

RELATÓRIO DE SCRUM

**- Introdução**

No caso deste projeto, a metodologia *Scrum* vai ser maioritariamente utilizada na parte da implementação do código do software. Esta metodologia vai assim ser útil pois permite-nos controlar melhor as diversas funções que irão ser implementadas e dividi-las pelos diversos elementos da equipa de desenvolvimento. As duas reuniões de sprint irão ser realizadas em tempo de aula junto do professor (cliente) e com todos os elementos do grupo. Na primeira reunião a ser realizada, irá ser apresentado um protótipo da aplicação assim como todos os relatórios exigidos pelo professor.

**- Stakeholders**

Irão fazer parte deste projeto todos os elementos do grupo e o professor. No nosso caso, o trabalho foi dividido igualmente entre todos os membros, ou seja, não existe uma distinção entre equipa de desenvolvimento e gestores pois todos têm tarefas em ambos os âmbitos.

O Flávio tem a função extra de Gestor de Projeto.

O Professor Ricardo Malheiro é o Gestor de Projeto Sénior e simultaneamente o cliente.

**- Product Backlog**

Como: Utilizador Comum

Gostaria de: Pesquisar livros e saber a sua disponibilidade

Para que: Não me tenha de dirigir ao funcionário para pedir essa informação

Como: Administrador

Gostaria de: poder registar os funcionários e outros administradores

Para que: cada funcionário tenha o seu login personalizado para um melhor controlo de segurança

Como: Administrador

Gostaria de: poder exportar e importar backups das bases de dados

Para que: possa aumentar a fiabilidade do sistema

Como: Administrador

Gostaria de: poder fazer tudo o que o funcionário faz

Para que: não seja preciso um login diferente

Como: Funcionário

Gostaria de: registar e editar os utilizadores comuns

Para que: possa ficar registado a quem se alugou os livros e se possa alterar os dados a qualquer momento

Como: Funcionário

Gostaria de: registar e editar livros

Para que: possa ficar registado todos os livros da biblioteca assim como editar os mesmos a qualquer momento

Como: Funcionário

Gostaria de: pesquisar livros

Para que: se consiga obter toda a informação sobre um livro e verificar a disponibilidade e localização do mesmo

