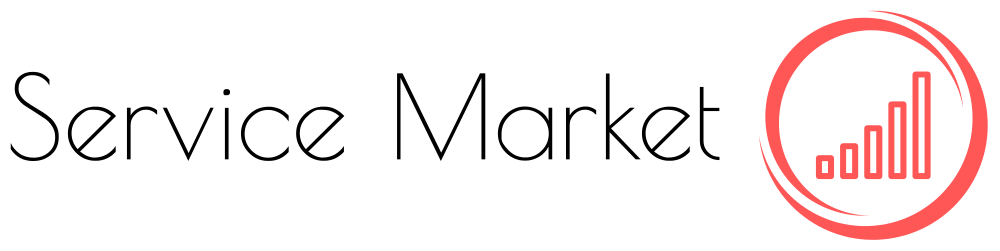
**SERVICE MARKET**

Gerenciamento de serviços domésticos

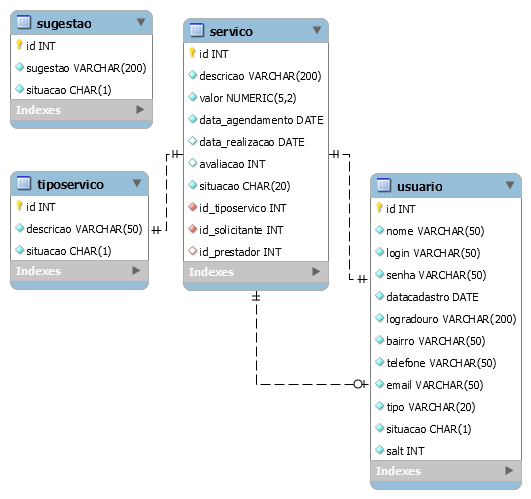
* **Logo:**



* **Função do projeto:** O projeto tem como principal objetivo desenvolver uma ferramenta web para gerenciar a demanda de serviços gerais, tais como: jardinagem, conserto de equipamentos, pintura, instalações elétricas, entre outros. Por meio de um cadastro simples, o usuário pode criar uma solicitação de um determinado tipo de serviço, que será atendida por um prestador. Qualquer um pode se cadastrar como prestador, porém deverá informar quais serviços estará prestando. Os serviços posteriormente poderão ser avaliados e a aplicação também apresentará relatórios sobre os mesmos. Ainda é possível sugerir novos tipos de serviço a serem disponibilizados, porém estes deverão ser aprovados por algum administrador.
* **Requisitos:**

| **ID** | **Descrição** | **Tipo** | **Prioridade** |
| --- | --- | --- | --- |
| RF001 | Manter cadastro de usuários | Funcional | Alta |
| RF002 | Manter cadastro de tipos de serviços | Funcional | Alta |
| RF003 | Manter cadastro de serviços | Funcional | Alta |
| RF004 | Gerenciar solicitações de sugestão de serviços | Funcional | Média |
| RF005 | Disponibilizar avaliações/feedback dos serviços | Funcional | Alta |
| RF006 | Permitir usuário do tipo “Prestador” atender a um serviço | Funcional | Alta |
| RF007 | Manter lista de serviços prestados/solicitados/atendidos | Funcional | Alta |
| RF008 | Emitir relatórios de serviços por data, tipo e/ou prestador/cliente | Funcional | Alta |
| RF009 | Emitir relatório de solicitações por status | Funcional | Alta |
|  |  |  |  |
| RNF001 | Utilizar HTML/JSP e funções JS | Não-funcional | Alta |
| RNF002 | Utilizar banco de dados PostgreSQL | Não-funcional | Alta |
| RNF003 | Utilizar Materialize CSS | Não-funcional | Alta |
| RNF004 | Validar campos de formulários | Não-funcional | Alta |
| RNF005 | Ser WEB | Não-funcional | Alta |
| RNF006 | Utilizar JasperReports | Não-funcional | Média |

* **Modelo ER:**



* **Funções dos integrantes:**

Airan: Criação do modelo ER, documento do projeto, classes .jsp principais (cadastro e movimentação central), servlets, classes de apoio (Conexão, SHA1, Formatação) entidades e DAO’s.

Ismael: Criação de .jsp para relatórios, manutenção e configuração do JasperReports, exportação de relatórios para PDF, criação e configuração na aplicação para classe e funcionalidade de envio de e-mail, manutenção de .jsp’s.