INSTITUTO SUPERIOR DE ENGENHARIA DO PORTO

MESTRADO EM ENGENHARIA INFORMÁTICA

Aplicação Móvel para suporte a Conferências Científicas - UO ISEP

JOÃO CARLOS FERREIRA MACHADO

Outubro de 2023

POLITÉCNICO
DO PORTO

Mobile Application to support Scientific Conferences - ISEP

João Machado

A dissertation submitted in partial fulfillment of the requirements for the degree of Master of Science, Specialisation Area of Computer Systems

Supervisor: Dr. Piedade Carvalho

Porto, October 14, 2023

Dedicatory

Dedicated to my parents, for their unwavering support and encouragement throughout my academic journey. Their love and guidance have been the driving force behind my success, and I am forever grateful. This dissertation is also dedicated to the software industry, for providing endless opportunities for innovation and growth, and for being the field that continues to inspire and challenge me.

Abstract

This dissertation aims to develop a mobile application for Instituto Superior de Engenharia do Porto (ISEP), to support scientific conferences. With the increase in the use of mobile devices and the growing trend of providing mobile applications for scientific conferences, the need for a mobile application that can enhance the conference experience and provide a convenient and user-friendly way to access information about a conference is becoming more important. This application is designed to be used for live conferences and includes features such as an interactive event schedule, speaker bios, ticket management, and rating of events among others. The attendees can have a seamless conference experience and make the most of the time spent at the conference.

The application also includes features for conference organizers such as event registration, attendee management, and real-time analytics. Organizers are able to manage attendees, track attendance, and access analytics on attendee behavior and preferences, and use these features to evaluate the success of the conference, gather insights for future conferences, and improve the attendee's experience.

The development of this application is based on best practices and guidelines for the design and development of mobile applications and user-centered design methodologies. The design process is based on user research, interviews, and usability testing, to ensure that the application meets the needs and expectations of the users.

The impact of this application on the conference experience is evaluated through user testing and feedback. The evaluation is focused on measuring the usability, usefulness, and user satisfaction of the application. The results of this evaluation are used to improve the application and ensure that it provides a high-quality user experience.

Overall, this thesis aims to contribute to the field of mobile applications for conferences by providing a comprehensive solution that can enhance the attendee experience and improve the overall effectiveness of the conference.

Keywords: Mobile Application, Scientific, Conferences

Resumo

Esta dissertação tem como objectivo desenvolver uma aplicação móvel para o ISEP com o intuito de auxiliar nas conferências científicas que o mesmo Instituto organiza. Com o aumento do uso de dispositivos móveis e a crescente tendência de fornecer aos visitantes de conferências aplicações móveis, a necessidade de uma aplicação móvel que permita melhorar a experiência e fornecer uma maneira conveniente e amigável de aceder informações sobre uma conferência organizada pelo ISEP é imperativo. A aplicação desenvolvida no âmbito desta dissertação é projectada para ser usada em conferências ao vivo e incluirá recursos como uma programação interactiva de eventos, biografias dos oradores e materiais de apresentação, além de um directório de participantes e oportunidades de *networking*. Com esta aplicação, o objectivo é que os participantes tenham uma experiência bastante positiva e que consigam aproveitar ao máximo o tempo gasto na conferência.

A aplicação também inclui recursos para os organizadores da conferência, como registo de eventos, gestão de participantes e análises em tempo real. Os organizadores podem gerir os participantes, acompanhar a presença e ter acesso a análises sobre o seu comportamento e preferências, usando essa informação para avaliar o sucesso da conferência, coleccionar e analisar métricas relativas à conferência a fim de melhorar a experiência do participante.

O desenvolvimento desta aplicação é baseado nas melhores práticas e directrizes para o desenho e desenvolvimento de aplicações móveis e metodologias de desenho centrada na experiência do utilizador. O processo de desenho é baseado em entrevistas com os organizadores actuais de conferências no ISEP e testes de usabilidade, para garantir que a aplicação atenda às necessidades e expectativas dos utilizadores.

O impacto desta aplicação na experiência da conferência é avaliado por meio de testes e feedback dos visitantes e organizadores da conferência. A avaliação é focada em medir a usabilidade, utilidade e satisfação dos utilizadores com a aplicação. Os resultados desta avaliação são utilizados para melhorar a aplicação e garantir que esta proporcione uma experiência de utilizador de alta qualidade.

Em geral, esta dissertação quer contribuir para o campo de aplicações móveis para conferências científicas, fornecendo uma solução abrangente que possa melhorar a experiência do participante e melhorar a eficácia geral da conferência. Além disso, a dissertação pretende apresentar uma abordagem de desenho centrado no utilizador e que possa ser utilizada como referência para o desenvolvimento de futuras aplicações móveis em outros domínios.

Acknowledgement

First and foremost, I would like to thank my family for all the support they have given me throughout my life, allowing me to reach this point of being a finalist in my course. I would like to thank my friends, who I could always count on and who always encouraged me to be better, both in my personal and professional life. To my classmates, João Cunha, André Santos, and Rafael Ferreira who have always been with me in all projects since the beginning of the course.

I would like to thank Professor Piedade Carvalho, my advisor during the development of this dissertation, who always helped me to clarify certain doubts regarding this new experience.

Finally, I would like to thank the ISEP, the institution where I completed my undergraduate degree and pursued my Master's degree. The way of thinking and the work methodology had a great influence on me and the person I am today. I will take everything I've learned in the past five years with me for the rest of my life.

Contents

List of Figures

List of Tables

List of Source Code

List of Acronyms

AHP	Analytic Hierarchy Process.
AOT	ahead-of-time.
DTOs	Data Transfer Objects.
ERD	Entity Relationship Diagram.
FAST	Function Analysis and System Technique.
FCM	Firebase Cloud Messaging.
FCS	Firebase Cloud Storage.
GDPR	General Data Protection Regulation.
ISEP	Instituto Superior de Engenharia do Porto.
MVC	Model-View-Controller.
OS	Operating System.
QEF	Quality Evaluation Framework.
REST	Representational State Transfer.
UC	Use Case.
UI	User Interface.
UML	Unified Modeling Language.

Chapter 1

Introduction

This chapter contains a general description of the dissertation and the context in which it is inserted, exposing the problem and the objectives to be achieved with it. Here, the working approach that led to this dissertation is also presented. Finally, a brief presentation of the remaining chapters of the document is made, in order to facilitate understanding of its structure and the content of each of them.

1.1 Context

This document was produced as part of the curriculum unit Thesis/Dissertation/Internship (TMDEI) of the second academic year of the Master's degree in Software Engineering, a specialization area of the Master's degree in Computer Engineering at the Instituto Superior de Engenharia do Porto (ISEP).

The work described in this document was carried out in an academic context, specifically at ISEP.

A mobile application for scientific conferences is a software application that can be installed on a mobile device such as a smartphone or tablet, which provides attendees with a convenient and easy-to-use interface to access information and engage with the conference. This application can include features such as the conference schedule, speaker biographies, session descriptions, and interactive maps of the conference venue. Additionally, it can also provide attendees with the ability to view and download presentation materials, participate in live polls and Questions-and-Answers sessions, and connect with other attendees through social networking features. The application can also include features such as push notifications to remind attendees of upcoming sessions, and the ability to provide real-time feedback on the conference (Haider, 2023).

The use of mobile applications in scientific conferences is becoming increasingly popular as it allows for greater engagement and interactivity among attendees, as well as provides a more efficient and streamlined way for conference organizers to share and disseminate information. The application can also be used to facilitate the submission and review of conference papers, as well as the management of the conference schedule and agenda. Furthermore, it can also be used to allow attendees to easily access conference proceedings and other materials even after the conference has ended (Campbell, 2023).

Mobile applications for scientific conferences can also be used as a tool for data collection and analysis. Conference organizers can use the data collected through the application to gain insight into attendee behavior, preferences, and interests, which can then be used to improve the overall conference experience. This data can also be used to track the impact and reach of the conference and to measure the effectiveness of different marketing and outreach efforts (Stova, 2023).

In summary, a mobile application for scientific conferences is a useful tool that can greatly enhance the conference experience for attendees and provide valuable data for conference organizers. It can be used to facilitate information sharing, engagement, and interactivity, as well as for data collection and analysis. The use of mobile applications for scientific conferences is becoming a standard in the industry, and it's important to consider the inclusion of one in order to meet the expectations of the attendees and the conference organizers.

1.2 Organization

The ISEP is a higher education institution located in Porto, Portugal. It offers undergraduate and graduate programs in various fields of engineering, including Computer Engineering, Electrical Engineering, Mechanical Engineering, Industrial Engineering and Management, and Environmental Engineering. The mission of ISEP is to provide students with a high-quality education and to prepare them to become competent engineers who can contribute to the development of society.

ISEP is known for its strong emphasis on practical training and hands-on experience, and it offers students the opportunity to participate in internships and research projects. The institution also has a strong focus on innovation and entrepreneurship, and it provides students with the opportunity to participate in startup incubation programs and innovation competitions.

ISEP also has a robust research program, with several active research centers and laboratories. These centers focus on areas such as renewable energy, smart cities, and advanced manufacturing. The institution also has partnerships and collaborations with a number of companies and organizations, both in Portugal and internationally, which provide students with opportunities for internships, research projects, and career development.

Overall, ISEP is a well-respected institution that is known for its strong academic programs, commitment to practical training, and focus on innovation and entrepreneurship. Its partnerships and collaborations with companies and organizations provide students with opportunities to gain valuable experience and prepare for their future careers.

1.3 Problem

The problem addressed in this dissertation is the lack of an official mobile application for the scientific conferences organized by ISEP. Despite the growing popularity of mobile applications in the field of scientific conferences, ISEP has yet to develop and implement an official application for its conferences. This results in a number of challenges for both conference organizers and attendees.

For conference organizers, the lack of an official mobile application makes it difficult to effectively communicate and disseminate information to attendees. This includes information such as the conference schedule, speaker biographies, and session descriptions. Additionally, it also makes it difficult for organizers to collect and analyze data on attendee behavior and preferences, which could be used to improve the overall conference experience.

For attendees, the lack of an official mobile application can make it difficult to navigate and engage with the conference. This includes access to information such as the conference schedule, session descriptions, and speaker biographies. Attendees may also miss out on interactive features and networking opportunities that are commonly found in mobile applications for scientific conferences.

In summary, the lack of an official mobile application for scientific conferences organized by ISEP presents a significant problem for both conference organizers and attendees. It makes it difficult to effectively communicate and disseminate information, and it limits the opportunities for engagement and interaction. This dissertation aims to address this problem by proposing and developing a mobile application that can be used to enhance the conference experience for attendees and provide valuable data for conference organizers.

1.4 Objectives

The main objective of this dissertation is to design and develop a mobile application that can be used to support scientific conferences organized by ISEP. This includes the following specific objectives:

- To provide conference attendees with a convenient and easy-to-use interface to access information and engage with the conference. This includes features such as the conference schedule, speaker biographies, session descriptions, and interactive conference venue maps.
- To allow attendees to easily access conference proceedings and other materials even after the conference has ended.
- To provide conference organizers with valuable data on attendee behavior and preferences, which can be used to improve the overall conference experience.
- To provide conference organizers with a tool for data collection and analysis, which can be used to track the impact and reach of the conference, and to measure the effectiveness of different marketing and outreach efforts.
- To provide a tool that can be used to facilitate information sharing, engagement, and interactivity.

Overall, the development of a mobile application for scientific conferences organized by ISEP will provide conference organizers with valuable data and attendees with easy access to the conference information, making the conference experience more interactive, efficient, and convenient.

1.5 Approach

The approach used in this dissertation to design and develop the mobile application for scientific conferences is as follows:

- Requirements gathering: The first step in the development process will be to gather and analyze the requirements for the application. This includes conducting interviews and surveys with conference organizers and attendees to understand their needs and preferences.

- Design and prototyping: Once the requirements have been gathered, the next step is to design the user interface and create a prototype of the application. This will include creating wireframes and mockups to demonstrate the proposed functionality and layout of the application.
- Implementation: Once the design and prototyping phase is complete, the next step is to implement the application using a suitable mobile development platform such as React Native, Flutter, or Ionic.
- Testing and evaluation: The final step in the development process is to conduct thorough testing and evaluation of the application. This includes usability testing with conference attendees, and performance testing to ensure the application is stable and reliable.
- Deployment: Once the testing and evaluation phase is complete, the mobile application will be deployed to the app stores (Google Play and Apple Store) and also it will be provided to ISEP to be used in future scientific conferences.

The Agile approach used in this dissertation is grounded in a profound acknowledgment of the inherent uncertainty and evolving nature of software development projects. Unlike traditional methodologies that rigidly fix requirements at the beginning, Agile methodologies recognize that the landscape of project requirements can shift, new insights can emerge, and user needs may evolve. This uncertainty is not seen as a hindrance but as an opportunity for growth and improvement (Chugh and Chugh, 2023).

Agile embraces change as an integral part of the development journey. It encourages adaptability and responsiveness to changing circumstances. This means that as new insights are gained or as user feedback is collected, the development team can readily integrate these learnings into the project. Agile methodologies place a high value on the ability to adjust and pivot in response to emerging challenges and opportunities (Doshi and Virparia, 2023; Rocha et al., 2023).

By doing so, the Agile approach ensures that the final product is not just a static reflection of initial specifications but a living, evolving solution that stays attuned to the ever-changing needs of its users. It prioritizes delivering value early and consistently, which is particularly beneficial in dynamic domains like mobile application development for scientific conferences.

Agile methodologies are particularly well-suited for projects that require flexibility, responsiveness to user feedback, and the ability to accommodate changes in requirements as the project unfolds. In this dissertation, the Agile approach serves as a guiding principle that enables the development process to align closely with the goals and objectives of the research. It ensures that the mobile application created is not just a one-time deliverable but an ongoing, adaptable tool that evolves in response to the evolving needs and preferences of its users.

1.6 Document Structure

The current chapter (Introduction) includes an initial contextualization of the document's theme, a small presentation of ISEP, the organization for which this project was carried out, the problem to be solved, a brief description of the current solution, and the objectives that are expected to be achieved, as well as the approach used to achieve these objectives, are defined.

In the second chapter (State of the Art), the current state of the art is raised, so that various important topics are introduced and explored for understanding the problem at hand and its possible solution. A study of the current market in this area is also carried out, related works are presented with the system to be developed, and some technologies used in the project of this dissertation, such as Flutter and Dart, are explored.

In the third chapter (Value Analysis), a value analysis of this project is carried out, to understand how it introduces value for both the customer and the organization, through various techniques and artifacts, such as opportunity identification and functional analysis.

The fourth chapter (Analysis and Design), refers to the evaluation model used to assess the development of the application, explains the process of designing the mobile application and all its integrated external components as well and details the engineering requirements, in which the actors of the system and the functional and non-functional requirements are identified.

The fifth chapter (Implementation), Section with actual implementation snippets and comprehensive explanations about how the mobile application was implemented.

In the sixth chapter (Evaluation), how the solution will be evaluated after its implementation is defined. Initially, the objectives of this experimentation and evaluation are traced, followed by a definition of the hypothesis to be tested. Afterward, the indicators and sources of information are presented, and finally, the methodology used in the evaluation itself.

In the seventh chapter, you will find the conclusion of this dissertation, summarizing the key findings and insights from the project. Additionally, this chapter outlines possible avenues for future work and research, providing recommendations and suggestions for further development and improvement of the mobile application for scientific conferences.

The Annexes Chapter, placed after the Conclusion chapter, contains additional supplementary materials, including the Google Form used during the testing phase, which provided valuable insights and data from the user testing process.

Chapter 2

State of the Art

Scientific conferences have become increasingly important as a means of sharing knowledge and building networks among researchers, practitioners, and professionals in various fields. As the popularity of these events grows, so does the need for effective tools to support them. In recent years, mobile applications have emerged as a promising technology to enhance the conference experience for both attendees and organizers. However, the choice of which type of mobile application to develop can be a challenging one. In this chapter, we review the state of the art in mobile applications for scientific conferences, focusing on the different types of mobile frameworks available to developers. Specifically, we compare native, hybrid, and cross-platform mobile frameworks, and discuss their advantages and disadvantages. We then examine four popular cross-platform frameworks (React Native, Flutter, Ionic, and NativeScript), and describe their features, strengths, and weaknesses. Finally, we present a review of three existing mobile applications for scientific conferences, highlighting their design, features, and user experience.

2.1 Scientific Conferences

Scientific conferences are a vital part of academic and research communities, serving as a platform for researchers, practitioners, and students to present and discuss their work, exchange ideas, and collaborate on new research projects. The conferences provide a space for attendees to learn about the latest developments in their fields, make connections with peers and potential collaborators, and gain inspiration for future research projects (Aboras, 2016).

The format and structure of scientific conferences can vary depending on the field and the conference's goals. However, they usually have a clear scientific or academic focus, with presentations and discussions based on research papers or scientific posters. These papers undergo a rigorous peer-review process to ensure their quality and relevance to the conference's theme. Moreover, scientific conferences typically include keynote speeches and invited talks by leading experts in the field, as well as parallel sessions and workshops that allow participants to interact and share their research.

Scientific conferences can be categorized based on different criteria, such as their scope, size, frequency, or location. Some conferences may focus on a specific topic, such as computer science or biotechnology, while others may cover broader areas, such as interdisciplinary studies. The size of a conference can vary significantly, from small, local events with a few dozen attendees to large international conferences with thousands of participants. The frequency of conferences may range from annual to biennial or triennial, depending on the field's pace and the available resources. Finally, conferences can take place in different locations, such as academic institutions, conference centers, hotels, or online platforms (sá et al., 2019).

One of the significant challenges facing scientific conferences is effective communication and dissemination of information to attendees. A large number of attendees, parallel sessions, and complex schedules can make it challenging for conference organizers to convey information effectively. The lack of efficient communication can frustrate attendees, making it difficult for them to navigate the conference and locate sessions that interest them (Sarabipour et al., 2021).

In recent years, mobile applications have become increasingly popular for enhancing the conference experience for both organizers and attendees. These applications offer a range of features, such as the conference program, interactive maps, social networking tools, and real-time feedback and polling, among others. The use of mobile apps has become a standard practice in many fields, including technology, marketing, and education.

Despite the popularity of mobile apps in many fields, some scientific conferences have yet to adopt an official mobile application. For example, conferences organized by ISEP do not have an official mobile application, which makes it difficult for attendees to access information about the conference and engage with the other participants. Moreover, the lack of an official app also makes it difficult for organizers to collect and analyze data on attendee's behavior and preferences, which could be used to improve the overall conference experience.

In summary, scientific conferences are an essential part of academic and research communities, providing a platform for knowledge dissemination and scientific discussions. The conferences vary widely in format, structure, scope, and location, and they undergo a rigorous peer-review process. Additionally, the use of mobile applications to enhance the conference experience is becoming increasingly popular, but some scientific conferences have yet to adopt an official app, which presents challenges for both organizers and attendees.

2.2 Native vs Hybrid vs Cross Platform Mobile Frameworks

When developing a mobile application, there are three main approaches to consider: native, hybrid, and cross-platform. Each approach has its own set of advantages and disadvantages, and the best approach will depend on the specific needs of the project.

Native applications are developed using the native programming languages and frameworks of the specific platform (e.g. Swift for iOS, Java for Android). They are developed specifically for a particular platform and are optimized for performance, providing the best user experience, and access to all platform-specific features. However, native development can be time-consuming and costly, as separate development teams are required for each platform (Nawrocki et al., 2021; Singh and Shobha, 2021).

Hybrid applications are developed using web technologies such as HTML, CSS, and JavaScript and are wrapped in a native container, such as Apache Cordova or Ionic. They are designed to be cross-platform and can be run on multiple platforms with a single codebase. They are cost-effective and faster to develop, but may not perform as well as native applications and may not have access to all platform-specific features (Denko et al., 2021; Xanthopoulos and Xinogalos, 2013).

Cross-platform applications are developed using a single codebase and are then compiled to run on multiple platforms. These applications can be built using a variety of frameworks such as React Native, Xamarin, and Flutter. They are faster to develop than native applications and can provide a better user experience than hybrid applications. However, they may not provide access to all platform-specific features and may not perform as well as native applications (Meirelles et al., 2019; Pinto and Coutinho, 2018).

In conclusion, the choice of whether to develop a native, hybrid, or cross-platform application depends on the specific needs of the project. Native applications provide the best user experience and access to all platform-specific features but are time-consuming and costly to develop. Hybrid applications are cost-effective and faster to develop, but may not perform as well as native applications. Cross-platform applications are faster to develop than native applications and can provide a better user experience than hybrid applications, but may not provide access to all platform-specific features and may not perform as well as native applications.

2.3 Cross-Platform Mobile Frameworks

Cross-platform mobile frameworks are a popular choice for developers looking to build mobile applications that can run on multiple platforms with a single codebase. These frameworks use a single programming language and set of tools to create applications that can run on both iOS and Android devices, thus reducing the time and cost of development (Sommer and Krusche, 2013).

Since the cross-platform approach was the chosen option for the development of the solution, in this section the most common frameworks will be enumerated.

2.3.1 React Native

React Native is a cross-platform mobile framework developed and maintained by Facebook. It has approximately 85k stars on GitHub and is the second most contributed open-source project on the platform. Like Ionic, React Native uses current web technologies such as HTML, CSS, and JavaScript, making it easy for web developers to quickly pick up (Kaushik et al., 2019).

One of the major strengths of React Native is its vast documentation and community support. The framework is used in several well-known mobile applications such as Facebook, Instagram, Discord, Pinterest, Uber, Tesla, and many more. This showcases that React Native can be used to fulfill a wide range of use cases.

On the performance side, React Native compiles its components into their native counterparts and uses a separate JavaScript thread to handle business logic, as well as to dispatch commands to the native User Interface (UI) and the React Native Bridge, which allows access to device functionality. This architecture ensures that the UI thread remains untouched, and is only responsible for rendering the UI. Additionally, React Native utilizes a similar concept to the Virtual DOM, which allows it to dispatch UI changes in batches and only when necessary, resulting in extremely responsive performance (Shevtsiv et al., 2019; Yudin, 2020).

Although fast, it may not provide the same level of performance as frameworks that completely compile to native, such as Flutter, it ultimately depends on the device's hardware. React Native also has a large number of third-party modules developed by the community to access native functionalities.

2.3.2 Flutter

Flutter is a free and open-source mobile UI framework created by Google. It is used to develop applications for Android, iOS, Linux, Mac, Windows, and the web. Unlike other cross-platform frameworks, Flutter uses a unique approach to building mobile apps, by using a reactive programming model and a custom high-performance rendering engine (Payne, 2019b).

Flutter's widget system is designed to be highly customizable and expressive, allowing developers to create beautiful and responsive user interfaces. The framework also includes a rich set of built-in widgets, including Material Design and Cupertino (iOS-style) widgets, as well as support for custom widgets.

One of the main advantages of Flutter is its high performance, which is achieved by using the Dart programming language, which compiles Ahead-of-time (AOT) to native code. This allows for smooth animations and fast app startup times. Additionally, Flutter's Hot Reload feature allows developers to quickly test and iterate on their code, without the need for a full app rebuild (Napoli, 2019).

Another advantage of Flutter is its strong support for native features, such as camera, GPS, and push notifications. This is achieved through the use of plugins, which provide a bridge to the native APIs of each platform. Flutter also supports integration with other languages and frameworks, such as Java and Swift, making it easy to use existing code in a Flutter app (Payne, 2019a).

Despite its many benefits, Flutter has some limitations, such as its relatively small community size and a limited number of existing packages and plugins compared to other frameworks. However, this is changing as Flutter continues to grow in popularity. Overall, Flutter is considered a powerful and versatile framework for building high-performance, visually stunning mobile apps.

2.3.3 Ionic

Ionic is an open-source framework that enables developers to build high-performance, high-quality mobile and desktop applications using web technologies such as HTML, CSS, and JavaScript. The Ionic team maintains both Ionic and Capacitor, which have a combined total of around 45k stars on GitHub. Developers can use Ionic standalone, with its JavaScript/-TypeScript and UI component library, or in conjunction with popular front-end frameworks such as Angular, React, and Vue, to leverage their powerful capabilities (Dunka et al., 2017).

Ionic is a framework-agnostic component library, that allows developers to choose the front-end framework that best suits their needs. Access to native device features such as GPS, File System, and Camera is provided through Cordova (formerly known as PhoneGap) or through its new application runtime, Capacitor. Both act as a bridge to access native device functionalities.

Applications built with Ionic are known as hybrid applications or Progressive Web Apps, as they run inside a browser but are packaged as native applications for distribution in app stores such as the App Store for iOS or the Play Store for Android. Due to the fact that Ionic does not build truly native applications, performance may not always be optimal. The app runs inside a native API called WebView (for Android) or UIWebView (for iOS), which wraps progressive web apps on mobile devices. This introduces additional overhead that can impact app performance. Additionally, Cordova and Capacitor may not cover 100% native features, leading to some limitations. However, they do allow developers to create and inject custom plugins.

On the other hand, the fact that Ionic applications are progressive web apps means that most, if not all, source code works on iOS, Android, and the web without the need for significant modification. Both Capacitor and Cordova provide API fallback based on the platform they're running on. Ionic's code is totally reusable across platforms, allowing developers to write once and run everywhere. Overall, Ionic provides clean UI components, uses modern web technologies, produces hybrid apps that can run on any device, and can be integrated with front-end frameworks with significant community support and documentation. However, performance may be a limitation with Ionic.

2.3.4 NativeScript

NativeScript is an open-source framework that allows developers to build cross-platform mobile applications for Android and iOS devices using web technologies such as HTML, CSS, and JavaScript. It is similar to React Native in that it also only compiles its UI components into their native equivalents, and it is similar to Ionic in that it can be used with top-tier front-end frameworks such as Angular, React, Vue, and Svelte. However, what sets it apart from these frameworks is its unique architecture that heavily relies on a single main thread and allows for 100% access to native APIs out-of-the-box (Brito et al., 2018).

One of the key features of NativeScript is its ability to inject all platform-specific APIs into its JavaScript runtime (V8 for Android and iOS), which runs in the main UI thread. This means that developers can expect full access to native functionalities without the need for additional plugins or third-party modules. However, this architecture also means that performance might not be as good as other frameworks that use separate threads for business logic and rendering, such as React Native or Flutter (Ahmed et al., 2020).

Another thing to consider when using NativeScript is its application showcase and community size. It is not as popular or well-known as React Native or Flutter, with only 18k stars on GitHub and 160 contributors at the time of writing. This means that there might be fewer resources and support available for developers compared to other frameworks. Despite this, it is still considered a viable alternative for those who are looking for a framework that allows for easy access to native APIs and can be used with multiple front-end frameworks (Majchrzak et al., 2015).

These frameworks offer a wide range of features and functionalities to help developers build high-performance, cross-platform mobile applications. They also provide developers with access to native functionality and APIs, as well as a wide range of third-party libraries and plugins.

However, it's worth noting that even though these frameworks are cross-platform, it's not always guaranteed that the final product will look and feel the same on both iOS and Android. Some additional work may be required to achieve a similar look and feel on both platforms.

In conclusion, cross-platform mobile frameworks are a popular choice for developers looking to build mobile applications that can run on multiple platforms with a single codebase. They provide a wide range of features and functionalities, as well as access to native functionality

2.4 Mobile Applications for Conferences

In recent years, there has been an increasing trend toward the use of mobile devices at scientific conferences. Many conference organizers are now providing mobile applications to support their events, in an effort to enhance the conference experience for attendees and provide a convenient and user-friendly way to access information about the conference.

In the field of mobile applications for conferences, several studies have been conducted to investigate the impact of these applications on the conference experience. These studies have shown that mobile applications can provide a convenient and user-friendly way to access information about the conference and its events, facilitate communication and collaboration among attendees, and provide conference organizers with valuable data on attendees' behavior and preferences (Ozturk et al., 2021; Talantis et al., 2020).

In terms of the design and development of mobile applications for conferences, there are several best practices and guidelines that have been established. These include the use of user-centered design methods, the adoption of agile development methodologies, and the use of the latest technologies and frameworks (Kounavis et al., 2011).

In terms of features and functionality, several studies have been conducted to investigate the most commonly used features. These include the provision of a centralized and easily accessible source of information about the conference and its events, the inclusion of a directory of attendees, networking opportunities, and the ability to provide feedback on presentations (Hussain et al., 2019; Zhan and Chiu, 2014).

2.4.1 Justification for Selection

The selection of Whova, Aventri, and Sched as the focus of this study was based on several key factors:

1. **High Ratings on App Stores:** Whova, Aventri, and Sched are among the most highly rated conference apps on major app stores, including both the Apple App Store and Google Play Store. Their positive reviews and high ratings indicate a strong user satisfaction level and a track record of delivering value to conference attendees and organizers.

2. **Prominence in Google Search Results:** These applications are prominent in Google search results when searching for conference management and attendee engagement solutions. Their visibility in search results indicates their recognition and widespread use in the industry.

3. **Comprehensive Feature Set:** Whova, Aventri, and Sched offer comprehensive feature sets that align with the functionalities envisioned for the ISEP Conference Application. These features include attendee engagement tools, registration management, speaker management, marketing, and analytics.

4. **Flexibility and Customization:** These platforms provide a level of flexibility and customization that can cater to various types and sizes of events. This flexibility is essential to adapt the applications to the specific needs of the ISEP conferences.

5. **User-Friendly Interfaces:** The user-friendly interfaces of these applications make them accessible to both event organizers and attendees. An intuitive interface is crucial for ensuring a smooth and hassle-free conference experience.

6. **Industry Recognition:** Whova, Aventri, and Sched have received industry recognition and awards for their contributions to event management and attendee engagement. These accolades further validate their suitability for evaluation and consideration.

Given these considerations, Whova, Aventri, and Sched emerged as strong candidates for evaluation as potential solutions for the ISEP Conference App.

2.4.2 Whova

Whova is an all-in-one event management platform designed to help organizers plan and execute successful events. The platform offers a wide range of features that allow organizers to create, promote, and manage their events with ease. From attendee engagement and registration to speaker management and event analytics, Whova provides comprehensive solutions for events of all sizes and types (Whova, 2023).

Whova was founded in 2013 by Dr. Zhao and Dr. Jia, who recognized the need for a more efficient and cost-effective solution for event management. Since its inception, Whova has become one of the leading event management platforms in the industry, serving over 10,000 events and millions of attendees worldwide.

One of the core features of Whova is its attendee engagement tools. The platform offers a mobile app that allows attendees to access event information easily, connect with other attendees, and engage with speakers and exhibitors. The app offers a personalized experience, enabling attendees to customize their event schedules and receive personalized recommendations based on their interests. Attendees can also access interactive maps, take notes, and participate in live polls and surveys.

In addition to attendee engagement, Whova offers robust registration tools that allow organizers to easily create and manage event registration pages. The platform offers customizable registration forms, ticketing options, and payment processing, making it easy for organizers to sell tickets and manage registration information.

Another key feature of Whova is its speaker management tools. The platform allows organizers to easily manage speaker information, including bios, headshots, and presentation materials. Organizers can also track speaker session schedules, send reminders and notifications, and collect feedback from attendees.

Whova also offers a variety of marketing and promotion tools to help organizers increase event visibility and attract attendees. The platform allows organizers to create event websites, send email campaigns, and promote events on social media. The platform also offers analytics tools that provide insights into attendee engagement, registration data, and other key event metrics.

Whova's platform is highly customizable and can be tailored to meet the specific needs of each event. The platform offers a variety of integrations with other event management tools, such as ticketing platforms, event apps, and CRM systems.

One of the key benefits of Whova is its user-friendly interface, which allows organizers to easily navigate the platform and access the tools they need. The platform also offers 24/7 customer support, ensuring that organizers have access to assistance whenever they need it.

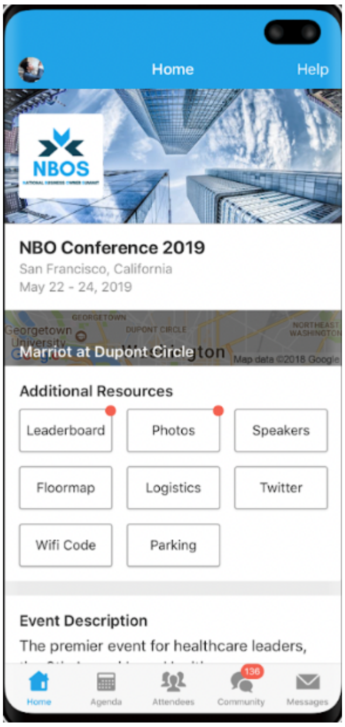


Figure 2.1: Home view for Whova app (Whova, 2023).

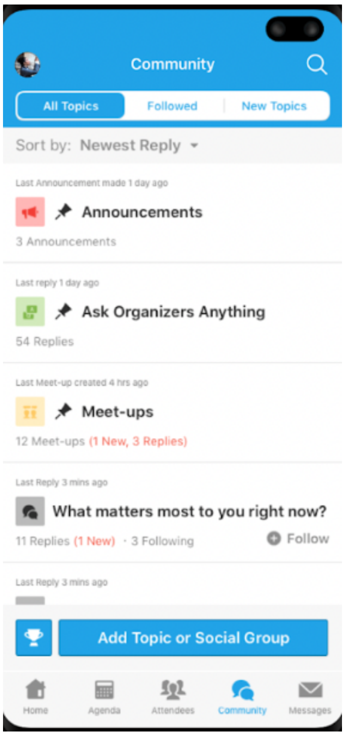


Figure 2.2: Community view for Whova app (Whova, 2023).

In conclusion, Whova is a comprehensive event management platform that provides a wide range of features and tools to help organizers plan and execute successful events. Its attendee engagement tools, registration management, speaker management, and marketing and promotion tools make it an all-in-one solution for events of all types and sizes. Its highly customizable platform, user-friendly interface, and 24/7 customer support make it an excellent choice for event organizers looking to streamline their event planning and management processes.

2.4.3 Aventri

Aventri is an event management software that is designed to streamline and automate the planning and execution of events. The platform provides a comprehensive suite of tools for managing every aspect of an event, from pre-event planning and marketing to on-site registration and attendee engagement. Aventri aims to simplify the event management process and enhance the attendee experience, while also providing organizers with data-driven insights to optimize their future events (Aventri, 2023).

Aventri's features are organized into several key areas, including event website and registration, marketing and communication, onsite check-in and badging, mobile event app, and analytics and reporting.

One of Aventri's core features is its event website and registration capabilities. The platform allows organizers to quickly and easily create a customized event website that can be used to promote the event and manage attendee registrations. Aventri provides a range of customization options, including branding, ticketing, and registration form fields, to ensure the event website aligns with the organizers' branding and messaging.

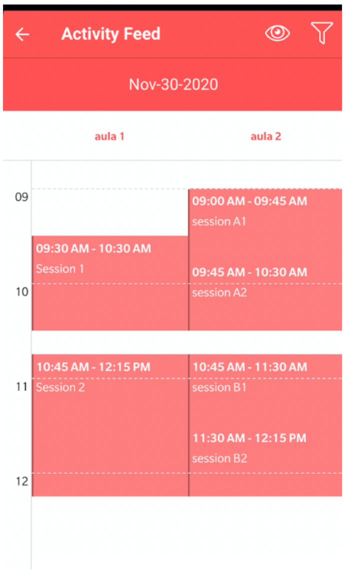
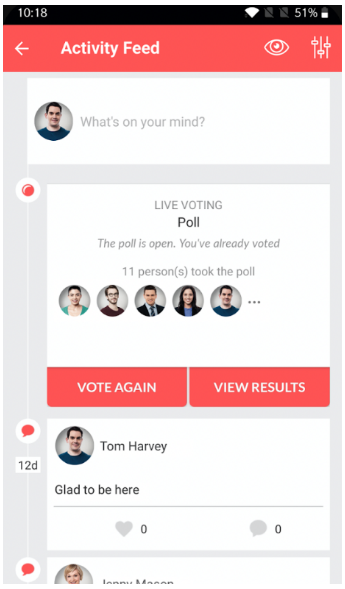


Figure 2.3: Schedule view for Aventri app (Aventri, 2023).



Tom Harvey

Glad to be here

Figure 2.4: Feed view for Aventri app (Aventri, 2023).

In addition to website and registration tools, Aventri provides robust marketing and communication features to promote the event and engage attendees. These include email marketing tools, social media integration, and personalized messaging that can be sent to attendees before, during, and after the event. Aventri also provides an event app that can be used to engage attendees, provide personalized schedules, and send notifications.

Aventri's onsite check-in and badging capabilities provide a streamlined and secure process for managing attendee check-ins. Organizers can use Aventri to print badges and manage access control for attendees, exhibitors, and other event stakeholders. Aventri's mobile app can also be used to scan badges and manage attendance in real-time.

Aventri's analytics and reporting tools provide valuable insights into attendee behavior and event performance. Organizers can use Aventri's dashboard to track registration and attendance metrics, monitor attendee engagement, and analyze event feedback. These insights can be used to optimize future events and improve the attendee experience.

Overall, Aventri is a comprehensive event management software that provides organizers with the tools they need to plan, execute, and optimize successful events. Its intuitive interface, robust feature set, and data-driven insights make it a valuable tool for event planners, marketers, and organizers.

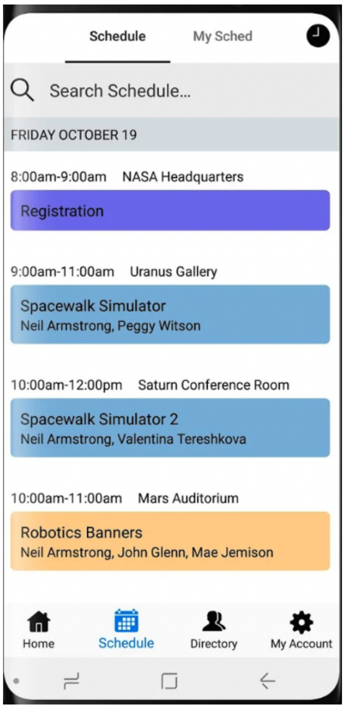
2.4.4 Sched

Sched is a web and mobile-based platform designed to streamline event management and enhance the overall attendee experience. This software provides a robust set of tools that enable event organizers to efficiently manage and promote their events, while attendees can use it to view and personalize their schedules, network with other attendees, and interact with event content (Sched, 2023).

The platform's user-friendly interface makes it easy for event organizers to create and manage a variety of activities, such as speaker sessions, workshops, and networking events. Organizers can create and publish event schedules, manage registration and ticketing, and monitor event attendance and participation metrics. Additionally, they can send customized notifications and reminders to attendees, ensuring they remain engaged and up-to-date on any event changes or updates.

