

1. Introdução

O AvatarSDK é uma plataforma baseada em IA que gera avatares 3D realistas a partir de uma foto. Este estudo de arte explora as capacidades do AvatarSDK e sua aplicação para criar um avatar completo com rosto, corpo e personalização.

2. Tipos de Avatares Suportados

O AvatarSDK oferece diferentes estilos de avatares:

- **Realista:** Gera um modelo detalhado com textura de alta qualidade.
- **Cartoon/Semi-realista:** Modelos estilizados, ideais para aplicações mais leves.
- **Low Poly:** Modelos otimizados para dispositivos móveis ou jogos.

3. Processo de Geração do Avatar

- **Upload da Foto:** O usuário fornece uma imagem frontal do rosto.
- **Processamento com IA:** A plataforma analisa a imagem e gera um modelo 3D.
- **Renderização:** O avatar é gerado e pode ser exportado em formatos como .glb, .fbx, e .obj.
- **Personalização:** Opções para editar cabelo, expressão e textura.

4. Possibilidades de Integração

- **Web e Mobile:** Suporte para Unity, Unreal Engine, Three.js, Babylon.js.
- **Exportação para Modelagem 3D:** Possibilidade de uso no Blender e Maya.
- **Aplicabilidade:** Moda virtual, realidade aumentada, experiências interativas.

5. Requisitos do Sistema

1. Funcionais

- O sistema deve permitir o upload de uma foto para geração do avatar.
- O backend deve processar a imagem e gerar um modelo 3D do rosto.
- O frontend deve exibir o avatar gerado ao usuário.

- O usuário deve poder inserir altura, peso e gênero para modelagem do corpo.
- O sistema deve integrar um motor 3D para visualização interativa do avatar.
- O avatar deve ser exportável em formatos padrão (.glb, .fbx, .obj).
- A plataforma deve oferecer opções de personalização de cabelo, pele e roupas.

2. Não Funcionais

- O sistema deve processar a imagem e gerar o avatar em até 10 segundos.
- O backend deve ser escalável para suportar múltiplos usuários simultaneamente.
- A segurança dos dados deve ser garantida, protegendo as imagens dos usuários.
- O frontend deve ser responsivo para dispositivos móveis e desktop.
- A renderização 3D deve ser otimizada para evitar latência excessiva.

