**1. Introdução**

**1.1. Propósito**

O objetivo deste documento é descrever detalhadamente os **requisitos funcionais** do sistema **ERP\_Ofício\_Nobre**, voltado para a gestão de uma marcenaria. O sistema permitirá o controle de estoque, gerenciamento de orçamentos e ordens de serviço, além de possibilitar interação entre **Clientes**, **Funcionários** e **Administradores**.

**1.2. Escopo**

O sistema será composto por um **backend (API REST com banco PostgreSQL)** e um **frontend (aplicativo Flutter)**, que fornecerão funcionalidades específicas de acordo com o tipo de usuário autenticado:

* **Administrador:** controle total do sistema.
* **Funcionário:** criação de orçamentos e ordens de serviço.
* **Cliente:** acompanhamento de projetos e aprovações.

## 2. Atores e Perfis

| **Ator** | **Descrição** |
| --- | --- |
| **Administrador** | Gerencia usuários, produtos, serviços, estoque e finanças. Recebe alertas de estoque baixo e atualizações de ordens de serviço. |
| **Funcionário** | Responsável por cadastrar clientes, emitir orçamentos e ordens de serviço, além de realizar visitas a clientes. |
| **Cliente** | Acompanha seus serviços, aprova ordens de serviço e pode selecionar preferências (como cores ou materiais). |
| **Desenvolvedor** | Responsável pela manutenção e atualização técnica do sistema. |
| **WebServer** | Ator externo que intermedeia solicitações e respostas entre backend e frontend. |

**3. Casos de Uso Principais**

| **Código** | **Caso de Uso** | **Atores** | **Descrição** |
| --- | --- | --- | --- |
| UC01 | **Logar no sistema** | Todos | Permite autenticação de usuários via e-mail e senha, com redirecionamento conforme o perfil. |
| UC02 | **CRUD Cliente** | Admin | Cadastrar, listar, atualizar e excluir clientes. |
| UC03 | **CRUD Funcionário** | Admin | Cadastrar e gerenciar informações de funcionários. |
| UC04 | **CRUD Produto** | Admin | Controlar o cadastro, estoque e valores de produtos. |
| UC05 | **CRUD Serviço** | Admin | Cadastrar e gerenciar tipos de serviços prestados. |
| UC06 | **CRUD Orçamento** | Admin | Criar, editar e enviar orçamentos vinculados a clientes e produtos. |
| UC07 | **CRUD Ordem de Serviço** | Admin, Funcionarios | Emitir e atualizar ordens de serviço baseadas em orçamentos aprovados. |
| UC08 | **CRUD Balanço Financeiro** | Admin | Registrar despesas e receitas associadas a serviços e materiais. |
| UC09 | **Visualizar todos os serviços** | Admin | Permite que o administrador acompanhe o status de todos os serviços em execução. |
| UC10 | **Receber alerta de estoque baixo** | Admin | Notifica o administrador quando o estoque de produtos atinge o nível mínimo. |
| UC11 | **Receber alerta de atualização de serviços** | Admin | Informa sobre alterações de status em ordens de serviço. |
| UC12 | **Visita a cliente** | Funcionário | Registra visitas técnicas e observações associadas a clientes. |
| UC13 | **Emitir ordem de serviço** | Funcionário | Gera uma OS a partir de um orçamento aprovado. |
| UC14 | **Visualizar seus serviços** | Cliente | Exibe lista e detalhes das ordens de serviço do cliente. |
| UC15 | **Aprovar ordem de serviço** | Cliente | Permite que o cliente aprove a execução do serviço antes do início. |
| UC16 | **Selecionar cores para o serviço** | Cliente | Cliente escolhe cores, acabamentos ou detalhes estéticos para o serviço. |
| UC17 | **Adicionar informações** | Desenvolvedor | Inserir informações técnicas ou atualizações no sistema. |
| UC18 | **Solicitar informações** | WebServer | Solicita ou envia dados entre o app e o backend. |

**4. Regras de Negócio**

1. Apenas **Funcionário** ou **Administrador** pode criar orçamentos.
2. Um **Orçamento** só pode virar **Ordem de Serviço** após aprovação do cliente.
3. O sistema deve atualizar automaticamente o estoque ao concluir uma OS.
4. O **Administrador** recebe alertas de estoque baixo e mudanças de status.
5. Cada **Cliente** só pode visualizar serviços vinculados ao seu cadastro.
6. Não é possível excluir um produto vinculado a um orçamento ativo.
7. O **lucro** de um orçamento é calculado com base em (Material + Mão de Obra) \* %Lucro.
8. Alterações financeiras devem ser registradas em log de auditoria.

## 5. Fluxos Alternativos (exemplos)

* **UC01 – Logar no sistema:**  
  Se o usuário informar credenciais inválidas, o sistema exibirá mensagem de erro e permitirá nova tentativa.
* **UC06 – Criar Orçamento:**  
  Caso o cliente ainda não esteja cadastrado, o funcionário poderá registrá-lo rapidamente durante o processo de criação do orçamento.

## 6. Requisitos de Interface

* Interface responsiva para dispositivos móveis (Android).
* Dashboard inicial com indicadores de estoque, ordens em andamento e orçamentos pendentes.
* Campos obrigatórios destacados em vermelho.
* Botão de confirmação com duplo clique para ações críticas (ex: exclusão de produto ou OS).

## 7. Requisitos de Dados

* Cada registro de **produto**, **cliente**, **funcionário**, **serviço**, **ordem** e **orçamento** deverá possuir identificador único (UUID).
* Logs de alteração deverão armazenar o usuário responsável e data/hora.
* As tabelas devem seguir o modelo definido no banco PostgreSQL (vide DDL do projeto).