

Projeto Sentinel Eyes

Aluno: Fabricio Pianovski f.pianovski@gmail.com
Felipe Camargo de Pauli fpauli@alunos.utfpr.edu.br
Rafael Campanhola rafaelcampanhola@alunos.utfpr.edu.br

Prof: Prof. Dr. César Benitez

Conteúdo

1	Introdução	2
1.1	Contexto e motivação	2
2	Descrição do Projeto	3
3	Escopo	4
3.1	Fora do Escopo	5
4	Requisitos do Projeto	5
4.1	Requisitos Funcionais do Software	5
4.2	Requisitos Não Funcionais do Software	5
4.3	Requisitos Funcionais do Hardware	6
5	Objetivos	6
6	Métodos	7
6.1	Software	10
6.2	Hardware	10
6.3	Ferramentas	11
7	Gantt	11
8	Riscos	12
9	Referências Bibliográficas	12

1 Introdução

Este trabalho tem como objetivo desenvolver o projeto Sentinel Eyes, **um sistema de identificação de pessoas por meio de escaneamento facial utilizando tecnologias de aprendizado de máquina e visão computacional**. A ideia é integrar componentes mecânicos, de hardware e software para criar um drone que verifica se pessoas são conhecidas ou não por meio da análise de parâmetros faciais. O sistema será capaz de processar imagens e vídeos capturados por câmeras de vigilância, reconhecer características de cada indivíduo e trazer dados sobre a pessoa em uma tela de monitoramento em tempo real.

O nome do projeto é uma referência às sentinelas do filme Matrix[1], que monitoravam áreas da Terra em busca de seres humanos não autorizados. Eles escaneavam os humanos para identificá-los e verificar se tinham autorização para estar onde estavam. O projeto Sentinel Eyes tem objetivo similar, que é o de dar a equipamentos voadores não tripulados capacidade de identificar pessoas por reconhecimento facial.

