

| | | | |
|--------------------|------------------------------------|----------------------|------------|
| Escola: | Escola Politécnica | Campus: | Curitiba |
| Disciplina: | Design de Software | Ano/Semestre: | 2025/2s |
| Atividade: | 1ª Avaliação Somativa | Data: | 12/09/2025 |
| Professora: | Andreza Vieira | Turma: | |
| Estudantes: | Guilherme Carvalho Nicolas Lobo | | |

1. ORIENTAÇÕES GERAIS

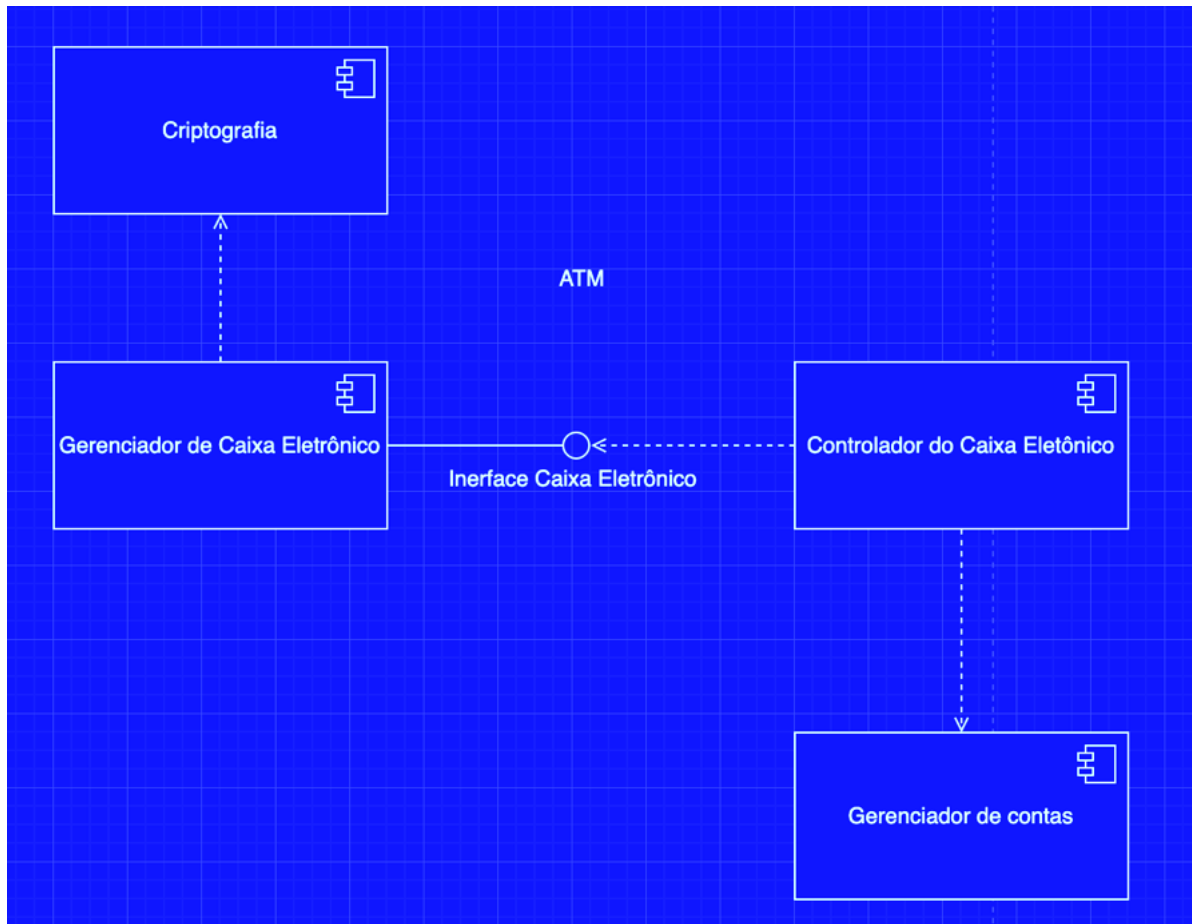
- Esta atividade refere-se à 1ª Avaliação Somativa, prevista no Plano de Ensino da disciplina de Design de Software.
- A avaliação é em dupla e terá a duração total de 1:30h.
- Não esqueça de preencher os nomes no cabeçalho da prova.
- Use a ferramenta de sua preferência.
- Deve ser enviado este mesmo arquivo word com as respostas.

PROVA

1) Você foi encarregado de modelar a estrutura de componentes de um novo software para um caixa eletrônico (ATM). As especificações são as seguintes:

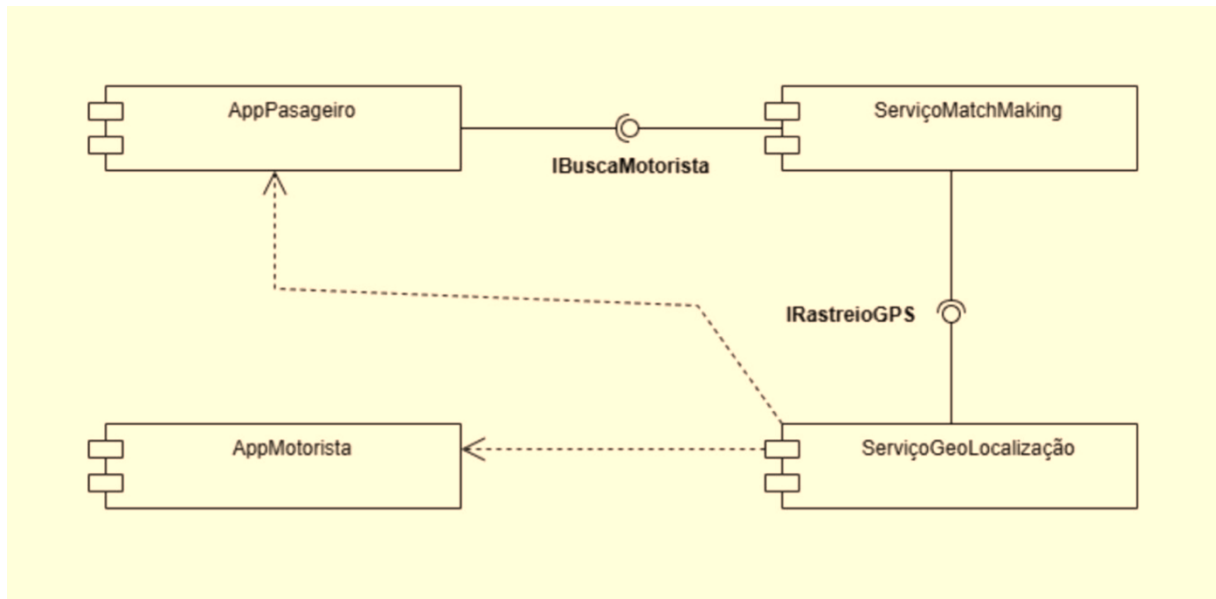
- O sistema possui um componente executável principal, o **Gerenciador de Caixa Eletrônico**.
- Este gerenciador implementa uma interface chamada **Interface Caixa Eletrônico**.
- Um outro componente, o **Controlador do Caixa Eletrônico**, utiliza os serviços da **Interface Caixa Eletrônico** para operar o hardware.
- O **Gerenciador de Caixa Eletrônico** depende de um componente de **Criptografia** para a segurança das transações.
- O **Controlador do Caixa Eletrônico** depende de um **Gerenciador de Contas** para processar as transações bancárias.

Desenhe o Diagrama de Componentes que representa a arquitetura descrita, utilizando corretamente as notações de componente, interface fornecida/requerida e dependência.



2) Veja a seguir as informações sobre a arquitetura de um aplicativo de transporte como o Uber e faça o diagrama de componentes com a ligação das interfaces requeridas e fornecidas. (3 pontos)

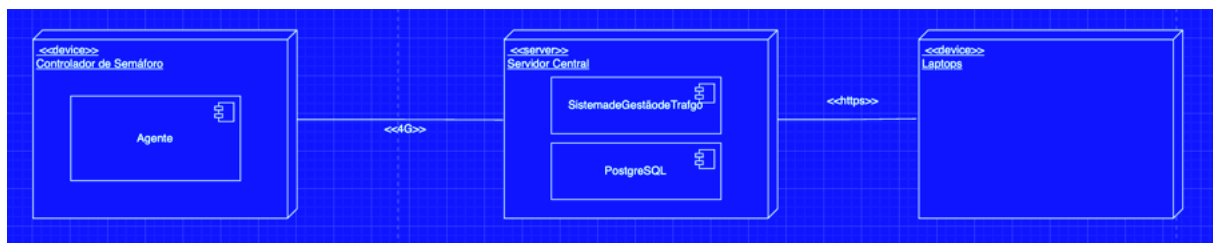
- **Início:** O usuário, através do **AppPassageiro**, inicia uma solicitação de corrida. Para encontrar motoristas próximos, o **AppPassageiro** utiliza a interface **IBuscaMotorista**.
- **Matchmaking:** O **ServicoMatchmaking** é o componente que implementa a interface **IBuscaMotorista**. Ele recebe a solicitação e é sua responsabilidade encontrar o motorista ideal.
- **Localização:** Para encontrar motoristas próximos, o **ServicoMatchmaking** utiliza a interface **IRastreoGPS**. Ele a utiliza para obter a localização em tempo real dos motoristas.
- **Fornecimento de Localização:** O **ServicoGeolocalizacao** é o responsável por rastrear os motoristas ativos. Ele implementa a interface **IRastreoGPS**, fornecendo os dados de localização para o **ServicoMatchmaking**.
- **Confirmação:** Uma vez que o **ServicoMatchmaking** encontra o motorista ideal, ele notifica tanto o **AppMotorista** quanto o **AppPassageiro**.



3) Uma cidade está implementando um sistema de trânsito inteligente. A infraestrutura física é composta por:

- Vários **Controladores de Semáforo** (dispositivos <<device>>) espalhados pela cidade, cada um rodando um software chamado **agente**
- Esses controladores se comunicam via rede **4G** com um **Servidor Central** (<<server>>).
- O **Servidor Central** hospeda o **SistemaDeGestaoDeTrafego** (componente) e um banco de dados **PostgreSQL**.
- Engenheiros de tráfego monitoram e controlam o sistema a partir de seus **Laptops** (<<device>>), que se conectam ao **Servidor Central** via **HTTPS**.

Modele o Diagrama de Implantação para este cenário, representando todos os nós e os protocolos de comunicação entre eles.



4) Um componente “AnalizadorDeSentimento” cresceu tanto que agora inclui lógica para processamento de texto, conexão com APIs de tradução e algoritmos de machine learning. De acordo

com o Princípio da Responsabilidade Única (SRP) aplicado a componentes, qual seria a melhor abordagem?

- a) Reescrever todo o componente usando uma linguagem de programação mais eficiente.
- b) Manter o componente como está para facilitar a implantação, já que tudo está em um só lugar.

[c)] Dividir o componente em componentes menores e mais coesos, como 'ProcessadorDeTexto', 'ClienteDeTraducao' e 'MotorDeML', cada um com uma única responsabilidade.

- d) Adicionar mais interfaces ao componente para cada uma das suas funcionalidades.