Informatics Institute of Technology

In Collaboration With

University of Westminster, UK



*University of Westminster, Coat of Arms*

Generalized Abstractive Text Summarization Using Optimized Transformers

Software Requirements Specification (SRS)

Mr. Nazhim Kalam

w1761265 / 2019281

Supervised by

Mr. Torin Wirasingha

September 2022

This Project Proposal is submitted in partial fulfilment of the requirements for

the BSc (Hons) Computer Science degree at

the University of Westminster.

Table of Contents

[List of Figures ii](#_Toc119321981)

[List of Tables ii](#_Toc119321982)

[1. CHAPTER OVERVIEW 1](#_Toc119321983)

[2. RICH PICTURE 1](#_Toc119321984)

[3. STAKEHOLDER ANALYSIS 1](#_Toc119321985)

[3.1 Stakeholder Onion Model 2](#_Toc119321986)

[3.2 Stakeholder Viewpoints 2](#_Toc119321987)

[4. REQUIREMENT ELICITION METHODOLOGIES 4](#_Toc119321988)

[5. ANALYSIS OF DATA & PRESENTATION OF THE OUTCOME THROUGH ELICITATION METHODOLOGIES 5](#_Toc119321989)

[5.1 Literature Review 5](#_Toc119321990)

[5.2 Survey 5](#_Toc119321991)

[5.3 Prototyping 6](#_Toc119321992)

[6. SUMMARY OF FINDINGS 6](#_Toc119321993)

[7. CONTEXT DIAGRAM 6](#_Toc119321994)

[8. USE CASE DIAGRAM 6](#_Toc119321995)

[9. USE CASE DESCRIPTIONS 6](#_Toc119321996)

[10. REQUIREMENTS 8](#_Toc119321997)

[10.1 Functional Requirements 8](#_Toc119321998)

[10.2 Non-functional Requirements 9](#_Toc119321999)

[11. CHAPTER SUMMARY 9](#_Toc119322000)

[REFERENCES I](#_Toc119322001)

[APPENDIX A – CONCEPT MAP II](#_Toc119322002)

# List of Figures

[Figure 12.1 - Prototype Feature Diagram (Self-composed) 13](#_Toc117550682)

[Figure 13.1 - Gantt Chart 16](#_Toc117550683)

[Figure 13.2 - Model development flow (Self-composed) 21](#_Toc117550684)

# List of Tables

[Table 5.1 - Related work in abstractive text summarization 3](#_Toc117584436)

[Table 11.1 - Research Objectives 9](#_Toc117584437)

[Table 13.1 - Research Methodology 13](#_Toc117584438)

[Table 13.2 - Deliverables and Dates 17](#_Toc117584439)

[Table 13.3 - Risk Mitigation Plan 19](#_Toc117584440)

**Acronyms**

|  |  |
| --- | --- |
| AI | Artificial Intelligence. |
| DL | Deep Learning |
| ML | Machine Learning |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

