**Requisitos (REQ)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Unidade:** Requisitos | **Data:** 14.11.2018 | **Status:** baselined | **Versão:** 3.0 |

**Lista de Membros da unidade:**

|  |  |  |  |
| --- | --- | --- | --- |
| Eduardo Guerra | LEI | Coordenador | uc2016231080@student.uc.pt |
| Carolina Batista | LDM | Vice-Coordenadora | uc2016240419@student.uc.pt |
| Bruna Rosas | LDM | Colaboradora | uc2016239564@student.uc.pt |
| Bruno Simões | LDM | Colaborador | uc2016238684@student.uc.pt |
| Joana Mateus | LDM | Colaboradora | uc2016225749@student.uc.pt |

**Trabalho desenvolvido pela equipa de requisitos:**

A equipa de requisitos é constituída por um grupo de elementos cujo trabalho é produzir mockups, tabelas de requisitos e o documento final de Software Requirement Specifications (SRS). Para isso são elaborados uma série de processos, métodos de trabalho que visam o desenvolvimento de um projeto contínuo e bem-sucedido, pretendendo repartir o esforço igualmente por cada membro da unidade.

**Objetivos da unidade de REQ:**

1. A equipa de requisitos trabalha com o foco de corresponder às exigências do cliente, realizando uma tabela de requisitos que têm de estar em sintonia com os mockups, que até serem validados permanecem em constante melhoramento. É um trabalho que tem de ser acompanhado pela unidade de implementação, pois serve-lhes de inputs e pode sofrer alterações consoante dificuldades ou escolhas de implementação.
2. Faz a ponte entre o cliente e a unidade de implementação, sendo que o trabalho desenvolvido inicialmente foi em conjunto com o desta unidade. Os outputs desta unidade servem como critérios de entrada e matéria primas para esta unidade e a de testes. Tem ainda o papel de investigar o melhor design possível a implementar e o melhor nível de usabilidade, para que o utilizador consiga perceber facilmente o funcionamento da plataforma e acima de tudo, usá-la.

**Comunicação e ferramentas de trabalho**

* Figma: www.figma.com - Plataforma para prototipagem e design destinada a mockups de alta fidelidade. Sendo uma plataforma online permite que estejam várias pessoas a trabalhar no mesmo projeto, podendo não só ver todo o trabalho desenvolvido como participar em tempo real em conjunto com os outros. É uma ótima plataforma não só para os designers da unidade, como também para os restantes membros que vão estando a par do trabalho verificando se este está a ser feito corretamente.
* Discord: Plataforma genérica ao resto da equipa. Para comunicação somente entre membros da unidade de requisitos existe o canal “requisitos” e para comunicação com a unidade de implementação existe o canal “requisitos-implementação”.

Google Drive, Google Docs, Google Sheets - plataformas genéricas ao resto da equipa. Todos os membros da PL têm acesso aos documentos desta unidade.

**Processos utilizados por esta unidade:**

|  |  |
| --- | --- |
| Criação da tabela de requisitos | **MQ\_REQ\_TABELA\_REQ** |
| Desenvolvimento de mockups | **MQ\_REQ\_MOCKUPS** |
| Construção do SRS | **MQ\_REQ\_SRS** |

**MQ\_REQ\_TABELA\_REQ**

|  |  |
| --- | --- |
| **Disciplina:** Requisitos | **Coordenador:** Eduardo Guerra |
| **Status:** concluded | **Vice:** Carolina Batista |

**Pessoas e os seus roles:**

Responsável pela elaboração da tabela de requisitos:

* Eduardo Guerra (LEI), coordenador de requisitos.

Responsáveis pela validação:

* Fábio Barata (LDM), colaborador de qualidade.
* Fernando Felício (LDM), coordenador de implementação.

**Inputs:**

Reuniões com o professor.

Informação disponibilizada no Fórum de Discussão da disciplina de Engenharia de Software.

**Critério de entrada/ativação:**

É através dos inputs dados pelo cliente, Mário Alberto da Costa Zenha Rela, que é possível, definir quais os requisitos necessários à resolução do problema, sendo ainda possível dar o nosso ponto de vista de forma a trabalhar em conjunto de forma a encontrar a melhor solução.

Usamos depois estes inputs para construir o documento de requisitos com o objetivo de fornecer a informação necessária às unidades de implementação e de testes para que possam incluir e testar a informação presente no artefacto gerado, servindo assim como critério de ativação deste processo.

