
Curso Ebac - Ti do Zero

Cartório de Registro
Documento de Arquitetura de Software

Versão <1.0>

| | |
|--------------------------------------|--------------------|
| Cartório de Registro | Version: <1.0> |
| Documento de Arquitetura de Software | Date: <25/02/2024> |
| | |

Índice Analítico

| | | |
|-----|---|---|
| 1. | Introdução | 3 |
| 1.1 | Finalidade | 3 |
| 1.2 | Escopo | 3 |
| 1.3 | Definições, Acrônimos e Abreviações | 3 |
| 1.4 | Referências | 3 |
| 1.5 | Visão Geral | 3 |
| 2. | Representação Arquitetural | 3 |
| 3. | Metas e Restrições da Arquitetura | 3 |
| 4. | Visão de Casos de Uso | 4 |
| 4.1 | Realizações de Casos de Uso | 4 |
| 5. | Visão Lógica | 4 |
| 5.1 | Visão Geral | 4 |
| 5.2 | Pacotes de Design Significativos do Ponto de Vista da Arquitetura | 4 |
| 6. | Visão de Processos | 4 |
| 7. | Visão de Implantação | 4 |
| 8. | Visão da Implementação | 4 |
| 8.1 | Visão Geral | 5 |
| 8.2 | Camadas | 5 |
| 9. | Visão de Dados (opcional) | 5 |
| 10. | Tamanho e Desempenho | 5 |
| 11. | Qualidade | 5 |

| | |
|--------------------------------------|--------------------|
| Cartório de Registro | Version: <1.0> |
| Documento de Arquitetura de Software | Date: <25/02/2024> |
| | |

Documento de Arquitetura de Software

1. Introdução

Esse documento tem como finalidade apresentar os requisitos de desenvolvimento para o Projeto de Registro de Cartório da EBAC, o cliente nos apresentou a necessidade de registrar os colaboradores, alunos e demais cargos que fazem parte da empresa. O Escopo foi definido para a criação de um menu principal com quatro opções de escolha, sendo elas Registrar, Consultar, Deletar e Sair da Aplicação. A função Registrar recebe os dados do usuário e os armazena em um banco de dados local não relacional. A função Consultar recebe um valor de referência (definido como chave primária), definido como CPF e busca no banco de dados as informações solicitadas. A função Deletar exclui os dados cadastrados a partir da chave primária apresentada pelo usuário, o CPF. Ao final dessas três funções foi criado uma Decisão a ser tomada, o usuário irá escolher se deseja sair e finalizar o programa ou se deseja retornar ao menu e continuar na aplicação. Por fim, a função Sair da Aplicação, última opção no menu que possibilita a saída e o encerramento do programa. Foi escolhida para o desenvolvimento deste software a linguagem C.

1.1 Finalidade

Este documento oferece uma visão geral arquitetural abrangente do sistema, usando diversas visões de arquitetura para representar diferentes aspectos do sistema. O objetivo deste documento é capturar e comunicar as decisões de arquitetura significativas que foram tomadas em relação ao sistema.

1.2 Escopo

Este documento auxilia os envolvidos no projeto a captar aspectos de arquitetura do sistema que são necessários para o desenvolvimento de uma solução que atenda às necessidades dos usuários finais. Além de auxiliar no entendimento do sistema por novos membros da equipe. O documento apresenta as principais funcionalidades da aplicação, instrução e explicação de uso.

1.3 Definições, Acrônimos e Abreviações

[Esta subseção contém as definições de todos os termos, acrônimos e abreviações necessários para interpretar corretamente o Documento de Arquitetura de Software. Essas informações podem ser fornecidas fazendo referências ao Glossário do projeto.]

1.4 Visão Geral

[Esta subseção descreve o que o restante do Documento de Arquitetura de Software contém e explica como o documento está organizado.]

2. Representação Arquitetural

[Esta seção descreve qual é a arquitetura de software do sistema atual e como ela é representada. Da Visão de Casos de Uso, Visão Lógica, Visão de Processos, Visão de Implantação e Visão de Implementação, enumera as visões necessárias e, para cada visão, explica quais tipos de elementos de modelo ela contém.]

