Software Requirements Engineering

Deliverable-2

Contents

Abstract

RequirementSage is an advanced chatbot that utilizes state-of-the-art Natural Language Processing (NLP) and Machine Learning (ML) technologies to revolutionize the software requirements-gathering process. This intelligent chatbot engages in natural, conversational interactions with stakeholders, extracting comprehensive software requirements to streamline the process, particularly for small-scale projects.

Vision Statement

The vision for RequirementSage is to become a cutting-edge solution for software requirements gathering, providing an efficient and user-friendly experience for stakeholders involved in software development. By leveraging NLP and ML, RequirementSage aims to:

- **Automate Requirements Gathering**: Eliminate manual and time-consuming processes by automating the collection of software requirements.
- **Improve Communication**: Facilitate clear and effective communication between end users and software developers through natural language interactions.
- **Enhance Accuracy**: Utilize ML to enhance requirement understanding and reduce the risk of miscommunication or incomplete requirements.
- **Support Small-Scale Projects**: Tailor the chatbot's capabilities to cater specifically to small-scale projects, enabling cost-effective and efficient development.
- **Learn and Adapt**: Continuously learn from interactions to provide more accurate and relevant requirements gathering over time.
- **Ensure Data Security**: Implement robust security measures to protect sensitive project information and maintain compliance with data privacy regulations.

Summary of Project Proposal

Scope of the Project:

1. Business Goal and Objectives: Aims to automate requirements gathering, reduce effort, summarize requirements, and enable real-time documentation for small-scale projects with a focus on e-commerce websites.

2. Project's Goal and Objectives: It focuses on the chatbot's language understanding, context recognition, user intent identification, and entity categorization specifically tailored for e-commerce scenarios.

3. Project Assumptions: Assumes reliable internet connectivity and prioritizes chatbot design and performance, specifically optimized for interactions related to e-commerce.

4. Project Deliverables: Includes user instructions, predefined scripts, and continuous validation of chatbot functionality for small-scale e-commerce projects.

5. Limitations and Constraints: Limited to small-scale e-commerce project requirements gathering due to complexity, ensuring the chatbot is specialized for the unique needs of e-commerce websites.

Main Features:

1. User Account Management: Allows users to create and manage accounts for tracking preferences and feedback.
2. Requirements Gathering via Chat: A core feature enabling users to discuss and specify software requirements
3. Project Proposal Generation: Automatically generates project proposals based on the gathered requirements.
4. Requirements Listing: Compiles a list of identified software requirements.
5. Creation of Software Requirement Document (SRD): Documents software requirements in a structured manner.
6. User Feedback Collection: Gathers user feedback for continuous improvement.

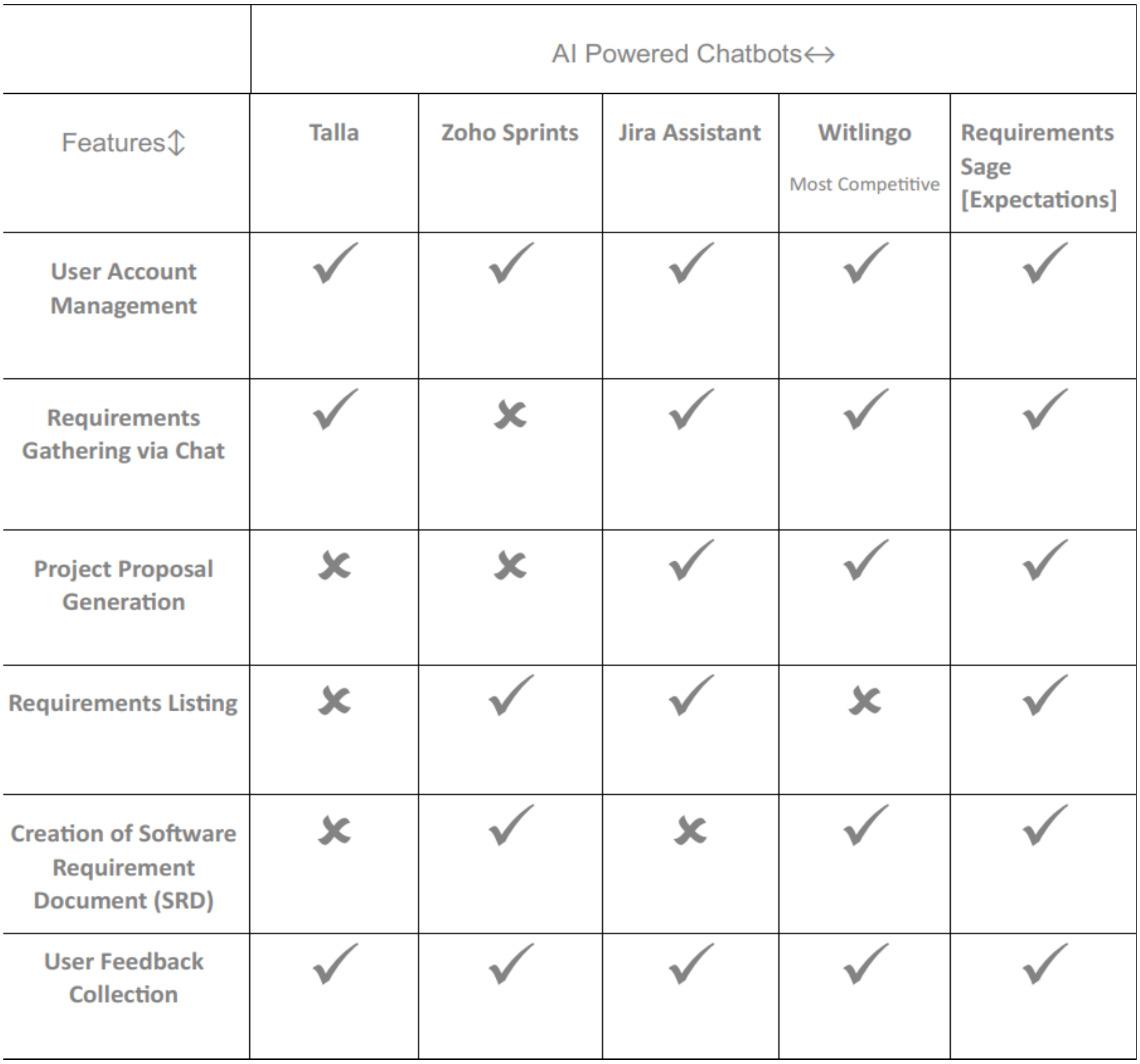
The project proposal outlined the ambitious goal of automating requirements gathering using an intelligent chatbot. It identifies key features, objectives, assumptions, constraints, and feasibility considerations, providing a comprehensive framework for the development of the "RequirementSage" chatbot.

Competitor Analysis:

The chatbots or frameworks that are already available in the market that can be used for requirements gathering are as follows:

1. **Talla:** Talla is an AI-powered chatbot that can assist in software requirements gathering by engaging with stakeholders, asking relevant questions, and documenting the gathered information.
2. **Zoho Sprints:** Zoho Sprints offers a chatbot feature that can help gather requirements and manage agile software development projects.
3. **Jira Assistant:** Some third-party plugins for Atlassian Jira, like "Jira Assistant," provide chatbot functionality for gathering requirements within the Jira environment.
4. **Witlingo:** Witlingo is a platform for building voice and chatbot applications. It can be customized to gather software requirements through voice or text-based interactions.
5. **Dialogflow:** Dialogflow by Google allows you to create conversational agents (chatbots) that can be used for requirements gathering through natural language conversations.

6. **Microsoft Bot Framework:** This framework allows users to build chatbots for various purposes, including software requirements gathering, using Microsoft technologies.
7. **Botpress:** Botpress is an open-source chatbot platform that can be customized to create chatbots for requirements gathering and other tasks.
8. **Twilio Autopilot:** Twilio Autopilot is a platform for building chatbots and virtual assistants that can be used to interact with stakeholders and gather software requirements.
9. **Conversational UI Frameworks:** You can also build custom chatbots for requirements gathering using frameworks like Rasa, Bot Framework, or custom development using Python, Node.js, or other programming languages.