**Descrição das tarefas:**

* O coordenador desta unidade deve reunir-se com o cliente apontando todos os aspetos importantes sobre o projeto.
* O responsável deverá realizar a tabela no Microsoft Office Word logo que disponha de grande parte da informação. No inicio da tabela deverão estar descritas as causas que levaram à alteração da versão anterior da tabela, nomeadamente que requisitos foram alterados/removidos (e a razão para tal) e erros de formatação. A cada requisito deverá estar associado o sprint e o número.
* Validação: Entrega da tabela aos responsáveis para que a possam validar. Submissão na pasta de Requisitos no GitHub e no Discord.

**Validação:**

Correção de erros ortográficos, legibilidade, coerência. O coordenador de implementação, consoante as capacidades da unidade e os pedidos do cliente também aprova ou rejeita a validação deste documento. A linguagem tem de ser coerente e os requisitos deverão ser possíveis de implementar. Se isto tudo se verificar, então é uma tabela válida.

Caso haja alguma falha na tabela de requisitos, é preenchido um google form que está na Drive e cujo link está disponível no canal “Links Importantes” do Discord. As respostas deste formulário são usadas para corrigir as falhas do documento e servem de evidências de inspeção.

Sempre que uma validação falha, a tabela é reformulada atualizando a sua versão.

**Critério de saída:**

O conteúdo deve ser legível para toda a equipa. Após validação feita pelo cliente o artefacto está pronto a ser transmitido à unidade de implementação para que possam implementar o projeto da melhor maneira.

**Métricas de avaliação deste processo:**

Métricas anotadas semanalmente e possíveis de consultar nos slides semanais.

● Número de horas total despendidas pelos membros que realizaram a tabela.

● Número de horas despendidas por cada membro.

● Número de pessoas atribuídas a cada tarefa.

● Se as tarefas atribuídas aos respetivos membros foram concluídas.

● Número de requisitos de versão para versão.

● Número de requisitos adicionados, alterados, eliminados.

**OUTPUTS gerados (artefactos):**

Tabela de requisitos presente na pasta de Requisitos no GitHub (LABSYNC\_REQ\_TABELA.v4.X), sendo a versão validada a com o X mais elevado.

**Lifecycle:**

Versões diferentes todas as semanas após breves conversações com membros desta unidade e de implementação. Existiram no total 17 versões, 2 na sprint 1, 2 na sprint 2, 10 na sprint 3 e 3 na versão 4.

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

Equipa de qualidade fica a par do documento a partir do momento em que é colocado no Discord. Todos os artefactos necessários à realização da tabela em questão encontram-se visíveis para a unidade de qualidade permitindo que o seu trabalho de validação seja concluído da melhor forma. Alguma dúvida ou falta de compreensão é esclarecida de imediato.

### **MQ\_REQ\_MOCKUPS**

Descrição do processo:

Este processo tem como objetivo a realização de mockups que vão formar o aspeto visual da plataforma.

Input:

O artefacto gerado pelo processo anterior, ou seja, a tabela de requisitos foi usada como input para iniciar este processo.

Critério de entrada/ativação:

Para iniciar a realização dos mockups é indispensável que o artefacto anterior (tabela de requisitos) esteja devidamente validado. Esta validação é conseguida através do esforço de várias entidades como o cliente, unidades de requisitos, implementação, testes e qualidade.

As razões pelas quais a validação da tabela de requisitos é tão importante para o desenvolvimento deste processo, são os seguintes:

- Como a equipa de requisitos tem um contacto direto com o cliente, deve analisar o documento, para verificar se todos os requisitos estão em sintonia com os objetivos do cliente, prevenindo eventuais requisitos desenvolvidos ou adicionados.

- Como a tabela de requisitos vai ser lida por várias pessoas, a equipa de qualidade tem o papel de corrigir incoerências gramaticais e elegibilidade, para que as mensagens sejam corretamente transmitidas. É importante que o autor da tabela de requisitos transmita a mensagem adequada para os autores dos mockups, assim todos os documentos se sintonizam. Também é importante que as outras unidades percebam o pretendido, para que não haja requisitos devolvidos.

- A tabela de requisitos tem de estar em conformidade com os objetivos do cliente, para que o cliente não seja surpreendido com mockups que não satisfaçam esses objetivos.