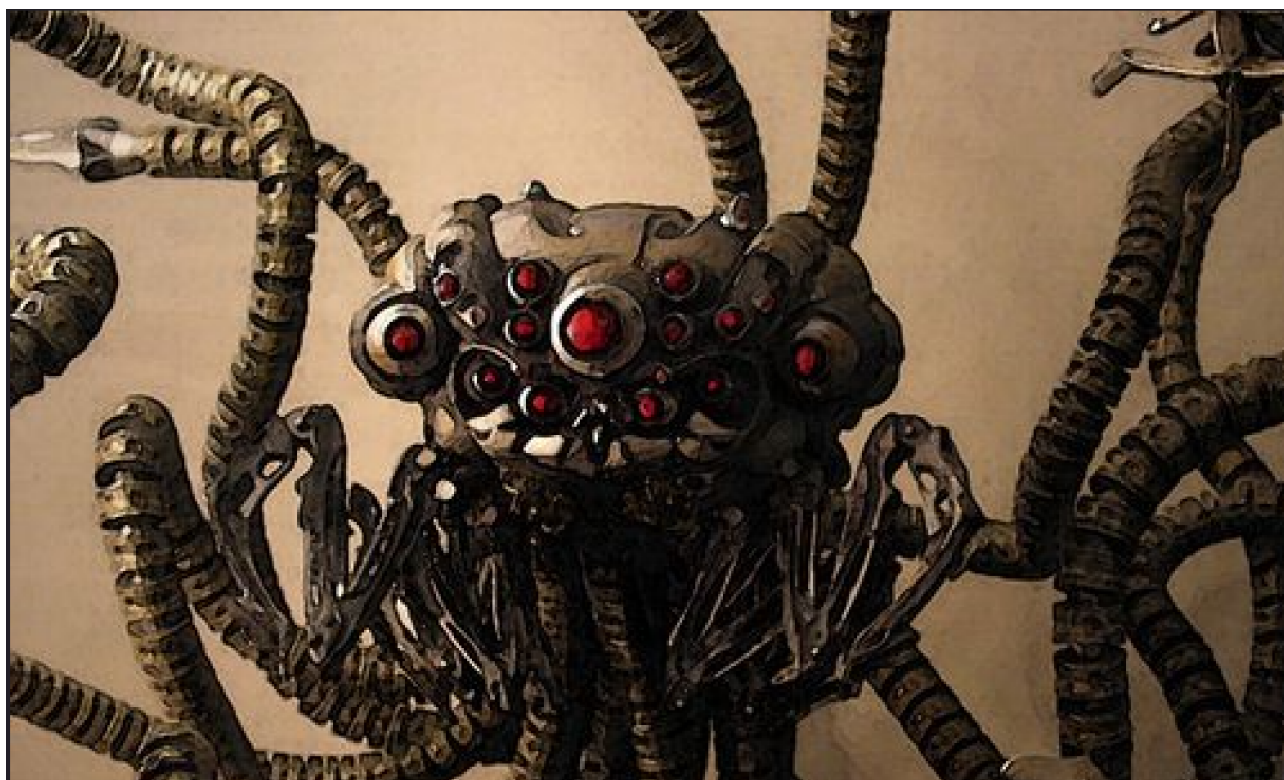


Figura 1: Sentinel, referência à Sentinel de Matrix

1.1 Contexto e motivação

Nos últimos anos, a segurança pública tem se tornado uma preocupação crescente em todo o mundo. Com o aumento da criminalidade e a necessidade de garantir a segurança das pessoas em espaços públicos e privados, a demanda por soluções tecnológicas eficientes tem crescido consideravelmente. Nesse cenário, a identificação e o monitoramento de indivíduos têm se mostrado ferramentas importantes no combate ao crime e na prevenção de incidentes.

A adoção de tecnologias de aprendizado de máquina e visão computacional tem re-

volucionado o setor de segurança, possibilitando o desenvolvimento de sistemas avançados de vigilância. Entre essas tecnologias, o reconhecimento facial se destaca como uma das mais promissoras, pois permite identificar pessoas de forma rápida e precisa, mesmo em ambientes lotados ou com condições adversas de iluminação.[2]

O projeto Sentinel Eyes busca contribuir para essa área, integrando tecnologias de ponta em um drone capaz de identificar e rastrear pessoas através de suas características faciais. Essa inovação tem o potencial de melhorar a eficácia das operações de segurança, auxiliando as autoridades na identificação de suspeitos, na busca por pessoas desaparecidas e na prevenção de atividades criminosas.

Além disso, o sistema pode ser aplicado em uma variedade de contextos, como controle de acesso em eventos, monitoramento de áreas restritas e gerenciamento de fluxo de pessoas em locais públicos. A motivação deste trabalho, portanto, é desenvolver um sistema robusto e eficiente que possa ser utilizado para melhorar a segurança e o bem-estar das pessoas, ao mesmo tempo em que respeita os princípios éticos e as preocupações relacionadas à privacidade[3].

2 Descrição do Projeto

Um drone, que será controlado por um controle remoto (ou aplicativo de celular ou um programa de computador), irá voar e parar em frente ao rosto de uma pessoa(3). Ele deve permanecer estático. Um Raspberry(1) controlará os atuadores do drone que o fará voar e parar em uma posição específica.

