

UNIVERSIDADE REGIONAL DE BLUMENAU
CENTRO DE CIÊNCIAS EXATAS E NATURAIS
CURSO DE CIÊNCIA DA COMPUTAÇÃO – BACHARELADO

**FURBMOBILE: UM PROTÓTIPO PARA VISUALIZAÇÃO DE
INFORMAÇÕES DO INTERAÇÃO FURB**

EVANDRO JOSÉ NUSS JUNIOR

BLUMENAU
2017

EVANDRO JOSÉ NUSS JUNIOR

**FURBMOBILE: UM PROTÓTIPO PARA VISUALIZAÇÃO DE
INFORMAÇÕES DO INTERAÇÃO FURB**

Trabalho de Conclusão de Curso apresentado ao curso de graduação em Ciência da Computação do Centro de Ciências Exatas e Naturais da Universidade Regional de Blumenau como requisito parcial para a obtenção do grau de Bacharel em Ciência da Computação.

Prof. Aurélio Faustino Hoppe, Mestre - Orientador

**BLUMENAU
2017**

FURBMOBILE: UM PROTÓTIPO PARA VISUALIZAÇÃO DE INFORMAÇÕES DO INTERAÇÃO FURB

Por

EVANDRO JOSÉ NUSS JUNIOR

Trabalho de Conclusão de Curso aprovado
para obtenção dos créditos na disciplina de
Trabalho de Conclusão de Curso II pela banca
examinadora formada por:

Presidente: _____
Prof. Aurélio Faustino Hoppe, Mestre – Orientador, FURB

Membro: _____
Profa. Joyce Martins, Mestre – FURB

Membro: _____
Prof. Dalton Solano dos Reis, Mestre – FURB

Blumenau, dia 11 de dezembro de 2017

Dedico este trabalho a todas as pessoas que sempre estiveram comigo e também para aquelas que me ajudaram de alguma maneira na realização deste trabalho.

AGRADECIMENTOS

À minha família que sempre me apoia em minhas escolhas e acredita em mim.

À minha namorada que sempre esteve ao meu lado.

Ao meu orientador, Aurélio Hoppe, por ter sugerido o tema em questão e dar todo o apoio necessário para a realização deste trabalho.

Aos meus amigos e colegas que de alguma forma me incentivaram e contribuíram para a realização deste trabalho.

Não deixe que o ruído da opinião alheia
impeça que você escute a sua voz interior.

Steve Jobs

RESUMO

Os dispositivos móveis estão cada vez mais inseridos no dia-a-dia das pessoas, com isso vem surgindo uma grande necessidade de disponibilizar as informações de maneira mais objetiva. A FURB não dispõe de um aplicativo que disponibilize informações para pessoas interessadas em ingressar na universidade. Estas informações são encontradas somente no site da FURB que não é preparado para receber acessos de dispositivos móveis. Dessa forma, este trabalho apresenta o desenvolvimento de um aplicativo Android que possibilita a visualização de informações relevantes para pessoas que estão interessadas em conhecer os cursos oferecidos pela FURB, assim como também tem o objetivo de auxiliar no controle de participação das oficinas do Interação FURB. O aplicativo foi desenvolvido utilizando a linguagem de programação JavaScript e a sintaxe JSX juntamente com o *framework* React Native. A parte do servidor foi desenvolvida na linguagem de programação C#. O aplicativo desenvolvido possibilita a visualização de informações dos cursos oferecidos pela FURB, algumas informações básicas sobre a Universidade, informações sobre o evento Interação FURB e suas oficinas, informações sobre matrícula, formas de ingresso e programas de apoio financeiro. Além disso, o aplicativo possui uma área restrita para ministrantes e visitantes das oficinas, para que sejam registradas as presenças dos visitantes via QRCode. Os resultados obtidos nos testes de usabilidade demonstraram que o protótipo desenvolvido foi capaz de facilitar o acesso as informações dos cursos oferecidos pela FURB e informações essenciais de ingresso na universidade.

Palavras-chave: Interação FURB. FURB-Mobile. React Native. Android.

ABSTRACT

Mobile devices are increasingly inserted in people's daily lives, with this a great need to make the information available more objectively has surged. FURB does not have an application that provides information for people interested in joining the university. This way, this work presents the development of an Android application that allows the visualization of relevant information for people who are interested in knowing the courses offered by FURB, as well as the objective of assisting in the participation control of the Interação FURB workshops. The application was developed using JavaScript programming language and JSX syntax along with React Native framework. The server part was developed in the programming language C#. The developed application allows you to view information about the courses offered by FURB, some basic information about FURB, information about the Interação FURB event and its workshops, enrollment information, entry forms and financial support programs. In addition, the application has a restricted area for attendees and workshop visitors to register attendance via QRCode. The results obtained in the usability tests demonstrated that the developed prototype was able to facilitate the access to the information of the courses offered by FURB and essential information of entrance in the university.

Key-words: Interação FURB. FURB-Mobile. React Native. Android.

LISTA DE FIGURAS

Figura 1 – Interação FURB no site da FURB.....	17
Figura 2 – Espaço no site da FURB reservado ao Interação FURB.....	17
Figura 3 – Site do Interação FURB	18
Figura 4 – Operações do React.....	20
Figura 5 – Telas disponíveis aos acadêmicos.....	22
Figura 6 – Telas disponíveis aos visitantes	22
Figura 7 – Telas de navegação	24
Figura 8 – Telas do aplicativo iAVA.....	25
Figura 9 – Diagrama de casos de uso do aplicativo móvel	30
Figura 10 – Diagrama de casos de uso do serviço para importação de informações dos sites e de arquivos Excel.....	31
Figura 11 – Arquitetura da solução	32
Figura 12 – Diagrama de classes do serviço de importação.....	34
Figura 13 – Diagrama de classes da API.....	35
Figura 14 – Diagrama de classe do pacote Model	36
Figura 15 – Diagrama de classes do pacote ModeloInterface.....	38
Figura 16 – Diagrama de classes do pacote Objetos.....	39
Figura 17 – Diagrama de classes do pacote Parametros e Util	40
Figura 18 – Tela inicial, menu lateral, tela de <i>login</i> e tela de <i>login</i> com e-mail do ministrante	48
Figura 19 – Tela A FURB, lista de letras, lista de cursos e informações do curso	49
Figura 20 – Tela ingresso, apoio financeiro e matrícula	50
Figura 21 – Tela do Interação FURB, lista de cursos com oficinas e informações das oficinas	51
Figura 22 – Menu lateral, tela de <i>check-in</i> , tela de minhas oficinas e informações das oficinas	52
Figura 23 – Menu lateral, tela de <i>check-in</i> e tela de inscritos por oficinas	53
Figura 24 – Notificação recebida pelo aplicativo	53

LISTA DE QUADROS

Quadro 1 – Requisitos Funcionais do aplicativo móvel.....	28
Quadro 2 – Requisitos Não Funcionais do aplicativo móvel	28
Quadro 3 – Requisitos Funcionais do serviço para importação de informações dos sites e de arquivos Excel	29
Quadro 4 – Requisitos Não Funcionais do serviço para importação de informações dos sites e de arquivos Excel.....	29
Quadro 5 – Código do método <code>ImportarOficinasSiteIteracao</code> da classe <code>ImportarDadosSite</code>	42
Quadro 6 – Localização das informações no arquivo <code>Inscrições.xlsx</code>	43
Quadro 7 – Localização das informações no arquivo <code>Ministrantes.xlsx</code>	43
Quadro 8 – Localização das informações no arquivo <code>Oficinas.xlsx</code>	43
Quadro 9 – Código do método <code>ImportarOficinas</code> da classe <code>ImportarDadosExcel</code>	44
Quadro 10 – Código do método <code>LerOficinas</code> da classe <code>LerDadosExcel</code>	45
Quadro 11 – Código do método <code>Post</code> da classe <code>Checkin</code>	47
Quadro 12 – Dispositivos móveis utilizados nos testes.....	62
Quadro 13 – Comparativo entre este trabalho e os trabalhos correlatos	62
Quadro 14 – Questionário de perfil dos visitantes	68
Quadro 15 – Lista de tarefas para visitantes.....	69
Quadro 16 – Questionário de usabilidade para visitantes.....	70
Quadro 17 – Questionário de perfil do ministrante	71
Quadro 18 – Lista de tarefas para ministrantes	72
Quadro 19 – Questionário de usabilidade para ministrantes	74

LISTA DE TABELAS

Tabela 1 – Perfil dos participantes do experimento atuando como visitante	55
Tabela 2 – Perfil dos participantes do experimento atuando como ministrante	56
Tabela 3 – Parte da lista de tarefas que não era necessário a realização de <i>login</i>	57
Tabela 4 – Parte da lista de tarefas que é necessária a realização de <i>login</i> com um visitante..	58
Tabela 5 – Parte da lista de tarefas para testar o aplicativo com um visitante sem internet.....	58
Tabela 6 – Parte da lista de tarefas que é necessário a realização de <i>login</i> com um ministrante	58
Tabela 7 – Parte da lista de tarefas para testar o aplicativo com um ministrante sem internet	59
Tabela 8 – Parte da lista que era necessário realizar tarefas no site da FURB	59
Tabela 9 – Questionário de avaliação do aplicativo atuando como visitante	60
Tabela 10 – Questionário de avaliação do aplicativo atuando como ministrante	61

LISTA DE ABREVIATURAS E SIGLAS

API – Application Programming Interface

AVA – Ambiente Virtual de Aprendizagem

DOM – Document Object Model

FURB – Fundação Universidade Regional de Blumenau

HTML – HyperText Markup Language

HTTP – HyperText Transfer Protocol

IDE – Integrated Development Environment

IIS – Internet Information Services

JSON – JavaScript Object Notation

JSX – JavaScript XML

RF – Requisitos Funcionais

RNF – Requisitos Não Funcionais

UC – Use Case

UI – User Interface

URL – Uniform Resource Locator

UX – User Experience

SUMÁRIO

1 INTRODUÇÃO.....	14
1.1 OBJETIVOS.....	15
1.2 ESTRUTURA.....	15
2 FUNDAMENTAÇÃO TEÓRICA	16
2.1 INTERAÇÃO FURB E SITE DA FURB	16
2.2 EXPERIÊNCIA DO USUÁRIO	18
2.3 REACT NATIVE.....	19
2.4 TRABALHOS CORRELATOS.....	21
2.4.1 FURB MOBILE: UMA APLICAÇÃO PARA VISUALIZAÇÃO E ACOMPANHAMENTO DA MATRIZ CURRICULAR.....	21
2.4.2 FURB-MOBILE: SISTEMA MÓVEL MULTIPLATAFORMA PARA NAVEGAÇÃO EM ROTAS INTERNAS.....	23
2.4.3 IAVA – APLICAÇÃO WEB PARA DISPONIBILIZAR RECURSOS DO AVA NO DISPOSITIVO MÓVEL IPHONE	24
3 DESENVOLVIMENTO.....	27
3.1 REQUISITOS.....	27
3.2 ESPECIFICAÇÃO	29
3.2.1 Casos de uso.....	29
3.2.2 Arquitetura	32
3.2.3 Diagramas de classes.....	33
3.3 IMPLEMENTAÇÃO	40
3.3.1 Técnicas e ferramentas utilizadas.....	40
3.3.2 Serviço de importação.....	41
3.3.3 Web API.....	46
3.3.4 Aplicativo móvel.....	48
3.4 ANÁLISE DOS RESULTADOS	54
3.4.1 Teste de usabilidade	54
3.4.2 Análise do perfil de usuário	55
3.4.3 Análise da lista de tarefas.....	57
3.4.4 Análise da avaliação do aplicativo.....	59
3.4.5 Teste de compatibilidade.....	61

3.4.6 Comparação com os trabalhos correlatos.....	62
4 CONCLUSÕES.....	64
4.1 EXTENSÕES	64
REFERÊNCIAS	66
APÊNDICE A – QUESTIONÁRIO DO PERFIL DE USUÁRIO, LISTA DE TAREFAS E AVALIAÇÃO DE USABILIDADE.....	68

1 INTRODUÇÃO

O uso de aplicativos vem facilitando a vida dos usuários e ao mesmo tempo fornecendo uma alternativa para se conectar à rede sem a necessidade de utilizar um navegador (ALTERMANN, 2010). Ao acessar um site, o usuário precisa abrir o navegador, acessar o site, que muitas vezes precisa de um *login*, para finalmente conseguir realizar uma determinada tarefa. Nos aplicativos, geralmente só é necessário realizar o *login* no primeiro acesso, nos demais acessos ele já estará pronto para ser utilizado (ALTERMANN, 2010).

Diante deste contexto, a FURB está trabalhando em um projeto chamado FURB-Mobile, que vai de encontro com o lema da Universidade, que acredita na importância de extrapolar os limites de salas e laboratórios, alcançando as pessoas onde quer que elas estejam, levando-as aonde elas desejam estar (UNIVERSIDADE REGIONAL DE BLUMENAU, 2017a). O FURB-Mobile não foi pensado somente para pessoas que já usufruem de serviços da FURB, mas também para pessoas que queiram conhecer um pouco mais sobre a Universidade. É neste cenário que entra o evento chamado Interação FURB. Ele tem como objetivo oportunizar à comunidade um primeiro contato com a universidade, conhecendo um pouco de sua estrutura, além de ter a possibilidade de participar de oficinas organizadas pelos cursos de graduação, intercâmbio, cursos de curta duração, ente outros (UNIVERSIDADE REGIONAL DE BLUMENAU, 2017c). Todavia, a universidade não dispõe de nenhum aplicativo móvel que facilite o acesso às informações dos cursos, tais como valor da mensalidade, formas de ingresso ou financiamento. É importante ressaltar que estas informações estão disponibilizadas no site da universidade. Porém, por ser um local com múltiplas finalidades (disponibilização de informações acadêmicas e administrativas), sua navegação e a localização das informações ficam bastante prejudicadas.

Outra problemática enfrentada em relação ao Interação FURB é o controle de acesso/presença nas oficinas oferecidas para os visitantes. Hoje, ao participar da oficina, o visitante ganha um adesivo comprobatório da sua participação. Ao juntar X adesivos, ele pode retirar um brinde junto à comissão organizadora. Por outro lado, a organização não possui um gerenciamento adequado e efetivo sobre a participação nas oficinas incluindo quais oficinas o visitante frequentou, qual é a relação entre elas, entre outras informações.

Diante do exposto, este trabalho apresenta o desenvolvimento de um aplicativo para a plataforma Android que disponibiliza, de maneira mais objetiva, informações para pessoas que estão interessadas em conhecer os cursos oferecidos pela FURB.

1.1 OBJETIVOS

O objetivo deste trabalho é desenvolver um aplicativo móvel para facilitar o acesso às informações dos cursos oferecidos pela FURB e, ao mesmo tempo, auxiliar na organização do Interação FURB.

Os objetivos específicos são:

- a) disponibilizar para consulta as informações sobre os cursos oferecidos pela FURB (apresentação, valor da mensalidade, intercâmbio, diferenciais do curso, projetos de iniciação científica e extensão, atividades extracurriculares, locais de trabalho e curiosidades);
- b) disponibilizar para consulta as informações sobre as oficinas do Interação FURB (resumo, horário e local);
- c) disponibilizar um mecanismo para que o usuário possa realizar o *check-in* ao participar de uma oficina no evento Interação FURB por meio de QR Code;
- d) disponibilizar um mecanismo de notificação para que o usuário seja informado sobre a proximidade do evento.

1.2 ESTRUTURA

A estrutura deste trabalho é composta por quatro capítulos. O primeiro contém uma introdução sobre o tema proposto. O segundo apresenta um embasamento teórico sobre o assunto abordado, sobre as tecnologias que serão utilizadas no decorrer do trabalho e também são apresentados os trabalhos correlatos. O terceiro expõe informações sobre o desenvolvimento do trabalho, bem como requisitos, especificação, implementação e análise de resultados. Por fim, no quarto capítulo são apresentadas as conclusões do trabalho e possíveis extensões.

2 FUNDAMENTAÇÃO TEÓRICA

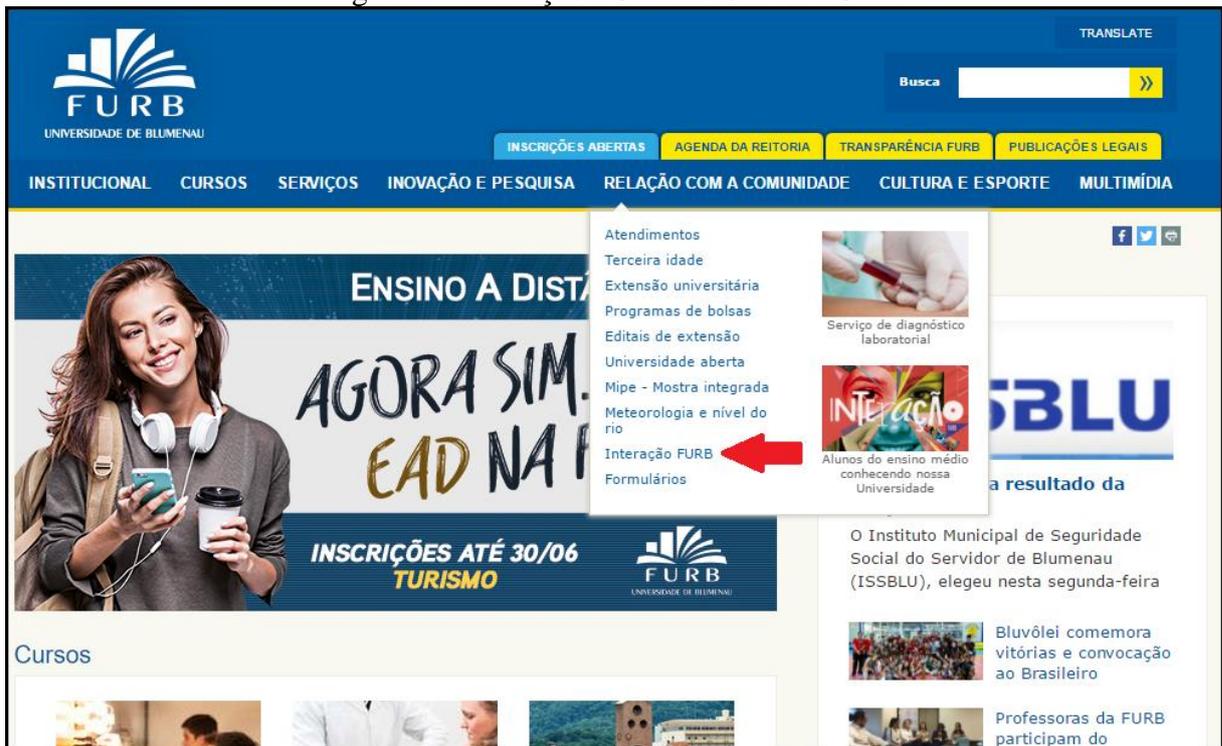
Neste capítulo são descritos os assuntos que fundamentaram o estudo realizado neste trabalho. Na seção 2.1 é apresentado o evento Interação FURB e o site da FURB. A seção 2.2 aborda o tema experiência do usuário. A seção 2.3 é dedicada a uma explicação sobre o *framework* React Native. Por fim, a seção 2.4 apresenta os trabalhos correlatos.

2.1 INTERAÇÃO FURB E SITE DA FURB

O evento Interação FURB vem sendo realizado anualmente pela FURB desde 2007, no mês de setembro (ANGELI, 2010). O Interação FURB é um programa que proporciona aos estudantes de ensino médio a oportunidade de conhecerem toda a estrutura da Universidade, participando de oficinas organizadas pelos cursos de graduação, intercâmbio, cursos de curta duração e entre outras atividades, no intuito de auxiliá-los na escolha profissional (UNIVERSIDADE REGIONAL DE BLUMENAU, 2017c).

