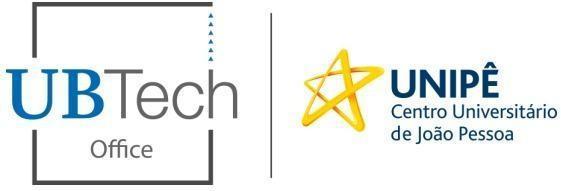
**Centro Universitário de João Pessoa - UNIPÊ Fábrica de Software do UBTech Office/UNIPÊ**

**Documento de Visão de Sistema**

# **OBJETIVO DESTE DOCUMENTO**

O objetivo deste documento é apresentar uma visão geral do AutoGestor, a plataforma de gerenciamento de oficinas mecânicas que será desenvolvida. Serão abordados tópico como escopo do produto, não escopo do produto, descrição dos envolvidos, visão geral do produto, restrições e diagramas representativos.

# **HISTÓRICO DE REVISÃO**

|  |  |  |  |
| --- | --- | --- | --- |
| ***Histórico de Revisão*** | | | |
| **Data** | **Autor** | **Descrição** | **Versão** |
| 25/02 | Alicia | Início do documento de visão | 1.0 |
| 25/02 | Alicia | Criação do nome e logo do projeto | 1.1 |
| 26/02 | Alicia | Estruturação dos requisitos funcionais | 1.2 |
| 26/02 | Alicia | Estruturação dos requisitos não funcionais | 1.3 |
| 26/02 | Alicia | Criação do Escopo e Não Escopo | 1.4 |
| 28/02 | Alicia | Criação dos diagramas de casos de uso e DER | 1.5 |
| 28/02 | Alicia | Criação do diagrama sequência ou fluxo BPMN | 1.6 |
| 01/03 | Alicia | Revisão final, ajustes e revisão geral | 1.7 |

Obs.: O redimensionamento das colunas das tabelas poderá ser alterado caso haja necessidade.

1. **ESCOPO DO PRODUTO**

A plataforma AutoGestor terá como objetivo principal otimizar a gestão de oficinas mecânicas, oferecendo controle financeiro, alocação de serviços e funcionários, gestão de estoque e emissão de notas fiscais. O sistema será baseado na web e incluirá funcionalidades essenciais para o gerenciamento eficiente da oficina.

O sistema deverá ser capaz de fornecer os seguintes serviços para atender as necessidades dos interessados:

Perspectiva do Usuário Atendente:

* O sistema deve permitir o cadastro e gerenciamento de clientes, garantindo a atualização de seus dados sempre que necessário.
* ● O sistema deve possibilitar a emissão de ordens de serviço, vinculando-as aos clientes e mecânicos responsáveis.
* ● O sistema deve oferecer um módulo de agendamento e organização de serviços, otimizando a distribuição de tarefas.
* ● O sistema deve fornecer suporte administrativo para mecânicos e clientes, incluindo comunicação e esclarecimento de dúvidas.

Perspectiva do Usuário Supervisor

* ● O sistema deve permitir o gerenciamento da equipe de mecânicos, distribuindo atribuições conforme a demanda.
* ● O sistema deve disponibilizar o monitoramento em tempo real das ordens de serviço em andamento.
* ● O sistema deve possibilitar a aprovação e priorização de serviços conforme critérios estabelecidos pela empresa.
* ● O sistema deve oferecer ferramentas para controle e análise do desempenho da equipe de mecânicos.

Perspectiva do Usuário Mecânico

* ● O sistema deve fornecer acesso rápido e intuitivo às ordens de serviço designadas a cada mecânico.
* ● O sistema deve permitir o registro do tempo gasto em cada serviço para controle de produtividade.
* ● O sistema deve possibilitar a atualização do status dos serviços, informando sobre o andamento das tarefas.
* ● O sistema deve permitir que o mecânico solicite peças e materiais ao estoque, garantindo disponibilidade para execução dos serviços.

Perspectiva do Usuário Cliente

* ● O sistema deve permitir que o cliente receba notas fiscais e comprovantes dos serviços prestados.
* ● O sistema deve oferecer acesso ao histórico de serviços realizados, permitindo consultas rápidas e detalhadas.
* ● O sistema deve disponibilizar um módulo de agendamento de serviços, permitindo a escolha de datas e horários convenientes.
* ● O sistema deve fornecer acompanhamento em tempo real do status dos serviços em andamento.
* ● O sistema deve enviar notificações via WhatsApp sobre atualizações de serviços e lembretes periódicos de manutenção preventiva.
* ● O sistema deve permitir a participação do cliente em um programa de fidelidade, oferecendo descontos e vantagens para serviços futuros.

