



ALUNO: CARLOS EDUARDO MONTOYA MONTOYA

PROFESSOR: CARLOS HENRIQUE VERISSIMO PEREIRA

CURSO: PROGRAMAÇÃO ORIENTADA A OBJETOS

SEMESTRE: 3

TURMA: A

2023

1. Descrição do Problema

Uma empresa administra um conjunto de propriedades, que administra como proprietário. Cada propriedade pode ser um apartamento, casa, local (instalações comerciais, escritórios, etc.), um andar ou um edifício que por sua vez possui andares e instalações. Como o número de propriedades que a empresa administra não é um número fixo, a aplicação deve permitir tanto introduzir novos imóveis, bem como descadastrá-los, modificá-los e consultá-los. Da mesma forma, só porque uma empresa gere um edifício específico não significa que ela gerencie todos os seus apartamentos e instalações, portanto o aplicativo também deve permitir a entrada novos apartamentos ou instalações, cancelá-los, modificá-los e fazer perguntas sobre eles. Qualquer pessoa que possua folha de pagamento, fiança bancária, contrato de trabalho ou vem endossado por outra pessoa, você pode alugar o prédio inteiro ou alguns andares ou instalações que ainda não estejam alugadas e, posteriormente, desalugá-las. Portanto, eles devem poder registrar-se, se forem novos inquilinos, com os dados correspondentes (nome, CPF, idade, sexo, ...), poder modificá-los, excluí-los, consultá-los, etc. O aplicativo oferece acesso web para que o inquilino possa modificar ou consultar seus dados, mas não cancele a inscrição ou registre-se. Para o desempenho de qualquer um destes operações, é necessária a identificação do inquilino.

2. Lista de Requisitos

2.1. Requerimentos funcionais.

- RF1: Cadastro dos usuários no sistema
- RF2: Pesquisa de propriedade filtrados.
- RF3: Conexão com proprietário do imóvel.
- RF5: Gestão das propriedades.
- RF6: Controlar o progresso de estado das propriedades.
- RF7: CRUD das propriedades, usuários, proprietário.
- RF8: Organizador de estatísticas das propriedades.
- RF9: Calcular média de contratos com sucesso das propriedades.

2.2. Requerimentos não funcionais.

- RNF1: Deverá ser desenvolvida um back-end robusto para dar sustentação (negócio)
- RNF2: Design de ícones familiares com seu uso, para reconhecimento imediato;
- RNF3: Terá uma interface de fácil uso tanto para usuários experientes como para leigos.
- RNF4: O sistema deverá se conectar com sites externos.
- RNF5: O desenvolvimento do software deverá prever entregas parciais ao cliente do produto de software.
- RNF6: O sistema será desenvolvido em Java.
- RNF7: Uso de design responsivo durante o desenvolvimento da aplicação;
- RNF8: O sistema deverá estar ativo 99,9% do tempo.
- RNF9: O sistema deverá fornecer confidencialidade dos dados sensíveis do usuário.
- RNF10: Deverá fornecer diferentes formas de login.
- RNF11: Deverá proteger os dados do usuário.
- RNF12: O sistema deve ser capaz de lidar com o número necessário de usuários sem qualquer degradação no desempenho.

3. Casos de Uso

3.1 Diagrama de Caso de Uso Geral

