

MANUAL DE QUALIDADE

ENGENHARIA DE SOFTWARE 2018/2019

PL6



ORCWALL

Data:

13-11-2018

Versão:

4.1

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: <ul style="list-style-type: none"> • HM • AA • LG • RR • VA 	Estrutura Inicial do Documento
2.0	2018.10.05	Departamento Qualidade: <ul style="list-style-type: none"> • HM • AA • LG • RR • VA 	Atualização do Documento
3.1	2018.11.17	Departamento Qualidade: <ul style="list-style-type: none"> • HM • AA • LG • RR • VA • MA 	Atualização do Documento
4.1	2018.12.05	Departamento Qualidade: <ul style="list-style-type: none"> • HM • AA • LG • RR • VA 	Atualização da secção de QUA Atualização da secção de IMP

Índice

Tabela de versões e revisões.....	2
1. Introdução	4
1.1 Contexto	4
1.2 Apresentação do projeto	4
1.3 Glossário	5
1.4 Referências	6
2. Regras gerais da equipa ORCWALL	7
3. Processos de Engenharia.....	8
3.1 Requisitos (REQ)	8
3.1.1 MQ_REQ_SRS.....	10
3.2. Implementação (IMP)	13
3.2.1 MQ_IMP_CODIGO	15
3.3 Testes (TST)	17
3.3.1 MQ_TST_TESTE_FUNCIONALIDADES	18
3.3.2 MQ_TST_TEST_VECTOR	20
3.3.3 MQ_TST_DEPLOYMENT	22
3.4 Qualidade (QUA).....	23
3.4.1 MQ_QUA_ATAS	25
3.4.2 MQ_QUA_ATUALIZACAO_MQ	26
3.4.3 MQ_QUA_VALIDACAO-DOCS	27
4. Disciplinas de suporte	28
4.1 Gestão de Projeto (PM).....	28
4.1.1 MQ_PM_PLANEAR	30
4.1.2 MQ_PM_SUPERVISAO.....	32
4.1.3 MQ_PM_RISCOS.....	34
4.2 Ambiente (ENV).....	35
4.2.1 MQ_ENV_REGULACAO	36
4.2.2 MQ_ENV_WORKSHOPS	37
4.2.3 MQ_ENV_SOFTWARE	38
4.2.4 MQ_ENV_ADS	39

1. Introdução

1.1 Contexto

No âmbito do projeto “Infor-Investigador” desenvolvido na disciplina de Engenharia de Software, lecionada pelos docentes Mário Zenha Relá 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.

1.2 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.

1.3 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

1.4 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.

2. 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.

3. Processos de Engenharia

3.1 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: REQ	Data: 16/11/2018	Coordenador: Paulo Sousa
Status: baselined	Versão: 3.0	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 com o 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.
- Apresentações:
<https://github.com/carladrim/carladrim.github.io/tree/master/Requisitos/Apresenta%C3%A7%C3%B5es%20semanais>

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 cada tarefa;
- 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).
- Atas:
<https://github.com/carladrim/carladrim.github.io/tree/master/Requisitos/Atas>
- Protótipos:
<https://github.com/carladrim/carladrim.github.io/tree/master/Requisitos/Rascunhos>
- Manual de Qualidade:
https://github.com/carladrim/carladrim.github.io/blob/master/Requisitos/ES2018_ORCWall_Requisitos_Manual-de-qualidade_v3.0.pdf
- Evidências de inspeção:
https://github.com/carladrim/carladrim.github.io/tree/master/Requisitos/evid%C3%A2ncias_inspe%C3%A7%C3%A3o

- Documento de Requisitos:
<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
ED	Eduarda Duarte	LDM	Colaborador FE	eduardamduarte@gmail.com
JS	João Simões	MIEEC	Colaborador BE	joaopedrosimoes@gmail.com
SO	Samuel Outeiro	LEI	Vice-Coordenador/BE	samuelbouteiro@gmail.com
ZT	Žiga 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 requisitos, 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

Comunicação dentro da unidade:

- Slack: www.slack.com - Plataforma utilizada maioritariamente 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 da equipa 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.

