**Centro Estadual de Educação Tecnológica Paula Souza**

**Faculdade de Tecnologia Zona Sul – Dom Paulo Evaristo Arns**

**Davi de brito junior**

**wesley silva dos santos**

**salv**

**São Paulo**

**2025**

DAVI BRITO JUNIOR

WESLEY SILVA DOS SANTOS

**salv**

Trabalho apresentado como requisito parcial para obtenção do título de graduação tecnológica em Desenvolvimento de Software Multiplataforma sob orientação do Prof. Dr. Winston Aparecido Andrade.

São Paulo

2025

**RESUMO**

**ABSTRACT**

**LISTA DE ILUSTRAÇÕES**

[Figura 1 - Home HIKVISION 13](#_Toc191203824)

[Figura 2 - Produtos HIKVISION 14](#_Toc191203825)

[Figura 3 -Soluções HIKVISION 14](#_Toc191203826)

[Figura 4 - Home Ring Alarm 15](#_Toc191203827)

[Figura 5 - Produtos Ring Alarm 16](#_Toc191203828)

[Figura 6 - Planos Ring Alarm 16](#_Toc191203829)

[Figura 7 - Home Dahua Technology 17](#_Toc191203830)

[Figura 8 - Soluções Dahua Technology 18](#_Toc191203831)

[Figura 9 - Produtos Dahua Technology 18](#_Toc191203832)

**LISTA DE TABELAS**

[Tabela 1- Tecnologias do Hardware e Sensores 5](#_Toc191377598)

[Tabela 2 - Tecnologias de Software e Backend 6](#_Toc191377599)

[Tabela 3- Protocolos e Automação 7](#_Toc191377600)

[Tabela 4- Requisitos Funcionais 7](#_Toc191377601)

[Tabela 5- Requisitos não funcionais 9](#_Toc191377602)

[Tabela 6- Comparativo SALV vs. Concorrentes 19](#_Toc191377603)

[Tabela 7- Valores dos componentes 20](#_Toc191377604)

**LISTA DE ABREVIATURAS E SIGLAS**

|  |  |
| --- | --- |
| SALV | Sistema de Alerta Laboratorial com Visão |
| IOT | Internet das coisas |
| PME | Pequenas e Medias Empresas |
| RFID | Radio Frequency Identification |
| RF | Requisitos funcionais |
| RNF | Requisitos não funcionais |
| NVR | Network vídeo recorder |
| SWOT | Forças, Fraquezas, Oportunidade e Ameaças |

Sumário

[1. APRESENTAÇÃO DO SOFTWARE 2](#_Toc191665622)

[2. APRESENTAÇÃO DO CLIENTE 3](#_Toc191665623)

[3. APRESENTAÇÃO DO PRODUTO 4](#_Toc191665624)

[3.1 Objetivos gerais 4](#_Toc191665625)

[3.2 Objetivos específicos 4](#_Toc191665626)

[3.3 Justificativa 5](#_Toc191665627)

[3.4 Tecnologias e APIs utilizadas 5](#_Toc191665628)

[3.4.1 Hardware e Sensores 6](#_Toc191665629)

[3.4.2 Software e Backend 7](#_Toc191665630)

[3.4.3 Protocolos e Automações 8](#_Toc191665631)

[4. ANÁLISE DE REQUISITOS 8](#_Toc191665632)

[4.1 Requisitos Funcionais 8](#_Toc191665633)

[4.2 Requisitos não funcionais 10](#_Toc191665634)

[5. ANÁLISE DE CONCORRENTES 13](#_Toc191665635)

[5.1 Hikvision 13](#_Toc191665636)

[5.1.1 Posicionamento no Mercado: 13](#_Toc191665637)

[5.1.2 Pontos Fortes (SWOT): 13](#_Toc191665638)

[5.1.3 Pontos Fracos (SWOT): 14](#_Toc191665639)

[5.2 Ring Alarm (Amazon) 16](#_Toc191665640)

[5.2.1 Posicionamento no Mercado: 16](#_Toc191665641)

[5.2.2 Pontos Fortes (SWOT): 16](#_Toc191665642)

[5.2.3 Pontos Fracos (SWOT): 16](#_Toc191665643)

[5.3 Dahua Technology 18](#_Toc191665644)

[5.3.1 Posicionamento no Mercado: 18](#_Toc191665645)

[5.3.2 Pontos Fortes (SWOT): 18](#_Toc191665646)

[5.3.3 Pontos Fracos (SWOT): 18](#_Toc191665647)

[5.4 Comparativo Estratégico (SALV vs. Concorrentes) 20](#_Toc191665648)

[5.5 Análise de custo de implementação 21](#_Toc191665649)

[6. CRONOGRAMAS DE TAREFAS 23](#_Toc191665650)

[7. PROTOTIPAÇÃO 24](#_Toc191665651)

[7.1 Prototipação do Aplicativo 24](#_Toc191665652)

[7.1.1 Paleta de Cores 24](#_Toc191665653)

[7.1 Prototipação do IOT 25](#_Toc191665654)

[8. PLANILHA DE CUSTOS DO PROJETO 26](#_Toc191665655)

[APÊNDICE A – MODELO MER 29](#_Toc191665656)

[APÊNDICE B – MODELO DER 31](#_Toc191665657)

[APÊNDICE C – DICIONÁRIO DE DADOS 32](#_Toc191665658)

[APÊNDICE D - SCRIPTS DE BANCO DE DADOS MYSQL 32](#_Toc191665659)

**INTRODUÇÃO**

A segurança em ambientes laboratoriais é um desafio constante, especialmente em contextos onde recursos financeiros são limitados e a necessidade de proteção de equipamentos, dados e materiais é crítica. Diante desses cenários, surge a proposta do SALV (Sistema de Alerta Laboratorial com Visão), um projeto acadêmico inovador que integra tecnologias de detecção de movimento e detecção facial para oferecer uma solução de segurança acessível e eficiente.