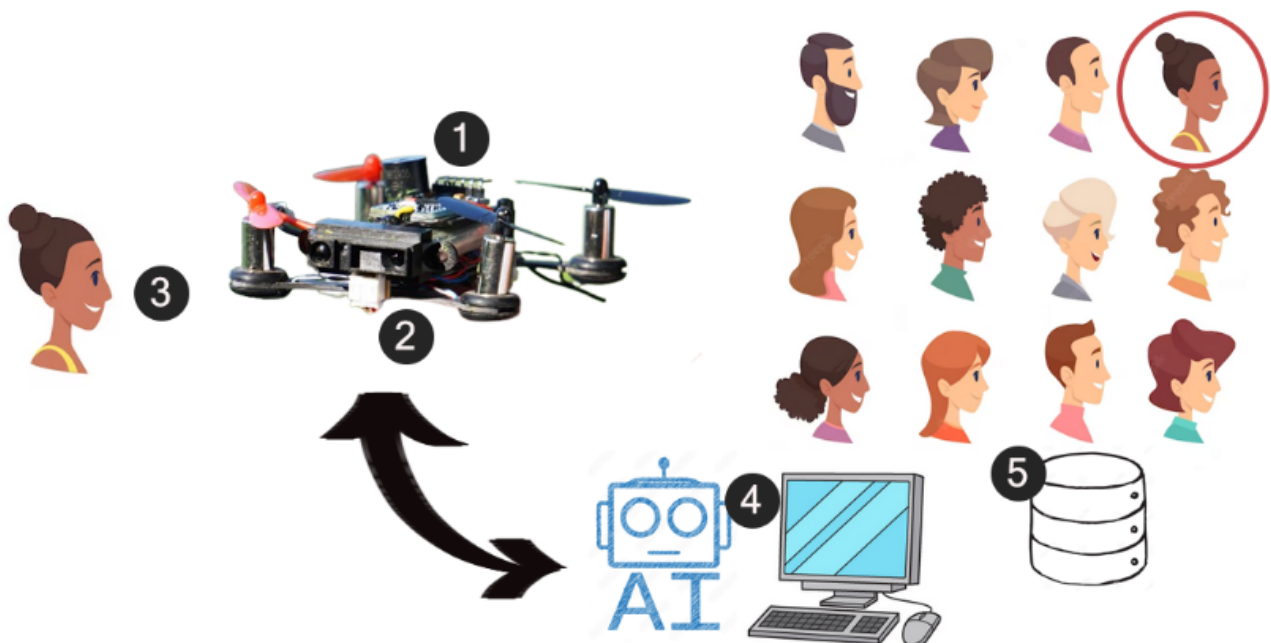


Figura 2: Desenho do Sentinel

A todo momento, desde o início do voo, o drone realiza captura de imagens a serem apresentadas em um programa cliente utilizando sua câmera acoplada(2). Ele fará isso por meio de um servidor que disponibilizará um streaming de frames a uma sessão ativa.

Essas imagens serão recebidas e, ao sinal de um comando do usuário, processadas pelo programa cliente instalado num computador, que utilizará algoritmos de aprendizado de máquina e visão computacional para identificar a pessoa sob análise(4).

Para verificar se é uma pessoa conhecida, o sistema terá que comparar parâmetros faciais da pessoa em análise a pessoas já conhecidas e mapeadas em um banco de dados(5).

Se identificar, trará uma descrição dessa pessoa. Senão, indicará que a pessoa é uma pessoa não autorizada.

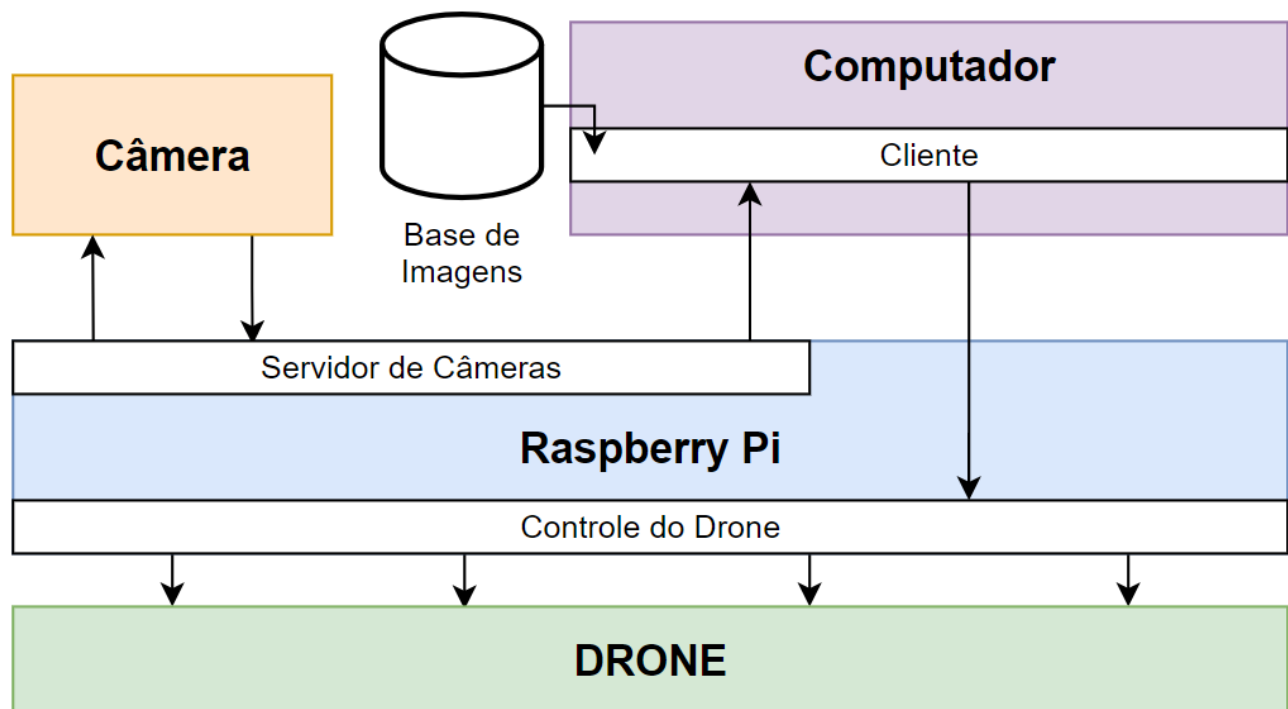


Figura 3: Drone

3 Escopo

Criação de um drone, com modelagem de suas peças e integração delas com o microcontrolador Raspberry, câmera e componentes eletrônicos, capaz de voar, parar e, em posição estável, capturar imagens e disponibilizá-la em um servidor, instalado no próprio Raspberry.