Apesar do site da FURB possuir um local reservado para o Interação FURB, a localização desta informação é de difícil acesso para pessoas que não estão habituadas a utilizar o site e não conhecem sua estrutura. Ao acessar o site, o usuário não consegue encontrar nada a respeito do Interação FURB porque não é o foco do site e sim mais uma informação no meio de tantas outras que o site dispõe. Com isso, é necessário que o usuário posicione o mouse sobre o item “Relação com a comunidade” ou clique sobre o mesmo para conseguir visualizar o item Interação FURB. Ao acessar com o celular fica ainda pior, porque o usuário não tem a opção de posicionar o mouse sobre o item “Relação com a comunidade”. Ainda, se o usuário não tem conhecimento que este item se refere ao Interação FURB, ele terá que clicar em cada item até encontrar algo sobre o evento. Na Figura 1 é mostrado onde é exibido o item Interação FURB no site da FURB.

Figura 1 – Interação FURB no site da FURB



Fonte: Universidade Regional de Blumenau (2017d).

Na Figura 2 pode-se ver que o site da FURB contém uma breve explicação sobre o Interação FURB. Na sequência é apresentado um link para o site do evento.

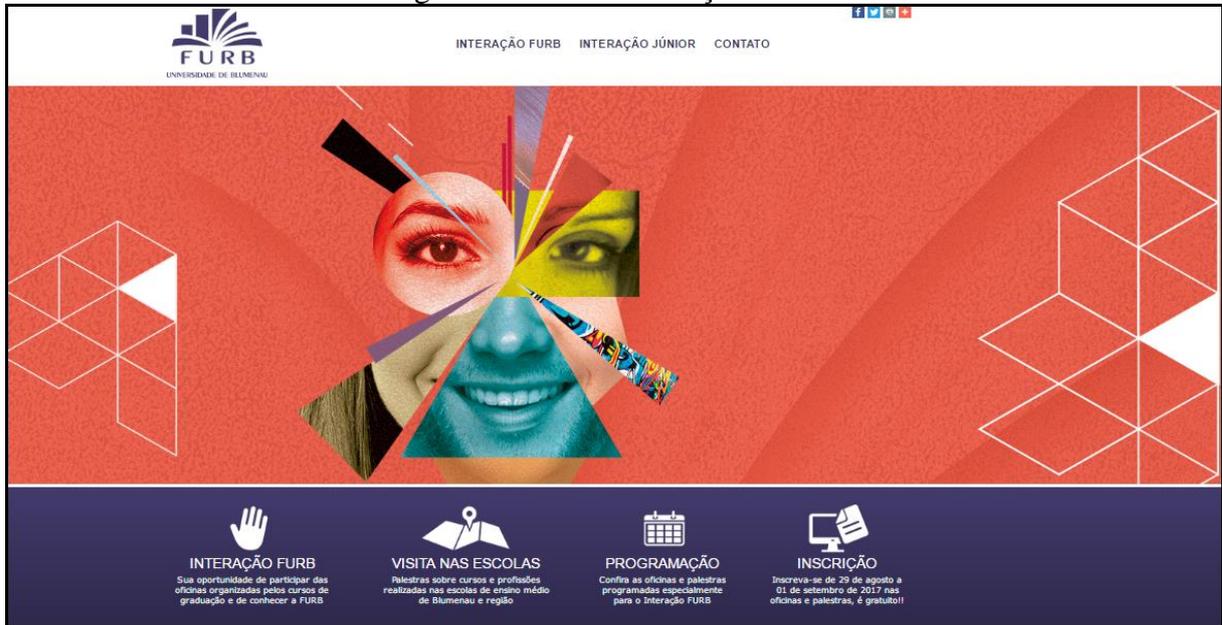
Figura 2 – Espaço no site da FURB reservado ao Interação FURB



Fonte: Universidade Regional de Blumenau (2017b).

O site do Interação FURB pode ser visto na Figura 3. Nele estão contidas informações sobre o evento, como participar, data em que será realizado, informações sobre as inscrições, cronograma do evento, contato, visitas da equipe do Interação FURB nas escolas, entre outras informações relevantes.

Figura 3 – Site do Interação FURB



Fonte: Universidade Regional de Blumenau (2017d).

A equipe do Interação FURB visita escolas de ensino médio de Blumenau e região durante o ano todo para conversar com os estudantes sobre profissões e apresentar a Universidade. Nestas visitas são abordadas informações sobre os cursos de graduação da FURB, profissões e área de atuação, mercado de trabalho, como ingressar na Universidade, programas de apoio financeiro para se manter estudando, possibilidades de intercâmbio acadêmico, definição de “Universidade” e importância da promoção do ensino, pesquisa e extensão, entre outras informações. No fim da palestra, os estudantes recebem a revista Escolha Certa, que traz informações detalhadas sobre todos os cursos de graduação da FURB (UNIVERSIDADE REGIONAL DE BLUMENAU, 2017c).

Em 2016 o Interação FURB recebeu mais de 7.000 mil estudantes do ensino médio, vindos de diversas cidades da região. Esses estudantes visitaram todos os quatro câmpus da Universidade, conhecendo os cursos, projetos e toda estrutura da FURB (CCM/FURB, 2016).

2.2 EXPERIÊNCIA DO USUÁRIO

A ISO 9241-210 define a experiência do usuário (User Experience - UX) como sendo as respostas e percepções de uma pessoa resultantes do uso de um produto, sistema ou serviço (CAELUM, 2016). O tema experiência do usuário é bastante subjetivo, mas é possível

aprender como desenhar um produto, serviço ou ambiente que proporcione uma experiência satisfatória para alguém que os use, identificando todos os aspectos da interação do usuário com esse produto, serviço ou ambiente (CAELUM, 2016).

Segundo Ayres (2016), existem seis fatores essenciais que devem ser levados em consideração para uma boa experiência do usuário: arquitetura da informação, estudar o comportamento e as necessidades do usuário para determinar um caminho que ele deve seguir no uso da aplicação; design de interação, facilitar e incentivar a interação entre usuário e a aplicação; design visual, criar uma identidade visual coerente que facilite a leitura, a compreensão das mensagens e as utilidades disponíveis na aplicação; estratégia de conteúdo, somente publicar conteúdos que tenham algum motivo para serem publicados e que estejam bem construídos; pesquisa com usuários, procurar entender o comportamento do usuário perante a aplicação; usabilidade, a aplicação deve ser funcional, intuitiva e prazerosa de usar.

Antigamente os sites eram feitos para um usuário que estava sentado na frente de um computador, com uma tela grande, um teclado e um mouse. Porém, com o uso dos dispositivos móveis com acesso à internet tudo isso mudou, pois um usuário pode estar acessando um site ou utilizando um aplicativo em qualquer lugar. Com isso se torna fundamental que no desenvolvimento de um site ou aplicativo sejam levadas em consideração todo o contexto ao qual a aplicação pode vir a ser utilizada (GOOGLE, 2015).

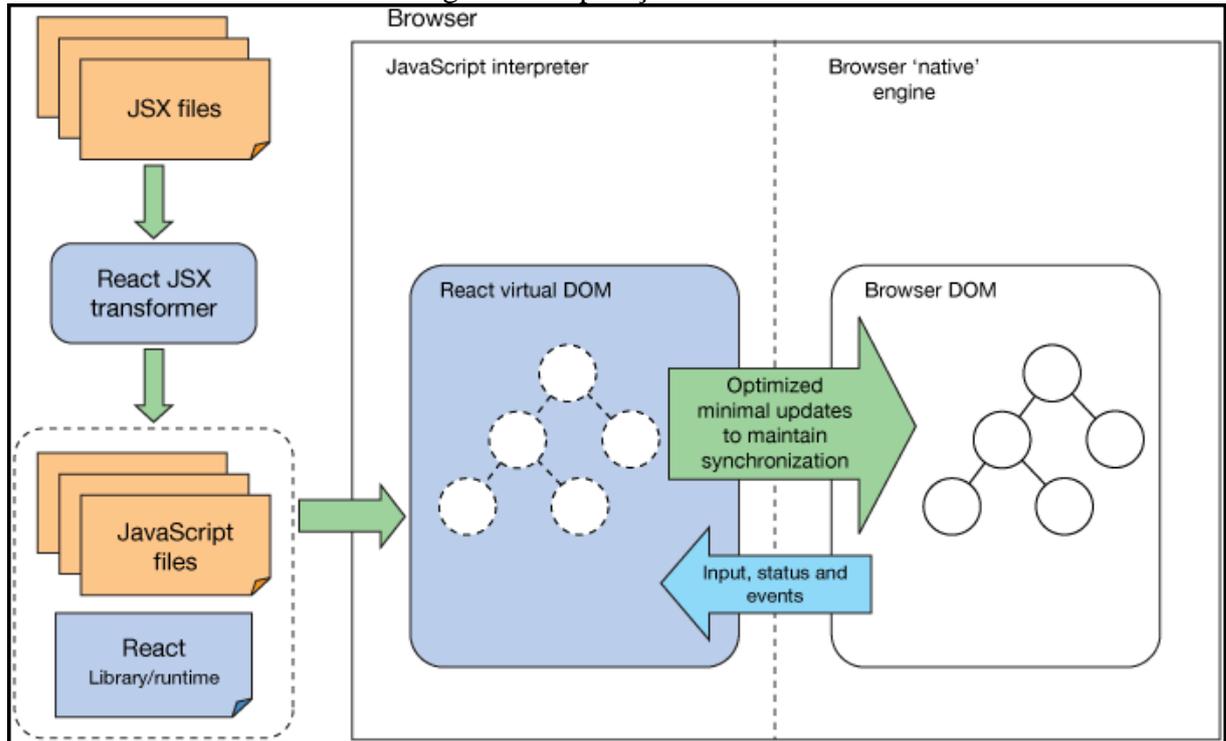
Atualmente na criação de interfaces é bastante adotado um conceito desenvolvido em 2014 pela Google, chamado Material Design. O motivo de ter a palavra Material no nome é porque este conceito busca trazer uma interface inspirada em experiências no mundo real. Desta forma, o conhecimento é transferido ao usuário por meio de experiências compartilhadas em sua vivência, fazendo com que ele consiga entender facilmente a função de cada elemento mesmo sem ter visto antes. Para criar esta semelhança com o mundo real, o conceito Material Design explora cores com alto contraste para dar destaque a ações relevantes, animações para revelar conteúdos mostrando os seus significados e também sombreamento que faz com que se entenda fácil o nível dos planos da interface (SERPA, 2016).

2.3 REACT NATIVE

Segundo Cabral (2016), React Native é um projeto desenvolvido pelos engenheiros do Facebook e que consiste em uma série de ferramentas que viabilizam a criação de aplicações móveis nativas para a plataforma iOS e Android, utilizando a linguagem de programação web JavaScript juntamente com o *framework* React.

React é uma biblioteca JavaScript com um compilador JavaScript XML (JSX) e as ferramentas do desenvolvedor associadas. O React facilita a criação de componentes de visualização de User Interface (UI) reutilizáveis, de alto desempenho, que podem ser compostos formando interfaces modernas. Para otimizar o desempenho do tempo de execução, os componentes React são primeiro renderizados em um Document Object Model (DOM) virtual gerenciado, como é mostrado na Figura 4 (LI, 2016).

Figura 4 – Operações do React



Fonte: Li (2016).

Segundo Li (2016), a especificação da hierarquia dos componentes de interface é feita de forma declarativa, ou seja, o desenvolvedor fornece uma descrição de como a UI deve aparecer e se comportar, mas no momento da execução o mecanismo de tempo de execução determina a maneira mais otimizada de mostrá-la. Ao carregar as informações da interface, elas são enviadas ao DOM virtual do React, depois disso, o React cuidará da sincronização da interface com o DOM real do navegador, no momento apropriado. A implementação do DOM virtual do React é autocontida e não depende de um navegador. Ela também pode ser usada para renderização do lado do servidor. O DOM virtual executa uma comparação otimizada de seu estado interno com relação aos DOMs dos navegadores e executa as atualizações mínimas necessárias para manter a UI consistente. Essa abordagem de modo retido elimina uma grande variedade de problemas de desempenho associados à modificação direta de elementos DOM do navegador, com uma transição visual de cada vez.

Apesar do React Native utilizar a linguagem de programação web JavaScript juntamente com o *framework* React, os aplicativos móveis criados com React Native são diferentes de aplicativos móveis web, aplicativos HTML5 e aplicativos híbridos, porque eles são verdadeiramente nativos assim como os aplicativos criados em Objective-C e Java (FACEBOOK, 2017).

Aplicativos móveis nativos proporcionam experiências melhores nas plataformas móveis. Algumas das vantagens se dá pelo fácil acesso a componentes específicos de cada plataforma. A utilização desses componentes faz com que a interface do aplicativo esteja em perfeita harmonia com relação a interface do próprio sistema. Ao ter uma atualização do sistema, a interface do aplicativo também será atualizada, pois utiliza componentes do sistema ao invés de criar um novo componente igual ao componente do sistema, assim como é feito em aplicativos web (OCCHINO, 2015). Outra vantagem de aplicativos nativos é com relação a performance, porque na web não existe um recurso de thread sofisticado que permite paralelizar o processamento, até podem ser utilizados alguns recursos que permitem que o processamento seja realizado em segundo plano, mas ainda assim não é possível executar cálculos que exigem um maior poder computacional fora do segmento principal do navegador (OCCHINO, 2015).

2.4 TRABALHOS CORRELATOS

Neste capítulo são apresentados três trabalhos com características semelhantes aos principais objetivos do trabalho desenvolvido. A seção 2.4.1 apresenta um aplicativo em Android que possibilita a visualização de informações e o acompanhamento da grade curricular dos cursos ofertados pela FURB (BARRETO, 2016). A seção 2.4.2 aborda um aplicativo multiplataforma que auxilia na locomoção dos visitantes do evento Interação FURB pelo campus da Universidade (ROCHA, 2016). Por fim, a seção 2.4.3 detalha um aplicativo para iPhone que disponibiliza alguns recursos do Ambiente Virtual de Aprendizagem (AVA) (SAMPARA, 2010).

2.4.1 FURB MOBILE: UMA APLICAÇÃO PARA VISUALIZAÇÃO E ACOMPANHAMENTO DA MATRIZ CURRICULAR

Barreto (2016) desenvolveu um aplicativo Android que possibilita o acompanhamento da grade curricular e a visualização de informações dos cursos ofertados pela FURB. Abrindo o aplicativo, o usuário se depara com duas possibilidades: realizar o *login* e visualizar informações acadêmicas da FURB ou acessar o aplicativo como um visitante.

O aplicativo oferece aos acadêmicos a possibilidade de visualizar o andamento do seu curso, status, pré-requisitos das disciplinas e os seus respectivos horários de aula. A Figura 5 apresenta as telas disponíveis aos acadêmicos, sendo elas respectivamente: tela de *login*, visualização da grade curricular, disciplinas com pré-requisitos e horários de aula.

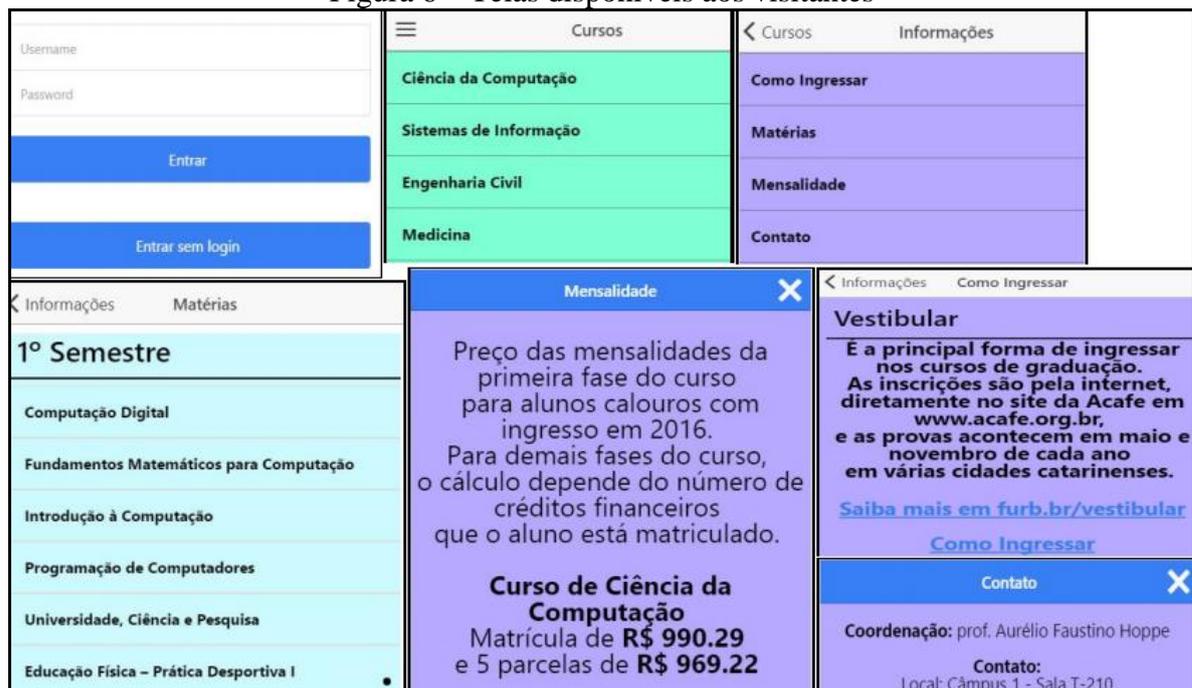
Figura 5 – Telas disponíveis aos acadêmicos



Fonte: Barreto (2016).

Aos visitantes, o aplicativo possibilita visualizar todos os cursos ofertados e suas disciplinas, informações de como ingressar, contato e valor da mensalidade. A Figura 6 apresenta as telas disponíveis aos visitantes, sendo elas: tela de *login*, cursos, informações do curso, grade curricular, mensalidade do curso, como ingressar e contato.

Figura 6 – Telas disponíveis aos visitantes



Fonte: Barreto (2016).

O aplicativo desenvolvido por Barreto (2016) possui uma arquitetura cliente-servidor. O cliente foi desenvolvido utilizando o *framework* Ionic, que possibilitou o uso de JavaScript, HTML5, AngularJS e CSS; e o servidor foi desenvolvido utilizando a linguagem de programação C# juntamente com o *framework* de serviços Web API.

Barreto (2016) relata que os resultados obtidos foram satisfatórios e que o aplicativo conseguiu contemplar em sua totalidade aquilo que havia sido proposto, uma vez que o aplicativo conseguiu montar uma grade curricular para a visualização das disciplinas divididas por seus respectivos semestres e mostrar os pré-requisitos de cada disciplina. Já para pessoas sem vínculo acadêmico com a FURB, possibilitou a visualização dos cursos e outras informações adicionais.

2.4.2 FURB-MOBILE: SISTEMA MÓVEL MULTIPLATAFORMA PARA NAVEGAÇÃO EM ROTAS INTERNAS

O trabalho proposto por Rocha (2016) consiste em um aplicativo multiplataforma para dispositivos móveis que permite ao usuário (aluno ou participante do Interação FURB) locomover-se até um ponto de interesse por meio de algumas rotas (normais ou com acessibilidade) sugeridas pelo aplicativo, assim como, visualizar sua posição atual em relação ao ambiente virtual.

Ao iniciar o aplicativo, o usuário é remetido à tela principal da aplicação, onde o mesmo pode visualizar uma planta de um determinado pavimento com vários vértices vermelhos. Para iniciar a navegação é necessário que o usuário defina um vértice que será a origem e outro para ser o destino, essa configuração é feita pressionando um vértice e selecionando a opção “marcar como origem” ou “marcar como destino”. Com a origem e o destino informados, basta o usuário acionar o botão “rota” que o aplicativo irá traçar uma linha pontilhada entre a origem e o destino, informando qual a rota de menor custo para o usuário. A Figura 7 mostra a tela principal, seguida pelo momento em que o usuário pressiona um vértice para configurá-lo como origem ou destino e na sequência, pode-se perceber a rota de menor custo traçada pelo aplicativo.

