

Projeto - Sistemas Web

Guilherme Jannotti Arantes¹, Helton Ribeiro Lustosa¹, Leonardo Souza¹

¹Instituto de Ciências Exatas e Aplicadas – Universidade Federal de Ouro Preto (UFOP)
Caixa Postal 24 – 35.930-970 – João Monlevade – MG – Brazil

{guilherme.jannotti}@gmail.com, {heltonrlustosa}@gmail.com, {leosouza3102}@gmail.com

Abstract. *This work has as main purpose to put into practice theoretical concepts learned during the graduation. This work includes technical and practical concepts in various disciplines, including: Software Engineering, Database, Configuration Management and Software Engineering, Computer Programming, Web Systems, Entrepreneurship, among others. The idea is to propose and develop a web system that can solve a real problem. Our group chose to design theme, proposing a management system for Republics, which will offer all the necessary structure to manage various needs arising from a republic (student, workers and / or community housing) as financial control and asset tracking.*

Resumo. *Este trabalho tem como principal finalidade colocar em prática conceitos teóricos aprendidos ao decorrer da graduação. O presente trabalho engloba conceitos técnicos e práticos de diversas disciplinas, entre elas: Engenharia de Software, Banco de Dados, Gerência de Configuração e Engenharia de Software, Programação de Computadores, Sistemas Web, Empreendedorismo, entre outras. A ideia é propor e elaborar um sistema web que possa solucionar algum problema real. O nosso grupo escolheu como tema do projeto, propor um Sistema de Gerenciamento para Repúblicas, que irá oferecer toda estrutura necessária para gerir várias necessidades advindas de uma república (estudantil, de trabalhadores e/ou moradia comunitária), como controle financeiro e controle de patrimônio.*

1. Introdução

2. Descrição do Problema

O nicho de mercado que desejamos atuar: Repúblicas Universitárias e de forma generalizada, todos os tipos de moradias comunitárias, cujo único requisito é ter ao menos dois moradores.

Problema encontrado: é relacionado à falta de um adequado gerenciamento financeiro e patrimonial das repúblicas estudantis, trabalhistas e moradias comunitárias. O que muitas vezes acarreta no fim (“falência”) da república.

3. Objetivo Geral

O Objetivo Geral é criar um sistema que gerencia todas as demandas de uma moradia comunitária, uma vez que ficou evidente que, para que a república prospere, é necessário uma gestão sólida por parte dos moradores, para que todas as informações importantes (contas, boletos, notas fiscais, comprovantes, documentos, etc.) fiquem armazenadas consistentemente e de forma segura, em um só local, que permita a análise dos dados contidos (gerado por relatórios), para que se possam planejar medidas de prevenção, manutenção, aquisição de bens e serviços, entre outras tomadas de decisões necessárias para a manutenção e desenvolvimento de repúblicas e moradias comunitárias.

3.1. Objetivos Específicos

Módulo de Controle de Usuários:

Módulo Financeiro:

Módulo Patrimonial:

4. Escopo da Aplicação

(descrever a área de domínio da aplicação, incluindo as limitações do projeto.)

5. Descrição do Produto

- Tipo de sistema a ser produzido:

O sistema web proposto visa controlar diversas questões provenientes de uma república estudantil, trabalhista e/ou moradia comunitária, a ideia é tratar questões como o gerenciamento financeiro e controle de patrimônio. No âmbito do controle financeiro, a proposta é criar uma plataforma que permita a inserção, armazenamento e controle das contas fixas e variadas, das notas fiscais individuais (e marcar quais moradores irão participar do rateio), controle de poupança da casa e reserva para urgências, entre outras

questões financeiras que envolvem todos os moradores. No âmbito do controle de patrimônio, a ideia é que a plataforma permita a inserção, armazenamento e controle de todos os bens materiais pertencentes à república, além de associar um valor de manutenção ou troca para cada bem material (opcional) dado uma data pré-estipulada, para que as repúblicas possam se planejar no sentido de fazer manutenções e/ou aquisições de novos bens.