# CHAPTER OVERVIEW

# RICH PICTURE

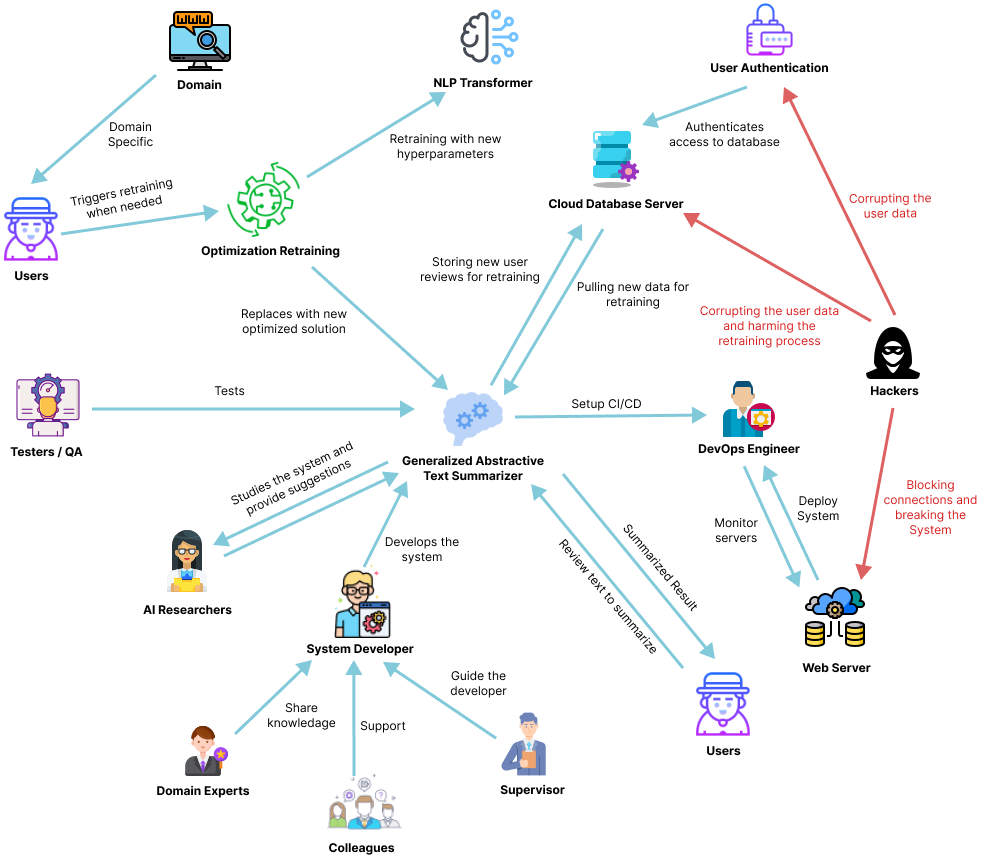
****

Figure 2.1 – Rich Picture Diagram (*self-composed*)

# STAKEHOLDER ANALYSIS

## 3.1 Stakeholder Onion Model

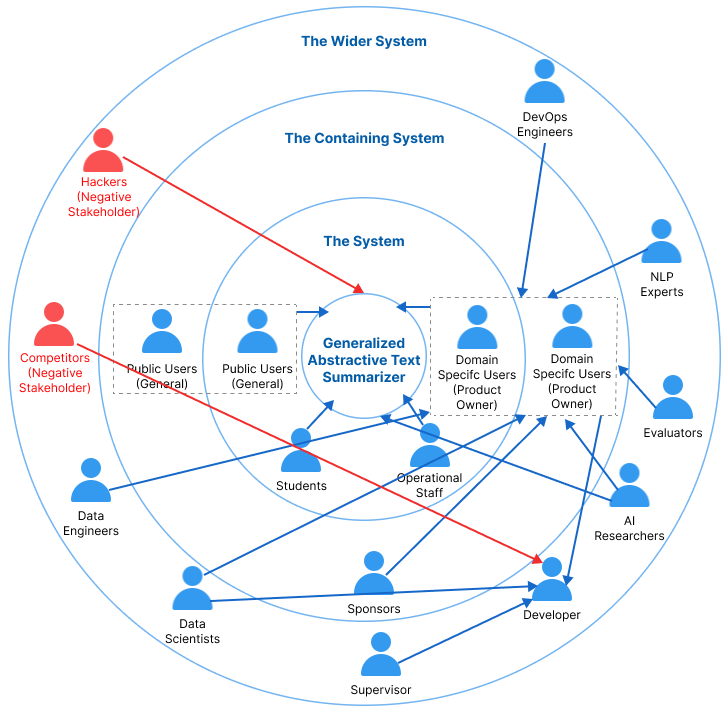


Figure 3.1 – Stakeholder Onion Model (*self-composed*)

## 3.2 Stakeholder Viewpoints

|  |  |  |
| --- | --- | --- |
| **Stakeholder** | **Role** | **Benefits/ Role Description** |
| Developer | Financial Contributor | Works on developing the system |
| Sponsors | Funds to market the system and allows developers to advance the system with time. |
| Data Scientists | Quality Control Regulator | Provides performance enhancements for the models and algorithms used in data science. |
| Data Engineers | Gives guidance on potential data that may be used to generate the best suggestions possible. |
| AI Researchers | Conduct research in the specified area to enhance and implement reliable text summarizing models. |
| NLP Experts | Offers specialized guidance and insights on the field  knowledge, to enhance the functionality of the system. |
| Students | Functional beneficiary | Understanding how the generalization works along with hyperparameter retraining with respect to the domain. |
| Domain Specific Users | System Owner and Operational | Inputs the text reviews for abstractive summarization and handles the need for model retraining when need to improve performance with the previously used inputs as new data for the model. |
| Public Users | General Usage | General public users (Not Domain Specific) will be using the general abstractive summarization model without any specific model assigned to them and hyperparameter retraining unless they want to. |
| Operational Staff | Operations support | Ensures that the system is up and functioning while responding to user requests and problems. |
| DevOps Engineers | Product Deployment & Maintenance | Makes ensuring the system is up and running in the cloud and is serving users without being throttled |
| Hackers | Negative Stakeholder | May manipulate the review data stored in the database which will affect the retraining process. |
| Competitors | May build competing systems that may outperform the existing system. |
| Evaluators | Quality Inspector | Checks to see if the system is ready for production use and puts it through its paces. |

