

# RELATÓRIO TÉCNICO – YOUVISA Sprint 2

Plataforma Cognitiva de Automatização Consular

## 1. RESUMO

Durante a Sprint 2, desenvolvemos o protótipo funcional do backend para a plataforma YOUVISA, focado na automação e digitalização do recebimento e validação de documentos consulares. O projeto aplica práticas modernas de modularização, segurança e integração de IA, alinhado às demandas acadêmicas e do desafio proposto.

## 2. ARQUITETURA DO PIPELINE

A solução é composta pelos seguintes módulos integrados:

### Backend FastAPI:

Responsável por receber uploads de documentos via endpoint REST, gerenciar o pipeline de automação e expor interfaces para integração com frontend/chatbot.

### Pipeline de Automação:

- Classificação por NLP:** Identifica tipo de documento usando regras baseadas em nome/atributos do arquivo (pronto para modelos transformers).
- Validação Visual simulada:** Confere formato/estrutura por extensão (preparado para integração com OpenCV).
- Automação RPA/Email:** Envia confirmação automática por e-mail ao usuário via SMTP.

### Banco de Dados (Firestore):

Armazena dados dos processos/documentos, status e histórico com timestamps automáticos.

### Integração com Frontend/Chatbot:

O endpoint `/upload` simula o recebimento via chatbot, e os endpoints REST estão prontos para serem consumidos por painel React ou outros clientes.

## 3. ETAPAS DE DESENVOLVIMENTO

- Definição e organização de módulos** na estrutura `/src/backend/app`, segmentando pipeline, RPA, DB e testes.

2. **Implementação do pipeline:** upload de documento, classificação automática, validação visual e resposta automatizada.
  3. **Configuração do Firestore** com credenciais protegidas, registro de cada caso e status.
  4. **Simulação e integração de e-mails automáticos** usando SMTP seguro (Gmail e App Password).
  5. **Testes do fluxo completo** via Swagger UI, com registros real-time em Firestore.
  6. **Documentação técnica e evidências:** README completo, diagrama de arquitetura, capturas e vídeo demonstrativo.
- 

## 4. DESAFIOS E SOLUÇÕES

### Estruturação de imports e caminhos:

Resolvemos conflitos usando `os.path` para caminhos dinâmicos, além de garantir módulos com arquivos

`__init__.py`.

### Integração SMTP:

Implementação funcional com logs e fallback seguro para falhas de envio.

### Validação visual inicial:

Realizada por análise de extensão; arquitetura preparada para OpenCV no futuro.

### Controle de segurança:

Uso de `.gitignore` para evitar upload de credenciais e artefatos desnecessários ao repositório.

### Gestão de repositório Git:

Configuração de remoto, resolução de conflitos e push limpo dos arquivos do projeto.

---

## 5. RESULTADOS OBTIDOS

- Pipeline modular e funcional: upload, classificação, validação, registro automático em Firestore e disparo de e-mail.
  - Todos os endpoints testados em Swagger UI, com evidências visuais e logs.
- Arquitetura preparada para integração com frontend React, bots reais e módulos de IA/visão avançados.
  - Documentação completa: README, diagrama de arquitetura, relatório técnico e vídeo demonstrativo.
    - Testes automatizados com pytest (100% pass rate nos testes principais).

### Evidências:

- Capturas do Swagger UI mostrando endpoints funcionais
  - Logs de console demonstrando pipeline de classificação e envio de email
  - Registros no Firestore com casos e status atualizados
  - Vídeo demonstrativo do fluxo completo (disponível em `/docs`)
-

# 6. PRÓXIMOS PASSOS

- Integrar modelos reais de NLP (spaCy, transformers) e visão computacional (OpenCV/CNN).
  - Implementar frontend React completo com visualização de casos e documentos.
  - Melhorar interface agente e integrar com bots reais (Telegram, WhatsApp).
  - Refinar controles de segurança, validação de inputs e logs de auditoria.
  - Expandir documentação para casos de erros e reprocessamento de documentos.
  - Deploy em ambiente cloud (AWS, GCP ou Azure).
- 

# 7. INTEGRANTES DO PROJETO

Nome	Papel	E-mail
[Nome 1]	Backend/API/Pipeline	[email]
[Nome 2]	Firestore/Integração	[email]
[Nome 3]	Frontend/Documentação	[email]
[Nome 4]	QA/Testes	[email]

(Preencha com os dados da sua equipe)

---

**Data:** Novembro de 2025

**Versão:** Sprint 2 - Protótipo Funcional

**Status:** Concluído e Testado

---

# ANEXOS

- README.md completo com instruções de instalação e uso
- Diagrama de arquitetura do pipeline (ver /docs/fluxograma\_pipeline.png)
- Capturas de tela do fluxo operacional
- Vídeo demonstrativo (2-4 minutos)
- Código fonte no repositório: [https://github.com/caliraselph/YouVisa\\_Sprint2](https://github.com/caliraselph/YouVisa_Sprint2)  
[\(https://github.com/caliraselph/YouVisa\\_Sprint2\)](https://github.com/caliraselph/YouVisa_Sprint2)