Um cliente se conecta usando um programa em C++ ou Python, recebe os frames e aplica algoritmos de Visão Computacional já existentes. Deste resultado, busca-se numa base de dados, que possui parâmetros faciais previamente preenchidos (serão inseridos ao longo do desenvolvimento do projeto), correspondência.

O sistema, por fim, apresentará uma imagem da pessoa com uma descrição ao lado. Se conhecida, com dados dela. Senão, um alerta de pessoa não autorizada.

3.1 Fora do Escopo

- **Desenvolvimento de algoritmos de visão computacional.** > Algoritmos prontos em Python serão implementados para a realização do pipeline de visão computacional.
- **Projeto de Chassi de Drone.** > Não há tempo hábil para criar um projeto do zero. Será utilizado um projeto existente do Thingiverse, modificando-o no Solidworks para encaixe das peças.
- **A comunicação com o drone será realizada por wifi** > Não será utilizado sistema de RF próprio, já que tanto o Raspberry quanto o computador possuem sistemas de wifi próprios.

4 Requisitos do Projeto

4.1 Requisitos Funcionais do Software

- RFS01 - O usuário decide quando voar com um comando
- RFS02 - O usuário controla o drone
- RFS03 - O usuário decide quando iniciar a autenticação
- RFS04 - O software mostra em tempo real a imagem da pessoa, capturada pelo Drone, na tela
- RFS05 - O usuário poderá inserir novas imagens no banco de dados de pessoas autorizadas.
- RFS06 - O processo de identificação da pessoa é feita automaticamente.
- RFS07 - O usuário decide quando desconectar do drone.

4.2 Requisitos Não Funcionais do Software

- RNFS01 - O servidor de frames rodará no Raspberry
- RNFS02 - O programa central rodará no computador
- RNFS03 - A base de dados será mongodb
- RNFS04 - Os algoritmos de Visão Computacional serão módulos de Python importados ao projeto
- RNFS05 - A apresentação na tela será em React
- RNFS06 - O servidor deverá escutar, enquanto ligado, por novas conexões, mas permitirá apenas uma conexão por vez
- RNFS07 - O software não permite desconexão durante o voo.
- RNFS08 - Se houver desconexão, o drone deverá executar um procedimento de segurança de pouso seguro

4.3 Requisitos Funcionais do Hardware

- RFH01 - O drone deve ser estável e de fácil controle
- RFH02 - A câmera deve enviar de forma ininterrupta imagens, a partir do momento que estiver voando

Requisitos Não Funcionais de Hardware

- RFNH01 - As peças deverão ser modeladas em 3D
- RFNH02 - O hardware deverá fazer um alerta luminoso e sonoro se tiver sem bateria.
- RFNH03 - O wifi a ser utilizado é do próprio Raspberry Pi

5 Objetivos

Os objetivos a serem alcançados podem ser divididos da seguinte maneira:

1. Desenvolver um módulo para gestão de câmera:

- (a) Integrar a câmera ao Raspberry[5] e configurá-la para capturar imagens em tempo real.
- (b) Implementar funções para controle e ajuste dos parâmetros da câmera, conforme necessário.

2. Desenvolver um módulo para servir frames:

- (a) Implementar um servidor de imagens em C++ no Raspberry para fazer o streaming de frames capturados pela câmera[?].
- (b) Garantir que o servidor seja capaz de lidar com múltiplas conexões e requisições de frames simultaneamente.
- (c) Otimizar o servidor para garantir baixa latência e bom desempenho no envio dos frames.

3. Desenvolver um módulo de conexão e comunicação entre cliente e servidor:

- (a) Implementar um protocolo de comunicação eficiente e confiável entre o programa cliente Python e o servidor de imagens no Raspberry.
- (b) Estabelecer conexão e troca de informações entre cliente e servidor, incluindo o envio de comandos e o recebimento de frames.

4. Desenvolver um módulo para receber frames no cliente:

- (a) Implementar um módulo no programa cliente Python para requisitar e receber frames do servidor de imagens no Raspberry.
- (b) Exibir os frames recebidos em uma página web em tempo real.

