

**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ**  
**DEPARTAMENTO ACADÊMICO DE INFORMÁTICA**  
**CURSO DE ESPECIALIZAÇÃO EM DESENVOLVIMENTO**  
**PARA DISPOSITIVOS MÓVEIS**

**MAX LUIZ DE CARVALHO**

**PROTÓTIPO DE APLICATIVO PARA GESTÃO DE EVENTOS ACADÊMICOS /**  
**CIENTÍFICOS**

**MONOGRAFIA DE ESPECIALIZAÇÃO**

**CURITIBA**

**2017**

**MAX LUIZ DE CARVALHO**

**PROTÓTIPO DE APLICATIVO PARA GESTÃO DE EVENTOS ACADÊMICOS /  
CIENTÍFICOS**

**Monografia apresentada como requisito parcial  
à obtenção do título de Especialista em  
Desenvolvimento para Dispositivos Móveis, do  
Departamento Acadêmico de Informática, da  
Universidade Tecnológica Federal do Paraná,  
Campus Curitiba.**

**Orientador: Prof. Robson Ribeiro Linhares**

**CURITIBA**

**2017**



Ministério da Educação  
**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ**  
**Câmpus Curitiba**  
Diretoria de Pesquisa e Pós-Graduação  
*Departamento Acadêmico de Informática*  
*Coordenação do Curso de Especialização em Desenvolvimento  
para Dispositivos Móveis*

## **TERMO DE APROVAÇÃO**

### **“Protótipo de Aplicativo para Gestão de Eventos Acadêmicos / Científicos”**

por

**“Max Luiz de Carvalho”**

Este Trabalho de Conclusão de Curso foi apresentado às 18:35 do dia 22 de novembro de 2017 na sala B201 como requisito parcial à obtenção do grau de Especialista em Desenvolvimento para Dispositivos Móveis na Universidade Tecnológica Federal do Paraná - UTFPR - Campus Curitiba. O(a) aluno(a) foi arguido pela Banca de Avaliação abaixo assinados. Após deliberação, a Banca de Avaliação considerou o trabalho aprovado.

|   |  |
|---|--|
| <hr/> <b>Prof. Robson Ribeiro Linhares<br/>(Presidente - UTFPR/Curitiba)</b>                | <hr/> <b>Prof. Paulo Mauricio de Lucchi Bordin<br/>(Avaliador 1 – Externo)</b> |
| <hr/> <b>Prof. Maria Claudia Figueiredo Pereira Emer<br/>(Avaliador 2 – UTFPR/Curitiba)</b> |  |

“A Ata de Aprovação assinada encontra-se na Coordenação do Curso.”

## **AGRADECIMENTOS**

Agradeço a Deus, pela força cedida para superar dificuldades e buscar constante evolução.

Aos amigos e à minha família, pelo suporte e compreensão nos momentos de ausência.

Ao Professor Robson Ribeiro Linhares, pela orientação, confiança e apoio na realização e conclusão deste trabalho.

Aos demais professores e colaboradores da UTFPR, por todo o comprometimento em entregar ensino de qualidade e elevar cada vez mais o conceito desta instituição de ensino e pesquisa.

Agradeço a todos que direta ou indiretamente prestaram apoio na realização deste trabalho.

## RESUMO

A rotina do dia-a-dia tem sofrido alterações por conta do notório crescimento do uso de dispositivos móveis e da influência do grande número de aplicações que prometem simplificar as tarefas cotidianas. A rotina diária inclui um smartphone como companheiro inseparável e os benefícios dessa relação dependem do desenvolvimento de aplicações úteis na rotina de seus usuários. Na organização de eventos ou congressos acadêmicos, a estratégia de divulgação pode ser o fator essencial de sucesso de forma a manter ou elevar seu conceito junto ao público-alvo. Neste contexto, este trabalho apresenta a especificação e desenvolvimento de um protótipo de aplicativo móvel multiplataforma (sistemas Android e IOS), utilizando o *framework* Cordova, para o gerenciamento de eventos sociais com finalidade acadêmica / científica. Por meio da aplicação proposta, os participantes podem obter informações atualizadas acerca da programação, palestrantes, local do evento além de receber notificações do tipo *push*, facilitando o fluxo de informações garantindo a atualização constante de todas as informações relevantes do evento. Na análise de usabilidade proposta, os participantes executaram com sucesso todas as tarefas propostas, em tempo adequado. O questionário de usabilidade SUS aplicado para esses usuários obteve pontuação média de 92,5, indicando uma interação satisfatória.

Palavras-chave: aplicações móveis; aplicações híbridas; framework Cordova; gerenciamento de eventos; dispositivos móveis.

## **ABSTRACT**

The day-to-day routine has undergone changes due to the notorious growth of the use of mobile devices and the influence of the large number of applications that promise to simplify everyday tasks. The daily routine includes a smartphone as an inseparable companion and the benefits of this relationship depend on the development of useful applications in the routine of its users. During the process of organizing events or academic conferences, the dissemination strategy can be the essential success factor in order to maintain or raise its standard with the target audience. In this context, this work presents the specification and development of a multiplatform mobile application prototype (Android and IOS systems), using the Cordova framework, for the management of social events for academic / scientific purposes. Through the proposed application, participants can get up-to-date information about the schedule, speakers, location of the event in addition to receiving push notifications, facilitating the flow of information ensuring constant updating of all relevant event information. In the proposed usability analysis, participants successfully performed all proposed tasks in a timely manner. The SUS usability questionnaire applied to these users obtained a mean score of 92.5, indicating a satisfactory interaction.

**Keywords:** mobile applications; hybrid applications; Cordova framework; event management; mobile devices.

## LISTA DE FIGURAS

|  |    |
|--|----|
| Figura 1 - Solicitação de uma página Web - Adaptado de: (DEVMEDIA, 2017) .....   | 18 |
| Figura 2 - Diagrama conceitual da arquitetura REST - Adaptado de: (PALIARI, 2017)<br>.....   | 20 |
| Figura 3 - Estrutura básica da notação JSON (Fonte: ZEMEL, 2011).....  | 20 |
| Figura 4 - Arquitetura típica de um aplicativo criado com o <i>framework</i> Apache<br>Cordova (versão 3) – Fonte: (WARGO, 2013) ..... | 22 |
| Figura 5 - Camadas no <i>framework</i> Cordova – Fonte: (JÚNIOR et al., 2012) .....  | 23 |
| Figura 6 - Ranking de classificação - SUS SCORE (BROOKE, 2013).....  | 26 |
| Figura 7 – Tela da aplicação “MED eventos”. Fonte: (UFENESP, 2017) .....   | 26 |
| Figura 8 – Tela da aplicação “32ª Reunião Anual SBPqO”. Fonte: (DEFINITY TEAM,<br>2017) .....  | 27 |
| Figura 9 – Visão arquitetural: modelo de funcionamento do aplicativo. Fonte: Autoria<br>própria .....                                  | 30 |
| Figura 10 - Tela principal.....  | 31 |
| Figura 11 - Tela de login .....  | 31 |
| Figura 12 – Tela de listagem de atividades com filtro .....  | 31 |
| Figura 13 – Tela de listagem de palestrantes.....  | 31 |
| Figura 14 – Tela de listagem de notícias.....  | 32 |
| Figura 15 – Tela de gerenciamento de favoritos .....   | 32 |
| Figura 16 - Diagrama de casos de uso: <i>front end e interação com back end</i> .....  | 32 |
| Figura 17 - Diagrama de casos de uso: <i>back end</i> da aplicação .....   | 33 |
| Figura 18 - Diagrama de implantação da aplicação proposta .....  | 49 |
| Figura 19 - DER - Base de dados MySQL .....  | 50 |
| Figura 20 - Diagrama de sequência: Sincronizar dados com o <i>web service</i> .....  | 51 |
| Figura 21 - Diagrama de sequência - Logar no sistema.....  | 52 |
| Figura 22 - Diagrama de sequência - Editar perfil.....   | 53 |
| Figura 23 - Diagrama de sequência - Listar atividades .....  | 54 |
| Figura 24 - Diagrama de sequência - Listar artigos externos .....  | 54 |
| Figura 25 - Diagrama de sequência - Marcar atividades na lista de favoritos .....  | 55 |
| Figura 26 - Diagrama de sequência - Listar palestrantes.....   | 55 |
| Figura 27 - Diagrama de sequência - Exibir credencial.....   | 56 |
| Figura 28 - Diagrama de atividades: Sincronizar informações com o <i>web service</i> ....  | 57 |

|   |    |
|---|----|
| Figura 29 – Graduação divisão em percentis dos escores (SAURO, 2011b).....                              | 60 |
| Figura 30 - <i>Back end</i> : tela de acesso .....  | 62 |
| Figura 31 - <i>Back end</i> : tela inicial .....  | 62 |
| Figura 32 - <i>Back end</i> : gerenciamento de atividades .....   | 63 |
| Figura 33 - Aplicação móvel: tela inicial .....   | 64 |
| Figura 34 - Aplicação móvel: menu lateral .....   | 64 |
| Figura 35 - Aplicação móvel: Página "sobre o evento" .....  | 64 |
| Figura 36 - Aplicação móvel: programa do evento .....   | 64 |
| Figura 37 - Aplicação móvel: filtrar categorias.....  | 64 |
| Figura 38 - Aplicação móvel em execução no emulador do dispositivo iPhone 5S<br>(sistema IOS 9.2) ..... | 65 |



## LISTA DE TABELAS

|  |    |
|--|----|
| Tabela 1 - Requisitos funcionais da aplicação proposta .....   | 28 |
| Tabela 2 - Requisitos não-funcionais da aplicação proposta .....   | 29 |
| Tabela 3 - Questionário de usabilidade baseado no SUS .....  | 58 |
| Tabela 4 - Índice de sucesso observado na execução das tarefas (0: tarefa não<br>completada, 1: tarefa completada com dificuldade e 2: tarefa completada sem<br>dificuldade) ..... | 59 |
| Tabela 5 - Pontuação no questionário de usabilidade (SUS) .....  | 60 |

## **LISTA DE ABREVIATURAS, SIGLAS E ACRÔNIMOS**

**API – Application Programming Interface**

**CRUD – Create, retrieve, update and delete**

**CSS – Cascading Style Sheets**

**GCM – Google Cloud Messaging**

**HTML – Hypertext Markup Language**

**HTTP – Hyper Text Transfer Protocol**

**JSON – Java Script Object Notation**

**PHP – Hypertext Preprocessor**

**QR – Quick Response**

**REST – Representational State Transfer**

**SGBD – Sistema Gerenciador de Banco de Dados**

**SUS – System Usability Scale**

**UML – Unified Modeling Language**

**URI – Uniform Resource Identifier**

## SUMÁRIO

|          |  |           |
|----------|--|-----------|
| <b>1</b> | <b>INTRODUÇÃO</b>                      | <b>13</b> |
| 1.1      | CONTEXTUALIZAÇÃO                       | 13        |
| 1.2      | MOTIVAÇÃO                              | 14        |
| 1.3      | OBJETIVOS                              | 14        |
| 1.3.1    | Objetivo Geral                         | 14        |
| 1.3.2    | Objetivos Específicos                  | 14        |
| 1.4      | METODOLOGIA                            | 15        |
| 1.5      | ORGANIZAÇÃO DO TRABALHO                | 15        |
| <b>2</b> | <b>FUNDAMENTAÇÃO TEÓRICA</b>           | <b>17</b> |
| 2.1      | REDES SOCIAIS E EVENTOS                | 17        |
| 2.2      | TECNOLOGIAS                            | 17        |
| 2.2.1    | Tecnologia WEB                         | 18        |
| 2.2.2    | Linguagem de marcação HTML             | 18        |
| 2.2.3    | Linguagem de Programação PHP           | 18        |
| 2.2.4    | <i>Web service</i>                     | 19        |
| 2.2.5    | Arquitetura REST                       | 19        |
| 2.2.6    | Notação JSON                           | 20        |
| 2.2.7    | Google Cloud Messaging (GCM)           | 20        |
| 2.2.8    | Banco de dados                         | 21        |
| 2.2.9    | <i>Framework Apache Cordova</i>        | 21        |
| 2.2.10   | Linguagem UML                          | 23        |
| 2.3      | USABILIDADE                            | 24        |
| 2.3.1    | Avaliação de Usabilidade de Interfaces | 24        |
| 2.3.2    | SUS (System Usability Scale)           | 25        |
| 2.4      | APLICATIVOS SIMILARES                  | 26        |
| 2.4.1    | MED eventos                            | 26        |
| 2.4.2    | Aplicativo da 32ª Reunião Anual SBPqO  | 27        |
| <b>3</b> | <b>DESENVOLVIMENTO DA APLICAÇÃO</b>    | <b>28</b> |
| 3.1      | REQUISITOS                             | 28        |
| 3.1.1    | Requisitos Funcionais                  | 28        |
| 3.1.2    | Requisitos não-funcionais              | 29        |
| 3.2      | VISÃO GERAL DA ARQUITETURA             | 29        |

|          |   |           |
|----------|---|-----------|
| 3.3      | MODELAGEM.....  | 30        |
| 3.3.1    | Prototipação .....  | 30        |
| 3.3.2    | Diagramas de Casos de Uso .....   | 32        |
| 3.3.3    | Especificações de Casos de Uso da Aplicação Móvel ( <i>front end</i> )..... | 33        |
| 3.3.4    | Especificações de Casos de Uso do <i>Web Service (back end)</i> .....       | 42        |
| 3.3.5    | Diagrama de Implantação.....  | 49        |
| 3.3.6    | Diagrama Entidade Relacionamento .....                                      | 50        |
| 3.3.7    | Diagramas de Sequência .....  | 51        |
| 3.3.8    | Diagrama de Atividades.....   | 56        |
| 3.3.9    | Análise de Usabilidade .....  | 57        |
| 3.3.9.1  | Resultados da análise de usabilidade.....                                   | 58        |
| 3.3.9.2  | Resultado do questionário de usabilidade (SUS).....                         | 60        |
| 3.4      | APRESENTAÇÃO DA APLICAÇÃO .....   | 61        |
| 3.4.1    | <i>Back End</i> da Aplicação.....   | 61        |
| 3.4.2    | Aplicação Móvel.....  | 63        |
| <b>4</b> | <b>CONCLUSÃO.....</b>   | <b>66</b> |
|          | <b>REFERÊNCIAS.....</b>   | <b>68</b> |
|          | <b>APÊNDICE A – QUESTIONÁRIO DE AVALIAÇÃO DE USABILIDADE .....</b>          | <b>71</b> |

# 1 INTRODUÇÃO

## 1.1 CONTEXTUALIZAÇÃO

Com o notório crescimento do uso de dispositivos móveis, em todo o mundo, a rotina do dia-a-dia tem sofrido alterações por conta da influência do grande número de aplicações que prometem simplificar as tarefas cotidianas. Desde simples aplicações de listas de compras até complexos sistemas de gerenciamento de empresas, a informática está cada vez mais íntima do ser humano, companheira inseparável de milhares de pessoas.

Os dispositivos móveis figuram como um canal ágil de comunicação global, pois com o uso da internet, seus usuários são capazes de interagir com milhares de pessoas em tempo real e em qualquer local que disponha de conexão. A rotina diária inclui um smartphone como companheiro inseparável e os benefícios dessa relação dependem do desenvolvimento de aplicações úteis no cotidiano de seus usuários, seja para entretenimento ou trabalho (SACCOL; REINHARD, 2007).

O mercado dispõe de diversas plataformas móveis, entre elas: Android, IOS, Windows Phone, etc. O sucesso e a disseminação de um aplicativo podem estar ligados diretamente à disponibilização da informação e interação, independentemente do ponto de acesso, além da sua usabilidade e desempenho.

Um sistema multiplataforma atinge um público maior, diluindo custos de equipes além de possibilitar o desenvolvimento em uma única linguagem de programação (FREIRE; RIBEIRO, 2013). Para a elaboração de sistemas multiplataformas, os desenvolvedores podem se utilizar de ferramentas para acelerar o processo, os *frameworks*. Estas ferramentas, a partir de uma aplicação semi completa, permitem a criação de aplicativos customizados (FAYAD; SCHMIDT, 1997).

O recurso de comunicação propiciado pelos dispositivos móveis pode funcionar como um agregador de informações e interação entre usuários para gerenciar eventos sociais tais como: eventos científicos, reuniões familiares, reuniões de negócio, entre outros.

## 1.2 MOTIVAÇÃO

Eventos acadêmicos figuram como importante instrumento na divulgação e propagação do conhecimento bem como na promoção social e cultural de uma instituição de ensino. A estratégia de divulgação pode ser o fator essencial no sucesso de um evento científico, de forma a manter ou elevar seu conceito junto ao público-alvo.

