**MANUAL DE QUALIDADE**

**ENGENHARIA DE SOFTWARE 2018/2019**

**PL6**



**Data:**

13-11-2018

**Versão:**

3.0

**Autores:**

Departamento de Qualidade:

* Hugo Marques
* Ana Almeida
* Luís Gonçalves
* Rui Reis
* Viviana Abreu

**Docentes:**

* Mário Alberto da Costa Zenha Rela
* Álvaro Manuel Reis da Rocha

# Tabela de versões e revisões

|  |  |  |  |
| --- | --- | --- | --- |
| Versão | Data | Autor | Descrição |
| 1.0 | 2018.09.27 | Departamento Qualidade:   * HM * AA * LG * RR * VA * MA | Estrutura Inicial do Documento |
| 2.0 | 2018.10.05 | Departamento Qualidade:   * HM * AA * LG * RR * VA * MA | Atualização do Documento |
| 3.0 | 2018.11.17 | Departamento Qualidade:   * HM * AA * LG * RR * VA * MA | Atualização do Documento |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Índice**

[Tabela de versões e revisões 2](#_Toc530077688)

[1. Introdução 5](#_Toc530077689)

[1.1 Contexto 5](#_Toc530077690)

[1.2 Apresentação do projeto 5](#_Toc530077691)

[1.3 Glossário 6](#_Toc530077692)

[1.4 Referências 7](#_Toc530077693)

[2. Regras gerais da equipa ORCWALL 8](#_Toc530077694)

[3. Processos de Engenharia 9](#_Toc530077695)

[3.1 Requisitos (REQ) 9](#_Toc530077696)

[3.1.1 MQ\_REQ\_Tabela\_Requisitos: 11](#_Toc530077697)

[3.1.2 MQ\_REQ\_*Mockups*: 13](#_Toc530077698)

[3.1.1 MQ\_REQ\_SRS: 16](#_Toc530077699)

[3.2. Implementação (IMP) 19](#_Toc530077700)

[3.2.1 MQ\_IMP\_ CODIGO 20](#_Toc530077701)

[3.2.2 MQ\_IMP\_CORRECAO 22](#_Toc530077702)

[3.2.3 MQ\_IMP\_REUNIOES 24](#_Toc530077703)

[3.3 Testes (TST) 26](#_Toc530077704)

[3.3.1 MQ\_TST\_TESTE\_FUNCIONALIDADES 27](#_Toc530077705)

[3.3.2 MQ\_TST\_TEST\_VECTOR 29](#_Toc530077706)

[3.3.3 MQ\_TST\_DEPLOYMENT 30](#_Toc530077707)

[3.4 Qualidade (QUA) 31](#_Toc530077708)

[3.4.1 MQ\_QUA\_APRESENTAÇÃO\_SEMANAL 32](#_Toc530077709)

[3.4.2 MQ\_QUA\_ATAS 33](#_Toc530077710)

[3.4.3 MQ\_QUA\_ATUALIZAÇÃO\_MQ 34](#_Toc530077711)

[3.4.4 MQ\_QUA\_MÉTRICAS\_E\_PROCESSOS 35](#_Toc530077712)

[3.4.5 MQ\_QUA\_VALIDAÇÂO-DOCS 36](#_Toc530077713)

[4. Disciplinas de suporte 37](#_Toc530077714)

[4.1 Gestão de Projeto (PM) 37](#_Toc530077715)

[4.1.1 MQ\_PM\_ PLANEAR 38](#_Toc530077716)

[4.1.2 MQ\_PM\_ORGANIZACAO 40](#_Toc530077717)

[4.1.3 MQ\_PM\_REUNIOES 40](#_Toc530077718)

[4.1.4 MQ\_PM\_SUPERVISAO 43](#_Toc530077719)

[4.1.5 MQ\_PM\_TAREFAS 45](#_Toc530077720)

[4.1.6 MQ\_PM\_RISCOS 47](#_Toc530077721)

[4.2 Ambiente (ENV) 48](#_Toc530077722)

[4.2.1 MQ\_ENV\_REGULACAO 49](#_Toc530077723)

[4.2.2 MQ\_ENV\_WORKSHOPS 50](#_Toc530077724)

[4.2.3 MQ\_ENV\_SOFTWARE 51](#_Toc530077725)

[4.2.4 MQ\_ENV\_ADS 52](#_Toc530077726)

# Introdução

## Contexto

No âmbito do projeto “Infor-Investigador” desenvolvido na disciplina de Engenharia de Software, lecionada pelos docentes Mário Zenha Rela e Álvaro Manuel Reis da Rocha, foi criado o presente manual de qualidade onde se descrevem todos os processos utilizados na realização do projeto, bem como todo o funcionamento da equipa responsável pelo mesmo, “ORCWALL”. Esta é formada por alunos da Licenciatura em Engenharia Informática, Design e Multimédia e Mestrado integrado em Engenharia Eletrotécnica e Computadores, pertencentes à Faculdade de Ciências e Tecnologia da Universidade de Coimbra.

## Apresentação do projeto

O projeto Infor-Investigador trata-se de um sistema de informação, proposto pela turma 2017/2018 do Programa Doutoral em Ciências da Informação da Faculdade de Letras da Universidade de Coimbra, que irá ser implementado no âmbito da disciplina de Engenharia de Software no decorrer do ano letivo de 2018/2019.

O Infor-Investigador apresenta-se como uma plataforma direcionada a investigadores. Com a utilização desta pretende-se que o investigador, numa só aplicação, possa ter acesso a informação relativa a áreas do seu interesse, contacto com outros investigadores, bem como maior facilidade na divulgação dos seus trabalhos e artigos.

Por outro lado, o Infor-Investigador pretende também ajudar os investigadores mais novos a integrar-se no seio de profissionais ligados à área da investigação. Para isso, os utilizadores menos experientes podem associar-se a padrinhos, investigadores mais entendidos que os ajudarão a inserirem-se no meio.

O contacto entre profissionais da área mostra-se, também, como uma funcionalidade determinante da aplicação. Uma vez que nesta existem fóruns sobre os mais diversos temas.

## Glossário

**E**

ENV – Unidade de Ambiente

**I**

IMP - Unidade de Implementação

**L**

LDM – Licenciatura em Design e Multimédia

LEI – Licenciatura em Engenharia Informática

**M**

MIEEC – Mestrado Integrado em Engenharia Eletrotécnica e de Computadores

**P**

PM – Unidade de Gestão de Projetos

**Q**

QUA – Unidade de Qualidade

**R**

REQ – Unidade de Requisitos

**T**

TST – Unidade de Testes

## Referências

O presente Manual de Qualidade tem como referência o documento *“ES2017\_MANUAL\_DE\_QUALIDADE\_SoftCreation\_v3.0a.pdf”,*disponibilizado pela equipa de docentes na plataforma Inforestudante.

# Regras gerais da equipa ORCWALL

Esta secção do manual tem como objetivo documentar as regras da equipa assim como as competências de cada um dos membros dentro destas mesmas.

1.Todas as decisões transversais ao projeto têm de ser tomadas, de forma democrática, pela equipa de gestores (em alguns casos poderá ser por todos os membros da equipa). O voto de qualidade fica a cargo do gestor do projeto.

2. As decisões que apenas tenham influência dentro de uma determinada equipa são tomadas de forma democrática dentro da mesma, ficando o voto de qualidade a cargo do coordenador da unidade em questão.

3. Todas as reuniões têm de resultar numa ata, que deverá ser enviada à equipa de qualidade para análise e arquivo.

4. Numa reunião geral, a ata será feita pela equipa de Qualidade.

5. Sempre que alguma decisão dentro de uma equipa cause a divisão ou conflito dentro desta mesma, a situação deverá ser levada para a equipa de Gestão.

6. Será facultado, todas as semanas, um formulário para preenchimento com o esforço e tarefas individuais, para registar o registo de dados e acompanhamento das unidades.

7. No final de cada Sprint será facultado um formulário de reflexão sobre os processos aplicados anteriormente e contentamento dentro das unidades de forma a aplicar melhorias.

8. O manual de qualidade poderá ser alterado sempre que forem tomadas decisões aprovadas pela equipa de Gestão em conformidade com os elementos do projeto. Poderão ser adicionados/retirados/alterados processos.