	AI Powered Chatbots↔				
Features↕	Talla	Zoho Sprints	Jira Assistant	Witlingo Most Competitive	Requirements Sage [Expectations]
User Account Management	✓	✓	✓	✓	✓
Requirements Gathering via Chat	✓	✗	✓	✓	✓
Project Proposal Generation	✗	✗	✓	✓	✓
Requirements Listing	✗	✓	✓	✗	✓
Creation of Software Requirement Document (SRD)	✗	✓	✗	✓	✓
User Feedback Collection	✓	✓	✓	✓	✓

For detailed review consult the deliverable-1.

Requirements Elicitation Techniques

1- Interview

Interviewee Information:

Name: Syed Uzair Shah

Role: Co-founder & Chief Product Owner at **Repair Desk**

Interviewer Information:

Name: Syed Huzaifa Mustansar

Role: Student of Software Engineering (3rd Semester) at FAST-NUCES and team leader for the project **Requirements Sage Chatbot**.

Purpose of the Interview:

This interview aims to understand the requirements engineering process and gather insights from Uzair, the Chief Product Owner, to define the essential features and scope for a chatbot that automates requirements engineering in the software industry.

Date: September 13, 2023.

1. **Uzair:** What's the idea of our project?

   Huzaifa: Build a chatbot using NLP to gather software requirements through two-way communication.

2. **Uzair:** How will the chatbot work?

   Huzaifa: Use NLP to process user responses, extract requirements, and narrow down through two-way communication.

3. **Uzair:** You will train the model with APIs or datasets?

   Huzaifa: Two options: APIs like GPT-3 or train on a dataset related to software requirements.

4. **Uzair:** Is it be One-way or two-way communication?

   Huzaifa: Two-way. Chatbot refines user-defined system requirements into specific, narrowed-down requirements.

5. **Uzair:** Chatbot response using NLP or APIs?

   Huzaifa: Similar to ChatGPT, using NLP specifically for requirements generation.

6. **Uzair:** Huzaifa! Requirements engineering isn't a piece of cake. There are multiple aspects involved in the requirements gathering. The biggest hurdle you'd face is that all project's requirements are different. When it comes to big projects, the requirements become even more complex. How will you tackle it?

   Huzaifa: Well yeah, you're right! This is the biggest concern we have, and my team is thinking about it. The basic algorithm that the chatbot would follow is that once it gets the user prompt, it will extract the information related to software requirements, and then based on its training it will form requirements.

7. **Uzair:** Again. This is not an easy task to automate such a complex process with the help of a chatbot. There is also data limitation to train the algorithm.

   Huzaifa: Well yeah, you're right, so how can we solve this problem? What would be the changes we need to incorporate in our chatbot to make it useful? What's your opinion about it since you're an industry expert?

8. **Uzair:** Huzaifa! You need to limit the scope of your project. Since it is just the start, limit your scope to very small-scale projects or small modules of projects. Try to make it for projects that are very small-scale, and the idea of their requirements is somewhat already known.

   Huzaifa: Yeah, that is exactly what I was discussing with my course instructor. We shall limit and define the scope of our project clearly. It should be just for the very small-scale projects, for example, a simple E-commerce store.

9. **Uzair:** Yeah! In this way, you can make it more effective and get an idea about how it would help the stakeholders to generate requirements.

Interview Outcomes:

The interview conducted with Uzair helped the team to identify the scope of the project and further elicitate some necessary requirements which are as follows:

1. The chatbot shall be just for small-scale projects
2. The target audience shall be small business owners or developers of small-scale projects.
3. The chatbot shall analyze the user prompts in real time and give responses based on its training on the prior data related to the software requirements.

2- Survey

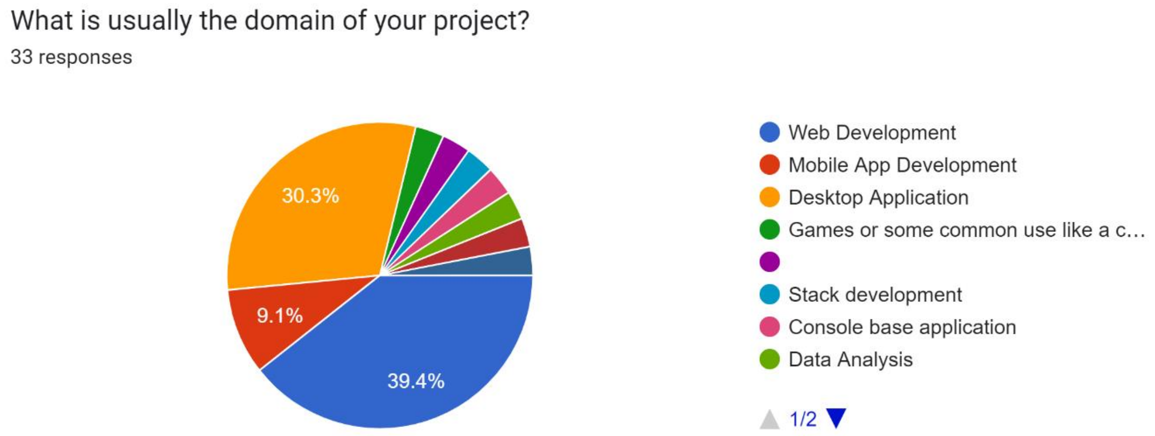
A thorough survey of our student body was conducted to gain important insights into the difficulties they run into when organizing and gathering information for their projects. This extensive study sought to comprehend the challenges that students encountered during the initial stages of project planning and data collection. By actively engaging with our student body, we aimed to pinpoint common problems and areas where students might need assistance or creative fixes. This survey acts as a crucial starting point for the creation of tools and strategies that can help students successfully negotiate the challenges of project planning and data collection. Link to survey: https://docs.google.com/forms/d/e/1FAIpQLSePxrHxzscQl6u77DRNnZINzgHBO-Aebc873WjDrw89dR4rqw/viewform?usp=sf_link

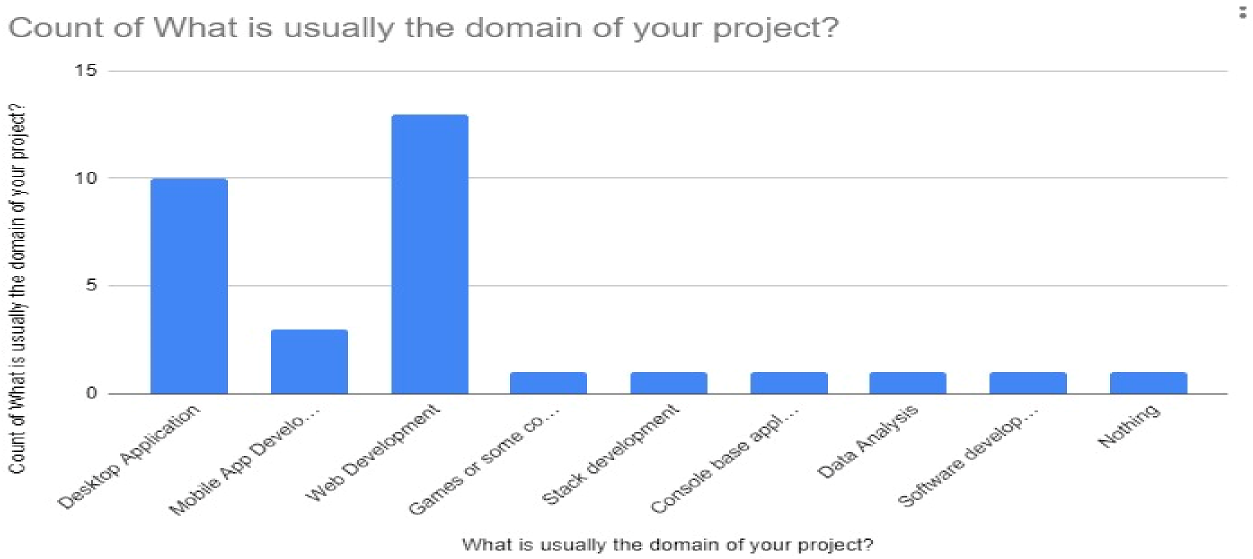
Respondent Information

Number of users Responded	36
Avg Age	18-21
Education level	Bachelors
Discipline	Software Engineering
Region	Lahore

Here’s a summary of each question in the context of the project for the "Requirement Sage Chatbot," which aims to streamline software requirements gathering through innovative NLP and ML techniques:

Question 1: What is usually the domain of your project?