One of the key features of Sched is its ability to personalize and optimize the attendee experience. Once registered, attendees can use the platform to view the event's schedule, create a customized itinerary, and receive notifications and reminders for sessions they plan to attend. Attendees can also use the platform to interact with event content and speakers, ask questions, and provide feedback.

Sched's mobile app further enhances the attendee experience by providing access to event information and schedules on the go. The app enables attendees to view their personalized schedules, find event locations, and access real-time event updates. Attendees can also use the app to network with other attendees, send private messages, and view speaker profiles.



FRIDAY OCTOBER 19

8:00am-9:00am NASA Headquarters
Registration

9:00am-11:00am Uranus Gallery
Spacewalk Simulator
Neil Armstrong, Peggy Witson

10:00am-12:00pm Saturn Conference Room
Spacewalk Simulator 2
Neil Armstrong, Valentina Tereshkova

10:00am-11:00am Mars Auditorium
Robotics Banners
Neil Armstrong, John Glenn, Mae Jemison

Figure 2.5: Schedule view for Shed app (Sched, 2023)

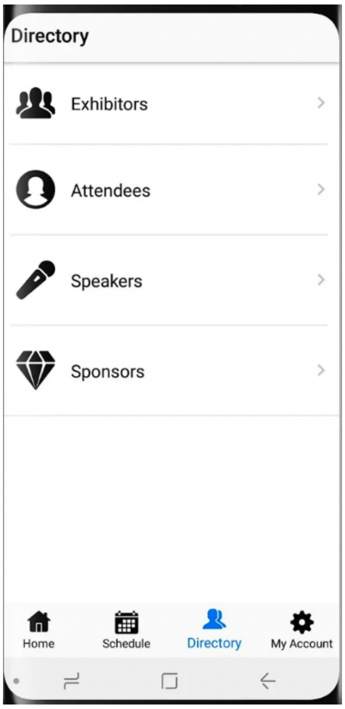


Figure 2.6: Directory view for Shed app (Sched, 2023)

In addition to its event management and attendee engagement tools, Sched provides advanced analytics and reporting capabilities. Organizers can leverage these features to track and measure event performance, identify areas for improvement, and generate post-event reports.

Overall, Sched offers a comprehensive set of event management tools that streamline the planning process, enhance attendee engagement, and provide insights into event performance. Its user-friendly interface and personalized experience features make it a valuable tool for organizers and attendees alike.

Whova offers advanced networking features and audience engagement tools, making it a great choice for conferences and trade shows. It also has a user-friendly interface and easy navigation. However, its pricing can be relatively expensive for small and medium-sized events, and some users may experience issues with customization and branding.

Aventri provides robust event management tools, including registration and ticketing, marketing and promotion, and on-site check-in and analytics. Its reporting and analytics features are particularly noteworthy. However, the platform's pricing is relatively high, and the learning curve can be steep for new users. Customer support can also be slow to respond to queries.

Sched offers a flexible and customizable event management platform with a wide range of features, including speaker and session management, sponsor promotion, and attendee networking. The platform is relatively affordable and can be a good option for smaller events. However, some users may find the interface cluttered and challenging to navigate, and the platform's mobile app can have performance issues.

Ultimately, the best option will depend on the specific needs of the event and its organizers. Careful consideration of each platform's strengths and weaknesses can help event planners choose the best option for their unique requirements.

2.4.5 Comparison of Whova, Aventri, and Sched

To assist in the evaluation process, a comparison table highlighting key features and considerations of Whova, Aventri, and Sched is presented below:

Table 2.1: Comparison of Whova, Aventri, and Sched.

Feature	**Whova**	**Aventri**	**Sched**
Attendee Engagement	Advanced networking features, personalized schedules, interactive maps, live polls	Event app for engagement, personalized schedules, notifications	Personalized schedules, attendee networking, interactive event content
Registration Management	Customizable forms, ticketing options, payment processing	Customized event websites, ticketing, registration form customization	Registration and ticketing features
Speaker Management	Speaker bios, presentation materials, session schedules	Speaker information management	Speaker sessions, engagement
Marketing and Promotion	Event websites, email campaigns, social media integration	Email marketing, social media integration	Promotion of event content
On-Site Check-In	Badge printing, access control	Streamlined attendee check-in	Real-time attendance management
Analytics and Reporting	Insights into attendee engagement, registration data, event metrics	Event performance analytics	Metrics on registration and attendance
Pricing	Pricing can be relatively expensive for small and medium-sized events	Pricing is relatively high	Affordable for smaller events
Customization	Highly customizable	Highly customizable, steep learning curve	Customizable, but interface can be cluttered
User-Friendliness	User-friendly interface	User-friendly, but new users may face challenges	Interface can be challenging to navigate
Customer Support	24/7 customer support	Customer support may be slow to respond	Support available

The table 2.1 provides an overview of the features and considerations of each platform, allowing event organizers to make informed decisions based on their specific event requirements. Ultimately, the choice among Whova, Aventri, and Sched will depend on factors such as event size, budget, customization needs, and user experience expectations.

Table 2.2: Feature envisioning for the proposed solution

Feature	Conference Application for ISEP
Attendee Engagement	Advanced networking features, personalized schedules, maps, and notifications
Registration Management	Attendees can associate a ticket to the application account
Speaker Management	Speaker bios, presentation materials, session schedules
Marketing and Promotion	No marketing and promotion integration planned
On-site check-in	Attendee check-in support, QR code generation for attendees
Analytics and Reporting	Metrics on registration and attendance, application usage, and real-time display of metrics using custom dashboards
Pricing	No pricing model for users. Costs regarding the hosting Firebase database and publishing to the Apple App Store
Customization	Multi conference support, multi-language support, and dark or light theme
User-Friendliness	User-friendly interface
Customer Support	No customer support

In table 2.2, an outline of the features of the proposed Conference Application for ISEP is presented. This table provides a glimpse into the key features and considerations that the custom conference application aims to offer. Attendee Engagement includes advanced networking features, personalized schedules, maps, and notifications. Registration Management allows attendees to associate a ticket with their application account. Speaker Management encompasses speaker bios, presentation materials, and session schedules. The application does not have Marketing and Promotion integration planned. For On-site check-in, the application will support attendee check-in and generate QR codes for attendees. Analytics and Reporting will include metrics on registration and attendance, application usage, and real-time display of metrics using custom dashboards. Pricing is not directly applied to users, but there may be costs associated with hosting Firebase databases and publishing to the Apple App Store. Customization options include support for multiple conferences, multiple languages, and dark or light themes. User-friendliness is emphasized with a user-friendly interface. However, there is no dedicated customer support planned for this application.

This table outlines the features specific to the proposed solution for the ISEP conference, making it easier for event organizers to understand the functionalities and capabilities of the custom conference application in comparison to the other event management platforms. It's important for event organizers to consider their unique requirements and priorities when selecting the right platform or solution for their event

2.5 Ethical and Responsibilities

Developing an application for managing scientific conferences entails a range of ethical and professional responsibilities for engineers and professionals involved in the process. These responsibilities are critical to ensuring the application's integrity, security, and ethical conduct.

2.5.1 Ethical Responsibilities

- **Transparency and Accuracy of Information**: The application must provide accurate and up-to-date information about scientific conferences. It is imperative to avoid any form of data or information manipulation that could mislead or misinform users.
- **Privacy and Data Protection**: Handling user data, including registration information for conferences, with the highest level of security and privacy is essential. All data protection regulations, such as General Data Protection Regulation (GDPR), must be strictly adhered to, respecting users' rights and ensuring their data remains confidential.
- **Neutrality and Impartiality**: The application should remain neutral and impartial in its treatment of scientific conferences. There should be no unjustified favoritism towards one conference or event over others, ensuring fairness and equity.
- **Accessibility and Inclusion**: The application should be designed to be accessible to all users, regardless of their physical or cognitive abilities. Inclusion features such as captions, screen reader support, and legible fonts are essential to make the application usable by a diverse audience.

2.5.2 Professional Responsibilities

- **Alignment with Best Software Development Practices**: Engineers should adhere to best practices in building the application, focusing on security, efficiency, and scalability. High-quality code and robust architecture are essential to provide a reliable user experience.
- **Continuous Updates and Application Maintenance**: Keeping the application up to date is crucial. This includes addressing bugs, and vulnerabilities, and implementing improvements as necessary to ensure the application's smooth and effective operation.
- **Effective Communication with Stakeholders**: Developers should maintain clear and transparent communication with stakeholders, including conference organizers and attendees. Timely responses to questions and regular updates on the project's progress foster trust and ensure that all parties are informed.
- **Ensuring Data Integrity**: Scientific conference data, as well as participant information, should be maintained with integrity and security. Regular backups, data encryption, and robust security measures are essential to protect user and event data.
- **Compliance with Industry Regulations and Standards**: The application must adhere to all relevant industry regulations and standards in the technology and scientific conference sectors. This includes following ethical guidelines, industry standards, and any applicable legal requirements to ensure responsible and professional conduct.

By embracing these ethical and professional responsibilities, engineers and professionals can ensure that the application for scientific confererence management maintains the highest standards of integrity, privacy, and professionalism in its operation. Such practices not only protect users and their data but also contribute to a positive and trustworthy user experience.

In this chapter, we have explored the state of the art in mobile applications for scientific conferences, focusing on the different types of mobile frameworks available to developers. We began by describing the concept and features of scientific conferences and highlighting the need for effective tools to support them. We then compared and contrasted native, hybrid, and cross-platform mobile frameworks, discussing their advantages and disadvantages. We also presented an in-depth review of four popular cross-platform frameworks (React Native, Flutter, Ionic, and NativeScript), examining their features, strengths, and weaknesses. Finally, we reviewed three existing mobile applications for scientific conferences, providing insights into their design, features, and user experience.

Overall, this review indicates that cross-platform frameworks offer a flexible and cost-effective way to develop mobile applications for scientific conferences. However, the choice of which framework to use depends on various factors, such as the development team's expertise, the desired features, and the target audience. This review also highlights the importance of user-centered design in the development of mobile applications for conferences, and the need to provide users with a seamless and intuitive experience.

Chapter 3

Value Analysis

Value Analysis is a systematic method for evaluating the worth of a product or service based on its functions and benefits. In the context of the mobile application for conferences, Value Analysis will be used to evaluate the benefits of the application to both the attendees and the event organizers. The goal of this chapter is to identify the key functions of the application and their associated benefits and to determine the overall value of the application based on its ability to enhance the conference experience and improve the overall effectiveness of the event.

3.1 Value Analysis Objectives

The purpose of Value Analysis is to examine the current process for achieving goals and determine how the solution being analyzed can improve or enhance the process. This analysis helps to clearly show the benefits that the solution brings to the current model, as well as any drawbacks. The ultimate goal is to determine if the advantages of the solution outweigh any negative factors, allowing for a conclusion to be made about whether the proposed solution is a valuable one.

The following is an analysis of how attendees and event organizers of conferences hosted at ISEP can benefit from using a mobile application similar to the object of this dissertation. To do that it's recommended to answer the following questions:

- What is your project?
- Who is your target customer? Or who do you provide the value to?
- What value does the solution provide?
- Why is the product unique?

With the world bouncing back from a two-year-long pandemic, where the majority of events were either canceled, postponed, or performed online, more and more conferences are being organized and performed physically at venues and institutes across the world revitalizing the market for mobile applications. These types of applications can either attain value by creating services and selling them to the customers or by connecting the service provider with their respective customers.

The application object of this dissertation aims to satisfy the latter with concerns regarding two main entities, the conference organizers and the conference attendees. The value generated is directly tied to how the attendees react and are able to easily access the information they might need and want while using the application. Reachleable and easy-to-access information translates to fewer logistics and more value. Additionally, value can also be generated from the ability to easily manage and access important metrics of an ongoing conference by the conference organizers. The application aims to provide value in the form of:

- Consistent and responsive application.
- Effortless process for attendees to find their important information and interact with the ongoing event.
- Conference management system to conveniently manage a conference.
- Gathering, processing, and displaying important metrics to the organizers.

When evaluating an application, it's important to consider the features offered and how they align with the target market. For example, if the target audience is self-employed drivers, the GPS tracking software should function efficiently. Conversely, if the platform caters to conference support, a robust and simple user interface with easy access to the most relevant information and the ability to interact with an ongoing conference would be more beneficial to the organizers and attendees. The value of the application can be determined by its ability to streamline processes for all parties involved, as well as the additional value it provides over existing models. One can gauge the perceived value for the organizers and attendees by examining the extent to which they rely on the application.

3.1.1 The Project

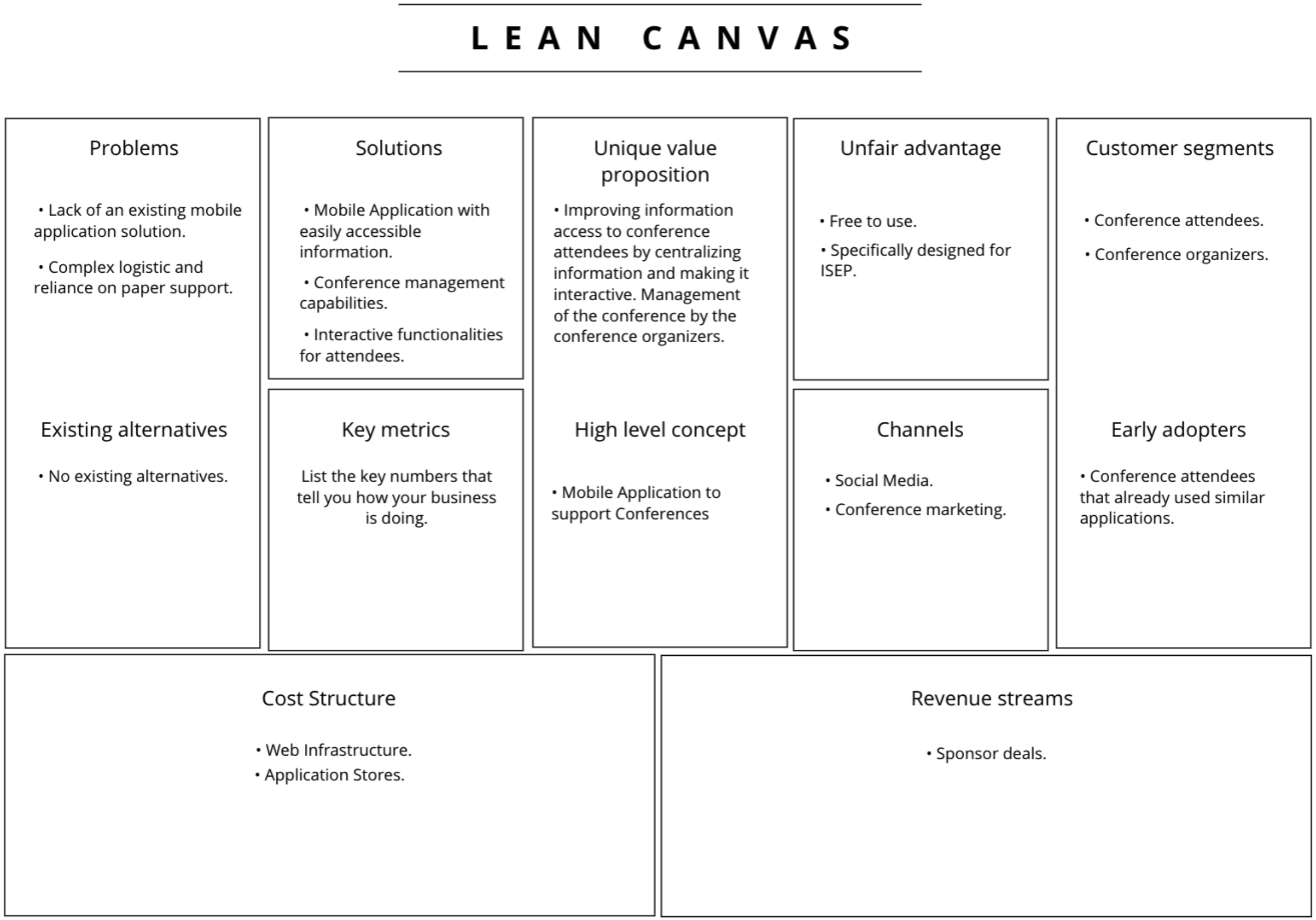
The mobile application developed in this dissertation is aimed at supporting scientific conferences hosted by ISEP. The purpose of the application is to enhance the conference experience for attendees, providing them with a convenient and user-friendly way to access important information about the event. The application is designed to be used at live events, offering features such as an interactive event schedule, speaker bios, and presentation materials.

The application not only focuses on the attendee experience but also includes features for event organizers. Organizers will have access to event registration, attendee management, and real-time analytics. These features will allow organizers to manage attendees, track attendance, and gather insights about attendee behavior and preferences. The information gathered through these features can then be used to evaluate the success of the event and make improvements for future events. The goal of this comprehensive solution is to improve the overall effectiveness of the event and provide a high-quality user experience for all participants.

3.1.2 Target Costumer

The target customer is the conference attendees with the aim to enhance the conference experience and allow attendees to make the most of their time at the conference, and conference organizers to manage the conference and use the information provided by the application to evaluate the success of the conference, gather insights for future events, and improve the attendee's experience.

3.1.3 Lean Canvas



LEAN CANVAS

Problems

- Lack of an existing mobile application solution.
- Complex logistic and reliance on paper support.

Existing alternatives

- No existing alternatives.

Solutions

- Mobile Application with easily accessible information.
- Conference management capabilities.
- Interactive functionalities for attendees.

Key metrics

List the key numbers that tell you how your business is doing.

Unique value proposition

- Improving information access to conference attendees by centralizing information and making it interactive. Management of the conference by the conference organizers.

High level concept

- Mobile Application to support Conferences

Unfair advantage

- Free to use.
- Specifically designed for ISEP.

Channels

- Social Media.
- Conference marketing.

Customer segments

- Conference attendees.
- Conference organizers.

Early adopters

- Conference attendees that already used similar applications.

Cost Structure

- Web Infrastructure.
- Application Stores.

Revenue streams

- Sponsor deals.

Figure 3.1: Lean Canvas.

3.2 Value as a Network

Value can be viewed as a network, a node-connected representation of all stakeholders and their specific roles that define a value chain. The network can be either internal, where value is traded within a company among employees, or external, where it is traded between a company, its suppliers, and customers. Value can be either tangible, such as money or physical documents, or intangible, such as knowledge, influence, or favors. In either case, value can only be perceived when it is converted into something that can be delivered. From a stakeholder perspective, this occurs when the input of one transaction, provided by another stakeholder, is transformed into actual benefits (Verna, 2008).

With this mobile application, value is external, meaning it is not traded internally within the organization but rather between all participants of it: the conference organizers, the conference attendees (both users of the application), and the application itself. The value between the organizers and registered attendees is classified as intangible in nature, as they do not trade services for money. This intangible value derives from offering the tools to the conference organizers providing an improved way to better manage their conferences and have access to important metrics in real-time about the conference and attendees. To the attendees, it provides a platform for them to find most if not all the information they might need regarding the event, as well as allowing for a way to interact with the content they enjoy. The following image illustrates how value is traded among the stakeholders.

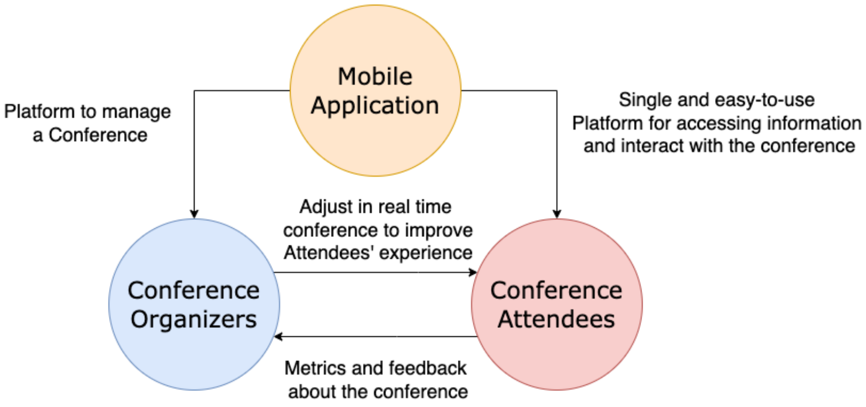

Mobile Application
Platform to manage a Conference
Single and easy-to-use Platform for accessing information and interact with the conference
Adjust in real time conference to improve Attendees' experience
Conference Organizers
Conference Attendees
Metrics and feedback about the conference

Figure 3.2: Value Network.

3.3 Development Value Analysis

Before beginning the development of the mobile application, it's crucial to evaluate the existing solutions and determine the one that best fits the needs. Performance is a crucial aspect to consider, as delivering a smooth and efficient experience to users is imperative. It's not enough to have a lot of features if navigating through the app is not consistent and fast. According to Android development guidelines, users can perceive slowness in an application if the response time exceeds 100ms to 200ms (Google, 2023). Performance and responsiveness are essential elements in creating a great mobile experience, as they are based on factors such as startup time, page loading speed, page transitions, animations, error handling, and waiting times. It's essential to provide a positive first impression, as a negative initial experience can deter users from using the platform.

Table 3.1: Perceived Value

Attributes	**Outcomes**	**Sacrifices**
Performance	User Experience	Development Time
Learning Curve	Experience, gradually decreasing development time	Slow initial start
Documentation and Supportive Community	Common issues easily addressed	Convuluted and outdated information
Integration and Features	Ability to easily access and use mobile-specific features	Performance
Cross-Platform Architecture	Faster time to market in both Android and iOS	Performance and inability to use specific Operating System (OS) features

When evaluating the development of the application, the **learning curve** and development time are important factors to consider. The framework should be straightforward, following current development practices, using a well-known programming language or one that is easy to learn, and simple enough for basic tasks such as page routing, layout, connecting to external services, and accessing native device functions.

The availability of **documentation and a supportive community** is also crucial. The framework must have detailed documentation with implementation examples and an active community to ensure its development is maintained and updated.

The **integration and features** offered by the framework are also critical factors. If it already has great integration with major mobile features, it has a head start.

The choice between **native and cross-platform** development is also important. Native development offers 100% access to mobile functions and improved performance, while cross-platform development means the same code can be used for both Android and iOS, eliminating the need for separate code bases. Currently, the decision is to move forward with a cross-platform solution, as all considered frameworks provide almost complete access to device features. However, some frameworks require less code to work properly on the desired platforms while others may require more. Some frameworks might already provide a wide range of styled-components, while others may require different internal implementations to achieve the desired experience on both platforms.

3.3.1 Perceived Value, Value Attributes and for the Developer

Building upon the concept of value for the customer proposed by T. Woodall in "Conceptualizing Value for the Customer: An Attributional, Structural, and Dispositional Analysis" (Woodall, 2003), a new table was created to succinctly outline the idea of value for the developer. This value is realized when the outcomes produced by the attributes surpass the sacrifices made. On the 3.1 table the attributes, outcomes, and sacrifices are enumerated.

- Performance is a key factor that contributes to a better user experience. The developer must ensure that the framework can handle all existing scenarios flawlessly. To achieve this, the developer must adopt good coding practices and optimize the application from the start. Although these practices may take more time to implement, especially at the start of the project, they will ultimately lead to a better end product.

- Learning Curve and Basics are also critical in adding value to the development process. The framework should be easy to start and implement basic functionality with smooth management. A framework that has a steep learning curve and is difficult to implement basic functionality will not be favored by developers.
- Documentation and Community play an important role in providing better development experiences. The framework should have comprehensive documentation that covers how-to-build examples and provides answers to common questions. If documentation is lacking, the developer can reach out to the community for help. However, it's recommended to read the documentation before starting development, as this may save time in the long run.
- Integration and Features can reduce development time and add value to the framework. A framework that easily integrates existing features such as payments, push notifications, GPS, and more, can be highly valuable. However, integrating these features may result in unwanted overhead for simple use cases.
- Cross-Platform capability is also a factor that can add value to the framework. With a single code base to maintain, the cross-platform framework can save the developer time and effort. The downside, however, is that cross-platform frameworks may not be as performant as their native alternatives.

The following frameworks have been taken into consideration for the development of the Mobile Application:

- Flutter;
- React Native;
- Ionic combined with Angular;
- Angular in conjunction with NativeScript.

3.3.2 The Analytic Hierarchy Process

The Analytic Hierarchy Process (AHP) is a systematic method for making complex, multi-criteria decisions that involve breaking down the decision problem into smaller, more manageable components. This process allows decision-makers to systematically evaluate a range of alternatives in terms of their relative effectiveness in satisfying a set of decision criteria and objectives (Saaty, 2008).

To apply the AHP method, one must first define the problem and determine the knowledge sought. This requires a clear understanding of the decision-making goal and the specific objectives that must be met in order to achieve that goal.

Next, the decision hierarchy must be structured from top to bottom, starting with the goal of the decision at the top and working down through intermediate levels of criteria and sub-criteria, to the bottom level of alternatives. The criteria and sub-criteria are used to define the aspects of the decision problem that must be considered when evaluating the alternatives.

Once the decision hierarchy has been established, the next step is to construct a set of pairwise comparison matrices. In these matrices, each element in an upper level is used to compare the elements in the level immediately below with respect to it. This allows the decision-maker to weigh the relative importance of each criterion and sub-criterion in terms of its impact on the decision.

The final step is to use the priorities obtained from the pairwise comparisons to weigh the priorities in the level immediately below. This process of weighing and adding the weighted values is repeated for each element, working downwards through the decision hierarchy, until the final priorities of the alternatives in the bottom-most level are obtained.

The AHP method provides a systematic, transparent, and robust approach to decision-making that allows decision-makers to evaluate a range of alternatives in a comprehensive and meaningful way. This method has proven to be effective in a wide range of decision-making contexts, and it is particularly well-suited to complex, multi-criteria problems where traditional approaches may fail to provide adequate guidance.

In the case of the development of a mobile application for supporting scientific conferences, the AHP method will be used to evaluate and compare a set of alternatives for developing the platform. The alternatives will be introduced and explained, and the criteria on which they will be evaluated will be chosen. The final output of the AHP process will be a set of priorities that will indicate which alternative is the most suitable for developing the application. This information will be used to guide the development process and ensure that the platform is developed in a way that meets the needs and expectations of its users (figure 3.3).

One crucial step in AHP is the definition of priority or the allocation of weight to each criterion. This step is referred to as Priority Definition, and it plays a vital role in determining the importance of each criterion in relation to the other. The assignment of weight or priority is accomplished through the use of the Fundamental Scale of Pairwise Comparison, which was introduced by Thomas Saaty in 2008 (Saaty, 2008). This scale provides a systematic approach to determining the priority of each criterion and how it relates to the other criteria. The end result of this step is a clear understanding of the relative importance of each criterion in relation to the overall goal, which is established in the first step of the AHP process. In figure 3.3, in blue the goal is identified and with the color green the criteria are enumerated.

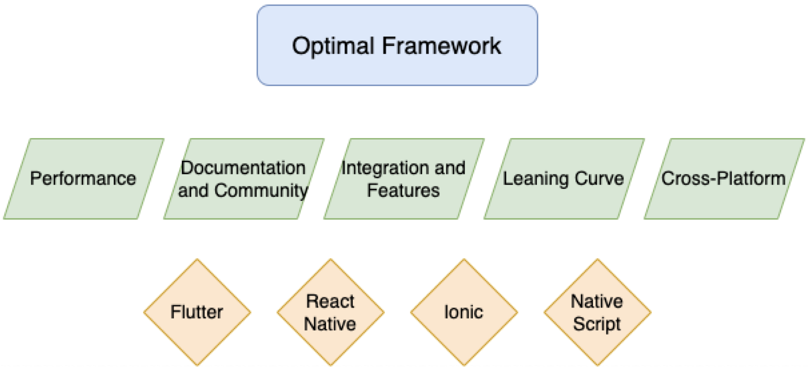


Figure 3.3: AHP Diagram.

To determine the significance of one criterion in relation to the others, the following categorization is made where **A** represents Cross-Platform, **B** represents Documentation and Community, **C** represents Integration and Features, **D** represents Learning Curve and **E** represents Performance:

Table 3.2: Criterion Comparison Matrix

	A	**B**	**C**	**D**	**E**
A	1	3	4	4	5
B	1/3	1	3	3	4
C	1/4	1/3	1	1	3
D	1/4	1/3	1	1	3
E	1/5	1/4	1/3	1/3	1

The classification of the criteria in the development of the software application is a crucial step in the process. The order in which the criteria are ranked is based on general experience in the field and aligns with the goals of this dissertation project.

Cross-Platform is considered the most important criterion as it is mandatory to develop a single codebase that can work on both Android and iOS platforms. While the framework may have a large community, be well-documented, and have a flat learning curve, not being cross-platform is not a viable option.

Documentation and Community come second in terms of importance as these precede all other criteria during the development process. Whether the intention is to integrate an already developed solution or to implement it oneself, it is necessary to read the documentation and gather information on how to do so. The learning curve is also smoothed out by having access to documentation and assistance from an active community.

Integration and Functionality and Learning Curve are given the same priority. Being able to quickly build something from scratch is just as important as integrating functionality into the app.

Performance is considered the least important criterion, despite being a crucial factor in providing quality user experiences. Performance is only relevant once the software has been fully implemented and is being used by users. This dissertation proposal is focused on developing a concept that will create a completely new solution where having the desired functionalities implemented and functional presents greater priority than optimizing performance and resource management.

While all the criteria play a critical role in ensuring successful development, it is important to note that ideally no aspect should be disregarded in favor of another. The comparison matrix is followed by the priority matrix, which is based on the pairwise comparison and the relative weight matrix. These matrices help to ensure that all criteria are considered in the decision-making process and that the most appropriate choice is made for the development of the software application.

Table 3.3: Priority Matrix

Criteria	**Priority**
A	46.4%
B	25.3%
C	11.4%
D	11.4%
E	5.5%

Table 3.4: Relative Weight Matrix

Criteria	Weight
A	0.464
B	0.252
C	0.114
D	0.114
E	0.55

The next step in the Analytic Hierarchy Process after computing the Relative Weight Matrix is to compare the alternatives. In this case, there are four alternatives to be evaluated, Ionic, React Native, NativeScript, and Flutter. Each of these alternatives must be compared against one another in terms of how well they fulfill each of the criteria. To do this, the same Fundamental Scale of Pairwise Comparison that was used to classify the criteria will be used to evaluate the alternatives.

When evaluating the alternatives, a brief explanation should be given on why the values were assigned in a particular way. This will help to provide context and ensure that the results of the comparison are accurate and meaningful. The weights of each alternative for each criterion will then be used to calculate a weight matrix, following the same process that was used to compute table 3.4.

It's important to note that all of the alternatives play a crucial role in ensuring the successful development of the application, and it's ideal to not disregard any one aspect in favor of another. The process of evaluating the alternatives based on the criteria is a critical step in the Analytic Hierarchy Process, and it's essential to ensure that it is performed with precision and care.

Comparison between **Cross-Platform** alternatives:

Table 3.5: Cross-Platform Criterion Comparison Matrix

	Ionic	Flutter	NativeScript	React Native
Ionic	1	2	3	5
Flutter	1/2	1	3	5
NativeScript	1/3	1/3	1	2
React Native	1/4	1/4	1/2	1

Table 3.6: Cross-Platform Relative Weight Matrix

Criteria	Weight
Ionic	0.455
Flutter	0.32
NativeScript	0.139
React Native	0.086

Cross-platform code reuse refers to the ability to use the same codebase across multiple platforms, such as Android and iOS. In the context of mobile app development, all of the frameworks being considered to have cross-platform capabilities and allow for a significant degree of code reuse. However, the extent to which this is possible varies among the frameworks.

Ionic stands out for its ability to also build for the web and its flexible style-based component library, which can be used on both Android and iOS. On the other hand, Flutter uses Material Design-styled components for Android and Cupertino-styled components for iOS. While this allows for a highly customized look and feel, a runtime check is required to determine the device platform, which may have an impact on performance.

NativeScript, in contrast, does not apply a specific styled theme and instead compiles the components into the platform-specific representation. However, components that do not exist on both platforms must be manually managed by the developer. React Native also compiles components to their native equivalents but only provides a basic set of components and requires additional platform-specific styling for custom components.

In conclusion, the extent of cross-platform code reuse varies among the frameworks and ultimately depends on the specific requirements and goals of the project. It is important to carefully consider the trade-offs between customization and performance when making a decision on the most appropriate framework.

Comparison between **Documentation and Community** criterion:

Table 3.7: Documentation and Community Comparison Matrix

	Ionic	**Flutter**	**NativeScript**	**React Native**
Ionic	1	1/3	3	1/4
Flutter	3	1	4	1/2
NativeScript	1/3	1/4	1	1/6
React Native	4	2	6	1

Table 3.8: CDocumentation and Community Weight Matrix

Criteria	**Weight**
React Native	0.499
Flutter	0.299
ionic	0.137
NativeScript	0.066

The documentation of each framework is comprehensive and equipped with well-explained examples and use cases. However, the documentation for NativeScript was found to be challenging to understand and navigate, especially for those with limited experience. On a personal note, Flutter has the best documentation with clear descriptions and rich examples. Additionally, the official Flutter YouTube channel provides quick and straightforward overviews of its extensive pool of components, known as widgets in Flutter. Both React and Ionic also have well-documented frameworks with plenty of examples.

In terms of community, React stands out as the most popular framework with a massive presence on GitHub, with 85k stars and over 2,057 contributors. It is used as a dependency by 324k GitHub projects. Flutter has a similar number of stars but with fewer contributors. Ionic and NativeScript (the core package) have 40k stars with 354 contributors and 18k stars with 159 contributors respectively.

In terms of their release date, Ionic was released first in 2013, followed by NativeScript in 2014, React Native in 2015, and Flutter in 2018, making it the most trending framework. In the GitHub developer survey of 2019, React Native was the most popular in the category of *Other Frameworks* within the *Most Popular Technologies*. Flutter was also mentioned in this list, although not as highly ranked as React, but still impressive given its relatively recent release. In the same category, Flutter was ranked as the third most loved, and React Native was ranked as the third most wanted in the *Most Loved, Dreaded, and Wanted Other Frameworks, Libraries, and Tools* title. In Slant, a product recommendation community website, React Native, Flutter, and Ionic are all highly regarded.

Comparison between **Integration and Features** criterion:

Table 3.9: Integration and Features Comparison Matrix

	Ionic	**Flutter**	**NativeScript**	**React Native**
Ionic	1	2	2	2
Flutter	1/2	1	1	1
NativeScript	1/2	1	1	1
React Native	1/2	1	1	1

Table 3.10: Integration and Features Weight Matrix

Criteria	**Weight**
Ionic	0.4
Flutter	0.2
React Native	0.2
NativeScript	0.2

The evaluation of the available features (core and third-party) and their ease of integration is a crucial aspect of this topic. These features could include access to native device functionalities, payments, data retrieval from external sources, maps, and even the utilization of Backend as a Service (BaaS) systems.

With Ionic, the developer has access to a vast pool of functionalities due to the fact that it uses JavaScript and runs as a web application. The npm repository, which is the second-largest package manager and the largest in the JavaScript ecosystem, has over 1 million packages available. Additionally, there are also some Ionic-specific packages developed by the community that can be found on the npm repository. With Ionic, there is a high likelihood of finding a solution to most problems a developer might encounter. In terms of native access, Ionic utilizes the Capacitor runtime engine that provides access to common device functionalities such as device storage, background tasks, and camera. Any missing native functionalities can be implemented and integrated through the Capacitor runtime engine. Most APIs have fallback options for each platform (Android, iOS, and Web), making their integration effortless.

React Native, being a JavaScript framework also has access to the npm repository. However, it only provides core functionalities for native access and heavily relies on its community for third-party tools. The same holds true for NativeScript, with a wider range of core functionalities available, but a smaller community compared to React Native.

Flutter, on the other hand, uses its own package manager known as Pub. While the size of its repository is significantly smaller compared to npm, the most significant packages have been or are in the process of being ported. In terms of native functionalities, Flutter provides multiple features out-of-the-box, and developers have the means to implement new ones. Although some alternatives come with more out-of-box functionalities, there are also those with larger communities that provide third-party functionalities, thus, the weight assigned was not much different.

Comparison between **Learning Curve** criterion:

Table 3.11: Learning Curve Comparison Matrix

	Ionic	**Flutter**	**NativeScript**	**React Native**
Ionic	1	1/3	2	4
Flutter	3	1	4	5
NativeScript	1/2	1/4	1	1
React Native	1/4	1/5	1	1

Table 3.12: Learning Curve Weight Matrix

Criteria	**Weight**
Flutter	0.541
Ionic	0.252
NativeScript	0.114
React Native	0.093

The ease of starting and implementing the basics was the main consideration in this criterion. Based on personal experience, React Native is known to be the most challenging platform to work with. Despite being built with JavaScript, a language widely used by web developers, its declarative paradigm can be confusing without prior experience and without the benefit of type-safe features. On the other hand, Ionic and NativeScript offer a variety of flavors for developers to work with, making it smoother for web developers who may already have experience with some of these technologies. Both platforms also support TypeScript, which enforces type-safety mechanics that are missing in JavaScript.

When it comes to ease of use, Flutter stands out as the simplest platform to get started with. Its Dart syntax is similar to JavaScript, and it's both dynamically and strongly typed. It's also compiled, as opposed to interpreted, which speeds up development by providing errors during compile time. Flutter, with its declarative approach, is the clear winner in terms of ease of use, with React Native coming in as a close second. The learning curve between Ionic and NativeScript is low when compared directly, but they share a similar learning curve since they can both be used with Angular and Vue, making it easier to transition from one to the other.

Comparison between **Performance** criterion:

The performance of the alternatives was considered in this criterion. Ionic was found to be the least performant among the alternatives, followed by NativeScript. Flutter and React Native were close in performance, with Flutter being the winner, as it compiles the entire codebase to native machine code for better performance.