9. A não conformidade com estas regras poderá levar a perda do cargo.

# Processos de Engenharia

## Requisitos (REQ)

|  |  |  |
| --- | --- | --- |
| **Disciplina:** REQ | **Data:** 16/11/2018 | **Coordenador:** Paulo Sousa |
| **Status:** baselined | **Versão:** 3.0 | **Vice:** Emanuel Pereira |

**Lista de membros desta unidade:**

|  |  |  |  |
| --- | --- | --- | --- |
| Paulo Sousa | MIEEC | Coordenador | uc2015248827@student.uc.pt |
| Emanuel Pereira | MIEEC | Vice | uc2012153856@student.uc.pt |
| Marco Silva | MIEEC | Colaborador | uc2015243074@student.uc.pt |
| Nuno Dinis | LDM | Colaborador | uc2016221901@student.uc.pt |

**Trabalho desenvolvido pela equipa de requisitos:**

A equipa de requisitos é constituída por 4 elementos. A estes elementos foram distribuídas tarefas de modo cumprir o objetivo da equipa, que é a realização do documento de System Requirements Specifications (SRS). As tarefas são: a realização dos mockups, tabelas de requisitos e ir editando o documento. As realizações destas tarefas definem o processo da equipa de requisitos, que é o documento de System Requirements Specifications (SRS).

O documento de System Requirements Specifications encontra-se no seguinte link: <https://github.com/carladrim/carladrim.github.io/tree/master/Requisitos>.

O trabalho da equipa de requisitos é importante porque o documento de System Requirements Specifications influencia o trabalho da equipa de implementação, que é desenvolver o produto que o cliente deseja, e também o das restantes equipas.

**Objetivos da unidade de REQ:**

1. Realização de Mockups para especificação dos requisitos.

2. Fazer a ligação entre o cliente e as restantes equipas da unidade.

3. Aprimorar o design e a usabilidade do produto a ser criado.

4. Criar um documento “Software Requirement Specifications

**Comunicação dentro da equipa:**

Segue a lista dos métodos de comunicação usados pela equipa de requisitos:

**Slack**: www.slack.com - Plataforma de mensagens usada para comunicação dentro da equipa. Apenas os membros da equipa de requisitos e um membro da equipa de ambiente tem acesso ao channel dedicado à equipa. A comunicação com a equipa de implementação é efetuada através de uma sala onde se encontram todos os membros de requisitos e todos os membros de implementação.

**Figma**: www.figma.com – Plataforma de design usada para a criação de mockups. Sendo uma plataforma online permite que várias pessoas editem e visualizem o mesmo projeto.

**Trello**: www.trello.com – Aplicação de gestão de projetos usada para saber o estado dos objetivos de cada equipa por sprint. Esta aplicação permite organizar as tarefas através de “notas” distinguidas por cores, uma para cada equipa.

**Diagrama de Gantt**: www.ganttproject.biz/ - Programa livre para realização de cronogramas que permite a gestão das tarefas realizadas e planeadas, com a informação do dia da realização e a dependência entre tarefas.

**Ficheiro semanal de tarefas da equipa**: Ficheiro disponibilizado pela equipa de gestão de projeto para ficar a par do tempo estimado e dedicado das tarefas de cada elemento das equipas.

**Presencialmente**: Usada como primeira opção de contacto uma vez que torna a passagem de informação mais clara. Esta forma de contacto foi feita através de reuniões de equipa e de reuniões com o cliente. Tanto as reuniões de equipa, como as reuniões com o cliente, foram registadas em ata.

**Processos utilizados por esta unidade:**

|  |  |
| --- | --- |
| 1. Realização de documento de SRS | MQ\_REQ\_SRS |

### 3.1.1 MQ\_REQ\_SRS

|  |  |
| --- | --- |
| **Disciplina:** Requisitos | **Coordenador:** Paulo Sousa |
| **Status:** | **Vice:** Emanuel Pereira |

O principal objetivo desta unidade é a criação deste documento, “Software Requirement Specifications”, que contém tabelas e mockups com todos os requisitos pretendidos pelo cliente. Este documento serve também de input a outras unidades pois mostra a informação do cliente necessária para o projeto.

**Pessoas e as suas funções:**

* Paulo Sousa (MIEEC) e Marco Silva (MIEEC) - Responsável pela elaboração das tabelas de requisitos.
* Emanuel Pereira (MIEEC) – Responsável pela formatação das tabelas no documento oficial de requisitos.
* Nuno Dinis (LDM) – Responsável pela validação das tabelas.
* Marco Silva (MIEEC) – Responsável pela criação das tabelas de requisitos.

**Inputs:**

* Reuniões como cliente, onde são definidos e validados os requisitos.
* Reuniões de equipa onde são tomadas decisões de trabalho e onde é feita a distribuição de tarefas.

**Critério de entrada/ativação:**

Através da proposta de trabalho feita pelo cliente, docente de Engenharia de Software, são formuladas as tabelas de requisitos e os respetivos mockups, através disto, é construído o documento de SRS. Através dos inputs, as tabelas e os mockups vão sendo alterados para estarem sempre de acordo com o avanço do projeto. Esta informação é posteriormente enviada à equipa de implementação que põe em prática todos estes requisitos. Para além disto, o documento de SRS serve como base para a equipa de testes averiguar se a implementação está de acordo com todos os requisitos.

**Descrição das tarefas:**

* Preparação: Levando em consideração os inputs apresentados, todo o trabalho é coordenado considerando as reuniões com o cliente e as decisões que daí surgem para fazer a divisão de tarefas.
* Implementação: A equipa de requisitos faz um levantamento de requisitos para levar a análise do cliente. Após isto e após a aprovação dos requisitos, são criadas as tabelas com os mesmos e os respetivos mockups.
* Validação: Assim que as tabelas de requisitos e os mockups estão terminados, ambos são colocados no channel de requisitos no slack para que toda a equipa possa opinar. Depois destes dois artefactos serem aprovados pela equipa são enviados para o channel do slack que reúne a equipa de implementação e de requisitos para que esta equipa possa ficar consciente dos requisitos pretendidos. Paralelamente a isto estes artefactos também são enviados para o Team Leader da equipa de testes via slack.
* Conclusão: Depois de as tabelas de requisitos com os respetivos mockups estarem terminados é criado o SRS para reunir estes dois artefactos num único. Após isto, este documento é aprovado pela equipa de qualidade e gestores, sendo por fim colocado na pasta de requisitos do GitHub.

**Validação:**

* Todas tabelas de requisitos criadas são validadas pelo cliente.
* Os Mockups são validados pelo Team Leader de implementação e pelo cliente.
* Os Mockups para serem validados têm de ser legíveis, têm de estar adequados à plataforma que os vai suportar, e de acordo com os requisitos do cliente.
* O Documento de SRS é validado por toda a equipa e depois é validado pelo professor da prática da cadeira.

**Critério de saída:**

Após todas as tarefas acima descritas terem sido concluídas e aprovadas por todos os membros da equipa de requisitos e pelo cliente, o documento de SRS serve como input para a equipa de implementação para que estes possam progredir no seu trabalho de acordo com o especificado no SRS que se encontra de acordo com o pretendido pelo cliente. Este documento serve também como input à unidade de testes.

**Métricas de avaliação deste processo:**

* Número total de horas de trabalho;
* Tempo de cadatarefa;
* Número de pessoas envolvidas por tarefa;
* Número de requisitos adicionados, alterados e eliminados.
* Número de tabelas de requisitos geradas;

**OUTPUTS gerados:**

* Tabelas de requisitos
* Mockups dos requisitos
* Documento SRS (ES2018\_ORCWall\_DocumentoDeRequisitosVx.x.pdf).
* Todos os outputs podem ser encontrados no seguinte link: https://github.com/carladrim/carladrim.github.io/tree/master/Requisitos

**Monitorização deste processo pela equipa de qualidade:**

Após o documento ter sido aprovado pela unidade de requisitos, é destacado um elemento da unidade de qualidade para fazer um controlo deste documento para verificar se este está de acordo com o pretendido.

## 3.2. Implementação (IMP)

|  |  |  |
| --- | --- | --- |
| Disciplina: IMP | Data: 15-11-2018 | Coordenador: Caio Brêda |
| Status: Baselined | Versão: v2.1 |  |

**Lista de membros desta Unidade:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Nome | Curso | Função | Contacto |
| CB | Caio Brêda | LDM | Coordenador/FE | [caiosimonbreda@gmail.com](mailto:caiosimonbreda@gmail.com) |
| ED | Eduarda Duarte | LDM | Colaborador FE | [eduardamduarte@gmail.com](mailto:eduardamduarte@gmail.com) |
| JS | João Simões | MIEEC | Colaborador BE | joaopedrosimoes@gmail.com |
| SO | Samuel Outeiro | LEI | Vice-Coordenador/BE | [samuelbouteiro@gmail.com](mailto:samuelbouteiro@gmail.com) |
| ZT | [Ž](https://www.facebook.com/ZTrontelj?fref=ufi)iga Trontelj | ERASMUS | Colaborador BE | svizec96@gmail.com |

FE: front-end| BE: back-end|

**Objetivos da Unidade de IMP:**

* Implementar as features indicadas no documento de requesitos, produzindo Código.
* Corrigir os erros detetados pela equipa de Testes.