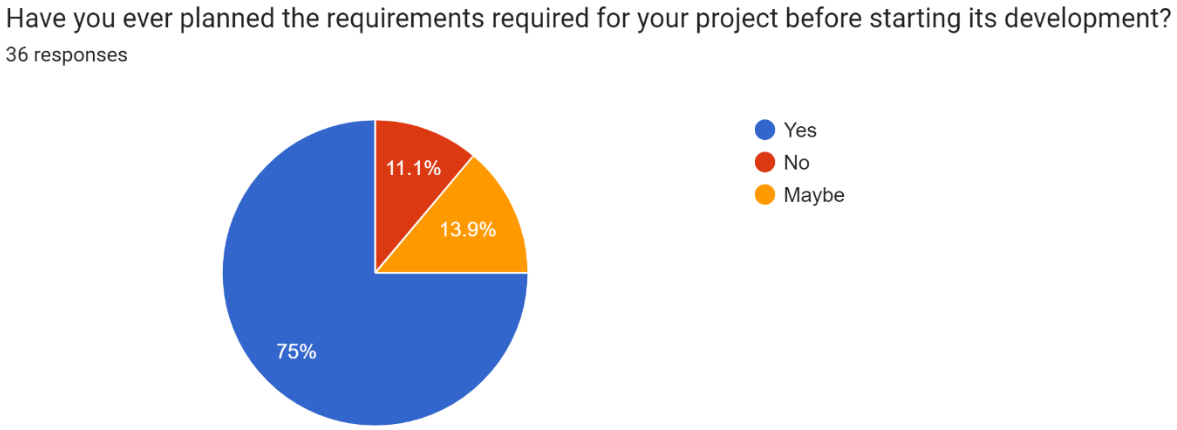


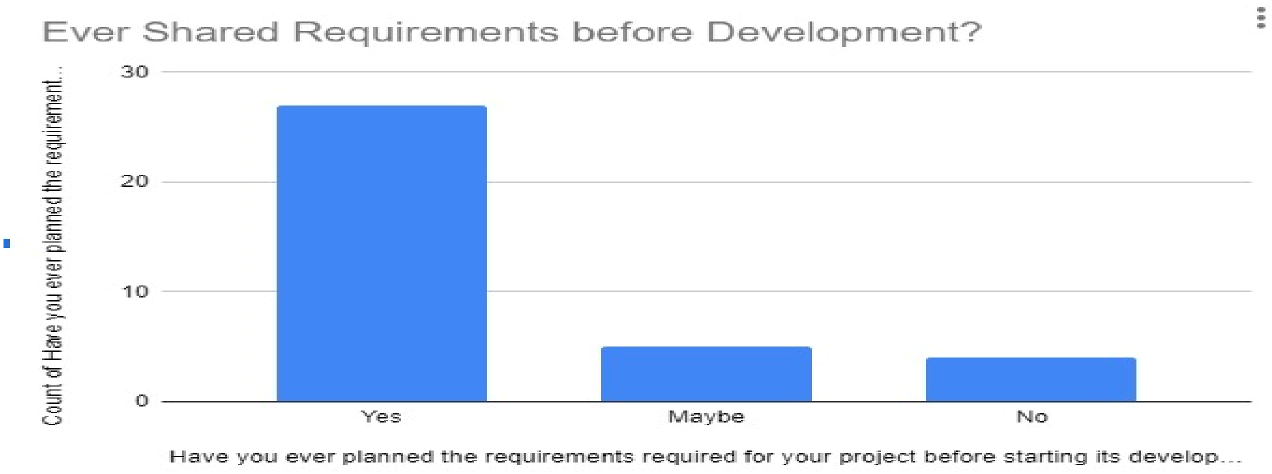


Most Dominant User Class: Web Development (39.4%)

Analysis: Web development is the most dominant domain among the respondents, indicating a strong presence of web developers in the survey.

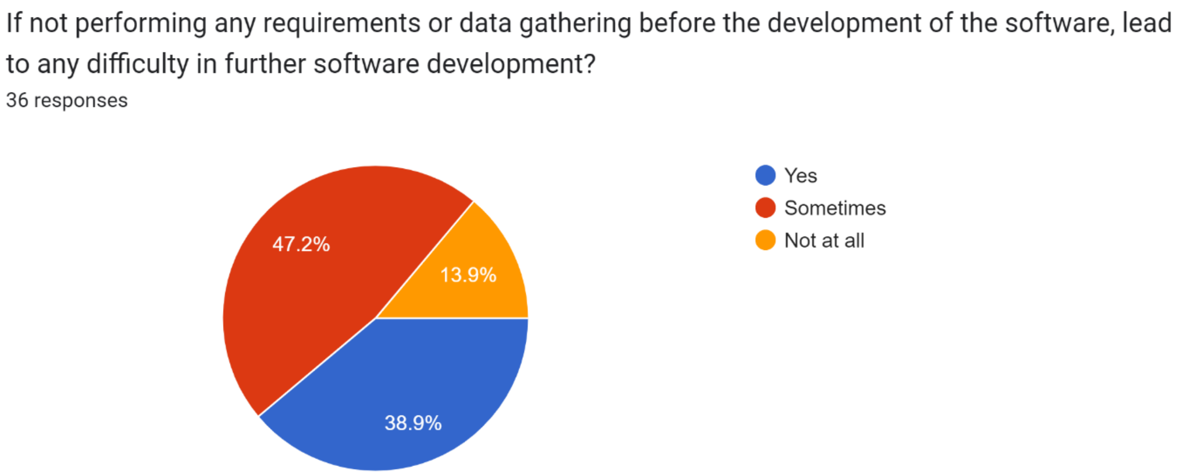
Question 2: Have you ever planned the requirements required for your project before starting its development?





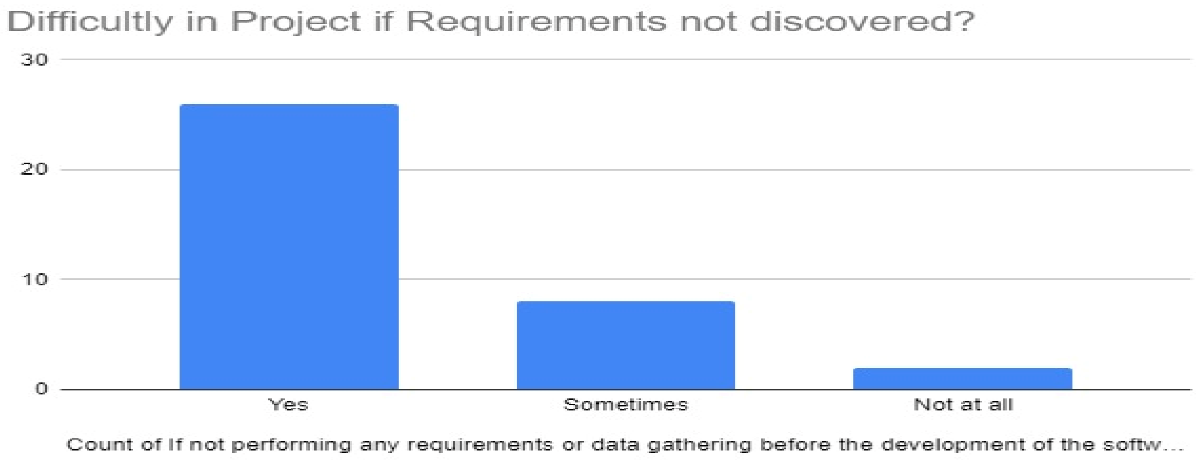
Most Dominant User Class: Yes (75%)
Analysis: Most respondents have planned project requirements before starting development, emphasizing the importance of project preparation in their workflow.

Question 3: If not performing any requirements or data gathering before the development of the software, lead to any difficulty in further software development?