O sistema visa automatizar a vigilância de laboratórios por meio de um programa computacional que, ao identificar movimentos verifica a identidade dos usuários por meio de RFID (Radio Frequency Identification). Caso uma pessoa seja detectada pelos sensores, mas não tenha se identificado anteriormente, o SALV inicia imediatamente a gravação do ambiente, destacando as pessoas capturadas em tempo real e envia um alerta para um aplicativo móvel, permitindo respostas rápidas a possível violação. Essa abordagem não apenas reduz custos com sistemas de monitoramento tradicionais, mas também promove um ambiente mais organizado e seguro.

Além do impacto prático o projeto busca consolidar conhecimentos adquiridos em diversas disciplinas como programação, visão computacional, redes de computadores, IOT e diversas outras, aplicando a um problema real. Com viabilidade técnica sendo consideravelmente alta e prazo de execução relativamente curta, o SALV representa um passo significativo na integração entre teoria acadêmica e inovação, demonstrando como tecnologias emergentes podem ser adaptadas para resolver desafios cotidianos de forma criativa e acessível.

A proposta reforça, ainda a importância da interdisciplinaridade e da colaboração já que envolve inúmeras matérias. Assim o SALV não é apenas uma ferramenta de segurança, mas também um exemplo de como a tecnologia pode ser democratizada para beneficiar comunidades acadêmicas ou pessoas cujo necessitam de uma segurança de baixo custo.

# APRESENTAÇÃO DO SOFTWARE

O SALV (Sistema de Alerta Laboratorial com Visão) é uma solução inovadora de segurança projetada para monitoramento de ambientes laboratoriais e outros espaços restritos. A proposta do software surge da necessidade de oferecer uma alternativa acessível e eficaz para instituições acadêmicas ou ambientes com recursos financeiros limitados, que enfrentam desafios em implementar sistemas de segurança tradicional,

Combinando tecnologias de ponta, o SALV integra detecção de movimentos, reconhecimento facial e autenticação por RFID para garantir a proteção contínua monitoramento de acesso não autorizados. O sistema é acionado automaticamente quando detecta movimentos de um indivíduo cujo não foi previamente autenticado, inicia-se a gravação do ambiente, destacando a pessoa capturada em tempo real. Além disso, o sistema envia alertas imediatos por meio de um aplicativo móvel, permitindo que os responsáveis tomem ações rápidas em caso de violação.

Embora desenvolvido inicialmente para atender a demanda de ambientes acadêmicos, o sistema é altamente adaptável e pode ser utilizado em diversos contextos, como residência, escritórios e outros espaços que necessitam de controle de acesso e vigilância eficiente. A solução oferece uma implementação simples, de baixo custo sem compromete a qualidade e a segurança dos dados, tornando-a ideal para um público-alvo diversificando que busca soluções acessíveis de segurança.

A proposta do SALV não só representa um avanço em termos de segurança e monitoramento, mas também demostra como a tecnologia pode ser democratizada para beneficiar instituições e indivíduos que buscam alternativas de proteção mais acessíveis e de fácil implementação.

Em resumo, o SALV é uma solução escalável, eficiente e de baixo custo que propõe transformar a maneira como a segurança de ambientes restritos é gerenciada, utilizando tecnologias emergentes para melhorar a segurança de forma prática, acessível e inteligente.

# APRESENTAÇÃO DO CLIENTE

Apesar de o sistema ter sido projetado para uma instituição acadêmica que busca aprimorar a segurança de seus laboratórios ou salas por meio de soluções tecnológicas acessíveis, sua aplicação pode ser ampliada para diversos contextos. A instituição enfrenta desafios relacionados ao controle de acesso e segurança em geral, devido a restrições orçamentárias que dificultam a adoção de sistemas tradicionais.

Dessa forma, o SALV surge como uma resposta direta às necessidades do cliente, proporcionando monitoramento automatizado e uma estrutura adaptável, permitindo sua implementação em diferentes locais sem grandes modificações. Embora tenha sido idealizado para o ambiente acadêmico, o sistema pode ser utilizado em outros espaços, como residências, escritórios ou qualquer ambiente que necessite de um sistema de segurança acessível e eficiente.

Assim, nosso público-alvo não se restringe apenas a instituições, mas também a qualquer pessoa interessada em uma solução de segurança de baixo custo e fácil implementação.

# APRESENTAÇÃO DO PRODUTO

O SALV é uma solução inovadora e acessível para monitoramento e controle de acesso em ambientes diversos. Desenvolvido inicialmente para instituições acadêmicas, o sistema combina detecção facial, sensores de movimento e notificações em tempo real para garantir um ambiente mais seguro e protegido contra acessos não autorizados.

## 3.1 Objetivos gerais

O SALV tem como objetivo principal garantir a segurança aprimorada em ambientes laboratoriais e outros espaços que necessitam de proteção. Através da combinação de tecnologias acessíveis, como detecção de movimento e detecção facial, o sistema visa automatizar o processo de vigilância, proporcionando uma solução eficiente e de baixo custo para monitoramento contínuo. Além disso, o projeto busca integrar tecnologias emergentes de forma prática e acessível, utilizando visão computacional, sensores de movimento e notificações em tempo real para gerar alertas imediatos em caso de violação. Outro objetivo do SALV é consolidar os conhecimentos adquiridos nas disciplinas de programação, visão computacional, redes de computadores e Internet das Coisas (IoT) e diversas outras disciplinas, aplicando esses conceitos em um projeto inovador e real. O SALV visa, ainda, oferecer uma solução escalável e adaptável a diferentes contextos, podendo ser implementado em diversos ambientes além do acadêmico, como residências e escritórios. Por fim, o sistema propõe a redução dos custos com soluções tradicionais de segurança, democratizando o acesso a tecnologias de proteção e tornando-as mais acessíveis para instituições e indivíduos com recursos financeiros limitados.