**Processos Utilizados por esta Unidade:**

|  |  |
| --- | --- |
| Processo | ID |
| Processo de Produção de Código e Design | MQ\_IMP\_CODIGO |
| Processo de Reuniões | MQ\_IMP\_REUNIAO |
| Processo de Organização | MQ\_IMP\_ORGANIZACAO |

**Comunicação dentro da unidade:**

* Slack: www.slack.com – Plataforma utilizada majoritariamente para comunicação com o resto da equipa, onde foram criadas salas conjuntas com as unidades de Requisitos e de Testes, com as quais a unidade de IMP mantém contacto mais frequentemente.
* Figma: www.figma.com – Plataforma utilizada pelos colaboradores de Front-end para fazer, visualizar e editar mockups da plataforma junto à unidade de Requisitos.
* Trello: www.trello.com – Usado para especificação e atribuição de tarefas (tanto no board geral daequipa quanto no board interno da unidade de IMP, onde tarefas detalhadas são postas em cartões e classificadas quanto a nível de dificuldade e área de especialização.
* Diagrama de Gantt: www.ganttproject.biz/ - Programa livre para realização de cronogramas que permite a gestão das tarefas realizadas e planeadas, com a informação do dia da realização e a dependência entre tarefas, usado para compor os gráficos presentes nas apresentações semanais.
* Ficheiro semanal de tarefas da equipa: Ficheiro disponibilizado pela equipa de Gestão de Projeto para ficar a par do tempo estimado e dedicado das tarefas de cada elemento das unidades. Preenchido por cada colaborador individualmente.
* Presencialmente: Usada como primeira opção de contacto uma vez que torna a passagem de informação mais clara. Esta forma de contacto foi feita através de reuniões de equipa e de reuniões com o cliente. Tanto as reuniões de equipa, como as reuniões com o cliente, foram registadas em ata.

### 3.2.1 MQ\_IMP\_ CODIGO

|  |  |  |
| --- | --- | --- |
| Disciplina: IMP | Data: 15.11.2018 | Coordenador: Caio Brêda |
| Status: baselined | Versão: v2.1 |  |

**Descrição do processo:**

Método de produção do código e design da plataforma.

**Pessoas e as suas funções:**

* Existem duas sub-unidades dentro da equipa: Back-End e Front-End. Cada membro só está inserido numa.
* Design Gráfico é tratado pela equipa de Front-End.

**Inputs:**

Lista de Tarefas ‘To do’ no trello de Implementação.

**Critério de entrada/ ativação:**

Atribuição das datas de início da tarefa e seus responsáveis, no início de cada semana. Em caso de imprevistos, os responsáveis pela tarefa devem comunicar ao coordenador, que então restabelece a prioridade das tarefas, comunicando com e identificando as pessoas no grupo de IMP no Messenger ou no Slack.

**Descrição de Tarefas:**

* Criação das tarefas no Trello, na lista ‘To do’.
* Mover a tarefa em desenvolvimento para a lista ‘Doing’.
* Produção de código das tarefas.
* Design da plataforma.
* Envio do código para o Github.
* Pull request por parte de cada grupo.
* Merge Request.
* Mover a tarefa em desenvolvimento para a lista ‘Done’.

**Validação:**

* Deverá ser feita pelo grupo de Testes.

**Critério de Saída:**

Todas as tarefas executadas encontram-se na lista ‘Done’ no trello de Implementação.

**Métodos de Avaliação deste processo:**

* Número de tarefas estimadas vs. tasks completas
* Número de features implementadas por semana

**Outputs gerados (artefactos):**

* Código fonte - PRoduto

**Monitorização deste processo pela equipa de Qualidade:**

A equipa de qualidade revê os processos criados

### 3.2.3 MQ\_IMP\_REUNIOES

|  |  |  |
| --- | --- | --- |
| Disciplina: IMP | Data: 15.11.2018 | Coordenador: Caio Brêda |
| Status: baselined | Versão: v2.1 |  |

**Descrição do processo:**

Este método descreve a estrutura da reunião semanal com o propósito de discussão de ideias e decisões, distribuição de tarefas e análise de resultados assim como a união e integração dos elementos da equipa.

**Pessoas e os seus roles:**

O coordenador de IMP deve manter o método da reunião, e tomar anotações para que o output seja o desejado.

Cada membro deve informar a equipa da situação atual de cada uma de suas tarefas, para que possam ser feitas alterações necessárias ao plano da Sprint. O coordenador informa a equipa de eventuais alterações, baseado nas atualizações das metas de cada sprint e no documento de Requisitos.

**Inputs:**

Documentos Reuniões IMP guardado no Google Docs e partilhado com todo o grupo. Todas as conversas registadas quer no Slack quer no Messenger do grupo de IMP. Manual de Requisitos.

**Critério de entrada/ativação:**

* Necessidade de tomada de decisões.
* Planeamento de atividades e distribuição de tarefas, quando necessário.

**Descrição das tarefas:**

* Em todas as aulas, a equipa reune-se entre si pessoalmente.
* O Team Leader será o moderador da reunião.
* Cada membro consulta o ficheiro Reunião Semanal.
* Cada membro anota a sua disponibilidade da semana no ficheiro.
* Cada membro fala à vez das suas tarefas e situação atual.
* Anotações são tomadas pela pessoa encarregue dessa semana no ficheiro.
* Tomada de decisões necessárias.
* Se necessário, alterar algum método de trabalho.
* Se necessário, comunicar com o gestor de outra equipa.
* Re-organização do Trello baseado na reunião.

**Validação:**

A validação é feita pelos membros da equipa de IMP, depois de chegada a uma conclusão e consenso

**Critério de saída de uma reunião:**

A reunião terminada, documentos validados.

**Métricas de avaliação deste processo:**

* Tempo da reunião;
* Percentagem da equipa presente;
* Número de decisões tomadas;
* Percentagem de assuntos resolvidos;

**Outputs gerados (artefactos):**

Documentação dos pontos importantes da reunião e da disponibilidade de cada um. Problemas encontrados, soluções possíveis a serem testadas na semana seguinte.

**Monitorização deste processo pela equipa de Qualidade:**

A equipa de Qualidade revê o processo.

## 3.3 Testes (TST)

**Lista de membros desta unidade:**

|  |  |  |  |
| --- | --- | --- | --- |
| Carlos | MIEEC | Coordenador | carlosfssobral@gmail.com |
| Ana Teresa | MIEEC | Colaboradora | anateresa.salgueiro31@gmail.com |
| Miguel | MIEEC | Colaborador | Migueldiogo7@gmail.com |
| João Tomás | LEI | Colaborador | joao.miguel.tomas@hotmail.com |

**Objetivos da unidade de TST:**

* Garantir a qualidade de conteúdo e de estrutura do trabalho realizado pela unidade de implementação, recorrendo ao processo de validação.
* Garantir que os requisitos do cliente são implementados e que o código desenvolvido é aplicado corretamente.
* Realizar testes de caixa-preta relativos aos diferentes aspetos da aplicação, nomeadamente o login, feed e perfil.
* Realizar testes automatizados.
* Deployment das diferentes versões do produto na plataforma web.

**Processos utilizados por esta unidade:**

|  |  |
| --- | --- |
| Processo para testar as funcionalidades implementadas | MQ\_TST\_TESTE\_FUNCIONALIDADES |
| Processo de escrita de testes | MQ\_TST\_TEST\_VECTOR |
| Processo de reuniões | MQ\_ENV\_REUNIÕES |
| Processo de *Deployment* | MQ\_TST\_DEPLOYMENT |

### 3.3.1 MQ\_TST\_TESTE\_FUNCIONALIDADES

|  |  |  |
| --- | --- | --- |
| Unidade: TST | Data: 2018.11.06 | Coordenador: Carlos Sobral |
| Status: baselined | Versão: 1.0 | Vice-Coordenador: João Tomás |

**Descrição do processo:**

Este processo tem como objetivo verificar se o código implementado por parte da equipa de implementação fornece ao utilizador as funcionalidades desejadas por parte dos requisitos do cliente.

**Inputs:**

Diferentes casos de testes, conforme o MQ\_TST\_TEST\_VECTOR (documento produzido no processo 3.5.2), de forma a testar as diferentes funcionalidades implementadas na aplicação ORCWall e também de acordo com o documento dos requisitos para o sprint em questão.