5. Desenvolver um módulo para processar frames no cliente:

- (a) Implementar um pipeline de visão computacional no programa cliente Python para realizar identificação facial.

- (b) Criar um modelo de dados com um gerenciador para inserir parâmetros faciais, e criar um processo de inclusão de rostos conhecidos.
- (c) Enviar comandos para o Raspberry iniciar e encerrar o envio de frames da pessoa a ser identificada.

6. Desenvolver um módulo para criar e controlar o equipamento voador não tripulado:

- (a) Modelar peças e circuitos necessários para criar um drone, e imprimi-los, deixando-os prontos.
- (b) Conectar e integrar as peças impressas, microcontrolador, câmera e componentes eletrônicos, de tal forma a fazê-los funcionamente unitariamente.
- (c) Criar um módulo para controlar o drone e fazê-lo voar e parar em uma determinada posição com segurança e estabilidade.

Além desses objetivos específicos, o projeto também tem como objetivo geral aplicar e aprimorar os conhecimentos adquiridos ao longo do curso de Engenharia Eletrônica, desenvolver habilidades multidisciplinares e aprofundar o entendimento dos alunos sobre o funcionamento de sistemas integrados.

6 Métodos

O desenvolvimento do projeto Sentinela envolve as seguintes etapas:

1. Levantamento das necessidades e requisitos: identificar os principais desafios e requisitos técnicos para a identificação de pessoas.
2. Seleção e implementação da câmera e servidor: escolher e instalar o dispositivo de captura de imagens mais adequados, levando em consideração fatores como resolução, alcance e condições ambientais. Ter o servidor que disponibilizará o streaming de frames para o cliente.
3. Projetar, modelar e criar drone que será utilizado para capturar imagens e vídeos de pessoas que deverão ser identificadas.
4. Desenvolvimento dos algoritmos de aprendizado de máquina e visão computacional: implementar e treinar modelos de inteligência artificial para reconhecer padrões de comportamento e identificar faces.
5. Desenvolvimento da interface de usuário: projetar e implementar um aplicativo ou portal web que permita aos usuários autorizados acessar o sistema e receber alertas em tempo

Serão utilizados os seguintes diagramas para planejamento e organização do desenvolvimento:

- Diagrama de Classes
- Diagrama de Estados
- Fluxograma do sistema

Uma prévia deles já foi modelada.