## 3.2 Objetivos específicos

O Sistema de Alerta Laboratorial com Visão (SALV)possui objetivos específicos que visam detalhar suas funcionalidades e contribuições. Primeiramente, busca-se desenvolver um sistema capaz de detectar movimentações no ambiente, a partir disso, iniciar o processo de detecção facial para identificação dos indivíduos presentes. Para aprimorar a segurança, pretende-se integrar um mecanismo de autenticação por meio de cartões de acesso usando RFID, garantindo que apenas pessoas autorizadas possam permanecer no local, sem a ativação do sistema de segurança. Além disso, objetiva-se implementar uma funcionalidade de gravação automática em tempo real, que será ativada sempre que um acesso não autorizado for detectado, possibilitando o armazenamento de registros visuais para análise posterior dentro de um tempo predeterminado.

Outro objetivo é criar um sistema de notificações em tempo real, que enviará alertas diretamente para um aplicativo móvel, permitindo que responsáveis possam tomar medidas imediatas diante de possíveis violações.

## 3.3 Justificativa

A segurança em ambientes laboratoriais e outros espaços restritos é um desafio crescente, especialmente em instituições acadêmicas e locais com recursos financeiros limitados. A necessidade de proteger equipamentos, dados e materiais contra acessos não autorizados exige soluções tecnológicas eficientes e acessíveis. Nesse contexto, o **Sistema de Alerta Laboratorial com Visão (SALV)** surge como uma alternativa inovadora que visa integrar detecção de movimento, detecção facial e notificações automatizadas para aprimorar o controle de acesso.

Muitos sistemas tradicionais de segurança são caros e demandam infraestrutura complexa, o que dificulta sua implementação em instituições que enfrentam restrições orçamentárias. O SALV propõe uma solução de monitoramento automatizado que reduz custos, dispensando a necessidade de vigilância humana constante e tornando o ambiente mais seguro de forma autônoma. Além disso, a integração com um aplicativo móvel permite que alertas sejam recebidos em tempo real, garantindo respostas rápidas e eficientes a possíveis incidentes.

Do ponto de vista acadêmico, o desenvolvimento do SALV contribui significativamente para a aplicação prática de conceitos estudados ao longo do curso, como programação, redes de computadores, inteligência artificial e Internet das Coisas, etc. Dessa forma, além de representar um avanço em segurança, o projeto promove a interdisciplinaridade e possibilita a experimentação e aprimoramento de tecnologias emergentes. Por ser escalável e adaptável a diferentes cenários, o sistema também se apresenta como uma solução versátil, podendo ser implementado não apenas em laboratórios acadêmicos, mas também em escritórios, residências e outros espaços que necessitem de segurança reforçada. Assim, o SALV se justifica tanto pela sua relevância prática quanto pelo seu impacto acadêmico e social.

## 3.4 Tecnologias e APIs utilizadas

Para garantir a eficiência e acessibilidade do SALV, utilizamos um conjunto de tecnologias que combinam a visão computacional, IoT, computação em nuvem e comunicação em tempo real. Afim de melhor organização e explicação o sistema é principalmente divido em três principais componentes:

* Hardware e Sensores: Responsável pela captura de dados do ambiente, utilizando sensores de movimento, câmeras e afins.
* Software e Backend: Processa as informações coletadas, realiza a detecção facial, gerencia permissões de acesso e notifica os responsáveis sobre atividades suspeitas.
* Protocolos de automação: Tecnologias responsáveis por garantir a operação contínua do sistema, incluindo a inicialização automática e o acionamento remoto dos dispositivos.

A seguir, detalhamos cada um desses componentes e suas respectivas tecnologias.

**3.4.1 Hardware e Sensores**

O SALV foi projetado para ser uma solução de baixo custo, permitindo a utilização de dispositivos acessíveis e amplamente disponíveis. Os componentes de hardware utilizados no sistema incluem:

Tabela 1- Tecnologias do Hardware e Sensores

|  |  |
| --- | --- |
| Tecnologia | Descrição |
| ESP32 | Controladores responsáveis pela integração dos sensores e comunicação com o sistema central. |
| Raspberry Pi Camera | Capturam imagens para reconhecimento |
| Webcam | Uma alternativa mais acessível, se comparado ao Raspberry Pi Camera (podendo-se usar ambos ou apenas um) |
|  |  |
| Leitor RFID | Permite autenticação por meios de cartão de acesso |
| LCD | Usado para uma melhor visibilidade do sistema, ou seja, mostrar para o usuário que tudo está funcionando (podendo ser alterado por uma série de leds, caso seja preferido) |
| Comunicação via MQTT | Garante a troca rápida e eficiente de dados entre os dispositivos |

**Fonte**: os autores.

**3.4.2 Software e Backend**

A estrutura do software é projetada para operar de forma independente dos sensores, garantindo um processamento mais eficiente e seguro. As tecnologias utilizadas no backend são:

Tabela 2 - Tecnologias de Software e Backend

|  |  |
| --- | --- |
| Tecnologia | Descrição |
| OpenCV | Utilizando para detecção de movimentos e faces em tempo real |
| Python | Principal recurso de controle e analise de informação das imagens registradas |
| Supabase | Banco de dados e autenticação segura |
| React Native | Aplicativo móvel para monitoramento e recebimento de alertas |
| Firebase Cloud Messaging | Envio de notificações instantâneas para os responsáveis |
| Supabase Store | Registro de imagens e vídeos de acessos não autorizados. |

**Fonte**: os autores.

**3.4.3 Protocolos e Automações**

Para garantir que o sistema funcione de forma autônoma e contínua, foram adotados protocolos e ferramentas que permitem automação de processos, como a inicialização automática de serviços e o acionamento remoto de dispositivos.

Tabela 3- Protocolos e Automação

|  |  |
| --- | --- |
| Tecnologia | Descrição |
| Wake-on-Lan (WOL) | Permite que o ESP32 ligue o PC remotamente, caso esteja desligado |
| Task Scheduler | Automatiza a inicialização do programa Python ao ligar o PC |
| AutoStart (Python Script) | Configuração para iniciar automaticamente os serviços ao ligar a máquina. |

**Fonte**: os autores.

# ANÁLISE DE REQUISITOS

A análise de requisitos é dos pontos fundamentais de qualquer projeto, e com o SALV não seria diferente, nessa questão a utilização de tal meio se torna necessário afim de garantir que a implementação atenda os objetivos propostos da melhor forma o possível. Esse processo envolveu a definição de requisitos funcionais e não funcionais, considerando fatores como segurança, eficiência e acessibilidade.