Figura 7 – Telas de navegação



Fonte: Rocha (2016).

Rocha (2016) desenvolveu o aplicativo utilizando uma arquitetura cliente-servidor, onde a parte de cliente foi desenvolvida utilizando o *framework* Ionic. Já a parte do servidor foi desenvolvida com Java, utilizando o *framework* RestEasy que permite, através de notações no código fonte, mapear as requisições web para as funções correspondentes.

Segundo Rocha (2016), o objetivo de desenvolver um aplicativo para dispositivo móvel que possibilitasse a apresentação de uma rota virtual para auxiliar a movimentação do usuário no mundo real foi atendido. Porém, dentre os objetivos específicos foi citado o uso do georreferenciamento para apresentar a localização do usuário, onde a intenção era apresentar ao usuário sua posição real em relação ao mapa interno dos blocos, ou ao menos, o local mais próximo de sua posição. Mas, devido a questões de tempo, este objetivo foi parcialmente atendido fazendo com que o aplicativo apresente ao usuário em qual bloco da universidade o mesmo se encontra.

2.4.3 IAVA – APLICAÇÃO WEB PARA DISPONIBILIZAR RECURSOS DO AVA NO DISPOSITIVO MÓVEL IPHONE

Elaborada por Sampara (2010), a aplicação iAVA é um aplicativo que contém alguns recursos do AVA. Foi desenvolvido para ser acessado por dispositivos móveis da plataforma iOS, no caso o iPhone.

Ao acessar o aplicativo, é apresentada a tela de *login*. Após realizar o *login*, o usuário tem acesso as funcionalidades disponíveis no iAVA. Dentre essas funcionalidades disponibilizadas, o aluno tem a possibilidade de visualizar a sua Área de Estudo, onde tem-se a lista de disciplinas as quais o usuário tem acesso. Ao acessar uma disciplina, são apresentados todos os recursos vinculados a ela. Os recursos de cada disciplina são personalizáveis pelos administradores, sendo assim, um administrador pode a qualquer momento acessar o aplicativo com sua conta e adicionar à disciplina um Fórum, uma Pasta, um Material ou um Texto Colaborativo, por exemplo. A Figura 8 mostra algumas telas do iAVA, sendo elas respectivamente: tela de *login*, visualização de disciplinas na área de estudo, visualização de recursos de uma disciplina, adicionar recursos em uma disciplina.

Figura 8 – Telas do aplicativo iAVA



Fonte: Sampara (2010).

Sampara (2010) utilizou no desenvolvimento da aplicação as linguagens de programação Java, JSP, JavaScript e CSS. Para criação da interface e estilos foram utilizadas as ferramentas DashCode e Interface Builder. A aplicação foi implementada utilizando uma arquitetura cliente-servidor, onde o servidor é um Web Service em Java, que recebe requisições do aplicativo e retorna as informações solicitadas.

No desenvolvimento do aplicativo, Sampara (2010) encontrou algumas dificuldades que precisou contornar, sendo uma delas a integração com o AVA que não dispõe de um Web Service. Com isso, surgiu a necessidade da criação de um Web Service para a realização de testes no aplicativo. Outra dificuldade encontrada foi na criação do recurso “arquivos” do usuário, porque o iPhone não suporta o componente HyperText Markup Language (HTML) responsável pela realização do envio de arquivos. Então, foi necessário alterar este recurso de maneira que o usuário passou a ter que informar o endereço de um arquivo disponível na internet para que o sistema baixasse uma cópia do mesmo. Porém, mesmo com as

dificuldades encontradas foram implementados os recursos para criar pasta, texto colaborativo, arquivos do usuário e fórum temático.

Como resultado final, Sampara (2010) concluiu que, mesmo sem perda de dados, uma interface voltada para o iPhone se torna muito diferente de uma interface para desktop, com tipos de elementos e estruturas diferentes. Assim, para atender todos os requisitos e expectativas que cada tipo de usuário deseja, as aplicações devem ser construídas separadamente, ou seja, uma aplicação para desktop e outra para iPhone.

3 DESENVOLVIMENTO

Neste capítulo são apresentadas as etapas de desenvolvimento da solução proposta. Na seção 3.1 são expostos os requisitos da solução. A seção 3.2 é dedicada à especificação do projeto, contendo casos de uso, apresentação da arquitetura e diagramas de classes. Na seção 3.3 é exibido um detalhamento das principais funcionalidades. Por fim, a seção 3.4 apresenta os experimentos realizados e resultados obtidos.

3.1 REQUISITOS

A solução desenvolvida pode ser separada em duas partes: aplicativo móvel e o serviço para importação de informações do site da FURB, do site do Interação FURB e de arquivos Excel colocados no servidor pelo administrador do sistema. Dessa forma, para melhor entendimento, os requisitos foram separados levando em consideração essas duas partes. Os Requisitos Funcionais (RF) e Requisitos Não Funcionais (RNF) do aplicativo móvel são apresentados no Quadro 1 e Quadro 2 respectivamente. Os requisitos estão relacionados com os casos de uso apresentados na Figura 9.

Quadro 1 – Requisitos Funcionais do aplicativo móvel

Requisitos Funcionais (RF)	Casos de uso (UC)
RF 01: permitir que o usuário possa visualizar informações sobre os cursos oferecidos pela FURB, tais como apresentação, valor da mensalidade, intercâmbio, diferenciais do curso, projetos de iniciação científica e extensão, atividades extracurriculares, locais de trabalho e curiosidades.	UC02
RF 02: permitir que o usuário possa visualizar informações sobre as oficinas do Interação FURB, tais como resumo, horário e local	UC07
RF 03: permitir que o usuário cadastrado no Interação FURB possa realizar <i>login</i>	UC11
RF 04: permitir que o usuário possa realizar <i>check-in</i> nas oficinas do evento Interação FURB apresentando um QRCode	UC12
RF 05: permitir que o usuário possa visualizar o local e o horário de atendimento da praça de atendimento ao estudante	UC01
RF 06: permitir que o usuário possa visualizar quais são as maneiras de ingressar na FURB	UC03
RF 07: permitir que o usuário possa visualizar informações de apoio financeiro, tais como bolsas, financiamento, estágios e monitoria	UC04
RF 08: permitir que o usuário possa visualizar quais são os procedimentos a serem seguidos para se matricular	UC05
RF 09: permitir que o usuário possa visualizar informações sobre o evento Interação FURB, tais como resumo, como participar, data, horário e local	UC06
RF 10: permitir que o usuário possa visualizar quais oficinas está inscrito	UC13
RF 11: permitir que o ministrante possa visualizar os visitantes que estão inscritos em suas oficinas	UC09
RF 12: permitir que o ministrante da palestra possa realizar <i>login</i>	UC08
RF 13: permitir que o ministrante da palestra possa realizar <i>check-in</i> dos visitantes participantes da sua oficina via QRCode	UC10
RF 14: permitir que o usuário receba notificações do aplicativo previamente cadastradas no servidor	UC14

Fonte: elaborado pelo autor.

Quadro 2 – Requisitos Não Funcionais do aplicativo móvel

Requisitos Não Funcionais (RNF)
RNF 01: rodar na plataforma Android
RNF 02: possuir uma interface simples e de fácil utilização
RNF 03: ser desenvolvido na linguagem JavaScript com <i>framework</i> React Native, utilizando a ferramenta de desenvolvimento Visual Studio Code
RNF 04: se comunicar com o servidor através de requisições REST
RNF 05: possuir uma API desenvolvida em C# com o <i>framework</i> Web API para disponibilizar as informações dos arquivos JSON
RNF 06: permitir o acesso ao aplicativo sem a necessidade de uma conexão com a internet

Fonte: elaborado pelo autor.

Os Requisitos Funcionais e Requisitos Não Funcionais do serviço para importação de informações dos sites e de arquivos Excel são expostos no Quadro 3 e Quadro 4 respectivamente. Os requisitos estão relacionados com os casos de uso apresentados na Figura 10.

Quadro 3 – Requisitos Funcionais do serviço para importação de informações dos sites e de arquivos Excel

Requisitos Funcionais (RF)	Casos de uso (UC)
RF 01: coletar os valores de mensalidades dos cursos disponíveis no site da FURB	UC01
RF 02: coletar as informações das oficinas disponíveis no site do Interação FURB, tais como nome, curso, resumo e horário	UC02
RF 03: coletar informações de local das oficinas disponíveis em arquivos Excel	UC03
RF 04: coletar informações das inscrições dos visitantes disponíveis em arquivos Excel, tais como nome, email, oficina, curso, local e horário	UC04
RF 05: coletar informações dos ministrantes das oficinas disponíveis em arquivos Excel, tais como nome, email, curso, oficina e <i>login</i>	UC05

Fonte: elaborado pelo autor.

Quadro 4 – Requisitos Não Funcionais do serviço para importação de informações dos sites e de arquivos Excel

Requisitos Não Funcionais (RNF)
RNF 01: ser desenvolvido na linguagem C# utilizando a ferramenta de desenvolvimento Visual Studio
RNF 02: armazenar os dados em arquivos JSON

Fonte: elaborada pelo autor.

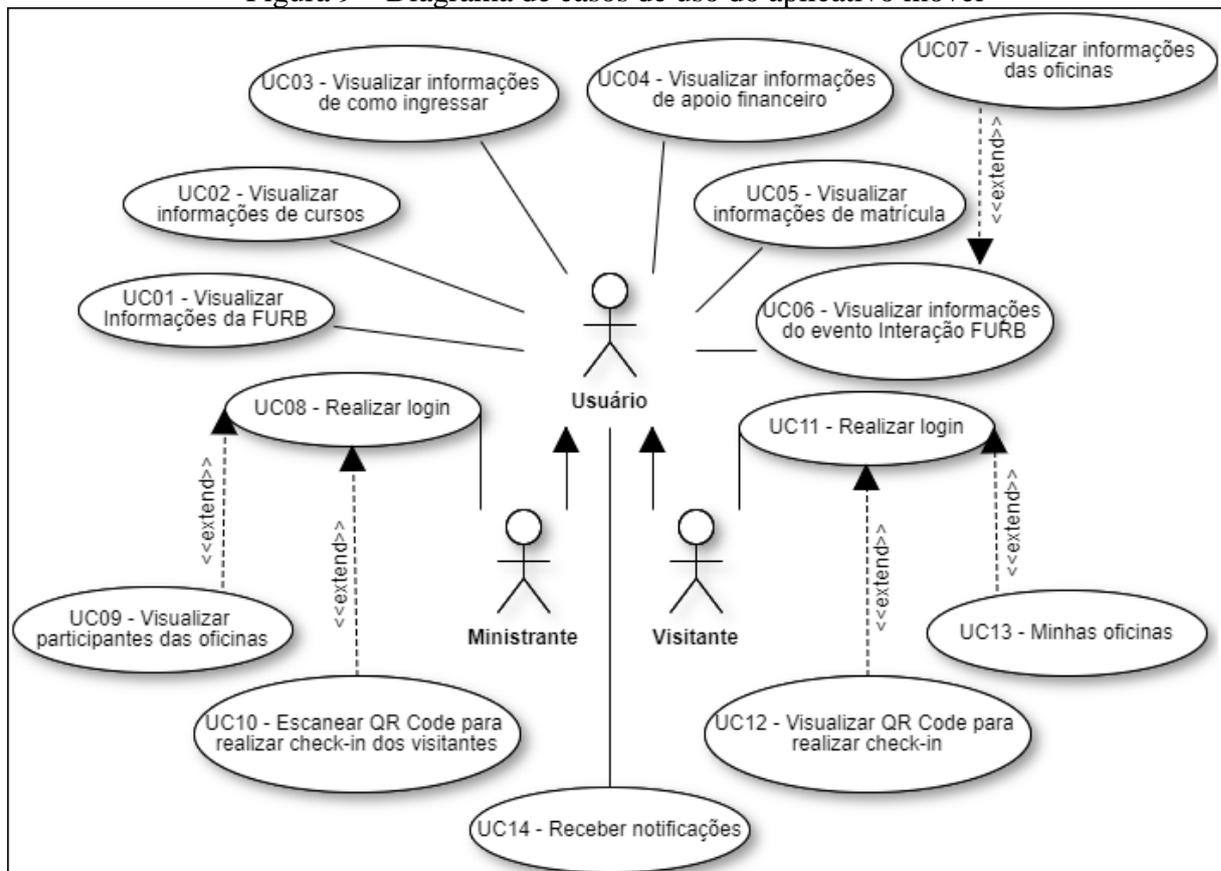
3.2 ESPECIFICAÇÃO

Nesta seção é apresentada a especificação da solução. A seção 3.2.1 apresenta os casos de uso do aplicativo e do serviço. A seção 3.2.2 detalha a arquitetura da solução. Por fim, a seção 3.2.3 expõe os diagramas de classes do serviço e da API que rodam no servidor. Para a realização dos diagramas foi utilizada a ferramenta online Cacao.

3.2.1 Casos de uso

Assim como nos Requisitos Funcionais e Requisitos Não Funcionais, os casos de uso foram separados em duas partes. Na Figura 9 são apresentadas as funcionalidades do aplicativo móvel para os atores Usuário, Ministrante e Visitante.

Figura 9 – Diagrama de casos de uso do aplicativo móvel

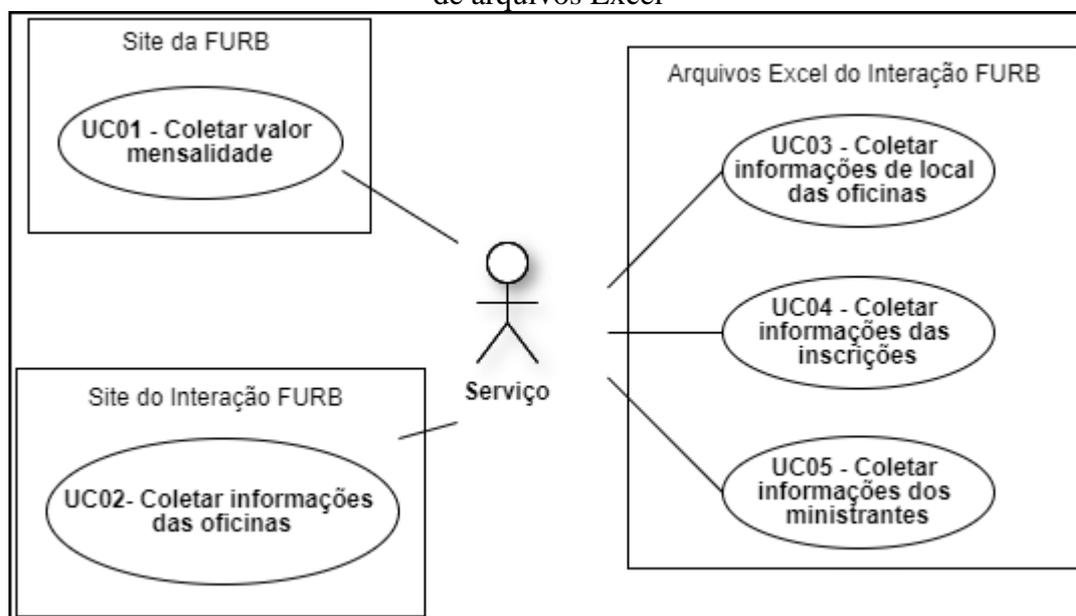


Fonte: elaborado pelo autor.

O caso de uso UC01 - Visualizar informações da FURB permite ao usuário visualizar o local e horário de funcionamento da praça de atendimento ao estudante e também visualizar algumas curiosidades, tais como: o que faz da FURB uma universidade e para que serve o centro de cada curso. O caso de uso UC02 - Visualizar informações de cursos permite ao usuário visualizar resumo do curso, valor da mensalidade, intercâmbio, diferenciais do curso, projetos de iniciação científica e extensão, atividades extracurriculares, locais de trabalho e curiosidades. O caso de uso UC03 - Visualizar informações de como ingressar apresenta os meios pelos quais é possível ingressar na FURB. No caso de uso UC04 - Visualizar informações de apoio financeiro o usuário consegue consultar os tipos de bolsas, financiamentos, estágios e monitorias que estão disponíveis para dar um apoio financeiro ao aluno no pagamento do curso. O caso de uso UC05 - Visualizar informações de matrícula proporciona ao usuário, acesso a um fluxograma com as etapas que devem ser seguidas para a realização da matrícula. No caso de uso UC06 - Visualizar informações do evento Interação FURB o usuário tem acesso a um breve resumo do evento Interação FURB, informações de como participar, local e data da realização do evento. O caso de uso UC07 - Visualizar informações das oficinas permite ao usuário

visualizar o resumo das oficinas do evento Interação FURB, horário e local. O caso de uso UC08 - Realizar login permite ao ministrante realizar o *login* no aplicativo. O caso de uso UC09 - Visualizar participantes das oficinas permite ao ministrante visualizar o nome de todos os visitantes cadastrados em suas oficinas. No caso de uso UC10 - Escanear QR Code para realizar check-in dos visitantes o ministrante consegue escanear o QRCode apresentado pelo visitante e com isso marcar a presença do visitante em sua oficina. O caso de uso UC11 - Realizar login permite ao visitante cadastrado no evento Interação FURB, realizar o *login* no aplicativo. O caso de uso UC12 - Visualizar QR Code para realizar check-in exibe ao visitante o QRCode que deve ser apresentado ao ministrante da oficina para realizar o seu *check-in*. No caso de uso UC13 - Minhas oficinas o usuário consegue visualizar as oficinas nas quais está cadastrado. Através do UC14 - Receber notificações o usuário receberá notificações cadastradas no servidor. Na Figura 10 são apresentadas as funcionalidades do serviço para importação de informações dos sites e de arquivos Excel, utilizando o ator Serviço.

Figura 10 – Diagrama de casos de uso do serviço para importação de informações dos sites e de arquivos Excel



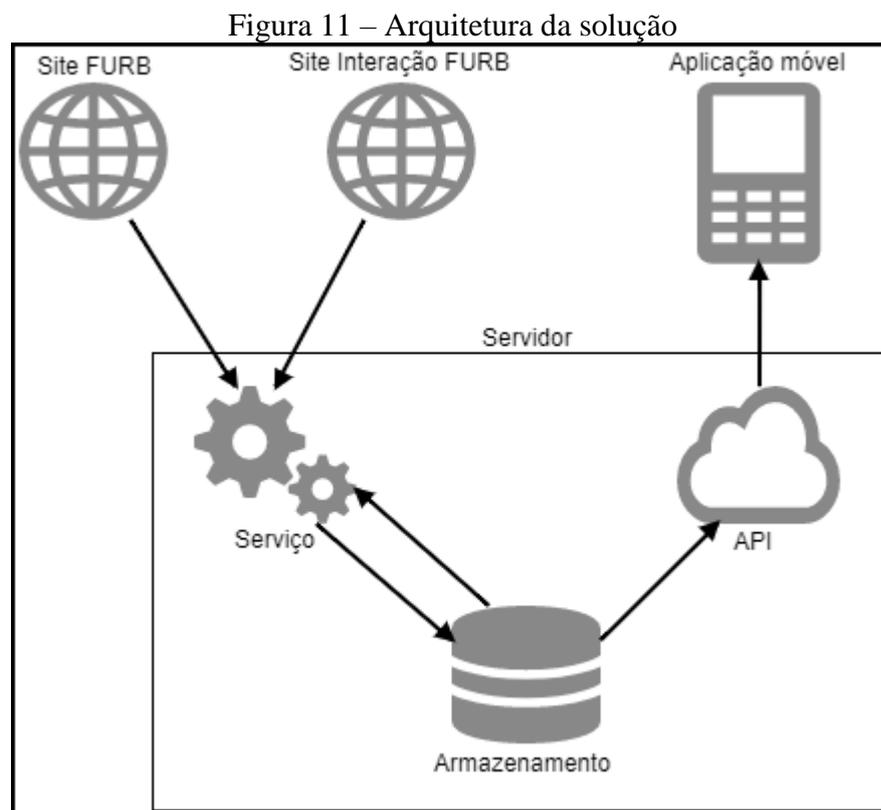
Fonte: elaborado pelo autor.

