



**Fundação Educacional do Município de Assis
Instituto Municipal de Ensino Superior de Assis
Campus "José Santilli Sobrinho"**

WILLIAM SARTI JOSÉ

**DESENVOLVIMENTO DE UM APLICATIVO ANDROID ROTEAMENTO
PARA TRANSPORTE COLETIVO**

**Assis/SP
2018**



**Fundação Educacional do Município de Assis
Instituto Municipal de Ensino Superior de Assis
Campus "José Santilli Sobrinho"**

WILLIAM SARTI JOSÉ

**DESENVOLVIMENTO DE UM APLICATIVO ANDROID ROTEAMENTO
PARA TRANSPORTE COLETIVO**

Projeto de pesquisa apresentado ao curso do Instituto Municipal de Ensino Superior de Assis – IMESA e a Fundação Educacional do Município de Assis – FEMA, como requisito parcial à obtenção do Certificado de Conclusão.

Orientando(a): William Sarti José

Orientador(a): Profº Me. Fábio Eder Cardoso

**Assis/SP
2018**

FICHA CATALOGRÁFICA

J83d	<p>JOSÉ, William Sarti Desenvolvimento de um aplicativo Android roteamento para transporte coletivo / William Sarti José. – Assis, 2018.</p> <p>60p.</p> <p>Trabalho de conclusão do curso (Ciência da Computação). – Fundação Educacional do Município de Assis-FEMA</p> <p>Orientador: Ms. Fábio Éder Cardoso</p> <p>1.Android 2.API 3. Google maps</p> <p>CDD 005.133</p>
------	--

DESENVOLVIMENTO DE UM APLICATIVO ANDROID ROTEAMENTO PARA TRANSPORTE COLETIVO

WILLIAM SARTI JOSÉ

Trabalho de Conclusão de Curso apresentado ao Instituto Municipal de Ensino Superior de Assis, como requisito do Curso de Graduação, avaliado pela seguinte comissão examinadora:

Orientador: _____
ProfºMe.Fábio Eder Cardoso

Examinador: _____
Prof. Dr.Luiz Carlos Begosso

Assis/SP
2018

DEDICATÓRIA

Dedico este trabalho à minha família que acreditou em mim e nos meus sonhos, me apoiando em todos os momentos que precisei.

AGRADECIMENTOS

Primeiramente a Deus por ter me capacitado e dado forças para prosseguir, pois sem Ele, não seria possível superar as minhas dificuldades.

Ao meu orientador, Prof^o Me. Fábio Eder Cardoso pela orientação, não somente por este trabalho, mas por toda minha vida acadêmica, por acreditar e tornar possível a construção de muitos de meus sonhos.

Aos meus familiares, Pai Alexandre Donizete José, Mãe Maria Angélica Sarti José, por estarem sempre ao meu lado, me apoiando e conduzindo-me durante minha caminhada.

“Uma pessoa com Ubuntu está aberta e disponível para as outras, apoia as outras, não se sente ameaçada quando outras pessoas são capazes e boas, com base em uma autoconfiança que vem do conhecimento de que ele ou ela pertence a algo maior que é diminuído quando outras pessoas são humilhadas ou diminuídas, quando são torturadas ou oprimidas.

SOU QUEM SOU PORQUE SOMOS TODOS NÓS.” Arcebispo Desmond Tutu.

RESUMO

Este trabalho apresenta a implementação de um aplicativo de gerenciamento de ônibus, auxiliando tanto os usuários do transporte coletivo, quanto as empresas; os passageiros poderão enviar *feedbacks* diretamente à empresa do ônibus, relatando os problemas e o que poderia melhorar no percurso; uma das vantagens da ferramenta é o chatterbot, um robô inteligente que traz informações fornecidas em tempo real, esclarecendo as possíveis dúvidas do passageiro, antes ou durante o trajeto, como horário de saída e de chegada do ônibus. O foco é melhorar a qualidade do serviço prestado pelo transportador e a satisfação do transportado. O presente trabalho utiliza a API (*Application Programming Interface*) da Google Maps, o manychat utilizando a tecnologia chatterbot para ajudar na informação.

Palavras-chave: Comunicação chatterbot, API, Google Maps.

ABSTRACT

This work will presents the implementation of a bus management application, helping both public transport users and companies; passengers can send feedback directly to the bus company, reporting problems and what could be improved on the route; one of the advantages of the tool is chatterbot, an intelligent robot that brings information provided in real time, clarifying the possible doubts of the passenger, before or during the journey, as the departure and arrival times of the bus. The focus is on improving the quality of the service provided by the transporter and the satisfaction of the transported. The present work will use the API (Application Programming Interface) of Google Maps, manychat using chatterbot technology to aid in information.

Keywords: Communication chatterbot, API, Google Maps.

LISTA DE ILUSTRAÇÕES

Figura 1-Etapas de compilação e execução de um programa Java (JUNIOR, 2017).....	21
Figura 2 - Camadas da Arquitetura Android (ANDROID DEVELOPERS, 2017)	24
Figura 3 - Ciclo de vida de uma atividade (ANDROID DEVELOPERS, 2017).....	25
Figura 4 - API da Google Maps distância entre pontos (Sobre o Google Maps, 2018)	29
Figura 5 - Logo Astah Community (Sobre o Astah, 2017).	30
Figura 6 - Painel de controle do Manychat (Sobre o Manychat, 2017)	31
Figura 7 - Chatbot Eliza (SGANDERLA, 2017).....	33
Figura 8 – Mapa mental chaterboots (Do autor)	35
Figura 9 - Diagrama Entidade e relacionamento (FIREBASE DATABASE, 2017)	36
Figura 10 - Diagrama de funcionamento do Usuario (Do autor)	37
Figura 11 - Diagrama de funcionamento do Contato (Do autor).....	38
Figura 12 - Diagrama de funcionamento da Mensagem (Do autor).....	38
Figura 13 – Diagrama de caso de uso geral (Do autor).....	39
Figura 14 - Caso de Uso: Usuário (Do autor)	40
Figura 15 - Caso de Uso 1: Login (Do autor).....	41
Figura 16 - Caso de Uso 2: Enviar de Mensagem (Do autor).....	42
Figura 17 - Caso de Uso 3: Recebimento de Mensagem (Do autor).....	43
Figura 18 - Caso de Uso 4: Visualização de Mensagem (Do autor).....	44
Figura 19 - Caso de Uso: Cadastro de Usuários (Do autor).....	45
Figura 20 - Tela de Login (Do autor).....	46
Figura 21 - Tela de Cadastro (Do autor).....	47
Figura 22 - Tela Bem-Vindo (Do autor).....	48
Figura 23 - Tela Localização (Do autor).....	49
Figura 24 - Tela sobre a empresa (Do autor).....	50

Figura 25 - Pagina do Facebook atendimento online 24hr (Facebook, 2018)	51
Figura 26 - Tela de contato (Do autor).....	52
Figura 27 – Problemas com Ônibus (Do autor)	53
Figura 28 – Aceitação do aplicativo (Do autor)	54
Figura 29 - Questionário feito em rede social Facebook (Do autor)	55
Figura 30 - Questionário do Google Forms compartilhado em rede social (Do autor)	56

LISTA DE TABELAS

Tabela 1 - Narrativa do 1º Caso de Uso: Login	41
Tabela 2 - Narrativa do 2º Caso de Uso: Envio de Mensagem.....	42
Tabela 3 - Narrativa do 3º Caso de Uso: Recebimento de mensagem.....	43
Tabela 4 - Narrativa do 4º Caso de Uso: Visualização de Mensagem.....	44
Tabela 5 - Narrativa do 5º Caso de Uso: Cadastro de Usuários.....	45

LISTA DE ABREVIATURAS E SIGLAS

API- Application Programming Interface

CLDC- Connected Limited Device Configuration

GPS- Global Positioning System

GSM- Global System for Mobile Communications

IDE- Integrated Development Environment

J2EE- Java 2 Enterprise Edition

J2ME- Java 2 Micro Edition

J2SE- Java 2 Standard Edition

JDK- Java Development Kit

JRE- Java Runtime Edition

JVM- Java Virtual Machine

OHA- Open Handset Alliance

SDK- Software Development Kit

SPTrans- São Paulo Transporte

UML- Unified Modeling Language

URBS- Urbanização e Saneamento de Curitiba

VM- Virtual Machine

XML- Extensible Markup Language

SUMÁRIO

1. INTRODUÇÃO	15
1.1. MOTIVAÇÃO	16
1.2. OBJETIVOS	16
1.3. JUSTIFICATIVA	16
1.4. PERSPECTIVA DE CONTRIBUIÇÃO.....	17
1.5. METODOLOGIA.....	17
1.6. ESTRUTURA DO TRABALHO.....	17
2. REVISÃO DA LITERATURA.....	18
3. METODOLOGIA DE DESENVOLVIMENTO	20
3.1. MATERIAIS	20
3.2. JAVA	20
3.3. PLATAFORMA ANDROID	22
3.4. ARQUITETURA.....	22
3.5. CONCEITOS BÁSICOS E ANATOMIA.....	23
3.6. BANCO DE DADOS NÃO-RELACIONAL	27
3.7. FIREBASE.....	27
3.8. FIREBASEUI	28
3.9. FREEMIND.....	28
3.10. API GOOGLE MAPS.....	28
3.11. ASTAH COMMUNITY	30
3.12. MANYCHAT	31
3.13. INTELIGÊNCIA ARTIFICIAL (IA)	32
3.14. CHATTERBOT	32
4. DESENVOLVIMENTO DO APLICATIVO	34
4.1. ESPECIFICAÇÃO DO APLICATIVO.....	34
4.1.1. DIAGRAMAS.....	34
4.1.2. MAPA MENTAL	34

4.1.3.	DIAGRAMA DE ENTIDADE E RELACIONAMENTO	35
4.1.4.	USUÁRIO.....	36
4.1.5.	CONTATO.....	37
4.1.6.	MENSAGEM	38
4.1.7.	DIAGRAMA DE CASO DE USO GERAL	39
4.1.8.	USUÁRIO.....	40
4.1.9.	ESPECIFICAÇÃO DOS CASOS DE USO	41
4.1.10.	LOGIN.....	41
4.1.11.	ENVIO DE MENSAGEM	42
4.1.12.	RECEBIMENTO DE MENSAGEM	43
4.1.13.	VISUALIZAÇÃO DE MENSAGEM	44
4.1.14.	CADASTRO DE USUÁRIOS.....	45
4.2.	IMPLEMENTAÇÃO DO APLICATIVO.....	46
4.2.1.	TELA DE LOGIN.....	46
4.2.2.	TELA DE CADASTRO	47
4.2.3.	TELA DE BEM-VINDO.....	48
4.2.4.	TELA DE LOCAL	49
4.2.5.	SOBRE A EMPRESA.....	50
4.2.6.	CHATTERBOT	51
4.2.7.	CONTATO.....	52
5.	AVALIAÇÃO DO APLICATIVO DESENVOLVIDO	53
5.1.	IDENTIFICAÇÃO PERFIL	53
5.2.	SUGESTÕES E OPINIÕES	57
6.	CONCLUSÃO	58
	REFERÊNCIAS.....	59

1. INTRODUÇÃO

Nos últimos anos, com o exponencial crescimento na área da tecnologia da informação, juntamente com uma alta demanda de soluções móveis, os smartphones tornaram-se um meio de comunicação muito utilizado por pessoas e empresas de pequeno, médio e grande porte; o crescimento do desenvolvimento de aplicativos para esses aparelhos cresce a cada dia, pois é muito mais fácil transportar um smartphone do que um notebook, além de se ter também um melhor custo/benefício. A partir da grande demanda do desenvolvimento de aplicativos para tais aparelhos, é necessário desenvolver outro tipo de software que verifique, automaticamente, o transporte público coletivo que é responsável pelo deslocamento diário dos cidadãos pela cidade (RABELLO, 2017).