- Plataforma e principais tecnologias:
 - A princípio, estamos utilizando as seguintes tecnologias:
1. HTML
 2. CSS
 3. BOOTSTRAP
 4. JAVASCRIPT
 5. PHP
 6. MYSQL
 7. APACHE
 8. WAMP (Windows Apache MySQL PHP)
 9. Brackets
 10. GitHub

Porém, devido algumas análises prévias, chegamos à conclusão que a melhor linguagem a ser utilizada para este projeto, devida sua complexidade, é o Ruby on Rails. Porém, dado que iremos aprender apenas no final do semestre, não temos tempo hábil para desenvolver utilizando essa tecnologia.

Pretendemos futuramente programar aplicativos mobile para Android e IOS que sincronizem com o sistema Web, de tal maneira que o usuário possa utilizar o sistema tanto pelo seu smartphone quanto pela web.

6. Casos de Uso

Listamos alguns casos de Uso com seus respectivos:

- *Nome.*
- *Nível.*
- *Ator Primário.*
- *Interessados.*
- *Pré-condições.*
- *Garantias de Sucesso.*
- *Cenário de Sucesso Principal.*
- *Extensões.*
- *Requisitos Especiais.*
- *Variação de Dados e Tecnologias.*
- *Frequência de Ocorrência.*
- *Outros.*

1º Caso de uso:

Nome: Inserir República.

Nível: Objetivo de usuário admin.

Ator primário: Fundador da república (admin).

Interessados:

- Fundador: Inserir uma nova república para ser gerenciada.
- Usuários: Participar de uma nova forma de gerenciar área financeira e patrimonial de republicas.
- Sistema: Registrar uma nova república e auxiliar no seu controle financeiro e patrimonial, facilitando a realização de tarefas complicadas como divisão de contas mensais.

Pré-condições:

- O grupo de pessoas deve ter um representante principal.
- O nome deve ser diferente do nome das outras repúblicas.
- O sistema deve verificar se já existe alguma república com o nome inserido.
- O fundador deve aceitar os termos de compromisso.

Garantias de sucesso:

- A república pode ser criada caso não exista nenhuma outra com o mesmo nome.

Cenário de sucesso principal: Não existir outra república com o mesmo nome.

Extensões: Não há.

Requisitos especiais: Utilização de letras e números apenas.

Variações de dados e tecnologia:

- A comparação dos nomes é feita através de uma comunicação com o banco de dados solicitando a existência do nome digitado na hora da criação.
- Serão utilizadas as linguagens HTML juntamente com CSS e Java Script para exibição da tela, PHP para comunicação com banco de dados e MySql para definição e manipulação do Banco de Dados.

Frequência de ocorrência: Ocorrerá somente quando for solicitada a criação de uma nova república.

Outros: Esta funcionalidade estará disponível na página inicial.

2º Caso de uso

Nome: Remover uma república.

Nível: Objetivo de usuário admin.

Ator primário: Fundador da república (admin).

Interessados:

- Fundador e usuários: Deixar de participar do software de gerenciamento de repúblicas.
- Sistema: Remover a república de sua base de dados.

Pré-condições:

- O grupo de pessoas deve estar de acordo com o representante principal.
- O fundador deve estar logado no sistema.
- O nome da república deve estar previamente inserido na base de dados.
- O sistema deve verificar se a república existe na base de dados.

Garantias de sucesso:

- A república pode ser removida caso seu nome esteja na base de dados da WeRep e o fundador esteja logado no sistema.

Cenário de sucesso principal: A república se encontra na base de dados e o fundador está logado no sistema.

Extensões: Durante um período de 30 dias é possível resgatar a república e trazê-la de volta à ativa, porém o fundador que deve realizar esta operação.

Requisitos especiais: O fundador deve estar logado e ele que deve remover a república.

Variações de dados e tecnologia:

- A comparação dos nomes é feita através de uma comunicação com o banco de dados solicitando a existência do nome digitado.

Frequência de ocorrência: Ocorrerá somente quando for solicitada a remoção de uma

república.

Outros: Esta funcionalidade estará disponível apenas na tela do fundador.

3º Caso de uso

Nome: Adicionar morador.

Nível: Objetivo de usuário admin.

Ator primário: Fundador da república (admin).

