# Overseer

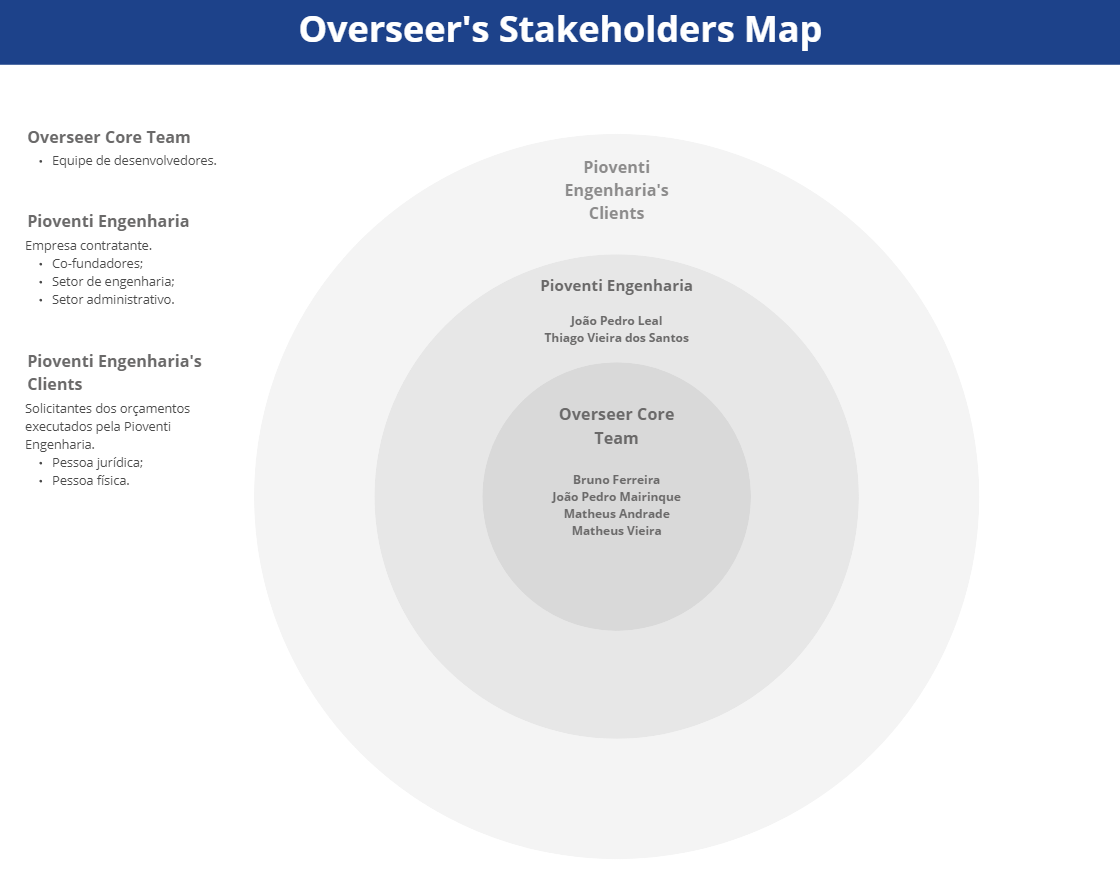
Bruno Ferreira, João Pedro Mairinque, Matheus Andrade e Matheus Vieira

1. **Apresentação do problema**

A Pioventi Engenharia é uma empresa que atua no ramo da engenharia civil em Belo Horizonte. Atualmente, a gestão financeira da empresa é realizada pelos donos da empresa, de forma completamente manual em uma planilha de dados. Foi identificado pelos mesmos a necessidade de um software que possa automatizar a parte do processo especialmente o gerenciamento dos orçamentos e permitir uma atualização orçamentária dos projetos em tempo real.

1. ***Stakeholders***

A Figura 1 a seguir apresenta o mapeamento das partes interessadas no software em confecção.