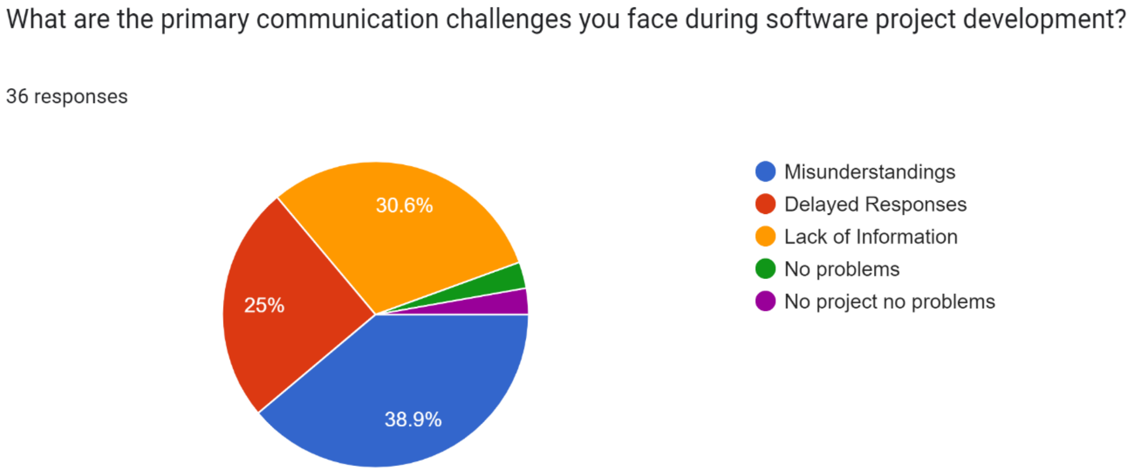
Most Dominant User Class: Sometimes (47.2%)
Analysis: Many respondents indicated that not gathering requirements or data before development can sometimes lead to difficulties, including resource issues, scope misalignment, and code rework.

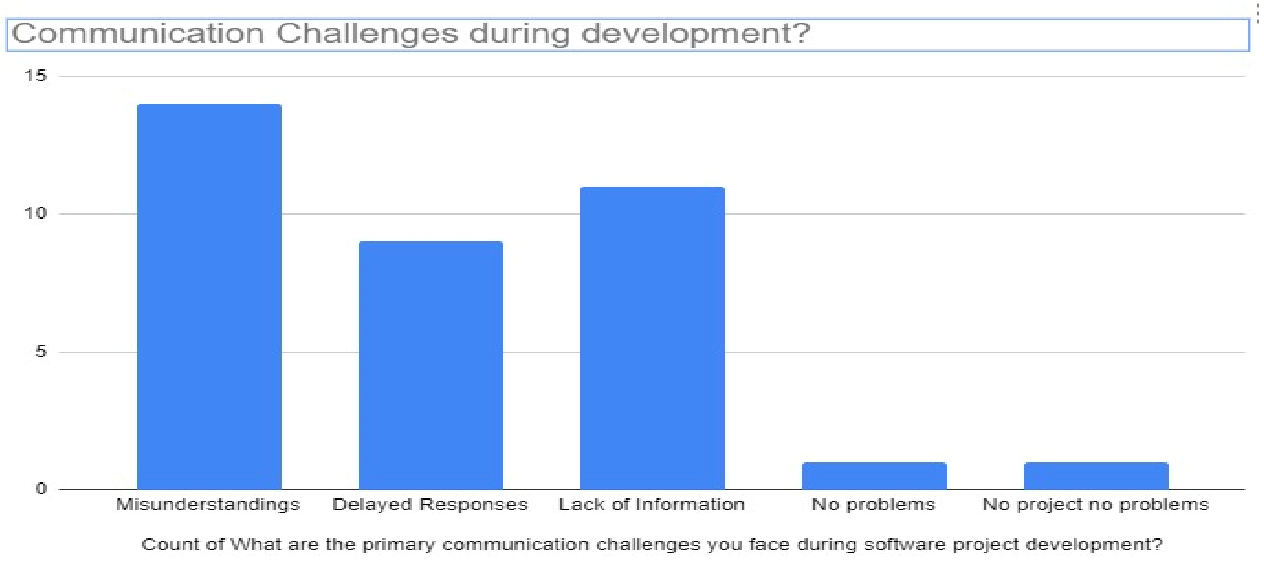
Question 4: If yes, what was the difficulty that you faced?



Most Dominant User Class: Hurdles, scope issues, and code problems were common difficulties mentioned.
Analysis: Respondents cited various challenges, including resource unavailability, scope clarity, code rework, and time management when requirements were not documented.

Question 5: What are the primary communication challenges you face during software project development?



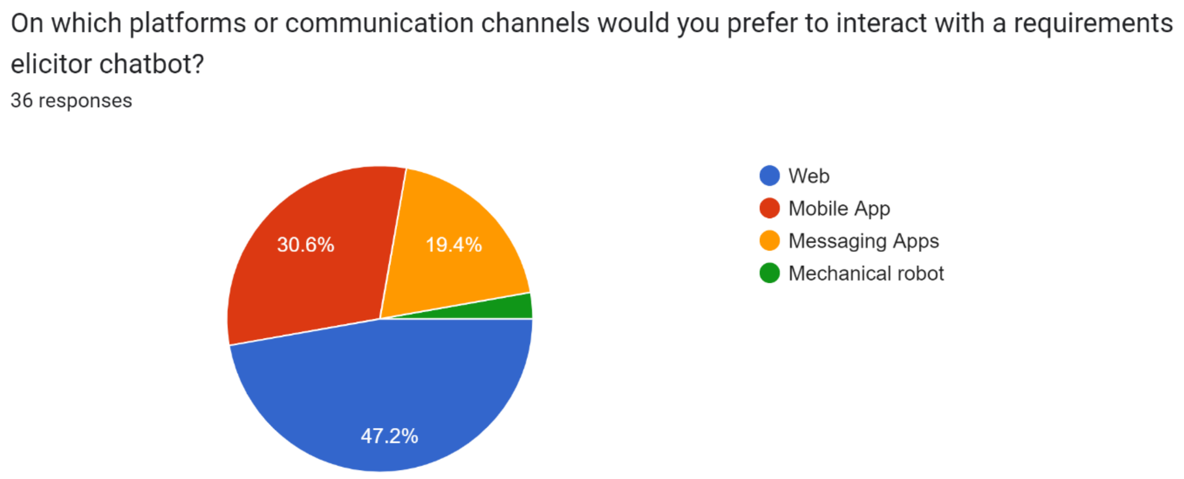


Most Dominant User Class: Misunderstandings (38.9%)
Analysis: Misunderstandings were the most common communication challenge mentioned, followed by delayed responses and a lack of information.

Question 6: If yes, What techniques or methods have you used for it? For example, Interviews, Surveys, etc.

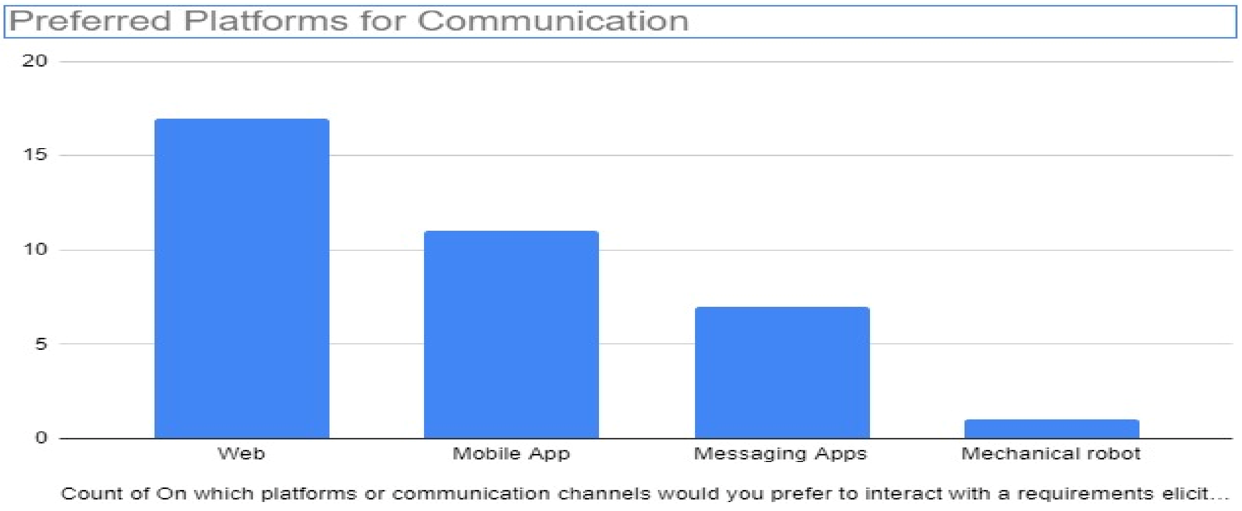
Most Dominant User Class: Interviews and Surveys were the most commonly used techniques.
Analysis: Respondents used a variety of techniques to address communication challenges, including workshops, interviews, surveys, research, and discussions.

