|  |
| --- |
| ESTG_simb.08.jpg |
|  |
| **WhatToDo** |
| [Subtítulo do Projeto/Estágio] |
|  |
| Relatório de Projeto Aplicado submetido como requisito parcial para obtenção do grau de Licenciatura em Engenharia Informática. |
| Orientador: Dr.ª Natália Gomes |
| **João Paulo Brás Delgado** |
|  |
| **[Mês] | [Ano]** |

[Dedicatorias/Frases]

**Agradecimentos**

[...]

**Resumo**

[...]

|  |  |
| --- | --- |
| **Palavras-chave:** | [Palavra-chave 1], [Palavra-chave 2], [Palavra-chave 3], [Palavra-chave 4], [Palavra-chave 5], [Palavra-chave 6]. |

**Abstract**

[...]

|  |  |
| --- | --- |
| **Keywords** | [Keyword 1], [Keyword 2], [Keyword 3], [Keyword 4], [Keyword 5], [Keyword 6]. |

Índice

[Índice de Figuras vi](#_Toc487016316)

[Índice de Tabelas vii](#_Toc487016317)

[Siglas 1](#_Toc487016318)

[1. Introdução 2](#_Toc487016319)

[1.1. Contextualização 2](#_Toc487016320)

[1.2. Motivação 2](#_Toc487016321)

[1.3. Descrição do problema 2](#_Toc487016322)

[1.4. Objetivos 3](#_Toc487016323)

[1.5. Estrutura do documento 4](#_Toc487016324)

[2. Estado da Arte 5](#_Toc487016325)

[2.1. Aplicações de âmbito distrital 6](#_Toc487016326)

[2.1.1. Lisbon by Triposo 6](#_Toc487016327)

[2.1.2. Oporto Insight 6](#_Toc487016328)

[2.1.3. Aqui Há Beira 7](#_Toc487016329)

[2.2. Aplicações de âmbito nacional 7](#_Toc487016330)

[2.2.1. Visit Portugal 7](#_Toc487016331)

[2.3. Aplicações âmbito global 8](#_Toc487016332)

[2.3.1. TripAdvisor 8](#_Toc487016333)

[2.4. Funcionalidades a reter 9](#_Toc487016334)

[2.5. Análise critica do estado da arte 9](#_Toc487016335)

[3. Metodologia e Funcionalidades esperadas 10](#_Toc487016336)

[3.1. Metodologia 10](#_Toc487016337)

[3.2. Descrição das Tarefas 11](#_Toc487016338)

[3.3. Requisitos do sistema 11](#_Toc487016339)

[4. Modelação 12](#_Toc487016340)

[4.1. Diagrama de Contexto 12](#_Toc487016341)

[4.2. Diagrama de Casos de Uso 12](#_Toc487016342)

[4.3. Diagramas de Sequência/ Descrição dos casos de Uso 12](#_Toc487016343)

[4.4. Diagramas de Classes 2 31](#_Toc487016344)

[4.5. Diagrama de Atividades 31](#_Toc487016345)

[4.6. Diagrama de Estados 31](#_Toc487016346)

[4.7. Diagrama de Componentes 31](#_Toc487016347)

[4.8. Diagrama de Instalação (talvez, ñ aplicável) 31](#_Toc487016348)

[4.9. Modelo ER e Semântica dos dados 31](#_Toc487016349)

[4.9.1. Modelo Entidade Relação 31](#_Toc487016350)

[4.9.2. Dicionário de Dados 31](#_Toc487016351)

[4.10. Tecnologias 32](#_Toc487016352)

[5. Implementação da Solução 33](#_Toc487016353)

[5.1. Introdução 33](#_Toc487016354)

[5.2. CODE (que seja necessário explicar) 33](#_Toc487016355)

[5.3. BD’S 33](#_Toc487016356)

[5.4. Interface 33](#_Toc487016357)

[5.5. Componentes Gráficos 33](#_Toc487016358)

[6. Testes 34](#_Toc487016359)

[7. Conclusões e Trabalho futuro 35](#_Toc487016360)

[7.1. Conclusões 35](#_Toc487016361)

[7.2. Trabalho Futuro 35](#_Toc487016362)

[7.3. 35](#_Toc487016363)

[Bibliografia 36](#_Toc487016364)

[[Anexo A] 37](#_Toc487016365)

[[Anexo B] 38](#_Toc487016366)

# Índice de Figuras

**No table of figures entries found.**  
This is an automatic table of contents. To use it, apply heading styles (on the Home tab) to the text that goes in your table of contents, and then update this table. If you want to type your own entries, use a manual table of contents (in the same menu as the automatic one).

# Índice de Tabelas

[Tabela 1 9](#_Toc484529645)

Lista de Acrónimos

PoI – point of interess

BackEnd –

API –

TI –

Mb –

XP –

UML –

API Rest –

# Introdução

Trabalho desenvolvido pelo aluno João Paulo Brás Delgado, no âmbito da disciplina de projeto da Licenciatura de Engenharia Informática no Instituto Politécnico da Guarda. Como etapa fundamental e final para a obtenção do grau académico de Licenciatura.