**Figura 1** - Mapa de stakeholders.

1. **Proposta da solução**

Overseer é um software com o propósito de auxiliar na criação de orçamentos com cálculos automatizados para os custos orçados, além da possibilidade de realizar adendos ao orçamento posteriormente.

O usuário pode inserir todos os custos relacionados à obra orçada e ao serviço em específico, para possuir uma gestão fina de gastos específicos. A atualização ocorre em tempo real e alertas são emitidos quando o custo real ultrapassar o custo orçado, o que pode significar a necessidade de ajustes no contrato por meio de adendos.

Por fim o software mostrará dados úteis abstraídos das obras, de forma a auxiliar no acompanhamento da gestão financeira com indicadores de desempenho.

1. **Projeto da solução**

O projeto implementado a partir de uma série de requisitos em estruturas de *Create, Read, Update and Delete* (CRUD), que por sua vez constituem o núcleo principal do software, o orçamento de um projeto de construção civil. Os CRUDs foram organizados em uma arquitetura de classes com informações associadas que derivam de um banco de dados relacional.

A implementação foi executada nas linguagens de programação JavaScript e TypeScript, respectivamente no *front-end* e *back-end*. O gerenciamento do banco de dados deu-se no sistema MySQL. Os principais *framewors* utilizados foram Node.js e o React, sendo o último para desenvolvimento da interface com usuário.

Foi utilizado o Docker como plataforma de serviço de virtualização do software, permitindo uma abordagem mais dinâmica no processo de desenvolvimento. Como repositório padrão do código fonte utilizou-se o GitHub.

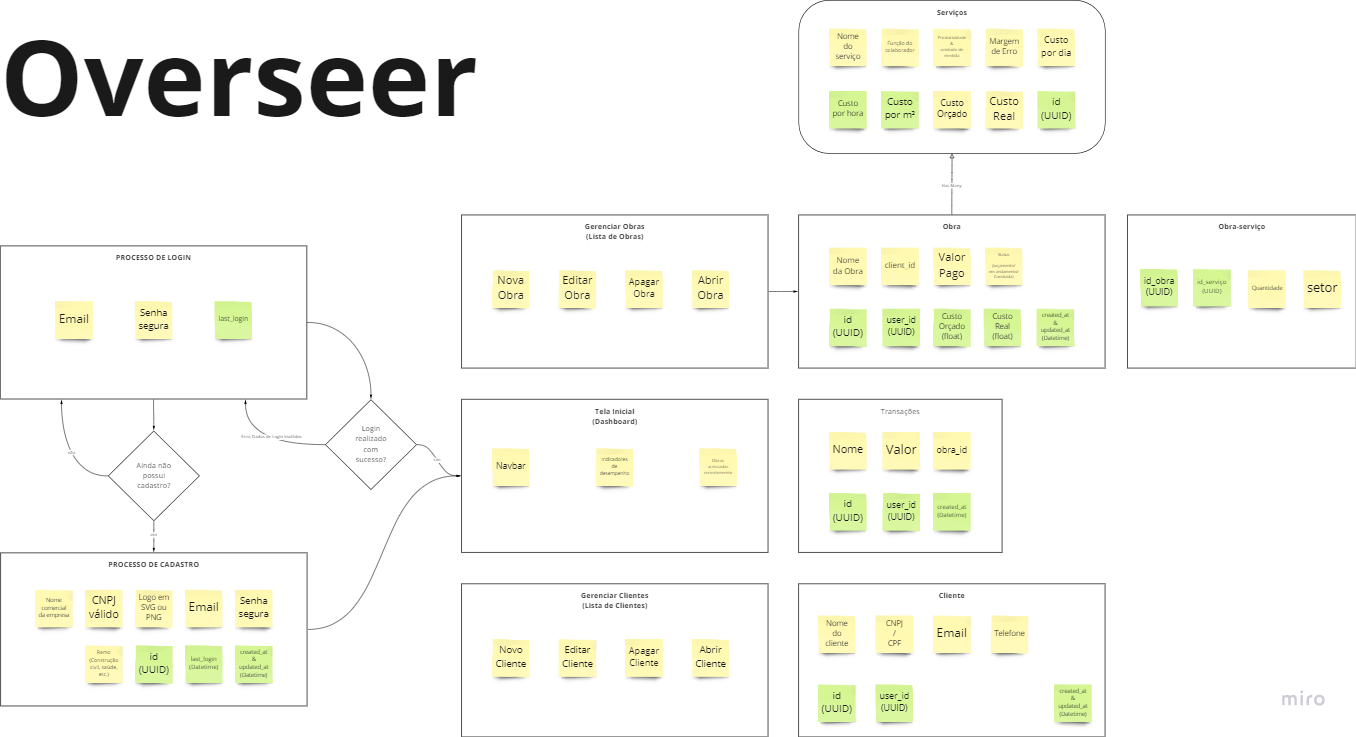
Os requisitos implementados no software estão listados na Tabela 1 abaixo.