**Critério de entrada/ativação:**

Uma mensagem automática do GitHub que apareça no Slack, enviada após uma submissão no GitHub por parte de algum dos membros da equipa de implementação. Esta mensagem informa que uma nova funcionalidade foi desenvolvida e está pronta para ser testada.

**Descrição das tarefas:**

* As tarefas são divididas entre os diversos membros da unidade de testes, ficando cada um encarregue de testar uma certa funcionalidade implementada. Os testes realizados estão de acordo com o MQ\_TST\_TEST\_VECTOR.
* O tester deve inserir em cada parâmetro o valor descrito para o mesmo. Por fim, deve alterar o estado do teste para passed or failed, caso o resultado esteja de acordo com o output descrito no vetor de testes. Caso o teste falhado, o requisito não é satisfeito e é gerada uma mensagem para a unidade de implementação, para que esta possa corrigir o erro, de forma a satisfazer o requisito presente na tabela de requisitos. Quando um conjunto de testes é concluído, é gerada uma mensagem para a equipa de implementação, a informar que o código passou nos testes, ou, em caso de falha, a informar que é necessário reabrir a tarefa em causa para que o código seja corrigido.

**Critério de saída:**

Assim que o código passar com sucesso em todos os testes presentes na tabela de testes e a aplicação cumpra os requisitos presentes na tabela de requisitos, o processo é dado como terminado e o processo de Deployment pode ser iniciado.

**Métricas de avaliação deste processo:**

* Número de pessoas envolvidas em cada teste;
* Horas de trabalho semanais por cada elemento da unidade;
* Número de testes realizados por requisito;
* Taxa de testes aprovados/reprovados por requisito;

**Outputs gerados (artefactos):**

O artefacto gerado é a tabela MQ\_TST\_TESTES, onde é apresentado o resultado dos testes efetuados (aprovado/reprovado), com a data em que foi testado e com as respetivas observações sobre o mesmo.

### 3.3.2 MQ\_TST\_TEST\_VECTOR

|  |  |  |
| --- | --- | --- |
| Disciplina: TST | Data: 2018.11.06 | Coordenador: Carlos Sobral |
| Status: baselined | Versão: 1.0 | Vice-Coordenador: João Tomás |

**Descrição do processo:**

Consiste na escrita dos testes a ser realizador por um tester (pessoa encarregue de testar o código), em particular as pré-condições, os inputs (texto a ser preenchido e botões a clicar) e o resultado esperado.

As pré-condições contêm o estado do sistema antes do teste ser realizado (dados da aplicações e página onde o utilizador se encontra), os inputs referem-se aos dados inseridos pelo testes e que pretendem alterar o sistema (p.e., criar um novo utilizador) e o output diz respeito ao estado final do sistema e resultados obtidos.

Este documento contém uma série de casos de testes para cada requisito criado.

**Inputs:**

O MQ\_REQ\_SISfoi o documento utilizado como input para este processo visto que contém a tabela de requisitos e listas de mockups, ambos estritamente necessários para a compreensão do código a implementar e dos consequentes testes a realizar.

**Critério de entrada/validação:**

Uma mensagem gerada no Slack, por parte de um dos elementos da unidade de implementação, para a unidade de testes, a informar qual a tarefa a ser implementada. Deste modo, o coordenador da unidade de testes pode distribuir as tarefas pelos elementos da sua unidade, para que seja possível aos elementos da mesma preparar testes para o requisito a ser implementado.

**Descrição das tarefas:**

As tarefas são distribuídas pelos elementos da unidade. Cada elemento terá que usar, como base, os artefactos de input e desenvolver um conjunto de testes que permitam averiguar se o sistema funciona como esperado e se corresponde aos requisitos do cliente.

**Critério de saída:**

A partir do momento em que cada requisito contenha um conjunto de casos de teste suficiente para que seja testado (cerca de cinco testes por requisito), então o processo dá-se por concluído. Este será mais tarde utilizado pela unidade de testes para testar as funcionalidades implementadas. Apesar de estar concluído e a ser usado por um outro processo, este documento pode ser reaberto, para adição ou edição de casos de teste.

**Métricas de avaliação deste processo:**

* Número de testes criados por requisito;
* Carga de trabalho semanal por elemento (em horas);
* Qualidade das pré-condições e do output;

**Outputs gerados (artefactos):**

É gerada uma tabela de testes que irá conter as etapas necessárias para realizar cada caso de teste e também campos para preencher num outro processo.

### 3.3.3 MQ\_TST\_DEPLOYMENT

|  |  |  |
| --- | --- | --- |
| Disciplina: TST | Data: 2018.11.06 | Coordenador: Carlos Sobral |
| Status: baselined | Versão:1.0 | Vice-Coordenador: João Tomás |

**Inputs:**

Verificar o código desenvolvido pela unidade de implementação, que está disponível no GitHub. Ver a Versão da aplicação.

**Critério de entrada/ativação:**

Aprovação por parte dos elementos da unidade, após a finalização, com sucesso, dos testes realizados.

**Descrição das tarefas:**

As tarefas são distribuídas pelos elementos da unidade. Cada elemento terá que usar as aplicações disponíveis para a realização dos testes.

**Validação:**

É feita a validação do produto, depois de todos os testes e operações realizadas estarem a funcionar sem erros fatais.

**Critério de saída:**

Os Testes realizados ao produto final estarem concluídos com sucesso. E o produto está pronto para ser entregue ao cliente.

**Métricas de avaliação deste processo:**

* Número de pessoas envolvidas em cada teste;
* Horas de trabalho semanal de cada elemento;
* Qualidade do output;
* Outputs gerados (artefactos):
* Número de horas despendidas na tarefa
* Versão da aplicação.

## 3.4 Qualidade (QUA)

|  |  |  |
| --- | --- | --- |
| Disciplina: QUA | Data: 2018.10.28 | Coordenador: Hugo Marques |
| Status: baselined | Versão: v2.0 | Vice: Ana Almeida |

**Lista de membros desta unidade:**

|  |  |  |  |
| --- | --- | --- | --- |
| Ana Almeida | MIEEC | Colaborador | anacatarinamalmeida@gmail.com |
| Hugo Marques | MIEEC | Coordenador | hugocm@sapo.pt |
| Luís Gonçalves | LEI | Colaborador | luis.pedro9511@gmail.com |
| Maria Alves | LDM | Colaborador | maria\_alves11@hotmail.com |
| Rui Reis | LEI | Colaborador | ruidsk@gmail.com |
| Viviana Abreu | MIEEC | Colaborador | viviana\_rubina@hotmail.com |

**Objetivos da unidade:**

* Garantir a qualidade de conteúdo e de estrutura dos documentos realizados nas diversas unidades recorrendo ao processo de avaliação.
* Avaliar os processos e cumprimento atempado de tarefas das outras unidades.
* Avaliar o estado do projeto.
* Garantir que os requisitos do cliente são implementados e que o código desenvolvido é bem aplicado
* Garantir a qualidade e o bom funcionamento do projeto final.

**Processos utilizados por esta unidade:**

|  |  |
| --- | --- |
| 1.Elaboração das apresentações semanais | MQ\_QUA\_APRESENTAÇÃO\_SEMANAL |
| 2.Elaboração das atas | MQ\_QUA\_ATAS |
| 3.Atualização do MQ | MQ\_QUA\_ATUALIZAÇÃO\_MQ |
| 4.Avaliação das métricas e processos de todo o grupo | MQ\_QUA\_MÉTRICAS\_E\_PROCESSOS |
| 5.Validação de documentos | MQ\_QUA\_VALIDAÇÂO-DOCS |

### 3.4.1 MQ\_QUA\_APRESENTAÇÃO\_SEMANAL

|  |  |  |
| --- | --- | --- |
| Disciplina: QUA | Data: 2018.10.28 | Coordenador: Hugo Marques |
| Status: baselined | Versão: v2.0 | Vice: Ana Almeida |

**Descrição do Processo:**

Definição de tarefas da nossa unidade e determinação da conclusão das mesmas e recolha do ponto de situação das diferentes unidades do grupo, usando o diagrama de Gantt que nos é enviado.

**Pessoas e os seus papéis:**

Somente uma pessoa ficou responsável da obtenção dos diagramas e elaboração da apresentação.

**Inputs:**

Os artefactos recebidos das outras unidades e o estado atual das mesmas são os inputs deste processo, através dos diagramas de Gantt.

**Critério de entrada/ativação:**

Para o arranque deste processo, é necessário que o responsável pelo documento solicite a sua validação ao Gestor de Qualidade.

**Descrição das tarefas:**

* Avaliação da estrutura do documento:
* Verificação da situação atual das métricas estabelecidas
* Correção de erros e formatação dos documentos
* Avaliação do conteúdo do documento.