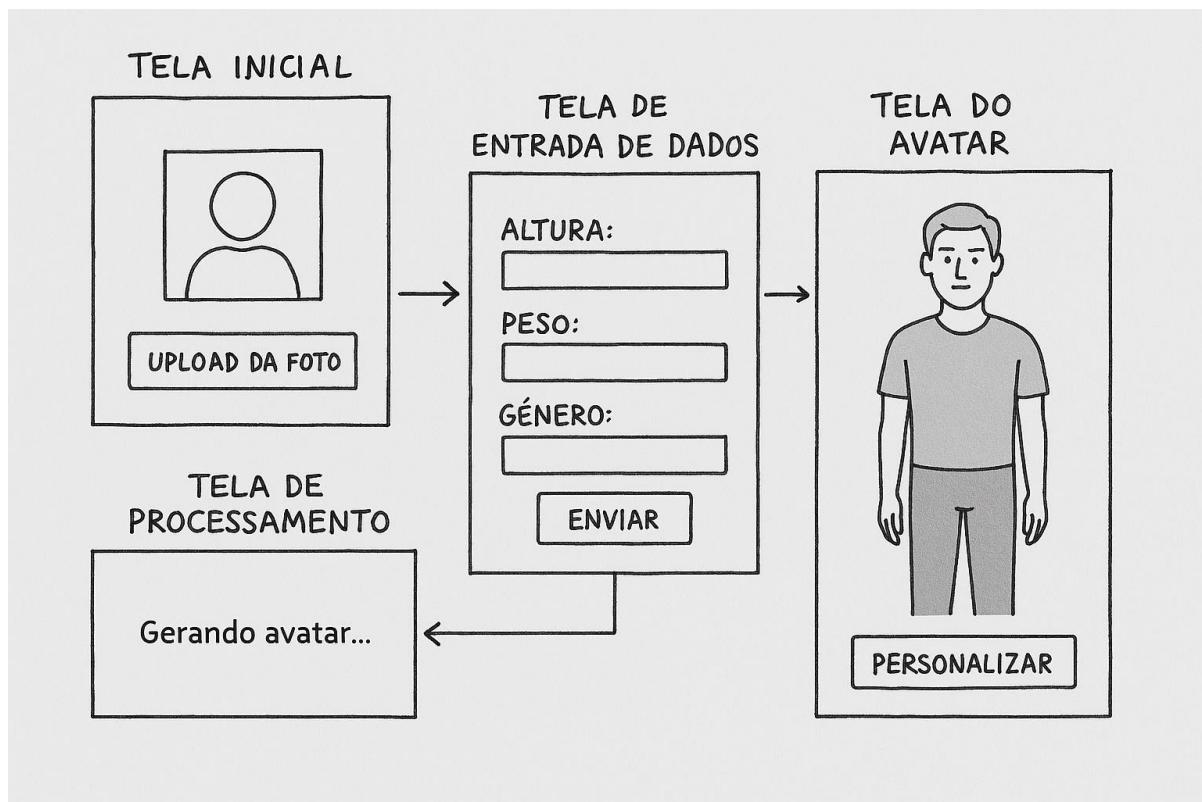
3. Tabela de Requisitos

ID	Tipo	Descrição	Prioridade
R1	Funcional	Upload de foto para geração do avatar	Alta
R2	Funcional	Processamento de imagem e criação do modelo 3D	Alta
R3	Funcional	Exibição do avatar no frontend	Alta
R4	Funcional	Inserção de altura, peso e gênero para modelagem do corpo	Média
R5	Funcional	Exportação do avatar em diferentes formatos	Média
R6	Não Funcional	Geração do avatar em até 10 segundos	Alta
R7	Não Funcional	Suporte para multiplataforma (mobile e desktop)	Média
R8	Não Funcional	Segurança dos dados e proteção de imagens	Alta

6. Implementação no Projeto

- O AvatarSDK será usado para gerar o rosto do avatar.
- O corpo será modelado proporcionalmente baseado em altura, peso e gênero.
- O avatar será exibido em uma plataforma interativa com opções de personalização.

7. protótipo de baixa fidelidade



8. Tecnologias Utilizadas

Backend (Servidor e Processamento de IA)

Linguagem: Python

- ◆ Framework: FastAPI (API rápida e escalável)
- ◆ Bibliotecas:

- requests – Para interagir com APIs externas (AvatarSDK)
- Pillow – Para manipulação de imagens
- opencv-python – Para pré-processamento de imagens
- numpy – Para cálculos matemáticos avançados

- torch + deepface – Para análise facial (se necessário)
- python-dotenv – Para gerenciamento de variáveis de ambiente

◆ **APIs:**

- **AvatarSDK API** – Geração do rosto 3D
 - **Custom Body Model API** – Modelagem do corpo baseado em altura, peso e gênero
- ◆ **Banco de Dados:**
- PostgreSQL ou MongoDB (armazenamento dos avatares e perfis)
-

Frontend (Interface do Usuário)

Web: React.js + Three.js para visualização 3D

◆ **Bibliotecas e Ferramentas:**

- Three.js – Para renderização do avatar 3D no navegador
- Babylon.js – Alternativa para gráficos interativos
- Axios – Para comunicação com o backend
- TailwindCSS – Para estilização rápida e responsiva

9. Desenvolvimento do Projeto

Desenvolvimento do Backend

Aqui, o foco será criar a API para processar as imagens, integrar com o AvatarSDK, e gerar o avatar do usuário.

- **Configuração do ambiente de backend (FastAPI e banco de dados).**
- **Criação de endpoints para upload de foto, geração do avatar e modelagem do corpo.**
- **Integração com a API AvatarSDK** para gerar o rosto 3D do avatar.
- **Desenvolver a lógica para modelagem do corpo** com base em altura, peso e gênero.
- **Armazenamento de dados no banco de dados** (dados do usuário e avatar).
- **Testes unitários** para garantir que cada parte do backend funcione corretamente.

