A red and white sign

AI-generated content may be incorrect.

**Regulamento Interno**

Consulta+

Guilherme Barreiro – 8220849

Diogo Loureiro – 8220238

André Oliveira – 8200066

Gonçalo Barbosa - 8230153

**Laboratório de Desenvolvimento de Software**

Índice:

[1. Equipa 3](#_Toc209795683)

[2. Riscos 3](#_Toc209795684)

[Riscos Técnicos 3](#_Toc209795685)

[Riscos de Gestão 3](#_Toc209795686)

[Riscos de Qualidade 3](#_Toc209795687)

[3. Avaliação Interna 4](#_Toc209795688)

[4. Periodicidade, métodos e ferramentas para reuniões 5](#_Toc209795689)

[5. Convocatória e Atas 5](#_Toc209795690)

[6. Versionamento 6](#_Toc209795691)

[7. Sprints 6](#_Toc209795692)

[8. Branching e Merge Requests 7](#_Toc209795693)

[9. User Stories e Testes 8](#_Toc209795694)

Índice de imagens:

[Figura 1- Pontuação a seguir pela Avaliação Interna 3](#_Toc209288813)

[Figura 2 - Tabela de Pontuação de cada membro 3](#_Toc209288814)

# 1. Equipa

A equipa é composta por Guilherme Barreiro, Diogo Loureiro, André Oliveira e Gonçalo Barbosa e serão responsáveis pelas funções de Product Owner e Dev Team.

O cargo de líder do projeto será desempenhado por Guilherme Barreiro.

# 2. Riscos

## Riscos Técnicos

* Falhas de integração entre backend, frontend web e app mobile.
* Problemas de autenticação/segurança (ex.: vulnerabilidades no JWT ou gestão de tokens).
* Dependência de serviços externos (notificações, email, SNS) que podem falhar.

## Riscos de Gestão

* Falta de comunicação eficaz entre membros da equipa.
* Ausência de membros em momentos-chave (reuniões, defesas).

## Riscos de Qualidade

* Requisitos mal definidos levando a funcionalidades incompletas ou mal implementadas.
* Testes insuficientes, resultando em bugs em produção.

# 3. Avaliação Interna

Falta de entrega, atrasos, a não comparência a reuniões, etc, serão penalizadas e geridas internamente pelo grupo, assim como o esforço e atitudes positivas para a equipa irão ser valorizadas.

A table with numbers and text

AI-generated content may be incorrect.A Dev Team deve analisar cada caso individual e decidir como equipa qual a penalização mais adequada para o membro inferido com base num sistema de pontos onde cada membro começa com 20 pontos, a retirada de pontos é considerada uma penalização e uma adição de pontos uma benesse ao membro a ser avaliado.

Figura 1- Pontuação a seguir pela Avaliação Interna

A screenshot of a computer

AI-generated content may be incorrect.

Figura 2 - Tabela de Pontuação de cada membro

# 4. Periodicidade, métodos e ferramentas para reuniões

As reuniões à distância serão realizadas através da plataforma Discord, enquanto os encontros presenciais decorrerão nas instalações da Escola Superior de Tecnologia e Gestão.

No início de cada sessão será redigida uma ata, onde ficarão registados os horários, os participantes, os assuntos discutidos e as decisões alcançadas.

Os recursos utilizados em cada reunião incluem:

* Aplicações do Microsoft Office (Word, Excel, …);
* Editor de texto simples;
* Ferramenta gráfica básica (Microsoft Paint).

O canal oficial de comunicação escrita do grupo será o Discord.

Os endereços de correio eletrónico institucionais dos membros são:

* Guilherme Barreiro – 8220849@estg.ipp.pt
* Diogo Loureiro – [8220238@estg.ipp.pt](mailto:8220238@estg.ipp.pt)
* André Oliveira – [8200066@estg.ipp.pt](mailto:8200066@estg.ipp.pt)
* Gonçalo Barbosa – [8230153@estg.ipp.pt](mailto:8230153@estg.ipp.pt)

# 5. Convocatória e Atas

O prazo mínimo para envio de convocatória antes de uma reunião deve ser de 3 dias de antecedência pelo email de cada membro nomeadamente:

* Guilherme Barreiro – 8220849@estg.ipp.pt
* Diogo Loureiro – 8220238@estg.ipp.pt
* André Oliveira – [8200066@estg.ipp.pt](mailto:8200066@estg.ipp.pt)
* Gonçalo Barbosa – 8230153@estg.ipp.pt

A data-limite para homologação de atas deve ser 2 dias após a realização da ata e o template desta encontra-se no Moodle disponibilizado pelo professor Célio.

As entregas parciais devem ser feitas até 2 dias antes da milestone.

# 6. Versionamento

As versões irão ser numeradas através de Versionamento Semântico, no formato X.Y.Z, onde X, Y, e Z são inteiros não negativos, não contendo zeros à esquerda. Cada elemento aumenta numericamente.

Por exemplo: 1.9.0 -> 1.10.0 -> 1.11.0.

Grande mudança ao projeto aumenta numericamente o X de forma que a versão 1.0.0 passa para 2.0.0, mudanças menores como adição de funcionalidades, por exemplo, aumenta numericamente o Y de forma que a versão 1.0.0 passa para 1.1.0, pequenas correções ao projeto (Patches) aumentam numericamente o Z de forma que a versão 1.0.0 passa para 1.0.1.

# 7. Sprints

Os sprints são semanais a finalizar na sexta-feira de cada semana, tendo segunda-feira a tarefa de monitorizar o avanço do projeto, quais foram os incrementos ao projeto e quais serão os incrementos futuros.

# 8. Branching e Merge Requests

Com base no repositório GitLab, a estratégia de branching segue as seguintes regras:

* dev: branch principal de desenvolvimento do projeto.
* master: branch que contém a versão estável do projeto / releases.
* feature/[nome-da-funcionalidade]: cada elemento cria uma branch específica para a funcionalidade/tarefa atribuída.

Todas as alterações devem:

1. Ser submetidas através de Merge Request (MR) para a branch dev.
2. Incluir descrição clara da alteração, associando a respetiva User Story.
3. Ser alvo de code review obrigatório por, pelo menos, um outro membro da equipa antes de serem aprovadas.
4. Conflitos de merge devem ser resolvidos pelo autor da branch em colaboração com o revisor, garantindo que nenhuma funcionalidade existente é afetada.
5. A integração em master apenas acontece após aprovação de todos os testes e validação em sprint.

# 9. User Stories e Testes

A estratégia de distribuição das User Stories e respetiva testagem será efetuada de forma rotativa, garantindo que:

* Todos os membros do grupo participam tanto no desenvolvimento como na validação.
* Nenhum elemento pode testar o seu próprio código.
* As User Stories são atribuídas com base em Story Points definidos pela equipa, de acordo com o nível de complexidade e esforço.
* A distribuição é feita de forma equilibrada para garantir justiça na carga de trabalho.

Regras específicas:

* Guilherme Barreiro testa o código de Diogo Loureiro.
* Diogo Loureiro testa o código de André Oliveira.
* André Oliveira testa o código de Gonçalo Barbosa.
* Gonçalo Barbosa testa o código de Guilherme Barreiro.

Uma User Story só pode ser marcada como “Done” quando estiver:

1. Implementada conforme a especificação.
2. Testada por outro membro.
3. Documentada e aprovada em sprint review.

Aprovado por: