****

**INSTITUTO POLITÉCNICO DE LEIRIA**

Ciclodias

RELATÓRIO DE projeto em sistemas de informação

Relatório de Projeto em Sistemas de Informação para cumprimento dos requisitos necessários à realização da prova de apresentação de projeto do Curso Técnico Superior Profissional (TeSP) de **Programação de Sistemas de Informação** realizado sob a orientação de **Marco Vicente**

Gabriel Silva Nº 2201133

Iuri Carrasqueiro Nº 2201127

Programação em Sistemas de informação

02/2022

**DECLARAÇÃO**

Declaro que este Relatório se encontra em condições de ser apreciada (o) pelo júri a designar.

O estudante 1,

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Torres Vedras, .... de ............... de ...............

Relatório de Projeto em Sistemas de Informação para cumprimento dos requisitos necessários à realização da prova de apresentação de projeto do Curso Técnico Superior Profissional (TeSP) de **Programação de Sistemas de Informação** realizado sob a orientação de **Marco Vicente**

**DECLARAÇÃO**

Declaro que este Relatório se encontra em condições de ser apreciada (o) pelo júri a designar.

O estudante 2,

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Torres Vedras, .... de ............... de ...............

# Agradecimentos

# Resumo

**[RELATÓRIO DE PROJETO - Ciclodias]**

**[AUTORES]**

Texto do resumo em Português, a 1 espaço, com o máximo de 300 palavras.

O Ciclodias é um projeto

PALAVRAS-CHAVE: xxxx, yyyy, zzzz

# Índices

O relatório deve conter o índice principal, o índice de figuras e o índice de tabelas.

Índice

[Agradecimentos iv](#_Toc95139223)

[Resumo v](#_Toc95139224)

[Índices vi](#_Toc95139225)

[Formatação do Texto Principal 1](#_Toc95139226)

[1. Introdução 2](#_Toc95139227)

[2. Metodologia 3](#_Toc95139228)

[3. Arquitetura do Sistema 4](#_Toc95139229)

[4. Gestão do Projeto 5](#_Toc95139230)

[5. Análise 6](#_Toc95139231)

[6. Desenho 10](#_Toc95139232)

[7. Implementação 15](#_Toc95139233)

[8. Testes 17](#_Toc95139234)

[9. Conclusão e trabalho futuro 18](#_Toc95139235)

[10. Bibliografia 19](#_Toc95139236)

[11. Anexos 20](#_Toc95139237)

[Figura 1 - Mockup de Registo do utilizador 10](file:///C:\Users\gabri\Desktop\Universidade_Ficheiros\Semestre_3\Projeto_Sistemas_Informacao\PSI_ProjetoSI\docs\Modelo_Relatório_Final%20_ProjetoSistemasInformação.docx#_Toc95139238)

[Figura 2 – Mockup de Login do utilizador 10](file:///C:\Users\gabri\Desktop\Universidade_Ficheiros\Semestre_3\Projeto_Sistemas_Informacao\PSI_ProjetoSI\docs\Modelo_Relatório_Final%20_ProjetoSistemasInformação.docx#_Toc95139239)

[Figura 3 – Mockup da página principal da app 11](file:///C:\Users\gabri\Desktop\Universidade_Ficheiros\Semestre_3\Projeto_Sistemas_Informacao\PSI_ProjetoSI\docs\Modelo_Relatório_Final%20_ProjetoSistemasInformação.docx#_Toc95139240)

[Figura 4 – Mockup do perfil de utilizador 11](file:///C:\Users\gabri\Desktop\Universidade_Ficheiros\Semestre_3\Projeto_Sistemas_Informacao\PSI_ProjetoSI\docs\Modelo_Relatório_Final%20_ProjetoSistemasInformação.docx#_Toc95139241)

[Figura 5 – Mockup da página para iniciar o treino 12](file:///C:\Users\gabri\Desktop\Universidade_Ficheiros\Semestre_3\Projeto_Sistemas_Informacao\PSI_ProjetoSI\docs\Modelo_Relatório_Final%20_ProjetoSistemasInformação.docx#_Toc95139242)

