



Sistema de gestão hoteleira usando dispositivos móveis

VICENTE DE SOUSA LOUREIRO FÉLIX

Outubro de 2016

Accommodation Management System using Mobile Devices

Vicente Sousa Loureiro Félix

**Dissertação para obtenção do Grau de Mestre em
Engenharia Informática, Área de Especialização em
Sistemas Computacionais**

Orientador: Prof. Doutor Lino Manuel Baptista Figueiredo

Co-orientador: Eng.º Ricardo Manuel Soares Anacleto

Dedicatória

Dedico este trabalho à minha família e aos meus amigos.

Resumo

Os smartphones são uma realidade deste tempo, sendo fácil de verificar que têm um excelente balanço entre portabilidade, tamanho e eficiência. Esta nova realidade também permite fazer um número muito elevado de diferentes operações, operações essas que poderão facilitar a vida no quotidiano de qualquer pessoa.

Esta dissertação irá debruçar-se neste contexto e em descrever a solução proposta para resolver a necessidade de haver uma aplicação que facilitasse a estadia dum hóspede num hotel.

A solução que se pretende passa por usar smartphones para automatizar os processos numa unidade hoteleira. Utilizando um dispositivo móvel os hóspedes do hotel vão poder fazer o check-in, encomendar um serviço para um quarto, fazer reservas e até abrir portas com o telemóvel utilizando a tecnologia NFC. Também a aplicação deve permitir ao smartphone do cliente conectar-se ao quarto através de Bluetooth para que este pudesse enviar ordens ao quarto como por exemplo abrir uma janela ou ligar a luz.

A tecnologia “Near Field Communication” ou simplesmente NFC permite trocar informações entre dispositivos utilizando uma comunicação sem fios. Basta os dispositivos estarem próximos que a comunicação acontece automaticamente sem haver a necessidade de configurações adicionais. Esta tecnologia ainda é relativamente pouco utilizada embora possa ter muitas utilidades.

As informações trocadas entre os dispositivos serão direcionadas para serviços externos que irão fazer as diversas operações para as quais estão programados.

Foi desenvolvida uma aplicação para satisfazer estes critérios e com o objetivo de tornar a estadia dos clientes dos hotéis numa estadia mais moderna, mais automática e mais simples. Para além da aplicação foram desenvolvidas mais ferramentas para suporte do bom funcionamento da aplicação, sendo necessário criar base de dados, uma aplicação servidora e serviços e ter uma estrutura de testes.

Palavras-chave: Smartphones, unidade hoteleira, NFC

Abstract

Smartphones are a reality of this time. It is easy to verify that they have an excellent balance between portability, size and efficiency. This new reality also allows a very large number of different operations, these operations that can make life easier in the daily life of any person.

This work will focus in this context and describe the proposed solution to address the need for an application that would facilitate the stay of a guest in a hotel.

The solution pretended is to use smartphones to automate the processes of a hotel unit. Using a mobile phone, guests of the hotel will be able to check in, order a service to a room, make reservations and open doors with their phone using NFC technology. Also, the application will allow the customer's smartphone to connect to the room via Bluetooth so that it could send orders to the room such as opening a window or turning on the lights.

The technology "Near Field Communication" or simply NFC allows to exchange information between devices using wireless communication. If the devices are close to each other the communication happens automatically without the need of additional settings. This technology is still relatively little used although it may have many uses.

The information exchanged between devices will be directed to external services that will make the various operations for which they are programmed to.

An application was developed in order to meet these criteria and in order to make the stay of the customers in the hotel more modern, automatic and simple. In addition to the application, it was developed more tools to support the smooth operation of the application, it is necessary to create the database, a server application and services. It is also necessary to have a testing framework.

Keywords: Smartphones, hotel unit, NFC

Agradecimentos

Gostaria de agradecer bastante aos meus orientadores que sempre se demonstraram disponíveis para orientação e esclarecimento de dúvidas.

Também gostaria de agradecer ao Ricardo Pinheiro, ao Ricardo Neves e ao Francisco Areias a disponibilidade que eles tiveram para me ajudar em alguns aspetos do projeto.

Índice

1	Interpretação do Problema	1
1.1	Enquadramento	3
1.2	Objetivos do trabalho	4
1.3	Caracterização do Tema	4
1.4	Motivação.....	5
1.5	Análise de valor.....	5
1.6	Organização do relatório.....	6
2	Contexto e Estado da arte	7
2.1	Tecnologias Utilizadas.....	8
2.1.1	Android.....	8
2.1.2	REST (Representational State Transfer).....	9
2.1.3	Node.js	13
2.1.4	Base de dados.....	14
2.1.5	Pushbots	17
2.2	Introdução à tecnologia NFC	19
2.2.1	Modo de emulação de cartão	20
2.2.2	Modo de leitura/escrita	24
2.2.3	Modo de ponto a ponto.....	27
2.3	Ameaças à utilização de NFC.....	28
2.3.1	Escuta das comunicações	28
2.3.2	Man-in-the-middle	28
2.3.3	Negação de serviço	28
2.3.4	Relay Attack.....	29
2.4	Aplicações NFC.....	29
2.4.1	Controlo de acesso.....	29
2.4.2	Pagamentos	34
2.4.3	Bilheteira ou “Ticketing”	36
2.4.4	Cultura	37
2.4.5	Tarefas rotineiras	38
2.4.6	Saúde	40
2.5	Avaliar soluções e abordagens existentes.....	41
3	Design da solução	45
4	Construção da Solução	53
4.1	Fluxo de Funcionamento	53
4.1.1	MyAccommodation	53
4.2	Interação com NFC	60
4.2.1	Fluxo NFC	63

4.3	Servidor REST	64
5	Avaliação da solução.....	71
5.1	HotelCardReader	72
5.1.1	Ecrã inicial.....	72
5.1.2	Ecrã de checkin/abrir porta	72
5.1.3	API	73
5.2	MyAccommodation	74
6	Conclusões	77
ANEXO A	- Análise de Valor	85

Lista de Figuras

Figura 1 - Interface Uniforme REST [47]	10
Figura 2 - Respostas Standard da arquitetura REST [47]	11
Figura 3 - Exemplo de ligação dos diversos componentes [47]	13
Figura 4 - Exemplo duma base de dados não relacional do tipo documental	15
Figura 5 - Exemplo duma base de dados não relacional do tipo Grafo [41]	16
Figura 6 - Exemplo duma base de dados não relacional do tipo Chave/Valor [42]	16
Figura 7 - Ilustração do funcionamento de notificações – fazer o pedido [54]	18
Figura 8 - Ilustração do funcionamento de notificações – enviar a notificação [54]	18
Figura 9 - Os três modos de operabilidade NFC [16]	20
Figura 10 - Emulação do Cartão baseado num Host [17]	21
Figura 11 - Emulação dum cartão através do Elemento Seguro [17]	22
Figura 12 - Pilha de Protocolos do modo Emulação de Cartão	23
Figura 13 - Android Tags NFC	24
Figura 14 - Pilha de protocolos do modo leitura/escrita de uma tag NFC -	26
Figura 15 - Pilha de protocolos do modo Ponto-a-Ponto	27
Figura 16 – a) comunicação normal b) comunicação com man-in-the-middle	28
Figura 17 - Aplicação SEOS e leitor iCLASS [13]	30
Figura 18 - NFC Porter [15]	31
Figura 19 - Processo de atribuição de chave do quarto do Clarion Hotel através de NFC [27] .	33
Figura 20 - Processo de atribuição de chave eletrónica do sistema Key2Share	34
Figura 21 - Exemplo da utilização da Google Wallet	36
Figura 22 - Interação com NFC no Museu de Londres [33]	38
Figura 23 - NFC Task Launcher [12]	39
Figura 24 - Gráficos circulares do Inquérito do projeto da empresa ASSA ABLOY [1]	42
Figura 25 - Arquitetura do sistema	45
Figura 26 - Diagrama de fluxo da comunicação NFC	46
Figura 27 - Diagrama de Fluxo de pedidos de serviços	47
Figura 28 - Diagrama Use Case	48
Figura 29 - Base de Dados da Solução	51
Figura 30 - Diagramas de ecrã principal da aplicação MyAccommodation	54
Figura 31 - Ecrã do Login	55
Figura 32 - Ecrã do Registo	55
Figura 33 - Ecrã do Menu Principal	56
Figura 34 - Ecrã do Menu de Alteração da Palavra-Chave	57
Figura 35 - Ecrã do Menu para efetuar Reservas	58
Figura 36 - Ecrã do Menu para consultar as Reservas	58
Figura 37 - Ecrã do Menu para pedir serviços ao quarto	59
Figura 38 - Ecrã do Menu de interação com o quarto	59
Figura 39 - Ecrã do Menu para fazer ações NFC	60

Figura 40 - Ecrã do Menu para visualizar informações do Hotel	60
Figura 41 - Código no ficheiro manifesto para o funcionamento de NFC.....	62
Figura 42 - Diagrama do Fluxo de NFC do sistema.....	63
Figura 43 - Camada do servidor – server.js (primeira camada)	65
Figura 44 - Ficheiro rest.js (segunda camada).....	67
Figura 45 - Ficheiro da instanciação do recurso dos utilizadores – usersRest.js (terceira camada)	68
Figura 46 - Implementação dos serviços (quarta camada)	69
Figura 47 - Ecrã do Menu para configuração da aplicação do leitor.....	72
Figura 48 - Ecrã do Menu que fica à espera de dispositivos NFC.....	73
Figura 49 - Extrato de código do ficheiro de manifesto Android da aplicação de leitura.....	74
Figura 50 - Sequência de ecrãs de teste de fazer reserva	75
Figura 51 - Sequência de ecrãs de teste de check in.....	75
Figura 52 - Modelo Canvas	88

Lista de Tabelas

Tabela 1 - Tipos de Tags existentes [24]	26
--	----

1 Interpretação do Problema

Neste capítulo é apresentada uma introdução ao documento. Começa por ser feito um enquadramento referente à utilização e evolução de Smartphones. Depois do enquadramento é importante ainda conhecer o problema proposto, a descrição dos objetivos a atingir e qual foi a motivação para este trabalho.

Os smartphones são uma realidade deste tempo, sendo fácil de verificar que têm um excelente balanço entre portabilidade, tamanho e eficiência.

Cada vez mais os dispositivos móveis estão a fazer parte da vida das pessoas e os utilizadores usam estes dispositivos móveis para um número indeterminado de diferentes ações. Uma das mais recorrentes é para aceder a serviços e informações. Outro aspeto a realçar é que a utilização dos dispositivos móveis é feita, geralmente, de forma pessoal e intransmissível sendo um identificador da pessoa a que lhe pertence e podendo ser utilizado como acesso a alojamentos, empresas, entre outros.

Estes pequenos mas poderosos dispositivos permitem ser um canivete suíço numa certa maneira. Um canivete suíço é um canivete que tem várias funções. Estes dispositivos também têm várias funções dentro do mesmo dispositivo. Permitem fazer chamadas, mandar e receber mensagens, jogar jogos, marcar tarefas, utilizar o GPS e muitas outras ações permitidas pelas inúmeras aplicações e sensores presentes no dispositivo tais como GPS, acelerómetro, giroscópio, barómetros e muito mais. Um dos pormenores importantes é a facilidade de acesso à internet e download das aplicações.

Os Smartphones utilizam um ecrã com uma interface “touch” que permite aos utilizadores uma maior facilidade em termos de acesso ao conteúdo do telemóvel.

O problema proposto passava pela criação dum sistema de gestão hoteleira. O sistema passava por usar smartphones para que os hóspedes dos hotéis vão pudessem fazer o check-in, encomendar um serviço para um quarto, fazer reservas e até abrir portas com o telemóvel utilizando a tecnologia NFC.

Esta aplicação permitirá ainda:

1. Fazer uma reserva num determinado hotel e todas as respetivas ações sobre a reserva caso o utilizador assim o deseje:

- a. Fazer Reserva
- b. Cancelar Reserva
- c. Editar Reserva

2. Encomendar um serviço ao hotel caso o utilizador assim o deseje:

- a. Pedir alguma coisa do bar
- b. Pedir serviço de lavandaria
- c. Pedir pequeno-almoço ao quarto
- d. Ou outro serviço que o hotel disponibilizar para esta plataforma

3. Interagir com o quarto utilizando apenas o dispositivo móvel (domótica):

- a. Ligar Ar Condicionado
- b. Abrir persiana ou janela
- c. Ligar Luzes do quarto
- d. Entre outras ações

4. Operações com NFC:

- a. Abrir Portas do quarto
- b. Fazer Check-In
- c. Fazer Check-Out

Um aspeto importante a realçar no mundo dos smartphones é que mesmo estes dispositivos sendo muito poderosos faltava uma coisa importante para que estes dispositivos conseguissem suportar todo este tipo de funções numa forma mais eficaz. Um bom sistema operativo. Atualmente, existem três grandes sistemas operativos para sistemas móveis: o Windows Phone criado pela Microsoft, o sistema operativo iOS criado pela Apple e o sistema operativo Android atualmente desenvolvido pela empresa de tecnologia Google.

Como este trabalho foi realizado para Android então este relatório irá se debruçar mais sobre este sistema operativo. Este sistema operativo foi trabalhado nas aulas de mestrado e é o sistema que mais pessoas usam no mundo pelo que terá mais visibilidade [11].

O sistema operativo Android começou a ser desenvolvido pela empresa Android Inc no ano de 2003. A empresa até então Android inc, foi adquirida pela Google em 2005 e somente em 2007 é que o SO Android foi apresentado ao público.

O Android é o sistema operativo móvel mais utilizado do mundo, possuindo assim a maior percentagem de vendas mundiais de sistemas operativos móveis. O número de vendas de dispositivos com o sistema Android é superior ao número de vendas de todos os dispositivos vendidos pela Microsoft, IOS juntos. Em julho de 2013, a loja de aplicações Google Play (loja do Android) possuía mais de 1 milhão de aplicações disponíveis, sendo feitos mais de 50 mil milhões de downloads. Também por isso uma pesquisa realizada em 2013 mostra que 71% dos programadores para sistemas móveis desenvolviam para o Android [11]. Já no mês de Junho do ano de 2016, existiam 2.2 milhões de aplicações disponíveis para Android o que gerou 6 biliões de dólares americanos em conteúdo não gratuito [61].

Uma pessoa quando sai à rua pode transportar várias coisas nos bolsos mas geralmente transporta, a chave ou chaves (respetivamente do carro, de casa, etc...), o telemóvel e a carteira. E se fosse possível transportar só o telemóvel tendo neste contido a carteira e as chaves? Neste projeto abordar-se-á mais a parte das chaves e num caso específico, caso o utilizador esteja alojado num hotel e tiver que transportar a chave ou cartão do hotel consigo. Em vez disso, transportará apenas e só o seu dispositivo móvel.

1.1 Enquadramento

Os Smartphones são uma realidade deste tempo. É fácil de verificar que têm um excelente balanço entre portabilidade, tamanho e eficiência como nenhuma coisa vista antes. O facto de serem portáteis e muito poderosos fizeram deles excelentes instrumentos para qualquer pessoa.

É importante dizer que um dispositivo móvel é um instrumento pertencente a um indivíduo e que de forma geral são dispositivos intransmissíveis. Com este tipo de características, estes aparelhos permitem inúmeras aplicações e ideais para que os utilizadores possam usufruir do máximo potencial que os dispositivos móveis podem oferecer, visto que estes telemóveis para além de serem igualmente portáteis tais como um telemóvel normal, também possuem inúmeros tipos de sensores (incluindo sensores de movimento, sensores de humidade e de temperatura, sensores de proximidade, acelerómetros, giroscópio, GPS, entre outros).

O projeto baseia-se num conjunto de funcionalidades no âmbito da indústria hoteleira. Nesta indústria existe uma forte necessidade de identificação pessoal por exemplo ao reservar o quarto ou a entrar no quarto e como os dispositivos móveis têm uma forte componente neste aspeto, o resultado é uma excelente combinação.

A possibilidade de utilização dum smart card recorrendo ao nosso dispositivo móvel é uma ideia bastante entusiasmante. Um smart card, ou cartão inteligente, é um cartão que

serve como identificação pessoal, geralmente temporária, e que permite passando num leitor apropriado, identificar de forma automática a pessoa que o está a passar. Utilizando a tecnologia NFC (Near Field Communication), é possível emular um cartão que irá tornar o processo de smart card existente bem mais simples.

Os smart cards podem ser usados para inúmeros propósitos e o que foi decidido abordar foi a abertura de portas pelo que os utilizadores poderão utilizar o seu próprio telemóvel para entrar no seu quarto de hotel.

O objetivo é criar um sistema que permitisse fazer reservas, check in e check out, aceder aos serviços do hotel, entre outras funcionalidades.

É importante ainda dizer que o projeto será desenvolvido para sistemas móveis Android.

1.2 Objetivos do trabalho

Será desenvolvida uma aplicação móvel que terá que cumprir os seguintes requisitos que foram inicialmente definidos que tinham que ser colmatados:

- Utilizador poder interagir com o seu telemóvel com os leitores NFC existentes no local de alojamento tais como acesso ao quarto, fazer check-in ou controlo de luzes;
- Utilizador poder reservar e consultar suas reservas no hotel registado no servidor;
- Possibilidade de acesso aos serviços do hotel. Pequeno-almoço ao quarto, lavandaria, room-service e serviços de massagens do hotel.

1.3 Caracterização do Tema

A informática ocupa uma posição de destaque na sociedade atual, com relevância em várias e diferentes áreas. Tornou-se possível devido ao contínuo desenvolvimento de novos recursos tecnológicos ou ao aperfeiçoamento da tecnologia existente para facilitar o quotidiano da sociedade.

As tecnologias têm vindo a tomar o seu lugar e criar um papel importante na vida do ser humano a nível pessoal e principalmente a nível organizacional indiferenciadamente de nacionalidade, raça, cultura ou religião.

O Turismo representa as atividades que as pessoas realizam durante as suas viagens e permanência em lugares distintos dos que vivem, por um período de tempo inferior a um ano consecutivo, com fins de lazer, negócios e outros. No mundo atual, o turismo representa uma

parte bastante relevante da economia dum país por isso este fator é sem dúvida a ter em conta chegando o seu contributo a atingir em média de 2,9% aos 3,4% do PIB dum país a nível mundial [51, 55]. É importante referir que em Portugal o contributo é de 16,4% do PIB estando Portugal muito acima da média relativamente à média mundial [62].

Este projeto é a junção destas duas grandes realidades em que a tecnologia vai dar uma experiência muito mais agradável a nível de hotelaria pelo que vai proporcionar ao utilizador uma maneira mais simples de reservar um quarto ou de fazer check-in.

1.4 Motivação

Desde início houve uma enorme motivação para desenvolver este projeto visto que sempre gostei e já tive experiência profissional a nível hoteleiro.

A motivação para desenvolver este projeto passa também pela possibilidade de trabalhar com uma tecnologia ainda pouco conhecida, mas com bastante potencialidade e bastante importância nos tempos que correm.

Em Portugal a utilização do NFC é inferior à utilização nos Estados Unidos da América, a utilização mais conhecida do NFC já começa a ter algum empato no quotidiano do ser Humano para por exemplo servir como modo de pagamento de compras ou serviços [50]. Embora este seja o modo mais conhecido, não é ainda utilizado pela maioria das pessoas, pois existe alguma incerteza como o processo de pagamento se desenrola e se é totalmente seguro.

Parte da motivação para desenvolver este projeto é também comum a todos os projetos em que se trabalha com tecnologias potentes mas ainda pouco utilizadas como é o NFC. Ou seja, tentar sempre descobrir novos modos de utilização e também simplificar a sua utilização de modo a que não se veja o NFC como uma tecnologia impossível e difícil, mas sim como uma grande oportunidade para fazer coisas muito inovadoras e úteis para a sociedade.

1.5 Análise de valor

É importante fazer a avaliação de qual o valor que o produto poderá ter visto que em qualquer negócio, o produto irá oferecer algum ou prestar certo serviço aos possíveis clientes.

O projeto desenvolvido compreenderá uma proposta de valor para os clientes hóspedes dos hotéis. A proposta para os clientes passará por uma aplicação que permitirá aos clientes não só poder realizar ações numa forma mais facilitada como também interagir de forma diferente com a unidade hoteleira que estes estão hospedados.

Em anexo estará presente uma secção a explicar melhor este tema.

1.6 Organização do relatório

Esta dissertação é composta por seis capítulos e as referências.

No primeiro capítulo é feita uma introdução e onde é interpretado o problema. Neste capítulo dá-se a conhecer o problema, o seu contexto e também uma versão resumida da sua análise de valor.

O segundo capítulo é referente ao contexto e ao estado da arte, sendo dadas a conhecer outras abordagens nesta área. Neste capítulo é detalhado em mais pormenor o contexto e o problema. Também este capítulo é dedicado a avaliar as abordagens previamente ditas do capítulo dois.

No terceiro capítulo é abordada a arquitetura do projeto. Neste capítulo serão apresentados diagramas e esquemas para uma melhor descrição da solução proposta ao problema.

O quarto capítulo relata a construção da solução, ou seja, como o sistema foi implementado.

O quinto capítulo apresenta os testes que foram feitos à aplicação para assegurar o seu bom funcionamento.

No capítulo seis são descritas as conclusões da dissertação e também as dificuldades que foram encontradas.

No final da dissertação estão as referências da dissertação.

2 Contexto e Estado da arte

Neste capítulo irá ser relatado aquilo que já existe na área em que o projeto se insere, neste caso aplicações de gestão de unidades hoteleiras e aplicações com NFC. Também serão expostas as diversas utilizações que um dispositivo NFC deve ter. Posteriormente, vão ser referidas todas as tecnologias utilizadas neste projeto, incluindo o NFC.

Como se sabe, o ser humano tem muita influência no mundo. Foi responsável por diversas ideias revolucionárias e avanços tecnológicos notáveis. A habilidade de construir prédios e monumentos de grande dimensão e de coabitar em comunidade com os outros é deveras admirável.

Embora o ser humano tenha sido capaz de grandes feitos também com ele surgiram vários problemas e, necessariamente, medidas para colmatar esses problemas. Falando mais especificamente da segurança e privacidade das nossas casas o ser humano rapidamente sentiu que precisava de ter medidas de proteção para as suas casas. Então surgiu a ideia da chave e fechadura. Chave que era um objeto que estava na posse da pessoa com permissão de acesso e a fechadura como interface entre a chave e a casa.

O sistema de chave e fechadura têm algumas vantagens mas também têm algumas desvantagens. A vantagem deste sistema é que é um sistema simples e que todas as pessoas já conhecem. Para além disso é um sistema em que a chave é uma coisa difícil de roubar ao utilizador se este tiver cuidado. A desvantagem é que o utilizador tem que andar com um objeto a mais no bolso, para além de que se perder a chave não pode entrar no sítio desejado.

A chave metálica e a fechadura ainda constituem uma percentagem avassaladora no método de controlo de acesso mas, nos tempos que ocorrem, começasse a notar uma mudança a nível hoteleiro. Em vez destes métodos tradicionais, os hotéis têm cada vez mais aderido a um sistema de chaves diferente, que consiste num cartão (chave) e num leitor de cartões (fechadura). O cartão para substituir a chave que servirá para o utilizador que o possui

poder mostrar a sua identidade, e o leitor de cartões para ler e verificar se o utilizador tem a permissão necessária para poder entrar no quarto.

Com a constante evolução tecnológica tem começado a aparecer, embora ainda seja em número muito reduzido, algumas alterações nesta matéria de chaves e fechaduras. Com o grande desenvolvimento tecnológico também houve uma grande evolução a nível de sistemas móveis, e está a começar a ser possível utilizar esses sistemas móveis como forma de chave. O utilizador possui um dispositivo móvel que servirá de chave e a porta dum designado hotel terá um leitor que fará a verificação do dispositivo móvel. Desde que o dispositivo móvel possua uma placa NFC o telemóvel pode ser utilizado neste contexto.

De seguida irão ser explicadas que tecnologias foram utilizadas para colmatar o problema descrito.

2.1 Tecnologias Utilizadas

Nesta secção são descritos em mais pormenor as diferentes tecnologias utilizadas. Será feita então a descrição do SO (Sistema Operativo) Android. Posteriormente, irá ser explicado a tecnologia Node.js que serve para fazer os serviços que irão ser chamados e a arquitetura REST (Representational State Transfer). A arquitetura Rest é um estilo de representação que para além de permitir uma melhor representação dos serviços, também permite ao sistema ter melhor performance no que diz respeito à sua escalabilidade. Será feito uma menção ao nível da base de dados e posteriormente também será relatado o sistema de controlo de versões utilizado de modo a registar as alterações que foram feitas e guardar tudo num servidor externo.