Atualmente a tecnologia é presente no quotidiano da sociedade atual, o número de dispositivos móveis tem aumentado exponencialmente [REF] fazendo com que o acesso à informação digital uma prática comum e fácil. Assim surge a oportunidade de difundir informação aos utilizadores desta tecnologia.

## Contextualização

Como requisito para a finalização da disciplina de projeto é necessário implementar uma solução, para resolver o problema existente, a divulgação de eventos num determinado local. Para tal é necessário a implementação dos conhecimentos adquiridos durante a licenciatura, conhecimentos estes, na área da modulação de software, programação, organização e gestão de recursos e conteúdos, ainda se prevê adquirir novos conhecimentos para a realização deste projeto.

## Motivação

O desenvolvimento deste projeto passa pela realização pessoal, finalização da licenciatura, consolidação e oportunidade de adquirir novos conhecimentos. Por outro lado, este projeto tem grande viabilidade e interesse, quer pela falta de softwares do género, quer pela utilidade apresentada pela aplicação.

## Descrição do problema

Atualmente o turista tem necessidade de planificar a sua visita, seja para aproveitar o tempo ao máximo, ou simplesmente para não perder um ponto de interesse (PoI) do seu agrado. Muitas vezes esta planificação é efetuada através de revistas, jornais, panfletos, guias turísticos, tudo em formato de papel, ou por outro lado em formato digital em que o conteúdo sofre poucas atualizações ou muitas vezes caindo em desuso por falta de administração competente.

No caso da informação recolhida via não informatizada, por norma, encontra-se desatualizada ou como é espectável sem capacidade de resposta a alterações súbitas, tais como, condições climatéricas muito adversas, obras de restauração e etc. Outro problema inerente é a falta de triagem por grupos etários nos eventos/pontos de interesse a decorrer, dificultando o planeamento das atividades levando muitas vezes ao desinteresse do turista[1].

Surgem então vários problemas, que faz com que muitas vezes os turistas fiquem insatisfeitos com a localidade que visitam, deste modo levantando um mau *feedback*. Assim para evitar situações deste tipo pretende-se implementar uma solução que combata estas dificuldades. Uma aplicação móvel atualizada em tempo real, com os pontos de interesse de uma cidade e principalmente focada nos eventos disponíveis na numa determinada localidade

## Objetivos

Como objetivo pretende-se o desenvolvimento de uma aplicação de turismo atualizada em tempo real que disponibilize informação dos diferentes PoI e eventos de um determinado local. Estes PoI e eventos devem estar organizados por faixa etária e/ou por categorias.

Sendo um projeto extenso prevê-se para âmbito de licenciatura apenas o desenvolvimento do *backEnd* necessário ao funcionamento da aplicação a desenvolver posteriormente.

Tendo como objetivos:

* Desenvolver uma solução eficiente para armazenar os dados referentes aos PoI e os eventos.
* Garantir que os dados armazenados estão disponíveis às diferentes plataformas, Android, iOS e Desktop.
* Permitir ao administrador através de um site criar/gerir novos eventos/locais de interesse e garantir que estes são organizados por diferentes categorias e ou faixa etária.
* Permitir ao administrador pesquisar por pontos de interesse diretamente de uma API disponível no mercado, para facilitar a introdução dos dados de localização dos PoI.
* Permitir ao administrador enviar notificações aos utilizadores.

Outro ponto a salientar apresenta-se com o facto de não se esperar que o administrador tenha qualquer tipo de formação especializada na área das TI, pelo que o processo de criação/gestão dos eventos seja simples e com uma curva de aprendizagem pequena.

## Estrutura do documento

[...

# Estado da Arte

Neste capitulo vamos abordar a importância das aplicações mobile e a sua tendência de desenvolvimento. Explorar alguns exemplos de aplicações semelhantes à aplicação a desenvolver, e por fim dar uma opinião critica sobre o estado da arte.

Sendo o leque de aplicações móveis disponíveis no mercado enorme, e de modo a existir um termo de comparação e análise coerente entre aplicações, a secção de aplicações existentes será subdividida em três. Aplicações de âmbito distrital, nacional e global. A seleção destas aplicações foi efetuada através do *feedback* positivo disponibilizado por parte dos utilizadores nas respetivas *Stores*, outro requisito considerado foi apenas o uso das funcionalidades grátis disponibilizadas por cada uma destas aplicações.

No caso de aplicação de âmbito nacional, optou-se por escolher a aplicação oficial do órgão máximo do turismo em Portugal, Turismo de Portugal.

Como guia de análise às aplicações vamos ter em conta as 25 boas práticas fornecidas pela entidade Google descritas no documento “Engage Users and Drive Conversions”[2].

Com esta análise pretende-se estruturar ideias e definir objetivos para o projeto. Segue então a análise dos sistemas mencionados.

