

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
CURSO DE CIÊNCIA DA COMPUTAÇÃO

PAOLA RAMOS MACALÃO

**Check in² Poa: um aplicativo Android para
turistas em Porto Alegre**

Trabalho de Graduação.

Prof. Dr. Leandro Krug Wives
Orientador

Porto Alegre, dezembro de 2013.

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Reitor: Prof. Carlos Alexandre Netto

Vice-Reitor: Prof. Rui Vicente Oppermann

Pró-Reitor de Graduação: Prof. Sérgio Roberto Kieling Franco

Diretor do Instituto de Informática: Prof. Luís da Cunha Lamb

Coordenador do Curso de Ciência da Computação: Prof. Raul Fernando Weber

Bibliotecário-Chefe do Instituto de Informática: Alexander Borges Ribeiro

AGRADECIMENTOS

Gostaria de agradecer primeiramente aos meus pais por serem a minha inspiração em acreditar que nenhum objetivo é difícil demais para ser alcançado. Todas as dificuldades que vocês superaram ao longo da vida me servem de energia para enfrentar os desafios de cada dia. Obrigada pelo apoio durante o vestibular, a graduação e o trabalho de conclusão. Se hoje eu sou uma vencedora, é porque vocês me ensinaram a sê-la.

Este trabalho não seria possível sem a ajuda e o incentivo do professor Leandro Krug Wives, o qual acreditou na minha ideia e me deu todo suporte necessário para desenvolvê-la. Muito obrigada por todos os ensinamentos passados durante a disciplina de Modelos de Linguagem de Programação e ao longo desse trabalho de graduação.

SUMÁRIO

LISTA DE ABREVIATURAS E SIGLAS.....	6
LISTA DE FIGURAS	7
LISTA DE TABELAS	8
RESUMO	9
ABSTRACT.....	10
1 INTRODUÇÃO.....	11
2 ESTADO DA ARTE	13
2.1 Trabalhos Relacionados	13
2.1.1 Análise Comparativa dos Trabalhos Apresentados	13
2.2 Soluções Existentes	14
2.2.1 Brasil Mobile – Guia Turístico Porto Alegre	14
2.2.2 TS Go! Rio Grande do Sul	14
2.2.3 Porto Alegre Guia.....	15
2.2.4 240 Prazeres de Porto Alegre	15
2.2.5 Field Trip	15
2.2.6 Foursquare	16
2.2.7 Análise Comparativa das Soluções Existentes	16
2.3 Identificação de Requisitos	16
3 PROJETO DO APLICATIVO	18
3.1 Modelagem das Funcionalidades.....	18
3.1.1 Padrão User Stories	18
3.2 Modelagem da Arquitetura.....	19
3.2.1 Padrão MVC	20

3.3 Modelagem do Armazenamento de Dados	21
3.3.1 Banco de Dados SQLite	21
4 PLATAFORMAS.....	23
4.1 Plataformas de Desenvolvimento	24
4.1.1 Eclipse	24
4.1.2 Android Studio	24
4.2 Plataforma Android.....	25
4.2.1 Histórico	25
4.2.2 Arquitetura.....	26
4.2.2.1 Sistema Operacional Linux	27
4.2.2.2 Bibliotecas Nativas	27
4.2.2.3 Android Runtime	27
4.2.2.4 Framework da Aplicação.....	28
4.2.2.5 Aplicação	28
4.2.3 Conceitos Básicos.....	28
4.2.4 Ciclo de Vida de uma Atividade.....	28
4.3 API Utilizadas	30
4.3.1 Facebook.....	30
5 IMPLEMENTAÇÃO DO APLICATIVO	31
5.1 Metodologia de Desenvolvimento	31
5.2 Dados necessários a Aplicação.....	31
5.2.1 Login e Compartilhamento usando o Facebook	32
5.2.2 Coordenadas Geográficas	32
5.2.3 Locais a serem visitados	32
5.3 Funcionamento do Aplicativo	32
5.3.1 Tela de Login.....	33
5.3.2 Tela de Menu Principal.....	33
5.3.3 Tela de Modo Completo	34
5.3.4 Tela de Informações do Local	35
5.3.5 Tela de Check-in.....	36
5.3.6 Compartilhamento no Facebook.....	37
5.3.7 Mapa de como chegar a um local	38
5.3.8 Seleção de Idiomas	38
5.3 Avaliação e Teste do Aplicativo	39
6 CONCLUSÃO	40
6.1 Trabalhos Futuros	40
REFERÊNCIAS	41

LISTA DE ABREVIATURAS E SIGLAS

API	Application Programming Interface
AVD	Android Virtual Device
EMBRATUR	Instituto Brasileiro de Turismo
EUA	Estados Unidos da América
IDE	Integrated Development Environment
JDK	Java Development Kit
MV	Máquina Virtual
MVC	Model-View-Controller
OHA	Open Handset Alliance
SDK	Software Development Kit
SGBD	Sistema de Gerenciamento de Banco de Dados

LISTA DE FIGURAS

Figura 1.1: Categorias de Aplicativos mais baixados na loja Google Play.....	12
Figura 3.1: Interação entre as camadas do padrão MVC.....	20
Figura 3.2: Utilizando o padrão MVC no projeto Android.....	20
Figura 4.1: Mercado mundial de smartphones por Sistema Operacional.....	23
Figura 4.2: Arquitetura da Plataforma Android.....	27
Figura 4.3: Ciclo de Vida de uma Atividade.....	29
Figura 5.1: Tela de Login e Logout.....	33
Figura 5.2: Tela de Menu Principal.....	34
Figura 5.3: Tela de Modo Completo.....	35
Figura 5.4: Tela de Informações do Local.....	36
Figura 5.5: Tela de Check-in.....	37
Figura 5.6: Compartilhamento no Facebook.....	37
Figura 5.7: Mapa de como chegar a um local.....	38
Figura 5.8: Idiomas disponíveis.....	39

LISTA DE TABELAS

Tabela 1.1: Análise comparativa dos trabalhos relacionados.....	14
Tabela 1.2: Análise comparativa das soluções existentes	16
Tabela 3.1: User Stories do Aplicativo.....	18
Tabela 3.2: Descrição da tabela “places”	21
Tabela 3.3: Descrição da tabela “player”.....	22
Tabela 4.1: Versões da Plataforma Android.....	26
Tabela 5.1: Ciclo de Desenvolvimento.....	31

RESUMO

Este trabalho consiste na elaboração e no desenvolvimento de um aplicativo Android no formato de um jogo direcionado a turistas em viagem a Porto Alegre ou moradores que desejem conhecer melhor a cidade em que vivem. O objetivo principal do aplicativo é incentivar o usuário a explorar a cidade de uma forma divertida, realizando check-in em cada um dos lugares disponíveis no jogo para ir acumulando a pontuação referente a cada destino. A ideia do jogo surgiu a partir de uma livre adaptação de um dos jogos de tabuleiro mais famosos no mundo inteiro: “Monopoly”, utilizando como moeda de compra a ação de realizar check-in no local. A partir de uma análise de mercado que demonstrou a predominância das vendas de dispositivos móveis com sistema operacional Android atualmente e a facilidade de desenvolvimento e teste dessa plataforma, o sistema foi desenvolvido visando atingir os usuários desses dispositivos.

Palavras-Chave: Android, jogo, turismo, check-in.

Check in² Poa: an Android app for tourists in Porto Alegre

ABSTRACT

This work consists in the design and development of an Android app in the format of a game directed to tourists traveling in Porto Alegre or local residents who wish to know better the city in which they live. The goal of this application is to encourage the user to explore the city in a fun way, performing check-in at each of the places available in the game to accumulate the score for each destination. The idea for the game occurred from a free adaptation of one of the most famous board games in the world: “Monopoly”, using as money to purchase the action of perform check-in in a location. From a market analysis that demonstrated the predominance of sales of mobile devices with Android operating system nowadays and the ease of development and testing of this platform, the system was developed to reach the users of these devices.

Keywords: Android, game, tourism, check-in.

1 INTRODUÇÃO

O objetivo principal desse trabalho é projetar e desenvolver um aplicativo Android para turistas na cidade de Porto Alegre. Buscando incentivar os visitantes a explorar os principais pontos turísticos da cidade de forma divertida, o aplicativo foi desenvolvido no formato de um jogo, baseando-se livremente no jogo de tabuleiro “Monopoly”. O jogador deverá realizar um check-in como moeda de compra em cada uma das localidades disponíveis no jogo.

A cidade de Porto Alegre diferentemente de outros municípios brasileiros não possui tantas paisagens naturais capazes de atrair milhares de turistas anualmente. Todavia, está estrategicamente localizada entre os países do Mercosul, é sede de uma série de eventos culturais de porte internacional como o Fórum Social Mundial, Fórum Internacional de Software Livre e a Bienal de Artes Visuais do Mercosul e ainda está prestes a receber jogos da Copa do Mundo Fifa de futebol em 2014.