Interessados:

- Fundador: Inserir um novo morador em sua república.
- Morador: Participar de uma nova forma de gerenciar área financeira e patrimonial.
- Sistema: Registrar um novo morador.

Pré-condições:

- O morador não deve estar previamente registrado.
- O morador deverá aceitar os termos de compromisso da república.

Garantias de sucesso:

- O morador pode ser registrado caso ainda não possua a sua identidade na base de dados da república.

Cenário de sucesso principal: Não existir uma pessoa com identidade igual à do morador previamente registrado.

Extensões: Não há.

Requisitos especiais – Utilização de letras e números apenas.

Variações de dados e tecnologia:

- A comparação do campo identidade é feita através de uma comunicação com o banco de dados solicitando a existência de uma pessoa com a mesma identidade.
- Serão utilizadas as linguagens HTML juntamente com CSS e Java Script para exibição da tela, PHP para comunicação com banco de dados e MySql para definição e manipulação do Banco de Dados.

Frequência de ocorrência: Ocorrerá somente quando for solicitada a inserção de um calouro na república.

Outros: Esta funcionalidade estará disponível dentro da página da república. Logo o fundador deve estar logado para realizar a operação.

4º Caso de uso

Nome: Remover morador.

Nível: Objetivo de usuário admin.

Atores primários: Fundador da república ou morador.

Interessados:

- Fundador: Remover um morador de sua república.

- Morador: Deixar de participar da república.
- Sistema: Remover um morador da base de dados de uma república.

Pré-condições:

- O morador deve estar previamente registrado.

Garantias de sucesso:

- O morador pode ser removido caso possua a sua identidade na base de dados da república.

Cenário de sucesso principal: Existir uma pessoa com identidade igual à do morador previamente registrada.

Extensões: Não há.

Requisitos especiais – O fundador deve estar logado e ele que deve remover o morador.

Variações de dados e tecnologia:

- A comparação do campo identidade é feita através de uma comunicação com o banco de dados solicitando a existência de uma pessoa com a mesma identidade.

Frequência de ocorrência: Ocorrerá somente quando for solicitada a remoção de um morador da república.

Outros: Esta funcionalidade estará disponível apenas na tela do fundador.

5º Caso de uso

Nome: Inserir dados sobre a república.

Nível: Objetivo de usuário admin.

Atore primário: Fundador da república.

Interessados:

- Fundador: Adicionar dados referentes a sua república como rua, bairro, cidade, CEP, número e telefone.
- Sistema: Adicionar a base de dados informações sobre a localização da república.

Pré-condições:

- A república deve estar previamente registrada e o fundador logado no sistema.

Garantias de sucesso:

- Todos os campos devem ser preenchidos.

Cenário de sucesso principal: Campos preenchidos.

Extensões: Caso o fundador queira alterar os dados basta digitar novamente o campo em questão que substituirá o antigo dado. Não haverá funcionalidade de remoção destes dados, pois eles podem ser alterados para nulo.

Requisitos especiais: O fundador deve estar logado e ele que deve inserir os dados.

Variações de dados e tecnologia:

- A inserção de dados será feita através de uma comunicação com o banco de dados.

Frequência de ocorrência: Ocorrerá somente quando for solicitada a remoção de um morador da república.

Outros: Esta funcionalidade estará disponível apenas na tela do fundador.

6º Caso de uso

Nome: Inserir patrimônio.

Nível: Objetivo de usuário admin.

Atores primários: Fundador da república.

Interessados:

- Fundador: Adicionar um patrimônio com data de substituição e valor a ser cobrado mensalmente para cobrir tal custo final.
- Sistema: Auxiliar no controle de patrimônios físicos da república.

Pré-condições:

- A república deve estar previamente registrada e o fundador logado no sistema.

Garantias de sucesso:

- O patrimônio foi adicionado a base de dados da república.

Cenário de sucesso principal: O patrimônio foi adicionado a base de dados.

Extensões – Não há.

Requisitos especiais – O fundador deve estar logado e ele que deve inserir os dados.

Variações de dados e tecnologia:

- A inserção de dados será feita através de uma comunicação com o banco de dados.

Frequência de ocorrência: Ocorrerá somente quando for solicitada a inserção de um novo patrimônio.