## Aplicações de âmbito distrital

### Lisbon by Triposo

A aplicação *Lisbon* é um sistema que pretende dar a conhecer todas as atividades e PoI disponíveis em Lisboa. Tem como módulos atrações turísticas, restauração, vida noturna, hotéis, visitas e espetáculos, atividades, guias de viagem, e algumas ferramentas úteis a turistas, tais como, conversor de moeda, temperatura, fuso horário e algumas frases essenciais na língua portuguesa.

Sendo uma aplicação específica, apenas de âmbito distrital, é bastante completa e aborda um grande leque de atividades disponíveis. Tem um design apelativo e de fácil navegação.

Como principais pontos negativos temos a dimensão da aplicação (336MB Android/ 600Mb iOS), que poderá afastar muitos turistas devido às limitações existentes nos planos de tráfego em roaming, e por outro lado muitas vezes estes têm espaço limitado nos dispositivos. Outro ponto negativo a salientar encontrado na aplicação é a falta de informação de eventos esporádicos, que apenas se realizam em datas definidas.

### Oporto Insight

O seguinte sistema divulga os PoI e atividades existentes no Porto. Inicialmente a aplicação esta dividida em 5 categorias, como dito anteriormente, funcionalidades pagas serão descartadas, assim sendo vamos focar-nos apenas em três secções grátis, locais a visitar, testemunhos de pessoas influentes, contatos de guias turistas.

Na secção dos locais a visitar esta divide-se em sítios a visitar (turismo cultural), restauração, dormidas, vida noturna, compras e informações uteis (contatos telefónicos de instituições de apoio à comunidade, tais como, policia municipal, bombeiros e etc.).

De uma forma geral a aplicação encontra-se bem conseguida, conjugação de cores apelativas, com um bom leque de informação e conteúdo ao utilizador.

Como notas negativas a falta de usabilidade da aplicação pode afastar os turistas menos pacientes, de salientar que tem muitas funcionalidades que apenas estão disponíveis caso se compre a versão *Pro* da aplicação, outro fator a mencionar é a falta de divulgação de eventos não periódicos.

### Aqui Há Beira

Aplicação destinada a divulgação da Beira Alta, tendo como objetivo principal dar a conhecer os eventos realizados nesta zona. É possível ver cinco áreas de foco, eventos, gastronomia, alojamento, o que fazer e por último, locais a visitar. De forma geral a aplicação tem um bom conceito, mas o seu desenvolvimento deixa muito a desejar, é de notar que a aplicação não é nativa nos diferentes sistemas operativos, o que resulta numa baixa performance. A termo de exemplo, até podermos utilizar a aplicação temos um tempo de espera de aproximadamente dez segundos, entre outros problemas. Em termos de usabilidade preenche poucos requisitos fazendo a navegação difícil e confusa, muitas das categorias apresentadas, em algumas localidades, encontram-se vazias resultando em falta de informação para o utilizador e tempo perdido em navegação.

Apenas de salientar que em relação às restantes aplicações analisadas esta divulga eventos não periódicos, o que acaba por ser uma funcionalidade que poucas aplicações têm e que distingue esta aplicação positivamente relativamente às outras.

## Aplicações de âmbito nacional

[REF]

### Visit Portugal

A aplicação Visit Portugal pretende divulgar a maior parte dos PoI existentes em Portugal. Dispondo de várias categorias, natureza, sol e mar, arte e cultura, entre muitas outras. Pontos a salientar é a possibilidade de usar realidade aumentada para encontrar PoI perto do utilizador, garantindo uma experiencia diferente ao utilizador, outro aspeto a salientar é existir uma categoria dedicada a pessoas com necessidades especiais, indicando quais as atividades recomendadas para estas. Por outro lado, é possível ter a perceção que a aplicação não é dedicada ao sistema operativo o que resulta numa má experiência e fraca performance existindo grandes períodos de *loading*.

Em usabilidade existem várias falhas, sendo as mais significativas relacionadas com a pesquisa. Outro fator negativo encontrasse na ausência de imagens na aplicação [figx].

Em resumo uma aplicação com bom conteúdo, mas mal explorada.

é após a escolha da categoria é mostrado uma lista com os diferentes PoI, apenas em formato de texto faltando informação com imagens tornando o uso da aplicação cansativa.

## Aplicações âmbito global

### TripAdvisor

Por último temos uma aplicação de cariz internacional, TripAdviser aplicação com mais de “[...] 500 milhões de avaliações e opiniões de viajantes, o [...]”, o que a torna uma das mais utilizadas. De uma forma geral a aplicação encontra-se bem conseguida e com poucas falhas. Aplicação dividida em 7 categorias, salientando as categorias “Things to Do” e “Destination Guides” sendo este o foco da nossa aplicação. Os pontos positivos desta aplicação são a sua execução, variedade de conteúdo e a proximidade com os utilizadores, através de um bom sistema de *feedback* e comentários. Como pontos negativos apenas salientar a falta de conteúdo em PoI menores, tais como cidades mais pequenas ou mesmo aldeias, e a falta de capacidade de divulgar eventos sem periodicidade.