O presente trabalho propõe a elaboração de um software que pode ser fornecido para servir àqueles que não possuem um veículo próprio para sua locomoção, e que se utilizam deste transporte para se deslocarem ao trabalho, escola, dentre outros inúmeros locais. Os usuários do transporte coletivo conhecem bem o trajeto percorrido pelo ônibus que ele geralmente utiliza, os seus horários e pontos de parada; porém, uma pessoa que não possui conhecimento das linhas de ônibus disponibilizadas pela empresa de transporte coletivo, ao chegarem ao terminal ficam perdidas, sem saber qual ônibus pegar para chegar ao seu destino; é neste contexto que o presente estudo se insere, tendo como objetivo facilitar o dia a dia do usuário do transporte coletivo da cidade de Assis/SP, por meio do acompanhamento das linhas do transporte municipal. Para esse projeto, foi desenvolvido um Aplicativo de Comunicação e Localização, que disponibilizará, ao usuário, em caso de dúvida, qual linha se deve pegar para ir ao destino esperado, ou o ponto próximo ao indicado na pesquisa, mostrando no mapa a localização do ônibus (RECK, 2017). A problematização encontrada no desenvolvimento do trabalho é apresentada, quando dois ou mais ônibus passam pela mesma rua. Como identificar o ônibus pesquisado pelo aplicativo no mapa já que na rua também estão passando outros ônibus com destinos diferentes? Para solucionar este problema cada linha disponibilizada terá um número de identificação associado à rota percorrida, pelo veículo, facilitando a sua localização.

1.1. MOTIVAÇÃO

A motivação para o desenvolvimento deste projeto de pesquisa foi o constante crescimento nas vendas de dispositivos móveis. Assim o aplicativo vem auxiliar os usuários que poderão utilizá-los em diversos locais.

Aplicar os conceitos estudados para o desenvolvimento do Plugin Google Maps na plataforma Google Android agregará muito valor aos desenvolvedores, pois não precisarão testar suas aplicações manualmente. Eles oferecem um Plugin pronto e específico para testar o roteamento em um aplicativo desenvolvido, assim tornando mais rápido o desenvolvimento e não lhe ocuparão bastante tempo.

1.2. OBJETIVOS

O presente projeto tem como objetivo a implementação de um aplicativo que venha auxiliar os usuários do transporte coletivo com informações em tempo real. O sistema proposto está disponível a visualização no mapa (local exato onde está o ônibus e o usuário no trajeto da viagem), com um marcador que possibilita o envio do feedback pelo passageiro para a empresa do transporte, elogiando ou criticando o serviço prestado. Este aplicativo também conta com um robô online integrado ao Facebook para divulgar informações sobre as rotas e os horários dos ônibus, pontos turísticos da cidade de Assis-SP, além de direcionar o usuário a um atendimento direto à empresa.

1.3. JUSTIFICATIVA

Espera-se que o desenvolvimento do Aplicativo venha facilitar o dia a dia dos usuários, trazendo informações necessárias para auxiliar nas dúvidas do usuário, aumentando a qualidade do transporte coletivo de Assis-SP.

1.4. PERSPECTIVA DE CONTRIBUIÇÃO

Ao término desta pesquisa, pretende-se publicá-la na Google Play Store, também em artigos e em instituições de ensino para que pessoas com interesse nesta área possam aproveitar esta pesquisa em seus trabalhos. Pretende-se também promover e compartilhar os conhecimentos e resultados alcançados. O aplicativo possibilitará uma grande ajuda aos usuários do transporte público na cidade de Assis/SP.

1.5. METODOLOGIA

Este trabalho utiliza como base, pesquisas com usuários, para levantamento de informações e problemas, pesquisa em artigos e sites referentes ao tema como forma de obtenção de informações relevantes para o desenvolvimento da aplicação, resultando na melhoria do transporte público assisense.

1.6. ESTRUTURA DO DESENVOLVIMENTO DO APLICATIVO

O trabalho está organizado da seguinte forma:

- **Capítulo 1 – Introdução**, apresenta os objetivos, o público alvo e a justificativa para sua realização. Neste tópico é apresentado o tema do trabalho a ser desenvolvido;
- **Capítulo 2 – Revisão da literatura**, apresenta a revisão da literatura onde são apresentados os trabalhos correlatos e o estado da arte no contexto da pesquisa;
- **Capítulo 3 – Metodologia de desenvolvimento**, onde são descritas as ferramentas utilizadas no desenvolvimento do trabalho proposto;
- **Capítulo 4 – Desenvolvimento do aplicativo**, apresenta o aplicativo desenvolvido;
- **Capítulo 5 – Avaliação do aplicativo desenvolvido**, apresenta uma pesquisa feita para que os usuários avaliem o aplicativo;
- **Capítulo 6 – Conclusão**, apresenta as conclusões obtidas na elaboração do trabalho e o direcionamento para os trabalhos futuros.
- **REFERENCIAS**

2. REVISÃO DA LITERATURA

No desenvolvimento deste trabalho foram encontrados vários projetos similares ao proposto, visando atender aos usuários de transporte coletivo.

A SPTrans, São Paulo Transporte, responsável pela gestão das linhas de ônibus operadas por empresas privadas (SPTrans, 2017), oferece aos seus usuários através de sua página na web, o acompanhamento em tempo real do deslocamento dos ônibus, o tempo médio do percurso, o horário previsto de chegada dele nas paradas e também sugere o ônibus a ser pego perante a indicação do ponto de partida e destino e a indicação dos pontos mais próximos (Redação Olhar Digital, 2017).

Outro exemplo de empresa que fornece as mesmas funções citadas acima é o URBS, Urbanização de Curitiba S.A. (URBS, 2017).

Os passageiros de ônibus da cidade de São José dos Campos, desde dezembro de 2012 podem ter acesso a informações como horários, linhas e pontos de embarque e desembarque de ônibus na cidade, além do tempo estimado da viagem, oferecidas por uma ferramenta de consulta *on line* de itinerários, devido a uma parceria da secretaria de Transportes e o Google Mapas (ALVES, 2017).

Abordando este assunto foram desenvolvidos alguns trabalhos acadêmicos como o de Pancholok e Rodrigues (2007), que desenvolveram um sistema em C++ que faz o controle e tráfego do transporte coletivo via GPS. Este programa foi dividido em módulos, o módulo GPS, módulo GPRS, módulo controlador e servidor.

O módulo GPS é responsável pelo armazenamento dos sinais enviados pelos satélites com as coordenadas de localização ele envia essas informações ao módulo GPRS que, por sua vez, envia a um servidor central para o processamento de tais informações.

O software processa as informações armazenadas no servidor e disponibiliza aos usuários os itinerários dos ônibus, o seu trajeto atual, a velocidade, o tempo esperado até a próxima estação.

Bastos e Jaques (2009) desenvolveram um sistema web que auxilia os usuários do transporte público. O sistema foi desenvolvido em PHP, utiliza o MYSQL como gerenciador de banco de dados e para calcular e gerar as rotas dos ônibus, o algoritmo A* de inteligência artificial. É formado por dois módulos, administrador e usuário. O módulo administrador é responsável pelos cadastrados que gerenciam o sistema e o módulo usuário fica a parte de consulta onde ele entra com os dados de origem e o destino que se pretende chegar e lhe é fornecido as linhas que deve pegar e as paradas, além de mostrar no mapa a sua trajetória prevista.

O Google Transit foi lançado em 2005 nos Estados Unidos pela Google. Esta ferramenta está presente em mais de 2,8 mil cidades, incluindo 22 no Brasil, e algumas de suas funcionalidades são: o fornecimento de paradas, trajetos públicos, horários, distâncias, além de informações sobre tarifas de transporte como ônibus, estações de metrô, trem e balsa. A aplicação está disponível para desktop e dispositivos móveis Android e IOS (KURTZ, 2017).

3. METODOLOGIA DE DESENVOLVIMENTO

3.1. MATERIAIS

Para o desenvolvimento do trabalho proposto será utilizada algumas ferramentas: a linguagem Java, o ambiente de desenvolvimento Android Studio, o servidor do tipo Cloud Computing, o API do Google Maps, o Manychat, Astah Community, o banco de dados Firebase, o Messenger do facebook com Chatboot integrado.

3.2. JAVA

A linguagem Java foi desenvolvida por um grupo de pesquisadores da Sun Microsystems, liderados por James Gosling, que tinham como objetivo o desenvolvimento de novos computadores portáteis capazes de se comunicar entre si. Para isso eles decidiram desenvolver uma plataforma que funcionasse em diferentes tipos de equipamentos (GONÇALVES, 2017).

A linguagem escolhida para o desenvolvimento foi o C++, mas durante a realização do projeto foi constatado que a mesma não disponibilizava recursos necessários para tudo o que a equipe visionara. A partir daí a equipe começou a trabalhar em uma linguagem que atendesse a tudo o que eles pretendiam, surge então a linguagem OAK, que mais tarde foi renomeada com o nome de Java.