Entregáveis:

- API de backend (FastAPI)
 - Integração com AvatarSDK
 - Lógica de modelagem do corpo
 - Banco de dados configurado
 - Testes do backend
-

Fase de Desenvolvimento do Frontend (*Semana 5-7*)

Agora que o backend está pronto, vamos focar no frontend que interage com o usuário e exibe o avatar gerado.

- **Configuração do ambiente frontend (React.js + Three.js).**
- **Desenvolver a interface para upload da foto** e entrada de dados (altura, peso, gênero).
- **Integração com a API do backend** para enviar a foto e os dados, e receber o avatar gerado.
- **Exibição do avatar 3D no frontend** usando Three.js.
- **Adição de funcionalidades de personalização** de avatar (cabelo, roupas, expressões).
- **Responsividade da interface** para garantir que funcione bem em dispositivos móveis e desktop.

Entregáveis:

- Interface de upload de foto e entrada de dados
- Exibição do avatar 3D no frontend
- Funcionalidades de personalização do avatar
- Testes do frontend

10. Diagramas

Diagrama de Sequência (Geração do Avatar)

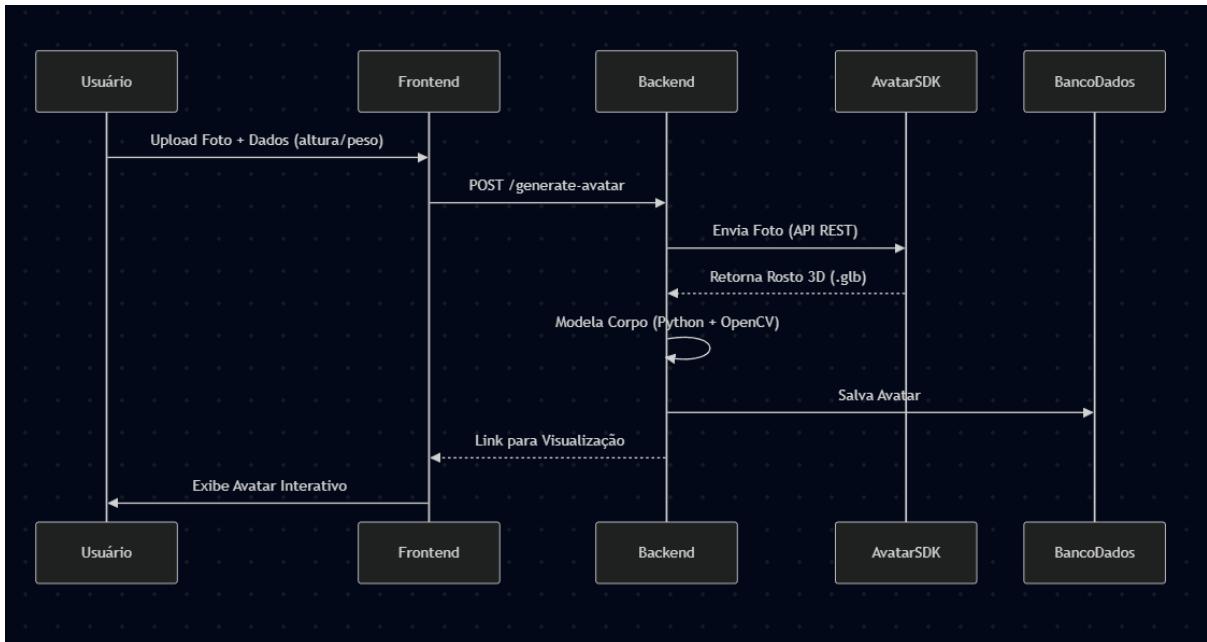


Diagrama de Casos de Uso

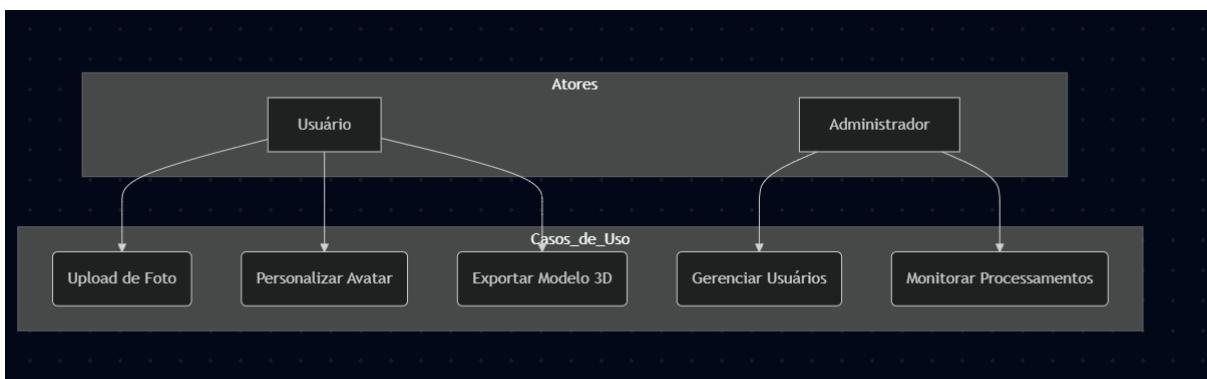


Diagrama de Arquitetura do Sistema

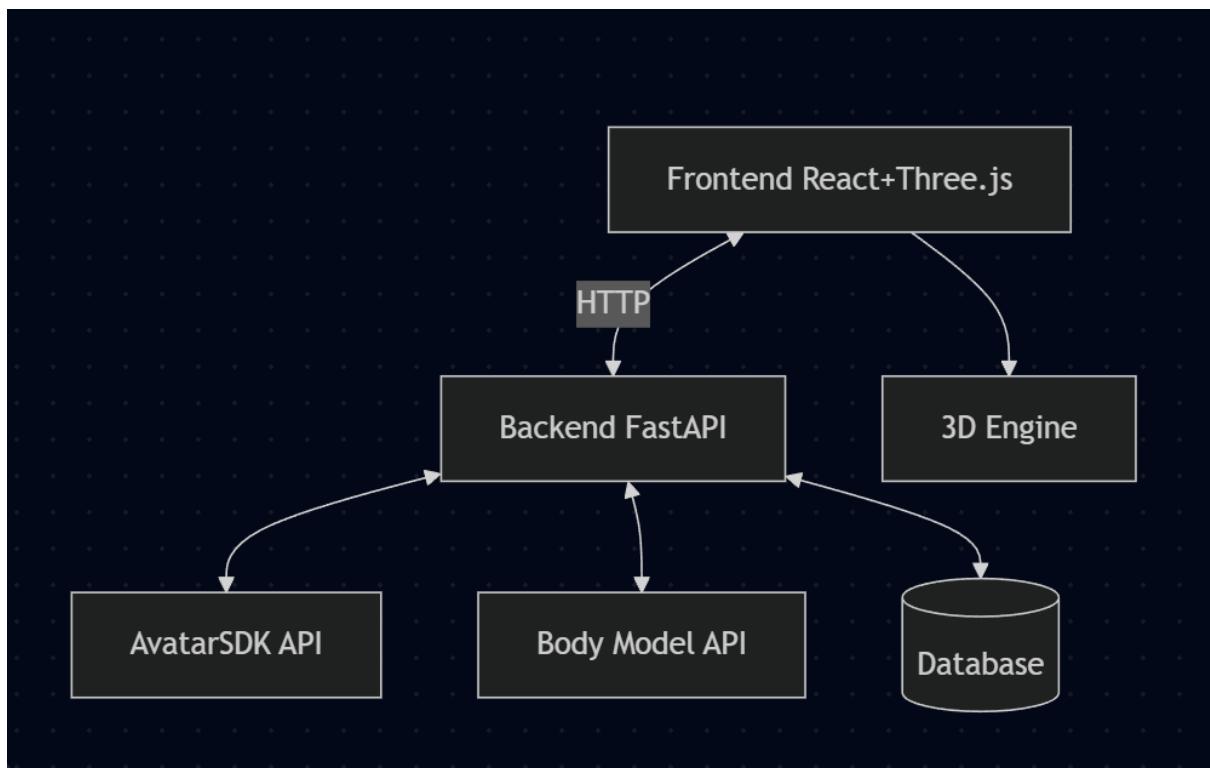


Diagrama de Componentes

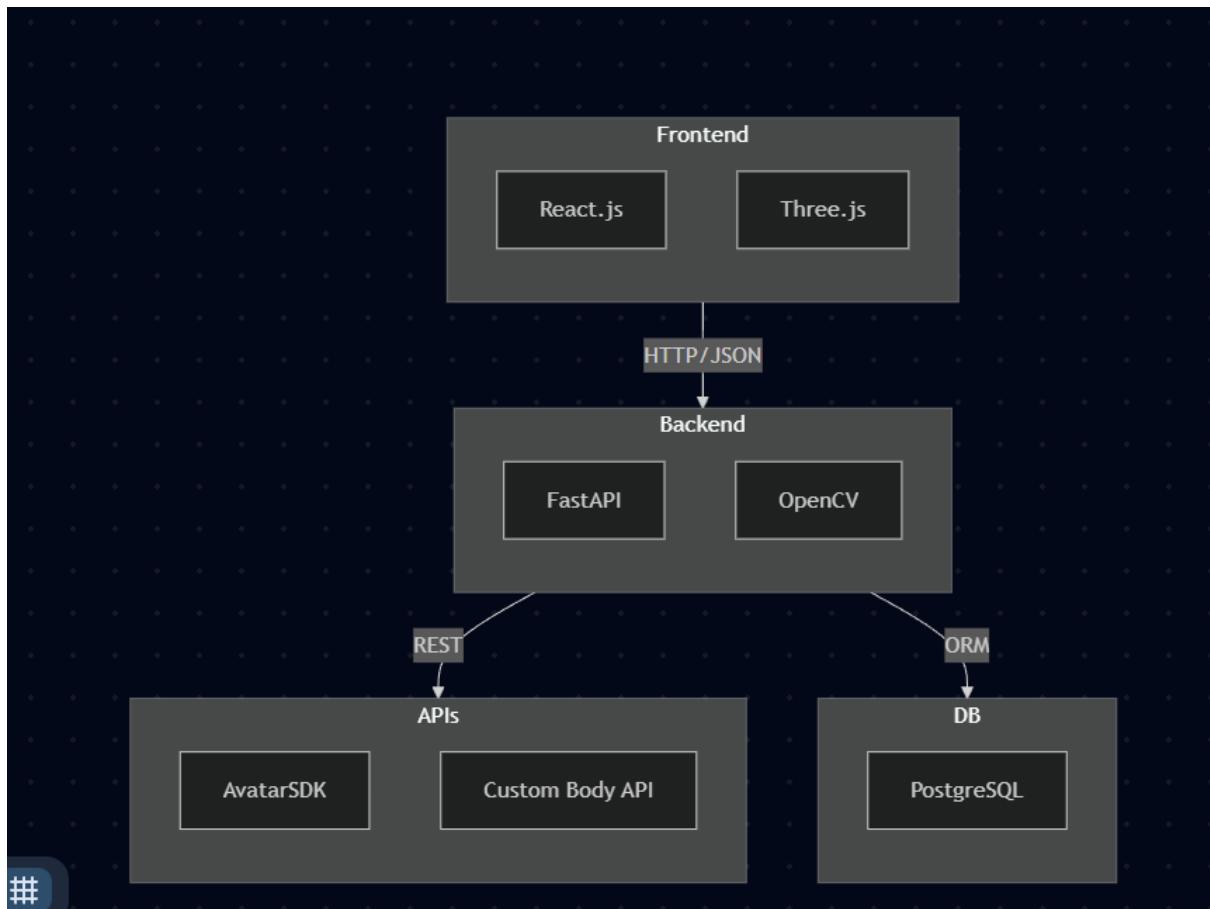
