



**UNIVERSIDADE
E D U A R D O
MONDLANE**

**FACULDADE DE ENGENHARIA
DEPARTAMENTO DE ELECTROTECNIA
LICENCIATURA EM ENGENHARIA INFORMÁTICA
ENGENHARIA DE SOFTWARE II**

**Relatório de Projecto
Sistema de Emissão de Certificados**

**Data início: 18/09/2024
Data fim prevista: 14/11/2024**

Estudante:

Lino, Miro Pedro Tipaneque

Docentes:

Sérgio Mavie

Kalid Bapú

Maputo, 20 de Novembro de 2024



**FACULDADE DE ENGENHARIA
DEPARTAMENTO DE ELECTROTECNIA
LICENCIATURA EM ENGENHARIA INFORMÁTICA
ENGENHARIA DE SOFTWARE II**

**Relatório de Projecto
Sistema de Emissão de Certificados**

Data início: 18/09/2024

Data fim prevista: 14/11/2024

Estudante:

Lino, Miro Pedro Tipaneque

Maputo, 20 de Novembro de 2024

Índice

1. Introdução	4
2. Objectivos	5
Geral	5
Específicos	5
3. Revisão de Conceitos	6
3.1. Sistemas de Informação Académica	6
3.2. Certificados Académicos	6
3.3. Interoperabilidade entre Sistemas	7
3.4. Plataforma X-Road	7
3.5. API (Interface de Programação de Aplicações)	7
3.6. Desenvolvimento de Sistemas Informatizados	8
4. Metodologia	8
4.1. Estrutura Organizacional do Projecto	9
4.2. Metodologia Scrum Aplicada no Projecto	9
4.3. Ferramentas de Apoio	10
4.4. Desafios Enfrentados	11
5. Actividades Desenvolvidas	11
6. Principais Conquistas	13
7. Desafios e riscos	14
8. Necessidade e suporte	15
9. Avaliação da cadeira	15
10. Conclusão	17
11. Referências:	18
12. Apêndices	19

1. Introdução

O Sistema de Emissão de Certificados (SISEC) foi desenvolvido como uma solução tecnológica para modernizar e otimizar o processo de emissão de certificados académicos na Universidade Eduardo Mondlane. Tradicionalmente, a gestão académica e a emissão de certificados são realizadas de forma manual, utilizando registos físicos, como pautas e documentos em papel, o que resulta em processos demorados, propensos a erros e com alto custo operacional. Esse cenário não apenas afeta a eficiência administrativa, mas também compromete a satisfação dos estudantes e a credibilidade das instituições de ensino.

O SISEC surge como uma proposta feita pelo docente desta cadeira, **Sérgio Mavie**, como um desafio para a turma que consistiu em desenvolver dois sistemas interoperativos que venham responder de forma directa às limitações mencionadas no parágrafo anterior, oferecendo um sistema informatizado capaz de consultar, verificar e processar as informações académicas de forma automática e segura. Sua principal funcionalidade é interagir com o Sistema de Gestão de Estudantes (SIGE) para acessar os dados necessários à emissão de certificados, como nomes, cursos, notas e disciplinas concluídas. Essa integração é possibilitada pela utilização da plataforma X-Road, que garante a interoperabilidade entre os sistemas, assegurando uma comunicação eficiente e confiável.

Este relatório apresenta o desenvolvimento do Sistema de Emissão de Certificados, detalhando os objectivos do projecto, a arquitetura do sistema, os requisitos funcionais e não funcionais, além das metodologias utilizadas para a implementação e os resultados alcançados. O documento também explora os desafios enfrentados durante o desenvolvimento e as soluções adotadas para garantir a interoperabilidade entre os sistemas envolvidos.

2. Objectivos

Geral

- ◆ Desenvolver um Sistema de Emissão de Certificados (SISEC) informatizado, que automatize o processo de emissão de certificados académicos, garantindo precisão, rapidez, padronização e integração com o Sistema de Gestão de Estudantes (SIGE) através da plataforma de interoperabilidade X-Road.

