

CRUD - App Pontos Farm

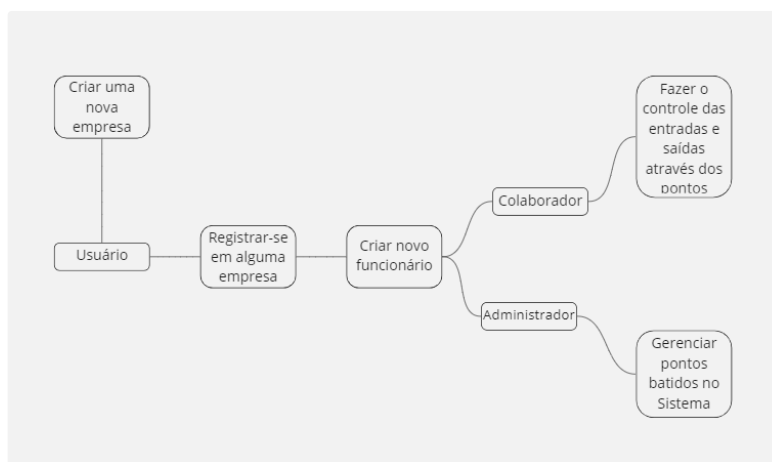
- [Objetivo](#)
- [Regras de negócio](#)
 - [Como funciona](#)
 - [Processos do negócio](#)
- [Tecnologias](#)
 - [Backend](#)
- [Banco de dados](#)
 - [Diagrama](#)
 - [Modelo Conceitual](#)
 - [Script SQL](#)
- [Desenhos](#)
 - [Telas wireframes - Jornada](#)
- [Aplicações](#)
 - [Spring](#)
 - [Gerenciador](#)
 - [Dependencies](#)
 - [Plugins](#)

Objetivo [↗](#)

- Promover e facilitar o gerenciamento de pontos para empresa e para o funcionário
- Aplicação projetada para ser centrada na utilização e necessidade do usuário
- Possibilitar ao usuário o registro em duas diferentes empresas utilizando o mesmo cadastro

Regras de negócio [↗](#)

Como funciona [↗](#)



- Usuário novo pode criar uma empresa ou se registrar em uma nova empresa como funcionário
- ao se registrar como funcionário deve-se escolher o tipo, sendo:
 - administrador → quem gerencia e cuida dos pontos

- colaborador → faz o controle de entradas e saídas através dos pontos

Processos do negócio [↗](#)