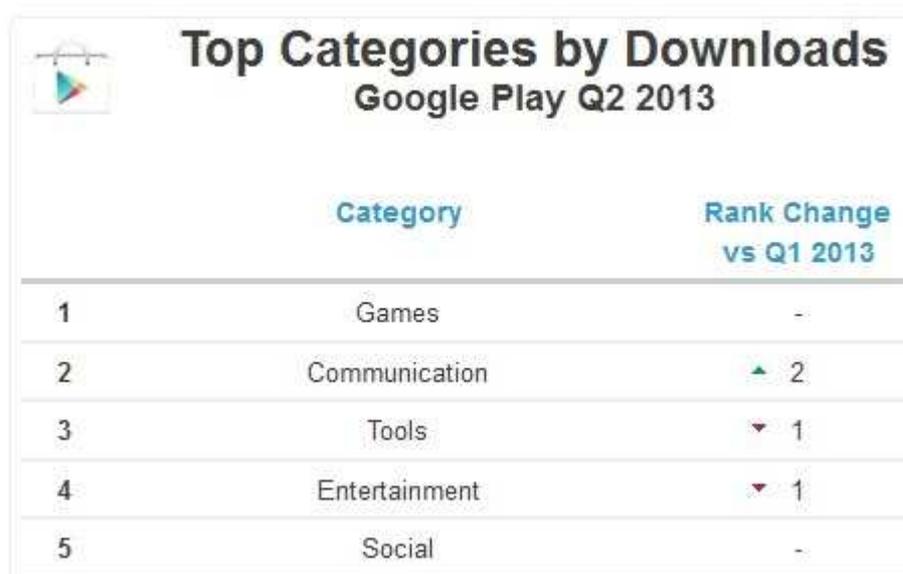
Segundo dados da Embratur (2012) entre Janeiro e Outubro do ano de 2012, Porto Alegre foi a capital com o maior crescimento no ingresso de turistas estrangeiros no país com base na entrada de visitantes estrangeiros nos aeroportos (31,7%), ficando a frente de cidades como Rio de Janeiro (19,3% no aeroporto de Galeão), Brasília (11,47%) e São Paulo (2,01% no aeroporto de Guarulhos). Esses turistas vieram principalmente participar de grandes eventos de negócios, os quais são organizados por empresas nacionais e internacionais instaladas no Estado.

Além disso, o aumento no número de ligações aéreas oferecidas entre o Aeroporto Internacional Salgado Filho e cidades no Exterior possibilita que os turistas venham diretamente para a cidade e fiquem aqui até o momento de retorno. Tendo a oportunidade de explorar nos horários livres tudo que a cidade tem a oferecer.

Visto que as vendas de smartphones não param de crescer no mundo inteiro, segundo dados do IDC (2013) as vendas desse tipo de dispositivo móvel devem alcançar a marca de um bilhão até o fim do ano, e os usuários desses aparelhos podem utilizá-los para buscar informações durante uma viagem, se torna importante o desenvolvimento de aplicativos e sites mobile visando suprir as necessidades dos turistas que chegam à cidade.

A partir de uma análise dos aplicativos para smartphones disponíveis nas duas maiores lojas desse gênero, Google Play e Apple Store, não foi encontrado nenhum aplicativo direcionado ao turismo exclusivamente na cidade de Porto Alegre com um número substancial de usuários e boas resenhas.

Desta forma, este trabalho apresenta o desenvolvimento de um aplicativo para Android visando levar os turistas estrangeiros, brasileiros de outros Estados, moradores do Interior do Rio Grande do Sul ou até mesmo moradores da própria cidade a visitar as principais atrações da cidade. Levando em consideração que a categoria Jogos teve o maior número de downloads no segundo trimestre do ano de 2013 na loja Google Play conforme o gráfico na Figura 1.1, surgiu a ideia de criar um aplicativo no formato de um jogo. Ao invés de desenvolver mais um aplicativo seguindo o formato dos demais disponíveis na categoria Turismo e Local da loja referida.



	Category	Rank Change vs Q1 2013
1	Games	-
2	Communication	▲ 2
3	Tools	▼ 1
4	Entertainment	▼ 1
5	Social	-

Figura 1.1: Categorias de Aplicativos mais baixados na loja Google Play
Fonte: (APPANNIE,2013)

O trabalho está organizado em 6 capítulos. Após a introdução e suas subseções, o segundo capítulo apresenta trabalhos relacionados a temática do turismo, os aplicativos voltados ao turismo existentes no mercado e uma análise de requisitos para o desenvolvimento da aplicação proposta nesse trabalho.

O terceiro capítulo aborda o projeto do aplicativo descrevendo a modelagem das funcionalidades, da arquitetura e do armazenamento de dados. Após, no quarto capítulo, encontra-se as informações referentes as plataformas e APIs utilizadas, sendo a plataforma Android a mais relevante entre elas.

O quinto capítulo descreve a metodologia utilizada para desenvolver o aplicativo proposto, como foram obtidos os dados mais importantes para o uso do aplicativo e também será descrito todo o funcionamento da aplicação, apresentando as telas e principais funcionalidades implementadas. O sexto e último capítulo apresenta a conclusão do trabalho.

2 ESTADO DA ARTE

Neste capítulo serão apresentados os principais trabalhos relacionados a temática do turismo e os principais aplicativos disponíveis para smartphones direcionados ao turismo na cidade de Porto Alegre. A partir desse estudo foi possível pensar em uma nova abordagem para o desenvolvimento de um aplicativo relacionado ao turismo no formato de um jogo.

2.1 Trabalhos Relacionados

O trabalho de Marcon (2013) consistiu no desenvolvimento de um aplicativo Android, chamado Tourist Helper, cujo objetivo era auxiliar turistas a encontrar lugares próximos a sua localização atual e a criação de rotas entre diversos pontos escolhidos pelo usuário. Também oferece a opção de criar listas de lugares favoritos e listas de viagens realizadas que podem ser importadas/exportadas para o sistema Touristy.

No trabalho de Maioli (2013) foi desenvolvido um Sistema Web chamado Touristy com a estrutura de uma rede social para realizar o planejamento e compartilhar informações sobre viagens. O usuário deve criar um login no próprio sistema e um perfil de usuário. Após, é possível cadastrar/editar viagens, visualizar as viagens realizadas, visualizar sua lista de amigos e acessar o perfil de outros usuários. Este sistema possui integração com o aplicativo Android Tourist Helper citado acima, cujo foco é ser utilizado durante a viagem enquanto este sistema tem por objetivo ser utilizado pelos usuários antes e depois das viagens. Sendo possível exportar os dados das viagens cadastradas para o Tourist Helper.

Por último, Lorenzi (2010) apresenta a abordagem multiagente MATRES baseada em suposições com um mecanismo de confiança aplicada em um sistema de recomendação multiagente. Esta abordagem foi validada em um cenário de recomendações de pacotes turísticos utilizando casos reais de uma agência de viagens. Um conjunto de agentes é responsável pelo processo de recomendação e cada agente é responsável por uma parte da recomendação. O conjunto dessas recomendações parciais torna-se a recomendação final que será apresentada ao usuário.

2.1.1 Análise Comparativa dos Trabalhos Apresentados

A Tabela 1.1 apresenta uma análise comparativa dos trabalhos apresentados na subseção anterior, destacando as principais características referentes a cada um dos sistemas desenvolvidos ligados a temática do turismo.

Tabela 1.1: Análise comparativa dos trabalhos relacionados

Característica/Trabalho	(MARCON, 2013)	(MAIOLI, 2013)	(LORENZI, 2010)
Plataforma	Android	Web	Aplicação Desktop
Check-in	Não	Não	Não
Assunto	Android/Sistemas Web	Serviços Web/Sistemas: Informação	IA/ Sistemas Multiagentes/ Informática: Turismo/ Sistemas: Recomendação
Arquitetura do Sistema	MVC	MVC	Multiagente
Armazenamento de Dados	Google App Engine/Nuvem	Banco de Dados/Servidor Web	Banco de Dados (Bancos de Conhecimentos)
Linguagem de Programação	Java + XML	Java	Java + XML

2.2 Soluções Existentes

Uma análise dos aplicativos relacionados a turismo disponíveis nas lojas virtuais Google Play Store e Apple App Store foi realizada. Visando compreender o que já era oferecido aos usuários e buscando desenvolver um aplicativo com uma dinâmica diferente dos demais.

2.2.1 Brasil Mobile – Guia Turístico Porto Alegre

Aplicativo desenvolvido pela Embratur para aparelhos com sistema operacional iOS, possui versões para diferentes capitais brasileiras. A partir da localização atual do usuário fornece algumas atrações próximas da referida localidade. Além disso, é possível escolher atividades entre as categorias: patrimônio e arquitetura, atividades culturais, atividades ao ar livre, lazer e compras. Não oferece informações sobre como chegar em algum lugar, apenas informa a qual distância o usuário se encontra daquele lugar e mostra no formato de pontos no mapa da cidade.

2.2.2 TS Go! Rio Grande do Sul

Este aplicativo pago para Android pertencente a família TS Go! de aplicativos para turismo, a qual abrange individualmente quase todos os estados brasileiros oferecendo o mapa da região com a posição atual do usuário e pontos interessantes próximos a sua

localização. Esta versão possui uma série de cidades, como: Porto Alegre, Caxias do Sul, Canoas, Pelotas, Viamão, Novo Hamburgo, Santa Maria entre outras.

