Crédito Pessoal

Sprints do Projeto

Versão 1.0

Histórico da Revisão

|  |  |  |  |
| --- | --- | --- | --- |
| **Data** | **Versão** | **Descrição** | **Autor** |
| 25/05/2023 | 1.0 | Especificação Inicial | Thiago Z. Araujo |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Sprints

**Sprint 1.**

* 1. – Criação do recurso no serviço de **API Gateway** para recebimento de propostas via **Clientes APP’s**.
  2. – Criação da rota de consumo (**00**), no **API Gateway**, para o componente de serviço de **Proposta de Crédito** (**sbootarchhex-orch-crdt-proposta-credito**).
  3. – Criação dos 4 componentes do fluxo principal e disponibilização para futuro desenvolvimento (**sboot-archhex-orch-crdt-proposta-credito, sboot-archhex-orch-crdt-contratacao-credito, sboot-archhex-orch-crdt-analise-credito, sboot-archhex-orch-crdt-liberacao-credito**).
  4. – Criação dos 3 componentes de persistência de dados e disponibilização para futuro desenvolvimento (**sboot-archex-atom-crdt-cadastro, sboot-archex-atom-crdt-financeiro, sboot-archex-atom-crdt-proposta**).
  5. – Criação das 5 filas para produção e consumo de mensagens **(Fila Proposta (Kafka), Fila Notificacao Email (Kafka), Fila Finalizacao (Kafka), Fila Contratacao (Kafka), Fila Liberacao Credito (Kafka)**).
  6. – Criação dos 3 bancos de dados (**Cadastro (MongoDB), Financeiro (MongoDB), Proposta (Oracle)**).

**Sprint 2.**

* 1. – Definir o contrato, utilizando **OpenApi** versão ***3.x***, dos endpoints de “CRUD – CREATE, RED, UPDATE e DELETE” para o componente de serviço de banco de dados (**sboot-archex-atom-crdt-cadastro**) do banco de dados schema **Cadastro (MongoDB)**.
  2. – Criar os recursos de “crud – create, red, update e delete” no componente **sboot-archex-atom-crdt-cadastro**.
  3. – Definir o contrato, utilizando **OpenApi** versão ***3.x***, dos endpoints de “CRUD – CREATE, RED, UPDATE e DELETE” para o componente de serviço de banco de dados (**sboot-archex-atom-crdt-financeiro**) do schema **Financeiro (MongoDB)**.
  4. – Criar os recursos de “crud – create, red, update e delete” no componente **sboot-archex-atom-crdt-financeiro**.
  5. – Definir o contrato, utilizando **OpenApi** versão ***3.x***, dos endpoints de “CRUD – CREATE, RED, UPDATE e DELETE” para o componente de serviço de banco de dados (**sboot-archex-atom-crdt-proposta**) do schema **Proposta (Oracle)**.
  6. – Criar os recursos de “crud – create, red, update e delete” no componente **sboot-archex-atom-crdt-proposta**.

**Sprint 3.**

* 1. – Criar o recurso no componente de Proposta de Crédito (**sboot-archhex-orch-crdt-proposta-credito**) que recebe dados de uma proposta (00), consome os serviços de cadastro (**01**, **02**, **03**) (**sboot-archex-atom-crdt-cadastro**, **sboot-archex-atom-crdt-financeiro**, **sboot-archex-atom-crdt-proposta**) e produz uma mensagem para a fila de Propostas (**04**) (**Fila Proposta (Kafka)**).
  2. – Criar um método no componente de Contratação de Crédito (**sboot-archhex-orch-crdt-contratacao-credito**) que consome informações do serviço Rest (**06.1**) de dados do SERASA.
  3. – Criar um método no componente de Contratação de Crédito (**sboot-archhex-orch-crdt-contratacao-credito**) que consome informações do serviço Rest (**06.2**) de dados do SPC.
  4. – Criar um método no componente de Contratação de Crédito (**sboot-archhex-orch-crdt-contratacao-credito**) que consome informações do serviço Rest (**06.3**) de dados do Risco de Crédito.
  5. – Criar um método no componente de Contratação de Crédito (**sboot-archhex-orch-crdt-contratacao-credito**) que consome informações do serviço Rest (**06.4**) de dados do Score.

**Sprint 4.**

* 1. – Definir o contrato, utilizando **OpenApi** versão ***3.x***, do endpoint de atualização de proposta (consumo dos serviços Rest de SERASA, SPC, Risco de Crédito e Score) para o componente de serviço de banco de dados (**sboot-archex-atom-crdt-proposta**) do schema **Proposta (Oracle)**.
  2. – cccccc
  3. – Definir o contrato, utilizando **OpenApi** versão ***3.x***, do endpoint de atualização de proposta (situação da proposta para aprovada ou reprovada) para o componente de serviço de banco de dados (**sboot-archex-atom-crdt-proposta**) do schema **Proposta (Oracle)**.
  4. – Criar um método no componente de Contratação de Crédito (**sboot-archhex-orch-crdt-contratacao-credito**) que consome o serviço de atualização de proposta no componente de banco de dados (**Sprint 4 – 4.3**) (**sboot-archex-atom-crdt-proposta**).
  5. – Definir o contrato, utilizando **OpenApi** versão ***3.x***, do endpoint de atualização de proposta (liberação do crédito) para o componente de serviço de banco de dados (**sboot-archex-atom-crdt-proposta**) do schema **Proposta (Oracle)**.
  6. – Criar um método no componente de Contratação de Crédito (**sboot-archhex-orch-crdt-contratacao-credito**) que consome o serviço de atualização de proposta no componente de banco de dados (**Sprint 4 – 4.5**) (**sboot-archex-atom-crdt-proposta**).