Question 7: On which platforms or communication channels would you prefer to interact with a requirements elicitor chatbot?



On which platforms or communication channels would you prefer to interact with a requirements elicitor chatbot?

36 responses



Most Dominant User Class: Web (47.2%)
Analysis: The web was the preferred platform for interacting with chatbots, followed by mobile apps and web interfaces.

Question 8: What features or functionalities do you believe would be essential for a requirements discovery chatbot?

Most Dominant User Class: Quick response time, accurate answers, and understanding vague sentences were commonly mentioned.
Analysis: Respondents emphasized the importance of chatbots providing quick, accurate, and human-like responses, along with the ability to understand ambiguous statements and market trends.

Question 9: Are there any specific tools, systems, or platforms you would like the chatbot to integrate with?

Most Dominant User Class: Integration with tools like calendars, GitHub, and code editors was mentioned.
Analysis: Some respondents expressed interest in integrating chatbots with various tools and platforms to enhance their development process.

Survey Outcomes

In summary, the survey questions offered insightful information about the needs, difficulties, and preferences of those involved in the software development process. These observations directly influence the "Requirement Sage Chatbot's" capabilities and design, ensuring that it effectively addresses important facets of software requirements gathering and project communication.

3- Brainstorming

Brainstorming Session 1ST October 2023.

Objective:

Generate creative ideas to help us discover business and user requirements through our Brainstorming Session.

Participants

- Syed Huzaifa Mustansar (Team Lead)
- Ahmad Shamail Butt
- Anas Khan
- Faizan Saleh

Review of Existing Requirements

To Plan our strategy at the foremost we must analyze what we have achieved so far. We have cleared our product's scope, finalized our initial features, and conducted a thorough feasibility study.

Brainstorming Session

1. How might we enhance data security and protect customer/user privacy?
Concluded: By enhancing security on our user account management feature.

2. What are the most significant challenges or pain points that developers in software houses currently face?
Concluded: Huge utilization of time and cost in long requirements elicitation phases.

3. What is a creative way to connect with our customers/users to gather feedback and insights?
Concluded: By a rating system.

4. How can we improve the customer/user experience to make it more seamless and enjoyable?
Concluded: Enhance the feedback feature.

5. How can we improve communication and collaboration among our users?
Concluded: By inserting emotions into the NLP API.

Requirements Discovered through Brainstorming

User Requirements:

- Enhance Feedback
- Create Collaborative Environment

Business Requirement:

- Develop A Rating System in Feedback

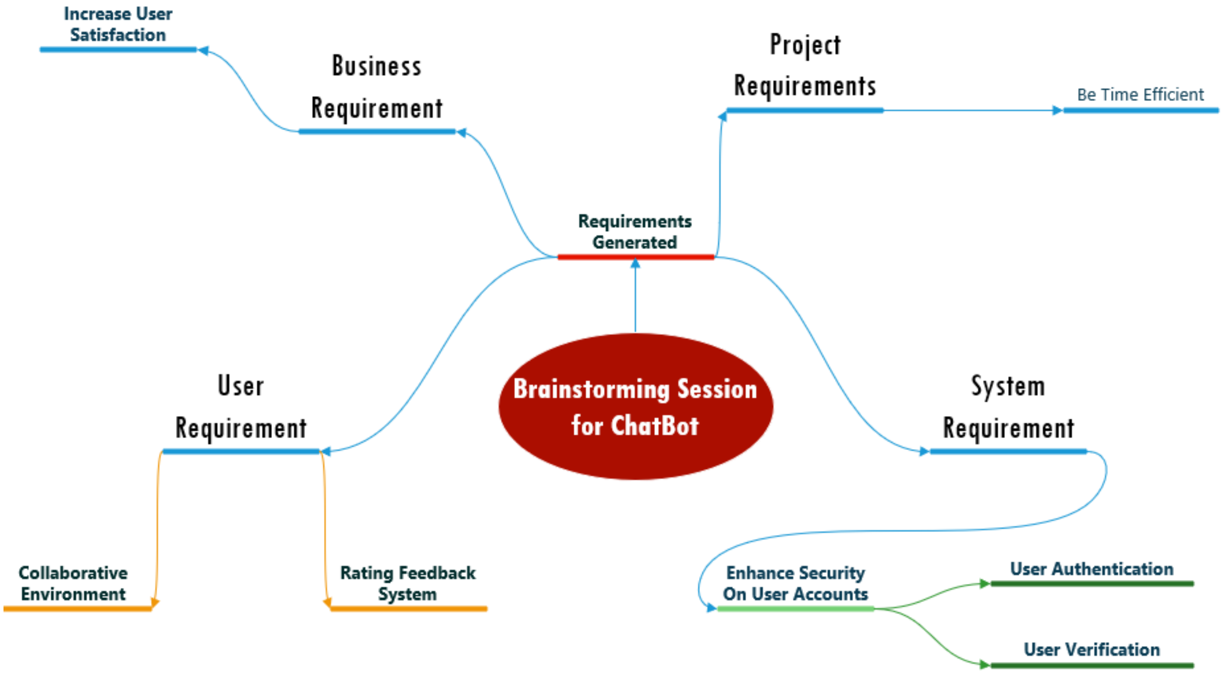
Project Requirements:

- Be Time Efficient

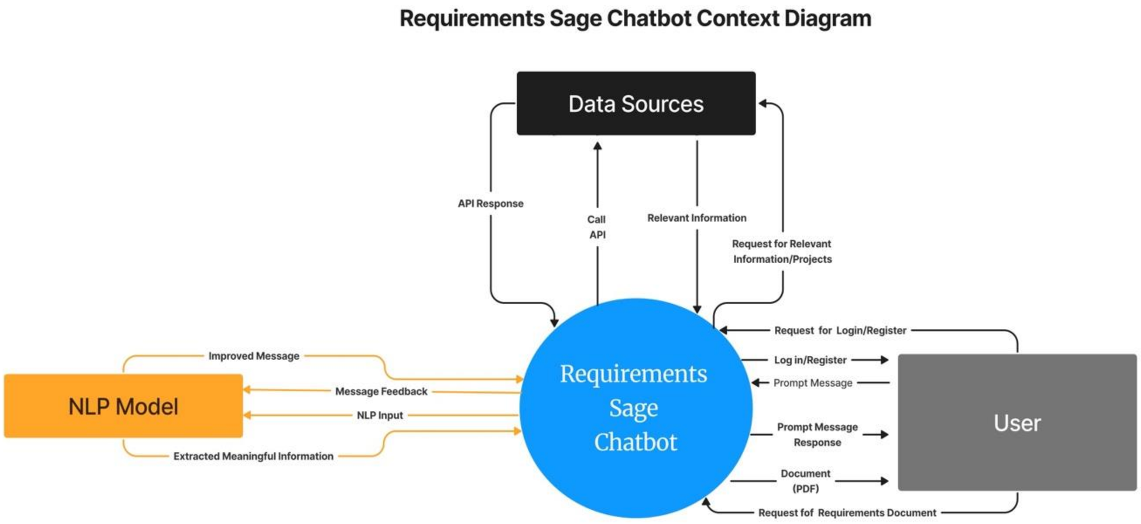
System Requirement:

- Enhance Security

Documentation Diagram for Discovered Requirements



4- Context Diagram


Requirements Sage Chatbot Context Diagram
Data Sources
API Response
Call API
Relevant Information
Request for Relevant Information/Projects
Improved Message
NLP Model
Message Feedback
NLP Input
Extracted Meaningful Information
Requirements Sage Chatbot
Request for Login/Register
Log in/Register
Prompt Message
Prompt Message Response
Document (PDF)
Request fof Requirements Document
User

5- Personas

Primary Persona

Name: Sarah

Age: 35 years

Role: Software Project Manager

Experience: 10 years

Background: Sarah is a 35-year-old software project manager with 10 years of experience in the industry. She works for a mid-sized software development company that specializes in custom solutions for clients in various domains.