A correta estruturação das atividades envolvidas no evento deve prever a constante atualização do cronograma além de garantir que todas as alterações no planejamento sejam prontamente comunicadas a todos os envolvidos, principalmente os participantes. Substituições de local de realização ou alteração de horário podem gerar baixa audiência se não comunicadas tempestivamente. Essa interação de informações, em tempo real, é de difícil manejo quando da ausência de ferramentas automatizadas para tal. A fim de atender à demanda de eventos científicos realizados por instituições de ensino, tais como: semanas acadêmicas, jornadas, workshops, pretende-se com o aplicativo em questão facilitar a interação e disseminação da informação dentre todos os participantes e organizadores, mesmo em diferentes plataformas. Com o envio de notificações do tipo *push*, facilita-se o fluxo de informações com os participantes e fica garantida a atualização constante de todas as informações relevantes do evento.

## 1.3 OBJETIVOS

### 1.3.1 Objetivo Geral

O objetivo deste trabalho é desenvolver um protótipo de aplicativo móvel multiplataforma para o gerenciamento de eventos sociais com finalidade acadêmica / científica.

### 1.3.2 Objetivos Específicos

- Estudo das plataformas envolvidas e o processo de desenvolvimento multiplataforma;
- Aplicação de análise sobre o problema e posterior planejamento;

- Estudo de interfaces simplificadas para disseminação de informação entre usuários;
- Realização de testes de usabilidade a fim de avaliar o grau de complexidade de uso da aplicação proposta bem como pontos positivos e negativos.

#### 1.4 METODOLOGIA

O início do presente trabalho deu-se pela revisão da literatura relacionada ao uso de dispositivos móveis como ferramenta na disseminação ágil de informações e facilitação da organização e gestão de eventos.

Em uma segunda etapa foi realizado um estudo das tecnologias envolvidas no desenvolvimento de aplicações móveis híbridas, tais como: linguagens de programação para o cliente e servidor, notação para tráfego de dados, banco de dados relacional, entre outras. Em seguida foi realizada a fase de projeto e análise, com a definição da arquitetura do sistema e a delimitação dos requisitos e funcionalidades desejados, além da elaboração de um protótipo de baixa fidelidade com a função de auxiliar no detalhamento da interface de interação com os usuários.

Após a codificação do *front end e back end*, foi realizada uma análise de usabilidade a partir da coleta de dados e da observação durante a execução de uma lista de tarefas que envolvia as principais funcionalidades da aplicação. Três usuários foram convidados a participar da referida avaliação em um protótipo de alta fidelidade. Um questionário baseado na escala SUS (*System Usability Survey*) foi aplicado aos usuários. Procedeu-se à análise dos resultados obtidos com o questionário e o teste de usabilidade.

#### 1.5 ORGANIZAÇÃO DO TRABALHO

O presente trabalho está organizado em 4 capítulos. O capítulo 2 apresenta a fundamentação teórica que embasa o protótipo, incluindo as tecnologias e conceitos que sustentam a proposta. No capítulo 3 é apresentada a modelagem e metodologia utilizados no processo de desenvolvimento do aplicativo, além da exposição das telas do sistema e dos resultados da análise de usabilidade.

Por fim, o capítulo 4 apresenta as considerações finais acerca do desenvolvimento do aplicativo móvel proposto bem como a discussão sobre futuras implementações que possam agregar maior valor ao sistema.



## **2 FUNDAMENTAÇÃO TEÓRICA**

### **2.1 REDES SOCIAIS E EVENTOS**

Na sociedade moderna, as redes sociais têm adquirido grande relevância por sua crescente utilização e facilidade na integração e conexão entre pessoas. Tais redes de relacionamento permitem que diversos grupos sociais se integrem socialmente, conectando e sustentando relacionamentos entre os atores (RECUERO, 2009). Uma rede social consiste no conjunto de relações entre pessoas agrupadas e as suas interações. Dentre os pontos positivos do seu uso, podemos destacar a democratização e compartilhamento de informações, conhecimento e interesses entre os atores envolvidos, além do auxílio na análise de perfis e comportamentos a partir das relações estabelecidas. O gerenciamento de eventos a partir de grandes redes sociais figura como uma ferramenta eficaz de marketing digital, propiciando uma ampla divulgação a partir das conexões entre usuários.

Em um contexto de pequenos grupos sociais, a divulgação das informações pode ser propositalmente limitada a poucos usuários, de forma a restringir a propagação de mensagens apenas aos envolvidos em um evento, por exemplo. Neste cenário, a interação entre os usuários pode influenciar o número de participantes e o planejamento na medida em que a disseminação da informação pode ou não viabilizar o evento em questão.

Em um evento acadêmico é importante possibilitar aos envolvidos a disseminação ágil da informação no que diz respeito à constante atualização dos detalhes das atividades propostas bem como na possibilidade de programação e organização da participação dentro dos interesses individuais. O envio de notificações instantâneas por meio de dispositivos móveis figura como meio cômodo, econômico e eficaz na divulgação de cronogramas, destaques e promoções relacionadas ao evento em questão.

### **2.2 TECNOLOGIAS**

Esta seção propõe-se a apresentar uma visão conceitual das tecnologias e ferramentas utilizadas bem como a arquitetura proposta no desenvolvimento deste trabalho.

### 2.2.1 Tecnologia WEB

A internet representa um sistema global de redes de computadores interligados utilizando protocolos de comunicação padronizados. Através da World Wide Web, WWW ou simplesmente Web, diversos recursos são disponibilizados através de uma grande variedade de tecnologias relacionadas, tais como: apresentação de hipertextos, servidores de e-mail, etc. A tecnologia Web está baseada no paradigma cliente-servidor, modelo de computação no qual o processamento é dividido, conforme o nome sugere, entre clientes e servidores. Um cliente realiza uma requisição e o servidor, após sua interpretação, gera uma resposta com informações através da criação ou recuperação de dados, conforme ilustra a Figura 1 (DEV MEDIA, 2017).



Figura 1 - Solicitação de uma página Web - Adaptado de: (DEV MEDIA, 2017)

### 2.2.2 Linguagem de marcação HTML

Hyper Text Markup Language (HTML) é uma linguagem de marcação utilizada para apresentação de documentos comumente distribuídos em redes de computadores, criada com especificações formais em meados de 1990. Permite a estruturação de documentos com cabeçalhos e referências com a finalidade de padronizar a exibição de documentos nos softwares clientes (navegadores).

### 2.2.3 Linguagem de Programação PHP

Hypertext Preprocessor (PHP) é uma linguagem de script criada em 1995 por Rasmus Lerdorf. Possui código aberto e é amplamente utilizada no desenvolvimento de aplicações web, com sintaxe similar à linguagem C. Foi criada com o objetivo

primário de gerar conteúdo dinâmico para páginas da internet. Em uma página HTML é possível embutir códigos PHP que serão executados ao se exibir a página. Tais códigos são interpretados pelo servidor Web que gera uma saída HTML (THONSON; WELLING, 2005).

A linguagem PHP possui uma curva de aprendizagem baixa, com facilidade na geração de conteúdo HTML ou mesmo inserida dentro deste. Dentre as vantagens da linguagem, podemos citar: manipulação de arquivos, geração de imagens dinâmicas para validação de formulários, criptografia de dados, definição de cookies e sessões, definição de interfaces para webservices, manipulação de arquivos XML, suporte a vários bancos de dados com acesso nativo (DB2, Firebird, MySQL, Oracle, etc.) (MELO; NASCIMENTO, 2007).

#### 2.2.4 *Web service*

Trata-se de uma aplicação autocontida, identificada por um URI (Uniform Resource Identifier), cujas interfaces e ligações são definidas em artefatos baseados em linguagem XML (Extensible Markup Language) (AUSTIN et al., 2004). O *web service* é uma aplicação capaz de interagir com outros sistemas informatizados através da troca de mensagens utilizando protocolos de comunicação. A partir do seu uso, sistemas podem consumir e agregar serviços disponibilizados em servidores independentes numa única aplicação.

Para o trabalho em questão, foi desenvolvido um *web service* na linguagem PHP para a comunicação com o dispositivo móvel (cliente) e o SGBD.

#### 2.2.5 Arquitetura REST

Desenvolvida em 2000 por Roy Fielding, REST - acrônimo de *Representation State Transfer* (transferência de estado representacional em português) - representa uma arquitetura baseada na observação das aplicações web como um conjunto de recursos, que representam um estado particular de um aplicativo. Ao acessar um recurso é transferido o estado (conteúdo) ou mesmo uma solicitação de alteração neste estado. Através dos métodos disponíveis pelo protocolo HTTP (PUT, GET, DELETE, entre outros) são realizadas as ações (FIELDING, 2000; RONDON, 2010). A Figura 2 ilustra o conceito típico da arquitetura REST.

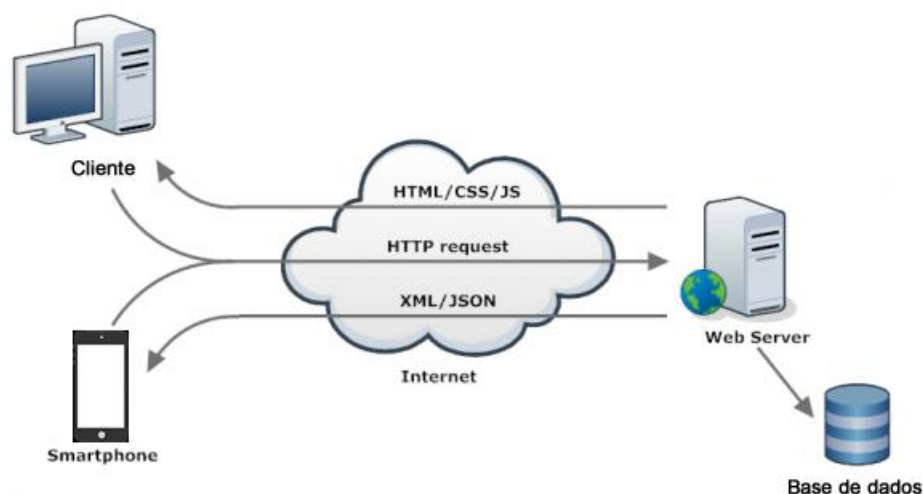


Figura 2 - Diagrama conceitual da arquitetura REST - Adaptado de: (PALIARI, 2017)

### 2.2.6 Notação JSON

A notação JSON consiste em um formato de texto utilizado para facilitar a troca de informações estruturadas, o JSON (*Javascript Object Notation*) é bastante utilizado em alternativa ao XML para trafegar dados em redes de computadores por apresentar facilidade de uso bem como suporte dentre os diferentes navegadores clientes. Dentre suas vantagens de uso, podemos destacar: sintaxe simples e de fácil implementação, tamanho menor dos arquivos, facilidade de desenvolvimento, alto desempenho, entre outros (ZEMEL, 2011).

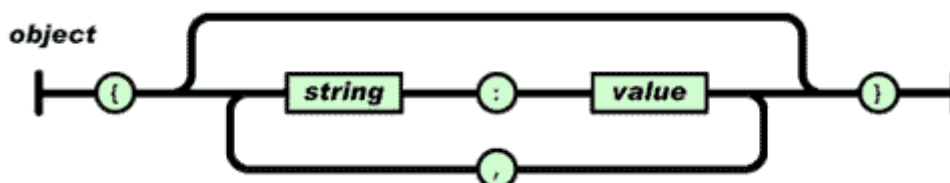


Figura 3 - Estrutura básica da notação JSON (Fonte: ZEMEL, 2011)

### 2.2.7 Google Cloud Messaging (GCM)

Trata-se de um serviço gratuito da empresa Google para troca de mensagens entre servidores e aplicações multiplataforma, como Android, iOS e Google Chrome. O GCM provê um mecanismo simples que permite aos servidores informarem a aplicações móveis para se conectarem e atualizar dados disponíveis. O serviço se

responsabiliza por todos os aspectos que envolvem a solicitação bem como a entrega dos dados à aplicação instalada em um dispositivo móvel. O GCM tem a capacidade de enviar uma mensagem informando à aplicação Android que novos dados devem ser obtidos a partir do servidor. As aplicações instaladas no dispositivo não precisam estar em execução para receber as mensagens, pois o serviço solicita ao sistema operacional que esta operação seja realizada (GOOGLE DEVELOPERS, 2017).

#### 2.2.8 Banco de dados

Um banco de dados consiste no conjunto de dados integrados com a finalidade de atender a uma aplicação ou comunidade de usuários. O gerenciamento de grandes quantidades de dados é facilitado pelos sistemas de gerência de dados –um *software* que agrega as funções de definição, recuperação e alteração de dados no contexto do banco de dados (HEUSER, 2009).

Os sistemas computadorizados de bancos de dados provêm um nível de abstração das informações armazenadas, omitindo detalhes de armazenamento que não são comumente necessários pelos usuários quando da utilização na requisição dos dados. O banco de dados utilizado no presente trabalho foi modelado conforme a representação contida no diagrama de entidade e relacionamento ilustrado na figura 16. O sistema gerenciador de banco de dados selecionado foi o MySQL, sistema gerenciador que utiliza a linguagem SQL e uma arquitetura cliente-servidor. Suas principais características são portabilidade, execução multithreads, velocidade otimizada no acesso aos dados, conexões criptografadas no tráfego de senhas, stored procedures, triggers, visões, integridade referencial, etc. (MILANI, 2007).

#### 2.2.9 *Framework* Apache Cordova

Trata-se de um *framework* livre e de código aberto que possibilita o acesso a funções nativas de um dispositivo móvel a partir da linguagem de programação *Javascript*. Aliado a outros *frameworks* como *jQuery Mobile*, *Dojo Mobile*, entre outros, possibilita o desenvolvimento de aplicações híbridas (combinação de tecnologias nativas e de aplicações web) em HTML, CSS e *Javascript* (THE APACHE SOFTWARE FOUNDATION, 2012). O *framework* Cordova possui como

componentes: um código fonte para cada dispositivo suportado, uma suíte de APIs que provêm uma aplicação web com acesso às funcionalidades nativas do dispositivo e, um conjunto de ferramentas para gestão do processo de criação de projetos de aplicações, gerenciando *plugins*, gerando aplicações nativas e possibilitando simulação ou emulação das aplicações. As plataformas suportadas, atualmente, são: Android (Google), bada (Samsung), BlackBerry 10 (BlackBerry), iOS (Apple), Firefox OS, Tizen, Windows Phone 7 e Windows Phone 8 (Microsoft) e Windows 8 (Microsoft) (WARGO, 2013).

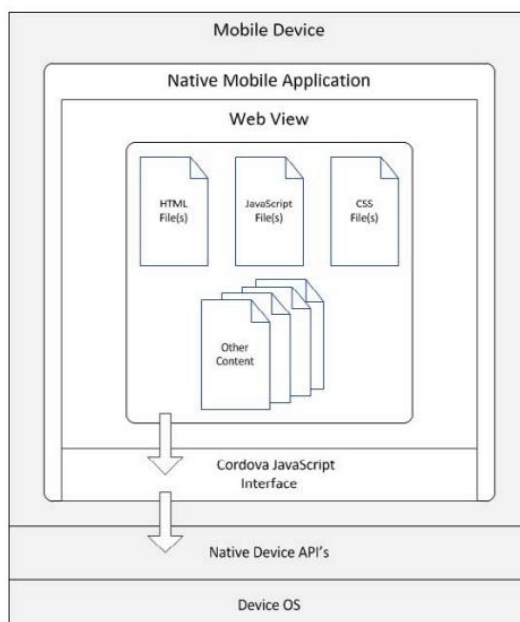


Figura 4 - Arquitetura típica de um aplicativo criado com o *framework* Apache Cordova (versão 3) – Fonte: (WARGO, 2013)

As aplicações *Cordova* são construídas a partir de tecnologias web (HTML, CSS e *Javascript*) com funções adicionais providas pelas bibliotecas especializadas no *framework* (WARGO, 2013). O uso do padrão web confere às aplicações vantagens como o menor custo para desenvolvimento além do acesso multiplataforma. O processo de criação de aplicações envolve o empacotamento na plataforma nativa a partir da construção dos pacotes com as ferramentas proprietárias. Aplicações híbridas, por conta de sua natureza de desenvolvimento, tendem a ser mais lentas que aplicações nativas. O conjunto de APIs disponibilizadas pelo *framework* acessa funcionalidades que são comuns nas diferentes plataformas, as quais incluem:

- Eventos de ciclo de vida do aplicativo;
- Armazenamento em banco de dados local;
- Gerenciamento de contatos;
- Câmera;
- Geolocalização;
- Bússola;
- Acelerômetro.

