# UNIVERSIDADE CATÓLICA DE PERNAMBUCO CENTRO DE CIÊNCIAS E TECNOLOGIA CIÊNCIA DA COMPUTAÇÃO

**João Pedro de Melo Cabana**

**Rodrigo Jorge de Santana França**

**Victor Vasconcelos Viana**

**Gabriel Souto Ferreira Pereira**

**PRODOC UNICAP**

**Recife**

**2019**

**João Pedro de Melo Cabana**

**Rodrigo Jorge de Santana França**

**Victor Vasconcelos Viana**

**Gabriel Souto Ferreira Pereira**

**PRODOC UNICAP**

Trabalho acadêmico apresentado a turma de Paradigmas de Linguagens de Programação do curso de Ciência da Computação, como parte dos requisitos necessários à obtenção de nota avaliativa para o primeiro GQ.

**Professor(a):** Clarissa Daisy **Disciplina:** Paradigmas de Linguagens de Programação **Turma:** Turma INF1125

# Recife

# 2019

**Resumo e Objetivo**

Nesse documento será apresentado as especificações do sistema PRODOC UNICAP, onde, para entendimento completado, está contido no mesmo as especificações solicitadas no documento de pre-réquisitos do projeto.

Entre os pré-requisitos se encontra a Justificativa linguagem e do paradigma utilizados, o manual do usuário visando deixar claro o fluxo e todas as funcionalidades atualmente implementadas ao usuário final e o manual do desenvolvedor explicando as ferramentas e configurações arquitetura e estrutura usada no projeto, as regras de negócio e o modelo MER da base de dados. Objetivo da elaboração dessa aplicação é fornecer aos docentes uma forma digital , segura e fácil de monitorar sua progressão na insittuição podendo assim além de registrar novas atividades, ter um espaço digital onde tudo será armazenado, possibilitando todo tipo de análise futura da insituição para com seus docentes. O sistema oferece aos docentes a possibilidade de registrar suas atividades e requerir suas pontuações para possíveis promoções. Para verificar essas requisições de pontuação é disponibilizado um módulo administrativo que será responsável por fazer a revisão e a validação dessas requisições feitas pelos docentes. O sistema ainda está limitado a sua forma mais básica de uso, melhorias serão realizadas para as próximas versões.

O projeto pode ser encontrado e baixado no seguinte repositório do github:

**https://github.com/rodrigorjsf/ProdocUnicapProject**

# Lista de ilustrações

# Figura 1 – Tela Inicial .................................................................................7

# Figura 2 – Login Docente ..........................................................................8

# Figura 3 – Sucesso de login.......................................................................8

# Figura 4 – Erro de login...............................................................................9

# Figura 5 – Tela docente...............................................................................9

# Figura 6 – Atualizar dados docente.........................................................10

# Figura 7 – Lista de atividades do docente..............................................10

# Figura 8 – Solicitação de avaliação de atividade...................................11

# Figura 9 – Login Administrador...............................................................11

# Figura 10 – Tela administrador.................................................................12

# Figura 11 – Tela Atividades Adiministrador............................................12

# Figura 12 – Tela Professor Adiministrador.............................................13

# Figura 13 – Atualizar atividade.................................................................13

# Figura 14 – Atualização bem sucedida....................................................13

# Figura 15 – Cadastrar professor..............................................................14

# Figura 16 – MER.........................................................................................20

# Sumário

# Justificativas ................................................................................................ 6

# Manual do Usuário ....................................................................................... 7

# Manual do Desenvolvedor ........................................................................ 15

# Bibliografia ................................................................................................. 21

# Justificativas

Para elaboração desse projeto foi utilizado o paradigma orientado a objetos com a linguagem Java, onde foi levado em consideração critérios como a grande possibilidade de expansão do projeto, a produtividade, legibilidadade, custo do ponto de vista de tempo de trabalho dos estudantes envolvidos no projeto visto que na instituição UNICAP são ferramentas que se tem mais vivência. A escolha da orientação a objeto se deve por oferecer uma estrutura de classes onde nelas são inseridas tudo que é relevante para mesma, possibilitando assim a coerência, confiabilidade e a manutenabilidade na aplicação, aumentando a produtividade da equipe e qualidade da aplicação. Foi cogitado a utilização do paradigma imperativo, porém a sua estrutura de simples execução de códigos e chamadas e suas responsabilidades não bem definidas, que possivelmente chegaria em uma situação onde ficaria muito complexo para entender, dificultando a manutenção. O java nos oferece um ferramental muito bom para garantir a segurando e o funcionamento da aplicação, apesar de ser verbosa, é de fácil legibilidade. A aplicação tem hospedagem toda centralizada no computador(aplicação e base de dados), sendo utilizado IDEs como eclipse e Intellij para desenvolvimento do código e o DB Browser (SQLite) para operar com a base de dados.