O aplicativo destaca como o seu grande diferencial não necessitar de conexão com a Internet. Assim, os turistas não gastariam com os custos de roaming¹ por utilizar a Internet fora da sua área de cobertura.

Possui posicionamento por GPS que mostra a posição atual do usuário e a distância real até o local que ele deseja visitar. Também oferece conectividade com o Google Maps.

2.2.3 Porto Alegre Guia

É um aplicativo para Android com o objetivo de encontrar atrações, hotéis, restaurantes, casas noturnas mais populares, pontos turísticos, museus, caixas eletrônicas e farmácias próximas a localidade atual do usuário. As principais funcionalidades do aplicativo só funcionam se o usuário estiver conectado a Internet.

O aplicativo é extremamente lento e não há uma uniformidade no idioma dos ícones e dados apresentados. Além disso, os nomes dos ícones e os botões muitas vezes não fazem sentido. Por exemplo, o botão para retornar a página anterior se chama “De volta”. Além disso, existem botões para acessar redes sociais a partir do aplicativo que não funcionam entre outras funcionalidades com problemas.

2.2.4 240 Prazeres de Porto Alegre

Este aplicativo foi uma homenagem da RBS aos 240 anos da fundação da cidade de Porto Alegre, assim 240 pessoas famosas ou não na região foram ouvidas e indicaram os lugares que elas consideravam especiais na cidade. A localização no mapa da cidade é fornecida para cada lugar indicado.

2.2.5 Field Trip

Este aplicativo para Android e iOS tem como diferencial notificar o usuário conforme ele vai se deslocando pela cidade sobre lugares interessantes de acordo com as categorias de locais definidas por ele. Quando o usuário se aproxima do local o aplicativo pode ler as informações disponíveis pelo fone de ouvido ou Bluetooth caso ele esteja dirigindo. É possível compartilhar os lugares descobertos nas seguintes redes sociais: Facebook, Twitter ou Google+. Atualmente, poucas localidades possuem informações disponíveis tornando restrita a área de abrangência do aplicativo.

¹ *Roaming*: Roaming ou itinerância é um termo empregado em telefonia móvel, mas também aplicável a outras tecnologias de rede sem fio. Designa a capacidade de um usuário de uma rede para obter conectividade em áreas fora da localidade geográfica onde está registrado, ou seja, obtendo conectividade através de uma outra rede onde é visitante. A rede que está sendo visitada pode ou não pertencer a mesma operadora.

O fornecimento de *roaming* para os usuários traz inúmeras implicações técnicas e comerciais para as operadoras, principalmente quando ocorre *roaming* entre operadoras distintas. É preciso realizar a autenticação dos usuários visitantes de outras redes de outras operadoras, bem como realizar a cobrança pela prestação dos serviços ao usuário e a sua operadora. Fonte: <http://pt.wikipedia.org/wiki/Roaming>

2.2.6 Foursquare

Aplicativo para Android e iOS que faz recomendações ao usuário a partir dos check-ins realizados pelo mesmo em restaurantes, bares e lojas. Também se baseia na rede de amigos do usuário que utilizam o aplicativo. Assim, busca traçar um perfil da pessoa e indicar lugares que se enquadrem no seu estilo de vida. Utiliza o Facebook como login ou um e-mail pessoal. É possível compartilhar os check-ins no Twitter e no Facebook do usuário.

2.2.7 Análise Comparativa das Soluções Existentes

A Tabela 1.2 apresenta uma análise comparativa dos aplicativos apresentados acima, destacando as principais características que influenciaram no desenvolvimento do aplicativo proposto nesse trabalho.

Tabela 1.2: Análise comparativa das soluções existentes

Aplicativo/Característica	Sistema Operacional	Loja	Preço	Avaliação do Conteúdo	Check-in	Redes Sociais
Brasil Mobile - Guia Turístico Porto Alegre	iOS	App Store	Gratuito	-	Não	Não
TS Go! Rio Grande do Sul	Android	Google Play	R\$ 2,12	Baixo nível de maturidade	Não	Facebook
Porto Alegre Guia	Android	Google Play	Gratuito	Baixo nível de maturidade	Não	Não
240 Prazeres de Porto Alegre	Android	Google Play	Gratuito	-	Não	Não
Field Trip	Android e iOS	Google Play e App Store	Gratuito	Baixo nível de maturidade	Não	Facebook, Twitter, Google+
Foursquare	Android e iOS	Google Play e App Store	Gratuito	Médio nível de maturidade	Sim	Facebook

2.3 Identificação de Requisitos

A partir da análise das informações obtidas nas subseções anteriores foram definidos requisitos funcionais e não-funcionais para a aplicação que está sendo proposta nesse trabalho.

- Requisitos Funcionais:
 - O sistema deve prover um botão de Login do Facebook;

- O sistema deve inserir no banco de dados o login do usuário e sua respectiva pontuação;
- O sistema deve guardar os lugares visitados pelo jogador;
- O sistema deve efetuar check-in se o jogador estiver na localidade selecionada;
- O sistema deve disponibilizar as informações no idioma do dispositivo móvel do jogador;
- O sistema deve mostrar o mapa de como chegar a uma localidade a partir da posição atual do jogador quando ele clicar no botão do Mapa;
- O sistema deve mostrar as informações do local quando o jogador selecioná-lo;
- Requisitos Não-funcionais:
 - A interface do sistema deve ser implementada como um arquivo XML;
 - O sistema será implementado em Java para dispositivos móveis com sistema operacional Android;
 - O sistema deverá utilizar um banco de dados SQLite;
 - Os dados de Login do jogador não podem ser exibidos pela aplicação, devem ficar guardados no banco de dados e utilizados somente pela parte interna do sistema;
 - O sistema deve realizar check-in de forma rápida;
 - O sistema deverá utilizar o Google Maps para mostrar ao jogador um mapa de como chegar a um local;

3 PROJETO DO APLICATIVO

Neste capítulo serão abordadas as modelagens utilizadas para desenvolver o aplicativo proposto, começando pelas funcionalidades que foram definidas no formato de User Stories. Após, será apresentada uma descrição da arquitetura do sistema e por último como foi feito o armazenamento dos dados do jogo.

3.1 Modelagem das Funcionalidades

Esta seção apresentará as funcionalidades propostas para o aplicativo no formato de User Stories.

3.1.1 User Stories

Uma User Story (história de usuário, em português) pode ser definida como uma parte do sistema suficientemente pequena que os desenvolvedores possam projetar, codificar, testar e desenvolvê-la em um curto período de tempo segundo Cockburn (2005). Este conceito está diretamente relacionado aos métodos ágeis de desenvolvimento de software. Neste trabalho foi utilizado o modelo proposto por Cohn (2004) no qual as User Stories devem ser descrições de funcionalidades simples, claras e breves que serão de grande utilidade para os usuários reais. Assim, as User Stories foram modeladas no formato: “Como um <tipo de usuário>, eu quero <algum objetivo> para <alguma razão>.”

Foram definidas as seguintes User Stories para o desenvolvimento dessa aplicação:

Tabela 3.1: User Stories do Aplicativo

User Story 1
Como um <i>Usuário</i> ,
eu quero <i>utilizar minha conta do Facebook como login</i>
para <i>não precisar fazer um cadastro</i> .

User Story 2
Como um <i>Usuário</i> ,
eu quero <i>fazer login no aplicativo</i>
para <i>visualizar os lugares que eu já visitei e saber a minha pontuação</i> .

User Story 3
Como um <i>Usuário</i> ,
eu quero <i>realizar check-in em um local</i>
para <i>ganhar mais pontos no jogo</i> .

User Story 4
Como um <i>Usuário</i> ,
eu quero <i>visualizar o menu Principal</i>
para <i>poder escolher entre os diferentes modos de jogo</i> .

User Story 5
Como um <i>Usuário</i> ,
eu quero <i>jogar no modo Completo</i>
para <i>poder visitar todos os lugares</i> .

User Story 6
Como um <i>Usuário</i> ,
eu quero <i>saber qual o local mais próximo de onde estou</i>
para <i>ir visita-lo</i> .

User Story 7
Como um <i>Usuário</i> ,
eu quero <i>saber como chegar em um local</i>
para <i>não precisar procurar em outro aplicativo ou usar um mapa</i> .

User Story 8
Como um <i>Usuário</i> ,
eu quero <i>compartilhar no Facebook que estou fazendo check-in em um local</i>
para <i>poder compartilhar com os meus amigos</i> .

User Story 9
Como um <i>Usuário</i> ,
eu quero <i>visualizar as informações sobre um local</i>
para <i>decidir se irei visita-lo</i> .

User Story 10
Como um <i>Usuário</i> ,
eu quero <i>que o aplicativo esteja no idioma do meu aparelho</i>
para <i>que eu possa compreender o que está escrito</i> .

3.2 Modelagem da Arquitetura

A seguir será detalhado o padrão arquitetural MVC que foi utilizado como base para a implementação do sistema proposto. Visto que, esse padrão busca separar a

interface gráfica, a lógica da aplicação e a modelagem dos dados e um aplicativo Android pode ser facilmente organizado dessa forma, decidiu-se por utilizá-lo.