Específicos

- ◆ Desenvolver funcionalidades que permitam a geração de certificados académicos de forma automatizada, minimizando erros manuais e garantindo a consistência das informações;
- ◆ Implementar uma interoperabilidade segura e eficiente que possibilite a consulta em tempo real dos dados académicos necessários para a emissão de certificados e;
- ◆ Estabelecer um modelo padrão para os certificados emitidos, contendo informações como nome do estudante, curso, disciplinas concluídas, notas, e demais dados relevantes.

3. Revisão de Conceitos

A revisão de conceitos é fundamental para contextualizar e fundamentar os aspectos técnicos e teóricos que sustentam o desenvolvimento do Sistema de Emissão de Certificados (SISEC). Nesta secção, serão abordados os conceitos-chave relacionados à gestão académica, interoperabilidade, sistemas de informação e tecnologias utilizadas.

3.1. Sistemas de Informação Académica

Os sistemas de informação académica são ferramentas computacionais projetadas para gerir e automatizar processos administrativos e pedagógicos em instituições de ensino. Esses sistemas centralizam informações sobre estudantes, disciplinas, notas, cursos e demais dados académicos, promovendo maior eficiência e organização.

Segundo Laudon & Laudon (2020), "os sistemas de informação são essenciais para gerir dados e transformar informações em conhecimento, facilitando a tomada de decisões e otimizando processos administrativos".

No contexto deste projecto, o SIGE é o responsável por armazenar e processar os dados académicos, sendo essencial para alimentar o Sistema de Emissão de Certificados (SISEC).

Benefícios:

- ✓ Redução de erros humanos associados à gestão manual.
- ✓ Centralização e padronização de informações.
- ✓ Aumento da eficiência operacional.

3.2. Certificados Académicos

Os certificados académicos são documentos oficiais emitidos por instituições de ensino que atestam a conclusão de cursos, disciplinas ou actividades específicas. Eles devem conter informações precisas e verificáveis, como nome do estudante, curso, disciplinas concluídas, e data de emissão.

De acordo com Oliveira (2018), "a digitalização dos certificados académicos reduz custos, aumenta a segurança e facilita o acesso, atendendo às demandas de um mundo cada vez mais informatizado".

Características dos Certificados Académicos Digitalizados:

- ✓ Formato padrão e padronizado.
- ✓ Inclusão de mecanismos de autenticação, como QR codes ou códigos de validação.
- ✓ Disponibilidade em formatos físico e digital (PDF).

3.3. Interoperabilidade entre Sistemas

A interoperabilidade é a capacidade de diferentes sistemas e organizações trabalharem em conjunto de maneira eficiente e segura, compartilhando dados e processos.

Segundo Gottschalk (2009), "a interoperabilidade é um aspecto crítico para sistemas complexos, permitindo a integração de serviços e a troca de dados de maneira segura e escalável".

No projecto, a interoperabilidade é implementada pela plataforma X-Road, que conecta o SIGE ao SISEC.

3.4. Plataforma X-Road

X-Road é uma infraestrutura de troca de dados segura e padronizada, que permite a comunicação entre diferentes sistemas de forma confiável.

Segurança: Garante criptografia e autenticação em todas as comunicações.

Escalabilidade: Suporta o crescimento do sistema e o aumento no número de transações.

Auditabilidade: Regista todas as transações realizadas entre os sistemas.

3.5. API (Interface de Programação de Aplicações)

Uma API (Application Programming Interface) é um conjunto de definições e protocolos que permitem que diferentes sistemas se comuniquem entre si. No contexto do projecto, a API do SIGE disponibiliza os dados necessários para o SISEC consultar as informações académicas.

Características da API:

- ✓ **RESTful APIs:** Adotadas pela simplicidade e compatibilidade com protocolos HTTP.
- ✓ **Documentação:** Necessária para facilitar a integração e manutenção.
- ✓ **Segurança:** Uso de autenticação por tokens para garantir que apenas sistemas autorizados acessem os dados.