A linguagem Java é orientada a objetos, oferece mecanismos de encapsulamento, abstração, polimorfismo e hereditariedade. É uma linguagem compilada e interpretada, como mostra a figura 1. Durante o processo de compilação é gerado um código intermediário chamado de *bytecodes*, o que torna o programa independente de plataforma, não vinculada com ambiente de desenvolvimento, a sua restrição é apenas ter o JDK que é composto pelo compilador, máquina virtual - JVM (Java Virtual Machine) e as bibliotecas

de desenvolvimento instaladas no seu computador. O processo de interpretação é feito em tempo de execução sendo gerenciado pela JVM, demonstrada na Figura 1.

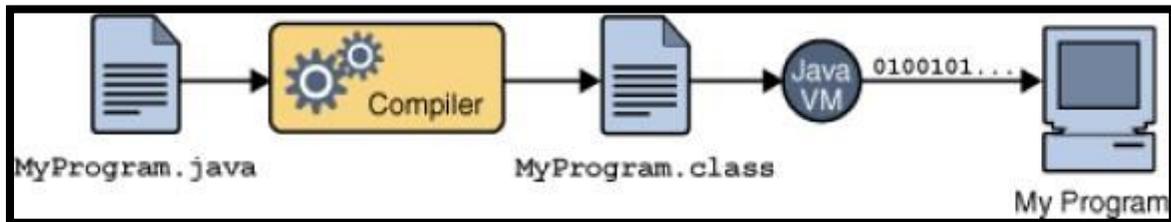


Figura 1-Etapas de compilação e execução de um programa Java (JUNIOR, 2017)

De acordo com (JUNIOR, 2017) a linguagem Java apresenta as seguintes características.

- Simples: permite o desenvolvimento de sistemas em diferentes sistemas operacionais e arquiteturas de hardware.
- Orientada a objetos: oferece mecanismos de abstração, encapsulamento e hereditariedade.
- Independente de plataformas: como descrito acima, no processo de compilação e interpretação dos *bytecodes* gerados pela JVM.
- Sem ponteiros: não permite a manipulação direta dos endereços de memória, possui o *garbage collector* que é responsável pelo mesmo.
- Permite *multithreads*: oferece recursos para aplicações capazes de executar rotinas concorrentemente através da sincronização destas rotinas, fluxos de exceções denominado threads.
- Segurança: pode executar programas através de uma rede especificando restrições de níveis de acesso (JUNIOR, 1999).

Java divide-se em três edições: Java 2 Standard Edition (J2SE), Java 2 Mobile Edition (J2ME), Java 2 Enterprise Edition (J2EE) destinadas a tipos específicos de aplicações.

- Java 2 Standard Edition (J2SE): Tecnologia projetada para computadores pessoais e ambientes de trabalho. Possui duas divisões JDK (Java Development Kit), composto pelo conjunto de desenvolvimento, tudo o que é necessário para execução de aplicações em Java e o JRE (Java Runtime Edition), versão preparada para o ambiente de execução.

3.3. PLATAFORMA ANDROID

O sistema operacional Android é um software open Source desenvolvido visando um vasto conjunto de dispositivos com características diferentes. Seu objetivo principal é criar uma plataforma onde desenvolvedores, operadoras e fabricantes possam inserir suas ideias e inovações resultando em um produto que realmente aprimora a experiência do usuário (Sobre o Android, 2017). Nesta subseção as principais camadas da arquitetura Android serão ilustradas juntamente com os principais conceitos de uma aplicação Android e sua anatomia.

3.4. ARQUITETURA

A arquitetura da plataforma Android é dividida em camadas onde cada uma é responsável por seus processos, essas camadas são:

Aplicações: camada onde são encontradas as aplicações que ficam à disposição do usuário, essas aplicações podem ser tanto os aplicativos nativos do sistema operacional como os aplicativos baixados pelo usuário, todos ficam localizados nesta camada.

Android framework: nesta camada se encontram as funções básicas do telefone tais como o sistema de localização, o sistema de chamadas, notificações, etc. A camada de aplicações interage diretamente com essa camada para utilizar essas funções básicas.

Bibliotecas nativas e Android runtime: neste nível de camadas do Android são encontradas os níveis de bibliotecas nativas e a camada de execução (Android runtime). A partir da versão 5.0, cada aplicativo executa seus próprios processos em uma instância

própria de execução, o que aumenta a velocidade de resposta e reduz o consumo energético. Anteriormente à versão 5.0 o Android utilizava o Dalvik, uma máquina virtual Java otimizada para ambientes móveis. A camada runtime é responsável por transformar o código compilado pela máquina virtual em instruções nativas. Nesta mesma camada se encontram as bibliotecas nativas do Android, dentre elas a Opendgl, Sqlite, entre outras.

Camada de abstração de hardware: essa camada providencia as interfaces para acesso aos componentes de hardware do smartphone por meio de bibliotecas, assim quando a camada de framework faz uma requisição para utilizar um elemento como, por exemplo, o bluetooth, essa camada que trata este serviço.

Kernel Linux: como camada mais profunda, temos a do kernel do Linux, que nunca interage com os usuários ou desenvolvedores, mas é o coração de todo o sistema. Ela garante a maioria das funcionalidades essenciais tais como: segurança, gerenciamento de memória, configurações de segurança (Sobre o Android, 2017).

3.5. CONCEITOS BÁSICOS E ANATOMIA

Nesta subseção iremos dissecar a anatomia de um aplicativo Android e destacar seus conceitos de funcionamento mais importantes. As aplicações Android são escritas utilizando a linguagem Java, e cada aplicativo tem seu próprio recipiente fechado de segurança, que o protege e o impede de executar funções fora de seu domínio. Um aplicativo Android possui quatro tipos de componentes, que são eles: atividades, serviços, provedores de conteúdo e receptores de transmissão, como demonstra na Figura 2.

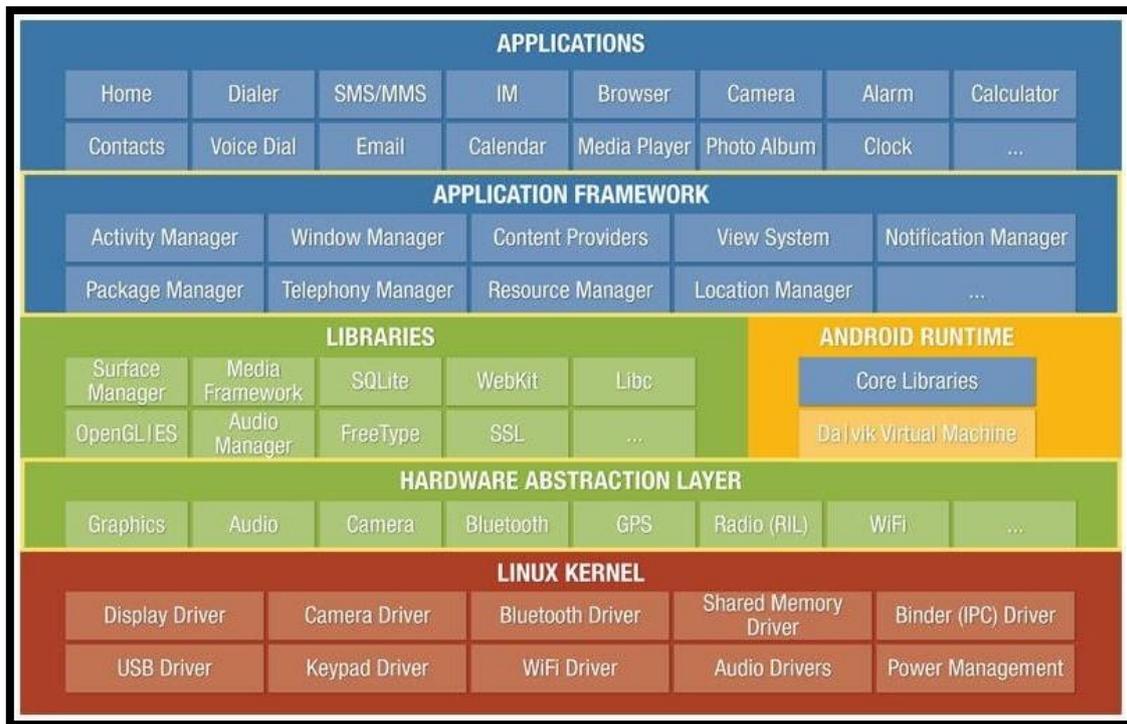


Figura 2 - Camadas da Arquitetura Android (ANDROID DEVELOPERS, 2017)

Atividades: uma atividade é uma das classes mais importantes de um aplicativo Android, é ela que gerência a interface visual apresentada ao usuário. Uma aplicação pode ter várias atividades e normalmente cada atividade representa uma funcionalidade do aplicativo. O entendimento do ciclo de vida de uma atividade é essencial para que o desenvolvedor possa construir um aplicativo que se comporte conforme desejado. Os estados em que uma atividade pode ser encontrada, demonstrada na Figura 3.

stopped: normalmente a atividade entra nesse estado quando ela é totalmente coberta, como por exemplo, por uma outra atividade.

destroyed: quando a atividade é encerrada ela entra nesse estado.

Sempre que uma atividade troca de estado alguns métodos da classe são chamados:

onCreate: chamado imediatamente após a criação da atividade e deve ser obrigatoriamente implementado, nela é definida o layout utilizado na atividade assim como seus atributos caso existam. (Sobre o Android, 2017).

onResume: método chamado sempre que a atividade inicia (em seguida da onCreate) ou reinicia (vindo dos estados pause e created)

onPause: método chamado quando uma atividade é deixada para segundo plano, recomenda-se gravar as informações necessárias, pois apesar de ser possível a retomada da atividade, nem sempre isso acontece.

onStop: método chamado quando uma atividade é totalmente encoberta e vai para segundo plano, também se recomenda salvar as informações desejadas.

onDestroy: método chamado para realizar qualquer limpeza final antes da atividade ser completamente destruída.

Serviços: os serviços da plataforma Android executam tarefas de propósito geral, que ficam em segundo plano rodando para todos tipos de motivo. É um componente que não possui uma interface com o usuário, por exemplo um aplicativo de música pode executar um serviço para ficar tocando música mesmo que outro aplicativo esteja aberto em primeiro plano.