Outros: Esta funcionalidade estará disponível apenas na tela do fundador.

7º Caso de uso

Nome: Remover patrimônio.

Nível: Objetivo de usuário admin.

Atores primários: Fundador da república.

Interessados:

- Fundador: Remover um patrimônio (Em caso de venda ou descarte).
- Sistema: Auxiliar no controle de patrimônios físicos da república.

Pré-condições:

- A república deve estar previamente registrada, o fundador logado no sistema e o patrimônio previamente inserido.

Garantias de sucesso:

- O patrimônio foi removido da base de dados da república.

Cenário de sucesso principal – O patrimônio foi removido da base de dados.

Extensões – Ao ser removido é adicionado em um histórico o valor acumulado com tal patrimônio.

Requisitos especiais – O fundador deve estar logado e ele que deve inserir os dados.

Variações de dados e tecnologia:

- A remoção de dados será feita através de uma comunicação com o banco de dados.

Frequência de ocorrência: Ocorrerá somente quando for solicitada a remoção de um

novo patrimônio.

Outros: Esta funcionalidade estará disponível apenas na tela do fundador.

8º Caso de uso

Nome: Rateio das despesas da República.

Nível: Sistema.

Atores primários – Sistema.

Interessados:

- Fundador e moradores: Fazer lançamento das notas referentes às respectivas despesas gastas por cada morador, além das despesas fixas.
- Sistema: Auxiliar no controle financeiro de divisão de contas.

Pré-condições:

- O sistema irá fazer o fechamento das contas num dia fixo todo mês, e disponibilizará o valor que cada morador deve pagar..

Garantias de sucesso:

- Ter sido inserido ao mínimo uma despesa.

Cenário de sucesso principal: a precisão e consistência dos dados calculados referente ao rateio das despesas..

Extensões: É possível desfazer as ações.

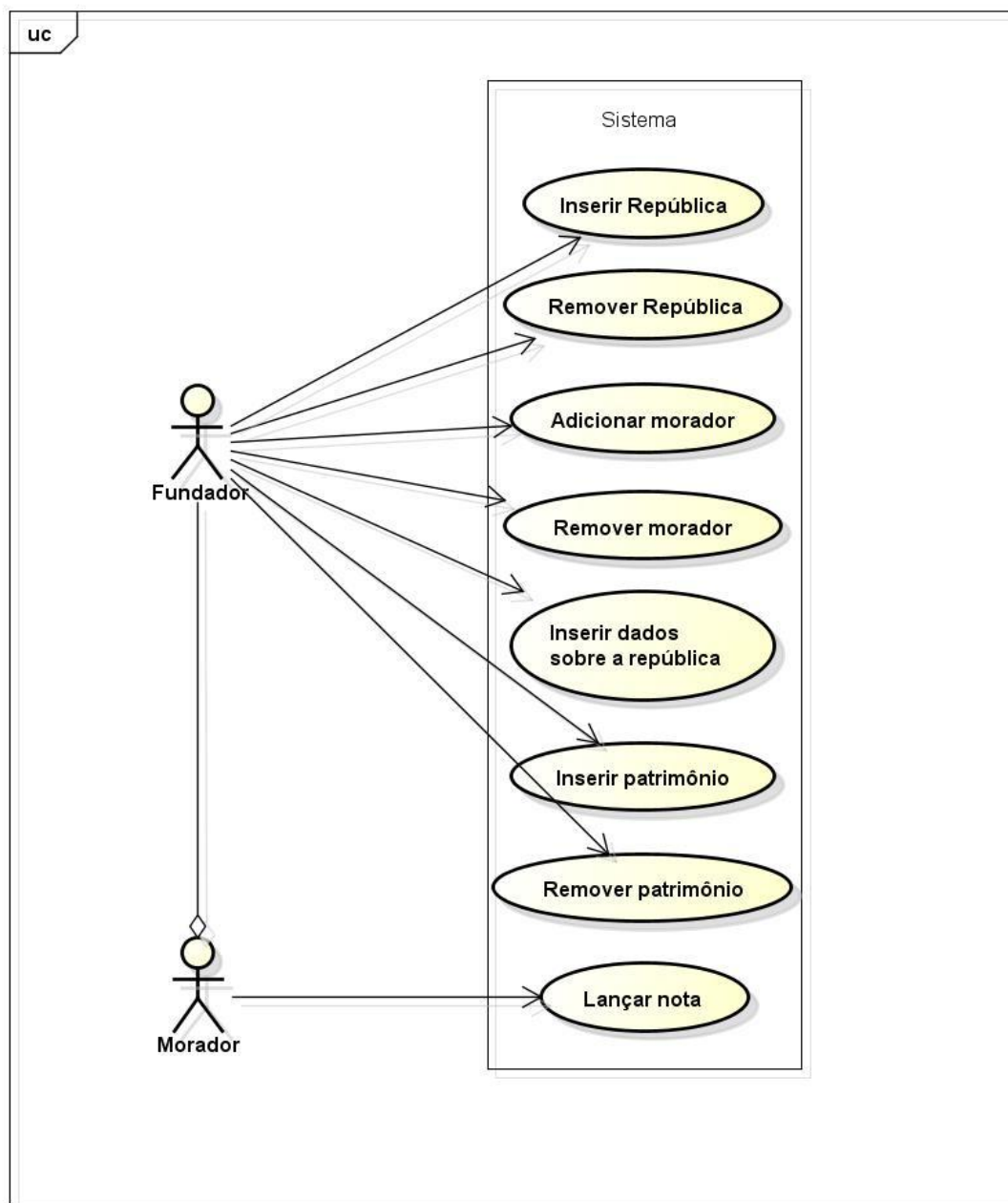
Variações de dados e tecnologia:

Frequência de ocorrência: o fechamento correrá em uma data fixa pré estipulada pelo usuário admin..

Outros:

- O admin poderá alterar ou excluir notas que ele avalia estar fora dos parâmetros estipulados pela república.

Diagrama de Casos de Uso:

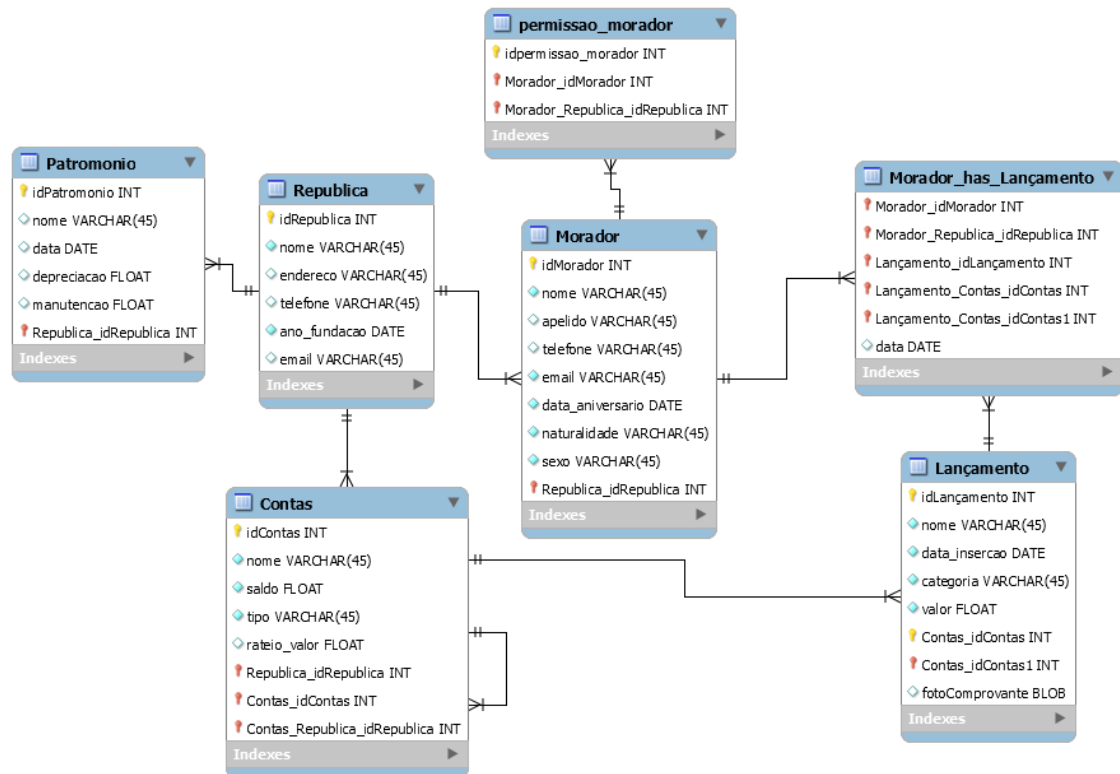


7. Classes

(definição das classes, relacionamentos, pacotes e os diagramas necessários.)

8. Banco de Dados

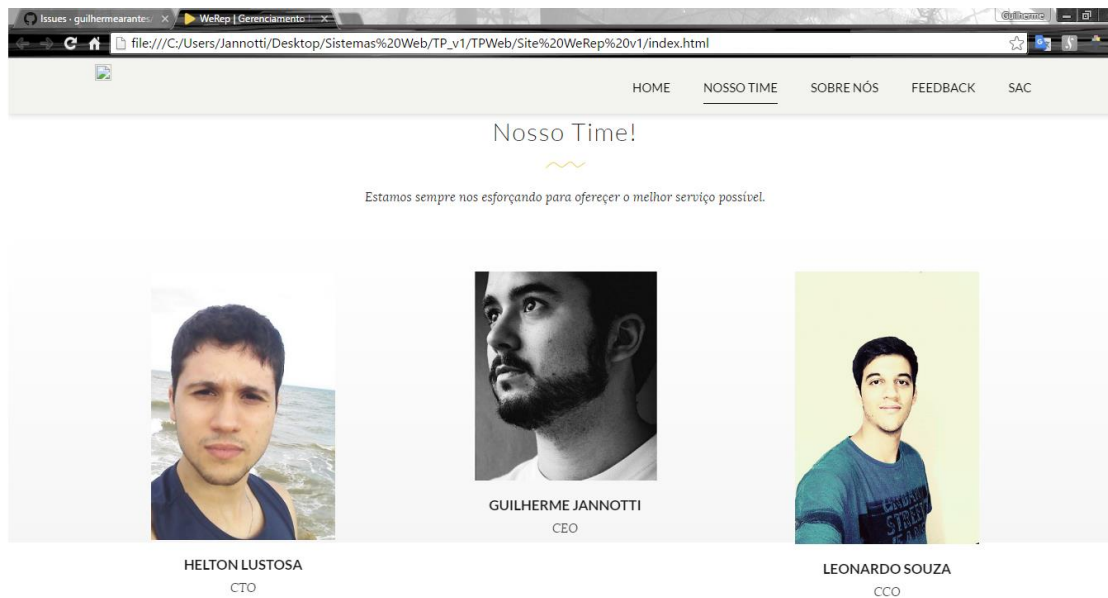
(diagrama de entidades-relacionamentos (DER) com a proposta para o banco de dados.)