## 4.1 Requisitos Funcionais

Tabela 4- Requisitos Funcionais

|  |  |
| --- | --- |
| **Requisitos Funcionais** | **Descrição dos requisitos** |
| **RF01** | Detecção de movimentos: O sistema deve detectar movimento em tempo real dentro do ambiente. Identificando movimentos deve iniciar o SALV. |
| **RF02** | Detecção Facial: O sistema deve capturar imagens e do ambiente tentando destacar o rosto do indivíduo cujo não foi autentificado anteriormente. |
| **RF03** | Autenticação via Cartão: O sistema deve permitir que usuários autentificados utilizam cartão RFID para identificação, nesses casos o sistema de segurança não deve ser ativado. |
| **RF04** | Gravação Automática: Caso o sistema detecte um usuário não autorizado, deve iniciar a gravação do ambiente. As imagens e vídeos capturados devem ser armazenados em um local seguro a fim de pessoas autorizadas verifiquem o que aconteceu. |
| **RF05** | Notificação em Tempo Real: O sistema deve enviar alertas para um aplicativo móvel sempre que um acesso não autorizado for detectado. O alerta deve conter informações como horário, caso o usuário tenha mais de uma sala, deve se conter qual sala esta sendo violada. |
| **RF06** | Controle Remoto: Permitir que administradores possam visualizar e administrar os acessos por meio de aplicativo. Possibilidade de desligar ou reconfigurar remotamente o sistema de monitoramento |
| **RF07** | Registo e Monitoramento de acesso: O sistema deve armazenar um histórico de uso de cartões RFDI incluído:   * Nome * Data e hora |
| **RF08** | Integração com Banco de dados: O sistema deve utilizar Supabase para armazenar dados dos acessos e dos usuários cadastrados |

**Fonte**: os autores.

## 4.2 Requisitos não funcionais

Tabela 5- Requisitos não funcionais

|  |  |
| --- | --- |
| **Requisitos não funcionais** | **Descrição dos requisitos** |
| **RNF01** | Segurança e Proteção de Dados:   * Os dados de usuários e imagens capturadas devem ser armazenados de forma segura e criptografada * Garantir que somente pessoas autorizadas tenham acesso as informações sensíveis * Apagar as filmagens que ultrapassa um tempo predeterminado |
| **RNF02** | Acessibilidade: O aplicativo móvel deve seguir as diretrizes de Acessibilidade Digital:   * Interface com alto contraste para usuários com baixa visão * Compatibilidade com leitores de tela * Botões grandes e interface intuitiva para facilitar o uso por pessoas com dificuldade motoras * O sistema deve oferecer um método alternativo de autenticação, como biometria, senhas e etc. |
| **RNF03** | Desempenho e Eficiência:   * O sistema deve processar a detecção facial em tempo real (tempo de resposta menor que 3 segundos). * A comunicação entre o ESP32, banco de dados e aplicativo deve ter latência mínima para envio de alertas rápidos |
| **RNF04** | Confiabilidades e Disponibilidade:   * O sistema deve continuar funcionando mesmo em caso de falha de algum componente. * O Wake-on-Lan deve garantir que o sistema seja iniciado novamente caso haja uma possível queda de energia |
| **RNF05** | Escalabilidade:   * O sistema deve permitir a adição de novos usuários e locais sem comprometer o desempenho |
| **RNF06** | Compatibilidade:   * O sistema deve ser compatível com Android e IOS. * O backend deve suportar a execução em servidões garantindo fácil manutenção |
| **RNF07** | Manutenibilidade:   * O código-fonte modular e bem documentado para facilitar futuras melhorias * Uso de padrões de desenvolvimento para garantir qualidade do software; |

**Fonte**: os autores.

# ANÁLISE DE CONCORRENTES

Para garantir que o SALV seja uma solução competitiva é necessário compreender o cenário atual dos sistemas de segurança que estão disponíveis no mercado. Nesta análise, identificamos concorrentes diretos e indiretos, avaliando suas principais soluções e funcionalidades. Pensando dessa forma podemos levar em consideração três pontos principais para realizamos a nossa análise, Posicionamento do mercado, Pontos fortes e fracos, que serão usadas como base em nossa analise.

## 5.1 Hikvision

A Hikvison é uma empresa chinesa fundada em 2001, especializada em soluções de vídeo monitoramento e segurança eletrônica. Reconhecida mundialmente, a empresa oferece uma série de serviços voltado para a área, apesar dessas questões podemos destacar alguns pontos com relação a isso.

**5.1.1 Posicionamento no Mercado:**

* Líder global em vigilância eletrônica, focado em grandes corporações e governos;
* Preço premium: sistemas completos custam a partir de R$ 20.000 (exemplo: pacote com 4 câmeras + NVR);
* Público-alvo: Empresas com alto orçamento para segurança física.

**5.1.2 Pontos Fortes (SWOT):**

* Tecnologia Avançada com alta precisão de reconhecimento facial;
* Integração com sistemas de segurança profissionais;
* Suporte para múltiplas câmeras e dispositivos IoT;
* Presença global em mais de 150 países.

**5.1.3 Pontos Fracos (SWOT):**

* Alto custo de aquisição e manutenção;
* Necessidade de infraestrutura robusta para funcionamento adequado;
* Preocupação com privacidade e conformidade com a LGPD.

Figura 1 - Home HIKVISION

Interface gráfica do usuário, Site

O conteúdo gerado por IA pode estar incorreto.

**Fonte**: HIKVISION (2025).

Figura 2 - Produtos HIKVISION

Uma imagem contendo screenshot, foto, olhando, pia

O conteúdo gerado por IA pode estar incorreto.