No caso de uso UC01 - Coletar valor mensalidade o serviço importa o valor da mensalidade diretamente do site da FURB a cada 5 minutos. No caso de uso UC02 - Coletar informações das oficinas o serviço importa as informações de nome, curso, resumo e horário das oficinas diretamente do site do Interação FURB a cada 5 minutos. Cada importação em arquivo Excel espera um arquivo com um nome específico, com isso o serviço fica monitorando uma pasta pré-definida aguardando que sejam colocados os arquivos com

estes nomes. O caso de uso UC03 - Coletar informações de local das oficinas espera um arquivo com o nome “Oficinas” para realizar a importação do local das oficinas. No caso de uso UC04 - Coletar informações das inscrições é aguardado um arquivo com o nome “Inscrições” para que seja realizada a importação das informações das inscrições, bem como nome do inscrito, e-mail, oficina, curso, local e horário. Por fim, o UC05 - Coletar informações dos ministrantes espera por um arquivo com o nome “Ministrantes” para que sejam importadas as informações dos ministrantes, bem como nome, e-mail, curso, oficina e *login*.

3.2.2 Arquitetura

Na Figura 11 é apresentada a arquitetura da solução que foi construída levando em consideração os requisitos e casos de uso elencados nas seções anteriores.



Fonte: elaborado pelo autor.

O serviço é instalado no servidor. Ele é responsável por alimentar a base de armazenamento utilizando três fontes de dados: o site da FURB, o site do evento Interação FURB e a própria base de armazenamento. As importações de informações dos sites são realizadas através de requisições HyperText Transfer Protocol (HTTP) que retornam o código HTML de determinadas páginas de cada site, após a requisição, os dados retornados são processados pelo serviço e salvos no servidor em arquivos no formato JavaScript Object

Notation (JSON). Essas importações são realizadas a cada 5 minutos para que não leve muito tempo para atualizar a base de armazenamento, caso seja realizada alguma alteração nas informações disponibilizadas em cada site. A importação realizada na própria base de armazenamento é feita em arquivos Excel que são colocados no servidor pelo administrador do sistema. O serviço fica monitorando uma pasta aguardando até que sejam colocados os arquivos Excel, quando os arquivos são colocados nesta pasta é iniciada a importação, após realizar o processamento dos dados de cada planilha, o serviço salva os dados em arquivos no formato JSON.

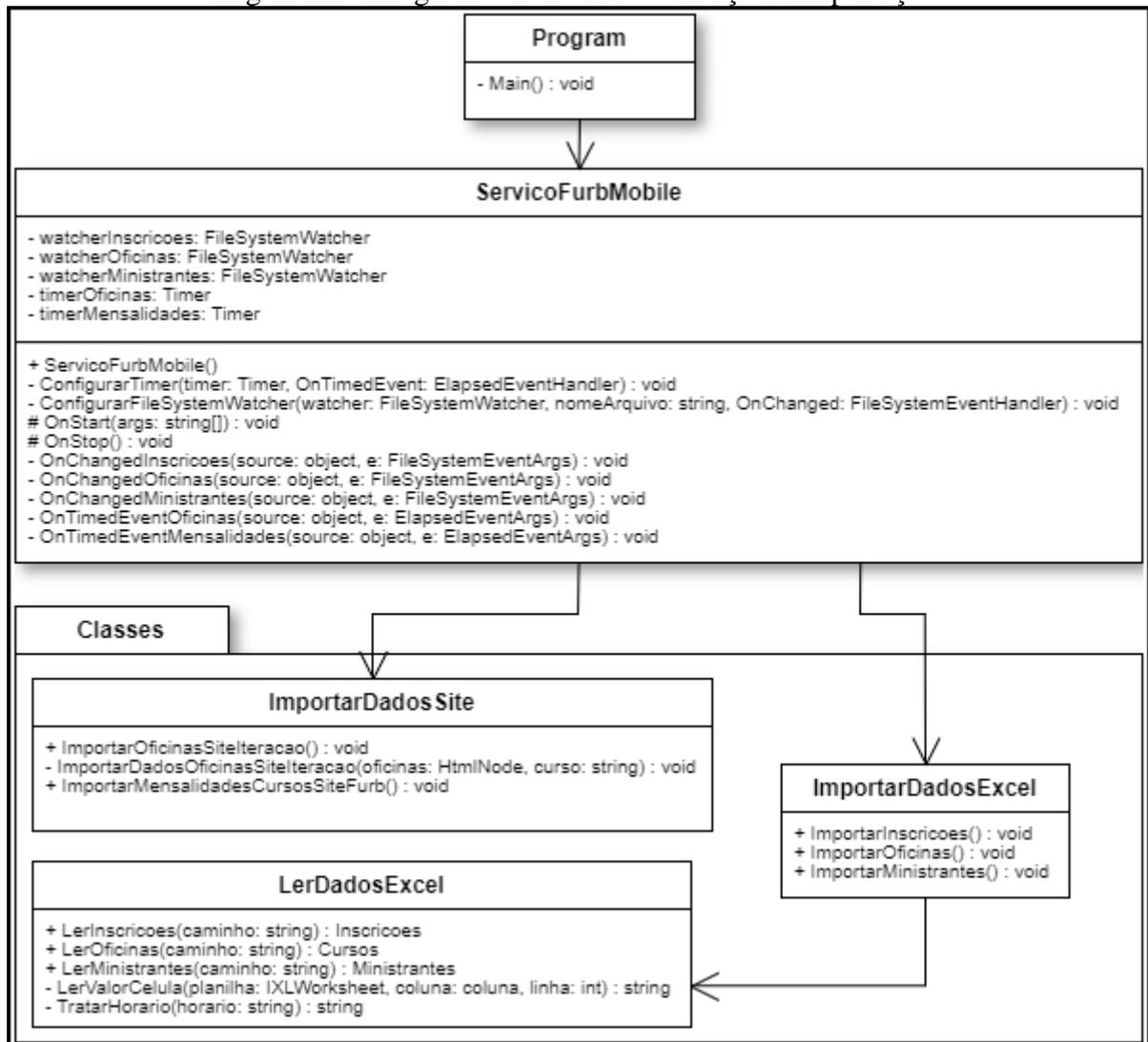
O aplicativo móvel exibe as informações que estão salvas nos arquivos JSON através de uma Application Programming Interface (API) disponibilizada no servidor. A API realiza a leitura dos arquivos JSON, processa as informações e as retorna para o aplicativo móvel.

3.2.3 Diagramas de classes

As aplicações que rodam no servidor foram separadas em três projetos levando em consideração a arquitetura e as aplicações existentes. O primeiro projeto diz respeito ao serviço de importação, o segundo projeto é a Web API que disponibiliza as informações para o aplicativo e o último projeto é uma biblioteca de classes que são utilizadas no serviço e na API. A seguir são apresentados os diagramas de classe de cada projeto.

A Figura 12 demonstra o diagrama de classes do serviço responsável pela importação dos dados do site da FURB, do Interação e dos arquivos Excel. A classe `Program` é a primeira a ser chamada, a função dessa classe é ativar o serviço. A classe `ServicoFurbMobile` é responsável por administrar os eventos do serviço (início e parada) e chamar os métodos de importação quando necessário. Nela são realizados os controles de tempo para execução das importações nos sites e também o monitoramento da pasta onde são colocados os arquivos Excel para realizar a importação. A classe `ImportarDadosSite` possui os métodos para fazer a importação dos dados do site da FURB e do Interação. A classe `ImportarDadosExcel` possui os métodos para fazer a importação dos dados dos arquivos Excel utilizando a classe `LerDadosExcel` para realizar a leitura de cada arquivo.

Figura 12 – Diagrama de classes do serviço de importação

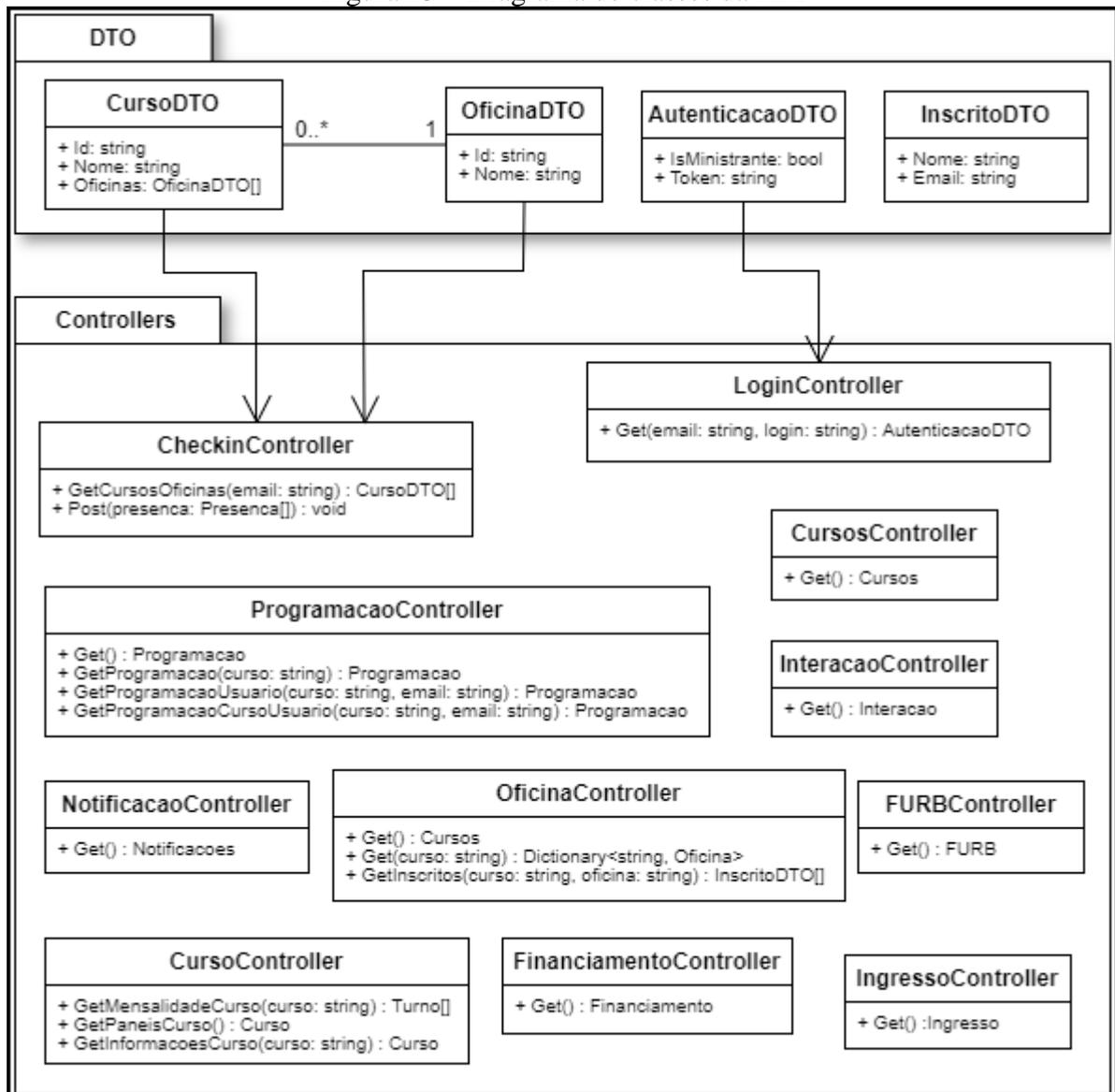


Fonte: elaborado pelo autor.

A Figura 13 demonstra o diagrama de classes da API responsável por ler os arquivos JSON, processar os dados e retorná-los para o aplicativo móvel. As classes do pacote `DTO` são utilizadas como modelos de objetos que serão retornados para a aplicação. As classes `CursoDTO` e `OficinaDTO` são utilizadas para retornar os cursos e oficinas do ministrante na tela de *check-in* do aplicativo móvel. A classe `AutenticacaoDTO` é utilizada no *login* do sistema para retornar o *token* do usuário e informar se o usuário é um ministrante ou um visitante. A classe `InscritoDTO` é utilizada para retornar a lista de usuários inscritos na oficina do ministrante. As classes do pacote `Controllers` processam as requisições feitas pelo aplicativo móvel, cada classe processa e retorna os dados de uma parte específica do aplicativo. A classe `CheckinController` retorna os cursos e oficinas do ministrante na tela de *check-in* e também realiza a gravação do *check-in* dos visitantes. Na classe `LoginController` é realizada a validação do *login* do usuário. Os dados relacionados ao evento Interação FURB,

bem como informações do evento, programação do evento e informações das oficinas, são retornados pelas classes `InteracaoController`, `ProgramacaoController` e `OficinaController`, respectivamente. A classe `CursosController` é responsável por retornar uma lista de cursos ofertados pela FURB, enquanto a classe `CursoController` retorna as informações de um curso específico. As notificações que serão exibidas aos usuários do aplicativo são retornadas pela classe `NotificacaoController`. A classe `FURBController` retorna informações sobre a FURB. Os tipos de financiamentos oferecidos pela universidade são retornados pela classe `FinanciamentoController`. Por fim, a classe `IngressoController` é responsável por retornar informações sobre como ingressar na FURB.

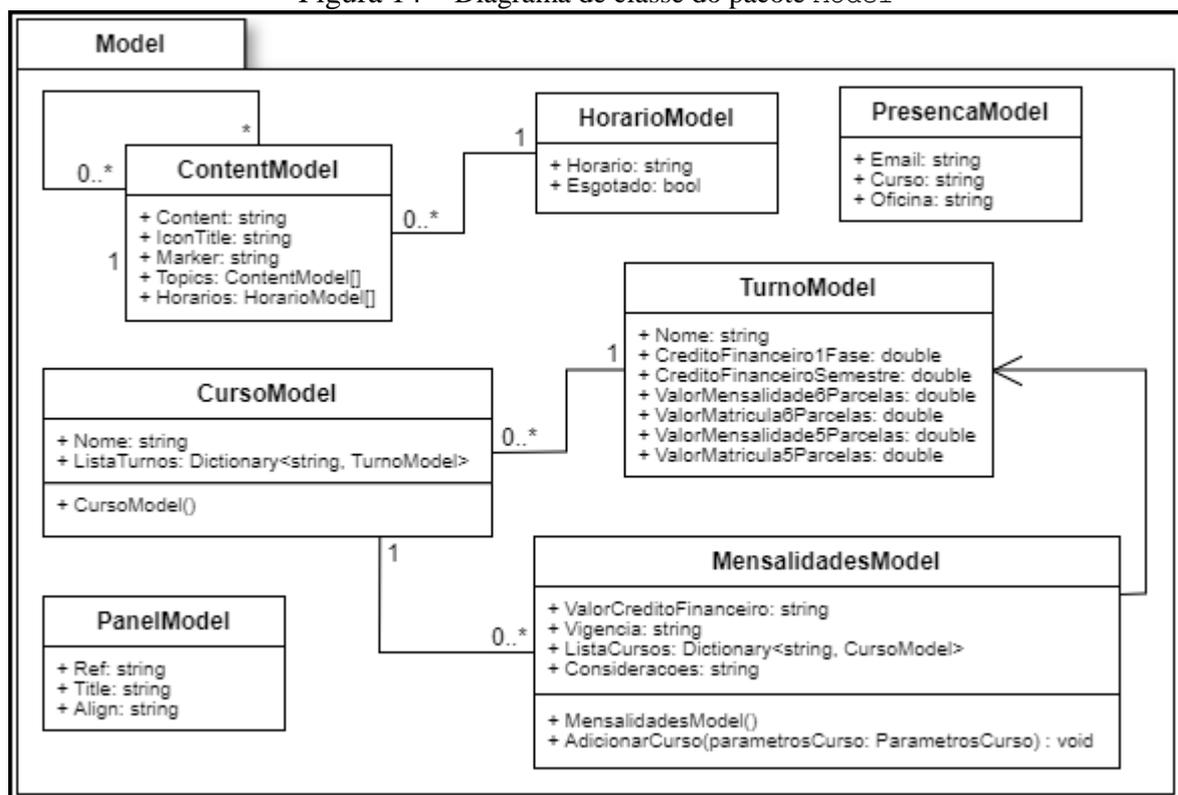
Figura 13 – Diagrama de classes da API



Fonte: elaborado pelo autor.

O diagrama de classes da biblioteca criada com classes de modelos de objetos para serem utilizados no serviço e/ou na API foi separado em quatro partes devido a sua complexidade e quantidade de classes. A Figura 14 demonstra as classes contidas no pacote `Model`, essas classes são utilizadas para modelar todas as informações exibidas no aplicativo. A classe `ContentModel` é utilizada para montar os conteúdos exibidos no aplicativo. Os conteúdos com as informações de oficinas do Interação FURB possuem os horários da oficina utilizando a classe `HorarioModel`. As informações de *check-in* dos visitantes são retornadas utilizando a classe `PresencaModel`. A classe `TurnoModel` é utilizada para montar as informações de valor de cada turno de um curso. A classe `CursoModel` é utilizada para agrupar os turnos no seu devido curso. A classe `MensalidadesModel` é utilizada para concentrar as informações de mensalidades de todos os cursos e informações gerais da mensalidade que se aplicam a todos os cursos. A classe `PanelModel` é utilizada para as configurações dos painéis que são exibidos no aplicativo.

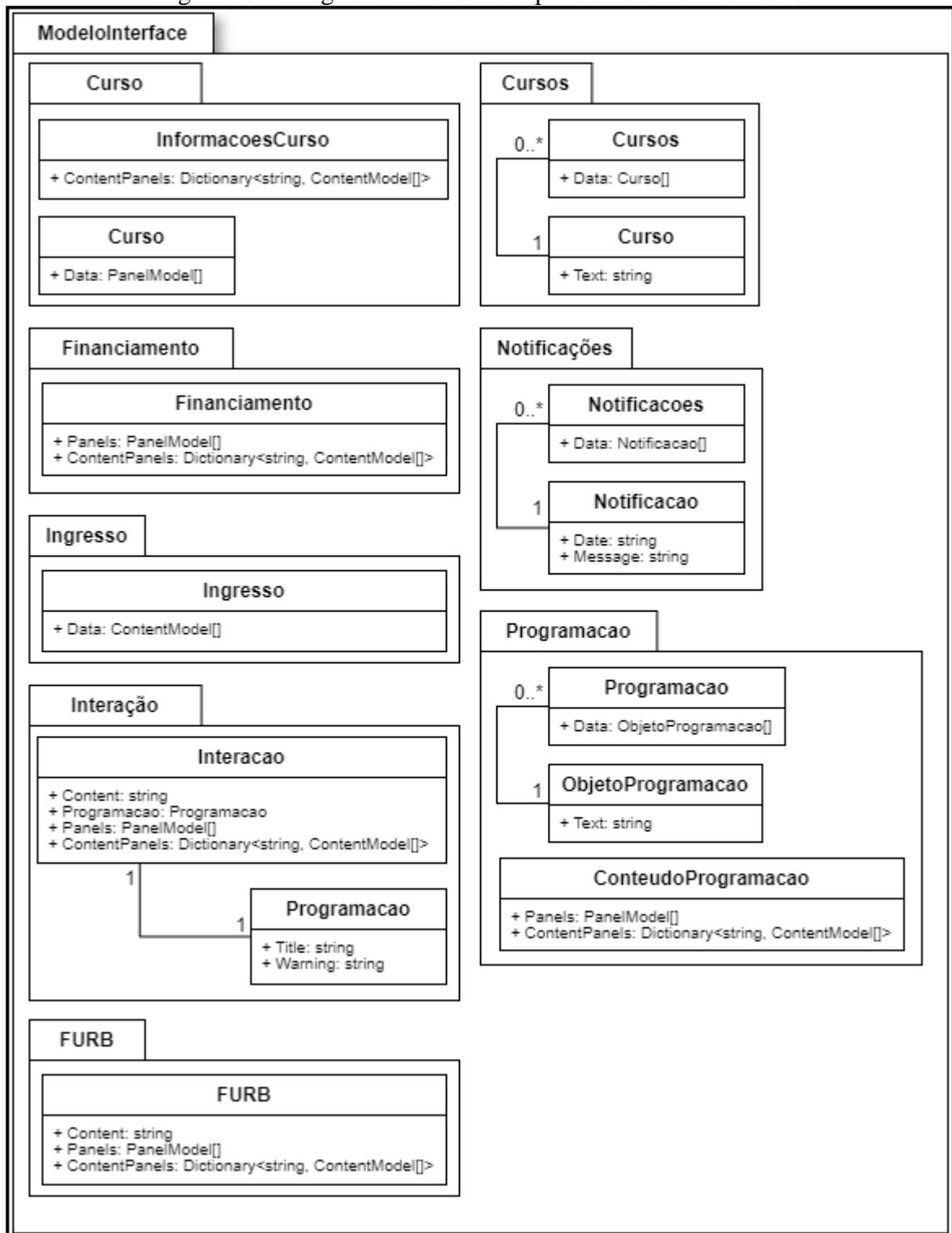
Figura 14 – Diagrama de classe do pacote `Model`



Fonte: elaborado pelo autor.

