



# Manutenção de Equipamentos

UFPR - SEPT - TADS  
Desenvolvimento WEB-II

---

## ENUNCIADO

Vocês deverão desenvolver um sistema de Controle de Manutenção de Equipamentos. Este sistema é baseado em solicitações de serviços e o histórico de alteração de estado deve ser mantido.

O acesso ao sistema é feito por meio de 2 perfis: Cliente e Funcionário. Todas as funcionalidades necessitam login no sistema, exceto o Autocadastro de clientes e o próprio Login.

- **RF001 - Autocadastro:** Qualquer pessoa pode se cadastrar no sistema, indicando: CPF (único), Nome, E-mail (único), Endereço completo, Telefone. Para o endereço o usuário deve digitar o CEP e os dados devem ser preenchidos (usando API ViaCEP). Mesmo assim, todos os dados do endereço devem ser armazenados. O login será o e-mail da pessoa, e uma senha aleatória de 4 números, que será enviada por e-mail no ato do autocadastro.
- **RF002 - Login:** Qualquer pessoa cadastrada (qualquer perfil) pode fazer login no sistema informando seu e-mail e senha. O sistema automaticamente identifica o perfil do usuário.

## PERFIL CLIENTE

- **RF003 - Página Inicial de Cliente:** Imediatamente após o login deve ser mostrada uma lista com todas as solicitações do cliente, ordenadas de forma crescente por data/hora. Nesta tela devem ser apresentadas: Data/Hora da solicitação, descrição do equipamento (limitado a 30 caracteres) e Estado da Solicitação. Deve ser apresentado um botão para Visualizar a Solicitação (RF008), onde são apresentadas todas as informações e o histórico completo de atualizações. Também deve ser apresentado um botão conforme a ação que o usuário deve efetuar (se não houver ação, não mostrar botão). As ações são as seguintes, conforma o estado:
  - a. **ORÇADA:** Botão Aprovar/Rejeitar Serviço, que apresenta a tela de Mostrar orçamento (RF005)
  - b. **APROVADA:** Sem botão de ação



# Manutenção de Equipamentos

UFPR - SEPT - TADS  
Desenvolvimento WEB-II

---

- c. REJEITADA: Botão Resgatar Serviço (RF009)
  - d. ARRUMADA: Botão Pagar Serviço (RF010)
  - e. Outros estados: Botão Visualizar Serviço
- **RF004 - Solicitação de Manutenção:** O cliente registra uma solicitação de manutenção que deve conter: Descrição do equipamento, Categoria do equipamento, Descrição do defeito. Esta solicitação é armazenada contendo data e hora e também o estado ABERTA, que vai para a empresa apresentar um orçamento.
  - **RF005 - Mostrar orçamento:** Após a empresa apresentar um orçamento, o usuário deve poder aprovar ou rejeitar o serviço. Deve ser apresentada uma tela contendo todos os dados da solicitação e o preço orçado, com destaque. Deve ser apresentado um botão para Aprovar o Serviço e outro para Rejeitar o Serviço.
  - **RF006 - Aprovar Serviço:** Na tela do RF005, ao clicar no botão Aprovar o Serviço, o sistema deve mostrar uma mensagem de "Serviço Aprovado no Valor R\$ xxxx" e, ao clicar OK, deve ser redirecionado para a tela da RF003. Então a solicitação passa para o estado APROVADA.
  - **RF007 - Rejeitar Serviço:** Na tela do RF005, ao clicar no botão Rejeitar o Serviço, deve ser apresentada uma tela para que o usuário possa escrever o motivo da rejeição. Ao confirmar, deve mostrar uma mensagem de "Serviço Rejeitado". Então a solicitação passa para o estado REJEITADA.
  - **RF008 - Visualizar Serviço:** Esse requisito vem do RF003. Deve ser aberta uma tela contendo todos os dados da solicitação e, na parte de baixo, todos os passos pelos quais a solicitação passou, mostrando data/hora e o funcionário que efetuou. Devem ser apresentados botões de ação
  - **RF009 - Resgatar Serviço:** Esse requisito vem do RF003. A solicitação que estava REJEITADA passa para APROVADA e entra novamente no fluxo. Deve ser armazenado no histórico da solicitação que ela passou de rejeitada para aprovada em determinada data/hora.
  - **RF010 - Pagar Serviço:** Assim que a solicitação entrar no estado AGUARDANDO PAGAMENTO significa que ele já foi arrumado. Deve ser apresentada uma tela com os dados da solicitação e com o valor em destaque, mais um botão que simplesmente confirma o pagamento do pedido, registrando a data/hora do pagamento.