**Fonte**: HIKVISION (2025).

Figura 3 -Soluções HIKVISION

Interface gráfica do usuário, Aplicativo

O conteúdo gerado por IA pode estar incorreto.

**Fonte**: HIKVISION (2025).

## 5.2 Ring Alarm (Amazon)

A Ring foi fundada em 2013 nos Estados Unidos e posteriormente adquirida pela Amazon. A empresa é conhecida por seus sistemas de segurança residencial e integração com assistentes de voz.

**5.2.1 Posicionamento no Mercado:**

* Foco em segurança residencial e usuários finais;
* Modelo Premium: dispositivos acessíveis (ex.: kit básico por R1.500).

**5.2.2 Pontos Fortes (SWOT):**

* Instalação fácil e configuração intuitiva;
* Integração com Alexa e outros dispositivos inteligentes;
* Notificações em tempo real pelo aplicativo.

**5.2.3 Pontos Fracos (SWOT):**

* Algumas funcionalidades são limitadas sem a assinatura premium;
* Dependência de conexão com a nuvem;
* Foco principal em segurança residencial, podendo não ser adequado para outros cenários.

Figura 4 - Home Ring Alarm

Interface gráfica do usuário, Aplicativo

O conteúdo gerado por IA pode estar incorreto. **Fonte**: Ring Alarm (2025).

Figura 5 - Produtos Ring Alarm

Interface gráfica do usuário, Aplicativo

O conteúdo gerado por IA pode estar incorreto.

**Fonte**: Ring Alarm (2025).

Figura 6 - Planos Ring Alarm

Interface gráfica do usuário, Aplicativo

O conteúdo gerado por IA pode estar incorreto.

**Fonte**: Ring Alarm (2025).

## Dahua Technology

A Dahua Technology é uma empresa chinesa fundada em 2001, especializada em soluções de monitoramento e controle de acesso. Seus produtos incluem sistemas de videomonitoramento com reconhecimento facial.

* + 1. **Posicionamento no Mercado:**
* Concorrente direto da Hikvision, com foco em câmeras de alta definição e análise por IA;
* Preço 20% menor que Hikvision, mas ainda elevado para PMEs.
  + 1. **Pontos Fortes (SWOT):**
* Soluções personalizáveis para diferentes segmentos;
* Integração com sensores e dispositivos de autenticação;
* Suporte a IA para análise de imagens.

**5.3.3 Pontos Fracos (SWOT):**

* Complexidade na configuração e implementação;
* Alto consumo de armazenamento e necessidade de servidores robustos;
* Histórico de vulnerabilidades em segurança cibernética.

Figura 7 - Home Dahua Technology

Interface gráfica do usuário, Site

O conteúdo gerado por IA pode estar incorreto.

**Fonte**: Dahua Technology (2025).

Figura 8 - Soluções Dahua Technology

Interface gráfica do usuário, Site

O conteúdo gerado por IA pode estar incorreto.

**Fonte**: Dahua Technology (2025).

Figura 9 - Produtos Dahua Technology

Interface gráfica do usuário, Site

O conteúdo gerado por IA pode estar incorreto.

**Fonte**: Dahua Technology (2025).

## Comparativo Estratégico (SALV vs. Concorrentes)

Com base nessa analise cujo realizamos dos nossos principais concorrentes que já estão fortemente alocados no mercado podemos montar uma tabela com os principais pontos cujo podemos vim a considerar na hora da escolha do produto.

Tabela 6- Comparativo SALV vs. Concorrentes

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Critério | Hikvision | Ring Alerm | Duhua | SALV |
| **Custo inicial** | Alto (R$ ≈ 20k) | Médio (R$ ≈ 1.5k) | Alto (R$ ≈ 16k) | Baixo (R$ ≈ 130) |
| **Tecnologia** | Reconhecimento facial | IoT e integração | IA comportamental | Reconhecimento + IoT |
| **Facilidade de Uso** | Complexa | Muito simples | Complexa | Simples (UI intuitiva) |
| **Conformidade com a LGPD** | Questionável | Parcial | Vulnerável | 100% compliance |
| **Público-Alvo** | Grandes empresas | Residências | Corporações | PMEs e Residências |

**Fonte**: os autores.

## Análise de custo de implementação

É necessário explicar o motivo do custo de implementação ser tão distinto entre o SALV e os concorrentes. Como mencionado anteriormente, o projeto foi concebido com foco na implementação de baixo custo. O objetivo era oferecer uma solução acessível, sem comprometer a qualidade e a segurança. Para isso, decidimos economizar ao máximo, utilizando componentes de IoT que se tornaram amplamente acessíveis ao longo dos anos.

Esses componentes estão disponíveis em lojas tradicionais (como no bairro da Santa Efigênia, em São Paulo) e no ambiente digital, através de diversos fornecedores online, o que proporcionou uma escolha mais vantajosa e econômica para o desenvolvimento do projeto. A seguir, apresentamos a listagem dos principais componentes utilizados, com seus respectivos preços:

Tabela 7- Valores dos componentes

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Componente | Descrição | Preço (R$) | Vendedor | Plataforma |
| ESP32 | Microcontrolador para integração de sensores | R$: 44,99 | ARDU ROBÓTICA E INFORMÁTICA LTDA | Shoppe |
| Display LCD 16X2 | Tela LCD de 16 colunas por 2 linhas, usada para exibição de informações de status do sistema. | R$: 26,36 | ARDU ROBÓTICA E INFORMÁTICA LTDA | Shoppe |
| Protoboard | Placa de ensaio para montagem de circuitos de maneira rápida e sem necessidade de solda. | R$: 13,20 | ARDU ROBÓTICA E INFORMÁTICA LTDA | Shoppe |
| Modulo sensor de movimento e presença | Sensor utilizado para detectar movimento e presença de pessoas no ambiente, essencial para a ativação do sistema de monitoramento. | R$: 12,97 | ARDU ROBÓTICA E INFORMÁTICA LTDA | Shoppe |
| Jumpers Macho X Macho | Fios de conexão com terminais machos nas duas extremidades, utilizados para realizar ligações entre os componentes eletrônicos. | R$: 11,31 | ARDU ROBÓTICA E INFORMÁTICA LTDA | Shoppe |
| Jumpers Macho ☓ fêmea | Fios de conexão com terminais macho de um lado e fêmea do outro, usados para conectar o ESP32 e outros sensores. | R$: 11,31 | ARDU ROBÓTICA E INFORMÁTICA LTDA | Shoppe |
| Leitor de RFDI | Leitor de cartão RFID utilizado para autenticação de acesso, permitindo identificar e controlar quem está autorizado no local. | R$: 15,00 | Curto Circuito Comp. Eletrônicos | Shoppe |