|  |  |  |  |
| --- | --- | --- | --- |
| Requisitos Funcionais | | | |
| ID | Descrição do Requisito | Prioridade | Complexidade |
| RF01 | Administrador pode cadastrar usuário no sistema | Alta | Média |
| RF02 | Usuário faz login no sistema | Alta | Alta |
| RF03 | Usuário pode cadastrar serviços | Alta | Media |
| RF04 | Usuário pode cadastrar clientes | Alta | Média |
| RF05 | Usuário pode cadastrar obras | Alta | Média |
| RF06 | Usuário pode cadastrar orçamentos | Alta | Média |
| RF07 | Usuário pode cadastrar transações | Alta | Média |
| RF08 | Usuário pode inserir o percentual de lucro relativo ao orçamento | Alta | Baixa |
| RF09 | Usuário pode visualizar o lucro ou prejuízo total da empresa | Média | Média |
| RF10 | Usuário pode visualizar gastos orçados e reais no orçamento | Média | Média |
| RF11 | Usuário pode visualizar custo real e custo orçado da obra | Média | Média |
| RF12 | Usuário pode criar adendos para modificar o orçamento com status de concluído | Alta | Alta |
| RF13 | Usuário deve ser capaz de exportar dados do orçamento em formato PDF | Alta | Alta |
| RF14 | Usuário pode visualizar despesas gerais da empresa | Média | Média |
| RF15 | Usuário pode visualizar os serviços em andamento | Média | Média |
| RF16 | Usuário pode verificar os prazos para término de execução dos serviços | Média | Baixa |
| RF17 | Usuário visualiza margem de erro de custo nos serviços | Média | Baixa |
| RF18 | Usuário pode visualizar página com dados financeiros da empresa | Alta | Média |
| RF19 | Usuário pode visualizar percentual margens financeiras por orçamento | Média | Baixa |
| RF20 | Usuário pode visualizar valor total por setor do orçamento | Média | Baixa |
| Requisitos Não Funcionais | | | |
| ID | Descrição do Requisito | Prioridade | Complexidade |
| RNF01 | Implementação deve ser em typescript | Alta | Média |
| RNF02 | Banco dados relacional deve ser implementado via MySQL | Alta | Baixa |
| RNF03 | Interface do usuário deve ser realizada em React | Média | Média |
| RNF04 | Sistema deve possuir interface mobile funcional | Alta | Alta |