## Funcionalidades a reter

Após a análise das seguintes aplicações foi possível reter ideias e definir objetivos de forma mais clara e precisa. Em baixo encontra-se a Tabela 1com as funcionalidades prestadas à aplicação a desenvolver pelas aplicações analisadas.

Tabela

|  |  |
| --- | --- |
| Funcionalidades | Pesquisa por localização. |
| Pesquisa por categoria. |
| Registo de PoI. |
| Registo de eventos não periódicos. |
| Ferramentas úteis a turistas (temperatura, fuso horário, etc.). |
| Sistema de classificação de atividades. |
| Sistema de comentários e *feedbacks.* |
| Sinalização de eventos/PoI para pessoas com deficiência. |
| Indexação da pesquisa. |
| Sistema de favoritos |

## Análise critica do estado da arte

Colocar tabela feita pelo pdf da Google, encontra-se em excell. E DePoIs fazer um pequeno texto a comentar.

Colocar transições entre capítulos

# Metodologia e Funcionalidades esperadas

Neste capítulo pretende-se apresentar e definir a metodologia a utilizar no projeto, definir tarefas e por último indicar resultados esperados.

## Metodologia

Para o desenvolvimento deste projeto pretende-se utilizar o movimento Ágil. Ágil é uma metodologia de desenvolvimento de *software* que equilibra três variáveis essenciais, custo, tempo e qualidade do produto. Esta metodologia é essencialmente utilizada em projetos curtos, com requisitos em constante mudança, com necessidade de ver resultados frequentemente e com equipas pequenas (no projeto a desenvolver composta por aluno e orientador). [REF]

Dentro da metodologia Ágil escolheu-se o *XP* (*Extreme Programming*). XP torna possível ter fases de escrita curtas, períodos contantes de codificação e sessões de reunião, validação e planificação frequentes e por último para que seja possível voltar a qualquer destas etapas sempre que necessário.

*XP* tem como atividades principais o planeamento, a projeção, a codificação e os testes. Assim aplicando ao projeto a desenvolver surgem quatro etapas:

* Planeamento, pretende-se discutir e documentar requisitos funcionais ao projeto.
* Projeto/Design, pretende-se documentar todo o projeto em linguagem *UML* (Unified Modeling Language).
* Codificação, pretende-se desenvolver a aplicação.
* Testes, fator de teste e validação do trabalho efetuado nas restantes iterações. Validação com o orientador do trabalho efetuado.

## Descrição das Tarefas

[...] Diagrama de Gant

## Requisitos do sistema

Distinguir requisitos funcionais de não funcionais?

A aplicação funcionará com três níveis de utilizadores, o utilizador administrativo que será responsável pela inserção e gestão dos PoI/Eventos e com capacidade de gerir permissões, o utilizador da aplicação não registado e o utilizador registado. Estes dois últimos funcionam como clientes, utilizando as funcionalidades prestadas pela aplicação a única diferença entre estes é a capacidade de o utilizador registado conseguir guardar PoI/Eventos nos favoritos, partilhar PoI/Eventos nas redes sociais (caso o utilizador permita) e comentar os diferentes PoI/Eventos existentes. Como descrito na secção 1.4. devido à dimensão do projeto apenas será desenvolvido o *backEnd* necessário ao funcionamento da aplicação final, deste modo apenas se falará das funcionalidades do utilizador administrativo, no entanto é de esperar que o *BackEnd* fique estruturado de forma a que encaixe na implementação aplicação cliente.

Temos as seguintes funcionalidades do utilizador administrado:

* Criar e editar Eventos.
* Criar e editar PoI.
* Pesquisar Eventos por categoria, faixa etária, coordenadas geográficas e nome.
* Pesquisar Pontos de Interesse por categoria, faixa etária, coordenadas geográficas e nome.
* Criar alertas ao utilizador (do tipo *push notification*).
* Visualizar estatísticas?

Pequeno texto de ligação com a modelação.

# Modelação

Referir que a modelação é referente a todo o Sistema (BackEnd/FrontEnd)[...].