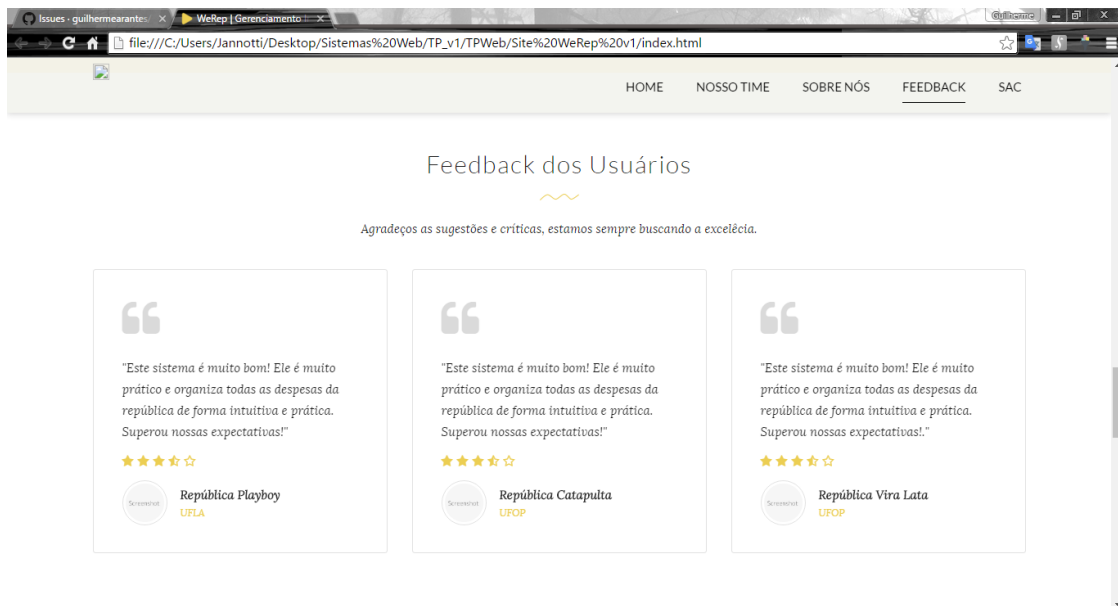
9. Protótipos



Home



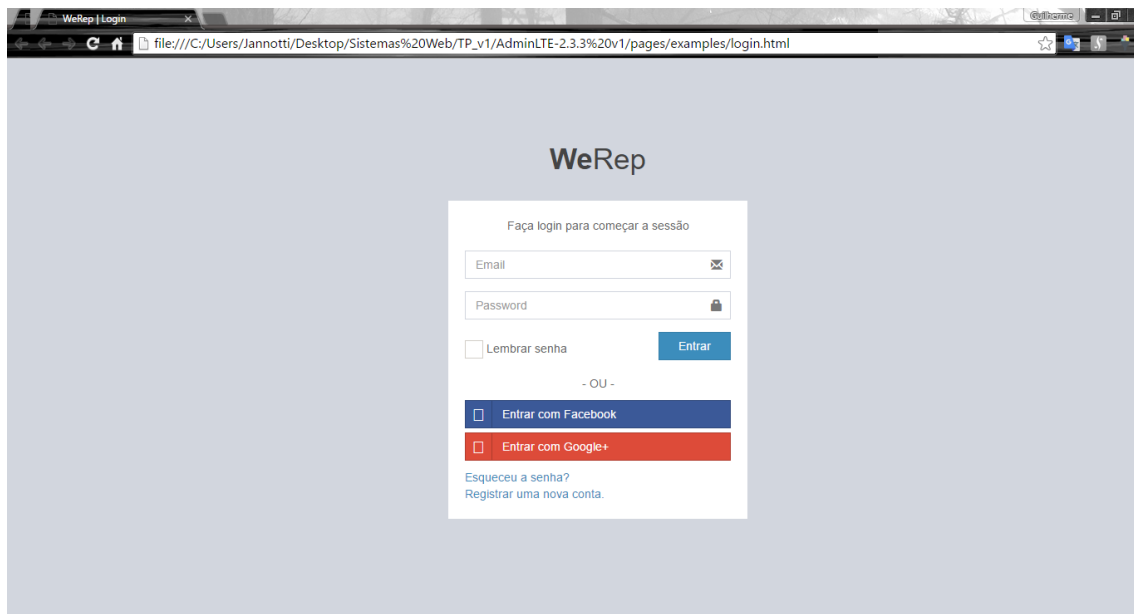
Nosso time.