3.2.1 Padrão MVC

Segundo Burbeck (1992), MVC (Model-View-Controller) é um padrão arquitetural que separa a modelagem dos dados da aplicação, a apresentação desses dados e as ações do sistema em três classes separadas conforme a Figura 3.1, baseando-se na interação entre o usuário e o sistema. Assim, é possível separar a lógica da interface e a lógica de negócio, tornar o código reutilizável e modificá-lo de forma fácil e segura, diminuindo a possibilidade de introduzir efeitos colaterais.

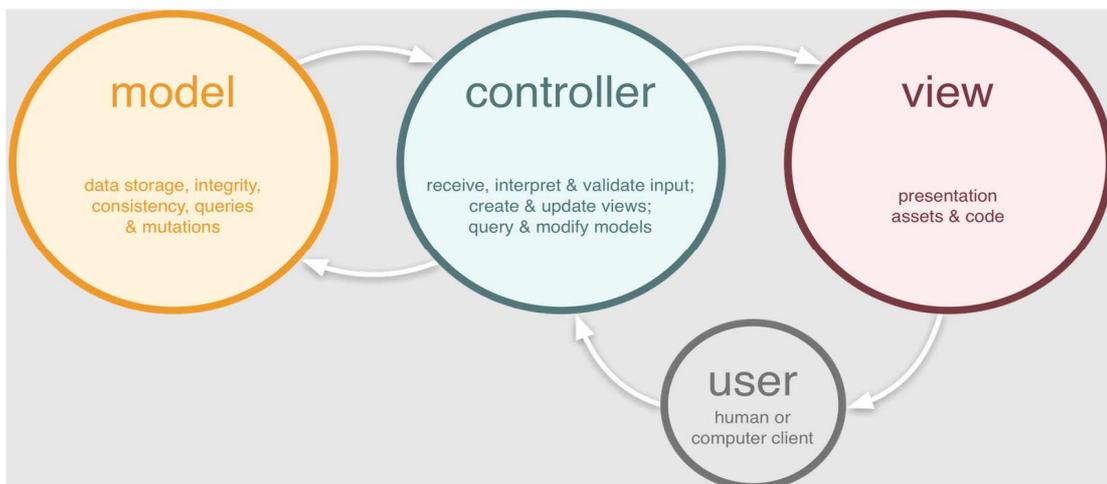


Figura 3.1: Interação entre as camadas do padrão MVC

Fonte: (BRINKMAN-DAVIS,2012)

A camada de Modelo gerencia o comportamento e os dados do domínio da aplicação, além de acessar o Banco de Dados e fornecer as operações necessárias para inserir, alterar ou apagar dados.

A camada de Visualização define e controla como os dados do Modelo serão apresentados ao usuário.

E por último, a camada de Controle interpreta as entradas do usuário pelo mouse e teclado, informando ao Modelo e a Visualização quais ações são necessárias.

Uma aplicação Android pode ser modelada seguindo o padrão MVC organizando o projeto da seguinte maneira: as Atividades constituem o Controle, as classes Java fazem parte do Modelo e a Visualização é representada pelos arquivos do tipo XML.

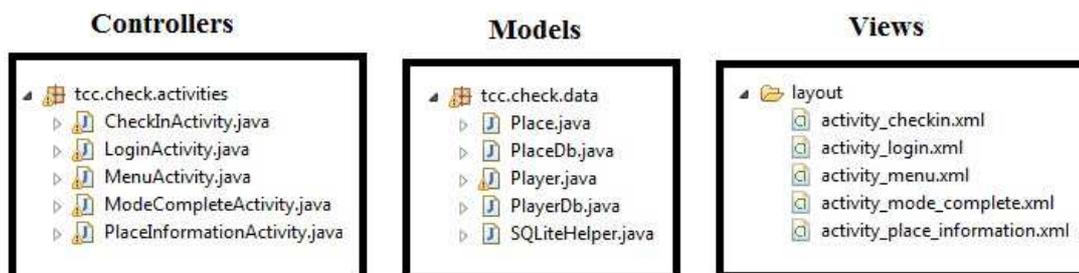


Figura 3.2: Utilizando o padrão MVC no projeto Android

3.3 Modelagem do Armazenamento de Dados

O armazenamento dos dados necessários para a aplicação foi feito criando-se tabelas em um banco de dados. Nesta seção será feito um detalhamento do Banco de Dados SQLite disponível na plataforma Android e das tabelas criadas para o armazenamento dos dados da aplicação.

3.3.1 Banco de Dados SQLite

O SQLite é uma biblioteca desenvolvida em C que oferece um banco de dados sem a necessidade de utilizar um SGBD (Sistema de Gerenciamento de Banco de Dados) para gerenciá-lo. A própria biblioteca lê e escreve os dados diretamente no arquivo do banco de dados em disco. É uma biblioteca de código aberto, multiplataforma, sem dependências externas e que suporta bases de dados com até dois terabytes. Tornando-se ideal para dispositivos móveis que necessitam de simplicidade para administrar, implementar e dar manutenção a uma base de dados por possuírem recursos limitados.

No desenvolvimento da aplicação proposta nesse trabalho foi criado um banco de dados utilizando a biblioteca SQLite. Foram introduzidas duas tabelas que serão descritas nas Tabelas 3.2 e 3.3.

Os dados necessários para preencher as informações correspondentes a cada localização do jogo encontram-se na tabela chamada “places”.

Tabela 3.2: Descrição da tabela “places”

Nome do campo	Tipo	Descrição
id	INTEGER	Identificador do local
numstars	TEXT	Pontuação do Local
name	TEXT	Nome
category	TEXT	Categoria
address	TEXT	Endereço
phone	TEXT	Telefone
website	TEXT	Site
description	TEXT	Descrição
latitude	TEXT	Coordenada geográfica da latitude
longitude	TEXT	Coordenada geográfica da longitude

Os dados relacionados as informações de um jogador do jogo foram inseridos na tabela chamada “player”.

Tabela 3.3: Descrição da tabela “player”

Nome do campo	Tipo	Descrição
id	INTEGER	Identificador do jogador
email	TEXT	Email de login do jogador
score	TEXT	Pontuação do jogador
places	TEXT	Locais que o jogador já visitou

4 PLATAFORMAS

Neste capítulo serão abordadas as plataformas utilizadas no desenvolvimento do aplicativo proposto neste trabalho. Conforme o gráfico apresentado na Figura 4.1 extraído de Gartner (2013), desde o terceiro trimestre do ano de 2010 os *smartphones* com sistema operacional Android são os mais vendidos no mercado mundial para usuários finais. Além disso, por se tratar de uma plataforma com código aberto e ser possível utilizar qualquer dispositivo físico com este sistema operacional para testar uma aplicação, escolheu-se o Android ao invés do iOS como base para o desenvolvimento da aplicação proposta neste trabalho.

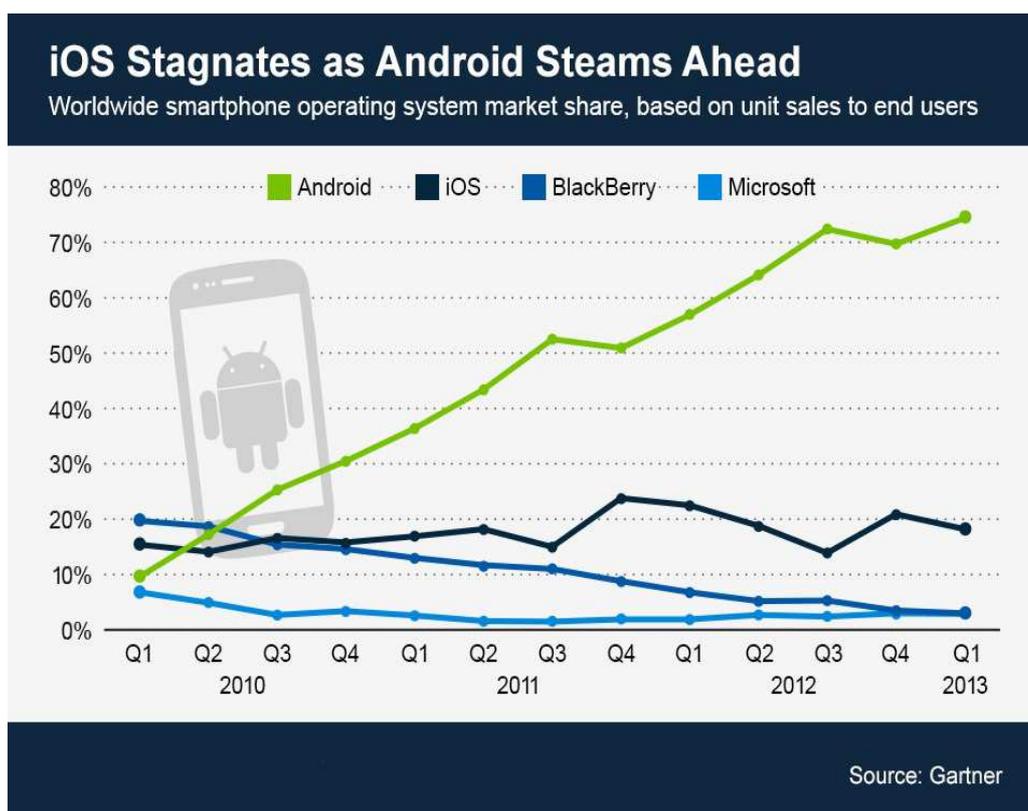


Figura 4.1: Mercado mundial de *smartphones* por Sistema Operacional

Fonte: (GARTNER,2013)

Primeiramente, serão discutidas as plataformas de desenvolvimento disponíveis para criar aplicativos Android. Após, a plataforma Android será explicada em maiores detalhes e por último serão referenciadas as outras API usadas.