2.1.1 Android

Android é um sistema operativo para dispositivos móveis, sendo desenvolvido e projetado principalmente para dispositivos móveis de touchscreen como tablets ou smartphones. Embora atualmente seja desenvolvido pela empresa de grande renome Google, o Android foi criado por uma empresa chamada Android, Inc. no ano de 2003. Quando houve a criação deste sistema operativo, as intenções iniciais era o desenvolvimento dum sistema operativo avançado para câmaras digitais, mas com os estudos de mercado a revelar que esse mercado não era grande o suficiente decidiu-se então que a empresa passaria a desenvolver um sistema operativo para dispositivos móveis. Este sistema operativo tinha então a interface baseada principalmente na manipulação direta através de gestos de toque, tais como deslizar, tocar, para que se pudesse manipular objetos presentes no ecrã do dispositivo [43].

É importante realçar que o sistema operativo Android está disponível em 70 idiomas diferentes e que foi o sistema operativo móvel que experienciou a maior taxa de crescimento de todos sendo que no ano de 2009 apenas 2,8% dos smartphones vendidos usavam o sistema operativo Android, no final do ano 2010 esta percentagem já era de 33%, no final do

ano de 2011 os dispositivos Android já faziam parte de mais de metade dos dispositivos vendidos atingindo um valor de 52,5% e finalmente no final do ano de 2012 a quota do mercado de vendas já chegara aos 75%. Estes números foram resultados de pesquisas realizadas pelas empresas Canalys, Gartner e IDC [43]. Mais atualmente no ano de 2015, a percentagem de utilização do sistema operativo Android chegou aos 82.8%, o que é bastante significativa [52].

O SO Android consiste numa pilha de softwares para dispositivos móveis, um middleware e um conjunto de aplicações-chave, tais como calendário, browser, contactos entre outras. As aplicações normalmente são escritas usando a linguagem de programação Java e executam sobre uma máquina virtual desenvolvida para o sistema operativo Android. Para desenvolver aplicações para esta plataforma é necessário o Software Development Kit (SDK) e conhecer a API (Application Programming Interface). O SDK consiste num conjunto de ferramentas de desenvolvimento contendo documentação, tutoriais, um emulador entre outras funcionalidades [19].

As aplicações Android são baseadas em activities. Cada ecrã numa dada aplicação Android tem o nome de activity, visto que é uma extensão da classe Activity. As activities têm ciclos de vida: Uma activity pode estar ativa, pausada, parada ou inativa.

Uma activity está ativa ou active quando está visível para o utilizador e quando está no topo da fila de activities. Pode dizer-se que uma activity está ativa quando está à espera de algum input do utilizador. Importante referir que o sistema Android vai tentar manter viva esta activity a todo o custo. Quando outra activity passa ao estado ativa então esta será pausada.

O estado de pausa ou paused é quando uma activity está visível e até pode ter uma thread a correr também mas não tem o foco do utilizador nem está à espera do seu input. Quando esta activity perde o foco então fica no estado de parada.

O estado de parada ou stopped é quando uma activity perde o foco do utilizador. A activity mantém-se em memória retendo a informação da mesma. Quando a activity é fechada torna-se inativa.

Uma activity está inativa ou inactive é quando a activity deixa de estar à espera de input do utilizador. As activities inativas são removidas da fila e precisam de ser reiniciadas antes de se mostrarem ao utilizador [64].

2.1.2 REST (Representational State Transfer)

Representational State Transfer (REST), ou em português, Transferência de Estado Representacional, é uma abstração da arquitetura da World Wide Web (WWW). A arquitetura REST é um estilo arquitetural que consiste num conjunto coordenado de restrições arquiteturais aplicadas a componentes, conectores e elementos de dados dentro de um sistema de hipermédia distribuído [47].

O estilo arquitetural de REST também é aplicado no desenvolvimento de serviços Web. Os web services podem-se caracterizar como "RESTful" se eles estiverem em conformidade com as restrições descritas na seção restrições arquiteturais.

A arquitetura rege-se por cinco princípios fundamentais: dar a todos os recursos um ID, usar métodos standards, fornecer representações múltiplas, comunicar sem estados e ligar os diversos componentes.

Dar a todos os recursos um ID:

Itens individuais :

<http://example.com/customers/1234>

<http://example.com/orders/2007/10/776654>

<http://example.com/products/4554>

<http://example.com/processes/salary-increase>

Coleções de Itens :

<http://example.com/orders/2007/11>

<http://example.com/products?color=green>

Usar métodos Standard como se fosse uma interface uniforme tal como se pode observar na Figura 1:

Interface Uniforme			
Método	Descrição	Seguro	Idempotente
GET	Faz o pedido dum recurso	Sim	Sim
PUT	Cria ou atualiza um recurso	Não	Sim
DELETE	Apaga um recurso	Não	Sim
POST	Submete a informação para ser processada por aquele recurso em específico	Não	Não

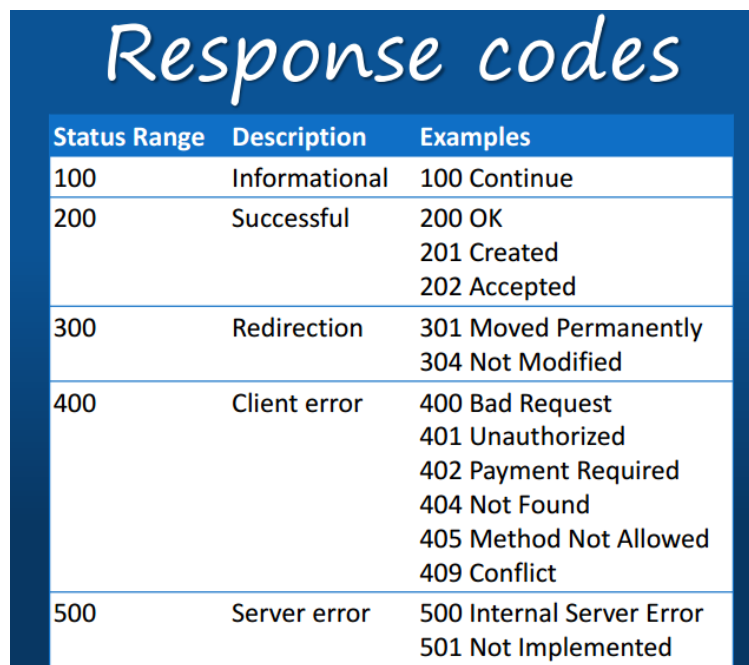
Figura 1 - Interface Uniforme REST [47]

Os nomes dos métodos encontram-se no lado esquerdo da imagem, ao centro a sua descrição e à direita encontram-se uma designação para cada um dos métodos se estes são Seguros e/ou Idempotentes.

Neste contexto, um método é seguro se este não modificar recursos como por exemplo o método GET.

Neste contexto, um método é idempotente se, por mais vezes que invocarmos o método o resultado não vai ser alterado.

Os códigos de resposta Rest são standards que podem ser observados na Figura 2.



Status Range	Description	Examples
100	Informational	100 Continue
200	Successful	200 OK 201 Created 202 Accepted
300	Redirection	301 Moved Permanently 304 Not Modified
400	Client error	400 Bad Request 401 Unauthorized 402 Payment Required 404 Not Found 405 Method Not Allowed 409 Conflict
500	Server error	500 Internal Server Error 501 Not Implemented

Figura 2 - Respostas Standard da arquitetura REST [47]

Fornecer múltiplas representações. A arquitetura REST pode ser representada de múltiplas formas. Entre elas estão:

- XML
- JSON
- YAML
- XHTML
- ATOM
- E muitas outras.

A comunicação stateless também é uma aspeto muito importante desta tecnologia. A comunicação stateless é devido ao fato de no servidor não ser guardado o estado da sessão. Um exemplo em relação a este tipo de comunicação vai ser dado de seguida através dum diálogo entre um cliente e um servidor:

[1.Cliente:] Mostra-me os produtos.

[1.Servidor:] Aqui está a lista de todos os produtos.

[2.Cliente:] Gostaria de comprar o produto A, eu sou o João e a minha password é xyz.

[2.Servidor:] Produto A adicionado ao carrinho do João.

[3.Cliente:] Gostaria de comprar o produto B, eu sou o João e a minha password é xyz.

[3.Servidor:] Produto B adicionado ao carrinho do João.

[4.Cliente:] Não quero comprar o produto A, eu sou o João e a minha password é xyz.

[4.Servidor:] Produto A removido do carrinho do João.

[5.Cliente:] Ok, já terminei. O meu nome é João e a password é xyz.

[5.Servidor:] Aqui está o total relativamente ao carrinho do João.

Como se pode verificar no diálogo, na fala 1 do cliente este apenas pede para o servidor mostrar todos os produtos. O servidor mostra pois é uma informação que não precisa de autenticação.

Posteriormente o cliente, neste caso o João, faz alguns pedidos ao servidor. Como a comunicação em REST é stateless então o cliente tem que enviar sempre qual é o seu username e a sua password pois o servidor não guarda essa informação na sessão. Isto é que quer dizer stateless.

Ligar os diversos recursos. Como se pode observar na Figura 3, existe uma encomenda (order) e a essa encomenda estão associados vários recursos, ou seja, esses recursos estão ligados à encomenda, sendo estes: a quantidade (amount), o produto (product), o cliente (customer) e a transportadora (dispatcher).

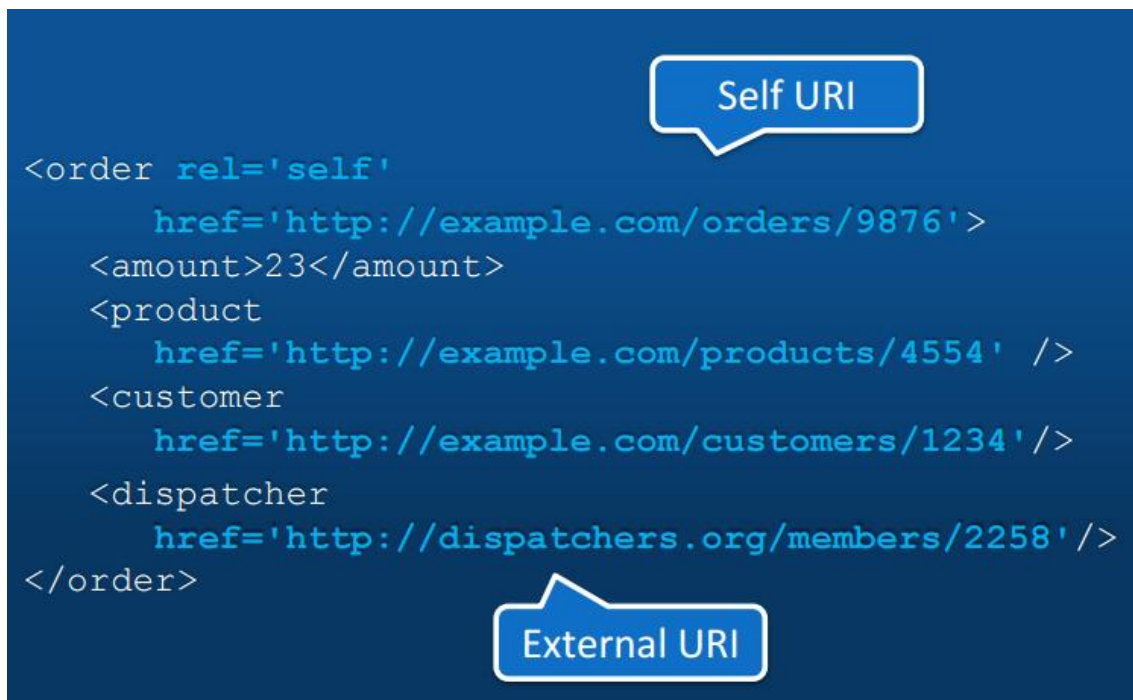


Figura 3 - Exemplo de ligação dos diversos componentes [47]

2.1.3 Node.js

Node.js é uma plataforma construída sobre o motor JavaScript do Google Chrome para facilmente construir aplicações de rede rápidas e escaláveis. Node.js usa um modelo de I/O direcionada a eventos não bloqueantes, o que o torna leve, eficiente e ideal para aplicações em tempo real com troca intensa de dados através de dispositivos distribuídos.

Numa conferência que aconteceu no ano de 2009, um jovem programador chamado Ryan Dahl, apresentou um projeto em que estava a trabalhar. Ryan deu a conhecer o projeto que consistia numa plataforma que combinava a máquina virtual JavaScript V8 da Google e um conjunto de eventos. O projeto, denominado de Node.js, apontava para uma direção diferente das outras plataformas em JavaScript que rodam no servidor: todos I/O primitivos são orientados a eventos. Aproveitando o poder e a simplicidade do Javascript, isso tornou tarefas difíceis de escrever aplicações assíncronas em tarefas fáceis.

Curiosamente a inspiração de Ryan Dahl para criar o Node.js foi o fato de ter visto a barra de progresso do upload de arquivos no Flickr, website de upload de fotos. Ele percebeu que o navegador não sabia o quanto do arquivo já tinha sido carregado e tinha que consultar o servidor web.

O Node.js foi criado para correr num servidor http dedicado e para que uma thread conseguisse estar responsável por um processo de cada vez. Aplicações em Node.js são baseadas em eventos e correm numa forma assíncrona. O Node não foi construído para seguir o modelo tradicional de receber, processar, enviar, esperar e receber. Em vez disso, os

pedidos dos processos Node ficam numa pilha e são enviados pequenos pedidos uns atrás dos outros sem esperar pelas respostas de cada um.

Esta é uma mudança muito rápida dos modelos dominantes que correm processos maiores e mais complexos e executam várias threads de forma simultânea. E cada thread espera a sua resposta apropriada antes de prosseguir com a sua execução.

Uma das principais vantagens do Node.js, de acordo com o seu criador Ryan Dahl, é que não bloqueia a entrada/saída (I/O) de dados.

Os desenvolvedores críticos do Node.js afirmam que um único processo requer um número significativo de ciclos do CPU, podendo assim a aplicação bloquear. Os defensores do modelo Node.js afirmam que o tempo de processamento do CPU é uma preocupação menor por causa do alto número de pequenos processos em que o código Node é baseado.

2.1.4 Base de dados

Base de dados são conjuntos de arquivos que têm relações entre si de forma a criar informação para representar o problema proposto e assim poder dar mais eficiência ao produto final.

Dentro das base de dados existem as:

- Base de Dados Relacionais
- Base de Dados não Relacionais

Uma base de dados relacional modela os dados numa forma que estes sejam percebidos pelo utilizador. Os dados permanecem em tabelas e as tabelas têm relações entre si.

As bases de dados relacionais surgiram em 1970 e foram desenvolvidas para facilitar o acesso aos dados, possibilitando assim aos utilizadores que utilizem uma grande variedade de abordagens no tratamento de informações. A linguagem padrão para os utilizadores puderem fazer perguntas relacionadas com os negócios é a Structured Query Language, mais conhecida como SQL.

Pontos fortes numa Base de Dados Relacional:

- Este tipo de base de dados oferece aos utilizadores possibilidade de validação, verificação e garantias de integridade de dados;
- Controlo de concorrência, recuperação de falhas e segurança;
- Controlo sobre as transações;
- Otimização de consultas.

Pontos fracos duma Base de Dados Relacional:

- Dificuldade em manter o mesmo modelo quando o sistema aumenta o seu tamanho – escalabilidade;
- Dificuldade na organização de dados quando se fala dum sistema distribuído trabalhando com particionamento de dados.

As bases de dados não relacionais são livres de qualquer tipo de esquemas e promovem a disponibilidade e escalabilidade. Duma forma muito simples são qualquer base de dados que não adote o paradigma relacional. Este tipo de base de dados é também denominado de NoSQL, que é o contrário de SQL das bases de dados relacionais.

Dentro das bases de dados não relacionais existem três categorias:

- Documental – Este tipo é talvez o mais popular das bases de dados não relacionais. Caracteriza-se por não haver tabelas com um número fixo de campos bem definidos, mas ao invés disso existem coleções. Cada coleção pode armazenar um ou mais documentos (no lugar de registros numa tabela). Ou seja, um documento é apenas um agregado de atributos que não possui uma regra rígida que define quais os tipos de cada atributo e qual tipo cada um deve possuir (são sem esquema, *schemaless*). Podemos observar um exemplo deste tipo de base de dados na Figura 4.

```
// Um documento usado para representar uma pessoa
{nome:"Ricardo", sobrenome:"Neves", apelido:"Isep Soldier", cidade:"Porto"}

// Um outro documento para representar uma pessoa na mesma coleção
{nome:"Luis", sobrenome:"Lima ", profissao:"Professor", gatos:[Micas,Rex]}
```

Figura 4 - Exemplo duma base de dados não relacional do tipo documental

- Baseada em Grafos – Talvez a menos popular das categorias, este modelo tem como foco o relacionamento entre os objetos. A metáfora neste caso é o grafo. Não existem coleções, apenas vértices representando os objetos de sistema em questão e arestas indicando o relacionamento entre estes. Normalmente cada vértice representa um agregado de atributos, muito similar ao modelo documental. A pesquisa pode ser feita tanto por algum dos atributos dos vértices quanto pelos relacionamentos entre os mesmos. Neste tipo de base de dados podemos fazer pesquisas do tipo “amigo do amigo”. A aplicação mais óbvia é a sua lista de contatos em uma rede social. Outro exemplo interessante poderia ser a representação de alguma estrutura do tipo causa/efeito, como uma fábrica, em que cada etapa de produção gera a entrada para uma ou mais etapas que precisem ser monitoradas [41]. Podemos observar um exemplo deste tipo de base de dados na Figura 5.

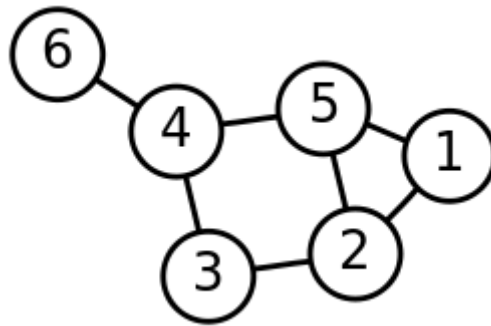


Figura 5 - Exemplo dum base de dados não relacional do tipo Grafo [41]

- Chave/valor - Trata-se da categoria mais simples e que talvez traga os maiores ganhos de performance quando bem aplicada. Neste modelo, toda consulta à base de dados é feita apenas através de uma chave, que pode ou não ter algum valor (de qualquer tipo) relacionado. É muito usado na implementação de caches de sistema ou acesso a informações que são alteradas em tempo real. De todos os modelos, é o que possui tempo de pesquisa mais curto: basicamente é uma pesquisa baseada no hash da chave. É o mais simples e é o que tem melhor performance, mas no entanto, de todos os modelos é também o menos confiável sob o ponto de vista de persistência, pois são normalmente desenvolvidos para armazenar informações em memória, com pouco foco na garantia de persistência [42]. Podemos observar um exemplo deste tipo de base de dados na Figura 6.

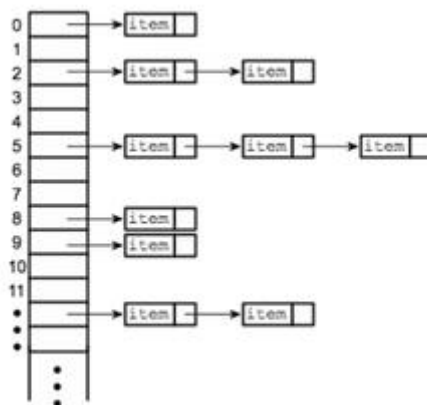


Figura 6 - Exemplo dum base de dados não relacional do tipo Chave/Valor [42]

O MySQL Workbench é um programa que permite a um programador ou arquiteto de dados, desenhar, modelar, gerar e criar base de dados [8]. O MySQL Workbench oferece ferramentas visuais de desenho para criar, executar e otimizar as consultas SQL que se pretende fazer ou que se pretende que os serviços desenvolvidos façam. O editor de SQL fornece destaque em termos de cor consoante a sintaxe para que o utilizador que o esteja a

utilizar faça uma leitura mais clara, também oferece a funcionalidade de auto-complete, que serve como auxílio ao programador para que este se guie mais facilmente quando está a escrever em SQL. Este programa também permite reutilizar trechos SQL e tem um histórico de execução de SQL para que o utilizador não se esqueça do que já foi programando de maneira a que este possa escrever com mais organização e clareza. O painel de conexão de base de dados permite que os desenvolvedores façam uma gestão mais facilitada das conexões existentes entre as diferentes tabelas.

2.1.5 Pushbots

PushBots é um sistema de notificação escalável e confiável para aplicações móveis que ajuda a economizar tempo e esforço para comunicar de forma eficaz com seu público. O sistema da PushBots oferece SDKs nativos, API REST, clientes API e um ambiente de trabalho online para gerir e enviar as notificações desejadas [48]. Este sistema é gratuito até cerca de mil e quinhentas notificações mensais o que para este projeto era suficiente. Esta tecnologia é utilizada para serem enviadas notificações quando o utilizador faz login, quando faz checkin e quando abre a porta do hotel para informar o utilizador que este conseguiu efetuar as suas ações com sucesso.

A intenção dum sistema de notificações é que o utilizador volte sempre para a aplicação pela qual as notificações estão a ser enviadas.

Numa primeira parte do processo o utilizador vai fazer um pedido de algum serviço e ao fazê-lo o dispositivo vai enviar a informação necessária para o pedido assim como um pedaço de informação que é o identificador do dispositivo para que depois o servidor saiba para qual dispositivo ele terá que enviar a informação. Esse pedaço de informação é denominado de notification token e vai ser explicado mais à frente. Este processo está bem patente na Figura 7.

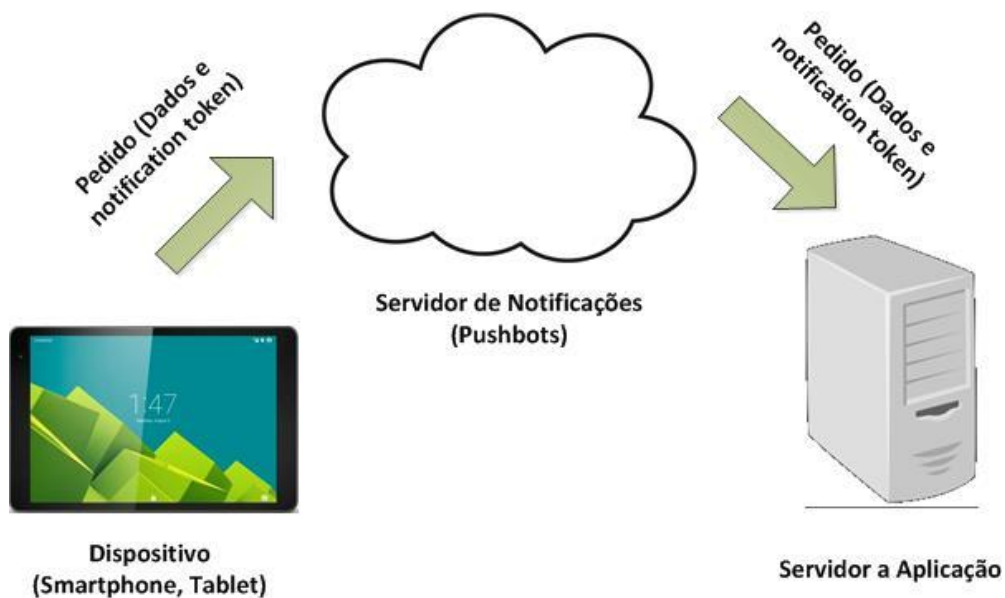


Figura 7 - Ilustração do funcionamento de notificações – fazer o pedido [54]

Posteriormente o servidor vai processar os dados e ao obter a resposta vai comunicar ao servidor de notificações que neste caso é o Pushbots enviando a informação necessária para responder ao pedido feito assim como o notification token para que o servidor de notificações saiba para que dispositivo tem que enviar a notificação. Posteriormente o servidor irá enviar a notificação ao dispositivo inicial tal e qual mostra a Figura 8.

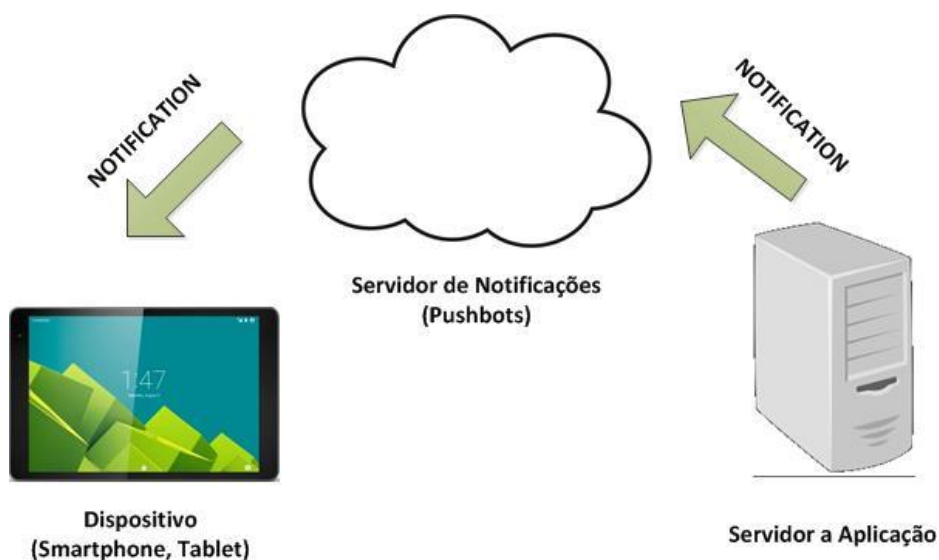


Figura 8 - Ilustração do funcionamento de notificações – enviar a notificação [54]

2.1.5.1 Notification Token

O Notification Token é o elemento que permite ao servidor saber qual o dispositivo para qual o programador pretende que seja enviada a notificação. Então este elemento é bastante útil pois o dispositivo que vai fazer os pedidos ao servidor é o leitor de NFC, não

sendo este o dispositivo para qual é desejado que seja enviada a informação, e sim para o dispositivo inicial.