# Manutenção de Equipamentos

UFPR - SEPT - TADS  
Desenvolvimento WEB-II

---

## PERFIL FUNCIONÁRIO

- **RF011 - Página Inicial de Funcionário:** Devem ser mostradas todas as solicitações no estado ABERTA, pois necessitam de atenção imediata do funcionário. Devem ser mostradas data/hora da abertura, nome do cliente, e descrição do produto, limitado a 30 caracteres. Para estas solicitações deve ser apresentado um botão para Efetuar Orçamento (RF012).
- **RF012 - Efetuar Orçamento:** Esse requisito vem do RF012. Deve ser aberta uma tela contendo todos os dados (completos) da solicitação e do cliente. Então o funcionário coloca um valor de orçamento que é registrado. Deve ser registrado que o orçamento foi feito por este funcionário logado, contendo data e hora. A solicitação deve passar para o estado ORÇADA.
- **RF013 - Visualização de Solicitações:** O funcionário lista todas as solicitações, que podem ser filtradas como: HOJE, PERÍODO DE DATAS DE ABERTURA (início e fim), ou TODAS. As solicitações devem ser mostradas em ordem crescente por data/hora. Só poderão aparecer solicitações REDIRECIONADAS, se este funcionário for o destino do redirecionamento. Deve ser usada a seguinte escala de cores:
  - a. Cinza : estado ABERTA
  - b. Marrom : estado ORÇADA
  - c. Vermelho : estado REJEITADA
  - d. Amarelo : estado APROVADA
  - e. Roxo: estado REDIRECIONADA
  - f. Azul : estado ARRUMADA
  - g. Alaranjado : estado PAGA
  - h. Verde : estado FINALIZADA

Devem ser apresentados links/botões de ação para cada solicitação em seu estado atual:

- a. ABERTA: botão/link para Efetuar Orçamento (RF012)
  - b. APROVADA/REDIRECIONADA: botão/link para Efetuar Manutenção (RF014)
  - c. PAGA: botão/link para Finalizar Solicitação (RF016)
- **RF014 - Efetuar Manutenção:** Deve ser aberta uma tela contendo todos os dados da



# Manutenção de Equipamentos

UFPR - SEPT - TADS  
Desenvolvimento WEB-II

---

solicitação e do cliente. Nesta tela o funcionário pode: 1) Efetuar a manutenção; ou 2) Redirecionar Manutenção. Quando ele efetuar a manutenção devem ser mostrados 2 campos para que o funcionário digite: a) Descrição da Manutenção; e b) Orientações para o Cliente. Deve ser registrada a data/hora da manutenção e qual foi o funcionário que a fez. A solicitação deve passar para o estado ARRUMADA. Se ele quiser redirecionar, o requisito RF015 deve ser executado.

- **RF015 - Redirecionar Manutenção:** Esse requisito vem do RF014. Se o funcionário que abriu a solicitação não se sentir capaz de corrigir o problema, ele pode redirecionar para um outro funcionário. Ele escolhe em uma caixa de seleção para qual funcionário será redirecionado, e efetua o redirecionamento. A solicitação passa para o estado REDIRECIONADA, e a alteração é armazenada no histórico, com data/hora, funcionário origem e funcionário destino. Uma solicitação pode ser redirecionada infinitas vezes e não pode ser redirecionada para si mesmo.
- **RF016 - Finalizar Solicitação:** Ao finalizar a solicitação, ela passa ao estado FINALIZADA e é registrada a data/hora da finalização, bem como o funcionário responsável pela finalização.
- **RF017 - CRUD de Categoria de Equipamento:** (CRUD - Inserção, Remoção, Atualização e Listagem) Um funcionário pode manter categorias de equipamentos. Por exemplo: Notebook, Impressora, Desktop, Microfone.
- **RF018 - CRUD de Funcionários:** (CRUD - Inserção, Remoção, Atualização e Listagem) O funcionário pode manter novos funcionários para acesso ao sistema, com os seguintes dados: e-mail único para login, nome, data de nascimento, senha. O funcionário não pode remover a si mesmo. Se houver somente um funcionário ele não poderá ser removido.
- **RF019 - Relatório de Receitas em PDF:** O funcionário visualiza a receita conseguida pela empresa em um determinado período por meio de um filtro do relatório: data inicial e data final, que podem ser vazias. Os dados devem ser mostrados agrupados por dia.
- **RF020 - Relatório de Receitas por Categoria em PDF:** O funcionário visualiza a receita conseguida pela empresa, desde sempre, agrupado por categoria de equipamento..