4.1 Plataformas de Desenvolvimento

Nesta seção serão apresentadas as informações obtidas a partir de uma pesquisa sobre as principais IDE (Integrated Development Environment) utilizadas no desenvolvimento de aplicativos Android. Esta pesquisa serviu como base para a escolha da plataforma de desenvolvimento Eclipse em detrimento do Android Studio que ainda encontra-se numa versão prévia.

4.1.1 Eclipse

Atualmente há duas maneiras de utilizar esta IDE para desenvolver aplicativos Android: baixando todas as ferramentas (Eclipse, plugin ADT e SDK Android) separadamente e depois configurando-as conjuntamente, ou baixando apenas o ADT Bundle que encapsula todas estas ferramentas. Em ambas as situações é necessário baixar e instalar o JDK (Java Development Kit), o qual fornece as bibliotecas de desenvolvimento Java.

No primeiro caso é preciso instalar uma versão do Eclipse e o Android SDK, instalar o plugin ADT pelas configurações do Eclipse e ainda alterar nas configurações deste plugin o caminho para a pasta onde foi instalado o Android SDK.

Já o processo de instalação no segundo caso é bem simples, bastando apenas descompactar a pasta do ADT Bundle no local desejado para instalação. A pasta do ADT Bundle é composta por:

- Pasta eclipse - contém os arquivos referentes ao software eclipse já configurado com o plugin ADT;
- Pasta sdk - contém os arquivos do SDK Android e ferramentas adicionais;
- Arquivo SDK Manager.exe - atualiza as versões do SDK do Android.

Além disso, inclui a última versão da plataforma Android e a versão mais recente do emulador. Enquanto no primeiro caso é fortemente recomendado que o usuário realize atualizações após instalar as ferramentas.

Para testar uma aplicação Android é necessário criar um AVD (Android Virtual Device) que é uma configuração de emulação que simula as características de um determinado celular real.

A partir dessas informações foi escolhido utilizar o ADT Bundle, pois era mais fácil de instalar e configurar, além de oferecer os mesmos recursos que a outra opção.

4.1.2 Android Studio

Esta IDE encontra-se em sua versão prévia (Early Access Preview) e foi anunciada em Maio de 2013 na conferência anual Google I/O, sendo disponibilizada gratuitamente a partir de Junho de 2013 para download. É baseada na versão comunitária do IntelliJ IDEA, porém foi desenvolvida especificamente para a criação de aplicativos Android.

O Android Studio usa o *Gradle*¹ para compilar as aplicações. Os projetos podem ser armazenados no Git, Mercurial ou Subversion. E o gerenciador de dispositivo virtual do Android (Android Virtual Device Manager) é completamente suportado. Possui versão para Windows, Mac OS X e Linux.

Oferece como principais diferenciais:

- Editor WYSIWYG: fornece um “live layout” em que as edições efetuadas no código são renderizadas simultaneamente no layout do aplicativo. Quando o valor de uma string, ou a cor ou um ícone é alterado, esta mudança ocorre no editor simultaneamente;
- Construtor de layout: um editor gráfico com Arrastar e Soltar (Drag and Drop) que oferece a possibilidade de visualizar layouts em várias configurações de tela concomitantemente (diferentes modelos de celulares e tablets);
- Console do Desenvolvedor: oferece dicas de otimização de código, assistência para tradução, rastreamento de onde foram efetuados os downloads do aplicativo, criar campanhas publicitárias e promoções;
- Ferramentas para avaliar desempenho, usabilidade, compatibilidade de versão entre outros;
- Projetos criados no Eclipse podem ser migrados para o Android Studio.

Estas funcionalidades visam aumentar a produtividade e tornar mais fácil para os desenvolvedores a criação de aplicativos com boa aparência necessitando de um pouco menos de esforço.

Entretanto, há um aviso na página de download da IDE informando que a versão atual possui vários recursos incompletos ou não implementados ainda e que bugs podem ser encontrados. Desta forma, se o usuário não se sentir confortável utilizando um produto inacabado é sugerido que ele faça o download do ADT Bundle.

4.2 Plataforma Android

Esta seção apresentará um breve histórico da Plataforma Android, uma lista com as versões da plataforma lançadas até o momento, além de uma descrição da arquitetura e dos principais conceitos envolvidos no desenvolvimento de aplicativos Android.

4.2.1 Histórico

Em Outubro de 2003, foi fundada uma empresa chamada Android, Inc na Califórnia (EUA), cujo foco era o desenvolvimento de sistemas operacionais para celulares, mas todos os seus projetos eram secretos. Em Agosto de 2005, a Google anuncia a compra dessa empresa criando rumores de que a empresa estaria entrando no mercado de telefonia móvel.

Somente em Novembro de 2007 foi revelado o consórcio de empresas pertencentes ao chamado OHA (Open Handset Alliance), como por exemplo: Google, HTC, Intel, LG, Dell, Broadcom Corporation, Marvell Technology Group, Motorola, Nvidia, Qualcomm, Samsung Electronics, Sprint Nextel, T-Mobile and Texas Instruments.

¹ *Gradle*: é uma ferramenta de automação de projeto que baseia-se nos conceitos de Apache Ant e Apache Maven. Fonte: <http://en.wikipedia.org/wiki/Gradle>

O principal objetivo desse grupo era padronizar uma plataforma de código aberto e livre para dispositivos móveis. Os fabricantes poderiam utilizá-la sem custos, realizar alterações no código-fonte conforme suas necessidades e não seriam obrigados a compartilhar essas alterações. Assim, surgiu o sistema operacional para dispositivos móveis Android. Atualmente, 84 empresas fazem parte dessa aliança.

O primeiro celular com Android disponível no mercado foi o T-Mobile G1 (também conhecido como HTC Dream) lançado em Outubro de 2008. Nesse mesmo mês, a Google disponibilizou todo o código fonte do Android sob a licença Apache (permite o uso e a distribuição do código-fonte).

Desde o seu lançamento surgiram diversas novas versões da plataforma Android, as quais se encontram na Tabela 4.1.

Tabela 4.1: Versões da Plataforma Android

Versão	Codiname	API	Distribuição
1.5	Cupcake	3	0,1% <
1.6	Donut	4	0,1% <
2.0	Eclair	5	0,1% <
2.2	Froyo	8	1,7%
2.3.3 - 2.3.7	Gingerbread	10	26,3%
3.0	Honeycomb	13	0,1%
4.0.3 – 4.0.4	Ice Cream Sandwich	15	19,8%
4.1.x – 4.3	Jelly Bean	16	52,1%
4.4	Kit Kat	19	-

Fonte: Elaborada pela autora com base em Android Developers (2013) e Wikipedia (2013)

4.2.2 Arquitetura

Conforme consta em Lecheta (2010), a arquitetura da plataforma Android é dividida em camadas, as quais são responsáveis por gerenciar seus respectivos processos. Cada aplicação é executada em um processo individual e possui uma thread dedicada. As camadas encontram-se divididas conforme a Figura 4.2.



Figura 4.2: Arquitetura da Plataforma Android
Fonte: (GALLERYANDROID,2013)

4.2.1.1 Sistema Operacional Linux

A camada mais inferior da arquitetura Android é composta por um kernel baseado na versão 2.6 do Linux. Essa camada não interage com usuários ou desenvolvedores, sendo responsável pelo gerenciamento de memória, processos e threads, pela segurança dos arquivos e pastas, pelas operações de redes e pelos drivers.

4.2.1.2 Bibliotecas Nativas

Logo acima está a camada que contém as bibliotecas nativas do Android, as quais são escritas em C e C++. Elas são pré-instaladas nos dispositivos móveis de acordo com a sua arquitetura, pois assim podem ser compiladas corretamente. Existem bibliotecas para navegação web, aplicações multimídia, suporte a banco de dados entre outros recursos.

4.2.1.3 Android Runtime

No mesmo nível da camada de bibliotecas nativas está a camada de execução Android que abrange bibliotecas do núcleo Java e a Máquina Virtual Dalvik. Estas bibliotecas são necessárias, uma vez que os aplicativos para dispositivos Android são programados utilizando a linguagem Java.

A Máquina Virtual Dalvik é responsável por executar códigos escritos em Java para Android. Primeiramente, arquivos do tipo .class e .jar são convertidos em tempo de compilação para o formato .dex. Assim, podem ser executados pela Dalvik com maior eficiência. Esta Máquina Virtual (MV) foi desenvolvida de forma otimizada para

consumir pouca memória, permitir que várias instâncias de MV rodem simultaneamente e delega ao sistema operacional o isolamento de processos, o gerenciamento de memória e de threads em cada instância.

