

RELATÓRIO DE SCRUM

Metodologias de Desenvolvimento de Software

RealStand,

2180601 Jason Viera Mendes.

2180662 Miguel João Guardado Ribeiro.

**Índice**

**- Introdução**

Este projeto foi gerido utilizando uma metodologia ágil chamada de Scrum. Esta metodologia ágil divide o projeto em ciclos que são as Sprints. Nessas sprints são dados e mostrados objetivos a serem cumpridos. No nosso caso tínhamos todas as semanas reuniões de 15 min (weekly scrum) onde mostrávamos os nossos progressos ao nosso Scrum Master e que de seguida passava a informação ao Product Owner, não foi possível fazer daily scrums devido ao facto de apenas haver um dia por semana de aula. Ao longo do desenvolvimento da aplicação tivemos duas Sprint Meetings, a primeira onde mostramos a gestão dos clientes e da oficina totalmente funcional, e última onde foi apresentado o produto final com todas as funcionalidades e validações de dados implementadas.

**- Stakeholders**

**Product Owners: Miguel Ribeiro, Jason Mendes, Professor Ricardo Malheiro**

Os Product Owners devem ter um diálogo com o cliente de modo a perceber todas as suas exigências, no fim do Product Owner reunir com o cliente e perceber todos os objetivos que o cliente pretende deve passar essa informação de forma clara e suscita para o Scrum Master fazer a ponte com a Development Team. O product owner ainda tem que definir objetivos para que se consiga cumprir tudo o que foi proposto.

**Scrum Master: Miguel Ribeiro, Jason Mendes**

Os Scrum Masters são a ponte entre o Product Owner e a Equipa de desenvolvimento, O scrum master é responsável pelos Daily Scrums para ir controlando o trabalho que está a ser desenvolvido pela equipa de programadores. O srum master também deverá criar os objetivos principais e dividir os objetivos em objetivos mais pequenos (user stories) para que a equipa de desenvolvimento coordene melhor o seu trabalho é faça as funcionalidades pedidas pelo cliente da melhor forma possível.

No final de cada sprint com a equipa de desenvolvimento os Scrum masters devem de reunir com o cliente e com o Product owner para verificarem se está tudo a correr como o planeado e se a aplicação esta como o cliente deseja.

**Development Team: Miguel Ribeiro, Jason Mendes**

A equipa de desenvolvimento recebe toda a informação do Scrum Master (user stories), esses user stories devem ser levados a cabo exatamente como foi pedido pelo Scrum Master.

A esquipa de desenvolvimento deve também ter atenção às priorização feita pelo Scrum Master, as tarefas com maior prioridade e que serão mostradas na próxima sprint deverão ser desenvolvidas em primeiro lugar e sem bugs ou erros.

No final ou início de cada dia a equipa de desenvolvimento deverá de mostrar todos os progressos ao scrum master.