[Figura 6 – Mockup da página de treino 12](file:///C:\Users\gabri\Desktop\Universidade_Ficheiros\Semestre_3\Projeto_Sistemas_Informacao\PSI_ProjetoSI\docs\Modelo_Relatório_Final%20_ProjetoSistemasInformação.docx#_Toc95139243)

[Figura 7- Mockup da página do mapa durante o treino 13](file:///C:\Users\gabri\Desktop\Universidade_Ficheiros\Semestre_3\Projeto_Sistemas_Informacao\PSI_ProjetoSI\docs\Modelo_Relatório_Final%20_ProjetoSistemasInformação.docx#_Toc95139244)

[Figura 8- Mockup da página de pausa 13](file:///C:\Users\gabri\Desktop\Universidade_Ficheiros\Semestre_3\Projeto_Sistemas_Informacao\PSI_ProjetoSI\docs\Modelo_Relatório_Final%20_ProjetoSistemasInformação.docx#_Toc95139245)

[Figura 9- Mockup da página de Resultados do treino 14](file:///C:\Users\gabri\Desktop\Universidade_Ficheiros\Semestre_3\Projeto_Sistemas_Informacao\PSI_ProjetoSI\docs\Modelo_Relatório_Final%20_ProjetoSistemasInformação.docx#_Toc95139246)

[Figura 10- Mockup da página de definições 14](file:///C:\Users\gabri\Desktop\Universidade_Ficheiros\Semestre_3\Projeto_Sistemas_Informacao\PSI_ProjetoSI\docs\Modelo_Relatório_Final%20_ProjetoSistemasInformação.docx#_Toc95139247)

# Formatação do Texto Principal

Configuração padrão: 1,5 espaço entre linhas; opções de tipo de letra: Garamond, Times New Roman; letra tamanho 12; texto justificado (6 pto antes / 6 pto depois – espaçamento entre parágrafos)[[1]](#footnote-2);

numeração das páginas: árabe, rodapé à direita; letra tamanho 10.

[Esta é sempre a página 1.]

# Introdução

Apresentação do trabalho. Deve incluir a organização e a estrutura do relatório.

# Metodologia

Para o desenvolvimento do Projeto em Sistemas de Informação, foi adotada a metodologia ágil, pela fácil divisão de tarefas e a adaptabilidade por adicionar novas funcionalidades.

O projeto foi dividido em duas partes, uma que inclui o Website e a API, ou seja, todo o código envolvendo a Yii Framework, e uma que inclui o programa Android, ou seja, todo o código envolvendo a programação Android. Cada um destes contém o seu próprio repositório Github e respetivos branches.

# Arquitetura do Sistema

Para a realização do projeto foram utilizadas as seguintes tecnologias

Website:

* Framework Yii2 para a manipulação e apresentação de dados no website;
* Framework Yii2-adminlte3 para a interface da Backend do Website;
* Framework Codeception para a realização de testes unitários, funcionais e de aceitação;
* Biblioteca Mapbox em Javascript para apresentação da rota dos percursos realizados pelos utilizadores da aplicação;
* Biblioteca ApexCharts para a apresentação do gráfico da velocidade instantânea dos treinos do utilizador;

Android:

* API Mapbox em Java para mostrar a localização do utilizador num mapa e o desenho do percurso feito pelo utilizador no mesmo;
* Biblioteca Volley para a comunicação com a API;
* Biblioteca EclipsePahoMqttJava para apresentação notificações de interações nas publicações do utilizador no Feed de Notícias e de alterações na Leaderboard do Website;

# Gestão do Projeto

Neste capítulo deve ser apresentado o plano do Projeto (Sugestão: utilizar a ferramenta de Gestão de projetos Redmine):

* Gantt Chart;
* Identificação das principais actividades;
* Deliverables;
* Milestones;
* Equipa de Projeto e funções respetivas.

# Análise

O projeto Ciclodias é um projeto que permite o monitoramento de exercício físico do utilizador, guardando informações dos treinos realizados, assim como a sua rota percorrida, permitindo a criação de publicações para todos os utilizadores poderem visualizar e interagir com essas.