As funcionalidades são expostas como uma API *Javascript*, disponíveis para uso no código da aplicação. Todo o processamento referente a correspondência na implementação nativa é realizada pelo *framework* nos diferentes sistemas operacionais suportados (JÚNIOR et al., 2012), conforme Figura abaixo:

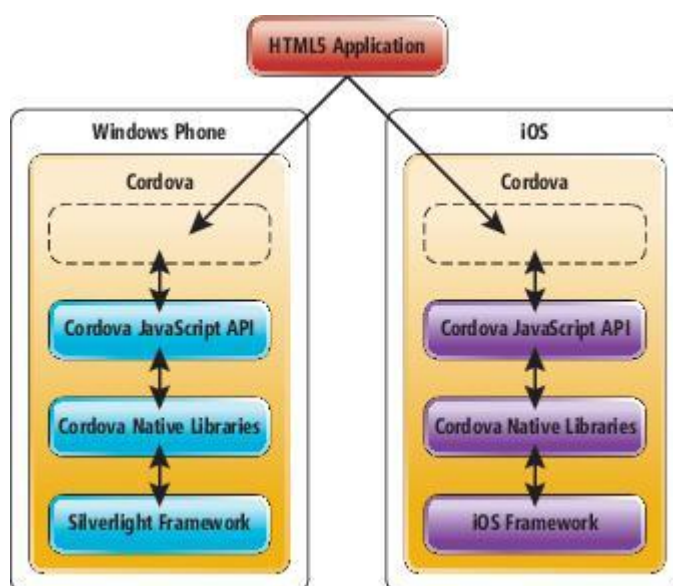


Figura 5 - Camadas no *framework* Cordova – Fonte: (JÚNIOR et al., 2012)

### 2.2.10 Linguagem UML

A UML (*Unified Modeling Language*), originada em 1990 a partir do trabalho de vários desenvolvedores, compreende uma linguagem para visualização, especificação, construção e documentação de artefatos de sistemas de software. Seu objetivo é disponibilizar, de forma padronizada, a apresentação de planos de arquitetura de projetos de sistemas (FURLAN, 1998). Esta linguagem é constituída de vários elementos gráficos que permitem construir diagramas a fim de representar sistemas de softwares sob diversas perspectivas de visualização, permitindo uma

maior compreensão do projeto por parte de todos os envolvidos no projeto além de facilitar a visualização dos requisitos bem como avaliar se estes podem ser atingidos dentro das metas estabelecidas (MELO; NASCIMENTO, 2007).

No presente projeto a linguagem UML foi aplicada na elaboração dos diagramas de casos de uso para representar as necessidades da aplicação, diagrama de implantação, diagrama entidade-relacionamento, diagramas de sequência, diagramas de atividades e especificações de casos de uso descrevendo as funcionalidades disponibilizadas aos usuários.

## 2.3 USABILIDADE

O termo usabilidade tem origem na década de 80 em substituição à expressão *user-friendly* (comumente utilizada nas áreas de psicologia e ergonomia). Podemos conceituar usabilidade como um fator que garante ao usuário facilidade de uso. Alguns aspectos que estão intimamente ligados à usabilidade são: condições eficazes de trabalho e eficiência na execução de uma tarefa (NIELSEN; LORANGER, 2007).

A avaliação da usabilidade pode garantir a continuidade e a afirmação competitiva de um sistema de informação, pois a partir da interação com o usuário e sua percepção de desempenho e satisfação evidenciamos seu grau de sucesso. Testes de usabilidade são úteis na investigação de questões que envolvem a navegação e entendimento da interface de interação com o usuário. Nesse contexto, Nielsen indica que um sistema interativo adequado deve possuir cinco fatores: facilidade de aprendizado, eficiência de uso, facilidade de memorização, suporte a erros e satisfação do usuário (COSTA; RAMALHO, 2010).

### 2.3.1 Avaliação de Usabilidade de Interfaces

Trata-se de um processo avaliativo para obtenção de informações sobre a usabilidade com a finalidade de estimar a qualidade de uso de um software além de possibilitar a identificação de problemas de interação ou de interface. A avaliação pode ser realizada sob diferentes dimensões: segundo a etapa do ciclo de design do software (durante o desenvolvimento ou em produtos finalizados), técnica para coleta de dados (entrevistas, questionários, entre outros), tipos de dados coletados (quantitativos ou qualitativos) e tipo de análise (prever problemas ou interpretação de



dados). As avaliações formativas são realizadas durante o processo de desenvolvimento enquanto as avaliações somativas são feitas em produtos já terminados, buscando verificar conformidade com um padrão estabelecido pelo desenvolvedor (PRATES; BARBOSA, 1991).

### 2.3.2 SUS (System Usability Scale)

A escala SUS foi desenvolvida por John Brooke em 1986 para avaliar interfaces de interação, a partir de uma escala de usabilidade, baseada nos seguintes critérios (BROOKE, 2013):

- Efetividade: os objetivos são alcançados;
- Eficiência: medição do esforço e recurso para alcançar os objetivos;
- Satisfação: índice de satisfação do usuário.

O respondente do questionário assinala as respostas em uma escala *Likert* - escala de resposta psicométrica usada habitualmente em pesquisas de opinião no qual os perguntados especificam seu nível de concordância com uma afirmação – e a partir de um cálculo realizado com as repostas obtidas, obtemos um valor de usabilidade que representa uma medida composta da usabilidade geral do sistema. Para calcular a pontuação subtrai-se o valor 1 das repostas obtidas nas perguntas ímpares e para as perguntas pares o valor respondido é subtraído de 5. Ao final somam-se todos os valores convertidos e o resultado é multiplicado por 4, obtendo-se a pontuação final que pode variar de 0 a 100. Essa pontuação é avaliada a partir do valor médio de 68 pontos (valor médio encontrado em mais de 500 estudos com mais de 5000 participantes envolvidos), considerando os resultados abaixo ou acima dessa média, conforme ranking de classificação ilustrado na Figura 6 (SAURO, 2011a; TEIXEIRA, 2015).

O questionário de usabilidade SUS permite obter uma avaliação rápida e de baixo custo, mesmo em pequenas amostras, da percepção da qualidade de interação a que os usuários estão submetidos quando do uso da aplicação / sistema em teste. A literatura menciona essa metodologia como uma das métricas mais populares para quantificar a satisfação do sistema, com muitos estudos acerca do seu uso para avaliar a facilidade de uso em uma ampla gama de produtos on-line e off-line (TRYMYUI, 2017).

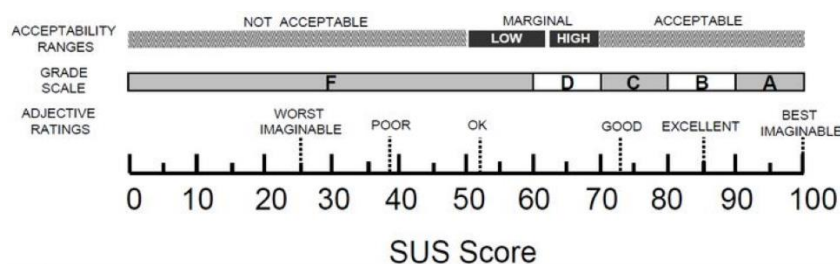


Figura 6 - Ranking de classificação - SUS SCORE (BROOKE, 2013)

## 2.4 APLICATIVOS SIMILARES

Na busca de aplicações com finalidade similar à proposta do presente trabalho, foram selecionados dois aplicativos disponíveis para *download* no *Google Play* (serviço de distribuição digital de aplicativos desenvolvido e operado pela *Google*).

### 2.4.1 MED eventos

O aplicativo MED eventos possui informações de divulgação de eventos do meio médico com suas principais informações, tais como datas, valores e inscrições (UFENESP, 2017). A Figura 7 apresenta a tela de funcionamento da aplicação MED eventos:



Figura 7 – Tela da aplicação “MED eventos”. Fonte: (UFENESP, 2017)

A aplicação proposta no presente trabalho busca a promoção e disseminação de informações para eventos de forma individualizada e bem detalhada. Esse aplicativo têm foco principal na agregação de informações de vários eventos

simultaneamente sem possibilitar pesquisas e detalhes maiores de um evento em específico.

#### 2.4.2 Aplicativo da 32ª Reunião Anual SBPqO

O aplicativo foi desenvolvido para visualização dos resumos e informações gerais do evento da 32ª Reunião Anual da SBPqO, com possibilidade de consulta à programação, resumo, favoritos e informações gerais do evento (DEFINITY TEAM, 2017). A Figura 8 apresenta a tela inicial da aplicação:



Figura 8 – Tela da aplicação “32ª Reunião Anual SBPqO”. Fonte: (DEFINITY TEAM, 2017)

A aplicação mencionada nesta seção possui funções similares à aplicação móvel proposta no presente trabalho, esta última possuindo mais opções de busca personalizada com filtros por categorias e título, além da geração de credencial e conexão com conta do twitter e visualização de artigos postados em plataforma no estilo blog (wordpress).

### 3 DESENVOLVIMENTO DA APLICAÇÃO

A aplicação constitui-se em solução multiplataforma para o gerenciamento de eventos sociais com finalidade acadêmica, facilitando a interação e disseminação da informação dentre os usuários envolvidos. O protótipo de aplicação foi desenvolvido para utilização plataforma, focando inicialmente em dispositivos com sistemas Android e IOS, com a possibilidade de adicionar mais plataformas com o uso do *framework* Apache Cordova.

#### 3.1 REQUISITOS

##### 3.1.1 Requisitos Funcionais

Consistem nas declarações das funcionalidades que a aplicação oferece ao usuário. A seguir são descritos os requisitos funcionais:

Tabela 1 - Requisitos funcionais da aplicação proposta

| Requisito                                  | Descrição   |
|--|---|
| Logar no sistema                           | O sistema deve permitir que um usuário efetue “login” por meio de validação de usuário e senha  |
| Listar categorias de atividades            | O sistema deve permitir a listagem de categorias de atividades                                  |
| Listar atividades                          | O sistema deve permitir a listagem de atividades  |
| Marcar atividade na lista de favoritos     | O sistema deve permitir a marcação de uma atividade como favorita e adicionar um lembrete       |
| Buscar atividades                          | O sistema deve permitir busca de atividades por categoria ou título                             |
| Listar palestrantes                        | O sistema deve permitir a listagem de palestrantes e possibilitar filtro por nome               |
| Exibir credencial                          | O sistema deve permitir a exibição de credencial do usuário para entrada no evento / atividades |
| Exibir informações sobre transporte        | O sistema deve permitir a exibição de informações sobre transporte                              |
| Exibir informações sobre o local do evento | O sistema deve permitir a exibição de informações sobre a cidade / local do evento              |
| Manter usuários                            | O sistema deve permitir a edição do perfil do usuário   |

|   |  |
|---|--|
| Notificação                             | O sistema deve possibilitar o envio de notificações <i>push</i> para os usuários através de cadastro de notificação no <i>back end</i> |
| Conexão com plataforma <i>wordpress</i> | O sistema deve possibilitar a exibição de artigos postados em uma categoria específica de um blog utilizando <i>wordpress</i>          |
| Conexão com twitter                     | O sistema deve exibir as últimas postagens da conta do <i>twitter</i>  |

### 3.1.2 Requisitos não-funcionais

Consistem nos requisitos relacionais ao uso da aplicação em termos de qualidade, desempenho, usabilidade segurança, etc. A seguir são descritos os requisitos não funcionais:

Tabela 2 - Requisitos não-funcionais da aplicação proposta

| Requisito              | Descrição  |
|------------------------|--|
| Acesso multiplataforma | O sistema deve ser acessível por dispositivos com sistema Android e IOS                            |
| Desempenho             | O sistema não deve apresentar tempo superior a 3 segundos na execução das funções                  |
| Conexão                | O sistema necessita de conexão com a internet para realizar as interações com o <i>web service</i> |

## 3.2 VISÃO GERAL DA ARQUITETURA

Esta seção apresenta a visão arquitetural proposta para a aplicação, de forma a atender aos requisitos previamente definidos. A aplicação foi desenvolvida com base na arquitetura cliente / servidor, a qual se caracteriza pela divisão de processos na obtenção de dados (responsabilidade do cliente) e na manutenção da informação, que fica a cargo do servidor. A Figura 9 ilustra a separação de responsabilidades da interface de interação com o usuário (cliente) e da interface de gerenciamento de dados (servidor).

O *web service* que compõe o *back end* da aplicação, no lado servidor, tem a responsabilidade de integrar o dispositivo cliente com a base de dados (acesso às funcionalidades do SGBD).

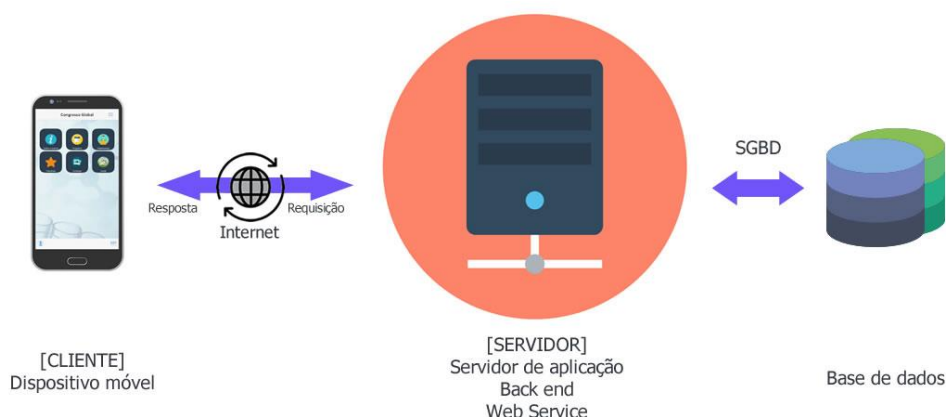


Figura 9 – Visão arquitetural: modelo de funcionamento do aplicativo. Fonte: Autoria própria

### 3.3 MODELAGEM

Nesta seção são apresentadas as especificações da aplicação proposta a partir da representação utilizando a linguagem UML. Foram utilizadas as ferramenta *ASTAH Community Edition* (ASTAH, 2017) e editor online do *Web Sequence Diagrams* (HANOV, 2007). A codificação da aplicação, utilizando a linguagem Javascript, não seguiu o paradigma de orientação a objetos. Desse modo, optou-se pela não elaboração de diagrama de classes, o qual não se demonstra apropriado em virtude da codificação orientada a comportamentos. Nesse contexto, a representação dos processos por meio de diagramas de sequência revelou-se mais adequada.

#### 3.3.1 Prototipação

Foi realizada prototipação de baixa fidelidade das telas do aplicativo, utilizando a ferramenta *Mockplus* (JONGDE SOFTWARE LLC, 2017), com a finalidade de representar a ideia do produto no projeto, focando nos principais requisitos funcionais e na estrutura de navegação. A Figura 10 apresenta a tela principal da aplicação, na qual o usuário poderá escolher entre as funcionalidades disponíveis. Na Figura 11 está representada a tela para login na aplicação, com os campos a serem preenchidos pelo usuário.

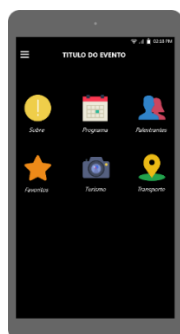


Figura 10 - Tela principal

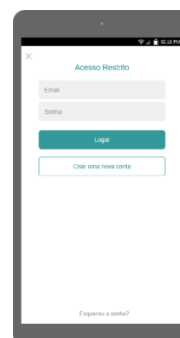


Figura 11 - Tela de login

A Figura 12 ilustra a tela de visualização de atividades quando o usuário acessa o programa do evento. A funcionalidade de listagem de palestrantes é representada na Figura 13.



Figura 12 – Tela de listagem de atividades com filtro

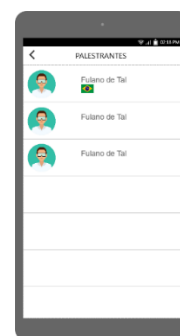


Figura 13 – Tela de listagem de palestrantes

As Figuras 14 e 15 representam, respectivamente, a tela de listagem de notícias externas (conexão com plataforma de blog) e a tela de gerenciamento de favoritos (exibição e possibilidade de exclusão de atividades marcadas como favoritas).