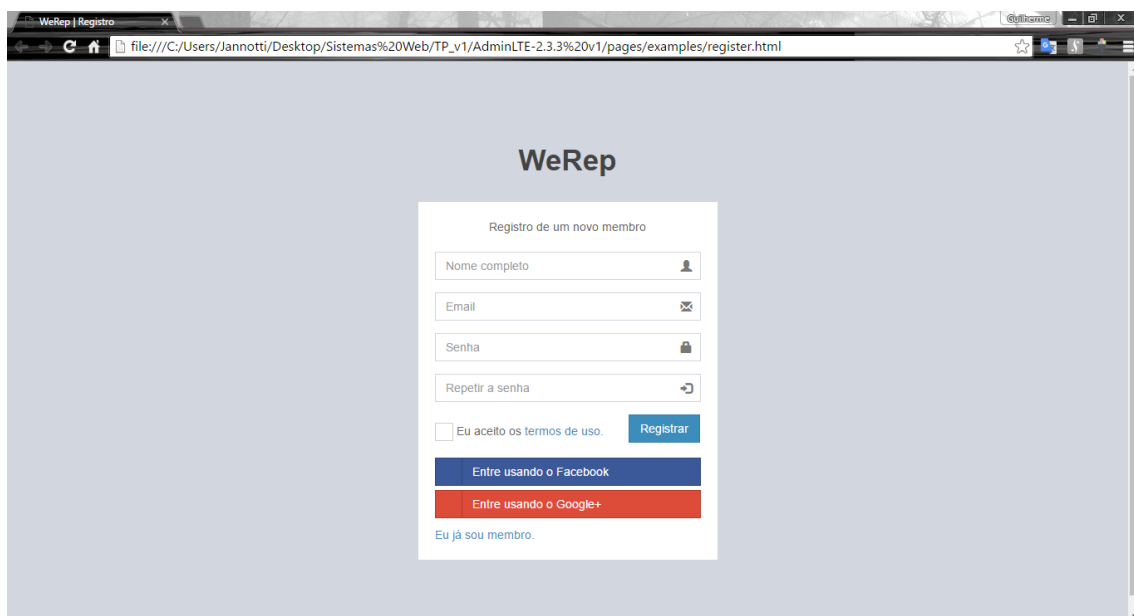
Feedback dos Usuários



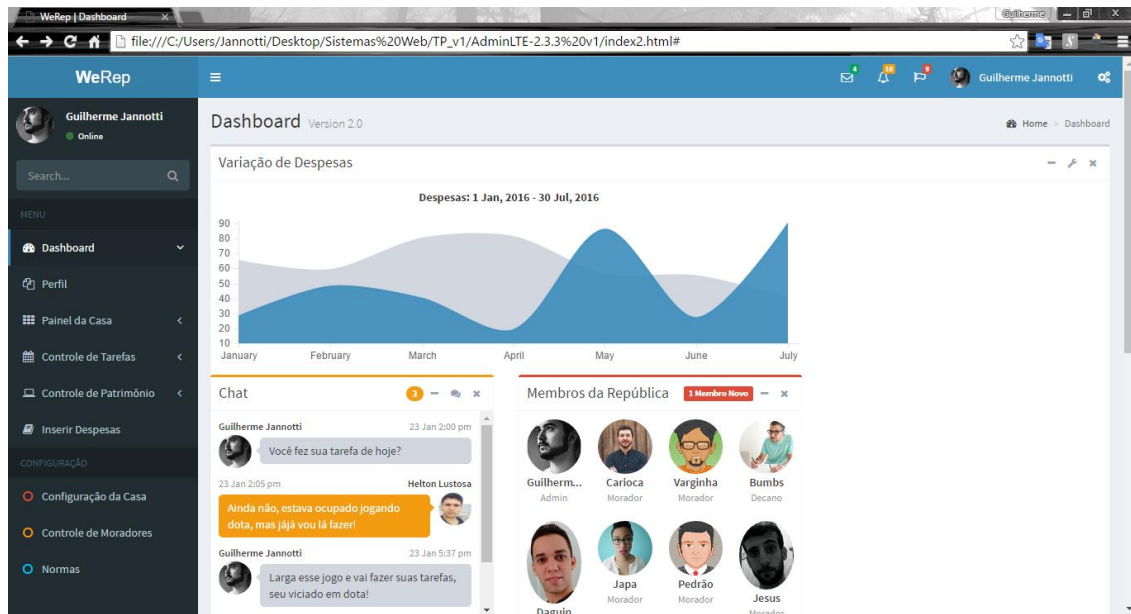
Perguntas Frequentes



Tela de Login do Dashboard



Tela de cadastro do Dashboard



Home do Dashboard

Códigos estão disponíveis no GitHub:

<<https://github.com/guilhermearantes/TPWeb>>

10. Cronograma

(apresentar um cronograma para execução do projeto.)

11. Referências Bibliográficas

(referências bibliográficas, caso se aplique.)