# Manual do Usuário

# PRODOC UNICAP :: Manual do Usuário

## 2.1- Usabilidade (Por tela)

### Início

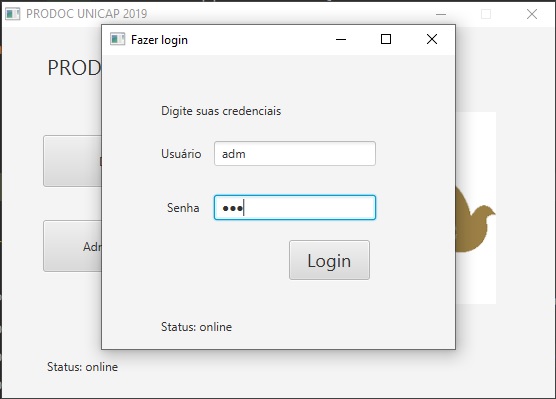
Na tela inicial da aplicação é mostrado ao usuário dois botões: Docente e Administrador [](https://github.com/rodrigorjsf/ProdocUnicapProject/blob/master/docs/manualUsuario/Telas/telaInicial.jpg)

* Botão Docente
  + Botão a ser utilizado pelos docentes UNICAP para realizar seu login no sistema e assim administrar suas atividades, informações da conta e requisições de pontuação em atividades;
* Botão Administrativo
  + Botão a ser utlizado pelo usuário administrador a fim de monitorar os professores cadastrados no sistema e as requisições de pontuação em atividades feitas pelos docentes, oferecendo o poder de avaliar essas atividades e validar as mesmas. Também permite o cadastro de novos docentes.

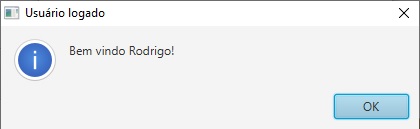
## 2.2- Módulos

### 2.2.1 Docente

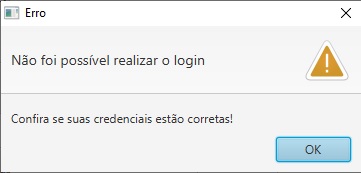
Após clicar no botão Docente, o sistema irá pedir as informações cadastrais (Usuário e Senha) do docente para que ele faça o login no módulo de docentes.

[](https://github.com/rodrigorjsf/ProdocUnicapProject/blob/master/docs/manualUsuario/Telas/senhaDocente.jpg)

