Análise e Especificação do Sistema de

Automatização

IA ESPAÇO MAKER

| VERSÃO: 0.1 | DATA DE REVISÃO: |
| --- | --- |

Sumário

[1. INTRODUÇÃO 1](#_heading=h.v87e9qjrfnkb)

[1.1Descrição do Problema 1](#_heading=h.fxkxwzqyti7p)

[1.2Alternativas existentes 1](#_heading=h.hmrrfngcitki)

[1.3Objetivos 1](#_heading=h.6a8ejbtprd2h)

[1.4Definição do usuário 1](#_heading=h.mb0y25nm25ii)

[2. FUNCIONALIDADES DO SISTEMA 2](#_heading=h.to20g7r38g5a)

[2.1. Requisitos funcionais 2](#_heading=h.mlnqlkb3one9)

[2.2. Requisitos não-funcionais 2](#_heading=h.eczlu37mq3a2)

[2.3. Escopo 3](#_heading=h.xe6hxd27vij7)

[3. DIAGRAMA DE CASO DE USO 4](#_heading=h.hiok4ljl402u)

[3.1. Diagrama de caso de uso 4](#_heading=h.n5ysk7fko6bj)

[3.2. Especificação de caso de uso 4](#_heading=h.gtgwl6621eeg)

[4. MODELO DE CLASSES 5](#_heading=h.kb843wkzj29r)

[4.1. Modelo de Domínio ( Fase de análise) 5](#_heading=h.rjm4z0q0ob9b)

[4.2. Diagrama de Classe (Fase de projeto) 5](#_heading=h.6cj49gafy6r6)

[5. DIAGRAMA DE SEQUÊNCIA 6](#_heading=h.6x33b1nlb73n)

[6. DIAGRAMA DE MÁQUINA 7](#_heading=h.9qmabu4ze0oh)

[7. CARACTERÍSTICAS OPERACIONAIS DO SOFTWARE 8](#_heading=h.vgqr18xur132)

[7.1. Ambiente operacional 8](#_heading=h.tk9uzp868q7m)

[7.2. Arquitetura do software 8](#_heading=h.y6tifqgfdg1p)

[7.3. Subsistemas e Componentes 8](#_heading=h.fi3w1pflmdgo)

[8. RISCOS 9](#_heading=h.9gx2hjvevox0)

[9. CRONOGRAMA 1](#_heading=h.eb0ihp63y3ll)

[10. CONCLUSÕES 2](#_heading=h.wpg7sgrgbh29)

[RESPONSABILIDADES 3](#_heading=h.bfydy4g8xsci)

# 1. INTRODUÇÃO

## **Descrição do Problema**

O crescimento acelerado da inteligência artificial (IA) e sua adoção em diferentes setores têm gerado uma demanda crescente por processamento computacional, o que, por consequência, aumenta o consumo energético. Grande parte das infraestruturas de IA atuais ainda depende de fontes não renováveis de energia, gerando impactos ambientais e aumentando os custos operacionais.

No contexto educacional e de inovação prática, como o Espaço Maker, essa realidade se torna um desafio ainda maior, pois os projetos desenvolvidos por estudantes e entusiastas geralmente enfrentam limitações de recursos financeiros e estruturais. A execução de modelos de linguagem de grande escala (LLMs) requer não apenas hardware compatível, mas também gestão eficiente de energia — algo que pode ser um gargalo em ambientes de baixo orçamento.

Ao mesmo tempo, o acesso a plataformas de interação como Discord e ferramentas de automação com APIs abertas oferece uma oportunidade para integrar sistemas inteligentes em interfaces populares, tornando a IA mais acessível para usuários comuns.

Manter a situação atual significa manter projetos educacionais e sustentáveis limitados por barreiras energéticas e técnicas, além de perder oportunidades de desenvolver soluções escaláveis que possam operar em ambientes com restrição de energia (zonas rurais, espaços educacionais, makerspaces etc.).

O projeto, portanto, visa resolver esse cenário ao propor uma solução de IA distribuída, com gerenciamento energético baseado em fontes renováveis, monitorado em tempo real via Prometheus, e com interface interativa via Discord.

### 

## **Alternativas existentes**

*[Identificar e descrever, de forma clara, as alternativas existentes para a solução do problema. Fornecer todos os detalhes que se julgue necessário para a perfeita compreensão da alternativa. As alternativas existentes devem incluir, além da solução que está sendo proposta pelo projeto, produtos similares já existentes no mercado ou pesquisas acadêmicas em andamento e que solucionam o problema de forma similar.]*

### 

## **Objetivos**

Desenvolver um sistema de inteligência artificial capaz de operar com energia proveniente de fonte solar, integrando modelos de linguagem de grande escala (LLMs) com monitoramento energético em tempo real via Prometheus. Integrar o sistema com o Discord para facilitar o envio de comandos e o uso educacional. Otimizar o uso de energia gerada para alocar recursos computacionais de forma eficiente e sustentável. Promover acessibilidade à IA de forma local, econômica e escalável, com foco em ambientes educacionais e makerspaces.

## **Definição do usuário**

Os principais usuários do sistema são estudantes, professores e entusiastas do Espaço Maker, que desejam explorar e aplicar inteligência artificial de forma prática, acessível e sustentável. Além disso, o sistema poderá ser utilizado por desenvolvedores, pesquisadores e educadores interessados em executar modelos de IA com recursos energéticos limitados, monitorar o consumo em tempo real e interagir com a plataforma por meio de comandos via Discord

# 2. FUNCIONALIDADES DO SISTEMA

## **2.1. Requisitos funcionais**

| **Id** | **Descrição** | **Solicitante** | **Prioridade** |
| --- | --- | --- | --- |
| **RF01** |  | **Patrocinador** | **Alta** |
| **RF02** |  | **Patrocinador** | **Alta** |
| **RF03** |  | **Usuário final** | **Alta** |
| **RF04** |  | **Usuário final** | **Alta** |
| **RF05** |  | **Usuário final** | **Média** |
| **RF06** |  | **Usuário final** | **Alta** |
| **RF07** |  | **Patrocinador** | **Alta** |
| **RF08** |  | **Patrocinador** | **Alta** |
| **RF09** |  | **Usuário final** | **Alta** |
| **RF10** |  | **Usuário final** | **Média** |
| **RF11** |  | **Usuário final** | **Média** |

## **2.2. Requisitos não-funcionais**

| **ID** | **Descrição** | **Solicitante** | **Prioridade** | **Detalhamento** |
| --- | --- | --- | --- | --- |
| RFN01 | Interface amigável | Usuário final | Alta | O sistema deve apresentar uma interface intuitiva e de fácil navegação. |
| RFN02 | Tempo de resposta | Patrocinador | Alta | O tempo de carregamento das páginas não deve exceder 2 segundos. |
| RFN03 | Compatibilidade com navegadores | Patrocinador | Alta | O sistema deve funcionar nos navegadores Chrome, Firefox e Edge (últimas 3 versões). |
| RFN04 | Adoção de boas práticas de desenvolvimento | Patrocinador | Alta | O sistema deve seguir padrões definidos pela organização para código e segurança. |
| RFN05 | Criptografia de comunicação | Patrocinador | Alta | Todas as transações devem ocorrer sob protocolo HTTPS com TLS ativo. |
| RFN06 | Alta disponibilidade | Patrocinador | Alta | O sistema deve estar disponível pelo menos 99,5% do tempo útil (horário comercial). |
| RFN07 | Tempo médio de reparo (MTTR) | Patrocinador | Média | O tempo médio para recuperação após falhas não deve ultrapassar 2 horas. |
| RFN08 | Controle de bugs críticos | Patrocinador | Alta | O sistema deve manter taxa máxima de 1 bug crítico a cada 5.000 linhas de código. |
| RFN09 | Suporte a múltiplos usuários simultâneos | Patrocinador | Média | O sistema deve suportar no mínimo 100 usuários ativos simultaneamente. |
| RFN10 | Manutenibilidade e padrão de código | Equipe técnica | Média | O código deve seguir padrão de nomenclatura e estar bem documentado. |

### 

## **2.3. Escopo**

*O projeto contempla o desenvolvimento de um sistema local de inteligência artificial alimentado por energia solar, capaz de executar modelos de linguagem de grande escala (LLMs) de forma sustentável e monitorada em tempo real. Serão implementadas as seguintes funcionalidades:*

* *Integração de um painel solar com sistema de monitoramento energético via Prometheus.*
* *Implementação de lógica de controle para alocação de carga computacional com base na disponibilidade energética.*
* *Execução local de LLMs especializadas com comunicação por meio de comandos enviados via Discord.*
* *Interface básica para visualização dos dados de energia e uso da IA.*
* *Testes práticos no ambiente do Espaço Maker com usuários reais.*

| **O projeto entregará** |
| --- |
|  |
|  |
|  |
|  |

| **O projeto não entregará** |
| --- |
|  |
|  |
|  |
|  |

# 3. DIAGRAMA DE CASO DE USO

*[Inserir o diagrama de caso de uso e as especificações de caso de uso: fluxo básico, alternativos exceções e regras de negócio.]*

## **3.1. Diagrama de caso de uso**

## **3.2. Especificação de caso de uso**

| **Nome do Caso de Uso:** |  |
| --- | --- |
| **Ator (s):** |  |
| **Objetivo:** |  |
| **Fluxo Básico** |  |
| **Fluxo Alternativo** |  |
| **Pré-Condições** |  |
| **Pós-Condições** |  |

# 4. MODELO DE CLASSES

## **4.1. Modelo de Domínio ( Fase de análise)**

*[Inserir o Modelo de Classes em um nível adequado de abstração que permita entendimento completo da solução proposta. Especificar os atributos e os relacionamentos]*

## **4.2. Diagrama de Classe (Fase de projeto)**

*[Inserir o Modelo de Classes em um nível adequado de abstração que permita entendimento completo da solução proposta. Especificar os atributos e os métodos.]*

### **Artefato 08: Diagrama de classe detalhado.**

* **4.1. Modelo de Domínio (Fase de Análise)**

*Representa os principais conceitos do sistema e como eles se relacionam, sem foco na implementação.*

* ***Classes e Atributos Principais:***
* ***Usuário***
  + *nome*
  + *email*
  + *senha*
  + *tipo (cliente, administrador)*
* ***Relacionamentos***
* *Usuário realiza muitos Orçamentos*

# 5. DIAGRAMA DE SEQUÊNCIA

*[Expressar os aspectos dinâmicos da solução utilizando Diagramas de Seqüência ou de Colaboração.]*

### **Artefato 09: Diagrama de sequência para três casos de uso.**

# 6. DIAGRAMA DE MÁQUINA

*[Incluir todos os diagramas de estados necessários, indicando as situações e métodos que permitem a mudança de estados. Os métodos devem ser consistentes com o diagrama de classes.]*

### **Artefato 10: Diagrama de máquina de estados para três situações importantes.**

# 7. CARACTERÍSTICAS OPERACIONAIS DO SOFTWARE

## **7.1. Ambiente operacional**

*[Descrever brevemente o ambiente de hardware e software necessário para o funcionamento do sistema. Pode-se ter uma representação gráfica da arquitetura de software e hardware se necessário, ou somente descritiva.*

*Utilizar o diagrama de implantação]*

### **Artefato 11: Diagrama de implantação.**

## **7.2. Arquitetura do software**

*Descrever como o sistema irá funcionar, em que plataforma, web, desktop, banco de dados, ambiente distribuído, acesso via telefone etc.*

*Utilizar um diagrama que apresente as camadas do sistema, principalmente que demonstre como o conceito de MVC (Model-View-Controller) está sendo utilizado. A UML oferece o Diagrama de Componentes ou Pacotes que permitem a criação de layers para separação dos componentes, conforme a estrutura MVC.*

*Deve-se representar no Diagrama de Componentes os principais componentes a serem gerados no projeto, bem como os componentes de terceiros que serão empregados. Incluir a versão dos componentes externos facilita posteriormente a manutenção e distribuição do sistema.*

### **Artefato 11: Diagrama de componentes e pacotes.**

## **7.3. Subsistemas e Componentes**

*Caso o sistema seja composto por subsistemas, eles podem ser melhor representados nesta secção. Recomenda-se utilizar os diagramas de Componentes ou de Pacotes para cada um deles.*

*O diagrama de componentes indica para a equipe técnica quais componentes devem fazer parte do sistema proposto e quais já estão disponíveis na arquitetura atual. Identificar adequadamente no artefato.*

### **Artefato 12: Diagrama de subsistemas ( se necessário).**

### **Artefato 13: Demais diagramas vistos em sala de aula.**

# 8. RISCOS

*[Identificar possíveis riscos associados à alternativa. Apresentar a tabela de riscos do projeto contendo obrigatoriamente as seguintes colunas, ordenadas em ordem decrescente de SEVERIDADE (do mais grave para o menos grave):*

* *Identificação do Risco (numeração seqüencial)*
* *Descrição do Risco (texto descrevendo o risco)*
* *Probabilidade: probabilidade de ocorrência do risco. Evitar percentuais, dando preferência ao uso de escalas (Ex.: 1=baixa, 2=média e 3=alta). As escalas facilitam as análises, pois é impossível prever com exatidão a probabilidade de um risco ocorrer.*
* *Impacto: impacto causado ao projeto caso o risco venha a ocorrer. Seguir as mesmas orientações da probabilidade, ou seja, usar escala (Ex.: 1=baixo, 2=médio e 3=alto).*
* *Severidade: indicador que permite classificar os riscos entre si. Pode ser definida uma fórmula que considere probabilidade e impacto com pesos diferenciados, no entanto, S=P\*I já é o suficiente.*
* *Ação de Prevenção: ação que visa reduzir a probabilidade do risco ocorrer. Esta ação precisa efetivamente ser executada para que se reduzam as chances do risco ocorrer.*
* *Ação de Contingência: ação que visa reduzir os impactos caso o risco ocorra. Esta ação deverá ser disparada caso um risco se torne efetivamente um problema que vai comprometer o sucesso do projeto.*

# 9. CRONOGRAMA

*[Apresentar um cronograma detalhado de atividades, com etapas do desenvolvimento da solução, datas e recursos envolvidos. Utilizar periodicidade semanal. Prever a utilização do tempo de forma realista, considerando semana de provas, férias e feriados. Avaliar, também de forma realista, o domínio do aluno em relação à compreensão do problema, da tecnologia e de quaisquer outros aspectos que possam influir no tempo do projeto. Considerar os marcos oficiais divulgados no Compromisso Pedagógico e também as atividades intermediárias necessárias para cada caso.]*

# 10. CONCLUSÕES

*[Resgatar os tópicos anteriores, revendo o panorama de inserção do projeto e voltando a analisar a viabilidade do projeto em função do progresso até o momento e da revisão dos riscos.]*