

Funcionalidades obrigatórias.

1 - Cadastro de usuário(Cliente)

Criação do usuário.

Alteração do usuário.

Consulta dos dados do usuário.

2 - Relatórios dos últimos acessos do usuário.

Deve constar os seguintes dados no relatório em tela.

1. Código do usuário
2. Nome do usuário
3. Data de acesso do usuário
4. Hora de acesso do usuário.

3 - Relatório mensal faturado sobre o uso da plataforma.

Valor bruto do faturamento mensal.

Valor bruto do lucro mensal. (usa fórmula de preço custo x valor vendido.)

4 - Relatório anual faturado sobre o uso da plataforma.

Valor bruto do faturamento anual.

Valor bruto do lucro anual. (usa fórmula de preço custo x valor vendido.)

5 - FeedBack do serviço/produto de 0 a 5 estrelas e observação.(tela de cliente)

6 - Relatório de feedBack:

Deve apresentar o nome do usuário, telefone, observação e quantidade de estrelas dadas no feedBack.

7 - Cadastro do produto/serviço:

O usuário(fornecedor), ou seja, adquirente da plataforma, deverá fazer o cadastro dos seus produtos/serviços, através da tela de administração.

8 - Função de modo claro/escuro para todo sistema.

Instruções:

Tela administrativa:

Relatórios ficam nesta tela.

Cadastro de produto/serviço.

Relatório de feedBack.

Tela de cliente:

Visualização e utilização da plataforma

Registro de FeedBack no final da compra/ou no momento do logOut do usuário.

Importante: caso o usuário já tenha feito o feedBack no caso de sistemas que não são vendas diretas, ou seja, feito um pedido, deve-se perguntar se ele deseja dar o feedBack da plataforma somente se ele ainda não fez o mesmo.

Cadastro, alteração e consulta do cliente

Critérios do trabalho de levantamento de requisitos e gestão de atividades.

1.Modelagem banco

- a. Padronização das nomenclaturas das tabelas.
 - i. A nomenclatura dos objetos obedecem um padrão com prefixo(ex: TB para tabelas, VW para views, TR para triggers...) ?
- b. Padronização das nomenclaturas dos campos (Os campos obedecem um padrão das nomenclaturas com prefixo indicando o tipo de campo, ex: Id ou cd para código, nu para número, vl para valores...).
- c. Chaves das tabelas.
 - i. As tabelas possuem chaves primárias, para persistência dos dados?

- ii. As tabelas possuem chaves estrangeiras mantendo os dados de forma coerente?

2.Modelagem Sistema

- a. Diagrama de UML(Diagrama de classes)
 - i. Foi estruturado o diagrama das classes do projeto?
- b. Fluxograma
 - i. Foi desenvolvido o fluxograma do projeto?
 - ii. O fluxograma possui as indicações corretamente?
 - iii. No caso de estrutura condicional, foi identificado corretamente no fluxograma?
- c. Caso de uso
 - i. Foi desenvolvido o caso de uso?
 - ii. O caso de uso possui os atores de acordo com suas permissões de acessos às telas?
 - iii. Caso de uso indica o fluxo de acordo com sistema criado?
 - iv. Modelos atualizados de acordo com mudanças no sistema?

3.Front End.

- a. Indentação do código.
 - i. O código estava indentado corretamente?
 - ii. O código estava limpo de acordo com os padrões de clean-code?
- b. Experiência do usuário.
 - i. O sistema é intuitivo?
 - ii. O sistema traz uma boa experiência no momento da navegação?
 - iii. Separação entre HTML, CSS e JS respeitada?
 - iv. JS organizado em funções reutilizáveis?
 - v. Responsividade implementada com media queries?
ex:

```
.container {  
  width: 80%;  
}
```



```
@media screen and (max-width: 600px) {  
  .container {  
    width: 100%;  
    font-size: 14px;  
  }  
}
```

- vi. Uso de bibliotecas/frameworks (ex: Bootstrap, Vue, Angular ou React) de forma consciente?

4.Back End

- a. Indentação do código.
 - i. O código estava indentado corretamente?
 - ii. O código estava limpo de acordo com os padrões de clean-code?
- b. funções do sistema.
 - i. As funcionalidades do sistema estão de acordo com as documentações.
- c. Padrões RESTful implementados corretamente?
 - i. A Api está aceitando requisições de POST,PUT,GET,DELETE?
 - ii. Organização de pacotes (controller, service, repository, model).
 - iii. Validação de entrada usando annotations (ex: @NotNull, @NotBlank)?
 - iv. Segurança básica com Spring Security ou JWT?
- d. Conexão com banco de dados funcional.
 - i. O sistema está conectado com o banco de dados?
 - ii. Uso adequado de DTOs para evitar exposição de entidades?
 - iii. Requisições testadas com Postman,Insomnia ou outra ferramenta?
 - iv. Configurações externas em application.properties ou .yaml?

5.Ferramenta de gerenciamento de atividades.

- a. Definição das etapas.
 - i. As estampas estavam definidas adequadamente, contendo: To do, In Progress, Quality Analysis and Done ?
- b. Definição das atividades.
 - i. As atividades estavam bem descritas?
 - ii. As atividades tiveram pontuações ou feedBack do desenvolvedor sobre as ações feitas?

6.Relatório.

- a. Relatório do sistema.
 - i. Relatório do sistema foi entregue nas regras da ABNT?
 - ii. Foi detalhado horário de cada reunião e participantes?
 - iii. Foi pontuada qual dor o sistema soluciona?

7.Acessibilidade

- a. Textos alternativos usados em imagens (alt)?
- b. Função Dark Funcional?

8.UI/UX Design

- a. Foi criado o Figma?.
- b. Foi utilizado wireframes no planejamento da interface(Figma)?

- c. Foi aplicado um padrão visual seguido em todos os componentes da interface.
- d. Foi aplicado um padrão de paleta de cores alinhada com o propósito do sistema?
- e. Os botões e elementos clicáveis são facilmente identificáveis?
- f. O sistema possui mensagens de erro claras e amigáveis?

9. Organização do Projeto

- a. Estrutura de pastas está organizada e sem arquivos desnecessários?
- b. Nome dos arquivos e pastas seguem convenções padronizadas?
- c. Cada integrante realizou ao menos 15 commits ()?
 - i. Para o commit ser válido é necessário ter alteração condizente com o projeto.
 - ii. Mensagem dos commits de acordo com a alteração no projeto?
 - iii. Caso não atendeu os 15 commits, foi justificado o motivo (Ter feito todos os commits em um dia só não é válido como resposta)?
- d. Git está disponível?
- e. Presença de testes automatizados ou manuais descritos.
 - i. Foi documentado cenários de testes?
 - ii. Foi documentada a organização de testes?