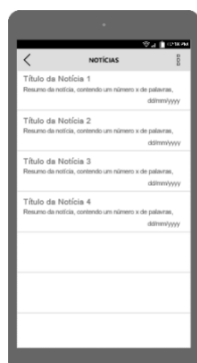


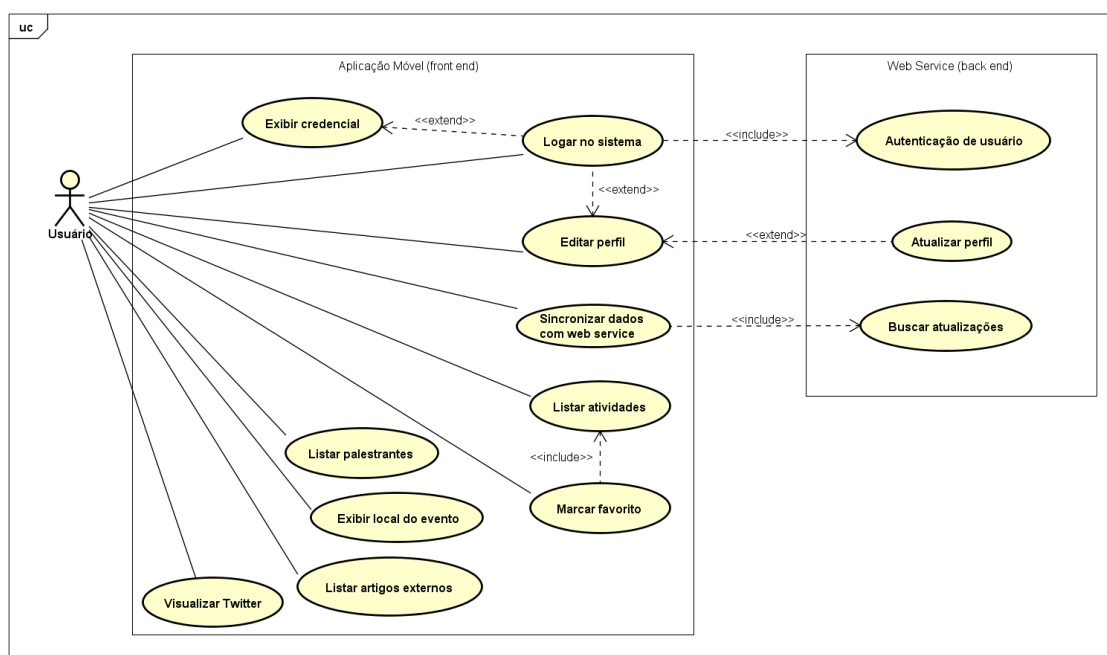
Figura 14 – Tela de listagem de notícias



Figura 15 – Tela de gerenciamento de favoritos

### 3.3.2 Diagramas de Casos de Uso

O diagrama de casos de uso representa a interação entre atores e a aplicação por meio do relacionamento entre esses elementos, descrevendo as funcionalidades a partir do ponto de vista do usuário. A construção de um caso de uso pode ser entendida como a descrição da sequência de mensagens trocadas entre um sistema e os atores envolvidos em seu contexto. A Figura 16 ilustra o diagrama de casos de uso com as funções da aplicação móvel e suas interações com o *web service*.



powered by Astah

Figura 16 - Diagrama de casos de uso: *front end* e interação com *back end*



A Figura 17 ilustra o diagramas de casos de uso referente às funções disponíveis ao administrador do sistema no *back end* da aplicação:

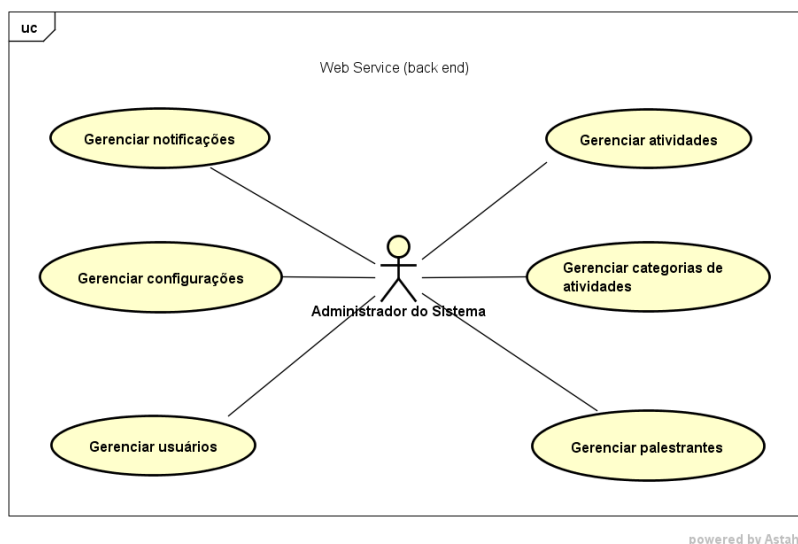


Figura 17 - Diagrama de casos de uso: *back end* da aplicação

### 3.3.3 Especificações de Casos de Uso da Aplicação Móvel (*front end*)

#### Sincronizar dados com o web service

**Descrição:** Este caso de uso descreve a sincronização das informações do banco de dados local com a base de dados do servidor (*back end*)

#### **Eventos:**

- O sistema exibe a tela inicial e atualiza o banco de dados local com as informações presentes na base de dados do servidor (*back end*)

#### **Atores:**

- Usuário
- Aplicativo
- *Web service*

#### **Pré-Condições:**

- O dispositivo móvel está conectado à internet

#### **Pós-Condições:**

- Não se aplica

#### **1. Conclusões com sucesso:**

- O aplicativo atualiza a base de dados local (SQLite) com os dados enviados pelo web service
- O aplicativo cadastra no *storage* local informações de configurações do evento

## 2. Conclusões sem sucesso:

- O aplicativo não consegue acesso ao *web service*

### Fluxo básico:

1. O usuário inicia a aplicação
2. O aplicativo exibe a tela inicial (E1)
3. O aplicativo requisita o caso de uso “buscar atualizações” no *web service* passando como parâmetro a última atualização realizada (data e hora)
4. O aplicativo atualiza a base de dados local com os dados fornecidos pelo *web service*
5. O aplicativo atualiza o *storage* local com informações de configurações do evento (título, descrição, latitude e longitude do local)
6. O caso de uso é encerrado

### Fluxos de exceção

E1. O aplicativo identifica que o dispositivo não está conectado à internet

E1.1. O aplicativo informa ao usuário que é necessária conexão ativa com a internet para a execução de algumas funcionalidades

E1.2. O caso de uso é encerrado

### Logar no sistema

**Descrição:** Este caso de uso descreve como o usuário realiza login na aplicação

### Eventos:

- O aplicativo exibe a tela e disponibiliza os campos a serem preenchidos.
- Usuário insere as informações referentes ao e-mail e a senha.
- O aplicativo valida os dados.
- O aplicativo retorna à tela inicial.

### Atores:

- Usuário
- *Web service*

### Pré-Condições:

- O usuário possui registro na base de dados do servidor
- O dispositivo móvel está conectado à internet

### Pós-Condições:

#### 1. Conclusões com sucesso:

- O aplicativo cadastra no *storage* local uma chave de autenticação do usuário
- O aplicativo cadastra no *storage* local e-mail e senha preenchidos.

## 2. Conclusões sem sucesso:

- Campos obrigatórios com conteúdo nulo
- Validação não encontra registro na base de dados

### Fluxo básico:

1. O aplicativo exibe a tela de acesso
2. Usuário preenche o campo de e-mail e senha
3. O aplicativo verifica os campos obrigatórios (E1)
4. O aplicativo requisita a execução do caso de uso “autenticação de usuário” no *web service* (E2)
5. O aplicativo informa o sucesso da operação
6. O caso de uso é encerrado

### Fluxos de exceção

E1. O aplicativo identifica campos obrigatórios não preenchidos

E1.1. O aplicativo exibe a mensagem “Campo E-mail (exemplo) não preenchido”.

E1.2. O aplicativo destaca o campo e aguarda o preenchimento

E2. O caso de uso “autenticação de usuário” informa que os dados enviados não foram encontrados nos registros

E2.1. O aplicativo exibe a mensagem: “Dados incorretos, cheque novamente”.

E2.2. O fluxo básico é retomado no item 1

### Editar perfil

**Descrição:** Este caso de uso descreve como o usuário edita seu perfil na aplicação

### Eventos:

- O aplicativo exibe a tela e disponibiliza os campos a serem editados
- O usuário edita as informações referentes ao seu cadastro
- O aplicativo valida os dados.
- O aplicativo informa sucesso na edição do perfil usuário

### Atores:

- Usuário
- *Web service*

### Pré-Condições:

- O usuário possui registro na base de dados do servidor e está logado
- O dispositivo móvel está conectado à internet

### Pós-Condições:

#### 1. Conclusões com sucesso:

- O aplicativo atualiza as informações do perfil do usuário no *storage* local e requisita ao *web service* atualização do banco de dados do *back end*

## 2. Conclusões sem sucesso:

- Campos obrigatórios com conteúdo nulo
- E-mail já cadastrado no sistema

### Fluxo básico:

1. O aplicativo exibe o formulário de cadastro com os dados do usuário (E1)
2. O usuário edita os campos de cadastro
3. O aplicativo verifica os campos preenchidos (E2) (E3)
4. O aplicativo executa o caso de uso “atualizar perfil” do *web service*
5. O aplicativo atualiza as informações no *storage* local
6. O aplicativo informa o sucesso da operação
7. O caso de uso é encerrado

### Fluxos de exceção:

E1. O aplicativo identifica que o usuário não está logado

E1.1. O sistema executa o caso de uso “logar no sistema”

E1.2. O caso de uso é encerrado

E2. O aplicativo identifica campos obrigatórios não preenchidos.

E2.1. O aplicativo Informa ao usuário a mensagem “Campo E-mail (exemplo) não preenchido”.

E2.2. O aplicativo destaca o campo e aguarda o preenchimento.

E3. O aplicativo identifica duplicidade de dados no registro de e-mail.

E3.1. O aplicativo exibe a mensagem: “E-mail já cadastrado”.

E3.2. O fluxo básico é retomado no item 1.

### Listar atividades

**Descrição:** Este caso de uso descreve a exibição da listagem de atividades

#### Eventos:

- O aplicativo exibe a tela e lista todas as atividades cadastradas para o evento
- O aplicativo filtra a lista de atividades quando usuário seleciona uma categoria

#### Atores:

- Usuário
- *Web service*
- Base de dados local

#### Pré-Condições:

- O aplicativo realizou sincronização de dados com o *web service* (ao menos uma vez)

**Pós-Condições:**

- Não se aplica

**1. Conclusões com sucesso:**

- O aplicativo exibe lista de atividades
- O aplicativo exibe lista de atividades de uma categoria (filtro selecionado)

**2. Conclusões sem sucesso:**

- Não se aplica

**Fluxo básico:**

1. O aplicativo exibe lista de atividades
2. O usuário seleciona uma categoria no filtro (A1)
3. O aplicativo busca dados referentes à categoria a partir de requisição à base de dados local
4. O aplicativo exibe lista de atividades da categoria selecionada no filtro
5. O caso de uso é encerrado

**Fluxo alternativo:**

A1. O aplicativo identifica que não há eventos para a categoria selecionada

A1.1. O aplicativo informa ao usuário a mensagem “Nenhuma atividade cadastrada para essa categoria”

A1.2. O caso de uso é encerrado

**Listar artigos externos (conexão com plataforma *wordpress*)**

**Descrição:** Este caso de uso descreve a exibição da listagem de artigos postados em um blog que forneça dados em formato JSON para consumo (foi utilizado um blog na plataforma *wordpress* para a aplicação proposta)

**Eventos:**

- O aplicativo exibe a tela e lista todas os artigos postados no blog *wordpress*
- O aplicativo acessa o artigo em uma nova janela quando usuário solicita

**Atores:**

- Usuário
- *Web service* da plataforma *wordpress*

**Pré-Condições:**

- O dispositivo móvel está conectado à internet

**Pós-Condições:****1. Conclusões com sucesso:**

- O aplicativo exibe lista de artigos
- O aplicativo acessa o artigo selecionado em uma nova janela

**2. Conclusões sem sucesso:**

- Não se aplica

**Fluxo básico:**

1. O aplicativo requisita os artigos ao *web service* da plataforma *wordpress*
2. O aplicativo exibe a lista de artigos recebidos
3. O usuário seleciona um artigo
4. O aplicativo gera uma nova janela com a exibição do artigo selecionado
5. O usuário toca sobre o botão voltar
6. A janela é fechada e o sistema retorna à tela de lista de artigos
7. O caso de uso é encerrado

**Marcar atividade na lista de favoritos**

**Descrição:** Este caso de uso descreve a marcação de uma atividade na lista de favoritos

**Eventos:**

- O aplicativo exibe uma lista de atividades
- A atividade selecionada é incluída na lista de favoritos

**Atores:**

- Usuário

**Pré-Condições:**

- O aplicativo realizou sincronização de dados com o *web service* (ao menos uma vez)
- O usuário requisitou lista de atividades de uma categoria, conforme descrito no caso de uso listar atividades

**Pós-Condições:**

- Não se aplica

**1. Conclusões com sucesso:**

- O aplicativo inclui uma atividade na lista de favoritos

**2. Conclusões sem sucesso:**

- Não se aplica

**Fluxo básico:**

1. O aplicativo executa o caso de uso listar atividades
2. O usuário seleciona uma atividade da lista
3. O usuário seleciona o ícone “favorito” para uma atividade (A1)
4. O aplicativo inclui a atividade na lista de favoritos da base de dados local do dispositivo
5. O aplicativo informa ao usuário o sucesso da operação
6. O caso de uso é encerrado

**Fluxo alternativo:**

A1. O usuário desmarca o ícone “favorito” para uma atividade

A1.1. O aplicativo remove a atividade da base de dados local

A1.2. O aplicativo informa ao usuário a remoção a remoção de atividade da lista de favoritos

A1.3. O caso de uso é encerrado

**Listar palestrantes**

**Descrição:** Este caso de uso descreve a exibição da listagem de palestrantes cadastrados no evento

**Eventos:**

- O aplicativo exibe a tela e lista todos os palestrantes cadastrados

**Atores:**

- Usuário

**Pré-Condições:**

- O aplicativo realizou sincronização de dados com o *web service* (ao menos uma vez)

**Pós-Condições:**

- Não se aplica

**1. Conclusões com sucesso:**

- O aplicativo exibe lista de palestrantes

**2. Conclusões sem sucesso:**

- Não se aplica

**Fluxo básico:**

1. O usuário requisita a lista de palestrantes (A1)
2. O aplicativo busca na base de dados local a lista de palestrantes cadastrados para o evento
3. O sistema exibe a lista de palestrantes cadastrados
4. O caso de uso é encerrado

**Fluxo alternativo:**

A1. O usuário realiza busca do palestrante por nome

A1.1. O aplicativo filtra a lista de palestrantes e exibe o resultado da busca

**Exibir credencial**

**Descrição:** Este caso de uso exibe a credencial do usuário

**Eventos:**

- O aplicativo gera um código QR e exibe na tela do dispositivo móvel

**Atores:**

- Usuário

**Pré-Condições:**

- O usuário possui cadastro
- O usuário está logado no sistema

**Pós-Condições:**

- Não se aplica

**1. Conclusões com sucesso:**

- O aplicativo exibe credencial do usuário

**2. Conclusões sem sucesso:**

- Não se aplica

**Fluxo básico:**

1. O usuário requisita a credencial (E1)
2. O aplicativo busca na base local do dispositivo móvel os dados do usuário
3. O aplicativo gera e exibe o código QR
4. O caso de uso é encerrado

**Fluxo de exceção:**

- E1. O aplicativo identifica que o usuário não está logado  
E1.2. O aplicativo dá opção de login e executa o caso de uso “logar no sistema”  
E1.3. O caso de uso é encerrado

**Visualizar twitter**

**Descrição:** Este caso de uso exibe a lista de postagens do twitter do evento

**Eventos:**

- O aplicativo exibe a página do twitter associada ao evento

**Atores:**

- Usuário

**Pré-Condições:**

- O dispositivo móvel está conectado à internet

**Pós-Condições:**

- Não se aplica



**1. Conclusões com sucesso:**

- O aplicativo exibe a página do twitter associada ao evento

**2. Conclusões sem sucesso:**

- Não se aplica

**Fluxo básico:**

1. O usuário requisita acesso à página do *twitter* associada ao evento
2. O aplicativo gera uma nova janela com a exibição da página do *twitter*
3. O usuário toca sobre o botão voltar
4. A janela é fechada e o sistema retorna à tela de lista de artigos
5. O caso de uso é encerrado

**Exibir local do evento**

**Descrição:** Este caso de uso exibe mapa com o local do evento

**Eventos:**

- O sistema gera um mapa do serviço Google Maps a partir dos dados de latitude e longitude cadastrados no *back end* da aplicação

**Atores:**

- Usuário

**Pré-Condições:**

- O aplicativo realizou sincronização de dados com o *web service* (ao menos uma vez)

**Pós-Condições:**

- Não se aplica

**1. Conclusões com sucesso:**

- O aplicativo exibe mapa interativo do local do evento

**2. Conclusões sem sucesso:**

- Não se aplica

**Fluxo básico:**

1. O usuário requisita o local do evento (E1)
2. O aplicativo faz uma requisição à *API Javascript* do *Google Maps*, passando os dados de latitude e longitude como parâmetro
3. O aplicativo exibe o mapa interativo na tela
4. O usuário seleciona o ícone de rotas
5. O aplicativo executa a aplicação de gerenciamento de mapas cadastrada no sistema operacional em uso
6. O caso de uso é encerrado