Table 3.13: Performance Comparison Matrix

	Ionic	**Flutter**	**NativeScript**	**React Native**
Ionic	1	1/9	1/8	1/8
Flutter	9	1	3	2
NativeScript	8	1/3	1	1/3
React Native	8	1/2	3	1

Table 3.14: Performance Weight Matrix

Criteria	**Weight**
Flutter	0.467
React Native	0.325
NativeScript	0.172
Ionic	0.036

To finalize this process, the last step is to find the global weight of an alternative, the product of each alternative's criterion weight, and the relative weight of the criteria, to reach the conclusion of which alternative is better suited for the case study. This value is calculated using the following formula:

Table 3.15: Global Weight Matrix

	A		**B**		**C**		**D**		**E**		**GW**
Ionic	0.212	+	0.0347	+	0.0456	+	0.0287	+	0.002	=	0.323
Flutter	0.1491	+	0.0756	+	0.0228	+	0.0617	+	0.0257	=	0.3349
NativeScript	0.0648	+	0.0167	+	0.0228	+	0.013	+	0.0095	=	0.1267
React Native	0.0401	+	0.1262	+	0.0228	+	0.0105	+	0.0179	=	0.2175

Table 3.16: Ranking of Cross-Platform Frameworks

Frameworks
1º: Flutter
2º: Ionic
3º: React Native
4º: NativeScript

3.4 Function Analysis and System Technique (FAST)

In this section, the functional analysis of the system will be carried out using the FAST methodology, which is a technique whose final product is a diagram that graphically represents the logical relationships between the functions of a given project, product, process, or service, based on questions such as "How?", "Why?" and "When?" This diagram can be used to consult the proposed solution for a given problem and to validate whether it meets the needs of the same (Bouchereau and Rowlands, 2000).

According to Prasad, 1998, the FAST diagram consists of several sections, represented in Figure 24. Initially, the higher-order function is defined, which is equivalent to the most basic function of the system. Then, the diagram is built from left to right, where each function further to the right corresponds to a secondary function of the previous function, answering the "How?" question of the previous function. This originates the main flow of the system. On the other hand, if the diagram is interpreted from right to left, the function furthest to the left answers the "Why?" question of the immediately previous function. In addition, each of these functions can give rise to other secondary functions that are represented vertically (top to bottom). Finally, there are other functions: the One-time Functions and All-time Functions (which are presented at the top of the diagram) and the Unwanted Functions (presented at the bottom of the artifact).

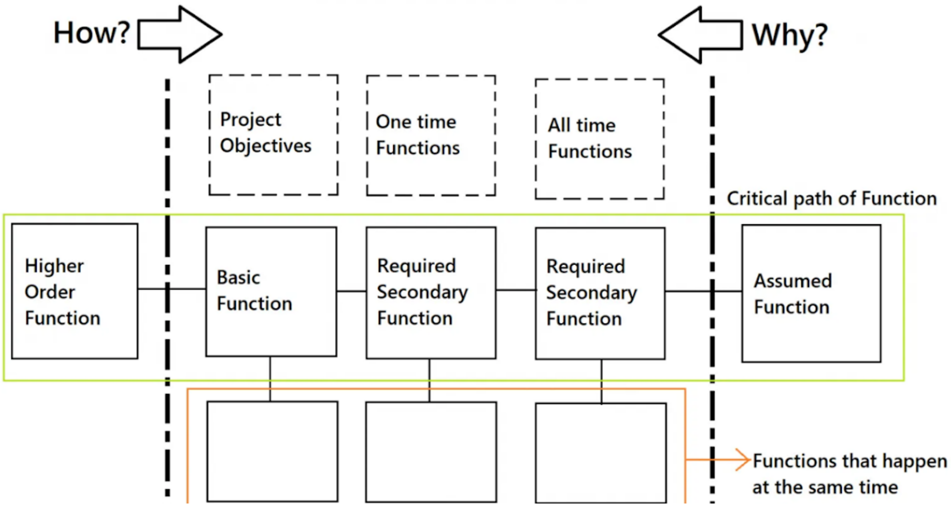

How?
Why?
Project Objectives
One time Functions
All time Functions
Critical path of Function
Higher Order Function
Basic Function
Required Secondary Function
Required Secondary Function
Assumed Function
Functions that happen at the same time

Figure 3.4: FAST diagram

In this context, it can be seen in Figure 3.5 the FAST analysis of the project associated with this document.

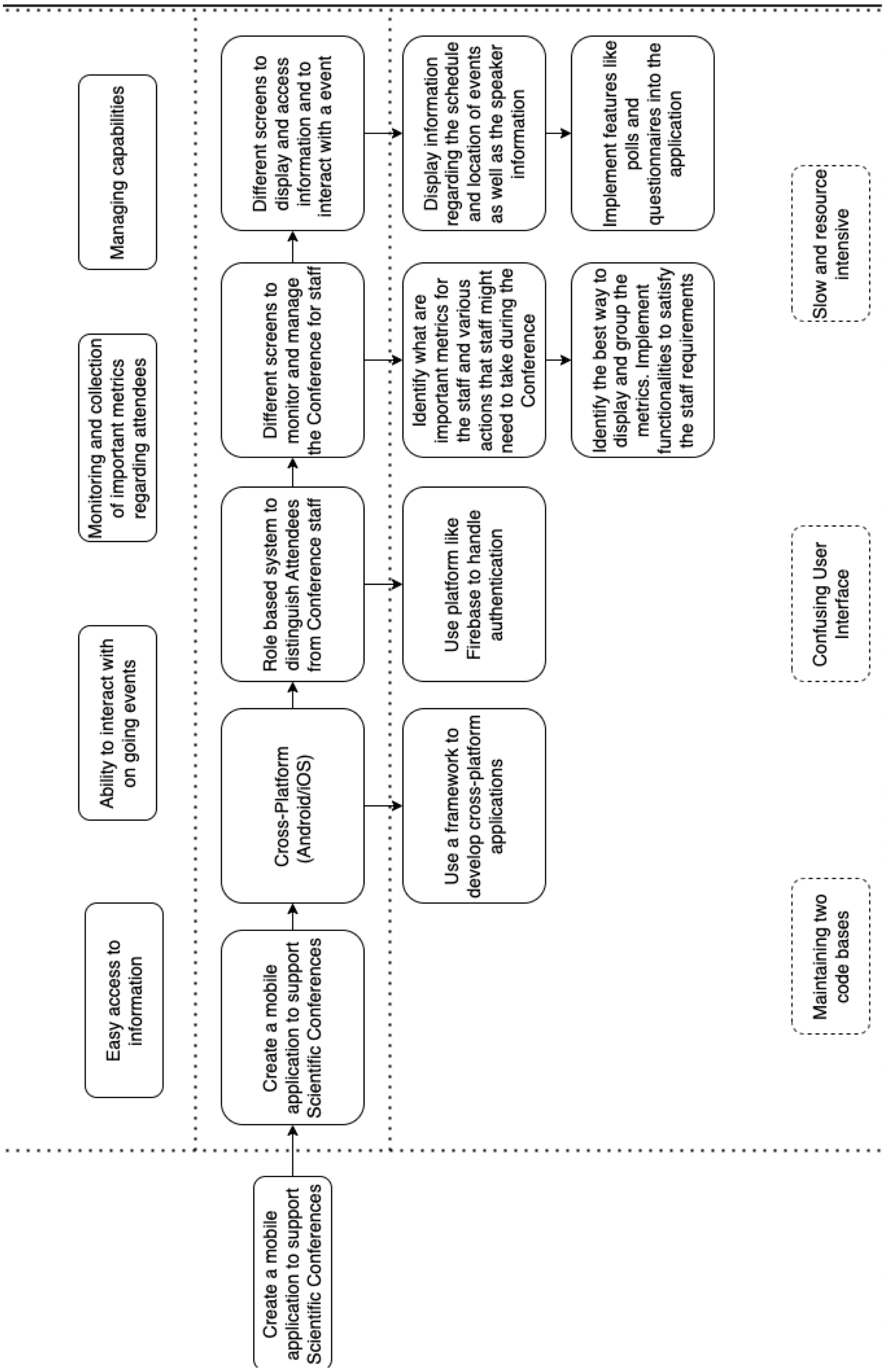

Create a mobile application to support Scientific Conferences
Easy access to information
Ability to interact with on going events
Monitoring and collection of important metrics regarding attendees
Managing capabilities
Create a mobile application to support Scientific Conferences
Cross-Platform (Android/iOS)
Role based system to distinguish Attendees from Conference staff
Different screens to monitor and manage the Conference for staff
Different screens to display and access information and to interact with a event
Use a framework to develop cross-platform applications
Use platform like Firebase to handle authentication
Identify what are important metrics for the staff and various actions that staff might need to take during the Conference
Identify the best way to display and group the metrics. Implement functionalities to satisfy the staff requirements
Display information regarding the schedule and location of events as well as the speaker information
Implement features like polls and questionnaires into the application
Maintaining two code bases
Confusing User Interface
Slow and resource intensive

Figure 3.5: FAST analysis

Chapter 4

Analysis and Design

In this chapter, the problem analysis is performed. Initially, the Quality Evaluation Framework (QEF) is applied to the project to measure the quality of the proposed solution, followed by a brief description of the system architecture. Then, the problem is deepened through requirement engineering. The system actors and the identified functional and non-functional requirements are presented along with a brief description of each of these requirements.

4.1 Quality Evaluation Framework

The proposed solution in this thesis requires a comprehensive understanding of the system and an evaluation of its quality to monitor the processes involved over time. The QEF is a tool that measures the quality of a system during its development cycle and facilitates communication between all parties involved, including stakeholders/clients, product owners, and the development team (Heidari and Loucopoulos, 2014). The QEF model will be used to:

- Divide all features into the Dimensions of the application.
- Identify the Quality Factors relevant to each Dimension.
- Determine Requirements for each Quality Factor.
- Set Metrics for these Requirements.
- Assess the overall quality of the system.

To determine the Dimensions and Quality Factors, a comprehensive list of functional and non-functional Requirements for the system must be created. Each Requirement is assigned a weight (2, 4, 6, 8, 10) to represent its significance, and metrics are identified to measure completion. The metrics are levels of fulfillment (%). Requirements are grouped into Quality Factors and then grouped into Dimensions. Keeping track of the fulfillment level of each Requirement with a model such as QEF is crucial to ensure the team accurately measures the system's overall quality.

From the analysis of the application's desired behavior, the following **Dimensions** and Quality Factors were established: Functional:

- General
- Social
- Settings

- Search

Non-Functional:

- Adaptability
- Performance

The tables in the next two subsections apply each dimension with the levels of **partly implemented** and **fully implemented**.

4.1.1 Functional Quality Factors:

Table 4.1: Functional, General Quality Factor

Requirement	**Description**	**Partly Implemented**	**Fully Implemented**
FGEN1 - User creates an account.	User creates an account in the application.		User creates an account using the application.
FGEN2 - Admin creates a staff account.	The admin creates an account for conference organizers.		Admin creates accounts for staff.
FGEN3 - The user can add and show the attendee ticket.	The user can add the attendee ticket via order number or ticket number and can show the generated QR code representing the ticket.		The user is able to add and show the attendee ticket.

Table 4.2: Functional, Social Quality Factor

Requirement	**Description**	**Partly Implemented**	**Fully Implemented**
FSO1 - The user can edit his profile.	User has a window to view and change his profile.	User can change personal details	User can change personal details, picture, and share LinkedIn profile.
FSO2 - The user can see the agenda of the conference.	The user can access a calendar view where all the events of a conference are displayed.	The user can access the calendar view to see the agenda of the conference	The user is able to access the calendar view and interact with each event to see more details about it.
FSO3 - The conference organizers can add and remove events to the calendar view.	The conference organizers can add or remove events to the calendar view and customize the information to display when the events are interacted with.	The conference organizers can only add and remove events to the calendar view.	The conference organizers can add and remove events to the calendar view and customize basic details about the event to be displayed when the calendar is interacted with.
FSO4 - The conference organizers can send notifications to the attendees.	The conference organizers can send important notifications in real-time to all the users of the application.		The conference organizers can easily send simple text notifications to all the users of the application.
FSO5 - The user can rate an event.	The user after attending an event can rate it.	All users can rate an event without validating if they attended it.	Only users that attended an event can rate the specific event and only after a specific amount of time has passed since it has started.

Table 4.3: Functional, Settings Quality Factor

Requirement	**Description**	**Partly Implemented**	**Fully Implemented**
FSE1 - The event organizers can customize basic event information.	The event organizers can customize basic details about events and the respective speakers.		The event organizers are able to customize information regarding events and speakers using the application.
FSE2 - The event organizers can access and change views and filters on the metrics dashboard.	The event organizers are able to access a window where various real-time metrics are displayed that can be filtered.	The staff organizers can only access the dashboard and metrics.	The event organizers are able to access and customize the dashboards and metrics in various ways.
FSE3 - Light and Dark modes.	The application supports light and dark modes and all users can change between them.		All users of the application can easily and briefly change from Light and Dark modes when using the application.

Table 4.4: Functional, Search Quality Factor

Requirement	**Description**	**Partly Implemented**	**Fully Implemented**
FSH1 - The user can search for events and speakers.	User can search for events to see access details and speakers to see the speaker profile.	User can search for either events or speakers.	User can search for both events and speakers in the same search box.
FSH2 - The user can search for specific zones to access details and directions.	The user is able to search for specific areas like coffee rooms, and bathrooms among others, and see details and directions to them.	The user can only search for some specific areas.	The user is able to query all available areas and get detailed information about them.

4.1.2 Non-Functional Quality Factors:

Table 4.5: Non-Functional, Adaptability Quality Factor

Requirement	Description	Partly Implemented	Fully Implemented
NFAD1 - Application supports Android and iOS Operating Systems.	The application is developed in one language and compiled into native Android and iOS applications.		The application compiles and runs natively in both Operating Systems.
NFAD2 - The application must resize to different screen sizes.	Depending on the type of smartphone, the application should resize correctly the windows and widgets on both Operating Systems.	Application partly resizes most windows and widgets correctly to both Operating Systems.	The application is tested and supports all different sizes of screens in both Operating Systems.
NFAD3 - User Interface Multiple Language Support.	The user can select Portuguese or English language to use the application.	The application only supports one language.	The application supports both English and Portuguese and can easily be extended to more languages.

Table 4.6: Non-Functional, Performance Quality Factor

Requirement	Description	Partly Implemented	Fully Implemented
NFPE1 - Assets and data are cached locally, improving responsiveness and network usage.	Cache data that depends on external services locally, particularly for information that is likely to be unchanging for the majority of the time, such as image data.		Assets and data ware cached.
NFPE2 - The application should load low-resolution images on a low network connection.	For improving the app's responsiveness, low-resolution assets are loaded when the connection quality is low. If the user stays on the same screen, the app then loads the original asset.	Always load low-quality assets first.	The application tests the network connection and decides which quality to load.

4.2 System Architecture

A deployment diagram is a visual representation of the physical topology of a system. It shows the allocation of software components and which hardware infrastructure, known as nodes, they are running on. In the context of this specific application, the deployment architecture can be described as follows (figure 4.1):

The overall system architecture of this application consists of two primary components: the mobile app, which runs on the user's smartphone, and a backend SpringBoot-based application that acts as a server connected to a MySQL database, which handles data persistence. The SpringBoot service exposes a REST API that can be consumed by the client, and data flows through this API using HTTP and exchanging JSON payloads.

To store image data, the mobile application uses Google Cloud Storage, a specialized service provided by Google, and accesses its API using the Firebase SDK to upload, download, and allocate assets. Data is sent and fetched via HTTP, and the URL assigned to each item is stored in the SpringBoot application. The decision to use an external dedicated server

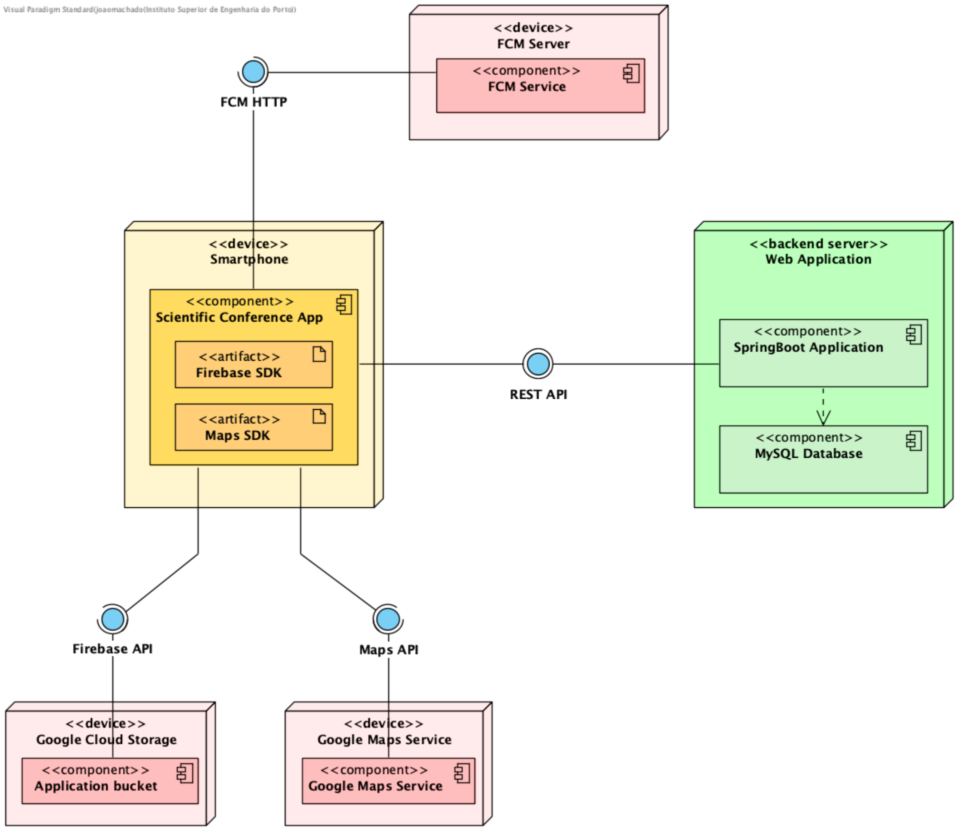


Figure 4.1: Deployment Diagram

for media storage was made because it was easy to integrate with the Flutter framework used to develop the mobile application, and it provides robust capabilities to handle uploads, downloads, and storage of large files, ensuring security even for unstable connections. It also has the ability to scale automatically and allows user-specific rules for each access type (read, write, delete). Storing image data in the same relational database used for business logic would significantly increase its size and reflect on its cost and performance.

The mobile application uses the Google Maps Platform for Maps and Distance Matrix API to integrate the map view and some map/location-related functionalities. The Maps API provides map tiles and information to identify locations, such as street names and places, while the Distance Matrix API is used for path computations, such as the shortest path, its distance, and the time it takes to travel from one point to another, using multiple travel options (by foot, bicycle, car).

Finally, the mobile application uses Firebase Cloud Messaging (FCM), a service provided by the Firebase suite, for push notification functionality. A token is generated for each device and sent to the SpringBoot application to be stored. When a specific event that triggers a notification occurs, the server uses that device's token to send a message, with limited payload, via the FCM server to either targeted devices or broadcast to all devices.

4.3 Core Features

In order to effectively develop a mobile application that caters to scientific conferences, it is essential to clearly understand the main features of the app and its intended use. The design of the application was formulated with careful consideration of the following key components:

- Profile Management.
- Ticket Management.
- Conference Management.
- Event Management.
- Search Functionality.
- Metrics View.
- Calendar View.
- Map View.

Prior to providing a detailed explanation of the aforementioned items, it is crucial to define the following terms utilized by the application:

- Conference Attendee - A user of the mobile application. When this user interacts with the application, he can log in to the application, associate a ticket, and perform attendee-related actions.
- Conference Manager - A user of the application. This user can log in to the application and depending on their permissions can perform several monitoring and management actions into an event or the entire conference.
- Conference - A conference is the concept of aggregating multiple events under a common topic. A conference may have a duration of several days, holding multiple events.
- Event - An Event is an occurrence within a conference held by one or multiple speakers where multiple attendees will gather in a specific place to interact with that speaker.

4.4 Functional Requirements

In this section, we will identify and analyze the functional requirements for the development of the mobile application to support scientific conferences organized by ISEP. Functional requirements are the features and capabilities that the application must provide to meet the needs of its users. They describe the tasks that the application must be able to perform and the interactions between the user and the application. Identifying and analyzing functional requirements is a critical step in the development process, as it provides a clear understanding of what the application should do and guides the design and implementation of its features. In this section, a set of functional requirements is presented and discussed based on the analysis of the needs of conference organizers and attendees. By defining the functional requirements in detail, we aim to ensure that the application will meet the expectations and requirements of its users and provide a seamless and enjoyable conference experience.

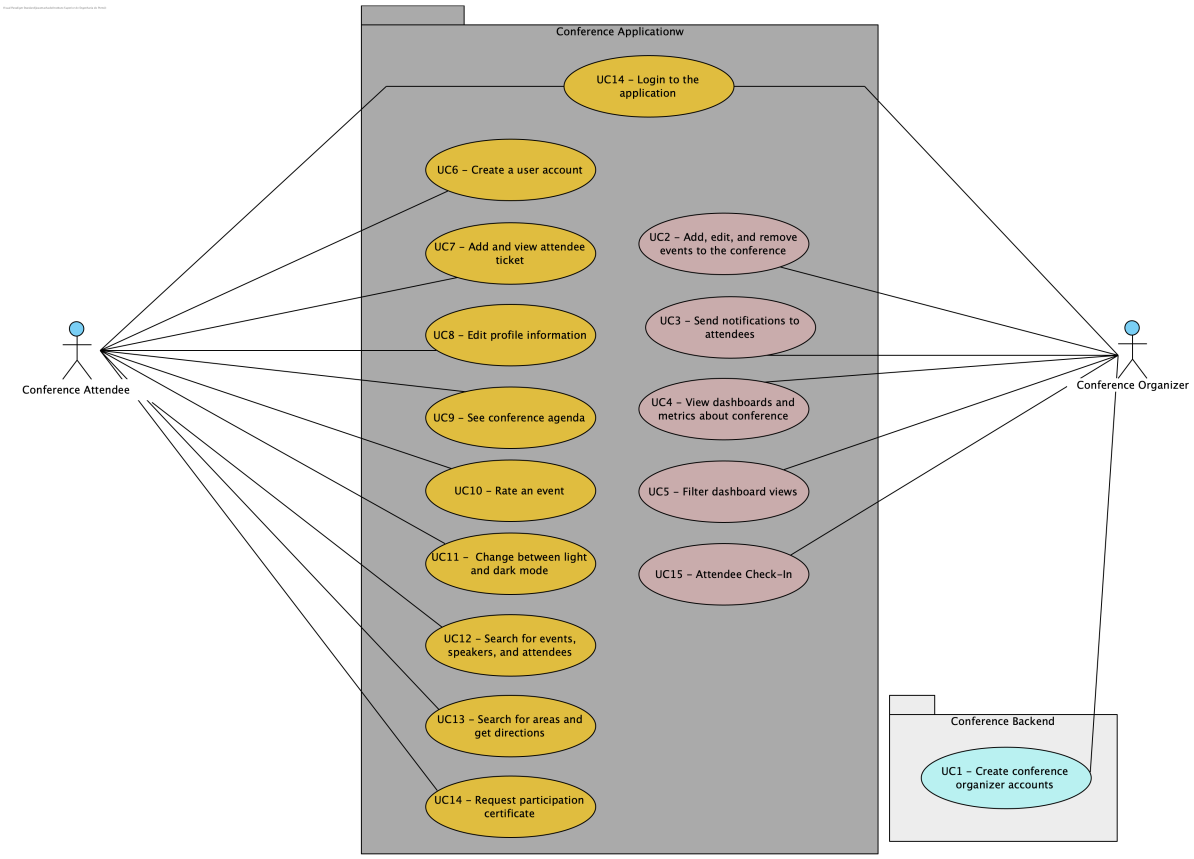

Conference Applicationw
UC14 – Login to the application
UC6 – Create a user account
UC7 – Add and view attendee ticket
UC8 – Edit profile information
UC9 – See conference agenda
UC10 – Rate an event
UC11 – Change between light and dark mode
UC12 – Search for events, speakers, and attendees
UC13 – Search for areas and get directions
UC14 – Request participation certificate
UC2 – Add, edit, and remove events to the conference
UC3 – Send notifications to attendees
UC4 – View dashboards and metrics about conference
UC5 – Filter dashboard views
UC15 – Attendee Check–In
Conference Attendee
Conference Organizer
Conference Backend
UC1 – Create conference organizer accounts

Figure 4.2: Use Case (UC) Diagram.

Figure 4.2 presents the associated use cases with this system using a use case diagram in the Unified Modeling Language (UML) notation.

4.4.1 UC1 - Create conference organizer accounts

Figure 4.3 represents the system-level sequence diagram of UC1, performed by the conference organizer. Initially, the conference organizer interacts with the button to add a new conference organizer account. The application presents a form with the necessary data to create a new conference organizer account. The conference organizer inputs the data, and the application validates and saves the new account.

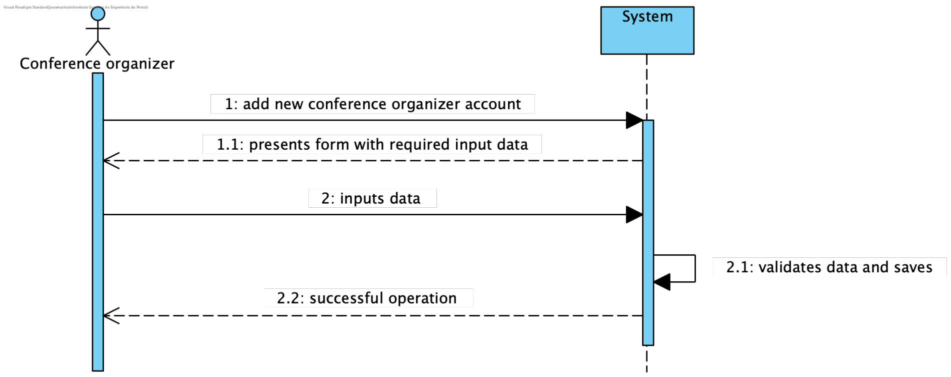

Conference organizer
System
1: add new conference organizer account
1.1: presents form with required input data
2: inputs data
2.1: validates data and saves
2.2: successful operation

Figure 4.3: UC1 - Sequence Diagram

4.4.2 UC2 - Add, edit, and remove events to the conference

Figure 4.4 represents the system-level sequence diagram of UC2, performed by the conference organizer. Initially, the conference organizer goes into the view of managing events of the conference. The system displays all events for that conference organized by date, where the earlier events appear first. The conference manager can interact with a button to add, the application prompts the user with a form where the conference manager inputs data regarding the new event. The application validates the data and saves the new event. Additionally, the conference manager can interact with the events already created to edit information regarding the event or delete it.

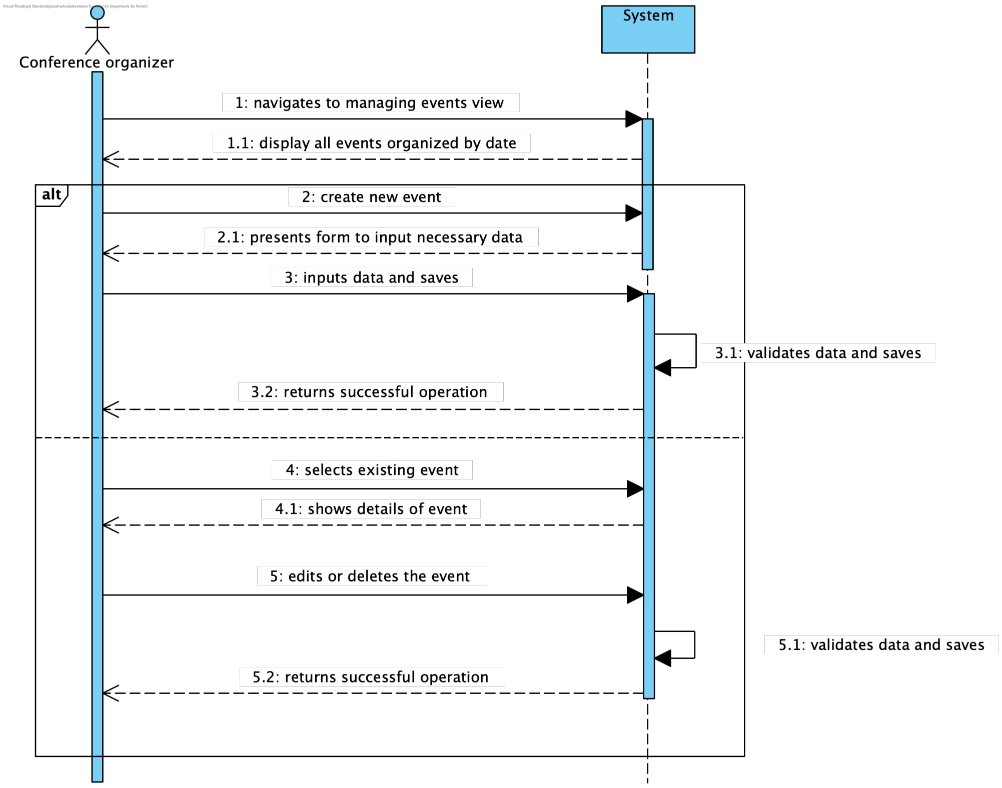

Conference organizer
System
1: navigates to managing events view
1.1: display all events organized by date
alt
2: create new event
2.1: presents form to input necessary data
3: inputs data and saves
3.1: validates data and saves
3.2: returns successful operation
4: selects existing event
4.1: shows details of event
5: edits or deletes the event
5.1: validates data and saves
5.2: returns successful operation

Figure 4.4: UC2 - Sequence Diagram

4.4.3 UC3 - Send notifications to attendees

Figure 4.5 represents the system-level sequence diagram of UC3, performed by the conference organizer. Initially, the conference organizer navigates to the view for sending notifications. The application displays an input box where the conference manager can input the text he wants to notify attendees with. The application validates the request and sends the notification to the attendee's devices.

4.4.4 UC4 - View dashboards and metrics about conference

Figure 4.6 represents the system-level sequence diagram of UC4, performed by the conference organizer. Initially, the conference organizer navigates to the view regarding information of the conference. The application displays multiple dashboards and metrics regarding the ongoing conference.

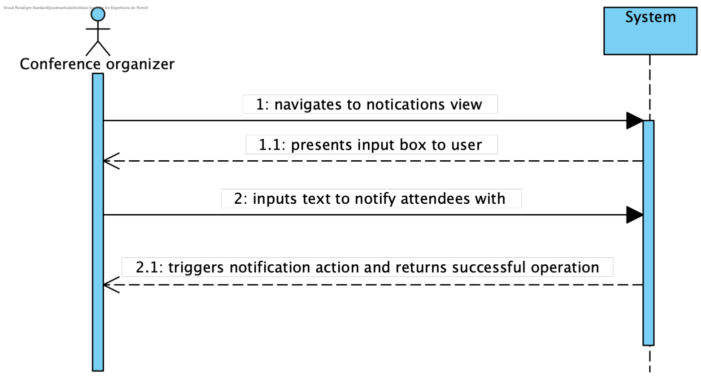

Conference organizer
System
1: navigates to notications view
1.1: presents input box to user
2: inputs text to notify attendees with
2.1: triggers notification action and returns successful operation

Figure 4.5: UC3 - Sequence Diagram

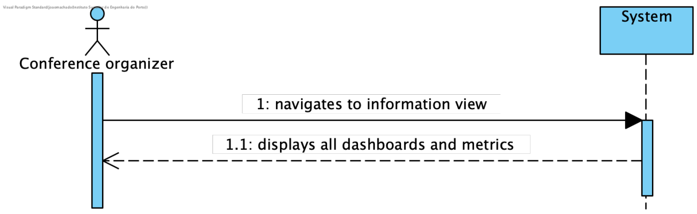

Conference organizer
System
1: navigates to information view
1.1: displays all dashboards and metrics

Figure 4.6: UC4 - Sequence Diagram

4.4.5 UC5 - Filter dashboard views

Figure 4.7 represents the system-level sequence diagram of UC5, performed by the conference organizer. After navigating to the view described in the above UC, the application, for each dashboard, displays a selection box with multiple predefined filters. When the conference organizer selects a filter for a specific dashboard, the information on the dashboard is rearranged accordingly.

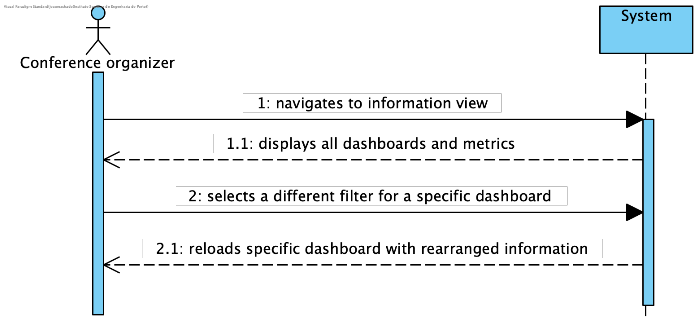

Conference organizer
System
1: navigates to information view
1.1: displays all dashboards and metrics
2: selects a different filter for a specific dashboard
2.1: reloads specific dashboard with rearranged information

Figure 4.7: UC5 - Sequence Diagram

4.4.6 UC6 - Create a user account

Figure 4.8 represents the system-level sequence diagram of UC6, performed by the conference attendee. Initially, the conference attendee interacts with the button to add a new conference organizer account. The application presents a form with the necessary data to create a new conference organizer account or to use Gmail or Facebook existing accounts. The conference organizer inputs the data, and the application validates and saves the new account.

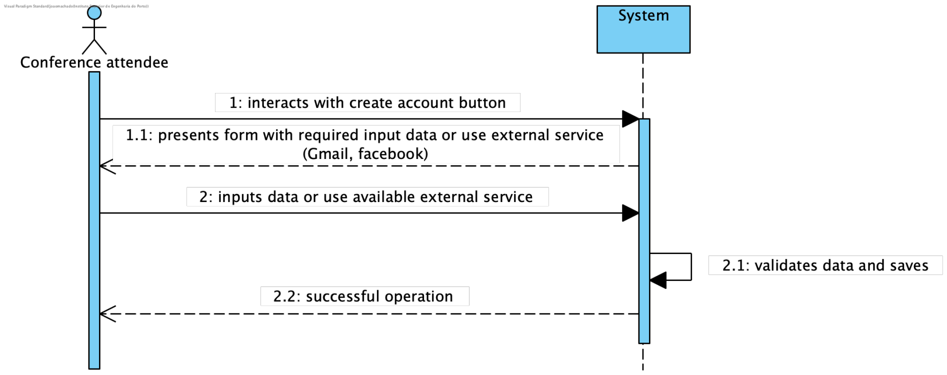

Conference attendee
System
1: interacts with create account button
1.1: presents form with required input data or use external service (Gmail, facebook)
2: inputs data or use available external service
2.1: validates data and saves
2.2: successful operation

Figure 4.8: UC6 - Sequence Diagram

4.4.7 UC7 - Add and view attendee ticket

Figure 4.9 represents the system-level sequence diagram of UC7, performed by the conference attendee. Initially, the conference attendee interacts with the button to add a ticket to the application. The application displays an input box where the user can input the order number or the ticket number. The application validates the number and generates a QR code. If the user already has a ticket associated with the account, the application automatically displays that QR code.

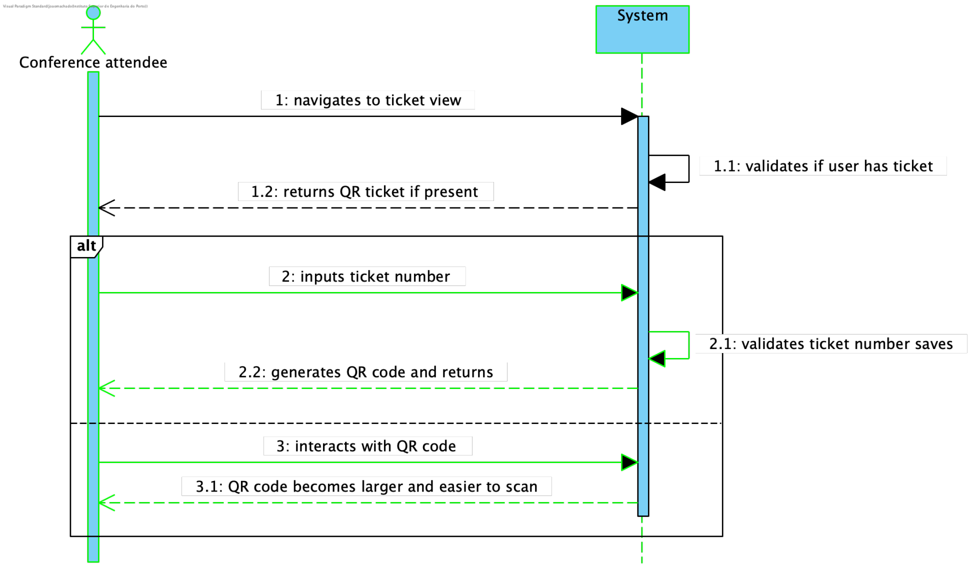

Conference attendee
System
1: navigates to ticket view
1.1: validates if user has ticket
1.2: returns QR ticket if present
alt
2: inputs ticket number
2.1: validates ticket number saves
2.2: generates QR code and returns
3: interacts with QR code
3.1: QR code becomes larger and easier to scan

Figure 4.9: UC7 - Sequence Diagram

4.4.8 UC8 - Edit profile information

Figure 4.10 represents the system-level sequence diagram of UC8, performed by the conference attendee. Initially, the conference attendee navigates to the personal profile view in the application. The application displays all the personal information about the user. The user can interact with the application to change specific information and to allow (default option) or disable the search functionality where they can be searched by other attendees. The application will validate the input data and save it.

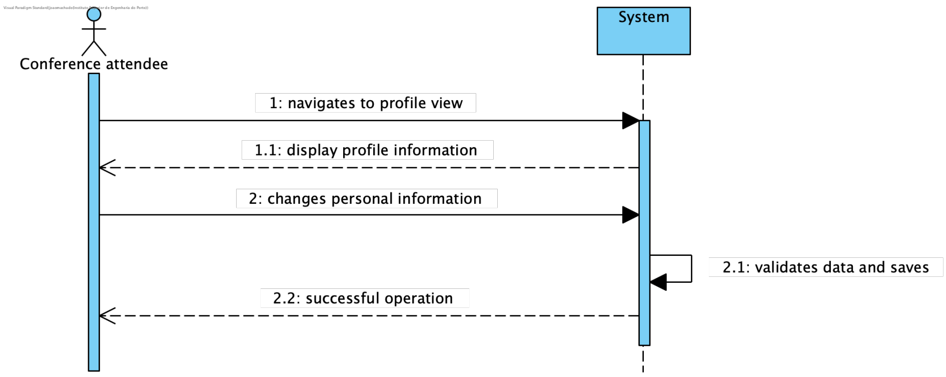

Conference attendee
System
1: navigates to profile view
1.1: display profile information
2: changes personal information
2.1: validates data and saves
2.2: successful operation

Figure 4.10: UC8 - Sequence Diagram

4.4.9 UC9 - See conference agenda

Figure 4.11 represents the system-level sequence diagram of UC9, performed by the conference attendee. Initially, the conference attendee selects the view of the conference agenda. The application displays all the events organized by date and time. The user can interact with each event to see details about it and the event speaker.