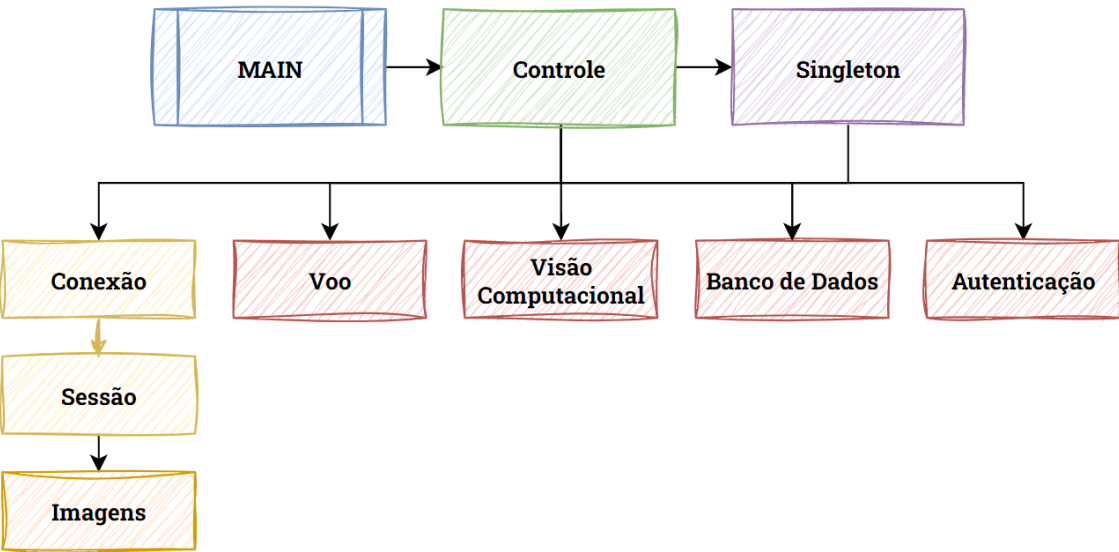


Figura 4: Diagrama de Classes

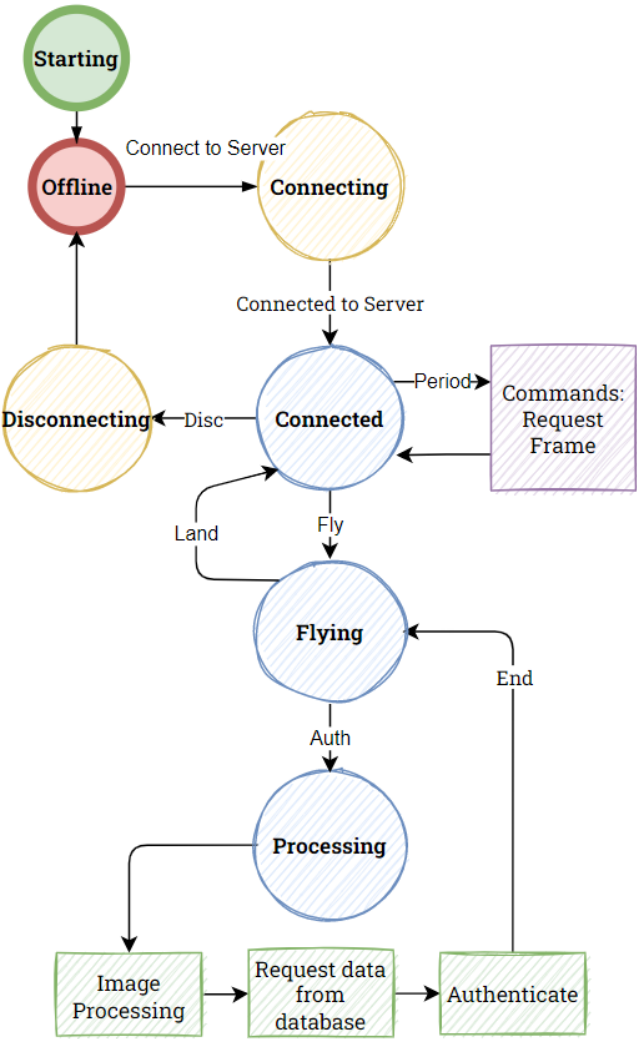


Figura 5: Diagrama de Estados do Cliente

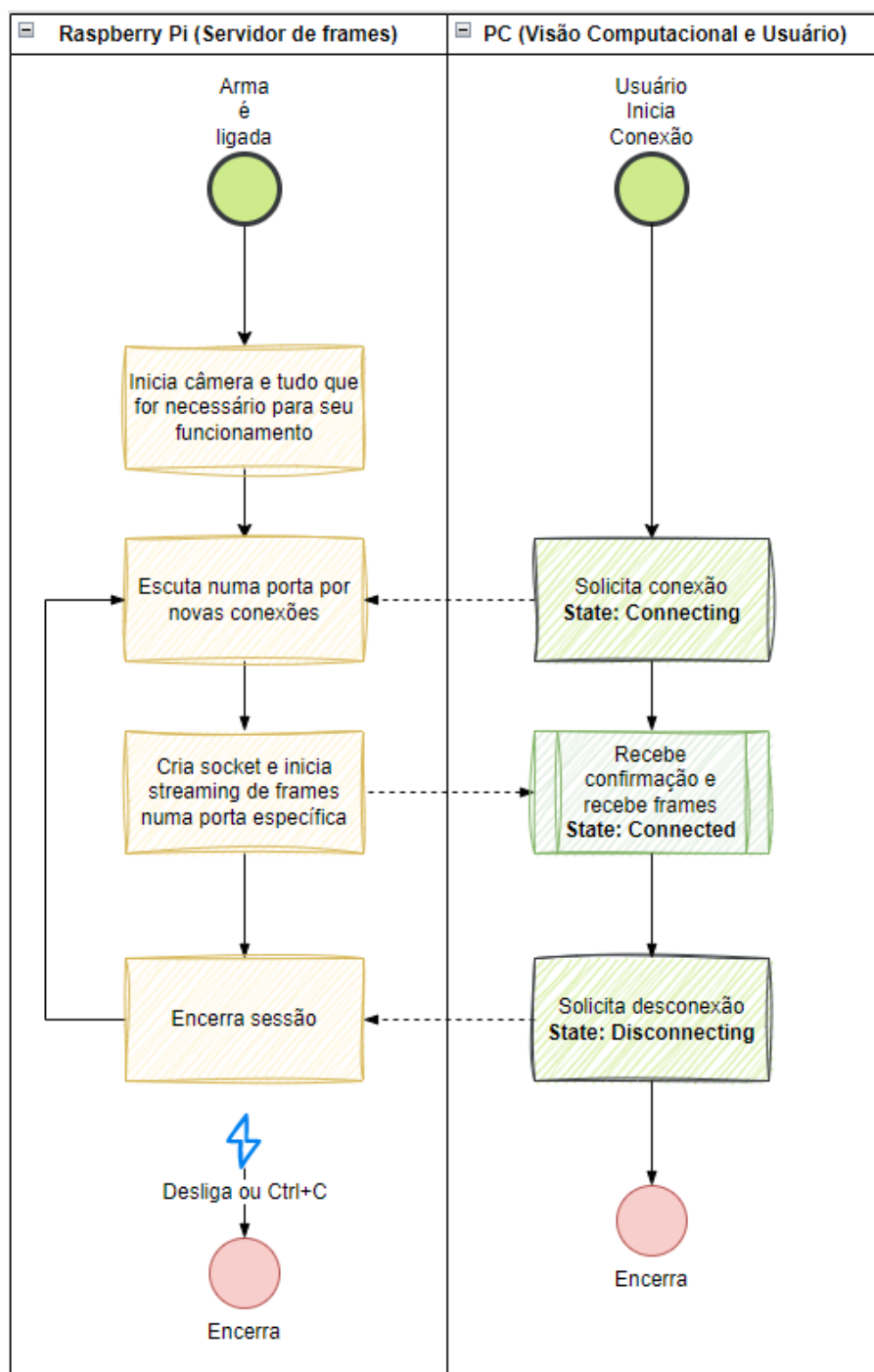


Figura 6: Fluxograma da relação Cliente Servidor

6.1 Software

Software	Uso	Licença	Custo
VSCode	Programação	MIT	Gratuito
CMake	Compilar código para Raspberry	LGPL/GPL	Gratuito
Solidworks	Modelagem de objetos 3D	Proprietária	Pago
Python	Programação de algoritmos de visão computacional	Python Software Foundation License	Gratuito
Node.js/C++	Servidor de imagens	MIT/GPL	Gratuito
Draw.io	Criação de diagramas	Apache 2.0	Gratuito
Overleaf	Editor LaTeX online	Gratuito	Gratuito
GitHub	Versionamento e compartilhamento de códigos	Gratuito	Gratuito
Git	Gerenciamento de código	GNU GPL	Gratuito
CircuitMaker	Criação de placas impressas	GPL	Gratuito
Fritzing	Diagramas de Raspberry mais visuais	GNU GPL	Gratuito

Tabela 1: Lista de softwares, usos, licenças e custos

6.2 Hardware