3. Metas e Restrições da Arquitetura

[Esta seção descreve os requisitos e objetivos do software que têm algum impacto sobre a arquitetura; por exemplo, segurança, garantia, privacidade, uso de um produto desenvolvido internamente e pronto para ser usado, portabilidade, distribuição e reutilização. Ela também captura as restrições especiais que podem ser aplicáveis: estratégia de design e implementação, ferramentas de desenvolvimento, estrutura das equipes, cronograma, código-fonte legado e assim por diante.]

| | |
|--------------------------------------|--------------------|
| Cartório de Registro | Version: <1.0> |
| Documento de Arquitetura de Software | Date: <25/02/2024> |
| | |

4. Visão de Casos de Uso

[Esta seção lista casos de uso ou cenários do modelo de casos de uso quando eles representam funcionalidade central e significativa do sistema final ou, quando têm uma grande cobertura arquitetural — eles experimentam muitos elementos arquiteturais ou quando enfatizam ou ilustram um ponto complexo e específico da arquitetura.]

4.1 Realizações de Casos de Uso

[Esta seção ilustra o funcionamento do software, apresentando algumas realizações (ou cenários) de casos de uso selecionadas e explica como os diversos elementos do modelo de design contribuem para a respectiva funcionalidade.]

5. Visão Lógica

[Esta seção descreve as partes significativas do ponto de vista da arquitetura do modelo de design, como sua divisão em subsistemas e pacotes. Além disso, para cada pacote significativo, ela mostra sua divisão em classes e utilitários de classe. Apresente as classes significativas do ponto de vista da arquitetura e descreva suas responsabilidades, bem como alguns relacionamentos, operações e atributos de grande importância.]

5.1 Visão Geral

[Esta subseção descreve toda a decomposição do modelo de design em termos de camadas e de hierarquia de pacotes.]

5.2 Pacotes de Design Significativos do Ponto de Vista da Arquitetura

[Para cada pacote significativo, inclua uma subseção com o respectivo nome, uma breve descrição e um diagrama com todos os pacotes e classes significativos nele contidos.]

[Para cada classe significativa no pacote, inclua o respectivo nome, uma breve descrição e, opcionalmente, uma descrição de algumas das suas principais responsabilidades, operações e atributos.]

6. Visão de Processos

[Esta seção descreve a decomposição do sistema em processos leves (threads simples de controle) e processos pesados (agrupamentos de processos leves). Organize a seção em grupos de processos que se comunicam ou interagem. Descreva os modos principais de comunicação entre processos, como transmissão de mensagens e interrupções.]

7. Visão de Implantação

[Esta seção descreve uma ou mais configurações da rede física (hardware) na qual o software é implantado e executado. Ela é uma visão do Modelo de Implantação. No mínimo, para cada configuração, ela deve indicar os nós físicos (computadores, CPUs) que executam o software e suas interconexões (barramento, LAN, ponto a ponto, etc.) É incluído também um mapeamento dos processos da Visão de Processos nos nós físicos.]

8. Visão da Implementação

[Esta seção descreve a estrutura geral do modelo de implementação, a divisão do software em camadas e os subsistemas no modelo de implementação e todos os componentes significativos do ponto de vista da arquitetura.]

| | |
|--------------------------------------|--------------------|
| Cartório de Registro | Version: <1.0> |
| Documento de Arquitetura de Software | Date: <25/02/2024> |
| | |

8.1 Visão Geral

[Esta subseção nomeia e define as diversas camadas e o seu conteúdo, as regras que determinam a inclusão em uma camada específica e as fronteiras entre as camadas. Inclua um diagrama de componentes que mostre os relacionamentos entre as camadas.]

8.2 Camadas

[Para cada camada, inclua uma subseção com o respectivo nome, uma lista dos subsistemas localizados na camada e um diagrama de componentes.]

9. Visão de Dados (opcional)

[Uma descrição da perspectiva de armazenamento de dados persistentes do sistema. Esta seção será opcional se os dados persistentes forem poucos ou inexistentes ou se a conversão entre o Modelo de Design e o Modelo de Dados for trivial.]

10. Tamanho e Desempenho

[Uma descrição das principais características de dimensionamento do software que têm um impacto na arquitetura, bem como as restrições do desempenho desejado.]

11. Qualidade

[Uma descrição de como a arquitetura do software contribui para todos os recursos (exceto a funcionalidade) do sistema: extensibilidade, confiabilidade, portabilidade e assim por diante. Se essas características possuírem significado especial, como implicações de segurança, garantia ou privacidade, elas deverão ser delineadas claramente.]