3.3 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 <i>Deployment</i>	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.
- Apresentações semanais:
<https://github.com/carladrim/carladrim.github.io/tree/master/Testes/Sprint%201%20-%20Testes/apresenta%C3%A7%C3%B5es%20semanais> (1ª sprint)

https://github.com/carladrim/carladrim.github.io/blob/master/Testes/Sprint%203%20-%20Testes/Sprint3_week2.pptx (3ª sprint)

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.
- Manual de testes:
<https://github.com/carladrim/carladrim.github.io/blob/master/Testes/Sprint%20-%20Testes/Manual%20Testes.docx>
- Tabela com testes de aceitação:
[https://github.com/carladrim/carladrim.github.io/blob/master/Testes/Sprint%20-%20Testes/Tabela%20com%20os%20testes%20de%20aceita%C3%A7%C3%A3o%20\(GATE\).pdf](https://github.com/carladrim/carladrim.github.io/blob/master/Testes/Sprint%20-%20Testes/Tabela%20com%20os%20testes%20de%20aceita%C3%A7%C3%A3o%20(GATE).pdf)
- Testes de aceitação:
<https://github.com/carladrim/carladrim.github.io/blob/master/Testes/Testes%20de%20aceita%C3%A7%C3%A3o.docx>

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_SIS foi 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):

- Tabela de testes que irá conter as etapas necessárias para realizar cada caso de teste e também campos para preencher num outro processo:
[https://github.com/carladrim/carladrim.github.io/blob/master/Testes/Sprint%20-%20Testes/Tabela%20com%20os%20testes%20de%20aceita%C3%A7%C3%A3o%20\(GATE\).pdf](https://github.com/carladrim/carladrim.github.io/blob/master/Testes/Sprint%20-%20Testes/Tabela%20com%20os%20testes%20de%20aceita%C3%A7%C3%A3o%20(GATE).pdf)

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	Vice	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 atas	MQ_QUA_ATAS
2.Atualização do MQ	MQ_QUA_ATUALIZACAO_MQ
3.Validação de documentos	MQ_QUA_VALIDACAO-DOCS

Comunicação dentro da equipa:

São usadas 2 plataformas para comunicação direta entre os membros:

- Slack: (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.
- 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 artefactos produzidos, para consulta do resto da equipa.
- <https://github.com/carladrim/carladrim.github.io/tree/master/Qualidade>
- 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, comentar e descrevê-la.
- <https://trello.com/invite/b/d9RQBALG/47e2545e3fa5a7061e1072a8f66d8f11/sprint-goals-and-tasks>
- 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

3.4.1 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 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 as anotações sobre o que foi discutido na aula pelos membros presentes; Folha de presenças da aula.

Critérios de entrada/ativação:

É necessário o começo de uma aula prática para o arranque deste processo.

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:

- <https://github.com/carladrim/carladrim.github.io/tree/master/Qualidade> (atas gerais)
- <https://github.com/carladrim/carladrim.github.io/blob/master/Qualidade/Qualidade/ATA%20-%20Reuni%C3%A3o%20Extraordin%C3%A1ria%2030.10.2018.docx> (ata de reunião extraordinária)
- <https://github.com/carladrim/carladrim.github.io/blob/master/Qualidade/Qualidade/ata%20pl6%2020-09-2018/Ata%20PL6%20-%2020-09-2018.pdf> (20/09/2018)
- https://github.com/carladrim/carladrim.github.io/blob/master/Qualidade/sprint%201%20-%20qualidade/ORCWALL_ATA_Geral-27.09.2018_v1.0.pdf (27/09/2018)
- <https://github.com/carladrim/carladrim.github.io/blob/master/Qualidade/Qualidade/ata%2011.10.2018.docx> (11/10/2018)

Monitorização deste processo pela equipa de qualidade:

O Coordenador de Qualidade valida o documento.