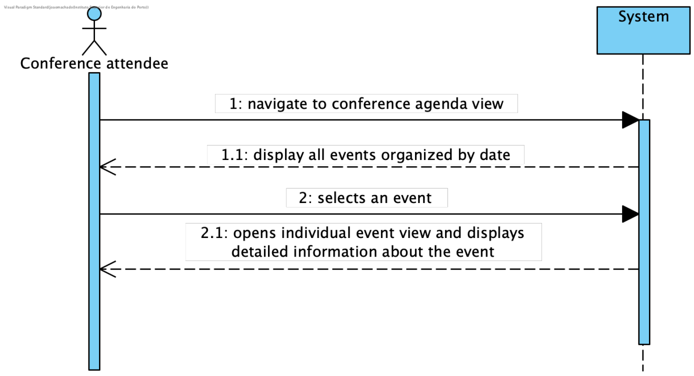

Conference attendee
System
1: navigate to conference agenda view
1.1: display all events organized by date
2: selects an event
2.1: opens individual event view and displays detailed information about the event

Figure 4.11: UC9 - Sequence Diagram

4.4.10 UC10 - Rate an event

Figure 4.12 represents the system-level sequence diagram of UC10, performed by the conference attendee. When the user is interacting with an event on the conference agenda view, a button is available to rate the event. After interacting with the button the application prompts the user with a basic star rating system from a one-star, minimum rating, to a five-star rating, max rating, and a textbox for additional feedback on the event. The application validates the input and saves it.

4.4.11 UC11 - Change between light and dark mode

Figure 4.13 represents the system-level sequence diagram of UC11, performed by the conference attendee. The conference attendee can toggle on or off a dark mode button to change the theme of the application from white to dark mode which changes the color scheme of all the applications to better suit dark or bright environments.

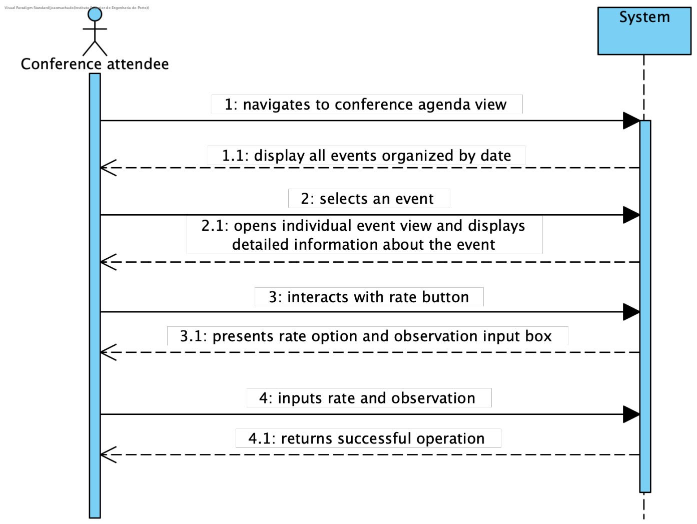

Conference attendee
System
1: navigates to conference agenda view
1.1: display all events organized by date
2: selects an event
2.1: opens individual event view and displays detailed information about the event
3: interacts with rate button
3.1: presents rate option and observation input box
4: inputs rate and observation
4.1: returns successful operation

Figure 4.12: UC10 - Sequence Diagram

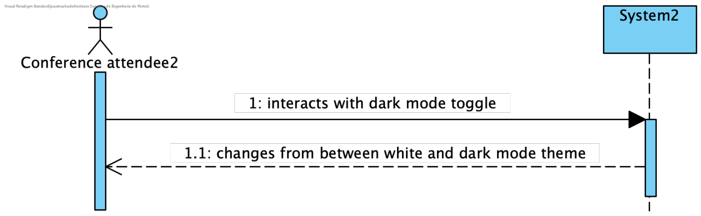

Conference attendee2
System2
1: interacts with dark mode toggle
1.1: changes from between white and dark mode theme

Figure 4.13: UC11 - Sequence Diagram

4.4.12 UC12 - Search for events, speakers, and attendees

Figure 4.14 represents the system-level sequence diagram of UC12, performed by the conference attendee. Initially, the conference attendee interacts with a search box on the application where the name of an event, speaker, or attendee can be introduced. The application validates this input searches all events, speakers, and attendees (that allow this feature), and returns a list of matches. The attendee sees the result set and can select a specific match to see more details about it.

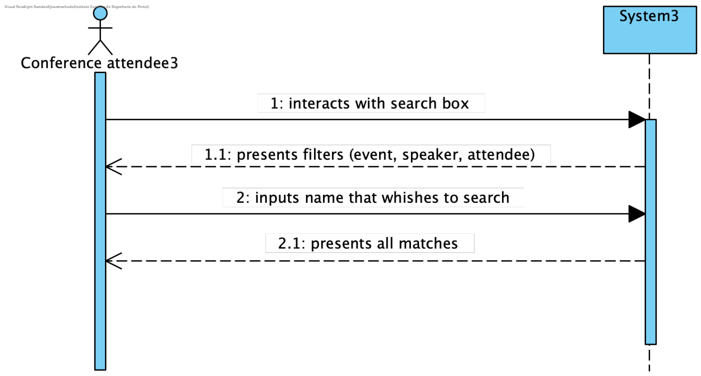

Conference attendee3
System3
1: interacts with search box
1.1: presents filters (event, speaker, attendee)
2: inputs name that whishes to search
2.1: presents all matches

Figure 4.14: UC12 - Sequence Diagram

4.4.13 UC13 - Search for areas and get directions

Figure 4.15 represents the system-level sequence diagram of UC13, performed by the conference attendee. Initially, the conference attendee navigates to the conference map view. The application displays a 2D map of the conference area. The user interacts directly with the map through drag and zoom movements. The user can also interact with the map to display the shortest way to move from his/her current position to the conference. The application validates this request and highlights the path on the map from the user's current location to the conference.

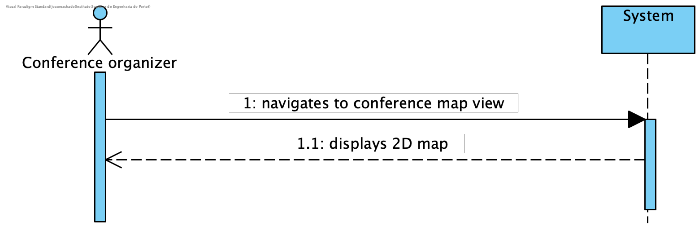


Figure 4.15: UC13 - Sequence Diagram

4.4.14 UC14 - Request participation certificate

Figure 4.16 represents the system-level sequence diagram of UC14, performed by the conference attendee. Initially, the conference attendee navigates to the conference certificate view. The application displays an input box for the name (mandatory) and email (optional to override the application account email) of the attendee. The attendee inputs the data and clicks on the save button. The application generates a participation certificate using the name provided and sends it by email to the attendee.

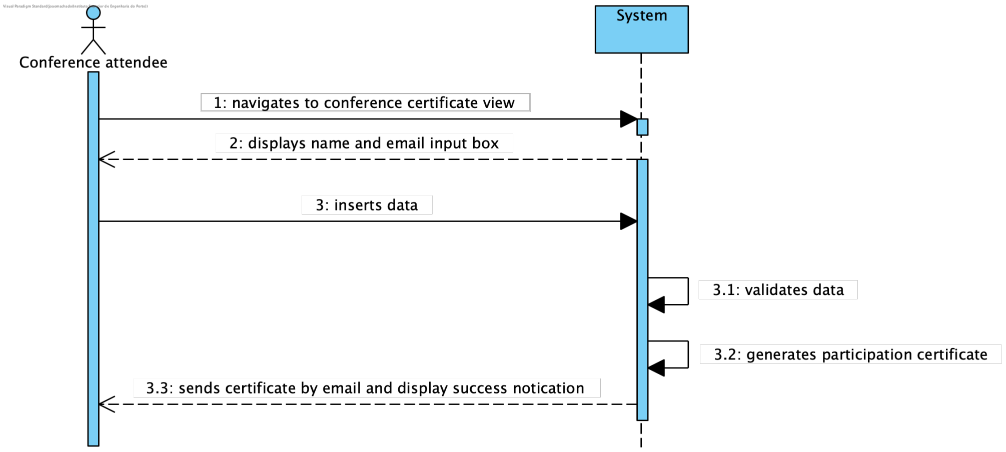

Conference attendee
System
1: navigates to conference certificate view
2: displays name and email input box
3: inserts data
3.1: validates data
3.2: generates participation certificate
3.3: sends certificate by email and display success notication

Figure 4.16: UC14 - Sequence Diagram

4.4.15 UC15 - Attendee Check-In

Figure 4.17 represents the system-level sequence diagram of UC15, performed by the conference organizer. Initially, the conference organizer navigates to the conference check-in view. The application uses the smartphone camera and displays what the camera is observing in the application. The conference organizer aims the smartphone camera, using the live feed displayed in the application, to the QR code provided by the conference attendee, on his smartphone or a printed paper. The application on the conference organizer's smartphone recognizes the QR code, validates it, and performs the check-in of the attendee displaying a success notification on the conference organizer's phone.

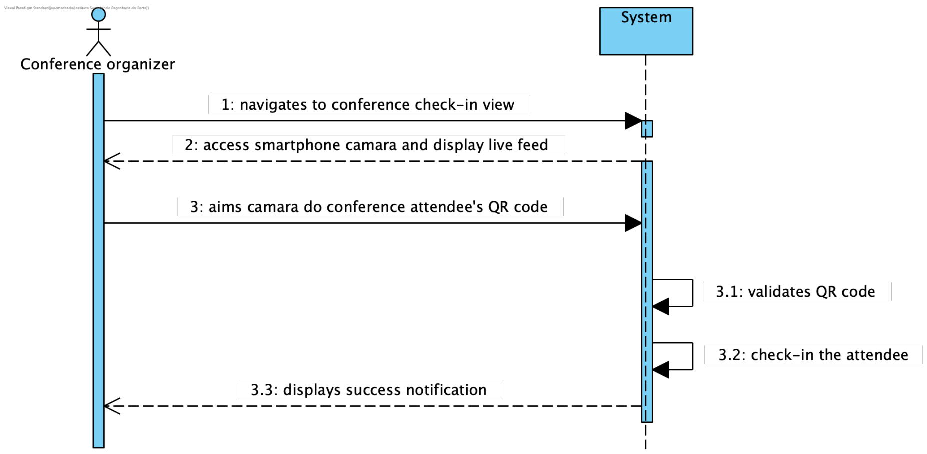

Conference organizer
System
1: navigates to conference check-in view
2: access smartphone camara and display live feed
3: aims camara do conference attendee's QR code
3.1: validates QR code
3.2: check-in the attendee
3.3: displays success notification

Figure 4.17: UC15 - Sequence Diagram

4.4.16 UC15 - Login to the application

Initially, the conference organizer or attendee opens the application. The application starts and prompts the user to log in to the application by inserting an email and password or to log in using Gmail or Facebook accounts. The user inputs the data, the application validates if the data is correct and if the user is a conference organizer or a conference attendee and displays the right views for each user.

4.5 Non-Functional Requirements

This section covers additional requirements that are not tied to the core features of the system:

- Interface: The application must be user-friendly and visually pleasing while following standard design practices. The UI should be adjustable to fit most screen sizes and should provide loading indicators such as shimmers, skeleton text, or low-resolution asset previews, to minimize loading times and enhance the user experience.
- Performance: The app's performance should be optimized by caching image data locally to avoid unnecessary downloads, and by caching remote data whenever possible to reduce redundant requests. This ensures that the app remains usable even in the absence of an internet connection or in cases of slow connectivity, providing users access to the most recent version of resources.
- Implementation: The mobile application must be compatible with both Android and iOS systems, and should not feature platform-specific behaviors or differences. This ensures that the app functions identically on both platforms, avoiding any potential confusion or user frustration.
- Scalability: The backend application must be capable of scaling up to meet increased demand, and handling heavy workloads as required without any slowdowns or issues. This ensures that the app can handle user traffic even during peak usage times.

In this chapter, the analysis and design of a mobile application to support scientific conferences organized by ISEP were presented. Initially, the Quality Evaluation Framework to guide our evaluation of the application's usability, functionality, and performance was defined. Followed by the System Architecture, outlining the high-level components and the interactions between them. Next, the Core Features of the application, which included Conference Management, Event Management, and Profile Management, and different views were identified and analyzed.

Subsequently, the Functional Requirements of the application were discussed, which describe the tasks that the application must perform to meet the needs of conference organizers and attendees. These requirements include providing a convenient and easy-to-use interface, allowing attendees to access the conference agenda, collecting and analyzing valuable data for conference organizers, and facilitating information sharing, engagement, and interactivity.

Finally, the Non-Functional Requirements of the application were addressed, which describe the attributes of the application such as interface, performance, implementation, and scalability. By defining both functional and non-functional requirements, we aim to ensure that the application meets the expectations and needs of its users and provides a seamless and enjoyable conference experience.

Overall, the analysis and design chapter provides a solid foundation for the development of the mobile application. By defining the requirements and architecture of the application, it is established a clear understanding of what the application should do, and how it should be built. In the next chapter, we will present the implementation of the application and discuss how it meets the requirements and goals established in this chapter.

Chapter 5

Implementation

This chapter delves into the extensive and intricate process of the implementation phase of the project, casting light on various features and conceptual elements that are crucial for the optimal functionality of the application, all viewed through the lens of a software engineer. This section not only elaborates on the features that were contemplated during the Design phase but also brings forth several essential concepts, integral for the efficacious operation of the app. These concepts, while fundamental, extend beyond mere functionality and are pivotal in the overall development lifecycle of mobile applications, given their crucial role in shaping the user experience and the structural integrity of the app.

The illustration of the navigation diagram as depicted in Figure 5.1 serves as a visual representation of the elaborate flow between the principal views of the app. This illustration is instrumental in demonstrating the inherent interactions and interconnectedness among the different sections of the application, providing a coherent and detailed perspective on the user journey and interaction model within the app. It acts as a blueprint, rendering a comprehensive overview of the various elements and their symbiotic relationships, allowing for a deeper understanding of the app's structural and functional dynamics.

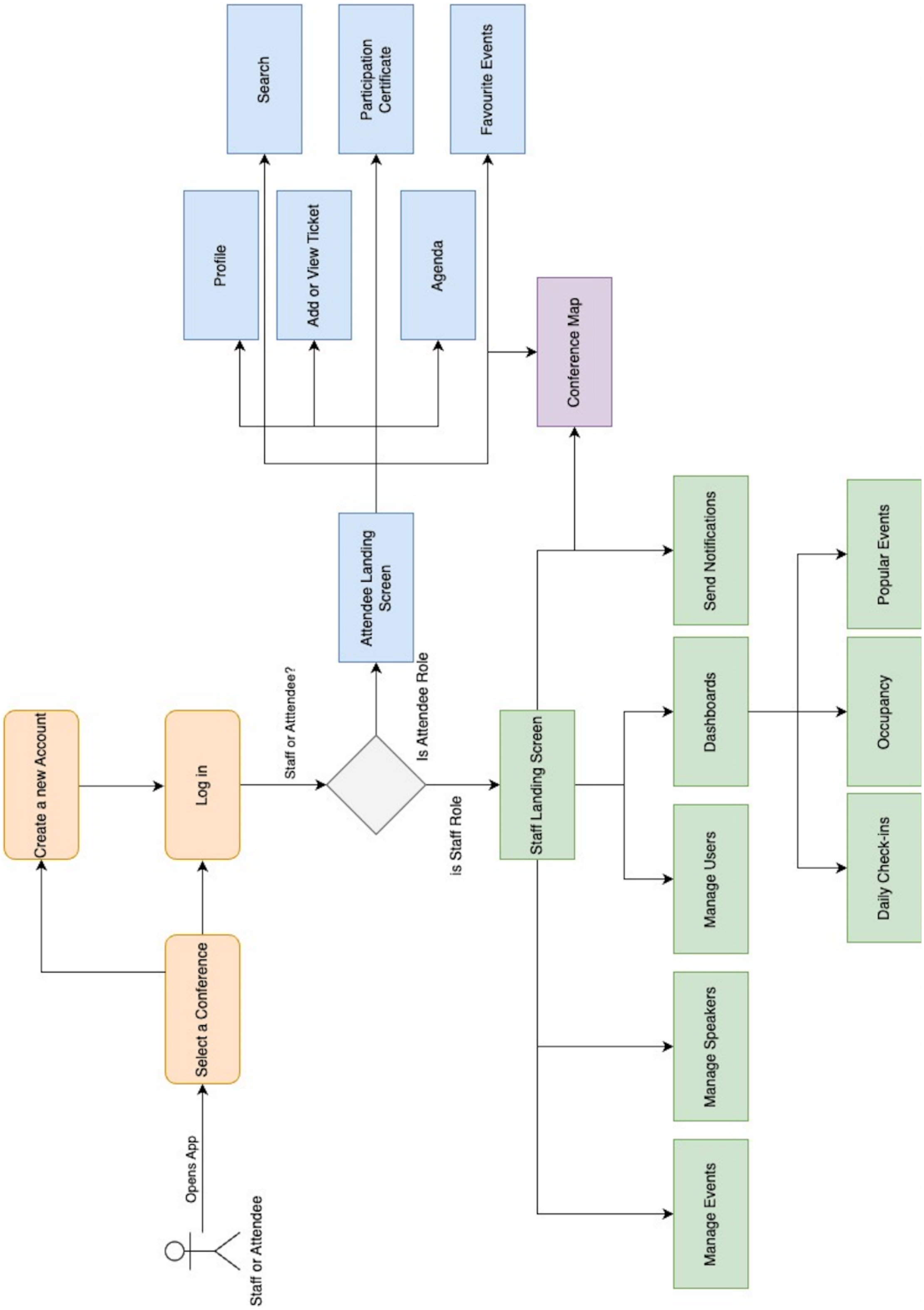


Figure 5.1: Flow Diagram.

5.1 Architecture

As stated in the previous chapter the proposed solution, tailored to manage and visualize conference attendance data, is multifaceted and consists of a Flutter mobile application and a limited-functionality Spring Boot backend application. The architecture sees Firebase Cloud Firestore as its central repository, ensuring seamless real-time data management, storage, and synchronization.

The system operates primarily through the Flutter application, which, apart from specific functionalities, acts as a self-contained unit, serving as the front and backend for most features. The Spring Boot application is specialized, focusing solely on participant certification, creation of conferences, and staff account management. For these select functionalities, the Flutter application resumes its role as the user interface, interacting with the Spring Boot application through RESTful APIs to ensure coherent and reliable data flow.

The mobile application is the core of the system, built using the Flutter framework, providing users with intuitive and real-time insights into conference attendance. It embraces modularity and scalability in its architectural structure. The user interface is interactive and user-centered, employing a variety of custom widgets and components to visualize attendance data efficiently and engagingly. The application employs extensive state management techniques, using StatefulWidget for managing mutable states, thus ensuring a responsive and reliable user interface.

5.1.1 Entity Relationship Diagram

In figure 5.2 the Entity Relationship Diagram (ERD) demonstrates how the scientific application entities (data objects) relate to one another and how (and what) data is stored in the Database.

The two participants mentioned in the Value-Analysis chapter, that constitute the value chain (the attendee and the staff), are represented by **User** with different values for *role*.

To use the app it is required an access account represented by Authentication in conjunction with the User entities. Users become staff by creating an account using the backend application and attendees by using the mobile app. Structuring the system this way allows for another layer of security, as only the hidden backend service can create accounts that may access and edit sensitive data.

After the diagram, each entity's role is explained in the system.

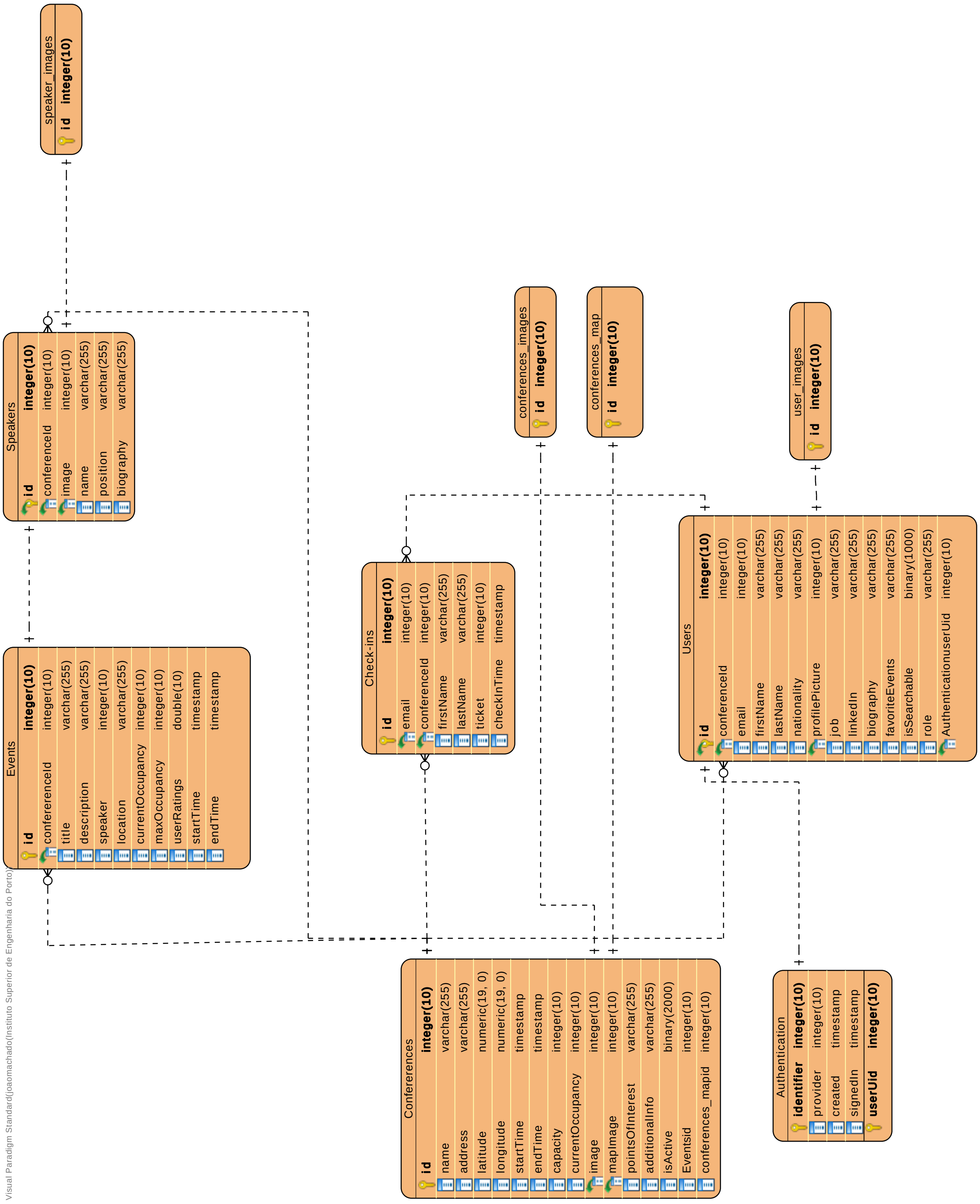


Figure 5.2: Entity Relationship Diagram.

- **Conferences**

  The Conference entity within the application is designed to comprehensively represent a scientific conference, offering users a detailed understanding of the conference's key attributes and information.

  Each conference is uniquely identified by an id, ensuring efficient database operations and distinguishing conferences from one another. The name field contains the title of the conference, providing a concise identification of the conference.

  The address attribute stores the physical location of the conference venue, which can include venue names or complete addresses. Geographical coordinates, represented by latitude and longitude, enable mapping and navigation to help attendees find the venue.

  Precise start and end times are captured by the startTime and endTime fields, aiding users in planning their participation and scheduling conference events. The capacity attribute specifies the maximum number of attendees the venue can accommodate.

  Real-time information on the number of registered attendees is available through the currentOccupancy field, offering insights into the conference's attendance status.

  Visual elements are also incorporated, with the image serving as a recognizable representation of the conference, and mapImage providing a layout of the venue for navigation.

  A list of pointsOfInterest highlights notable locations, exhibitor booths, rest areas, and other points of interest within or around the conference venue, enhancing the attendee experience.

  The additionalInfo attribute accommodates supplementary information, including announcements, special instructions, or essential notes.

  Finally, the isActive indicator informs whether the conference is currently active or not, and facilitates the filtering and display of active conferences to users.

- **Authentication**

  Firebase's Authentication entity uniquely identifies users by their email addresses and userUids. The email is utilized for authentication purposes and serves as a unique identifier. The userUid, on the other hand, is automatically generated by the database and plays a key role in creating the User entity. This establishes a link between the Authentication entity and the User entity without requiring the use of personal data, such as the email address.

  The Authentication entity also stores vital audit information, including the provider (indicating the type of authentication used), created (indicating when the account was created), signedIn (indicating the timestamp of the last login), and the automatically generated userUid.

  Firebase places a strong emphasis on password security by employing advanced cryptographic techniques. When a user registers or logs in with their email and password, Firebase utilizes a cryptographic hash function to transform the provided plain-text password into an irreversible and unique string of characters known as a password hash. This password hash is securely stored within Firebase's authentication database, ensuring robust security for user credentials.

- **Users**

  The User entity represents the diverse community of conference attendees and organizers who engage with the scientific conferences facilitated by the mobile application. It encompasses a wide range of attributes that collectively define and personalize each user's experience.

  Each user is uniquely identified by their id, ensuring the precise management and differentiation of user accounts within the application. This unique identifier is crucial for tracking user-specific interactions, preferences, and activities throughout the conference.

  The conferenceId attribute establishes a direct association between a user and a specific conference, enabling attendees to actively participate in and contribute to the selected conference. This linkage ensures user interactions, such as registration and check-ins, are contextually relevant and aligned with the chosen conference.

  User-specific information, including firstName and lastName, is captured to provide a personalized and friendly user experience. These attributes enable users to identify themselves and interact with others in a more personal and approachable manner.

  The nationality attribute records each user's nationality, offering insights into the diversity of attendees participating in scientific conferences. This information fosters a sense of inclusivity and enables users to connect with others from similar backgrounds or interests.

  The role attribute defines the user's position or status within the conference, whether they are an attendee or organizer. This classification helps tailor the user's experience by granting appropriate privileges and access to specific features and functionalities.

  The profilePicture attribute allows users to customize their profiles with a visual representation, such as a profile picture. This image adds a personal touch to user profiles and makes it easier for attendees to recognize and connect with each other.

  Contact information, including email, is collected to facilitate communication and networking among users. It serves as a primary point of contact for conference-related updates and interactions.

  Users can specify their job and LinkedIn details, providing insights into their professional backgrounds and expertise. This information supports networking opportunities and enables users to connect with others who share similar career interests.

  The biography field allows users to showcase their backgrounds, interests, and achievements. Attendees can use this space to introduce themselves, share their research focus, or provide additional context about their participation in the conference.

  The isSearchable attribute determines whether a user's profile is discoverable by others within the application. Users can choose to make their profiles searchable, fostering connections and interactions with fellow attendees.

Lastly, the favoriteEventIds attribute maintains a record of the events that a user has marked as favorites. This feature enhances the user experience by allowing attendees to curate their schedules and easily access information about their preferred sessions.

- **Events**

  The Event entity encapsulates essential information about a specific event within a scientific conference. It serves as a comprehensive representation of individual conference sessions, workshops, talks, or any scheduled activities.

  Each event is uniquely identified by an id, which ensures effective data management and differentiation between events. Additionally, the conferenceId attribute establishes a connection to the parent conference, enabling the organization of events within their respective conferences.

  The startTime and endTime fields pinpoint the exact timing of the event, offering users a clear understanding of when the event will occur. The title attribute provides a concise, descriptive name for the event, aiding attendees in identifying and selecting sessions of interest.

  A detailed description field provides additional context about the event, including its content, objectives, and any relevant information. This information assists attendees in making informed decisions regarding event attendance.

  Events often feature guest speakers, and the speaker attribute links the event to a specific speaker entity, allowing users to explore the speaker's background and expertise.

  The location field specifies where the event will take place within the conference venue, guiding attendees to the correct venue area.

  Real-time data on the number of attendees currently registered for the event is available through the currentOccupancy attribute, while the maxOccupancy field indicates the event's maximum capacity, ensuring attendees have information about space availability.

  Furthermore, a userRatings map provides attendees with the ability to submit ratings for events, fostering user engagement and helping other attendees make informed event selections based on peer feedback.

- **Speakers**

  The Speaker entity plays a vital role in providing attendees with valuable insights into the individuals delivering presentations and talks at scientific conferences. It serves as a comprehensive representation of the conference's guest speakers.

  Each speaker is uniquely identified by an id, ensuring effective management and distinction between speakers. Additionally, the conferenceId attribute establishes a direct connection between the speaker and the conference to which they are associated, allowing for the organization and categorization of speakers by conference.

  A prominent feature of the Speaker entity is the image attribute, which allows for the inclusion of the speaker's visual representation, such as a profile picture. This visual element aids attendees in recognizing and connecting with speakers on a personal level.

Information about the speaker's professional background is captured in the position attribute, providing context about their expertise and qualifications. Attendees can gain a deeper understanding of the speaker's area of specialization and the relevance of their contributions to the conference.

The biography field offers a comprehensive overview of the speaker's background, accomplishments, and contributions. Attendees can access this information to learn more about the speaker's journey, research interests, and past achievements.

- **Check-Ins**

  Within the application, the Check-Ins entity plays a pivotal role in tracking and recording attendee check-ins during a particular conference. It serves as a comprehensive repository, storing essential data to ensure both data integrity and facilitate auditing processes.

  Each check-in event is uniquely identified within the system, ensuring precise tracking and categorization of attendee interactions within the specified conference. This unique identifier is linked to two crucial pieces of information: the user's email and the conference ID. This linkage guarantees the consistency and reliability of the data, making it valuable for auditing purposes.

  In addition to user identification, the Check-Ins entity also captures attendee-specific details, including their firstName and lastName. These particulars are documented to support auditing efforts and provide a complete record of attendee interactions within the conference.

  Furthermore, the entity records the specific ticket associated with each check-in, contributing to the auditing process by maintaining a trail of attendee participation. This information ensures that conference organizers have access to a comprehensive dataset that assists in the verification of attendee access and engagement.

  The checkinTime attribute is another integral component of the Check-Ins entity. It chronicles the precise timing of attendee check-ins, offering real-time insights into attendee engagement patterns. This temporal data is instrumental in powering the dashboard functionalities, enabling organizers to monitor and analyze attendee interactions throughout the conference.

- **ConferencesImages**

  The ConferencesImages entity serves as a storage database dedicated to housing conference images or logos. It's uniquely identified by the conferenceId, ensuring seamless retrieval and association of visual assets with specific scientific conferences. This entity streamlines the process of enhancing the visual representation and brand recognition of each conference within the application.

- **ConferencesMap**

  The ConferencesMap entity acts as a storage repository for conference maps, with each map associated with a specific conference through its conferenceId. This entity simplifies the access and display of conference layouts and spatial information within the application, aiding attendees in navigating the event venue effectively.

- **SpeakerImages**

The SpeakerImage entity serves as a dedicated storage resource for speaker images within the application. Each image is uniquely associated with a speaker through their corresponding speakerId, facilitating the seamless integration of speaker visuals into the user interface for an enhanced user experience.

- **UserIamges**

  Similar to the SpeakersImages entity, this entity is the dedicated storage resource for user images within the application. Each image is uniquely associated with a user through their corresponding userId.

5.1.2 Backend Class Diagram

The backend class diagram, as illustrated in Figure 5.3, provides a comprehensive overview of the backend architecture, listing all its essential classes and their relationships. The backend application is developed using Spring Boot, a Java-based framework that adheres to the Model-View-Controller (MVC) architectural pattern. This section offers an insightful exploration of how the backend components interact and function within the application.

The backend application is initiated by invoking the main method on the ConferenceApplication class. During this initialization phase, the FirebaseConfig class plays a crucial role. The initialize method within this class is automatically triggered, setting up a Firebase database instance that remains accessible throughout the application's lifecycle. This strategic initialization ensures seamless data management and interaction with Firebase services.

The backend system employs two primary controllers: ConferenceController and UserController. These controllers serve as key components responsible for handling incoming HTTP requests and facilitating the communication between the client and the server. They expose a wide range of Representational State Transfer (REST) endpoints, each designed to fulfill specific functionalities.

When a client sends an HTTP request to one of the exposed REST endpoints, the backend system initiates a series of actions. First, the JSON body of the incoming request is transformed into corresponding Data Transfer Objects (DTOs). These DTOs serve as intermediaries, encapsulating the data and ensuring compatibility with the application's internal structures.

The core processing of the HTTP request is delegated to the service classes. These services are responsible for converting the DTOs into the appropriate domain models. The domain models represent the underlying data structures used within the application. Once the conversion is complete, the services orchestrate the persistence of data by interacting with the Firebase database.

The global Firebase database object, established during the initialization phase, plays a pivotal role in data storage and retrieval. It serves as the interface through which the application communicates with Firebase services, ensuring efficient and secure data transactions. The Firebase database object is a cornerstone of the backend architecture, enabling the seamless flow of data between the application and the cloud-based database.

In summary, the backend class diagram provides a visual representation of the intricate relationships and interactions within the backend application. By adhering to the MVC pattern, employing controllers and services, and leveraging the power of Firebase, the backend system achieves robust data processing and management capabilities, contributing to the overall functionality of the ISEP Conference App.

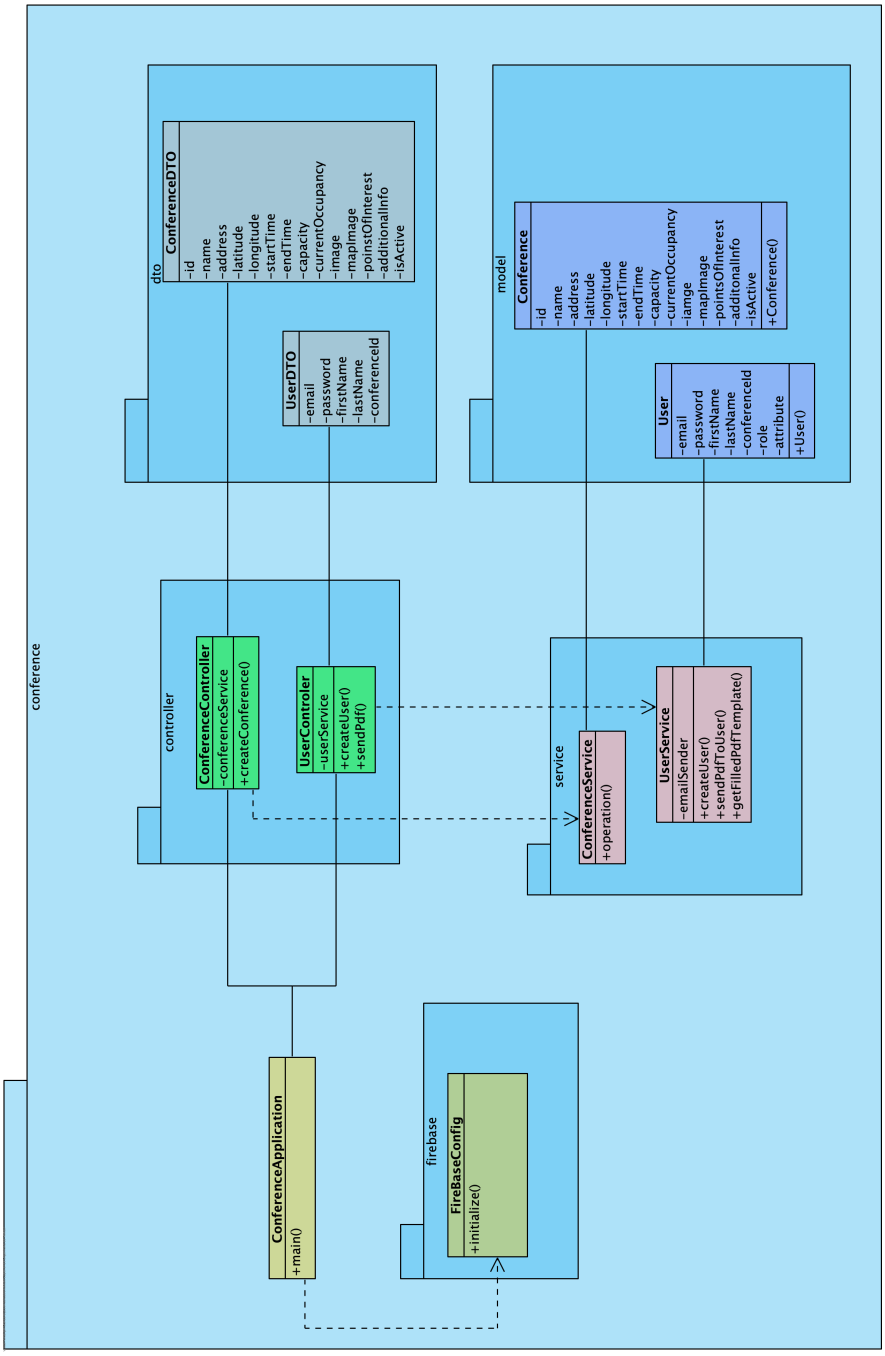


Figure 5.3: Backend Class Diagram.

5.1.3 Mobile Application Class Diagram

The mobile application class diagram, depicted in Figure 5.4, offers a high-level overview of the core classes and their primary interactions within the application. Due to the complexity of the application, this diagram presents a simplified yet insightful representation of the key components and their relationships.

At the heart of the mobile application lies a central main class, positioned at the root of the project's structure. This main class assumes a pivotal role in orchestrating the initialization and launch of various dependencies and modules. Among these dependencies, one of the critical components is the automatically generated Firebase configuration class. This class serves as the foundation for configuring the application to seamlessly function on both Android and iOS devices. Additionally, the auth class is responsible for managing user authentication, and presenting login and account creation screens when necessary. The *selectConference* class handles the presentation of available conferences, enabling users to choose their desired conference.

The main class also establishes references to the theme and notifier classes. These classes are instrumental in ensuring a consistent and user-friendly experience throughout the application. They are responsible for propagating changes related to the chosen theme (dark or light) and the selected language across all screens within the application. This mechanism enhances the overall user experience by allowing users to customize their app settings.

Upon successful login to the application, users are seamlessly redirected to either the staff or attendee landing screens, depending on their assigned role. These landing screens serve as gateways to a myriad of functionalities, each of which is implemented in dedicated classes within their respective directories.