Goals and Needs:

- Sarah needs to efficiently gather and document requirements from clients, ensuring that they are clear, complete, and aligned with the project objectives.
- She wants to streamline the requirements elicitation process, reduce misunderstandings between stakeholders, and minimize scope changes during the development phase.

- Sarah values tools and methods that save her time, as she often juggles multiple projects simultaneously.

Challenges:

- Balancing the needs and expectations of diverse clients can be challenging.
- Managing and prioritizing requirements from various stakeholders can become overwhelming.
- Ensuring that the requirements are well-documented and can be easily communicated to the development team is crucial.

Expectations from the Chatbot:

- Sarah expects the chatbot to help her structure and clarify requirements through natural language conversations.
- She anticipates that the chatbot can assist in identifying potential conflicts or ambiguities in requirements and offer suggestions to resolve them.
- The chatbot should be user-friendly and integrate seamlessly with her existing project management tools.

Secondary Persona

Name: Alex

Age: 27 years

Role: Junior Software Developer working for a small-scale startup

Background: Alex is a 27-year-old junior software developer working for a startup that develops mobile applications. Alex is relatively new to the industry and has been in the role for about two years.

Goals and Needs:

- Alex needs clear and well-documented requirements to ensure that they can be effectively translated into code.
- They want to better understand client expectations and project goals to contribute meaningfully to the development process.
- Alex values learning opportunities and tools that can help them gain expertise in software development.

Challenges:

- Grasping complex requirements and translating them into practical code can be challenging for a junior developer.
- Communication with clients and project managers to clarify requirements is sometimes intimidating.
- Keeping up with the latest industry best practices and standards can be overwhelming.

Expectations from the Chatbot:

- Alex expects the chatbot to provide explanations and examples related to requirements, helping them comprehend the context and significance.
- They anticipate that the chatbot can assist in generating code snippets or suggesting coding patterns based on requirements.
- The chatbot should be supportive and encourage their professional development by providing resources and tips related to software development.

6- User Scenario for the conversation with RequirementsSage chatbot for generating requirements for an E-commerce website

The user opens the chatbot app and initiates a conversation. The chatbot app welcomes the user and informs them that it is here to assist with gathering project requirements and generating a requirement document. If a user asks for something related to other than requirements discovery, the chatbot responds negatively with “context not available.” Also, if the user prompt is too long the chatbot shall display prompt too long and ask for input again. Other than that, if the user asks that he needs help gathering requirements for a new website project. The chatbot app acknowledges the user's request for assistance with a website project and asks for more details. It prompts the user to specify the type of website, target audience, desired features, and any specific design preferences. The user responds by saying that it is an e-commerce website for fashion products, targeting young adults and they want a clean and modern design. The chatbot app records the user's input and asks further questions to refine the requirements. It asks about the expected number of products, preferred payment methods, shipping options, and any unique features. The user provides more details to narrow down the requirements. The chatbot app summarizes the information provided and asks if there are any specific branding or color preferences. It also inquires about the project timeline and budget constraints. The user answered accordingly. The chatbot app records the newly provided information. It thanks the user for providing the details. The chatbot app provides recommendations for e-commerce platforms based on the user's project requirements and budget. If the user does not demand a requirements document the chat should end. If not, The Chatbot generates a requirement document based on the gathered information and sends it to the user through the chat interface, ensuring they have a record of their project requirements. After that, the user ends the conversation.

Major Features w.r.t Elicitation Techniques

- User Account Management
- Requirements Gathering via Chat
- Project Proposal Generation
- Requirements Listing
- Creation of Software Requirement Document (SRD)
- User Feedback Collection

1- Functional Requirements

F-1.1 User Account Management

- **F-1.1.1:** Users shall be able to register him/herself with a unique username, email address, and password.
- **F-1.1.2:** The software shall allow registered users to log in using their username and password.
- **F-1.1.3:** Users shall be able to change their email address and password.
- **F-1.1.4:** The software shall provide an option for users to deactivate their accounts.
- **F-1.1.5:** The software shall support different user roles (e.g., regular user, admin).

F-1.2 Requirements Gathering via Chat

- **F-1.2.1:** Users shall be able to initiate a chat session for gathering software requirements by typing “Hello”.
- **F-1.2.2:** The software shall process user-provided input in the English language that shall be parsed to identify requirements, questions, and clarifications.
- **F-1.2.3:** The software shall generate relevant requirement elicitation questions based on user-provided information.
- **F-1.2.4:** The software shall capture and store user responses to elicitation questions that shall be associated with specific requirements for documentation.
- **F-1.2.5:** The software shall maintain a record of gathered requirements.
- **F-1.2.6:** The software shall provide assistance and suggestions to users during requirement discussions.
- **F-1.2.7:** Users shall have the ability to confirm and finalize requirements in the chat that shall be marked as accepted and ready for further development.

F-1.3 Project Proposal Generation

- **F-1.3.1:** The software shall define the project scope based on the gathered requirements.
- **F-1.3.2:** The software shall identify, and document project objectives based on the information gathered.

- **F-1.3.3:** The software shall list project features and functionalities derived from the gathered requirements.
- **F-1.3.4:** The software shall generate a project proposal document, based on the integrated details, including scope, features, requirements, etc.
- **F-1.3.5:** The software shall allow users to export the project proposal document that is saved in PDF format.

F-1.4 Requirements Listing

- **F-1.4.1:** The software shall compile a comprehensive list of identified software requirements through information gathered from various sources through user interactions and documents.
- **F-1.4.2:** The software shall classify requirements into different categories or types i.e. functional requirements, non-functional requirements, and user-specific requirements.
- **F-1.4.3:** The software shall allow users to export the compiled list of requirements that saved in PDF format
- **F-1.4.4:** The software shall accommodate updates and changes to individual requirements allowing the user to able to add, modify, or remove requirements from the compiled list.

F-1.5 Creation of Software Requirement Document (SRD)

- **F-1.5.1:** The software shall create a structured Software Requirement Document (SRD) that shall include an introduction, purpose, scope, requirements, dependencies, and any other relevant sections.
- **F-1.5.2:** The software shall allow users to export the SRD document that may be saved in PDF format.
- **F-1.5.3:** Users shall be able to add, modify, or remove requirements and associated content.

F-1.6 User Feedback Collection

- **F-1.6.1:** The software shall provide a mechanism for users to submit feedback on software requirements and the software requirement document.
- **F-1.6.2:** User feedback shall be integrated into the requirements listing and software requirement document that may lead to updates or revisions in the documentation.
- **F-1.6.3:** When user feedback leads to requirement changes, the software shall update the requirements listing.
- **F-1.6.4:** When user feedback leads to document revisions, the software shall update the software requirement document (SRD) and the user shall have access to the latest version of the SRD.

2- Non-Functional Requirements

NF-2.1 Performance

- **NF-2.1.1:** The chatbot app shall respond to user inputs within 2 seconds to ensure a seamless and efficient user experience.
- **NF-2.1.2:** It should be able to handle concurrent interactions with multiple users without significant performance degradation.

NF-2.2 Availability

- **NF-2.2.1:** The chatbot app shall be available 24/7 to cater to user requests and provide assistance at any time.

NF-2.3 Security

- **NF-2.3.1:** User data and project requirements gathered by the chatbot app shall be stored securely and protected from unauthorized access.
- **NF-2.3.2:** The chatbot app shall use encryption protocols to ensure the confidentiality of user data during transmission.