## Diagrama de Contexto

[...] DONE

## Diagrama de Casos de Uso

[...] DONE

## Diagramas de Sequência/ Descrição dos casos de Uso

|  |  |
| --- | --- |
| Nome | Registar utilizador |
| Objetivo | Criar conta de utilizador |
| Atores envolvidos | URegistado |
| Pré-Condição |  |
| Prioridade |  |
| Fluxo Principal | 1. O ator seleciona a opção criar conta; 2. O sistema apresenta duas opções ao ator, criar conta utilizando rede social ou registo; 3. O utilizador seleciona criar conta com rede social; 4. O sistema recolhe as informações da rede social sobre o ator e preenche os campos com a informação recolhida. 5. O sistema cria utilizador registado e devolve mensagem: “Registo efetuado com sucesso”. |
| Fluxos Alternativos | 3A   * O ator seleciona criar conta com registo; * O sistema devolve uma nova janela com as informações necessárias ao registo; * O ator preenche os campos obrigatórios, e opcionalmente preenche os campos não obrigatórios; * O Sistema devolve mensagem: “Registo com sucesso” e termina o caso de uso. |
| Fluxos de Exceção | * Não tem conexão à internet.   4A   * A rede social não dá permissão à aplicação. E o sistema devolve a mensagem: “A rede social negou o registo, tente novamente, se o problema persistir contate um administrador.”.   5A   * O Sistema devolve mensagem: “Ocorreu um erro no registo, tente novamente”. |
| Pós-Condição |  |
| Casos de Teste | 1. Testar a criação através das diferentes redes sociais. Validar se a rede social devolve permissão positiva; 2. Verificar se os utilizadores ficam registados. Ao efetuar o registado, o sistema faz um pedido ao *webService* para validar se o utilizador foi registado e este devolve um valor do tipo *boolean* para confirmar o registo com sucesso ou não; 3. Verificar se o não preenchimento ou preenchimento incorreto dos campos obrigatórios (definir os campos) impede o registo do utilizador. 4. Antes de fazer o pedido à API Rest verificar a conexão do dispositivo móvel à internet. |

|  |  |
| --- | --- |
| Nome | Efetuar *login* |
| Objetivo | Aceder a conta pessoal de um utilizador registado |
| Atores envolvidos | URegistado/Administrador |
| Pré-Condição | Tirar linhas |
| Prioridade |  |
| Fluxo Principal | 1. O ator seleciona a opção *login;* 2. O sistema devolve duas opções. *Login* e *login* por rede social; 3. O ator seleciona login por rede social; 4. O sistema acede às *passwords* guardadas no dispositivo móvel e valida o *login* por rede social; 5. O sistema devolve uma mensagem: “*Login* efetuado com sucesso”. |
| Fluxos Alternativos | 3A   * O ator seleciona *login* simples; * O sistema acede às cadenciais guardadas no dispositivo móvel do ator e valida o *login*; * O sistema devolve uma mensagem: “*Login* efetuado com sucesso”.   3B   * O ator não preenche os dados de acordo com o formato indicado; * O sistema devolve uma mensagem: “Preencha os dados de acordo com os formatos válidos”. |
| Fluxos de Exceção | * Não tem conexão à internet.   5A   * O *login* está incorreto, o sistema devolve uma mensagem: “Cadenciais erradas, tente novamente.” |
| Pós-Condição |  |
| Casos de Teste | 1. Testar os diferentes tipos de *Login*; 2. Verificar se o dispositivo armazena as credenciais; 3. Verificar se a aplicação encripta a password antes de enviar o pedido ao webService; 4. Verificar se o sistema emite as mensagens corretas mediante os fluxos de exceção. 5. Antes de fazer o pedido à API Rest verificar a conexão do dispositivo móvel à internet. |

|  |  |
| --- | --- |
| Nome | Pesquisar Evento |
| Objetivo | Pesquisar pontos de interesse ou eventos |
| Atores envolvidos | UVisitante, URegistado |
| Pré-Condição |  |
| Prioridade |  |
| Fluxo Principal | 1. O ator seleciona a linguagem pretendida. 2. O ator insere uma localidade, nome do evento ou informação relevante à pesquisa ou pode escolher um filtro para que a pesquisa fique mais limitada; 3. O utilizador clica em pesquisar; 4. O sistema devolve os PoI e eventos de acordo com a pesquisa efetuada. |
| Fluxos Alternativos | 4A   * O sistema não devolve nenhum resultado na pesquisa, então sugere eventos/PoI próximos ao utilizador. |
| Fluxos de Exceção | * Não tem conexão à internet. |
| Pós-Condição |  |
| Casos de Teste | 1. Antes de fazer o pedido à API Rest verificar a conexão do dispositivo móvel à internet. |

|  |  |
| --- | --- |
| Nome | Partilha PoI/Evento nas Redes Sociais |
| Objetivo | Partilhar informações sobre PoI/Evento nas redes sociais |
| Atores envolvidos | URegistado |
| Pré-Condição | *Login* válido |
| Prioridade |  |
| Fluxo Principal | 1. O ator encontra-se no ecrã com o PoI/Evento pretendido, e seleciona a opção partilhar PoI/Evento; 2. O sistema devolve um ecrã com o PoI/Evento a partilhar e a opção de escrever uma mensagem se o ator pretender; 3. O ator seleciona a opção partilhar PoI/Evento; 4. O sistema devolve uma mensagem: “PoI/Evento partilhado com sucesso”. |
| Fluxos Alternativos | 3A   * O ator seleciona a opção cancelar; * O sistema retorna ao PoI/Evento selecionado anteriormente.   4A   * O sistema devolve uma mensagem: “Não foi possível partilhar o PoI/Evento tente novamente”; * O sistema apresenta de novo o PoI/Evento a partilhar. |
| Fluxos de Exceção | * Não tem conexão à internet. |
| Pós-Condição |  |
| Casos de Teste | 1. Verificar se o PoI/Evento foi partilhado na rede social; 2. Verificar se a mensagem consta da partilha. 3. Antes de fazer o pedido à API Rest verificar a conexão do dispositivo móvel à internet. |