**Fluxo de exceção:**

- E1. O aplicativo identifica que o dispositivo não está conectado à internet
- E1.1. O aplicativo informa ao usuário que é necessária conexão ativa com a internet para a execução de algumas funcionalidades
- E1.2. O caso de uso é encerrado

3.3.4 Especificações de Casos de Uso do *Web Service (back end)***Autenticação de usuário**

**Descrição:** Este caso de uso descreve a checagem de login, pelo *web service*, a partir dos dados recebidos do aplicativo móvel

**Eventos:**

- O *web service* recebe a requisição de checagem de usuário e realiza a consulta a partir dos parâmetros fornecidos pelo usuário no caso de uso da aplicação móvel: “logar no sistema”

**Atores:**

- Usuário
- Aplicativo
- *Web service (back end)*

**Pré-Condições:**

- O dispositivo móvel está conectado à internet
- O usuário possui registro na base de dados do servidor

**Pós-Condições:**

- Não se aplica

**1. Conclusões com sucesso:**

- Os dados da base de dados do servidor são sincronizados com os dados da base local do aplicativo

**2. Conclusões sem sucesso:**

- Não se aplica

**Fluxo básico:**

1. O usuário preenche os dados durante o caso de uso “logar no sistema”
2. O aplicativo requisita autenticação ao *web service* (E1)
3. O *web service* realiza a checagem dos parâmetros fornecidos
4. O *web service* devolve mensagem de sucesso ao aplicativo (A1)
5. O caso de uso é encerrado

**Fluxo alternativo:**

A1. O *web service* não localiza registro de usuário com os dados fornecidos pelo aplicativo

A1.1. O *web service* devolve mensagem de insucesso ao aplicativo

A1.2. O caso de uso é encerrado

**Fluxo de exceção:**

E1. O aplicativo identifica que o dispositivo não está conectado à internet

E1.1. O aplicativo informa ao usuário que é necessária conexão ativa com a internet para a execução de algumas funcionalidades

E1.2. O caso de uso é encerrado

**Atualizar perfil**

**Descrição:** Este caso de uso descreve a atualização das informações do perfil do usuário no banco de dados do servidor (*back end*), a partir dos dados fornecidos pelo aplicativo móvel

**Eventos:**

- O *web service* recebe a requisição de atualização e responde com a informação de sucesso ao aplicativo móvel

**Atores:**

- Aplicativo

- *Web service (back end)*

**Pré-Condições:**

- O dispositivo móvel está conectado à internet

- O usuário possui registro na base de dados do servidor

**Pós-Condições:**

- Não se aplica

**1. Conclusões com sucesso:**

- Os dados do perfil do usuário no banco de dados do servidor (*back end*) são atualizados

**2. Conclusões sem sucesso:**

- Os dados não são atualizados

**Fluxo básico:**

1. O aplicativo requisita ao *web service* a atualização do perfil na base de dados do servidor enviando os dados preenchidos pelo usuário (E1)
2. O *web service* realiza a checagem dos parâmetros fornecidos
3. O *web service* atualiza os dados enviados pelo aplicativo
4. O caso de uso é encerrado

**Fluxo de exceção:**

- E1. O aplicativo identifica que o dispositivo não está conectado à internet
- E1.1. O aplicativo informa ao usuário que é necessário conexão ativa com a internet para a execução de algumas funcionalidades
- E1.2. O caso de uso é encerrado

**Buscar atualizações**

**Descrição:** Este caso de uso descreve o fornecimento de dados pelo *web service* a partir da última informação de atualização fornecida pelo aplicativo móvel

**Eventos:**

- O *web service* recebe a requisição de atualização e responde com os dados solicitados a partir dos parâmetros fornecidos pelo aplicativo

**Atores:**

- Aplicativo
- *Web service (back end)*

**Pré-Condições:**

- O dispositivo móvel está conectado à internet
- O usuário possui registro na base de dados do servidor

**Pós-Condições:**

- Não se aplica

**1. Conclusões com sucesso:**

- Os dados da base de dados do servidor são sincronizados com os dados da base local da aplicação móvel

**2. Conclusões sem sucesso:**

- Não se aplica

**Fluxo básico:**

1. O aplicativo requisita atualização da base de dados local ao *web service*, fornecendo a data e hora da última atualização (E1)
2. O *web service* realiza a checagem dos parâmetros fornecidos
3. O *web service* fornece os dados solicitados pelo aplicativo
4. O caso de uso é encerrado

**Fluxo de exceção:**

- E1. O aplicativo identifica que o dispositivo não está conectado à internet
- E1.1. O aplicativo informa ao usuário que é necessário conexão ativa com a internet para a execução de algumas funcionalidades
- E1.2. O caso de uso é encerrado

## Gerenciar notificações

**Descrição:** Este caso de uso descreve o cadastro e envio de notificações para o aplicativo móvel

### **Eventos:**

- O administrador do sistema cadastra a notificação e o *web service* a envia ao servidor do GCM que inclui a mensagem em uma fila até que os dispositivos móveis se conectem à internet para receber as notificações

### **Atores:**

- Administrador do sistema
- *Web service (back end)*
- Aplicativo

### **Pré-Condições:**

- O servidor (*back end*) possui conexão com a internet

### **Pós-Condições:**

- Não se aplica

### **1. Conclusões com sucesso:**

- Os dispositivos móveis com o aplicativo instalado recebem a mensagem de notificação quando conectados à internet

### **2. Conclusões sem sucesso:**

- Não se aplica

### **Fluxo básico:**

1. O administrador do sistema cadastra uma mensagem de notificação no *back end* da aplicação
2. O *web service* envia a mensagem ao servidor do GCM que distribui a mensagem para os dispositivos móveis com o aplicativo instalado
3. O caso de uso é encerrado

## Gerenciar usuários

**Descrição:** Este caso de uso descreve o gerenciamento de usuários do aplicativo, cadastrados no servidor do *back end*

### **Eventos:**

- O administrador do sistema realiza a operação de criação, recuperação, atualização ou exclusão de registros de usuários (operações CRUD) na base de dados do servidor (*back end*)

**Atores:**

- Administrator do sistema
- *Web service (back end)*

**Pré-Condições:**

- O servidor (*back end*) possui conexão com a internet

**Pós-Condições:**

- Não se aplica

**1. Conclusões com sucesso:**

- As operações CRUD são realizadas na base de dados do servidor

**2. Conclusões sem sucesso:**

- Não se aplica

**Fluxo básico:**

1. O administrador do sistema realiza uma operação CRUD na base de dados do servidor (criação, recuperação, atualização ou exclusão de registro)
2. O *web service* informa o sucesso da operação
3. O caso de uso é encerrado

**Gerenciar atividades**

**Descrição:** Este caso de uso descreve o gerenciamento de atividades cadastradas no servidor do *back end*

**Eventos:**

- O administrador do sistema realiza a operação de criação, recuperação, atualização ou exclusão de atividades (operações CRUD) na base de dados do servidor (*back end*)

**Atores:**

- Administrator do sistema
- *Web service (back end)*

**Pré-Condições:**

- O servidor (*back end*) possui conexão com a internet

**Pós-Condições:**

- Não se aplica

**1. Conclusões com sucesso:**

- As operações CRUD são realizadas na base de dados do servidor

**2. Conclusões sem sucesso:**

- Não se aplica

**Fluxo básico:**

1. O administrador do sistema realiza uma operação CRUD na base de dados do servidor (criação, recuperação, atualização ou exclusão de registro)
2. O *web service* informa o sucesso da operação
3. O caso de uso é encerrado

**Gerenciar categorias de atividades**

**Descrição:** Este caso de uso descreve o gerenciamento de categorias de atividades cadastradas no servidor do *back end*

**Eventos:**

- O administrador do sistema realiza a operação de criação, recuperação, atualização ou exclusão de categorias (operações CRUD) na base de dados do servidor (*back end*)

**Atores:**

- Administrador do sistema
- *Web service (back end)*

**Pré-Condições:**

- O servidor (*back end*) possui conexão com a internet

**Pós-Condições:**

- Não se aplica

**1. Conclusões com sucesso:**

- As operações CRUD são realizadas na base de dados do servidor

**2. Conclusões sem sucesso:**

- Não se aplica

**Fluxo básico:**

1. O administrador do sistema realiza uma operação CRUD na base de dados do servidor (criação, recuperação, atualização ou exclusão de registro)
2. O *web service* informa o sucesso da operação
3. O caso de uso é encerrado

**Gerenciar palestrantes**

**Descrição:** Este caso de uso descreve o gerenciamento de palestrantes cadastrados no servidor do *back end*

**Eventos:**

- O administrador do sistema realiza a operação de criação, recuperação, atualização ou exclusão de palestrantes (operações CRUD) na base de dados do servidor (*back end*)

**Atores:**

- Administrador do sistema  
- *Web service (back end)*

**Pré-Condições:**

- O servidor (*back end*) possui conexão com a internet

**Pós-Condições:**

- Não se aplica

**1. Conclusões com sucesso:**

- As operações CRUD são realizadas na base de dados do servidor

**2. Conclusões sem sucesso:**

- Não se aplica

**Fluxo básico:**

1. O administrador do sistema realiza uma operação CRUD na base de dados do servidor (criação, recuperação, atualização ou exclusão de registro)
2. O *web service* informa o sucesso da operação
3. O caso de uso é encerrado

**Gerenciar configurações**

**Descrição:** Este caso de uso descreve o gerenciamento de configurações do evento (título, descrição e informações do local) cadastradas no servidor do *back end*

**Eventos:**

- O administrador do sistema realiza cadastro ou alteração das informações do evento na base de dados do servidor (*back end*)

**Atores:**

- Administrador do sistema  
- *Web service (back end)*

**Pré-Condições:**

- O servidor (*back end*) possui conexão com a internet

**Pós-Condições:**

- Não se aplica

**1. Conclusões com sucesso:**



- O cadastro ou alteração das informações é realizado na base de dados do servidor

## 2. Conclusões sem sucesso:

- Não se aplica

### Fluxo básico:

1. O administrador do sistema realiza uma operação de cadastro ou alteração na base de dados do servidor
2. O *back end* informa o sucesso da operação
3. O caso de uso é encerrado

### 3.3.5 Diagrama de Implantação

O diagrama de implantação representa o relacionamento entre os recursos de hardware e software disponíveis para a execução da aplicação além de ilustrar qual a forma de comunicação entre os entes envolvidos.

O fluxo de informações da aplicação proposta se dá pela troca de mensagens via protocolo *HTTP (REST)* entre o dispositivo móvel e um servidor de aplicação que gerencia os recursos do sistema de administração (*back end* e base de dados), conforme a Figura 18.

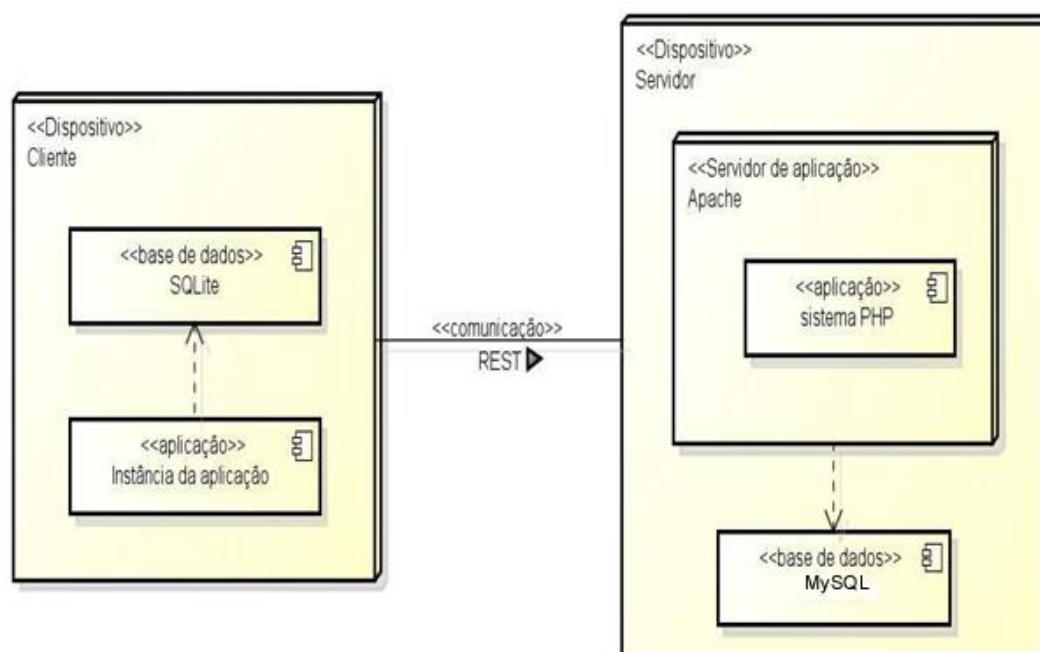


Figura 18 - Diagrama de implantação da aplicação proposta

### 3.3.6 Diagrama Entidade Relacionamento

O diagrama entidade relacionamento tem como objetivo ilustrar a estrutura lógica geral de um banco de dados enfatizando os dados e seus relacionamentos. A Figura 19 representa o modelo de dados do servidor.

A tabela de dados 1 contém as informações referentes ao cadastro de usuários: dados cadastrais, dados de acesso e perfil (participante, organizador ou palestrante). Algumas funcionalidades da aplicação móvel são habilitadas para determinados usuários conforme seu perfil cadastrado no servidor.

A tabela de dados 2 armazena as informações gerais de atividades (título, data, carga horária, etc.) e está relacionada com sua categorização (tabela de dados 3) e a sala onde a atividade vai ser realizada (tabela de dados 4).

A tabela de dados 5 armazena os dados dos convidados e se relaciona com a Tabela 2 (atividades) por meio dos registros dos convidados em suas respectivas atividades na tabela de dados 6. A tabela de dados 7 armazena configurações de exibição do evento (título, logo, etc.).

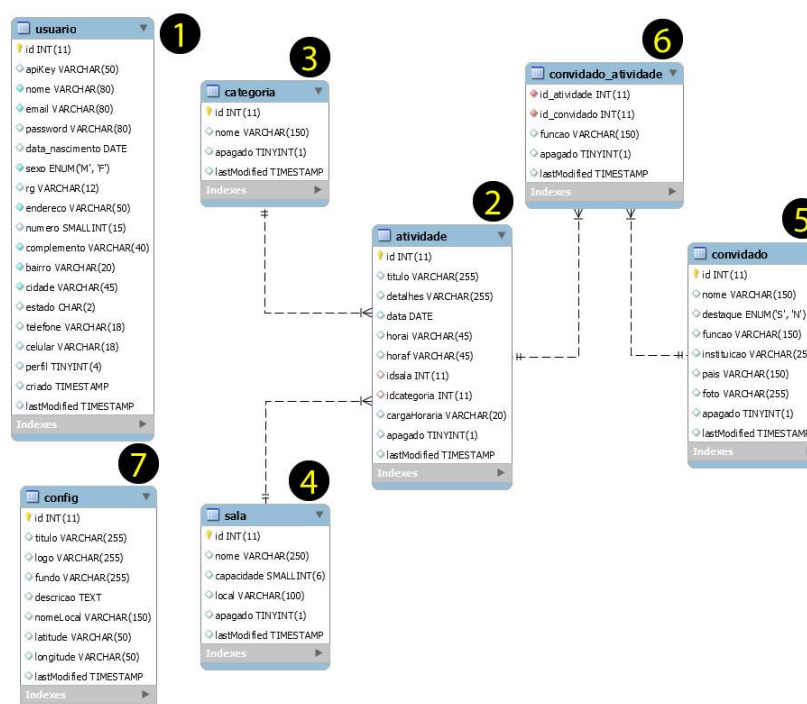


Figura 19 - DER - Base de dados MySQL

Na aplicação móvel, são persistidas as tabelas de dados 2, 5 e 6 referentes, respectivamente, às atividades, convidados e o relacionamento entre essas duas tabelas. Toda vez que a aplicação é executada é realizada uma sincronização dos dados do *web service* com a base de dados local SQLITE por meio de checagem do campo “*lastModified*” das tabelas mencionadas e no caso em que a última atualização do servidor é posterior à atualização da base local, apenas os dados que foram alterados são atualizados no dispositivo móvel.

### 3.3.7 Diagramas de Sequência

Os diagramas de sequências foram elaborados para representar os principais fluxos entre os módulos do sistema e suas interações com os atores envolvidos nas operações realizadas.

#### Sincronizar dados com o web service

A Figura 20 descreve o diagrama de sequência “sincronizar dados com o web service”, que demonstra a sequência de eventos para sincronização de informações do banco de dados local com a base de dados do servidor (*back end*).