Em caso de sucesso no login é exibida a seguinte mensagem

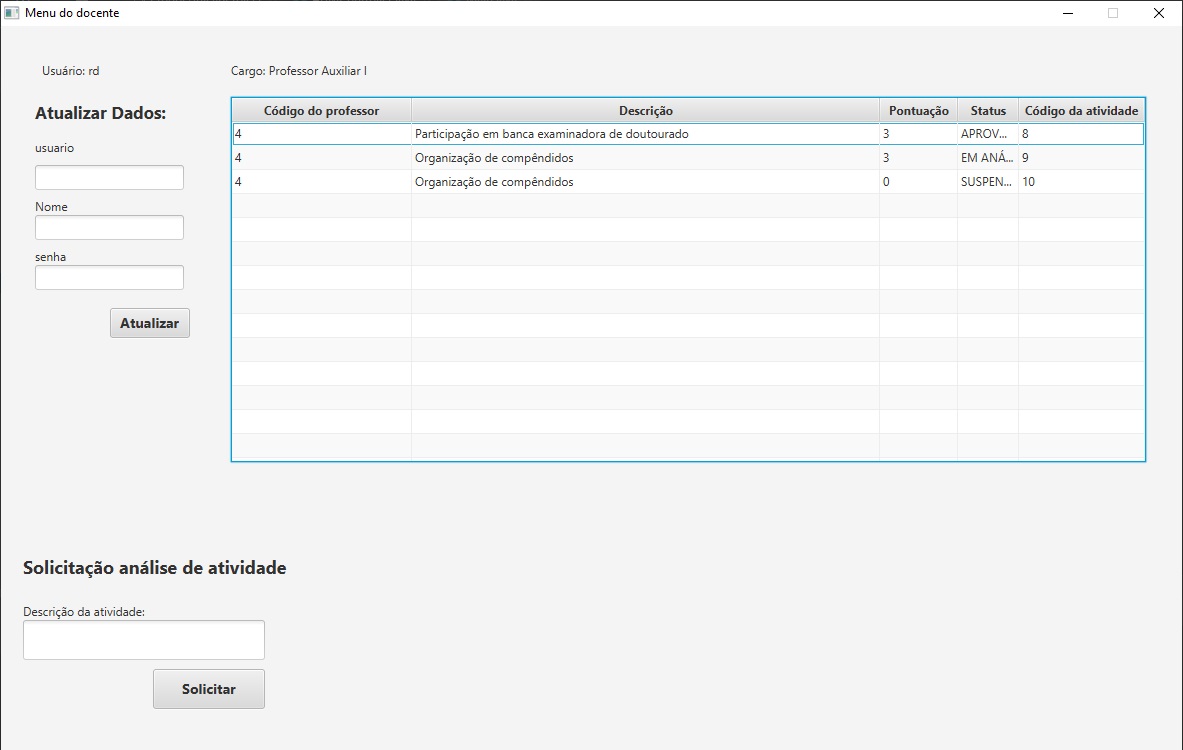
[](https://github.com/rodrigorjsf/ProdocUnicapProject/blob/master/docs/manualUsuario/Telas/loginSucess.jpg)

Em caso de não sucesso no login é exibida a seguinte mensagem

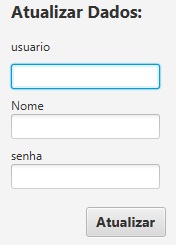
[](https://github.com/rodrigorjsf/ProdocUnicapProject/blob/master/docs/manualUsuario/Telas/loginError.jpg)

**As mesmas regras se aplicam ao módulo ADMINISTRADOR**

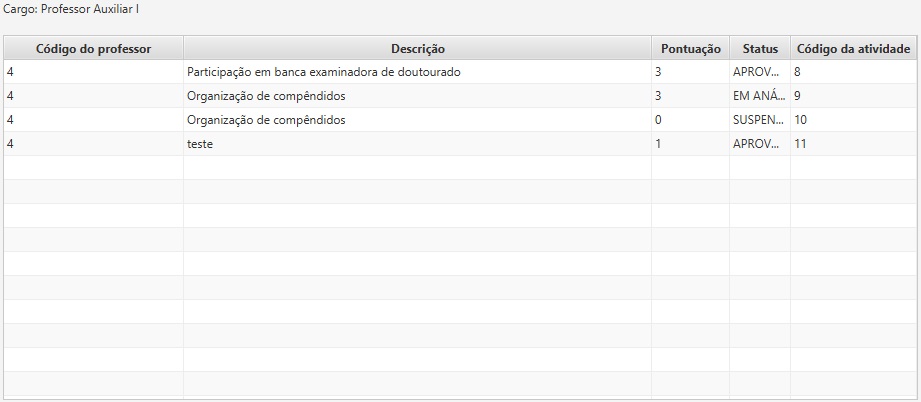
* Feito o Login, na tela do docente irá ser informado seu Usuário e Cargo acima das iterações disponíveis.

[](https://github.com/rodrigorjsf/ProdocUnicapProject/blob/master/docs/manualUsuario/Telas/mainMenu.jpg)

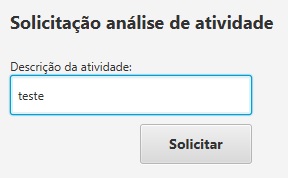
* + Na área "Atualizar Dados" é permitido ao usuário a atualização de seus dados cadastrais (usuário, nome e senha).

[](https://github.com/rodrigorjsf/ProdocUnicapProject/blob/master/docs/manualUsuario/Telas/attDadosDoc.jpg)

* + Ao lado, será exibido uma tabela onde mostra todas as requisições de atividades feitas pelo docente, onde nessa tabela é informado para cada requisição o código do professor, descrição, pontuação, status e código da atividade.