Como citado por Fielding (2000), "as APIs RESTful são amplamente utilizadas devido à sua simplicidade, escalabilidade e eficiência em sistemas distribuídos".

3.6. Desenvolvimento de Sistemas Informatizados

A construção de sistemas informatizados segue metodologias ágeis, como Scrum, para garantir que os requisitos sejam atendidos de maneira eficiente.

Etapas de Desenvolvimento:

- i. **Levantamento de Requisitos:** Identificação de necessidades e funcionalidades.
- ii. **Modelagem do Sistema:** Definição da arquitetura, banco de dados e fluxos de informação.
- iii. **Implementação:** Codificação das funcionalidades com linguagens modernas e ferramentas adequadas.
- iv. **Testes:** Verificação e validação para garantir o funcionamento correto.
- v. **Implantação:** Integração do sistema no ambiente operacional.

De acordo com Schwaber & Sutherland (2020), "metodologias ágeis permitem que equipas entreguem software funcional com maior eficiência, respondendo rapidamente a mudanças nos requisitos".

4. Metodologia

Para garantir a organização, a eficiência e a qualidade no desenvolvimento do Sistema de Emissão de Certificados, foi adoptada uma metodologia ágil, baseada principalmente no framework **Scrum**. Essa metodologia foi escolhida por ser amplamente utilizada em projectos de desenvolvimento de software devido à sua

capacidade de promover a colaboração, priorização de tarefas e entrega incremental de funcionalidades. Além disso, considerando a divisão do grupo em subgrupos especializados, o Scrum permite flexibilidade e integração contínua das equipas.

4.1. Estrutura Organizacional do Projecto

A turma foi dividida em dois grupos principais:

Grupo 1: Desenvolvimento do Sistema de Gestão de Estudantes (SIGE).

Grupo 2: Desenvolvimento do Sistema de Emissão de Certificados (SISEC).

Dentro do Grupo 2, responsável pelo SISEC, houve uma subdivisão funcional:

Backend	Frontend	Administração	Cybersecurity
Tembe, Hector	Macaneta, Lino	Deve, Yuren	Pacule, Manuel
Matimbe, José	Massingue, Dionalde	Golias, Marcos	Sibiya, Ndiwani
Macucule, Evgueni	Nhambire, Gabriel	Semo, Agneila	
Almeida, Henriques	Chavele, Fernando		
Buque Júnior, Nicolau	Ntema, Alexandre		
Niaure, Geraldo			

Tabela 1: Grupos por área de preferência

Essa divisão permitiu que cada equipa se concentrasse em sua área de especialização, promovendo maior produtividade e qualidade nas entregas.

4.2. Metodologia Scrum Aplicada no Projecto

4.2.1. Papéis no Projecto

- i. **Product Owner:** Representado pelo Docente, **Sérgio Mavie**, que definiu os objectivos do sistema e priorizou os requisitos de acordo com as necessidades da instituição.
- ii. **Scrum Master:** **Miro Pedro Tipaneque Lino**, o gestor do grupo principal que coordenou a execução das tarefas, removeu impedimentos e garantiu que a metodologia fosse aplicada correctamente.
- iii. **Equipas de Desenvolvimento:** Formadas pelos subgrupos (backend, frontend, administração e cybersecurity), que executaram as tarefas específicas.

4.2.2. Etapas do Scrum no Projecto:

i. planeamento do Sprint (Sprint Planning):

No início de cada sprint (um ciclo de trabalho com duração de 1 a 2 semanas), as equipas se reuniram para:

- ✓ Definir as funcionalidades a serem desenvolvidas.
- ✓ Dividir as tarefas em itens menores e atribuí-los aos membros do grupo.
- ✓ Estabelecer metas claras para cada sprint.

ii. Reuniões Diárias (Daily-Standups):

Cada equipa realizou reuniões rápidas (15 minutos) para:

- ✓ Relatar o que foi feito no dia anterior.
- ✓ Compartilhar o que seria feito no dia atual.
- ✓ Identificar e discutir possíveis impedimentos.

iii. Desenvolvimento Iterativo:

As tarefas foram divididas em ciclos curtos e iterativos, permitindo a entrega incremental de funcionalidades.

