|  |
| --- |
| **PLANO DE AULA** |
| **TEMA:**  Visão global do modelo arquitetural AReS 2 do Bacen. |

|  |
| --- |
| **OBJETIVOS** |
| **GERAL:**  Capacitar os profissionais em conceitos técnicos do modelo arquitetural AReS 2. |
| **ESPECÍFICO:**  1. Apresentar aos profissionais o modelo arquitetural AReS 2.  2. Apresentar conceitos sobre o protocolo HTTP e como o REST funciona.  3. Apresentar como construir um CRUD utilizando Spring Rest.  4. Apresentar como construir um CRUD utilizando Angular.  5. Apresentar como construir um batch.  6. Apresentar principais comandos Docker.  7. Apresentar a ferramenta de ferramenta de testes de aceitação. |

|  |
| --- |
| **CONTEÚDO** |
| * O que é modelo arquitetural AReS 2? * Organização de conteúdo dentro da arquitetura. * Fundamentos protocolo HTTP. * Fundamentos REST. * Fundamentos da Injeção de dependência. * Fundamentos dos Testes de aceitação. * Back-end:   + Spring   + Spring boot   + Lombok   + Hibernate   + QueryDSL   + Validation builder   + Permissão   + Apache POI   + Orika   + Adapter (Informação externa)   + Organização de conteúdo   + Cucumber   + Junit * Front-end:   + JavaScript   + TypeScript   + Reactive X   + Angular cli   + Angular   + Organização de conteúdo. * Batch:   + Spring Batch   + Shell script (run.sh)   + Arquivo configuração por ambiente (dsen, homo, prod)   + Organização de conteúdo.   + Organização do pacote gerado. * Exercício CRUD e BATCH * Remoção de dúvidas. |

|  |
| --- |
| **METODOLOGIA** |
| |  |  |  | | --- | --- | --- | |  | **Tempo** | **Descrição da etapa** | | 1 | 00h 15m | Acolhimento e abertura | | 2 | 00h 15m | Apresentação do modelo arquitetural | | 3 | 00h 30m | Apresentação do protocolo HTTP e REST | | 4 | 03h 00m | Apresentação de um CRUD no back-end | | 5 | 02h 00m | Apresentação de um CRUD no front-end | | 6 | 01h 00m | Apresentação de um CRUD no batch | | 7 | 00h 45m | Remoção de dúvidas | | 8 | 00h 15m | Exercício proposto para estudo. | |

|  |
| --- |
| **AVALIAÇÃO** |
| * Perguntas durante o treinamento * Exercícios * Prova |

|  |
| --- |
| **REFERÊNCIAS** |
| * **Angular**   Internet: https://angular.io/  Internet: https://angular-templates.io/tutorials/about/learn-angular-from-scratch-step-by-step  Internet: https://loiane.training/curso/angular  Internet: https://cli.angular.io/   * **Apache POI**   Internet: https://poi.apache.org/   * **Cucumber**   Internet: https://cucumber.io/   * **Docker**   Internet: https://www.docker.com/   * **Hibernate**   Internet: https://hibernate.org/     * **Injeção de dependência**   Internet: https://en.wikipedia.org/wiki/Dependency\_injection   * **Junit**   Internet: https://junit.org/junit4/   * **Lombok**   Internet: <https://projectlombok.org/>   * **QueryDSL**   Internet: http://www.querydsl.com/   * **Reactive X**   Internet: http://reactivex.io/  Internet: https://www.learnrxjs.io/   * **Shell script**   Internet: https://www.shellscript.sh/   * **Spring**   Internet: https://spring.io/projects/spring-framework  Internet: https://spring.io/projects/spring-boot   * **TypeScript**   Internet: <https://www.typescriptlang.org/> |

|  |
| --- |
| **PROJEÇÃO DE HORAS** |
| * **Construção de Conteúdo Responsável: Thiago Garbazza**   **30 horas.**   * **Aplicação Treinamento Responsável: Thiago Garbazza**   **10 horas.**   * **Total de horas de construção e aplicação: 40 horas.** |
|  |
| **RECURSOS NECESSÁRIOS** |
| * Computador com ferramentas de desenvolvimento   + Maven   + JDK 8   + Node > 10   + GIT   + Power point (PPT) |

|  |
| --- |
| **IMPORTANTE!** |
| **É essencial que o plano de aula, assim como todo o material do treinamento seja encaminhado para a Gerência de Desenvolvimento de Pessoas.** |