- A unidade de implementação é relevante para perceber se é possível implementar determinado requisito. Assim, apenas é necessário adaptar a tabela de requisitos, e não os mockups.

- A unidade de testes também pode sugerir alterações na tabela de requisitos, prevenindo erros que possam surgir quando estiverem a testar, assim não é necessário perder tempo a realizar mockups ou criar páginas que depois não vão ser utilizadas.

Todos os pontos referidos anteriormente têm como consequências sucessivas alterações e atrasos no desenvolvimento de mockups, por isso a tabela de requisitos deve estar devidamente validada antes de se iniciar este processo.

Descrição das tarefas:

- São atribuídas tarefas para realização de wireframes (rascunhos dos mockups), os mockups de alta fidelidade.

- Os mockups são feitos com base nos rascunhos e na tabela de requisitos.

- Verificação por parte da unidade de requisitos dos mockups, para perceber se esses elementos estão de acordo com a tabela de requisitos.

- Reunião com o cliente para perceber se os mockups e os seus objetivos condizem.

- Colocação dos mockups no Discord, no canal de requisitos-implementação para que implementação possa validar.

- Colocação dos documentos finais no GitHub para que outras unidades como qualidade e testes possam validar.

Validação:

Os mockups para serem validados têm de ser legíveis, têm de estar adequados à plataforma que os vai suportar, verificar se não há impossibilidades de implementação, se a usabilidade é adequada e se cumpre os requisitos.

As validações dos mockups são feitos pelas unidades de requisitos, implementação e de qualidade e cliente.

Requisitos verifica se os mockups e a tabela de requisitos estão em sintonia, observam se os mockups estão coerentes graficamente e se não falta nenhuma funcionalidade.

Implementação verifica se tem possibilidade de implementar e sugere soluções, com o objetivo de facilitar o seu trabalho.

Qualidade verifica se não existem erros ortográficos ou incoerências.

O cliente verifica se os seus objetivos foram cumpridos, sugere alterações com o intuito de melhorar o design, a usabilidade e para facilitar o trabalho a implementação.

Caso tenham sido encontradas falhas, a equipa de requisitos, deve validar de novo os documentos corrigidos.

Critério de saída:

Assim que os mockups são validados como descrito acima, o processo dá-se como concluído. Os mockups são enviados à unidade de implementação para que possam elaborar o front-end e para que entendam o fluxo de navegação da plataforma. Deverão ainda seguir para a equipa de testes de maneira a que possam testar cada um dos requisitos.

Métricas de avaliação deste processo:

Métricas anotadas semanalmente:

- Número de pessoas envolvidas em cada tarefa;

- Número de *mockups*(novos, finalizados, alterados, total)

Outputs (artefactos):

1 documento: conjunto de mockups.

Lifecycle:

Mockups:

Versão 1 - Mockups terminados a 25-09-2018, após primeira reunião com o cliente em conjunto com a unidade de requisitos.

Versão 2- Mockupps terminados a 28-09-2018 com as alterações pedidas pelo cliente.

Versão 3- Mockups terminados a 15-10-2018 com os mockups da sprint 2.

Versão 4- Mockups terminados a 28-10-2018 após implementação pedir mockups mais específicos e mockups de páginas em falta.

Versão 5- Mockups terminados em 10-11-2018 com o redesign dos mockups, após reunião com cliente no dia 09-11-18.

Versão 6- Mockups terminados em 12-11-2018 com os objetivos de sprint 3 quase estabilizados.

Versão 7- Mockups terminados a 13-11-2018, após remoção de fotografias dos utilizadores nos posts (versão final dos mockups da sprint 3).

Nota: Não ocorreram mais versões dos mockups, já que deixou de haver um documento para os mesmos. Estes passaram a ser editados e submetidos diretamente no SRS.

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

Assim que os mockups e o diagrama de navegação são concluídos, são colocados no GitHub para que a equipa de Qualidade possa validar.

### **MQ\_LABSYNC\_REQ\_SRS**

|  |  |
| --- | --- |
| **Disciplina:** Requisitos | **Coordenador:** Eduardo Guerra |
| **Status:** concluded | **Vice:** Carolina Batista |

**Descrição do processo:**

Este é o documento que contem grande volume de trabalho da unidade de requisitos, ou seja contém os mockups com as respetivas legendas, o diagrama de navegação, fluxo de interação e os casos de uso.