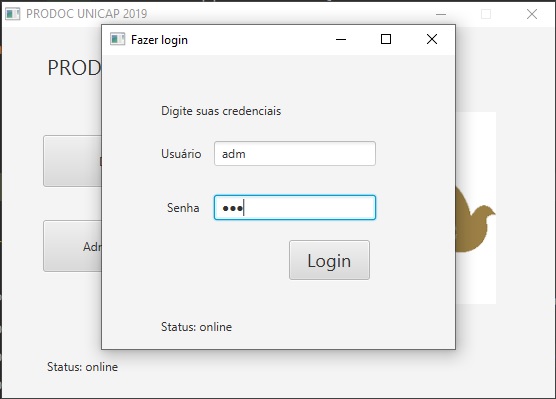
[](https://github.com/rodrigorjsf/ProdocUnicapProject/blob/master/docs/manualUsuario/Telas/docenteAtividades.jpg)

* + No canto inferior esquerdo da tela, na área "Solicitação análise de atividade" é possível fazer uma requisição de avaliação de uma atividade a ser pontuada.

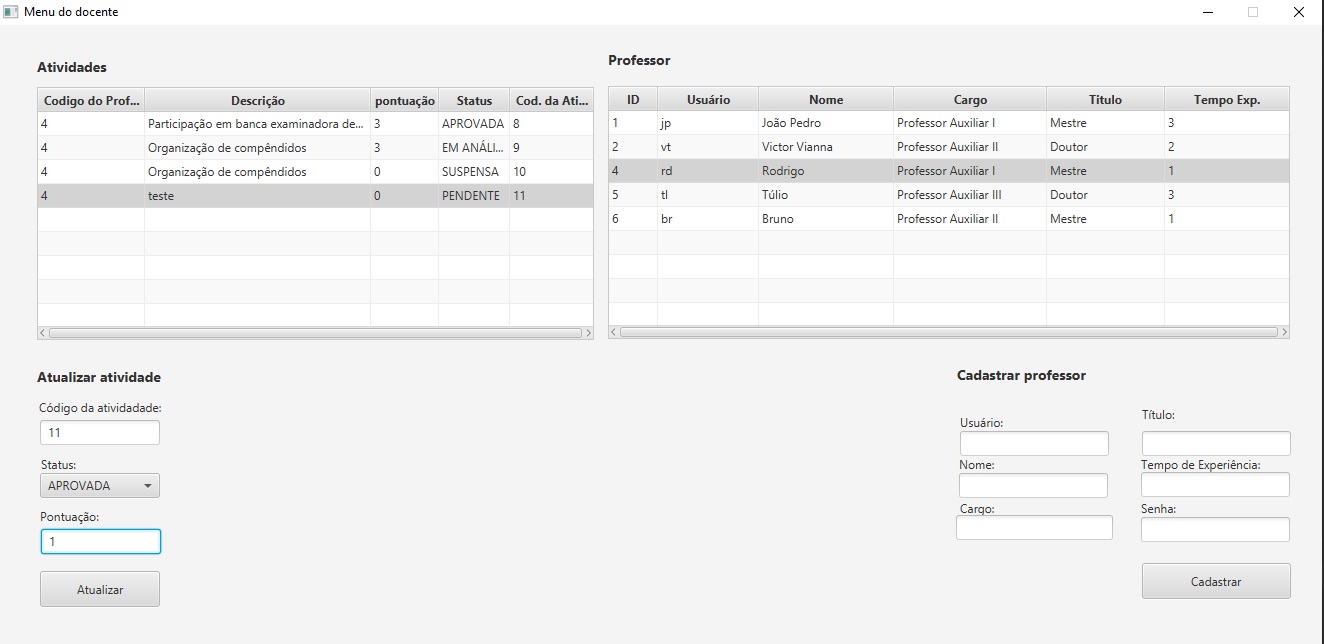
[](https://github.com/rodrigorjsf/ProdocUnicapProject/blob/master/docs/manualUsuario/Telas/novaAtividade.jpg)

### 2.2.2 Administrador

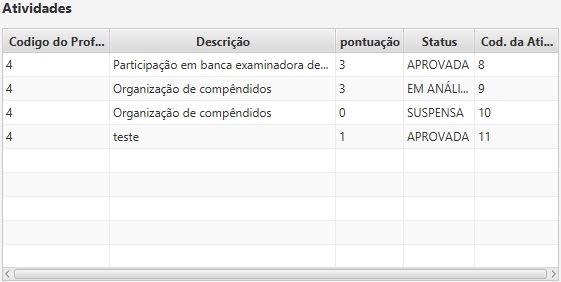
Após clicar no botão Administrador, o sistema irá pedir as informações cadastrais (Usuário e Senha) do usuário Administrador para que ele faça o login no módulo Administrador.