Este componente pode ser obtido através dum método disponibilizado pela API do Pushbots e pode ser visto a seguir:

- “Pushbots.sharedInstance().regID();”

Este método irá então retornar a variável identificativa do dispositivo em questão que posteriormente será enviada através dos pedidos para que depois todas as outras componentes saibam a origem do seu pedido.

Para efetuar o login é apenas preciso que seja enviado o email, a password e o notification token.

Para efetuar o checkin e o abrir porta é necessário o ID do utilizador e o notification token.

2.2 Introdução à tecnologia NFC

O NFC significa em inglês Near Field Communication, que em português pode ser traduzido para Comunicação por Campo de Proximidade, é uma tecnologia que permite que dois dispositivos que estejam próximo um do outro possam trocar informação entre si numa comunicação sem fios, utilizando ondas eletromagnéticas.

Basta os dois aparelhos estarem perto um do outro que a comunicação é estabelecida automaticamente sem a necessidade de configurações adicionais nem de qualquer evento que acione uma ação.

A tecnologia RFID (Radio-Frequency Identification) é um método de identificação automática através de sinais de rádio. Foi desenvolvida por uma equipa de cientistas liderados por Charles Walton e é a base do funcionamento do NFC. Esta tecnologia inovadora encontra-se atualmente normalizada através da norma ISO/IEC 18092. Esta norma é também conhecida por NFC Interface and Protocol (NFCIP-1).

É importante realçar que o NFC permite a comunicação entre dois dispositivos até uma distância máxima de 4 cm e com uma velocidade máxima de transmissão de 424 Kbps.

Existe um fórum [23] que contém todas estas especificações e boas práticas para a utilização de NFC. O objetivo deste fórum é estimular a utilização do NFC através do desenvolvimento de especificações, certificar a interoperabilidade entre os dispositivos e serviços, assegurar a segurança entre as comunicações e ainda educar o mercado sobre as potencialidades da utilização da tecnologia NFC.

Podem existir até três componentes no processo de comunicação por NFC: o Controlador NFC, o elemento seguro (não obrigatório) e o CPU. O controlador NFC é o

componente principal da funcionalidade NFC enquanto que o elemento seguro é um microchip cuja finalidade é garantir a segurança de dados privados, tais como informações de cartões de crédito ou números de identificação pessoais. O CPU (Central Processing Unit) do dispositivo é o principal processador do dispositivo NFC.

Os três modos de funcionamento são [21]:

- Modo de emulação de cartão: No modo de emulação de cartão, um dispositivo pode emular um smart card e assim ser capaz de comunicar com leitores existentes de RFID de forma a dar a um dispositivo móvel a possibilidade de ter novas funções;
- Modo de leitura e escrita: Neste modo os dispositivos têm a possibilidade de acesso a smart cards e tags NFC para poderem fazer uma leitura ou escrita e despoletarem um conjunto designado de funções;
- Modo P2P (Ponto-a-Ponto): Neste modo dois dispositivos podem comunicar entre si, transferir informação, imagens, documentos e até mesmo a troca de contactos virtuais.

A Figura 9 expressa os três modos de funcionamento:



Figura 9 - Os três modos de operabilidade NFC [16]

2.2.1 Modo de emulação de cartão

O modo de emulação de cartão é o modo em que o programador pode dar a um dispositivo NFC o comportamento semelhante a um smart card. Ou seja, este modo permite pagamentos e processos de ticketing (processo em que o utilizador utiliza o dispositivo NFC como se fosse um bilhete) sem ter que mudar a infraestrutura já existente. Neste modo, o dispositivo obedece à norma ISO/IEC 14443-4 de modo a poder usufruir de todas as

potencialidades que a norma dos smart cards proporciona, isto é conseguir interagir com leitores de forma com sucesso.

De todos os modos existentes de funcionamento do NFC este é o único que pode utilizar um Elemento Seguro (ES). Este Elemento Seguro é muito importante dado que ações que utilizem a emulação de cartões por vezes podem englobar certas operações em que é imperativo um determinado conjunto de dados não estarem disponíveis para consulta, como por exemplo, num pagamento através do dispositivo [17].

Este modo pode englobar dois modos de atuar diferentes:

O modo em que não se utiliza o elemento seguro. Este modo é designado por “Emulação do Cartão baseado num Host”. Neste tipo de ação a informação é encaminhada para o CPU do telemóvel no qual aplicações Android estão ligadas diretamente, em vez de encaminhar pelo Elemento Seguro. Na Figura 10 está representado este modo.

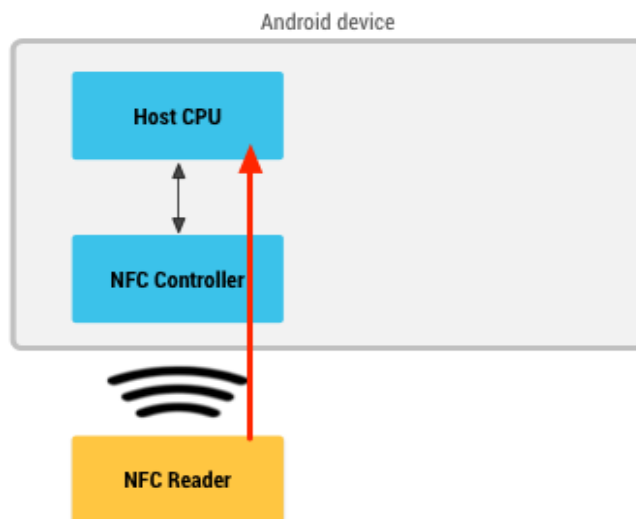


Figura 10 - Emulação do Cartão baseado num Host [17]

E o modo em que utiliza um elemento seguro de forma a tornar o utilizador menos vulnerável em situações mais críticas. Quando a emulação cartão NFC é fornecida através de um elemento de segurança, o cartão a ser emulado é levado através do elemento seguro no dispositivo por meio de uma aplicação Android. Então, quando o utilizador segura o dispositivo através de um terminal NFC, o controlador NFC envia toda a informação desde o leitor até ao elemento de segurança. Na Figura 11 representa isso mesmo.

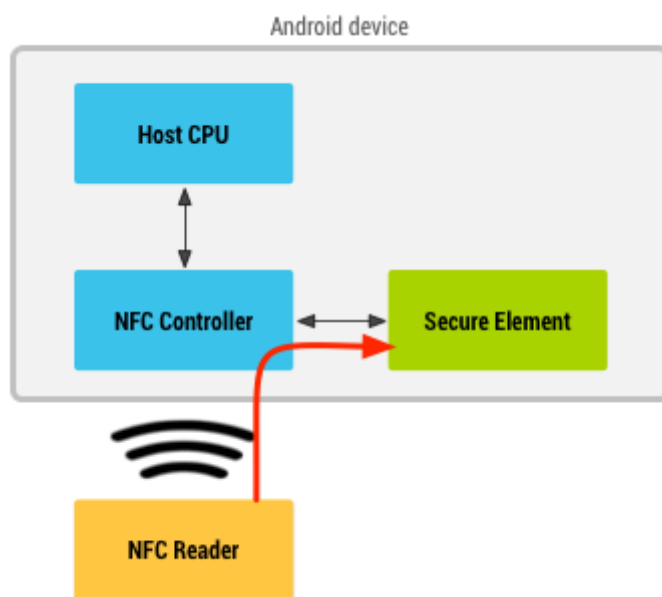


Figura 11 - Emulação dum cartão através do Elemento Seguro [17]

Ainda sobre este modo irão ser abordados vários assuntos, assim como, o que é o elemento seguro, qual é a pilha de protocolos que suporta este modo de funcionamento, a emulação de cartão com elemento seguro por software e a emulação de cartão com elemento seguro por hardware. Ainda sobre este modo irão ser abordados vários assuntos, assim como, o que é o elemento seguro, qual é a pilha de protocolos que suporta este modo de funcionamento, a emulação de cartão com elemento seguro por software e a emulação de cartão com elemento seguro por hardware.

2.2.1.1 Elemento seguro

A ideia deste Elemento Seguro é certificar-se que as aplicações possam ter um ambiente seguro para guardar e executar ações e operações de carácter crítico. Através de operações de controlo de memória para cada aplicação, para funções de encriptação e desencriptação e de assinaturas virtuais de pacotes.

Este Elemento Seguro é na verdade um chip que pode estar incorporado dentro de uma de três possibilidades: pode estar incorporado no mecanismo de NFC do aparelho, pode estar dentro dum cartão SD (Secure Digital), e ainda pode estar incorporado no SIM (Subscriber Identity Module), pelo que neste caso encontra-se controlado pelas operadoras de telecomunicações [18].

2.2.1.2 Pilha de protocolos

A pilha de protocolos do modo de emulação de cartão pode visualizar-se na Figura 12. Existem 3 camadas nesta pilha: Uma camada analógica, uma camada de protocolo digital e uma camada das aplicações.

A camada analógica está relacionada com as características de RF (Rádio Frequência) existentes nos dispositivos NFC e determina ainda os alcances operacionais dos dispositivos.

A camada do protocolo digital refere-se à construção dos blocos de comunicação bem como os aspetos digitais contidos nas normas ISO/IEC 18092 e ISO/IEC 14443.

A camada analógica e a camada de protocolo digital são camadas que não são só empregues no NFC como também empregues nos smart cards. Neste modo poderá existir uma total compatibilidade com este tipo de cartões.

A camada das Aplicações é referente a aplicações proprietárias, tais como aplicações de pagamento, baseadas nas normas ISO/IEC 14443 Tipo A, Tipo B e FeliCa [19].

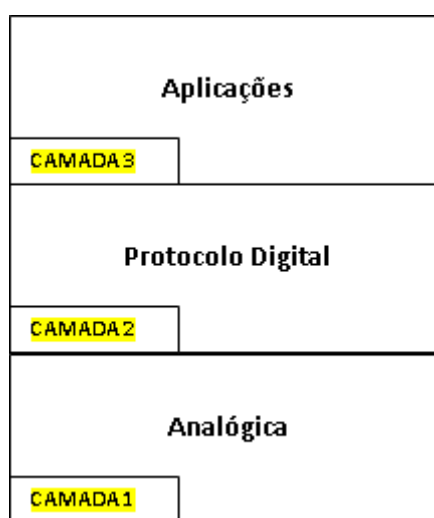


Figura 12 - Pilha de Protocolos do modo Emulação de Cartão

2.2.1.3 Emulação de Cartão com Elemento Seguro por Hardware

Uma característica comum a todos os modos de operabilidade da tecnologia NFC é que todos eles interagem primeiro com o controlador NFC do dispositivo. De seguida, caso não seja necessário nenhuma informação armazenada pelo chip do elemento seguro para comprovar a identidade do utilizador, o controlador NFC interage diretamente com o CPU do dispositivo de modo a executar o pretendido. Pelo contrário, caso seja necessário aceder à informação armazenada no elemento seguro, o Controlador NFC interage com o mesmo de modo a obter esses dados e de seguida sim interage com o CPU [20].

2.2.1.4 Emulação de Cartão com Elemento Seguro por Software

Já foi mencionado que o Elemento Seguro é um chip e quais as suas funções. Posteriormente foi introduzido nos dispositivos móveis o Elemento Seguro por Software. Com a introdução do Elemento Seguro por Software ou Soft-SE deixou de haver a necessidade de existir um chip que pudesse ter as responsabilidades de segurança. Ao invés disso, essas responsabilidades são executadas por um software do dispositivo que dado aos seus certificados e normas garantem a segurança do sistema. Por exemplo, nos telemóveis Samsung Galaxy 7, o elemento seguro integrado denomina-se de PEARL by OT [63].

2.2.2 Modo de leitura/escrita

É importante perceber o que é uma tag para se conseguir entender de fato em que consiste este modo de operabilidade. No modo de Leitura/Escrita usa-se tags ao invés de cartões visto que estas ações dispõem uma reação (ligam o bluetooth, colocam o telemóvel em silêncio, entre outras) e não são responsáveis por processar informação.

Neste modo de utilização um dispositivo móvel equipado com NFC consegue ler ou alterar informação armazenada em tags NFC.

Uma tag é um dispositivo reativo, isto é, responde a um sinal enviado por um elemento ativo que neste caso é o smartphone. Vários exemplos de uma utilização desta tecnologia numa forma inteligente são a leitura e atualização de horários, atualizar e ler certos tipos de papéis, entre outros [22, 23].

2.2.2.1 Tags

As NFC Tags ou simplesmente Tags são pequenos rótulos (Figura 13) totalmente configuráveis que podem ser colados ou afixados em diversos locais tais como habitação, trabalho, entre outros.



Figura 13 - Android Tags NFC

Já existem algumas aplicações para facilitar a programação de tags tais como a “NFC TagWriter”, “NFC Tools” e “NFC Task Launcher”.

Estas aplicações são uma grande maneira de trazer os benefícios do NFC para a vida diária dos utilizadores, facilitando as tarefas do utilizador de inúmeras formas possíveis:

Um utilizador pode, por exemplo, colocar uma destas tags na sua mesinha de cabeceira e programa-la para que quando o utilizador coloque o telemóvel em cima da mesinha de cabeceira para ir dormir, a tag configura o telemóvel para o pôr em silêncio e para colocar um alarme a uma designada hora. Depois quando acordar e passar outra vez o telemóvel na tag o telemóvel já não fica mais em silêncio e tira qualquer alarme. Outro caso prático deste tipo de tecnologia pode ser numa chegada dum filho a casa. A tag é colocada à beira da porta e sempre que a criança chegar a casa e passar o seu telemóvel naquela tag, o telemóvel irá enviar automaticamente enviar uma mensagem ao pai e à mãe a dizer que o seu

filho chegou a casa com segurança e também pode fazer com que o telemóvel se ligue à rede wireless de casa por exemplo.

É importante realçar que existem quatro tipos diferentes de tags. Estas tags podem variar em capacidade e até em formato:

- Tag Tipo 1 -> Este tipo de tag pode ser usada para escrita ou leitura mas os utilizadores podem ainda configurar estas tags para que sejam apenas de leitura. A memória destas tags é de 96 bytes o que permite guardar um pequeno conjunto de dados ou informação e até armazenar um URL. A velocidade destas tags pode chegar até aos 106 kbit/s. É importante também realçar que esta tag é baseada na norma base de ISO14443A . Como resultado das suas características este tipo de tag é a melhor em termos de custo/eficiência [23].

- Tag Tipo 2 -> Este tipo de tag também é baseado na norma base de ISO14443A. Também os utilizadores podem configurar estas tags para leitura ou escrita ou somente para leitura se assim acharem mais conveniente. O que difere as tags do tipo 2 das tags do tipo 1 é o fato das tags do tipo 2 terem uma memória inicial de apenas 48 bytes [23].

- Tag Tipo 3 -> A tag do tipo 3 é um pouco diferente das anteriores porque é baseado no sistema de Sony FeliCa. Ou seja é desenvolvido pela Sony e é no fabricante que as tags podem ser pré-configuradas para poderem ser de leitura/escrita ou somente de leitura consoante as necessidades do utilizador. É importante realçar que as tags desenvolvidas no sistema de Sony FeliCa seguem a norma de ISO 18092. Atualmente tem uma capacidade de memória de 2kbytes e a velocidade de comunicação é de 212kbit por segundo. Este tipo de tags é o mais caro que as anteriores sendo apenas utilizado em situações ou problemas mais complexos [23].

- Tag Tipo 4 -> Este tipos de tags NFC são já pré-configurados pelos responsáveis do seu fabrico que podem ser de leitura/escrita ou de somente de leitura tal como as anteriores. A memória pode ir até aos 32 Kbytes e a velocidade de comunicação pode ir dos 106 Kbits por segundo até aos 424 Kbits por segundo. Este Tag está definido para ser compatível com a norma de ISO14443 do Tipo A e ISO14443 do Tipo B [23].

Na Tabela 1 irá ser apresentada uma tabela a explicar os tipos de tag mais sucintamente:

Tabela 1 - Tipos de Tags existentes [24]

	Type 1 Tag	Type 2 Tag	Type 3 Tag	Type 4 Tag
Compatible Products	Innovision Topaz	NXP Mifare Ultralight/ NXP Mifare Ultralight C	Sony FeliCa	NXP DESFire / NXP SmartMX- JCOP
Memory Size	96 Bytes	48 Bytes / 144 Bytes	1, 4, 9 KB	4KB / 32KB
Unit Price	Low	Low	High	Medium / High
Data Access	Read/Write or Read-only	Read/Write or Read-only	Read/Write or Read-only	Read/Write or Read-only

2.2.2.2 Pilha de protocolos

É possível verificar na Figura 14 que tal e qual como no modo de emulação de cartão a pilha de protocolos começa exatamente da mesma forma, sendo a camada mais baixa analógica e logo a cima desta está a camada de protocolo digital.

A camada referente às operações com tags é uma camada responsável pelas instruções e pelos comandos a serem utilizados pelos dispositivos de modo a poderem comunicar com as diferentes tags existentes.

A camada de aplicações NDEF (NFC Data Exchange Format) contém as aplicações que são baseadas nas especificações NDEF, como por exemplo, smart posters, entre outros.

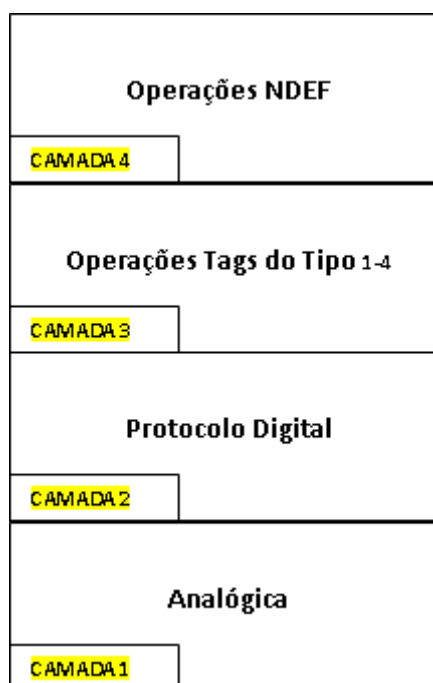


Figura 14 - Pilha de protocolos do modo leitura/escrita de uma tag NFC -

2.2.3 Modo de ponto a ponto

O modo Ponto-a-Ponto permite que dois dispositivos que possuam a tecnologia NFC comuniquem entre si e que assim possam efetuar trocas de informação e de ficheiros. Desta forma os utilizadores podem fazer essa comunicação muito rapidamente apenas com uma simples aproximação de dispositivos e um toque para efetuar a partilha. É ainda importante realçar que este modo segue a norma de ISO/IEC 18092.

2.2.3.1 Pilha de protocolos

Tal como se pode observar na Figura 15, a pilha de protocolos do modo Ponto-a-Ponto tal como nas anteriores possui duas camadas que se encontram na parte inferior que são exatamente as mesmas e que possuem as mesmas funções das pilhas de protocolos dos outros modos de operabilidade, sendo estas a camada analógica e a camada de protocolo digital.

A seguir a estas camadas encontra-se a camada do Logical Link Control Protocol (LLCP). O protocolo LLCP é o que suporta toda a comunicação do modo Ponto-a-Ponto entre dois dispositivos e é essencial para que exista uma comunicação bidirecional. Este protocolo foi desenvolvido em específico para suportar transferências de ficheiros pequenos ou protocolos de rede tais como OBEX e TCP/IP e é baseado na norma IEEE 802.2.

Na camada mais a cima encontram-se o Protocol Bindings e o protocolo Simple NDEF Exchange Protocol. O Protocol Bindings tem como função permitir a interoperabilidade entre os diferentes protocolos, e ao NDEF Exchange Protocol cabe permitir que sejam trocadas mensagens do tipo NDEF.

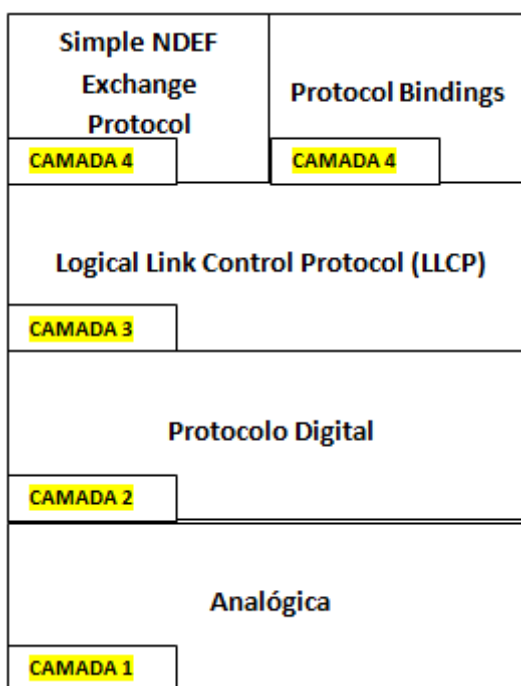


Figura 15 - Pilha de protocolos do modo Ponto-a-Ponto

2.3 Ameaças à utilização de NFC

Existem alguns riscos aquando se utiliza a tecnologia NFC. Diversos especialistas identificaram ameaças a esta tecnologia que advêm de ataques executados por terceiros que podem pôr em causa a segurança e confidencialidade das operações. Vão ser explicados em que é que estes ataques consistem em mais detalhe no próximo capítulo.

2.3.1 Escuta das comunicações

O ataque de escuta das comunicações ou simplesmente “sniffing” é um ataque em que o atacante está à escuta das comunicações de modo a intercetar os dados ou ficheiros enviados pelos dispositivos dos utilizadores. Este ataque está associado a uma comunicação sem fios em que o atacante utiliza uma simples antena de modo a conseguir captar a informação proveniente dos dispositivos que estão a fazer a comunicação. Os atacantes ou invasores podem obter cópias de arquivos importantes ou até senhas de acesso durante a comunicação dos dispositivos.

Uma forma de detetar este ataque é através da análise física de todas as conexões Ethernet e da verificação individual de todas as interfaces do sistema.

2.3.2 Man-in-the-middle

Em português pode-se traduzir este ataque como: “Homem no meio”. Este ataque é quando no decorrer duma comunicação entre dois agentes em que se estão a trocar dados ou informação, existe outro agente que está a intercetar a conversa podendo não só ver o que está a ser comunicado, como também pode enviar mensagens em nome de um dos outros agentes.

Na Figura 16 irá ser apresentado uma comunicação normal e uma comunicação com Man-in-the-Middle.

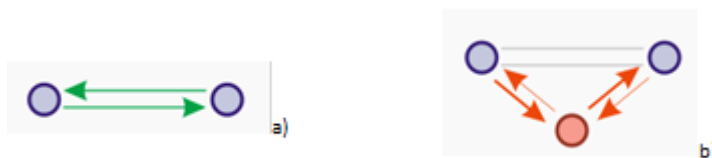


Figura 16 – a) comunicação normal b) comunicação com man-in-the-middle

2.3.3 Negação de serviço

Este tipo de ataque é mais conhecido por “Deny Of Service”, é bastante comum e pode fazer imensos estragos a qualquer sistema. O atacante que faz este tipo de ataque tem como objetivo tornar os recursos indisponíveis sobrecarregando o sistema com pedidos,

dados ou outro tipo de requisição. O atacante tenta fazer muitos pedidos para que os serviços não consigam lidar com tantas solicitações e deixe de funcionar.

2.3.4 Relay Attack

Este tipo de ataque é um tipo de técnica relacionada com o ataque “Man-in-the-Middle”. Num ataque de “Man-in-the-Middle”, o invasor escuta, interceta e até manipula as mensagens enviadas por os outros agentes, mas no “Relay Attack” é o atacante que inicia a comunicação controlando desta forma as duas partes intervenientes.

2.4 Aplicações NFC

Os responsáveis pelo sistema operativo Android e os fabricantes de plataformas móveis já começavam a lançar nos seus modelos uma placa NFC para que os dispositivos possam fazer estas inúmeras operações. E já se pode dizer que existem várias aplicações de utilização de NFC e também para poder configurar tags que permitem, através da tecnologia NFC, fazer algumas tarefas de forma a facilitar a vida aos utilizadores.

