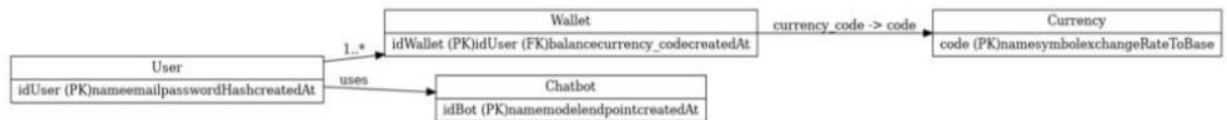


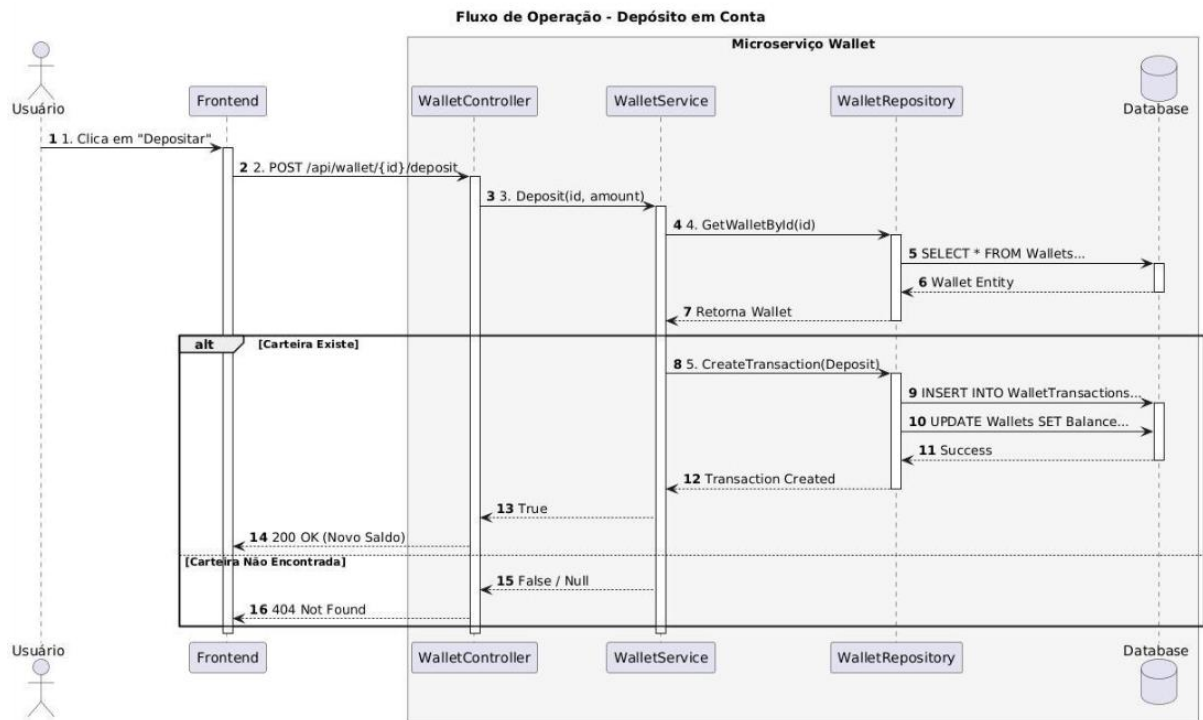
## Diagrama Entidade-Relacionamento (DER)

Este modelo lógico de dados apresenta as principais entidades do sistema e seus relacionamentos. O **User** (usuário) possui um relacionamento de "um para muitos" com a **Wallet** (carteira), permitindo múltiplas carteiras de diferentes moedas. A entidade **Currency** (moeda) define os tipos de ativos suportados, enquanto o **Chatbot** armazena o histórico de interações para auditoria e treino do modelo de IA.