4.2.1.4 Framework da Aplicação

A penúltima camada é responsável por fornecer os programas básicos que gerenciam o acesso dos aplicativos aos recursos de hardware disponíveis, por exemplo: sensores, WiFi e localização. As partes mais importantes dessa camada são:

- Activity Manager: controla o ciclo de vida das aplicações;
- Content Providers: gerencia o compartilhamento de dados entre aplicativos;
- Telephony Manager: gerencia todas as chamadas de voz;
- Location Manager: responsável pela localização geográfica do dispositivo via GPS ou torre de celular;
- Resource Manager: controla os recursos utilizados por um aplicativo;
- Notification Manager: notifica ao usuário sobre eventos, como novas mensagens, alertas e alarmes.

4.2.1.5 Aplicação

A última camada é composta pelos aplicativos que vêm pré-instalados nos dispositivos móveis, como agenda de contatos, relógio e calendário, além dos aplicativos instalados pelo usuário. Esta camada que interage com os usuários.

4.2.3 Conceitos Básicos

Os principais elementos de uma aplicação Android são: as Atividades e os Serviços. Uma Atividade consiste em uma tela composta por componentes gráficos suportados pela plataforma, a qual interage com os usuários. Uma aplicação pode conter uma ou mais Atividades. Um Serviço compreende trechos de código que representam uma tarefa executada em segundo plano, sem interface com o usuário e com um ciclo de vida próprio.

Conforme descrito em Gargenta (2011), uma aplicação Android consiste em um arquivo do tipo APK que possui três componentes principais:

- Executável Dalvik: todo o código fonte da aplicação em Java foi compilado para um arquivo executável Dalvik, o qual rodará a aplicação;
- Recursos: todos os componentes da aplicação que não são código. Podem ser: imagens, arquivos de áudio e vídeo, arquivos XML de layouts e pacotes de idiomas, entre outros;
- Bibliotecas Nativas: uma aplicação pode incluir código nativo juntamente com o seu arquivo APK, como bibliotecas em C ou C++.

4.2.4 Ciclo de Vida de uma Atividade

Cada Atividade possui um ciclo de vida dividido em diversos estados, desde a sua criação até o seu encerramento. A cada mudança de estado, um método específico é

chamado e uma notificação é enviada ao aplicativo sendo tudo controlado pelo sistema operacional do dispositivo que está rodando a aplicação.

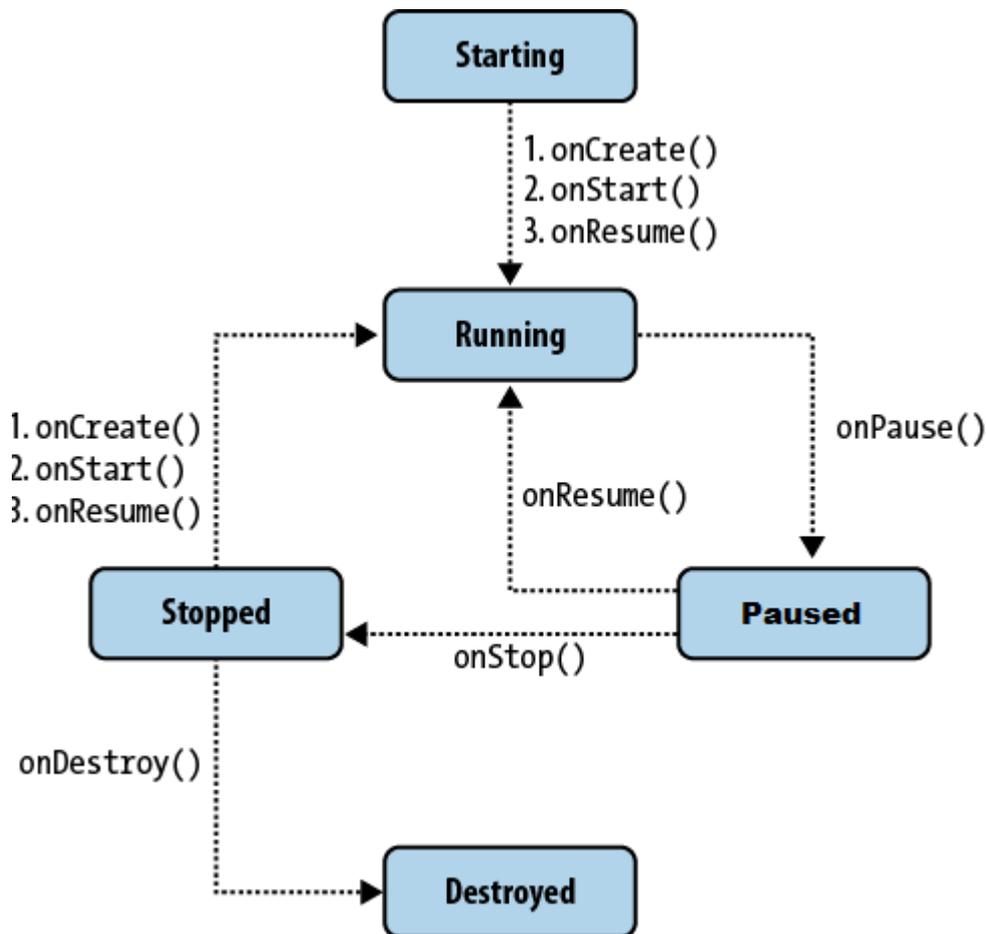


Figura 4.3: Ciclo de Vida de uma Atividade

Fonte: (GARGENTA, 2011)

Os principais métodos chamados ao longo do ciclo de vida de uma Atividade são:

- `onCreate()`: é chamado apenas uma única vez quando a Atividade é iniciada. Quando este método é finalizado, o método `onStart()` é chamado para iniciar o ciclo de vida visível da Atividade, por isso pode ser usado para fazer a inicialização da interface do usuário;
- `onStart()`: é chamado quando a Atividade está prestes a ficar visível ao usuário, pode ser chamado depois dos métodos `onCreate()` ou `onRestart()` dependendo do estado da aplicação;
- `onRestart()`: usado quando uma Atividade está parada temporariamente e deve ser iniciada novamente. Esse método chama o método `onStart()` de forma automática.
- `onResume()`: é chamado quando a Atividade está executando como a Atividade Principal do aplicativo e está pronta para interagir com o usuário;

- `onPause()`: método chamado quando o sistema está prestes a retomar ou iniciar outra Atividade. Ele salva o estado da aplicação, para que a Atividade interrompida possa voltar a ser executada do ponto em que parou;
- `onStop()`: chamado quando a Atividade está sendo encerrada e não encontra-se mais visível ao usuário;
- `onDestroy()`: método que encerra a execução de uma Atividade, pode ser chamado automaticamente pelo sistema operacional para liberar recursos ou pela aplicação pelo método `finish()` da classe `Activity`.

4.3 API Utilizadas

Nesta subseção serão tratadas outras API utilizadas no desenvolvimento do aplicativo com o intuito de oferecer aos usuários certas comodidades, como fazer Login no aplicativo utilizando sua conta do Facebook.

4.3.1 Facebook

Algumas funcionalidades oferecidas pelo SDK do Facebook foram integradas ao aplicativo, como realizar Login e fazer uma publicação na linha do tempo do usuário. Foi necessário criar um aplicativo Facebook, vinculando-o a uma conta de desenvolvedor no site do Facebook. Assim, uma chave para acessar a API do Facebook foi obtida e inserida ao código do aplicativo seguindo as instruções disponíveis em Facebook (2013).

5 IMPLEMENTAÇÃO DO APLICATIVO

Este capítulo tem por objetivo descrever como foi realizada a implementação do aplicativo, descrevendo a metodologia de desenvolvimento utilizada, como foram obtidos os dados necessários para o uso da aplicação e o funcionamento geral do sistema.

5.1 Metodologia de Desenvolvimento

Baseando-se nas User Stories definidas na seção 3.1.1, a aplicação foi desenvolvida de modo incremental, buscando a cada ciclo de desenvolvimento agregar novas funcionalidades a aplicação. Ao final de cada ciclo era entregue uma nova versão funcional do aplicativo. A tabela 5.1 descreve as funcionalidades desenvolvidas em cada ciclo de desenvolvimento.

Tabela 5.1: Ciclo de Desenvolvimento

Ciclo	Descrição
1	Implementação das User Stories 1, 2 e 4 – Tela de Login e Tela do Menu Principal
2	Implementação das User Stories 5 e 9 – Tela de Modo Completo e Tela de Informações do Local
3	Implementação da User Story 3 – Tela de Check-in
4	Implementação das User Stories 7 e 8 - Mapa de como chegar a um local e Compartilhamento no Facebook
5	Implementação das User Stories 6 e 10 – Funcionalidade Ir para o lugar mais próximo e Seleção de Idiomas

5.2 Dados necessários a Aplicação

Algumas funcionalidades do aplicativo necessitam de dados que são obtidos externamente, como as informações do jogador solicitadas ao Facebook para identificá-lo e realizar Login e coordenadas geográficas obtidas através do GPS do dispositivo móvel utilizado pelo jogador para executar a aplicação. Nessa seção, será descrito como esses dados são adquiridos.

5.2.1 Login e Compartilhamento usando o Facebook

Para utilizar a API do Facebook é necessário cadastrar um aplicativo Android na página do Facebook Developers, adquirir uma chave única de identificação para o aplicativo e inseri-la no código fonte. Assim, quando o aplicativo fizer uma requisição a API do Facebook, ela será capaz de reconhecê-lo e irá fornecer os serviços requisitados.

Na aplicação proposta foram utilizados os serviços de Login e Compartilhamento na linha do tempo do usuário, seguindo a documentação disponível no portal do Facebook Developers.

Para Compartilhar uma frase na linha do tempo do usuário, por exemplo, é verificado se o usuário aceitou as permissões necessárias, caso contrário é solicitado ao usuário permiti-las. Após, a mensagem é enviada através de uma requisição HTTP e caso a mensagem tenha sido publicada com sucesso recebe-se uma mensagem no formato JSON com o número identificador da publicação.

5.2.2 Coordenadas Geográficas

A principal funcionalidade do aplicativo é a realização de check-in em um determinado lugar, o que só é possível sabendo-se a exata localização geográfica do jogador. Assim, é fundamental para a aplicação obter as informações de latitude e longitude da posição atual do jogador. Essa informação é adquirida através do uso do GPS do dispositivo móvel do jogador.

As coordenadas geográficas também são necessárias para solicitar o Mapa de como chegar a um local a partir da posição atual do jogador e para o botão de Ir para o local mais próximo.

Para utilizar o GPS do dispositivo móvel é necessário solicitar permissão ao usuário. Assim, quando ele for instalar o aplicativo deverá permitir o uso do GPS pela aplicação.

A classe nativa do Android chamada LocationManager é a responsável por fornecer acesso aos serviços de localização. Através dela é possível ativar o GPS e determinar a frequência de leitura dos dados através de intervalos de tempo ou distância. A cada leitura é fornecido um objeto da classe Location com informações como: latitude, longitude, data e hora, rumo (medida náutica em graus), altitude e velocidade.

Por conseguinte, quando alguma funcionalidade do jogo necessitou das informações de latitude e longitude da posição atual do jogador, foram utilizadas as classes descritas acima para obtê-las.

5.2.3 Locais a serem visitados

Os locais a serem visitados no Modo Completo de jogo foram selecionados a partir de uma análise dos principais pontos turísticos disponíveis no Portal do Turista. Este portal foi desenvolvido pela Prefeitura de Porto Alegre e encontra-se disponível em seu site. Foram levados em consideração os principais eventos culturais que acontecem na cidade e os seus respectivos públicos-alvo.

5.3 Funcionamento do Aplicativo

Para exemplificar o funcionamento do jogo essa seção apresentará as principais telas e funcionalidades disponíveis no aplicativo.

5.3.1 Tela de Login

A primeira tela do aplicativo tem por objetivo efetuar o login e o logout do jogador através da sua conta do Facebook. Quando o jogador clicar no botão de login uma requisição será realizada para a API do Facebook e uma janela para inserir suas credenciais aparecerá.



Figura 5.1: Tela de Login e Logout

5.3.2 Tela de Menu Principal

Após o jogador efetuar o login, a tela de Menu Principal aparecerá disponibilizando as opções de jogo. No momento, só está implementado o Modo Completo de jogo com todos os locais disponíveis. O próximo passo seria implementar o Modo Personalizado no qual o jogador poderia selecionar uma ou mais categorias e criar listas personalizadas com os lugares das categorias escolhidas.



Figura 5.2: Tela de Menu Principal

5.3.3 Tela de Modo Completo

Nessa tela são apresentados os locais que o jogador deverá visitar e realizar check-in no Modo Completo de jogo. Também apresenta a pontuação atual do jogador e um botão que indica ao jogador qual o local mais próximo da sua localização atual, exibindo a tela com as informações do respectivo lugar.



Figura 5.3: Tela de Modo Completo

5.3.4 Tela de Informações do Local

Quando o jogador seleciona um dos locais, essa tela se abre apresentando ao jogador informações sobre o local, como: nome, categoria, endereço, telefone, website, uma breve descrição e a pontuação referente ao check-in nesse local. Além disso, existem 3 botões disponíveis nessa tela.

O botão Share serve para compartilhar na linha do tempo do Facebook do jogador uma mensagem dizendo que ele está visitando aquele local. Ao lado, o botão do Google Maps serve para abrir um mapa com o caminho entre a localização atual do jogador e a localização do local que ele selecionou. E por último há o botão de realizar check-in no local.



Figura 5.4: Tela de Informações do Local

5.3.5 Tela de Check-in

Essa tela apresenta o resultado do check-in efetuado pelo jogador. Se o jogador estiver realmente no local selecionado receberá uma mensagem informando e ganhará os pontos referentes aquele local. Caso contrário, receberá uma mensagem avisando que ele não se encontra naquele local. Além disso, a tela possui um botão que retorna a tela de Modo Completo referente a subseção 5.3.3.

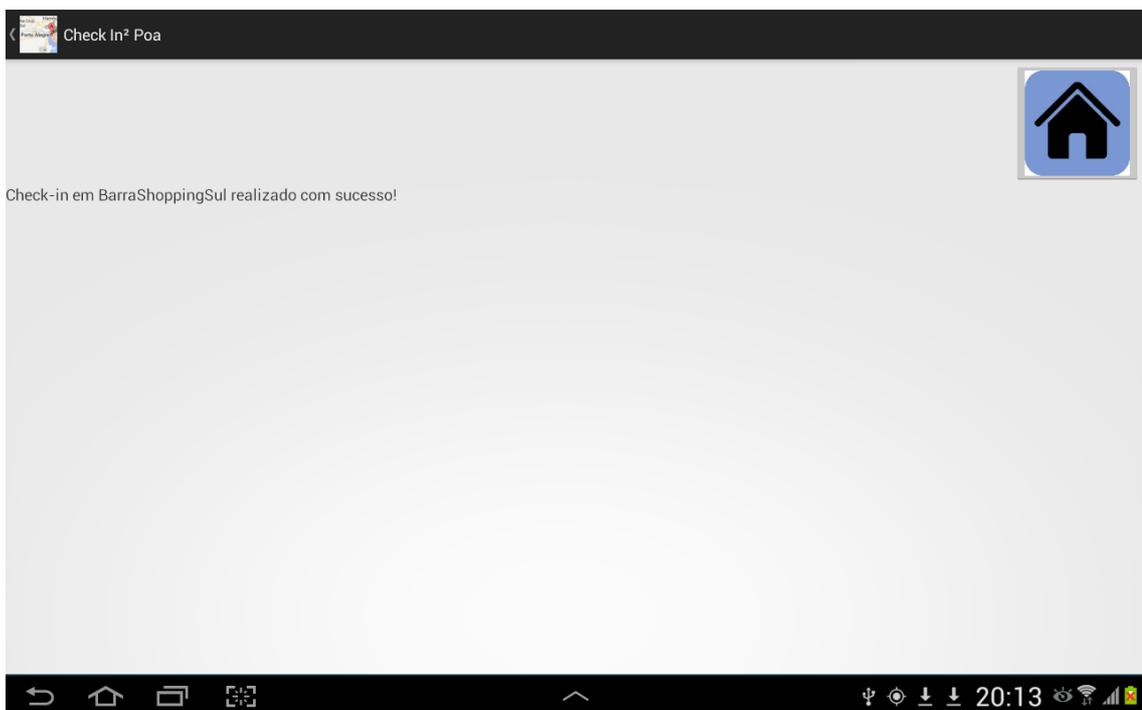




Figura 5.5: Tela de Check-in

5.3.6 Compartilhamento no Facebook

Quando o jogador estiver na tela de Informações do Local referente a subseção 5.3.4, ele poderá selecionar o botão Share e compartilhar na linha do tempo da sua conta do Facebook uma mensagem pré-definida no aplicativo, dizendo que está visitando aquele local.



Figura 5.6: Compartilhamento no Facebook

5.3.7 Mapa de como chegar a um local

Quando o jogador estiver na tela de Informações do Local referente a subseção 5.3.4, ele poderá selecionar o botão de Mapa que utilizará a Internet ou o aplicativo do Google Maps caso o jogador o tenha instalado, para mostrar o mapa com o caminho entre a posição atual do jogador e aquele local.

