# 3. Projeto da Solução

## 3.1. Requisitos funcionais

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Processo/tarefa** | **Descrição** | **Prioridade** |
| **RF001** | Identificação de usuário | O usuário faz um cadastro e em seguida efetua o login para entrar no seu perfil em que tem os projetos e tarefas. | Alta |
| **RF002** | Criação de projetos | O usuário cria um projeto e pode incluir várias tarefas no mesmo projeto | Alta |
| **RF003** | Criação de tarefas | O usuário cria uma tarefa e pode atribuir a tarefa a mais usuários | Alta. |
| **RF004** | Status da tarefa | De acordo com o andamento da tarefa o status pode estar como em andamento, em atraso, concluído | Média |
| **RF005** | Pesquisa de tarefas | Pesquisar tarefas a partir das tarefas criadas | Média. |
| **RF006** | Comentário de tarefas | Pode comentar quem está atrelado a tarefa | Alta |

## 

## 3.2. Diagrama de Classe

O objetivo deste diagrama é representar a estrutura que vai estar presente no sistema. Assim entender melhor em uma visão geral a aplicação e expressar visualmente as necessidade específicas de nossa aplicação.

# DCN2

# 3.3. Metodologia

Foi utilizado a metodologia ágil Scrum para o gerenciamento de equipe e projeto, também fizemos uso da ferramentas de apoio Trello. As técnicas utilizadas para coletas de informação foram:

* Primeiramente foi realizado uma observação direta nos executores do processo, no caso os desenvolvedores. Foi observado que o desenvolvedor executa as demandas enviadas pelo supervisor ponderando se é possível, realiza testes convencionais e caso tenha encontrado possível solução ele envia para o supervisor.
* Foi realizado também observação direta com supervisor das demandas do processo. Foi observado que o supervisor é responsável por identificar se uma demanda do cliente vai ser feita, se o resultado de uma demanda enviada pelo desenvolvedor é satisfatória e ele mesmo deve notificar o cliente sobre o projeto.
* Também foi realizado uma pesquisa sobre qualquer documentação ou notas existentes do processo. Foi encontrado descrições escritas do processo pelo supervisor sobre as demandas consideradas importantes.
* Então foi realizado entrevista com o usuário identificado como usuário chave pela equipe de analise. As perguntas foram:

- Como é decido qual demanda realizar atualmente ?

-> Orientação da supervisão ou decisão do desenvolvimento.

- Como é solicitado a alteração de um componente no software ?

-> O supervisor envia o email ou solicita verbalmente a demanda ao desenvolvimento.

- Existe alguma forma do supervisor ou desenvolvedor identificar em qual status está a demanda de um software ?

-> Somente se o supervisor questionar o andamento da demanda.

- Como é identificado se o resultado é satisfatório pelo cliente ?

-> Não havendo retorno da demanda já solucionada.

- Quem realiza o testes da demanda ?

-> O próprio desenvolvimento.

Então com as informações coletadas foi possível modelar os diagramas de processos de negócios (bpmn) e a execução do diagrama de classes. Para a modelagem BPMN foi utilizado o Bizaggi e para o diagrama de classes um plugin da IDE eclipse e o Astah UML. Já no software em si, no back-end foi utilizado a IDE eclipse, linguagem Java, serviços web Java, framework Spark e gerenciador de projetos Maven. Para o Front-end foi utilizado o Javascript, VueJs, HTML e CSS.

## 3.4. Layout da tela

O objetivo do wireframe é auxiliar o desenvolvedor no entendimento dos requisitos que foram recolhidos junto ao cliente com relação as funções e objetos que um sistema deverá conter. Com está motivação, realizamos os wireframes necessários para elucidar as possíveis telas de nossa aplicação.

## 3.4.1 Adicionar Tarefa



## 3.4.2 Adicionar Usuário

## 3.4.3 Editar Descrição da Tarefa

## 3.4.4 Editar Comentário da Tarefa

## 3.4.5 Editar Usuário

## 

## 3.4.6 Login

**3.4.7 Tarefa**

## 

## 3.4.8 Listar Tarefas 3.4.9 Listar Usuários