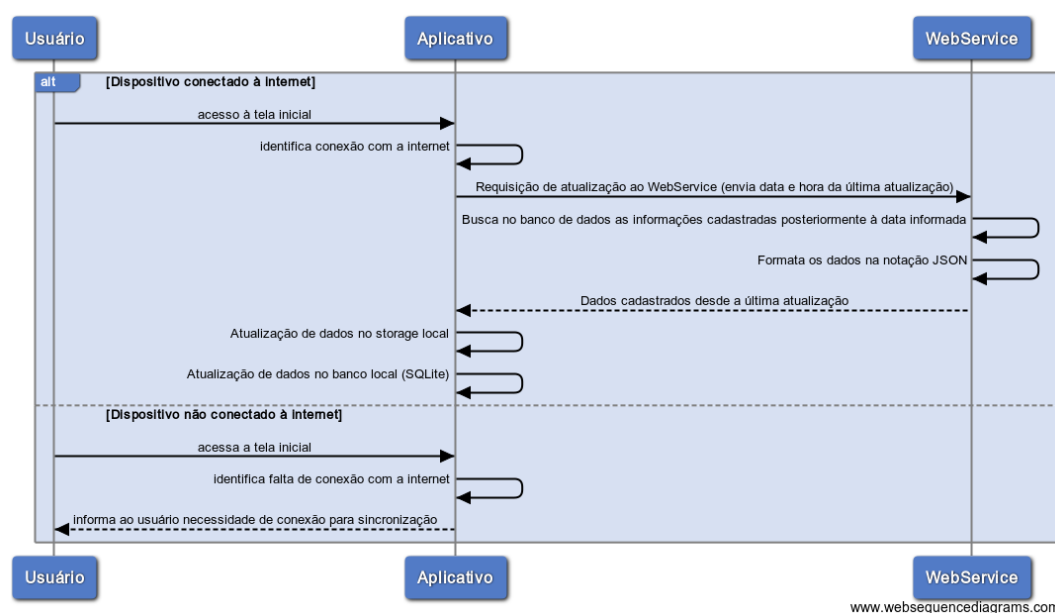


Figura 20 - Diagrama de sequência: Sincronizar dados com o *web service*

## Logar no sistema

A Figura 21 descreve o diagrama de sequência “logar no sistema”, que demonstra a sequência de eventos para que o usuário inicie uma sessão na aplicação móvel através da inserção de dados cadastrais e posterior validação na tabela de usuários da base de dados do *back end*.

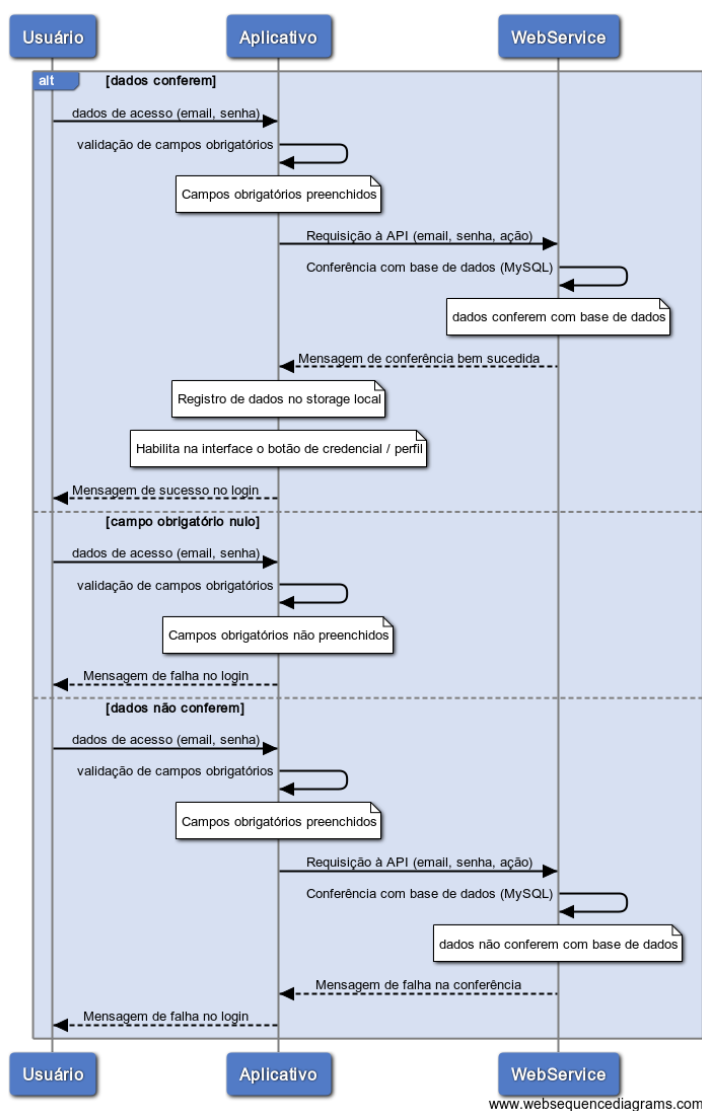


Figura 21 - Diagrama de sequência - Logar no sistema

## Editar perfil

A Figura 22 descreve o diagrama de sequência “editar perfil”, que demonstra a sequência de eventos para que o usuário realize a alteração de dados cadastrais previamente inseridos na base de dados por meio da alteração das informações no formulário e posterior validação e atualização.

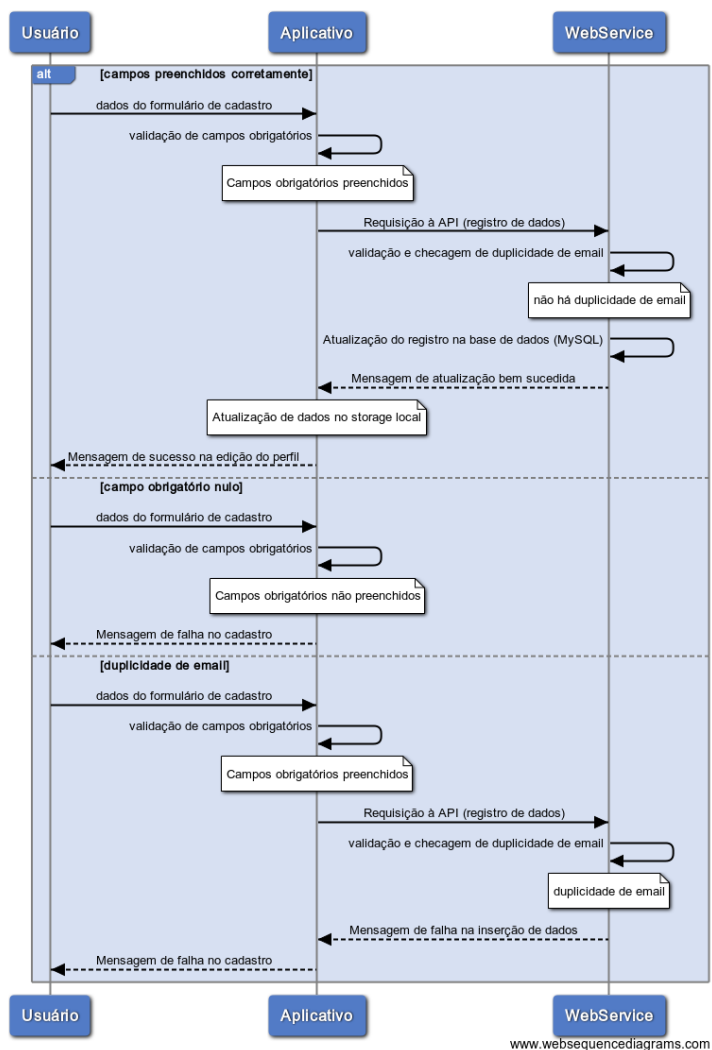


Figura 22 - Diagrama de sequência - Editar perfil

### Listar atividades

A Figura 23 ilustra o diagrama de sequência “listar atividades”, que descreve a exibição de atividades cadastradas para o evento possibilitando a busca por título além do filtro de categorias cadastradas.

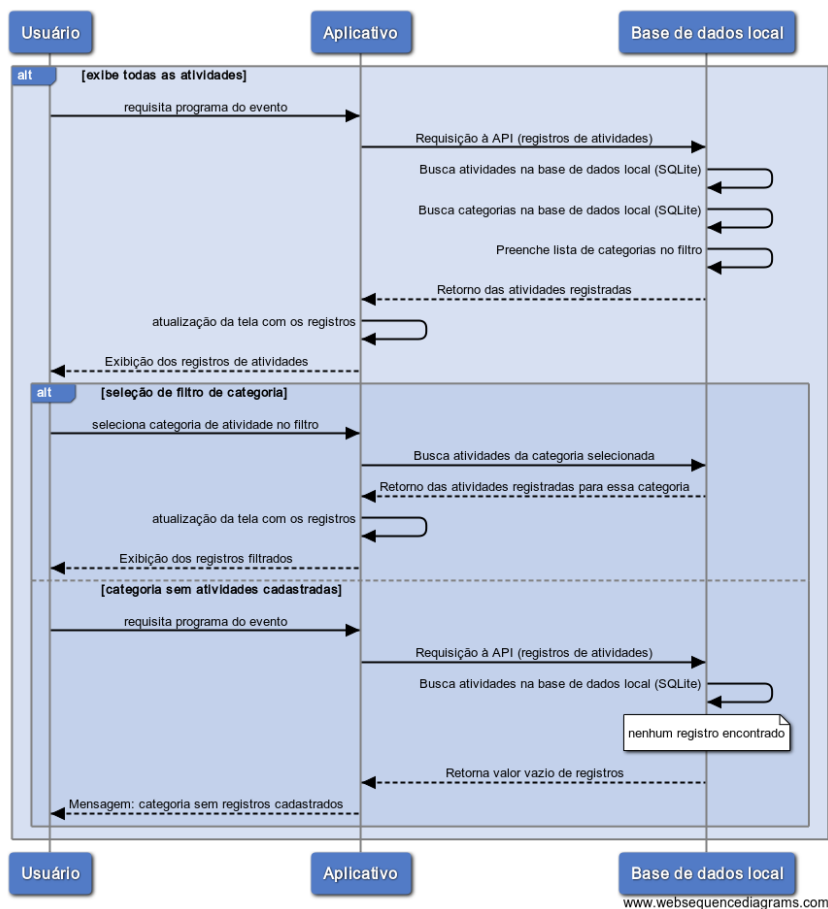


Figura 23 - Diagrama de sequência - Listar atividades

### Listar artigos externos (conexão com plataforma *wordpress*)

A Figura 24 descreve o diagrama de sequência “listar artigos externos”, que descreve a exibição de artigos postados em uma categoria específica de um blog com plataforma *wordpress*.

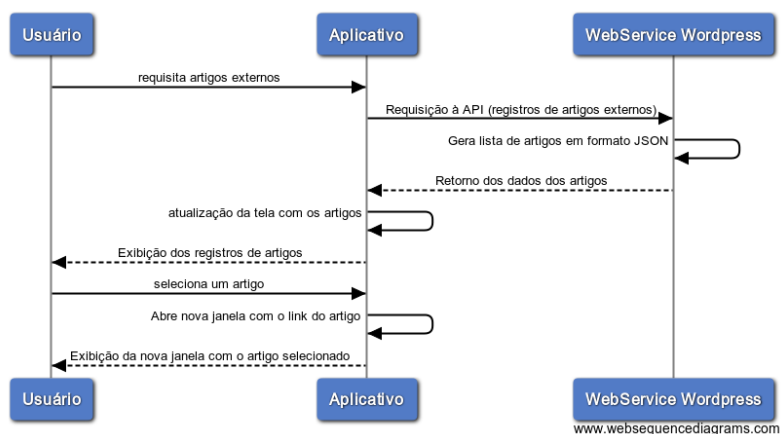


Figura 24 - Diagrama de sequência - Listar artigos externos

### Marcar atividades na lista de favoritos

A Figura 25 descreve o diagrama de sequência “marcar atividades na lista de favoritos”, que descreve a sequência de eventos para que o usuário registre localmente no dispositivo móvel uma lista de atividades de seu interesse.

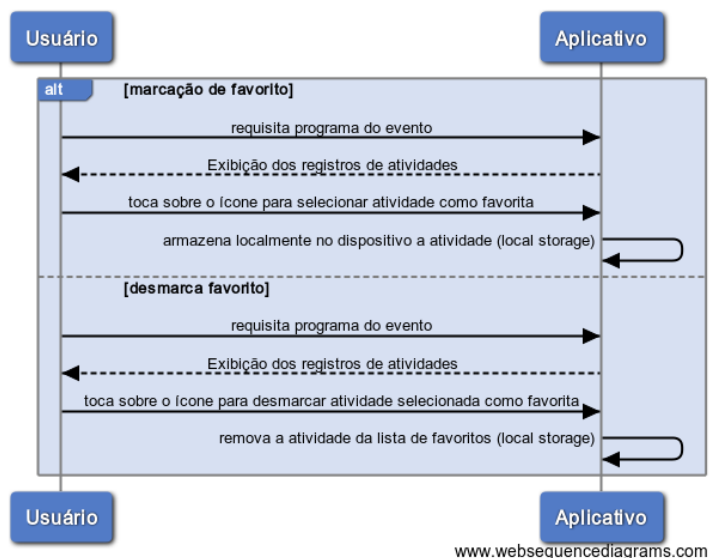


Figura 25 - Diagrama de sequência - Marcar atividades na lista de favoritos

### Listar palestrantes

A Figura 26 descreve o diagrama de sequência “listar palestrantes”, que descreve a exibição de categorias de atividades cadastradas para o evento possibilitando a busca por título.

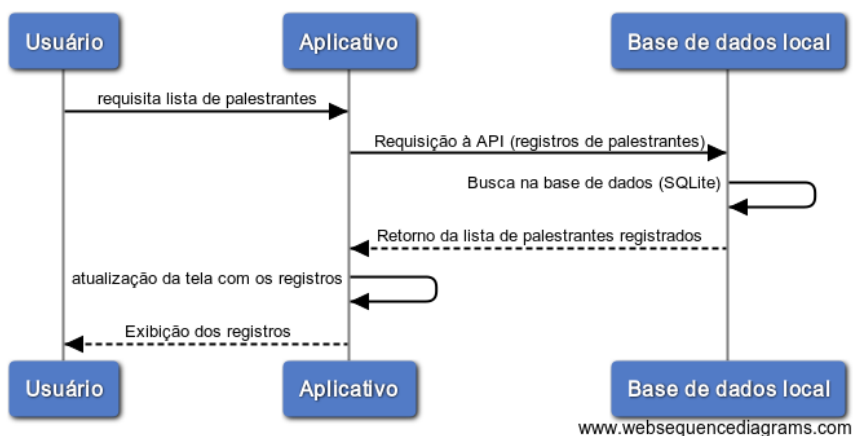


Figura 26 - Diagrama de sequência - Listar palestrantes

## Exibir credencial

A Figura 27 descreve o diagrama de sequência “exibir credencial”, que descreve a exibição da credencial para um usuário previamente logado no sistema.

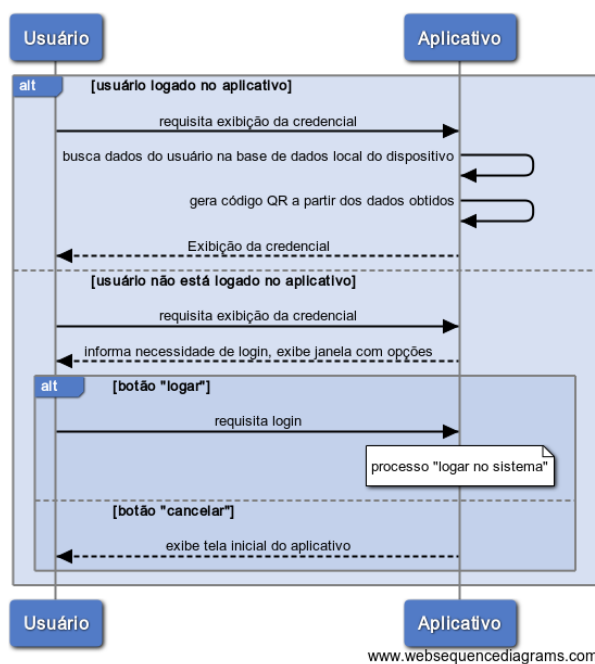


Figura 27 - Diagrama de sequência - Exibir credencial

### 3.3.8 Diagrama de Atividades

O diagrama de atividades representa os passos a serem percorridos para a conclusão de um processo, descrevendo o fluxo de atividades e suas interações. A Figura 28 ilustra o diagrama de atividades do processo de sincronização de dados entre o banco de dados local e a base de dados do servidor.