Os objetivos principais deste projeto são:

* Permitir o utilizador criar, entrar e gerir a sua conta o mais eficientemente possível;
* Permitir o utilizador visualizar todos os seus treinos detalhadamente;
* Permitir ao utilizador realizar um treino tanto em modo online como modo offline fazendo a sincronização dos treinos assim que houver Internet;
* Permitir a aplicação ser executada em segundo plano;
* Permitir a gestão das sessões de treino do utilizador;
* Permitir a criação de publicações no feed de notícias, a partir de um treino do utilizador;
* Permitir a visualização e interação de qualquer utilizador com sessão iniciada nas publicações presentes no feed de notícias;
* Receber notificações na aplicação Android, providenciadas pelo Website;

Requisitos do projeto Geral:

* Permitir a criação de uma conta no projeto;
* Permitir iniciar e terminar sessão o mais eficientemente possível;
* Permitir ao utilizador editar o primeiro e último nome assim como a data de nascimento do seu perfil;
* Permitir ao utilizador apagar o seu perfil. Quando se realiza esta operação todos os dados da conta são apagados, assim como os treinos realizados pelo utilizador, publicações feitas pelo mesmo, e quaisquer interações feitas pelo mesmo em publicações de outros utilizadores;
* Apresentar ao utilizador todos os treinos que este realizou com os dados mais relevantes;
* Permitir ao utilizador visualizar os detalhes pormenorizados de um treino;
* Permitir o utilizador apagar um treino a partir dos detalhes do mesmo. Neste caso, é verificado se existe alguma publicação criada para esse treino. Caso exista, todas as interações feitas para a mesma são apagadas para depois apagar a própria publicação e o treino, sem quaisquer erros e sem deixar dados desnecessários na API. Caso não exista nenhuma publicação, o treino é simplesmente apagado;
* Permitir ao utilizador criar uma publicação no feed de notícias a partir dos detalhes de um treino;

Requisitos da aplicação Android:

* Carregar da API para as SharedPreferences os dados do utilizador que fez login, incluindo o seu token para pedidos da API, permitindo assim fazer automaticamente login na aplicação sem que se tenha de cada vez que se saia da aplicação inserir os dados novamente;
* Carregar da API todos os treinos realizados pelo utilizador que se iniciou sessão para a Base de dados local, para permitir a visualização em modo offline;
* Terminar a sessão do utilizador na aplicação elimina todos os dados na base de dados local e SharedPreferences, deixando a aplicação sem dados inutilizados;
* Permitir a obtenção da localização GPS do utilizador para realizar sessões de treino na aplicação. É pedido ao utilizador estas permissões na primeira vez que este entra na activity para iniciar um treino, ou para visualizar os detalhes de um. Caso o utilizador não o permita, este é redirecionado para a página principal, isto acontece até que este permita a obtenção da localização. Depois de permitido, estas permissões não são mais pedidas ao utilizador;
* Permitir ao utilizador realizar treinos em modo offline, guardando-os na Base de dados local, fazendo-se a sincronização com a API assim que houver Internet e quando se navegar para a página principal da aplicação;
* Permitir ao utilizador iniciar uma sessão de treino;
* Permitir a aplicação ser executada em segundo plano durante o treino;
* Apresentar ao utilizador no menu principal durante o treino os dados mais relevantes deste, de modo a permitir a fácil visualização dos mesmos;
* Permitir a visualização do mapa durante o treino;
* Permitir colocar o treino em pausa e retomar quando o utilizador pretender;
* Permitir ao utilizador terminar o treino apresentando caixas de confirmação;
* Permitir ao utilizador rever a sua sessão de treino, com a rota desenhada no mapa e permitir este escolher um nome para a sessão. caso a pretenda guardar. Este também pode sair sem guardar a sessão;
* Guardar na API as alterações feitas aos treinos e utilizador;
* Receber notificações providenciadas pelo Website;
* Caixas de confirmação em todas as operações que possam ter algum impacto na aplicação, para que o utilizador não faça operações indesejadas;

Requisitos do Website Frontend:

* Permitir a visualização do menu principal com a Leaderboard do website, a qualquer utilizador mesmo que não tenha login feito;
* Bloquear o acesso a todas as páginas exceto a página principal, login e registo a utilizadores sem login feito;
* Bloquear o acesso a administradores de fazer login.
* Bloquear o acesso a utilizadores a páginas para moderadores;
* Permitir a visualização de publicações no feed de notícias a qualquer utilizador com login feito;
* Permitir a interação dos utilizadores nas publicações de qualquer utilizador (colocar um gosto ou escrever comentários);
* Permitir apagar uma publicação que foi criada pelo mesmo utilizador a partir do feed de notícias;
* Permitir ao utilizador visualizar apenas as suas publicações, para as poder gerir facilmente;
* Permitir ao moderador aceder a uma página para gerir todos os comentários das publicações;

Requisitos do Website Backend:

* Permitir o acesso a esta parte do Website apenas a administradores, outros utilizadores não têm acesso;
* Visualização de um dashboard com informações gerais do projeto (Ex: Número de sessões de treino);
* Permitir a visualização e fazer a gestão de todos os utilizadores e alterar os seus cargos rapidamente;

# Desenho

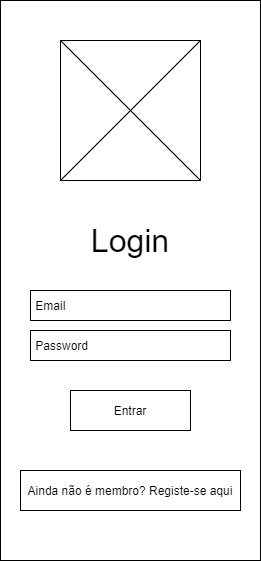
Especificar Protótipos, Casos de Uso, Modelo de Dados, Mockup’s, etc.

Figura 1 - Mockup de Registo do utilizador

Figura 2 – Mockup de Login do utilizador

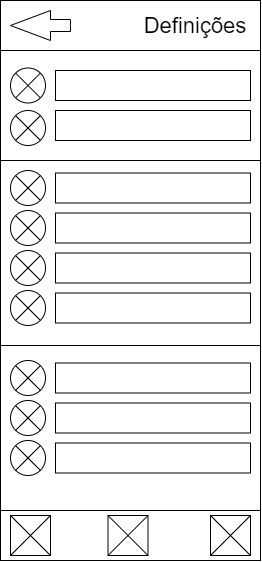
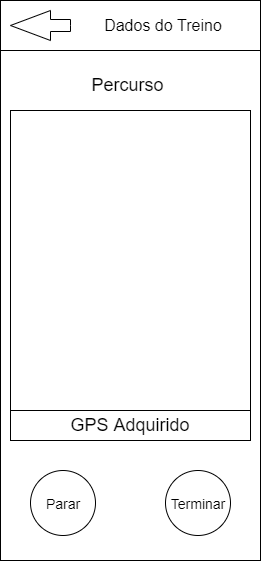


Figura 3 – Mockup da página principal da app

Figura 4 – Mockup do perfil de utilizador

Figura 5 – Mockup da página para iniciar o treino

Figura 6 – Mockup da página de treino

Figura 7- Mockup da página do mapa durante o treino

Figura 8- Mockup da página de pausa

Figura 9- Mockup da página de Resultados do treino

Figura 10- Mockup da página de definições

# Implementação

Em termos de implementação, o desenvolvimento do nosso projeto começou pela pesquisa de informação, o estudo de diversas API’s externas, com as funcionalidades necessárias para o nosso projeto, e a criação dos repositórios Github.

Ao terminar a pesquisa de informação, começou-se a estudar a documentação da API externa escolhida, MapBox API, e iniciou-se a criação dos *mockups* das *activitys* da aplicação Android. Ao terminar esta última tarefa, começou-se a criação das próprias *activitys* no Android Studio.

Para o estudo da API externa, foi criado um projeto de testes, onde foi implementado todo o código que era estudado.

Além destas tarefas, desenvolveu-se a página principal do Website e os relatórios para as primeiras etapas do projeto das unidades curriculares de AMSI e PlatSI.

Após as primeiras etapas, desenvolveu-se o código da nossa API, com todos os métodos necessários a serem utilizados pela aplicação Android, e a implementação do código da API externa na aplicação Android.