[](https://github.com/rodrigorjsf/ProdocUnicapProject/blob/master/docs/manualUsuario/Telas/senhaDocente.jpg)

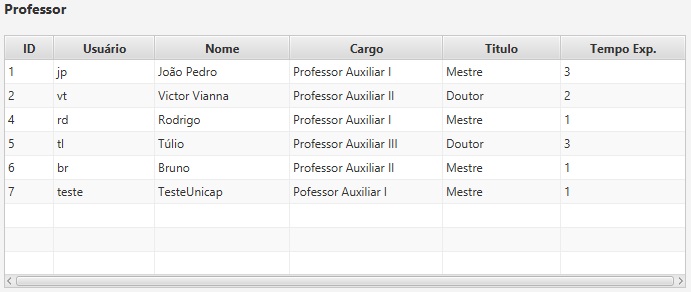
* Feito o Login, na tela do administrador irá ser mostrado telas para gerenciamento da aplicação.



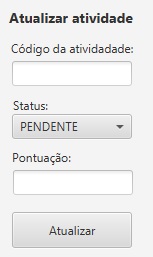
* + Na área Atividades será informado todas as Atividades, independente de status, de um docente selecionado. Entre as informações estão: Código do Professor, Descrição, Pontuação, Status e Código da atividade.

[](https://github.com/rodrigorjsf/ProdocUnicapProject/blob/master/docs/manualUsuario/Telas/atividades.jpg)

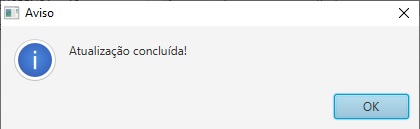
* + Na área Professor será informado todos os docentes cadastrados no sistema da aplicação. Entre as informações estão: ID, Usuário, Nome, Cargo, Título e tempo de Experiência.

[](https://github.com/rodrigorjsf/ProdocUnicapProject/blob/master/docs/manualUsuario/Telas/professor.jpg)

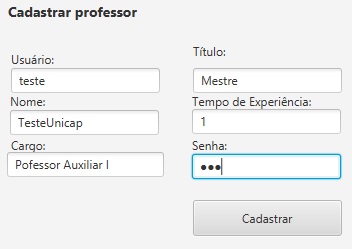
* + No canto inferior esquerdo, na área "Atualizar atividade" é possível atualizar o Status e Pontuação de uma atividade selecionada através de seu código.

[](https://github.com/rodrigorjsf/ProdocUnicapProject/blob/master/docs/manualUsuario/Telas/attAtividade.jpg)

É exibido uma mensagem de sucesso caso a validação ocorra.

[](https://github.com/rodrigorjsf/ProdocUnicapProject/blob/master/docs/manualUsuario/Telas/attRequisicao.jpg)

* + No canto inferior direito, na área "Cadastrar professor" é possível cadastrar um novo docente ao sistema informando seu Usuário, Nome, Cargo, Título, Tempo de Experiência e Senha.

[](https://github.com/rodrigorjsf/ProdocUnicapProject/blob/master/docs/manualUsuario/Telas/newProfessor.jpg)

# Manual do Desenvolvedor

Projeto de desenvolvimento da aplicação **PRODOC UNICAP**. Este módulo é implementado em linguagem de programação Java.

Este projeto usa as seguintes tecnologias:

* [Java 8](https://www.oracle.com/technetwork/pt/java/javase/downloads/jdk8-downloads-2133151.html)
* [SQLite](https://www.sqlite.org/)
* [j forms]

## 3.1- Ferramentas e Configurações

### 3.1.1- Eclipse

**Instalação**

* Baixar e descompactar um dos pacotes (Win ou Linux 64-bit) da versão Eclipse IDE for Java Developers.