Um dispositivo equipado com NFC apresenta uma nova variedade de utilizações que até então não se encontravam disponíveis. Nas próximas secções serão apresentadas algumas dessas utilizações, bem como casos de estudo para cada uma delas.

2.4.1 Controlo de acesso

A característica ou funcionalidade do controlo de acesso é uma forte componente da tecnologia NFC. O fato de haver possibilidade de abrir portas de casa, portas dum quarto de hotel, acesso a edifícios ou locais. Como se pode averiguar estas funcionalidades já se começaram a manifestar a nível de habitação, a nível hoteleiro e também a nível empresarial.

Este modo de funcionamento resulta do fato da tecnologia NFC ter a capacidade de emular um smartphone como sendo um cartão inteligente. O facto de vários cartões poderem ser armazenados dentro do mesmo dispositivo dá a este sistema grande potencialidade sendo um dos principais pontos fortes para a sua utilização.

Em Novembro do ano de 2012, a empresa ASSA ABLOY criou uma solução NFC para desbloquear portas e posteriormente foi testada dentro de hotéis, escritórios, empresas e habitações. A solução denominada de Seos permite que as empresas forneçam aos seus colaboradores acesso a certas portas duma forma muito específica através duma aplicação no seu telemóvel com tecnologia NFC.

O sistema Seos utiliza leitores iCLASS SE e Software de credenciais iCLASS fornecidos pela HID Global, divisão do Grupo ASSA ABLOY com sede na Califórnia [1].

Na Figura 17 pode verificar um exemplo deste sistema:



Figura 17 - Aplicação SEOS e leitor iCLASS [13]

Os leitores iCLASS SE fazem parte da plataforma iCLASS SE da HID Global e garantem adaptabilidade, interoperabilidade e um nível elevado de segurança. A tecnologia de cartão inteligente sem contato iCLASS SE 13,56 MHz tem um consumo muito baixo e faz com que sejam possíveis novas classes de credenciais digitais que podem ser implantadas e incorporadas de forma segura tanto em dispositivos fixos quanto móveis [1].

A empresa ASSA ABLOY para além de fornecer leitores NFC em dispositivos de controlo de acesso em portas também fornece a aplicação necessária para telefones móveis e infraestrutura de software seguro. Esse Software seguro já vem pré-configurado para conseguir determinar quem está autorizado a entrar na área e quando é que está autorizado a entrar. Esta informação está presente nas bases de dados da empresa e através dessa informação, consegue gerar chaves que serão posteriormente associadas aos telemóveis dos colaboradores autorizados e desta forma autorizá-los a entrar na empresa. Importante realçar é que o sistema também oferece outras funcionalidades, tais como, códigos PIN ou a capacidade de negar acesso a determinados indivíduos específicos consoante os interesses da empresa.

Este sistema já foi testado em 28 hotéis com uma arquitetura que compreende os seguintes aspetos: o utilizador faz check-in utilizando uma aplicação presente no seu telemóvel, o sistema de segurança do hotel consegue gerar uma chave para o utilizador e o quarto fica atribuído ao utilizador, a chave é posteriormente enviada para plataforma da empresa via um canal seguro, posteriormente a empresa envia para o telemóvel via o TSM (Trusted Service Manager) uma chave encriptada para o SIM Card do telemóvel e depois o utilizador já pode abrir a porta do quarto com o seu dispositivo móvel.

Existe também uma aplicação denominada de “NFCPorter” [14] que consegue utilizar a placa NFC e assim permitir um número considerado de ações tal e qual como demonstra na Figura 18.

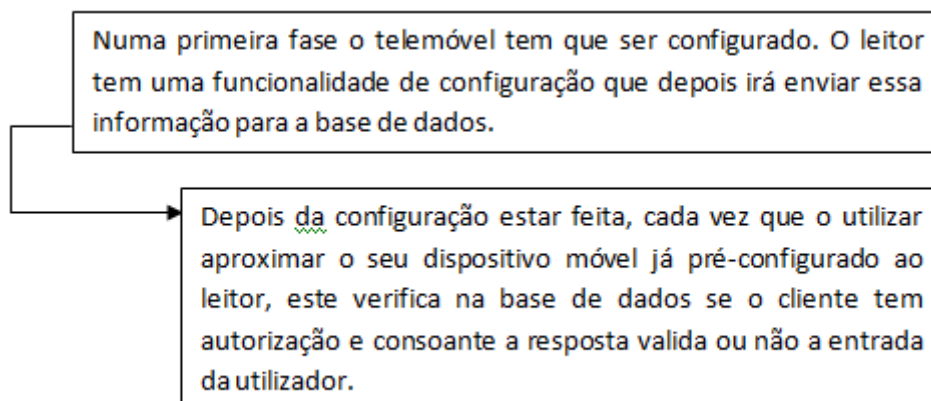


Figura 18 - NFC Porter [15]

A aplicação denominada de NFCPorter permite:

- Abrir a porta com um simples clique no nosso telemóvel
- Substituir os atuais cartões de identificação
- É compatível com a uma grande parte deste tipo de sistemas
- Barato e muito fácil de instalar

O NFCPorter veio para substituir os cartões eletrónicos de acesso pelos sistemas móveis com NFC. Basta ligar o leitor NFCPorter a um sistema de controlo de acesso, fazer o download da aplicação de identificação numa forma gratuita e usufruir das vantagens da identificação móvel [14].

Se não existe a possibilidade de aceder a um sistema de controlo de acesso, o NFCPorter é capaz de trabalhar no modo Stand-alone para controlar apenas uma única entrada. O NFCPorter é uma boa opção para o controlo de abertura da porta de casa, da garagem, dum escritório, entre outros.

O Sistema NFCPorter é projetado para a máxima compatibilidade com sistemas de controlo de acesso atuais, bem como o padrão mais frequente de cartões de identificação. Portanto, o sistema está preparado para a instalação em locais onde a maioria das pessoas ainda utilizam cartões sem contato próprio e assim o sistema garante o funcionamento paralelo de ambas as tecnologias para um período ilimitado de tempo.

Do ponto de vista de qualquer sistema de controlo de acesso já existente é exatamente igual. Tudo o que se tem que fazer é trocar os leitores e o NFCPorter fará o resto.

A identificação do utilizador é tão fácil como aproximar o leitor NFCPorter com um smartphone. Depois da aproximação do smartphone existe uma transferência do ID do utilizador. O ID é então transferido para um sistema de identificação superior, a fim de validar os direitos do utilizador e desbloquear então a entrada caso este tenha permissão.

No entanto este sistema possui uma diferença significativa comparado com outros sistemas pois não opera sobre leitores NFC/RFID já implementados. Para este sistema ser utilizado é necessário que sejam adquiridos leitores NFC próprios criados pela empresa que disponibiliza o NFCPorter. Esta necessidade de adquirir novo hardware é uma grande desvantagem perante os outros sistemas que funcionam sobre os leitores NFC/RFID existentes.

A nível de habitação, cada vez mais tem-se falado em casas inteligentes mas o problema é que envolve um investimento inicial grande. Agora com apenas um dispositivo móvel como o telemóvel, o utilizador pode ter a abertura de portas automatizada contribuindo assim para uma casa mais inteligente.

Esta tecnologia NFC no controlo de acessos também já está a ser utilizada nos hotéis. A experiência pioneira a utilizar NFC como método de acesso foi feita pelo Clarion Hotel em Estocolmo na Suécia no ano de 2010. De maneira a se poder realizar testes sobre este novo método de acesso foi criado um projeto teste em que alguns dos clientes do hotel receberiam um telemóvel com NFC e com uma aplicação própria para o efeito. Esta aplicação permitir-lhes-ia não só, aceder ao quarto, como realizar o processo de check-in e check-out sem ter de se deslocarem à receção do hotel [26].

Depois da realização da reserva os clientes recebiam no dia selecionado um mensagem na qual vinha explicito uma ligação para uma aplicação on-line onde os clientes teriam que realizar o check-in. Este processo de check-in sendo bem-sucedido, o cliente receberia, no seu telemóvel, o número do quarto bem como o acesso ao mesmo. Após o check-in ser realizado e de o hotel atribuir um quarto ao cliente, uma chave é enviada do hotel para um servidor externo, que irá encriptar a chave e posteriormente enviar para o cartão SIM do cliente [27]. Aquando a chegada ao hotel o cliente não precisa de se dirigir à receção para fazer o check-in, podendo simplesmente deslocar-se para o seu quarto. Chegando à porta do seu quarto, basta colocar o telemóvel em contato com o leitor existente na porta e caso o acesso fosse válido a porta abrir-se-ia.

No intuito de fazer check-out era apenas necessário o cliente colocar o telemóvel em contato com uma tag NFC existente na receção do hotel e confirmar o processo. Podemos verificar na Figura 19 todas as fases do processo.

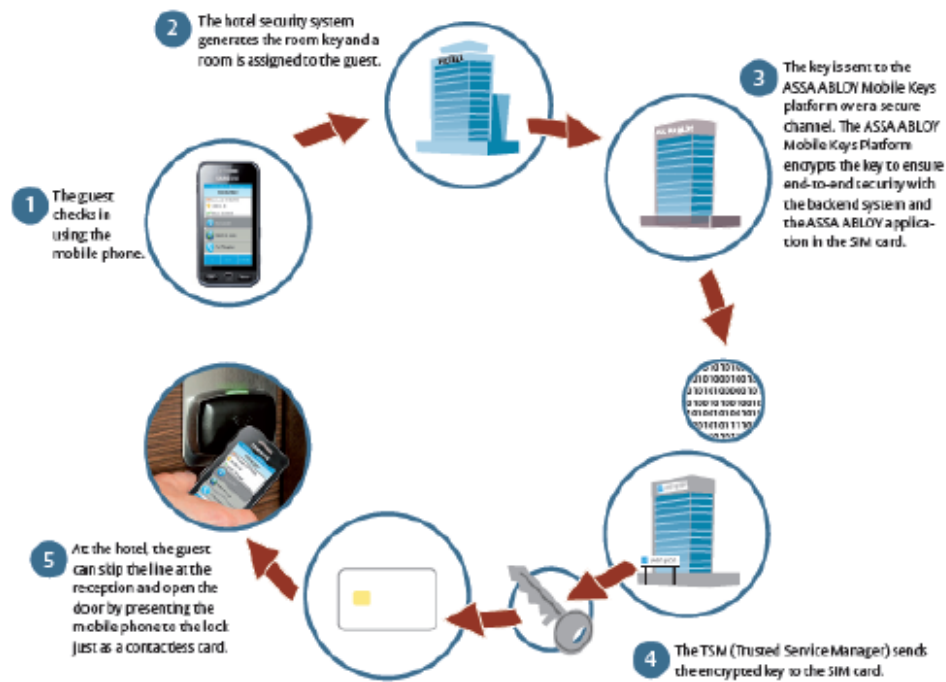


Figura 19 - Processo de atribuição de chave do quarto do Clarion Hotel através de NFC [27]

Quando se realizaram estas experiências foi feito um estudo no qual se concluiu que as respostas tinham sido bastante positivas e encorajadoras para a utilização desta tecnologia. Grande parte das pessoas que participaram fizeram referência ao facto de pouparem tempo no processo de check-in e referiram ainda que caso o seu telemóvel suportasse NFC, utilizá-lo-iam para este tipo de acesso.

Outro exemplo de sistemas utilizando NFC a nível de controlo de acessos é o Key2Share e possui um modo de operação semelhante ao controlo de acesso em Estocolmo. Porém este sistema foi desenhado de modo a suportar outros casos de uso relacionados com a autorização de acesso, como receber uma chave eletrónica de um automóvel, entre outros. Um colaborador duma dada empresa que possua este sistema recebe uma chave eletrónica de modo a que este se registe no servidor da empresa, podendo então criar o seu registo, utilizando a respetiva chave. Posteriormente, recebe da parte do servidor uma chave definitiva que lhe garante acesso ao edifício pré-definido. Após este processo o novo colaborador fica com acesso ao edifício utilizando o seu smartphone. Na Figura 20 pode-se verificar todo este processo.

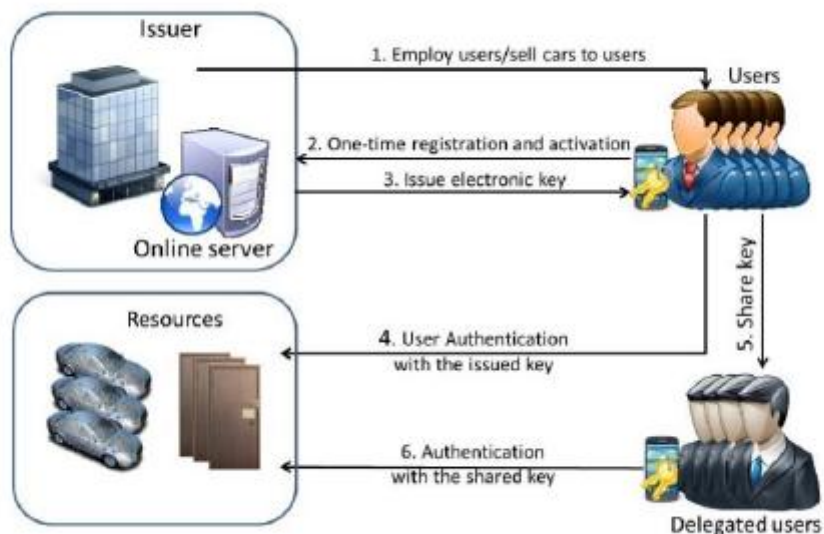


Figura 20 - Processo de atribuição de chave eletrônica do sistema Key2Share

Também a nível hoteleiro, a cadeia de Hotéis “Starwood Hotels” já anunciou um sistema sem chave e através do telemóvel. Esta cadeia de hotéis conta já com mais de 150 hotéis por todo o mundo em cidades como Pequim, Hong Kong, Los Angeles, Nova Iorque. Depois de fazerem a reserva, os clientes serão notificados, com 24 horas de antecedência, do seu número de quarto e todas as informações que vão precisar. Posteriormente, basta os utilizadores colocarem o telemóvel perto da fechadura da porta do seu quarto e a porta abrir-se-á [25].

A nível empresarial também já se começa a notar alguns avanços embora este seja o menos desenvolvido dos três. Por exemplo, em Madrid já existem algumas empresas que utilizam a tecnologia NFC para controle de acesso ao edifício da empresa.

2.4.2 Pagamentos

Uma característica importante e interessante da tecnologia NFC é a sua capacidade de realizar pagamentos. De fato, parece uma coisa do futuro mas fazer pagamentos através dum dispositivo móvel já é possível. Atualmente uma pessoa desloca-se a uma loja ou a um supermercado e perante os modos de pagamento atuais tem de optar por dois métodos de pagamento: pagar com cartão multibanco ou com dinheiro. Se optar por pagar com dinheiro implica ter de levar sempre consigo o valor necessário para as referidas compras e quanto maior o valor das compras, maior quantidade de dinheiro este terá de levar, o que não é nem o mais cómodo nem o mais seguro. Comparando com o pagamento em dinheiro, o processo de pagamento por cartão multibanco é mais rápido e mais eficiente. No entanto, no caso de o cliente possuir vários cartões multibanco terá de escolher o mais conveniente e de seguida colocá-lo pela orientação correta e introduzir o PIN (Personal Identification Number) para

concluir a compra. Quer a pessoa escolha um ou outro método implica sempre levar algo mais no seu bolso, pelo que se houver a opção de pagar com o telemóvel usando a tecnologia NFC, essa pessoa tem que carregar menos coisas, tomando de princípio que o telemóvel é sempre imprescindível.

Os pagamentos através da tecnologia NFC é um mecanismo que precisa de uma aplicação que permita fazer essas ações tornando o processo de pagamento muito mais rápido e eficiente. Um utilizador que tenha este tipo de aplicação basta simplesmente passar o seu dispositivo móvel com NFC num dispositivo que faça de leitor e deverá ser inserido um código PIN de modo a tornar o método de pagamento mais seguro e após a inserção do código, em caso de validação a compra é feita com sucesso. Não é necessário a introdução do código PIN para todas as transações, apenas para algumas. Uma grande vantagem ao usar um telemóvel para fazer pagamentos em relação a cartões de crédito é que se um cliente usar um telemóvel pode armazenar vários cartões num único telemóvel, enquanto que se usar cartões de crédito e se possuir mais que um, essa pessoa poderá ter que levar mais que cartão e isso por si só é uma grande vantagem para quem usa NFC como modo de pagamento. É ainda possível que se possa guardar cupões de promoções no telemóvel, acabando deste modo, com a utilização de papel ou de outro dispositivo que contenha essas informações, simplificando ainda mais as coisas para o lado do cliente.

Isto tudo com o objetivo de tornar a experiência e tarefa do utilizador o mais simplificada possível, não esquecendo o aspeto da segurança.

Foi no ano de 2011 que a empresa americana Google lançou uma aplicação para os clientes realizar pagamentos recorrendo à tecnologia NFC [29]. A Google Wallet veio transformar o smartphones num cartão de crédito. A Google Wallet é uma aplicação que qualquer cliente pode fazer download na PlayStore da Google e que permite aos utilizadores associar vários cartões de crédito ao dispositivo assim como também utilizar cupões promocionais [30]. Com a simples aproximação dum telemóvel a um terminal com NFC e com esta aplicação é possível realizar uma transação para fazer o pagamento dum artigo ou serviço como se pode visualizar na Figura 21.



Figura 21 - Exemplo da utilização da Google Wallet

Como se pode verificar na Figura 21, basta aproximar o telemóvel do leitor e, em caso de haver mais que um cartão armazenado no smartphone, o utilizador deverá seleccionar o cartão pelo qual deseja efetuar o pagamento.

Nos anos seguintes Google Wallet passou a chamar-se Android Pay tendo surgido vários concorrentes a este sistema, entre eles a Apple Pay e a Isis Mobile Wallet. Existem poucas diferenças entre estes sistemas sendo elas apenas a nível de países autorizados à utilização e a nível de guardar as transações feitas [29].

2.4.3 Bilheteira ou “Ticketing”

Uma outra possibilidade para o uso de NFC consiste na compra de bilhetes para um determinado número de eventos ou transportes. Esta nova maneira de comprar bilhetes poderá ser a resposta a vários problemas, entre eles tornar muito mais vantajosa a compra de bilhetes por parte do utilizador, visto que este não precisa de se deslocar a um certo local para comprar os bilhetes e também evita as filas de espera. Outra vantagem em usar um smartphone como bilhete é a robustez dada pelo telemóvel que é maior do que um bilhete tradicional de papel.

Esta utilização do NFC não tem vantagens apenas para os utilizadores mas também para as empresas que para além de reduzir os custos do fabrico de bilhetes também reduzem o custo de manutenção das máquinas e dos funcionários das bilheteiras. Do ponto de vista do ambiente, o bilhete tradicional feito em papel engloba sempre grandes desvantagens a nível ambiental pelo processo de fabrico dos bilhetes, processo esse que seria reduzido com a utilização deste método.

Tal como na utilização dos pagamentos também no Ticketing ou bilheteira é preciso uma aplicação. Essa aplicação possibilita a consulta e compra dos bilhetes. Esses bilhetes podem ser de várias categorias: transportes, cinema, museus, entre outros.

No ano de 2008 na Alemanha foi implementado um projeto que tinha como objetivo utilizar o NFC como substituição dos bilhetes de transportes. O projeto denominado de Touch&Travel nasceu. Para os utilizadores poderem usufruir deste novo modo de viajar tinham então que primeiro ter instalado no seu telemóvel a aplicação de forma gratuita e posteriormente criar um perfil. Depois, antes de cada viagem, o utilizador teria que se registar na viagem e poderia fazê-lo apenas com o aproximar do seu smartphone a um leitor disponível nas paragens. No final de cada viagem o utilizador teria que passar o seu dispositivo para fazer o logoff da viagem para dizer que a viagem terminou e posteriormente proceder ao pagamento [31]. O sucesso deste projeto foi muito grande de forma a que este projeto ainda se encontra ativo nos dias de hoje.

2.4.4 Cultura

Também é possível dentro dos museus os clientes terem a oportunidade de interagirem com estátuas, pinturas, esculturas, entre outros. Este novo conceito permite que aos clientes interagirem com a exposição, utilizando o seu *smartphone* para consultarem informações sobre as amostras, tais como quem é o autor daquela peça, o ano e outras informações relevantes.

Um exemplo deste tipo de utilização é o museu de Londres, *Museum of London* e no *Museum of London Docklands*, que desde 2011 disponibiliza aos seus visitantes uma aplicação que permite a quem possuir um *smartphone* com NFC, utilizar uma variedade de serviços tais como uma maior aprendizagem sobre os artigos em exposição, compra de bilhetes para futuras exposições ou eventos e caso os clientes desejem podem receber notícias no seu telemóvel acerca do museu. Podemos verificar essa interação na Figura 22.

Como o museu de Londres, já vários museus em todo o mundo adotaram este sistema para agradarem ao máximo os seus clientes. Alguns desses museus são: *Wolfsonian Museum*, em Miami Beach nos Estados Unidos da América [34], *Please Touch the Exhibits!* em Filadélfia nos Estados Unidos da América [35], *Ancient Italian Renaissance Villas* em Londres [36], entre outros.



Figura 22 - Interação com NFC no Museu de Londres [33]

2.4.5 Tarefas rotineiras

A tecnologia NFC tem tantas funcionalidades que até pode ser utilizada fora do contexto empresarial e fora de qualquer tipo de contexto de negócio sendo possível utilizar até nas tarefas mais rotineiras.

Existe uma aplicação para programação de tags. A aplicação denomina-se de “NFC Task Launcher” tal como se pode verificar na Figura 23. O utilizador primeiro cria um conjunto de ações, depois coloca ações nesse conjunto e depois com o aproximar da tag, a aplicação vai programar a tag com aquele conjunto de ações.

Por exemplo, pode-se criar um conjunto com o nome dormir, depois adiciona-se a ação de despertador a uma designada hora e também colocar o dispositivo em silêncio. Posteriormente ao aproximar a tag, a tag vai ficar programada com essas duas ações e quando o utilizador aproximar o dispositivo da tag, ele vai ficar automaticamente programado para ter um despertador à hora designada e em silêncio, tornando a tarefa do utilizador bem mais simples.

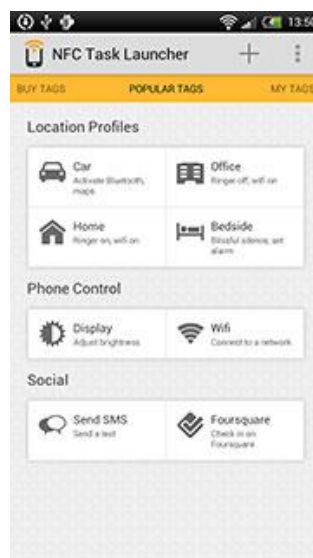


Figura 23 - NFC Task Launcher [12]

Uma ideia interessante que se poderá ter recorrendo à tecnologia NFC é por exemplo ter uma tag NFC na entrada da habitação da parte de dentro e quando uma criança chega a casa passa o seu telemóvel nessa designada tag. Essa tag pode estar pré-programada para quando o telemóvel da criança passar nessa interface, esta envia automaticamente uma mensagem aos pais a dizer que a criança chegou em segurança a casa e pode também ligar o telemóvel da criança à rede sem fios de sua casa.

Uma ideia bastante útil para usar esta tecnologia é ter uma tag NFC na mesinha de cabeceira programada para que ao estar em contato com um telemóvel vai colocar o telemóvel em silêncio e programar o despertador para o dia seguinte. E de manhã ao aproximar novamente o dispositivo vai colocar com som [37].

Outra ideia inteligente é colocar uma tag NFC no carro e quando se passar o telemóvel nessa interface o telemóvel liga-se logo por Bluetooth ao carro, lança a aplicação do mapa e ativa o GPS.

Aqui está uma lista de ideias interessantes que se poderá fazer com esta tecnologia [37, 38]:

- Definir um temporizador para 30 minutos quando se coloca as roupas na lavandaria.
- Para se estar completamente focado e imperturbável enquanto se está no ginásio ou num sítio de exercício, apenas colocar uma tag no seu saco de desporto que coloca o telemóvel em modo de voo e lança uma lista de reprodução treino.
- Colocar uma tag no portátil ou tablet e deixar ligar ao hotspot Wi-Fi do telefone. Isto pode ser muito útil se uma pessoa estiver em movimento ou trabalhar nalgum transporte público e precisar de uma conexão à internet para procurar alguma coisa rapidamente.

Também tem que se deixar um tag de alternância no hotspot Wi-Fi, para que se possa desligar novamente com um toque rápido.