A Figura 15 demonstra a segunda parte do diagrama de classes que apresenta as classes contidas no pacote `ModeloInterface`. É possível visualizar que dentro do pacote principal existem outros pacotes, cada pacote contém classes para montar conteúdos para uma ou mais telas do aplicativo. O pacote `Curso` contém classes que são utilizadas para preparar as

informações que são exibidas na visualização de um curso. As informações dos cursos ficam separadas em painéis, esses painéis são agrupados na classe `Curso`, enquanto o conteúdo de cada painel é configurado na classe `InformacoesCurso`. As classes do pacote `Cursos` são responsáveis por montar conteúdos para as telas de listagem de cursos, tanto a tela que exibe somente as letras iniciais dos cursos, quanto a listagem que exibe o nome dos cursos ao acessar uma letra. A classe `Cursos` é utilizada para reunir as classes `Curso` que guardam o nome de cada curso. O pacote `Financiamento` é composto apenas pela classe `Financiamento` que armazena quais são os painéis e os conteúdos exibidos na tela de apoio financeiro. O conteúdo exibido na tela de ingresso é organizado utilizando a classe `Ingresso` do pacote `Ingresso`. As notificações exibidas pelo aplicativo são configuradas utilizando as classes do pacote `Notificações`, a classe `Notificacoes` concentra todas as notificações que são montadas na classe `Notificacao`. As informações exibidas na tela do Interação FURB são configuradas utilizando as classes do pacote `Interação`. A classe `Programacao` contém o link para redirecionar para a programação do evento e também o alerta sobre a possibilidade de mudança da programação. As informações sobre o Interação FURB, os painéis e os conteúdos dos painéis são montados pela classe `Interacao`. A classe `FURB` do pacote `FURB` monta o conteúdo sobre a FURB, bem como os painéis exibidos e os conteúdos de cada painel. A parte de programação do Interação FURB é organizada utilizando as classes do pacote `Programacao`. A classe `ObjetoProgramacao` é utilizada para retornar o nome do curso e a classe `Programacao` é utilizada para agrupar todos os nomes dos cursos. A classe `ConteudoProgramacao` é responsável por montar quais são os painéis exibidos e os conteúdos de cada painel.

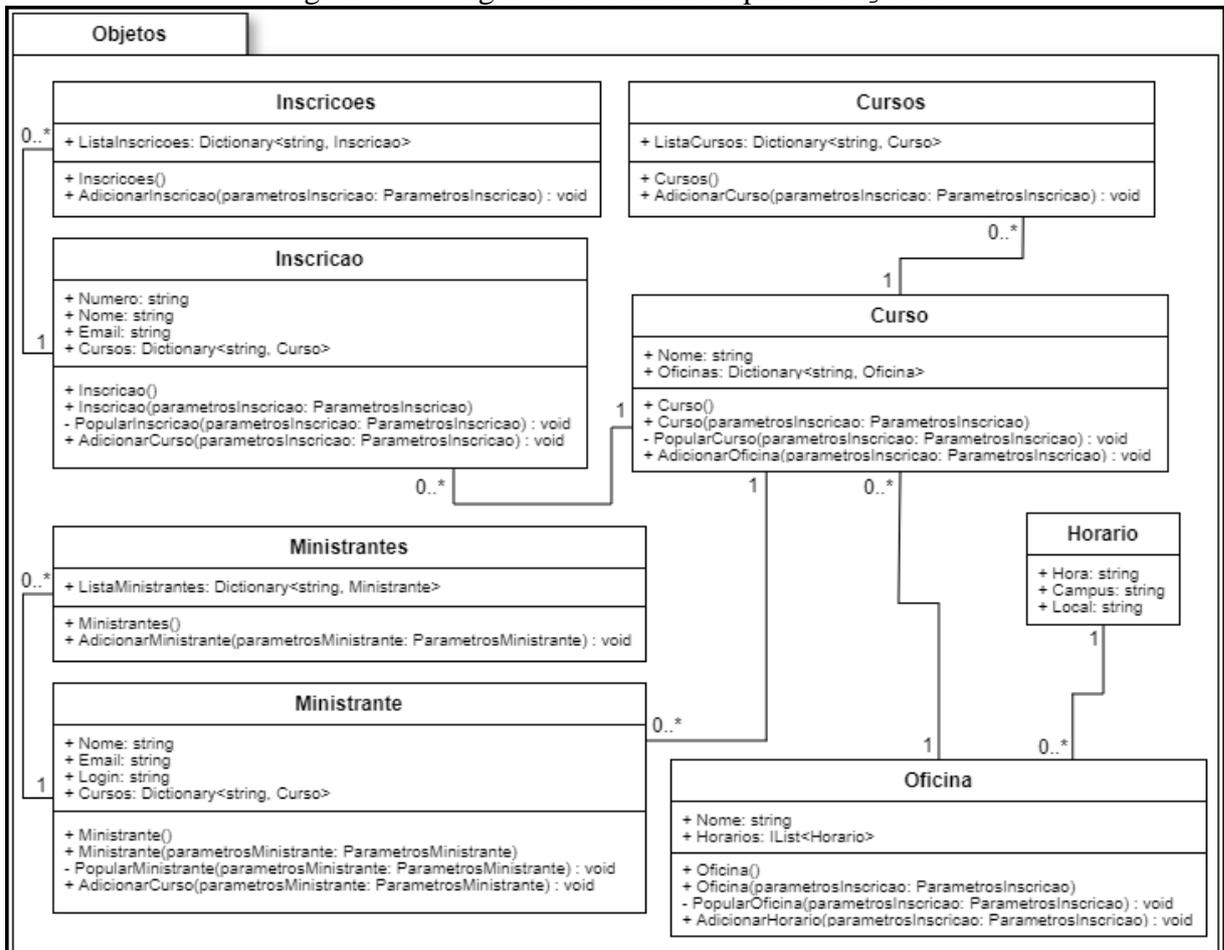
Figura 15 – Diagrama de classes do pacote `ModeloInterface`

Fonte: elaborado pelo autor.

A Figura 16 demonstra a terceira parte do diagrama de classes que apresenta as classes contidas no pacote `Objetos`. A classe `Inscricoes` agrupa as informações de todas as inscrições que são configuradas na classe `Inscricao`. As informações de um ministrante do Interação FURB são organizadas utilizando a classe `Ministrante`, que por sua vez é

agrupada na classe `Ministrantes`. A classe `Oficina` é utilizada para montar as informações de uma oficina e agrupar os horários da oficina que são configurados utilizando a classe `Horario`. Por fim, as oficinas são agrupadas na classe `Curso` juntamente com o nome do curso das oficinas, que por sua vez são agrupadas na classe `Cursos`.

Figura 16 – Diagrama de classes do pacote `Objetos`

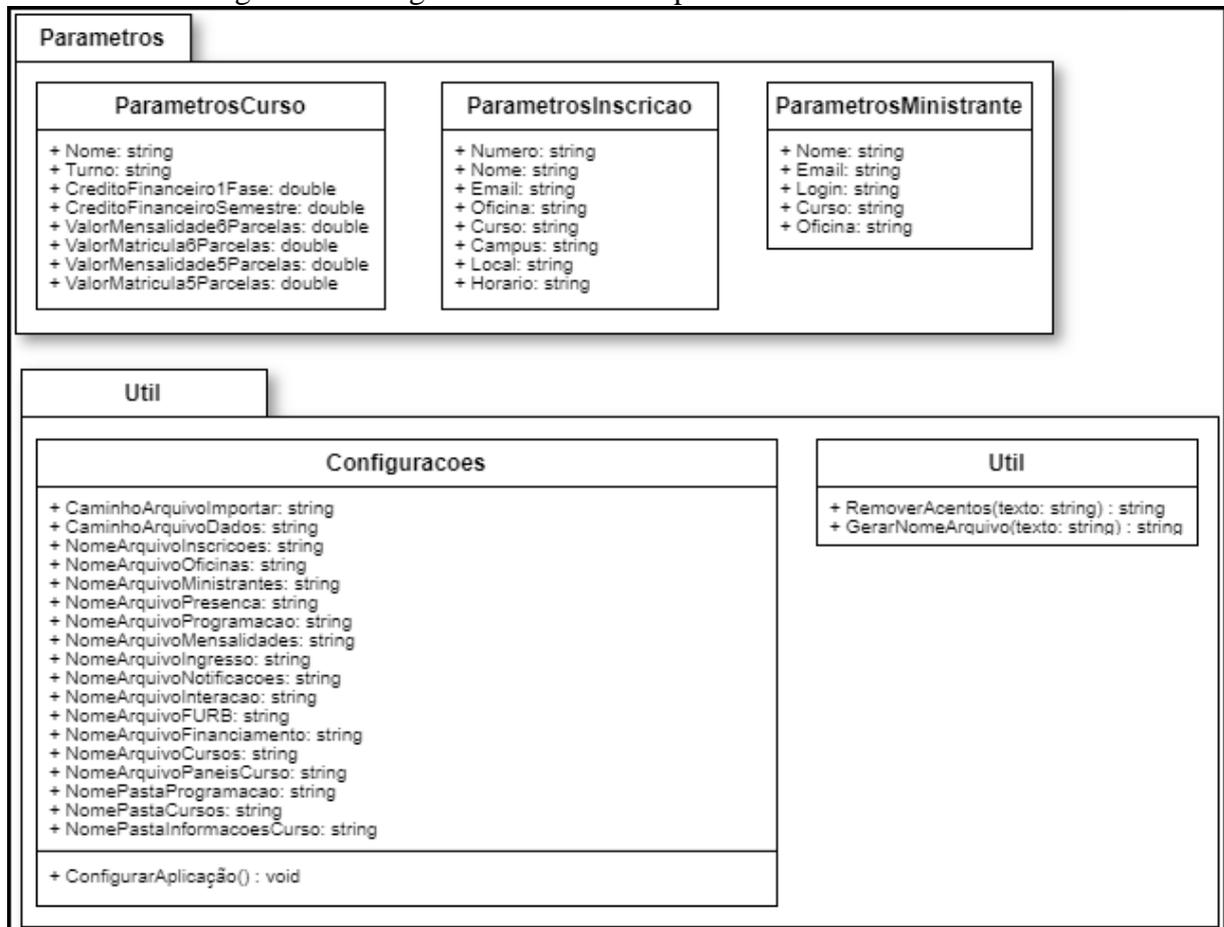


Fonte: elaborado pelo autor.

A Figura 17 demonstra a quarta parte do diagrama de classes que apresenta as classes contidas nos pacotes `Parametros` e `Util`. A classe `ParametrosCurso` é utilizada na importação de valores de mensalidade do site da FURB, através dela que são passadas as informações importadas para montar os cursos e as mensalidades. A classe `ParametrosInscricao` é utilizada nas importações de inscrições e ministrantes em arquivos Excel, através dela que são passadas as informações importadas para montar as inscrições, cursos e oficinas. A classe `ParametrosMinistrantes` é utilizada na importação de ministrantes em arquivos Excel, através dela que são passadas as informações importadas para montar os ministrantes. A classe `Configuracoes` é utilizada para armazenar todos os caminhos e nomes de arquivos que serão utilizados no serviço e na API. A classe `Util` é

utilizada para colocar os métodos que podem ser utilizados em qualquer lugar e não são específicos de uma funcionalidade.

Figura 17 – Diagrama de classes do pacote `Parametros` e `Util`



Fonte: elaborado pelo autor.

3.3 IMPLEMENTAÇÃO

Neste capítulo são mostradas as técnicas e ferramentas utilizadas, detalhes da implementação e também das funcionalidades do aplicativo. A seção 3.3.1 exibe informações sobre as técnicas e ferramentas utilizadas no desenvolvimento da solução. A seção 3.3.2 expõe detalhes da implementação do serviço de importação. A seção 3.3.3 demonstra detalhes da implementação da Web API que retorna informações para o aplicativo. Por fim, a seção 3.3.4 apresenta as funcionalidades do aplicativo móvel.

3.3.1 Técnicas e ferramentas utilizadas

A parte do servidor é composta pelo serviço e pela API desenvolvida em C#, utilizando a Integrated Development Environment (IDE) Visual Studio 2017. No desenvolvimento do serviço foi utilizada a biblioteca `HtmlAgilityPack` para ler os dados importados dos sites através de uma técnica chamada *Web scraping* e também a biblioteca

ClosedXML para ler os dados dos arquivos Excel. A API foi desenvolvida utilizando ASP.NET Core para dispensar a necessidade de utilizar o Internet Information Services (IIS) e também o *framework* ASP.NET Web API. Para o armazenamento e recuperação dos dados em JSON foi utilizado o *framework* Newtonsoft.Json.

O desenvolvimento do aplicativo móvel foi realizado com a linguagem de programação JavaScript juntamente com o *framework* React Native. A codificação também foi realizada utilizando a sintaxe JSX utilizada pelo React. As configurações iniciais do projeto Android foram feitas utilizando a IDE Android Studio 2.3.3, porém todo o desenvolvimento foi feito utilizando a IDE Visual Studio Code.

3.3.2 Serviço de importação

A FURB não disponibiliza um banco de dados ou uma API com as informações necessárias para a criação deste aplicativo. Dessa forma, algumas informações ficam diretamente em arquivos JSON e outras são importadas de diferentes locais por um serviço que fica rodando no servidor.

O serviço importa os dados de cinco fontes diferentes, duas delas são realizadas em sites e as outras três em arquivos Excel. As importações em sites são realizadas a cada 5 minutos e utilizam a técnica Web scraping que consiste em ler o código HTML de uma página web através de requisições HTTP e na sequência extrair as informações desejadas do HTML obtido.

O serviço realiza a importação das mensalidades dos cursos ofertados pela FURB através da Uniform Resource Locator (URL) <http://www.furb.br/web/mensalidades.php> e também realiza a importação das informações das oficinas disponíveis no evento Interação FURB através da URL <http://www10.furb.br/interacao/web/4265/interacao-furb/interacao-furb/programacao>. O Quadro 5 mostra a implementação do método que faz a importação dos cursos das oficinas disponíveis no evento Interação FURB.

Quadro 5 – Código do método ImportarOficinasSiteIteracao da classe ImportarDadosSite

```

1 public void ImportarOficinasSiteIteracao()
2 {
3     var wc = new WebClient();
4     var pagina =
wc.DownloadString("http://www10.furb.br/interacao/web/4265/interacao-
furb/interacao-furb/programacao");
5
6     var htmlDocument = new HtmlDocument();
7     htmlDocument.LoadHtml(pagina);
8
9     var divProgramacao = htmlDocument.GetElementById("programacao");
10
11    var data = new List<ObjetoProgramacao>();
12
13    foreach (var htmlNode in divProgramacao.ChildNodes)
14    {
15        if (htmlNode.Attributes["class"] != null &&
htmlNode.Attributes["class"].Value.IndexOf("curso") != -1)
16        {
17            var curso = htmlNode.ChildNodes[1].InnerText;
18
19            data.Add(new ObjetoProgramacao()
20            {
21                Text = curso
22            });
23
24            ImportarDadosOficinasSiteIteracao(htmlNode.ChildNodes[3], curso);
25        }
26    }
27
28    var programacao = new Programacao();
29    programacao.Data = data.ToArray();
30
31    using (StreamWriter file = new
StreamWriter(Path.Combine(Configuracoes.CaminhoArquivoDados,
Configuracoes.NomeArquivoProgramacao + ".json")))
32    {
33        file.WriteLine(JsonConvert.SerializeObject(programacao));
34    }
35 }

```

Fonte: elaborado pelo autor.

Na linha 3 é criada a instância da classe `WebClient` que é uma classe disponibilizada pelo pacote .NET para realização de requisições HTTP. Na linha 4 é realizada a requisição para buscar o conteúdo da página web. Na linha 6 é criada a instância da classe `HtmlDocument` que é uma classe disponibilizada pela biblioteca `HtmlAgilityPack` para facilitar a busca de informações em conteúdo HTML. Na linha 7 é carregado o conteúdo da página web para a instância da classe `HtmlDocument`. Na linha 9 é procurado no conteúdo HTML por um elemento com `id` igual a `programacao` pois neste elemento que se encontram as informações das oficinas do Interação FURB. Na linha 11 é criada uma lista de `ObjetoProgramacao` para guardar os cursos das oficinas encontrados. Na linha 13 é iniciada

uma varredura nos elementos filhos do elemento com `id` igual a `programacao`. Na linha 15 é verificado se o elemento filho possui uma classe chamada `curso`. Na linha 17 é pego o nome do curso. Na linha 19 é adicionado um novo `ObjetoProgramacao` com o nome do curso na lista de `ObjetoProgramacao`. Na linha 24 é chamado o método que realiza a importação das informações das oficinas do curso encontrado. Na linha 28 é criada uma instância da classe `Programacao`. Na linha 29 é atribuída a lista de `ObjetoProgramacao` no atributo `Data` da instância da classe `Programacao`. Na linha 31 é criado um arquivo JSON com o nome `Programação` para que seja salva a instância de `Programacao` em forma de JSON na linha 33.

As importações dos arquivos Excel ficam monitorando a pasta `C:\FurbMobile\Importar` utilizando o `FileSystemWatcher` que se trata de uma classe disponibilizada pelo pacote `.NET`. Quando é colocado um arquivo Excel com extensão `.xlsx` com o nome `Inscrições`, `Ministrantes` ou `Oficinas`, a importação desses arquivos é iniciada no mesmo instante. Os valores devem estar sempre nas mesmas colunas e guias de planilha, a importação foi feita levando em consideração o formato conforme são utilizados atualmente pela FURB. No Quadro 6, Quadro 7 e Quadro 8 é possível visualizar como devem estar presentes as informações em cada arquivo para que a importação funcione corretamente.

Quadro 6 – Localização das informações no arquivo `Inscrições.xlsx`

Informação	Guia de planilha	Coluna
número da inscrição	1	A
nome do visitante	1	B
e-mail do visitante	1	C
nome da oficina	1	D
nome do curso da oficina	1	E
local onde será realizada a oficina	1	F
horário de início da oficina	1	G

Fonte: elaborado pelo autor.

Quadro 7 – Localização das informações no arquivo `Ministrantes.xlsx`

Informação	Guia de planilha	Coluna
nome da oficina	1	A
nome do curso da oficina	1	B
e-mail do ministrante	1	I
nome do ministrante	1	H
nome do ministrante	2	A
<i>login</i> do ministrante	2	B

Fonte: elaborado pelo autor.

Quadro 8 – Localização das informações no arquivo `Oficinas.xlsx`

Informação	Guia de planilha	Coluna
nome da oficina	1	A
nome do curso da oficina	1	B
horário de início da oficina	1	C
campus onde será realizada a oficina	1	D
local onde será realizada a oficina	1	E

Fonte: elaborado pelo autor.