OU

* Baixar e Instalar o Intellij

**Importação do Projeto (Git)**

Após abrir o Eclipse, importar projeto do Git:

1. File -> Import... -> Projects from Git -> Clone URI
2. URI: git@github.com:rodrigorjsf/ProdocUnicapProject.git -> Next
3. Escolher branch master
4. Local destination: pasta-do-workspace
5. Import as general project -> Next -> Finish

**Configuração do Projeto**

Para Intellij

1. Adicionar o arquivo sqlite-jdbc-3.27.2.1.jar na pasta src -> dbUtil
2. Clicar com botão direito no projeto -> File -> Project\_Structure -> Libraries...
3. Apertar no + e selecionar a pasta dbUtil
4. OK

**Para baixar** o JAR do SQLite ir na pasta docs, baixar o .ZIP e descompactar na pasta dbUtil.

Incluir Run Configurations:

**Java Application** (para rodar a aplicação localmente)

* New launch configuration:
  + Main
    - Name: prodoc-unicap
    - Project: selecionar-nome-do-projeto
    - Main class: Main.Main

### 3.1.2- DB Browser (SQLite)

Ferramenta de gerenciamento de BDs SQLite.

## 3.2- Arquitetura do Sistema

Este projeto segue um padrão arquitetural MVC onde é que um padrão de arquitetura de software, separando sua aplicação em 3 camadas. A camada de interação do usuário(view), a camada de manipulação dos dados(model) e a camada de controle(controller).Para criação da view foi utilizado o J Forms utiliza FXML para criação da exibição de dados. A manipulação de dados e regras de negócio é através do Java seguindo o paradigma da orientação a objetos, é responsável pela leitura e escrita de dados, e também de suas validações. Foi desenvolvida de maneira centralizada com aplicação local, utilizando base de dados SQLite local.

A organização e significado de cada um dos pacotes do projeto segue abaixo:

src

├── ADM -> classe e FXML do adiministrador

├── Atividades -> classe das atividades

├── dbUtil -> classe e arquivo jar da camada de persistência de dados

├── DOC -> classe e FXML de docentes

├── Login -> classe e FXML de controle de login

├── Main -> classe e FXML da inicialização da aplicação

├── Model -> classe e FXML dos models

**Para cada pacote:**

* ADM
  + ADM.fxml
  + ADMController
* Atividades
  + AtividadesDAO
* dbUtil
  + dbConection
* DOC
  + DocenteCOntroller
  + DocenteView.fxml
  + usuarioDAO
* Login
  + LoginController
  + LoginDAO
  + LoginView.fxml
* Main
  + Main
  + MainController
  + MainDAO
  + MainView.fxml
* Model
  + AtividadeModel
  + DocenteModel

## 3.3- Regras de negócio

### 3.3.1 Entidades com Manutenção (CRUD)

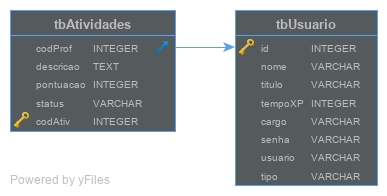
#### **Usuário (tabela tbUsuario)**

* Inclusão
  + Não permitir a inclusão de um novo registro cujo código já se encontra cadastrado;
  + Preencher o campo de nome (nome) com o nome atual do usuário;
  + Preencher o campo de título (titulo) com o título atual do usuário;
  + Preencher o campo de tempo de experiência (tempoXP) com o tempo de exercício na instituição do usuário;
  + Preencher o campo de cargo (cargo) com o cargo atual do usuário;
  + Preencher o campo de senha (senha) com a senha do usuário;
  + Preencher o campo de usuário (usuario) com o nome de usuário que será utilizado para identificar o mesmo;
  + Preencher o campo de tipo de usuário (tipo) com o tipo de usuário: Docente ou Administrador;
* Alteração
  + Não permitir a alteração de um registro cujo código não está cadastrado;
  + Permitir atualização do campo nome (name).
  + Permitir atualização do campo usuário (usuario).
  + Permitir atualização do campo senha (senha).
* Exclusão
  + Não permitir a remoção de um registro;

#### **Atividade (tabela tbAtividades)**

* Inclusão
  + Não permitir a inclusão de um novo registro cujo código já se encontra cadastrado;
  + Gera um código de Atividade auto-increment (codAtiv);
  + Preencher o campo descrição (descricao) com a descrição da atividade cadastrada;
  + Preencher o campo pontuação (pontuacao) com um valor da pontuação da atividade;
  + Preencher o campo status (status) com o status da atividade;
  + Associar um Professor à atividade através da chave estrangeira (codProf);
* Alteração
  + Não permitir a alteração de um registro cujo identificador não está cadastrado;
  + Alteração é feita através da aprovação de uma atividade por um adiministrador, essa alteração é feita nos campos (pontuacao) e (status);
* Exclusão
  + Não permitir a remoção de um registro;

## - MER



# Bibliografia

https://tableless.com.br/mvc-afinal-e-o-que/