Serviços Adicionais

* ● O sistema deve garantir a segurança e integridade dos dados dos clientes e mecânicos, seguindo as normas de privacidade.
* ● O sistema deve fornecer relatórios gerenciais para análise de desempenho da equipe e fluxo de serviços.

# **NÃO ESCOPO DO PRODUTO**

O AutoGestor **NÃO** deverá fornecer os serviços para atender as necessidades dos interessados:

* Versões para aplicativos móveis nativos (inicialmente será apenas web).
* Integração direta com sistemas de contabilidade externa.
* Personalização visual detalhada para cada oficina.

# **DESCRIÇÃO DOS ENVOLVIDOS**

Os principais envolvidos na plataforma serão Atendente, Supervisor, Mecânico e cliente.

# Resumo dos Usuários

|  |  |  |
| --- | --- | --- |
| Nome | Responsabilidades | Perfil |
| Atendente | São os usuários que vão: Cadastrar e gerenciar clientes; Emitir ordens de serviço; Organizar e agendar serviços; Oferecer suporte administrativo para mecânicos e clientes | Utilizador do Aplicativo Web como usuário logado |
| Supervisor | São os usuários que vão: Gerenciar mecânicos e suas atribuições; Monitorar ordens de serviço em andamento; Aprovar e priorizar serviços; Controlar o desempenho da equipe. | Utilizador do Aplicativo Web como usuário logado |
| Mecânico | São os usuários que vão: Acessar ordens de serviço; Registrar o tempo gasto em cada serviço; Atualizar o status dos serviços; Solicitar peças e materiais ao estoque. | Utilizador do Aplicativo Web como usuário logado |
| Cliente | São os usuários que vão: Acompanhar o status dos serviços em andamento; Acessar o histórico de atendimentos; Receber notas fiscais e comprovantes de serviço; Receber notificações via WhatsApp sobre atualizações e manutenções preventivas; Participar do programa de fidelidade para descontos e vantagens futuras. | Utilizador do Aplicativo Web como usuário logado |

# **VISÃO GERAL DO PRODUTO**

O AutoGestor será uma ferramenta baseada na web para gerenciar oficinas mecânicas, otimizando fluxos internos e garantindo maior controle sobre operações financeiras e operacionais. A plataforma permitirá aos atendentes organizar os agendamentos, aos supervisores gerenciar a equipe de mecânicos, aos mecânicos acessar ordens de serviço e aos clientes acompanhar o status dos serviços.

* 1. **Requisitos Funcionais**

|  |  |  |
| --- | --- | --- |
| **Nº** | **Nome** | **Descrição** |
| RF001 | Acesso a plataforma | Os usuários devem ser capazes de acessar a plataforma AutoGestor através de um navegador web em qualquer dispositivo. |
| RF002 | Cadastro de Funcionários | Os usuários devem poder cadastrar e gerenciar mecânicos na plataforma. |
| RF003 | Controle de Serviços | Os usuários devem poder criar, editar e monitorar ordens de serviço, garantindo o acompanhamento do status. |
| RF004 | Controle de Estoque | O sistema deve permitir o gerenciamento de peças e insumos, controlando a disponibilidade para os serviços. |
| RF005 | Emissão de Notas Fiscais | O sistema deve gerar notas fiscais eletrônicas para os clientes de acordo com os serviços prestados. |
| RF006 | Relatórios e Dashboards | O sistema deve possibilitar a geração de relatórios financeiros e operacionais para análise de desempenho. |
| RF007 | Integração com WhatsApp | O sistema deve permitir o envio automático de atualizações sobre serviços e lembretes de manutenção preventiva via WhatsApp. |
| RF008 | Programa de Fidelidade | O sistema deve gerenciar pontos e benefícios para clientes recorrentes, possibilitando vantagens em serviços futuros. |
| RF009 | Gestão de Pagamentos | O sistema deve registrar formas de pagamento e associá-las às respectivas notas fiscais emitidas. |
| RF010 | Registro de Materiais Utilizados | O sistema deve permitir o controle detalhado dos materiais alocados a cada serviço, garantindo a rastreabilidade. |
| RF002 | Cadastro de Funcionários | Os usuários devem poder cadastrar e gerenciar mecânicos na plataforma. |

