**EAP – Estrutura Analítica do Projeto**

A Estrutura Analítica do Projeto (EAP) é uma ferramentaria de gerenciamento de projetos que divide o trabalho em partes menores e mais gerenciáveis. A EAP é uma representação gráfica da decomposição do trabalho a ser executado em um projeto que permite a visualização de todas as entregas que o projeto deve realizar. A EAP é uma ferramenta fundamental para o planejamento e controle de projetos, pois permite a visualização de todas as entregas que o projeto deve realizar.

Este documento ficará estruturado da seguinte forma

* Análise de Riscos
* Escopo do projeto - Requisitos
* Cronograma - Gráfico GANTT
* CPM - Caminho Crítico
* Custos - Orçamento do Projeto

**Análise de Riscos – SWOT**

Análise de riscos utilizando a Matriz SWOT (Strengths, Weaknesses, Opportunities, Threats), uma sigla em inglês que significa Forças, Fraquezas, Oportunidades e Ameaças. FOFA em português.

*Figura 1 - Analise de risco SWOT do projeto.*

Uma imagem contendo Interface gráfica do usuário

Descrição gerada automaticamente

*Fonte: Feito pelo Autor, 2024.*

A análise conclui que o projeto é viável pois as forças e oportunidades são maiores do que as ameaças e fraquezas.

**Escopo do projeto – Requisitos**

O propósito deste documento é definir o escopo do projeto a partir das regras de negócio estabelecidas, que foram elencadas de diversas formas e fontes.

**Regras de negócio**

* **[RN001]** Para usuários que desejam testar o sistema antes de realizar o cadastro, foi criado a opção de entra como visitante.
* **[RN002]** Para utilizar o sistema o usuário precisa realizar um cadastro através de seu nome, e-mail e senha.
* **[RN003]** Após o cadastro, o usuário terá a disponibilidade de acessar os dados do aplicativo em qualquer aparelho móvel através do login.
* **[RN004]** O usuário poderá cadastrar, remover e editar eventos e lembretes na aba do calendário.
* **[RN005]** O usuário será capaz de administrar uma lista de itens, adicionando, removendo ou editando os itens dela.
* **[RN006]** O usuário conseguirá, através da aba de atividades diárias verificar todas as atividades que ele tem para determinado dias da semana, além de poder adicionar, editar e remover essas atividades a qualquer momento.
* **[RN007]** O sistema utilizara o MySQL como banco de dados, e será divido em API, front-end e mobile.
* **[RN008]** A API back-end deve ser desenvolvida em NodeJs, e utilizando o ORM Prisma para o banco de dados, além da presença de verificação e autenticação por JWT.
* **[RN009]** O front-end web deve ser desenvolvido utilizando HTML, CSS e JavaScript.
* **[RN010]** O front-end mobile deve ser desenvolvido utilizando a biblioteca do react.
* **[RN011]** O sistema deve ser desenvolvido em português, com documentação seguindo as normas ABNT

**Requisitos funcionais**

Os requisitos funcionais do sistema EPONA foram obtidos com base nas regras de negócio e estão elencados a seguir de forma ordenada a partir da sigla RF (Requisito Funcional) seguida da ordem do requisito, conforme exemplo a seguir: [RF001]

De acordo com a metodologia Ágil XP cada requisito é obtido através de uma breve história, que será ilustrada neste documento através da UML DCU (Diagrama de Casos de Uso).

Também será atribuído um nível de prioridade para o desenvolvimento entre os três: essencial, importante e desejável.

[RF001] **Cadastrar-se** – O sistema deve permitir o cadastro de usuários.

*Figura 2 - Requisito funcional de cadastro.*

Diagrama

Descrição gerada automaticamente

*Fonte: Feito pelo Autor, 2024.*

Prioridade: [x] Essencial [ ] Importante [ ] Desejável

Referência: [RN001] [RN002]

[RF002] – **Login** - O sistema deve permitir o login dos usuários cadastrados aplicando a autenticação JWT

*Figura 3 - Requisito funcional de login.*

Diagrama

Descrição gerada automaticamente

*Fonte: Feito pelo Autor, 2024.*

Prioridade: [x] Essencial [ ] Importante [ ] Desejável

Referência: [RN003]

[RF003] **Remover** – O sistema deve permitir que o usuário remova quaisquer itens dentre as opções

*Figura 4 - Requisito funcional de remover.*

Diagrama

Descrição gerada automaticamente

*Fonte: Feito pelo Autor, 2024.*

Prioridade: [x] Essencial [ ] Importante [ ] Desejável

Referência: [RN004] [RN005] [RN006]

[RF004] **Editar** – O sistema deve permitir que o usuário edite quaisquer itens dentre as opções

*Figura 5 - Requisito funcional de editar.*

Diagrama

Descrição gerada automaticamente

*Fonte: Feito pelo Autor, 2024.*

Prioridade: [x] Essencial [ ] Importante [ ] Desejável

Referência: : [RN004] [RN005] [RN006]

[RF005] **Adicionar** – O sistema deve permitir que o usuário adicionar quaisquer itens dentre as opções

*Figura 6 - Requisito funcional de adicionar.*

Diagrama

Descrição gerada automaticamente

*Fonte: Feito pelo Autor, 2024.*

Prioridade: [x] Essencial [ ] Importante [ ] Desejável

Referência: : [RN004] [RN005] [RN006]

[RF006] **Exibir Atividades** – O sistema deve permitir ao usuário a visualizar todas as suas atividades previamente adicionadas

*Figura 7 - Requisito funcional de exibir atividades.*

Diagrama

Descrição gerada automaticamente *Fonte: Feito pelo Autor, 2024.*

Prioridade: [x] Essencial [ ] Importante [ ] Desejável

Referência: : [RN006]