Receptores de transmissão broadcast: os receptores de transmissão broadcast de uma aplicação funcionam de modo que ela precise ficar rodando para que ela execute alguma instrução caso um evento específico ocorra, basta ela configurar um receptor de transmissão para esse determinado evento, que o sistema faz o tratamento e anuncia à aplicação sempre que ele ocorrer, fazendo com que as instruções configuradas no receptor sejam executado.

Provedores de conteúdo: os provedores de conteúdo cuidam da parte de gerenciamento de acesso a um conjunto de dados. Os provedores são projetados para serem utilizados por outras aplicações através de um cliente, podendo fazer com que duas aplicações diferentes compartilhem dos mesmos dados.

3.6. BANCO DE DADOS NÃO-RELACIONAL

Nessa seção serão fornecidas informações sobre o banco de dados utilizado, no qual pertence à classe de banco não relacionais(NoSQL), uma tecnologia de banco de dados projetada para suportar os requisitos de aplicações em nuvens e arquitetado para superar a escala, desempenho, modelo de dados e as limitações de banco de dados relacionais (DATA SCIENCE ACADEMY, 2017).O banco utilizado foi o Firebase realtime database que consistem em um banco de dados não-relacional localizado na nuvem fornecido pela Firebase (FIREBASE DATABASE, 2017).Devido ao pequeno tamanho e complexidade do projeto, não havia necessidade da utilização de um banco NoSQL que possui como seu ponto forte a estabilidade, porém devido ao grande crescimento na utilização de bancos NoSQL que tem se notado, e ao crescimento de tecnologias que se beneficiam com esse tipo de banco, Tais como Big Data e Internet of things e etc.

3.7. FIREBASE

O Firebase é uma plataforma para a construção de aplicativos mobile e web através de ferramentas e infraestrutura que visam ajudar os desenvolvedores e construir aplicativos de qualidade (FIREBASE DATABASE, 2017) Onde são agrupados diversos serviços importantes tais como o sistema de análise (Firebase Analytics), sistema de autenticação de usuário (Firebase Auth), armazenamento (Firebase Storage) entre outros. Porém para este trabalho foi utilizado apenas o serviço de banco de dados não relacional, o Firebase Realtime Database. Esse banco de dados nada mais é do que uma árvore JSON gigante em que todos seus dados estão armazenados nos nodos, o que facilita uma modelagem simples de dados. O maior benefício do Firebase Database Realtime é que ele já possui um sistema de sincronização instantânea implementando, fazendo com que, caso ocorra uma modificação no banco, todos os aplicativos que tenham a referência daquele item, o atualizem automaticamente, ao invés de trabalhar com requisição e resposta normalmente utilizado em outros bancos.

3.8. FIREBASEUI

Em conjunto o Firebase realtime database também foi utilizado uma biblioteca FirebaseUI fornecida pela própria Firebase que possibilita uma integração rápida e fácil entre elementos UI e APIs do Firebase Database ou Firebase authentication (FirebaseUI for Android – UI Bindings for Firebase) abstraindo algumas funções do lado do usuário e automatizando o processo de sincronização entre banco e a UI.

3.9. FREEMIND

O FreeMind é um programa que permite o desenvolvimento de projetos, com ele pode-se montar o mapa mental do projeto, com todas as funcionalidades que o sistema deve possuir. (Sobre o Freemind, 2017).

3.10. API GOOGLE MAPS

O Google Maps é um serviço gratuito de pesquisa e visualização de mapas e imagens via satélite, desenvolvido pela empresa Google. Além de oferecer o serviço de mapas, possui uma ferramenta que calcula as direções entre dois pontos como demonstrado na figura 2, logo abaixo, também exibe rotas passo a passo, disponibiliza informações como tempo estimado por rota, se possui pedágios em sua extensão e também possibilita a identificação de pontos através dos seus marcadores, apresentado na Figura 4.

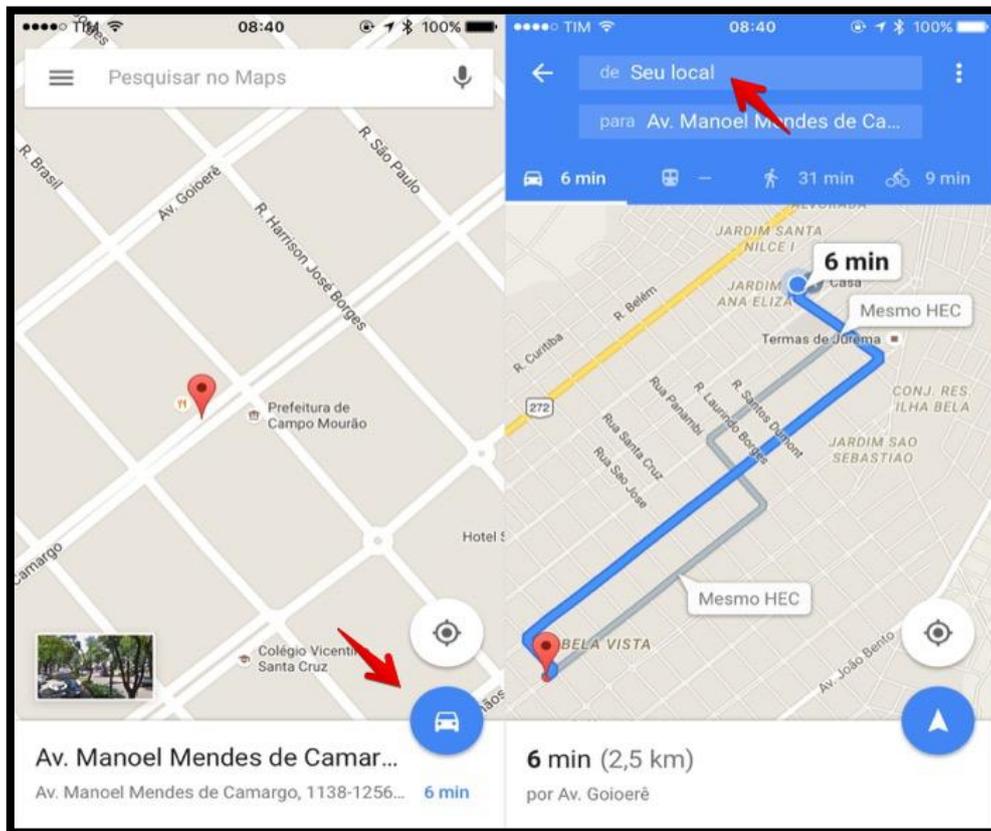


Figura 4 - API da Google Maps distância entre pontos (Sobre o Google Maps, 2018)

É possível integrá-lo em aplicações web ou dispositivos móveis, através da Google Maps API, um conjunto de classes Javascript que proporciona aos desenvolvedores, acesso aos serviços disponibilizados pelo Google Maps. Com ela é possível construir aplicativos que realizem consultas por endereços e utilizem funções de zoom, além de proporcionar a criação de mapas personalizados (Sobre o Google Maps, 2017).

3.11. ASTAH COMMUNITY

É uma ferramenta paga e disponibiliza também aos usuários uma versão Open Source, é utilizada na elaboração de diagramas UML (Unified Modeling 24 Language), permitindo uma melhor visualização dos produtos e processos a serem realizados pelo projeto (MUCIN, 2018). A UML é um modo de padronizar as formas de modelagem, é utilizado na especificação, documentação, estruturação para sub visualização lógica do desenvolvimento completo de sistemas de informação. Alguns de seus modelos de diagramas são: -Diagrama de Classes; -Diagrama de Caso de Uso; -Diagrama de Estados; -Diagrama de Atividades; -Diagrama de Sequência; -Diagrama de Objetos; -Diagrama de Implantação; -Diagrama de Estrutura Composta (Sobre o Astah, 2017), como demonstrado na Figura 5.



Figura 5 - Logo Astah Community (Sobre o Astah, 2017).

3.12. MANYCHAT

O Manychat é uma ferramenta especializada em bot (robô para entrega de mensagens), sendo mais específico, em bot para facebook Messenger. O uso de bot nos dias de hoje para ajudar durante o processo de construção de marca se tornou algo fundamental para a maioria das empresas que estão presente na internet e por isso que em hoje em dia existe inúmeras ferramentas na área de marketing digital para diversas funcionalidade. Podemos encontrar ferramentas para entregar e-mail, criar site, páginas de captura, gravar vídeo e muito mais. Isso para facilitar o trabalho dentro de uma empresa que quer crescer perante os desafios que enfrenta no empreendedorismo digital, por isso essas ferramentas podem ser muito vantajosas para nós. Logo abaixo o painel de controle do manychat que fornece detalhes gerais (Sobre o Manychat, 2017), como demonstrado na Figura 6.



Figura 6 - Painel de controle do Manychat (Sobre o Manychat, 2017)

3.13. INTELIGÊNCIA ARTIFICIAL (IA)

O termo Inteligência Artificial (IA) vem da junção de duas palavras distintas, mas que no início do seu surgimento criaram esperança em algumas pessoas de que as máquinas fossem capazes de possuir inteligência, e em outras, deixaram dúvidas da veracidade e capacidade de uma máquina ser inteligente (Berkeley, 1997). Em 1950, Alan Turing propôs o chamado Teste de Turing, que tinha como objetivo identificar se o computador e um interrogador que poderia ser de qualquer sexo. Cada um ficava em uma sala separada dos outros dois e a comunicação era realizada via teclado. O objetivo do jogo, para o interrogador, era determinar qual dos outros dois era o homem. A máquina tinha como objetivo responder as perguntas filosóficas por meio de uma definição de inteligência totalmente operacional, fazendo com que o interrogador pensasse que estava realmente conversando com um ser humano e não com uma máquina: se no final da conversa o interrogador não conseguisse distinguir quem era humano, então, dizia-se que o computador possui a inteligência (BROOKS, 1991). Dessa forma, percebe-se que ao longo dos anos a IA foi ganhando espaço e se desenvolvendo, e hoje existem diversos sistemas que aplicam algum conceito. Os Chatterbots ou agentes de conversação são exemplos da aplicação de conceitos de IA.