The widgets directory houses notifiers and other classes that encapsulate generic widgets used across multiple parts of the application. These reusable widgets contribute to code modularity and efficiency, streamlining the development process.

Similarly to the backend application, the model directory plays a vital role by containing classes that represent the domain model of the mobile application. These classes define the structure and behavior of core data entities, ensuring data consistency and integrity.

Lastly, the l10n directory encompasses the Application Resource Bundle (ARB) files, which are essential for implementing multi-language support. These ARB files map all the text used within the application to specific languages, facilitating the localization of content and enabling users to interact with the app in their preferred language.

In summary, the mobile application class diagram provides a simplified yet comprehensive view of the application's key components and their interconnections. This architectural design, coupled with careful organization and support for multi-language localization, contributes to the robustness and versatility of the ISEP Conference App.

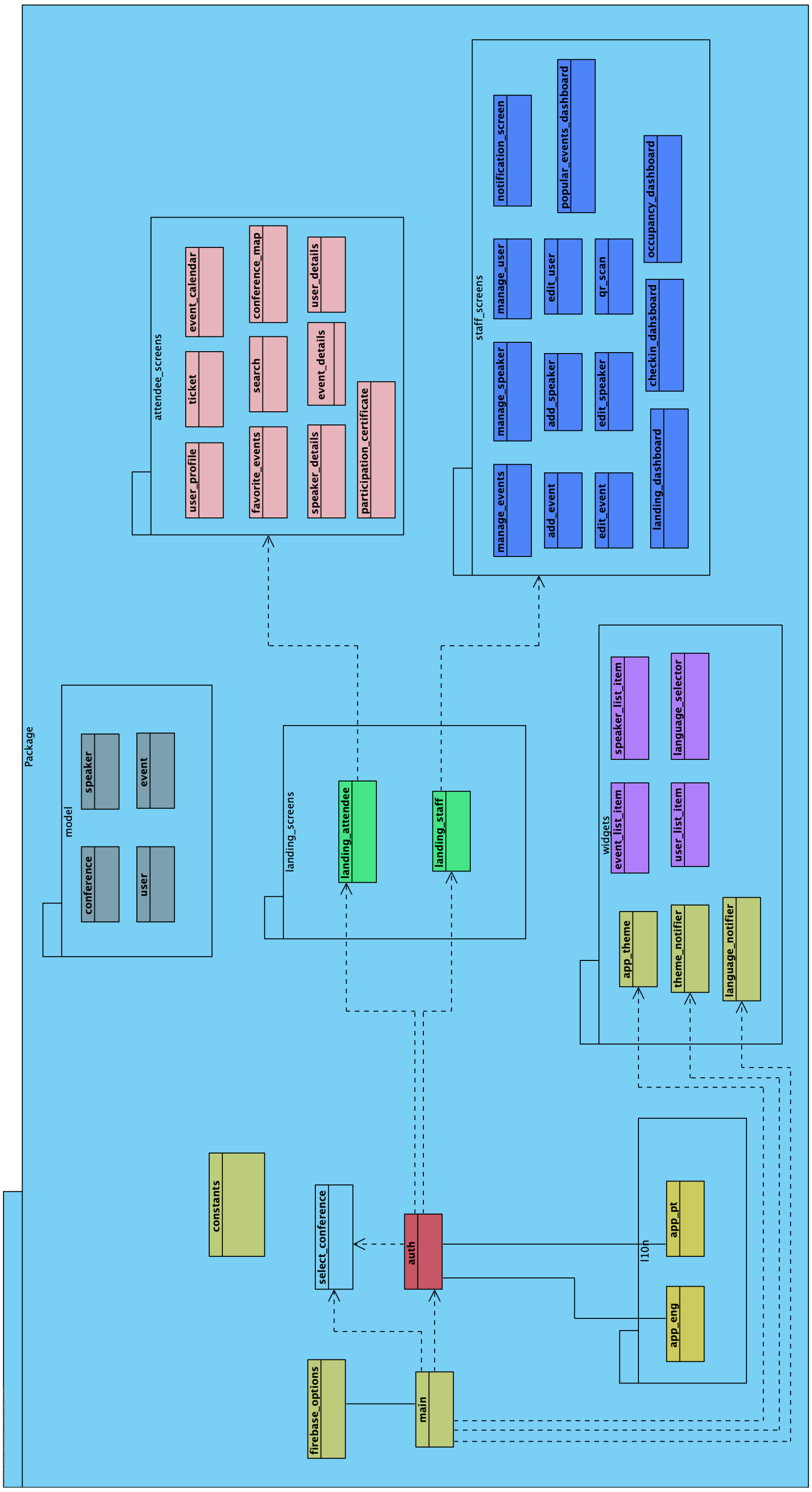


Figure 5.4: Mobile Application Class Diagram.

5.2 Security

In the realm of application development, ensuring user data protection and maintaining system integrity are paramount. A foundational component of the security architecture in the developed mobile application is the integration of Firebase Email and Password Authentication, augmented with secure session storage on the user device.

5.2.1 Login

Firebase Authentication furnishes the application with a state-of-the-art identity solution, employing an integration of Email and Password Authentication methods, elucidated in figures 5.5 and 5.6. This integration serves as a cornerstone in establishing a secure environment, leveraging advanced and proven authentication protocols to securely verify user credentials, ensuring users are genuine and authentic, thus mitigating any risks related to unauthorized access.

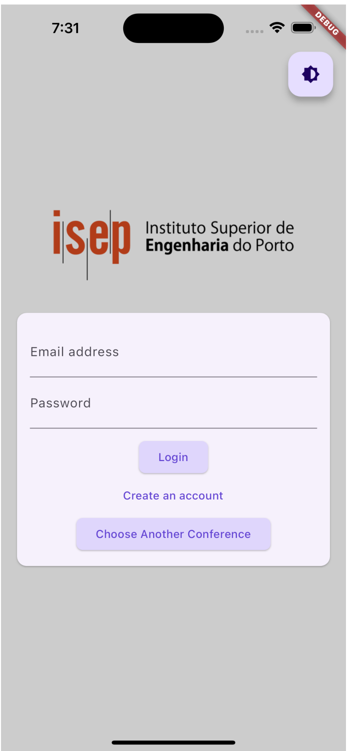
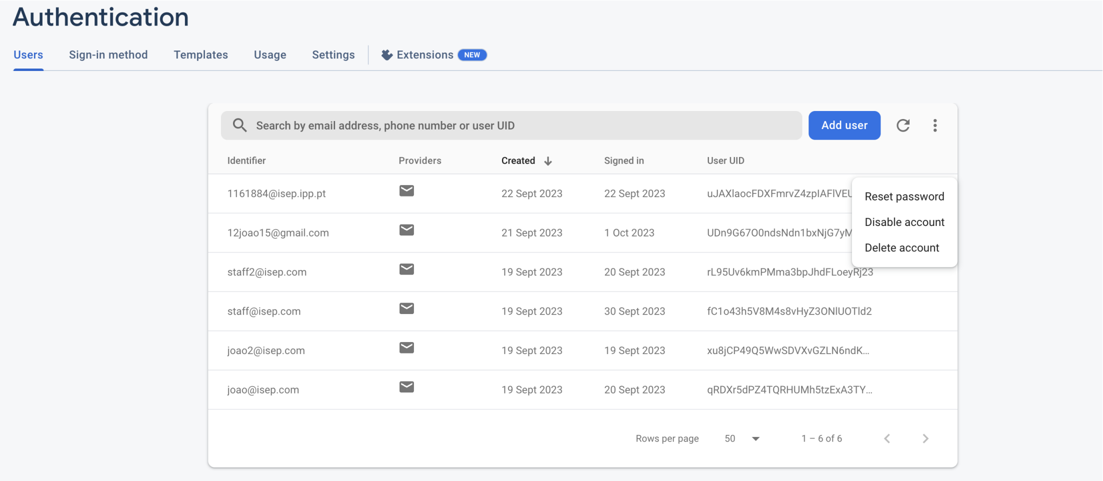

7:31
DEBUG
isep Instituto Superior de Engenharia do Porto
Email address
Password
Login
Create an account
Choose Another Conference

Figure 5.5: Login and Create Account Screen.

The illustrated Firebase Authentication Database in Figure 5.6 and the sequence diagram in Figure 5.7 are a testament to the intricate layers of security and user-friendly interfaces integrated within the application. It demonstrates the secure environment where user authentication processes are streamlined, requiring users to verify their identities using their email addresses and passwords. This methodology fortifies the application's security measures and hinders unauthorized access to sensitive user data and features.



Identifier	Providers	Created ↓	Signed in	User UID
1161884@isep.ipp.pt	✉	22 Sept 2023	22 Sept 2023	uJAXlaocFDXFmrvZ4zpIAFIVEL
12joao15@gmail.com	✉	21 Sept 2023	1 Oct 2023	UDn9G67O0ndsNdn1bxNjG7yM
staff2@isep.com	✉	19 Sept 2023	20 Sept 2023	rL95Uv6kmPMma3bpJhdFLoeyRj23
staff@isep.com	✉	19 Sept 2023	30 Sept 2023	fC1o43h5V8M4s8vHyZ3ONlUOTId2
joao2@isep.com	✉	19 Sept 2023	19 Sept 2023	xu8jCP49Q5WwSDVXvGZLN6ndK...
joao@isep.com	✉	19 Sept 2023	20 Sept 2023	qRDXr5dPZ4TQRHUMh5tzExA3TY...

Figure 5.6: Firebase Authentication Database.

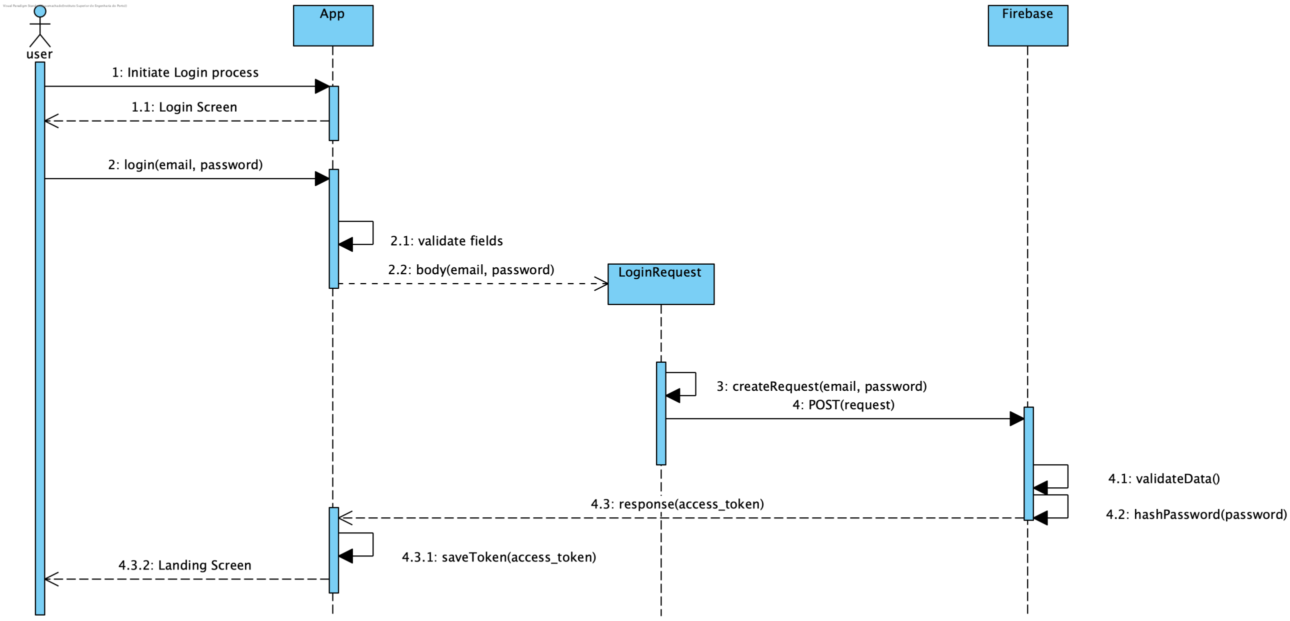

user
App
Firebase
1: Initiate Login process
1.1: Login Screen
2: login(email, password)
2.1: validate fields
2.2: body(email, password)
LoginRequest
3: createRequest(email, password)
4: POST(request)
4.1: validateData()
4.2: hashPassword(password)
4.3: response(access_token)
4.3.1: saveToken(access_token)
4.3.2: Landing Screen

Figure 5.7: Firebase Authentication Sequence Diagram.

Furthermore, the application ensures user passwords are meticulously encrypted and securely stashed, unavailable for viewing by administrators, hence, accentuating user privacy and data security. The features enabling users to reset passwords, and administrators the ability to disable or delete accounts, empower users and administrators with elevated control and quick resolution to any security concerns or account-related discrepancies. These capabilities, coupled with the use of a separate and secure database for authentication, heighten the application's security standards, providing an additional layer of protection against potential data breaches and unauthorized access.

Post-authentication, the application emphasizes user-centric experiences by securely maintaining the user's session on the device, allowing continuous interaction without recurrent logins. The session information, encrypted and securely stored locally, remains immune to unauthorized access or potential security vulnerabilities, maintaining its confidentiality and integrity. This secure and innovative storage mechanism serves to prevent any manipulation or interception of the session data, ensuring a seamless and secure user experience.

This meticulous balance between strengthening security and optimizing user experience is pivotal. The secure session management acts as a bulwark against unauthorized access and potential security breaches, while concurrently eliminating the need for repetitive user authentication, thereby cultivating a seamless and user-centric interaction paradigm.

The integration of such robust Firebase Email and Password Authentication mechanisms, in conjunction with secure session management, is a confluence forming a formidable security infrastructure for the application. This intricate security matrix not only verifies user identities securely but also preserves the confidentiality and integrity of user sessions. The relentless commitment to security in this application guarantees users a secure and user-friendly environment, fostering trust and reliability, and assuaging the risks related to unauthorized access and data breaches. The synergy of these security mechanisms propels the application into a realm of enhanced trust and user satisfaction, providing a harmonious blend of security and user-friendliness.

5.2.2 Data protection

Firebase Database Rules play a critical role in establishing a secure and structured data access framework for applications. They allow developers to create conditions under which data read and write operations are permitted. These rules are instrumental in securing data stored in Firebase, ensuring the integrity and confidentiality of sensitive information while still allowing access to public data.

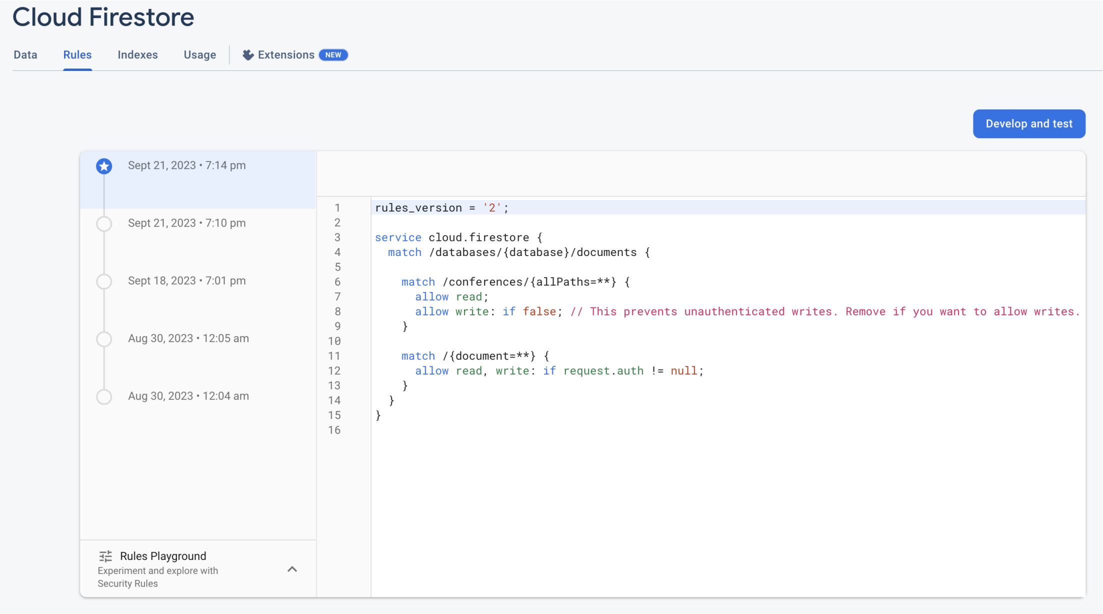
Within the application, the rules are meticulously defined to delineate between public and private data. Public data, such as conference names and images, are readily accessible, providing essential information required for the initial selection screen. This accessibility ensures that users can view necessary details without authentication, offering a seamless user experience.

Conversely, private data are stringently protected, and accessible only to authenticated users. This segregation is pivotal to maintaining user data privacy and application data integrity. The rules are articulated to ensure that unauthorized access to sensitive data is unequivocally restricted, thereby mitigating risks associated with data breaches and unauthorized data manipulation.

The rules (5.8,5.9) are designed to work harmoniously with Firebase Authentication, reinforcing the secure environment established through email and password authentication. Once users are authenticated, they are granted permission to access and interact with private data relevant to them, ensuring a secure and personalized user experience.

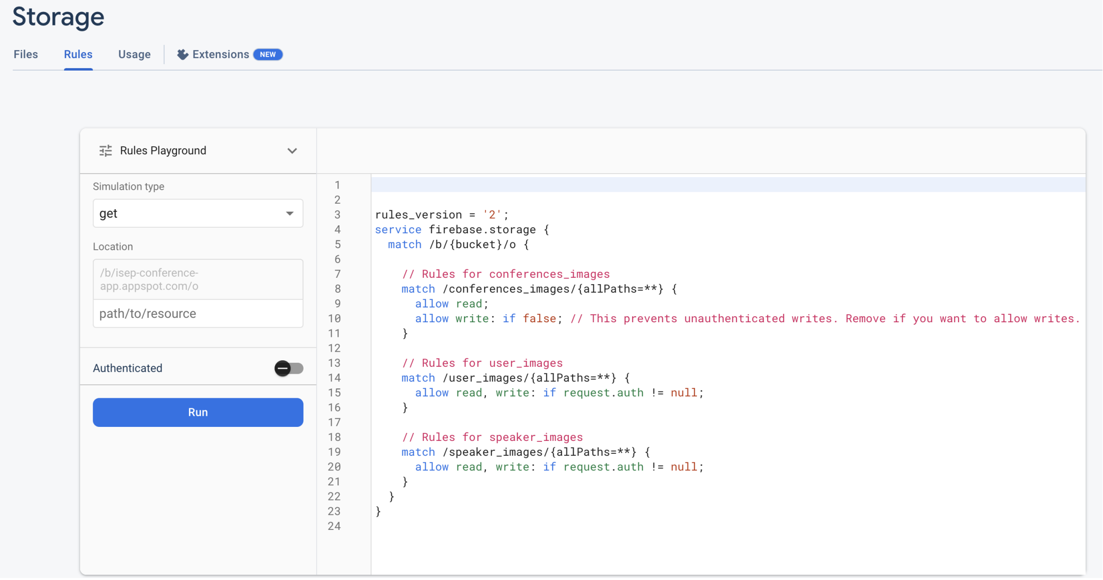
The meticulous crafting of Firebase Database Rules is driven by the need to balance security with accessibility. By allowing public access to non-sensitive data, the application maintains an open and user-friendly interface. At the same time, the strict access controls on private data ensure that the user's information remains secure and the application's data structure is upheld.

Firebase Database Rules are crucial in the establishment of a secure and coherent data interaction model within the application. They ensure that data access is regulated and secured, protecting sensitive information while maintaining the availability of public data. The integration of these rules with Firebase Authentication amplifies the application's security



```
rules_version = '2';

service cloud.firestore {
  match /databases/{database}/documents {

    match /conferences/{allPaths=**} {
      allow read;
      allow write: if false; // This prevents unauthenticated writes. Remove if you want to allow writes.
    }

    match /{document=**} {
      allow read, write: if request.auth != null;
    }
  }
}
```

Figure 5.8: Firebase Database Rules.



Simulation type: get

Location: /b/isep-conference-app.appspot.com/o path/to/resource

Authenticated

```


rules_version = '2';
service firebase.storage {
  match /b/{bucket}/o {

    // Rules for conferences_images
    match /conferences_images/{allPaths=**} {
      allow read;
      allow write: if false; // This prevents unauthenticated writes. Remove if you want to allow writes.
    }

    // Rules for user_images
    match /user_images/{allPaths=**} {
      allow read, write: if request.auth != null;
    }

    // Rules for speaker_images
    match /speaker_images/{allPaths=**} {
      allow read, write: if request.auth != null;
    }
  }
}
```

Figure 5.9: Image Database Rules.

framework, providing a structured, secure, and user-friendly environment that fosters trust and reliability in the application's data interactions.

By employing such detailed and stringent data access rules, the application ensures adherence to data protection best practices, paving the way for a robust and resilient application architecture.

5.2.3 Input Validation and Optimization

In the development of our application, critical emphasis has been laid on Input Validation and Optimization to ensure seamless user interaction and to fortify the application's security.

The sophisticated design involves implementing context-appropriate keyboards and precise input validations, ensuring the correctness and reliability of the user input data.

For instance, in the email field of the form, the application mandates the inclusion of the "@" character, a fundamental criterion for an email address. The employed code snippet 5.1 delineates the implementation of this validation:

```
TextFormField(
  decoration: InputDecoration(
    labelText: _isLoggin ? 'Email address' : 'Email address *',
  ),
  keyboardType: TextInputType.emailAddress,
  autocorrect: false,
  textCapitalization: TextCapitalization.none,
  validator: (value) {
    if (value == null || value.trim().isEmpty || !value.contains('@')) {
      return 'Invalid email address';
    }
    return null;
  },
  onSaved: (newValue) => _enteredEmail = newValue!,
),
```

Listing 5.1: Input Validation for email field.

This snippet also demonstrates the optimization of the keyboard context, where autocorrect is set to false, preventing any unnecessary and potentially confusing corrections while the user is entering their email address.

Moreover, the application employs distinct keyboard layouts adaptive to the context of each screen, enhancing the user interaction experience. For example, when a field accepts only numerical input, the displayed keyboard is numeric, and similarly, autocorrect is disabled for password fields, preventing inadvertent corrections and enhancing user experience and interaction efficiency.

The code snippet 5.2 is an illustration from the code where the password field is both obscured and validated for a minimum length of 6 characters to ensure robustness:

```
TextFormField(
  decoration: InputDecoration(
    labelText: _isLoggin ? 'Password' : 'Password *',
  ),
  obscureText: true,
  validator: (value) {
    if (value == null || value.trim().length < 6) {
      return 'Invalid password, needs to be 6 characters long';
    }
    return null;
  },
  onSaved: (newValue) => _enteredPassword = newValue!,
),
```

Listing 5.2: Hiding text on password field.

These meticulously implemented input validations and optimized user interfaces are pivotal in maintaining the application's integrity, reliability, and user-friendliness. They not only assure the accuracy and validity of the user input data but also significantly contribute to the application's overall user experience and security posture, making interactions intuitive and secure.

5.3 Roles

In a multi-functional application environment, it is imperative to implement a refined navigation system that caters to the diverse needs of different user roles. The application developed encompasses two distinct roles: Staff and Attendee. The seamless navigation experience and the availability of functionalities are meticulously crafted based on these roles, ensuring an intuitive and role-appropriate user interface.

Upon successful authentication, the system identifies the role of the user, which fundamentally influences the subsequent user experience. The role-based direction ensures that users are guided to a landing screen tailored to their roles, staff, or attendees, presenting functionalities and information pertinent to their responsibilities and needs.

- Staff Landing Screen (5.11): The staff landing screen is imbued with functionalities integral to managing and overseeing conference proceedings. It serves as a hub where staff members can access tools and information essential for maintaining the order and flow of the conference, addressing the multifarious tasks that staff members are anticipated to execute.

- Attendee Landing Screen (5.10): In contrast, the attendee landing screen is streamlined to offer a user-friendly interface, focusing on providing attendees with essential information and functionalities, such as conference schedules, speakers, and real-time updates. It is constructed to offer a seamless and enriching experience to the attendees, allowing them to navigate through the conference with ease and acquire pertinent information promptly.

The divergence in functionalities and information presented on the landing screens is pivotal in offering a personalized and role-centric user experience. Staff members are equipped with tools and insights necessary for operational excellence, while attendees enjoy an enriching and informative experience, facilitating their engagement and participation in the conference.

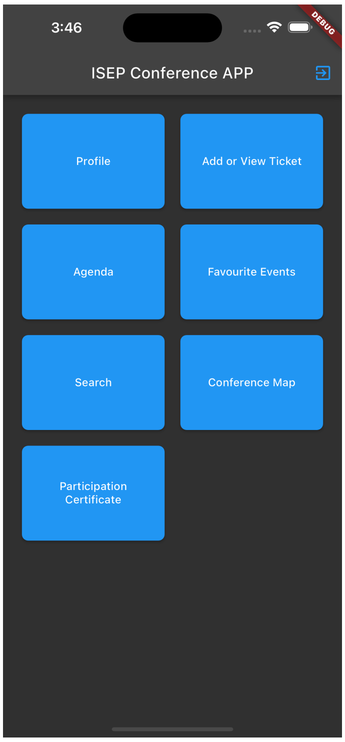


Figure 5.10: Attendee Landing Screen on iPhone.

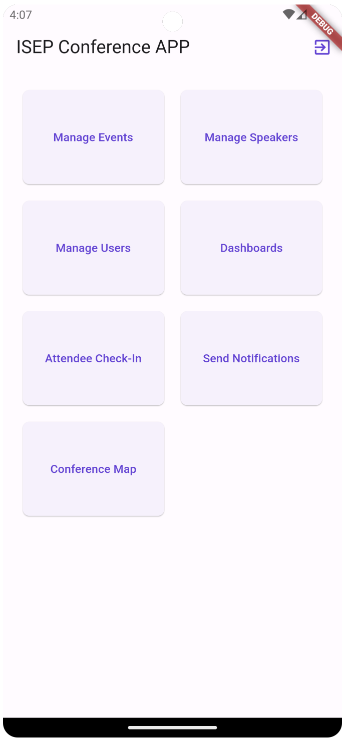


Figure 5.11: Staff Landing Screen on Android.

The role-based navigation is intertwined with an integrated security mechanism, ensuring that access to functionalities and sensitive information is stringently regulated based on user roles. This mechanism is paramount in maintaining the security and integrity of the application, averting unauthorized access and potential discrepancies in user experience.

The dynamic role-based interface ensures the relevancy and appropriateness of the content and functionalities displayed, enhancing user interaction and satisfaction. By adopting this approach, the application accentuates its user-centric design philosophy, catering to the distinct needs and expectations of staff and attendees meticulously.

Role-based access and Navigation are essential components in the architecture of the application, providing a structured, secure, and intuitive user environment. The intricate alignment of roles with corresponding landing screens ensures the delivery of role-specific functionalities and information, enriching the user experience while upholding the application's security standards. This role-based design philosophy is pivotal in fostering an environment of operational coherence and user satisfaction, contributing significantly to the overall success and reliability of the application.

5.4 Storage

In the domain of mobile application development, meticulous examination and strategic implementation of storage utilization are indispensable. The overarching objective is to furnish users with a seamless and expedient experience, devoid of recurrent password prompts or conspicuous loading durations, aligning with contemporary user expectations.

The strategic allocation and utilization of storage are pivotal in achieving an optimized user experience. This is realized by securely conserving authentication data within encrypted files and caching resources locally to obviate the necessity of incessant remote fetches. These practices are elemental in mitigating superfluous network requests and reducing loading intervals, thereby providing a fluid interaction paradigm for the users.

The developed application utilizes four types of storage:

- Local Storage — JSON file-based storage. Allows the applications to create multiple JSON files, each one acting as a database for key-value pairs. These files are created and stored locally on the user's device.
- Firebase Cloud Storage — Versatile online object storage from Google. Stores user-generated content. Integrates customizable access rules and is accessible using the Firebase SDK. GigWho utilizes this storage for image-related data only.
- Secure Storage — Secure Key-value storage. Uses a Keychain for iOS and, Android, it encrypts data using AES, keeping the AES secret key encrypted with RSA and storing the RSA key in Keystore.
- In-Memory Storage 5.3— Extra fast key-value in-memory storage. Data is stored using the device's main memory. Differs from the other types of storage in that if the application is closed, the memory is released and data is lost. This is not a reliable persistence mechanism and is only used for data that is frequently accessed within a specific context.

```
return GestureDetector(
                onTap: () async {
                  final prefs = await SharedPreferences.getInstance();
                  prefs.setString('selectedConferenceId', conference.id);
                  widget.onConferenceSelected();
                },
```

Listing 5.3: In-Memory Storage of the Conference ID for easy access during app utilization.

The following diagram 5.12 provides a clear view of the process for uploading an image using Firebase Cloud Storage.

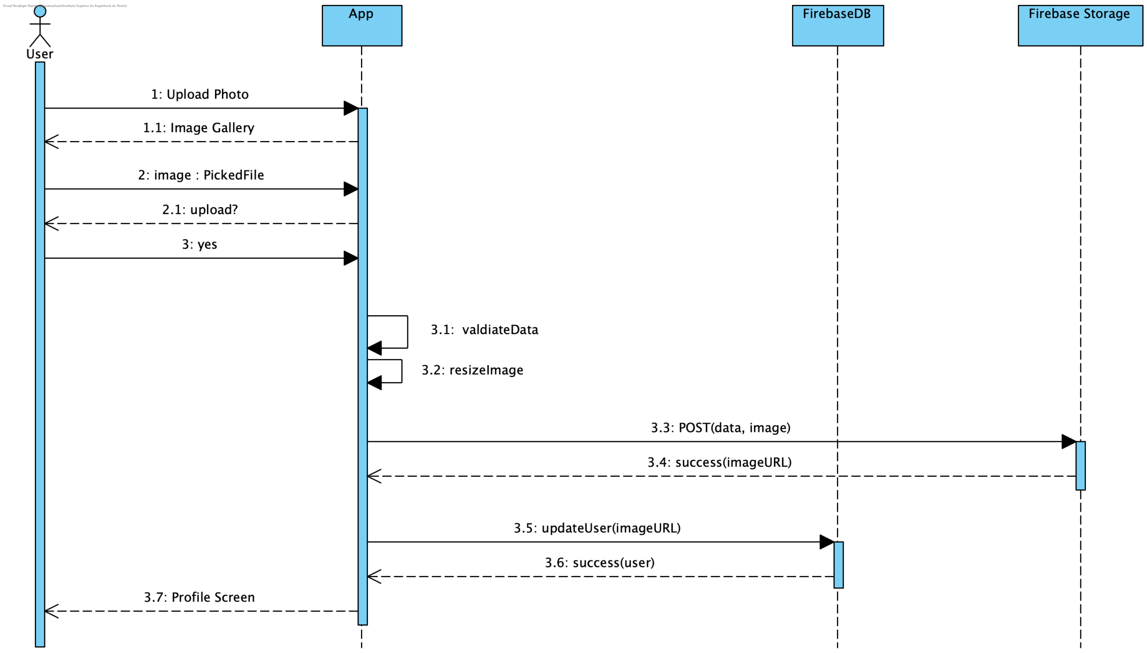


Figure 5.12: Firebase Cloud Storage Sequence Diagram.

5.4.1 Management and Utilization of Media Assets via Firebase Cloud Storage (FCS)

This subsection elucidates the procedure of importing assets from the local gallery, subsequently uploading them to Firebase Cloud Storage (FCS), and retaining them on the application server to facilitate universal access by every user of the app. The assets are initially imported from the user's smartphone gallery to the app and then relayed to a bucket within Firebase Cloud Storage. A bucket in this context refers to a virtual container, furnished by Firebase, which acts as a repository where developers can store various objects pertinent to their application.

The process of depositing an image asset into a bucket initiates the creation of a condensed version, modified to specific dimensions. The incorporation of Image and Image Picker packages aids in streamlining the import process, harmonizing it with both Android and iOS galleries. The initiation of the upload activity by users prompts the opening of the filesystem for file selection. The raw data of the file is subsequently imported and utilized to generate its placeholder equivalent.

In this context, the ImagePicker API facilitates file selection from the local gallery, while PickedFile acts as a simplified, universal wrapper of File (from dart:io) that encapsulates raw data. In the following excerpt (5.4), the path allocated for the selected file is transmitted to an actual File object, which is then conveyed to a widget responsible for rendering the resultant image.

```
Future<void> _pickImage() async {
  final pickedImage = await ImagePicker().pickImage(
    source: ImageSource.gallery,
    imageQuality: 50,
    maxWidth: 150,
  );

  if (pickedImage == null) return;

  final pickedImageFile = File(pickedImage.path);
  setState(() {
    _userImageFile = pickedImageFile;
  });
}
```

Listing 5.4: Function to select an image from the gallery.

Once the image is manifested on the device, the user has the autonomy to decide on its upload. If the user opts to proceed with the upload, Firebase Storage furnishes the essential APIs to establish a connection with the remote Firebase Cloud Storage, enabling a secure transmission of the selected file. The culmination of this process is the generation of a URL that directs to the remote location of the image. The next excerpt (5.5) represents a continuation of the preceding one, delineating the procedure of uploading to the remote site:

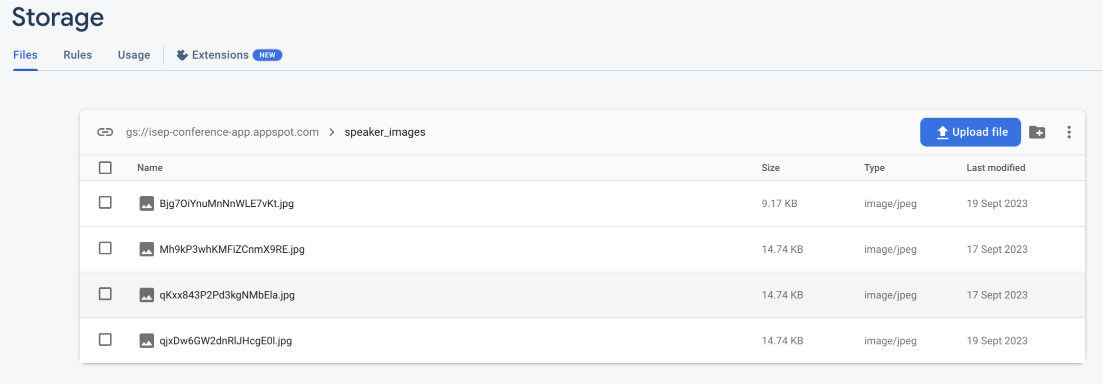
```
Future<String> _uploadImage(File image) async {
  FirebaseStorage storage = FirebaseStorage.instance;
  Reference ref = storage.ref().child('user_images/$_userId.jpg');
  UploadTask uploadTask = ref.putFile(image);
  await uploadTask.whenComplete(() => {});

  if (uploadTask.snapshot.state == TaskState.success) {
    return await ref.getDownloadURL();
  } else {
    return '';
  }
}
```

Listing 5.5: Fuction to upload attendee image and using the userId (unique) as the name of the picture.

Moreover, the approach involves the caching of ImageProvider for Firebase Cloud Storage image objects. This advanced mechanism ensures efficient management and retrieval of images, optimizing the overall performance and user experience of the application by reducing the load times and preserving bandwidth, especially when dealing with high-resolution images. It underlines the application's commitment to delivering a responsive and seamless interaction, allowing users to access and manage media assets with unparalleled ease and convenience.

5.4.2 Local Storage

The application's storage usage schema has been architected with meticulous consideration to facilitate future integrations and modifications. The adopted strategies and implementations are designed to be highly scalable and adaptable, allowing for seamless integrations of new storage solutions or modifications to the existing schema as the application evolves.



Storage

gs://isep-conference-app.appspot.com > speaker_images

□	Name	Size	Type	Last modified
□	Bjg7OiYnuMnNnWLE7vKt.jpg	9.17 KB	image/jpeg	19 Sept 2023
□	Mh9kP3whKMFiZCnmX9RE.jpg	14.74 KB	image/jpeg	17 Sept 2023
□	qKxx843P2Pd3kgNMbEla.jpg	14.74 KB	image/jpeg	17 Sept 2023
□	qjxDw6GW2dnRlJHcgE0l.jpg	14.74 KB	image/jpeg	19 Sept 2023

Figure 5.13: Stored images on Database.

This forethought ensures that the application remains robust and adaptable to emerging requirements and technological advancements.

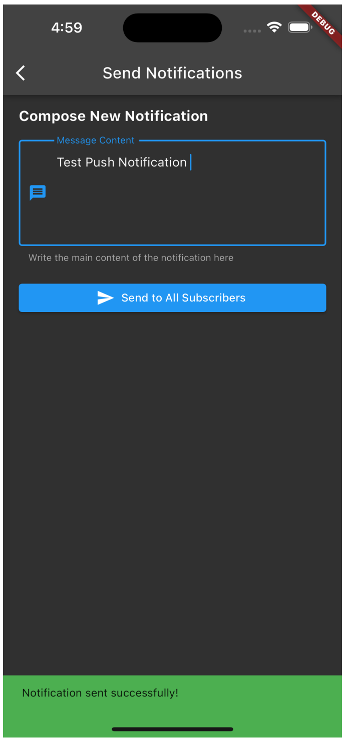
The conscientious evaluation and strategic implementation of storage solutions are quintessential in mobile development, underpinning the delivery of a streamlined and responsive user experience. The techniques adopted within the developed application are oriented towards ensuring secure, efficient, and scalable storage utilization, thereby providing a robust foundation for future enhancements and adaptability to evolving technological landscapes.

5.5 Push Notifications and Firebase Cloud Messaging (FCM)

Push Notifications act as vital conduits between the application and the users, enhancing the user experience by delivering real-time updates, alerts, and important information directly to the user's device. These notifications transcend the boundaries of app usage, reaching users even when the application is not active, thus ensuring that essential information is conveyed promptly and efficiently. This section delves into the incorporation of Push Notifications within the application, facilitated through Firebase Cloud Messaging (FCM), and details its operational dynamics, functionalities, and significance in ensuring seamless and dynamic communication.

FCM is a cloud solution for messages on iOS, Android, and web applications. It provides a reliable and efficient connection between the server and devices that allow server applications to send messages to the client. FCM serves as the backbone for the delivery of push notifications, enabling the transmission of notifications or messages to users, with options for customization, targeting, and analytics.

Within the developed application, Firebase Cloud Messaging operates as the messenger between staff users and the rest of the users. It is explicitly designed to allow staff users the capability to send push notifications (figures: 5.14,5.15), serving as a medium for the dissemination of crucial information, emergency alerts, or important updates related to the conference.