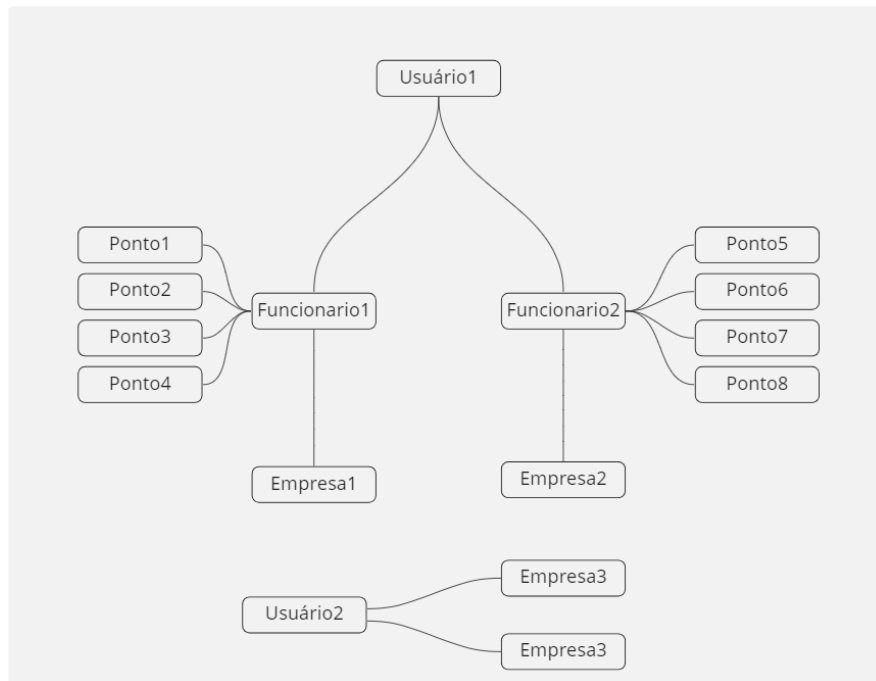


Ilustração das relações que um usuário pode possuir de acordo com as regras de negócio da aplicação

- Usuário1 atribuído a dois funcionários que pertencem a diferentes empresas
- Usuário2 atribuído à duas diferentes empresas
- Funcionários 1 e 2, independentes entre si, batendo diferentes pontos

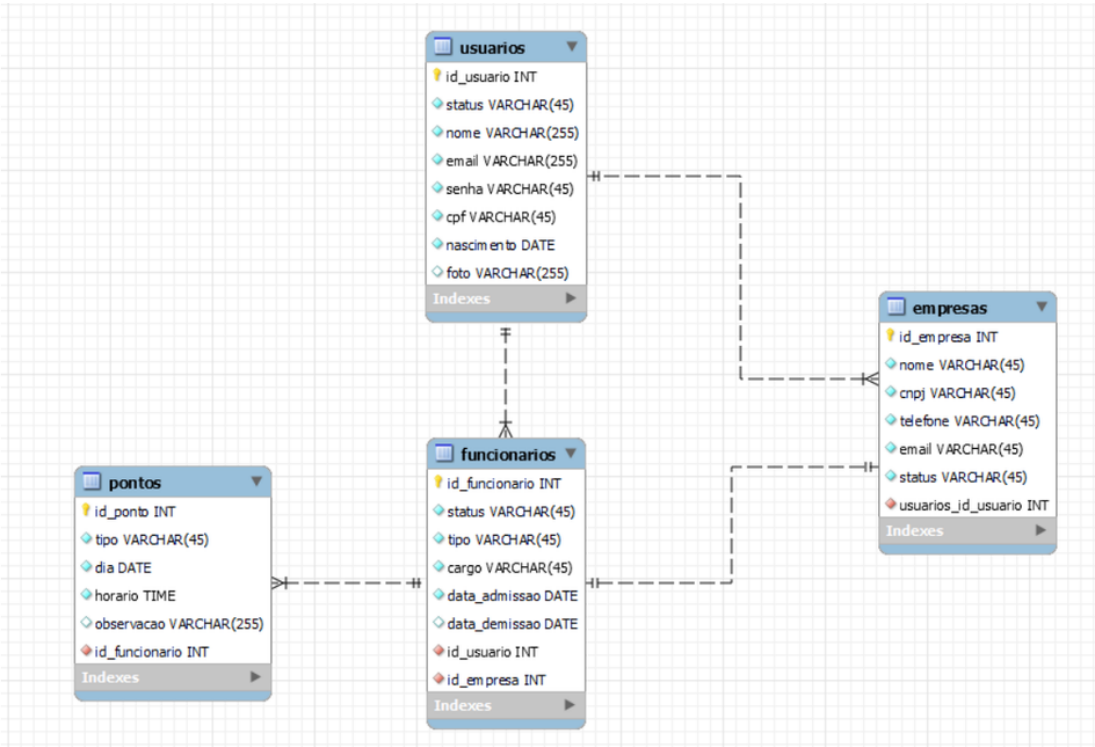
Tecnologias [↗](#)

Backend [↗](#)