3.14. CHATTERBOT

O chatterbot é um sistema que utiliza linguagem natural para conversar com outras pessoas, ou seja, responde como se fosse um atendente humano. Além disso, o diálogo entre a pessoa e o chatterbot fica registrado em um arquivo de log. Dessa forma, a gerência poderá acompanhar o histórico das conversas (log), obtendo, as principais dúvidas dos clientes; auxiliando, com isso, em novas tomadas de decisões para alcançar os objetivos e metas. A palavra chatterbot surgiu da junção das palavras chatter (a pessoa que conversa) e da palavra bot (abreviatura de robot), ou seja, um robô (em forma de software) que conversa com as pessoas (CHATTERBOT, AIRES, 2017). Por muito tempo, desde o início da humanidade, a comunicação e os meios de comunicação vêm sofrendo fortes atualizações. Considerando o grau de dificuldade que o usuário tem para interagir com

diversos sistemas, e ainda o tempo gasto com estudos e treinamentos nesse segmento, o mais correto seria as máquinas adaptarem-se ao usuário, possibilitando uma forma de interação comum entre sistemas e humanos (SGANDERLA, 2017).

Exemplos de Chatterbots. Há alguns anos, o termo chatterbot, ou agente de conversação, ainda não era conhecido. Contudo, hoje já é possível encontrar muitos chatterbots disponíveis na internet. Em pesquisas realizadas, descobriu-se que já existem diversos chatterbots, os quais possuem como função atender clientes de empresas. Cada vez mais as empresas investem nessa nova modalidade de atendimento. Os chatterbots estão inclusos em diversas áreas, como: saúde, educação, comercial, governamental, entre outras. A seguir, destacam-se os principais agentes de conversação disponíveis na internet.

Eliza, Iniciando a primeira geração de chatterbot em 1966 e desenvolvido por Joseph Weizenbaum, no Massachusetts Institute of Technology (Instituto de Tecnologia de Massachusetts), nasceu o primeiro chatterbot, Eliza. Este tinha por objetivo fazer uma psicanálise e diagnosticar, por meio de diálogos amigáveis, problemas psicológicos em pacientes (LEONHARDT MENEZES, 2003), como demonstra a Figura 7.



Figura 7 - Chatbot Eliza (SGANDERLA, 2017)

4. DESENVOLVIMENTO DO APLICATIVO

O Aplicativo possui dois módulos: usuário e empresarial.

No módulo usuário está concentrada a parte disponível, a visualização no mapa (local exato onde está o ônibus e o usuário no trajeto da viagem), com um marcador que possibilita o envio do *feedback* pelo passageiro para a empresa do transporte, elogiando ou criticando o serviço prestado.

O módulo empresarial está voltado mais para parte de *Marketing*, com um robô online integrado ao Facebook para divulgar informações sobre as rotas e os horários dos ônibus, pontos turísticos da cidade de Assis-SP, além de direcionar o usuário a um atendimento direto à empresa.

4.1. ESPECIFICAÇÃO DO APLICATIVO

Será feita uma descrição da especificação do aplicativo desenvolvido neste projeto.

4.1.1. DIAGRAMAS

Os diagramas são uma modelagem muito útil para o desenvolvimento de sistemas que demonstram como é o funcionamento do sistema, pois define como o sistema deve interagir.

4.1.2. MAPA MENTAL

O mapa mental é um diagrama com um assunto principal, onde dele saem ramificações com informações relacionadas. Essas informações são normalmente cores, desenhos e palavras-chave para facilitar a memorização.

Na figura abaixo foi feita uma pesquisa sobre os locais e os horários de saída dos ônibus, cada um conforme seus respectivos cronogramas; elaborados para os usuários do serviço do transporte coletivo assisense, como demonstra a Figura 8.

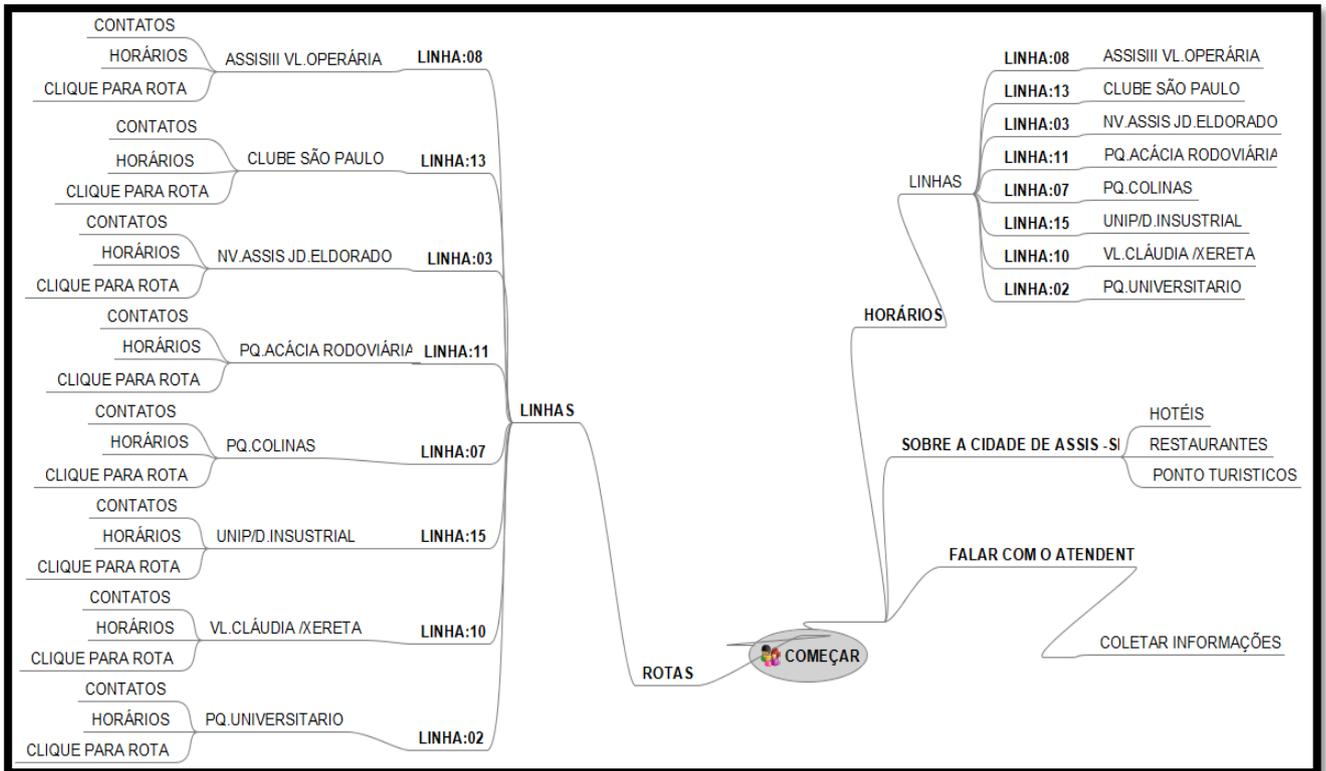


Figura 8 – Mapa mental chaterbots (Do autor)

4.1.3. DIAGRAMA DE ENTIDADE E RELACIONAMENTO

A modelagem de entidades e relacionamento, baseada em estórias, facilita o desenvolvimento da aplicação, pois orienta como suas estruturas devem ser construídas, organizadas e relacionadas para prover um funcionamento conforme esperado. Tendo em vista que para este trabalho foi utilizado um banco de dados não relacional (FIREBASE, 2017), entidades de relação e outras menores não são necessárias, portanto acabaram sendo incorporadas em duas entidades robustas.

É possível de se observar a forma com que as entidades e seus relacionamentos foram projetadas. Por se tratar de um aplicativo que utiliza um banco NoSQL, foi necessário adaptar um diagrama de entidades e relacionamento para que ele pudesse refletir como realmente ocorre esse relacionamento, como demonstra a Figura 9.

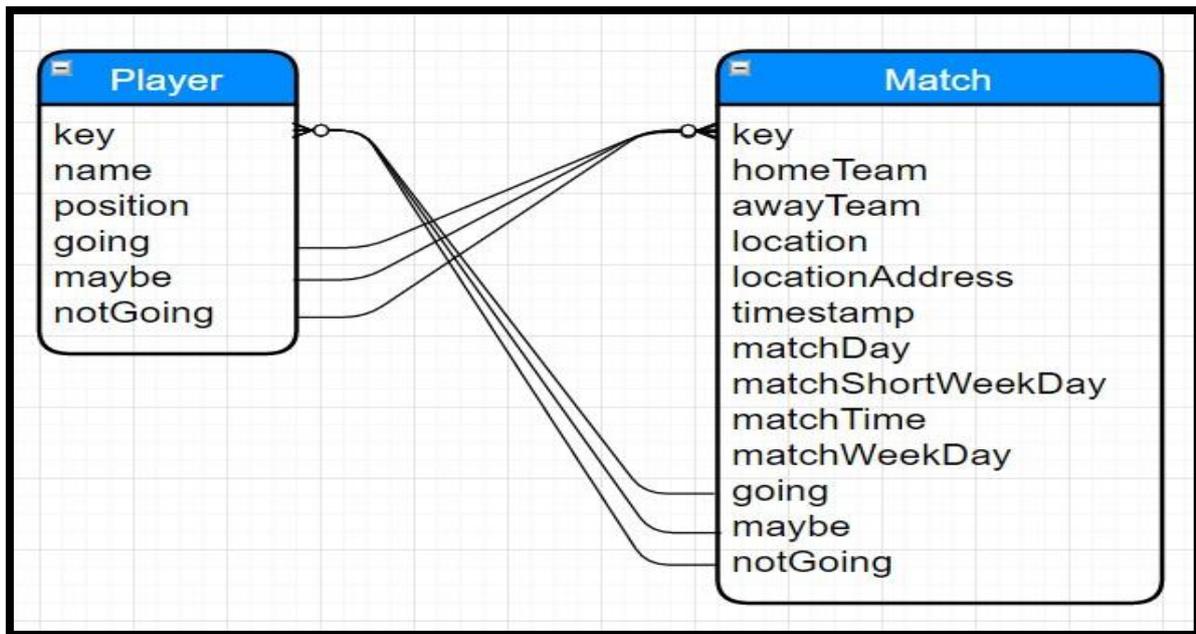


Figura 9 - Diagrama Entidade e relacionamento (FIREBASE DATABASE, 2017)

4.1.4. USUÁRIO

Essa entidade Usuário representa um chat, onde ao cadastrar, ela armazena o E-mail, id, nome, senha; o objetivo é de tornar o aplicativo mais simples e intuitivo ao usuário, demonstrado na Figura 10.

- **key:** chave única criada pelo Firebase ao inserir o dado no banco.
- **E-mail:** e-mail do Usuários.
- **ID:** Id para identificar o usuário.
- **Nome:** nome do Usuários.
- **Senha:** senha do Usuários.



