Relatório do Sistema de Assistente de Priorização de Tarefas

1. Visão Geral e Contexto do Sistema

Este projeto visa a criação de um sistema assistente de priorização de tarefas, com o objetivo de melhorar a produtividade dos usuários por meio da organização e priorização automática de suas atividades diárias. A interface será web e permitirá ao usuário adicionar, gerenciar e priorizar tarefas automaticamente com base em prazos e no histórico de execução. Além disso, o sistema contará com a integração com uma API de Inteligência Artificial (IA) para sugerir a ordem de execução das tarefas.

Objetivos:

* Organização: Facilitar o gerenciamento das tarefas diárias.
* Priorização Automática: Utilizar IA para sugerir a ordem de execução das tarefas com base no prazo e na análise histórica.
* Notificações: Enviar lembretes e avisos para tarefas próximas da data de vencimento.

Módulos do Sistema:

* Web: Interface web para interação com o usuário.
* Login: Módulo de autenticação de usuários.
* CRUD de Tarefas: Módulo para criar, visualizar, editar e excluir tarefas.
* IA para Priorização de Tarefas: Módulo que utiliza IA para sugerir a ordem das tarefas.
* Notificações e Relatórios: Envio de lembretes e geração de relatórios de atividades.

1. Requisitos Funcionais

2.1 Cadastro e Autenticação de Usuários

* Objetivo: Permitir que o usuário se cadastre e faça login no sistema.
* Requisitos:
* O sistema deve permitir que os usuários façam login com nome de usuário e senha.
* O sistema deve exibir mensagens de erro caso o usuário insira credenciais inválidas ou um nome de usuário já existente.

2.2 Gerenciamento de Tarefas (CRUD)

* Objetivo: Permitir ao usuário gerenciar suas tarefas de forma simples e intuitiva.
* Requisitos:
* O usuário deve ser capaz de criar, visualizar, editar e excluir tarefas.
* O sistema deve validar as informações das tarefas antes de armazená-las no banco de dados.

2.3 Priorização de Tarefas

* Objetivo: Priorizar automaticamente as tarefas de acordo com critérios predefinidos (prazos e histórico).
* Requisitos:
* O sistema deve utilizar IA para sugerir a ordem de execução das tarefas.
* O sistema deve notificar o usuário caso a API de priorização esteja indisponível.

2.4 Notificações de Tarefas

* Objetivo: Enviar lembretes para as tarefas próximas da data de vencimento.
* Requisitos:
* O sistema deve enviar notificações baseadas nas datas de vencimento das tarefas.
* O sistema deve verificar se as notificações estão ativadas para cada tarefa antes de enviá-las.

2.5 Relatórios e Estatísticas

* Objetivo: Fornecer relatórios detalhados sobre as tarefas realizadas.
* Requisitos:
* O sistema deve gerar relatórios de atividades concluídas.
* O sistema deve possibilitar o acompanhamento do progresso do usuário.

2.6 Integração com Calendário

* Objetivo: Integrar as tarefas com sistemas de calendário para uma gestão centralizada.
* Requisitos:
* O sistema deve sincronizar as tarefas com o calendário do usuário.

2.7 Criação de Tarefas Frequentes

* Objetivo: Facilitar a criação de tarefas recorrentes.
* Requisitos:
* O sistema deve permitir a criação de tarefas frequentes com facilidade.

3. Requisitos Não Funcionais

3.1 Banco de Dados

* O sistema utilizará MySQL para armazenar dados relacionados ao usuário e às tarefas.

3.2 Disponibilidade da API de IA

* A API de IA será configurada para garantir que as sugestões de priorização sejam fornecidas de forma rápida e precisa.

3.3 Desempenho

* O sistema deve ser ágil, com tempo de resposta rápido para as operações de CRUD e para a priorização das tarefas.

3.4 Segurança

* O sistema deve garantir a segurança dos dados do usuário durante o processo de autenticação e ao acessar informações pessoais.

