

**كلية العلوم الحاسوبية والمعلوماتية**

Faculty of CSI

Briefly News

***Student:***

|  |  |
| --- | --- |
| Amneh Mohammad Shawish | 202220114 |
|  |  |
|  |  |

*Supervisor:*

Dr. Ala’a Abuthawabeh

*Amman - Jordan*

*2024/2025*

**Table of Contents**

Chapter One

Introduction page 3

1.1 Problem Statement page 3

1.2 Business Case page 3

1.3 Project Scope page 3

1.4 Project Goals page 3

Chapter Two

Project Planning and Requirements page 4

2.1 Project Plan and Schedule page 4

2.2 Functional Requirements page 4

2.3 Non-Functional Requirements page 4

2.4 SDLC Model page 5

Chapter Three

System Analysis and Design page 7

3.1 Use Case Diagram page 7

3.2 User Interface page 7

3.3 System Architecture MVC page 8

3.4 Class Diagram page 9

List of Figures

|  |  |  |
| --- | --- | --- |
| Figure No. | Figure Title | Page No. |
| Figure 1 | Gantt Chart of Project | 4 |
| Figure 2 | Use Case Diagram | 7 |
| Figure 3 | Login Page | 7 |
| Figure 4 | News Article with "Generate Summary" Button | 8 |
| Figure 5 | Class Diagram | 9 |

Chapter One

Introduction

The system has been facing problems due to its paper-based appointment

system. With the increase in the number of patients visiting, it has

become diﬃcult to manage the appointment system manually. Recording

of appointments and creating registers by pen and paper has become a

* 1. Problem Statement:  
     The increasing volume of news articles published daily presents a challenge for readers who want to stay informed but have limited time to read lengthy content. While news platforms provide a wealth of information, the sheer size and complexity of articles make it difficult for users to quickly grasp the key points. Manual summarization, although helpful, is inefficient and time-consuming.

To address this challenge, an automated news summarization tool is needed. The goal is to develop a system that can analyze and extract the most relevant information from lengthy news articles, providing users with brief and meaningful summaries without losing essential context. Using Drupal for content management and a Natural Language Processing (NLP) library for summarization, this system aims to enhance the user experience by offering quick access to condensed news content, helping readers stay up to date without investing significant time.

* 1. Business Case:

As digital news articles become longer, they are much harder to keep interested and find out key information quickly. This project aims to develop an automated news summarization tool using Drupal as well as an NLP library which will provide quick access to the important information that the reader is looking for. This would provide a competitive advantage to news platforms.

1.3 Project Scope:

* Platform: Built using Drupal to manage and store articles.
* Summarization Engine: An NLP library to automatically generate summaries from text-based articles.
* Content Types: If you have a news article that has a lot of text you will want to see a summary and the full article.
* Limitations: The initial release will only support English language articles and will exclude multimedia content.

1.4 Project Goals:

* Develop an NLP-based summarization engine.
* Integrate the plugin into Drupal to summarize news articles.
* Make sure summaries are appropriate and brief but contain significant information.
* Improve user experience by offering quick summaries without overwhelming users.
* Ensure scalability for future expansion.

Chapter Two

Project Planning and Requirements

2.1 Project Plan and Schedule:

* Week 1: Requirements Gathering
* Week 2: System Analysis and Planning
* Week 3: System Design
* Weeks 4-5: System Implementation
* Week 6: Testing and Debugging

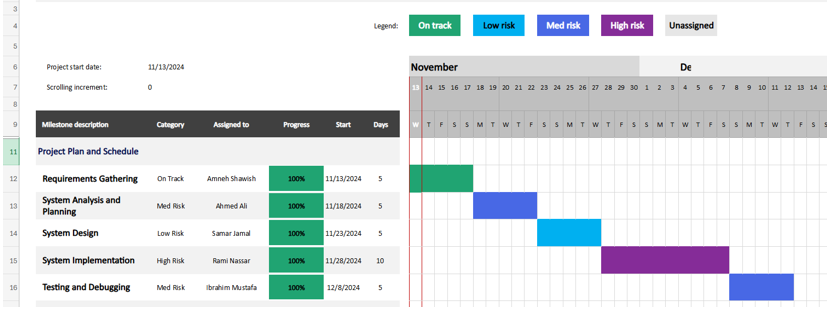


Figure 1: Gantt Chart of Project

2.2 Functional Requirements:

* FR1: The system must automatically summarize news articles using an NLP library.
* FR2: The NLP summarization should extract key points while maintaining the context and meaning.

2.3 Non-Functional Requirements:

* NFR1: Performance

The system must generate summaries within 5 seconds for articles up to 2000 words.

* NFR2: Accuracy

Summarization accuracy should maintain at least 80% relevance to the original article content, based on human evaluation.

* NFR3: Usability

The user interface should be simple and intuitive, requiring no more than 5 minutes of training for new users.

2.4 SDLC Model:

For this project, the Agile software development lifecycle (SDLC) model will be used. Agile is well-suited for this project because it allows for iterative development and frequent feedback, which is important for refining the summarization tool and user interface.