Figura 10 - Diagrama de funcionamento do Usuário (Do autor)

4.1.5. CONTATO

Diferentemente da entidade Usuário, o Contato necessita de informações para que a aplicação funcione de modo desejado. O contato irá guardar as informações trocadas pelos Usuários, conforme demonstrado na Figura 11.

- **Key:** chave única criada pelo Firebase ao inserir o dado no banco.
- **E-mail:** e-mail do Usuários.
- **Identificador do Usuário:** Id para identificar o usuário.
- **Nome:** nome do usuários.



Figura 11 - Diagrama de funcionamento do Contato (Do autor)

4.1.6. MENSAGEM

Essa entidade Mensagem é bem complexa, como ilustra a figura 12; o usuário em amarelo é a Ana, que enviou mensagem para o vermelho que é o Sarti, ocorrendo um fluxo de mensagens entre ambos, apresentado na Figura 12.



Figura 12 - Diagrama de funcionamento da Mensagem (Do autor)

4.1.7. DIAGRAMA DE CASO DE USO GERAL

Este Diagrama de Caso de Uso Geral demonstra as operações possíveis para cada ator; O usuário poderá realizar quatro operações: enviar mensagem, receber mensagem, fazer login e visualizar mensagem. As três primeiras operações descritas serão realizadas por meio do envio de uma requisição na qual será uma operação, porém automática, como demonstrada na Figura 13.

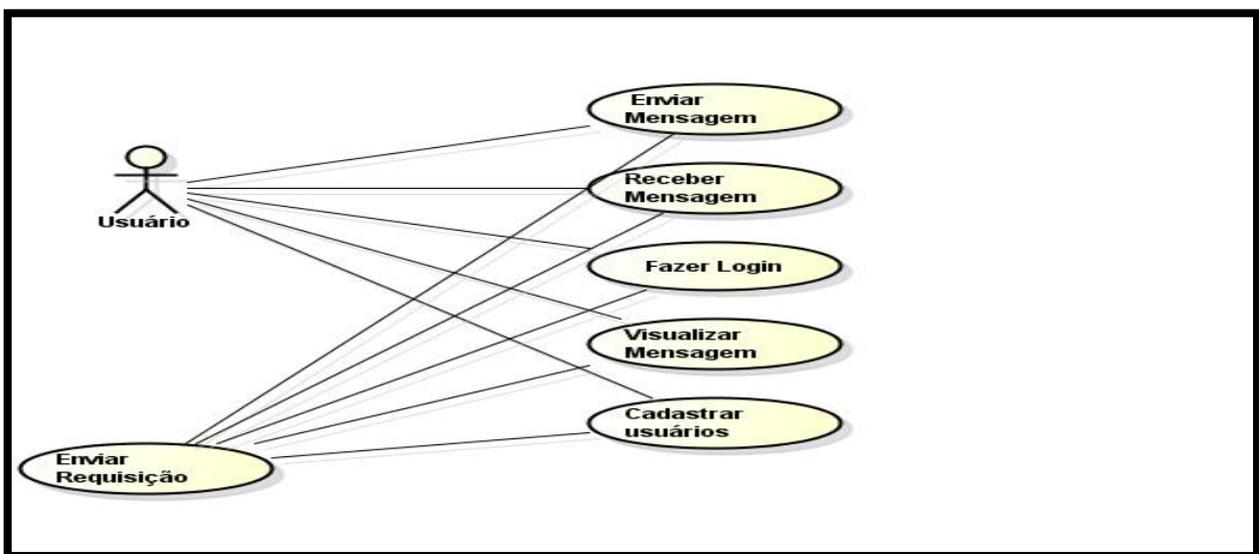


Figura 13 – Diagrama de caso de uso geral (Do autor)

4.1.8. USUÁRIO

A figura 14 ilustra as iterações do usuário com o sistema e as ações que deverão ser tomadas no uso da aplicação.

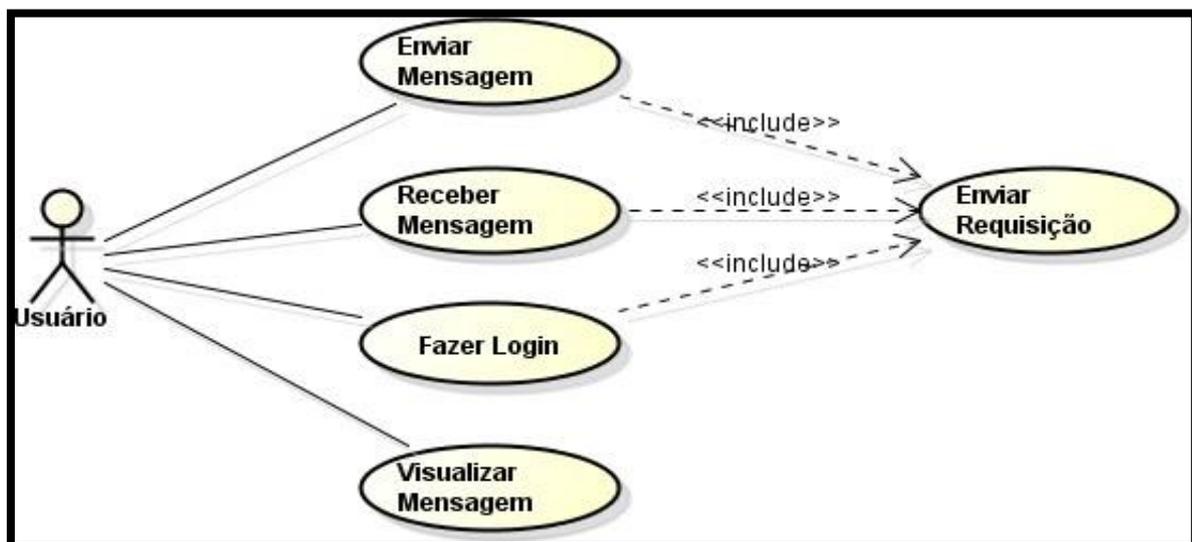


Figura 14 - Caso de Uso: Usuário (Do autor)

4.1.9. ESPECIFICAÇÃO DOS CASOS DE USO

A seguir serão apresentados os casos de uso específicos para cada iteração.

4.1.10. LOGIN

A figura 15 exemplifica o caso de uso do Login.

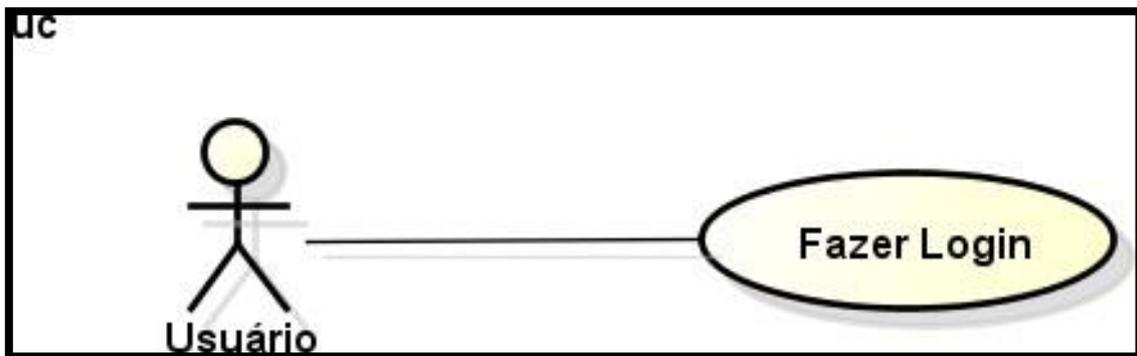


Figura 15 - Caso de Uso 1: Login (Do autor)

Finalidade/Objetivo:	Conectar o usuário ao Aplicativo, para ter certeza de Que ele pode ter acesso.
Ator (es):	Usuário.
Pré-condições:	Já estar cadastrado no Aplicativo.
Fluxo principal:	O usuário preenche os campos e pressiona o botão para fazer o login. O Aplicativo fará uma requisição e validará o usuário. Se houver dados incorretos ou falha durante a autenticação Retornará uma mensagem de falha.

Tabela 1 - Narrativa do 1º Caso de Uso: Login

O usuário acessará o Aplicativo e terá que fazer o login, quando ele acionar o botão para logar no aplicativo, os dados preenchidos serão enviados ao *banco de dados* verificando se há usuário cadastrado na base de dados.

4.1.11. ENVIO DE MENSAGEM

A figura 16 exemplifica o caso de uso do envio de mensagem.

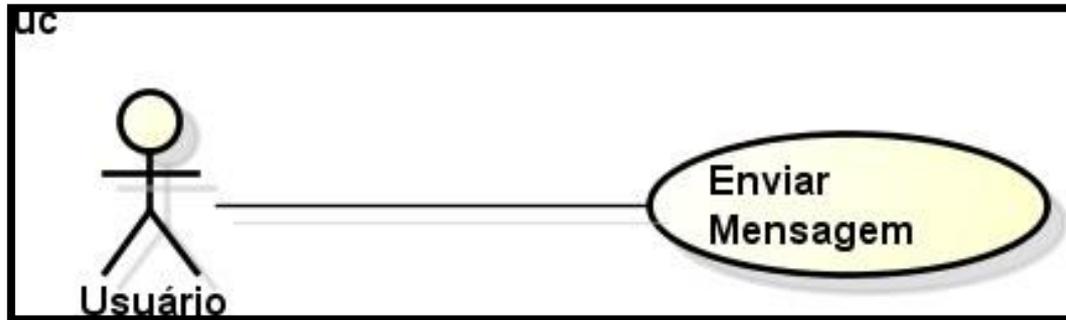


Figura 16 - Caso de Uso 2: Enviar de Mensagem (Do autor)

Finalidade/Objetivo:	Enviar mensagem.
Ator (es):	Usuário.
Pré-condições:	Estar logado no sistema e ter um outro usuário adicionado.
Fluxo principal:	<p>O usuário preenche os campos e pressiona o botão para enviar a mensagem.</p> <p>O sistema fará uma requisição para salvar os dados da mensagem no banco de dados. Se acontecer algum erro, durante o processo, uma mensagem de erro será retornada.</p>

Tabela 2 - Narrativa do 2º Caso de Uso: Envio de Mensagem

Enviar mensagem fará com que a aplicação envie uma requisição para *banco de dados*, passando por parâmetro, qual usuário aquela mensagem irá, quem está enviando. O usuário analisará e enviará os dados para serem salvos no banco de dados.