Componente	Qtd	Valor	Total	Gasto	Fornecedor	Status
Projeto - Sentinel Eyes	\$		2,824.20	\$ 1,304.20		
Raspberry Pi 4 Model B	1	\$1,000.00	\$ 1,000.00	\$ -	Já possuo	Disponível
Câmera Raspberry V2	1	\$ 320.00	\$ 320.00	\$ -	Já possuo	Disponível
R-Line Tattu 4S 1300mAh 95C	1	\$ 211.07	\$ 211.07	\$ 211.07	Aliexpress	Correios
Carregador de Bateria Imax B6 80W	1	\$ 166.66	\$ 166.66	\$ 166.66	Aliexpress	Correios
LiPo Battery Monitor and Buzzer	1	\$ 33.56	\$ 33.56	\$ 33.56	Aliexpress	Correios
Diatone Mamba F405 mk2 & F40 ESC	1	\$ 402.83	\$ 402.83	\$ 402.83	Aliexpress	Correios
Motor XING-E XING 2207 1800KV	4	\$ 73.30	\$ 293.18	\$ 293.18	Aliexpress	Correios
Hélice tripla Hurricane	4	\$ 15.93	\$ 63.71	\$ 63.71	Aliexpress	Correios
XT60 3-6S to USB 5V Adapter	1	\$ 46.58	\$ 46.58	\$ 46.58	Aliexpress	Correios
XT60 Y-Splitter	1	\$ 37.91	\$ 37.91	\$ 37.91	Aliexpress	Correios
Cartão de memória 64GB	1	\$ 48.70	\$ 48.70	\$ 48.70	Aliexpress	Correios
Impressão 3D	1	\$ 200.00	\$ 200.00	\$ -	Procurar	A realizar