* 1. **Requisitos Não Funcionais**

|  |  |  |
| --- | --- | --- |
| **Nº** | **Nome** | **Descrição** |
| **RNF001** | Segurança da Informação | O sistema deve garantir a segurança e confidencialidade dos dados dos clientes. |
| **RNF002** | Usabilidade | A interface do usuário deve ser intuitiva e fácil uso. |
| **RNF003** | Disponibilidade | A plataforma deve estar disponível 24 horas por dia, 7 dias por semana, com mínimo tempo de inatividade. |
| **RNF004** | Desempenho | O sistema deve ser responsivo e fornecer tempos de resposta rápidos, mesmo em momentos de pico de uso. |
| **RNF005** | Escalabilidade | O sistema deve ser capaz de lidar com um aumento significativo no número de usuários e consultas sem comprometer o desempenho. |

**6.1 Requisitos Técnicos**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **REQ TEC** | **Nome** | **Descrição** | **Função** | **Ferramenta e Versão** |
| RT001 | **Plataforma Web** | **O sistema deve ser acessado via navegador web, sem necessidade de instalação local.** | **Full Stack** | **React.js 18, Next.js 14** |
| RT002 | **Responsividade** | **A interface deve ser responsiva para garantir compatibilidade com desktops, tablets e smartphones.** | **Frontend** | **Tailwind CSS 3.3** |
| RT003 | **Compatibilidade de Navegadores** | **O sistema deve ser compatível com os navegadores mais populares: Chrome, Edge, Safari e Firefox.** | **Frontend** | **React.js 18, Next.js 14** |
| RT004 | Segurança da Informação | Deve seguir normas de segurança, incluindo autenticação segura (OAuth 2.0 ou JWT) e criptografia SSL/TLS. | Backend | Spring Security (Java) ou Express.js (Node.js) |
| RT005 | Escalabilidade | O sistema deve suportar crescimento de usuários sem comprometer o desempenho. | Backend | Kubernetes, Docker |
| RT006 | Desempenho | Tempo de resposta inferior a 2 segundos para operações críticas. | Backend | Redis 7, PostgreSQL 15 |
| RT007 | Disponibilidade | Plataforma disponível 24/7 com monitoramento contínuo e recuperação de falhas. | DevOps | AWS EC2, CloudWatch |
| RT008 | Backup Automático | Deve possuir um mecanismo de backup automatizado para evitar perda de dados. | Banco de Dados | PostgreSQL 15, Firebase Backup |
| RT009 | Banco de Dados | Utilizar um banco de dados relacional otimizado para alto volume de transações. | Banco de Dados | PostgreSQL 15, MySQL 8 |
| RT010 | APIs e Integrações | Deve permitir integração com APIs externas, como sistemas de pagamento e envio de notificações. | Backend | Node.js 20, Java 17 |
| RT011 | Controle de Acesso | Deve suportar múltiplos níveis de acesso (cliente, atendente, mecânico, supervisor, administrador). | Backend | Spring Boot (Java) ou Express.js (Node.js) |
| RT012 | Gerenciamento de Estoque | Deve permitir o controle detalhado de peças e insumos utilizados nos serviços. | Backend | Java 17, Node.js 20 |
| RT013 | Relatórios e Dashboards | Deve fornecer relatórios gerenciais e dashboards para análise de desempenho e fluxo de serviços. | Full Stack | Power BI, Metabase, React.js |
| RT014 | Suporte a Pagamentos | Deve permitir diferentes formas de pagamento, como cartão, Pix e dinheiro. | Backend | Stripe API, Mercado Pago API |
| RT015 | Sistema de Notificações | O sistema deve enviar notificações automáticas para clientes sobre status de serviços e lembretes de manutenção. | Backend | Firebase Cloud Messaging (FCM), Twilio |