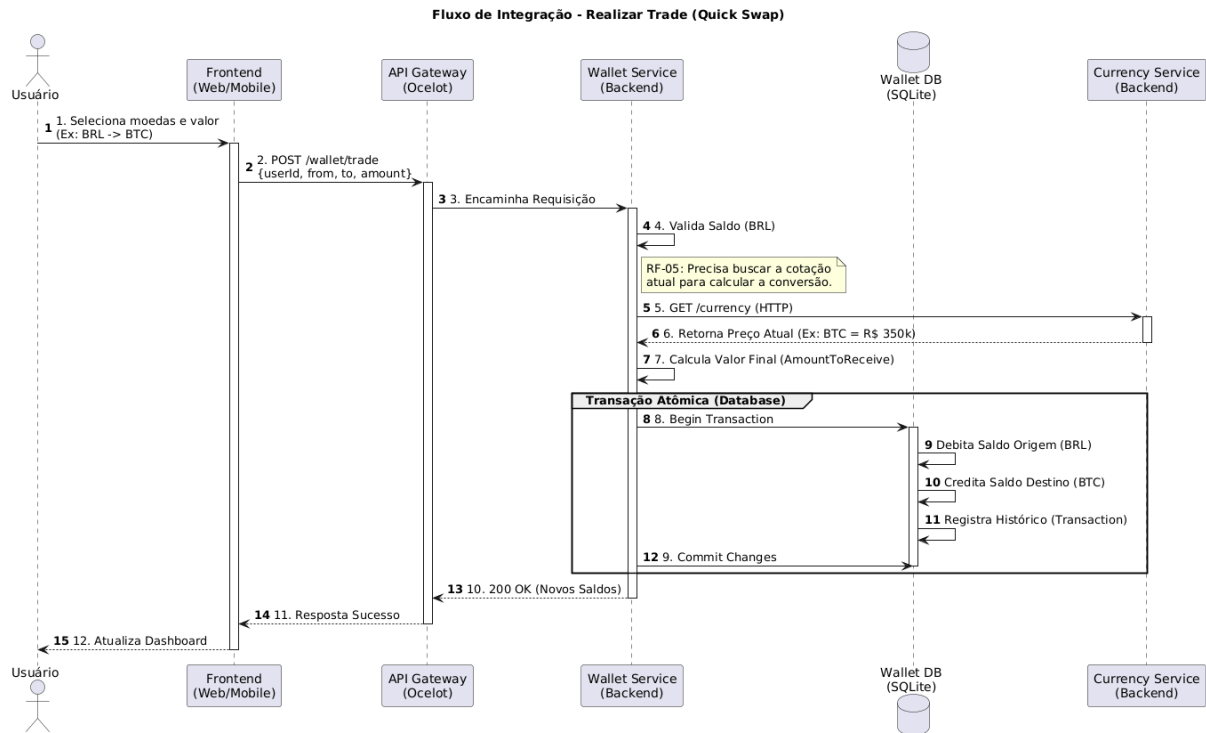
## Diagrama de Sequência - Fluxo de Depósito (Visão Interna)

Detalhamento da interação entre as camadas do microserviço de Wallet (Controller, Service e Repository) durante uma operação de depósito. O fluxo demonstra a aplicação dos princípios de orientação a objetos e camadas de responsabilidade única, desde a recepção da requisição HTTP até à persistência segura da transação e atualização do saldo no banco de dados.



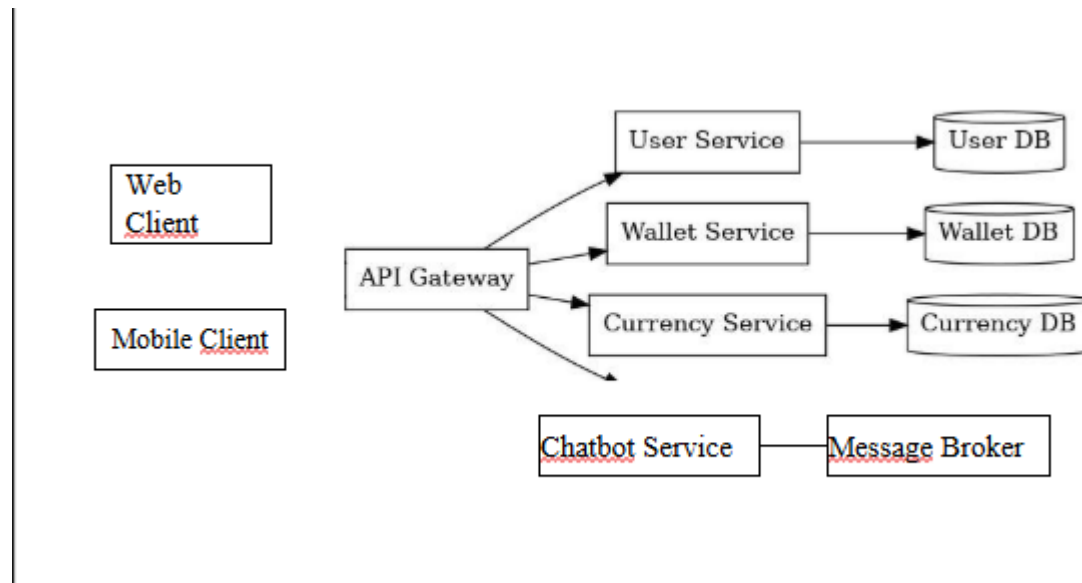
## Diagrama de Sequência - Fluxo de Integração de Trade (Quick Swap)