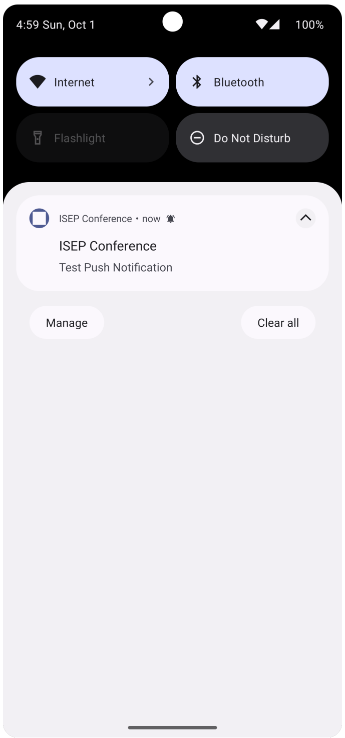
Compose New Notification

Message Content: Test Push Notification

Write the main content of the notification here

Notification sent successfully!

Figure 5.14: Staff Screen to send push notifications on iPhone.



ISEP Conference • now

ISEP Conference
Test Push Notification

Figure 5.15: Attendee on Android receiving a push notification.

This implementation aligns with the application's emphasis on role-based functionalities. Given the supervisory nature of staff roles, equipping them with the ability to send notifications ensures that the information conveyed is authenticated, relevant, and timely. The notifications sent via FCM can reach users irrespective of the app's operational state, ensuring uninterrupted communication flow.

The utilization of FCM is particularly crucial in instances requiring immediate attention or action from the users. In emergency scenarios or when vital updates are necessitated, staff users can promptly broadcast notifications to all users, thereby averting potential lapses in communication and ensuring the safety and awareness of the attendees.

The flexibility of FCM allows the crafting of messages that are concise, clear, and informative, thus maximizing the impact and comprehension of the conveyed information (snippet: 5.6). In addition, FCM's analytics capabilities enable the monitoring of notification effectiveness and user engagement, offering insights that can be leveraged to enhance communication strategies.

```
Future<void> sendNotification() async {
    const String serverToken = Constants.serverToken;
    const String topic = Constants.topic;

    final response = await http.post(
      Uri.parse(Constants.pushNotificationEndpoint),
      headers: {
        'Content-Type': 'application/json',
        'Authorization': 'key=$serverToken',
      },
      body: json.encode({
        'notification': {
          'body': _messageController.text,
          'title': 'ISEP Conference'
        },
        'priority': 'high',
        'data': {
          'click_action': 'FLUTTER_NOTIFICATION_CLICK',
        },
        'to': '/topics/$topic',
      }),
    );

    if (response.statusCode == 200) {
      ScaffoldMessenger.of(context).showSnackBar(
        const SnackBar(
          content: Text('Notification sent successfully!'),
          backgroundColor: Colors.green,
        ),
      );
  }
```

Listing 5.6: Fuction responsible for creating the HTTP request to send push notifications.

The integration of Firebase Cloud Messaging for push notifications significantly elevates the user experience within the application. It empowers users with real-time information, enriching their engagement and interaction with the application. The immediate relay of important updates and alerts ensures that users remain informed and prepared, thus fostering a sense of reliability and responsiveness within the app ecosystem.

Incorporating Firebase Cloud Messaging as the facilitator of push notifications within the application underscores the commitment to delivering a seamless, informed, and dynamic user experience. By enabling staff users to communicate important updates and alerts directly to users' devices, the application ensures the prompt delivery of essential information, thereby enhancing user awareness, safety, and interaction. The strategic implementation of FCM aligns with the overall architectural philosophy of the application, emphasizing the importance of real-time communication, user-centric design, and operational efficiency in the modern app environment.

5.6 User Interface and Experience

In any software solution, especially mobile applications, the emphasis on user interface and experience is paramount, acting as the medium through which users interact with the offered functionalities. The provided application demonstrates a meticulous commitment to delivering an intuitive, user-friendly, and inclusive interface, thereby enhancing overall user satisfaction and interaction quality. Below is a detailed discourse on the pivotal aspects of the application's user interface and experience

5.6.1 Intuitive and User-Friendly Layout

The application under discussion has been designed with a user-centric approach where the emphasis is on intuitive design and user-friendly layout. The layout is simplistic yet effective, aiming to reduce cognitive load and avoid overwhelming the users with unnecessary details. It employs a landing page system where functionalities are meticulously organized and easily accessible. This organization seeks to maintain a clean and uncluttered aesthetic, allowing users to identify and use the application's functionalities with minimal effort. The buttons are large and distinguishable, enhancing ease of interaction and reducing any potential interaction ambiguities.

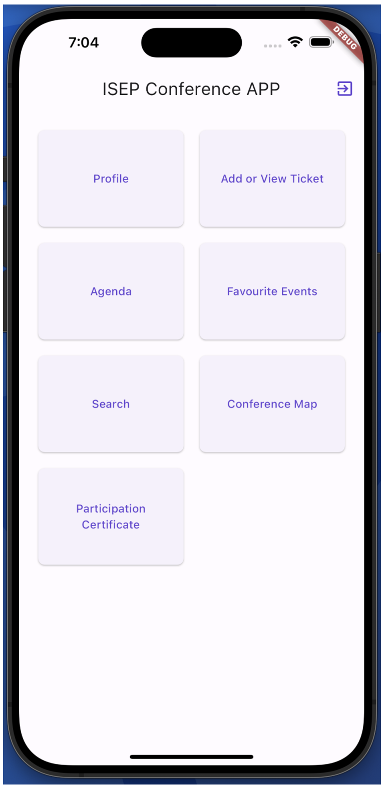


Figure 5.16: Main Attendee Screen on iOS.

Each functionality is displayed in a structured two-column grid view (figure 5.16), acting as a direct gateway to the varied features of the application. The use of large buttons is intentional, aiming to provide a hassle-free navigation experience. This design philosophy is grounded in making the application inclusive and user-friendly, catering to users with varying levels of familiarity with mobile applications.

5.6.2 Simplified Navigation

The navigational structure of the application is meticulously designed to avoid complexity and offer users a seamless and intuitive experience. It is deliberately crafted to be devoid of multi-level menus and intricate structures, avoiding potential user confusion and frustration. Each feature and functionality of the application is accessible with minimal navigation, allowing users to easily revert to the main screen with a single back action.

The code snippet below 5.7 depicts a simplified navigation approach using Dart, which ensures that all functionalities are just a level away. It demonstrates the creation of navigation buttons for each functionality in the attendee role, utilizing a generic approach to button creation.

```
  Widget _buildButton(String text, Widget screen) {
    return ElevatedButton(
      onPressed: () => _navigateToScreen(screen),
      style: AppTheme.elevatedButtonStyle,
      child: Text(
        text,
        overflow: TextOverflow.ellipsis,
        maxLines: 2,
        textAlign: TextAlign.center,
      ),);}}
```

Listing 5.7: Basic function for navigating to another screen received by parameter.

This method, *buildButton*, is a generic approach to creating buttons, which takes the text to be displayed and the screen to navigate to as parameters, thus providing a highly reusable and maintainable solution. This streamlined approach ensures that adding more functionalities or modifying existing ones is straightforward, fostering a flexible and adaptive development environment.

The application employs an AnimatedBuilder in conjunction with a GridView to organize these buttons in a user-friendly manner, providing a smooth visual transition and interaction experience:

```
child: GridView(
    padding: const EdgeInsets.all(24),
    gridDelegate: const SliverGridDelegateWithFixedCrossAxisCount(
        crossAxisCount: 2,
        childAspectRatio: 3 / 2,
        crossAxisSpacing: 20,
        mainAxisSpacing: 20,
    ),
    children: [
        _buildButton('Profile', UserProfile(onPickImage: (pickedImage)
   {})),
        _buildButton('Add or View Ticket', const TicketScreen()),
        _buildButton('Agenda', const EventCalendarScreen()),
        _buildButton('Favourite Events', const FavoritesScreen()),
        _buildButton('Search', const SearchScreen()),
        _buildButton('Conference Map', const ConferenceMapScreen()),
        _buildButton('Participation Certificate',
            const ParticipationCertificateScreen()),
        ],),
```

Listing 5.8: Function that creates navigation buttons for each functionality of the attendee role.

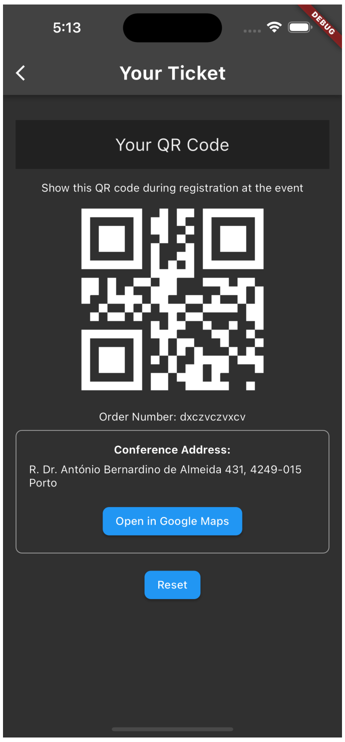
This fluid navigational structure, coupled with the reusability of the developed code, facilitates the quick addition of new features without the need to refactor existing code significantly, thereby reducing developmental efforts and minimizing the risk of introducing errors. It provides a solid foundation for future developments and enhancements while maintaining a clean and user-centric design.

This commitment to straightforward and user-friendly navigation is pivotal in maintaining user satisfaction and engagement levels, epitomizing the application's emphasis on user convenience and operational efficiency. By avoiding any potential navigational ambiguities, this approach guarantees a fluid and delightful user experience, promoting user retention and overall application success.

5.6.3 Visual Versatility – Dark and Light Mode

Visual versatility is another crucial aspect of the user interface in the application. The application offers both dark and light modes (figures: 5.17, 5.18), allowing users to choose their preferred visual interaction space based on comfort, lighting conditions, and personal preference. This feature is not just an aesthetic variation but a user-centric approach to accommodate diverse user preferences and environmental conditions.

The integration of adaptive themes contributes significantly to user satisfaction, allowing users to personalize their experience. The availability of these varied visual themes is crucial in creating an inclusive application environment, meeting the diverse visual preferences of the user base, and ensuring an enriching user experience.



Your QR Code

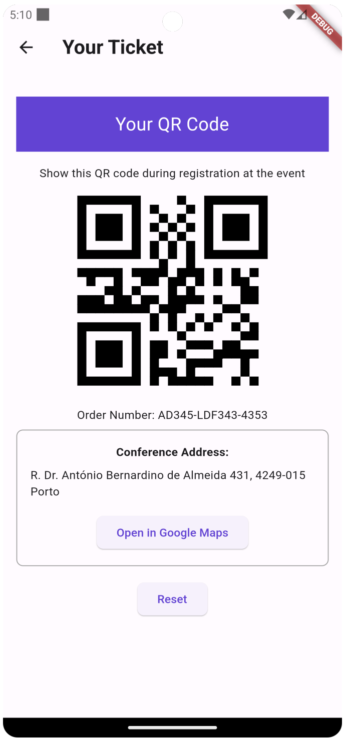
Show this QR code during registration at the event

Order Number: dxczvczvxcv

Conference Address:

R. Dr. António Bernardino de Almeida 431, 4249-015 Porto

Figure 5.17: QR code Screen adapted for dark mode.



Your QR Code

Show this QR code during registration at the event

Order Number: AD345-LDF343-4353

Conference Address:

R. Dr. António Bernardino de Almeida 431, 4249-015 Porto

Figure 5.18: QR code Screen adapted for light mode.

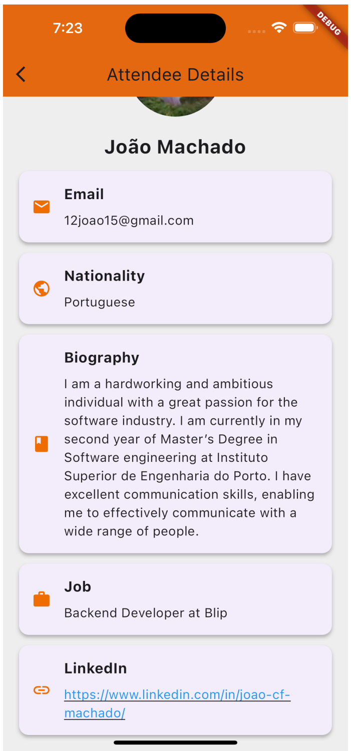
5.6.4 Effective Loading Indicators

Loading indicators are integrated within the application to provide real-time feedback and assure users about the ongoing processes. These indicators are vital in keeping the user informed about the application's operational state, thereby preventing potential frustration and confusion. They act as real-time communicators, signaling the users about the various processes and transitions occurring within the application.

This feature ensures that the users are not left in a state of uncertainty regarding the app's responsiveness. The visibility and clarity provided by these indicators are instrumental in maintaining user trust and engagement, helping to retain user attention and mitigating any potential uncertainties during transitions and task executions.

5.6.5 Universal Iconography

The application commits to a universally inclusive design by employing universally recognized icons, enhancing global understanding and acceptance. These icons represent a global visual language that surpasses linguistic boundaries, allowing users from diverse backgrounds to intuitively understand and navigate through the application. Such universally comprehended icons are strategically employed, optimizing navigability and functionality for users from various linguistic and cultural backgrounds.



7:23

DEBUG

Attendee Details

João Machado

Email

12joao15@gmail.com

Nationality

Portuguese

Biography

I am a hardworking and ambitious individual with a great passion for the software industry. I am currently in my second year of Master's Degree in Software engineering at Instituto Superior de Engenharia do Porto. I have excellent communication skills, enabling me to effectively communicate with a wide range of people.

Job

Backend Developer at Blip

LinkedIn

https://www.linkedin.com/in/joao-cf-machado/

Figure 5.19: The Attendee Details Screen employs recognizable icons and clear imagery to convey information effectively.

Figure 5.19 illustrates the Attendee Details Screen, where clear and concise visual elements are paired with universal icons to provide users with an instantaneous understanding of the available features and information. This screen exemplifies the careful integration of user-centric design principles, offering users detailed insights through a harmonious blend of text, iconography, and imagery.

The depicted SpeakerDetailsScreen class in Dart demonstrates how each piece of information is paired with relevant, universally understood icons and is encapsulated within a card-like structure, facilitating readability and visual appeal.

```
Widget _buildDetailCard(String title, IconData icon, String detail) {
  return Card(
    // ... Other properties and children widgets
    child: Row(
      children: [
        Icon(icon, color: Colors.orange.shade800, size: 24),
        // ... Other widgets to display title and detail
         Text(
            detail,
            style: const TextStyle(fontSize: 16, color: Colors.black87),
            ),
        ],
    ),
  );
}
```

Listing 5.9: Fuction to create generic card elements on the screen.

The code snippet 5.9 reflects the method *buildDetailCard*, which takes a title, an icon, and detail, creating a consistent and structured display of information, resonating well with the application's emphasis on user-friendly design and universal accessibility. The combination of the appropriate icons with the corresponding details emphasizes clarity, ensuring users can easily perceive and understand the provided information.

This meticulous attention to visual communication is integral to the broader vision of inclusivity and user-centric design. It enables users across the globe to interact with the application with confidence and ease, elevating the overall user experience and reinforcing the application's role as a universally accessible tool.

In conclusion, the convergence of universal iconography, meticulous detail representation, and seamless navigation illustrates the profound commitment of the application to delivering an intuitive and enriching user experience. Every component, from its universally recognized icons to its detailed representation of information, contributes to an interaction landscape that is harmonious and resonant with a diverse user spectrum, aligning impeccably with evolving paradigms of user-centric technology interaction. The application stands as a beacon of seamless integration between technology and human interaction, providing a platform that is both enriching and universally inclusive.

5.7 Multi-Conference Support and Modularity

The developed application is architected with the flexibility to support multiple conferences concurrently. This feature is vital in ensuring the application's scalability and versatility, enabling it to accommodate a variety of conferences without any overlap or interference. Users, upon initiating the application, are presented with a list of active conferences, allowing them to choose their conference of interest efficiently. This interface is clean and intuitive, ensuring that users can easily navigate through the available options and make their selection with minimal effort and time.

The list of conferences displayed to the users (figure 5.20) is dynamic and is controlled by an *isActive* field associated with each conference. This field acts as a toggle, manipulated by administrators, to either display or conceal conferences based on their active status. This ensures that administrators have direct control over the visibility and accessibility of conferences, allowing them to manage the active conferences efficiently and maintain an uncluttered and relevant interface for the users.

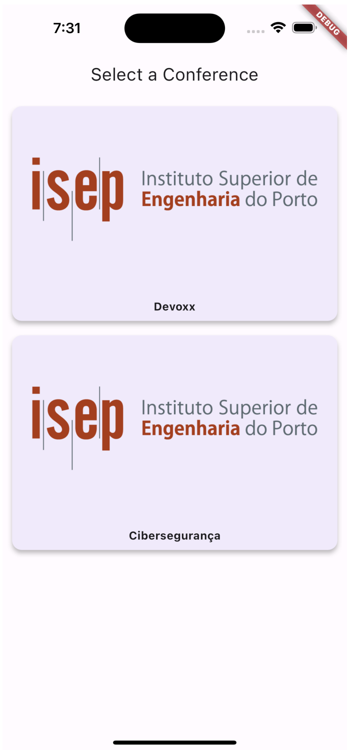

7:31
DEBUG
Select a Conference
isep Instituto Superior de Engenharia do Porto
Devoxx
isep Instituto Superior de Engenharia do Porto
Cibersegurança

Figure 5.20: Displaying available conferences.

Upon selecting a conference, users have the option to create a new account associated with that specific conference or log in using existing credentials. This delineation ensures that users and their interactions are confined to the selected conference, maintaining clear boundaries and avoiding any cross-conference interactions or data mixing. All subsequent activities, data interactions, and functionalities accessed by the users are exclusively related to their chosen conference.

For instance, the events, speakers, and information accessed or created by staff accounts pertain solely to the associated conference. Whether it is viewing dashboards containing conference-specific information or managing profiles, the application ensures that all interactions are within the context of the selected conference. Attendees can tailor their profiles, upload pictures, and edit information while maintaining the conference-specific context.

The integration and management of conferences are facilitated by a robust and extensible backend, developed using Java. This backend serves as the backbone for adding new conferences and managing existing ones, providing a modular and scalable solution. The extensible nature of the backend means that new conferences can be seamlessly integrated without the need for extensive code modifications or system overhauls.

Administrators can, through the backend application, easily add, modify, or deactivate conferences, ensuring that the application remains dynamic and adaptable. This modularity is paramount in maintaining a future-proof and scalable application, allowing for continuous expansion and adaptation based on evolving needs and requirements.

In essence, the multi-conference support coupled with a modular and extensible backend creates a versatile and scalable application environment. The user-centric approach ensures that users can navigate and interact with multiple active conferences effortlessly while maintaining distinct and independent conference ecosystems. Administrators have the flexibility to manage the conferences efficiently, ensuring the application's relevance and adaptability in the dynamically evolving conference landscape.

The encapsulated and conference-specific interaction spaces, the dynamic and adaptable user interfaces, and the robust backend collectively reflect the application's commitment to providing a versatile, user-friendly, and scalable solution, catering to the diverse and evolving needs of the conference ecosystem.

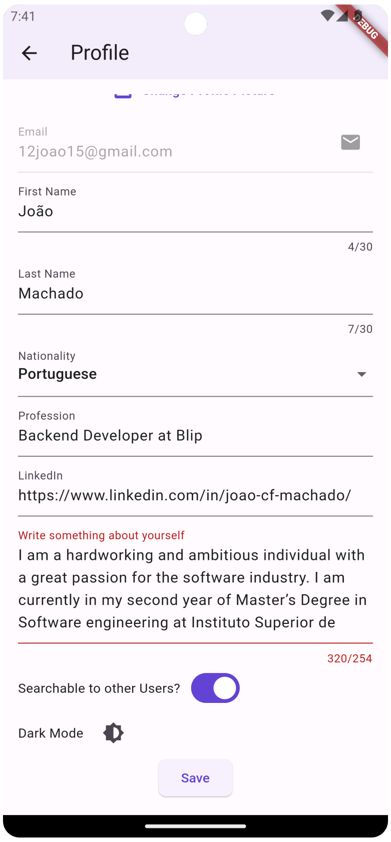
5.8 Attendee Functionalities

The attendee role within the application is fundamental, designed to enrich the user experience during the conference, offering an extensive range of features and functionalities. The attendee is essentially a conference participant who leverages the application to enhance their conference experience, interact with the events, and manage their profiles and activities efficiently. The application serves as a comprehensive tool for attendees, encapsulating various functionalities that are tailored to meet their needs and preferences, providing a seamless and intuitive interface for interaction, information retrieval, and management.

5.8.1 Profile Management

Profile Management stands as a cornerstone for user interaction within the app. Attendees can effortlessly modify different elements of their profiles such as images, bio, profession, and LinkedIn profiles. This feature ensures the most authentic and current representation of the user within the application, granting them the autonomy to project a virtual presence that aligns with their real-world persona.

Beyond the mere modification of personal details, this feature plays a critical role in allowing attendees to control the visibility and accessibility of their profiles (figure 5.21). This means users can manage their interaction preferences, deciding whether their profiles should be searchable or remain private, offering an enhanced layer of privacy and interaction management within the application environment.



Email
12joao15@gmail.com

First Name
João
4/30

Last Name
Machado
7/30

Nationality
Portuguese

Profession
Backend Developer at Blip

LinkedIn
https://www.linkedin.com/in/joao-cf-machado/

Write something about yourself
I am a hardworking and ambitious individual with a great passion for the software industry. I am currently in my second year of Master's Degree in Software engineering at Instituto Superior de
320/254

Searchable to other Users?

Dark Mode

Figure 5.21: Profile screen where users can select if they want to be searchable.

The user interface in this section is meticulously designed to be user-friendly, providing clear navigation paths and intuitive operation mechanisms. This ensures that users can effectuate any adjustments with ease, enhancing the overall user experience and interaction within this domain of the application.

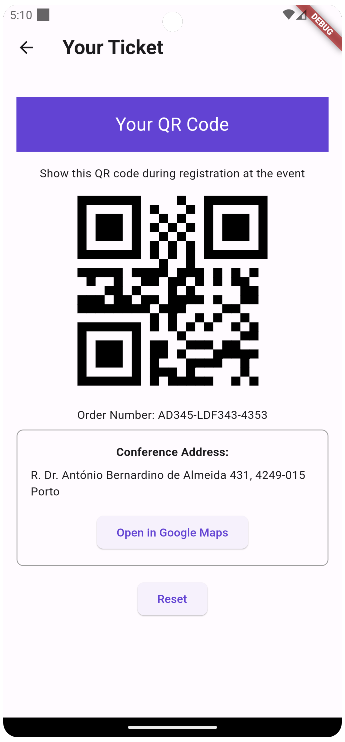
5.8.2 Enhanced Ticket Management

The introduction of the "Add or View Ticket" feature marks a pivotal innovation in redefining and enhancing ticketing processes for attendees. This feature allows users to generate QR codes for their tickets, serving as the pivotal medium for ticket validation and check-ins. By doing so, it simplifies and accelerates the initial entry procedures, reducing the complexities and redundancies associated with traditional ticketing methods.

Moreover, the feature meticulously integrates with Google Maps, a pivotal interaction delineated in figures 5.22 and 5.23. This implies that attendees have access to not only succinct and user-friendly directions to the conference location but also to highly reliable and pinpoint accurate navigation. The seamless transition to Google Maps eliminates the need for manual input, offering a path from the user's current location to the exact location of the conference, ensuring an error-free and timely arrival.

This integration is an embodiment of precision and reliability, especially considering the feature's ability to dynamically fetch the conference locations from the database in real-time. Whether a conference is located within ISEP or at another venue, the feature adjusts and provides directions to the correct and current place. This adaptation and integration illustrate the application's commitment to fostering a sense of reliability, convenience, and user-centric design, ensuring attendees can reach varying conference venues without any inconveniences or inaccuracies.

Additionally, this dynamic and real-time adaptation to multiple conference venues underscores the application's relentless pursuit to deliver a robust and reliable user experience. It eliminates any discrepancies or ambiguities in the conference locations, ensuring that users are provided with the most accurate and up-to-date navigational information, irrespective of the constant variability in conference venues. This guarantees that attendees, regardless of their familiarity with the venue or the city, can reach their destination with confidence and ease.



Your Ticket

Your QR Code

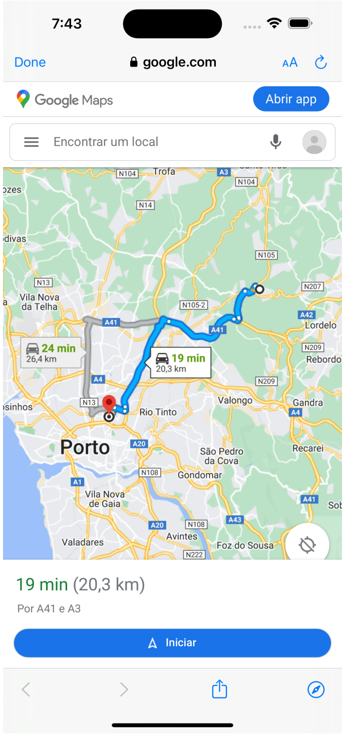
Show this QR code during registration at the event

Order Number: AD345-LDF343-4353

Conference Address:

R. Dr. António Bernardino de Almeida 431, 4249-015 Porto

Figure 5.22: QR code Screen adapted for dark mode.



19 min (20,3 km)

Por A41 e A3

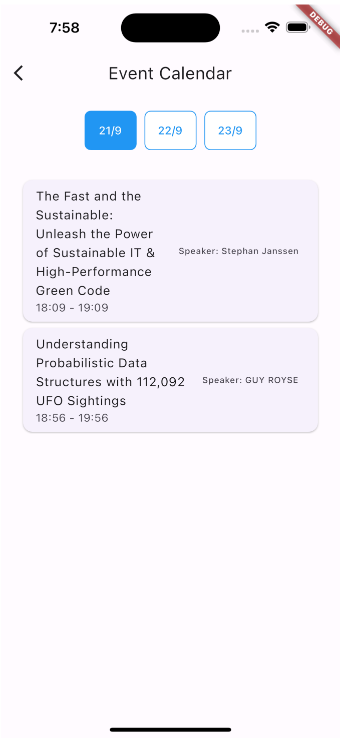
Figure 5.23: Seamless transition to Google Maps, showcasing the dynamic navigation from the user's current location to the exact conference location.

In conclusion, this feature of the application truly exemplifies the implementation of advanced and user-centric design principles. By offering concise yet comprehensive interaction mechanisms, it enables users to effortlessly manage their tickets and navigate through different locations with optimum ease and efficiency. The integration of real-time accurate navigation, dynamic location fetching, and seamless interaction with Google Maps are revolutionary steps in eradicating the hassles and time consumption traditionally associated with ticket management and navigation processes.

5.8.3 Agenda Interaction

The Agenda Screen acts as a multifunctional platform that provides attendees with a panoramic view of the conference schedule. It offers detailed insights into each event, allowing attendees to interact with the event listings and access extensive details including speaker information, event ratings, locations, descriptions, and capacities.

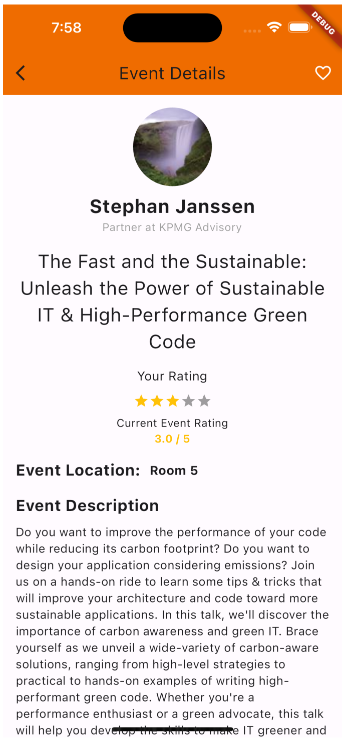
Moreover, this screen is a hub for personal schedule management. It allows attendees to favorite and rate events (figures 5.24, 5.25), providing feedback thus contributing to the dynamic evolution of the conference experience. The feedback collected from the ratings is invaluable, providing organizers with real-time insights into attendee preferences and perceptions.



The Fast and the Sustainable: Unleash the Power of Sustainable IT & High-Performance Green Code
Speaker: Stephan Janssen
18:09 - 19:09

Understanding Probabilistic Data Structures with 112,092 UFO Sightings
Speaker: GUY ROYSE
18:56 - 19:56

Figure 5.24: Agenda Screen.



Stephan Janssen
Partner at KPMG Advisory

The Fast and the Sustainable: Unleash the Power of Sustainable IT & High-Performance Green Code

Your Rating
★★★☆☆
Current Event Rating
3.0 / 5

Event Location: Room 5

Event Description

Do you want to improve the performance of your code while reducing its carbon footprint? Do you want to design your application considering emissions? Join us on a hands-on ride to learn some tips & tricks that will improve your architecture and code toward more sustainable applications. In this talk, we'll discover the importance of carbon awareness and green IT. Brace yourself as we unveil a wide-variety of carbon-aware solutions, ranging from high-level strategies to practical to hands-on examples of writing high-performant green code. Whether you're a performance enthusiast or a green advocate, this talk will help you develop the skills to make IT greener and

Figure 5.25: Event Details Scrren.

The design of the Agenda Screen is meticulously crafted to facilitate seamless interaction. The interface is user-friendly, offering intuitive navigation and interaction paths. This ensures that users can access comprehensive information and manage their schedules effectively, enhancing their overall conference experience.

5.8.4 Personalized Event Management

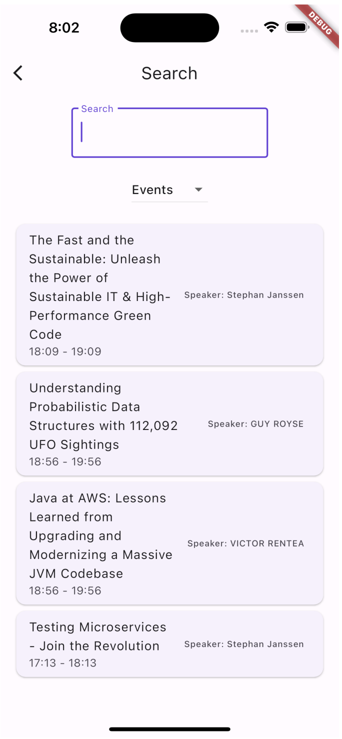
The Favourite Events Screen is a user-specific interface where attendees can create and manage their preferred events list. It provides users with the autonomy to create a personalized schedule that aligns with their interests and preferences, allowing them to optimize their conference experiences based on their individual needs.

This functionality is designed with user convenience at its core. It provides a seamless and intuitive interaction environment, allowing users to modify and access their favorite events effortlessly. The ease of navigation and the interaction mechanisms within this feature facilitates swift schedule management, enhancing user experience.

Moreover, this personalized space within the application stands as a testament to the application's commitment to offering a customizable user experience. It allows attendees to interact, manage, and modify their schedules based on their preferences, ensuring that every attendee can tailor their conference experience to their liking.

5.8.5 Enhanced Search Capabilities

The Search Screen stands as a fundamental component within the application, offering a multifunctional domain where users can effortlessly navigate through Events, Speakers, and other Attendees, as illustrated in figure 5.26. This functionality is pivotal in providing detailed insights about each search result, fostering clarity and a comprehensive understanding of the diverse elements within the conference.



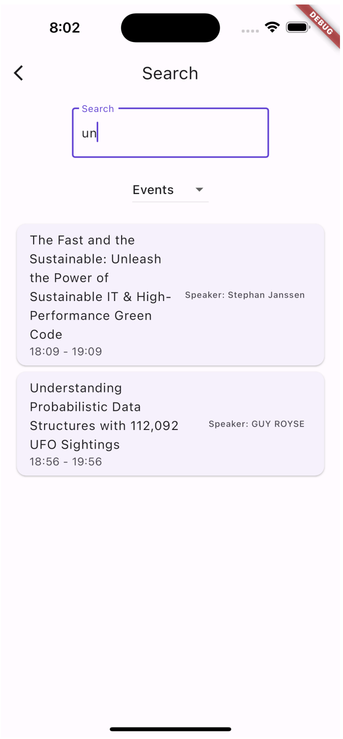
The Fast and the Sustainable: Unleash the Power of Sustainable IT & High-Performance Green Code
18:09 - 19:09
Speaker: Stephan Janssen

Understanding Probabilistic Data Structures with 112,092 UFO Sightings
18:56 - 19:56
Speaker: GUY ROYSE

Java at AWS: Lessons Learned from Upgrading and Modernizing a Massive JVM Codebase
18:56 - 19:56
Speaker: VICTOR RENTEA

Testing Microservices - Join the Revolution
17:13 - 18:13
Speaker: Stephan Janssen

Figure 5.26: Search Screen displaying all events.



The Fast and the Sustainable: Unleash the Power of Sustainable IT & High-Performance Green Code
18:09 - 19:09
Speaker: Stephan Janssen

Understanding Probabilistic Data Structures with 112,092 UFO Sightings
18:56 - 19:56
Speaker: GUY ROYSE

Figure 5.27: Search screen displaying events with "un" on the title.

One noteworthy aspect of this feature is the implementation of real-time search capability, seen as a commitment to enhancing efficiency and accuracy in user interactions (5.27). This dynamic approach significantly diminishes the time and effort users invest to locate specific items, elevating the overall user experience by providing immediate and precise results.

The underlying implementation employs a generic method, *fetchFromFirebase<T>*, allowing for versatile interactions with different data types, thereby minimizing code redundancy and ensuring a clear and concise codebase. This abstract logic not only simplifies the workflow but also enables easy extendability for future developments, serving as a testament to the diligent adherence to good coding practices throughout the project.

```
void _fetchFromFirebaseBasedOnSelectedOption() async {
    if (_selectedOption == 'Events') {
      await _fetchFromFirebase<Event>('events', Event.fromDocument);
    } else if (_selectedOption == 'Speakers') {
      await _fetchFromFirebase<Speaker>('speakers',
          (doc) async => Speaker.fromMap(doc.data() as Map<String, dynamic>));
    } else if (_selectedOption == 'Users') {
      await _fetchFromFirebase<User>(
          'users', (doc) async => User.fromDocument(doc));
    }
  }

  Future<void> _fetchFromFirebase<T>(String collection,
      Future<T> Function(DocumentSnapshot doc) fromDocumentFunction) async {

      final List<Future<T>> futureResults =
          snapshot.docs.map((doc) => fromDocumentFunction(doc)).toList();
      final List<T> results = await Future.wait(futureResults);

      setState(() {
        _isLoading = false;
        if (T == Event) {
          _eventResults = results as List<Event>;
          filteredEvents = List.from(_eventResults);
        } else if (T == Speaker) {
          _speakerResults = results as List<Speaker>;
          filteredSpeakers = List.from(_speakerResults);
        } else if (T == User) {
          _userResults = results as List<User>;
          filteredUsers = List.from(_userResults);
        }
      });
    } catch (error) {
      ScaffoldMessenger.of(context).showSnackBar(
          SnackBar(content: Text('Error fetching from $collection.')));
      setState(() => _isLoading = false);
    }
  }
```

Listing 5.10: Dart method showcasing the generic approach to fetch data based on the selected option.

This abstracted and generic approach forms the backbone of the Search Screen, allowing the application to adapt and expand seamlessly. The emphasis on user-centric design and efficient coding practices is emblematic of the application's dedication to delivering optimal user interaction and a fluid user experience. By integrating streamlined search processes and robust coding practices, the application enables users to interact with various conference elements with heightened ease and precision, reflecting a harmonious blend of functionality and adaptability.

5.8.6 Navigational Assistance

The Conference Map Screen is a vital component, providing a detailed spatial representation of the conference venue. It offers extensive navigational assistance by highlighting various points of interest and providing labels for important marks, acting as a comprehensive informational tool that enhances user navigation within the conference space.

In addition to acting as a navigational guide, this screen also offers supplementary information regarding the conference, ensuring users have a contextual understanding of the conference environment. The provision of such detailed and contextual information significantly enhances the user's spatial awareness and understanding of the conference layout.

The incorporation of this feature within the application is crucial in contributing to a more informed and enriched conference experience for the attendees. It ensures that every participant is well-acquainted with the conference space, allowing them to navigate through different sections of the venue with ease and confidence.

5.8.7 Certificate Procurement

The Participation Certificate Screen is a significant feature that refines and simplifies the process of certificate generation and acquisition. By establishing a seamless interaction with the backend service, this feature enables the automated creation of a request to leverage a customizable PDF template, injecting the attendee's name accurately, and dispatching the resultant certificate to the attendee's registered email as verified proof of attendance.

```java
    private byte[] getFilledPdfTemplate(String userFirstName, String userLastName) throws Exception {
        Resource pdfTemplate = new ClassPathResource("isep_conference_template.pdf");
        try (InputStream is = pdfTemplate.getInputStream()) {
            PdfReader reader = new PdfReader(is);
            ByteArrayOutputStream out = new ByteArrayOutputStream();
            PdfStamper stamper = new PdfStamper(reader, out);

            stamper.getAcroFields().setField("userFirstName", userFirstName);
            stamper.getAcroFields().setField("userLastName", userLastName);

            stamper.setFormFlattening(true);
            stamper.close();
            reader.close();

            return out.toByteArray();
        }
    }
```

Listing 5.11: Java method showcasing the backend service's approach to PDF generation.

The utilization of a PDF template in the backend is a practical approach to accommodate future modifications and enhancements. The template can be easily replaced or modified, and new fields can be incorporated into the template with minimal adjustments to the code, enabling a flexible and adaptable system for changing needs and requirements.

This streamlined process contrasts starkly with traditional methods, where attendees would typically need to approach staff organizers to receive their certificates, leading to potential delays and inaccuracies. The automated process, thus, eliminates these hassles and allows attendees to focus more on their conference experience, reducing the time spent on administrative tasks.

The emphasis on user convenience and operational efficiency is evident in the implementation of the Participation Certificate Screen. The application's dedication to providing a user-friendly experience is apparent through every interaction within the application, ensuring users find value and reliability in the services provided. By refining user interactions and simplifying administrative procedures, the application aims to offer a seamless and efficient user experience in the realm of conference management.

In conclusion, the Attendee Functionalities within the application serve as an extensive suite of tools meticulously designed to enrich and streamline the experience of every conference participant. Each functionality, from Profile Management to Certificate Procurement, is conceived with a user-centric approach, ensuring that attendees can interact, navigate, and manage their conference experiences with optimum ease and efficiency.