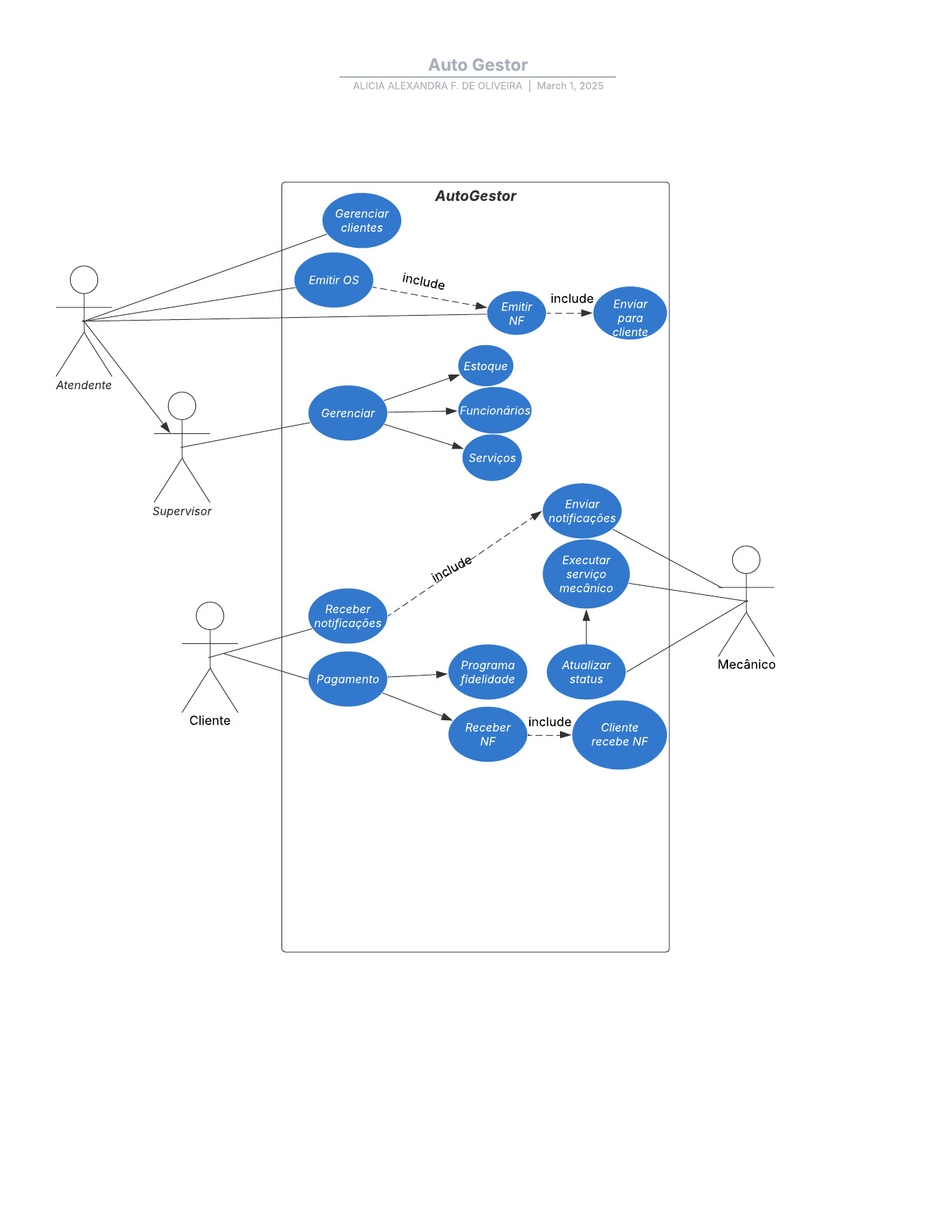
**6.2 Diagrama de Casos de Uso**

O diagrama de casos de uso do Auto Gestor representa as interações entre diferentes atores e funcionalidades do sistema, focado na gestão de serviços automotivos. Nele, vemos que os principais usuários incluem o cliente, o atendente, o supervisor e o mecânico, além do próprio Auto Gestor, que automatiza várias operações.

Os clientes podem receber notificações e notas fiscais, participar de um programa de fidelidade e efetuar pagamentos. Já o atendente e o supervisor têm a responsabilidade de gerenciar clientes, estoque e funcionários, além de emitir ordens de serviço e notas fiscais. O mecânico, por sua vez, executa os serviços e pode atualizar o status das ordens de serviço no sistema.