- ✓ O grupo de Backend focou na criação de APIs REST para atender às solicitações do Frontend e garantir a integração com o banco de dados.
- ✓ O grupo de Frontend desenvolveu e testou as interfaces do usuário, garantindo responsividade e usabilidade.
- ✓ Os grupos de Administração e CyberSecurity concentraram-se na configuração segura e no teste de interoperabilidade via X-Road.

iv. Revisão do Sprint (Sprint Review):

- ✓ Ao final de cada sprint, as equipas apresentaram os resultados ao Product Owner e aos colegas.
- ✓ O feedback recebido foi incorporado aos sprints subsequentes.
- ✓ Alterações ou melhorias foram priorizadas e planeadas para os próximos ciclos.

v. Retrospectiva do Sprint (Sprint Retrospective):

Após cada sprint, as equipas avaliaram:

- ✓ O que funcionou bem.
- ✓ O que poderia ser melhorado.
- ✓ Planos de ação para aumentar a produtividade e minimizar problemas futuros.

4.3. Ferramentas de Apoio

Durante o projecto, várias ferramentas foram utilizadas para apoiar a metodologia ágil e o trabalho colaborativo:

Ferramenta	Descrição
Trello	Para gestão de tarefas e acompanhamento do progresso dos sprints.
GitHub	Para controle de versão e integração contínua do código.
Postman	Para testar e validar as APIs desenvolvidas pelo grupo de Backend.
Docker	Para padronização de ambientes e garantir que os sistemas funcionem uniformemente em diferentes plataformas.

Tabela 2: Ferramentas de apoio

4.4. Desafios Enfrentados

Apesar das vantagens do Scrum, o projecto enfrentou alguns desafios:

- ✧ Coordenação entre Subgrupos: A integração entre o backend e o frontend exigiu um alinhamento constante para evitar incompatibilidades.
- ✧ Aprendizado da Plataforma X-Road: A configuração inicial e o entendimento dos mecanismos de segurança da X-Road demandaram mais tempo do que o previsto.
- ✧ Gestão de Conflitos de Agenda: Com a participação de vários membros em diferentes equipas, a coordenação de horários foi um desafio.

5. Actividades Desenvolvidas

Abaixo temos tabela de actividades para o projecto, incluindo os responsáveis por cada subgrupo, as actividades realizadas e as datas correspondentes.

Actividade	Subgrupo Responsável	Responsável	Data de Início	Data de Término	Descrição
Planeamento do projecto	Todos	Todos	18/09/2024	20/09/2024	Reuniões iniciais para definir escopo, requisitos e divisão de tarefas.
Configuração do ambiente de	Backend	Hector Tembe	21/09/2024	28/09/2024	Instalação e configuração do

backend					Laravel, banco de dados e preparação de endpoints iniciais.
Desenvolvimento do frontend inicial	Frontend	Gabriel Nhambire	21/09/2024	05/10/2024	Criação das interfaces básicas em ReactJS, incluindo telas de login e solicitação.
Integração inicial com X-Road	Administração	Yuren Deve	28/09/2024	06/10/2024	Configuração inicial da interoperabilidade entre sistemas utilizando a plataforma X-Road.
Implementação de medidas de segurança	Cibersegurança	Manuel Pacule	28/09/2024	08/10/2024	Configuração de certificados digitais e políticas de acesso seguro.
Desenvolvimento de endpoints de API	Backend	Hector Tembe	29/09/2023	12.10/2024	Desenvolvimento de APIs REST para atender às funcionalidades de emissão de certificados.
Desenvolvimento avançado de frontend	Frontend	Gabriel Nhambiri	06/10/2024	15/10/2024	Implementação de funcionalidades detalhadas e integração com APIs do backend.
Testes de interoperabilidade	Administração	Yuren Deve	09/10/2024	20/10/2024	Testes para validar a comunicação entre sistemas