- Ao viajar no metro significa que por vezes o telefone está a ter uma tarefa muito complicada para conseguir ter sinal no telemóvel, e isto acaba por gastar muita energia da bateria. Para evitar isso, poderia se colocar nas estações de metro e as pessoas antes de entrarem no metro passavam o telemóvel nessas tags para entrarem em modo de voo e ao saírem do metro voltavam a passar para desligar o modo voo.
- Embora seja mais complicado de se realizar já é possível ligar um portátil e luzes através da tecnologia NFC. Basta passar o telemóvel nas tags designadas e essas ações já são possíveis.
- Existem muito mais ideias para além destas que podem fazer com esta tecnologia.

2.4.6 Saúde

A tecnologia NFC pode ainda ser utilizada na área da saúde. Embora nesta área ainda esteja pouco explorada a tecnologia NFC pode contribuir muito para uma mais-valia na área da saúde.

Esta tecnologia nos hospitais pode facilitar na identificação de pacientes para que seja melhor e mais rápido as ações que os profissionais de medicina têm que fazer ou também podem permitir a redução das taxas de erro existentes na utilização de medicamentos.

Estes lapsos podem ocorrer mas com tags NFC a ajudar na identificação de cada paciente, os profissionais sabem logo quem é o paciente, o que ele tem e que tratamentos está a ser sujeito, reduzindo assim erros tais como a incorreta dosagem de medicação, confusão de pacientes, administração de medicamentos fora de horas, entre outros.

Com a utilização de tags NFC também em caso de emergência é possível aceder com maior rapidez às informações pessoais do paciente, como por exemplo, o tipo de tratamento que o doente está a ter, as alergias que este tem, o tipo de sangue e outras características que possam fazer a diferença nestes casos.

Por mais abstrato que isto possa parecer já houve projetos desta natureza a ser implementados. O projeto pioneiro surgiu no Kuopio University Hospital na Finlândia no ano 2008 que começou a utilizar o NFC como forma para reduzir erros entre os seus profissionais [39].

Na cidade de Karachi no Paquistão um sistema foi implementado no ano de 2009 um sistema em tempo de real de rastreamento de doenças usando a tecnologia NFC. Na cidade de Karachi era muito complicado de manter registos correspondentes a cada pessoal quer fosse no papel ou a nível eletrónico. Este fato coloca vários desafios, incluindo a falta de um método confiável para a identificação do doente ou compreensão da doença.

A incapacidade de guardar registros ou breves relatórios médicos impede a continuidade dos cuidados, não só com pacientes a moverem-se de hospitais para hospitais como também dentro da mesma clínica ou hospital.

Os doentes podem já ter sido observados, lhes terem realizadas prescrições, feito exames através de laboratórios de teste, radiografias, e essas informações não serem facilmente acessíveis e são até frequentemente ignoradas. Para além disso, os centros de saúde são insuficientes, os hospitais muitas vezes têm um grande número de doentes num curto período de tempo, e podem não ter tempo para aconselhar ou encaminhar os pacientes de forma adequada.

A fim de superar estes desafios e facilitar a deteção de pneumonia em crianças pequenas, começaram-se a desenvolver ferramentas que utilizam telefones que possuem a tecnologia NFC para permitir a identificação do paciente e rastreamento em vários hospitais.

Estas ferramentas permitem que os pacientes sejam identificados por uma única frequência de rádio (RFID) e depois segue-se a transmissão para uma tag cada vez que os pacientes visitam uma clínica ou hospital.

Este sistema foi desenvolvido a pensar no combate à doença da pneumonia especialmente no caso das crianças. Então quando se usa um telemóvel com NFC para enviar um ID do paciente para um servidor central, o médico será capaz de conseguir fazer um trabalho muito mais eficaz ajudando assim a combater as doenças. Então se a pneumonia é diagnosticada, existe uma equipa móvel que é alertada para levar a criança doente para mais avaliação e posterior tratamento [40].

2.5 Avaliar soluções e abordagens existentes

Existem algumas abordagens já desenvolvidas nesta área pelo que serão descritas e analisadas.

Uma das abordagens que já foram descritas foi a da empresa ASSA ABLOY que criou uma solução que passava por fazer testes piloto em 28 hotéis distintos pelo que no final 17 desses hotéis confirmaram um grande melhoramento a nível de performance dos processos de hotelaria [1].

Nesta solução os clientes faziam reservas pelos canais normais já existentes e no dia em que chegavam ao hotel, os clientes recebiam uma chave encriptada via SMS. Nesse SMS estava presente um link, o cliente ia aceder a uma aplicação online onde iram fazer o check-in e receber o número do quarto.

Quando o check-in estava completo, os utilizadores recebiam uma chave do quarto digital que ia estar interligada com o dispositivo móvel NFC do utilizador. Quando chegassem ao hotel os clientes podiam simplesmente ignorar a receção onde, geralmente, se faz o check-

in e podiam ir diretamente para o seu quarto e abrir a respetiva porta aproximando apenas o seu telemóvel à fechadura da porta.

O check-out também foi gerido através da aplicação online de chaves móveis e as chaves do quarto digitais do hotel foram desativadas quando o check-out foi concluído.

Após estes testes pilotos foram feitos questionários aos clientes para se poder avaliar a eficácia e a adaptabilidades dos clientes a este projeto e os resultados foram bastante positivos.

Dos 28 hotéis em que esta experiencia ocorreu, 17 desses mesmos disseram que grande parte dos clientes usaram esta aplicação repetitivamente e que os clientes tiveram uma fácil adaptação ao sistema. Foi feito um inquérito de satisfação aos clientes em que estes puderam classificar a experiência em vários aspetos, de resposta aberta e também dar sugestões para melhorar a qualidade do serviço.

O inquérito demonstra que os hóspedes apreciaram o fato de não terem que estar à espera na receção do hotel. Cerca de 90% dos hóspedes afirmaram que economizaram tempo por não terem que fazer check-in na receção e 60% disseram que pouparam pelo menos 10 minutos nesse processo. Por fim, concluiu-se que 9 em cada 10 pessoas preferiam usar este método para suas estadias em hotéis se o seu telemóvel assim o suportasse.

Aqui estão outros resultados importantes do inquérito (Figura 24):

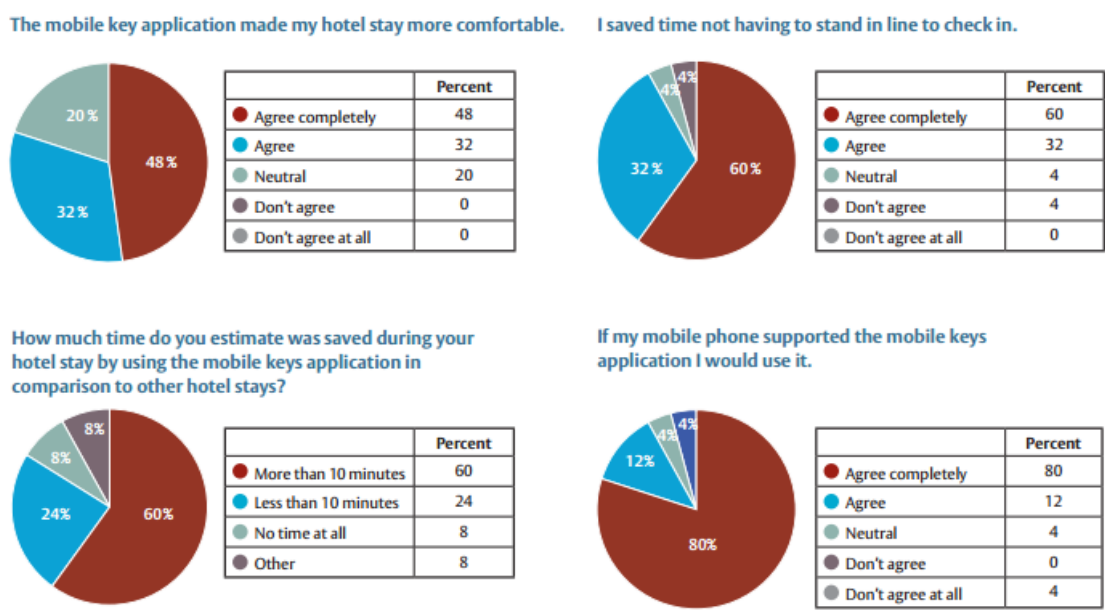


Figura 24 - Gráficos circulares do Inquérito do projeto da empresa ASSA ABLOY [1]

Conforme os resultados indicam a experiência foi bastante positiva na opinião dos clientes pois para além da maioria dos inquestionados não só a experiência no hotel foi mais

agradável como também pouparam tempo nas ações da sua estadia como usariam a aplicação outra vez. Isso são notícias muito entusiasmantes para os desenvolvedores de aplicações que utilizam a tecnologia NFC.

3 Design da solução

Este trabalho é um de três componentes que constituem o projeto em si. O trabalho desta dissertação é a parte do dispositivo móvel ou smartphone que irá por si só interagir com as outras duas componentes do projeto sendo elas os serviços do hotel (internal services e external services) e o componente do quarto (Room). A Figura 25 demonstra a aplicação em questão, representada pelo SmartPhone. A aplicação é denominada de MyAccommodation.

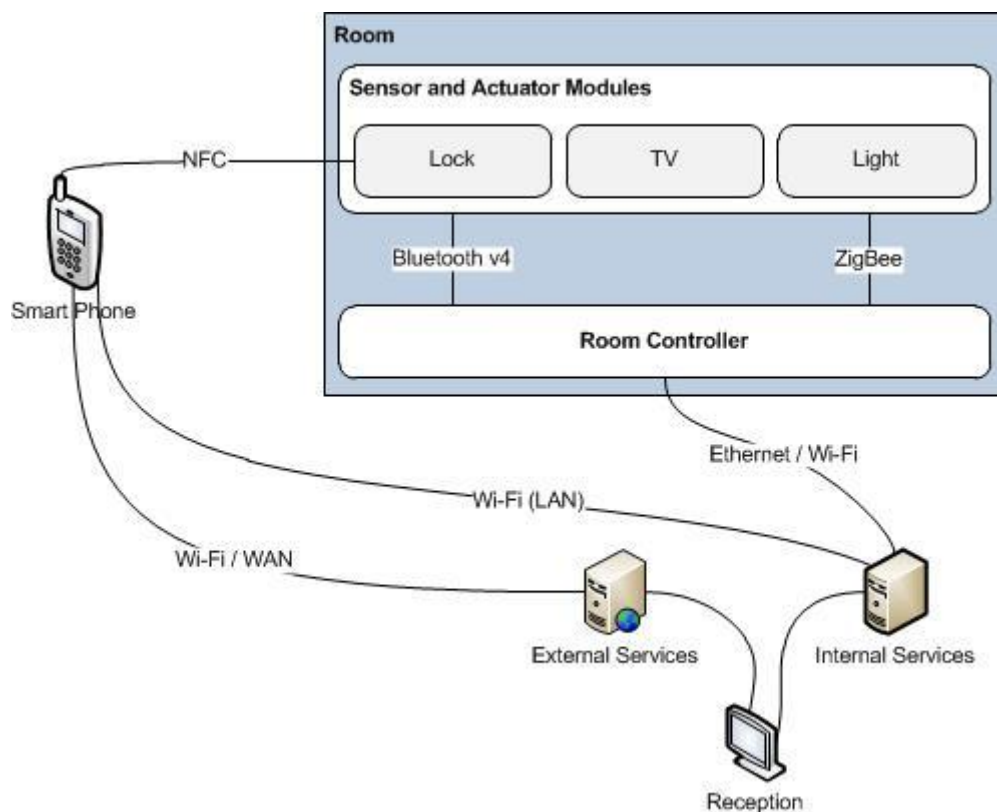


Figura 25 - Arquitetura do sistema

O trabalho da dissertação, que é a aplicação Android (representado na figura por SmartPhone) interage com a componente do Quarto (Room) no sentido do utilizador poder entrar no quarto através da simples aproximação do seu telemóvel e no sentido do utilizador poder enviar ordens ao quarto a nível de domótica, nomeadamente, acender ou apagar as luzes do quarto e abrir ou fechar as janelas do quarto. A parte de abrir a porta, acender a luz e fechar a janela irá ser depois consumada por um aluno de eletrotécnica que vai ser responsável por receber o comando informático e consequentemente fazer responder o mecanismo da porta, lâmpada ou janela consoante o resultado recebido.

De seguida irá ser demonstrado, através dum diagrama de fluxo (Figura 26), como a solução está estruturada no caso do utilizador queira efetuar o check-in ou abrir a porta do seu quarto:

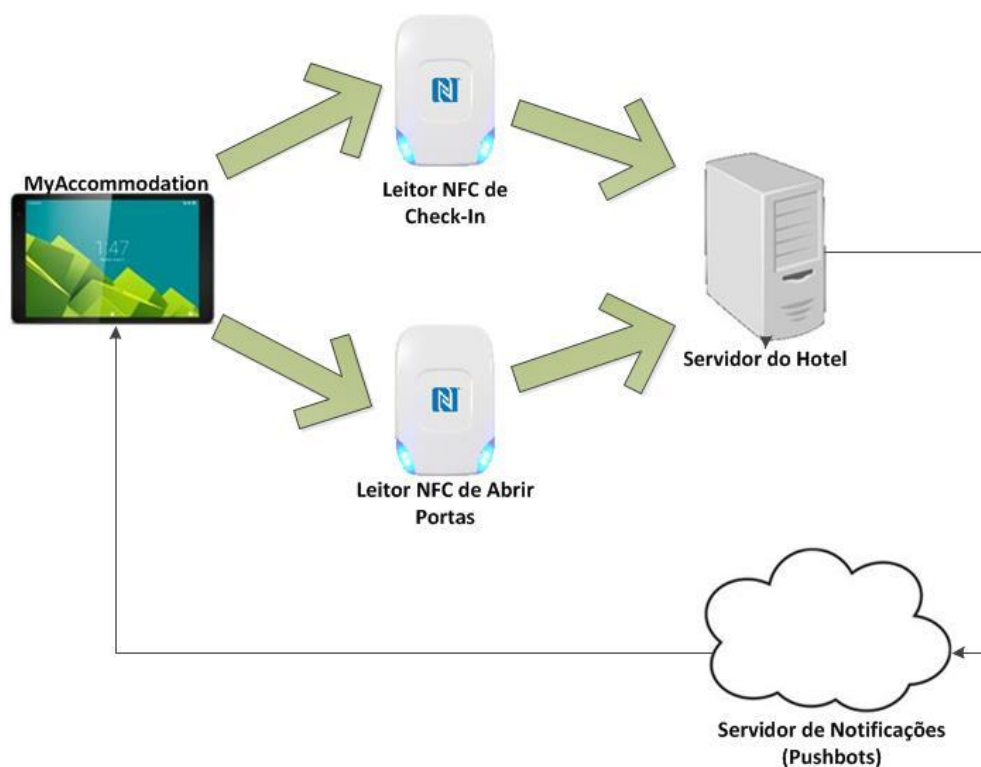


Figura 26 - Diagrama de fluxo da comunicação NFC

Os dados que serão passados pela aplicação cliente para o leitor serão as seguintes:

- Id do utilizador
- Notification Token

O Id do utilizador irá indicar que utilizador pretender fazer a ação, enquanto que o Notification Token será responsável por indicar qual o dispositivo está a pretender fazer a comunicação NFC. O Notification Token é indispensável para a aplicação do leitor porque esta aplicação terá que comunicar ao serviço qual o dispositivo para qual ele tem que enviar a notificação de resposta.

Como se pode observar na Figura 26, depois de aproximar o seu dispositivo a um leitor NFC de check-in ou a um leitor NFC de abertura de porta dum designado quarto a aplicação do leitor irá comunicar com o servidor. Este processo irá ser explicado no capítulo seguinte que diz respeito à implementação da solução.

Depois a aplicação cliente ficará à espera duma notificação que irá ser recebida pelo servidor de notificações Pushbots tal como está demonstrado na Figura 26.

Ainda em relação à componente do quarto e no que diz respeito à parte do utilizador conseguir controlar os componentes do quarto, foi definido um protocolo que vai ser utilizado para comunicar por Bluetooth com o sistema central do quarto. Ao clicar no botão, a aplicação irá enviar informação necessária para ligar/desligar uma dada luz ou abrir/fechar a janela:

- ID: O ID será responsável por identificar que componente o utilizador escolheu interagir. Pode ser Luz1, Luz2, Janela1, Janela2, etc.
- Resultado: O Resultado será responsável por identificar se o utilizador escolheu ligar/desligar ou abrir/fechar o respetivo componente identificado pelo ID.

Outra parte do projeto como um todo é a parte que se encontra na parte inferior da Figura 25, que diz respeito aos serviços do hotel (internal services e external services) que irão ser chamados através da aplicação presente no Smartphone para garantir que as ações pretendidas ocorram com sucesso.

Os utilizadores após fazerem determinada ação que requeira informação ou intervenção dos serviços, a aplicação irá contactar com os mesmos, presentes num servidor externo para garantir as funcionalidades requeridas ao bom funcionamento da aplicação. Estes serviços REST vão fazer a comunicação com a base de dados para poderem confirmar valores recebidos fazendo comparações ou até para fazer edição de informação se o sistema assim o pretender. De seguida irá ser apresentado o diagrama de fluxo de pedidos de serviços (Figura 27).

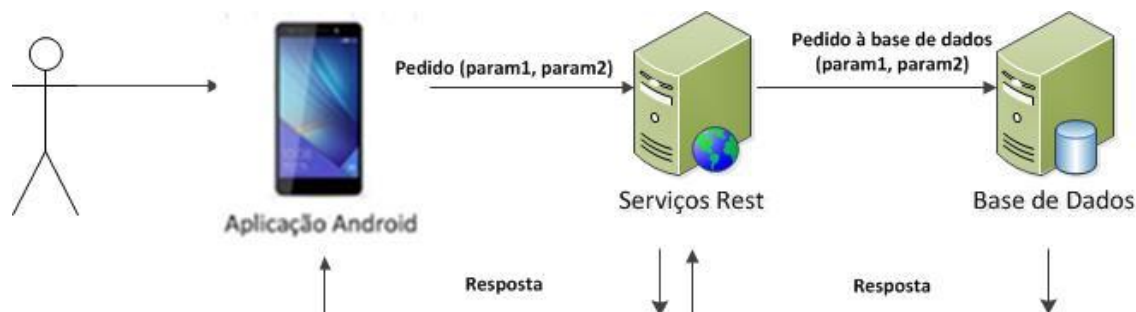


Figura 27 - Diagrama de Fluxo de pedidos de serviços

Tal como se pode observar na Figura 28 o utilizador vai através da aplicação Android fazer ações, por exemplo, efetuar reserva. Depois a aplicação vai fazer um pedido ao servidor de efetuação de reserva, enviando-lhe diversos dados (id_do_cliente, data_inicio, data_fim).

Posteriormente o servidor irá fazer o insert na base de dados e preencher a reserva no sistema e depois irá receber a resposta da base de dados que irá transmitir ao cliente.

A aplicação cliente vai ser a aplicação através da qual os utilizadores poderão fazer as diversas ações que a solução compreende tais como efetuar reserva e consultar as reservas do cliente, fazer check in e abrir a porta com o telemóvel, pedir serviços ao quarto. A aplicação chama-se MyAccommodation.

Na Figura 28 está um diagrama de Use Case a explicar o que os utilizadores podem fazer através da aplicação Android. Existe um utilizador que poderá, com a aplicação Android, fazer as seguintes ações: Fazer e consultar reservas, ver informações do hotel, pedir um serviço ao quarto, efetuar o check-in e o check-out e abrir a porta do quarto com a aproximação do telemóvel.

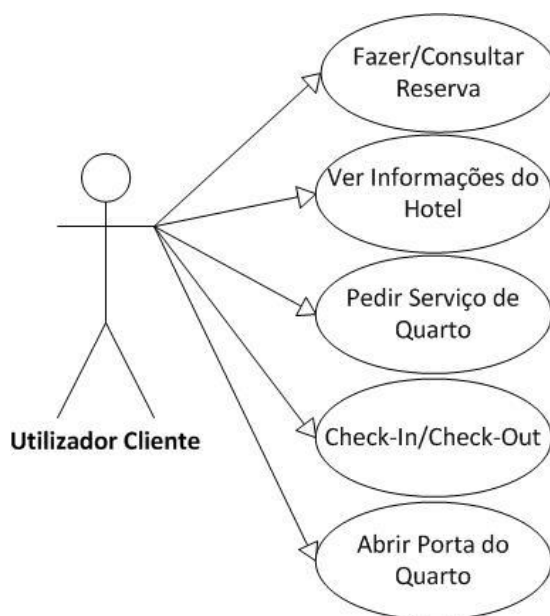


Figura 28 - Diagrama Use Case

A aplicação será desenvolvida no Android Studio. Foi escolhido o Android Studio porque é o ambiente de programação oficial da Android, por ser fácil de utilizar e por haver muita documentação disponível online para auxílio de trabalho. Existe outro ambiente de programação que se chama Eclipse. A decisão em utilizar o Android Studio como ambiente de programação foi devido a vários fatores que vão ser de seguida inumerados [46]:

- O Android Studio é um produto oficial da Google e o Eclipse não iria ter mais suporte.
- Conclusão de código avançada: Quer o Android Studio, quer o Eclipse usam um sistema de auto-conclusão do código como ajuda aos programadores mas esse sistema está melhor no Android Studio, pelo que muitas vezes a auto-conclusão no Eclipse tem resultados que não são muito precisos.

- A opção de Drag-and-Drop: Embora a função de Drag-and-Drop para desenhar a interface do utilizador não seja importante para os desenvolvedores, ela está presente no Android Studio e não no Eclipse.

É também importante realçar que o Android Studio pode ser adquirido de forma gratuita e através do website da Android Studio [53].

Esta aplicação será suportada por serviços web feitos em Node.js utilizando uma arquitetura REST. Esses serviços permitiram fazer todas ações que o utilizador precisa de fazer, sendo elas: fazer e consultar reservas, ver informações do hotel, pedir um serviço ao quarto, efetuar o check-in e o check-out e abrir a porta do quarto com a aproximação do telemóvel.

A tecnologia Node.js é uma tecnologia que é bastante favorável no que toca a aplicações de rede escaláveis pois tem uma boa capacidade de lidar com um elevado número de conexões simultâneas com alto rendimento [6]. A arquitetura REST é uma arquitetura e não uma linguagem de programação. Esta arquitetura permite uma maior organização de forma a tornar mais simples os procedimentos mais complexos. REST é baseado no conceito de servidor/cliente e quando um cliente faz um pedido ao servidor, o servidor tem que dar uma resposta. Os pedidos e respostas são construídos em torno de “representações” e “recursos”. Um recurso pode ser essencialmente qualquer conceito coerente e significativo a que este é dirigido e uma representação de um recurso é tipicamente um documento que capta o estado atual ou previsível de um recurso [7].

Por fim, a base de dados foi planeada utilizando a ferramenta Visio da Microsoft e implementada no MySQL Workbench, sendo ambas excelentes ferramentas de trabalho. A base de dados foi desenvolvida em na linguagem de MySQL.

A base de dados podia ser relacional ou não-relacional. Depois de muita ponderação, optou-se pela base de dados relacional. Esta decisão foi devido às bases de dados relacionais terem uma base matemática que é codificada numa linguagem de programação apropriada como o SQL [9]. Embora não seja sempre possível definir o esqueleto e arquitetura da solução desde início, quando é possível o melhor a utilizar é uma base de dados relacional [10]. Esse foi o caso da solução em questão em que foi possível desde logo fazer um estudo e planear a arquitetura da nossa solução numa forma bastante elaborada. Utilizando uma base de dados relacional foi possível chegar à conclusão de quais seriam as classes, atributos e relações entre todos estes componentes que permitissem o desenvolvimento da solução em questão.

Como se pode observar na Figura 29, existem diversas tabelas que vão ser responsáveis pelo armazenamento de informação requerida para resolver o problema inicialmente proposto. As tabelas da base de dados são: Tabela “User” para os diferentes clientes do sistema, a tabela para o hotel, a tabela para as reservas, a tabela para os serviços, a tabela para os quartos e outras mais que são importantes para o bom funcionamento da aplicação.

A tabela dos utilizadores irá ligar com diversas tabelas. Uma dessas tabelas é a tabela “HotelServiceUser” que será responsável para guardar todos os serviços do hotel que o

utilizador pediu. Os serviços disponíveis que o utilizador poderá pedir estão presentes na tabela "HotelServiceItem" que está sempre associado a uma descrição (tabela "Description") e cada descrição ao seu código de língua (tabela "LanguageCode"). Desta forma a aplicação poderá ser traduzida numa forma mais clara e coerente. Outra tabela relevante que está ligada à tabela dos utilizadores é a tabela "HotelReservation" que será responsável por guardar a informação das reservas feitas pelo utilizador.

A tabela "HotelReservation" terá várias ligações: estado do pagamento (tabela "Status Payment"), o método de pagamento (tabela "MethodPayment"), o estado da reserva (tabela "ReservationStatus"), e a tabela que representa o quarto ou os quartos presentes naquela reserva (tabela "RoomReservation").