|  |  |
| --- | --- |
| Nome | Editar perfil |
| Objetivo | Alterar as informações do perfil do ator |
| Atores envolvidos | URegistado |
| Pré-Condição | *Login* válido |
| Prioridade |  |
| Fluxo Principal | 1. O ator encontra-se no ecrã de perfil; 2. O ator altera as suas informações pessoais; 3. O ator seleciona a opção guardar alterações; 4. O sistema devolve uma mensagem: “Alterações efetuadas com sucesso”. |
| Fluxos Alternativos | 3A   1. O utilizar seleciona a opção cancelar;  * O sistema devolve uma mensagem: “Pretende cancelar as alterações?”; * O ator confirma a ação e o sistema descarta as alterações feitas pelo ator.   3B   * O utilizar seleciona a opção cancelar; * O sistema devolve uma mensagem: “Pretende cancelar as alterações?”; * O ator cancela a ação e continua as alterações ao perfil com as informações digitadas anteriormente. |
| Fluxos de Exceção | * Não tem conexão à internet.   4A   * O sistema devolve uma mensagem: “Não foi possível efetuar as alterações pretendidas.”. |
| Pós-Condição |  |
| Casos de Teste | 1. Verificar se as alterações são descartadas após o utilizador cancelar o editar perfil; 2. Verificar se as alterações são efetuadas na aplicação e na base de dados. 3. Antes de fazer o pedido à API Rest verificar a conexão do dispositivo móvel à internet. |

|  |  |
| --- | --- |
| Nome | Adicionar Evento/PoI aos favoritos |
| Objetivo | Adicionar Evento/PoI aos favoritos do utilizador. |
| Atores envolvidos | URegistado |
| Pré-Condição | *Login* válido. |
| Prioridade |  |
| Fluxo Principal | 1. O ator no ecrã de visualização dos detalhes de Eventos/PoI seleciona a opção adicionar evento/PoI aos favoritos; 2. O sistema devolve uma confirmação através da alteração gráfica do ícone dos favoritos. |
| Fluxos Alternativos |  |
| Fluxos de Exceção | * Não tem conexão à internet.   2A   * O sistema devolve uma mensagem: “Não foi possível adicionar aos favoritos.”. |
| Pós-Condição |  |
| Casos de Teste | 1. Verificar se os favoritos se encontram na área dos favoritos do ator. Através de um pedido ao webservice e este devolve que o determinado favorito se encontra na base de dados. 2. Antes de fazer o pedido à API Rest verificar a conexão do dispositivo móvel à internet. |

|  |  |
| --- | --- |
| Nome | Eliminar PoI/Evento dos favoritos |
| Objetivo | Retirar um determinado PoI/Evento dos favoritos |
| Atores envolvidos | URegistado |
| Pré-Condição | *Login* válido |
| Prioridade |  |
| Fluxo Principal | 1. O utilizador encontra-se na área de favoritos; 2. Clica na opção remover, ou arrasta para o lado esquerdo o PoI/Evento; 3. O PoI/Evento é removido da área de favoritos. |
| Fluxos Alternativos | 1A   * O utilizador encontra-se na área da descrição do PoI/Evento; * O ator clica no ícone dos favoritos (encontra-se selecionado, PoI/Evento pertence aos favoritos); * O ícone dos favoritos devolve um feedback visual. |
| Fluxos de Exceção | * Não tem conexão à internet.   2A   * O sistema devolve uma mensagem: “Não foi possível remover dos favoritos.”. |
| Pós-Condição |  |
| Casos de Teste | * Verificar se os favoritos são removidos da área dos favoritos do ator. Através de um pedido ao webservice e este devolve que o determinado favorito foi removido da base de dados. * Antes de fazer o pedido à API Rest verificar a conexão do dispositivo móvel à internet. |

|  |  |
| --- | --- |
| Nome | Efetuar comentário ao Evento/PoI. |
| Objetivo | Fazer um comentário ao Evento/PoI adicionando uma informação ao PoI/Evento. |
| Atores envolvidos | URegistado |
| Pré-Condição | *Login* Válido. |
| Prioridade |  |
| Fluxo Principal | 1. O ator encontra-se no ecrã do Evento/PoI, e seleciona a opção inserir comentário; 2. O ator escreve o comentário; 3. O ator seleciona a opção inserir comentário; 4. O sistema insere o comentário, e mostra a área dos diferentes comentários inseridos. |
| Fluxos Alternativos | 2A   * O ator cancela o comentário; * O sistema devolve o ecrã com os detalhes do Evento/PoI.   3A   * O ator cancela a escrita do comentário; * O sistema devolve uma mensagem: “Pretende cancelar o comentário?”; * O ator seleciona a opção confirmar; * O sistema devolve o ecrã com os detalhes do Evento/PoI. |
| Fluxos de Exceção | * Não tem conexão à internet. |
| Pós-Condição |  |
| Casos de Teste | 1. Verificar se o comentário foi inserido com sucesso; 2. Verificar se o comentário foi registado na base de dados. 3. Antes de fazer o pedido à API Rest verificar a conexão do dispositivo móvel à internet. |

