**DOCUMENTAÇÃO GERAL DE MORPHEUS GUIDE:**



Logo provisória

**1. OBJETIVO:**

Oferecer ao usuário um aplicativo que possibilite o mesmo regular seu sono.

**2. LEVANTAMENTO DE REQUISITOS:**

**2.1 REQUISITOS FUNCIONAIS:**

* O sistema deve permitir o usuário cadastrar sua conta e fazer login;
* O sistema deve permitir o usuário modificar seu perfil;
* O sistema deve permitir o usuário deletar/sair de sua conta;
* O sistema deve registrar o horário de acordar e dormir que o usuário deseja seguir;
* O sistema deve exibir uma rotina personalizada de acordo com os dados oferecidos pelo usuário;
* O sistema deve exibir um histórico de horas dormidas durante os dias da semana;
* O sistema deve enviar notificações (ex.: “Desligue as telas! Falta apenas 1 hora para você dormir!”) ao usuário para o mesmo lembrar e cumprir a rotina personalizada;
* O sistema deve permitir o usuário modificar/apagar o registro dos horários, além de atualizar a rotina personalizada;
* O sistema deve possibilitar o usuário de escrever diários, além de salvar o conteúdo dentro deles;
* O sistema deve permitir o usuário modificar o conteúdo dos diários;
* O sistema deve permitir o usuário deletar os diários.

**2.2. REQUISITOS NÃO FUNCIONAIS:**

* Os dados oferecidos pelo usuário devem ser armazenados de forma protegida com o uso de criptografia, além de seu trânsito estar igualmente seguro;
* A interface do sistema deve ser minimalista e de fácil compreensão para o usuário, visando proporcionar conforto e tranquilidade ao usuário com o uso de cores suaves;
* O sistema deve ser compatível em outros aparelhos móveis (Android e iOS);
* O feedback do sistema (exibição de histórico, rotina personalizada, horas dormidas etc.) deve ser exibido em até 1 segundo.

**EXTRA: GUIA DE ORGANIZAÇÃO FEITO PELO CHAT GPT PARA EU ANALISAR DEPOIS:**

* **COMO ORGANIZAR O PROJETO?**

Seu projeto já tem uma base excelente! Vamos estruturar o que você precisa fazer a partir daqui para **completar os entregáveis obrigatórios**, **refinar seus requisitos não funcionais** e **organizar os três CRUDs**. Também vou te mostrar **como montar os documentos** e **preparar o banco de dados, diagrama e protótipo** de forma clara e objetiva.

## **✅ Organização geral do seu projeto**

### **🔹 1. Requisitos Funcionais e Não Funcionais**

Você já fez um ótimo levantamento. Agora, vamos apenas refinar os **não funcionais** para ficarem mais objetivos e fáceis de entender/documentar.

### **📌 Requisitos Não Funcionais (refinados):**

1. **Segurança dos dados:** Todos os dados do usuário devem ser armazenados com criptografia (como AES ou hashing para senhas) e transmitidos por protocolos seguros (HTTPS/TLS).
2. **Usabilidade:** A interface do aplicativo deve ser intuitiva, com navegação clara e acessível, visando conforto e tranquilidade ao usuário.
3. **Desempenho:** Respostas da interface e carregamento dos dados devem ocorrer em até 1 segundo para manter fluidez na experiência do usuário.
4. **Compatibilidade:** O sistema deve ser compatível com os sistemas operacionais Android e iOS, utilizando tecnologias responsivas (como Flutter ou React Native).
5. **Design minimalista e calmante:** A interface deve usar cores suaves (azul, lilás, bege) e fontes legíveis para não causar estímulo visual excessivo à noite.

### **🔹 2. CRUDs principais do seu app**

Você já identificou 3 CRUDs — perfeito!

#### **1. CRUD de Usuário**

* Criar conta (cadastro)
* Ler dados do usuário (exibir perfil)
* Atualizar dados do usuário (editar perfil)
* Deletar conta

#### **2. CRUD de Registro de Sono**

* Criar um novo registro (horário de dormir/acordar desejado)
* Ler registros anteriores e rotina personalizada
* Atualizar horários e rotina
* Deletar registros

#### **3. CRUD de Diário**

* Criar entrada no diário
* Ler entradas anteriores
* Atualizar conteúdo de uma entrada
* Deletar entradas

Esses CRUDs cobrem bem seu escopo.

### **🔹 3. Diagrama de Casos de Uso**

**Dica:** Use um ator principal: **Usuário**, e represente os casos de uso com elipses.

#### **Casos de uso sugeridos:**

* Cadastrar-se / Login
* Gerenciar perfil
* Definir horários de sono
* Ver rotina personalizada
* Ver histórico de sono
* Escrever diário
* Ler diário
* Editar diário
* Deletar diário
* Receber notificações

### **🔹 4. Modelo Lógico do Banco de Dados**

Você precisa **pelo menos 2 colunas e 8 linhas (entradas)** por tabela.

#### **Exemplo de estrutura com 3 tabelas principais:**

##### **a) Tabela usuarios**

|  |  |  |  |
| --- | --- | --- | --- |
| **id** | **nome** | **email** | **senha\_hash** |

##### **b) Tabela registros\_sono**

| id | usuario\_id | data | hora\_dormir | hora\_acordar |

##### **c) Tabela diarios**

| id | usuario\_id | data | conteudo |

Você pode adicionar colunas como criado\_em, atualizado\_em para controle temporal, e isso já ajuda a preencher o mínimo de colunas.