3.4.2 MQ_QUA_ATUALIZACAO_MQ

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

Descrição do processo:

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

Pessoas e os seus papéis:

Cada membro do grupo está encarregue de atualizar uma ou mais unidades, dependendo do necessário.

Inputs:

Manual de Qualidade de cada unidade do projeto.

Crítérios de entrada/ativação:

Para o arranque deste processo, é necessário que os responsáveis pelos documentos solicitem a sua validação ao Coordenador de Qualidade.

Descrição da tarefa:

Atualizar o manual de qualidade segundo os documentos reunidos e critérios necessários.

Métricas de avaliação deste processo:

- Número de processos dos MQs revistos.
- Número de horas despendidas pelos membros.

Outputs gerados:

- [https://github.com/carladrim/carladrim.github.io/blob/master/Qualidade/sprint%201%20-%20qualidade/ES2018_PL6_ORCWALL_QUA_ManualdeQualidade%20\(2\).docx](https://github.com/carladrim/carladrim.github.io/blob/master/Qualidade/sprint%201%20-%20qualidade/ES2018_PL6_ORCWALL_QUA_ManualdeQualidade%20(2).docx) (sprint 1)
- https://github.com/carladrim/carladrim.github.io/blob/master/Qualidade/Qualidade/MQ_ORCWALL_v1.1.pdf (sprint 1 - protótipo)
- https://github.com/carladrim/carladrim.github.io/blob/master/Qualidade/ES2018_PL6_ORCWALL_QUA_ManualdeQualidade_v3_1.pdf (sprint 3, versão 1)
-

Monitorização deste processo pela equipa de qualidade:

O Coordenador de Qualidade revê e valida o documento final.

3.4.3 MQ_QUA_VALIDACAO-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 20 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. Dependendo do tamanho ou quantidade de erros, ou é criado um documento com os erros encontrados, ou o documento em si é corrigido, ou então são feitos comentários no documento usando a ferramenta própria do *Microsoft Word*.

Pessoas e ou seus roles:

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

Inputs:

Qualquer documento realizado por um ou mais elemento, pertencentes à equipa, que necessite de validação.

Crítérios 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 de 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.
- Quantidade 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; Ou documento corrigido, se este necessitar de poucas alterações; Ou documento gerado com respectivos erros a corrigir.

4. 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 (através do Trello, e notificando pelo Slack) pelas várias unidades definindo datas limite;
- Planos de Riscos partilhado no Github;
- Analisar dados recolhidos com a ajuda das ferramentas da Google;
- Manter a comunicação entre todas as unidades, esclarecendo dúvidas, e solucionando problemas ou puramente para recolher dados mais específicos do progresso;
- Tomar decisões e geri-las para que o projeto prossiga, com vista a melhorar os processos a cada sprint, facilitando a circulação de informação, comunicação...
- 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 3 plataformas para comunicação direta entre os membros:

- Slack: 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 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 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 artefactos 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

Processos Utilizados por esta Unidade:

Processo	ID
Processo de Planeamento	MQ_PM_PLANEAR
Processo de Supervisão	MQ_PM_SUPERVISAO
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_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:

Gestora Eduarda Duarte deve partilhar semanalmente o documento que permite a recolha de informação do esforço individual assim como tomar decisões e intervir quando necessário.

Gestor de riscos Francisco Miranda e a Colaboradora Rita Salema devem 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 ou em caso de haver anomalias, o ciclo do processo dá-se como encerrado quando é tomada uma nova decisão.

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

4.1.3 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 equipas.
- 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 por alguém experiente no software em questão.

Critério de Saída:

Tutoriais, workshop.

Métodos de Avaliação deste processo:

- Número de workshops 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)