|  |  |
| --- | --- |
| Nome | Avaliar (*rate*) PoI/Evento |
| Objetivo | Atribuir uma classificação a um PoI/Evento |
| Atores envolvidos | URegistado |
| Pré-Condição | *Login* Válido |
| Prioridade |  |
| Fluxo Principal | 1. O ator encontra-se no ecrã dos detalhes da atração; 2. O ator seleciona a classificação pretendida; 3. O sistema devolve uma mensagem: “Obrigado pela avaliação.” |
| Fluxos Alternativos |  |
| Fluxos de Exceção | Não tem conexão à internet. |
| Pós-Condição |  |
| Casos de Teste | 1. Verificar se a classificação foi introduzida na base de dados. 2. Antes de fazer o pedido à API Rest verificar a conexão do dispositivo móvel à internet. |

|  |  |
| --- | --- |
| Nome | Inserir PoI/Evento |
| Objetivo | Inserir um evento/PoI que permitirá ao utilizador final consultar. |
| Atores envolvidos | Administrador |
| Pré-Condição | *Login* válido |
| Prioridade |  |
| Fluxo Principal | 1. O ator seleciona a opção criar PoI/Evento; 2. O sistema devolve um ecrã com o formulário a preencher; 3. O ator preenche os campos (nome do evento, localização, data...) ou apenas aqueles que são obrigatórios; 4. O ator seleciona a opção guardar; 5. O sistema devolve uma mensagem: “PoI/Evento introduzido com sucesso.”. |
| Fluxos Alternativos | 3A   * O ator não preenche os campos obrigatórios. * O sistema assiná-la os campos em falta.   3B   * O ator não preenche o formulário corretamente, o sistema alerta o ator dando um feedback visual, passando os campos para vermelho.   5A   * O sistema devolve uma mensagem: ”PoI já existente.”. |
| Fluxos de Exceção | * Não tem conexão à internet. |
| Pós-Condição |  |
| Casos de Teste | 1. Ir à aplicação móvel e verificar se o PoI inserido é apresentado. 2. Antes de fazer o pedido à API Rest verificar a conexão do dispositivo móvel à internet. |

|  |  |
| --- | --- |
| Nome | Editar PoI/Evento |
| Objetivo | Efetuar alterações a um determinado PoI/Evento já existente na aplicação. |
| Atores envolvidos | Administrador |
| Pré-Condição | Login válido |
| Prioridade |  |
| Fluxo Principal | 1. O ator encontra-se num determinado evento/PoI; 2. O ator seleciona a opção editar; 3. O sistema devolve um novo ecrã com possibilidade de editar as informações do evento/PoI selecionado; 4. O ator edita o evento; 5. O ator seleciona a opção gravar alterações; 6. O sistema devolve uma mensagem: “Alterações efetuadas com sucesso.”, e retorna o ator ao evento/PoI anteriormente selecionado. |
| Fluxos Alternativos | 5A   * As alterações efetuadas não estão de acordo com as normas, e o sistema evidencia os erros de inserção. E o fluxo continua normalmente.   5B   * O ator seleciona a opção cancelar; * O sistema pergunta se pretende descartar as alterações efetuadas. * O utilizador seleciona a opção não e continua a edição, e o fluxo continua normalmente.   5C   * O ator seleciona a opção cancelar; * O sistema pergunta se pretende descartar as alterações efetuadas. * O utilizador seleciona a opção sim e o fluxo continua normalmente. |
| Fluxos de Exceção | * Não tem conexão à internet.   6A   * O sistema devolve a mensagem: “As alterações não foram efetuadas, tente novamente.”. |
| Pós-Condição |  |
| Casos de Teste | 1. Verificar se as alterações se verificaram na aplicação. 2. Verificar se as alterações foram efetuadas na base de dados, comparando os parâmetros de edição com os na base de dados. 3. Antes de fazer o pedido à API Rest verificar a conexão do dispositivo móvel à internet. |

|  |  |
| --- | --- |
| Nome | Criar notificações para os utilizadores |
| Objetivo | Criar notificações para o sistema enviar aos utilizadores |
| Atores envolvidos | Administrador |
| Pré-Condição | *Login* válido |
| Prioridade |  |
| Fluxo Principal | 1. O ator seleciona a opção criar notificação. 2. O sistema devolve o formulário da notificação. 3. O ator preenche a informação ou apenas os campos obrigatórios. 4. O ator seleciona a opção criar notificação. 5. O sistema devolve uma mensagem: “Notificação criada com sucesso.”, e apresenta a lista das notificações. |
| Fluxos Alternativos | 3A   * O ator não preenche os campos obrigatórios. * O sistema assiná-la os campos em falta.   3B   * O ator não preenche o formulário corretamente. * O sistema assiná-la os campos errados. |
| Fluxos de Exceção | * Não tem conexão à internet.   5A   * O sistema devolve uma mensagem: “Não foi possível criar a notificação. (mais mensagem de possível erro.).”. |
| Pós-Condição | 1. Enviar notificação aos utilizadores. |
| Casos de Teste | 1. Verificar se a notificação foi enviada com sucesso analisando a resposta fornecida pela *Framework* do firebase. 2. Antes de fazer o pedido à API Rest verificar a conexão do dispositivo móvel à internet. |