**Fonte**: os autores.

# CRONOGRAMAS DE TAREFAS

O cronograma de um projeto é uma ferramenta fundamental para o planejamento e a execução eficaz de qualquer iniciativa, incluindo o desenvolvimento do SALV. Ele serve como um guia que orienta a equipe ao longo de todas as fases do projeto, garantindo que as atividades sejam realizadas dentro dos prazos estabelecidos e que os objetivos sejam alcançados de maneira organizada e eficiente.

Para facilitar esse processo, utilizamos o Trello como base para o nosso cronograma, que nos ajudou a estruturar o desenvolvimento. Consideramos tanto as tarefas que tínhamos total controle, como a implementação de funcionalidades especificas, quanto questões mais complexas que poderiam lavar a variação de tempo, como a entrega de componentes essenciais para a continuidade do projeto. Todas essas variáveis foram cuidadosamente consideradas ao montamos o cronograma.

Por fim, apresentamos a montagem desse cronograma, que reflete nosso planejamento e as etapas necessárias para o sucesso do SALV, vale ressaltar que nesse ponto nos encontrávamos no início do projeto.

Figura 10 - Cronograma base do projeto

Interface gráfica do usuário, Aplicativo

O conteúdo gerado por IA pode estar incorreto.

**Fonte**: os autores.

# PROTOTIPAÇÃO

A prototipação é uma das etapas cruciais no desenvolvimento de qualquer projeto, pois permite a visualização e a validação de ideias antes da implementação final. No contexto do SALV, a prototipação foi aplicada tanto no desenvolvimento do aplicativo quanto na montam do sistema IoT, garantindo que a solução proposta atendesse as necessidades dos usuários e funcionassem de maneira integrada.

## 7.1 Prototipação do Aplicativo

A prototipação do aplicativo móvel foi realizada com o objetivo de criar uma interface intuitiva e funcional, que permitisse aos usuários interagir facialmente com o sistema de segurança. Utilizamos ferramentas de design de interfaces, como o Figma para criar Wire frames etc.

Imagem

Imagem

Imagem

Durante essa fase, realizamos testes de usabilidade e acessibilidade afim do melhor resultado possível. Essa abordagem iterativa nos permitiu ajustar a interface, garantindo que as funcionalidades, como o recebimento de alertas em tempo real e o controle remoto fossem acessíveis e compreensível.

**7.1.1 Paleta de Cores**

A escolha da paleta de cores é um elemento essencial no design do aplicativo, influenciando tanto a usabilidade quanto a identidade visual. Cores bem selecionadas podem melhorar a experiência do usuário, facilitando a leitura, aumentando a compreensão das informações e reforçando a identidade do sistema SALV.

Utilizamos uma paleta que equilibra contraste e harmonia, garantindo acessibilidade para todos os usuários, incluindo aqueles com deficiências visuais. O uso adequado de cores também auxilia na diferenciação de elementos importantes, como alertas e comandos principais, proporcionando uma experiência de interação intuitiva e eficiente.

## 7.1 Prototipação do IOT

A prototipação do sistema IoT envolveu a montagem física dos componentes que compões o SALV. Utilizamos um conjunto de diversificando de componentes para a criação desse protótipo. Embora não houvesse uma plataforma que oferecesse todos os elementos necessários para a montagem, conseguimos desenvolver um conceito básico utilizando o Wokwi. Essa ferramenta foi extremamente útil para determinar as portas de dados e de alimentação que seriam utilizadas no projeto.

É importante ressaltar que esta é apenas uma fase de prototipação. Em etapas futuras, abordaremos detalhadamente as conexões o código do ESP32, garantindo que todos os aspectos do sistema sejam bem compreendidos e implementados

Figura 11 – Protótipo do IoT

Diagrama, Esquemático

O conteúdo gerado por IA pode estar incorreto.

**Fonte**: os autores.

# PLANILHA DE CUSTOS DO PROJETO

Embora já tenhamos mencionado de forma geral os custos do projeto anteriormente, é essencial detalharmos com mais precisão os valores associados a cada componente. Além disso é importante considerar outros custos adicionais que podem surgir durante o desenvolvimento, como o espaço em nuvem, licenciamento de tecnologias especificas e outros serviço. Neste documento, vamos nos aprofundar em cada uma dessas despesas, fornecendo uma visão completa e transparente dos gastos necessários para a execução do projeto.

Figura 12 - Gráfico de custos do projeto

**Fonte**: os autores.

**CONSIDERAÇÕES FINAIS**

**REFERÊNCIAS**

# APÊNDICE A – MODELO MER

Uma parte importante em relação à estruturação do banco de dados são as questões dos modelos de entidade, tanto o Modelo Entidade-Relacionamento (MER) quanto o Modelo de Entidade Relacional (DER). No contexto deste projeto, essas questões não   
seriam diferentes. Para garantir uma compreensão abrangente da estrutura do banco de   
dados e abranger todos os leitores desta documentação, torna-se necessário incluir uma   
descrição do que seria o modelo MER.

O Modelo Entidade-Relacionamento (MER) é uma representação visual dos elementos de dados e de como eles se relacionam entre si em um sistema de banco de dados. Ele consiste em entidades, que são objetos ou conceitos do mundo real, atributos, que são as propriedades das entidades, e relacionamentos, que são as associações entre as entidades.