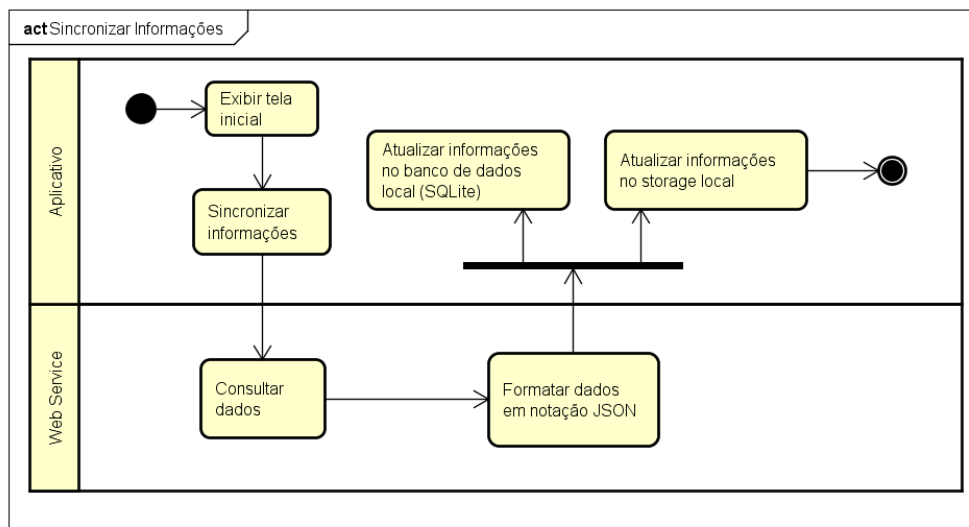


Figura 28 - Diagrama de atividades: Sincronizar informações com o *web service*

### 3.3.9 Análise de Usabilidade

Com a finalidade de validar as soluções de interface da aplicação proposta, foram realizados testes de usabilidade em protótipo de alta fidelidade com 3 usuários, para ajustes e avaliação da interface conforme a descoberta de padrões e sugestão de melhorias. A metodologia utilizada para os testes foi a avaliação somativa com coleta de dados quantitativos e análise interpretativa, a partir da observação de usuários e preenchimento de tabela com o índice de sucesso do cenário proposto.

Como métrica foi utilizada a quantidade de tarefas completadas com sucesso a partir da lista de tarefas que envolvem as principais funções da aplicação proposta, utilizando-se índice 0 (zero) para tarefa não completada, 1 (um) para tarefa completada com dificuldade e 2 (dois) para tarefa completada sem dificuldade. Quando da obtenção do índice 1, foi registrada a dificuldade apresentada. Segue a lista de tarefas propostas para os usuários convidados:

- Tarefa 1: Logar no sistema (usuário: user@email.com; senha: 123456)
- Tarefa 2: Acessar a credencial do evento;
- Tarefa 3: Acessar as informações do palestrante “John Doe”;
- Tarefa 4: Buscar a atividade “Facetas estéticas de Zircônia” na categoria “Workshop”;
- Tarefa 5: Marcar a atividade “Facetas estéticas de Zircônia” como favorita;

- Tarefa 6: Visualizar a lista de favoritos;
- Tarefa 7: Desmarcar a atividade “Facetas estéticas de Zircônia” da lista de favoritos;
- Tarefa 8: Acessar o perfil e realizar alteração do número de endereço;
- Tarefa 9: Acessar as informações sobre o local do evento.

Após a realização das tarefas, foi aplicado um questionário baseado na escala SUS com a finalidade de analisar a qualidade da interação sob a percepção do usuário, composto por 10 perguntas cujas respostas variam em uma escala de 1 a 5 (1: discordo completamente e 5: concordo completamente):

| Tabela 3 - Questionário de usabilidade baseado no SUS   |  |
|---|--|
| Pergunta  | Resposta: (1 Discordo completamente – 5 Concordo completamente)  |
| 1. Eu acho que gostaria de usar esse aplicativo com frequência.                                     | <input type="checkbox"/> 1 <input type="checkbox"/> 2 <input type="checkbox"/> 3 <input type="checkbox"/> 4 <input type="checkbox"/> 5 |
| 2. Eu achei o aplicativo desnecessariamente complexo.   | <input type="checkbox"/> 1 <input type="checkbox"/> 2 <input type="checkbox"/> 3 <input type="checkbox"/> 4 <input type="checkbox"/> 5 |
| 3. Eu achei o aplicativo fácil de usar.   | <input type="checkbox"/> 1 <input type="checkbox"/> 2 <input type="checkbox"/> 3 <input type="checkbox"/> 4 <input type="checkbox"/> 5 |
| 4. Eu acho que precisaria de ajuda de uma pessoa com conhecimentos técnicos para usar o aplicativo. | <input type="checkbox"/> 1 <input type="checkbox"/> 2 <input type="checkbox"/> 3 <input type="checkbox"/> 4 <input type="checkbox"/> 5 |
| 5. Eu acho que as várias funções do aplicativo estão muito bem integradas.                          | <input type="checkbox"/> 1 <input type="checkbox"/> 2 <input type="checkbox"/> 3 <input type="checkbox"/> 4 <input type="checkbox"/> 5 |
| 6. Eu acho que o aplicativo apresenta muita inconsistência.   | <input type="checkbox"/> 1 <input type="checkbox"/> 2 <input type="checkbox"/> 3 <input type="checkbox"/> 4 <input type="checkbox"/> 5 |
| 7. Eu imagino que as pessoas aprenderão como usar esse aplicativo rapidamente.                      | <input type="checkbox"/> 1 <input type="checkbox"/> 2 <input type="checkbox"/> 3 <input type="checkbox"/> 4 <input type="checkbox"/> 5 |
| 8. Eu achei o aplicativo atrapalhado de usar.   | <input type="checkbox"/> 1 <input type="checkbox"/> 2 <input type="checkbox"/> 3 <input type="checkbox"/> 4 <input type="checkbox"/> 5 |
| 9. Eu me senti confiante ao usar o aplicativo.  | <input type="checkbox"/> 1 <input type="checkbox"/> 2 <input type="checkbox"/> 3 <input type="checkbox"/> 4 <input type="checkbox"/> 5 |
| 10. Eu precisei aprender várias coisas novas antes de conseguir usar o aplicativo.                  | <input type="checkbox"/> 1 <input type="checkbox"/> 2 <input type="checkbox"/> 3 <input type="checkbox"/> 4 <input type="checkbox"/> 5 |

### 3.3.9.1 Resultados da análise de usabilidade

Os três participantes convidados conseguiram realizar com êxito todas as tarefas propostas para a análise de usabilidade, com uma média de tempo de execução menor que 30 segundos para cada tarefa. Relativo ao perfil dos participantes, 1 se classificou como usuários básico (conhecimentos básicos sobre informática e uso casual de aplicativos de celular), 1 como usuário intermediário (uso

corriqueiro de softwares de edição, sistemas e aplicativos de celular) e o outro participante é usuário avançado (formação acadêmica e atuação profissional na área de informática, com experiência no uso de sistemas e aplicativos de celular). A tabela abaixo apresenta os índices de sucesso observados durante o teste de usabilidade:

Tabela 4 - Índice de sucesso observado na execução das tarefas (0: tarefa não completada, 1: tarefa completada com dificuldade e 2: tarefa completada sem dificuldade)

| Tarefa  | Índice usuário 1<br>(básico)   | Índice usuário 2<br>(intermediário)  | Índice usuário 3<br>(avançado)                                |
|---|--|--|---|
| Tarefa 1: Logar no sistema  | 2  | 2  | 1 (errou a digitação do email)                                |
| Tarefa 2: Acessar a credencial do evento  | 2  | 2  | 2   |
| Tarefa 3: Acessar as informações do palestrante John Doe                              | 2  | 2  | 2   |
| Tarefa 4: Buscar a atividade "Facetas estéticas de Zircônia" na categoria Workshop    | 2  | 2  | 2   |
| Tarefa 5: Marcar a atividade "Facetas estéticas de Zircônia" como favorita            | 1 (demorou para identificar que o favorito é marcado pelo ícone de estrela e questionou sobre uma confirmação) | 1 (clicou duas vezes sobre o ícone estrela, aguardava uma mensagem de confirmação) | 2   |
| Tarefa 6: Visualizar a lista de favoritos   | 2  | 2  | 2   |
| Tarefa 7: Desmarcar a atividade "Facetas estéticas de Zircônia" da lista de favoritos | 2  | 2  | 2   |
| Tarefa 8: Acessar seu perfil e realizar alteração do número de endereço               | 1 (não preencheu a senha de acesso na primeira tentativa)  | 2  | 1 (deixou dados obrigatórios em branco na primeira tentativa) |
| Tarefa 9: Acessar as informações do local do evento                                   | 2  | 2  | 2   |

### 3.3.9.2 Resultado do questionário de usabilidade (SUS)

Seguem os escores obtidos a partir da aplicação do questionário de usabilidade baseado no instrumento SUS (*System Usability Scale*):

Tabela 5 - Pontuação no questionário de usabilidade (SUS)

| Usuário   | Nível de usuário | Pontuação Final |
|-----------|------------------|-----------------|
| Usuário 1 | Básico           | 90              |
| Usuário 2 | Intermediário    | 100             |
| Usuário 3 | Avançado         | 87,5            |

A pontuação média obtida no presente estudo, de 92,5 pontos representa um valor superior à média comumente citada na literatura (68 pontos) e sua interpretação é melhor representada quando ilustrada em uma curva normalizada, destacando os escores em percentis (medidas que dividem a amostra ordenada - por ordem crescente dos dados - em 100 partes), conforme a Figura 27 (SAURO, 2011b).

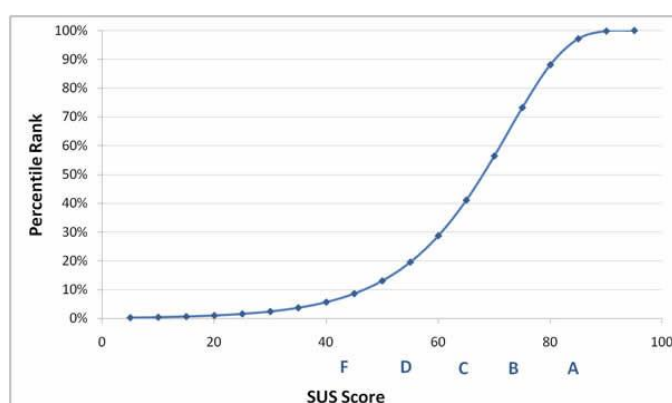


Figura 29 – Graduação divisão em percentis dos escores (SAURO, 2011b).

A média de 92,5 pontos corresponde aproximadamente ao percentil 94, indicando que o escore é maior que 94% das aplicações testadas, conforme a metodologia, e resulta em uma graduação “A”.

Sauro (2011b), afirma que o questionário de usabilidade não detecta problemas de usabilidade e tem como finalidade disponibilizar informação acerca da percepção da usabilidade a partir do ponto de vista do usuário. Importante consideração deve ser feita acerca da pequena amostra utilizada e do entendimento de que tal valor não representa um diagnóstico de usabilidade e sim uma ferramenta para avaliar a percepção do usuário ao longo das fases de desenvolvimento da aplicação. A partir

dessa percepção é possível realizar ciclos de atualização e melhorias para que pontos considerados não adequados sejam aperfeiçoados até a disponibilização da versão final do aplicativo.

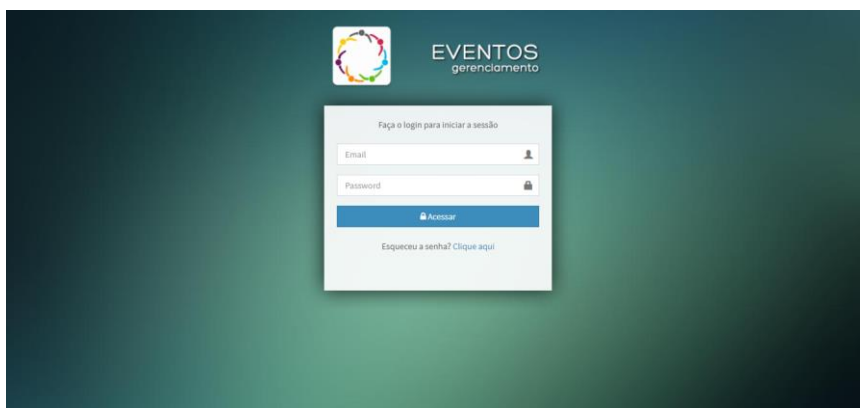
### 3.4 APRESENTAÇÃO DA APLICAÇÃO

A partir dos protótipos, foram elaboradas as telas da aplicação em linguagem HTML com a utilização do *framework7* para elaboração do *front end* da aplicação móvel.

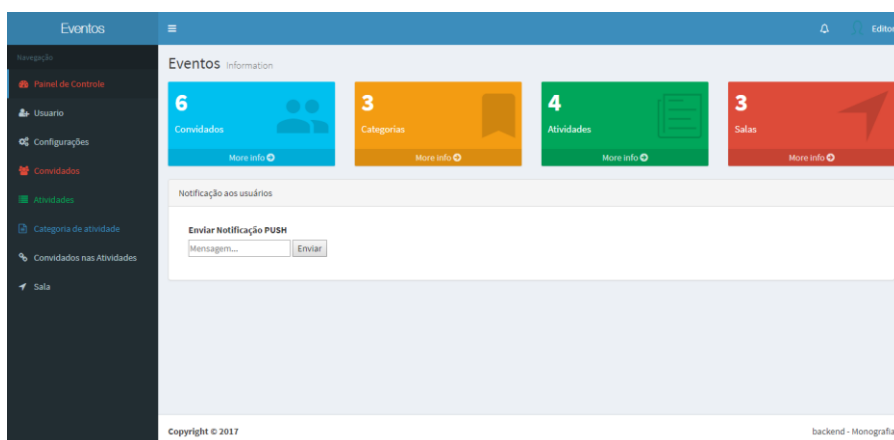
#### 3.4.1 *Back End* da Aplicação

O gerenciamento dos dados da aplicação, bem como o envio de mensagens *push* para os usuários são realizados por meio de uma aplicação desenvolvida em linguagem PHP, implementada utilizando duas ferramentas livres e de código aberto: o *framework* Laravel (OTWELL, 2017) em conjunto com a ferramenta *Crudbooster* (CROCODIC STUDIO, 2017), para geração de camadas de visão e controle das operações CRUD do *back end*. As permissões de acesso configuradas na administração do *back end* definem o tipo de acesso concedido a um usuário cadastrado no sistema no que diz respeito às tarefas de criação, visualização, atualização e exclusão de dados da base MySQL que serve como fonte de informações para a aplicação móvel proposta. Os dados da referida base são disponibilizados através de uma *API REST* para consumo pela aplicação móvel, nas diferentes requisições do usuário.

A Figura 30 ilustra a tela de *login* do *back end*. Foram definidos dois perfis de usuários: administrador e editor. O usuário com permissão de administrador pode realizar a alteração dos perfis de todos os usuários além de realizar alterações nas configurações do menu e aparência do *web service*.

Figura 30 - *Back end*: tela de acesso

A Figura 31 apresenta a tela inicial do *back end*, quando acessado por um usuário com permissão de editor. O envio de notificações, para os dispositivos móveis com a aplicação instalada, é realizado a partir da tela inicial com o preenchimento do formulário.

Figura 31 - *Back end*: tela inicial

A partir dessa tela, o usuário pode acessar as funcionalidades de gerenciamento de dados através do menu lateral que permite acesso às funções CRUD das tabelas do banco de dados (exemplo de gerenciamento de atividades na Figura 32). As informações são disponibilizados pela API REST em notação JSON, para que sejam consumidas pela aplicação móvel.

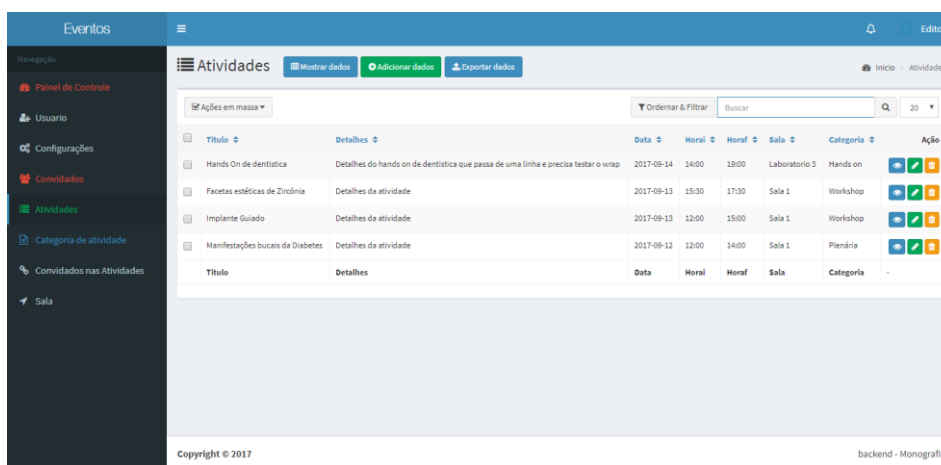


Figura 32 - *Back end*: gerenciamento de atividades

### 3.4.2 Aplicação Móvel

A aplicação proposta no presente trabalho foi exportada para as plataformas Android e IOS. As figuras expostas nesta seção foram obtidas a partir da execução do aplicativo em um emulador da plataforma Android. As telas da aplicação foram divididas em função dos requisitos funcionais, a partir das necessidades apontadas. O layout sugerido para o aplicativo utiliza-se de ícones a fim de facilitar o entendimento das funções disponíveis.