					utilizando X-Road.
Auditoria de segurança do sistema	Cibersegurança	Manuel Pacule	16/10/2024	22/10/2024	Verificação das vulnerabilidades e melhorias na segurança dos dados trocados.
Testes funcionais e ajustes	Todos	Todos	23/10/2024	06/11/2024	Validação das funcionalidades desenvolvidas e correção de erros.
Documentação do sistema	Todos	Todos	07/11/2024	12/11/2024	Elaboração da documentação técnica e manual do sistema.
Entrega e apresentação final	Todos	Todos	13/11/2024	14/11/2024	Apresentação do projecto para a turma/docente, destacando resultados e desafios superados.

Tabela 3: Actividades realizadas

6. Principais Conquistas

Apesar das dificuldades enfrentadas, o projecto alcançou resultados notáveis, incluindo:

- **Desenvolvimento do Sistema de Emissão de Certificados:** A equipa conseguiu implementar funcionalidades principais, como a emissão automatizada de certificados, integrando o sistema de gestão de estudantes via X-Road.
- **Configuração Inicial do X-Road:** Mesmo enfrentando limitações técnicas, a equipa de administração conseguiu configurar parcialmente a plataforma, possibilitando a interoperabilidade básica entre os sistemas.

- **Capacitação em Novas Tecnologias:** Os membros do backend e frontend desenvolveram maior proficiência em Laravel e ReactJS, o que representou um aprendizado significativo.

7. Desafios e Riscos

I. Equipa de Administração

- Falta de Conhecimento Técnico:** A falta de familiaridade com o X-Road dificultou a configuração da interoperabilidade.
- Hardware Insuficiente:** A incapacidade de executar a virtualização do Ubuntu devido à falta de computadores com memória RAM suficiente atrasou o progresso.

II. Equipa de Backend

- Aprendizado do Laravel:** Muitos membros da equipa não possuíam experiência prévia com Laravel, o que resultou em atrasos no desenvolvimento das APIs.

III. Equipa de Frontend:

- Integração Frontend e Backend:** Dificuldades em conectar as interfaces desenvolvidas com os endpoints do backend prejudicaram o fluxo de dados esperado.

IV. Gestão do Projecto

- Gestão de Problemas Técnicos:** Os gestores enfrentaram desafios significativos ao tentar resolver os problemas técnicos das equipas, especialmente por não possuírem total domínio de todas as ferramentas usadas.
- Gestão do Tempo:** Ajustar o cronograma às dificuldades técnicas foi um desafio constante.

Riscos Identificados:

- Atraso no Prazo:** As dificuldades técnicas poderiam comprometer o prazo final do projecto.
- Falhas na Interoperabilidade:** Caso o X-Road não fosse configurado corretamente, o principal objetivo do projecto (emissão de certificados automatizada) não seria alcançado.

c) **Desmotivação das Equipas:** As barreiras técnicas e a pressão por resultados poderiam desmotivar os membros, afetando a produtividade.

8. Necessidades e Suporte

As necessidades identificadas foram:

a) **Capacitação Técnica:** Treinamento em X-Road para a equipa de administração, abordando tanto a configuração quanto a integração segura. Cursos ou workshops sobre Laravel e ReactJS para as equipas de backend e frontend, reduzindo a curva de aprendizagem.

b) **Melhoria de Hardware:** Computadores com maior capacidade de RAM e processamento para suportar ambientes de virtualização e ferramentas de desenvolvimento mais avançadas.

c) **Documentação Técnica:** Acesso a guias e manuais detalhados sobre Laravel, ReactJS e X-Road para facilitar a resolução de problemas.

O suporte que se achou necessários foi:

a) **Orientação de Especialistas:** Consultores externos ou profissionais com experiência em X-Road poderiam auxiliar na configuração e na superação de dificuldades técnicas. Mentores em Laravel e ReactJS poderiam ajudar as equipas de backend e frontend a resolver problemas complexos.

b) **Recursos Financeiros:** Investimentos para adquirir hardware adequado e financiar capacitações técnicas para os membros do projecto.