O Quadro 9 mostra a implementação do método que faz a importação das oficinas no arquivo Excel Oficinas.xlsx.

Quadro 9 – Código do método ImportarOficinas da classe ImportarDadosExcel

```
1 public void ImportarOficinas ()
2 {
3     var caminho = Path.Combine(Configuracoes.CaminhoArquivoImportar,
4     Configuracoes.NomeArquivoOficinas + ".xlsx");
5
6     if (File.Exists(caminho))
7     {
8         var lerDadosExcel = new LerDadosExcel ();
9
10        var cursos = lerDadosExcel.LerOficinas(caminho);
11
12        var json = JsonConvert.SerializeObject(cursos);
13
14        using (StreamWriter file = new
15        StreamWriter(Path.Combine(Configuracoes.CaminhoArquivoDados,
16        Configuracoes.NomeArquivoOficinas + ".json")))
17        {
18            file.WriteLine(json);
19        }
20    }
21 }
```

Fonte: elaborado pelo autor.

Na linha 3 é montado o caminho onde o Excel Oficinas.xlsx se encontra. Na linha 5 é verificado se o arquivo existe para ser importado. Na linha 7 é criada a instância da classe LerDadosExcel que é a classe onde ficam os métodos para leitura dos arquivos Excel. Na linha 9 é chamado o método responsável por fazer a leitura das oficinas no arquivo Excel Oficinas.xlsx e retornar os cursos e oficinas importadas. Na linha 11 os cursos e oficinas importados são serializados em forma de JSON. Na linha 13 é criado um arquivo JSON com o nome Oficinas para que sejam salvos os cursos e oficinas em forma de JSON na linha 15. O Quadro 10 mostra a implementação do método que faz a leitura das oficinas no arquivo Excel Oficinas.xlsx.

Quadro 10 – Código do método LerOficinas da classe LerDadosExcel

```

1 public Cursos LerOficinas(string caminho)
2 {
3     var cursos = new Cursos();
4
5     using (var wb = new XLWorkbook(caminho))
6     using (var planilha = wb.Worksheet(1))
7     {
8         var linha = 2;
9         while (true)
10        {
11            var oficina = LerValorCelula(planilha, "A", linha);
12            var curso = LerValorCelula(planilha, "B", linha);
13            var horario = LerValorCelula(planilha, "C", linha);
14            var campus = LerValorCelula(planilha, "D", linha);
15            var local = LerValorCelula(planilha, "E", linha);
16
17            if (string.IsNullOrEmpty(oficina) &&
18                string.IsNullOrEmpty(curso) &&
19                string.IsNullOrEmpty(horario) &&
20                string.IsNullOrEmpty(campus) &&
21                string.IsNullOrEmpty(local))
22                break;
23            cursos.AdicionarCurso(new ParametrosInscricao()
24            {
25                Oficina = oficina,
26                Curso = curso,
27                Horario = TratarHorario(horario),
28                Campus = campus,
29                Local = local
30            });
31
32            linha++;
33        }
34        planilha.Dispose();
35    }
36    return cursos;
37 }

```

Fonte: elaborado pelo autor.

Na linha 3 é criada uma instância da classe `Cursos` para guardar os cursos e oficinas encontrados no arquivo Excel. Na linha 5 é criada uma instância da classe `XLWorkbook` que é uma classe disponibilizada pela biblioteca `ClosedXML` para realizar a leitura em arquivos Excel, passando como parâmetro o caminho do arquivo `Oficinas.xlsx`. Na linha 6 é utilizada a instância da classe `XLWorkbook` para retornar o conteúdo da primeira guia da planilha. Na linha 8 é declarada a variável responsável pelo controle da linha da planilha que está sendo importada, iniciando pela linha 2 para que não seja importado o cabeçalho da planilha que se encontra na linha 1. Na linha 9 é realizado um loop para ler todas as linhas da planilha. Na linha 11 é lida a célula na linha atual e coluna A que é onde se encontra o nome da oficina. Na linha 12 é lida a célula na linha atual e coluna B que é onde se encontra o nome do curso da oficina. Na linha 13 é lida a célula na linha atual e coluna C que é onde se encontra o horário da oficina. Na linha 14 é lida a célula na linha atual e coluna D que é onde se encontra o

campus onde será realizada a oficina. Na linha 15 é lida a célula na linha atual e coluna E que é onde se encontra a sala em que será realizada a oficina. Na linha 18 é verificado se todas as colunas da linha atual estão vazias, se todas as informações da linha estiverem vazias é executado o comando na linha 23 para que seja finalizada a importação das oficinas. Na linha 23 são armazenadas as informações que foram lidas no Excel na instância da classe `Cursos`. Na linha 32 é incrementada a linha atual para que seja importado a próxima linha do Excel. Na linha 34 é liberada a planilha do Excel que estava sendo lida. Na linha 36 são retornados os cursos e oficinas que foram lidos da planilha.

3.3.3 Web API

A Web API é responsável por realizar a comunicação entre o aplicativo móvel e o servidor. Para isso, a API disponibiliza métodos para a recuperação de informações através de uma URL específica. A URL `http://ip:porta/api/Checkin`, por exemplo, disponibiliza acesso ao método para salvar as presenças dos visitantes. A implementação deste método pode ser visualizada no Quadro 11.

Quadro 11 – Código do método Post da classe Checkin

```

1  [HttpPost]
2  public void Post([FromBody]PresencaModel[] presenca)
3  {
4      Configuracoes.ConfigurarAplicação();
5
6      var caminho = Path.Combine(Configuracoes.CaminhoArquivoDados,
7      Configuracoes.NomeArquivoPresenca + ".json");
8      List<PresencaModel> presencas;
9
10     if (System.IO.File.Exists(caminho))
11     {
12         var jsonPresencas = System.IO.File.ReadAllText(caminho);
13         presencas =
14         JsonConvert.DeserializeObject<List<PresencaModel>>(jsonPresencas);
15     }
16     else
17     {
18         presencas = new List<PresencaModel>();
19     }
20     presencas.AddRange(presenca);
21
22     presencas = presencas.GroupBy(p => new { p.Email, pCurso, p.Oficina
23     }).Select(p => new PresencaModel()
24     {
25         Email = p.Key.Email,
26         Curso = p.Key.Curso,
27         Oficina = p.Key.Oficina
28     }).ToList();
29     using (StreamWriter file = new StreamWriter(caminho))
30     {
31         file.WriteLine(JsonConvert.SerializeObject(presencas));
32     }
33 }

```

Fonte: elaborado pelo autor.

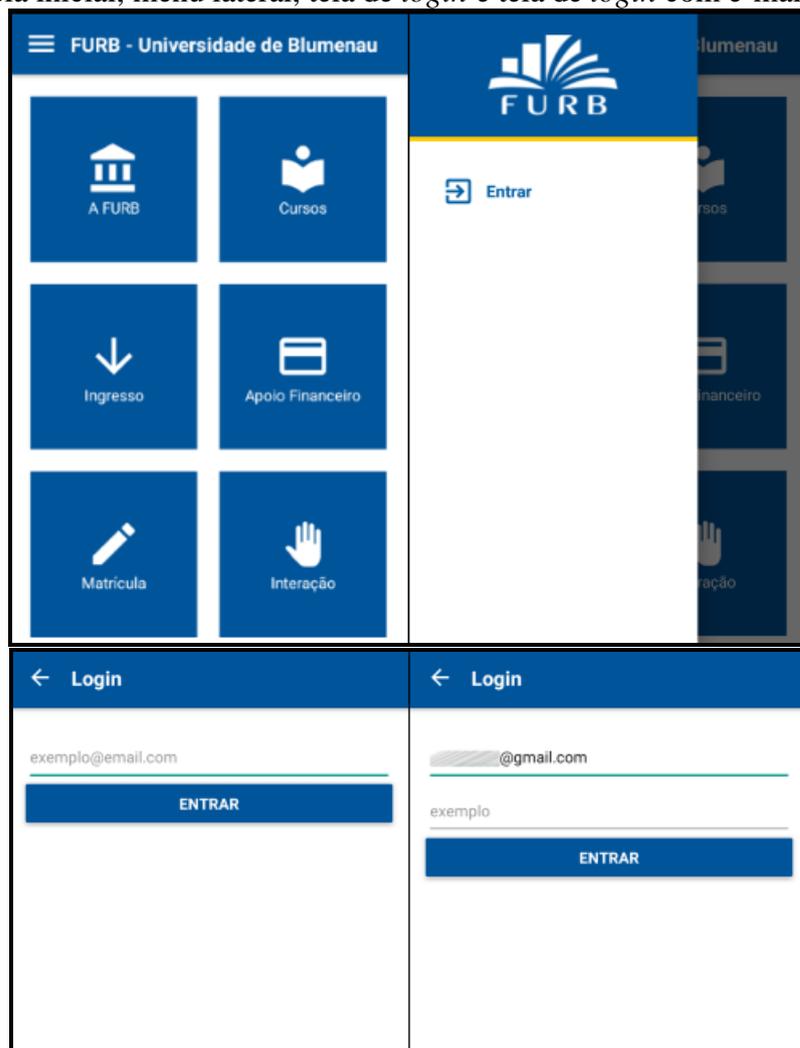
Na linha 1 é utilizada a notação `HttpPost` para definir que este método seja chamado somente quando a requisição para a URL for do tipo `Post`. Na linha 2 é possível visualizar que o método `Post` recebe como parâmetro um array de `PresencaModel` que é pego no corpo da requisição, por isso na frente do parâmetro existe a notação `FromBody`. Na linha 4 é chamado o método `ConfigurarAplicação` da classe `Configuracoes` para carregar os nomes e caminhos dos arquivos. Na linha 6 é montado o caminho do arquivo onde são salvas as presenças dos visitantes. Na linha 7 é criada uma lista de `PresencaModel` para guardar as presenças. Na linha 9 é verificado se o arquivo de presença já existe, se o arquivo já existir o mesmo é lido na linha 11 e as presenças salvas no arquivo são carregadas para a lista de presenças na linha 12, mas se o arquivo de presenças ainda não existir, é criada uma nova instância da lista de `PresencaModel` na linha 15. Na linha 17 as presenças que estão na lista de presenças recebida por parâmetro, são adicionadas na lista de presenças que foram carregadas. Na linha 19 a lista com todas as presenças é agrupada por e-mail, curso e oficina respectivamente, afim de remover os registros duplicados de presença. Na linha 25 é criado

um arquivo JSON com o nome Presença para que seja salva a lista de presenças em forma de JSON na linha 27.

3.3.4 Aplicativo móvel

Ao acessar o aplicativo o usuário é direcionado para a tela inicial que possibilita o acesso a seis itens: A FURB, Cursos, Ingresso, Apoio Financeiro, Matrícula e Interação. Também é possível acessar o menu através do botão que está localizado no topo ou arrastando o dedo da esquerda para a direita. No menu estará disponível o botão Entrar para realizar o *login* no aplicativo com um e-mail cadastrado no evento Interação FURB ou com o e-mail e *login* de um ministrante do evento. A Figura 18 mostra a tela inicial do aplicativo, o menu sem que o usuário tenha realizado o *login*, a tela de *login* e a tela de *login* ao informar um e-mail de um ministrante.

Figura 18 – Tela inicial, menu lateral, tela de *login* e tela de *login* com e-mail do ministrante



Fonte: elaborado pelo autor.

No item **A FURB** o usuário tem acesso a algumas informações básicas sobre a universidade. Ao acessar o item **Cursos** o usuário será direcionado inicialmente para uma tela com as letras iniciais dos cursos disponíveis. Ao acessar uma letra, o usuário irá se deparar com uma lista de cursos que iniciam com a letra selecionada. Ao clicar sobre um curso o usuário será redirecionado para a tela de visualização do curso, onde estarão disponíveis várias informações sobre o curso. A Figura 19 mostra a tela com informações básicas sobre a FURB, a lista de letras iniciais dos cursos, a lista de cursos que iniciam com a letra selecionada e a tela com as informações do curso selecionado.

Figura 19 – Tela A FURB, lista de letras, lista de cursos e informações do curso



Fonte: elaborado pelo autor.

Ao acessar o item **Ingresso** o usuário se depara com as possíveis formas de ingressar na FURB. No item **Apoio Financeiro** o usuário tem acesso as formas de apoio financeiro

Figura 21 – Tela do Interação FURB, lista de cursos com oficinas e informações das oficinas

The screenshot displays the 'Interação FURB' app interface, divided into three main sections:

- Interação (Left Panel):** Contains introductory text about the opportunity to know the University of Blumenau and participate in workshops. It also includes a section titled 'Programação de 2017' with a note: '* Programação sujeita a alterações até o início das inscrições.' Below this is a 'Como participar' section listing requirements:
 - ser estudante do 1º, 2º e 3º ano do ensino médio ou comunidade em geral;
 - fazer a inscrição on-line nas oficinas e palestras de interesse;
 - imprimir comprovante de inscrição e apresentar no dia do evento;
 - comparecer no câmpus 1 da FURB no dia do evento.
- Programação (Middle Panel):** A grid of course categories:
 - Administração
 - Arquitetura e Urbanismo
 - Artes Visuais
 - Atividade Especial
 - Atividade Interação
 - Biomedicina
- Programação (Right Panel):** Details for the 'Aprenda A Programar De Forma Divertida' workshop:
 - Objective: ensinar alguns conceitos básicos de programação por meio do FURBOT.
 - Horários: 10h30 às 11h30* and Câmpus 1 - S-415 - 15h30 às 16h30*.
 - Note: * Horários que não possuem mais vagas disponíveis.
 - Other workshop options listed below: Aprendendo A Programar Robôs, Computação Gráfica E Entretenimento Digital, Iot - Internet Das Coisas, Tecnologias De Redes De Computadores, and Um Dia De Trabalho Em Uma Empresa De Informática.

Fonte: elaborado pelo autor.

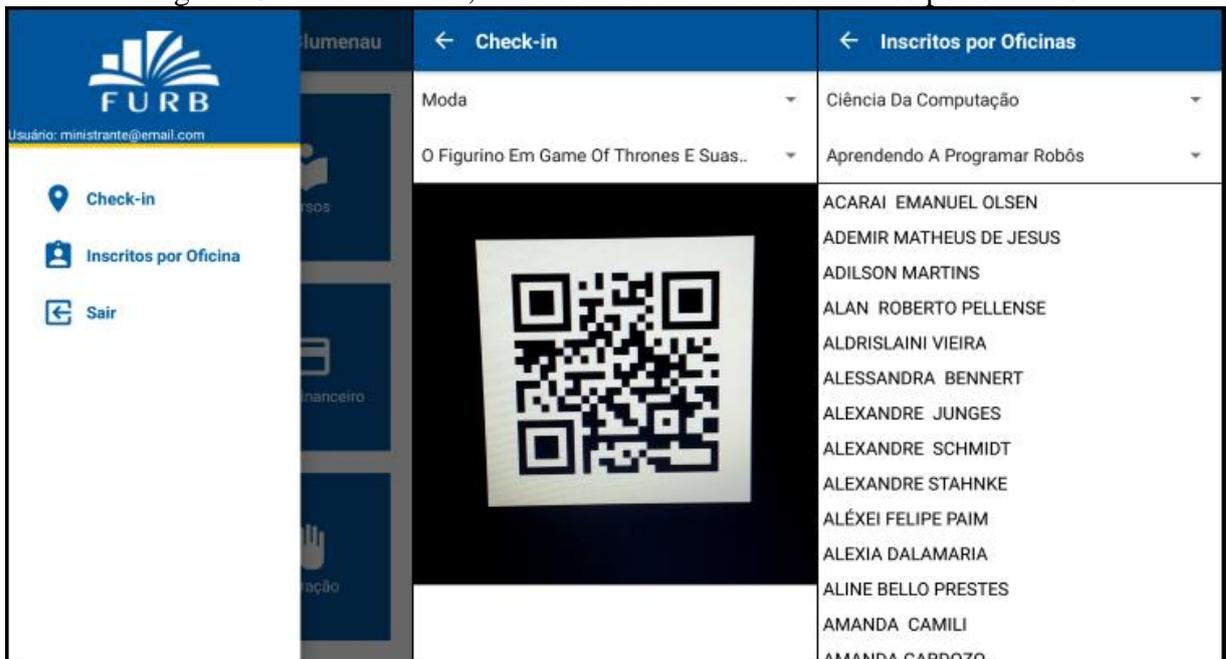
Ao acessar o aplicativo como visitante, é possível visualizar no menu lateral o e-mail do visitante e mais três opções: Check-in, Minhas Oficinas e Sair. Na tela de *check-in* o visitante se depara com um QRCode que possibilita a realização de um *check-in* nas oficinas, o visitante precisa apresentar o QRCode para o ministrante do curso escanear. Ao acessar o item Minhas Oficinas o usuário é redirecionado para a tela de programação do Interação FURB, porém filtrando somente os cursos das oficinas nas quais o mesmo está cadastrado e ao acessar um curso, o visitante pode visualizar informações das oficinas nas quais está cadastrado. Ao clicar em Sair as opções disponíveis para visitantes não são mais exibidas. A Figura 22 apresenta o menu lateral, a tela de *check-in*, a tela com os cursos das oficinas do visitante e a tela com as informações das oficinas.

Figura 22 – Menu lateral, tela de *check-in*, tela de minhas oficinas e informações das oficinas



Fonte: elaborado pelo autor.

Ao acessar o aplicativo como ministrante, é possível visualizar no menu lateral o e-mail do ministrante e mais três opções: Check-in, Inscritos por Oficina e Sair. Na tela de *check-in* o ministrante tem a possibilidade de registrar a presença dos visitantes em suas oficinas escaneando o QRCode do visitante, sendo assim somente estão disponíveis os cursos e oficinas nas quais o usuário é ministrante. O ministrante pode registrar a presença de todos os visitantes que possuem QRCode, independentemente de estarem cadastrados na oficina ou não. Na tela de Inscritos por Oficina, o ministrante pode consultar o nome de todos os visitantes que estão cadastrados em cada uma de suas oficinas. Ao clicar em Sair as opções disponíveis para ministrantes não são mais exibidas. A Figura 23 mostra respectivamente o menu lateral, a tela de *check-in* e a tela de inscritos por oficinas.

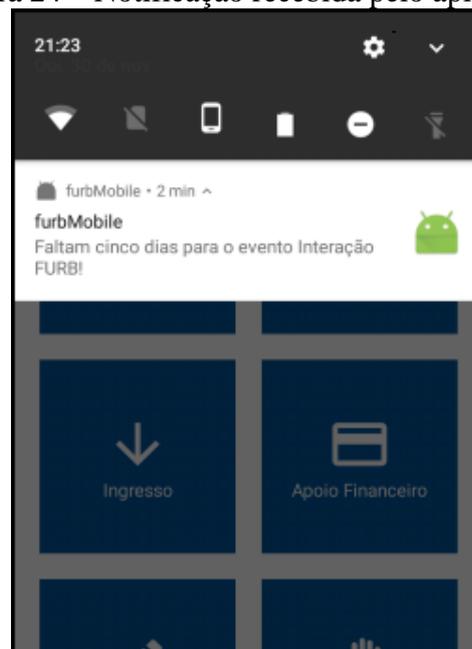
Figura 23 – Menu lateral, tela de *check-in* e tela de inscritos por oficinas

Fonte: elaborada pelo autor.

Os usuários dos aplicativos receberão notificações cadastradas no servidor, é possível configurar um texto e uma data para que a notificação seja exibida no celular. Sempre que o usuário acessar o aplicativo, as notificações cadastradas no servidor são atualizadas no celular, dessa forma quando chegar a data configurada, a notificação é apresentada mesmo se o usuário não estiver com o aplicativo aberto no momento. A

Figura 24 mostra uma notificação recebida pelo aplicativo.

Figura 24 – Notificação recebida pelo aplicativo



Fonte: elaborado pelo autor.