No modelo MER, as entidades são representadas por retângulos, os atributos    
listados dentro desses retângulos e os relacionamentos são mostrados por linhas que   
conectam as entidades, indicando como elas estão relacionadas umas com as outras.

Após essa breve explicação, apresentamos o nosso modelo:

Figura 13 - Modelo Mer

Diagrama

O conteúdo gerado por IA pode estar incorreto.

**Fonte**: os autores.

# APÊNDICE B – MODELO DER

O Modelo de Entidade Relacional (DER) é uma representação lógica e estruturada dos dados em um sistema de banco de dados, focando na forma como as entidades estão relacionadas entre si e como esses relacionamentos são armazenados. Enquanto o MER se concentra nas entidades, atributos e relacionamentos, o DER vai além, definindo tabelas específicas e os vínculos entre elas.

No modelo DER, as entidades são mapeadas para tabelas no banco de dados, onde cada entidade corresponde a uma tabela e cada atributo se torna um campo nessa tabela. Os relacionamentos entre as entidades são representados pelos vínculos entre as tabelas, geralmente por meio de chaves estrangeiras.

O DER também aborda aspectos como cardinalidade (ou seja, quantos registros de uma entidade estão relacionados a quantos registros de outra entidade), restrições de integridade referencial e outras regras de negócio que garantem a consistência e a integridade dos dados.

Figura 14 - Modelo Der

Diagrama

O conteúdo gerado por IA pode estar incorreto.

**Fonte**: os autores.

# APÊNDICE C – DICIONÁRIO DE DADOS

O dicionário de dados é uma ferramenta essencial na gestão e desenvolvimento de sistemas de informação. Ele funciona como um catálogo ou glossário que contém informações detalhadas sobre os dados armazenados em um sistema, incluindo definições, formatos, tipos, relações e restrições.

Este recurso serve como uma fonte de referência centralizada para todas as partes interessadas no sistema, incluindo desenvolvedores, analistas de dados, administradores de banco de dados e usuários finais. Ele desempenha um papel fundamental na garantia da consistência e na compreensão compartilhada dos dados, facilitando a comunicação e colaboração entre as equipes envolvidas no desenvolvimento e manutenção do sistema.

Além disso, o dicionário de dados é uma ferramenta valiosa para o controle de qualidade dos dados, ajudando a identificar e corrigir inconsistências, erros e ambiguidades nos dados. Ele também pode ser usado como parte do processo de documentação do sistema, fornecendo informações importantes para a compreensão da estrutura e funcionamento do sistema de informação.

Tabela 8 - Dicionário de dados Pessoa

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Nome da tabela | | | Tb\_Pessoas | | |
| Nome do Campo | Tipo de Dado | Tamanho Máximo | Descrição dos Campos | Tipo de Chave | Observações |
| Id\_Pessoas | INT | - | Id para identificar a pessoa | Pk |  |
| Email | VARCHAR | 45 | Email usado no cadastro |  |  |
| Nome | VARCHAR | 45 | Nome do usuário |  |  |
| Celular | VARCHAR | 45 | Identificação do celular do usuário, cujo vai chegar a notificação |  |  |
| Token | TEXT | - | Token do google caso a pessoa queira fazer login dessa forma |  | Esse campo só é usado para login com o Google |

**Fonte**: Autores (2024).

# APÊNDICE D - SCRIPTS DE BANCO DE DADOS MYSQL

Os scripts de geração de tabelas e procedures do banco de dados MySQL desempenham um papel crucial na arquitetura e na funcionalidade de qualquer projeto que dependa de armazenamento de dados. Ao criar e manter um banco de dados MySQL, os scripts não apenas definem a estrutura das tabelas que armazenam informações essenciais, mas também estabelecem os procedimentos que manipulam esses dados de maneira consistente e eficiente.

A principal importância desses scripts reside na sua capacidade de garantir a consistência e a integridade dos dados. Ao definir cuidadosamente os tipos de dados, as restrições de chave primária e estrangeira, e outras propriedades das tabelas, os scripts ajudam a evitar inconsistências e erros de dados que podem surgir durante o ciclo de vida do projeto.

Além disso, os procedimentos armazenados definidos nos scripts fornecem uma maneira padronizada de acessar, manipular e atualizar os dados do banco de dados.

Com o objetivo de evitar a contaminação da documentação por conteúdo externo, optamos por demonstrar o script de criação de uma maneira que está anexada externamente. No entanto, entendemos que alguns leitores podem preferir uma abordagem mais direta. Por esse motivo, além do link externo para o script completo, também fornecemos um exemplo do script na forma de imagem, para facilitar o acesso rápido e a visualização para aqueles que preferem não acessar recursos externos.

*-- -----------------------------------------------------*

*-- Schema SALV*

*-- -----------------------------------------------------*

DROP SCHEMA IF EXISTS `SALV` ;

*-- -----------------------------------------------------*

*-- Schema SALV*

*-- -----------------------------------------------------*

CREATE SCHEMA IF NOT EXISTS `SALV` DEFAULT CHARACTER SET utf8 ;

SHOW WARNINGS;

USE `SALV` ;

*-- -----------------------------------------------------*

*-- Table `SALV`.`Tb\_Usuários`*

*-- -----------------------------------------------------*

DROP TABLE IF EXISTS `SALV`.`Tb\_Usuários` ;

SHOW WARNINGS;

CREATE TABLE IF NOT EXISTS `SALV`.`Tb\_Usuários` (

  `ID\_Usuários` INT NOT NULL,

  `Nome` VARCHAR(55) NULL,

  `Email` VARCHAR(55) NULL,

  `Celular` VARCHAR(55) NULL,

  PRIMARY KEY (`ID\_Usuários`))

ENGINE = InnoDB;

SHOW WARNINGS;

*-- -----------------------------------------------------*

*-- Table `SALV`.`Tb\_Dispositivo`*

*-- -----------------------------------------------------*

DROP TABLE IF EXISTS `SALV`.`Tb\_Dispositivo` ;

SHOW WARNINGS;