1. Planning Phase:
   * Define project scope, goals, and high-level requirements.
   * Establish the timeline and prioritize features for each sprint.
2. Design Phase:
   * Design system architecture, including the integration of the NLP library with Drupal.
   * Create wireframes for the user interface.
3. Development Phase:
   * Implement the summarization engine using the chosen NLP library.
   * Integrate the engine with Drupal and develop the user interface.
   * Develop the content types, such as news articles and summaries, and build the system’s back end.
4. Testing Phase:
   * Perform unit testing, integration testing, and user acceptance testing (UAT).
   * Ensure that the summarization works as expected and that the system is scalable and responsive.
5. Deployment Phase:
   * Deploy the system to a live environment.
   * Monitor for issues and gather user feedback for future improvements.
6. Maintenance Phase:
   * Address any bugs or performance issues post-deployment.
   * Add additional features based on user feedback, such as multi-language support.

Chapter Three

System Analysis and Design

* 1. Use Case Diagram:

The system consists of two main roles: Admin and User (Reader). Below is a list of the core use cases for each role.

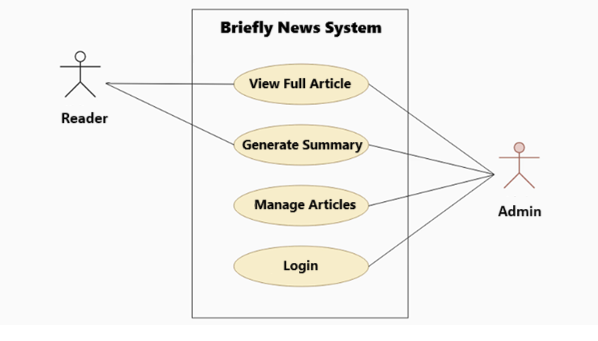


Figure 2: Use Case Diagram

* 1. User Interface:

The UI should follow a modern and clean design, making it easy for users to read the article and generate an AI summary of the content.

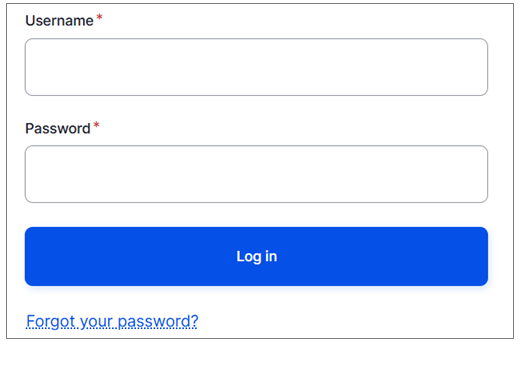


Figure 3: Login Page

A screenshot of a computer

AI-generated content may be incorrect.

Figure 4: News Article with "Generate Summary" Button

* 1. System Architecture MVC:

Drupal follows a modified MVC pattern:

* Model: Entities like Article, Summary, User, etc.
* View: Twig templates and Views module output.
* Controller: Custom modules, routes, and hooks (e.g., a custom module using an NLP library to generate summaries).

The NLP summarization logic is integrated into the Controller layer, fetching the full article from the Model (Node), summarizing it, and returning the output to the View.

* 1. Class Diagram:
* User: Represents users of the system, including both readers and administrators. It includes user credentials, roles, and methods for login/logout and role retrieval.
* Article: Encapsulates the properties of news articles such as title, content, and publication metadata. It is linked to the user who created the article and provides a method to fetch the associated summary.
* Summary: Represents the summarized version of an article, generated by an NLP library.

Relationships:

* A User can author multiple Articles, establishing a one-to-many relationship.
* Each Article has a corresponding Summary, forming a one-to-one relationship.

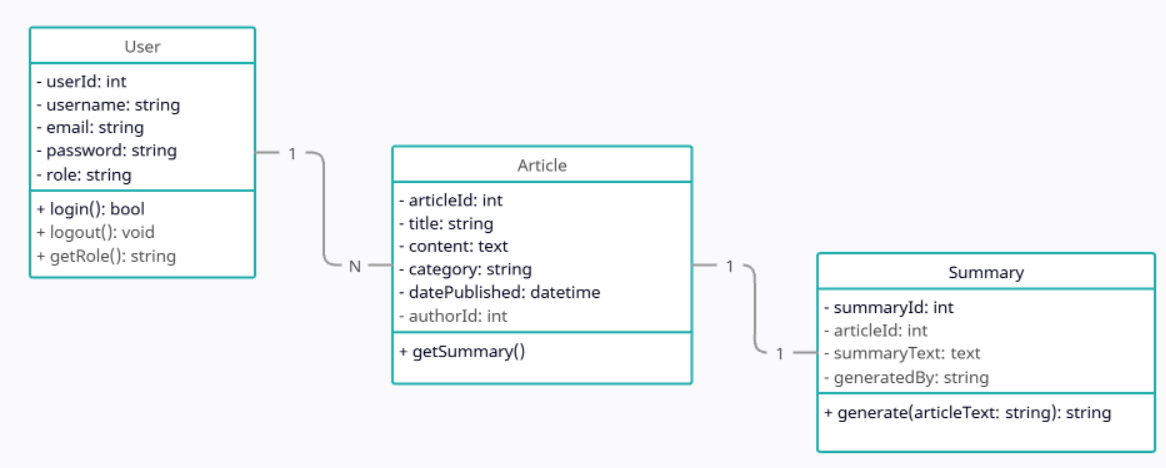


Figure 5: Class Diagram