Figura 7: Previsão Orçamentária

6.3 Ferramentas

7 Gantt

		Abril				Maio					Junho			
		1	2	3	4	1	2	3	4	5	1	2	3	4
M01 - Apresentação dos Planos de Projetos		×												
M01.T01	Identificar as necessidades e requisitos técnicos.													
M01.T02	Definir os objetivos e escopo do projeto.													
M01.T03	Elaborar a proposta e cronograma.													
Entrega	Proposta do Projeto com objetivo, descrição, atividades e cronograma.	04/04												
M02 -Apresentação do Site/Blog de Acompanhamento			×											
M02.T01	Procurar e definir qual site/blog será utilizado.													
M02.T02	Preencher o site/blog com os dados da proposta.													
Entrega	Site pronto e com todas as informações de acompanhamento do projeto.		11/04											
M03 - Apresentação do projeto, desenvolvimento e testes do hardware						×								
M03.T01	Modelar peças e circuitos necessários para criar um drone, e imprimi-los, deixando-os prontos (objetivo 6).													
M03.T02	Pesquisar, escolher e implementar a câmera no Raspberry (objetivo 1).													
M03.T03	Projetar, modelar e construir placas auxiliares para funcionamento do drone (objetivo 6).													
M03.T04	Implementar funções para controle e ajuste dos parâmetros da câmera conforme necessário (objetivo 1).													
Entrega	Drone voando em frente à pessoa de interesse, capturando imagem da mesma e apresentando em algum lugar essa imagem.					02/05								
M04 - Apresentação do projeto, desenvolvimento e testes do software									×					
M04.T01	Desenvolver um módulo para servir frames do drone para o cliente (objetivo 2).													
M04.T02	Desenvolver um módulo de conexão e comunicação entre cliente e servidor (objetivo 3).													
M04.T03	Desenvolver módulo para receber frames no cliente (objetivo 4).													
M04.T04	Apresentar em página web a imagem da pessoa a ser identificada em tempo real (objetivo 4).													
M04.T05	Implementar pipeline de visão computacional para identificação de parâmetros faciais.													
M04.T06	Criar base de dados e sistema de gerenciamento de parâmetros faciais conhecidos (objetivo 5).													
Entrega	Algoritmos funcionando com hardcode (sem interconexão com outros módulos) e base de dados pronta.								23/05					
M05 - Apresentação da Integração mecânica/hardware/software + testes funcionais.											×			
M03.T01	Integrar identificação ao sistema cliente, identificando pessoas de forma hardcoded.													
M03.T02	Integrar identificação à base de dados, identificando pessoas de forma hardcoded.													
M03.T03	Integrar identificação ao sistema cliente e à base de dados, identificando a pessoa e descrevendo-a de acordo com seus parâmetros.													
Entrega	Apresentar o Sentinel pronto e funcionando.										06/06			
Evento 01 Entrega do Relatório, Vídeo e Blog.												13/06		
Evento 02	Avaliação dos professores.												20/06	

Figura 8: Diagrama de Gantt

8 Riscos

Id	Risco	Prob	Grav	PG
1	Não conseguir modelar as peças.	2	5	10
	Contornar: comprar peças prontas para Drones e pontuar no relatório.			
2	O drone não estabilizar em frente à pessoa.	1	5	5
	Aceitar: fazer ele pousar próximo à pessoa e iniciar o streaming.			
3	O raspberry ou a câmera queimar.	1	5	5
	Contornar: se não houver reposição sem custo (são equipamentos muito caros), utilizar o computador para captura de imagens e pontuar no relatório.			
4	Comunicação entre drone e computador com problema.	1	5	5
	Eliminar: fazer o drone capturar imagens em um pendrive ou similar, e transferir os frames para o computador.			

Tabela 2: Tabela de riscos

9 Referências Bibliográficas

- [1] **Matrix Fandom**
<https://matrix.fandom.com/wiki/Sentinel>, Acesso em 25 de março de 2023.
- [2] **Taigman, Y., Yang, M., Ranzato, M., & Wolf, L. (2014).** DeepFace: Closing the Gap to Human-Level Performance in Face Verification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1701-1708. doi:10.1109/CVPR.2014.220
- [3] **Al-Nawashi, M., Al-Hazaimh, O., & Saraee, M. (2018).** Drone-Based Intelligent Transportation Systems for Smart Cities: The Role of Drone Technology in Smart Cities. In *Proceedings of the 2018 9th International Conference on Information and Communication Systems (ICICS)*, 186-191. doi:10.1109/IACS.2018.8355459
- [4] **Microdone**
<https://dojofordrones.com/raspberry-pi-pico-drone-diy-micro-drones/>, Acesso em 25 de março de 2023.
- [5] **Raspberry**
<https://www.raspberrypi.com/products/camera-module-v2/>, Acesso em 25 de março de 2023.