**Sprint 5.**

* 1. – Criar o recurso no componente de Contratação de Crédito (**sboot-archhex-orch-crdt-contratacao-credito**) que consome uma mensagem da fila de Propostas (**05**) (**Fila Proposta (Kafka)**), utiliza os métodos (**Sprint 3 – 3.2, 3.3, 3.4 e 3.5**) das consultas aos serviços de SERASA, SPC, Risco de Crédito e Score, consome o serviço de atualização de proposta no componente de banco de dados (**15**) (**sboot-archex-atom-crdt-proposta**) e produz uma mensagem para a fila de Contratação (**07**) (**Fila Contratacao (Kafka)**).
  2. – Criar o recurso no componente de Análise de Crédito (**sboot-archhex-orch-crdt-analise-credito**) que consome uma mensagem da fila de Contratação (**08**) (**Fila Contratacao (Kafka)**), avalia os dados da proposta, consome o serviço de atualização de proposta no componente de banco de dados (**16**) (**sboot-archex-atom-crdt-proposta**) e produza uma mensagem para a fila de Finalização (**09.1**) (**Fila Finalizacao (Kafka)**).
  3. – Criar para o recurso de análise de crédito (**Sprint 4 - 4.2**), fluxo para a liberação de crédito caso a situação da proposta seja aprovada (produzir mensagem para a fila de Liberação de Crédito (**09.2**) (**Fila Liberacao Credito (Kafka)**)).

**Sprint 6.**

* 1. – Criar o recurso no componente de Liberação de Crédito (**sboot-archhex-orch-crdt-liberacao-credito**) que consome uma mensagem da fila de Liberação de Crédito (**11**) (**Fila Liberacao Credito (Kafka)**), libera o crédito para o cliente e consome o serviço de atualização de proposta no componente de banco de dados (**17**) (**sboot-archex-atom-crdt-proposta**).
  2. – Criar o recurso no componente de Contratação de Crédito (**sboot-archhex-orch-crdt-contratacao-credito**) que consome uma mensagem da fila de Finalização (**10**) (**Fila Finalizacao (Kafka)**) e produz uma mensagem para a fila de Notificação de E-mail (**12**) (**Fila Notificacao Email (Kafka)**).
  3. – Criar o recurso no componente de Notificação de E-mail (**Sistema Notificacao Email**) que consome uma mensagem da fila de Notificação de E-mail (**13**) (**Fila Notificacao Email (Kafka)**) com os dados para a produção de e-mails de comunicação com o cliente (aceite ou reprovação de proposta, contrato, etc.).

**Sprint 7.**

* 1. – Criar camada de segurança **JWT** entre os componentes **API Gateway** e Proposta de Crédito (**sboot-archhex-orch-crdt-proposta-credito**).
  2. – Criar integração entre o componente de Proposta de Crédito (**sboot-archhex-orch-crdt-proposta-credito**) como Prometheus (**Promethrus (Serviço Global)**).
  3. – Criar integração entre o componente de Contratação de Crédito (**sboot-archhex-orch-crdt-contratacao-credito**) como Prometheus (**Promethrus (Serviço Global)**).
  4. – Criar integração entre o componente de Análise de Crédito (**sboot-archhex-orch-crdt-analise-credito**) como Prometheus (**Promethrus (Serviço Global)**).
  5. – Criar integração entre o componente de Liberação de Crédito (**sboot-archhex-orch-crdt-liberacao-credito**) como Prometheus (**Promethrus (Serviço Global)**).
  6. – Criar referência no Grafana Loki (**Grafana Loki (Serviço Global)**), Alertmanager (**Alert Manager (Serviço Global)**) e Slack do componente de Proposta de Crédito (**sboot-archhex-orch-crdt-proposta-credito**).
  7. – Criar referência no Grafana Loki (**Grafana Loki (Serviço Global)**), Alertmanager (**Alert Manager (Serviço Global)**) e Slack do componente de Contratação de Crédito (**sboot-archhex-orch-crdt-contratacao-credito**).
  8. – Criar referência no Grafana Loki (**Grafana Loki (Serviço Global)**), Alertmanager (**Alert Manager (Serviço Global)**) e Slack do componente de Avaliação de Crédito (**sboot-archhex-orch-crdt-analise-credito**).
  9. – Criar referência no Grafana Loki (**Grafana Loki (Serviço Global)**), Alertmanager (**Alert Manager (Serviço Global)**) e Slack do componente de Liberação de Crédito (**sboot-archhex-orch-crdt-liberacao-credito**).

**Sprint 8.**

*Reservada para ajustes, melhorias e outras atividades decorrentes do processo de homologação.*