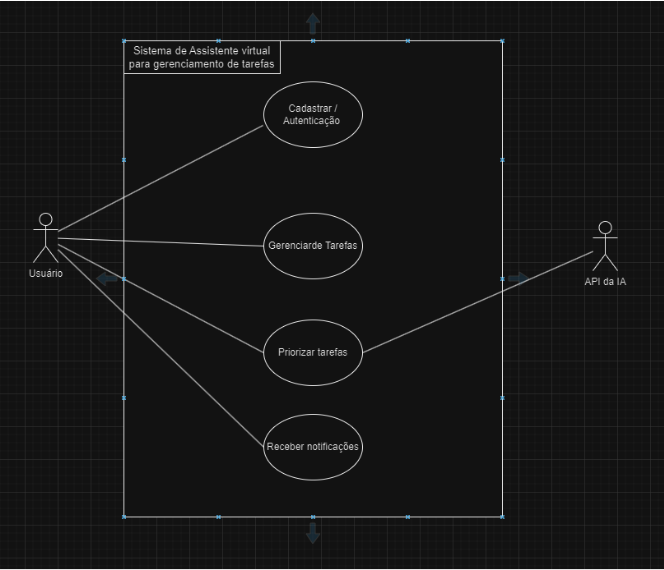
3.5 Flexibilidade para Extensões Futuras

* O sistema deve ser desenvolvido de forma modular, permitindo futuras extensões, como integração com outros sistemas ou funcionalidades adicionais (por exemplo, categorização de tarefas).

4. Diagrama de Caso de Uso

* O Diagrama de Caso de Uso ilustra os atores e os casos de uso principais do sistema, com foco nas interações do usuário com o sistema de login, gerenciamento de tarefas, priorização de tarefas, e notificações.

**Diagrama de caso de uso geral**



**Cenários de caso de uso**

**UC-01**

| Nome do caso de uso | Cadastrar/Autenticação de usuário |
| --- | --- |
| Descrição | Usuário vai fazer login ou se cadastrar no site |
| Atores | Usuário |
| Pré-Condições | Estar no navegador |
| Fluxo principal | 1.Usuário abre o aplicativo  2.Preenche os dados de nome e senha  3.O sistema confirma os dados  4.O usuário tem acesso ao sistema |
| Fluxo alternativo | 1.O usuário digita nome ou senha inválido  2.O sistema exibe uma mensagem de erro e avisa para o usuário |
| Regra de execução | Se já existir um nome de usuário, uma mensagem de erro deve ser exibida solicitando que o usuário escolha outro nome. |

**UC-02**

| Nome do caso de uso | Gerenciar tarefas |
| --- | --- |
| Descrição | Criar, ler, editar e excluir tarefas |
| Atores | Usuário |
| Pré-Condições | Usuário deve estar logado |
| Fluxo principal | 1.Usuário o que quer fazer com a tarefa  2.O dado vai ser atualizado no banco de dados |
| Fluxo alternativo | 1.O sistema não consegue atualizar o banco de dados  2.O sistema informa que não foi possível confirmar a ação |
| Regra de execução | O sistema deve validar os dados (campos) da tarefa antes de atualizar o banco de dados |

**UC-03**

| Nome do caso de uso | Priorizar tarefas |
| --- | --- |
| Descrição | O sistema sugere uma ordem para a execução das tarefas |
| Atores | Usuário e API da IA |
| Pré-Condições | As tarefas devem estar cadastradas |
| Fluxo principal | 1.Usuário solicita uma sugestão  2.API analisa as informações de cada tarefa e sugere as prioridades das tarefas |
| Fluxo alternativo | 1.Não há nenhuma sugestão  2.O sistema mostra uma frase padrão avisando que a prioridade já está adequada |
| Regra de execução | Se a API de IA estiver fora do ar ou inacessível, o sistema exibe uma mensagem informando que a priorização não está disponível |

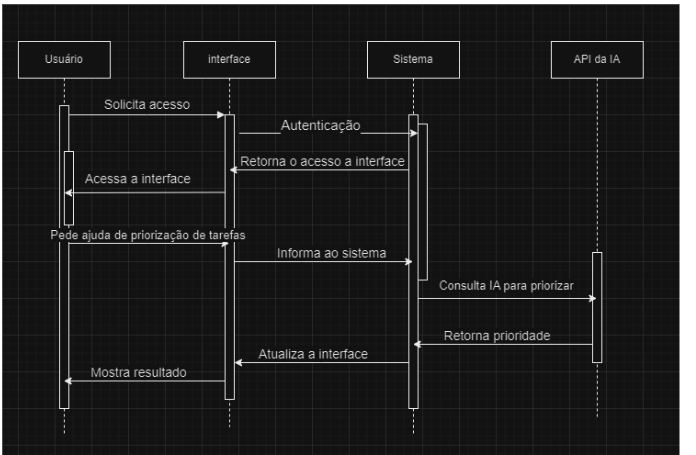
**UC-04**

| Nome do caso de uso | Receber notificações |
| --- | --- |
| Descrição | Sistema envia lembrete das tarefas |
| Atores | Usuário |
| Pré-Condições | Tarefa deve ter algum lembrete configurado |
| Fluxo principal | 1.Sistema verifica as tarefas  2.Envia a notificação de acordo com a data da tarefa |
| Fluxo alternativo | O usuário ignora a notificação e arquiva a notificação |
| Regra de execução | O sistema deve verificar se as notificações estão ativadas para cada tarefa antes de enviar o lembrete. |