Além disso, o diagrama destaca a troca de informações entre esses elementos. Por exemplo, a emissão de nota fiscal pode incluir o envio automático para o cliente. O gerenciamento de clientes e funcionários também pode abranger a atualização de status e o envio de notificações. Há também a relação com o controle de estoque, garantindo que as peças e materiais necessários estejam disponíveis para os serviços.

No geral, o Auto Gestor funciona como um sistema centralizador que facilita a administração de todos esses processos, garantindo que cada usuário tenha suas funções bem definidas e que todas as operações fluam de maneira eficiente.



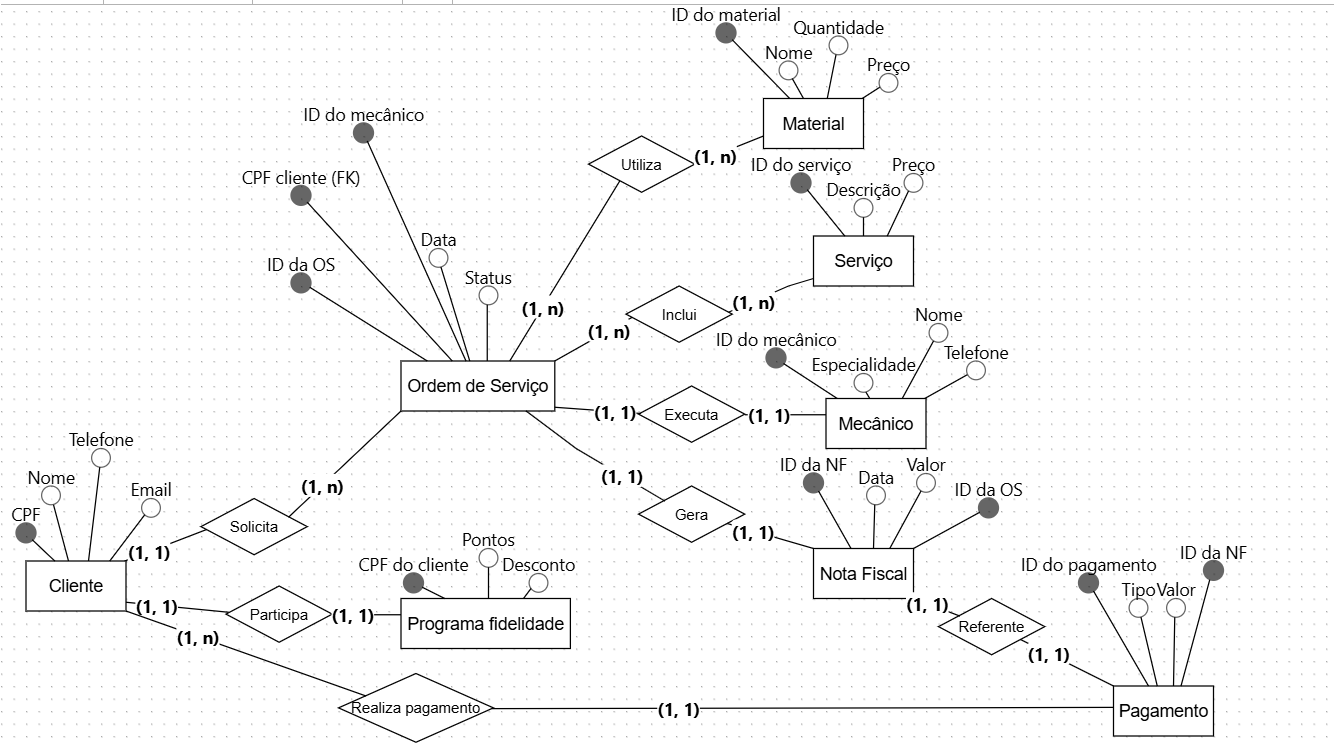
**6.1.1 Diagrama Entidade-Relacionamento**

O diagrama de entidade-relacionamento do AutoGestor organiza de maneira estruturada os principais elementos do sistema, destacando como eles se conectam. No centro de tudo está a Ordem de Serviço, peça-chave que liga clientes, mecânicos, serviços e pagamentos.

O Cliente, identificado pelo CPF, pode solicitar diversas ordens de serviço. Além disso, ele pode participar de um Programa de Fidelidade, acumulando pontos e recebendo descontos. Cada ordem de serviço está associada a um Mecânico, responsável por executar o serviço. Esse mecânico tem um ID próprio, nome, telefone e especialidade.