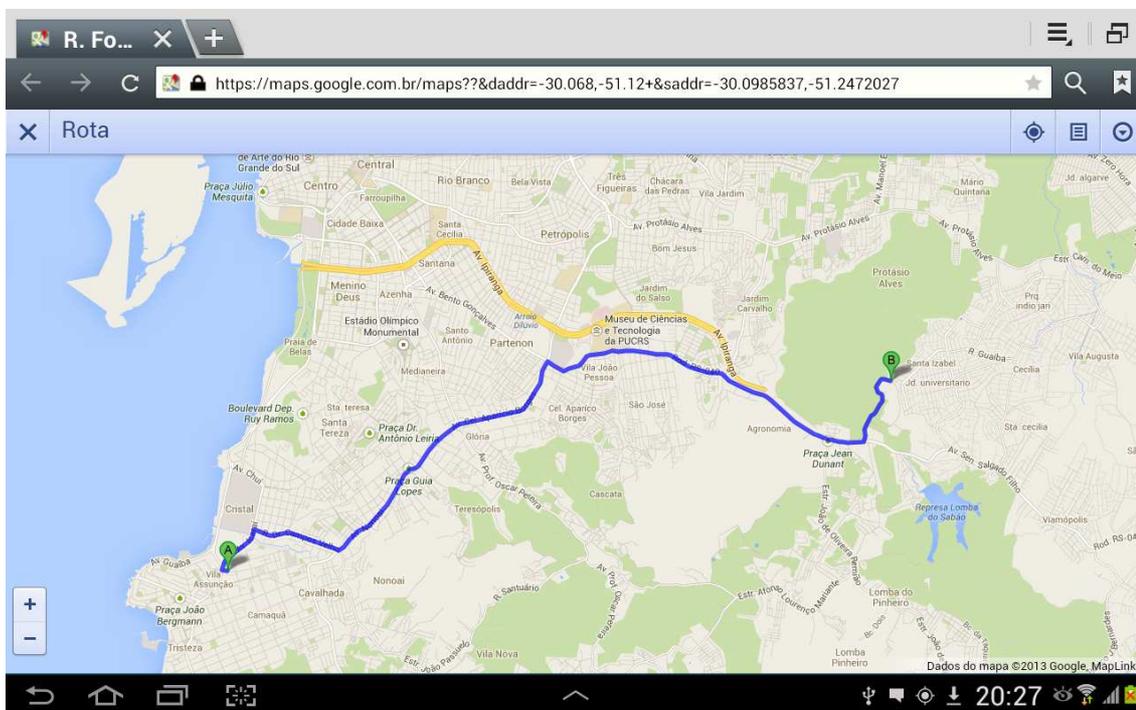


Figura 5.7: Mapa de como chegar a um local

5.3.8 Seleção de Idiomas

Buscando oferecer o aplicativo a usuários de diferentes nacionalidades que venham visitar a cidade de Porto Alegre, torna-se fundamental disponibilizá-lo em mais de um idioma. Para implementar a seleção de idiomas em um aplicativo Android, as Strings da aplicação podem ser definidas com um identificador único, o qual deve ser sempre referenciado no código ao invés de utilizar Strings fixas.

Para adicionar um idioma na aplicação deve ser criada uma pasta no diretório /res chamada “values-(abreviação do idioma conforme o padrão Android)” e um arquivo XML chamado “strings.xml” com as Strings traduzidas para aquele idioma dentro da pasta. Assim, quando o usuário abrir a aplicação, será selecionado o idioma correspondente ao do seu dispositivo móvel, ou caso não exista Strings para o seu idioma será utilizado o idioma padrão contido na pasta “values”.

Nesse aplicativo foram definidos dois idiomas: português e inglês conforme demonstrado na Figura 5.8.



Figura 5.8: Idiomas disponíveis

5.4 Avaliação e Teste do Aplicativo

Foram realizados testes de unidade a cada nova funcionalidade codificada dentro de um ciclo de desenvolvimento e testes de integração ao final de cada ciclo. Assim, a cada ciclo de desenvolvimento novas funcionalidades eram adicionadas com a garantia de que não haviam sido introduzidos erros nas funcionalidades implementadas anteriormente. Ao final do quinto ciclo foi feito um teste de sistema, realizando-se check-in em diferentes pontos turísticos. Visando conferir a pontuação do jogo, a precisão do check-in em diferentes locais, a corretude dos mapas oferecidos aos jogadores, o login e compartilhamento através do Facebook e o funcionamento geral da aplicação.

Todas as dez User Stories propostas foram implementadas e testadas. Além disso, o aplicativo atingiu os requisitos funcionais e não-funcionais definidos no início do projeto.

Fazendo uma análise comparativa entre as soluções existentes e o aplicativo desenvolvido é possível verificar que somente o Foursquare também oferece o recurso de realizar check-in em uma localidade e o compartilhamento no Facebook. Porém, o Check in² Poa é o único aplicativo no formato de um jogo.

6 CONCLUSÃO

Este trabalho apresentou um novo tipo de aplicativo voltado ao gênero do Turismo, diferenciando-se dos aplicativos existentes no mercado por ser desenvolvido no formato de um jogo. A partir da análise das vendas de aplicativos Android constatou-se que o gênero mais vendido eram os Jogos. Além disso, não existiam muitos aplicativos voltados ao turismo na cidade de Porto Alegre e os poucos que existiam apresentavam falhas. Assim, surgiu a ideia de desenvolver um protótipo de um jogo para turistas em Porto Alegre direcionado a dispositivos móveis Android, os quais são os mais vendidos no mercado mundial atualmente conforme abordado ao longo do trabalho.

O principal objetivo do aplicativo é levar os turistas a conhecer os pontos turísticos mais importantes da cidade. Para isso, são oferecidas funcionalidades como: realizar check-in nos locais para ganhar pontos no jogo, mapa de como chegar até um local, o ponto turístico mais próximo da posição atual do jogador e a possibilidade de compartilhar os check-ins em uma rede social. Essas eram as principais funcionalidades propostas para o trabalho e que foram implementadas com sucesso.

6.1 Trabalhos Futuros

Durante o desenvolvimento do aplicativo surgiram ideias de novas funcionalidades que poderiam ser agregadas ao projeto, caso os dados dos usuários e dos locais fossem armazenados na nuvem. Seria possível criar um ranking a partir da pontuação dos jogadores, os jogadores poderiam sugerir novos locais para serem adicionados ao jogo, criar novos modos de jogo baseando-se no meio de transporte utilizado pelos jogadores (a pé, de carro ou de ônibus) e ainda a possibilidade de firmar parcerias com agências de turismo adicionando roteiros personalizados aos seus clientes.

Além disso, uma avaliação de usabilidade do aplicativo poderia ser realizada, buscando aprimorar a interface e a dinâmica do jogo. Através de um roteiro com um passo-a-passo de como utilizar o jogo seguido por um questionário, a ser respondido por turistas que se encontrassem passeando em algum ponto turístico disponível no jogo.

REFERÊNCIAS

- COCKBURN, Alistair. **Escrevendo Casos de Uso Eficazes**. Porto Alegre: Bookman, 2005.
- COHN, Mike. **User Stories Applied: For Agile Software Development**. 2004. 304f.
- COHN, Mike. **Advantages of the “As a user, I want” user story template**. Disponível em: < <http://www.mountangoatsoftware.com/blog/advantages-of-the-as-a-user-i-want-user-story-template> >. Acesso em: Agosto de 2013.
- LECHETA, Ricardo R. **Google Android: Aprenda a criar aplicações para dispositivos móveis com o Android SDK**. São Paulo: Novatec Editora, 2010.
- GARGENTA, Marko. **Learning Android**. Sebastopol: O’Reilly, 2011.
- MAIOLI, Marília. **Touristy: Um Sistema Web Colaborativo para Planejamento e Compartilhamento de Viagens**. 2013, 57f. Trabalho Individual (Graduação em Ciência da Computação) – Instituto de Informática, UFRGS, Porto Alegre.
- MARCON, Diego. **Touristy: Um Sistema Android para Gerenciamento de Roteiros Turísticos**. 2013, 55f. Trabalho Individual (Graduação em Ciência da Computação) – Instituto de Informática, UFRGS, Porto Alegre.
- LORENZI, Fabiana. **Uma aborgagem multiagente de recomendação baseada em suposições e confiança para cenários dinâmicos**. 2010, 84f. Trabalho Individual (Doutorado em Ciência da Computação) – Instituto de Informática, UFRGS, Porto Alegre.
- GOOGLE. **Android Developers**. Disponível em:
<<http://developer.android.com/index.html> >. Acesso em: Outubro de 2013.
- FACEBOOK. **Facebook Developers**. Disponível em:
< <https://developers.facebook.com/> >. Acesso em: Outubro de 2013.
- MICROSOFT. **Model-View-Controller**. Disponível em:
<<http://msdn.microsoft.com/en-us/library/ff649643.aspx>>. Acesso em: Outubro de 2013.
- GARTNER. **iOS Stagnates as Android Steams Ahead**. Disponível em:
< <http://www.gartner.com/technology/home.jsp> >. Acesso em: Novembro de 2013.

OPEN HANDSET ALLIANCE. **Alliance**. Disponível em:

<http://www.openhandsetalliance.com/oha_overview.html/>. Acesso em: Novembro de 2013.

GALLERY ANDROID. **Android Building Framework**. Disponível em:

<<http://galleryandroid.com/android-development-framework/>>. Acesso em: Novembro de 2013.

BRINKMAN-DAVIS. **Essence of MVC**. Disponível em:

<<http://www.essenceandartifact.com/2012/12/the-essence-of-mvc.html>>. Acesso em: Novembro de 2013.

APP ANNIE. **Top Categories by Downloads**. Disponível em:

< <http://www.appannie.com/>>. Acesso em: Novembro de 2013.

PORTAL DO TURISTA. **Espaços Culturais**. Disponível em:

< http://www.portoalegre.travel/site/espacos_culturais.php >. Acesso em: Outubro de 2013.