**Métricas e avaliação deste processo:**

* Número de horas despendidas pelo membro
* Número de tarefas realizadas

**Outputs gerados (artefactos):**

PowerPoint da apresentação semanal.

**Monitorização deste processo pela equipa de Qualidade:**

O Gestor de Qualidade monitoriza, sempre que um documento é validado, as métricas geradas.

### 3.4.2 MQ\_QUA\_ATAS

|  |  |  |
| --- | --- | --- |
| Disciplina: QUA | Data: 2018.10.28 | Coordenador: Hugo Marques |
| Status: baselined | Versão: v2.0 | Vice: Ana Almeida |

**Descrição do processo:**

Realização de um documento que sumariza tudo que que foi desenvolvido na aula prática pelas diferentes unidades da equipa.

**Pessoas e os seus papéis:**

Um membro da unidade fica responsável por recolher a informação e reuni-la num só documento.

**Inputs:**

Documentos com a anotações sobre o desenvolvido na aula das unidades.

**Critérios de entrada/ativação:**

Para o arranque deste processo, é necessário que o responsável pelo documento solicite a sua validação ao Gestor de Qualidade.

**Descrição da tarefa:**

* Obtenção de informação acerca do que foi desenvolvido e discutido por cada unidade na aula pratica.
* Junção de todos os dados obtidos num documento Word.

**Métricas de avaliação deste processo:**

* Número de equipas que enviou o sumário do que desenvolveu na aula.
* Número de horas despendidas pelo membro.

**Outputs gerados:**

Documento Word de formato ATA

**Monitorização deste processo pela equipa de qualidade:**

O Gestor de Qualidade monitoriza, sempre que um documento é validado, as métricas geradas.

### 3.4.3 MQ\_QUA\_ATUALIZAÇÃO\_MQ

|  |  |  |
| --- | --- | --- |
| Disciplina: QUA | Data: 2018.10.28 | Coordenador: Hugo Marques |
| Status: baselined | Versão: v2.0 | Vice: Ana Almeida |

**Descrição do processo:**

Análise ao atualmente escrito e reformulação de acordo com os novos critérios e informação.

**Pessoas e os seus papéis:**

Cada membro do grupo está encarregue de uma ou mais unidades.

**Inputs:**

Manual de Qualidade de cada unidade do projeto.

**Critérios de entrada/ativação:**

Para o arranque deste processo, é necessário que os responsáveis pelo documento solicitem a sua validação ao Gestor de Qualidade.

**Descrição da tarefa:**

Atualizar o manual de qualidade segundo os documentos reunidos.

**Métricas de avaliação deste processo:**

* Número de equipas que enviaram o respetivo manual de qualidade.
* Número de horas despendidas pelos membros.

**Outputs gerados:**

Documento Word

**Monitorização deste processo pela equipa de qualidade:**

O Gestor de Qualidade monitoriza, sempre que um documento é validado, as métricas geradas.

### 3.4.4 MQ\_QUA\_MÉTRICAS\_E\_PROCESSOS

|  |  |  |
| --- | --- | --- |
| Disciplina: QUA | Data: 2018.10.28 | Coordenador: Hugo Marques |
| Status: baselined | Versão: v2.0 | Vice: Ana Almeida |

**Descrição do processo:**

Análise da qualidade e quantidade de documentos produzidos ou analisados pela unidade para determinar a produtividade da unidade.

**Pessoas e os seus papéis:**

Cada elemento do grupo cria ou analisa vários documentos, cuja conta final, feita por um desses membros, serve para contabilizar a produtividade.

**Inputs:**

Documentos produzidos e analisados por unidade de tempo

**Critérios de entrada/ativação:**

Para o arranque deste processo, é necessário que seja entregue um documento ao responsável pelas métricas e processos e que seja solicitada a validação ao Gestor de Qualidade.

**Descrição da tarefa:**

* Obtenção de informação acerca do que foi desenvolvido e discutido por cada unidade na aula pratica.
* Junção de todos os dados obtidos num documento Word.
* Cálculo de artefactos analisados por unidade de tempo

**Métricas de avaliação deste processo:**

* Número de documentos produzidos
* Número de documentos analisados

#### Número de horas despendidas pelo membro.

### 3.4.5 MQ\_QUA\_VALIDAÇÂO-DOCS

|  |  |  |
| --- | --- | --- |
| Disciplina: QUA | Data: 2018.11.15 | Coordenador: Hugo Marques |
| Status: Baselined | Versão: v1.0 | Vice: Ana Almeida |

**Descrição do processo:**

Quando um documento é solicitado para a validação, caso este seja pequeno, um elemento da unidade fica encarregue de encontrar os erros e de o corrigir. Caso o documento tenha mais das 30 páginas, a validação será então dividida por duas ou mais pessoas, em que um elemento será o responsável e o outro(s) o(s) revisor(es). Para este documento ser aceite, este necessita de respeitar o template definido pela equipa, sendo este o primeiro aspeto a conferir. De seguida, deve ser avaliado o conteúdo, assim como, possíveis erros frásicos e/ou ortográficos.

**Pessoas e ou seus roles:**

Qualquer elemento da unidade de qualidade pode ficar responsável pela validação de um documento.

**Inputs:**

Documento realizado por um ou mais elemento pertencentes à equipa. Estes documentos podem ser atas, documentos a serem entregues ao cliente, entre outros.

**Critérios de entrada/ativação:**

Solicitação da validação do documento por parte do elemento da equipa que o criou.

**Descrição das tarefas:**

1.Avaliação da estrutura do documento:

* Verificação do template usado, tendo este ser o que foi definido pela equipa. Averiguação dos componentes presentes na capa, tais como, autores, versão, data, nome da equipa, turma, nome da disciplina, logo, informação presente no cabeçalho e rodapé.

2. Avaliação do conteúdo

* Verificação do conteúdo de modo a apurar se este se enquadra no tipo de documento em que se enquadra, assim como possíveis correções ao mesmo.
* Possível correção de construções frásicas/erros ortográficos.

**Métricas de avaliação deste processo:**

* Número de documentos validados.
* Número de horas gastas por revisor.
* Número de alterações feitas ao documento.

**Outputs gerados:**

* Documento validado e devidamente entregue aos seus autores com os

comentários dos revisores para ser feita a implementação das alterações

pelo autor do documento original.

# Disciplinas de suporte

## 4.1 Gestão de Projeto (PM)

|  |  |  |
| --- | --- | --- |
| Disciplina: PM | Data: 10-11-2018 | Coordenador: Eduarda Duarte |
| Status: baselined | Versão: v1.1 | Vice: António Madaleno |

**Lista de membros desta Unidade:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Nome | Curso | Função | Contacto |
| ED | Eduarda Duarte | LDM | Coordenador | eduardamduarte@gmail.com |
| AM | António Madaleno | MIECC | Vice-Coordenador | uc2015231841@student.uc.pt |
| AP | Alex Pinheiro | LEI | Colaborador | uc2014227184@student.uc.pt |
| FM | Francisco Miranda | LEI | Gestor Riscos | uc2015250592@student.uc.pt |
| RS | Rita Salema | LEI | Colaborador | uc2016244347@student.uc.pt |

**Objetivos da Unidade de PM:**

* Organização e distribuição das tarefas pelas várias equipas.
* Analisar dados recolhidos e arranjar táticas para melhorar.
* Manter a comunicação entre todas as unidades.
* Tomar decisões e geri-las para que o projeto prossiga.
* Intervir, e lidar com conflitos quando necessário.
* Entregar o produto desejado com a qualidade possível no final de cada sprint.

**Comunicação dentro da equipa:**

São usadas 2 plataformas para comunicação direta entre os membros:

* Slack: [www.slack.com](http://www.slack.com) Esta plataforma é usada para comunicação quer com o resto das equipas, quer dentro da unidade (há um channel puramente para a unidade, onde também está presente o coordenador de Ambiente). É preferível que todas as conversas estejam aqui registadas, de forma a canalizar a informação e saber onde esta foi transmitida.
* Messenger: [www.messenger.com](http://www.messenger.com) apenas para casos de urgência, por se tratar de uma ferramenta de menor organização, porém de maior frequência no uso diário dos nossos membros.
* Presencial: Esta opção é preferível em assuntos extensos e de complexidade maior como por exemplo uma reunião.

Como suporte desta comunicação é efetuada a partilha de documentos nas seguintes duas plataformas:

* Github (onde todos os documentos produzidos e artefactos produzidos são guardados). Plataforma que permite à unidade partilhar e armazenar os artefatos produzidos, para consulta do resto da equipa.