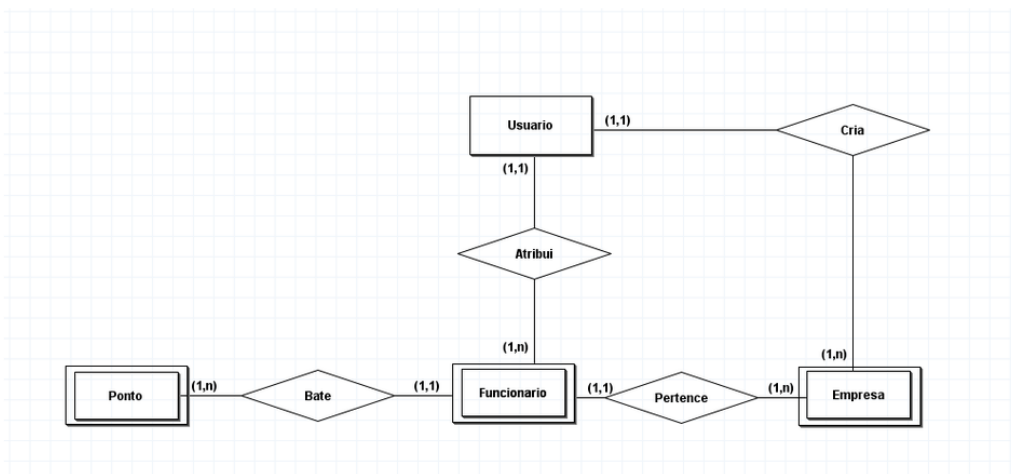
- SpringBoot
- Mysql

Banco de dados

Diagrama



Modelo Conceitual [↗](#)



Script SQL [↗](#)

```
1 -- MySQL Workbench Forward Engineering
2
3 SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
4 SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0;
5 SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO_DATE,ERF
6
7 -----
8 -- Schema mydb
9 -----
10
11 -----
12 -- Schema mydb
13 -----
```

```

14 CREATE SCHEMA IF NOT EXISTS `mydb` DEFAULT CHARACTER SET utf8 ;
15 USE `mydb` ;
16
17 -----
18 -- Table `mydb`.`usuarios`
19 -----
20 CREATE TABLE IF NOT EXISTS `mydb`.`usuarios` (
21   `id_usuario` INT NOT NULL AUTO_INCREMENT,
22   `status` VARCHAR(45) NOT NULL,
23   `nome` VARCHAR(255) NOT NULL,
24   `email` VARCHAR(255) NOT NULL,
25   `senha` VARCHAR(45) NOT NULL,
26   `cpf` VARCHAR(45) NOT NULL,
27   `nascimento` DATE NOT NULL,
28   `foto` VARCHAR(255) NULL,
29   PRIMARY KEY (`id_usuario`))
30 ENGINE = InnoDB;
31
32
33 -----
34 -- Table `mydb`.`empresas`
35 -----
36 CREATE TABLE IF NOT EXISTS `mydb`.`empresas` (
37   `id_empresa` INT NOT NULL AUTO_INCREMENT,
38   `nome` VARCHAR(45) NOT NULL,
39   `cnpj` VARCHAR(45) NOT NULL,
40   `telefone` VARCHAR(45) NOT NULL,
41   `email` VARCHAR(45) NOT NULL,
42   `status` VARCHAR(45) NOT NULL,
43   `id_usuario` INT NOT NULL,
44   PRIMARY KEY (`id_empresa`),
45   INDEX `fk_empresas_usuarios1_idx` (`id_usuario` ASC) VISIBLE,
46   CONSTRAINT `fk_empresas_usuarios1`
47     FOREIGN KEY (`id_usuario`)
48     REFERENCES `mydb`.`usuarios` (`id_usuario`)
49     ON DELETE NO ACTION
50     ON UPDATE NO ACTION)
51 ENGINE = InnoDB;
52
53
54 -----
55 -- Table `mydb`.`funcionarios`
56 -----
57 CREATE TABLE IF NOT EXISTS `mydb`.`funcionarios` (
58   `id_funcionario` INT NOT NULL AUTO_INCREMENT,
59   `status` VARCHAR(45) NOT NULL,
60   `tipo` VARCHAR(45) NOT NULL,
61   `cargo` VARCHAR(45) NOT NULL,
62   `data_admissao` DATE NOT NULL,
63   `data_demissao` DATE NULL,
64   `id_usuario` INT NOT NULL,
65   `id_empresa` INT NOT NULL,
66   PRIMARY KEY (`id_funcionario`),
67   INDEX `fk_funcionarios_usuarios_idx` (`id_usuario` ASC) VISIBLE,
68   INDEX `fk_funcionarios_empresas1_idx` (`id_empresa` ASC) VISIBLE,
69   CONSTRAINT `fk_funcionarios_usuarios`
70     FOREIGN KEY (`id_usuario`)
71     REFERENCES `mydb`.`usuarios` (`id_usuario`)

