



**GRADUAÇÃO EM TECNOLOGIA DE ANÁLISE E DESENVOLVIMENTO DE  
SISTEMAS**

**GABRIEL RAMAGLIA TOLENTINO PEREIRA**

**GUILHERME DE MORAES DOMINGUES**

**IAN SOUZA DE OLIVEIRA**

**PEDRO DO PRADO HARDMANN**

**VICTOR DE OLIVEIRA VILAS BOAS**

**Software para Gerenciamento de Estacionamento**

## **Gerenciador de estacionamento**

O projeto consiste no desenvolvimento de um software para gerenciar um estacionamento, focado em facilitar o controle de entradas e saídas de veículos, além da administração de planos mensais para clientes. O sistema é operado por funcionários, que realizam todas as ações no sistema. Clientes ocasionais utilizam o serviço sem a necessidade de cadastro, enquanto os clientes interessados em planos mensais têm seus dados registrados para controle e faturamento. O sistema busca automatizar e organizar os processos operacionais do estacionamento, oferecendo maior praticidade, segurança e eficiência na gestão.

### **Requisitos funcionais**

- RF1 - O sistema deve permitir o cadastro e a exclusão de veículos
- RF2 - O sistema deve calcular o preço do serviço final
- RF3 - O software deverá cadastrar clientes que participem de planos mensais;
- RF4 - O sistema deve permitir remover clientes cadastrados em planos mensais;
- RF5 - O sistema deve armazenar o registro de todos os carros que passaram pelo estacionamento em um histórico;
- RF6 - O sistema deve permitir o administrador acessar ao histórico dos veículos que já foram cadastrados;
- RF7 - Sistema deve exibir as vagas disponíveis;
- RF8 - O sistema deve alertar caso todas as vagas estejam ocupadas ou caso restem poucas vagas;

### **Requisitos não funcionais**

- RNF1 - As tecnologias utilizadas para o desenvolvimento frontend serão HTML, CSS e Javascript;
- RNF2 - A tecnologia utilizada para o desenvolvimento backend será Python;
- RNF3 - O banco de dados deverá ser feito utilizando MySql por meio do Workbench 8.0;
- RNF4 - O código deve ser modular e documentado para facilitar futuras manutenções e atualizações.
- RNF5 - Alterações no plano de mensalistas ou regras de cobrança devem ser facilmente configuráveis sem alterações no código-fonte.

- RNF6 - A interface do sistema deve ser intuitiva, permitindo que funcionários com pouco treinamento consigam operá-la.
- RNF7 - O sistema deve utilizar cores e ícones claros para indicar o status de vagas (ocupada ou livre)

