Logo

Description automatically generated

Software Project Management 2022/2023

Gestão de Medicamentos

D1.2.2 - Plano de Garantia de Qualidade

### Autores

* Rúben Mendes
* Diogo Silva
* Nuno Domingues
* Daniel Albino
* Miguel Neves

### Revisto por:

* Rúben Mendes

### Aprovado por:

* Nuno Domingues

### Índice

[1. Introdução](#_Toc117490361)

[2. Objetivos de qualidade](#_Toc117490362)

[3. Avaliações](#_Toc117490363)

[3.1. Inspeções](#_Toc117490364)

[3.2. Deskchecks](#_Toc117490365)

[3.3. Walkthroughs](#_Toc117490366)

[4. Testes](#_Toc117490367)

[4.1. Unidade de Testagem](#_Toc117490368)

[4.2. Teste de integração](#_Toc117490369)

[4.3. Teste de Aceitação](#_Toc117490370)

[5. Risk Management](#_Toc117490371)

[5.1. Risk identification and assessment](#_Toc117490372)

[5.2. Risk monitoring and control](#_Toc117490373)

[6. Padrões de Código](#_Toc117490374)

[7. Gestão da Mudança](#_Toc117490375)

# Introdução

A nossa aplicação de gestão de medicamentos vai ser uma aplicação *user friendly*, simples, robusta, rápida, intuitiva, fácil de usar e ser funcional.

Para isto, entendemos que o nosso projeto, necessita de corresponder a alguns objetivos de qualidade, tais como a usabilidade do sistema, aspeto que será estudado tendo em conta as heurísticas de usabilidade. Neste aspeto, pretendemos dar um foco importante à parte gráfica para garantir que os utilizadores têm a experiência que desejamos.

Para isto iremos fazer o uso de mockups, assim temos uma implementação digital de acordo com os nossos objetivos.

Ainda importante de referir, que a fiabilidade do sistema é um dos aspetos mais importantes, isto porque não é expectável que o sistema simplesmente pare de funcionar, o que pode levar a más consequências, nomeadamente a falta da toma de medicamentos, sendo que os testes das funcionalidades da aplicação devem ser testados de uma forma exaustiva. O código deve ser o mais compacto e simples possível, para prevenir potenciais “*bugs*”.

Link para o ficheiro D2.3.2 - Quality Assessment Report, [aqui](https://isecpt.sharepoint.com/sites/GPS2223-Team14/Project%20Files/Deliverables/D2.3.2%20-%20Quality%20Assessment%20Report.docx?web=1).

# Objetivos de qualidade

As qualidades externas mais importantes para este projeto são as seguintes:

* Usabilidade

As qualidades internas mais importantes para este projeto são as seguintes:

* Compreensibilidade

Para cumprirmos com as qualidades externas, iremos usar heurísticas de usabilidade.

Relativamente às qualidades internas, certas convenções permitem-nos compreensibilidade do código, como também, iremos pedir a uma pessoa externa, que ainda não está definida para analisar o nosso código e avaliar se é compreensível ou não.

As heurísticas que iremos utilizar como as convenções de código, estão presentes no ficheiro D2.3.2 – Quality Assessment Report, Secção 2, [aqui](https://isecpt.sharepoint.com/:w:/r/sites/GPS2223-Team14/Project%20Files/Deliverables/D2.3.2%20-%20Quality%20Assessment%20Report.docx?d=w6febc6faaa794586aad681a24015a81d&csf=1&web=1&e=iDfeUm&nav=eyJoIjoiMTMzMzc1Njc4NyJ9).

# Revisões

## Inspeções

No âmbito do nosso projeto iremos utilizar as inspeções no documento de especificações de requisitos de software.

As inspeções, permitem que os inspetores identifiquem defeitos, que possam encontrar, de modo a transmitir esses defeitos ao autor do documento.

A inspeção vai seguir o processo normal, processo esse que está descrito em [GPS2223-7\_Reviews.pdf](https://isecpt.sharepoint.com/:b:/r/sites/GPS2223/Shared%20Documents/General/Slides/GPS2223-7_Reviews.pdf?csf=1&web=1&e=k1BKEy) (1. Inspections). O autor do documento estará na equipa do projeto e será o elemento da equipa que mais envolvido está na escrita do documento.

No entanto as funções, da restante equipa de inspeção será desempenhada por membros externos.

Link para a pasta de relatório de inspeções, [aqui](https://isecpt.sharepoint.com/:f:/r/sites/GPS2223-Team14/Project%20Files/Relat%C3%B3rios%20de%20inspe%C3%A7%C3%A3o?csf=1&web=1&e=elgzL6).

## Deskchecks

Iremos maioritariamente utilizar os Deskchecks para fazer as revisões dos documentos.

Essencialmente um Deskcheck é uma simples revisão informal, onde o autor do produto do trabalho distribui pelos seus revisores, este processo está mais detalhado em [GPS2223-7\_Reviews.pdf](https://isecpt.sharepoint.com/:b:/r/sites/GPS2223/Shared%20Documents/General/Slides/GPS2223-7_Reviews.pdf?csf=1&web=1&e=k1BKEy) (2. Deskchecks).

O autor do documento será sempre aquele, que mais envolvido está na escrita do documento. Os revisores serão os restantes elementos da equipa e o tempo que despendem a ler e analisar o documento deve ser registado.

Não são necessários registos escritos, anexados ao documento.

## Walkthroughs

Os Walkthroughs, são:

* Uma formal informal de apresentar documentos técnicos por meio de uma reunião, a revisores, de preferência alguns não devem possuir conhecimentos técnicos.
* O autor tem a responsabilidade de preparar, marcar a reunião, como também convidar os revisores e deve guiar a reunião, solicitando opiniões aos revisores, de forma a melhorar o documento.