**Diagrama de caso de uso de Priorização de Tarefas**

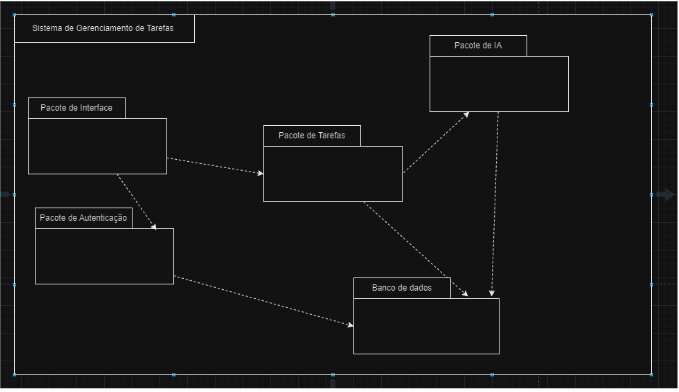


5. Diagrama de Sequência

O Diagrama de Sequência mostra o fluxo de interações entre os objetos do sistema. Ele ilustra como o usuário interage com a interface web, como as tarefas são processadas no backend e como a API de IA sugere a priorização das tarefas.

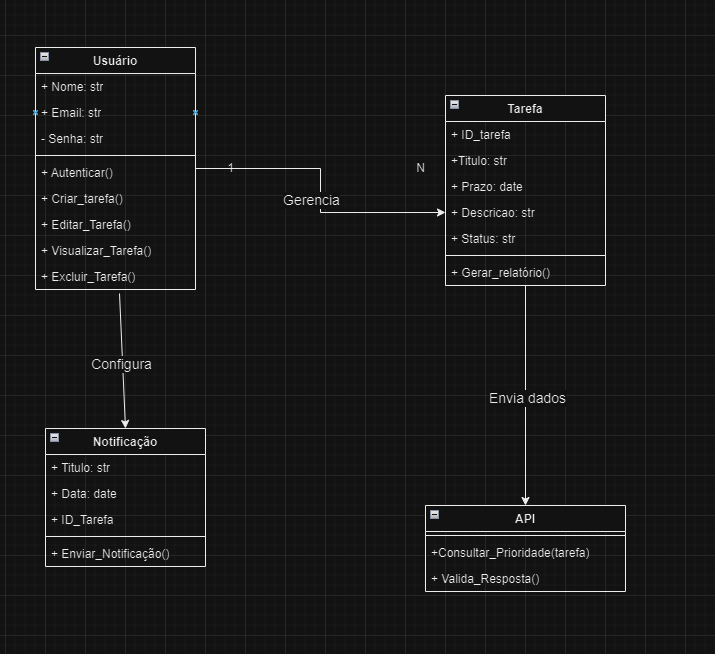
6. Diagrama de Pacotes

O Diagrama de Pacotes organiza o sistema em pacotes distintos, como:



7. Diagrama de Classes

O Diagrama de Classes detalha a estrutura das classes principais do sistema, incluindo:



8. Diagrama de Objetos

O Diagrama de Objetos mostra exemplos concretos de objetos no sistema, como uma instância de um usuário e suas tarefas no momento da execução, foi feito utilizando programação orientada a objetos por meio da linguagem python e o código está disponível no github

9. Plano de Implementação Inicial

A implementação será dividida em fases:

* Configuração do Banco de Dados: Iniciar com a configuração do banco de dados MySQL.
* Desenvolvimento da Interface Web: Criar a interface de usuário para login, gerenciamento de tarefas e visualização de relatórios.
* Implementação do CRUD de Tarefas: Permitir a criação, leitura, atualização e exclusão de tarefas.
* Integração com a API de IA: Configurar a API de IA para priorização das tarefas por meio do sistema multi-agente
* Notificações e Relatórios: Implementar o envio de notificações e a geração de relatórios.
* Testes Finais: Realizar testes de integração para garantir a funcionalidade de todos os módulos.

10. Conclusão

Este relatório apresenta o planejamento e a estrutura do sistema assistente de priorização de tarefas. Através de uma abordagem modular e com foco em IA, o sistema visa otimizar a gestão de tarefas dos usuários, promovendo maior produtividade. As fases de implementação e os requisitos funcionais e não funcionais foram claramente definidos, e a arquitetura do sistema está bem estruturada para atender às necessidades do projeto.