4.1.12. RECEBIMENTO DE MENSAGEM

A figura 17 exemplifica o caso de uso do recebimento de mensagens.

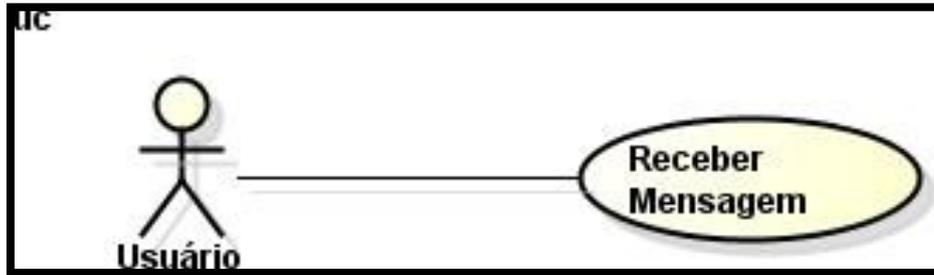


Figura 17 - Caso de Uso 3: Recebimento de Mensagem (Do autor)

Finalidade/Objetivo:	Receber mensagens.
Ator (es):	Usuário.
Pré-condições:	Estar logado no aplicativo.
Fluxo principal:	A cada período de tempo o aplicativo fará uma Requisição para obter novas mensagens.

Tabela 3 - Narrativa do 3º Caso de Uso: Recebimento de mensagem

Para receber mensagem o usuário não irá fazer nenhum estímulo; esta atividade será feita por um temporizador direto no aplicativo onde, a um determinado tempo, se fará uma requisição; ele confirmará se este usuário tem acesso ao aplicativo, caso tenha, ele irá até o banco pegar as mensagens novas e o banco entregará a mensagem ao usuário.

4.1.13. VISUALIZAÇÃO DE MENSAGEM

O caso de uso da visualização de mensagem, é apresentada na Figura 18.

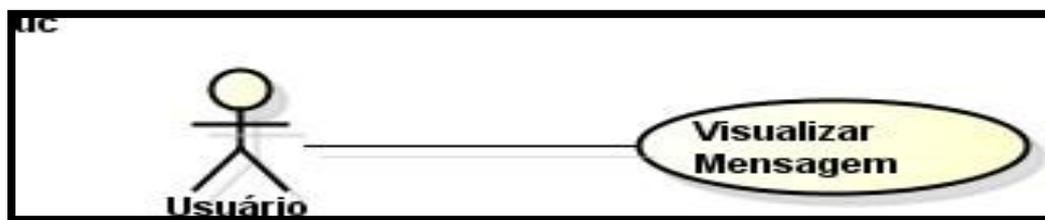


Figura 18 - Caso de Uso 4: Visualização de Mensagem (Do autor)

Finalidade/Objetivo:	Visualizar as mensagens .
Ator (es):	Usuário.
Pré-condições:	Estar logado no aplicativo.
Fluxo principal:	Receber as mensagens. Abrir a aba de mensagens.

Tabela 4 - Narrativa do 4º Caso de Uso: Visualização de Mensagem

4.1.14. CADASTRO DE USUÁRIOS

A figura 19 exemplifica o caso de uso de Cadastro de Usuários.

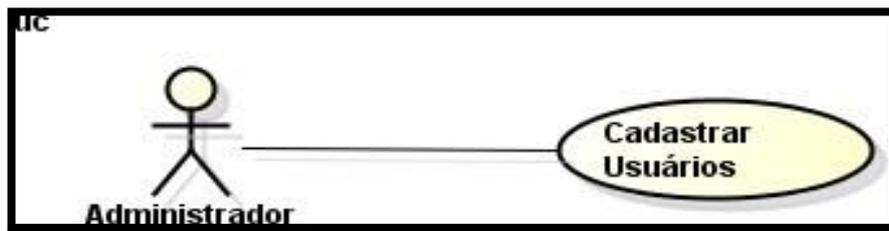


Figura 19 - Caso de Uso: Cadastro de Usuários (Do autor)

Finalidade/Objetivo:	Permitir inserção de novos usuários.
Ator (es):	Usuários.
Pré-condições:	Estar logado no aplicativo.
Fluxo principal:	Preenche os campos na tela e pressiona o botão cadastrar. Uma requisição será feita, e retornará uma mensagem de confirmação. Caso haja algum erro, uma mensagem de falha será o retorno.

Tabela 5 - Narrativa do 5º Caso de Uso: Cadastro de Usuários

Para o cadastro de usuário será enviada uma requisição que mandará os dados ao banco, para que possam ser salvos; o banco retornará uma mensagem: “cadastro feito com sucesso”.

4.2. IMPLEMENTAÇÃO DO APLICATIVO

O presente trabalho apresenta a implementação do aplicativo chatbus para o roteamento e gerenciamento do transporte coletivo. A implementação do aplicativo foi feita utilizando o Android Studio.

4.2.1. TELA DE LOGIN

A figura 20 apresenta a tela inicial para acessar o app; o usuário informa seus dados de acesso (Login e Senha) ou, se preferir, autenticar-se direto com o Facebook.

proto.io



Figura 20 - Tela de Login (Do autor)

4.2.2. TELA DE CADASTRO

A Figura 21 apresenta a tela “ao clicar em não tem conta?”; é o responsável por cadastrar os usuários, permitindo inserir novos usuários ao aplicativo, de acordo com os campos necessários.

proto.io



Figura 21 - Tela de Cadastro (Do autor)

4.2.3. TELA DE BEM-VINDO

A figura 22 apresenta a ilustração na tela principal quando o usuário já estiver logado, podendo acessar através dos ícones.

proto.io

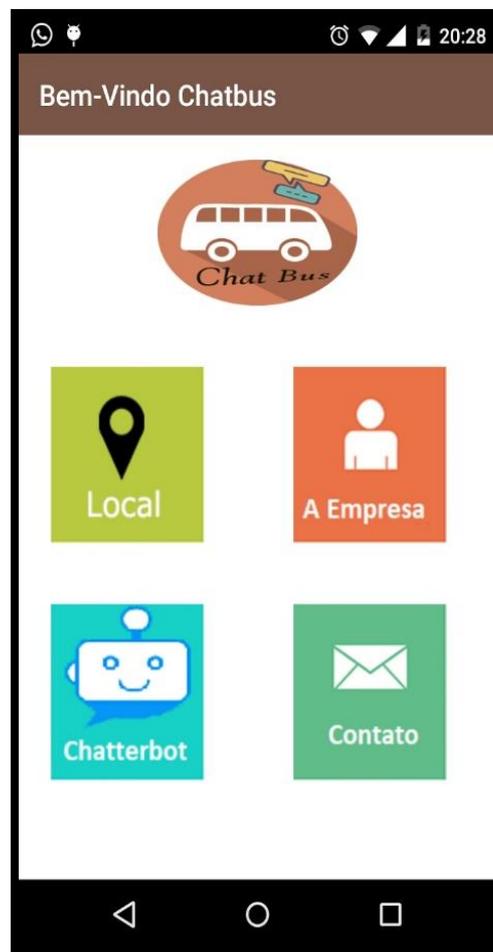


Figura 22 - Tela Bem-Vindo (Do autor)

4.2.4. TELA DE LOCAL

A figura 23 mostra a “tela de localização” com a visualização no mapa (local exato onde está o ônibus e o usuário no trajeto da viagem), com um marcador que possibilita o envio do feedback, pelo passageiro, para a empresa do transporte (elogiando ou criticando o serviço prestado).

proto.io



Figura 23 - Tela Localização (Do autor)

4.2.5. SOBRE A EMPRESA

Na figura 24, uma reportagem (<https://www.assiscity.com/?b=74821>) que conta como surgiu a ideia do aplicativo.

proto.io



Figura 24 - Tela sobre a empresa (Do autor)

4.2.6. CHATTERBOT

A tela que conta com um robô online, integrado ao Facebook, para divulgar informações (sobre as rotas, os horários dos ônibus, e os pontos turísticos, da cidade de Assis-SP), é a ilustração da figura 25.



Figura 25 - Pagina do Facebook atendimento online 24hr (Facebook, 2018)

4.2.7. CONTATO

A figura 26 ilustra a tela que direciona o usuário à um atendimento direto na empresa.

proto.io



Figura 26 - Tela de contato (Do autor)

5. AVALIAÇÃO DO APLICATIVO DESENVOLVIDO

O presente capítulo apresenta as análises dos resultados obtidos por meio de um questionário (elaborado após o desenvolvimento de uma ideia criada em um evento ocorrido na cidade de Assis em 2017, o *Startup Weekend*), que foi apresentado para as pessoas na rua, e nas redes sociais (Facebook e Google Forms). O objetivo desta análise é ter um feedback sobre algumas funcionalidades do aplicativo (como a facilidade de uso, as necessidades atendidas e não atendidas), aprimorando-o futuramente em conjunto com a equipe ChatBus (que é um aplicativo ainda em desenvolvimento para ser utilizado no dia-dia, criado com o objetivo de melhorar a experiência no transporte coletivo); o aplicativo demonstra os dados com antecedência, informando se o ônibus está atrasado, lotado, quebrado ou, de algum outro qualquer imprevisto, que possa ocorrer durante o percurso.

5.1. IDENTIFICAÇÃO PERFIL

Na identificação de perfil analisou-se os problemas que os usuários têm ao usar o ônibus, demonstrado na Figura 27.

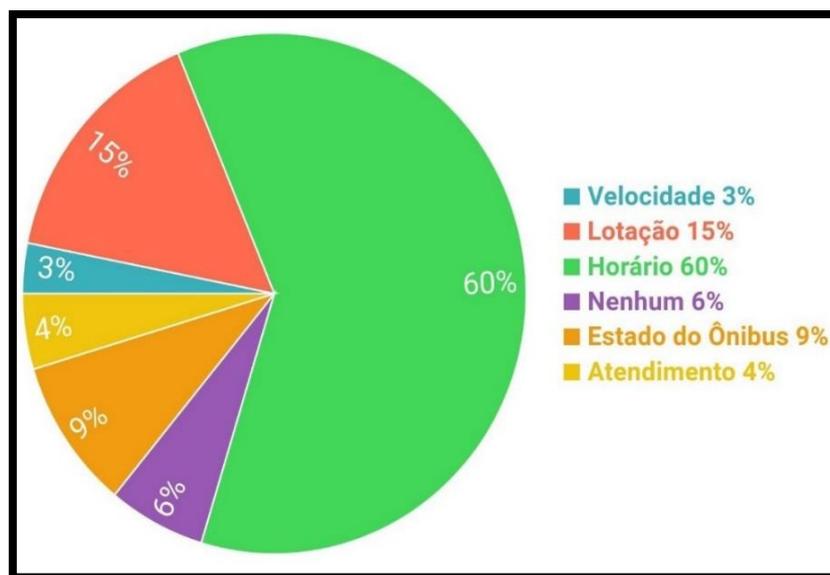


Figura 27 – Problemas com Ônibus (Do autor)

A figura 28 ilustra uma estatística de quem usaria ou não o aplicativo.

