



CENTRO UNIVERSITÁRIO DE BRASÍLIA
CIÊNCIA DA COMPUTAÇÃO

FELIPE BARCELOS DE CARVALHO (RA: 22350044)
MARCOS VINICIUS ROCHA (RA: 22352865)
JOÃO MARCELO GUIMARÃES DOURADO (RA: 22350653)
EDUARDO ARAÚJO UCHOA (RA: 22353207)
DAVI MAIA (RA: 22305561)

PROJECT AEGIS:
PAINEL DE INTELIGÊNCIA DE AMEAÇAS

BRASÍLIA
2025

Cronograma de Execução — Projeto Integrador I: Project Aegis

Status: Atualização para refletir o fluxo de Desenvolvimento Real (Backend Python → Frontend → Automação)

Detalhamento das Sprints (Quinzenais)

Sprint 1: Concepção e Arquitetura

Foco: Definição estrutural e configuração do ambiente.

- **Definição do problema e Personas:** Alinhamento do escopo referente à fragmentação de dados de segurança e definição do público-alvo.
- **Escolha da Stack Tecnológica:** Determinação do uso de Python para o desenvolvimento do backend e GitHub Actions para a automação de processos.
- **Configuração do Ambiente:** Criação do repositório de código, configuração do arquivo `.gitignore` e estabelecimento do ambiente virtual de desenvolvimento.

Sprint 2: Backend e Coleta de Dados (ETL)

Foco: Desenvolvimento dos scripts de extração, transformação e carga (ETL).

- **Scripts Python:** Desenvolvimento e implementação dos módulos de coleta de dados (`coletor_cve.py`, `coletor_otx.py`, etc.).
- **Conexão com APIs:** Testes de requisição, resposta e autenticação com as APIs do NVD (National Vulnerability Database), OTX AlienVault e HIBP.
- **Tratamento de Dados:** Normalização das informações coletadas para garantir consistência e exportação estruturada para arquivos JSON.

Sprint 3: Frontend e Visualização

Foco: Interface do usuário, experiência e apresentação dos dados.

- **Estrutura Web:** Desenvolvimento da estrutura base utilizando HTML5 semântico e estilização com CSS3.
- **Dashboard:** Implementação da lógica de visualização de dados para o painel administrativo.
- **Gráficos:** Integração da biblioteca **Chart.js** para renderização visual das estatísticas de vulnerabilidades e ameaças.
- **Integração:** Desenvolvimento da camada de leitura para exibição dinâmica dos dados processados nos arquivos JSON gerados pelo Backend.

Sprint 4: DevOps e Automação (Fase Atual)

Foco: Integração Contínua (CI), Entrega Contínua (CD) e Segurança.

- **Workflow YAML:** Configuração e otimização do arquivo de fluxo de trabalho (`deploy_diario.yml`) no GitHub Actions.
- **Segurança:** Implementação de *GitHub Secrets* para gerenciamento seguro de chaves de API e credenciais sensíveis, evitando exposição no código-fonte.
- **CI/CD:** Execução de testes de deploy automático, agendamento de execução de scripts (Cron jobs) e correção de falhas no pipeline de integração.

Sprint 5: Documentação e Entrega Final

Foco: Finalização acadêmica, revisão técnica e preparação para banca.

- **Manuais:** Elaboração da documentação técnica detalhada da API e do guia de usuário para o Dashboard.
- **QA Final:** Revisão de garantia de qualidade, validação de requisitos e testes finais de ponta a ponta (End-to-End).
- **Banca:** Preparação dos slides, roteiro de apresentação e consolidação dos artefatos para a defesa final do projeto.

Cronograma de Execução (Resumo)

Sprint	Fase	Atividades Chave	Entregas Principais
1	Concepção e Arquitetura	Definição de Stack e Ambiente	Repositório Configurado
2	Backend e ETL	Scripts Python e APIs	Arquivos JSON de dados tratados
3	Frontend e Visualização	HTML/CSS e Chart.js	Dashboard funcional integrado
4	DevOps	Workflow YAML e Secrets	Pipeline de CI/CD em execução
5	Documentação e Entrega	Manuais e QA	Relatório Técnico e Apresentação