Ilustração do fluxo completo de uma operação de troca de moedas (Trade) num ambiente distribuído. O diagrama evidencia a orquestração realizada pelo **API Gateway**, a comunicação síncrona entre os microserviços de **Wallet** e **Currency** para obtenção da cotação em tempo real, e a execução de uma transação atômica no banco de dados para garantir a consistência financeira (débito e crédito simultâneos).



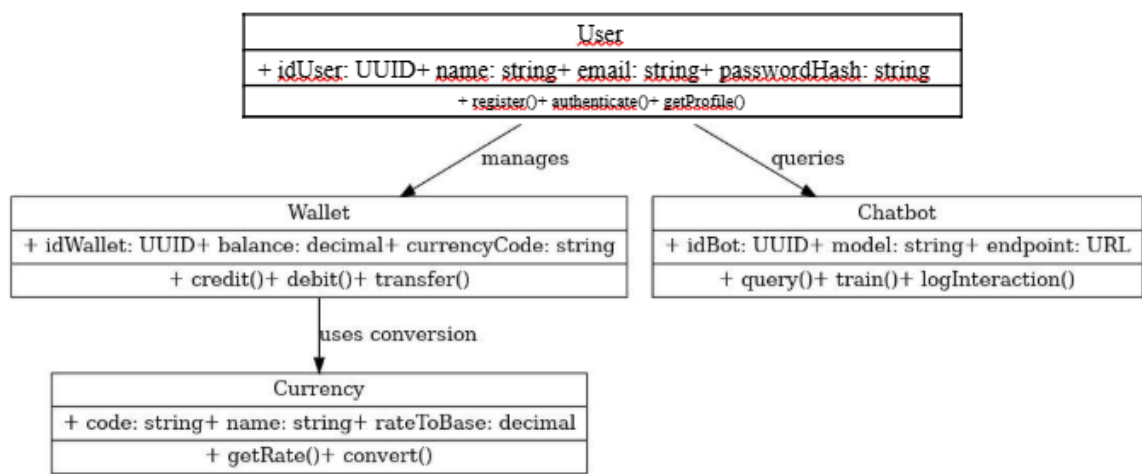
## Diagrama de Componentes (Arquitetura de Microserviços)

Visão geral da arquitetura do sistema baseada em microserviços. O **API Gateway** (Ocelot) atua como ponto único de entrada, encaminhando as requisições dos clientes (Web e Mobile) para os serviços de backend apropriados (**User**, **Wallet**, **Currency**, **Chatbot**). Cada serviço possui o seu próprio banco de dados isolado, garantindo o desacoplamento e a escalabilidade independente.



### Diagrama de Classes

Representação da estrutura orientada a objetos dos principais domínios do sistema. As classes encapsulam os dados (atributos como balance, rateToBase) e comportamentos (métodos como credit(), convert()) essenciais para a regra de negócio. O diagrama destaca as associações de gestão entre Usuário e Carteira, e de dependência para conversão entre Carteira e Moeda.



## Diagrama de Implantação (Deployment)

Visão da infraestrutura física e lógica do sistema. Demonstra como os componentes de software são implantados em contêineres/serviços distintos. O **Load Balancer/Gateway** gerencia o tráfego de entrada, distribuindo-o para os serviços de autenticação e negócios, que por sua vez comunicam-se com os seus respectivos servidores de banco de dados e o Message Broker para operações assíncronas do Chatbot.