**Tabela 1** - Mapa de stakeholders.

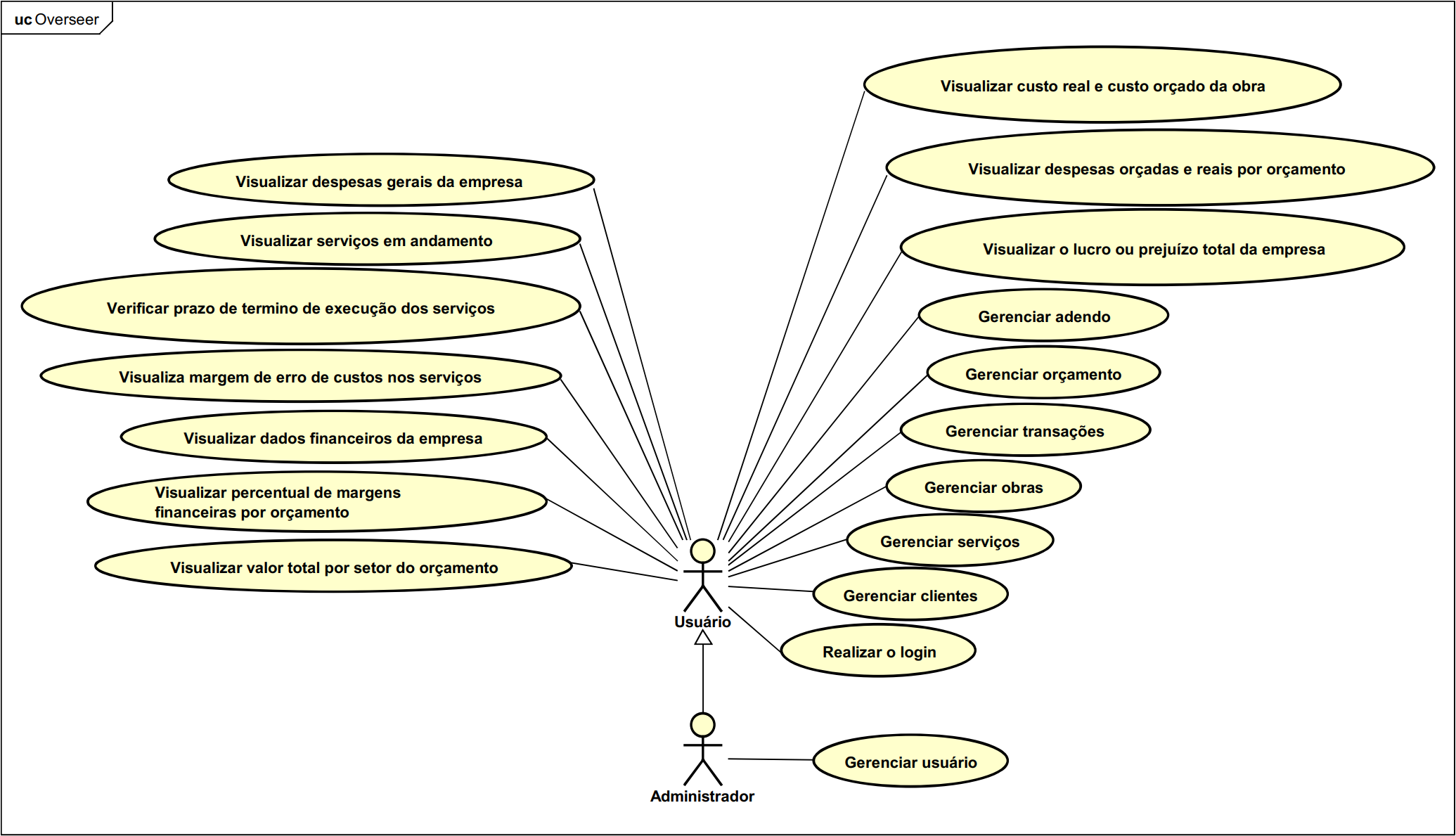
1. **Artefatos principais**

Inicialmente a concepção do software deu-se por entrevistas com os co-fundadores da Pioventi Engenharia, seguido de um processo de *brainstorm* (Figura 2) da equipe de desenvolvimento, para tal foi utilizada a plataforma Miro.