The integrative design of each functionality ensures seamless interaction, allowing users to access, modify, and manage various aspects of their conference participation effectively. The application encapsulates a diverse range of features, allowing users to manage their profiles, navigate through the conference, curate personalized schedules, and much more, all while maintaining a user-friendly and intuitive interface. The real-time search capability and the integration with Google Maps are exemplary of the application's commitment to providing cutting-edge solutions, ensuring precise and efficient user interactions.

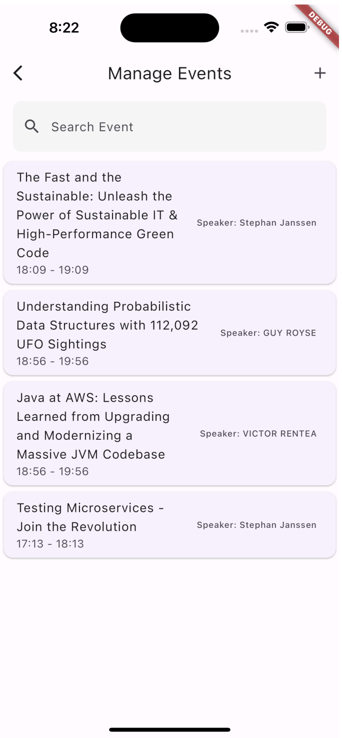
Ultimately, the attendee functionalities are reflective of a broader commitment to creating a customizable, user-friendly, and enriched conference experience. They echo a relentless pursuit of excellence in user experience design, delivering functionalities that are not only innovative but also highly responsive to the unique needs and preferences of each attendee. The application, through its multifarious features, stands as a comprehensive solution, anticipating and addressing the varied needs of the conference attendees, thereby significantly contributing to crafting a more engaging, informative, and seamless conference experience.

5.9 Staff Functionalities

The Staff Functionalities are integral, providing various tools and features enabling efficient conference management. The staff role is pivotal, requiring functionalities that allow seamless event, speaker, and user management, ensuring the smooth operation of the conference. The suite of staff functionalities is imbued with advanced features, intuitive design, and real-time capabilities, all aimed at maximizing efficiency and control in managing the diverse aspects of the conference.

5.9.1 Manage Events

The "Manage Events" feature stands as a pivotal element, allowing staff to create, edit, or delete events with remarkable precision and efficiency. This feature is integral to devising the conference itinerary and remains essential for implementing real-time adjustments to assure the schedule's contemporaneity and accuracy.



Manage Events +

Search Event

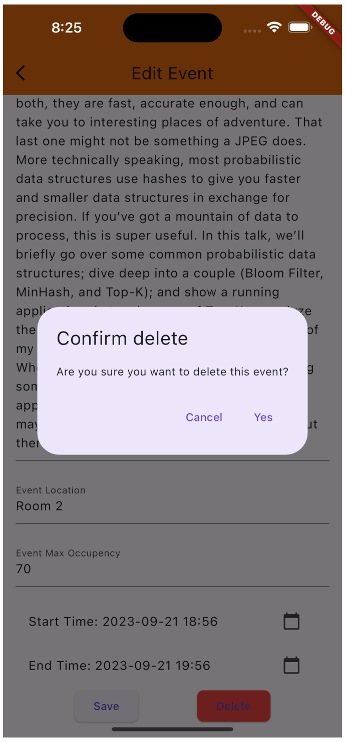
The Fast and the Sustainable: Unleash the Power of Sustainable IT & High-Performance Green Code
Speaker: Stephan Janssen
18:09 - 19:09

Understanding Probabilistic Data Structures with 112,092 UFO Sightings
Speaker: GUY ROYSE
18:56 - 19:56

Java at AWS: Lessons Learned from Upgrading and Modernizing a Massive JVM Codebase
Speaker: VICTOR RENTEA
18:56 - 19:56

Testing Microservices - Join the Revolution
Speaker: Stephan Janssen
17:13 - 18:13

Figure 5.28: Manage Events Screen.



Edit Event

both, they are fast, accurate enough, and can take you to interesting places of adventure. That last one might not be something a JPEG does. More technically speaking, most probabilistic data structures use hashes to give you faster and smaller data structures in exchange for precision. If you've got a mountain of data to process, this is super useful. In this talk, we'll briefly go over some common probabilistic data structures; dive deep into a couple (Bloom Filter, MinHash, and Top-K); and show a running

Confirm delete

Are you sure you want to delete this event?

Cancel Yes

Event Location
Room 2

Event Max Occupency
70

Start Time: 2023-09-21 18:56

End Time: 2023-09-21 19:56

Save Delete

Figure 5.29: Confirm deletion of event pop-up.

Incorporating a real-time search functionality, this feature ensures a streamlined approach to managing existing events, allowing staff to swiftly locate and modify events, thereby reducing the time and effort traditionally associated with schedule management. This is especially useful for making instantaneous modifications, ensuring the agenda accurately mirrors the planned activities.

The user interface of this functionality is meticulously designed to be straightforward and intuitive, catering to staff members with varying levels of technological proficiency. It provides an uncluttered and simple UI, allowing all staff members to navigate and operate effortlessly, regardless of their tech background, ensuring inclusivity and user satisfaction.

Moreover, to prevent inadvertent alterations or deletions, editing and deleting actions are accompanied by a confirm action. This thoughtful inclusion adds a layer of security and reassurance, safeguarding against unintentional mistakes and allowing staff to confirm their intentions before committing to changes. This verification step exemplifies the application's focus on providing reliable and error-resistant solutions to its users.

This functionality is not merely a tool but a testament to the application's commitment to offering robust, responsive, and user-centric solutions. It provides staff with the flexibility and reliability to adapt the event schedule as needed and to ensure the accuracy and relevance of event information efficiently. By blending dynamic functionalities with user-friendly design and secure interaction mechanisms, the "Manage Events" feature optimizes the management process and enhances overall user experience, contributing to the seamless orchestration of the conference.

5.9.2 Manage Speakers

Similar to event management, the “Manage Speakers” functionality allows the creation, editing, or deletion of speakers, using a real-time search and filter feature. It enables staff to maintain an accurate and updated list of speakers, ensuring that any modifications are reflected instantaneously. This feature is essential in managing the array of speakers and their respective information efficiently, providing an organized and structured approach to speaker management.

The immediate reflection of modifications and the real-time search functionality are crucial in maintaining the integrity of the speaker’s information, allowing swift adjustments and updates. This feature is meticulously designed to support the dynamic nature of conference management, offering a responsive and intuitive interface for managing speakers, their details, and their association with respective events.

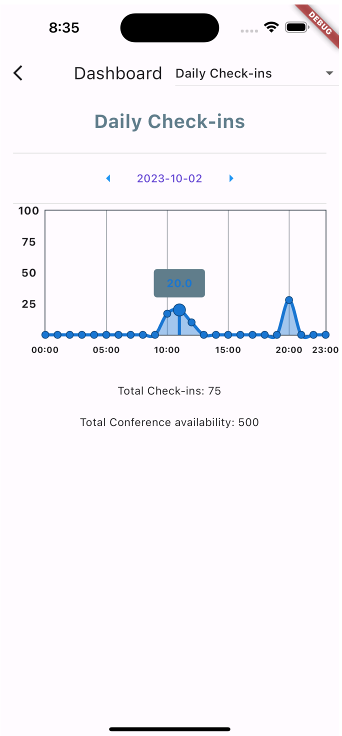
5.9.3 Manage Users

“Manage Users” is another crucial functionality allowing staff to edit or delete users, utilizing real-time search functionality for efficient user management. It provides the staff with the necessary tools to oversee and manage user information and roles effectively. The inclusion of real-time filtering in this section ensures a streamlined approach to locating and managing users, enabling staff to maintain an updated and accurate user base.

This functionality supports the continuous need for user management within the application, providing a comprehensive overview and control over user information and statuses. It is crucial for maintaining a secure and organized user environment, allowing staff to manage user access, information, and roles with precision and reliability.

5.9.4 Dashboards

The Dashboards serve as dynamic and insightful tools, offering staff a myriad of perspectives on varying aspects of the conference. These dashboards encompass a wide range of analytical capabilities, including visualizing daily check-ins (5.30), a graph detailing the quantity and timings of check-ins, to providing an overall conference occupancy pie chart (5.31) and a comprehensive dashboard on event popularity and ratings (5.32). These multifaceted dashboards are instrumental in providing staff with a broad and in-depth understanding of ongoing activities and their statuses.


8:35

Dashboard Daily Check-ins

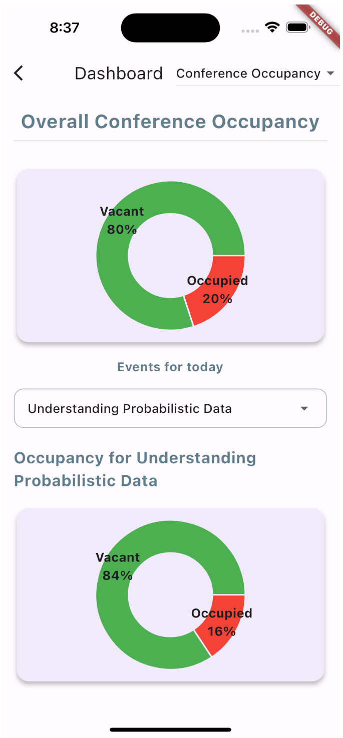
Daily Check-ins

2023-10-02

Total Check-ins: 75

Total Conference availability: 500

Figure 5.30: Daily Check-ins Screen.


8:37

Dashboard Conference Occupancy

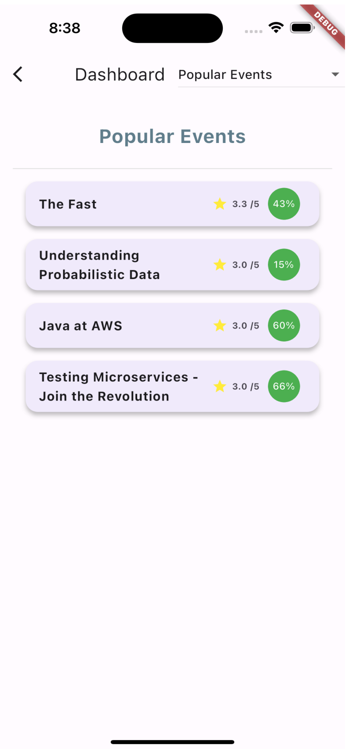
Overall Conference Occupancy

Events for today

Understanding Probabilistic Data

Occupancy for Understanding Probabilistic Data

Figure 5.31: Conference Occupancy and Event Ocuppancy.


8:38

Dashboard Popular Events

Popular Events

The Fast	3.3 /5	43%
Understanding Probabilistic Data	3.0 /5	15%
Java at AWS	3.0 /5	60%
Testing Microservices - Join the Revolution	3.0 /5	66%

Figure 5.32: All events and their rating.

The dynamic depiction of daily check-ins allows staff to ascertain rush hours easily, enabling them to allocate more staff at the gates during peak times and manage resources more efficiently. This proactive approach ensures smoother operation and enhanced attendee experience.

The conference and event occupancy insights serve as essential tools for optimal resource allocation within different areas of the venue. By monitoring the real-time capacity of the conference and specific events, staff can manage resources and crowd control more effectively, ensuring a balanced distribution of services and preventing overcrowding in specific areas.

The event rating dashboard acts as an immediate feedback tool, enabling staff to discern the effectiveness and appeal of each event, allowing for real-time adjustments and future planning based on attendee preferences and feedback.

```
enum DashboardType {
  dailyCheckins,
  conferenceOccupancy,
  popularEvents,
  demographics,
}

extension DashboardTypeExtension on DashboardType {
  String toPrettyString() {
    switch (this) {
      case DashboardType.dailyCheckins:
        return 'Daily Check-ins';
      case DashboardType.conferenceOccupancy:
        return 'Conference Occupancy';
      case DashboardType.popularEvents:
        return 'Popular Events';
      case DashboardType.demographics:
        return 'Demographics';
      default:
        return 'Unknown';
    }
  }

Widget _buildDashboardContent(DashboardType type) {
  switch (type) {
    case DashboardType.dailyCheckins:
      return const DailyCheckinsChart();
    case DashboardType.conferenceOccupancy:
      return const ConferenceOccupancyChart();
    case DashboardType.popularEvents:
      return const PopularEventsDashboard();
    default:
      return const Center(
        child: Text('Select a dashboard from the top-right dropdown.'));
  }
}
```

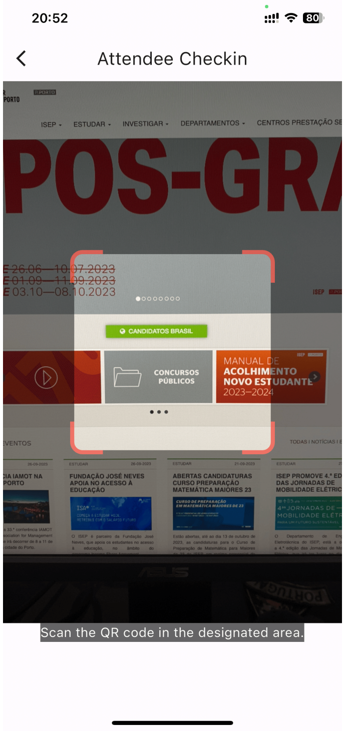
Listing 5.12: Dashboard content build method showcasing easy extensibility.

These dashboards are inherently dynamic and designed to be easily extendable. The provided code snippet (5.12) illustrates the method used for building dashboard content, demonstrating the simplicity of adding new dashboard types. The enum *DashboardType* and its extension method ‘toPrettyString()‘ make it convenient and straightforward to introduce new dashboards that could bring further value and insights.

This adaptability means that the application can easily evolve to meet the emerging and diverse analytical needs of conference management. The flexible design of the dashboards, combined with their diverse range of insights, underlines the application's commitment to providing versatile, adaptable, and insightful tools that aid in informed decision-making and efficient conference management.

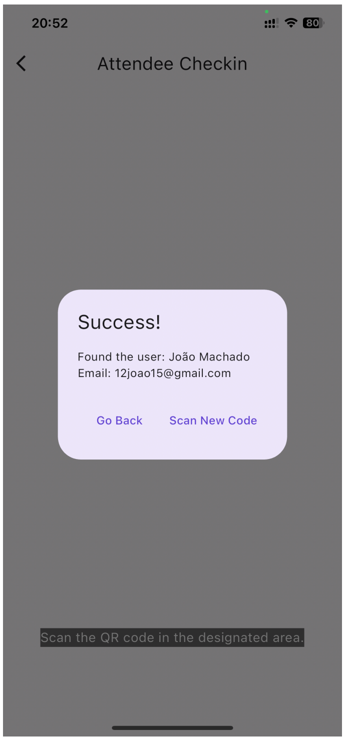
5.9.5 Attendee Check-In

The "Attendee Check-In" feature innovatively transforms the staff's phone into a potent tool for seamless management by leveraging its camera to scan QR codes. It's not merely about verifying the validity of the codes but about instantly displaying the attendee's information and promptly recording the check-in on the database. This feature crystallizes as a cornerstone for managing attendee entry and represents a paradigm shift towards more fluid, efficient, and precise methods of verifying attendance and maintaining impeccable records.



Scan the QR code in the designated area.

Figure 5.33: Real iOS device scanning for QR code.



Success!

Found the user: João Machado
Email: 12joao15@gmail.com

Scan the QR code in the designated area.

Figure 5.34: Real iOS device finding a real QR code.

This feature acts as an enabler for staff, allowing them to effectively manage attendee access, thereby reinforcing the security and the exactness of the attendance logs. The immediate recording of check-in data not only maintains real-time attendance accuracy but also fuels the analytical dashboards with instantaneous data, allowing for a more coherent and up-to-date overview of attendee activities and conference occupancy.

The transformation of a simple phone camera into a QR scanner stands out as a remarkable resource optimization, replacing the need for additional devices and manual labor and thus, saving invaluable time and resources. The quick scan-and-verify mechanism ensures a hassle-free and swift entry experience for attendees, allowing staff to concentrate on more pressing and complex tasks related to conference management.

Moreover, the implications of this feature are far-reaching. It opens up a plethora of opportunities for future expansions and enhancements. Imagine a scenario where staff can validate not only the entry tickets but also verify whether an attendee has collected a preset or availed of a free meal. Such capabilities can significantly optimize resource distribution and prevent any potential discrepancies in service avails.

This versatility and the potential for future integrations mean that this feature can act as a precursor to many innovative solutions aimed at refining and elevating conference management standards. It is not just about streamlining the existing processes but about envisioning and implementing new methods for a more organized, efficient, and enriched user and management experience.

In essence, the “Attendee Check-In” feature is not merely a functionality; it’s a vision for future-proof and advanced conference management, offering immediate benefits and laying down a robust foundation for countless possibilities and enhancements.

5.9.6 Send Notifications

The “Send Notifications” feature is crucial, enabling staff members to send push notifications to all users. Staff can easily input text into the screen and disseminate vital information promptly, ensuring that attendees are kept informed about important updates or emergency situations. This feature plays a pivotal role in maintaining open lines of communication between the staff and the attendees, reinforcing the application’s commitment to ensuring a well-informed and cohesive conference environment.

The ease and swiftness of sending notifications are vital in maintaining a responsive and adaptive communication channel, allowing staff to address the attendees’ collective promptly and effectively. This feature stands as a testament to the application’s focus on creating a harmonious and informed conference environment, ensuring the timely dissemination of crucial information.

5.9.7 Conference Map

The Conference Map screen, shared with the attendee role, provides a detailed representation of the conference venue. It is a pivotal tool for staff members, offering insights into various points of interest within the conference space. It ensures that staff have a comprehensive understanding of the venue layout, aiding them in effectively guiding and assisting attendees.

The availability of detailed and contextual information within this feature ensures that staff can effectively manage and navigate the conference.

The culmination of the "Implementation" chapter furnishes a holistic view of the intricate and robust architecture that underpins the conference management application. It lays bare the amalgamation of meticulous planning, strategic design, and progressive technologies, leading to the realization of an application that stands resilient, functional, and user-centric.

The architecture of the application is not just a structural skeleton; it’s a harmonious orchestration of various components and services working in unison, ensuring seamless interactivity and unassailable reliability. It is a reflection of a thoughtful approach aimed at delivering optimal performance, scalability, and resilience, essential for managing dynamic conference environments.

Security, a paramount concern in the development lifecycle, has been given its due diligence. The adoption of proven security protocols, coupled with the meticulous enforcement of secure practices, assures the safeguarding of sensitive user information and the overall integrity of the application. The measures implemented form a solid defense line against potential breaches and unauthorized accesses, substantiating the application's commitment to user privacy and data protection.

Storage solutions, meticulously chosen and implemented, have empowered the application with efficient data handling and management capabilities. The leveraging of cloud-based solutions assures the accessibility, reliability, and scalability of the data storage, making it conducive to the fluctuating and expanding needs of conference management.

The emphasis on user interface and experience is evident in the aesthetic and interactive design elements embedded within the application. The commitment to delivering a coherent, intuitive, and enriching user experience manifests through every interaction, every visual cue, and every piece of information presented within the application. It is a testament to the application's dedication to fostering user engagement and satisfaction.

The diverse functionalities provided for both attendees and staff spotlight the application's versatility and adaptability. Innovative features like real-time search, dynamic dashboards, and QR code scanning enhance the overall management experience and introduce avenues for future enhancements and integrations. The continual evolution and adaptability of these functionalities signify the application's potential to meet the ever-evolving demands of conference management.

In retrospect, the diverse sections in this chapter coalesce to portray a comprehensive picture of a versatile and forward-looking application. The integration of advanced technologies, the attention to user-centric design, the focus on robust and secure practices, and the infusion of innovative features illuminate the application's pursuit of excellence and its readiness to embrace future advancements. This chapter stands as a testament to the rigorous and enlightened approach adopted in implementing a solution that is not only contemporaneously relevant but also poised to adapt and evolve in the foreseeable future.

Chapter 6

Evaluation

In this segment, a meticulous verification process is undertaken to ascertain the quality and integrity of the developed application in its entirety. Each requirement is attributed a value, selected from the set {2, 4, 6, 8, 10}, which denotes its significance within the respective quality factor it pertains to.

6.1 Functional Dimension

The Functional Dimension serves as a representation of the functional requirements delineated in this dissertation, establishing a foundation to quantify the application's efficacy in satisfying the outlined functionalities.

6.1.1 General Dimension

Table 6.1: Functional, General Quality Factor

Requirement	**Weight**	**Level of Fulfilment**
FGEN1 - User creates an account.	10	100%
FGEN2 - Admin creates a staff account.	10	100%
FGEN3 - The user can add and show the attendee ticket.	10	100%

The impeccable accomplishment of the general requirements elucidates the robustness and precision embedded within the application. The creation of user and staff accounts (**FGEN1**, **FGEN2**) are fundamental pillars, essential for maintaining the operational integrity and coherence of the application. Owing to their pivotal role in sustaining the foundational structure, these elements have been ascribed a paramount weight value of 10. The functionality enabling the user to incorporate and display the attendee ticket (**FGEN3**) is equally pivotal and is, thus, allocated a synonymous weight value, echoing its indispensable nature and the consequential implications delineated previously in this dissertation.

Every requirement within this dimension has been successfully attained and meticulously materialized, emblematic of an advanced and secure authentication and authorization flow, coupled with a fortified storage solution ensuring the impenetrable safeguarding of user-specific sensitive information. Attendees are provided with a streamlined approach to effortlessly amalgamate the received ticket into the application, which subsequently triggers the seamless generation of the corresponding QR code. This integration not only reinforces the robustness of the identity verification process but also enhances the overall practicality and functionality of the application, thereby enriching the user experience.

It is significant to highlight that the 100% fulfillment of these requirements is indicative of the application's steadfast dedication to achieving operational supremacy, maintaining user-centric design principles, and upholding stringent data security standards. These fundamental functionalities, perfected to the finest detail, embody a harmonious synthesis of innovation, reliability, and user accessibility. The outcome is an enriched and intuitive user interface, reinforcing the application's versatility and adaptability to meet the evolving and multifaceted demands inherent to contemporary conference management scenarios.

6.1.2 Social Dimension

Table 6.2: Functional, Social Quality Factor

Requirement	Weight	Level of Fulfilment
FSO1 - The user can edit his profile.	6	100%
FSO2 - The user can see the agenda of the conference.	10	100%
FSO3 - The conference organizers can add and remove events to the calendar view.	10	100%
FSO4 - The conference organizers can send notifications to the attendees.	4	100%
FSO5 - The user can rate an event.	6	100%

The elements within the social dimension are crucial in fostering a communicative and user-interactive platform, and they have been executed with meticulous precision to reach a level of fulfillment of 100%. The components, **FSO2** and **FSO3**, are integral, acting as the cornerstone of the platform's social interface owing to their cardinal roles in establishing a user-interactive milieu, thereby warranting a maximal weightage of 10.

FSO1 is instrumental in user personalization, enabling users to revise their profiles, thereby holding substantial importance in enhancing user autonomy and experience. However, its relative weight is moderated to 6, acknowledging the comprehensive customization prospects available during the initial account creation phase.

Similarly, **FSO5** is a vital feature facilitating users in expressing their event experiences through ratings, thus generating instantaneous and collective feedback, which acts as a reservoir of insights for both attendees and the organizational staff. It enriches the communal understanding and interactive experience of the event, yet its elemental importance is balanced with a weight of 6, considering it is not a determinant of the core operational functionality of the application.

The feature, **FSO4**, provides a conduit for the organizers to relay notifications to the attendees, establishing a real-time communicative channel. This ensures that attendees remain informed and updated about ongoing developments and announcements, enhancing the fluidity of information dissemination. However, its utilitarian importance is recognized with a weight of 4, considering its non-central role in the foundational operational framework and its selective application.

To encapsulate, the social dimension embodies a spectrum of user-centric functionalities, intricately interwoven to create a responsive and engaging user environment. The meticulous integration of each functionality emphasizes user engagement, seamless interaction, and enriched user experience, underlining the commitment to developing a versatile, adaptive, and comprehensive platform. This careful orchestration of interactive features illustrates the application's endeavor to elevate the user interaction paradigm, ensuring the platform's relevance and adaptability in addressing the multifarious needs of the contemporary conference ecosystem.

6.1.3 Settings Dimension

Table 6.3: Functional, Settings Quality Factor

Requirement	**Weight**	**Level of Fulfilment**
FSE1 - The event organizers can customize basic event information.	10	100%
FSE2 - The event organizers can access and change views and filters on the metrics dashboard.	10	100%
FSE3 - Light and Dark modes.	2	100%

The ability of event organizers to tailor event information **(FSE1)** is fundamental in the conference application environment. This facility to create, modify, and manage events and associated speaker information crafts the primary information stratum that attendees seek within the application. Given its pivotal role in shaping user interactions and information exchange, it is assigned a maximum weight of 10.

FSE2 enables diverse dashboard viewability and ensures smooth navigational transitions for staff members. This characteristic is foundational for the administrative facet of a conference application, offering real-time data visualization in a user-friendly and intuitive format, thus enhancing decision-making and resource adaptation based on attendee feedback. Given its imperative nature in managing conference dynamics and enhancing operational efficiency, it rightfully merits a weight of 10.

On the other hand, **FSE3**, representing the theme customization feature allowing transition between Light and Dark modes, has become a ubiquitous presence in most contemporary applications. It extends an additional layer of user interface personalization to the user. However, within the specialized context of a conference application, its significance is comparatively minimal, and hence, it is accorded a lower weight of 2.

To sum up, the Settings functionality serves as the nexus between user interaction and administrative adaptability. The precise execution of each feature within this dimension emphasizes the seamless synergy between user-centric design and administrative functionality, ultimately contributing to the robust and adaptable nature of the platform. The meticulous integration of these elements reflects the application's commitment to delivering a versatile and intuitive user experience while maintaining operational excellence and administrative flexibility in a dynamic conference environment.

6.1.4 Search Dimension

Table 6.4: Functional, Search Quality Factor

Requirement	Weight	Level of Fulfilment
FSH1 - The user can search for events and speakers.	8	100%
FSH2 - The user can search for specific zones to access details and directions.	6	100%

The implementation of a comprehensive and user-friendly search functionality distinguishes this application from its counterparts, having all requirements met with unmitigated precision and exceeding the standards set by other conference applications. The ability to promptly search for events, speakers, and attendees, especially with the streamlined and intuitive interface provided, significantly amplifies the application's social and informational value. It serves as an enabler for enriched networking opportunities and optimal access to essential event details, enhancing the overall user experience.

FSH1, which allows the seamless search for events and speakers, is particularly laudable, serving as a conduit for enhanced interaction and information exchange. Any modifications to event, speaker, or attendee information are instantaneously reflected in the search results, ensuring real-time accuracy and reliability. Given the paramount importance of this feature in fostering user engagement and social connectivity, it has been assigned a weight of 8.

FSH2, albeit not as critical as **FSH1**, accentuates the application's user-centric approach by offering navigational aid through integration with Google Maps and a dedicated screen for venue information. This feature, particularly beneficial for international attendees unfamiliar with the locale, offers convenience and essential details, thereby contributing to the application's overall appeal and functionality.

In conclusion, the search functionality, meticulously developed and integrated, stands as a testament to the application's commitment to delivering a premium user experience and unrivaled utility. The symbiosis of intuitive design and real-time accuracy within this feature not only underscores the application's innovative edge but also solidifies its standing as a holistic and forward-thinking solution in the realm of conference management applications.

6.2 Non-Functional Dimension

This section encapsulates the non-functional requirements, which, although indirect in their influence on core application features, play a pivotal role in shaping the overall user experience. They address attributes beyond the immediate functionalities, encompassing performance, system compatibility, and overarching usability metrics.

6.2.1 Adaptability Dimension

NFAD1, which emphasizes the compatibility of the application across both Android and iOS platforms, is of paramount importance. The universality it offers ensures the application reaches a vast demographic, eliminating potential biases tied to specific device preferences. Given the adoption of Flutter as the development framework, coupled with the use of platform-agnostic extensions and rigorous testing on an array of devices (both virtual and

Table 6.5: Non-Functional, Adaptability Factor

Requirement	Weight	Level of Fulfilment
NFAD1 - Application supports Android and iOS Operating Systems.	10	100%.
NFAD2 - The application must resize to different screen sizes.	10	100%
NFAD3 - User Interface Multiple Language Support.	4	100%

tangible) for both OS, it can be confidently asserted that this requirement has been fulfilled to perfection.

NFAD2 underscores the significance of adaptability, ensuring the application's user interface gracefully conforms to a spectrum of screen dimensions. This requirement was meticulously verified using diverse devices spanning both Android and iOS ecosystems, with comprehensive attention to device variations within the Android landscape. Dedicated techniques, such as SafeArea, have been incorporated to guarantee seamless resizing, making the application resilient to potential discrepancies in device screen/resolutions. Recognizing that any inconsistency here could severely impair the user experience and render the application non-operational, this requirement justifiably holds a weight of 10.

NFAD3, which pertains to multi-language support in the user interface, has been successfully implemented. This feature allows users to switch between languages, enhancing usability and accessibility for a global audience. Currently, the application supports both English and Portuguese languages, with the capability to easily extend support for additional languages. The inclusion of a language selection option in the dropdown menu ensures users can choose their preferred language seamlessly. As a result, this requirement has achieved a level of fulfillment of 100%.

In conclusion, while the non-functional requirements may not directly influence the primary features of the application, their meticulous realization ensures optimal performance, wide accessibility, and a refined user experience. With a majority of them achieved with unparalleled precision, it underlines the application's commitment to versatility, adaptability, and inclusiveness.

6.2.2 Performance Dimension

Table 6.6: Non-Functional, Performance Factor

Requirement	Weight	Level of Fulfilment
NFPE1 - Assets and data are cached locally, improving responsiveness and network usage.	6	100%
NFPE2 - The application should load low-resolution images on a low network connection.	4	75%

NFPE1 emphasizes the significance of localized data caching to amplify application responsiveness while judiciously consuming network resources. It achieved an impeccable 100% level of fulfillment. While this feature reinforces performance efficiency, it's noteworthy that the application could remain operational even in its absence. Additionally, users on stable

network connections might not discernibly perceive this optimization, explaining its weight of 6. Nevertheless, its full realization underscores a commitment to delivering an optimal user experience across varied network scenarios.

NFPE2 targets the adaptability of the application to varying network strengths by promoting the use of low-resolution images under compromised connectivity. Despite its weight of 4, the requirement saw 75% fulfillment. The nuanced challenge here is the automatic detection of network speed to dictate image resolution. However, the application employs a strategic approach to mitigate this: it compresses all images uploaded to economize on both storage and bandwidth. Furthermore, images are initially rendered in low resolution, with a seamless transition to higher quality when feasible. This dynamic, often indiscernible owing to the inherently diminutive image dimensions, ensures users get a fluid experience without taxing their network bandwidth unduly.

In essence, the performance dimension, while seemingly nuanced, underlines the meticulous efforts to optimize user experience. The application's commitment to ensuring smooth and efficient operation, regardless of the variable network conditions, highlights its user-centric design philosophy.

According to the QEF (Quality Evaluation Framework), the importance each dimension (Di) has to the final solution is given by:

$$D_i = \sum_n (p_n \times factor_n), \sum_n (p_n) = 1, p_n \in [0, 1]$$

Figure 6.1: Formula to calculate dimension importance.

And the contribution from each quality factor (Fn) is given by:

$$F_n = \frac{1}{\sum_m pr_m} \times \sum_m (pr_m \times pc_m), p_n \in [0, 1]$$

Figure 6.2: Formula to calculate quality factor contribution.

With a global deviation (D) given by:

$$D = \sqrt{\sum_j (1 - \frac{Dim_j}{100})^2}$$

Figure 6.3: Global deviation formula.

The quality of the entire solution (Q) can be computed using:

$$Q = 1 - \frac{D}{\sqrt{n}} \times 100, Q \in [0, 100]$$

Figure 6.4: Formula to calculate the quality of the solution.

The formulas above were used to compute the importance of each quality factor to the solution and each completion level.

Table 6.7: QEF — Quality factor weight contribution and fulfillment level

Dimension	Quality Factor	Weight	Contribution
Functional	General	0.091	100%.
Functional	Social	0.227	100%
Functional	Settings	0.227	100%
Functional	Search	0.045	100%
Non-Functional	Adaptability	0.38	100%
Non-Functional	Performance	0.38	83%

After applying the formulas, the quality score for the quality of the proposed solution, in its current state, is 98%.

6.3 Testing Phase

Testing is an indispensable component in the software development lifecycle. It serves not just as a validation tool but as a continuous integration instrument, reinforcing the application's reliability, functionality, and user experience. In essence, without a comprehensive testing phase, developers might be releasing a product into the wild with hidden snags and glitches, potentially undermining user trust and the product's overall reputation.

The importance of testing grows multifold in the context of mobile applications. Given the plethora of devices, operating systems, screen sizes, and network conditions, the challenge of ensuring consistent performance and user experience across all these variables is formidable. Thus, the testing phase becomes not just a checkpoint but a crucible, refining and honing the application for the real world.

For this application, it was embraced a multi-pronged testing approach. During the development phase, preliminary tests were conducted to check the functionality and responsiveness of the application. However, it is acknowledged the fact that developers, being closely entwined with the product, might overlook certain usability issues or interface glitches that a fresh pair of eyes could spot.

Hence, it incorporated feedback mechanisms, using Google Forms, to solicit user experiences and insights during our beta testing phase. Google Forms provided an easy-to-use, accessible platform for testers to report bugs, suggest enhancements, and offer general feedback about their interaction with the application. The simplicity of the tool ensured that testers, regardless of their technical proficiency, could provide valuable input without feeling overwhelmed.

The feedback collected served as a goldmine of information. It highlighted not just the technical pitfalls but also user interface design choices that might not be intuitive to the end-user. Each piece of feedback was carefully analyzed, categorized, and prioritized. This informed our iterative development process, ensuring that these problems were addressed not just the critical bugs but also made necessary refinements to the user experience.

6.3.1 Alpha Testing Phase

The alpha testing phase is not just an early stage in the software testing process but a pivotal point where a product is closely examined under genuine conditions. This phase can significantly influence a software's eventual success, especially when catering to a tech-savvy audience.

Recognizing the importance of this phase, the alpha testing was approached with meticulous planning. The primary objective was to evaluate the application's functionality and to gather insightful feedback that could inform future iterations. The MVP of the application was thus presented to a select group of 8 individuals with strong tech backgrounds, ensuring that the feedback would be comprehensive and critical.

It was crucial to test the application across a range of devices. Of the 8 participants, 3 used Android phones, each from a different brand. This diversity ensured a comprehensive assessment across varying Android hardware and software configurations. The remaining 5 participants used iPhones, specifically models from the iPhone 12 and 13 series. The selection of these recent iPhone models ensured that the app's compatibility with Apple's latest devices was thoroughly evaluated.

The application was tested in the context of the Devoxx tech conference. The app was tailored to simulate the events and occurrences of this esteemed conference, providing an authentic usage scenario.

Feedback was methodically collected using Google Forms (figures:6.5, 6.6, 6.7, 6.8, 6.9, 6.10, 6.11, 6.12, 6.13, 6.14, 6.15). The structured nature of this tool allowed for detailed insights. The feedback revealed several areas of improvement:

- Screen size discrepancies led to widgets being incorrectly rendered on different devices.
- The static nature of navigation made the user experience feel disjointed due to the lack of animated transitions.
- The positioning of search boxes at the bottom contradicted the conventional design, leading to user confusion.
- A pronounced need for a 'favorite' function for events was expressed, underlining its significance for a conference application.
- The obligation to click a separate button to initiate searches was identified as counterintuitive.
- Various bugs across functionalities were highlighted, pinpointing immediate areas for rectification.

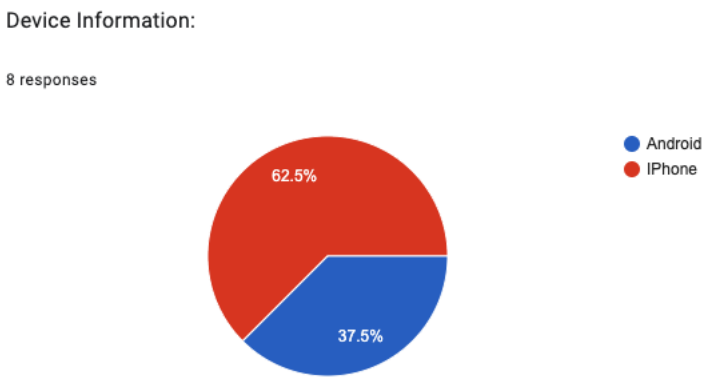
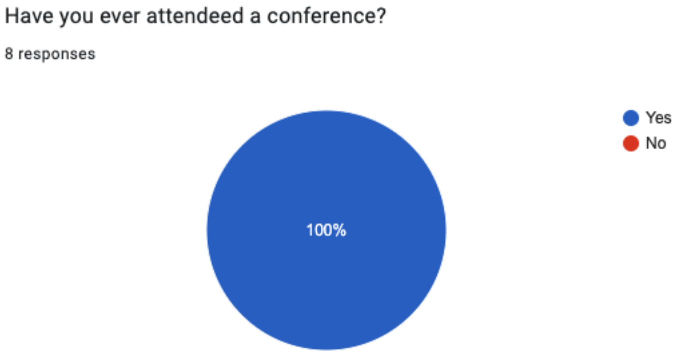


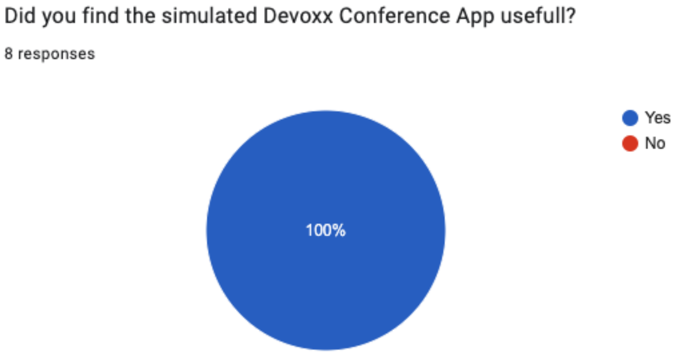
Figure 6.5: Google Form question 1.



Have you ever attendeed a conference?