3.4 ANÁLISE DOS RESULTADOS

Neste capítulo são apresentados os experimentos realizados para validar se o aplicativo cumpriu seus objetivos. A seção 3.4.1 descreve como foi realizado o teste de usabilidade com os voluntários. A seção 3.4.2 mostra os resultados obtidos com relação ao questionário para verificar qual o perfil do usuário. A seção 3.4.3 apresenta a lista de tarefas a serem realizadas no aplicativo e no site da FURB. A seção 3.4.4 expõe o resultado dos questionários de avaliação do aplicativo. A seção 3.4.5 aborda o teste de compatibilidade realizado em dispositivos móveis diferentes. Por fim, a seção 3.4.6 apresenta um comparativo entre este trabalho e os trabalhos correlatos.

3.4.1 Teste de usabilidade

O teste de usabilidade foi realizado com dez usuários, sendo oito usuários atuando como visitante e os outros dois atuando como ministrante. Cada usuário esteve sujeito a um questionário de perfil, uma lista de tarefas e um questionário de avaliação do aplicativo, específico para o tipo de usuário. Os questionários e a lista de tarefas estão disponíveis no Apêndice A. Na realização dos testes, alguns usuários foram orientados a instalar o `apk` nos seus dispositivos móveis e para outros foi disponibilizado um aparelho com o aplicativo instalado previamente.

Primeiramente, foi solicitado aos participantes, que não utilizaram o aparelho disponibilizado, que instalassem o `apk` em seus dispositivos. Em seguida, foi apresentada uma breve explicação sobre o objetivo do aplicativo, aplicando o questionário de perfil de usuário. Na sequência, os usuários foram motivados a realizar quatorze tarefas descritas na lista de tarefas, sendo que sete das tarefas eram sem a realização do *login*, três tarefas eram realizando *login* com o tipo de usuário específico, duas tarefas acessando o aplicativo sem conexão com a internet e as duas últimas tarefas foram realizadas no site da FURB para efeitos de comparação.

Após a realização das tarefas os participantes foram convidados a responder o questionário de avaliação do aplicativo. O questionário foi composto por nove perguntas com o objetivo de validar se o aplicativo é de fácil utilização e se as informações são encontradas com mais facilidade no aplicativo ou no site da FURB. No final do questionário foi colocada uma pergunta aberta para que o usuário deixasse suas críticas e sugestões sobre o aplicativo. Os resultados desse experimento são apresentados nas próximas seções.

3.4.2 Análise do perfil de usuário

A primeira análise foi realizada através das respostas obtidas no questionário de perfil de usuário. Este questionário teve como objetivo identificar qual era o público avaliador, quais são os seus hábitos e possíveis necessidades que o participante poderia ter em relação ao assunto abordado. A Tabela 1 apresenta o resultado dos questionários de perfil que foram respondidos pelos participantes que atuaram como visitante no experimento.

Tabela 1 – Perfil dos participantes do experimento atuando como visitante

Sexo	75% masculino 25% feminino
Idade	0% menos de 18 anos 87,5% entre 18 e 25 anos 12,5% entre 26 e 35 anos 0% mais de 35 anos
Profissão	12,5% analista de qualidade 37,5% bolsista 25% desenvolvedor 12,5% operador de micro 12,5% suporte
Quanto você deseja algo e precisa da aprovação dos seus responsáveis, você normalmente mostra o que você quer através do	50% celular 50% computador 0% revista/jornal
Qual meio de comunicação você normalmente utiliza para buscar informações?	37,5% celular 62,5% computador 0% jornais 0% revistas
Se a resposta de algumas das duas perguntas anteriores for Celular, você prefere utilizar	40% navegador 60% aplicativo específico
Você utiliza o celular todos os dias?	100% Sim 0% Não
Você já acessou o site da FURB?	87,5% Sim 12,5% Não
Você teve dificuldade para encontrar informações dos cursos no site da FURB?	50% Sim 37,5% Não 12,5% Nunca acessei
No seu dia a dia, você prefere utilizar meios digitais ou materiais (papeis impressos)?	25% materiais 75% digitais

Fonte: elaborado pelo autor.

A partir dos dados apresentados na Tabela 1 é possível visualizar que os participantes que atuaram como visitante no experimento, em sua maioria são do sexo masculino, com idade entre 18 e 25 anos e atuam nas mais variadas profissões. A metade dos visitantes optam por mostrar algo que desejam e precisam da aprovação dos seus responsáveis através do celular e a outra metade através do computador. O meio de comunicação utilizado para buscar informações em sua maioria é o computador, porém dos que preferem o celular, 60% optam pelo uso de aplicativos específicos ao invés de acessar sites através do navegador do celular. Todos os participantes utilizam o celular todos os dias. A metade dos participantes já teve

dificuldades em encontrar informações dos cursos no site da FURB, enquanto a outra metade não teve dificuldade ou nunca acessou o site. Por fim, 75% dos participantes optam por utilizar meios digitais, ao invés de utilizar meios materiais. A Tabela 2 apresenta o resultado dos questionários de perfil que foram respondidos pelos participantes que atuaram como ministrante no experimento.

Tabela 2 – Perfil dos participantes do experimento atuando como ministrante

Sexo	50% masculino 50% feminino
Idade	0% menos de 18 anos 0% entre 18 e 25 anos 50% entre 26 e 35 anos 50% mais de 35 anos
Profissão	100% Professor
Quanto tempo você gasta para passar informações referentes ao curso e mensalidade?	0% 5 a 10 minutos 50% 10 a 15 minutos 50% 15 a 20 minutos
Você entrega algum material com informações a respeito do curso?	0% Sim 100% Não
Você acha interessante ter um aplicativo com estas informações para aumentar o tempo útil da oficina?	100% Sim 0% Não
Você gostaria que receber feedback das oficinas mediante avaliação dos visitantes?	100% Sim 0% Não
Você costuma verificar quantos participaram das suas oficinas?	100% Sim 0% Não
Você já teve dificuldades em saber o que fazer com as listas de presenças com as assinaturas dos visitantes?	0% Sim 100% Não
Qual meio de comunicação você normalmente utiliza para buscar informações?	50% celular 50% computador 0% jornais 0% revistas
Se a resposta da alternativa anterior for Celular, você prefere utilizar	0% navegador 100% aplicativos específicos
Você utiliza o celular todos os dias?	100% Sim 0% Não
Você normalmente acessa o site da FURB?	50% Sim 50% Não
Você já teve dificuldade para encontrar informações dos cursos no site da FURB?	50% Sim 50% Não 0% Nunca acessei
No seu dia a dia, você prefere utilizar meios materiais (papeis impressos) ou digitais?	0% materiais 100% digitais

Fonte: elaborado pelo autor.

Com base nos dados apresentados na Tabela 2 é possível visualizar que a metade dos participantes que atuaram como ministrante no experimento são do sexo feminino e a outra metade do sexo masculino. Além disso, todos os participantes: possuem mais de 25 anos, trabalham no cargo de professor, preferem utilizar meios digitais, utilizam o celular todos os

dias, gostariam de receber feedback dos visitantes com relação as suas oficinas e costumam verificar quantos visitantes participaram de suas oficinas.

A metade dos participantes normalmente utilizam o celular para buscar informações e dessa metade todos preferem utilizar aplicativos ao invés de utilizar o navegador do celular.

Também é possível visualizar que os ministrantes gastam entre 10 e 20 minutos para passar informações sobre cursos e mensalidades em suas oficinas, além de não entregar nenhum material para que os visitantes possam consultar posteriormente sobre os cursos mencionados. Com isso todos acharam interessante a possibilidade de possuir um aplicativo que disponibilize informações sobre os cursos para que não gastem muito tempo na oficina abordando este assunto.

Nenhum dos participantes tiveram dificuldades em saber o que fazer com a lista de presenças com assinaturas dos visitantes. Por fim, metade dos participantes normalmente acessam o site da FURB e metade também já teve dificuldade em encontrar informações dos cursos no site da FURB.

3.4.3 Análise da lista de tarefas

A segunda análise foi realizada através das respostas obtidas na lista de tarefas que o participante foi orientado a executar no aplicativo e no site da FURB, com o objetivo de verificar se as informações no aplicativo ficaram fáceis de serem encontradas e realizar uma comparação com o site da FURB. A Tabela 3 apresenta o resultado da parte da lista de tarefas que não era necessário realizar *login* no aplicativo, esta parte da lista de tarefas era a mesma tanto para visitantes, quanto para ministrantes.

Tabela 3 – Parte da lista de tarefas que não era necessário a realização de *login*

Visualizar informações sobre a FURB. A tarefa foi executada?	100% Sim 0% Não
Visualizar o valor da mensalidade do curso “Engenharia Elétrica”. A tarefa foi executada?	100% Sim 0% Não
Visualizar informações de Ingresso. A tarefa foi executada?	100% Sim 0% Não
Visualizar informações sobre os tipos de apoios financeiros. A tarefa foi executada?	100% Sim 0% Não
Visualizar o processo de matrícula. A tarefa foi executada?	100% Sim 0% Não
Visualizar a data em que será realizado o evento Interação FURB. A tarefa foi executada?	100% Sim 0% Não
Visualizar horário de uma oficina do evento Interação FURB. A tarefa foi executada?	40% Sim 60% Não

Fonte: elaborado pelo autor.

A partir das tarefas listadas na Tabela 3 é possível visualizar que quase todas as tarefas foram executadas pelos participantes sem dificuldades, a única tarefa que 60% dos participantes não conseguiram realizar foi a de visualizar o horário de uma oficina do evento Interação FURB. O motivo apontado pela maioria dos participantes foi que não estava claro a possibilidade de clicar sobre o link “Programação de 2017” e alguns falaram que não estava claro que ao acessar a programação do evento seriam encontrados os horários das oficinas. A Tabela 4 mostra o resultado da parte da lista que era necessário realizar *login* no aplicativo com um visitante.

Tabela 4 – Parte da lista de tarefas que é necessária a realização de *login* com um visitante

Realizar o login com um visitante cadastrado. Utilize o e-mail “visitante@email.com” para realizar o login como visitante. A tarefa foi executada?	100% Sim 0% Não
Visualizar as oficinas que o visitante está matriculado. A tarefa foi executada?	100% Sim 0% Não
Visualizar o QRCode do visitante para realização do check-in. A tarefa foi executada?	100% Sim 0% Não

Fonte: elaborado pelo autor.

A partir das tarefas listadas na Tabela 4 é possível visualizar que todas as tarefas foram executadas pelos visitantes sem dificuldades. A Tabela 5 mostra o resultado da parte da lista que era necessário testar o aplicativo com um visitante sem acesso à internet.

Tabela 5 – Parte da lista de tarefas para testar o aplicativo com um visitante sem internet

Visualizar o valor da mensalidade do curso “Engenharia Elétrica”. A tarefa foi executada?	100% Sim 0% Não
Visualizar as oficinas que o visitante está matriculado. A tarefa foi executada?	100% Sim 0% Não

Fonte: elaborado pelo autor.

A partir das tarefas listadas na Tabela 5 é possível visualizar que foi possível realizar as tarefas solicitadas mesmo sem que o visitante possuísse acesso à internet. A Tabela 6 mostra o resultado da parte da lista que era necessário realizar *login* no aplicativo com um ministrante.

Tabela 6 – Parte da lista de tarefas que é necessário a realização de *login* com um ministrante

Realizar o login com um ministrante de uma oficina. Utilize o e-mail “ministrante@email.com” para realizar o login como ministrante e no segundo campo que será exibido informe “ministrante”. A tarefa foi executada?	100% Sim 0% Não
Registrar a presença de um visitante escaneando um QRCode. A tarefa foi executada?	50% Sim 50% Não
Visualizar os participantes de suas oficinas. A tarefa foi executada?	100% Sim 0% Não

Fonte: elaborado pelo autor.

A partir das tarefas listadas na Tabela 6 é possível visualizar que quase todas as tarefas foram executadas pelos ministrantes sem dificuldades, exceto a tarefa de registrar a presença

de um visitante escaneando um QRCode que foi realizado apenas por metade dos participantes. O motivo apontado para que não fosse possível realizar esta tarefa é que o nome do item “Check-in” não está correto, pois entendesse que *check-in* é a ação de registrar a presença do próprio usuário do aplicativo e não de outros usuários. A Tabela 7 mostra o resultado da parte da lista que era necessário testar o aplicativo com um ministrante sem acesso à internet.

Tabela 7 – Parte da lista de tarefas para testar o aplicativo com um ministrante sem internet

Visualizar o valor da mensalidade do curso “Engenharia Elétrica”. A tarefa foi executada?	100% Sim 0% Não
Registrar a presença de um visitante escaneando um QRCode. A tarefa foi executada?	100% Sim 0% Não

Fonte: elaborado pelo autor.

A partir das tarefas listadas na Tabela 7 é possível visualizar que foi possível realizar as tarefas solicitadas mesmo sem que o ministrante possuísse acesso à internet. Por fim, a Tabela 8 mostra o resultado da parte da lista que era necessário realizar tarefas no site da FURB, esta parte da lista de tarefas era a mesma tanto para visitantes, quanto para ministrantes.

Tabela 8 – Parte da lista que era necessário realizar tarefas no site da FURB

Visualizar o valor da mensalidade do curso “Engenharia Elétrica”. A tarefa foi executada?	100% Sim 0% Não
Visualizar informações sobre os tipos de apoios financeiros. A tarefa foi executada?	90% Sim 10% Não

Fonte: elaborado pelo autor.

A partir das tarefas listadas na Tabela 8 é possível visualizar que todas as tarefas foram executadas, exceto a tarefa para visualizar os tipos de apoios financeiros que não foram realizadas por 10% dos participantes.

3.4.4 Análise da avaliação do aplicativo

A terceira e última análise foi realizada através das respostas obtidas no questionário de avaliação do aplicativo. Este questionário teve como objetivo avaliar a usabilidade do aplicativo, verificar se o aplicativo trouxe algum benefício em relação ao site da FURB e se os participantes utilizariam o aplicativo. A Tabela 9 apresenta o resultado dos questionários de avaliação do aplicativo que foram respondidos pelos participantes que atuaram como visitante no experimento.

Tabela 9 – Questionário de avaliação do aplicativo atuando como visitante

O aplicativo é visualmente agradável e atraente?	100% Sim 0% Não
A navegação pelo aplicativo é fácil?	100% Sim 0% Não
Os ícones são intuitivos?	100% Sim 0% Não
O aplicativo em algum momento deixou você sem ação?	0% Sim 100% Não
É possível utilizar o aplicativo sem o apoio de outra pessoa?	100% Sim 0% Não
Você utilizaria este aplicativo?	87,5% Sim 12,5% Não
Você recomendaria esse aplicativo para outra pessoa?	87,5% Sim 12,5% Não
Foi mais fácil encontrar o valor da mensalidade do curso no aplicativo do que no site da FURB?	100% Sim 0% Não
Foi mais fácil encontrar informações sobre apoio financeiro no aplicativo do que no site da FURB?	100% Sim 0% Não

Fonte: elaborado pelo autor.

A partir dos dados apresentados na Tabela 9 é possível visualizar que todos os participantes que atuaram como visitante no aplicativo responderam que o aplicativo é visualmente agradável e atraente, a navegação pelo aplicativo é fácil, os ícones são intuitivos e que é possível utilizar o aplicativo sem o apoio de outra pessoa.

O aplicativo em nenhum momento deixou algum dos participantes sem ação, neste quesito é levado em consideração se em algum momento o usuário chegou em uma tela que não teve aonde clicar para voltar ou prosseguir. Também foi informado por todos os participantes que é mais fácil encontrar o valor da mensalidade de um curso e informações sobre apoio financeiro no aplicativo do que no site da FURB. No entanto, apesar de todos os itens terem sido avaliados positivamente nem todos os participantes utilizariam o aplicativo, no total 12,5% não utilizaria o aplicativo e nem recomendaria. O motivo apontado por esses participantes foi de que não via necessidade para eles utilizarem o aplicativo e como eles não utilizariam o aplicativo, também não iriam recomendar o mesmo. A Tabela 10 apresenta o resultado dos questionários de avaliação do aplicativo que foram respondidos pelos participantes que atuaram como ministrante no experimento.

Tabela 10 – Questionário de avaliação do aplicativo atuando como ministrante

O aplicativo é visualmente agradável e atraente?	100% Sim 0% Não
A navegação pelo aplicativo é fácil?	100% Sim 0% Não
Os ícones são intuitivos?	50% Sim 50% Não
O aplicativo em algum momento deixou você sem ação?	0% Sim 100% Não
É possível utilizar o aplicativo sem o apoio de outra pessoa?	100% Sim 0% Não
Você utilizaria este aplicativo?	100% Sim 0% Não
Foi mais fácil encontrar o valor da mensalidade do curso no aplicativo do que no site da FURB?	50% Sim 50% Não
Foi mais fácil encontrar informações sobre apoio financeiro no aplicativo do que no site da FURB?	100% Sim 0% Não
Você acha que o uso do QRCode facilita o registro da presença?	50% Sim 50% Não

Fonte: elaborado pelo autor.

A partir dos dados apresentados na Tabela 10 é possível visualizar que todos os participantes que atuaram como ministrante no aplicativo responderam que o aplicativo é visualmente agradável e atraente, a navegação pelo aplicativo é fácil e que é possível utilizar o aplicativo sem o apoio de outra pessoa. Com relação aos ícones, 50% responderam que os ícones são intuitivos. O aplicativo em nenhum momento deixou algum dos participantes sem ação, neste quesito é levado em consideração se em algum momento o usuário chegou em uma tela que não teve aonde clicar para voltar ou prosseguir.

Todos os participantes informaram que é mais fácil encontrar as informações sobre apoio financeiro no aplicativo do que no site da FURB, porém a metade respondeu que não é mais fácil encontrar as informações de valor da mensalidade de um curso no aplicativo.

A metade dos ministrantes acham que o registro de presença através do QRCode não vai facilitar o registro de presença pelo fato de que atualmente com a folha de presença é possível passar a folha para os alunos assinarem enquanto a oficina é ministrada e com este novo método seria necessário o ministrante parar um instante para registrar a presença de cada visitante. Por fim, todos os ministrantes responderam que utilizariam o aplicativo.

3.4.5 Teste de compatibilidade

O teste de compatibilidade foi realizado em sete dispositivos Android, possibilitando verificar como o aplicativo se comporta executando em diferentes aparelhos, com versões diferentes do Android e diferentes tamanhos de tela. O Quadro 12 apresenta características dos dispositivos móveis utilizados nos testes.

Quadro 12 – Dispositivos móveis utilizados nos testes

Quantidade	Modelo do dispositivo	Versão do Android	Tamanho da tela (polegadas)	Resolução
1	Samsung Galaxy J3	5.1.1	5	720 x 1280
2	Samsung Galaxy J5	6.0.1	5	720 x 1280
1	Alcatel Pixi	6.0	4	480 x 800
1	Galaxy Win Duos	4.1.2	4.7	480 x 800
1	Galaxy Note 2	7.1.2	5.5	720 x 1280
1	Xiaomi Mi 5	7.0	5.15	1080 x 1920

Fonte: elaborado pelo autor.

O aplicativo executou corretamente em todos os dispositivos móveis testados, inclusive com todas as funcionalidades funcionando corretamente. As telas do aplicativo se ajustaram de acordo com o tamanho da tela e resolução dos dispositivos. Mesmo nos dispositivos com versões mais antigas foi possível utilizar o aplicativo sem problemas. O único problema que aconteceu no dispositivo Alcatel Pixi foi que em um determinado momento ao clicar muito rápido nas funcionalidades o menu lateral permaneceu aberto quando deveria ter sido fechado automaticamente, porém nada que não poderia ser resolvido fechando o menu manualmente.

3.4.6 Comparação com os trabalhos correlatos

No Quadro 13 é apresentado um comparativo entre este trabalho e os trabalhos correlatos, onde as linhas representam as características e as colunas representam os trabalhos.