9. Avaliação da Cadeira

A cadeira de Engenharia de Software II ofereceu uma oportunidade prática de aplicar conceitos fundamentais de desenvolvimento de sistemas em um projecto desafiador e interconectado. A experiência proporcionou uma compreensão mais profunda sobre:

I. Gestão de Projectos:

- ✓ O uso de técnicas para planeamento e divisão de tarefas, incluindo a aplicação de metodologias ágeis como Scrum, contribuiu para organizar os esforços das equipas.

- ✓ No entanto, as dificuldades na gestão de problemas técnicos evidenciaram a importância de maior treinamento em liderança e resolução de crises.

II. Trabalho em Equipe

- ✓ A divisão de responsabilidades entre subgrupos (backend, frontend, administração e cibersegurança) refletiu a dinâmica do trabalho colaborativo no setor de TI.
- ✓ A necessidade de comunicação eficiente destacou a relevância de boas práticas de integração entre equipes.

III. Tecnologias Utilizadas

- ✓ Ferramentas como Laravel, ReactJS e X-Road foram desafiadoras, mas enriqueceram o aprendizado, proporcionando competências práticas que podem ser aplicadas em contextos reais.
- ✓ As limitações técnicas enfrentadas por algumas equipes também demonstraram a importância de preparar o ambiente tecnológico antes de iniciar um projeto.

Apesar dos desafios encontrados, a cadeira incentivou os estudantes a buscar soluções criativas, trabalhar sob pressão e desenvolver habilidades que vão além da programação, incluindo gestão de recursos e colaboração interdisciplinar.

10. Conclusão

O projecto de desenvolvimento dos sistemas de gestão de estudantes e emissão de certificados, com interoperabilidade via X-Road, foi uma experiência enriquecedora. Ele permitiu aos estudantes aplicar teorias aprendidas em sala de aula em um contexto prático e desafiador, que reflete cenários reais do mercado de trabalho. Apesar das dificuldades técnicas, limitações de hardware e falta de conhecimento prévio de algumas ferramentas, as equipas demonstraram resiliência e disposição para aprender.

As principais lições aprendidas incluem:

- A importância do planeamento detalhado e da preparação técnica antes de iniciar um projecto.
- O valor da comunicação eficiente e da colaboração entre equipas multidisciplinares.
- A necessidade de investir em capacitação contínua para lidar com tecnologias emergentes.

11. Referências

Fielding, R. T. (2000). Architectural Styles and the Design of Networkbased Software Architectures.

Gottschalk, P. (2009). Maturity Model for IT Interoperability in Digital Government.

Laudon, K. C., & Laudon, J. P. (2020). Management Information Systems: Managing the Digital Firm.

Menezes, A. J., van Oorschot, P. C., & Vanstone, S. A. (1996). Handbook of Applied Cryptography.

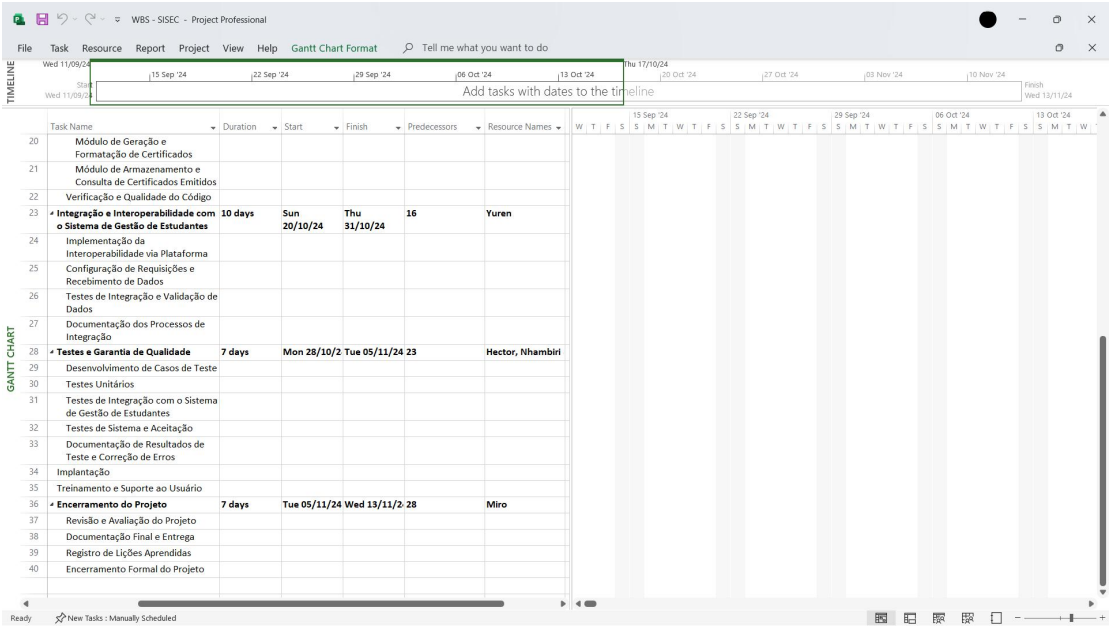
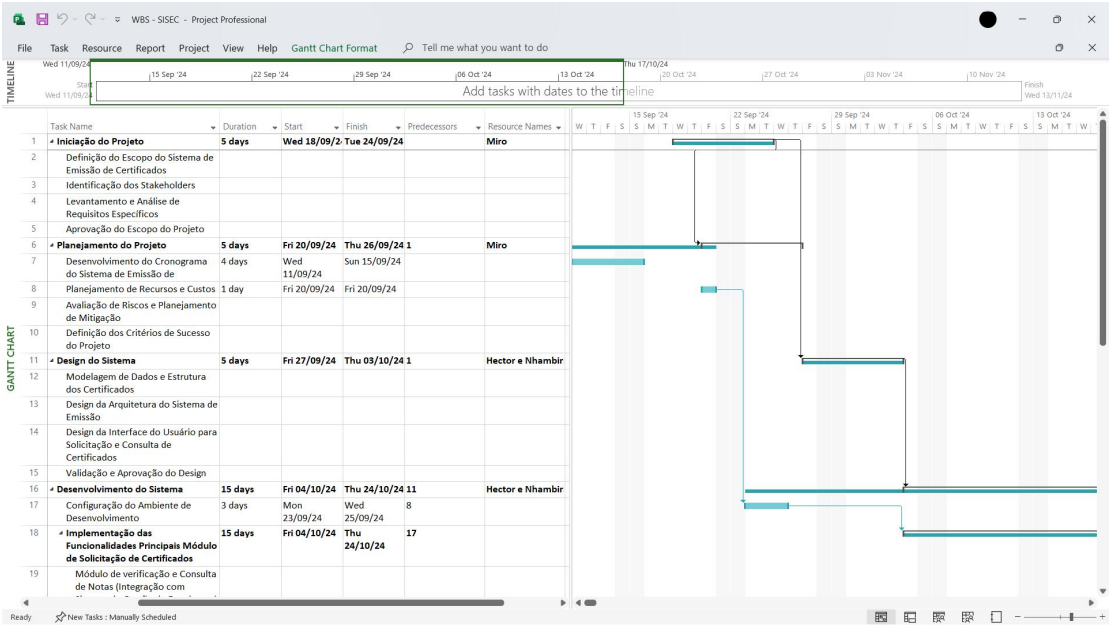
Oliveira, A. S. (2018). Digital Transformation in Education: Opportunities and Challenges.

Schwaber, K., & Sutherland, J. (2020). The Scrum Guide.

Stallings, W. (2020). Cryptography and Network Security: Principles and Practice.

APÊNDICES

WBS DO SISEC



Produto final