Dentro da ordem de serviço, há a inclusão de um ou mais Serviços, cada um podendo envolver o uso de Materiais, que possuem nome, preço e quantidade. A ordem de serviço também gera uma Nota Fiscal, contendo um identificador único, data e valor. Essa nota fiscal, por sua vez, está vinculada ao Pagamento, que registra informações como tipo e valor do pagamento.

O modelo garante que todos os processos do sistema estejam bem interligados, permitindo um controle eficiente das operações, desde o atendimento ao cliente até a finalização do serviço e pagamento.



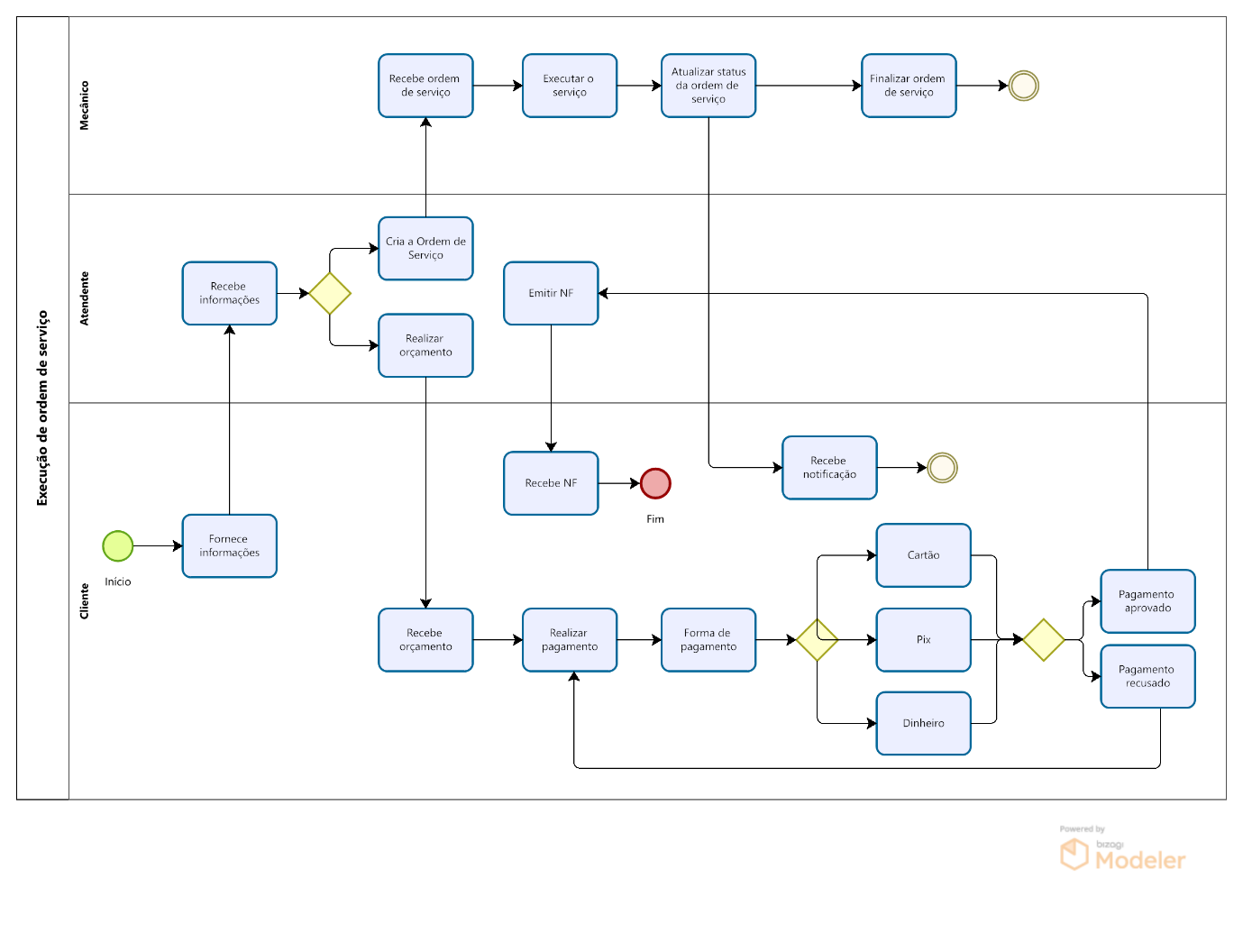
**6.1.2 Diagrama de fluxo BPMN**

O fluxo BPMN do **AutoGestor** descreve o processo de execução de uma ordem de serviço, organizando as etapas entre os principais envolvidos: **cliente, atendente e mecânico**.

Tudo começa com o **cliente fornecendo informações** para iniciar o atendimento. O **atendente** recebe essas informações e realiza um **orçamento**, que é então enviado ao cliente para aprovação. Se aprovado, o atendente cria a **Ordem de Serviço**, formalizando o pedido.

A partir desse ponto, o **mecânico entra em ação**, recebendo a ordem e executando o serviço. Durante o processo, ele pode **atualizar o status** da ordem, garantindo que o fluxo de trabalho seja acompanhado. Assim que o serviço é concluído, a ordem é finalizada.

Enquanto isso, o atendente **emite a Nota Fiscal (NF)**, que é enviada ao cliente. Em seguida, o cliente deve **realizar o pagamento**, escolhendo entre **cartão, Pix ou dinheiro**. O sistema verifica a aprovação do pagamento e, se recusado, o processo pode precisar ser repetido. Caso seja aprovado, o cliente recebe uma **notificação de confirmação** e o fluxo se encerra.



# **7. RESTRIÇÕES**

Descreve as restrições que são impostas ao sistema ou ao processo de desenvolvimento. Para a plataforma AutoGestor, listo as restrições abaixo:

* O sistema deve ter um registro de domínio próprio.
* Deve ser funcional e responsivo no ambiente web.
* Compatível com os navegadores mais populares: **Safari, Edge, Chrome e Firefox**.
* Deve seguir a **LGPD** para proteção de dados dos usuários.
* Deve estar em conformidade com **ISO 27001** para segurança da informação.
* Utilização de **autenticação segura (OAuth 2.0 ou JWT)**.
* Implementação de **múltiplos níveis de acesso** (cliente, atendente, mecânico, supervisor, administrador).
* Todas as comunicações devem ser **criptografadas via SSL/TLS**.
* Tempo de resposta inferior a **2 segundos** para as principais ações.
* Deve ser escalável para suportar crescimento de usuários e serviços.
* Implementação de **backups automáticos** para evitar perda de dados.
* **Monitoramento contínuo** para identificação rápida de falhas.
* Deve permitir integração com **APIs externas**, como serviços de pagamento e envio de notificações.
* Interface seguindo **boas práticas de UX/UI** para facilitar a navegação.
* Os diagramas foram construídos utilizando **BrModelo, Lucidchart** e **Draw.io.**

# **8. POLÍTICA DE VERSIONAMENTO** Descreve como será realizada a política de versionamento do projeto.

* *Gitlab;*
* *Branch Master;*
* *Cada alteração é acompanhada pela equipe antes de cada commit;*
* É mantido um histórico de versão por meio do gitlab e a cada *commit* são executados testes automatizados, por meio do componente de integração contínua do gitlab.

# **9. REFERÊNCIAS**

São referências para este documento de visão:

* **Pressman, Roger S. & Maxim, Bruce R.**. *Engenharia de Software: Uma Abordagem Profissional*. 8ª edição. McGraw-Hill, 2016. - Exploração detalhada de modelagem de sistemas, incluindo diagramas UML e BPMN.
* Descrição da Metodologia – Master of Puppets- <https://desenho-de-software-2019-02.github.io/Documentation/dinamica_e_seminario_2/desenho_metodologia-bpmn/bizagi/>
* ISO/IEC 27001:2013 – Information Security Management Systems. - Padrão internacional utilizado como referência para as diretrizes de segurança do AutoGestor.
* LGPD – Lei Geral de Proteção de Dados (Lei nº 13.709/2018). - Diretrizes de privacidade e proteção de dados aplicáveis ao sistema AutoGestor.
* BrModelo, Lucidchart e Draw.io para a construção dos diagramas de casos de uso, entidade-relacionamento e BPMN.
* GitLab para versionamento e controle das mudanças realizadas durante o desenvolvimento.
* Atas de reuniões do workshop sobre funções e responsabilidades do PO, realizado para aprimoramento do AutoGestor.
* Documentação interna da Fábrica de Software do UBTech Office/UNIPÊ, incluindo discussões sobre requisitos e prototipação.