**- Product Backlog**

**User Story Cliente:**

**Criar Cliente:**

* Estimativa (Story Points): 20
* Prioridade da Tarefa: Elevada

**Editar Cliente:**

* Estimativa (Story Points): 20
* Prioridade da Tarefa: Baixa

**Eliminar Cliente:**

* Estimativa (Story Points): 20
* Prioridade da Tarefa: Media

**User Story Carro Oficina**

**Criar Carro Oficina:**

* Estimativa (Story Points): 40
* Prioridade da Tarefa: Elevada

**Eliminar Carro Oficina:**

* Estimativa (Story Points): 10
* Prioridade da Tarefa: Baixa

**Criar Serviço:**

* Estimativa (Story Points): 30
* Prioridade da Tarefa: Elevada

**Eliminar Serviço:**

* Estimativa (Story Points): 10
* Prioridade da Tarefa: Baixa

**Criar Parcela:**

* Estimativa (Story Points): 30
* Prioridade da Tarefa: Elevada

**Eliminar Parcela:**

* Estimativa (Story Points): 10
* Prioridade da Tarefa: Baixa

**Criar Fatura Carro Oficina:**

* Estimativa (Story Points): 20
* Prioridade da Tarefa: Media

**User Story Carro Venda**

**Criar Carro Venda:**

* Estimativa (Story Points): 40
* Prioridade da Tarefa: Elevada

**Eliminar Carro Venda:**

* Estimativa (Story Points): 10
* Prioridade da Tarefa: Baixa

**User Story Venda:**

**Criar Venda:**

* Estimativa (Story Points): 30
* Prioridade da Tarefa: Elevada

**Eliminar Venda:**

* Estimativa (Story Points): 10
* Prioridade da Tarefa: Baixa

**Criar Fatura Venda:**

* Estimativa (Story Points): 10
* Prioridade da Tarefa: Media

**User Story Carro Aluguer:**

**Criar Carro Aluguer:**

* Estimativa (Story Points): 40
* Prioridade da Tarefa: Elevada

**Eliminar Carro Venda:**

* Estimativa (Story Points): 10
* Prioridade da Tarefa: Baixa

**User Story Aluguer:**

**Criar Aluguer:**

* Estimativa (Story Points): 30
* Prioridade da Tarefa: Elevada

**Eliminar Aluguer:**

* Estimativa (Story Points): 10
* Prioridade da Tarefa: Baixa

**Criar Fatura Aluguer:**

* Estimativa (Story Points): 10
* Prioridade da Tarefa: Media

**Testes Aceitação**

**Teste Erros Possíveis: (Programa Parar de Correr)**

* Estimativa (Story Points): 40
* Prioridade da Tarefa: Elevada

**Teste Bugs: (Erros no decorrer do programa que troca informação)**

* Estimativa (Story Points): 40
* Prioridade da Tarefa: Elevada

**Teste Validação de dados:**

* Estimativa (Story Points): 40
* Prioridade da Tarefa: Elevada

|  |  |
| --- | --- |
| Story Points | Duração em Horas |
| 2.5 | 15 min (0.25h) |
| 5 | 30 min (0.5h) |
| 10 | 60 min (1h) |

|  |  |
| --- | --- |
| Prioridade | Cor Prioridade |
| Baixa |  |
| Media |  |
| Elevada |  |

**- Sprints**

Ao longo do desenvolvimento deste projeto foram planeados dois sprints meetings, o primeiro foi no dia 20 de maio, com a duração de 3 horas. Neste Sprint meeting foi apresentado as funcionalidades Gestão de Clientes e da Oficina da nossa aplicação RealStand.

A gestão do cliente tem as seguintes funcionalidades:

* Criar Cliente – validação: Número Identificação Fiscal (nif) tem de ser único, e deve ser apenas numérico.
* Editar Cliente
* Eliminar Cliente – validação: o cliente não deverá ser eliminado caso tenha registos associados.

De seguida, no fim de apresentar ao Scrum master e product owner a gestão de clientes passamos para a segunda fase desta meeting, a gestão da oficina.

Funcionalidades Gestão Oficina

* Adicionar Carro Oficina- validação: o número de quadro deverá ser único.
* Eliminar Carro Oficina- validação: o carro não poderá ser eliminado caso tenha serviços associados.
* Criar Serviço- validação: a data do serviço não deverá ser anterior a atual.
* Eliminar Serviço.
* Criar Parcela
* Eliminar Parcela
* Emitir Fatura

Por fim falta o último sprint que será a aplicação completa com todas as funcionalidades pedidas pelo cliente. As últimas funcionalidades implementadas na última sprint foram todas aceites com sucesso pelo cliente logo não foram necessárias alterar para mostrar na segunda meeting.

Neste último sprint serão apresentadas as seguintes funcionalidades.

Funcionalidades Gestão Vendas:

* Criar Carro para Venda
* Eliminar Carro Venda -validação: impossível apagar carro de venda caso esteja vendido
* Vender Carro Venda
* Emitir Fatura da Venda

Funcionalidades Gestão Aluguer:

* Criar Carro Para aluguer
* Eliminar Carro Aluguer -validação: impossível apagar carro aluguer caso esteja alugado
* Alugar Carro Aluguer -validação: impossível alugar um carro que ainda esteja alugado
* Emitir Fatura Aluguer

Resumo Funcionalidades apresentadas nas Sprints

1ª Sprint

|  |  |
| --- | --- |
| Gestão Clientes | Criar Clientes |
| Editar Clientes |
| Eliminar Clientes |
|  |  |
| Gestão Oficina | Criar Carro Oficina |
| Eliminar Carro Oficina |
| Criar Serviço |
| Eliminar Serviço |
| Criar Parcela |
| Eliminar Parcela |
| Emitir Fartura Serviço |
| 2ª Sprint | |
| Gestão Vendas | Criar Carro Venda |
| Eliminar Carro Venda |
| Vender Carro |
| Emitir Fatura Venda |
|  |  |
| Gestão Aluguer | Criar Carro Aluguer |
| Eliminar Carro Aluguer |
| Alugar Carro |
| Emitir Fatura Aluguer |

BurnDown Chart – Sprint 1

BurnDown Chart – Sprint 2