Quadro 13 – Comparativo entre este trabalho e os trabalhos correlatos

Características / Trabalhos	Trabalho desenvolvido	Barreto (2016)	Rocha (2016)	Sampara (2010)
plataforma	Android	Android	iOS e Android	iOS
utiliza Web Service	Sim	Sim	Sim	Sim
auxilia participantes do evento Interação FURB	Sim	Não	Sim	Não
informações sobre cursos ofertados	Sim	Sim	Não	Não
informações úteis para alunos da FURB	Sim	Sim	Sim	Sim

Fonte: elaborado pelo autor.

A partir do Quadro 13, pode-se observar que todos os trabalhos são aplicativos móveis e se comunicam com um Webservice. O aplicativo de Rocha (2016) é o único multiplataforma, o de Sampara (2010) foi desenvolvido para a plataforma iOS, enquanto o aplicativo deste trabalho e o de Barreto (2016) rodam na plataforma Android. Em relação as funcionalidades de cada trabalho, pode-se perceber que todos fornecem informações que podem ser úteis para alunos da FURB. Entretanto, para pessoas que estejam interessadas em ingressar na universidade, somente o aplicativo de Barreto (2016) e o aplicativo desenvolvido neste trabalho mostram informações sobre os cursos ofertados.

Somente o aplicativo desenvolvido e o de Rocha (2016) auxiliam os participantes do Interação FURB. Cada um dos dois aplicativos auxiliam os participantes de uma maneira. O aplicativo desenvolvido fornece a possibilidade de realizar *check-in* nas oficinas, disponibiliza informações sobre o evento e também permite disparar notificações para lembrar os inscritos sobre o evento. O aplicativo de Rocha (2016) tem foco no deslocamento dos participantes do Interação FURB. Por fim, é possível visualizar que o aplicativo desenvolvido foca na disponibilização de informações aos participantes do Interação FURB e pessoas que estejam interessadas em conhecer os cursos disponibilizados pela FURB.

4 CONCLUSÕES

Este trabalho propôs o desenvolvimento de um aplicativo Android com o objetivo de facilitar o acesso as informações para pessoas que tem interesse em conhecer os cursos que a FURB oferece e que tem interesse em ingressar na universidade. O aplicativo também tem como objetivo auxiliar no controle de presenças do evento Interação FURB, fornecendo uma maneira de realizar o registro de presenças dos visitantes.

O aplicativo foi desenvolvido na IDE Visual Studio Code utilizando o *framework* React Native e foi escrito na linguagem de programação JavaScript juntamente com a sintaxe JSX do React. A parte do servidor foi desenvolvida na IDE Visual Studio utilizando a linguagem de programação C#. No desenvolvimento do serviço vale destacar a utilização da biblioteca HtmlAgilityPack para ler os dados importados dos sites e da biblioteca ClosedXML para ler os dados dos arquivos Excel. Enquanto na API foi utilizado o *framework* ASP.NET Web API. As tecnologias utilizadas no desenvolvimento do trabalho se mostraram muito eficientes, apesar da falta de conhecimento nessas tecnologias, foi possível desenvolver as funcionalidades desejadas.

O aplicativo fornece várias informações sobre os cursos oferecidos pela FURB, os tipos de apoios financeiros, primeira matrícula na universidade, informações sobre o evento Interação FURB e suas oficinas. Além de disponibilizar uma alternativa para a realização de *check-in* nas oficinas do evento Interação FURB e permitir que sejam emitidas notificações para os usuários do aplicativo. Outra funcionalidade que merece destaque é o acesso off-line que permite aos usuários utilizarem o aplicativo sem ter acesso à internet.

Com base nos experimentos, verificou-se que alguns pontos poderiam ser melhorados com pequenos ajustes, porém no geral o aplicativo ficou intuitivo e de fácil utilização. Quase todas as pessoas que participaram dos experimentos falaram que utilizariam o aplicativo. O teste de compatibilidade mostrou que o aplicativo funcionou em vários modelos de celulares, com diferentes tamanhos de tela, resoluções e versões do Android.

Por fim, o aplicativo móvel atingiu o objetivo de facilitar o acesso às informações dos cursos oferecidos pela FURB e, ao mesmo tempo, oferecer uma ferramenta para registrar a presenças dos visitantes que frequentam o evento Interação FURB.

4.1 EXTENSÕES

Algumas das extensões possíveis para este trabalho são:

- a) fazer com que no primeiro *login* seja enviado um link para o e-mail informado no aplicativo, solicitando a criação de uma senha para acessos futuros;

- b) otimizar o acesso off-line para que o usuário possa ter acesso a todas as informações do aplicativo mesmo sem ter acessado elas enquanto estava online;
- c) disponibilizar o aplicativo na plataforma iOS;
- d) disponibilizar um módulo que permita ao ministrante visualizar quantos visitantes realizaram *check-in* em sua oficina, permitindo saber quantos estavam inscritos na oficina e quantos foram assistir sem se inscrever;
- e) criar uma aplicação para edição do conteúdo do aplicativo que não é importado automaticamente;
- f) criar uma aplicação para cadastrar as notificações que serão disparadas para os usuários do aplicativo.

REFERÊNCIAS

- ALTERMANN, Dennis. **WWW Vs Apps. Navegadores estão sendo substituídos pelos aplicativos?**. [S.l.]: Midiatismo, 2010. Disponível em: <<http://www.midiatismo.com.br/www-vs-apps-navegadores-estao-sendo-substituıda-pelos-aplicativos>>. Acesso em: 04 mar. 2017.
- ANGELI, Flaviano José. **Protótipo de mundo virtual para relacionamento com participantes do Interação FURB**. 2010. 66f. Trabalho de Conclusão de Curso (Bacharelado em Sistemas de Informação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau.
- AYRES, Julia. **O que é experiência do usuário (UX)?**. [S.l.], 2016. Disponível em: <<http://www.tekoa.com.br/blog/experiencia-do-usuario-ux/>>. Acesso em: 30 nov. 2017.
- BARRETO, Filipe Franz. **FURB MOBILE: Uma aplicação para visualização e acompanhamento da matriz curricular**. 2016. 47f. Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau.
- CABRAL, Carlos. **React Native: Construa aplicações móveis nativas com JavaScript**. [S.l.], 2016. Disponível em: <<https://tableless.com.br/react-native-construa-aplicacoes-moveis-nativas-com-javascript/>>. Acesso em: 29 mar. 2017.
- CAELUM. **UX e Usabilidade aplicados em Mobile e Web**. [S.l.], [2016]. Disponível em: <<https://www.caelum.com.br/apostila-ux-usabilidade-mobile-web/>>. Acesso em: 29 mar. 2017.
- CCM/FURB. **Folder Interação FURB**. Elaborado pela equipe Interação - CCM/FURB. Blumenau: FURB, 2016. Recebido por e-mail em: 20 set. 2016.
- FACEBOOK. **React Native**. [S.l.], 2017. Disponível em: <<https://facebook.github.io/react-native/>>. Acesso em: 20 maio 2017.
- GOOGLE. **UX – A experiência do usuário**. [S.l.], 2015. Disponível em: <<https://www.thinkwithgoogle.com/intl/pt-br/articles/ux-user-experience.html>>. Acesso em: 20 maio 2017.
- LI, Sing. **React: crie componentes de UI de alto desempenho, que podem ser mantidas**. [S.l.], 2016. Disponível em: <<https://www.ibm.com/developerworks/br/library/wa-react-intro/>>. Acesso em: 30 maio 2017.
- OCCHINO, Tom. **React Native: Bringing modern web techniques to mobile**. [S.l.], 2015. Disponível em: <<https://code.facebook.com/posts/1014532261909640/react-native-bringing-modern-web-techniques-to-mobile/>>. Acesso em: 21 maio 2017.
- ROCHA, Marcus Otávio. **FURB-MOBILE: Sistema móvel multiplataforma para navegação em rotas internas**. 2016. 61f. Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau.
- SAMPARA, Fernanda. **IAVA – Aplicação web para disponibilizar recursos do AVA no dispositivo móvel iphone**. 2010. 93f. Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau.
- SERPA, Thuany. **Dissecando o Material Design**. [S.l.], 2016. Disponível em: <<https://imasters.com.br/design-ux/dissecando-o-material-design/?trace=1519021197>>. Acesso em: 29 maio 2017.

UNIVERSIDADE REGIONAL DE BLUMENAU. **FURB-MOBILE**. Blumenau, [2017a?]. Disponível em: <http://gcg.inf.furb.br/?page_id=3117>. Acesso em: 04 mar. 2017.

_____. **Interação FURB**. Blumenau, [2017b?]. Disponível em: <<http://www.furb.br/web/1368/relacao-com-a-comunidade/interacao-furb>>. Acesso em: 19 maio 2017.

_____. **Interação FURB**. Blumenau, [2017c?]. Disponível em: <<http://www10.furb.br/interacao/web/4263/interacao-furb/interacao-furb/interacao-furb>>. Acesso em: 04 mar. 2017.

_____. **FURB**: Universidade de Blumenau. Blumenau, [2017d?]. Disponível em: <<http://www.furb.br/web/10/portugues>>. Acesso em: 30 maio 2017.

APÊNDICE A – Questionário do perfil de usuário, lista de tarefas e avaliação de usabilidade

Neste apêndice constam o questionário de perfil dos visitantes, a lista de tarefas para visitantes, o questionário de usabilidade aplicado ao final do experimento com os visitantes, questionário de perfil dos ministrantes, a lista de tarefas para ministrantes e o questionário de usabilidade aplicado ao final do experimento com os ministrantes.

Quadro 14 – Questionário de perfil dos visitantes

PERFIL DOS VISITANTES			
A seguir, é apresentado um questionário para melhor entender o perfil do usuário que está avaliando o aplicativo.			
Sexo:			
<input type="checkbox"/> Feminino	<input type="checkbox"/> Masculino		
Idade:			
<input type="checkbox"/> menos de 18 anos	<input type="checkbox"/> entre 18 e 25 anos		
<input type="checkbox"/> entre 26 e 35 anos	<input type="checkbox"/> mais de 35 anos		
Profissão: _____			
Quando você deseja algo e precisa da aprovação dos seus responsáveis, você normalmente mostra o que você quer através do:			
<input type="checkbox"/> Celular	<input type="checkbox"/> Computador	<input type="checkbox"/> Revista/Jornal	
Qual meio de comunicação você normalmente utiliza para buscar informações?			
<input type="checkbox"/> Celular	<input type="checkbox"/> Computador	<input type="checkbox"/> Jornais	<input type="checkbox"/> Revistas
Se a resposta de algumas das duas perguntas anteriores for Celular, você prefere utilizar:			
<input type="checkbox"/> Navegador	<input type="checkbox"/> Aplicativo específico		
Você utiliza o celular todos os dias?			
<input type="checkbox"/> Sim	<input type="checkbox"/> Não		
Você já acessou o site da FURB?			
<input type="checkbox"/> Sim	<input type="checkbox"/> Não		
Você teve dificuldade para encontrar informações dos cursos no site da FURB?			
<input type="checkbox"/> Sim	<input type="checkbox"/> Não	<input type="checkbox"/> Nunca acessei	
No seu dia a dia, você prefere utilizar meios digitais ou materiais (papeis impressos)?			
<input type="checkbox"/> Materiais	<input type="checkbox"/> Digitais		

Fonte: elaborado pelo autor.

Quadro 15 – Lista de tarefas para visitantes
LISTA DE TAREFAS PARA VISITANTES

A seguir, é apresentada uma lista com 14 tarefas que têm por objetivo avaliar o aplicativo desenvolvido. Procure realizar as tarefas na sequência estabelecida.

1. Visualizar informações sobre a FURB.
A tarefa foi executada?
() Sim () Não. Por quê? _____

2. Visualizar o valor da mensalidade do curso “Engenharia Elétrica”.
A tarefa foi executada?
() Sim () Não. Por quê? _____

3. Visualizar informações de Ingresso.
A tarefa foi executada?
() Sim () Não. Por quê? _____

4. Visualizar informações sobre os tipos de apoios financeiros.
A tarefa foi executada?
() Sim () Não. Por quê? _____

5. Visualizar o processo de matrícula.
A tarefa foi executada?
() Sim () Não. Por quê? _____

6. Visualizar a data em que será realizado o evento Interação FURB.
A tarefa foi executada?
() Sim () Não. Por quê? _____

7. Visualizar horário de uma oficina do evento Interação FURB.
A tarefa foi executada?
() Sim () Não. Por quê? _____

As tarefas a seguir requerem a realização do *login* como visitante no aplicativo, conforme estabelecido na próxima tarefa.

1. Realizar o *login* com um visitante cadastrado.
Utilize o e-mail “visitante@email.com” para realizar o *login* como visitante.
A tarefa foi executada?
() Sim () Não. Por quê? _____

2. Visualizar as oficinas que o visitante está matriculado.
A tarefa foi executada?
() Sim () Não. Por quê? _____

3. Visualizar o QRCode do visitante para realização do *check-in*.
A tarefa foi executada?
() Sim () Não. Por quê? _____

As tarefas a seguir devem ser realizadas com a internet desligada para verificar o funcionamento do acesso *off-line* ao aplicativo.

<p>1. Visualizar o valor da mensalidade do curso “Engenharia Elétrica”. A tarefa foi executada? () Sim () Não. Por quê? _____ _____</p> <p>2. Visualizar as oficinas que o visitante está matriculado. A tarefa foi executada? () Sim () Não. Por quê? _____ _____</p> <p>Para fins de comparação, as 2 tarefas a seguir devem ser realizadas no site da FURB.</p> <p>1. Visualizar o valor da mensalidade do curso “Engenharia Elétrica”. A tarefa foi executada? () Sim () Não. Por quê? _____ _____</p> <p>2. Visualizar informações sobre os tipos de apoios financeiros. A tarefa foi executada? () Sim () Não. Por quê? _____ _____</p>

Fonte: elaborado pelo autor.

Quadro 16 – Questionário de usabilidade para visitantes

QUESTIONÁRIO DE AVALIAÇÃO DO APLICATIVO PARA VISITANTES

A seguir, é apresentado um questionário com o objetivo de avaliar a utilidade do aplicativo, bem como suas funcionalidades.

O aplicativo é visualmente agradável e atraente?

() Sim () Não

A navegação pelo aplicativo é fácil?

() Sim () Não

Os ícones são intuitivos?

() Sim () Não

O aplicativo em algum momento deixou você sem ação?

() Sim () Não

É possível utilizar o aplicativo sem o apoio de outra pessoa?

() Sim () Não

Você utilizaria este aplicativo?

() Sim () Não

Você recomendaria esse aplicativo para outra pessoa?

() Sim () Não

Foi mais fácil encontrar o valor da mensalidade do curso no aplicativo do que no site da FURB?

() Sim () Não

Foi mais fácil encontrar informações sobre apoio financeiro no aplicativo do que no site da FURB?

() Sim () Não

Fonte: elaborado pelo autor.

Qual é a sua opinião sobre o aplicativo quanto ao seu uso e funcionalidades? Fique à vontade para fazer críticas e sugestões.

Quadro 17 – Questionário de perfil do ministrante
PERFIL DOS MINISTRANTES

A seguir, é apresentado um questionário para melhor entender o perfil do usuário que está avaliando o aplicativo.

Sexo:

Feminino Masculino

Idade:

menos de 18 anos entre 18 e 25 anos
 entre 25 e 35 anos mais de 35 anos

Profissão: _____

Quanto tempo você gasta para passar informações referentes ao curso e mensalidade?

5 à 10 minutos 10 à 15 minutos 15 à 20 minutos

Você entrega algum material com informações a respeito do curso?

Sim Não

Você acha interessante ter um aplicativo com estas informações para aumentar o tempo útil da oficina?

Sim Não

Você gostaria que receber feedback das oficinas mediante avaliação dos visitantes?

Sim Não

Você costuma verificar quantos participaram das suas oficinas?

Sim Não

Você já teve dificuldades em saber o que fazer com as listas de presenças com as assinaturas dos visitantes?

Sim Não

Qual meio de comunicação você normalmente utiliza para buscar informações?

Celular Computador Jornais Revistas

Se a resposta da alternativa anterior for Celular, você prefere utilizar:

Navegador Aplicativos específicos

Você utiliza o celular todos os dias?

Sim Não

Você normalmente acessa o site da FURB?

Sim Não

Você já teve dificuldade para encontrar informações dos cursos no site da FURB?

Sim Não Nunca acessei

No seu dia a dia, você prefere utilizar meios materiais (papeis impressos) ou digitais?

Materiais Digitais

Fonte: elaborado pelo autor.

Quadro 18 – Lista de tarefas para ministrantes
LISTA DE TAREFAS PARA MINISTRANTES

A seguir, é apresentada uma lista com 14 tarefas que têm por objetivo avaliar o aplicativo desenvolvido. Procure realizar as tarefas na sequência estabelecida.

1. Visualizar informações sobre a FURB.
A tarefa foi executada?
() Sim () Não. Por quê? _____

2. Visualizar o valor da mensalidade do curso “Engenharia Elétrica”.
A tarefa foi executada?
() Sim () Não. Por quê? _____

3. Visualizar informações de Ingresso.
A tarefa foi executada?
() Sim () Não. Por quê? _____

4. Visualizar informações sobre os tipos de apoios financeiros.
A tarefa foi executada?
() Sim () Não. Por quê? _____

5. Visualizar o processo de matrícula.
A tarefa foi executada?
() Sim () Não. Por quê? _____

6. Visualizar a data em que será realizado o evento Interação FURB.
A tarefa foi executada?
() Sim () Não. Por quê? _____

7. Visualizar horário de uma oficina do evento Interação FURB.
A tarefa foi executada?
() Sim () Não. Por quê? _____

As tarefas a seguir requerem a realização do *login* como ministrante no aplicativo, conforme estabelecido na próxima tarefa.

1. Realizar o *login* com um ministrante de uma oficina.
Utilize o e-mail “ministrante@email.com” para realizar o *login* como ministrante e no segundo campo que será exibido informe “ministrante”.
A tarefa foi executada?
() Sim () Não. Por quê? _____

2. Registrar a presença de um visitante escaneando um QRCode.
A tarefa foi executada?
() Sim () Não. Por quê? _____

3. Visualizar os participantes de suas oficinas.
A tarefa foi executada?
() Sim () Não. Por quê? _____

As tarefas a seguir devem ser realizadas com a internet desligada para verificar o funcionamento do

acesso *off-line* ao aplicativo.

1. Visualizar o valor da mensalidade do curso “Engenharia Elétrica”.

A tarefa foi executada?

Sim Não. Por quê? _____

2. Registrar a presença de um visitante escaneando um QRCode.

A tarefa foi executada?

Sim Não. Por quê? _____

Para fins de comparação, as 2 tarefas a seguir devem ser realizadas no site da FURB.

1. Visualizar o valor da mensalidade do curso “Engenharia Elétrica”.

A tarefa foi executada?

Sim Não. Por quê? _____

2. Visualizar informações sobre os tipos de apoios financeiros.

A tarefa foi executada?

Sim Não. Por quê? _____

Fonte: elaborado pelo autor.

Quadro 19 – Questionário de usabilidade para ministrantes

QUESTIONÁRIO DE AVALIAÇÃO DO APLICATIVO PARA MINISTRANTES

A seguir, é apresentado um questionário com o objetivo de avaliar a utilidade do aplicativo, bem como suas funcionalidades.

O aplicativo é visualmente agradável e atraente?

Sim Não

A navegação pelo aplicativo é fácil?

Sim Não

Os ícones são intuitivos?

Sim Não

O aplicativo em algum momento deixou você sem ação?

Sim Não

É possível utilizar o aplicativo sem o apoio de outra pessoa?

Sim Não

Você utilizaria este aplicativo?

Sim Não

Foi mais fácil encontrar o valor da mensalidade do curso no aplicativo do que no site da FURB?

Sim Não

Foi mais fácil encontrar informações sobre apoio financeiro no aplicativo do que no site da FURB?

Sim Não

Você acha que o uso do QRCode facilita o registro da presença?

Sim Não

Fonte: elaborado pelo autor.

Qual é a sua opinião sobre o aplicativo quanto ao seu uso e funcionalidades? Fique à vontade para fazer críticas e sugestões.