A tabela "RoomReservation" irá ter a si associado a tabela HotelReservation para que seja guardado a informação da reserva. Irá ter associado o hotel correspondente à reserva (tabela "Hotel") e também irá ter associado a tabela dos quartos que fazem parte da reserva (tabela "Room"). A tabela "Room" irá ter associado a si o estado do quarto (tabela "RoomStatus"), o tipo de quarto (tabela "RoomType").

A tabela "Hotel" irá ser a tabela para guardar informações sobre o Hotel. Esta tabela tem algumas ligações associadas tais como: a tabela "HotelContacts" (contatos de hotel) para guardar todos os contatos do hotel, a tabela "Room" para guardar que quartos o hotel possui e também a tabela "HotelServiceUser" para guardar o registo dos serviços que foram pedidos pelos utilizadores.

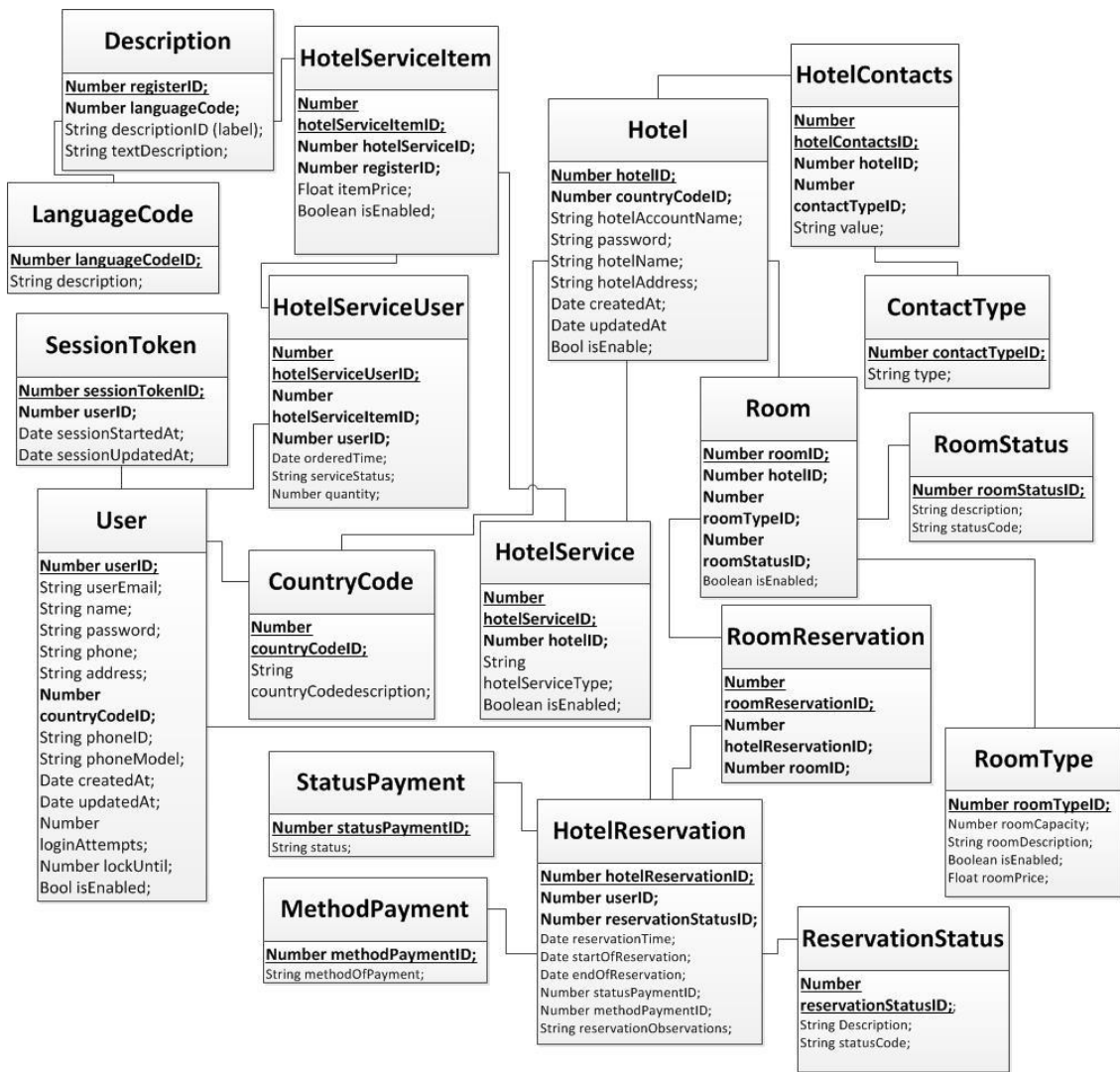


Figura 29 - Base de Dados da Solução

4 Construção da Solução

De seguida irá ser explicado como foi construída a solução assim como o seu funcionamento.

4.1 Fluxo de Funcionamento

De seguida irá ser explicado o funcionamento da aplicação. Para além disso irá ser explicado o que consiste cada menu e para que serve. A aplicação do lado dos clientes chama-se: “MyAccommodation”. Posteriormente, irá ser explicado a interação NFC e o seu fluxo.

4.1.1 MyAccommodation

Esta aplicação vai estar no lado dos clientes. Através desta aplicação, os clientes vão poder realizar as operações básicas de autenticação (registar, fazer login, logout e alteração da palavra-chave), vão poder proceder à marcação de reservas, consultar as suas reservas efetuadas, pedir serviços ao quarto, saber informações do hotel, efetuar o checkin/checkout e abrir a porta do quarto.

Na Figura 30 irá ser apresentado um diagrama de explicação dos ecrãs desta aplicação.

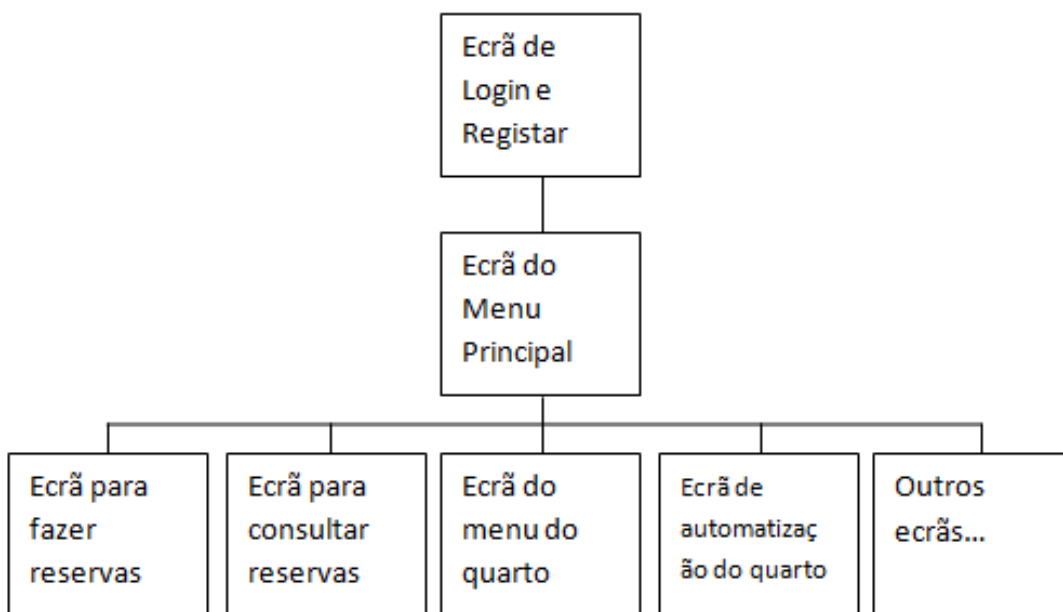


Figura 30 - Diagramas de ecrã principal da aplicação MyAccommodation

4.1.1.1 Ecrã de Login

Na Figura 31 pode-se observar o menu do Login. Este é o primeiro menu da aplicação.

O utilizador tem duas opções. Por um lado se ainda não tem conta criada pode apenas clicar no botão mais à esquerda que diz “register” para poder criar uma conta, por outro lado pode completar os campos de preenchimento e efetuar o login clicando no botão mais à direita que diz “login” depois de preencher os dados.

Este menu contém dois campos onde o utilizador pode introduzir o seu email e a sua palavra-chave com os quais se registaram na aplicação.

Para além das respetivas pistas ou hints, este menu permite ainda gravar sessão para depois de fechar a aplicação não ter que voltar a reintroduzir as suas credenciais. Para fazer isso, o utilizador basta antes de efetuar o login clicar na check-box onde diz “Remember me?” que significa “lembrar-me”. Em caso de efetuar o login com sucesso irá para o Menu Principal da aplicação. A opção lembrar-me utiliza a classe SharedPreferences para conseguir executar a operação de guardar os dados.

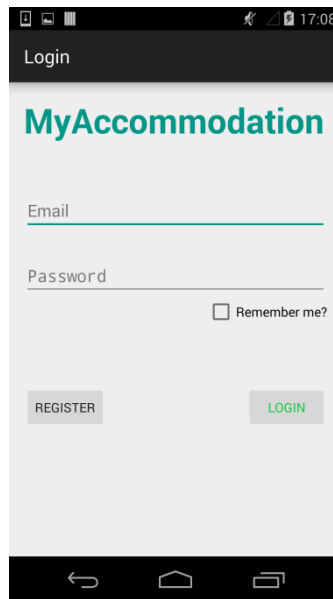


Figura 31 - Ecrã do Login

4.1.1.2 Ecrã do registo

Na Figura 32 pode-se observar o menu onde os utilizadores se podem registar.

Tal como o menu anterior, este menu permitem saltar para o login caso o utilizador ainda já possua conta. Para tal basta clicar no botão da direita que diz “login”.

Este menu possibilita aos utilizadores efetuarem o seu registo sendo apenas pedidos a estes para introduzirem o seu nome, o seu email e a palavra-chave e a sua respetiva repetição. Importante ainda referir que todos os campos usam hints como pistas para os utilizadores saberem o que têm que preencher.

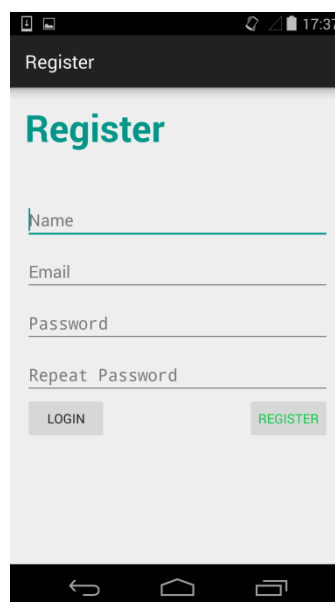


Figura 32 - Ecrã do Registo

4.1.1.3 Ecrã do menu principal

Na Figura 33 pode-se observar o menu principal da aplicação.

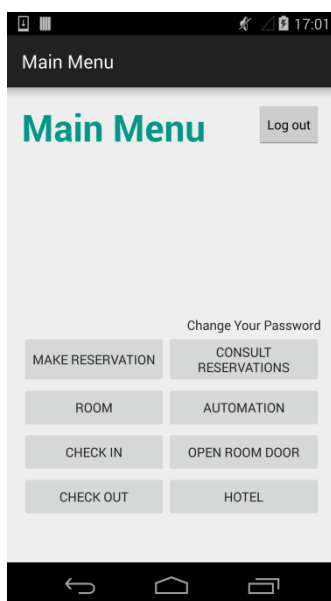


Figura 33 - Ecrã do Menu Principal

Neste menu o utilizador poderá navegar pelos diversos menus da aplicação para desta forma poder usufruir e desfrutar das múltiplas funcionalidades desta. Existem um botão para fazer logout que o utilizador poderá clicar se desejar sair da aplicação e apagar os seus dados de sessão da aplicação. Este botão fará a aplicação voltar ao menu de login anteriormente explicada.

Depois existe ainda também um texto onde o utilizador pode clicar e mais oito botões de navegação. O texto que se consegue ler “Change Your Password” serve para o utilizador trocar a palavra-chave. Os oito botões servem para:

- Make Reservation – O utilizador clicando aqui irá para um menu onde poderá efetuar reservas.
- Consult Reservations – O utilizador ao clicar neste botão irá para um menu onde poderá visualizar as suas reservas.
- Room – Ao carregar neste botão o utilizador poderá visualizar qual é o número do seu quarto e pedir serviços para o seu quarto. Esses serviços variam de hotel para hotel mas poderão ser serviços de cafetaria, lavandaria, entre outros.
- Automation – Este menu irá servir para o utilizador enviar comandos via Bluetooth para o quarto como por exemplo abrir uma janela ou preciana, ligar uma luz.
- Check In – Este menu corresponde à utilização da tecnologia NFC. É um menu que apenas diz ao utilizador que pode encostar o seu dispositivo a um designado leitor que este está pronto para efetuar uma comunicação. Importante realçar que não é obrigatório o

utilizador estar neste menu para que haja comunicação, sendo possível, com um apenas aproximar do smartphone, haver comunicação.

- Open Room Door – Este menu tem a mesma utilidade que o menu Check In anteriormente explicado, servindo para o mesmo efeito.
- Check Out – Este botão permite ao utilizador efetuar o Check Out do hotel. Este botão é um botão de ação não levando o utilizador para outro menu, servindo apenas para este efetuar o Check Out.
- Hotel – Este botão servirá para o utilizador ir para um menu onde poderá visualizar informações sobre o hotel em que está com o checkin efetuado.

4.1.1.4 Ecrã de alteração da palavra-chave

Na Figura 34 pode-se visualizar o ecrã em que o utilizador poderá proceder à alteração da sua palavra-chave se assim o desejar. No primeiro campo, o utilizador terá que colocar a palavra-chave atual e nos próximos dois campos terá que colocar a nova palavra-chave.

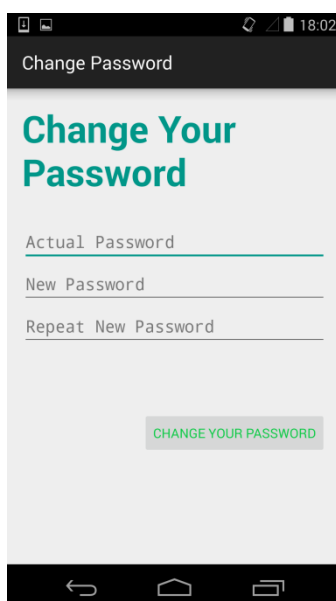


Figura 34 - Ecrã do Menu de Alteração da Palavra-Chave

4.1.1.5 Ecrã de fazer reserva – Make Reservation

Estão representadas na Figura 35 duas imagens que fazem parte do ecrã onde os utilizadores fazem reservas. Assim que a página é carregada, existe um campo que diz o nome do hotel. Depois existem duas datas que serão a data de entrada e a data de saída.

Primeiro os utilizadores terão que selecionar qual o hotel que desejam fazer a reserva, de seguida terão que selecionar o intervalo dos dias que desejam lá ficar e depois irão aparecer todos os tipos de quartos disponíveis naquele hotel para aquela data.

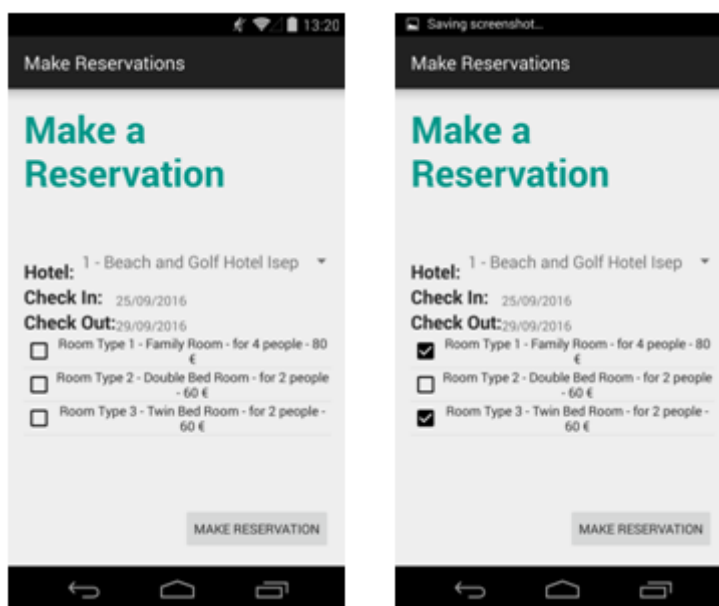


Figura 35 - Ecrã do Menu para efetuar Reservas

4.1.1.6 Ecrã de consultar reservas – Consult Reservations

Na Figura 36 serão apresentadas todas as reservas ainda ativas do utilizador. Ainda ativas significa que são reservas que ainda não acabaram ou que ainda estão para chegar.

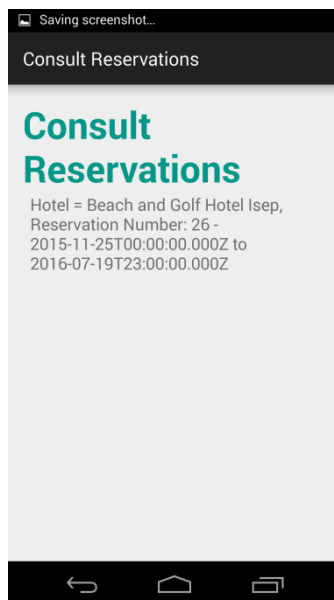


Figura 36 - Ecrã do Menu para consultar as Reservas

4.1.1.7 Ecrã de pedir serviços ao quarto – Room

A Figura 37 representa o menu do quarto onde o utilizador poderá pedir serviços para o quarto.

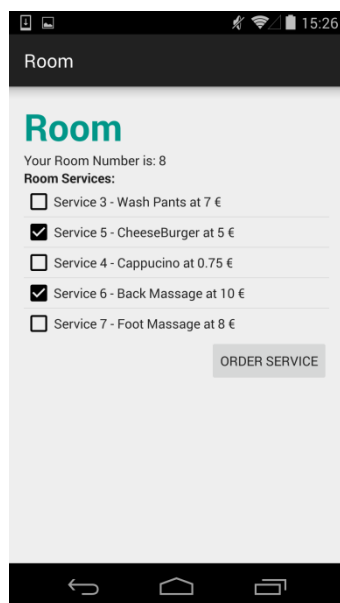


Figura 37 - Ecrã do Menu para pedir serviços ao quarto

4.1.1.8 Ecrã de enviar comandos ao quarto – Automation

Na Figura 38 pode-se ver o menu de automação do quarto onde o utilizador poderá interagir com o seu quarto.

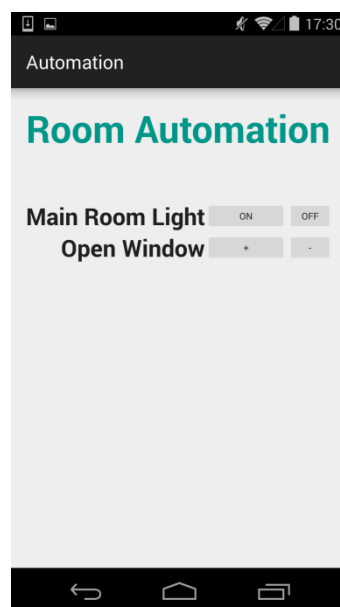


Figura 38 - Ecrã do Menu de interação com o quarto

4.1.1.9 Ecrã para ações NFC – Check In/Open Room Door

Na Figura 39 apenas servirá para explicar ao utilizador que o dispositivo está pronto para fazer a comunicação NFC. A parte que diz “Sending Orders” é um fragmento. Um fragmento em Android diz respeito a um certo comportamento que uma parte do ecrã da aplicação poderá ter. Neste caso esse fragmento representa a informação que aparece em

baixo no ecrã. Quando existir então a comunicação NFC, o texto “Sending Orders” será substituído pelos dados passados nesta comunicação.

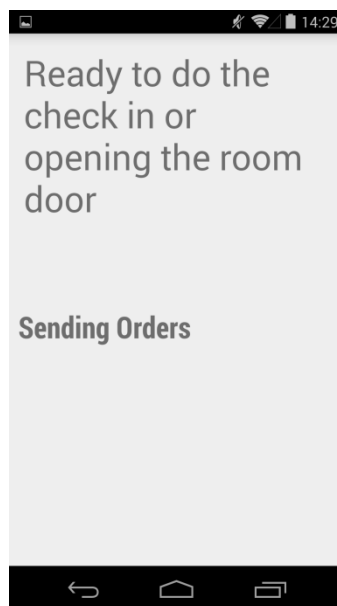


Figura 39 - Ecrã do Menu para fazer ações NFC

4.1.1.10 Ecrã para visualizar informações do hotel – Hotel

Na Figura 40 está representado o menu onde o utilizador irá poder visualizar as diversas informações disponibilizadas pelo próprio hotel.

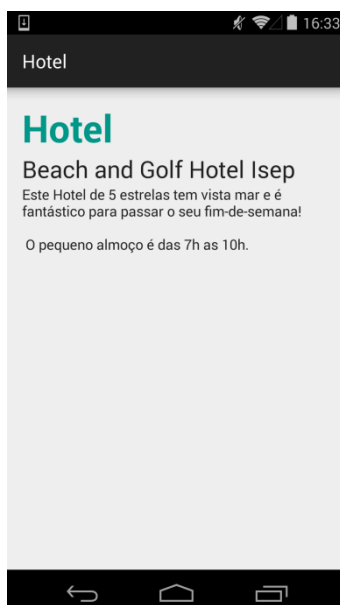


Figura 40 - Ecrã do Menu para visualizar informações do Hotel

4.2 Interação com NFC

A aplicação está estruturada para poder recolher informação através da tecnologia NFC. Na aplicação cliente existem um conjunto de permissões que têm que ser dadas no ficheiro manifesto em relação à comunicação via NFC.

O ficheiro manifesto é obrigatório em qualquer aplicação Android e é responsável por várias questões [49]:

- Dar o nome ao package principal da aplicação.
- Descrever todos os componentes da aplicação – as activities, os serviços, os broadcast receivers e os content providers pelos quais a aplicação é composta.
- Nomear todas as classes que implementam cada um dos componentes e publica as suas capacidades (por exemplo, que tipo de mensagens estes conseguem lidar). Estas declarações fazem com que o sistema Android possa saber que componentes existem e com que condições é que podem ser lançadas.
- Determinar que processos serão responsáveis pelos componentes de aplicação.
- Declarar que permissões uma aplicação deve ter para aceder partes protegidas da API e interagir com outras aplicações. Importante referir neste ponto que visto que esta aplicação irá utilizar a tecnologia NFC, também é necessário dar explicitamente a esta tecnologia no ficheiro manifesto.
- Também tem a responsabilidade de declarar que permissões as outras aplicações terão que ter para poderem comunicar com a sua aplicação com sucesso.
- Listar as classes de instrumentação que fornecem perfis e outras informações enquanto a aplicação está a correr. Estas declarações estão presentes no manifesto apenas enquanto a aplicação está a ser desenvolvida e testada, sendo retiradas antes da sua publicação.
- Declarar o valor mínimo da API Android que a aplicação precisa para funcionar corretamente.
- Listar todas bibliotecas que a aplicação necessita.

Para a comunicação acontecer é preciso primeiro dar permissão no ficheiro de manifesto do Android para que esta aplicação possa utilizar e comunicar através de NFC através do código representado pela Figura 41.

```

<service android:name="nfc.CardService"
    android:exported="true"
    android:permission="android.permission.BIND_NFC_SERVICE">
    <intent-filter>
        <action
            android:name="android.nfc.cardemulation.action.HOST_APDU_SERVICE"/>
        <category android:name="android.intent.category.DEFAULT"/>
    </intent-filter>
    <meta-data android:name="android.nfc.cardemulation.host_apdu_service"
        android:resource="@xml/aid_list"/>
</service>

```

Figura 41 - Código no ficheiro manifesto para o funcionamento de NFC

Como podemos observar na Figura 41 existe um código específico para garantir o bom funcionamento da tecnologia NFC. Basicamente este código consiste na implementação dum serviço responsável por manipular a comunicação NFC entre os terminais.

Numa primeira fase o código está a dizer à aplicação que dá permissão para este poder usar o hardware que diz respeito ao NFC.

De seguida, é iniciado um Intent. Um Intent é uma descrição abstrata de uma operação a ser executada. E a operação executada neste caso é a comunicação NFC no modo de emulação de cartão. Também aqui é definida a classe nativa do Android responsável pela comunicação NFC. Essa classe é a HostApduService que terá que ser estendida na classe responsável por receber os dados NFC.

No final, é dado a conhecer ao sistema um ficheiro de configuração XML que lista os números identificadores das aplicações para qual esta comunicação irá fazer. Isto também define o protocolo ou os protocolos de emulação de cartão que a aplicação suporta.