# REQUIREMENT ELICITION METHODOLOGIES

There were several requirement elicitation approaches used to collect needs for the creation of the research project. The approaches selected for this were literature review, survey, and prototype. The following is a discussion of the rationales behind selecting the mentioned requirement elicitation approaches.

|  |  |
| --- | --- |
| **Method** | **Description** |
| Literature Review | To determine research gaps in the chosen domain of interest and the intended topic of study at the project's outset, the author conducted a thorough literature analysis. Current systems were researched together with comparable technologies that might be applied to the existing systems that were referenced in literature in order to discover research gaps available in technologies that can be used. |
| Survey | A questionnaire was utilized as a survey instrument to obtain requirements and opinions from possible users of the suggested system. The author will benefit from this sort of poll in understanding people's thought processes and expectations for the prototype. It will also enable the author to explain whether or not the targeted users will benefit from the suggested solution. |
| Prototyping | The project was chosen to follow the Agile Software Development Life-cycle, thus prototyping would allow the author to test and evaluate the prototype while iteratively trying out several alternative implementations to find any potential areas for improvement. |

# ANALYSIS OF DATA & PRESENTATION OF THE OUTCOME THROUGH ELICITATION METHODOLOGIES

The data analysis from the requirement elicitation methods that were selected are shown below.

## 5.1 Literature Review

Table 5.1 - Literature review findings

|  |  |
| --- | --- |
| **Finding** | **Citation** |
|  |  |
|  |  |
|  |  |
|  |  |

## 5.2 Survey

Table 5.2 – Questionnaire Analysis

|  |  |
| --- | --- |
| **Question** | **How will you decide which NFT to purchase?** |
| **Aim of question** |  |
| **Findings & Conclusion** | |
| **Aim of question** |  |
| **Findings & Conclusion** | |
| **Aim of question** |  |
| **Findings & Conclusion** | |

## 5.3 Prototyping

# SUMMARY OF FINDINGS

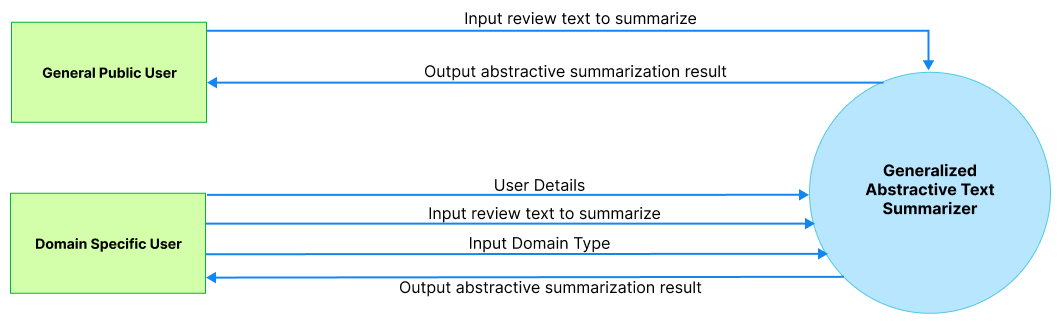
Table 5.3 – Summary of Findings

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Id** | **Finding** | **Literature Review** | **Survey** | **Prototyping** |
| 1 |  |  |  |  |
| 2 |  |  |  |  |
| 3 |  |  |  |  |