Para implementar esta última, desenvolveu-se as *activitys* necessárias para a realização de um treino, a ligação entre estas e a classe da API externa, e o armazenamento dos dados dos treinos na base de dados local, exceto a informação da rota e do gráfico de velocidade.

Para mostrar os dados dos treinos realizados, implementou-se uma RecyclerView na página principal da aplicação Android, onde são apresentados os dados principais dos treinos, e a *activity* de detalhes das sessões de treino, onde são exibidos todos os dados sobre esse.

Após isso, passou-se a desenvolver o código de comunicação com a nossa API na aplicação Android, mais precisamente, os métodos de login, registo e gestão do utilizador. Logo a seguir, os métodos de criação e listagem de sessões de treinos, onde, também, se desenvolveu a funcionalidade de criação de novas sessões de treinos, sem haver acesso à Internet, com um sistema de sincronização com a nossa API, no momento que houver novamente Internet.

A este ponto do desenvolvimento da aplicação Android, iniciou-se uma fase de testes, onde foi-se testando a criação de novos treinos, a criação da rota do mesmo e se os dados estavam a ser armazenados corretamente. Também iniciou-se o desenvolvimento da Backend do Website.

Após esta fase de testes, implementou-se os métodos da nossa API de alteração e eliminação de uma sessão de treino, eliminação do próprio perfil na aplicação Android, e o armazenamento da rota do treino, tanto na base de dados local, como na nossa API.

Ao terminar as tarefas anteriores, passou-se a desenvolver o Website, onde se criou página de histórico de sessões de treino, implementou-se a API externa e a página de detalhes de cada treino. Enquanto isso, foi-se corrigindo alguns erros que apareciam na aplicação Android.

Depois destas páginas, implementou-se as páginas do feed de notícias e todas as funcionalidades do mesmo, como a possibilidade de reagir a uma publicação ou a possibilidade de comentar a mesma. Esta tarefa levou algum tempo a ser desenvolvida, por envolver bastantes funcionalidades diferentes. Além disto, começou-se a desenvolver os testes unitários, funcionais e de aceitação da Codeception na Yii Framework.

A este ponto, começou-se a armazenar os dados do gráfico da velocidade, tanto na base de dados local do Android, como na nossa API, e implementou-se o gráfico da velocidade nos detalhes de cada treino, utilizando esses dados.

Neste momento, começou-se a preparar o Website para a entrega na unidade curricular de PlatSI, comentando o código e realizando o relatório para a UC. Depois dessa entrega, criou-se alguns métodos na nossa API que ainda não tinham sido criados, como por exemplo, os métodos para a criação de novas publicações, e a implementação de funcionalidades baseadas em *messaging*, onde se desenvolveu um sistema de notificações.

Realizou-se o relatório para a unidade curricular de SISis e fez-se a devida entrega.

Por fim, implementou-se o sistema de notificações no Android e corrigiu-se os últimos erros da aplicação, com isso, preparando-a para a sua entrega na unidade curricular de AMSI.

Realizou-se o relatório e fez-se a entrega.

# Testes

Este capítulo deve incluir:

* Desenho de testes
* Ferramentas de testes utilizadas
* Execução dos testes
* Resultados obtidos e correção de erros

# Conclusão e trabalho futuro

# Bibliografia

<https://docs.mapbox.com/android/navigation/guides/>

<https://docs.mapbox.com/android/beta/maps/guides/>

<https://forum.yiiframework.com/t/how-to-create-a-pretty-url/79879/5>

<https://www.yiiframework.com/doc/guide/2.0/en/rest-authentication>

<https://stackoverflow.com/questions/27474059/yii-override-generic-rest-functions/27496376>

<https://stackoverflow.com/questions/27648504/yii2-override-generic-create-action-in-rest-activecontroller/50744982#50744982>

<https://github.com/yiisoft/yii2-app-advanced/blob/master/docs/guide/start-testing.md>

<https://stackoverflow.com/questions/43224842/how-to-add-multiple-icons-on-action-bar-in-android/43225435>

<https://stackoverflow.com/questions/28296708/get-clicked-item-and-its-position-in-recyclerview>

# Anexos

1. Notas de rodapé: fonte 10 pt; 1 espaço entre linhas. [↑](#footnote-ref-2)