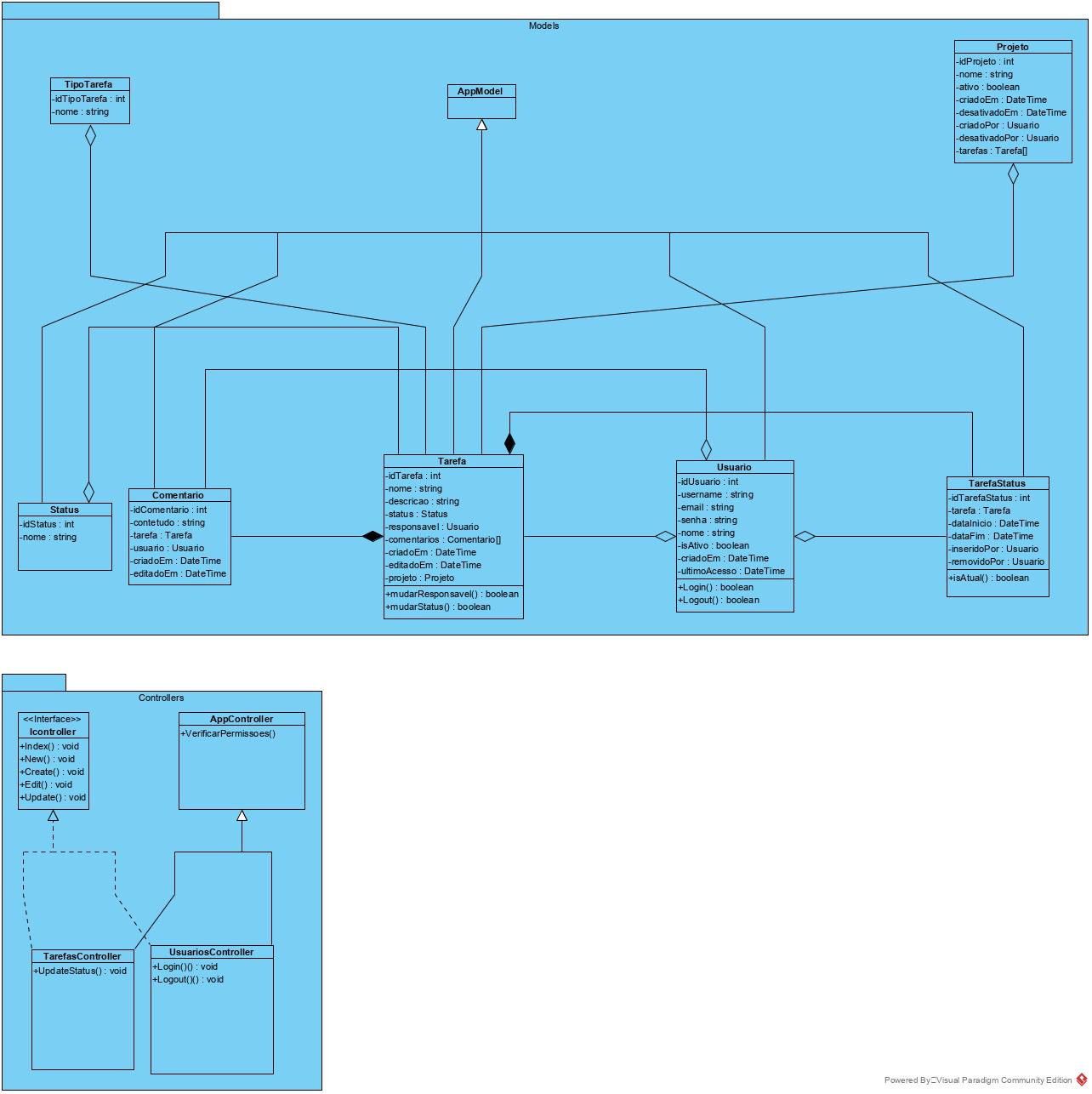
# 3. Projeto da Solução

## 3.1. Requisitos funcionais

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Processo/tarefa** | **Descrição** | **Prioridade** |
| **RF001** | Criação de tarefas. | O usuário pode cadastrar novas tarefas. | Alta |
| **RF002** | Criação de projetos. | O usuário pode cadastrar novos projetos. | Alta |
| **RF003** | Mudança de status. | O usuário pode mudar o status das tarefas. | Alta. |
| **RF004** | Descrever tarefa | O usuário pode adicionar descrição para uma tarefa. | Média |
| **RF005** | Listar tarefas | O usuário pode visualizar uma lista de tarefas filtrando por status e atribuição | Média. |
| **RF006** | Atribuição de tarefas | O usuário pode atribuir um usuário como responsável por uma tarefa. | Média |
| **RF007** | Login | O usuário deve executar login para acessar o sistema. | Média |
| **RF008** | Logout | O usuário pode encerrar sua sessão para bloquear o acesso ao sistema. | Média |
| **RF009** | Tipo de tarefa | O usuário pode alterar o tipo de tarefa. | Média |
| **RF010** | Cadastro de tipo de tarefa | O usuário pode cadastrar novos tipos de tarefas e editar os existentes. | Média |
| **RF011** | Cadastro de usuários | O usuário administrador pode cadastrar novos usuários e editar os existentes. | Média |
| **RF012** | Desativar projeto | O usuário administrador pode desativar projetos, bloqueando a criação de novas tarefas e encerrando todas as existentes. | Baixa |
| **RF013** | Alterar perfil | O usuário pode alterar seus dados básicos acessando seu perfil. | Baixa |

## 3.2. Diagrama de Classes

Observação: O diagrama de classes também está disponível no seguinte link para visualização em maior resolução: <https://github.com/sarahjfreitas/tis2/blob/master/docs/Diagrama%20de%20Classes.jpg>



# 

# 3.3. Metodologia

Para coleta de informação foi usado como base principalmente o estudo de soluções similares que são concorrentes no mercado.

O software será disponibilizado na plataforma web sendo publicado em um servidor que será definido no futuro.

O código será feito principalmente em Java com o auxílio do framework Spring. Será usado também componentes do Bootstrap no front end e TypeScript. Além disso PostgreSQL será usado como gerenciador de banco de dados para armazenamento de informações.

Para produzir o código será utilizada a IDE IntelliJ IDEA e a ferramente pgAdmin será usada para facilitar o gerenciamento do banco de dados. Além disso o versionamento do código será feito usando o GIT e será armazenado em um repositório no GitHub.

## 3.4. Layout da tela

Observação: Todos os layouts de tela podem ser acessados no seguinte link para visualização em maior resolução: <https://github.com/sarahjfreitas/tis2/tree/master/docs/Templates%20de%20Telas>

## 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

## 

## 3.4.6 Listar Tarefas

## 

## 3.4.6 Listar Usuários