Como: Funcionário

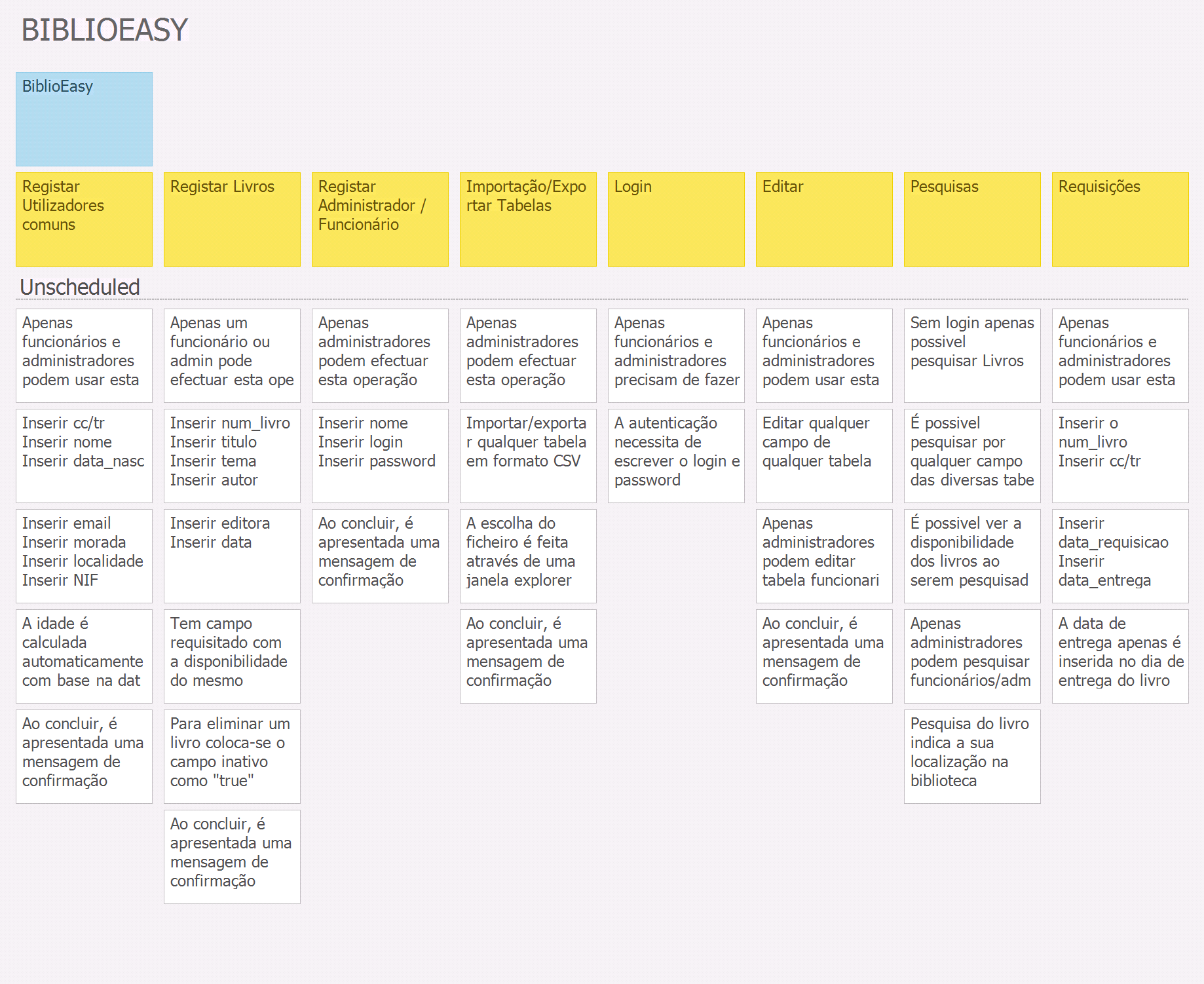
Gostaria de: pesquisar utilizadores

Para que: obter toda a informação sobre cada um dos utilizadores assim como um histórico de requisições do mesmo

Como: Funcionário

Gostaria de: poder registar requisições

Para que: o utilizador possa requisitar um livro e essa informação fique registada num histórico



|  |  |  |
| --- | --- | --- |
| ***User Story*** | ***Story Points*** | **Tempo (horas)** |
| Registar Utilizadores Comuns | 6 | 3 |
| Registar Livros | 6 | 3 |
| Registar Administrador/Funcionário | 12 | 6 |
| Importação/Exportação de Tabelas | 4 | 2 |
| Editar | 30 | 15 |
| Pesquisas | 20 | 10 |
| Requisições | 40 | 20 |

Estimamos que cada *Story Point* equivale a meia hora.

**- Sprints**

O nosso projeto engloba dois sprints com duas reuniões.

O primeiro sprint engloba as seguintes tarefas relativas á criação do código da aplicação:

|  |  |
| --- | --- |
| WBS | Task Name |
| **2** | **Sprint 1** |
| **2.1** | **Investigação javafx** |
| 2.1.1 | criação tabelas |
| 2.1.2 | adicionar campos tabela |
| 2.1.3 | editar campos tabela |
| 2.1.4 | apagar campos tabela |
| **2.2** | **Tabelas** |
| 2.2.1 | criar tabela utilizadores comuns |
| 2.2.2 | criar tabela livros |
| 2.2.3 | criar tabela requisições |
| 2.2.4 | Registar utilizadores comuns |
| 2.2.5 | Registar livro |
| 2.2.6 | editar linhas tabelas |
| 2.2.7 | apagar linhas tabelas |
| 2.3 | Janela de Informação |
| 2.4 | Registo de requisições |
| 2.5 | Devolução de requisições |
| 2.6 | Edição de requisições |
| 2.7 | pesquisa de requisições por aluno |
| 2.8 | pesquisa de requisições por livro |
| 3 | Reunião 22/5 |
| 4 | Conclusão Sprint 1 |

Este Sprint preocupa-se principalmente em criar as tabelas e as funções base delas (criar/editar/apagar linha), assim como funções para pesquisar as tabelas.

Segundo Sprint:

|  |  |
| --- | --- |
| WBS | Task Name |
| **5** | **Sprint 2** |
| 5.1 | Atualizar relatório de Requisitos (com os novos adquiridos através da reunião) |
| 5.2 | Atualizar outros relatórios |
| **5.3** | **save/load** |
| 5.3.1 | Save state .bin |
| 5.3.2 | load state .bin |
| 5.3.3 | importar tabelas .csv |
| 5.3.4 | exportar tabelas .csv |
| 5.4 | criar tabela funcionarios/admins |
| 5.5 | bloquear falta de permissões |
| 5.6 | Ocultar colunas nas pesquisas a pedido do utilizador |
| 5.7 | Validar NIF segundo formula oficial e não deixar nifs iguais |
| 5.8 | Adicionar coluna "dias em atraso" |
| 5.9 | Tipo de Utilizador (aluno, func ou docente) - Nova coluna |
| 5.10 | Listagem por entregas (ordenar por mais antigo) |
| 5.11 | Pesquisar por nome e outros campos (fazer o menu) |
| 5.12 | Bloquear utilizadores que tenham livros por entregar |
| 5.13 | Alerta de utilizadores com x dias em atraso (a pedir o x) |
| 5.14 | Defenir multas, x por dia (a pedir x) |
| 5.15 | Login |
| 5.16 | Reunião 12/6 |
| 5.17 | Conclusão sprint 2 |

Este Sprint preocupa-se principalmente nas funções de administração.