8 responses

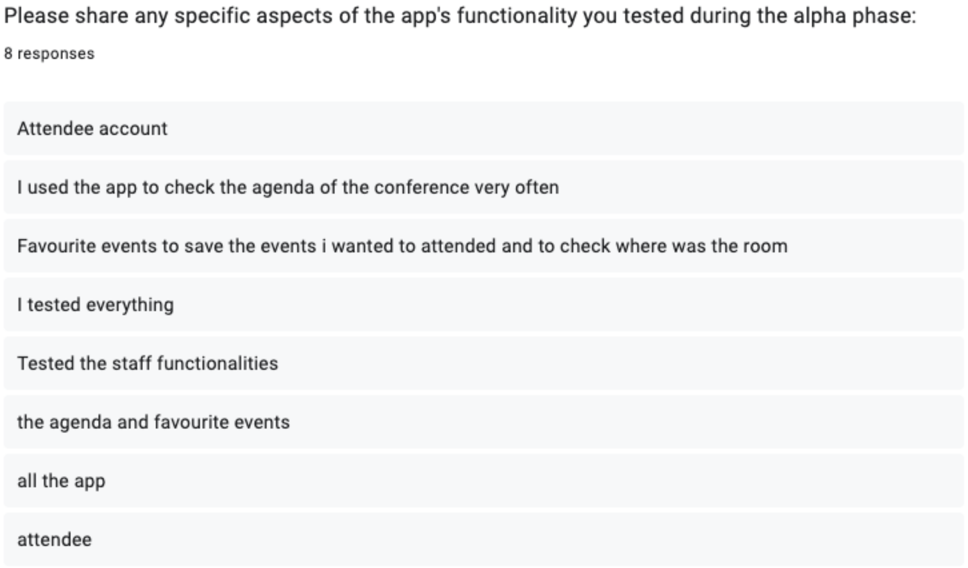
Figure 6.6: Google Form question 2.



Did you find the simulated Devoxx Conference App usefull?

8 responses

Figure 6.7: Google Form question 3.



Please share any specific aspects of the app's functionality you tested during the alpha phase:

8 responses

Attendee account

I used the app to check the agenda of the conference very often

Favourite events to save the events i wanted to attended and to check where was the room

I tested everything

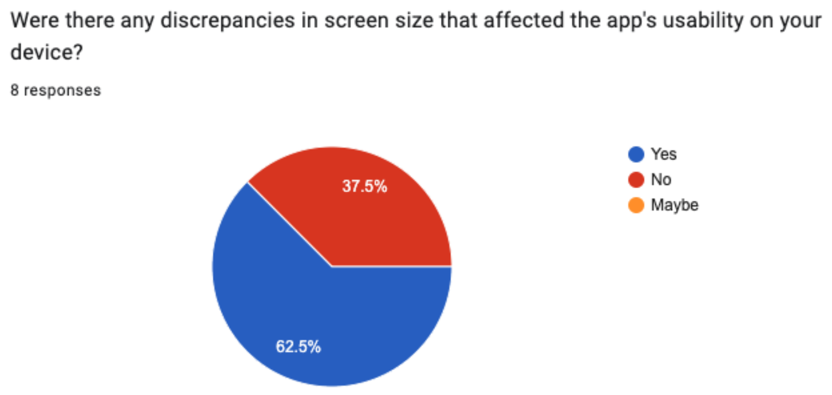
Tested the staff functionalities

the agenda and favourite events

all the app

attendee

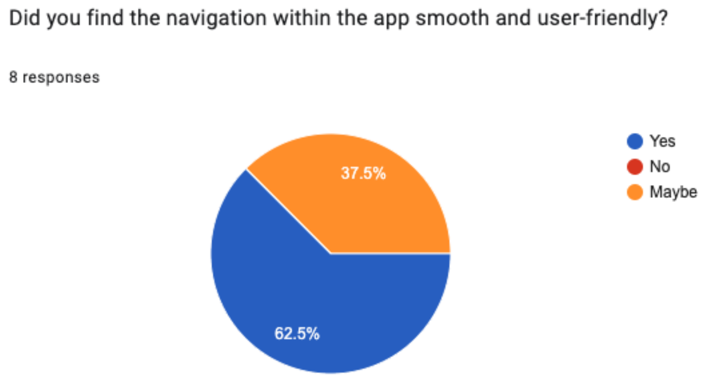
Figure 6.8: Google Form question 4.



Were there any discrepancies in screen size that affected the app's usability on your device?

8 responses

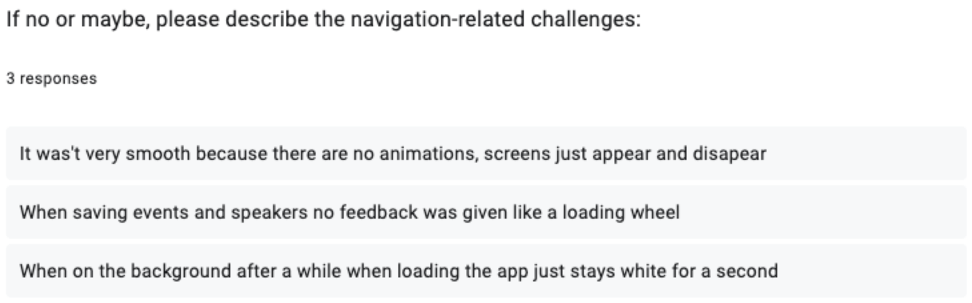
Figure 6.9: Google Form question 5.



Did you find the navigation within the app smooth and user-friendly?

8 responses

Figure 6.10: Google Form question 6.



If no or maybe, please describe the navigation-related challenges:

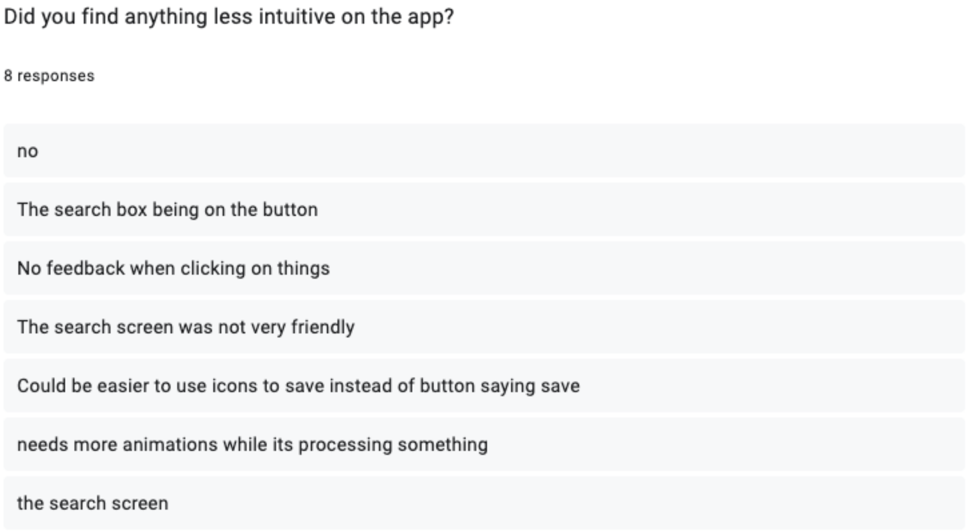
3 responses

It was't very smooth because there are no animations, screens just appear and disapear

When saving events and speakers no feedback was given like a loading wheel

When on the background after a while when loading the app just stays white for a second

Figure 6.11: Google Form question 7.



Did you find anything less intuitive on the app?

8 responses

no

The search box being on the button

No feedback when clicking on things

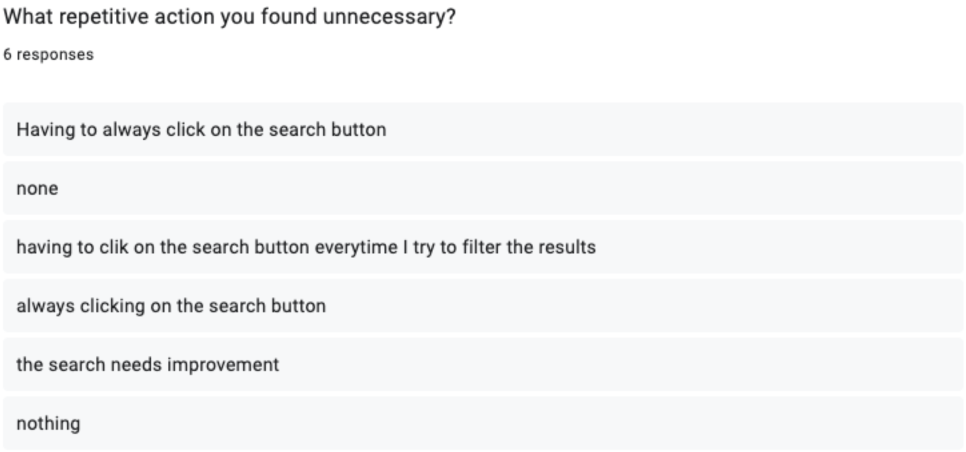
The search screen was not very friendly

Could be easier to use icons to save instead of button saying save

needs more animations while its processing something

the search screen

Figure 6.12: Google Form question 8.



What repetitive action you found unnecessary?

6 responses

Having to always click on the search button

none

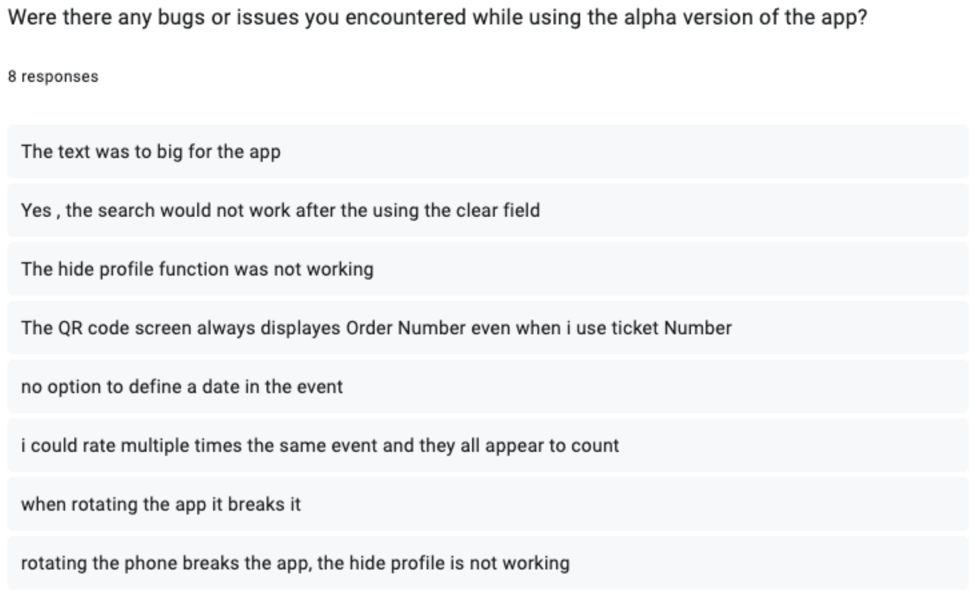
having to clik on the search button everytime I try to filter the results

always clicking on the search button

the search needs improvement

nothing

Figure 6.13: Google Form question 9.



Were there any bugs or issues you encountered while using the alpha version of the app?

8 responses

The text was to big for the app

Yes , the search would not work after the using the clear field

The hide profile function was not working

The QR code screen always displayes Order Number even when i use ticket Number

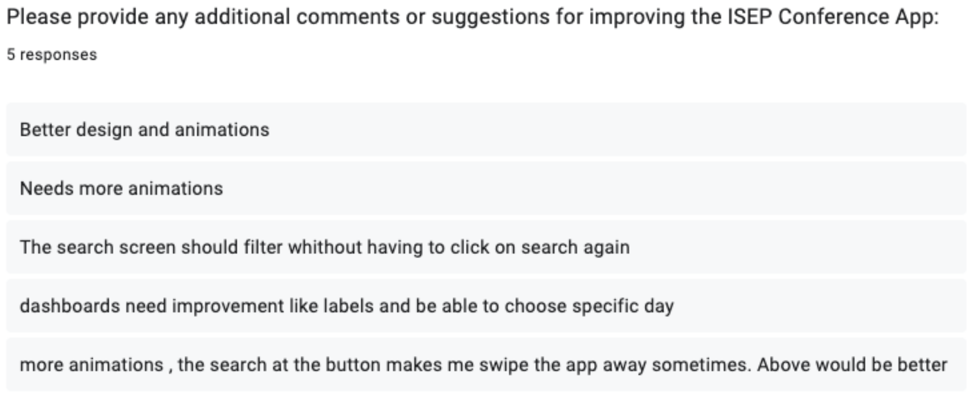
no option to define a date in the event

i could rate multiple times the same event and they all appear to count

when rotating the app it breaks it

rotating the phone breaks the app, the hide profile is not working

Figure 6.14: Google Form question 10.



Please provide any additional comments or suggestions for improving the ISEP Conference App:

5 responses

Better design and animations

Needs more animations

The search screen should filter whithout having to click on search again

dashboards need improvement like labels and be able to choose specific day

more animations , the search at the button makes me swipe the app away sometimes. Above would be better

Figure 6.15: Google Form question 11.

In addition to user feedback, objective data was gathered using the Firebase analytical framework. This tool tracked metrics such as logins, database requests, and user actions, highlighting which functionalities were most popular. This fusion of qualitative feedback and quantitative data provided a robust foundation for refining the application further.

6.3.2 Beta Testing Phase

The beta testing phase was strategically divided into two distinct stages to ensure a thorough evaluation of the application's usability, functionality, and overall user experience.

Beta Phase 1: Feature Importance Assessment

In the initial stage, a diverse group comprising colleagues and friends from varying backgrounds was engaged. These individuals, by virtue of their varied experiences and expectations, brought a wealth of perspectives to the testing process. They were prompted to brainstorm and articulate features they deemed indispensable in a conference application. This exercise aimed to ascertain if the application aligned with the general expectations and to identify potential areas that might have been overlooked.

Beta Phase 2: Usability Evaluation

Following the feature brainstorming, testers were granted access to the application. Some were given staff accounts while others were set up with attendee profiles, ensuring that the evaluation process covered multiple user scenarios. Testers were encouraged to explore the app without rigid guidelines. The idea was to discern if users could intuitively navigate through the application and easily access the information relevant to a conference.

Feedback was collected in a less structured manner, allowing for more spontaneous and candid insights. Overall, this phase of testing yielded positive outcomes. The majority of testers reported a seamless experience, successfully finding the information they sought without encountering significant hurdles. Their ability to navigate the platform with ease was a reassuring indication of the application's user-centric design and intuitive interface.

However, a recurring piece of constructive criticism pertained to the application's design aesthetics. Many testers opined that the visual design appeared too monotonous, even describing it as "bland" and "boring". This feedback highlighted the importance of not only functional but also visually appealing design, an aspect that would need attention in the subsequent development iterations.

6.3.3 Pre-Release Testing Phase

Upon reaching this crucial juncture, the application had undergone several iterations based on feedback from previous testing phases. Most of the functionalities were seamlessly integrated, with the notable exception of multi-language support, which remained unimplemented at this stage.

To conduct an exhaustive final evaluation, the application was introduced to a broader audience, specifically a group of work colleagues totaling 20 individuals. These participants were entrusted with the application for a span of three days. This testing period coincided with a small internal conference organized at the workplace, presenting a fitting scenario for evaluation. The conference was characterized by a packed schedule spanning three days, featuring multiple simultaneous events, diverse topics, and an array of speakers.

To facilitate this real-world testing scenario, the application was meticulously set up to reflect the structure, events, and dynamics of this internal conference. This approach aimed to gauge how effectively the application would support attendees in navigating such a bustling event.

Feedback channels were once again established through Google Forms and direct conversations, fostering an environment where testers felt comfortable sharing both their praises and critiques. The testing group's device distribution consisted of 8 Android smartphones and 12 iPhones, allowing for a well-rounded assessment of the application's performance across popular platforms.

Analysis of application metrics provided additional insights into user behavior. It was discerned that the 'Agenda Event Search' feature was extensively utilized, underscoring its importance to attendees aiming to plan and prioritize their schedules. Additionally, the 'Favorite' screen emerged as a favorite, with users highlighting the ability to bookmark and swiftly check the time and location of events they intended to attend. The event rating feature was commended for its utility, guiding participants in determining the value of watching event recordings.

Moreover, users complimented the application's responsiveness, particularly the swift loading times upon logging in and the expedited closing process. The streamlined design, while simpler, was praised for its intuitive layout and inclusion of all the anticipated functionalities. Lastly, the implementation of push notifications, which served as crucial last-minute updates concerning venue alterations, was lauded for enhancing real-time communication and improving the overall conference experience.

The testing phase, complemented by user feedback, played a pivotal role in shaping our application. It ensured that the product, once released, would not just meet functional requirements but would genuinely resonate with its intended user base, providing a seamless, intuitive, and reliable experience. This phase is a testament to our commitment to quality and user-centric design, underscoring the importance of real-world testing in the development process.

In the Evaluation chapter, a comprehensive exploration of the application's various dimensions – both functional and non-functional – was undertaken. This rigorous evaluation was essential in identifying the strengths and areas of improvement in the application, ensuring it not only met its intended objectives but also provided a high-quality user experience.

The Functional Dimensions section delineated the core features and functionalities of the application. These were meticulously evaluated against set criteria, and it was heartening to observe that most requirements were met to the fullest extent. Such fulfillment underscores the meticulous planning and development processes that underpinned the creation of the application.

Simultaneously, the Non-Functional Dimensions section casts a spotlight on the attributes that, while not directly contributing to the application's primary functions, play a pivotal role in its overall performance, adaptability, and user experience. Here, too, the application demonstrated considerable success, although there were areas, such as multi-language support, where further work was needed.

The Testing Phase presented the real-world evaluation of the application, moving from controlled alpha testing environments to broader beta phases, and culminating in a pre-release test. These stages were invaluable, revealing insights that might have been overlooked in a purely theoretical evaluation. Feedback from actual users, especially during real-world conference scenarios, was instrumental in refining the application to its final form. The iterative process of testing, feedback collection, and subsequent improvements ensured that the application was not only functional but also user-centric.

In conclusion, through a blend of theoretical evaluation and real-world testing, the application has been thoroughly vetted and improved upon. While there are always areas to refine and expand, the current state of the application stands as a testament to robust planning, development, and a commitment to user experience. As the application moves forward, these evaluations will serve as a foundation, guiding future enhancements and iterations.

Chapter 7

Conclusion

Embarking on this dissertation journey, the goal was clear yet ambitious: to craft a comprehensive mobile conference application designed specifically for ISEP conferences. This application, compatible with the dominant mobile Operating Systems, would transform the traditional experience of attending a scientific conference by providing an integrated platform for both attendees and staff.

In the process of achieving this objective, a door was opened to the fascinating world of mobile development. The landscape of mobile development is vast and ever-evolving, and the opportunity to delve into a new paradigm and experiment with novel frameworks was both challenging and enlightening. The intricacies of Flutter as a framework and the broader ecosystem of tools available for modern mobile development provided invaluable learning experiences. This dissertation, therefore, stands not just as a testament to a technical endeavor but also as a chronicle of academic growth and professional evolution.

ISEP, the esteemed institution where I've spent significant years of my academic journey, stands at the core of this project. It was essential for the solution presented not just to serve a theoretical purpose but to genuinely address the needs of the university. As I wrap up this phase of my academic career, it is my sincere hope that the application developed can find its place in enhancing the conference experiences at ISEP. It's not just a product of technical know-how but also a reflection of the dedication and commitment to contributing meaningfully to the university community.

All the intended functionalities were implemented fully with the exception of the automatic detection of network quality, which, upon evaluation, was deemed less critical for the application's core purpose. Additionally, significant enhancements were introduced, such as the support for multiple conferences. This feature is a game-changer, allowing the easy addition of new conferences without requiring any code changes. Furthermore, it facilitates the management of simultaneous conferences within a single application.

Even though my personal career path is geared towards becoming a backend software engineer rather than a mobile software developer, I found the challenge of creating this application to be exhilarating. It pushed me to learn an entirely new programming paradigm and navigate a multitude of different frameworks. This experience expanded my skill set and exposed me to a diverse range of programming tools and technologies.

In summation, this dissertation not only marks a significant academic achievement but also serves as a stepping stone into the expansive world of mobile application development. As the curtain falls on this chapter, the narrative of continual learning, refining, and contributing to the world of technology unfurls.

7.1 Limitations and Future Work

Every endeavor, no matter how meticulously executed, has areas that beckon further refinement. This application is no exception. While the underlying code is robust and follows good coding practices, there's room for enhancement. A touch from an expert in user interface design could elevate the user experience, ensuring that the application is not only functional but also visually captivating.

Future iterations could benefit from extensive testing, particularly focusing on the features designed for staff members. While substantial efforts were invested in ensuring a seamless experience for attendees, a more balanced testing approach can ensure holistic functionality.

Furthermore, the potential for additional functionalities remains vast. For instance, the application can be expanded to incorporate more features, such as real-time data feeds from existing projects at ISEP. Integrating sensor data from event rooms to track real-time attendance and attendee behavior could provide valuable insights and improve event management. Additionally, the following functionalities could be considered:

- **Live Polls**: Implementing real-time polling features that enable event organizers to gather instant feedback from attendees.
- **Customizable Forms**: Allowing event organizers to create customized registration forms tailored to the unique requirements of each conference.
- **Ticketing and Payment Integration**: Integrating a robust ticketing and payment system to facilitate ticket sales and event registration.
- **Social Media Integration**: Expanding the application's capabilities to seamlessly connect with social media platforms like Twitter for enhanced event promotion and attendee engagement.

These enhancements can further solidify the application's position as an indispensable tool for both conference attendees and organizers. In summation, this dissertation not only marks a significant academic achievement but also serves as a stepping stone into the expansive world of mobile application development. As the curtain falls on this chapter, the narrative of continual learning, refining, and contributing to the world of technology unfurls.

In summation, this dissertation not only marks a significant academic achievement but also serves as a stepping stone into the expansive world of mobile application development. As the curtain falls on this chapter, the narrative of continual learning, refining, and contributing to the world of technology unfurls.

References

Aboras, M. (2016). *The importance of scientific conferences in research*. https://doi.org/10.13140/RG.2.1.2637.9127

Ahmed, A., Kok, B., & Howard, C. (2020). Online community-based design of free and open source software for transgender voice training.

Aventri. (2023). *The simplest way to run all your events*.

Bouchereau, V., & Rowlands, H. (2000). Methods and techniques to help quality function deployment (qfd). *Benchmarking: An International Journal*, *7*, 8–20. https://doi.org/10.1108/14635770010314891

Brito, H., Gomes, A., & Bernardino, J. (2018). Javascript in mobile applications: React native vs ionic vs nativescript vs native development, 1–6. https://doi.org/10.23919/CISTI.2018.8399283

Campbell, B. (2023). *Guide to mobile conference apps*.

Chugh, M., & Chugh, N. (2023). A deep drive into software development agile methodologies for software quality assurance. https://doi.org/10.1002/9781119896838.ch12

Denko, B., Pecnik, S., & Fister jr, I. (2021). A comprehensive comparison of hybrid mobile application development frameworks. *International Journal of Security and Privacy in Pervasive Computing*, *13*, 78–90. https://doi.org/10.4018/IJSPPC.2021010105

Doshi, M., & Virparia, P. (2023). Agile development methodology for software re-engineering. https://doi.org/10.1007/978-981-19-9888-1_32

Dunka, B., Emmanuel, E., & Oyerinde, Y. (2017). Hybrid mobile application based on ionic framework technologies. *International Journal of Recent Advances in Multidisciplinary Research*, *04*, 3121–3130.

Google. (2023). *Keeping your app responsive*. Retrieved January 20, 2023, from https://developer.android.com/training/articles/perf-anr.html

Haider, S. A. (2023). *Mobile apps for events and conferences*.

Heidari, F., & Loucopoulos, P. (2014). Quality evaluation framework (qef): Modeling and evaluating quality of business processes. *International Journal of Accounting Information Systems*, *15*, 193–223. https://doi.org/10.1016/j.accinf.2013.09.002

Hussain, A., Leong, C., Puteh, N., & Zaaba, Z. (2019). Requirement model of conference management system: A mobile app for creating and managing conference paper. *International Journal of Interactive Mobile Technologies (iJIM)*, *13*, 197. https://doi.org/10.3991/ijim.v13i11.11280

Kaushik, V., Gupta, K., & Deepali, G. (2019). React native application development.

Kounavis, C., Zamani, E., & Giaglis, G. (2011). A mobile application supporting conference attendees and organizers. *Proceedings - 2011 Panhellenic Conference on Informatics, PCI 2011*, 333–337. https://doi.org/10.1109/PCI.2011.39

Majchrzak, T. A., Ernsting, J., & Kuchen, H. (2015). Model-driven cross-platform apps: Towards business practicability. *CEUR Workshop Proceedings*, *1367*, 129–136.

Meirelles, P., Rocha, C., Assis, F., Siqueira, R., & Goldman, A. (2019). A students' perspective of native and cross-platform approaches for mobile application development. https://doi.org/10.1007/978-3-030-24308-1_47

Napoli, M. (2019). Introducing flutter and getting started. https://doi.org/10.1002/9781119550860.ch1

Nawrocki, P., Wrona, K., Marczak, M., & Śnieżyński, B. (2021). A comparison of native and cross-platform frameworks for mobile applications. *Computer*, *54*, 18–27. https://doi.org/10.1109/MC.2020.2983893

Ozturk, A., Wei, W., Hua, N., & Qi, R. (2021). Factors affecting attendees continued use of mobile event applications. *Journal of Hospitality and Tourism Technology*, *ahead-of-print*. https://doi.org/10.1108/JHTT-03-2020-0058

Payne, R. (2019a). Developing in flutter. https://doi.org/10.1007/978-1-4842-5181-2_2

Payne, R. (2019b). Hello flutter. https://doi.org/10.1007/978-1-4842-5181-2_1

Pinto, C., & Coutinho, C. (2018). From native to cross-platform hybrid development, 669–676. https://doi.org/10.1109/IS.2018.8710545

Prasad, B. (1998). Review of qfd and related deployment techniques. *Journal of Manufacturing Systems - J MANUF SYST*, *17*, 221–234. https://doi.org/10.1016/S0278-6125(98)80063-0

Rocha, F., Misra, S., & Soares, M. (2023). Guidelines for future agile methodologies and architecture reconciliation for software-intensive systems. *Electronics*, *12*, 1582. https://doi.org/10.3390/electronics12071582

sá, M., Ferreira, C., & Serpa, S. (2019). Virtual and face-to-face academic conferences: Comparison and potentials. *Journal of Educational and Social Research*, *9*, 35–47. https://doi.org/10.2478/jesr-2019-0011

Saaty, T. (2008). Decision making with the analytic hierarchy process. *Int. J. Services Sciences Int. J. Services Sciences*, *1*, 83–98. https://doi.org/10.1504/IJSSCI.2008.017590

Sarabipour, S., Khan, A., Seah, S., David, A., Mumoki, F., Sáez, P., Schwessinger, B., Debat, H., & Mestrovic, T. (2021). Changing scientific meetings for the better. *Nature Human Behaviour*, *5*. https://doi.org/10.1038/s41562-021-01067-y

Sched. (2023). *Event scheduling software that puts the professional into your event*.

Shevtsiv, N., Shvets, D., & Karabut, N. (2019). Prospects for using react native for developing cross-platform mobile applications. *Central Ukrainian Scientific Bulletin. Technical Sciences*, 208–213. https://doi.org/10.32515/2664-262X.2019.2(33).208-213

Singh, M., & Shobha, G. (2021). Comparative analysis of hybrid mobile app development frameworks. https://doi.org/10.35940/ijsce.F3518.0710621

Sommer, A., & Krusche, S. (2013). Evaluation of cross-platform frameworks for mobile applications. *Lecture Notes in Informatics (LNI), Proceedings - Series of the Gesellschaft fur Informatik (GI)*.

Stova. (2023). *Event application strategy*.

Talantis, S., Shin, Y., & Severt, K. (2020). Conference mobile application: Participant acceptance and the correlation with overall event satisfaction utilizing the technology acceptance model (tam). *Journal of Convention Event Tourism*, *21*, 1–23. https://doi.org/10.1080/15470148.2020.1719949

Verna, A. (2008). Value network analysis and value conversion of tangible and intangible assets. *Journal of Intellectual Capital*, *9*, 5–24. https://doi.org/10.1108/14691930810845777

Whova. (2023). *All-in-one event management software*.

Woodall, T. (2003). Conceptualising 'value for the customer': An attributional, structural and dispositional analysis. *Academy of Marketing Science Review*, *12*.

Xanthopoulos, S., & Xinogalos, S. (2013). A comparative analysis of cross-platform development approaches for mobile applications. *ACM International Conference Proceeding Series*. https://doi.org/10.1145/2490257.2490292

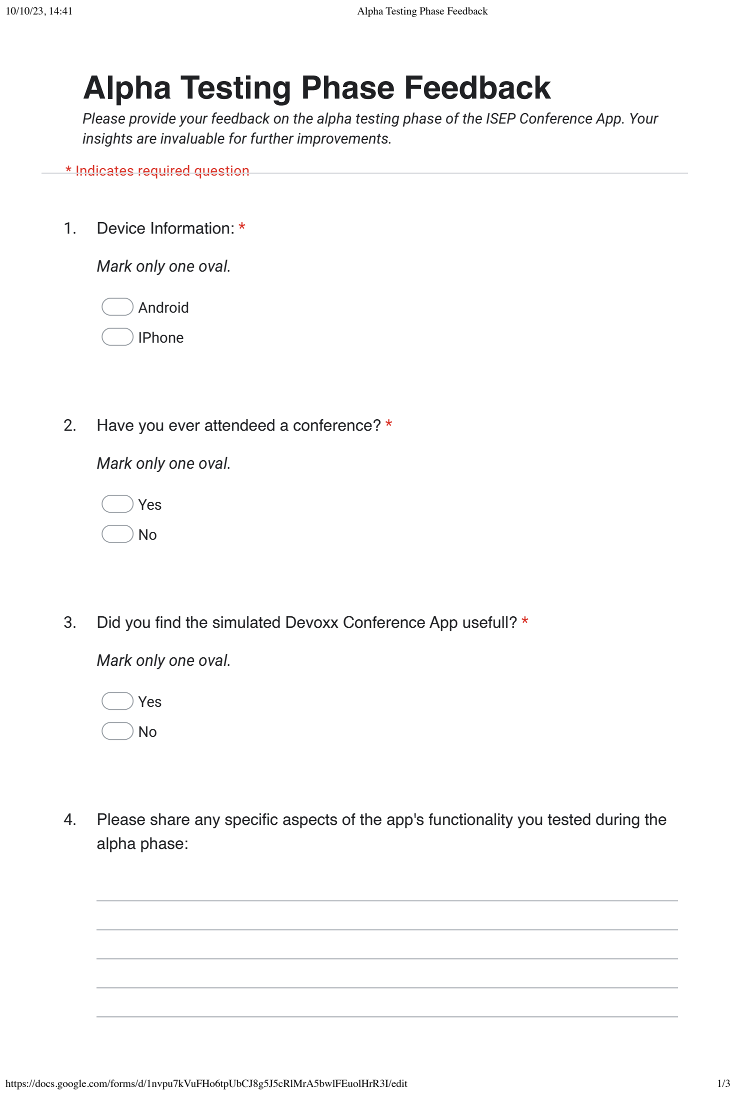
Yudin, A. (2020). Introduction to react native. https://doi.org/10.1007/978-1-4842-6333-4_5

Zhan, L., & Chiu, D. M. (2014). Mckit: A mobile app for conferences. *ACM SIGCOMM Computer Communication Review*, *44*, 59–64. https://doi.org/10.1145/2602204.2602214

Appendix A

Appendix Title Here

A.1 Alpha Testing Phase Google Form



10/10/23, 14:41 Alpha Testing Phase Feedback

Alpha Testing Phase Feedback

Please provide your feedback on the alpha testing phase of the ISEP Conference App. Your insights are invaluable for further improvements.

* Indicates required question

1. Device Information: *

   Mark only one oval.

   ○ Android
   ○ IPhone

2. Have you ever attendeed a conference? *

   Mark only one oval.

   ○ Yes
   ○ No

3. Did you find the simulated Devoxx Conference App usefull? *

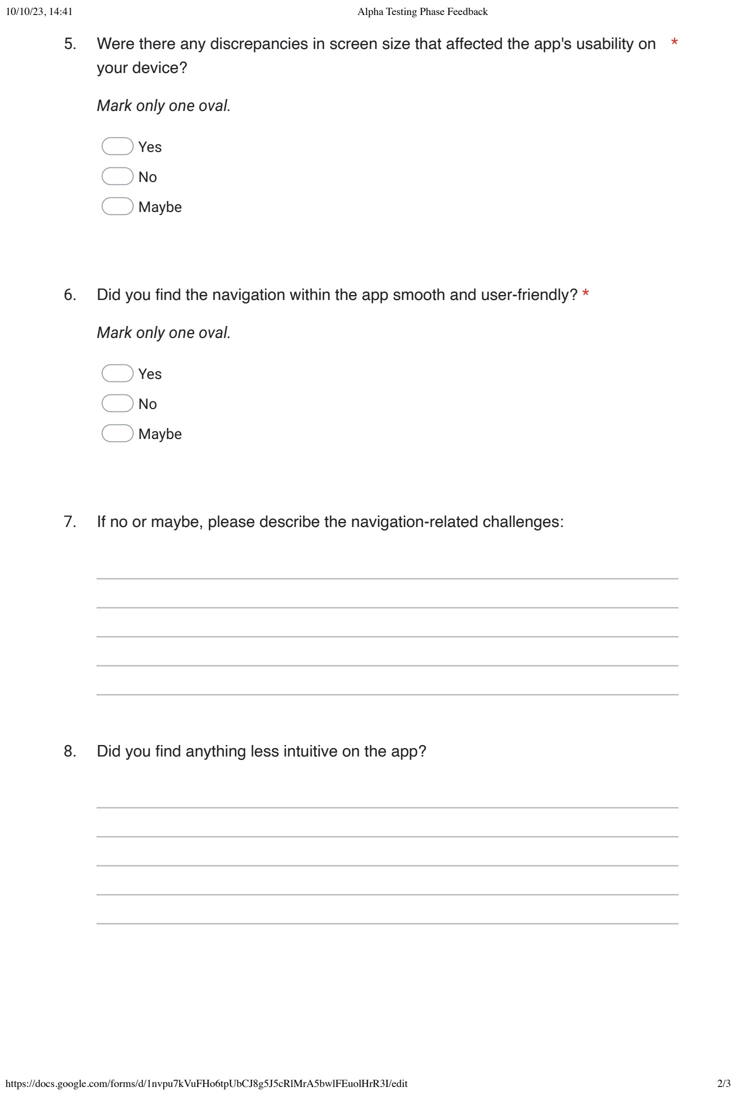
   Mark only one oval.

   ○ Yes
   ○ No

4. Please share any specific aspects of the app's functionality you tested during the alpha phase:

https://docs.google.com/forms/d/1nvpu7kVuFHo6tpUbCJ8g5J5cRlMrA5bwlFEuolHrR3I/edit 1/3

Figure A.1: Google Form questions first page.



10/10/23, 14:41 Alpha Testing Phase Feedback

5. Were there any discrepancies in screen size that affected the app's usability on your device? *

Mark only one oval.

○ Yes
○ No
○ Maybe

6. Did you find the navigation within the app smooth and user-friendly? *

Mark only one oval.

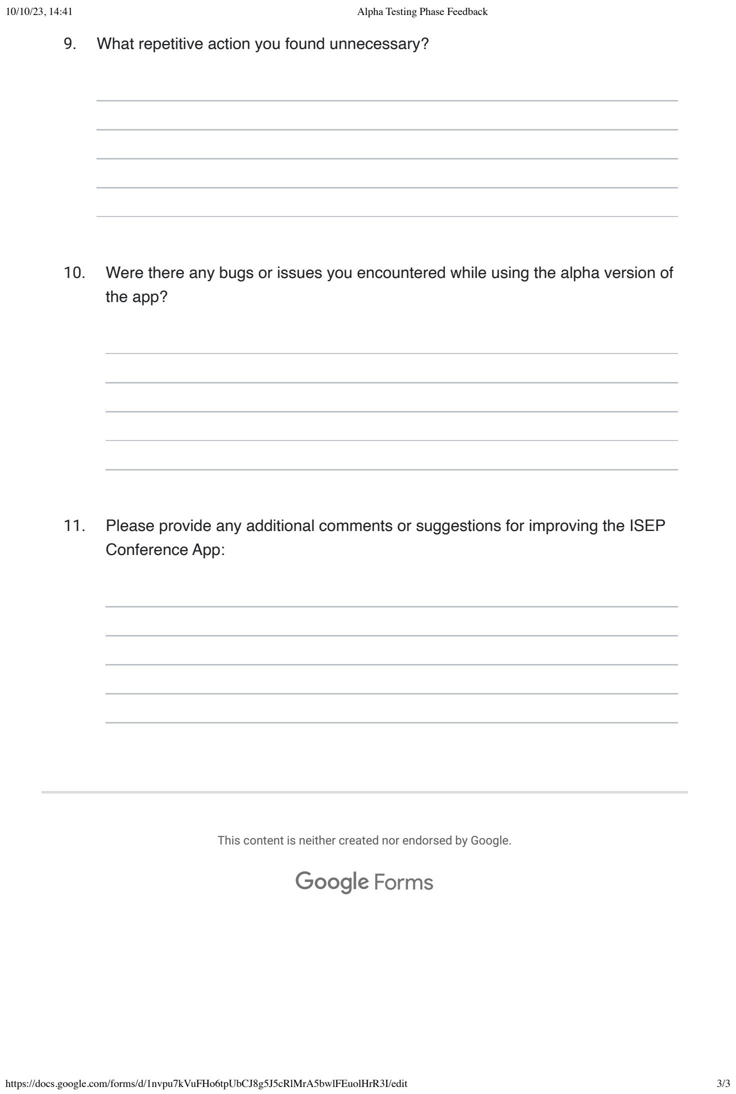
○ Yes
○ No
○ Maybe

7. If no or maybe, please describe the navigation-related challenges:

8. Did you find anything less intuitive on the app?

https://docs.google.com/forms/d/1nvpu7kVuFHo6tpUbCJ8g5J5cRlMrA5bwlFEuolHrR3I/edit 2/3

Figure A.2: Google Form questions second page.



10/10/23, 14:41 Alpha Testing Phase Feedback

9. What repetitive action you found unnecessary?

10. Were there any bugs or issues you encountered while using the alpha version of the app?

11. Please provide any additional comments or suggestions for improving the ISEP Conference App:

This content is neither created nor endorsed by Google.

Google Forms

https://docs.google.com/forms/d/1nvpu7kVuFHo6tpUbCJ8g5J5cRlMrA5bwlFEuolHrR3I/edit 3/3

Figure A.3: Google Form questions third page.