```

```

72     ON DELETE NO ACTION
73     ON UPDATE NO ACTION,
74     CONSTRAINT `fk_funcionarios_empresas1`
75     FOREIGN KEY (`id_empresa`)
76     REFERENCES `mydb`.`empresas` (`id_empresa`)
77     ON DELETE NO ACTION
78     ON UPDATE NO ACTION)
79 ENGINE = InnoDB;
80
81
82 -----
83 -- Table `mydb`.`pontos`
84 -----
85 CREATE TABLE IF NOT EXISTS `mydb`.`pontos` (
86   `id_ponto` INT NOT NULL AUTO_INCREMENT,
87   `tipo` VARCHAR(45) NOT NULL,
88   `dia` DATE NOT NULL,
89   `horario` TIME NOT NULL,
90   `observacao` VARCHAR(255) NULL,
91   `id_funcionario` INT NOT NULL,
92   PRIMARY KEY (`id_ponto`),
93   INDEX `fk_pontos_funcionarios1_idx` (`id_funcionario` ASC) VISIBLE,
94   CONSTRAINT `fk_pontos_funcionarios1`
95     FOREIGN KEY (`id_funcionario`)
96     REFERENCES `mydb`.`funcionarios` (`id_funcionario`)
97     ON DELETE NO ACTION
98     ON UPDATE NO ACTION)
99 ENGINE = InnoDB;
100
101
102 SET SQL_MODE=@OLD_SQL_MODE;
103 SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;
104 SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;
105

```

Desenhos [↗](#)

Telas wireframes - Jornada [↗](#)

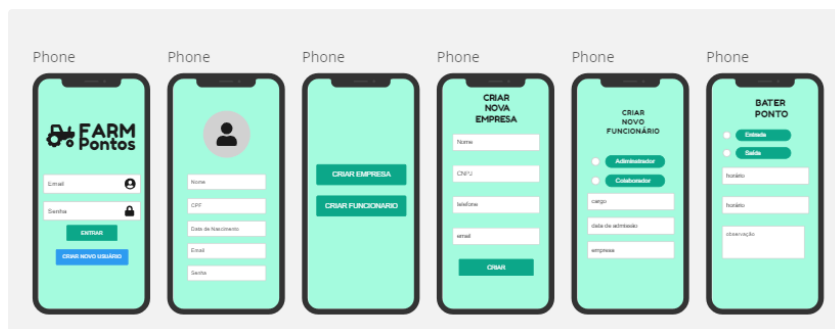


Ilustração da idealização da jornada do usuário dentro de um app mobile utilizando a aplicação desenvolvida [FARM PONTOS]

→ Tela de Login

- Cadastro de novo usuário
- Opção de criar nova empresa ou novo funcionário
- Criação de nova empresa
- Criação de novo funcionário
- Tela destinada a ação de bater o ponto

Aplicações [↗](#)

Spring [↗](#)

Gerenciador [↗](#)

- Maven

Dependencies [↗](#)

- lombok
- mysql
- data-jdbc
- data-jpa
- web
- flyway
- jakarta
- openApi

Plugins [↗](#)

- OpenAPI Generator Plugin
- Surefire
- Flyway