<https://github.com/carladrim/carladrim.github.io/tree/master/Gest%C3%A3o%20de%20Projeto>

* Google Docs (para automatização de ficheiros) guardados na seguinte Drive: <https://drive.google.com/open?id=1pL3D3wqhrHNXAfNl56nLRdoNFdU5nzUA>
* Trello (para distribuição de tarefas). A plataforma é usada para comunicação das tarefas e atribuição assim como definição de datas limites, e ainda supervisionar o seu estado. Também é possível a anexação de ficheiros a uma tarefa, e comentar e descrevê-la. Na unidade de Gestão usamo-la para comunicar as tarefas a todas as unidades (através da labels) e ainda para organização interior da nossa unidade.
  + Trello da equipa: <https://trello.com/invite/b/d9RQBALG/47e2545e3fa5a7061e1072a8f66d8f11/sprint-goals-and-tasks>

- Trello da unidade: <https://trello.com/invite/b/QrRyDCIf/83e0cfe1b68d1fe7d6c663f2ccb3e46b/pm>

* Diagrama de Gantt: Esta ferramenta é usada para planear e organizar dentro da unidade, as tarefas de cada membro assim como datas limite. Para além de ser uma ferramenta útil para comunicação destes dados, é usado como complemento das apresentações.

[www.ganttproject.biz](http://www.ganttproject.biz)

**Processos Utilizados por esta Unidade:**

|  |  |
| --- | --- |
| Processo | ID |
| Processo de Planeamento | MQ\_PM\_PLANEAR |
| Processo de Documentação | MQ\_PM\_DOCUMENTOS |
| Processo de Reunião | MQ\_PM\_APRESENTACAO |
| Processo de Supervisão | MQ\_PM\_SUPERVISAO |
| Processo de Organização da Unidade | MQ\_PM\_ORGANIZACAO |
| Processo de Plano de Riscos | MQ\_PM\_RISCOS |

### 4.1.1 MQ\_PM\_PLANEAR

|  |  |  |
| --- | --- | --- |
| Disciplina: PM | Data: 10-11-2018 | Coordenador: Eduarda Duarte |
| Status: baselined | Versão: v1.1 | Vice: António Madaleno |

**Descrição do processo:**

Método de organização, distribuição e ‘update’ de tarefas de todas as unidades com uma data limite.

Este processo tem como objetivo facilitar a comunicação entre toda a equipa, permitindo ver o estado das tarefas dependentes assim como as suas datas finais. Usando um trello a que todas as unidades têm acesso, e atribuindo uma ‘’label’’ a cada equipa, é possível observar os objetivos de cada unidade para a Sprint.

Apenas os coordenadores podem mover e criar Cards.

**Pessoas e as suas funções:**

* O Gestor e o Vice-Gestor do Projeto devem analisar os objetivos e distribuir as tarefas.
* O resto da equipa de Gestão deve ajudar a melhorar a qualidade do plano proposto para que o esforço fique bem distribuído.

**Inputs:**

* Ficheiro escrito pelo cliente com os objetivos desta Sprint.
* Dados recolhidos na reunião com o cliente acerca do mesmo ficheiro.
* Lista de Tarefas não terminadas na Sprint anterior.

**Critério de entrada/ativação:**

Atribuição de cards, por parte do Gestor, no Trello “Goals and Tasks” a cada unidade com as tarefas e respetivas datas finais, assim como outros dados necessários à finalização da tarefa.

**Descrição de Tarefas:**

* Análise do ficheiro de requisitos do cliente no início de cada Sprint;
* Anotação das tarefas dadas, artefactos pedidos;
* Consultar o cliente, para esclarecimento de dúvidas e especificações;
* Organização da Informação;
* Consultar os coordenadores para atribuição de um esforço estimado para as tarefas;
* Colocação de tarefas, respetivas datas, detalhes e unidades no Trello “Goals and Tasks”;
* Notificar os coordenadores das diferentes unidades da disponibilização das tarefas;
* Esclarecer e apontar novas dúvidas apontadas pelos coordenadores.

**Validação:**

Deverá ser feita pelos coordenadores de cada unidade se as tarefas forem objetivas e satisfazerem o tempo disponibilizado pelos membros, assim como as capacidades destes.

**Critério de Saída:**

Colocação de todas as tarefas (cards) disponibilizadas no Trello de “Goals and Tasks”

**Métodos de Avaliação deste processo:**

* Número de Tasks criadas
* Número de horas de esforço estimado para cada unidade

**Outputs gerados (artefactos):**

* Documento ES2018\_PL6\_ORCWALL\_PM\_PlanoSemanalUnidades com o histórico de tarefas, disponibilizado no Github.
* Documento ES2018\_PL6\_ORCWALL\_PM\_PlanoGeralUnidades

### 4.1.2 MQ\_PM\_DOCUMENTOS

|  |  |  |
| --- | --- | --- |
| Disciplina: PM | Data: 10-11-2018 | Coordenador: Eduarda Duarte |
| Status: baselined | Versão: v1.1 | Vice: António Madaleno |

**Descrição do processo:**

Método de automatização e partilha dos ficheiros com informação e dados sobre as unidades. A finalidade deste processo é que seja automatizada a recolha e atualização de informação e acessada facilmente (no Github ou pinned no Slack).

É usada o Google Docs, Slides e Sheets, pela agilização de partilha e conexão de dados. Todos os documentos desta natureza são armazenados numa só Google Drive e deverão ser guardados em PDF (sempre que necessário ou terminada uma recolha) no Github. Todos os documentos que são usados de forma cíclica deverão ser cuidadosamente limpos, para que não haja erros na recolha dos dados.

**Pessoas e as suas funções:**

Todos os membros da unidade de gestão devem pensar e concretizar documentos que facilitem a automatização da recolha de dados necessários para a melhoria da equipa.

O desenvolvimento do documento será atribuído a um membro dependendo da disponibilidade durante a semana.

Esse mesmo membro ficará encarregue de limpar o documento se necessário.

**Inputs:**

Documentos e dados que se tornam repetitivos e cíclicos ou que seja necessário o preenchimento individual dos membros da equipa.

**Critério de Entrada/Ativação:**

O Gestor de Projeto ou outro membro propõe a necessidade de recolher um novo género de dados ou recolha de opiniões. Esta necessidade é votada e anunciada no Slack da equipa.

**Descrição das Tarefas:**

* A equipa de gestão decide sobre a necessidade de um novo processo de recolha de um novo tipo de informação;
* É criado um Google Docs, ou Slides ou Sheets ou Forms dependendo de qual das tarefas é de facto a mais adequada;
* É atribuído a um membro de Gestão, a edição e formatação do ficheiro;
* O ficheiro é partilhado e apresentado na Apresentação Semanal para que toda a gente seja notificada e preencha;
* O link fica disponível no Slack (pinned) e no GitHub na página principal do projeto.

**Validação:**

O gestor do projeto deve aprovar e rever cada documento, e ao fim do uso dele confirmar a sua necessidade.

**Critério de Saída:**

Obter a informação e ser capaz de editá-la facilmente, tendo tempo poupado.

**Métodos de Avaliação deste processo:**

* Esforço gasto na recolha de informações;
* Nº de membros que preencheram o documento.

**Outputs gerados (artefactos):**

Documentos no Google Docs partilhados na página principal do Github (aqui são apenas apresentados os que são acessiveis à equipa);

<https://github.com/carladrim/carladrim.github.io>

Documentos na Drive (os documentos organizados em folders, são os que estão a ser usados);

<https://drive.google.com/drive/folders/1pL3D3wqhrHNXAfNl56nLRdoNFdU5nzUA?usp=sharing>

### 4.1.3 MQ\_PM\_APRESENTACAO

|  |  |  |
| --- | --- | --- |
| Disciplina: PM | Data: 10-11-2018 | Coordenador: Eduarda Duarte |
| Status: baselined | Versão: v1.1 | Vice: António Madaleno |

**Descrição do processo:**

Este processo descreve a forma como uma apresentação semanal se deve organizar, com a finalidade de obter sempre uma conclusão dessa mesma ou apenas certificar que toda a equipa tem conhecimento da situação atual.

**Pessoas e os seus roles:**

* Vice-Gestor António Madaleno: atualiza a apresentação da semana em questão;
* Colaborador Alex Pinheiro: trata de re-organizar e atualizar os diagramas de Gantt;
* Gestora Eduarda Duarte: trata de recolher as outras apresentações para tudo seja apresentado e unido num computador.
* Todos os membros da equipa: preenchem o slide dos issues com o conteúdo que acharam mais relevante de questionar durante a semana.