[RF007] **Exibir eventos/lembretes** – O sistema deve permitir ao usuário a visualizar todos os seus lembretes do calendário

*Figura 8 - Requisito funcional de exibir eventos e lembretes*.Diagrama

Descrição gerada automaticamente

*Fonte: Feito pelo Autor, 2024.*

Prioridade: [x] Essencial [ ] Importante [ ] Desejável

Referência: : [RN004]

[RF008] **Exibir lista** – O sistema deve permitir ao usuário a visualizar todos os itens da lista

*Figura 9 - Requisito funcional de exibir lista*.Diagrama

Descrição gerada automaticamente *Fonte: Feito pelo Autor, 2024.*

Prioridade: [x] Essencial [ ] Importante [ ] Desejável

Referência: : [RN005]

[RF009] **Escolher opção do menu –** Quando logado o usuario deve ter acesso a um menu, com as opções de ir para as atividades, calendário ou lista, além de voltar a sua Home

*Figura 10 - Requisito funcional de selecionar as opções de páginas através de um menu.*Diagrama

Descrição gerada automaticamente

*Fonte: Feito pelo Autor, 2024.*

Prioridade: [ ] Essencial [ ] Importante [x] Desejável

Referência: ?

**Requisitos não funcionais**

Os requisitos não funcionais do sistema EPONA foram obtidos com base em pesquisas de mercado realizadas através da internet, observação direta das tecnologias de implantação em nuvem atuais. Para a classificação a sigla NF (Não Funcional) foi utilizada seguida da ordem do requisito, conforme exemplo a seguir:

**[NF001]** Será observada a UX (User Experience) e a UI (User Interface) para garantir uma boa experiência do usuário.

Prioridade: [x] Essencial [ ] Importante [ ] Desejável  
Referência: [RN009]

**[NF002]** A questão de segurança será observada com a utilização de autenticação JWT e a utilização de HTTPS.

Prioridade: [x] Essencial [ ] Importante [ ] Desejável  
Referência: [RN008]

**[NF003]** O desempenho do sistema será observado com a utilização de um banco de dados relacional e a utilização de um ORM para otimizar as consultas.

Prioridade: [ ] Essencial [x] Importante [ ] Desejável  
Referência: [RN008]

**[NF004]** Mobile desenvolvido com o framework react, consumindo a API Back-End.

Prioridade: [ ] Essencial [ ] Importante [x] Desejável  
Referência: [RN010]

**Cronograma – Gráfico Gantt**

*Figura 11 - Requisito funcional de selecionar as opções de páginas através de um menu* **Linha do tempo, Gráfico de cascata

Descrição gerada automaticamenteTabela

Descrição gerada automaticamente**

*Fonte: Feito pelo Autor, 2024.*

**CPM – Caminho crítico**

Para a elaboração do cronograma inicial acima foram aplicadas as ferramentas PERT e CPM.

O PERT (Program Evaluation and Review Technique) e o CPM (Critical Path Method) são técnicas de gerenciamento de projetos que ajudam a planejar, agendar e coordenar tarefas dentro de um projeto. O PERT é usado principalmente em projetos de pesquisa e desenvolvimento, enquanto o CPM é usado para projetos de construção e manufatura, em nosso caso utilizamos o CPM pois o tempo para cada tarefa já foi estimado pela equipe e acordado entre os stakeholders no TAP (Termo de Abertura do Projeto).

A revisão do tempo das atividades está ilustrada na tabela a seguir:

Tabela

Descrição gerada automaticamente

**Caminho crítico**

O caminho crítico traçado na rede apresentada na imagem abaixo, calculado somando as atividades de maior duração em sequência, totalizando 24 dias

D3 + E3 = 6 dias

D3 + E3 + G4 + H3 + I1 + J10 = 24 dias

*Figura 12 – Diagrama de Pert e COM.*

Uma imagem contendo Forma

Descrição gerada automaticamente

*Fonte: Feito pelo Autor, 2024.*

**Orçamento**

**Estimativa de custos/ orçamento inicial**

|  |  |  |  |
| --- | --- | --- | --- |
| **Recurso** | **Hora** | **Valor Hora** | **Total** |
| Design | 5 | 30,00 | 150,00 |
| Back-end + BD | 15 | 30,00 | 450,00 |
| Front-end | 15 | 30,00 | 450,00 |
| Mobile | 15 | 30,00 | 450,00 |
| Integração FullStack | 10 | 30,00 | 300,00 |
| Documentação do Projeto | 10 | 30,00 | 300,00 |
| Testes | 10 | 30,00 | 300,00 |
| **Total: R$ 2400,00** | | | |

**Orçamento oficial por funcionalidade**

|  |  |  |  |
| --- | --- | --- | --- |
| **Requisito** | **Valor Tela** | **Funcionalidades** | **Total** |
| RF001 – Cadastro | 100,00 | 100,00 | 200,00 |
| RF002 – Login | 100,00 | 100,00 | 200,00 |
| RF003 – Remover | 50,00 | 50,00 | 100,00 |
| RF004 – Editar | 50,00 | 50,00 | 100,00 |
| RF005 – Adicionar | 50,00 | 50,00 | 100,00 |
| RF006 – Exibir atividades | 150,00 | 150,00 | 300,00 |
| RF007 – Exibir lembretes | 150,00 | 150,00 | 300,00 |
| RF008 – Exibir lista | 150,00 | 150,00 | 300,00 |
| RF009 – Menu | 100,00 | 100,00 | 200,00 |
| **Total R$ 2400,00** | | | |