Na comunicação principal, para além destas permissões que têm que ser dadas, são implementadas quatro classes que servem para suportar a comunicação NFC:

- AccountStorage – Classe que serve para instanciar variáveis importantes à comunicação NFC que são responsáveis pelo bom funcionamento da API. Algumas variáveis presentes nesta classe são strings que servem para guardar e comparar informação. Para além destas variáveis também existe uma variável de sincronização.
- CardEmulationFragment – Classe que é responsável apenas pelas variáveis de texto que aparecem no ecrã do utilizador.
- CardService – Classe que vai fazer o extend da classe nativa Android responsável pela comunicação NFC (HostApduService). Classe que contém diversas variáveis próprias do tipo string e arrays do tipo byte. A manipulação e comparação dessas variáveis juntamente com diversas operações da classe estendida irão garantir a boa comunicação NFC. O método mais

importante é o método `processCommandApdu`. Este método vai permitir enviar o `UserId` que deseja efetuar o comando NFC e o token identificativo do seu dispositivo móvel.

- `SampleActivityBase` – esta classe será uma classe de instanciação apenas para o fragmento que vai ser mostrado no ecrã. Esse fragmento será para demonstrar que informação está a ser passada para o leitor.

4.2.1 Fluxo NFC

A interação NFC na aplicação acontece quando o utilizador aproxima o seu dispositivo a um leitor NFC. No caso da aplicação `MyAccommodation` basta o utilizador ter a aplicação instalada e aproximar o seu dispositivo num leitor apropriado que a comunicação terá lugar.

Como se pode observar na Figura 42, o utilizador começa por aproximar o seu dispositivo móvel a um leitor NFC para efetuar o check-in. Nesta comunicação a aplicação cliente irá transmitir o ID correspondente ao utilizador que aproxima o dispositivo que, neste caso, deseja efetuar o check-in ou abrir a porta do seu quarto. Também nesta comunicação irá ser passado uma variável denominada de `Notification Token` que servirá para identificar o dispositivo da aplicação cliente para que depois esta possa receber a resposta do servidor.

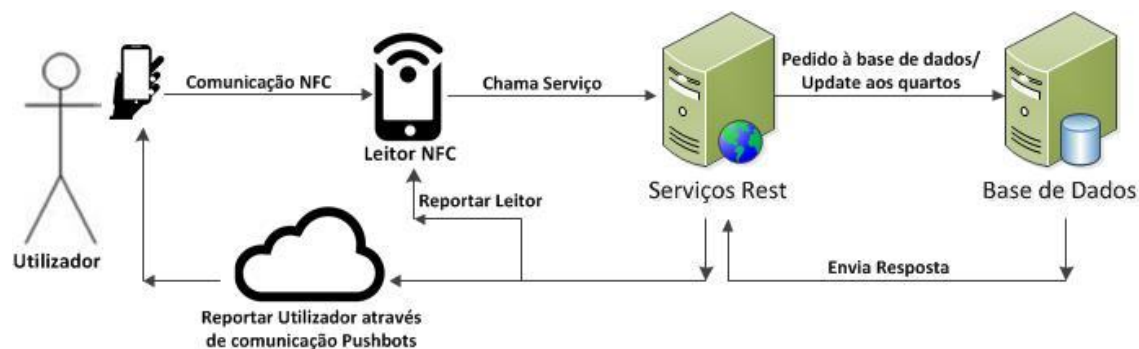


Figura 42 - Diagrama do Fluxo de NFC do sistema

Posteriormente, a aplicação do leitor irá chamar um serviço, enviando-lhe o ID do utilizador que pretende efetuar o check-in. O leitor de check-in irá chamar o serviço responsável pelo check-in presente no servidor que recebe uma variável que é o ID do utilizador que quer efetuar o check-in. O serviço faz uma query à base de dados para verificar se o utilizador tem uma reserva feita no hotel naquele dia (no dia em que o utilizador aproxima o seu dispositivo ao leitor).

Caso seja um leitor de abertura de porta, então o serviço que vai ser chamado é o serviço de abertura de porta que por sua vez vai comunicar com a base de dados para ver se aquele `userID` tem uma reserva para aquele quarto em específico. Ou seja, para do `userID` também vai enviar o `roomID` (variável identificativa do quarto).

A base de dados irá responder à query feita pelo serviço. No caso do checkin e caso a resposta da base de dados seja positiva então o serviço irá também verificar quantos e quais

os quartos que estão presentes na reserva visto que o utilizador pode efetuar uma reserva com um ou mais quartos. Depois irá fazer um update na tabela dos quartos na base de dados para que os quartos fiquem no estado de ocupado.

Posteriormente irá informar duas entidades: o leitor NFC que irá mostrar o resultado no seu ecrã para que o cliente possa ver o resultado do seu check-in ou da sua tentativa de abrir a porta e também irá enviar uma notificação através da tecnologia Pushbots à aplicação utilizador através da variável (Notification Token) inicialmente passada que identifica o dispositivo da aplicação cliente.

Caso a resposta seja negativa, então o serviço irá fazer o mesmo, exceto o update na base de dados da situação dos quartos.

4.3 Servidor REST

Um aspeto fulcral para o funcionamento do sistema são os serviços implementados que irão estar prontos para responder a qualquer pedido da aplicação.

Os serviços foram implementados no ambiente de desenvolvimento NetBeans e estão estruturados em quatro camadas que irão ser explicadas de seguida. Foram definidas estas quatro camadas porque desta forma os serviços ficariam muito mais organizados na medida em que cada camada tem a sua responsabilidade.

A primeira camada, representada pela Figura 43, é a camada do servidor que tem que estar a correr para a aplicação poder comunicar com os serviços. Este ficheiro tem os métodos de start e de stop que servirão para iniciar ou parar o servidor caso seja necessário. No método start também se pode verificar que foi definida por defeito a porta 3000 pelo que as rotas terão que passar por esta porta.

Embora não esteja representado na Figura 43, esta camada também possui uma variável que vai ser responsável por requerer acesso à segunda camada, sendo ela:

- `var rest = require("./rest.js");`

```

REST.prototype.configureExpress = function(connection) {
    var self = this;

    //Body Parser é o responsável por interpretar os http requests e os responses
    app.use(bodyParser.urlencoded({ extended: true }));
    app.use(bodyParser.json());

    var router = express.Router();
    app.use('/api', router);
    var rest_router = new rest(router,connection,md5);
    self.startServer();
}

REST.prototype.startServer = function() {
    var port = process.env.PORT || 3000;
    app.listen(port, function(){
        console.log("[STARTED] Express server listening on port %d in %s mode", port, app.settings.env);
    });
}

REST.prototype.stop = function(err) {
    console.log("ISSUE WITH MYSQL \n" + err);
    process.exit(1);
}

new REST();

```

Figura 43 - Camada do servidor – server.js (primeira camada)

Este ficheiro denominado de server.js também contém o método denominado de “configureExpress” que será o responsável por fazer a ligação com a segunda camada e fazer uma rota para os serviços poderem ser chamados. Também importante realçar que este método diz que todas as rotas que serão designadas começaram por “/api”.

Esta primeira camada comunica então com a segunda camada que é o ficheiro denominado de rest.js. O ficheiro rest.js, que atua como segunda camada, contém todos os recursos, todos os seus métodos e seus caminhos que têm que ser chamados para a aplicação conseguir obter os resultados desejados.

Na Figura 44 está representado o recurso “User” ou utilizador e pode observar pela mesma figura que estão representados três métodos diferentes para este recurso:

- Método Get User – método designado para encontrar um utilizador enviando-lhe o userID ou número de identificação do utilizador. O caminho para este método é “/api/users/userID”. O “/api” foi definido na primeira camada do servidor pelo que todas as rotas começaram por aqui. Depois “/users” que está a dizer que irá selecionar a partir dum conjunto de utilizadores, no caso do Rest é geralmente mencionado como uma coleção, neste caso de utilizadores. E por final o userID que irá ser substituído pelo ID do utilizador que se deseja encontrar;
- Método Get All Users – método designado para encontrar todos os utilizadores do sistema, não sendo necessário enviar nenhuma informação. O caminho para este método é “/api/users”. Neste caso, pode-se observar que a rota diz respeito à coleção dos utilizadores;

- Método Create User - método designado para criar um utilizador no sistema, sendo necessário enviar um conjunto de informação. Esta informação será enviada num body. Esse conjunto de informação que será enviado será o nome do novo utilizador, o seu email e a sua password. O caminho para este método é `"/api/users"`, sendo que desta vez irá ser aplicado o método POST ao invés de do método GET como é no método `"Get All Users"`.

Então por exemplo, para ser chamado o método `"GET USER"` teria que ser executado um GET para este caminho: `http://192.168.0.102:3000/api/users/16` sendo:

- `http://192.168.0.102` – o endereço da máquina na qual o servidor está a correr;
- `3000` – a porta definida no servidor;
- `/api` – o caminho inicial de todas as rotas também definida no servidor;
- `/users` – a coleção dos utilizadores na qual o utilizador que se pretende encontrar estará inserido;
- `/16` – o número identificador do utilizador que se pretende que o sistema retorne.

Embora a Figura 44 apenas mostre o recurso User, esta camada possui todos os recursos para que o servidor possa efetuar ações relacionadas com os mesmos. Os recursos presentes nesta camada são:

- Recurso User ou utilizador;
- Recurso Hotel;
- Recurso RoomTypes ou tipos de quarto;
- Recurso Room ou quarto;
- Recurso Hotel Service ou serviço do hotel em termos de categoria (por exemplo: cafetaria);
- Recurso HotelServiceItem ou cada serviço em específico (por exemplo: café) ;
- Recurso HotelServiceUser ou cada serviço que está a ser consumido por um determinado utilizador;
- Recurso login;
- Recurso logout;
- Recurso HotelReservation ou reserva do hotel;
- Recurso checkin;
- Recurso checkout;

- Recurso description ou descrição;
- Recurso OpenDoor ou abertura de porta.

```

//-----USER-----\\
//Get User
router.get("/users/:userID", function (req, res) {
  var callback = function (r) {
    return res.json(r);
  };

  usersRest.getUserByID(req.params.userID, callback);
});

//Get All Users
router.get("/users", function (req, res) {
  var callback = function (r) {
    return res.json(r);
  };
  usersRest.getAllUsers(callback);
});

//Create User
router.post("/users", function (req, res) {
  var callback = function (r) {
    return res.json(r);
  };
  usersRest.post(req.body, callback);
});

```

Figura 44 - Ficheiro rest.js (segunda camada)

No final da segunda camada está uma variável que serve para o servidor saber que tem que se ligar a este ficheiro para que possa rotear por este caminho. Esta variável é: `module.exports = REST_ROUTER;`

É ainda importante realçar que no início da segunda camada estão representados todos os recursos e o caminho destes de interligação com a terceira camada. Irá ser atribuída a uma variável um requerimento do caminho do ficheiro para que na camada dois possam ser chamados os métodos que estão instanciados na camada três. Portanto, no início deste ficheiro e para o recurso de utilizadores está definido desta forma:

```
var usersRest = require('./lib/models/usersRest');
```

Então podemos verificar que a terceira camada vai ser constituída por diversos ficheiros. Os ficheiros vão ter o nome constituído por o nome do recurso mais a palavra “Rest”. Ou seja, para o caso do ficheiro da terceira camada que representa os métodos instanciados que dizem respeito ao recurso “User”, o nome do ficheiro será “usersRest.js”.

O exemplo de um dos possíveis ficheiros presentes na terceira camada é o ficheiro dos Utilizadores ou “users” tal como está representado na Figura 45. Podemos observar na figura os diversos métodos relacionados com o recurso “user” representados.

```

function UsersRest(){
  return {

    //Get a specific User
    getUserByID: function(Id, callback){
      console.log('Getting User with ID: ' + Id);
      return userModel.getUserById(Id, callback);
    },

    //Get all Users
    getAllUsers: function(callback){
      console.log('Getting All Users...');
      return userModel.getAllUsers(callback);
    },

    //Create a new User
    post: function(data, callback){
      console.log('Creating a new User...');
      return userModel.createNewUser(data, callback);
    },

    //Delete a specific User
    delete: function(Id, callback){
      console.log('Deleting a specific User...');
      return userModel.deleteUser(Id, callback);
    },
  },
}

```

Figura 45 - Ficheiro da instanciação do recurso dos utilizadores – usersRest.js (terceira camada)

É importante realçar ainda duas variáveis importantes nesta terceira camada:

- `module.exports = UsersRest();` – variável que se encontra no final do ficheiro. Esta variável serve para a camada dois conseguir rotear para este ficheiro quando o sistema requerer algum método sobre o recurso “user” ou utilizador.
- `var userModel = require('./user');` – variável que se encontra no início do ficheiro. Esta variável serve de ligação entre a camada três e a camada quatro pelo que irão ser invocados os métodos presentes na camada quatro. Por exemplo: `userModel.getUserById(Id, callback)`, o que quer dizer que, irá ser chamado sobre o `userModel` (camada quatro) o método `getUserById`.

A quarta camada, representada na Figura 46, consiste na implementação dos serviços que estão instanciados na terceira camada. Aqui existem vários cuidados com a ligação à base de dados, com exceções e também com os possíveis resultados que os serviços puderam obter da base de dados.

```

function User() {
  return {
    getUserById: function(Id, callback){

      //Get DataBase Connection
      pool.getConnection(function(err, connection) {

        //Variable responsible for the response
        var res;

        if (err) {
          connection.release();
          res = {
            success: false,
            msg: constants.MESSAGES.ERROR.GENERIC
          };
          return callback(res);
        }

        var getQuery = 'select userID, name from User where userID = ?';

        connection.query(getQuery, [Id], function (err, rows) {
          connection.release();
          if (err || rows.length === 0) {
            res = {
              success: false,
              msg: constants.MESSAGES.ERROR.WRONGDATA,
              result: ( err ? err: rows )
            };
          }
        }
      )
    }
  }
}

```

Figura 46 - Implementação dos serviços (quarta camada)

É importante realçar uma variável presente neste ficheiro:

- `module.exports = User();` – variável que se encontra no final do ficheiro. Esta variável serve para a camada três conseguir rotear para este ficheiro quando o sistema requerer algum método sobre o recurso “user” ou utilizador;

5 Avaliação da solução

Para além da aplicação ter que ser testada continuamente durante o desenvolvimento da mesma é preciso definir um conjunto de parâmetros para poder fazer uma avaliação constante das diferentes funcionalidades do projeto e verificar se estas estão a fazer sentido ou não.

Os parâmetros que vão ser considerados são: a eficácia das funcionalidades e o tempo em que as ações demoram a concretizar-se.

A eficácia das funcionalidades será medida através duma análise para determinar se de facto as ações que o utilizador poderá fazer na aplicação cumprem os objetivos da solução. Alguns exemplos de testes serão mencionados mais à frente.

O tempo que as ações demoram a concretizar-se não deverá ser muito longo para que o utilizador não se canse da aplicação, por isso foi definido que o cliente não poderia passar mais de 5 segundos à espera que a aplicação reagisse sendo uma mudança de menu, sendo um requisito de algum serviço.

Uma aplicação cliente foi desenvolvida para os clientes hóspedes dos hotéis. Essa aplicação foi desenvolvida de forma a permitir aos utilizadores poderem realizar as ações permitidas pelo sistema, isto é, fazer e consultar reservas, fazer check-in e abrir portas com a tecnologia NFC, pedir serviços ao hotel, etc.

Porém outra aplicação foi também desenvolvida para poder testar se a primeira aplicação estaria a funcionar corretamente no que diz respeito à tecnologia NFC.

A aplicação principal e a aplicação de testes como interagem uma com a outra vão ser uma forte componente de testes, podendo dizer-se assim que uma irá testar a outra em termos de funcionalidades NFC.

A aplicação de teste denomina-se HotelCardReader e de seguida irá ser explicada.

5.1 HotelCardReader

Esta aplicação é a aplicação do lado dos hotéis.

Antes de a aplicação estar funcional o utilizador terá que escolher se quer que a aplicação se comporte como um leitor de check-in ou como um leitor de porta.

A aplicação HotelCardReader é constituída por três ecrãs apenas:

- Ecrã inicial onde o utilizador poderá escolher a configuração da aplicação.
- Ecrã pronto para receber check-ins.
- Ecrã pronto para receber pedidos para abrir a porta.

5.1.1 Ecrã inicial

Este ecrã (Figura 47) é logo o que aparece inicialmente e é onde o utilizador poderá optar por configurar a aplicação para estar apta a fazer checkin ou apta para verificar permissão para abrir a porta dum quarto.

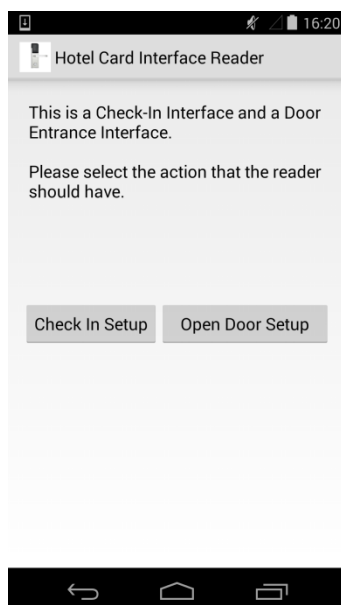


Figura 47 - Ecrã do Menu para configuração da aplicação do leitor

5.1.2 Ecrã de checkin/abrir porta

Quando a aplicação está neste ecrã (Figura 48) significa que está à espera que haja alguma interação com um dispositivo móvel dum utilizador e fará a ações para qual foi configurada no menu inicial.

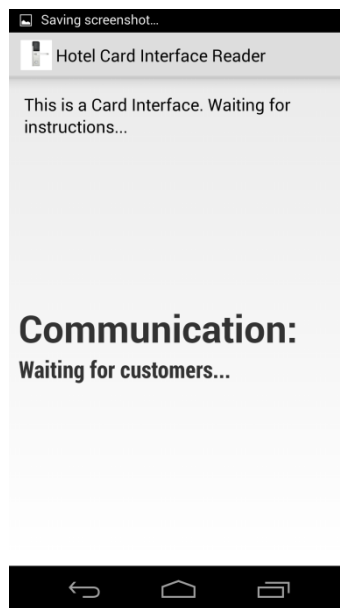


Figura 48 - Ecrã do Menu que fica à espera de dispositivos NFC

Na aplicação irá aparecer um texto a dizer “Waiting for customers” que significa “à espera de clientes. Após a aproximação do telemóvel do cliente ao leitor este irá aparecer um texto no lugar onde diz à espera de clientes. O texto varia consoante o resultado.

Se a aplicação está à espera de leituras de check-in então o leitor irá dizer:

- Em caso de sucesso: “Check in Successfully, your room number is X”, ou seja, “Check in concluído com sucesso, o número do seu quarto é X”, sendo X o número do quarto atribuído ao cliente.
- Em caso de falha: “Check In failed”, ou seja, “Check In sem sucesso”.

Se a aplicação está à espera de leituras de abertura de porta então o leitor irá dizer:

- Em caso de sucesso: “Welcome to your room”, ou seja, “Bem-vindo ao seu quarto”.
- Em caso de falha: “Open door failed”, ou seja, “abertura de porta sem sucesso”.

5.1.3 API

No que diz respeito a API desta aplicação existiam diversas classes e métodos que permitem a receção dos dados através da tecnologia NFC:

- Classe customHandler: esta classe vai ser responsável por receber os dados NFC e por inicializar a classe Pushbots que depois irá transmitir a resposta à aplicação NFC.
- Classe Main Activity: esta classe representa a activity que está à escuta de informação, ou seja, diz respeito ao ecrã menu checkin/abrir porta.

- Classe LoyaltyCardReader: esta classe vai ser responsável por tratar os dados NFC. Esta classe tem uma variável que se designa de SELECT_OK_SW que se, o tipo de dados passados por NFC for igual a esta variável então vai ter sucesso e vai dar correto. Os dados antes de serem comparados com esta variável têm que ser processados.
- Classe CardReaderFragment: esta classe servirá para pegar na informação e chamar os serviços, ou de checkin ou da abertura de portas.

Também nesta aplicação foi preciso preparar o ficheiro de manifesto Android para poder efetuar a comunicação NFC. Foi introduzido o código representado na Figura 49.

```
<receiver
  android:name="com.pushbots.google.gcm.GCMBroadcastReceiver"
  android:permission="com.google.android.c2dm.permission.SEND" >
  <intent-filter>
    <action android:name="com.google.android.c2dm.intent.RECEIVE" />
    <action android:name="com.google.android.c2dm.intent.REGISTRATION" />
    <category android:name="com.example.vicente.myaccommodation" />
  </intent-filter>
</receiver>
<receiver android:name="com.example.vicente.myaccommodation.customHandler" />
```

Figura 49 - Extrato de código do ficheiro de manifesto Android da aplicação de leitura

Como se pode verificar na Figura 49, é introduzido no ficheiro um receiver. O receiver é um componente responsável por receber ou tratar eventos provenientes do próprio sistema ou de outras aplicações. Este receiver está a registar na aplicação que esta pode receber informação da aplicação “MyAccommodation”, ou seja, da aplicação cliente.

5.2 MyAccommodation

A aplicação cliente (“MyAccommodation”) foi testada nas suas diversas funcionalidades. Neste capítulo irão ser demonstrados alguns exemplos dos testes feitos e como era suposto estes se comportarem.

O primeiro exemplo a ser dado é quando o utilizador tenta fazer uma reserva. Então numa primeira fase, o utilizador seleciona no menu a opção de Make Reservation. Posteriormente, o utilizador terá que proceder à alteração das duas datas que se representam debaixo do nome do hotel, sendo estas: a data do checkin e a data do checkout. Quando o cliente acaba de introduzir as datas, automaticamente o sistema irá mostrar os tipos de quartos que o utilizador poderá reservar, sendo que a cada tipo de quarto existe uma check box que o utilizador terá que usar para dizer ao sistema se quer ou não quer aquele tipo de quarto. Por fim o utilizador carrega no botão que se encontra em baixo à direita que diz “Make Reservation”, ou em português, “Fazer Reserva”. Na Figura 50 está demonstrado a sequência de ecrãs que irão aparecer neste tipo de funcionalidade.

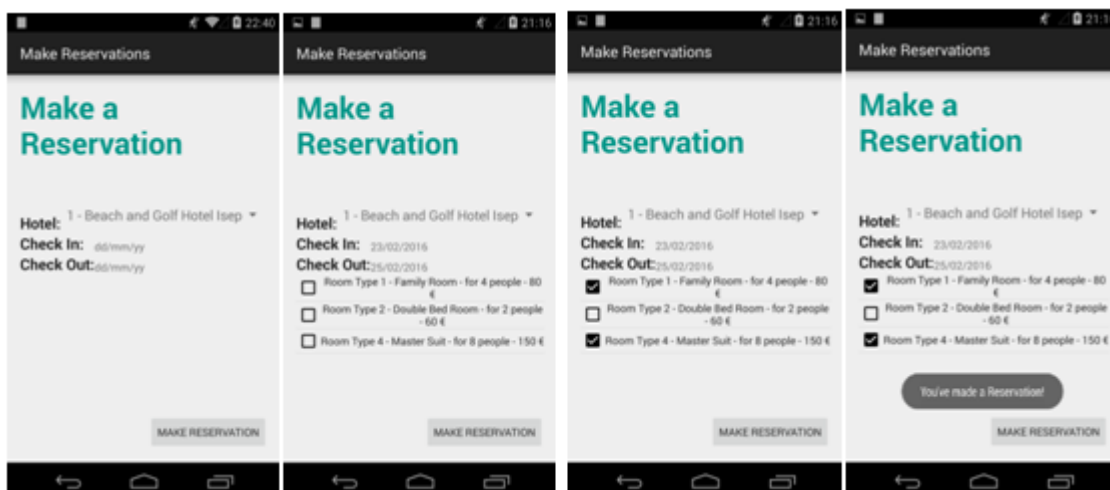


Figura 50 - Sequência de ecrãs de teste de fazer reserva

Outro exemplo que será dado é quando o utilizador tenta efetuar o check-in através do NFC. Numa primeira fase, o utilizador seleciona no menu a opção de Check In. Posteriormente, o utilizador irá aproximar o seu dispositivo ao leitor NFC e depois terá que esperar para receber a notificação passada por PushBots no seu dispositivo.

Na Figura 51 estão representados a sequência de ecrãs especificados para testar esta funcionalidade. De notar que o ecrã que se encontra à direita, ecrã que demonstra a notificação, corresponde ao menu de notificações do dispositivo, pelo que se encontra fora da aplicação.