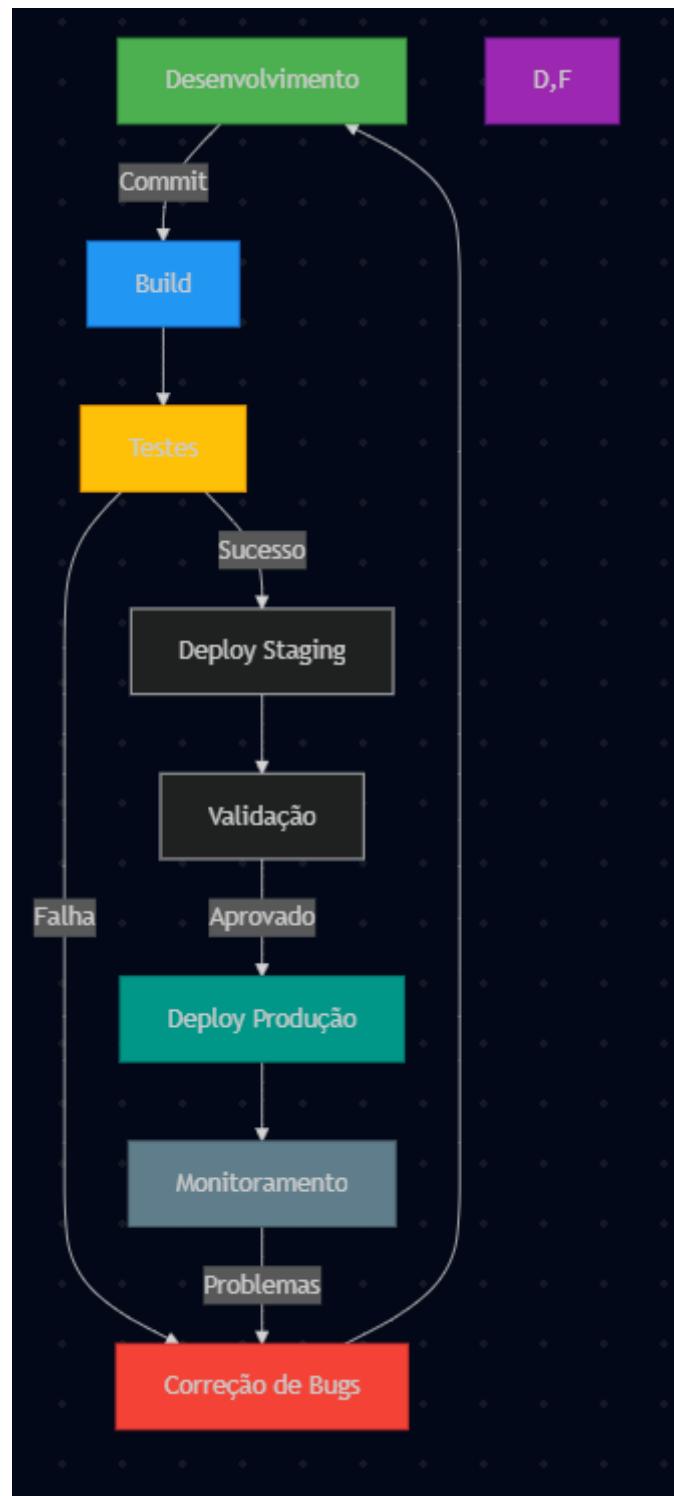
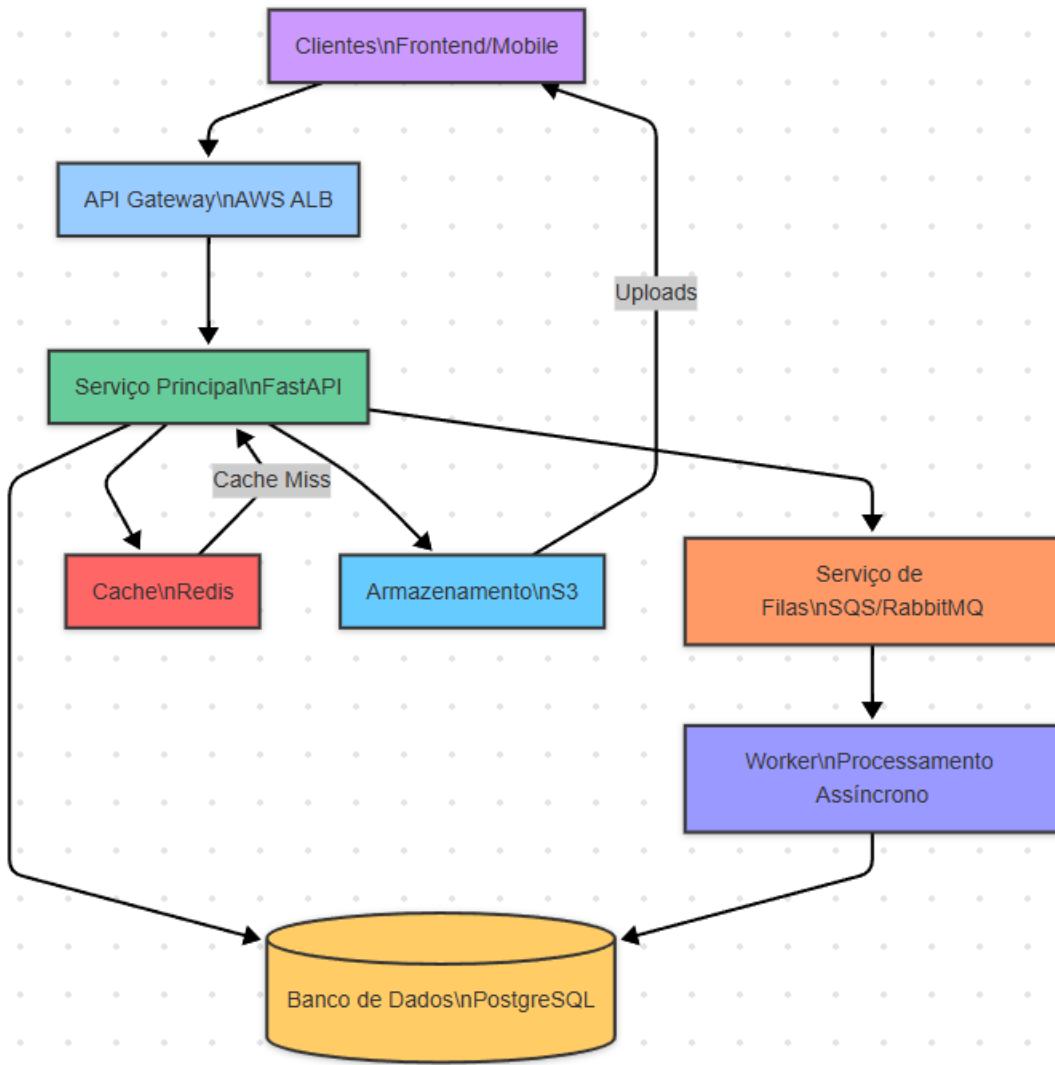


Diagrama de Estados (Avatar)



11. implementação do projeto





Explicação por etapas (para leigos):

1. Tirar a Foto

- Você tira uma selfie normal (como faria para redes sociais)

2. Cozinha Digital

- Nossa "chef digital" (a IA) analisa sua foto
- Como um escultor, ele esculpe seu rosto em 3D

3. Vestir o Avatar

- O sistema adiciona um corpo proporcional
- Você escolhe roupas e cortes de cabelo, como em jogos

4. Resultado Final

- Recebe um boneco digital que:
- Pode ser usado em vídeos
- Funciona em celulares e computadores
- Pode virar figurinha ou personagem de jogo

12. Conclusão

O AvatarSDK fornece uma base sólida para a criação de avatares 3D realistas e personalizáveis. A integração com ferramentas 3D e IA permitirá um sistema interativo e adaptável à experiência do utilizador.