1. **Objetivo da nossa aplicação:**

(qual o propósito de nossa aplicação, qual problema ela pode vir a resolver, quem são os usuários envolvidos)

Ser um gerenciador de tarefas simples e genérico, semelhante a uma agenda, com o propósito de facilitar a organização da rotina de diversos tipos de clientes.

1. **Requisitos encontrados (liste os requisitos encontrados):**

**Funcionais:**

Formulários para criar e editar tarefas;

Formulários para criar e editar contatos;

Opções para excluir tarefas e marcá-las como concluída ou cancelada;

Opções para excluir contatos;

Opção para excluir um usuário do sistema;

Conexão com um banco de dados para manipular tarefas, contatos e usuários;

Exibição na tela das tarefas e contatos criados;

Exibição na tela do usuário em sessão;

Formulários para cadastro, entrada e edição da senha de um usuário;

**Não Funcionais:**

Ferramentas de pesquisa;

Mensagens de erro;

Linguagem clara para o usuário;

Opções para ocultar a exibição temporariamente de tarefas e contatos;

1. **Tecnologias, softwares e linguagens que serão utilizados:**

(descreva aqui as tecnologias de desenvolvimento envolvidas bem como softwares utilizados, a linguagem que utilizaremos tanto para codificação como banco e dados).

**Linguagens de programação, marcação e estilo:**

Primárias: PHP, HTML, CSS;

Secundárias: JavaScript.

**Linguagem do banco de dados:**

MySQL.

**Software de edição de código:**

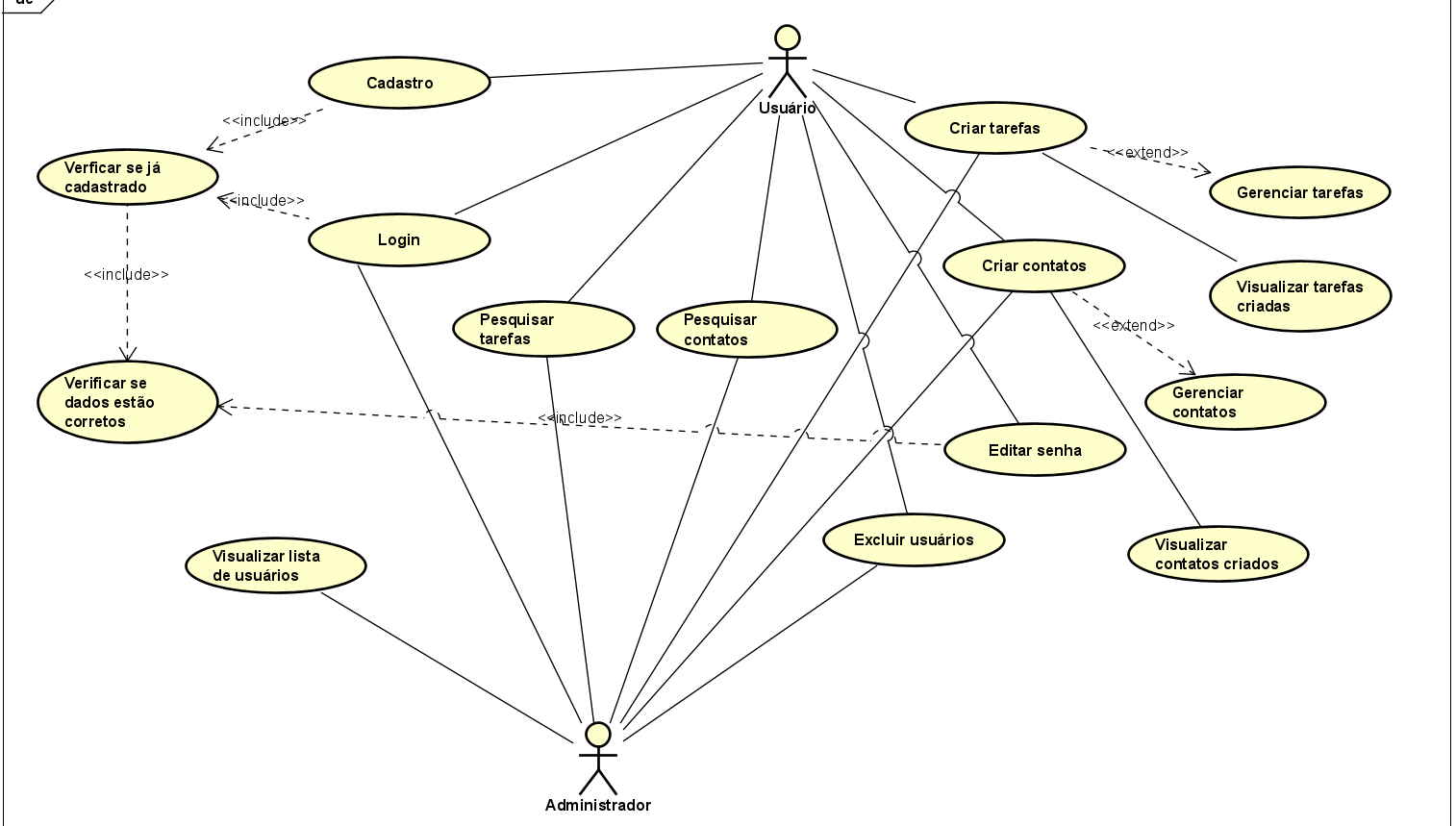
VSCode.

**Ferramentas para diagramação, wireframes, tabelas, etc.:**

Astah, Excel, PowerPoint, Figma, Lucidchart.

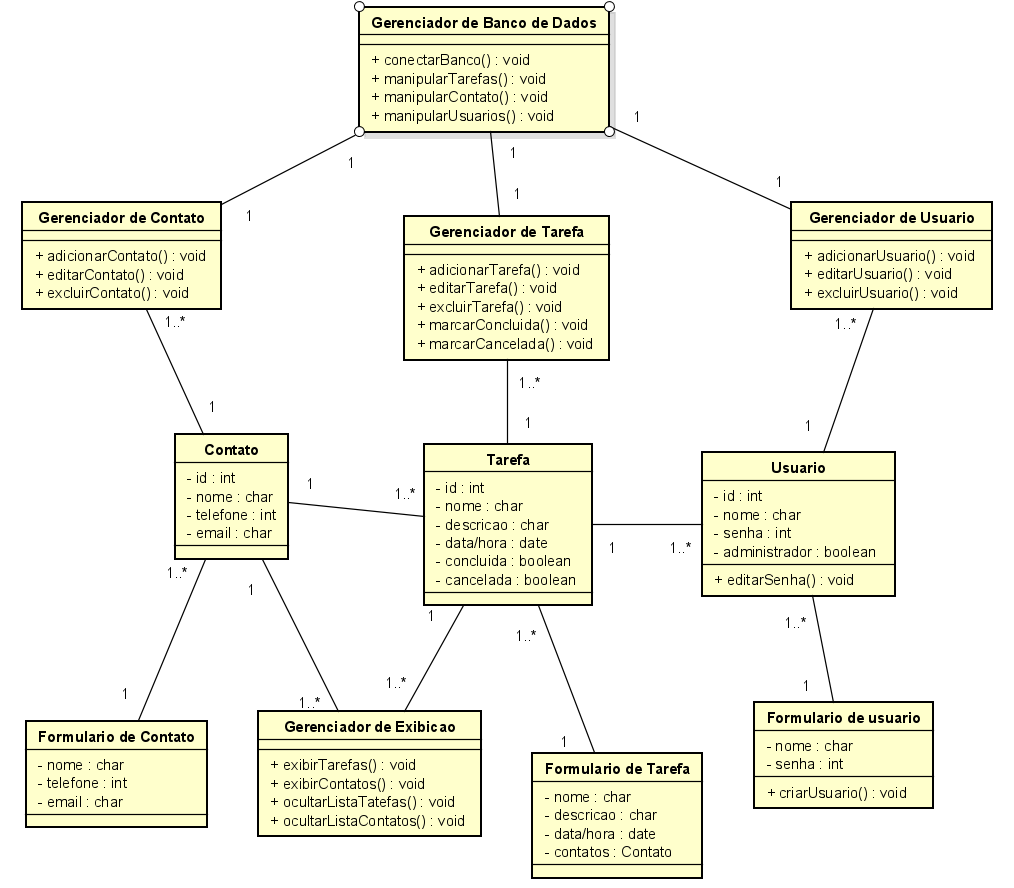
1. **Diagrama de estudo de casos:**

(faça um print da tela e coloque aqui seu diagrama de caso de uso)



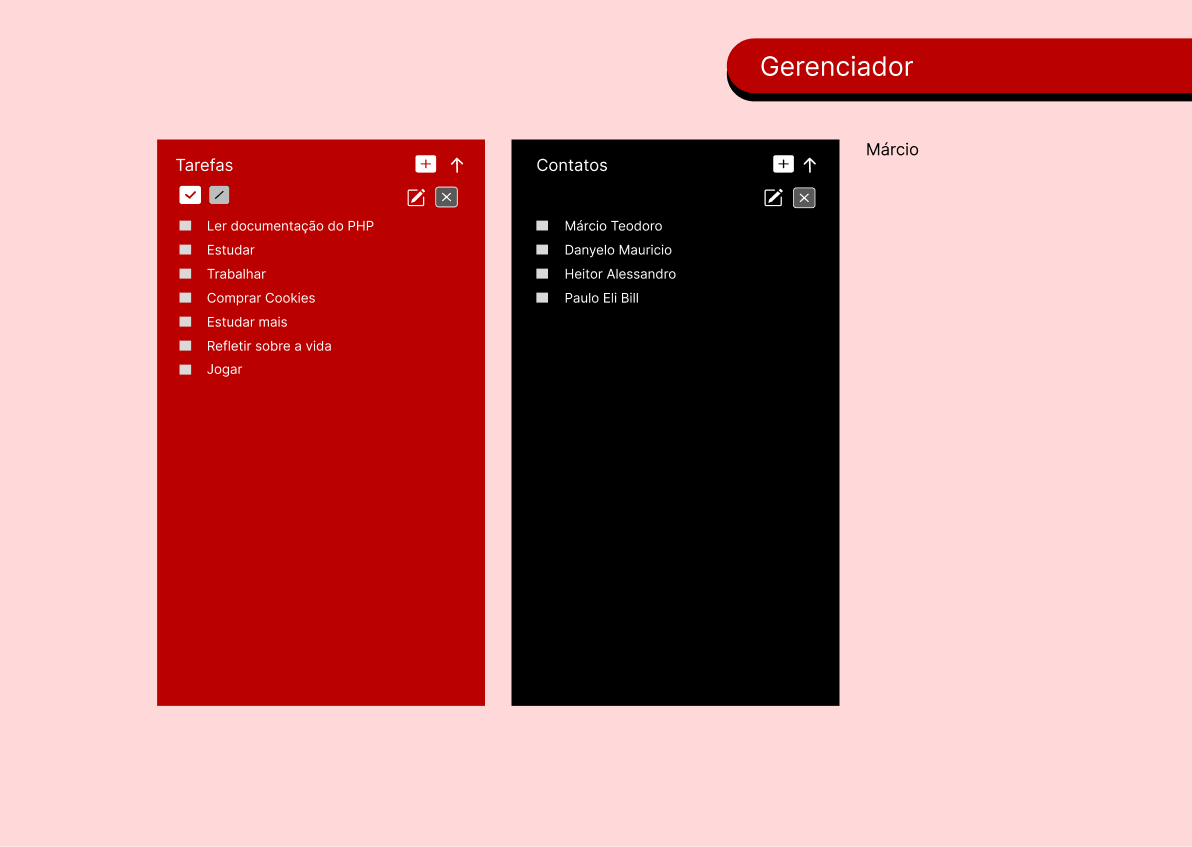
1. **Diagrama de Classes:**

(semelhante ao diagrama acima, print seu diagrama de classes e coloque uma imagem aqui).



1. **Wireframe da tela principal:**

(utilizando qualquer programa de desenho, rascunhos ou próprios para wireframes, faça um esboço da tela inicial da sua aplicação, lembre-se de não inventar a roda).



1. **Descrição das tabelas de banco de dados que serão implementadas:**

(Já comece a pensar em seu banco de dados e liste aqui as possíveis tabelas envolvidas).

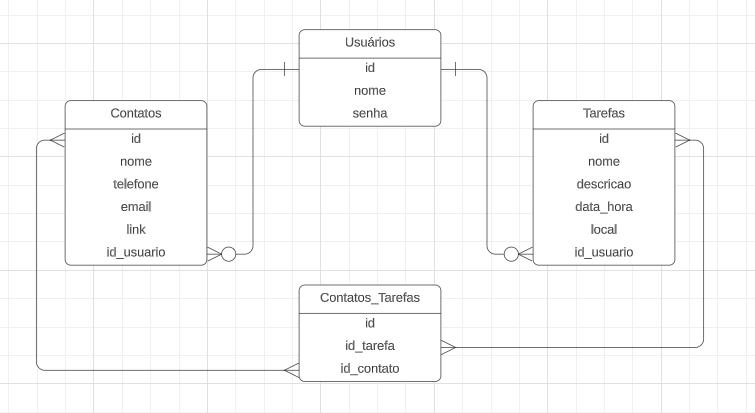
**Usuários** - Todos os usuários cadastrados no sistema, incluindo os administradores.

**Tarefas –** Todas as tarefas criadas, cada uma vinculada a um usuário específico.

**Contatos –** Todos os contatos criados, cada um vinculado a um usuário específico.

**Contatos\_Tarefas –** Todos os contatos que foram vinculados a uma determinada tarefa.

**Diagrama de entidade e relacionamento:**



1. **Teste que serão realizados:**

(pesquise na internet os tipos de testes existentes, e escolha um desses que pode ser aplicado na sua aplicação afim de validar a programação).

### **Testes de unidade**

Testes de unidade são feitos em um nível muito baixo, próximo ao código-fonte do aplicativo. Eles consistem em testar métodos e funções individuais de classes, componentes ou módulos usados pelo software. Testes de unidade, em geral, têm custo baixo para automatizar e podem ser executados com rapidez por um servidor de integração contínua.

1. **Requisitos mais detalhados:**

Em arquivo próprio.

1. **?**

**Dupla: Danyelo Muñoz e Márcio Rodriguês Teodoro**