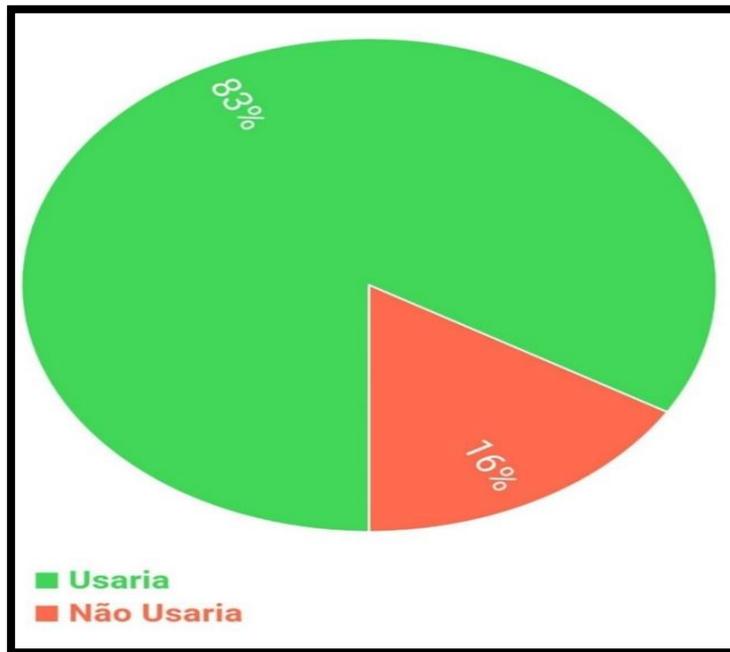


Figura 28 – Aceitação do aplicativo (Do autor)

Foram elaborados questionários em rede social para obter dados sobre quem usaria ou não o aplicativo, demonstrados na Figura 29.



Figura 29 - Questionário feito em rede social Facebook (Do autor)

Uma pesquisa foi elaborada para fazer o levantamento do aplicativo, conforme o Link do formulário:”https://docs.google.com/forms/d/18QTC8e8p1QkT8tIkzhkVOSOKPOddhXM6_fNEIrFlsVM/edit?usp=sharing”

Foi feita na pesquisa, entrevista com 25 pessoas sendo 15 mulheres e 10 homens; 6 deles com idade até 20 anos, 8 com idade de 20 a 30 anos, e 11 com mais de 30 anos. A frequência de utilização do transporte publico apontou para 12 pessoas que utilizam deste transporte diariamente; 8 pessoas semanalmente; 5 pessoas mensalmente. Das 25 pessoas entrevistadas, 24 gostariam de ser informadas dos pontos da rotas, 22 pessoas gostariam de ser informada do horário que o seu ônibus passaria; concluindo a importância do aplicativo.

The image shows a social media post from William Sarti Jose, dated October 21st. The post shares a Google Forms survey titled 'ChatBus'. The survey content is as follows:

ChatBus

Qual o maior problema que você tem hoje ao pegar o transporte público?

Your answer

Com que frequência você costuma utilizar o transporte público ?

- Diariamente
- Semanalmente
- Mensalmente
- Outros

Quanto tempo você passa esperando o ônibus?

ChatBus
DOCS.GOOGLE.COM

At the bottom of the post, there are icons for 'Curtir', 'Comentar', and 'Compartilhar'.

Figura 30 - Questionário do Google Forms compartilhado em rede social (Do autor)

5.2. SUGESTÕES E OPINIÕES

Ao finalizar o questionário, permite-se que o usuário deixe sua sugestão para a melhoria do aplicativo, podendo também opinar sobre sua utilidade e funcionalidade.

Foram realizados alguns pedidos de melhoria na interface e na correção de alguns bugs, como por exemplo: “A tela principal do aplicativo poderia ser mais organizada, mais bonita; poderiam haver filtros para classificar os feedbacks, os locais, os horários, etc.”.

Em relação às opiniões, teve-se um parecer bem positivo com todas as respostas (a maioria elogiando, demonstrando a utilização do aplicativo como uma possível solução). Eis uma das sugestões coletadas num dos maiores eventos de tecnologia da América Latina (Campus Party), por Dario Pires, mentor de startups: “O ChatBus é uma ótima ideia, creio que o aplicativo será bastante útil para sociedade”.

6. CONCLUSÃO

A proposta de trabalho foi o desenvolvimento de um aplicativo de gerenciamento de ônibus, com o intuito de auxiliar os usuários do transporte coletivo em um trajeto na cidade de Assis; criou-se um levantamento do funcionamento das linhas de ônibus e a identificação, para distinguir cada rota. Foi utilizada uma tecnologia de inteligência artificial: um robô online integrado ao Facebook para divulgar informações sobre as rotas e os horários dos ônibus, pontos turísticos da cidade de Assis-SP, além de direcionar o usuário a um atendimento direto com empresa responsável pelo transporte na cidade; o aplicativo auxiliará os usuários no dia a dia, atuando como uma ferramenta de busca e localização das linhas disponibilizadas pelo transporte coletivo da cidade.

Num trabalho futuro pretende-se desenvolver uma nova versão desta aplicação, tendo-se uma versão gratuita com serviços limitados, e uma paga com todos os recursos disponíveis; o projeto está sendo organizado em duas partes: uma na versão disponível na *playstore*, gratuitamente para os usuários baixarem (contendo recursos como os horários da rota do ônibus através de um campo de seleção de rotas ou de horários, contendo alguns conteúdos de publicidades); A outra na versão paga onde será cobrado um valor mensal, disponibilizando todos os conteúdos da inteligência artificial, que avisará o usuário onde é o ponto de ônibus mais próximo, e os horários, de acordo com seu histórico de uso, e não haverá conteúdos de publicidade.

REFERÊNCIAS

ANDROID DEVELOPERS. Android. Disponível em: Acesso em: Fevereiro de 2017.

AIRES, João Clayton. Desenvolvimento de um Chatterbot para o Portal de Ensino da Universidade do Oeste de Santa Catarina. Disponível em: <http://www.intercom.org.br/papers/regionais/nordeste2008/resumos/R12-0355-1.pdf>. Acesso em 01 janeiro 2017.

DATA SCIENCE ACADEMY. <<http://datascienceacademy.com.br/blog/2016/quando-utilizar-rdbms-ou-nosql/>> Acesso em novembro de 2017.

FIREBASE DATABASE. . <<https://firebase.google.com/docs/database/>> Acesso em Novembro de 2017.

GONÇALVES, E. – Dominando Java Server Faces e Facelets Utilizando Spring 2.5, Hibernate e JPA – Editora Ciência Moderna – Ano 2017.

JUNIOR, Peter Jandl. Introdução ao Java. Núcleo de Educação a Distância. Universidade de São Francisco, 1999. Disponível em <<http://markmail.org/download.xqy?id=zdem6rhmgwp5sefm&number=1>>. Acesso em Outubro de 2017.

Java. Sobre o Tutorial Java. O que é Java? Disponível em <<http://javafree.uol.com.br/artigo/871498/Tutorial-Java-O-que-e-Java.html>>. Acesso em Novembro de 2017.

KURTZ, João. Google Transit orienta sobre transporte público em 12 cidades sede da Copa. Disponível em <<http://techtudo.com.br/<http://www.techtudo.com.br/noticias/noticia/2017/04/google-transit-orienta-sobre-transporte-publico-em-12-cidades-sede-da-copa.html>>. Acesso em 03 Dezembro de 2017.

LEONHARDT, Michelle Denise; NEISSE, Ricardo; TAROUÇO, Liane Margarida Rockenbach. Meara: Um Chatterbot Temático para Uso em Ambiente Educacional. Disponível em Acesso em: Março de 2018

MUCIN, Samuel Paiva. Astah Community, um software para trabalhar com UMLs Disponível em<<http://www.plantaonerd.com/blog/2011/04/18/astah-community-um-software-para-trabalha-com-umls/>>. Acesso em: Janeiro de 2018

RABELLO, R.R.: Android – Um Novo Paradigma de Desenvolvimento Móvel,
http://www.cesar.org.br/files/file/WM20_Android_LBS.pdf Acesso em: Março de 2018

RECK, Garrone. Apostila Transporte Público. Disponível em <
http://www.dtt.ufpr.br/Transporte%20Publico/Arquivos/TT057_Apostila.pdf> Acesso em:
Fevereiro de 2018

Redação Olhar Digital. SPTTrans lança serviço online que divulga localização dos
ônibus em tempo real. Disponível em< <http://olhardigital.uol.com.br/noticia/sptrans-lanca-servico-online-que-divulga-informacoes-em-tempo-real-dos-onibus/25100>>.
Acesso em Outubro de 2017.

Sobre o Android. Introduction to Android. Disponível em <
<http://developer.android.com/guide/index.html>>. Acesso em: Março de 2017.

Sobre o manychat. o que é o manychat. Disponível em
<https://inicioavancadodigital.com/manychat-para-facebook-messenger/>. Acesso
em: Março de 2017.

Sobre o Astah. É hora de tentar Astah de graça! Disponível
em<<http://astah.net/editions/community>>. Acesso em: Janeiro de 2018.

Sobre o Freemind. FREEMIND. Disponível em <
<http://www.elirodrigues.com/2011/07/12/freemind/>> Acesso em: Março de 2018.

Sobre o Google Maps. Google Maps API. Disponível em <
<http://pesquompile.wikidot.com/google-maps-api>>. Acesso em: Março de 2018.

SGANDERLA, Rachele Bianchi. Chatterbot para Interação com Usuários. 2018.

Disponível em:

<http://tecnociencia.inf.br/comunidade/index.php?option=com_content&task=view&id=119&Itemid=41>. Acesso em: Janeiro de 2017.