**Inputs:**

* Documento ES2018\_PL6\_ORCWALL\_PM\_DadosSemanais contendo o esforço individual, o progresso da sprint e evolução do esforço.
* Cards do Trello.
* Outros documentos relacionados com a convocação da reunião, mencionados pelo Gestor.
* Documento Google Slides com o modelo da apresentação.
* Apresentações das outras unidades.

**Critério de entrada/ativação:**

O começo de uma nova semana inicia as alterações na apresentação.

**Descrição das tarefas:**

* Transferir o plano da semana para a apresentação com os esforços estimados;
* Correção de problemas que o professor tenha apontado anteriormente;
* Todos os membros devem recolher de issues ao longo da semana;
* Todos os membros devem preencher o esforço aplicado e o status da tarefa no fim da semana;
* Atualizar dos gráficos;
* Inserir os novos diagramas de Gantt;
* Guardar em PDF a apresentação finalizada dessa semana assim como outros documentos da semana;
* Apresentar, ouvindo as questões da equipa e reunindo soluções;
* Dar upload no Github;
* Limpar os documentos depois de feita uma cópia de segurança.

**Validação:**

A validação é feita pelo Moderador da Reunião, depois da chegada a uma conclusão e consenso.

**Critério de saída:**

A validação é feita pelo Gestor de Projeto (Eduarda Duarte) que revê a apresentação e corrige.

**Métricas de avaliação deste processo:**

* Esforço gasto na atualização da apresentação;
* Feedback dado pelo professor (nota);

**Outputs gerados (artefactos):**

Ata da reunião no GitHub no folder Qualidade com a data da reunião.

Documento de Apresentação semanal no folder do Github de Gestão de Projeto.

https://github.com/carladrim/carladrim.github.io/tree/master/Gest%C3%A3o%20de%20Projeto/Apresenta%C3%A7%C3%B5es%20Semanais

### 4.1.4 MQ\_PM\_SUPERVISAO

|  |  |  |
| --- | --- | --- |
| Disciplina: PM | Data: 10-11-2018 | Coordenador: Eduarda Duarte |
| Status: baselined | Versão: v1.1 | Vice: António Madaleno |

**Descrição do processo:**

Este processo descreve a forma como a unidade de Gestão deve supervisionar o progresso do plano e o esforço das outras equipas, assim como detetar anomalias para poder resolvê-las rapidamente.

**Pessoas e os seus papéis:**

* O gestor do projeto deve partilhar semanalmente o documento que permite a recolha de informação do esforço individual.
* O gestor de riscos deve atualizar o plano e estar atento à situação de cada unidade.
* Os membros da equipa de gestão deverão verificar o plano e ajustar a conformidade de cada unidade com as tarefas atribuídas, mantendo sempre um diálogo e verificando a organização dentro de cada unidade para que a sprint seja concluída com sucesso.

**Inputs:**

* Documento ES2018\_PL6\_ORCWALL\_PM\_DadosSemanais
* Cards do Trello.
* Documento ES2018\_PL6\_ORCWALL\_PM\_PlanoDeRiscos
* Outros documentos relacionados com a convocação da reunião, mencionados pelo Gestor.

**Critério de entrada/ativação:**

Anomalia presente em alguma das unidades, ou em algum dos ficheiros com os dados semanais, ou mensagem a descrever uma dificuldade dentro das unidades.

**Descrição das tarefas:**

* Gestor envia o documento DadosSemanais para o Slack, para que todos os membros coloquem o esforço aplicado;
* Gestor consulta os coordenadores para verificarem a fidelidade dos dados recolhidos;
* Se houver membros a não aplicar esforço: tentar atribuir tarefas de um nível diferente;
* Verificar o cumprimento dos prazos, e o progresso através do Trello;
* Gestor de Riscos fala com os coordenadores das unidades através do Slack (grupo de coordenadores) para perceber os riscos e se tudo está a correr de acordo com o plano inicial. Comunicar a necessidade de mudança do plano no caso de anomalia;
* Gestor de Riscos partilha um formulário semanal, no Slack, para que todos os membros possam contribuir para a recolha de riscos.

**Validação:**

A validação é feita pelo Gestor do Projeto, que deve ser informado dos acontecimentos, aprovando ou ajudando a tomar decisões.

**Critério de saída:**

Unidades todas estáveis e a cumprir os prazos desejados. Inexistência de problemas ou conflitos.

**Métricas de avaliação deste processo:**

* Feedback da equipa;
* Esforço de todos os membros;
* Metas cumpridas e Tarefas Acabadas;

**Outputs gerados (artefactos):**

Documento ES2018\_PL6\_ORCWALL\_PM\_DadosSemanais

https://github.com/carladrim/carladrim.github.io/tree/master/Gest%C3%A3o%20de%20Projeto/Artefactos/Esfor%C3%A7o%20da%20equipa%20individual\_%20de%20cada%20semana

### 4.1.5 MQ\_PM\_ORGANIZACAO

|  |  |  |
| --- | --- | --- |
| Disciplina: PM | Data: 10-11-2018 | Coordenador: Eduarda Duarte |
| Status: baselined | Versão: v1.1 | Vice: António Madaleno |

**Descrição do processo:**

Método que descreve o procedimento de reconhecimento, atribuição e organização das tarefas que cabe à unidade de Gestão de Projeto finalizar. Este processo visa a facilitar e automatizar a comunicação indireta entre os membros, dando a conhecer, e notificar cada um destes, as tarefas e os detalhes desta.

**Pessoas e as suas funções:**

Gestor de Projeto e Vice-gestor devem fazer o plano da sprint, distribuir as tarefas, atualizar o manual de qualidade, e tomar decisões.

**Inputs:**

* Ficheiro escrito pelo cliente com os objetivos desta Sprint.
* Dados recolhidos na reunião com o cliente acerca da sprint atual.
* Lista de Tarefas não terminadas na Sprint anterior nas cards do Trello

**Critério de entrada/ ativação:**

Começo de uma nova semana/ reset da card no Trello

**Descrição de Tarefas:**

* Reset dos Cards de repetição no Trello a cada início de semana;
* Analisar os documentos e anotar tarefas para PM;
* Atribuir tarefas específicas da semana através do Trello “PM” na lista ‘’To Do’’
* Fazer o Diagrama de Gantt baseado nas cards;
* Partilhar o Diagrama no folder do Github correspondente;
* Colocar datas limite, assim como os owners das tarefas;
* Ao desenvolver uma tarefa colocá-la em ‘Doing’;
* Se uma tarefa estiver em atraso, avisar no Slack da unidade;
* Quando uma tarefa está completa passá-la para a lista de ‘Done’;

**Validação:**

Deverá ser feita pelo Gestor de Projeto, ou pelo Vice-Gestor, que deverão confirmar o progresso das tarefas assim como a quantidade de esforço exigido. Os membros da unidade também se deverão manifestar sobre as tarefas que lhes são atribuídas, aceitando ou não.

**Critério de Saída:**

Cards das tarefas na lista ‘Done’ no trello.

**Métodos de Avaliação deste processo:**

* Nº de tarefas distribuídas;
* Esforço exigido de cada membro;

**Outputs gerados (artefactos):**

Cards no Trello.

Diagramas de Gantt

https://github.com/carladrim/carladrim.github.io/tree/master/Gest%C3%A3o%20de%20Projeto/Artefactos/Reportings%20e%20Outros%20Documentos/Diagramas%20de%20Gantt

### 4.1.6 MQ\_PM\_RISCOS

|  |  |  |
| --- | --- | --- |
| **Disciplina:** PM | **Data:** 10-11-2018 | **Coordenador:** Eduarda Duarte |
| **Status:** baselined | **Versão:** 1.1 | **Vice:** António Madaleno |

**Descrição do processo:**

Método que descreve o processo de criação e atualização do Plano de Riscos

**Pessoas e as suas funções:**

O Gestor de Riscos com a ajuda de um colaborador supervisiona e atualiza o manual de riscos conforme os objetivos da sprint e estado das unidades.

**Inputs:**

* Documento de ES2018\_PL6\_ORCWALL\_PM\_DadosSemanais
* Documento de ES2018\_PL6\_ORCWALL\_PM\_PlanoSprint

**Critério de entrada/ ativação:**

Alguma anomalia dentro de uma das unidades ou um novo objetivo ou tarefa para a sprint em questão.

**Descrição de Tarefas:**

* Avaliar e analisar os objetivos e as tarefas da sprint;
* Recolher informação sobre os riscos possíveis de cada unidade;
* Partilhar um formulário no Slack com os membros da equipa;
* Imaginar casos possíveis e planos de solução no caso destes acontecerem;
* Preencher o documento com o Plano;
* Atualizar o documento;
* Supervisionar a equipa para saber se é necessário pôr um destes planos em ação.

