Análise e Especificação do Sistema de

Automatização

IA Maker para organização de projetos

|  |  |
| --- | --- |
| VERSÃO: 0.1 | DATA DE REVISÃO: 22/07/2025 |

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

### Espaços makers são ambientes colaborativos que estimulam a criatividade e a inovação, permitindo que ideias se transformem em protótipos e soluções tecnológicas. Contudo, um desafio recorrente nesses espaços é a falta de registro sistemático e de ferramentas que organizem e orientem os projetos em andamento. Essa carência de documentação e acompanhamento reduz a capacidade de aprendizado coletivo, dificulta a continuidade dos trabalhos e limita o impacto social das iniciativas.

### A aplicação de agentes inteligentes pode oferecer uma solução prática para esse cenário. Ao integrar um sistema de Inteligência Artificial que organize, registre e oriente o fluxo dos projetos, torna‑se possível criar um histórico estruturado, fornece recomendações baseadas em dados anteriores e até mesmo automatizar relatórios de progresso. Com isso, os participantes podem focar no desenvolvimento e reduzir o tempo gasto em tarefas administrativas.

### Este trabalho tem como proposta o desenvolvimento de um agente de IA local, operando no contexto do Espaço Maker, dedicado a registrar, organizar e orientar projetos de forma autônoma. O objetivo é melhorar a gestão interna, aumentar a produtividade dos participantes e criar uma base de conhecimento contínua, acessível e sustentável.

## Alternativas existentes

### Plataformas de gestão de projetos com IA integrada

#### **Notion**

O Notion é um aplicativo de prganização e gestão de tarefas, com capacidade colaborativa. Pode ser utilizado com funções de inteligência artificial para resumir textos organizar layouts, agendas e documentação. No entanto, essa ferramenta em espaços colaborativos maiores é paga e suas interações de IA limitadas com armazenamento estritamente em nuvem.

#### **ClicUp**

### ClicUp é uma plataforma de gerenciamento de tarefas e projetos que oferece um ambiente colaborativo completo, permitindo a criação de tarefas, fluxos de trabalho personalizados, dashboards relatórios de desempenho. Conta com recursos de inteligência artificial para auxiliar na descrição de atividades, priorização de tarefas e geração de resumos automáticos. Entretanto, apesar de ser bastante flexível e poderosa, suas funções de IA também estão limitadas ao armazenamento em nuvem, e os planos com recursos avançados requerem assinatura paga.

**Monday**

**Monday.com** é uma ferramenta de gestão de projetos focada em visualização de processos e integração com diferentes serviços. Oferece quadros personalizáveis, automações internas e um assistente de IA que ajuda na criação de relatórios, no acompanhamento de prazos e na organização de informações. Assim como o ClickUp, o Monday.com funciona principalmente em ambiente online e seus recursos de inteligência artificial dependem de planos pagos, além de não oferecer suporte nativo para operação off‑grid ou armazenamento totalmente local, o que limita seu uso em projetos que exigem maior controle sobre os dados.

### Soluções open-source ou locais

#### **OpenProject**

**OpenProject** é uma ferramenta de gerenciamento de projetos de código aberto que pode ser executada em um servidor local, garantindo controle total sobre dados e configurações. Ela oferece funcionalidades como registro de tarefas, cronogramas, gerenciamento de prazos e colaboração entre equipes. Por ser open-source, elimina custos de licença e permite personalizações conforme a necessidade do espaço maker. Contudo, não possui integração nativa com inteligência artificial, exigindo desenvolvimento adicional ou integração de agentes externos para alcançar funções avançadas, como geração de relatórios inteligentes ou recomendações automáticas.

#### **Redmine**

**Redmine** é uma plataforma também open-source voltada para gestão de projetos, muito utilizada em times de desenvolvimento de software, mas flexível o suficiente para outros tipos de projeto. Permite controle de tarefas, acompanhamento de progresso, criação de campos personalizados e histórico detalhado de atividades, tudo isso podendo ser instalado e operado localmente. Apesar de oferecer uma base sólida para organização e registro, assim como o OpenProject, não conta com IA integrada, demandando customização ou integração de modelos inteligentes para alcançar um agente que auxilie ativamente na organização dos projetos.

**Taiga**

**Taiga** é uma solução open-source focada em metodologias ágeis (Scrum e Kanban), ideal para equipes criativas e colaborativas. Pode ser hospedada localmente, o que garante autonomia sobre o armazenamento e maior privacidade. Oferece recursos como gestão visual de tarefas, sprints e acompanhamento do fluxo de trabalho. Assim como as demais ferramentas citadas, não possui nativamente funções de IA, mas fornece APIs e um ecossistema aberto que facilitam a integração com agentes inteligentes desenvolvidos para dar suporte à documentação e à orientação de projetos.