NF-2.4 Scalability

- **NF-2.4.1:** The chatbot app should be designed to scale horizontally to accommodate increased user demand during peak usage periods.

NF-2.5: Accuracy

- **NF-2.5.1:** The chatbot app's natural language processing (NLP) capabilities shall have a minimum accuracy rate of 95% in understanding and interpreting user emotions.
- **NF-2.5.2:** It should consistently provide accurate recommendations and responses based on the gathered requirements.

NF-2.6: User Interface

- **NF-2.6.1:** The chatbot app shall have an intuitive and user-friendly interface to facilitate easy and efficient interaction with users.
- **NF-2.6.2:** It should support multimedia inputs and voice commands, for enhanced user engagement.

NF-2.7 Data Retention and Privacy

- **NF-2.7.1:** User data collected by the chatbot app for requirement gathering shall be retained for a specified period and then automatically deleted to ensure compliance with data privacy regulations.

- **NF-2.7.2:** Users shall have the option to request the deletion of their data at any time.

3- Business Requirements

B-3.1: Purpose

- **B-3.1.1:** The chatbot shall clearly automate the process of discovering requirements
- **B-3.1.2:** The chat sets to streamline the process for elicitation of requirements.

B-3.2: User Satisfaction

- **B-3.2.1:** Provide an enhanced User Feedback system to better understand user needs.
- **B-3.2.2:** The performance of the product should be properly optimized, so users will not get upset.

B-3.3 Time Considerations

- **B-3.3.1:** The project timeline should be as efficient as possible to counter our time constraints.

B-3.4 Security

- **B-3.4.1:** The product's user management should have no loopholes to guarantee user safety.
- **B-3.4.2:** The user must be aware of our security policies to avoid future problems.

B-3.5 Data Privacy

- **B-3.5.1:** Agreement shall be made with the user in accordance with our privacy policy.
- **B-3.5.2:** The user shall only use our product if they agree to our Terms and Conditions.

B-3.6 Business Scalability

- **B-3.6.1:** The system should handle a good sum of users at live time.
- **B-3.6.2:** Performance must not be degraded at times of peak usage of our hosting resources.

B-3.7 SEO (Search Engine Optimization)

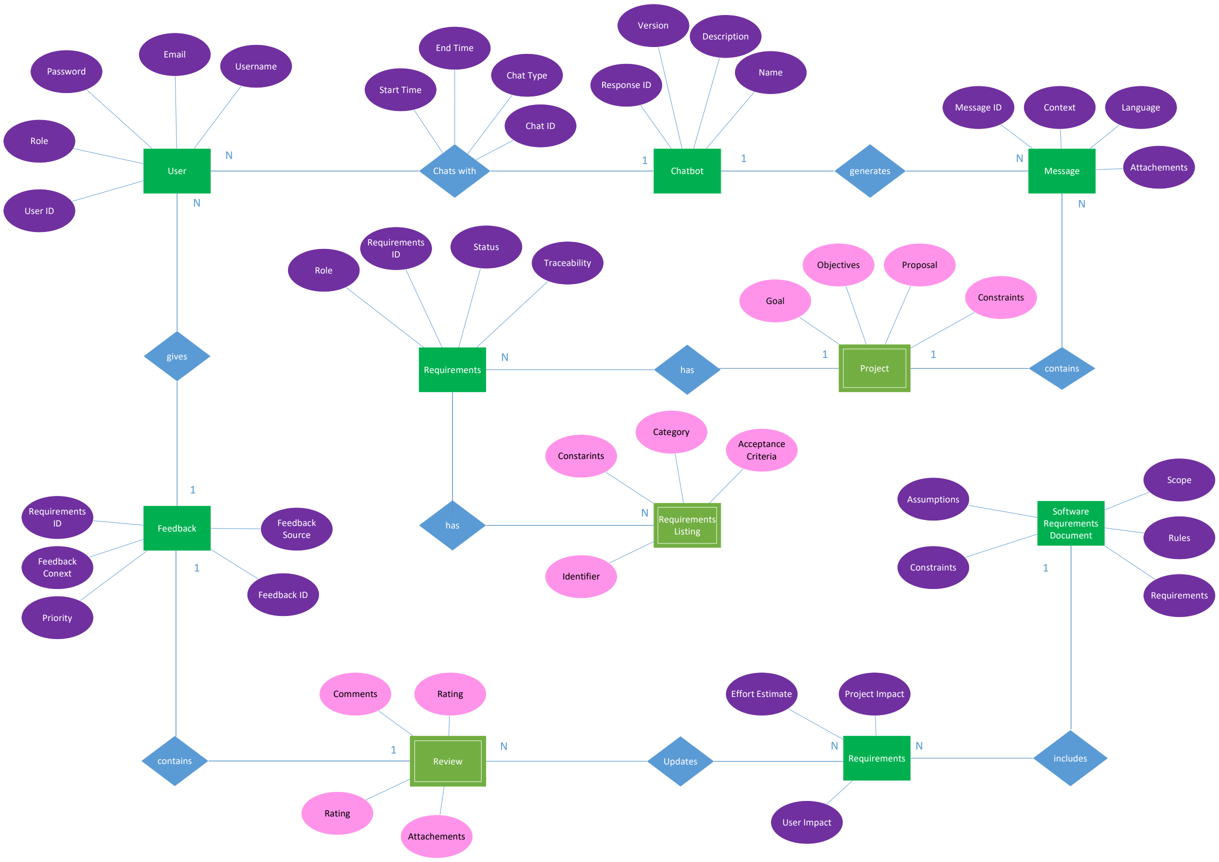
- **B-3.7.1:** Search results must be according to popular keywords so the user can receive documentation according to the trend.
- **B-3.7.2:** Source of Information must be free to use due to our budget constraints