Toda e qualquer suposição, que não esteja definida aqui e que a equipe faça, deve ser devidamente documentada e entregue em um arquivo .doc/.odt que acompanha o trabalho.



# Manutenção de Equipamentos

UFPR - SEPT - TADS  
Desenvolvimento WEB-II

---

## DADOS INICIAIS PARA TESTES

Deve-se apresentar um conjunto mínimo de dados previamente cadastrados (pode-se apresentar mais):

- 2 funcionários (Maria e Mário)
- 4 clientes (João, José, Joana, Joaquina)
- 5 categorias (Notebook, Desktop, Impressora, Mouse, Teclado)

Cria uma massa de solicitações relevante para testar todos os requisitos do sistema, com datas, horas, históricos de alteração dos status das solicitações. Pelo menos 20 solicitações devem ser apresentadas, com estados diferentes, datas distintas, funcionários e clientes diferentes.

## REQUISITOS NÃO-FUNCIONAIS

Os requisitos não-funcionais deste sistema são:

- Leitura de telas deve ser bem elaborado;
- Deve ser usado DHTML (html/xhtml, css, dom e javascript);
- Deve-se usar as tecnologias vistas em aula: Angular, REST, Spring Boot. Bem como as boas práticas de programação: Padrões de Projeto, Repository, Serviços, etc;
- Usar padrão *standalone* do Angular v17, para criação de componentes;
- Deve-se seguir boas práticas de programação e orientação a objetos: ocultamento de informações, baixo acoplamento, nomeação de atributos, classes e métodos, etc;
- Deve-se usar um *framework* para desenvolvimento das telas. Sugere-se o Bootstrap, Material ou Tailwind. Também deve-se usar um conjunto de bibliotecas Javascript para alterar o comportamento de telas, de forma dinâmica, quando necessário. Sugere-se o jQuery;
- Todos os campos devem possuir validação tanto no *front-end* (Angular) como no *back-end* (Spring);



# Manutenção de Equipamentos

UFPR - SEPT - TADS  
Desenvolvimento WEB-II

---

- Todas as senhas devem ser criptografadas usando Hash SHA-256 + SALT (pesquisar o que é isso :-);
- Todas as tabelas no banco de dados (exceto a de endereço com cidade/estado) devem estar normalizadas (3FN) e devem seguir um padrão de codificação, inclusive as que não possuem cadastro e devem estar previamente preenchidas;
- O preenchimento do endereço do Cliente deve ser feito de forma automática consultando o CEP com a API Viacep (<https://viacep.com.br/>);
- No banco de dados deve ser armazenado o endereço completo, mas não há necessidade de normalizar Cidade e Estado;
- Queries no banco de dados devem favorecer o desempenho por meio de JOINS e boas práticas de consultas;
- No caso da entrega do protótipo, todas as funcionalidades devem ser implementadas e todos os dados devem ser fictícios;
- Todas as datas e valores monetários devem ser entrados e mostrados no formato brasileiro;
- Todos os campos que tiverem formatação devem possuir máscara;
- Todas as datas poderão ser entradas através de calendários;
- Qualquer tipo de remoção deve ser confirmada antes de ocorrer;
- Remoções devem usar um mecanismo de desativação dos registros para evitar problemas de integridade referencial;
- O sistema será testado usando o navegador FIREFOX, versão mais recente.

## REQUISITOS MÍNIMOS PARA DEFESA

São requisitos mínimos para a equipe ir para a defesa:

1. Implementação em Angular (v17+ e *standalone*) com Spring Boot e REST para comunicação;
2. O sistema deve estar rodando;
3. A API apresentada deve estar integrada ao front-end, API só com testes (ex Postman) serão desconsideradas;
4. Deve-se ter o banco de dados projetado, normalizado e implementado;



# Manutenção de Equipamentos

UFPR - SEPT - TADS  
Desenvolvimento WEB-II

---

5. Deve-se ter um banco de dados preenchido com informações relevantes para os testes;
6. Os seguintes requisitos **devem estar completamente funcionais** dentro do software (menus, rotas, etc), fluxo completo desde a tela até o banco de dados, sem erros e de forma consistente;
  - a. RF001
  - b. RF002
  - c. RF003
  - d. RF004
  - e. RF005
  - f. RF006
  - g. RF011
  - h. RF012
  - i. RF017
  - j. RF018