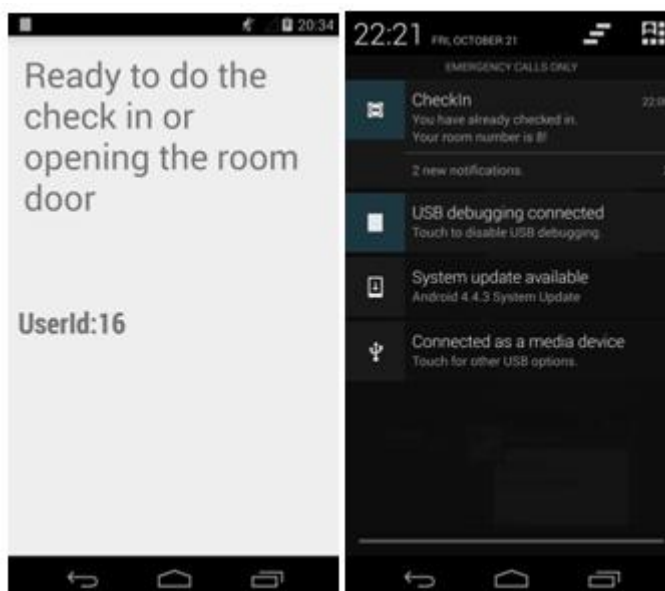


Figura 51 - Sequência de ecrãs de teste de check in

6 Conclusões

Neste capítulo vão ser apresentadas as conclusões finais do trabalho, os resultados obtidos, se o trabalho feito cumpriu ou não cumpriu os objetivos definidos inicialmente e também que melhorias poderão ser desenvolvido no futuro.

Vive-se num mundo que a tecnologia está em constante mudança e o ser humano e a sociedade adapta-se facilmente a estas alterações. A nível de sistemas móveis tem-se notado uma evolução enorme existindo um elevado número de aplicações para um espectro enorme de diferentes ações que os utilizadores poderão desejar que o seu telemóvel faça. Um desses exemplos é este trabalho em que o utilizador pode utilizar a aplicação que não só permita ao utilizador efetuar e consultar as suas reservas, como também permite ao utilizador uma maneira mais fácil e interativa de estar no hotel.

O que foi proposto inicialmente consistia na criação duma aplicação mobile que passava por usar smartphones para automatizar os processos duma unidade hoteleira. Esta aplicação iria utilizar diversos serviços que permitiriam aos clientes fazer o check-in, encomendar um serviço para um quarto, fazer reservas e até abrir portas com o telemóvel utilizando a tecnologia NFC. Também era proposto fazer com que o dispositivo se ligasse ao quarto através de Bluetooth para que o utilizador pudesse enviar ordens ao quarto como por exemplo abrir uma janela ou ligar o ar condicionado.

O resultado obtido no final foi uma aplicação para dispositivos móveis que de fato cumpre os objetivos do trabalho inicialmente propostos com a exceção da parte da domótica do quarto em que o utilizador se ligava ao quarto através da tecnologia Bluetooth. Este ponto não foi cumprido devido à prioridade do trabalho a ser desenvolvido. O trabalho foi sendo desenvolvido para ir cumprindo os objetivos e o menos prioritário acabou por não ser alcançado, neste caso era o objetivo da domótica.

Depois do trabalho houve a ponderação que existiriam algumas melhorias que se podiam considerar para um trabalho futuro. Para além da conclusão de domótica do quarto, também se poderia considerar algumas melhorias a nível do aspeto da interface do utilizador que poderia ser mais trabalhada. Também era importante ter a opção de poder conter o

mapa de como chegar ao hotel desde o local onde o utilizador se encontra, para que desta forma o utilizador pudesse chegar ao hotel mais facilmente. Outra melhoria poderia ser a colocação de sons musicais para cada operação de NFC e respetiva resposta tornando assim a aplicação ainda mais interativa com o utilizador. Quando acontecesse a primeira entrada no quarto por parte do utilizador utilizando o Smartphone, a aplicação poderia dar ao utilizador a possibilidade de poder criar uma palavra-chave para que depois deste tentar abrir a porta novamente lhe pedisse essa palavra-chave de forma a tornar este processo mais seguro.

Referências

- [1] Assa Abloy, "Evaluation of the world's first pilot using NFC phones for check-in and hotel room keys", 2011, <http://www.assaabloy.com/Global/Products/Products-old/ASSA-ABLOY-Mobile-Keys/Report-ASSA-ABLOY-Mobile-Keys-Pilot-Clarion.pdf>; available up to 12/07/2016
- [2] M. Walerych, W. Zabierowski, Application Supporting Hotel Management as an Example of Web Technologies Usage, 2012 ; available up to 08/02/2016
- [3] C. Faria, O QUE É VALOR PARA O CLIENTE?, 2004, http://www.merkatus.com.br/10_boletim/96.htm; available up to 08/02/2016
- [4] D. Monteiro, O Conceito de Valor para o Consumidor – Uma análise sob a óptima da metedologia Laddering , 2010; available up to 08/06/2016
- [5] T. Woodall, Conceptualising 'Value for the Customer': An Attributional, Structural and Dispositional Analysis, 2003
- [6] T. Capan Why The Hell Would I Use Node.js? A Case-by-Case Tutorial, 2013, <http://www.toptal.com/nodejs/why-the-hell-would-i-use-node-js>; available up to 10/02/2016
- [7] P. Glowacki, Why Use "REST" Architecture for Web Services?, 2010, <http://edn.embarcadero.com/article/40467>; available up to 13/02/2016
- [8] MySQL, 2016, <https://www.mysql.com/products/workbench/>; available up to 1/02/2016
- [9] Duffymo, 2011, <http://stackoverflow.com/questions/4811744/what-is-the-difference-between-a-relational-and-non-relational-database>; available up to 13/02/2016
- [10] D. Lamire, 2013, <http://lemire.me/blog/2011/11/14/are-relational-databases-good-for-anything-anymore/>; available up to 10/02/2016
- [11] Prat, 6 Best Resources to learn Android Lollipop Development, 2015, <http://codecondo.com/6-best-resources-to-learn-android-lollipop-development/>; available up to 10/02/2016
- [12] NFCTask Luncher Inc, 2016, https://rapidnfc.com/imgs/new_nfc_task_launcher_2.jpg; available up to 2/02/2016

- [13] S. Clark, Assa Abloy launches Seos mobile keys, 2012, <http://www.nfcworld.com/2012/09/10/317763/assa-abloy-launches-seos-mobile-keys/>; available up to 4/02/2016
- [14] System NFCPorter, 2016, <http://www.nfcporter.com/>; available up to 4/02/2016
- [15] System NFCPorter, 2016, <https://lh6.ggpht.com/tQz4rSD21d5LNzYQMrVcOtoBsnIfc8ZUvmkMimN3M4M2Y8pDVet4ht0AtbE7POymU3U=w300>; available up to 3/02/2016
- [16] NXP Systems, 2015, http://www.nxp.com/files/graphic/nxp/other_type/Three-modes-3.jpg; available up to 4/02/2016
- [17] Google, Host-based Card Emulation, 2016, <http://developer.android.com/guide/topics/connectivity/nfc/hce.html>; available up to 7/02/2016
- [18] MS4B, Safe Element, 2016, <http://www.ms4b.com/soluciones/ecosistema>; available up to 14/02/2016
- [19] D. Garcia, Accommodation Management System using Mobile Devices, 2014
- [20] L. Johnson, Secure elements vs cloud-based HCE: What is more secure for NFC mobile payments?, 2015, <http://www.sequent.com/secure-elements-vs-cloud-based-hce-secure-nfc-mobile-payments/>; available up to 27/02/2016
- [21] NFC Forum, What are the operating modes of NFC devices?, 2013, <http://nfc-forum.org/resources/what-are-the-operating-modes-of-nfc-devices/>; available up to 4/03/2016
- [22] E. Alecrim, O que é NFC (Near Field Communication)?, 2016, <http://www.infowester.com/nfc.php>; available up to 7/03/2016
- [23] NFC Forum, What it Does, 2016, <http://nfc-forum.org/what-is-nfc/what-it-does/>; available up to 12/03/2016
- [24] NFC Forum, NFC Tag Types Platform, 2015, <http://www.oprfid.com/products-NFC-1.jpg>; available up to 13/03/2016
- [25] R. Boden, Starwood Hotels picks Bluetooth for mobile room keys that work on smartphones and Apple Watch, 2014, <http://www.nfcworld.com/2014/11/03/332389/starwood-hotels-picks-bluetooth-mobile-room-keys-work-smartphones-apple-watch-devices/>; available up to 15/03/2016

- [26] S.Clark, BMW uses NFC car keys to open hotel room doors, 2012, <http://www.nfcworld.com/2012/04/23/315235/bmw-uses-nfc-car-keys-to-open-hotel-room-doors/>; available up to 16/03/2016
- [27] Assa Abloy, Evaluation of the world's first pilot using NFC phones for check-in and hotel room keys, <https://www.assaabloy.com/Global/Products/Products-old/ASSA-ABLOY-Mobile-Keys/Report-ASSA-ABLOY-Mobile-Keys-Pilot-Clarion.pdf>; available up to 16/03/2016
- [28] C. Krauß, Access Control in Enterprises with Key2Share, <https://www.sit.fraunhofer.de/en/key2share/>; available up to 2/04/2016
- [29] C. Rinaldi, Google Wallet vs. Apple Pay: O que você precisa saber, 2014, <http://www.androidpit.com.br/google-wallet-vs-apple-pay-comparacao>; available up to 9/04/2016
- [30] Google Wallet – Como funciona?, 2016, <http://www.creditooudebuto.com.br/google-wallet-como-funciona/>; available up to 11/04/2016
- [31] Touch&Travel, Como funciona? , 2015, <https://www.touchandtravel.de/web/cupo/touchandtravel/funktionsweise>; available up to 11/04/2016
- [32] S. Clark, Museum of London adds NFC, 2011, <http://www.nfcworld.com/2011/08/16/39129/museum-of-london-adds-nfc/>; available up to 11/04/2016
- [33] Museum of London, 2013, <http://qfuse.com/blog/wp-content/uploads/2013/07/Museum-of-London2.jpg>; available up to 11/04/2016
- [34] The Wolfsonian-Florida International University, 2015, <http://www.wolfsonian.org/>; available up to 12/04/2016
- [35] Please Touch Museum, 2016, <http://www.pleasetouchmuseum.org/experience/>; available up to 13/04/2016
- [36] Victoria and Albert Museum, 2016, <http://www.vam.ac.uk/visit>; available up to 14/04/2016
- [37] W. Gordon, How to Automate Your Phone for Every Room in the House with NFC Tags, 2013, <http://lifehacker.com/how-to-automate-your-phone-for-every-room-in-the-house-473409963>; available up to 14/04/2016
- [38] M. Knoll, 18 Creative & Useful Ways To Use NFC Tags With Your Smartphone, 2014, <http://trendblog.net/creative-and-useful-ways-to-use-nfc-tags-with-your-smartphone/>; available up to 16/04/2016

- [39] A. Lahtela, RFID and NFC in healthcare: Safety of hospitals medication care, 2008, <http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=4571079&url=http%3A%2F%2Fieeexplore.ieee.org%2Fiel5%2F4562769%2F4571002%2F04571079.pdf%3Farnumber%3D4571079>; available up to 17/04/2016
- [40] Marcus, A. et al. "Using NFC-Enabled Mobile Phones for Public Health in Developing Countries." Near Field Communication, 2009. NFC '09. First International Workshop on. 2009. 30-35. 2009 IEEE.
- [41] L. Higgs, M. Wilson, From Relational to Neo4j, 2016, <http://neo4j.com/developer/graph-db-vs-rdbms/>; available up to 19/04/2016
- [42] J. Serra, Relational databases vs Non-relational databases, 2015, <http://www.jamesserra.com/archive/2015/08/relational-databases-vs-non-relational-databases/>; available up to 21/04/2016
- [43] Mobile Nations, Android History, 2016, <http://www.androidcentral.com/android-history>; available up to 23/04/2016
- [44] TutorialsPoint Simply Easy Learning, Android Architecture, 2016, http://www.tutorialspoint.com/android/android_architecture.htm; available up to 25/04/2016
- [45] TutorialsPoint Simply Easy Learning, Android Architecture, 2016, <http://zdnet1.cbsistatic.com/hub/i/2014/10/04/3149f833-4b82-11e4-b6a0-d4ae52e95e57/becc354a4a1fde2c8be525214f6771da/android-architecture-485b.jpg>; available up to 25/04/2016
- [46] M. Rajput, Why Android Studio Is Better For Android Developers Instead Of Eclipse, 2015, <https://dzone.com/articles/why-android-studio-better>; available up to 25/04/2016
- [47] P. de Sousa, Rest Apis For The Web, 2015, <https://moodle.isep.ipp.pt/acesso/>; available up to 25/04/2016
- [48] Pushbots, Pushbots Inc., 2016, <https://pushbots.com/>; available up to 25/05/2016
- [49] Android, App Manifest, 2016, <https://developer.android.com/guide/topics/manifest/manifest-intro.html>; available up to 25/05/2016
- [50] Smart Alliance, NFC in the United States, 2016, <http://www.smartcardalliance.org/activities-videos-nfc-in-the-united-states/>; available up to 25/05/2016
- [51] Turismo de Portugal, 2016, <http://guiastecnicos.turismodeportugal.pt/pt/museus-monumentos/>; available up to 13/06/2016

- [52] IDC Research, Smartphone OS Market Share, 2015, <http://www.idc.com/prodserv/smartphone-os-market-share.jsp>; available up to 15/06/2016
- [53] Android, Android Studio, 2016, <https://developer.android.com/studio/index.html>; available up to 15/06/2016
- [54] A. Ravulavaru, Pushbots and Cordova – Easy Push Notifications for your app, 2015, <http://thejackalofjavascript.com/pushbots-and-cordova/>; available up to 23/06/2016
- [55] A. Silva, Peso do turismo na economia do país está muito acima da média mundial, 2014, <https://www.publico.pt/economia/noticia/turismo-pesa-mais-no-emprego-e-na-economia-em-portugal-do-que-no-resto-do-mundo-1629417/>; available up to 15/07/2016
- [56] S.V. Dominguez, O Valor percebido como elemento estratégico para obter lealdade dos clientes, 2000, <http://www.regeusp.com.br/arquivos/v07-4art05.pdf>; available up to 1/8/2016
- [57] K. Scheuer, Valor percebido pelo Cliente, 2010, <http://www.administradores.com.br/artigos/marketing/valor-percebido-pelo-cliente/37326/>; available up to 2/8/2016
- [58] S. Araújo, A Quantitative Model for Decomposing and Assessing the Value for the Customer, 2014, <https://repositorio-aberto.up.pt/bitstream/10216/75552/2/101049.pdf>; available up to 2/8/2016
- [59] J. Borbinha, Redes de colaboração: alguns elementos para análise e reflexão, 2004, <http://eprints.rclis.org/10979/1/Borbinha.pdf>; available up to 4/8/2016
- [60] L. Maia e H. Afsarmansh, Collaborative Networks: Reference Modeling, 2008, <https://books.google.pt/books?id=pkkPSQL9UIC&pg=PA121&lpg=PA121&dq=arcon+model+marketing&source=bl&ots=kahZ9BsYoZ&sig=MF0kET9cQjAaDokKk7ttCwuWnFY&hl=pt-PT&sa=X&ved=0ahUKEwjj-tv7xrLOAhWKbxQKHjY0AC6oQ6AEIHjAA#v=onepage&q=arcon%20model%20marketing&f=false>; available up to 5/8/2016
- [61] Statista, 2016, <https://www.statista.com/statistics/276623/number-of-apps-available-in-leading-app-stores/>; available up to 19/9/2016
- [62] Público, 2016, http://www.ahresp.com/news_article.php?id=2588; available up to 19/9/2016
- [63] Eastern Daylight Time, 2016, <http://www.businesswire.com/news/home/20160628005853/pt/>; available up to 5/10/2016

[64]Luis Lino Ferreira, 2014, Android Programming V 1.1/; available up to 5/10/2016

ANEXO A – Análise de Valor

Na perspectiva de negócio e de marketing, o valor pode assumir várias definições. Uma delas pode ser que o valor dum produto é quanto o produto irá satisfazer as necessidades dum determinado consumidor em detrimento a quantos recursos este teve que gastar para o obter. Outra definição interessante é o grau de benefício obtido como resultado da utilização ou daquilo que o consumidor desfrutou do produto ou serviço [3, 4].

O produto em questão irá ter um certo valor. É importante fazer esta avaliação visto que em qualquer negócio o produto irá oferecer algum ou prestar certo serviço aos possíveis clientes.

Antes de fazer a análise de valor da solução em questão é ainda importante descrever as quatro fases importantes do valor de cada produto. Existe uma primeira fase que é a fase antes da compra do produto que ocorre antes do consumidor ainda não ter adquirido o produto. A segunda fase é a fase do momento exato da compra. Posteriormente, a fase que ocorre depois da venda, denominando-se de pós-venda, e por último a última fase que é a fase de utilização [5].

O valor é bastante importante de identificar para que exista um objetivo claro de que é que o cliente vai usufruir mas existem dois conceitos muito importantes e muito distintos no que diz respeito a esta matéria, sendo estes o valor desejado e o valor percebido. O valor desejado é aquele valor que é concentrado apenas nos atributos de valor, sem fazer uma abordagem adequada às várias situações de uso e das consequências dessas situações de uso enquanto que o valor percebido está vinculado ao uso do produto ou serviço, ou seja, está relacionado com a percepção do cliente e não com o posicionamento da empresa fornecedora e envolve a noção de troca de benefícios por custos [56,57].

É importante realçar que relacionamentos longos entre o fornecedor e cliente podem ajudar no processo de criação de valor. Valor esse que pode trazer benefícios a ambas as

partes. Nos mercados de negócio são poucas as empresas que têm conhecimento para avaliar o valor e isso é muitas vezes um tremendo erro [58].

Um conjunto de pessoas ou trabalhadores que trabalham em prole dum objetivo comum formam uma rede de colaboração [59]. As redes de colaboração podem ser mais horizontais ou verticais dependendo de como a empresa está estruturada. Uma rede de colaboração horizontal é se todos os membros tiverem responsabilidades e objetivos semelhantes e pode ser uma rede de colaboração mais vertical se os membros forem heterogéneos em relação ao seu cargo e aos seus objetivos.

Para compreender melhor estas redes foram criados modelos que ajudam a explicar melhor este assunto e um desses modelos é o modelo de ARCON. Segundo este modelo as redes de distribuição estão organizadas em três pilares: as funções que cada elemento desempenha, as relações entre cada um dos colaboradores e outras características do nó da rede tais como a quantidade de ligações entre os membros, a sua localização, entre outros e são estas características da rede que possibilitam a criação de valor [60].

Para melhor realizar o estudo da análise de valor do produto é de boa prática realizar o Modelo Canvas. O Modelo Canvas é constituído por 9 componentes muito importantes. Os 9 componentes são: Segmentos de clientes, a proposta de valor, os canais, o relacionamento com os clientes, as fontes de receita, os recursos chave, as atividades chave, as parcerias chave e a estrutura de custos e vão ser apresentados de seguida:

A.1 Segmentos de Clientes

Um aspeto importante e comum a todo o tipo de negócios é que é preciso identificar os clientes alvo, fazendo assim uma análise da segmentação dos clientes alvo a atingir, de forma a identificar o ambiente ou ambientes onde o produto será introduzido. Tendo isto em conta, os clientes alvo são os hóspedes de unidades hoteleiras que se interessam por novas tecnologias, novidades tecnológicas e que queiram dar uma experiência diferente aos seus hóspedes.

Também como clientes terão que ser considerados os hotéis que queiram investir neste tipo de tecnologia como possibilidade de aumentar a felicidade dos seus clientes. É importante ainda realçar que no grupo de clientes hóspedes só podem ser considerados clientes com Smartphones que possuam a tecnologia NFC para puderem usufruir de todas as funcionalidades do produto pelo que isso pode ser um obstáculo.

A.2 Proposta de Valor

A proposta de valor para os clientes passará por uma aplicação que permitirá aos clientes não só poder realizar ações numa forma mais facilitada como também interagir de forma diferente com a unidade hoteleira que estes estão hospedados.

A.3 Canais

Pretende-se chegar aos clientes através de anúncios na Internet. Através do anúncio pretende-se que se dê a conhecer o produto e suas funcionalidades de forma a tentar empolgar e atrair possíveis clientes.

A.4 Relacionamento com os Clientes

O relacionamento irá apenas acontecer entre a empresa e as unidades hoteleiras pelo que estas terão que saber ajudar os seus clientes diretos. A única altura em que os clientes irão comunicar diretamente com a empresa é quando se registam através da aplicação estando assim aptos para poderem usufruir das funcionalidades do produto.

A.5 Fontes de Receita

Serão as unidades hoteleiras que irão ter os custos de compra e manutenção do produto com o objetivo final dos seus próprios clientes/hóspedes ficarem satisfeitos com os serviços.

A.6 Recursos Chave

O projeto envolve muitos recursos físicos como base do seu funcionamento.

Por parte dos clientes apenas envolve o Smartphone do cliente e este tem que ter a funcionalidade de NFC. Por outro lado, por parte da unidade hoteleira já envolve leitores de check-in NFC na receção e leitores de acesso aos quartos em cada uma das portas dos quartos de todo o hotel. Um aspeto ainda a realçar é que esta solução também requer que quer os clientes ou a unidade hoteleira possuam internet para que os serviços do produto possam comunicar.

A.7 Atividades Chave

As atividades chave necessárias que irão possibilitar a construção e o bom funcionamento de todas as componentes do projeto serão: levantamento de informação relacionado com as tecnologias envolvidas na construção e funcionamento do projeto, verificar e analisar outras ideias, aplicações ou projetos já existentes relacionados com o projeto em questão, identificar os possíveis clientes alvos e suas necessidades, construir um produto que satisfaça essas mesmas necessidades e criar uma estrutura de apoio ao funcionamento do produto para que os clientes fiquem satisfeitos.

A.8 Parcerias Chave

Inicialmente, no projeto em questão não se pretende realizar nenhum tipo de parceria, contudo futuramente talvez se possa fazer certas parcerias com alguns hotéis como forma de publicidade para que os clientes possam observar e, com isso, tentar estender e aumentar as vendas do nosso produto.

A.9 Estrutura de Custos

A estrutura de custos associada a todo o projeto é: o desenvolvimento das duas aplicações, o desenvolvimento dos serviços e base de dados que irão possibilitar o bom funcionamento das aplicações e os possíveis contratos de manutenção que as unidades hoteleiras poderão fazer com a empresa.

A seguir na Figura 52 irá ser apresentado o Modelo Canvas.

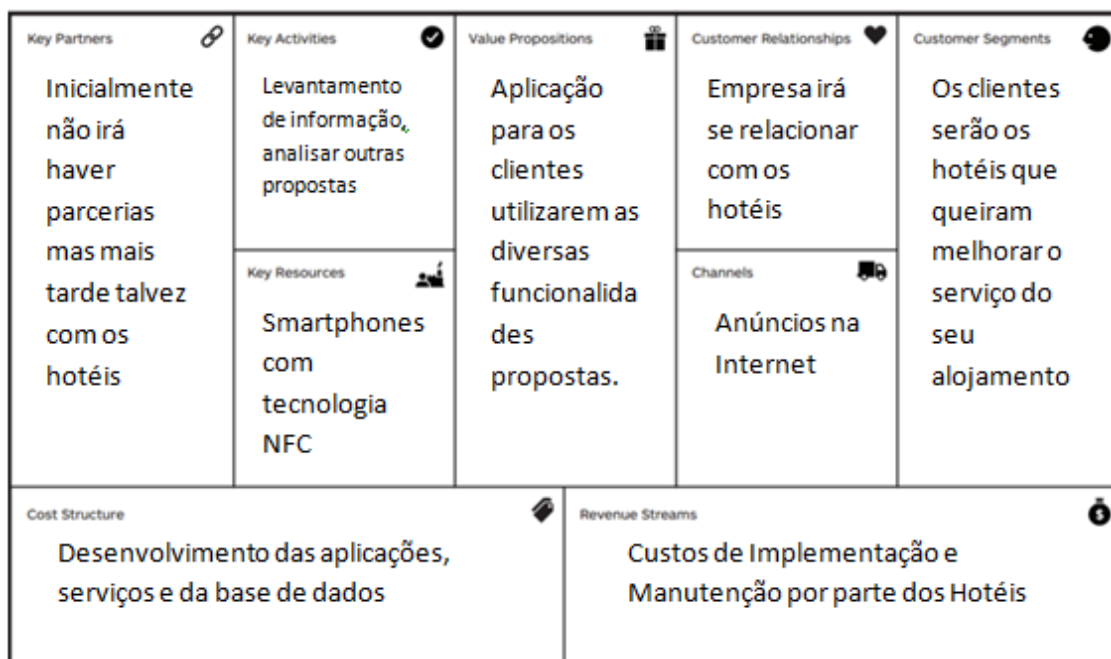


Figura 52 - Modelo Canvas

A ideia de negócio deste projeto irá basear-se em efetuar a venda do produto às unidades hoteleiras para que estas possam dar uma melhor experiência aos seus hóspedes. Os hotéis terão o custo de implementação do sistema, isto é, colocar as interfaces de check-in na receção e em todas as portas dos quartos do hotel e também o custo de manutenção no sistema.

A negociação, ato de tomar uma decisão sobre as duas partes sobre um acordo de benefício mútuo, deverá ocorrer junto dos clientes, neste caso os hotéis.