****

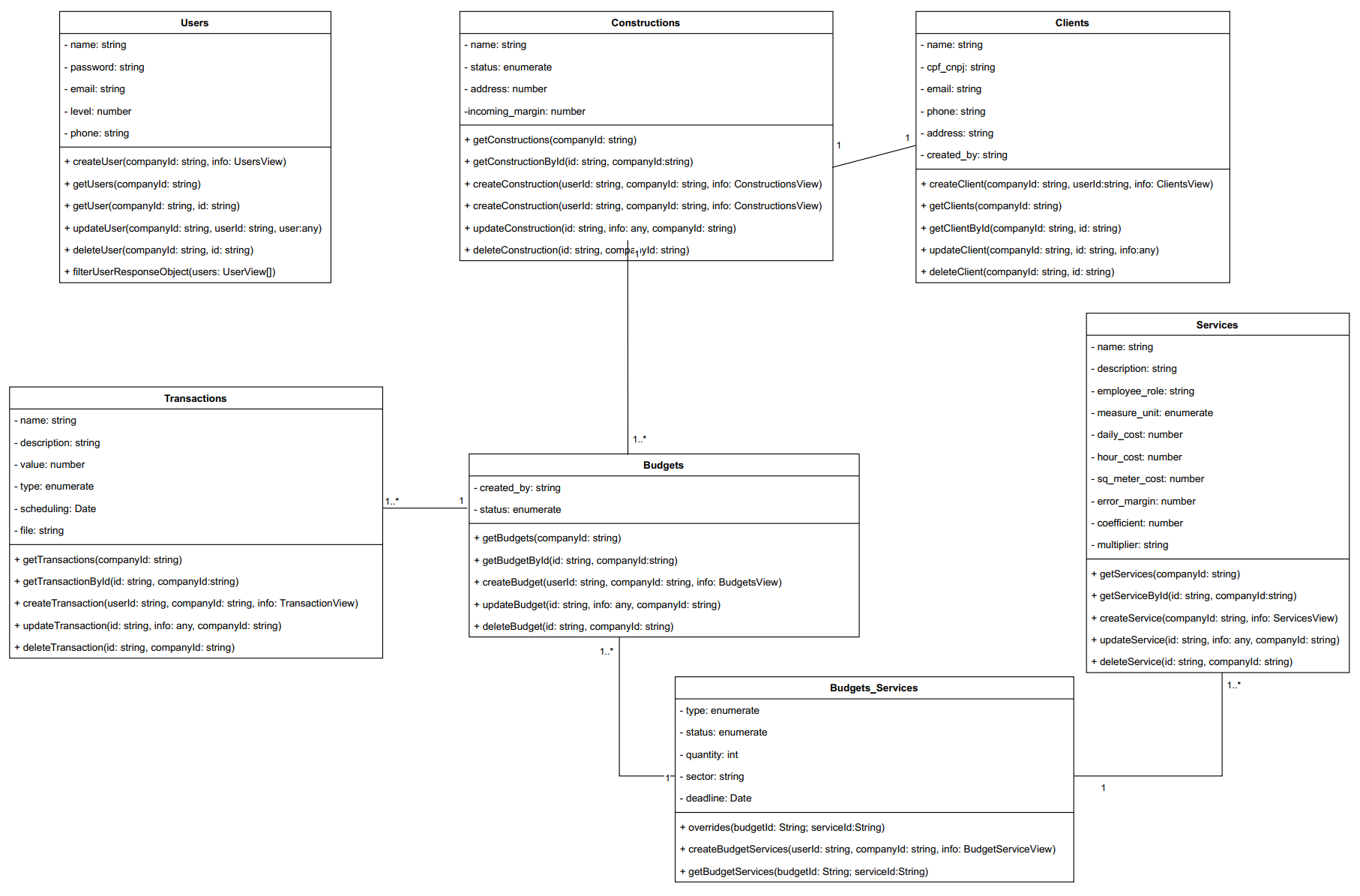
**Figura 2** - Brainstorm na plataforma Miro.