## 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 e acessível 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** | Criar um novo projeto, definindo nome, descrição e participantes. | **Membro Maker** | **Alta** |
| **RF02** | Ler mensagens e histórico de interações do projeto. | **LLM(Rag) + Banco de dados** | **Alta** |
| **RF03** | Conversar em tempo real com o modelo de IA integrado ao sistema. | **Cliente + LLM** | **Alta** |
| **RF04** | Gera documentação automática do projeto em andamento. | **LLM** | **Alta** |
| **RF05** | Consulta status do projeto e progresso das tarefas. | **Projetista** | **Média** |
| **RF06** | Edita informações de um projeto existente | **Projetista** | **Alta** |
| **RF07** | Visualiza relatórios de energia e desempenho do sistema. | **Admin** | **Alta** |
| **RF08** | Exporta dados do projeto (PDF, CSV ou outro formato. | **Projetista** | **Alta** |
| **RF09** | Recebe alertas/notificações sobre atualizações ou prazos. | **Projetista** | **Alta** |
| **RF10** | Atribui tarefas a membros do projeto | **Projetista** | **Média** |
| **RF11** | Realiza busca por projetos, mensagens ou documentos específicos. | **Projetista** | **Média** |
| **RF12** | Resume automaticamente reuniões ou registros longos, gerando tópicos e lista de ações sugeridas. | **LLM** | **Alta** |
| **RF13** | S**ugere próximos passos para o projeto** com base no histórico e no status atual das tarefas. | **LLM(Rag) + Banco de dados** | **Alta** |
| **RF14** | Detecta inconsistências ou informações duplicadas entre as tarefas e projetos e alertar o projetista. | **LLM** | **Alta** |
| **RF15** | Prioriza tarefas automaticamente, propondo uma ordem de execução com base em prazos e dependências. | **LLM** | **Alta** |
| **RF16** | Cria etiquetas ou categorias automaticamente ao analisar descrições de tarefas e documentos. | **LLM** | **Média** |
| **RF17** | Realiza Análise de sentimento das mensagens trocadas, indicando riscos de comunicação ou engajamento. | **LLM** | **Alta** |
| **RF18** | Fornece respostas a perguntas diretas sobre o projeto, usando o histórico e dados armazenados. | **LLM** | **Alta** |

## 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 sistema de IA Maker para organização de projetos será desenvolvido para atuar de forma local (on‑premises) dentro do Espaço Maker, com possibilidade de operação off‑grid utilizando energia proveniente de fonte solar. O objetivo principal é gerenciar e automatizar o fluxo de projetos colaborativos, oferecendo recursos de documentaçãoautomática, análise de dados históricos, priorização de tarefas e recomendações inteligentes, tudo integrado a uma interface acessível aos usuários.

O sistema deverá:

* Registrar e organizar projetos, permitindo criação, edição, categorização e atribuição de tarefas.
* Fornecer interação em linguagem natural com um agente de IA, capaz de responder perguntas, gerar resumos, sugerir próximos passos e alertar inconsistências.
* Gerar relatórios e documentação automática do andamento dos projetos (em formatos como PDF ou CSV).
* Permitir envio e recebimento de comandos via Discord, facilitando uso educacional e remoto.
* Operar com controle local dos dados, garantindo privacidade, e permitir integração com soluções open‑source existentes

| **O projeto entregará** |
| --- |
| Sistema local para criação, edição e organização de projetos no Espaço Maker. |
| Integração com LLM para respostas, sugestões e geração automática de relatórios. |
| Exportação de dados e documentação automática em PDF/CSV. |
| Monitoramento energético em tempo real. |
| Interface de interação via Discord para comandos e notificações. |

| **O projeto não entregará** |
| --- |
| Sistema 100% off-grid com geração solar |
| Integração nativa com ferramentas 100% em nuvem (Notion, ClickUp, Monday). |
| Desenvolvimento de novos modelos de IA do zero (serão usados modelos existentes). |
| Garantia de operação fora do ambiente do Espaço Maker ou sem infraestrutura local. |
| Suporte a sensores ou dispositivos não especificados no levantamento inicial. |

# 3. DIAGRAMA DE CASO DE USO

## 3.1. Diagrama de caso de uso

## 3.2. Especificação de caso de uso

|  |  |
| --- | --- |
| **Nome do Caso de Uso:** | Criar um novo projeto |
| **Ator (s):** | Admin, Projetista, Membro Maker, Discord, Banco de dados, LLM |
| **Objetivo:** | Permitir que um usuário (projetista ou membro) crie um novo projeto no sistema, registrando nome, descrição, participantes e configurando as primeiras tarefas. |
| **Fluxo Básico** | 1. O usuário acessa a interface principal do sistema. 2. Seleciona a opção **“Criar Novo Projeto”**. 3. Informa O sistema valida os dados inseridos. 4. O sistema registra o projeto no banco de dados e confirma a criação. 5. O sistema disponibiliza o projeto na lista geral e envia confirmação ao usuário. |
| **Fluxo Alternativo** | * Dados obrigatórios não preenchidos → o sistema solicita correção antes de salvar. * Banco de dados indisponível → o sistema exibe mensagem de erro e registra log de falha. |
| **Pré-Condições** | * O usuário está autenticado no sistema. * O banco de dados e a LLM estão ativos. |
| **Pós-Condições** | * O novo projeto está salvo e visível para os membros autorizados. * O sistema gera automaticamente um ID único e cria um histórico inicial de registro. |

### 

# 4. MODELO DE CLASSES

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

#### **Classes e Atributos Principais**

**Projetista**

* nome : String
* email : String
* senha : String
* tipo : Enum (membroMaker, admin)

**Projeto**

* nomeProjeto : String
* descricao : String
* dataCriacao : DateTime
* status : Enum (ativo, finalizado)

**Tarefa**

* titulo : String
* descricao : String
* prioridade : Enum (alta, média, baixa)
* prazo : DateTime
* status : Enum (pendente, concluída)

**Relatório**

* tipo : String (PDF, CSV)
* dataGeracao : DateTime
* conteudo : String

## 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.]*

# 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.]*