Walkthroughs não serão usados neste projeto, pelo facto de ser requerido preferencialmente revisores sem conhecimentos técnicos.

# Testes

## Teste unitário

Os testes de unidade vão servir para testar pequenos pedaços de código de cada vez por exemplo funções de pesquisa ou consulta de medicamentos sendo realizadas por quem as desenvolve facilitando o processo.

Estes testes vão ser maioritariamente utilizados para as classes medicamentos, utentes e encomendas pois vão exigir muitas funções separadas podendo ser realizadas por qualquer membro da equipa.

Quando estivermos no processo de testes o nosso programa vai estar carregado com dados fictícios, facilitando no processo de testes.

## Teste de integração

Os testes de Integração vão servir para encontrar falhas na integração de unidades/módulos, fazendo com que exista um teste a nível de interface e dependências entre si

Este tipo de teste vai ser realizado no nosso modelo de dados servindo para integrar os módulos dos utentes, encomendas e medicamentos gerando assim um modelo de dados funcional para a nossa aplicação.

Quando estivermos a testar a integração de módulos vamos ter de abordar se cada segmento de dados estão a funcionar em conjunto. Sendo este realizado por todos os membros em conjunto que desenvolveram os módulos.

## Teste de Aceitação

Os testes de aceitação é um teste realizado após o desenvolvimento do produto para determinar se os requisitos iniciais foram atendidos de acordo com as suas necessidades decidindo assim se o produto desenvolvido é aceite ou rejeitado pelo cliente. Em caso de rejeição, uma negociação com o cliente poderá ser uma solução.

Link para os documentos:

* [Plano de Testes de Aceitação](https://isecpt.sharepoint.com/:w:/s/GPS2223-Team14/Ecps_cWRi9pOh9iqTbiRqxwBC2zcPqlILa5vtWly0r2EjQ?e=aYchIM)
* [Relatório de teste](https://isecpt.sharepoint.com/:w:/r/sites/GPS2223-Team14/Project%20Files/Deliverables/D2.3.2%20-%20Quality%20Assessment%20Report.docx?d=w6febc6faaa794586aad681a24015a81d&csf=1&web=1&e=dMGQWu&nav=eyJoIjoiMTM3MzIxOTkwOCJ9)

# Gestão de risco

Link para o documento [D2.1.2 – Risk Plan](https://isecpt.sharepoint.com/:w:/r/sites/GPS2223-Team14/Project%20Files/Deliverables/D2.1.2%20-%20Risk%20Plan.docx?d=webb93aae0c024f64b46985936fb3dc82&csf=1&web=1&e=Bg2t7G) e D2.3.2 - Quality Assessment Report, Secção 5, [aqui](https://isecpt.sharepoint.com/:w:/r/sites/GPS2223-Team14/Project%20Files/Deliverables/D2.3.2%20-%20Quality%20Assessment%20Report.docx?d=w6febc6faaa794586aad681a24015a81d&csf=1&web=1&e=9d86Mn&nav=eyJoIjoiMzQ4NTYyNzQyIn0%3D)

## Identificação e avaliação de riscos

Descreve como os riscos são identificados e avaliados.

Cada risco é conectado a um impacto de:

tolerável (1), sério (3), or catastrofrico(5)

e a probabilidade de ocorrência com, um impacto de:

muito baixo(1), baixo(2), moderado(3), alto(4) ou muito alto(5).

## Monitoramento e controlo de riscos

Os riscos são revistos cada semana

Riscos com P x I ≥ 20 devem ser mitigados

# Padrões de Código

Link para o ficheiro D2.3.2 - Quality Assessment Report, Secção 6, [aqui](https://isecpt.sharepoint.com/:w:/r/sites/GPS2223-Team14/Project%20Files/Deliverables/D2.3.2%20-%20Quality%20Assessment%20Report.docx?d=w6febc6faaa794586aad681a24015a81d&csf=1&web=1&e=jCM0qU&nav=eyJoIjoiNDYzMzczMDU3In0%3D).

Decidimos utilizar um conjunto de padrões standard na linguagem Java tais como:

1. Código dividido em módulos para serem fáceis de testar;
2. Nomes de variáveis, classes e funções explícitos e claros;
3. Utilização dos métodos getter e setter;
4. Campos das classes com nomes explícitos e claros;
5. Utilização de Interfaçes e controladores;
6. Utilização do design *pattern* Decorator;
7. Utilização de pastas e subpastas para organização;

# Gestão da Mudança

A gestão de mudança, permite implementar mudanças em produtos de trabalho que valem a pena, ou seja que tragam benefícios para o projeto. Serve para evitar que mudanças desnecessárias ou excessivamente caras atrapalhem o projeto.

Todos os pedidos, passam pelo “Change Control Board”, que é constituído por 3 elementos, sendo o gestor de projeto, gestor técnico e o gestor de teste, para que não haja empate na decisão.

Se houver um pedido de mudança, nesse pedido deve constar os potenciais benefícios dessa mudança. O “Change Control Board”, deve estimar o impacto que terá no projeto e decidir se é viável, aprovando ou rejeitando o pedido de mudança.

Link para o ficheiro D2.3.2 - Quality Assessment Report, Secção 7, [aqui](https://isecpt.sharepoint.com/:w:/r/sites/GPS2223-Team14/Project%20Files/Deliverables/D2.3.2%20-%20Quality%20Assessment%20Report.docx?d=w6febc6faaa794586aad681a24015a81d&csf=1&web=1&e=I60ofE&nav=eyJoIjoiMTM5MTI3MjA3NSJ9).