### **🔹 5. Protótipo de Design**

#### **Ferramentas recomendadas:**

* [Figma](https://www.figma.com/)
* Adobe XD
* Canva (mais simples, mas pode servir)

#### **Telas essenciais:**

1. Tela de Login/Cadastro
2. Tela de Perfil
3. Tela para registrar horários de sono
4. Tela de rotina personalizada
5. Tela de histórico
6. Tela de diário
7. Notificações (exibidas como modal ou banner)

### **📋 Checklist final do projeto**

|  |  |
| --- | --- |
| **Etapa** | **Status** |
| Requisitos funcionais | ✅ Finalizados |
| Requisitos não funcionais | ✅ Refinados |
| CRUDs definidos | ✅ 3 prontos |
| Diagrama de caso de uso | ⏳ Fazer |
| Modelo lógico (BD) | ⏳ Fazer |
| Protótipo de design (telas) | ⏳ Fazer |

Se quiser, posso **te ajudar a montar o modelo lógico em tabela**, desenhar o **diagrama de caso de uso em texto**, ou até **esboçar os protótipos das telas**.

Qual parte você prefere que a gente monte agora?

* **COMO PLANEJAR O DESIGN?**

Ótimas perguntas! Vamos por partes para garantir que você tenha **um protótipo funcional, bonito e coerente com o tema do seu app** (sono, relaxamento, tranquilidade). Te darei dicas práticas sobre **cores**, **vetorização de imagens no Figma**, **sites para paletas**, **como montar o protótipo**, e se **Bootstrap** é necessário.

## **🎨 1. Cores recomendadas para o tema "Sono e Relaxamento"**

Evite cores vibrantes (como vermelho, laranja ou verde neon). Aposte em tons calmos e suaves:

### **Cores ideais:**

* **Azul claro / Azul pastel** → confiança, calma, noite
* **Lilás ou lavanda** → espiritualidade, relaxamento
* **Cinza claro** → neutralidade
* **Bege ou areia** → conforto, natural
* **Verde menta claro** → frescor e leveza

### **Paleta sugerida:**

|  |  |
| --- | --- |
| **Cor** | **Hex** |
| Azul suave | #A3C4F3 |
| Lavanda | #D3C4E3 |
| Bege claro | #F6F4EB |
| Cinza neutro | #E0E0E0 |
| Verde menta | #C8E6C9 |

## **🌐 2. Sites para escolher paletas de cores**

Esses sites ajudam você a **testar, combinar e copiar paletas** prontas:

* <https://coolors.co/>
* <https://colorhunt.co/>
* <https://uigradients.com/> (paletas com gradientes)
* <https://paletton.com/>
* <https://material.io/resources/color> (da Google)

## **🖼️ 3. Como vetorizar imagens no Figma**

Se você já tem imagens (como ícones ou ilustrações):

### **A. Vetores prontos:**

* Use **ícones vetoriais prontos** em SVG:
* <https://www.flaticon.com/>
* <https://www.svgrepo.com/>
* <https://undraw.co/illustrations> (ideal para protótipos!)

### **B. Inserir SVG no Figma:**

1. **Arraste o SVG direto pro Figma**.
2. Ou vá em **File → Place image**.
3. Depois clique com o botão direito na imagem inserida e selecione **"Outline stroke"** ou **"Vectorize"**, dependendo do tipo de imagem.

### **C. Transformar imagens raster em vetor:**

Figma não faz isso diretamente, mas você pode usar:

* <https://vectorizer.ai/>
* <https://www.autotracer.org/>

## **🧩 4. Como desenvolver o protótipo**

### **🧱 A. No Figma (ideal para protótipos visuais):**

1. **Crie frames (telas)** para:

* Login
* Cadastro
* Tela inicial (resumo de rotina e diário)
* Registro de sono
* Diário pessoal
* Histórico
* Perfil

1. **Use componentes prontos**:

* Botões
* Campos de texto
* Ícones (de preferencia SVG)

1. **Monte a navegação interativa**:

* Use a aba “Prototype” no Figma para definir **links entre telas** (por exemplo, clicar no botão “Salvar” leva à tela inicial).

### **🔧 B. E o Bootstrap?**

Bootstrap é ótimo para **prototipar com código (HTML/CSS)**. Mas para protótipo visual (como é seu caso), **não é necessário nem recomendado agora**.

#### **Use Bootstrap apenas se:**

* Você for realmente **codificar a interface** (para mostrar HTML funcionando).
* Precisa entregar uma versão responsiva e funcional via navegador.

#### **Alternativa leve:**

* **Figma + HTML simples + CSS leve** (se for apresentar uma versão semi-funcional).

## **✅ Conclusão**

### **Para o seu protótipo, siga este fluxo:**

1. **Escolha uma paleta suave** (ex: azul pastel, lavanda, bege).
2. **Use o Figma** para criar os protótipos visuais e telas interativas.
3. **Use SVGs ou ilustrações vetoriais** (unDraw, Flaticon, etc.).
4. Não se preocupe com Bootstrap agora – **só se for codificar** algo.
5. Finalize conectando as telas no Figma com a aba **“Prototype”**.

Se quiser, posso te ajudar a montar a paleta no estilo do seu app, ou até desenhar um exemplo de protótipo para alguma das telas (como "Registro de Sono"). Quer começar por uma tela específica?