CREATE TABLE IF NOT EXISTS `SALV`.`Tb\_Dispositivo` (

  `ID\_Dispositivo` INT NOT NULL,

  `Localização` VARCHAR(60) NULL,

  `Status` TINYINT NULL,

  `Nome` VARCHAR(55) NULL,

  PRIMARY KEY (`ID\_Dispositivo`))

ENGINE = InnoDB;

SHOW WARNINGS;

*-- -----------------------------------------------------*

*-- Table `SALV`.`Tb\_Evento`*

*-- -----------------------------------------------------*

DROP TABLE IF EXISTS `SALV`.`Tb\_Evento` ;

SHOW WARNINGS;

CREATE TABLE IF NOT EXISTS `SALV`.`Tb\_Evento` (

  `ID\_Evento` INT NOT NULL,

  `DataHora` DATETIME NULL,

  `Local` VARCHAR(55) NULL,

  `Tipo\_Evento` TINYINT NULL,

  `Tb\_Dispositivo\_ID\_Dispositivo` INT NOT NULL,

  PRIMARY KEY (`ID\_Evento`, `Tb\_Dispositivo\_ID\_Dispositivo`),

  INDEX `fk\_Tb\_Evento\_Tb\_Dispositivo1\_idx` (`Tb\_Dispositivo\_ID\_Dispositivo` ASC) VISIBLE,

  CONSTRAINT `fk\_Tb\_Evento\_Tb\_Dispositivo1`

    FOREIGN KEY (`Tb\_Dispositivo\_ID\_Dispositivo`)

    REFERENCES `SALV`.`Tb\_Dispositivo` (`ID\_Dispositivo`)

    ON DELETE NO ACTION

    ON UPDATE NO ACTION)

ENGINE = InnoDB;

SHOW WARNINGS;

*-- -----------------------------------------------------*

*-- Table `SALV`.`Tb\_Notificacao`*

*-- -----------------------------------------------------*

DROP TABLE IF EXISTS `SALV`.`Tb\_Notificacao` ;

SHOW WARNINGS;

CREATE TABLE IF NOT EXISTS `SALV`.`Tb\_Notificacao` (

  `ID\_Notificacao` INT NOT NULL,

  `Mensagem` VARCHAR(45) NULL,

  `Destino` VARCHAR(55) NULL,

  `DataHoraEnvio` DATETIME NULL,

  `Tb\_Evento\_ID\_Evento` INT NOT NULL,

  `Tb\_Evento\_Tb\_Dispositivo\_ID\_Dispositivo` INT NOT NULL,

  PRIMARY KEY (`ID\_Notificacao`, `Tb\_Evento\_ID\_Evento`, `Tb\_Evento\_Tb\_Dispositivo\_ID\_Dispositivo`),

  INDEX `fk\_Tb\_Notificacao\_Tb\_Evento1\_idx` (`Tb\_Evento\_ID\_Evento` ASC, `Tb\_Evento\_Tb\_Dispositivo\_ID\_Dispositivo` ASC) VISIBLE,

  CONSTRAINT `fk\_Tb\_Notificacao\_Tb\_Evento1`

    FOREIGN KEY (`Tb\_Evento\_ID\_Evento` , `Tb\_Evento\_Tb\_Dispositivo\_ID\_Dispositivo`)

    REFERENCES `SALV`.`Tb\_Evento` (`ID\_Evento` , `Tb\_Dispositivo\_ID\_Dispositivo`)

    ON DELETE NO ACTION

    ON UPDATE NO ACTION)

ENGINE = InnoDB;

SHOW WARNINGS;

*-- -----------------------------------------------------*

*-- Table `SALV`.`Tb\_Registro\_de\_Acesso`*

*-- -----------------------------------------------------*

DROP TABLE IF EXISTS `SALV`.`Tb\_Registro\_de\_Acesso` ;

SHOW WARNINGS;

CREATE TABLE IF NOT EXISTS `SALV`.`Tb\_Registro\_de\_Acesso` (

  `ID\_Registro\_de\_Acesso` INT NOT NULL,

  `Data e Hora` DATETIME NULL,

  `Statis\_de\_Acesso` TINYINT NULL,

  `Filmagem` BLOB NULL,

  `Tb\_Usuários\_ID\_Usuários` INT NOT NULL,

  `Tb\_Dispositivo\_ID\_Dispositivo` INT NOT NULL,

  PRIMARY KEY (`ID\_Registro\_de\_Acesso`, `Tb\_Usuários\_ID\_Usuários`, `Tb\_Dispositivo\_ID\_Dispositivo`),

  INDEX `fk\_Tb\_Registro\_de\_Acesso\_Tb\_Usuários\_idx` (`Tb\_Usuários\_ID\_Usuários` ASC) VISIBLE,

  INDEX `fk\_Tb\_Registro\_de\_Acesso\_Tb\_Dispositivo1\_idx` (`Tb\_Dispositivo\_ID\_Dispositivo` ASC) VISIBLE,

  CONSTRAINT `fk\_Tb\_Registro\_de\_Acesso\_Tb\_Usuários`

    FOREIGN KEY (`Tb\_Usuários\_ID\_Usuários`)

    REFERENCES `SALV`.`Tb\_Usuários` (`ID\_Usuários`)

    ON DELETE NO ACTION

    ON UPDATE NO ACTION,

  CONSTRAINT `fk\_Tb\_Registro\_de\_Acesso\_Tb\_Dispositivo1`

    FOREIGN KEY (`Tb\_Dispositivo\_ID\_Dispositivo`)

    REFERENCES `SALV`.`Tb\_Dispositivo` (`ID\_Dispositivo`)

    ON DELETE NO ACTION

    ON UPDATE NO ACTION)

ENGINE = InnoDB;

SHOW WARNINGS;

SET SQL\_MODE=@OLD\_SQL\_MODE;

SET FOREIGN\_KEY\_CHECKS=@OLD\_FOREIGN\_KEY\_CHECKS;

SET UNIQUE\_CHECKS=@OLD\_UNIQUE\_CHECKS;