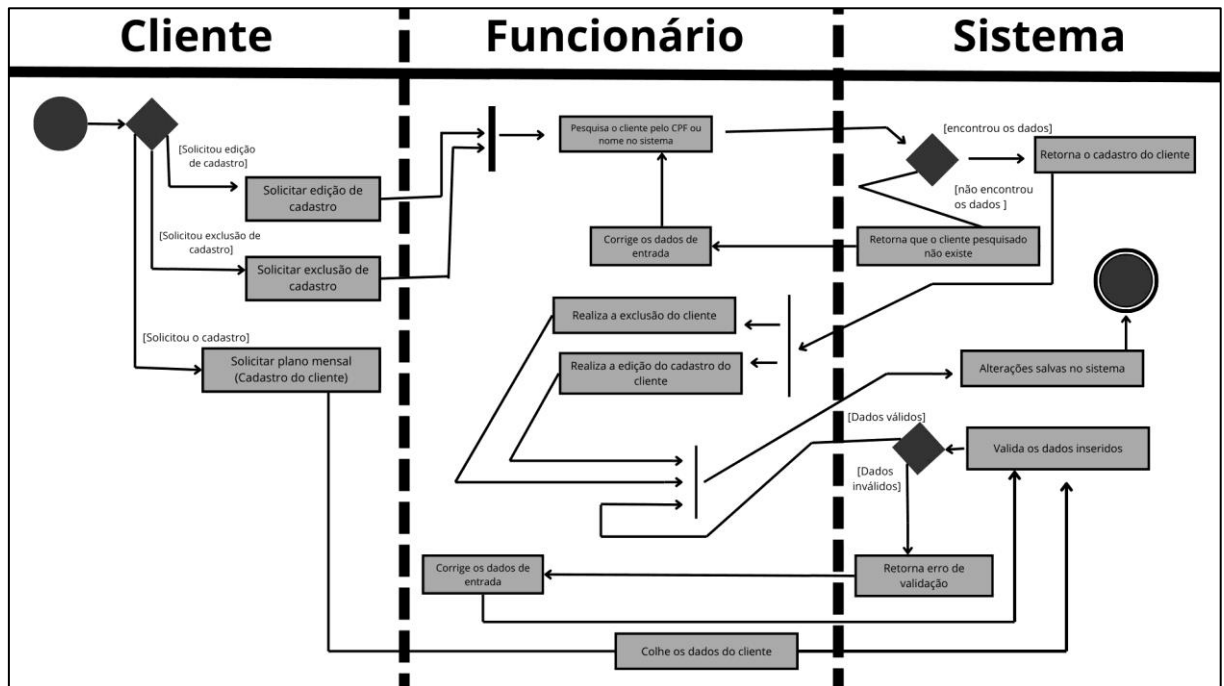
## **Regras de negócio**

- RN01 - O acesso ao histórico será restrito somente ao gestor do estacionamento e NÃO poderá ser editado.
- RN02 - Todo carro cadastrado fará parte do histórico
- RN03 - Não será obrigatório o cadastro do cliente, somente caso o cliente queira ter um plano mensal
- RN04 - O cálculo do pagamento de carros que utilizam o estacionamento de forma avulsa, isto é, que não possuem um plano de pagamento mensal, deve ser feito levando em consideração o tempo de utilização do estacionamento e o tipo de vaga escolhida.
- RN05 - Os clientes que possuem planos realizaram o pagamento mensalmente;
- RN06 - Os planos mensais serão divididos em categorias, de acordo com o tipo da vaga, seja ela padrão, coberta ou para motocicleta
- RN07 - Às vagas do estacionamento serão divididas nas seguintes categorias: vagas padrão, vagas cobertas e vagas para motocicleta.

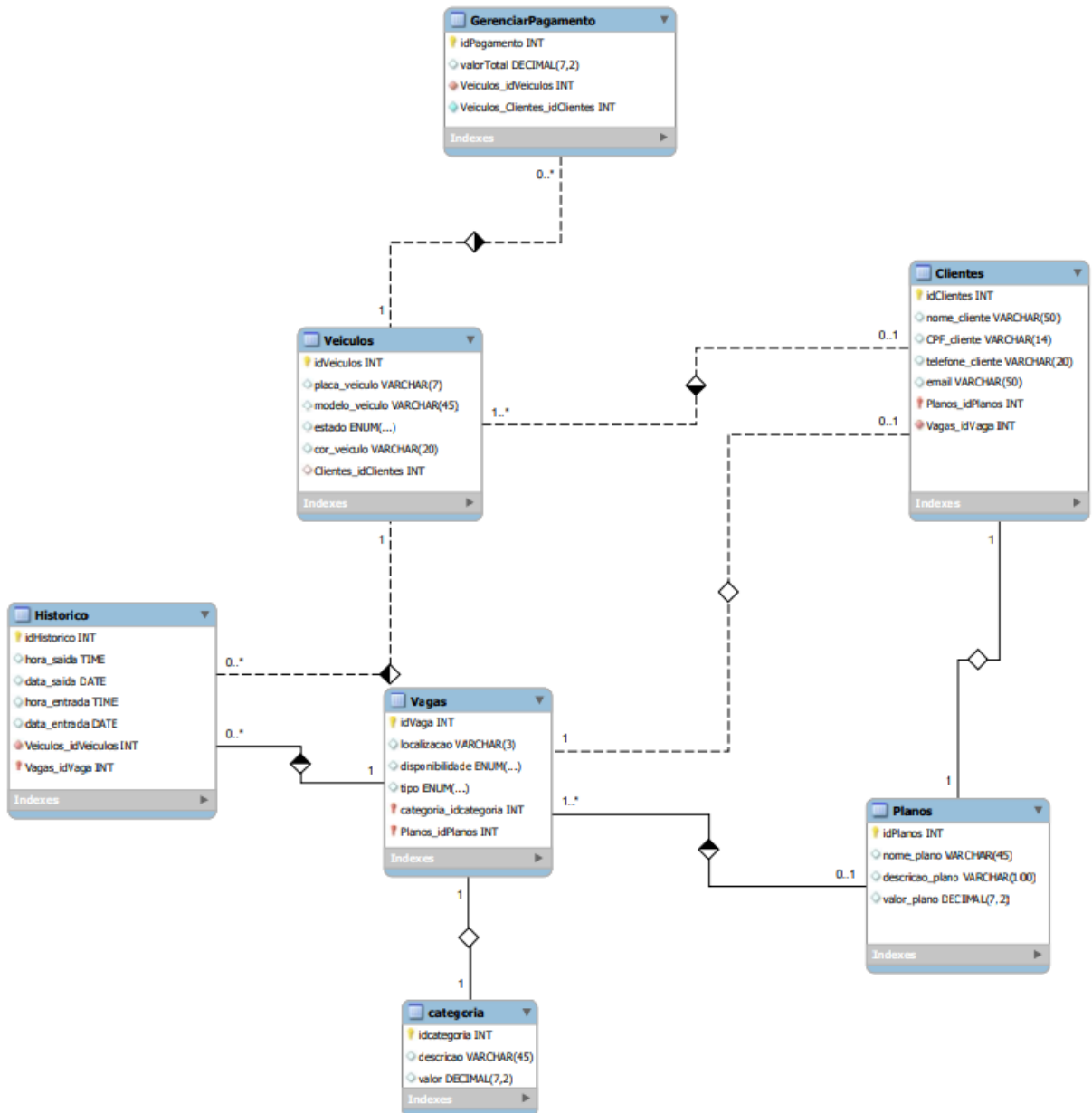
## Diagrama de caso de uso



## Diagrama de Atividades: Processo de cadastro de cliente



## MER: Modelagem Entidade Relacionamento



## Diagrama de Classes:

