Name

Scaffold

Table of Contents

[Section 1 - Identifying and Defining (15 Marks) 2](#_Toc183681920)

[1.1 — Client Scenario 2](#_Toc183681921)

[1.2 — Environmental Specification 2](#_Toc183681922)

[1.3 — List of Objectives 3](#_Toc183681923)

[1.4 — Performance Criteria 4](#_Toc183681924)

[1.5 — IPO (Input-Process-Output) 4](#_Toc183681925)

[1.6 — Gantt Chart 0](#_Toc183681926)

[Milestone 1: This should be completed by term 1, week 2. 0](#_Toc183681927)

[Section 2 - Research and Planning (10 Marks) 0](#_Toc183681928)

[2.1 — Research 0](#_Toc183681929)

[2.2 — Data Flow Diagram (DFD) 1](#_Toc183681930)

[2.3 — Interface Design (Wireframe) 0](#_Toc183681931)

[2.4 — Storyboard 1](#_Toc183681932)

[2.5 — Pseudocode 0](#_Toc183681933)

[Milestone 2: This should be completed by term 1, week 5. 1](#_Toc183681934)

[Section 3.1 - Development (10 Marks) 0](#_Toc183681935)

[3.1.2 — Logbook 0](#_Toc183681936)

[Milestone 3.1: This will be checked at various points each term. 2](#_Toc183681937)

[Section 4 - Testing and Maintaining (10 Marks) 0](#_Toc183681938)

[4.1 — Acceptance Testing 0](#_Toc183681939)

[4.2 — Load Testing 1](#_Toc183681940)

[4.3 — User Survey 2](#_Toc183681941)

# Section 1 - Identifying and Defining (15 Marks)

## 1.1 — Client Scenario

|  |
| --- |
| **Client Requirements:**   1. Content Processing and Transformation 2. Ability to upload and process various file formats (PDF, DOCX, URL, YT) 3. Multi-language support for international students 4. Study Material Generation: 5. AI summaries of uploaded material 6. AI flashcards with questions and answers 7. AI practice quizzes to revise 8. Personalization Features: 9. Customizable study schedules 10. Progress tracking and analytics 11. User Experience: 12. Modern design interface suitable for students 13. Cross-platform web app (Desktop, Mobile) 14. Accessibility features   **Problem Statement:**  Students today face many challenges in their academic journey, these include:   * Information overload from multiple sources and formats * Limited time to process and organize study materials * Difficulty maintaining focus with traditional study methods * Varying learning styles that aren't addressed by education facilities * The need for more engaging and interactive study experiences   The primary problem Notelo AI addresses is the inefficiency and one-size-fits-all approach of traditional studying methods. Many students spend countless hours trying to organize and memorize information, often using methods that don't align with their learning style. This leads to decreased motivation, poor retention, and suboptimal academic performance.  **Vision and Expected Impact:**  Notelo AI aims to help students approach studying more effectively by:   * Reducing study time by 40-50% using AI * Improving information retention through revision material * Increasing student engagement and motivation * Supporting diverse learning styles and needs * Making education more accessible to all students   The application envisions becoming an indispensable learning companion that adapts to each student's unique needs and circumstances. By leveraging AI technology, Notelo AI will transform the studying experience from a often frustrating and time-consuming process into an efficient, personalized, and engaging journey.  **Long-term impact goals include:**   * Improved academic performance across diverse student populations * Reduced academic stress and anxiety * Enhanced student confidence and self-directed learning skills * Better preparation for professional careers through improved study habits * Greater educational equity through personalized learning support   Through these features and goals, Notelo AI positions itself as more than just a study tool – it's a comprehensive learning platform that empowers students to reach their full academic potential while developing crucial lifelong learning skills. |

Notelo AI is designed to help students that face challenges when studying for school and university. The application's primary purpose is to transform various study materials (PDFs, articles, videos) into formats like summaries, flashcards, and quizzes that will help students save time studying. The target audience is primarily students with a focus on those who may struggle with traditional studying methods or staying focused.

## 1.2 — Environmental Specification

Outline the technical environment in which the application will operate. This includes hardware, software, network specifications, and other relevant environmental factors. Describe the operating systems, development tools, servers, and other infrastructure components necessary for the application.

Hardware

|  |  |  |
| --- | --- | --- |
| Image | Part | Specification |
| 🖥️ | Client Device | Any modern computer/mobile device with web browser |
| 💾 | Memory (RAM) | Minimum 4GB RAM recommended |
| 💽 | Storage | Minimum 1GB free space for caching |

Software

|  |  |  |
| --- | --- | --- |
| Image | Platform | Specification |
| 🌐 | Frontend | React.js, Next.js, TailwindCSS, Framer Motion |
| ⚙️ | Backend | Supabase, PostgreSQL, PayPal |
| 🧠 | AI Services | OpenAI API integration |
| 🔒 | Authentication | Supabase Auth |
| 🌍 | Web Browsers | Chrome, Firefox, Safari, Edge (latest versions) |

Network

|  |  |  |
| --- | --- | --- |
| Image | Part | Specification |
|  |  |  |
|  |  |  |

## 1.3 — List of Objectives

Enumerate the specific objectives that the application aims to achieve. These should be clear, measurable goals that directly address the client’s needs. The objectives could include functionality, user engagement, performance targets, and other relevant criteria.

Functional Features

|  |  |  |
| --- | --- | --- |
| Feature | Priority | Description |
|  |  |  |
|  |  |  |
|  |  |  |

Non-Functional Features

|  |  |  |
| --- | --- | --- |
| Feature | Priority | Description |
|  |  |  |
|  |  |  |
|  |  |  |

## 1.4 — Performance Criteria

Specify the application's performance criteria, including speed, reliability, scalability, and user satisfaction benchmarks. Define how performance will be measured and the application's standards.

|  |
| --- |
|  |

## 1.5 — IPO (Input-Process-Output)

Develop an IPO model that outlines the application's inputs, processes, and outputs.

|  |  |  |
| --- | --- | --- |
| Input | Process | Output |
|  |  |  |
|  |  |  |
|  |  |  |

## 1.6 — Gantt Chart

Create a Gantt chart to outline the project timeline. The chart should include all major milestones, tasks, and deadlines. It should also visually represent the project schedule, showing each task's start and end dates and the dependencies between tasks.

|  |
| --- |
|  |

### Milestone 1: This should be completed by term 1, week 2.

# Section 2 - Research and Planning (10 Marks)

## 2.1 — Research

Conduct thorough research to gather information relevant to the project. This should include market analysis, user research, competitor analysis, and other pertinent data. Summarise the findings and explain how they will inform the project’s development.

|  |
| --- |
|  |

## 2.2 — Data Flow Diagram (DFD)

Design a DFD for each feature of your project to illustrate how data will flow through the system. The DFD should show the interactions between inputs, processes, and outputs. Provide detailed diagrams and explanations for each feature.

**DFD 1**

|  |
| --- |
|  |

**DFD 2**

|  |
| --- |
|  |

**DFD 3**

|  |
| --- |
|  |

## 2.3 — Interface Design (Wireframe)

Develop wireframes for the application's user interface. These wireframes should visually represent the application's layout and design. Include annotations and explanations for each element of the interface.

|  |
| --- |
|  |

## 2.4 — Storyboard

Create a storyboard to outline the user journey through the application. The storyboard should include key screens, user actions, and interactions. Provide a narrative to explain how users will navigate the application and achieve their goals.

|  |
| --- |
|  |

## 

## 2.5 — Pseudocode

Write five pseudocode algorithms for critical functions and algorithms in the web application's backend. These pseudocode algorithms should provide a clear foundation for coding the application. Include detailed explanations and comments for each algorithm.

**Algorithm 1**

|  |
| --- |
|  |

**Algorithm 2**

|  |
| --- |
|  |

**Algorithm 3**

|  |
| --- |
|  |

**Algorithm 4**

|  |
| --- |
|  |

**Algorithm 5**

|  |
| --- |
|  |

### Milestone 2: This should be completed by term 1, week 5.

# Section 3.1 - Development (10 Marks)

## 3.1.2 — Logbook

Develop a logbook for tracking changes and implementing significant features and changes/additions to your web application. The logbook should follow the styling guide provided in the course specs. Include entries for each significant development activity, with dates, descriptions, and outcomes.

|  |  |
| --- | --- |
| Date: |  |
| Title: |  |
| Author: |  |
| Tasks Description: |  |
| Image: |  |
| Logbook Entry: | Progress since the last entry:  Task Achieved:  Stumbling blocks or issues encountered and how they were managed:  Possible approaches for upcoming tasks:  Reflective comments: |
| Resources: |  |

|  |  |
| --- | --- |
| Date: |  |
| Title: |  |
| Author: |  |
| Tasks Description: |  |
| Image: |  |
| Logbook Entry: | Progress since the last entry:  Task Achieved:  Stumbling blocks or issues encountered and how they were managed:  Possible approaches for upcoming tasks:  Reflective comments: |
| Resources: |  |

|  |  |
| --- | --- |
| Date: |  |
| Title: |  |
| Author: |  |
| Tasks Description: |  |
| Image: |  |
| Logbook Entry: | Progress since the last entry:  Task Achieved:  Stumbling blocks or issues encountered and how they were managed:  Possible approaches for upcoming tasks:  Reflective comments: |
| Resources: |  |

|  |  |
| --- | --- |
| Date: |  |
| Title: |  |
| Author: |  |
| Tasks Description: |  |
| Image: |  |
| Logbook Entry: | Progress since the last entry:  Task Achieved:  Stumbling blocks or issues encountered and how they were managed:  Possible approaches for upcoming tasks:  Reflective comments: |
| Resources: |  |

### Milestone 3.1: This will be checked at various points each term.

# Section 4 - Testing and Maintaining (10 Marks)

## 4.1 — Acceptance Testing

Apply methodologies to test and evaluate whether the application meets all outlined requirements and objectives. Describe the acceptance testing process, including test cases, scenarios, and criteria. Document the results and any necessary actions.

|  |
| --- |
|  |

## 4.2 — Load Testing

Apply methodologies to test and evaluate the application’s stability and performance under different levels of usage and load. Detail the load testing procedures, tools, and metrics. Provide a report on the findings and recommendations for improvement.

|  |
| --- |
|  |

## 4.3 — User Survey

Collect feedback through a survey to analyse and respond to feedback on the application’s usability, functionality, and overall satisfaction. Design a user survey with relevant questions and distribute it to a representative sample of users. Summarise the survey results and outline the changes made based on user feedback.

|  |
| --- |
|  |