A partir da tela inicial (Figura 33) é possível acessar praticamente todas as funções disponíveis ao usuário. O menu lateral (Figura 34) apresenta um atalho para as referidas funções além de incluir acesso às notícias (informações postadas no blog), acesso ao *twitter* do evento e possibilidade de *login* e *logout* da aplicação.

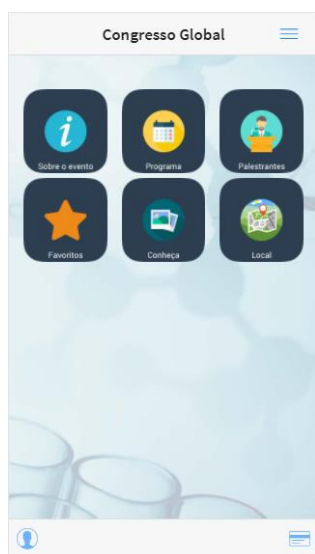


Figura 33 - Aplicação móvel: tela inicial

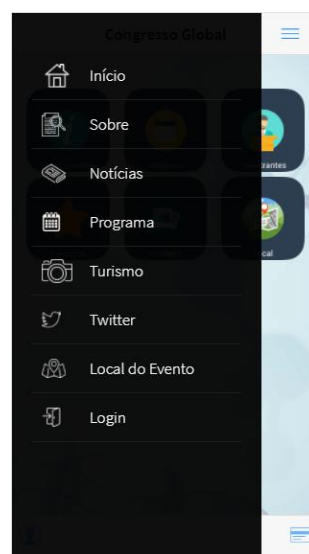


Figura 34 - Aplicação móvel: menu lateral

A Figura 35 ilustra as informações referentes à descrição do evento, obtidas a partir dos dados cadastrados no *back end*. Acessando o programa do evento, são exibidas todas as atividades com suas respectivas categorias, títulos, local, horário e data conforme a Figura 36. Na lista de atividades é possível filtrar os resultados por categoria através do ícone localizado no canto superior direito. Após a seleção da categoria desejada, a lista de atividades é atualizada e posteriormente o usuário pode limpar o filtro para que seja exibida a grade completa (Figura 37).

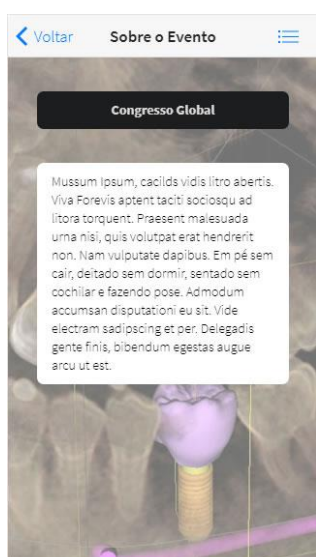


Figura 35 - Aplicação móvel: Página "sobre o evento"



Figura 36 - Aplicação móvel: programa do evento

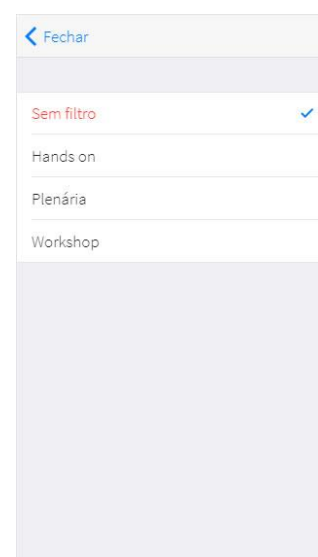


Figura 37 - Aplicação móvel: filtrar categorias



A fim de validar a proposta de aplicativo multiplataforma em razão do requisito não funcional elencado no item 3.1.2 do presente trabalho, foi realizado teste no sistema IOS a partir da emulação de um dispositivo iPhone 5S com a versão de sistema IOS 9.2, conforme a Figura 38. A aplicação executada no sistema IOS obteve desempenho idêntico à execução no sistema Android e foram identificadas pequenas diferenças visuais em botões e ícones que não interferem na usabilidade.



Figura 38 - Aplicação móvel em execução no emulador do dispositivo iPhone 5S (sistema IOS 9.2)

## 4 CONCLUSÃO

Atualmente, a vasta gama de aplicações de dispositivos móveis possibilita a simplificação de tarefas cotidianas, além da facilidade e agilidade na disseminação de informações. Podemos interagir com milhares de pessoas em tempo real e praticamente em qualquer local onde haja conexão viável com a internet.

Na organização e realização de um evento acadêmico, a divulgação de informações é um fator de extrema importância, influenciando diretamente no número de participantes ou mesmo na viabilidade do evento. A partir dos recursos presentes nos dispositivos móveis é possível garantir atualização constante de cronogramas, envio de destaques relacionados ao evento, além de disponibilizar aos participantes meios cômodos de planejamento no que diz respeito a interesses individuais por categorias e assuntos.

A implementação da aplicação proposta neste trabalho evidencia a possibilidade do desenvolvimento multiplataforma a partir de apenas um código fonte, por meio da utilização do framework *Cordova*, otimizando o processo de entrega para diferentes dispositivos móveis e simplificando o desenvolvimento. A partir do uso de tecnologias web e linguagens como HTML5, CSS e JavaScript, foi possível gerar uma aplicação híbrida e executá-la em dispositivos móveis reais (celular com sistema operacional Android e *tablet* iPad mini com sistema IOS). A estratégia de desenvolvimento da API, do *back end* da aplicação, com a arquitetura *REST*, se mostrou eficiente e com relativa simplicidade. A comunicação através de mensagens HTTP e a exposição dos dados por URIs dispensa a criação de protocolos específicos para uma aplicação, simplificando o desenvolvimento.

A etapa de prototipação de baixa fidelidade permitiu uma revisão dos requisitos propostos além de auxiliar na definição da forma de interação com o usuário, facilitando a etapa de desenvolvimento. A qualidade da interação do usuário foi avaliada por meio de teste de usabilidade e avaliação do índice de sucesso em uma lista de tarefas que representa as principais funções da aplicação. A observação do usuário durante a execução das tarefas e os resultados dessa avaliação, proporcionaram o ajuste de pequenas falhas que não ficaram evidentes durante o

desenvolvimento do aplicativo, como por exemplo alguns problemas de transição de telas durante a navegação.

Dentre as dificuldades encontradas, vale destacar que algumas instruções utilizadas nos *plug-ins* do *framework* Cordova, possuem parâmetros diferenciados para cada uma das plataformas (Android e IOS), bem como formas distintas de armazenamento e recuperação de dados. Em vista da diferença da estrutura dos sistemas operacionais, em alguns layouts de telas houve a necessidade de ajustes finos na folha de estilos a fim de manter similaridade do aspecto da aplicação nos diferentes dispositivos testados.

Como sugestão para continuidade do projeto e trabalhos futuros, pode-se apontar:

- Implementar funcionalidade adicional de controle de entrada de alunos em atividades restritas, a partir da leitura dos códigos de barras gerados na credencial;
- Implementar funcionalidade de confirmação de participação em evento por meio de identificação do dispositivo dentro da sala da atividade em questão;
- Implementar geração automatizada de certificados a partir da leitura dos códigos de barra dos participantes nas diferentes atividades;
- Implementar geração automatizada de *Proceedings (anais)* do evento a partir de um modelo padronizado;
- Implementar geração automatizada dos pacotes de instalação para as diferentes plataformas a partir de uma interface no *back end*;
- Ampliar o número de participantes nos testes de usabilidade e questionário SUS.

## REFERÊNCIAS

- ASTAH. **Astah Community - Free UML Modeling tool**, 2017. Disponível em: <<http://astah.net/editions/community>>. Acesso em: 7 dez. 2017
- AUSTIN, D. et al. Web services architecture requirements. **W3C Working Group Notes**, 2004.
- BROOKE, J. SUS: a retrospective. **Journal of usability studies**, v. 8, n. 2, p. 29–40, 2013.
- COSTA, L. F. DA; RAMALHO, F. A. A usabilidade nos estudos de uso da informação: em cena usuários e sistemas interativos de informação. **Perspectivas em Ciência da Informação**, v. 15, n. 1, p. 92–117, 2010.
- CROCODIC STUDIO. **CRUDBooster**. Disponível em: <<https://crudbooster.com/>>. Acesso em: 11 nov. 2017.
- DEFINITY TEAM. **32ª Reunião Anual SBPqO – Apps para Android no Google Play**, 2017. Disponível em: <<https://play.google.com/store/apps/details?id=com.definitysolutions.apps.sbpqo32>>. Acesso em: 5 dez. 2017
- DEV MEDIA. **Como funcionam as aplicações web**. Disponível em: <<http://www.devmedia.com.br/como-funcionam-as-aplicacoes-web/25888>>. Acesso em: 28 jul. 2017.
- FAYAD, M.; SCHMIDT, D. C. **Object-oriented application frameworks** *Communications of the ACM*, 1997.
- FIELDING, R. T. **Architectural styles and the design of network-based software architectures**. University of California, Irvine, , 2000.
- FREIRE, P. J.; RIBEIRO, R. Revisão de Literatura de Frameworks de Desenvolvimento Móvel Multiplataforma CAPSI/2013. 2013.
- FURLAN, J. D. **Modelagem de objetos através da UML-the unified modeling language**. Makron books, 1998.
- GOOGLE DEVELOPERS. **Cloud Messaging | Google Developers**. Disponível em: <<https://developers.google.com/cloud-messaging/>>. Acesso em: 23 nov. 2017.
- HANOV, S. **WebSequenceDiagrams** Waterloo, Ontario, Canada Hanov Solutions Inc., , 2007. Disponível em: <<https://www.websequencediagrams.com>>
- HEUSER, C. A. **Projeto de banco de dados: Volume 4 da Série Livros didáticos informática UFRGS**. Rio Grande do Sul: Bookman Editora, 2009.

JONGDE SOFTWARE LLC. **Mockplus**, 2017. Disponível em: <<https://www.mockplus.com>>. Acesso em: 7 nov. 2017

JÚNIOR, G. DE P. S. et al. **Aplicação Multiplataforma da Realidade Aumentada Móvel para Geolocalização utilizando o PhoneGap**. Porto Alegre: Sociedade Brasileira de Computação, 2012

MELO, A. A.; NASCIMENTO, M. G. F. **PHP Profissional**. 1. ed. São Paulo: Novatec Editora, 2007.

MILANI, A. **MySQL - Guia do Programador**. São Paulo: Novatec Editora, 2007.

NIELSEN, J.; LORANGER, H. **Usabilidade na web**. Rio de Janeiro: Elsevier Brasil, 2007.

OTWELL, T. **Laravel - The PHP Framework For Web Artisans**. Disponível em: <<https://laravel.com/>>. Acesso em: 10 nov. 2017.

PALIARI, M. **Como funciona um Webservice REST**. Disponível em: <<http://www.matera.com/br/2012/10/22/como-funciona-um-webservice-rest/>>. Acesso em: 20 ago. 2017.

PRATES, R. O.; BARBOSA, S. D. J. **Avaliação de Interfaces de Usuário – Conceitos e Métodos**. Congresso Nacional da Sociedade Brasileira de Computação. **Anais...**Rio de Janeiro: Anais do XXIII Congresso Nacional de Informática, 1991

RECUERO, R. **Redes sociais na internet**. Porto Alegre: Sulina, 2009.

RONDON, T. B. **Arquitetura REST e o serviço web “RESTful”**. Disponível em: <<http://sao-paulo.pm.org/pub/arquitetura-rest-e-o-servico-web-restful->>. Acesso em: 10 fev. 2015.

SACCOL, A. Z.; REINHARD, N. **Tecnologias de informação móveis, sem fio e ubíquas: definições, estado-da-arte e oportunidades de pesquisa** *Revista de Administração Contemporânea*, 2007.

SAURO, J. **MeasuringU: Measuring Usability with the System Usability Scale (SUS)**. Disponível em: <<https://measuringu.com/sus/>>. Acesso em: 6 out. 2017.

SAURO, J. SUSTisfied? little-known System Usability Scale facts. **UX Magazine**, v. 10, n. 3, p. 2011–2013, 2011b.

TEIXEIRA, F. **O que é o SUS (System Usability Scale) e como usá-lo em seu site**. Disponível em: <<https://brasil.uxdesign.cc/o-que-é-o-sus-system-usability-scale-e-como-usá-lo-em-seu-site-6d63224481c8>>. Acesso em: 6 out. 2017.

THE APACHE SOFTWARE FOUNDATION. **Apache Cordova**. Disponível em: <<http://cordova.apache.org/>>. Acesso em: 21 abr. 2015.

THONSON, L.; WELLING, L. **PHP e MySQL: desenvolvimento web**. Rio de Janeiro: Editora Campus, 2005.

TRYMYUI. **SUS: System Usability Scale**. Disponível em: <<https://www.trymyui.com/sus-system-usability-scale>>. Acesso em: 10 out. 2017.

UFENESP. **MED Eventos – Apps para Android no Google Play**, 2017. Disponível em: <<https://play.google.com/store/apps/details?id=br.ufenesp.medeventos>>. Acesso em: 5 dez. 2017

WARGO, J. M. **Apache Cordova 3 programming**. San Francisco: Addison-Wesley Professional, 2013.

ZEMEL, T. **JSON: JavaScript Object Notation**. Disponível em: <<http://desenvolvimentoparaweb.com/javascript/json-javascript-object-notation/>>. Acesso em: 23 jul. 2017.

## APÊNDICE A – QUESTIONÁRIO DE AVALIAÇÃO DE USABILIDADE

Idade: \_\_\_\_\_ Formação: \_\_\_\_\_

Prezado participante, as perguntas deste questionário visam avaliar a facilidade de uso do aplicativo que você acabou de utilizar

Responda às perguntas abaixo, utilizando-se da seguinte escala:

|   |                               |   |   |   |                               |
|---|-------------------------------|---|---|---|-------------------------------|
|   | Discordo<br>completament<br>e |   |   |   | Concordo<br>completament<br>e |
|   |                               |   |   |   |                               |
| 1 | 2                             | 3 | 4 | 5 |                               |

| Pergunta  | Escala |   |   |   |   |
|---|--------|---|---|---|---|
| 1. Eu acho que gostaria de usar esse aplicativo com frequência.                                     | 1      | 2 | 3 | 4 | 5 |
| 2. Eu achei o aplicativo desnecessariamente complexo.   | 1      | 2 | 3 | 4 | 5 |
| 3. Eu achei o aplicativo fácil de usar.   | 1      | 2 | 3 | 4 | 5 |
| 4. Eu acho que precisaria de ajuda de uma pessoa com conhecimentos técnicos para usar o aplicativo. | 1      | 2 | 3 | 4 | 5 |
| 5. Eu acho que as várias funções do aplicativo estão muito bem integradas.                          | 1      | 2 | 3 | 4 | 5 |
| 6. Eu acho que o aplicativo apresenta muita inconsistência.   | 1      | 2 | 3 | 4 | 5 |
| 7. Eu imagino que as pessoas aprenderão como usar esse aplicativo rapidamente.                      | 1      | 2 | 3 | 4 | 5 |
| 8. Eu achei o aplicativo atrapalhado de usar.   | 1      | 2 | 3 | 4 | 5 |
| 9. Eu me senti confiante ao usar o aplicativo.  | 1      | 2 | 3 | 4 | 5 |
| 10. Eu precisei aprender várias coisas novas antes de conseguir usar o aplicativo.                  | 1      | 2 | 3 | 4 | 5 |

### Consentimento

Afirmo que sou maior de 18 anos e desejo participar do estudo de usabilidade da aplicação proposta no presente trabalho. Todas as informações coletadas neste estudo são confidenciais, e meu nome não será identificado em momento algum. Estou ciente que posso fazer perguntas ou desistir da colaboração em qualquer momento, sem qualquer tipo de penalidade.

Assinatura: \_\_\_\_\_ Data: \_\_\_/\_\_\_/\_\_\_