Neste documento, descrevo os conceitos de programação, bem como suas vantagens em relação a outras abordagens de software, utilizados nos projetos SIAD-CR e Facilite Já, suas funcionalidades e o seu objetivo no negócio.

**1. Conceitos de Programação**

**Interfaces claras:**

Definir interfaces entre os módulos para reduzir dependências.

**Injeção de dependência:**

Utilizamos padrões como injeção de dependência para fornecer componentes externos, reduzindo o acoplamento.

**Mensagens Assíncronas:**

Utilizar camadas de abstração para isolar componentes e reduzir a necessidade de conhecimento interno.

Por se tratar de uma aplicação na qual seus objetivos são, de primeira instancia, armazenar ações realizadas pelo administrador, a fim de garantir o monitoramento, diagnóstico de problemas, segurança (Para o registro e análise de ações não suspeitas, como por exemplo a ação em horários não comercial), auditoria e Histórico de ações, descrevo abaixo as principais razões para a escolha da tecnologia.

Arquitetura Limpa:

Utilizo um meu projeto de POC, um design de software propostos por Robert C. Martin, também conhecido como "Uncle Bob", denominada de clear arquitecture. Essa arquitetura busca criar sistemas flexíveis, escaláveis e fáceis de manter, mantendo uma separação clara de responsabilidades.

Os principais componentes da Arquitetura Limpa, na POC, incluem:

Entidades:

Representam os objetos de negócios do domínio. Elas encapsulam o estado e as regras de negócios.

Casos de Uso (Use Cases):

Contêm as regras de negócios específicas da aplicação. Eles orquestram a interação entre as entidades e os demais componentes.

**Interfaces de Entidades (Gateways):**

São interfaces que permitem a comunicação com elementos externos, como bancos de dados, serviços web, etc. Essas interfaces são implementadas na camada externa.

**Controladores:**

Recebem as solicitações do exterior e orquestram a execução dos casos de uso.

**Presenters/DTOs:**

Responsáveis pela apresentação dos dados para a interface do usuário. Eles traduzem os dados provenientes dos casos de uso para um formato compreensível para a camada de interface do usuário.

**Frameworks e Drivers:**

Contêm os detalhes técnicos, como frameworks específicos, bibliotecas e componentes de infraestrutura. Essa é a camada mais externa e a única que depende de frameworks específicos.

A arquitetura é organizada em camadas concêntricas, com a Regra da Dependência (Dependency Rule) indicando que a direção da dependência deve apontar sempre para o centro da arquitetura, mantendo as dependências internas às camadas mais internas e as camadas externas dependentes apenas de abstrações.

Os benefícios da Arquitetura Limpa incluem a manutenção da flexibilidade do sistema, facilitação da testabilidade, redução da dependência de frameworks externos e melhor compreensão do código.

**Vantagens de armazenar logs no MongoDB:**

1. Estrutura de Dados Mais Complexa:

Se seus logs têm uma estrutura de dados complexa e exigem consultas mais avançadas, um banco de dados como o MongoDB pode oferecer mais flexibilidade.

2. Recursos de Consulta e Indexação:

O MongoDB oferece recursos avançados de consulta e indexação, o que pode ser útil para análise de logs em profundidade.

3. Integração com Outros Dados:

Facilidade de integração e armazenamento com a comunicação de serviços externos.

4. Persistência de Longo Prazo.

Por se tratar de uma aplicação na qual seus objetivos são, de primeira instancia, de uso exclusivo para clientes externos da companhia, no qual sua principal função é o pagamento de boletos, segue abaixo as principais razões para a escolha do cloud watch para armazenamento de logs: 1.

1 Monitoramento em Tempo Real:

O CloudWatch permite monitorar logs em tempo real, facilitando a detecção e resposta rápida a eventos importantes ao realizar um pagamento.

2. Escala Automática:

O CloudWatch é dimensionado automaticamente para lidar com grandes volumes de logs, oferecendo escalabilidade eficiente.

3. Ferramentas de Análise:

O CloudWatch fornece ferramentas de análise e consulta de logs, permitindo a extração de insights úteis, a fim de conhecermos mais o cliente final.

4. Alertas e Ações Automáticas:

Pode-se configurar alertas no CloudWatch para acionar ações automáticas em resposta a eventos específicos, como erros ao realizar um pagamento.

**Visando segurança, todos os dados sensíveis que a aplicação deverá** compartilhar**, bem como senhas e tokens, deverão ser armazenados no serviço da aws secrets manager.**

**2. Objetivo no Negócio**

O SIAD-CR é de uso interno da companhia, voltado exclusivamente para a área de cobrança e renegociação de crédito.

Suas principais funcionalidades são:

1. Ações de Cobrança ao Cliente via E-mail.
2. Negativação junto aos birôs de crédito.
3. Emissão de Boletos.

O Facilite já é de uso externo, ou seja, para o usuário final poder realizar os principais objetivos:

1. Funcionalidades de Renegociação (Agrupamento da Dívida e Simulação de Pagamento);
2. Funcionalidades de Pagamento (Pagamento de Boletos).