Desenhado o conceito e as responsabilidades gerais do Overseer iniciou-se o levantamento dos requisitos funcionais e não funcionais. Ao longo do processo de desenvolvimento esses sofreram mudanças a partir da análise detalhada dos casos de uso modelados no Astah, apresentados na Figura 3.



**Figura 3** - Diagrama de caso de uso do Overseer.

A Figura 4 apresenta o Modelo de Classes UML representativo do sistema. Definidos os requisitos a confecção do banco de dados ocorreu no sistema MySQL.



**Figura 4** - Modelo de Classes UML do Overseer.

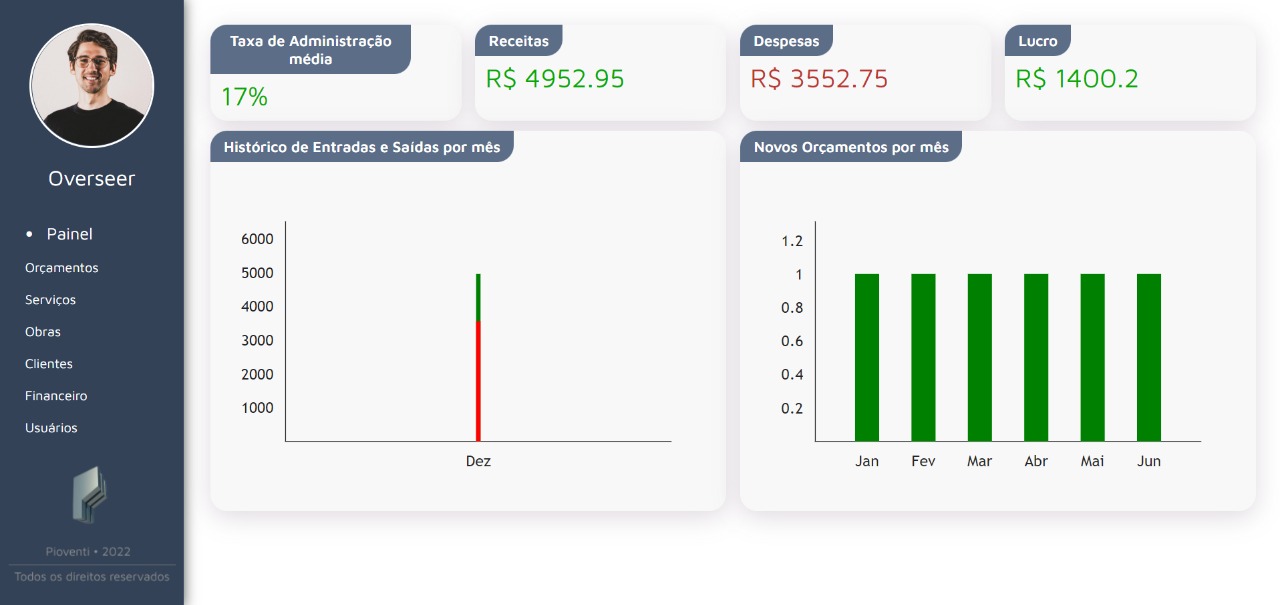
A concepção visual e o *layout* das páginas da aplicação foram desenvolvidos no Figma. As figuras 5, 6 e 7 abaixo apresentam o resultado final das telas de orçamento, controle financeiro e painel de controle administrativo, respectivamente.



**Figura 5** - Tela orçamentos.



**Figura 6** - Tela Financeiro.



**Figura 7 -** Painel de Controle Administrativo.

1. **Conclusões**

O feedback das reuniões entre a equipe de desenvolvimento e os representantes da Pioventi Engenharia foram os melhores possíveis. A empresa contratante mostrou-se muito satisfeita com o resultado final do software apresentado, além do interesse em expandir o projeto. Sendo assim entende-se que o Overseer cumpre todos os requisitos definidos junto a Pioventi Engenharia com eficiência.