|  |  |
| --- | --- |
| Nome | Devolver Acesso |
| Objetivo | A API da Rede social dá permissão ao utilizador para aceder as suas informações. |
| Atores envolvidos | URegistado, APIRedeSocial |
| Pré-Condição | Caso de Uso: Registar Utilizador |
| Prioridade |  |
| Fluxo Principal | 1. O sistema pede acesso às informações da rede Social (nome, idade, mail, tokenValidação, foto, etc). 2. A API da rede social devolve a informação pedida pelo sistema. 3. O sistema regista a informação, e criar um utilizador do tipo rede social. 4. O sistema devolve uma mensagem: “Utilizador criado com sucesso”; |
| Fluxos Alternativos |  |
| Fluxos de Exceção | * Não tem conexão à internet.   2A   * A API da rede social nega o pedido do sistema. |
| Pós-Condição |  |
| Casos de Teste | * Antes de fazer o pedido à API Rest verificar a conexão do dispositivo móvel à internet. |

|  |  |
| --- | --- |
| Nome | Enviar Notificação |
| Objetivo | O sistema envia notificações aos utilizadores |
| Atores envolvidos | URegistado |
| Pré-Condição | *Login* Válido |
| Prioridade |  |
| Fluxo Principal | 1. O sistema valida quais os utilizadores que contém as notificações ativadas; 2. Envia notificação aos utilizadores; |
| Fluxos Alternativos |  |
| Fluxos de Exceção | 2A.   * O sistema não tem conexão co os dispositivos móveis; |
| Pós-Condição |  |
| Casos de Teste | * Verificar se os utilizadores recebem as notificações. * Verificar se os utilizadores que têm bloqueado as notificações não as recebem. * Antes de fazer o pedido à API Rest verificar a conexão do dispositivo móvel à internet. |

|  |  |
| --- | --- |
| Nome | Fornece PoI ao sistema |
| Objetivo | A API da Google fornece pontos de interesse ao administrador facilitando o seu registo; |
| Atores envolvidos | Administrador |
| Pré-Condição | *Key* de acesso à API válida/ Caso de Uso Inserir PoI |
| Prioridade |  |
| Fluxo Principal | 1. O ator seleciona fazer pesquisa por nome do PoI; 2. O Sistema faz um pedido à API da Google com o nome introduzido do PoI; 3. A API da Google devolve a informação referente ao PoI procurado; 4. O sistema recebe a informação e cria um PoI com a informação recebida pela Google mais a informação adicionada pelo utilizador; 5. O sistema devolve uma mensagem: “Ponto de interesse criado com Sucesso”. |
| Fluxos Alternativos | 3A.   * A API devolve vários pontos de interesse; * O utilizador tem de escolher o pretendido e o sistema faz um novo pedido, desta vez com o ID do PoI; * A API devolve a informação do PoI em questão.   3B.   * A API não devolve nenhum PoI; * O utilizador faz uma nova pesquisa; |
| Fluxos de Exceção | * Não tem conexão à internet. |
| Pós-Condição |  |
| Casos de Teste | * Verificar se quando o fluxo 3A é acionado corre como esperado; * Antes de fazer o pedido à API Rest verificar a conexão do dispositivo móvel à internet. |

## Diagramas de Classes 2

[...].

## Diagrama de Atividades

[...]

## Diagrama de Estados

[...]

## Diagrama de Componentes

[...]

## Diagrama de Instalação (talvez, ñ aplicável)

[...]

## Modelo ER e Semântica dos dados

[...]

### Modelo Entidade Relação

[...]

### Dicionário de Dados

[...]

## Tecnologias

[...]

# Implementação da Solução

[...]

## Introdução

## CODE (que seja necessário explicar)

[...]

## BD’S

[...]

## Interface

[...]

## Componentes Gráficos

[...]

# Testes

# Conclusões e Trabalho futuro

[...]

## Conclusões

[...]

## Trabalho Futuro

[...]

## 

[...].

# Bibliografia

[1] D. Gavalas, M. Kenteris, C. Konstantopoulos, and G. Pantziou, “Web application for recommending personalised mobile tourist routes,” *IET Softw.*, vol. 6, no. 4, p. 313, 2012.

[2] Google, “Engage Users and Drive Conversions,” pp. 1–30.

# [Anexo A]

# [Anexo B]

Apagar esta table;

|  |  |
| --- | --- |
| Nome |  |
| Objetivo |  |
| Atores envolvidos |  |
| Pré-Condição |  |
| Prioridade |  |
| Fluxo Principal |  |
| Fluxos Alternativos |  |
| Fluxos de Exceção |  |
| Pós-Condição |  |
| Casos de Teste |  |