**Input:**

Os artefactos gerados pelo processo anterior, ou seja, os documentos da tabela de requisitos e das mockups foram usados como input para iniciar este processo.

**Critério de entrada/ativação:**

Para iniciar este processo é de extrema importância que a tabela de requisitos e as mockups estejam devidamente validados, já que vão servir de base na construção do documento ou farão parte do mesmo.

**Descrição das tarefas:**

- São atribuídas tarefas para cada membro da unidade de requisitos.

- Verificação da validação dos inputs do SRS.

- Organização do documento (junção dos mockups, diagrama de navegação e criação de coerência visual do SRS)

- As tabelas de casos de usos são feitas com base na tabela de requisitos, mockups e diagrama.

- Colocação do SRS no discord e no GitHub para posterior validação.

**Validação:**

A função de validação cabe às unidades de requisitos, qualidade e implementação. É necessário que o documento esteja coerente gramaticalmente e visualmente, paginado, intitulado, legendado, deve ter índice, tabela de versões, autores do documento e nome da equipa (Labsync). A unidade de qualidade analisa os fatores linguísticos e de estrutura, enquanto que requisitos e implementação centra a sua análise no conteúdo. Como neste documento consta toda a informação necessária para que a plataforma seja implementada, é de extrema importância que a unidade de implementação valide o SRS.

**Critério de saída:**

Assim que todos os dados do SRS estejam estabilizados e validados, o processo é concluído. Este documento vai servir essencialmente a unidade de implementação que terá as bases para realizar o seu trabalho, mas também poderá ser útil para outras unidades que terão num único documento as principais informações da unidade de requisitos.

**Métricas de avaliação deste processo:**

Métricas anotadas semanalmente:

- Número de pessoas envolvidas em cada tarefa;

**Outputs (artefactos):**

O documento de requisitos.

**Lifecycle:**

Diagrama de navegação:

Versão 1- Diagrama terminado a 04-10-2018, em sintonia com os objetivos da sprint 1.

Versão 2- Diagrama terminado a 11-10-2018 com os objetivos da sprint 2.

Versão 3- Diagrama terminado a 15-10-2018 com acréscimo de notas para que implementação percebesse melhor o fluxo de navegação.

Versão 4 – Diagrama terminado a 6-11-2018 de acordo com a tabela de requisitos da sprint #3.

Nota: Não ocorreram mais versões do diagrama, já que deixou de haver um documento para os mesmos. Este passou a ser editado e submetido diretamente no SRS.

SRS:

Versão 1.0 - SRS terminado a 04-10-2018.

Versão 1.1- SRS terminado a 24-10-2018, após adição dos mockups da sprint 2 com as respetivas tabelas de caso de uso e atualização do diagrama de navegação.

Versão 1.2- SRS terminado a 28-10-2018, após implementação fazer pedido de novos mockups e aperfeiçoamento dos antigos. Portanto, as alterações do SRS centraram-se na substituição de mockups e adição de novos.

Versão 3.0- SRS terminado a 11-11-2018, com a adição dos mockups, diagrama de navegação e tabelas de caso de uso da sprint 3.

Versão 3.1- SRS terminado a 19-11-2018, após inspeção, as informações foram corrigidas.

Versão 3.2- SRS terminado a 24-11-2018, foram efetuadas as alterações necessárias para a sprint 4.

Versão 3.3- SRS terminado a 30-11-2018, logo que os mockups foram aperfeiçoados e as tabelas atualizadas.

Versão 3.4- SRS terminado a 01-12-2018, posteriormente à refinação dos mockups.

Versão 3.5- SRS terminado a 05-12-2018, com a devida correção do diagrama de navegação, mockups e tabelas de casos de uso após o envio da checklist, por parte do professor.

Versão 3.6- SRS terminado a 09-12-2018, com alterações de pequenos pormenores nos mockups.

Versão 3.7- SRS terminado a 11-12-2018, assim que o mockups ver bookmark e respetiva tabela foram atualizados.

Versão 3.8- SRS terminado a 11-12-2018, com a atualização dos mockups see bookmark e criar perfil.

Versão 3.9- SRS terminado a 13-12-2018, a versão final do documento.

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

Assim que o SRS foi finalizado, foi colocado no GitHub para que a equipa de Qualidade pudesse validar.