**Validação:**

Deverá ser feita pelo Gestor de Projeto, ou pelo Vice-Gestor, que deverão confirmar a qualidade do Plano de Riscos.

**Critério de Saída:**

Todos os casos de interesse analisados e prevenidos.

**Métodos de Avaliação deste processo:**

* Quantidade de Riscos encontrados;
* Quantidade de Riscos cobertos;
* Issues de cada unidade.

**Outputs gerados (artefactos):**

## Documento ES2018\_PL6\_ORCWALL\_PM\_PlanoDeRiscos

## https://github.com/carladrim/carladrim.github.io/tree/master/Gest%C3%A3o%20de%20Projeto/Artefactos/Reportings%20e%20Outros%20Documentos/Plano%20de%20Riscos

## 4.2 Ambiente (ENV)

|  |  |  |
| --- | --- | --- |
| Disciplina: ENV | Data: 2018.10.05 | Coordenador: Carla Mendes |
| Status: baselined | Versão: 1.0 | Vice: Sara Costa |

**Lista de membros desta unidade:**

|  |  |  |
| --- | --- | --- |
| Carla Mendes | LDM | carladrianamendes@gmail.com |
| Helena Tavares | LDM |  |
| Sara Costa | LDM | lenasofmt@gmail.com |

**Objetivos da unidade de ENV:**

|  |  |
| --- | --- |
| 1.Regulação e registo do ambiente do trabalho da equipa. | MQ\_ENV\_REGULACAO |
| 2.Workshops internos ou, caso necessário, externos. | MQ\_ENV\_WORKSHOPS |
| 3.Escolha dos programas a usar na elaboração do trabalho | MQ\_ENV\_SOFTWARE |
| 4.Desenvolvimento da Arquitetura de Software. | MQ\_ENV\_ADS |

**Processos comuns da Equipa**

* Assegurar o bom funcionamento de toda a equipa a partir de recursos auxiliares como tutoriais, slack entre outros.
* Pesquisa profunda sobre os melhores materiais a serem utilizados.

**Critério de entrada/ativação**

Falta de conhecimento de alguma app ou recurso auxiliar presente em alguma das unidades, ou em algum dos ficheiros com os dados semanais, ou mensagem a descrever uma dificuldade dentro das unidades.

### 4.2.1 MQ\_ENV\_REGULACAO

|  |  |  |
| --- | --- | --- |
| Disciplina: ENV | Data: 2018.10.05 | Coordenador: Carla Mendes |
| Status: baselined | Versão: 1.0 | Vice: Sara Costa |

**Descrição do processo:**

Regulação dos softwares utilizados pelas várias equipas, nomeadamente Slack, Trello, GitHub, Google Docs, entre outros.

**Pessoas e as suas funções:**

Qualquer membro da unidade pode efetuar a regulação, sempre que seja necessário atualizar/regular algum dos critérios dos softwares mencionados, a pedido ou auxílio de qualquer outro membro da equipa.

**Inputs:**

* Pedidos de auxílio por qualquer membro da equipa.
* Regulação consoante qualquer alteração que ocorra na equipa.

**Critério de entrada/ ativação:**

Alteração e regulação de ambientes de trabalho, mudança de cargos de membros, atualizações de softwares, etc.

**Descrição de Tarefas:**

* Análise de um pedido de um membro da equipa;
* Alterações de cargos/mudanças de quipas.
* Atualizações de softwares.
* Organização da Informações/documentos/pastas.
* Esclarecer e apontar novas dúvidas apontadas por membros da equipa.
* Constante apoio a toda a equipa para verificar se estão todos em sintonia com o novo programa ou método.

**Validação:**

Deverá ser feita pelo coordenador da Unidade para que seja aceite e validada.

**Critério de Saída:**

Regulação efetuada, mudanças feitas, novos links.

**Métodos de Avaliação deste processo:**

* Número de softwares em utilização;
* Correto funcionamento dos softwares.

**Outputs gerados (artefactos):**

Plataformas online (GitHub, Slack, Trello, Google Docs);

### 4.2.2 MQ\_ENV\_WORKSHOPS

|  |  |  |
| --- | --- | --- |
| Disciplina: ENV | Data: 2018.10.05 | Coordenador: Carla Mendes |
| Status: baselined | Versão: 1.0 | Vice: Sara Costa |

**Descrição do processo:**

Workshops com unidades/membros de unidades para que as dúvidas geradas pelos softwares sejam esclarecidas e os tutoriais postos em prática.

**Pessoas e as suas funções:**

Qualquer membro da unidade pode dar os workshops desde que estejam aptos para estes e após a validação do coordenador.

**Inputs:**

* Pedidos de auxílio por qualquer membro da equipa.
* Dúvidas em relação a um software em utilização.
* Dificuldades em trabalhar com um certo software por parte de membros/unidades da equipa.

**Critério de entrada/ ativação:**

Alteração e regulação de ambientes de trabalho, mudança de cargos de membros, atualizações de softwares, etc.

**Descrição de Tarefas:**

* Análise de um pedido de um membro da equipa;
* Estudo da dúvida em causa;
* Estudo do funcionamento correto do software (básico e avançado);
* Reunião/marcação do workshop;
* Esclarecer e apontar novas dúvidas apontadas por membros da equipa;
* Constante apoio a toda a equipa para verificar se estão todos em sintonia com o novo programa ou método.

**Validação:**

Deverá ser feita pelo coordenador da Unidade para que seja aceite e validada, ou pro alguém experiente no software em questão.

**Critério de Saída:**

Tutoriais, workshop.

**Métodos de Avaliação deste processo:**

* Número de worshops efetuados;
* Número de dúvidas existentes.

**Outputs gerados (artefactos):**

Tutoriais (GitHub), Workshops, atas de reuniões.

### 4.2.3 MQ\_ENV\_SOFTWARE

|  |  |  |
| --- | --- | --- |
| Disciplina: ENV | Data: 2018.10.05 | Coordenador: Carla Mendes |
| Status: baselined | Versão: 1.0 | Vice: Sara Costa |

**Descrição do processo:**

Procura/estudo aprofundado de possíveis softwares a utilizar.

**Pessoas e as suas funções:**

Coordenador ou vice-coordenador.

**Inputs:**

* Pedidos de um novo software por parte de uma unidade/membros da equipa/professor da cadeira.
* Mudança de software.

**Critério de entrada/ ativação:**

Pedido de um novo software, ou necessidade de implementar um novo software de modo a desenvolver certas funcionalidades. Pedido por parte de um membro/professor auxiliar.

**Descrição de Tarefas:**

* Análise de um pedido de um membro da equipa/professor/unidade;
* Estudo de possíveis softwares;
* Apresentação das possibilidades aos membros/professores;
* Escolha do software;
* Implementação do software;
* Esclarecimento/anotação de dúvidas;
* Constante apoio a toda a equipa para verificar se estão todos em sintonia com o novo programa ou método.

**Validação:**

* Deverá ser feita pelo coordenador da Unidade/professor para que seja aceite e validada.
* Critério de Saída:
* Software aplicado, tutoriais.

**Métodos de Avaliação deste processo:**

* Número de softwares em uso;
* Facilidade de uso dos softwares em causa.

**Outputs gerados (artefactos):**

Tutoriais (GitHub), produtos dos softwares.

### 4.2.4 MQ\_ENV\_ADS

|  |  |  |
| --- | --- | --- |
| Disciplina: ENV | Data: 2018.10.05 | Coordenador: Carla Mendes |
| Status: baselined | Versão: 1.0 | Vice: Sara Costa |

**Descrição do processo:**

Elaboração do documento de Arquitetura e Design do Software.

**Pessoas e as suas funções:**

Coordenador.

**Inputs:**

* Tarefa do sprint, validada pela Unidade de Gestão.
* Critério de entrada/ ativação:
* Cartão de tarefa no Trello, validado pelo Unidade de Gestão.

**Descrição de Tarefas:**

* Análise do todo o software utilizado pela equipa.
* Estudo aprofundado de todos os softwares.
* Criação do documento.
* Criação dos diagramas e métodos.

**Validação:**

Deverá ser feita pela equipa de Qualidade e Gestão.

**Critério de Saída:**

Artefacto (ES2018\_OrcWall\_Arquitectura-de-Software\_v1.0.pdf)

**Métodos de Avaliação deste processo:**

* Número de horas despendido para a elaboração do artefacto.

**Outputs gerados (artefactos):**

Artefacto (ES2018\_OrcWall\_Arquitectura-de-Software\_v1.0.pdf)