# CONTEXT DIAGRAM

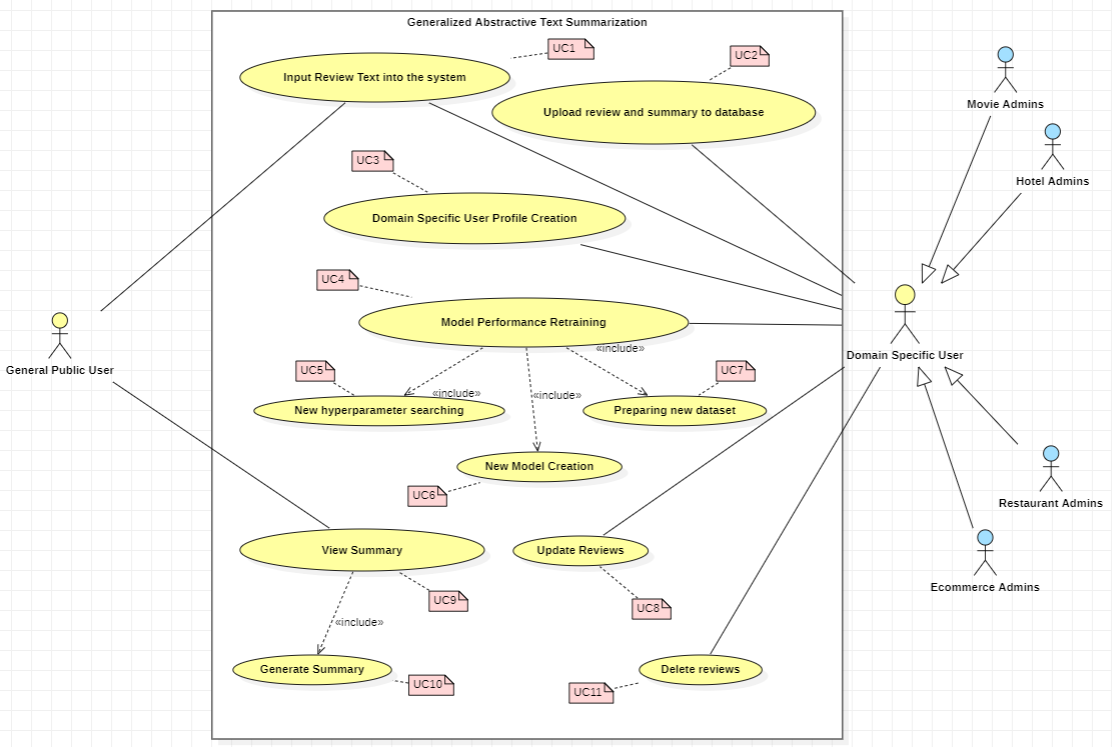
The boundaries and interactions of the system should be established before development. The graphic below shows how the system is situated.

Figure 7.1 – Context Diagram (self-composed)



# USE CASE DIAGRAM

Figure 8.1 – Use case Diagram (self-composed)



# USE CASE DESCRIPTIONS

Table 9.1 – Use case description UC:0X

|  |  |
| --- | --- |
| Use Case |  |
| Id |  |
| Description |  |
| Primary Actor |  |
| Supporting Actors (if any) |  |
| Stakeholders and Interests (if any) |  |
| Pre-Conditions |  |
| Post Conditions |  |
| Trigger |  |
| Main Success Scenario |  |
| Variations |  |

Table 9.2 – Use case description UC:0X

|  |  |
| --- | --- |
| Use Case |  |
| Id |  |
| Description |  |
| Primary Actor |  |
| Supporting Actors (if any) |  |
| Stakeholders and Interests (if any) |  |
| Pre-Conditions |  |
| Post Conditions |  |
| Trigger |  |
| Main Success Scenario |  |
| Variations |  |

# REQUIREMENTS

## 10.1 Functional Requirements

Based on the significance of the system demands, the MoSCoW approach was utilized to identify their priority levels.

Table 10.1 – Priority Levels

|  |  |
| --- | --- |
| **Priority Level** | **Description** |
| Must have (M) | The demand at this level is the fundamental functional requirement for a prototype, and it must be carried out. |
| Should have (S) | Although not strictly required for the anticipated prototype to function, important criteria do provide a lot of value. |
| Could have (C) | Optional, non-essential desirable needs are crucial to the project's scope. |
| Will not have (W) | Requirements that the system might not meet right now and that are not given first consideration. |

Table 10.2 – Functional requirements

|  |  |  |  |
| --- | --- | --- | --- |
| **FR ID** | **Requirement** | **Priority Level** | **Use Case** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

## 10.2 Non-functional Requirements

Table 10.3 – Non-functional requirements

|  |  |  |  |
| --- | --- | --- | --- |
| **NFR ID** | **Requirement** | **Priority Level** | **Use Case** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# CHAPTER SUMMARY

# REFERENCES

# APPENDIX A – CONCEPT MAP