Requirements Elaboration

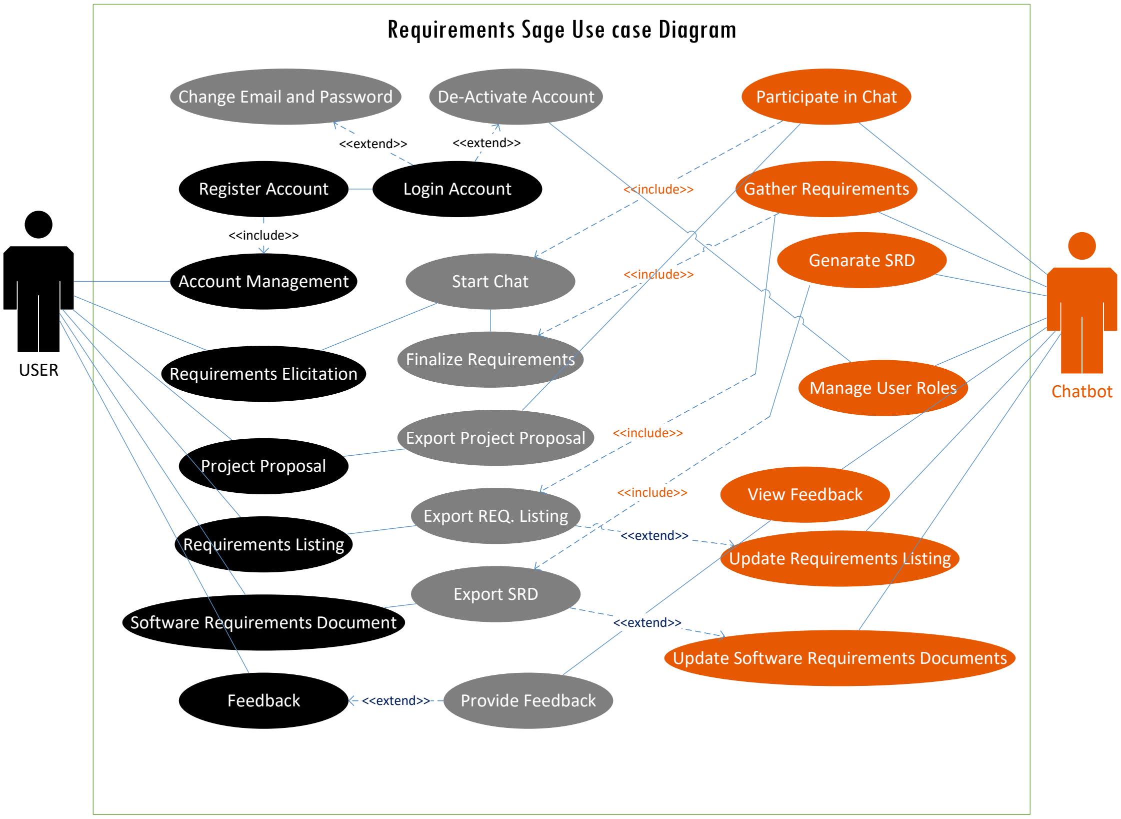
Diagrams

1. Entity Relationship Diagram
2. Use Case Diagram
3. Class Diagram
4. Swim Lane Diagram
5. Data Flow Diagrams

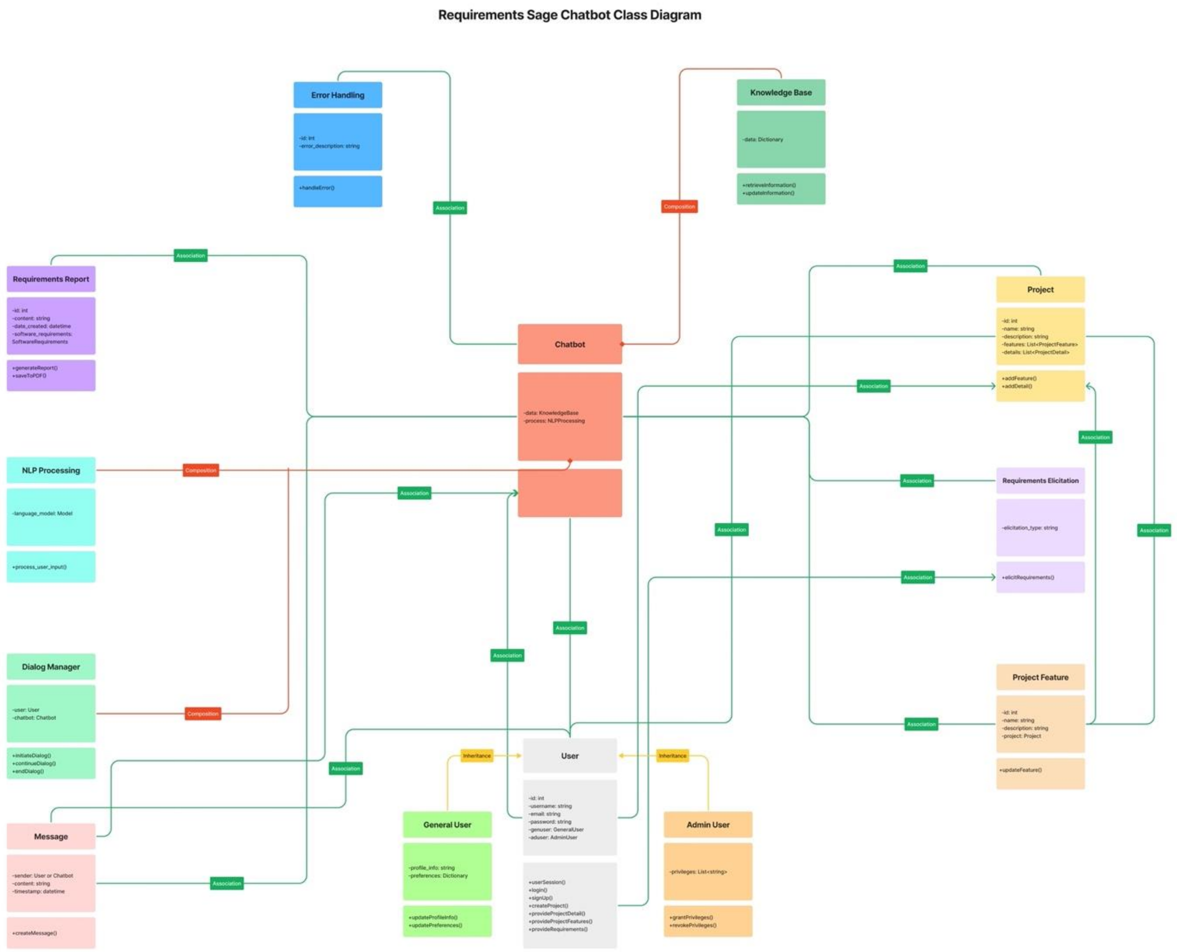
1. Entity Relationship Diagram



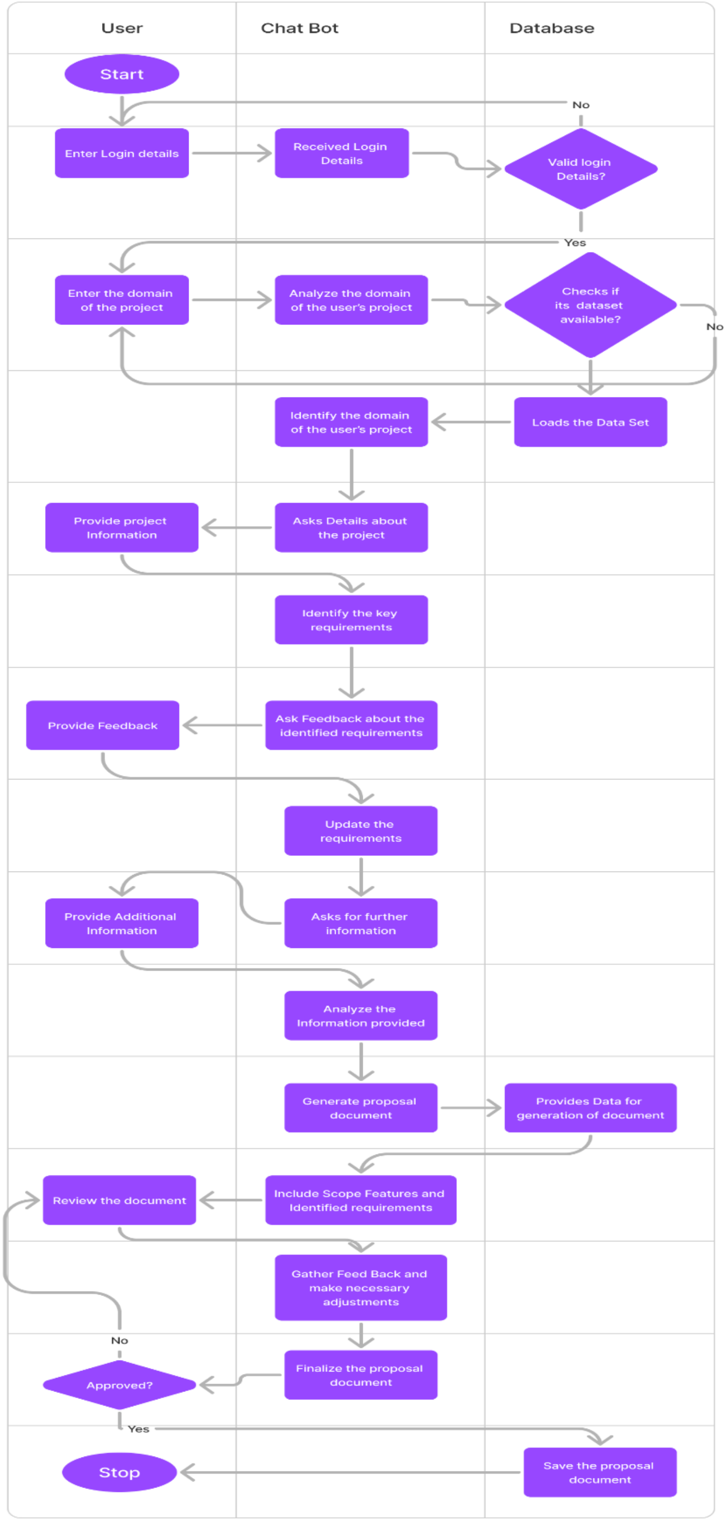
2. Use Case Diagram



3. Class Diagram



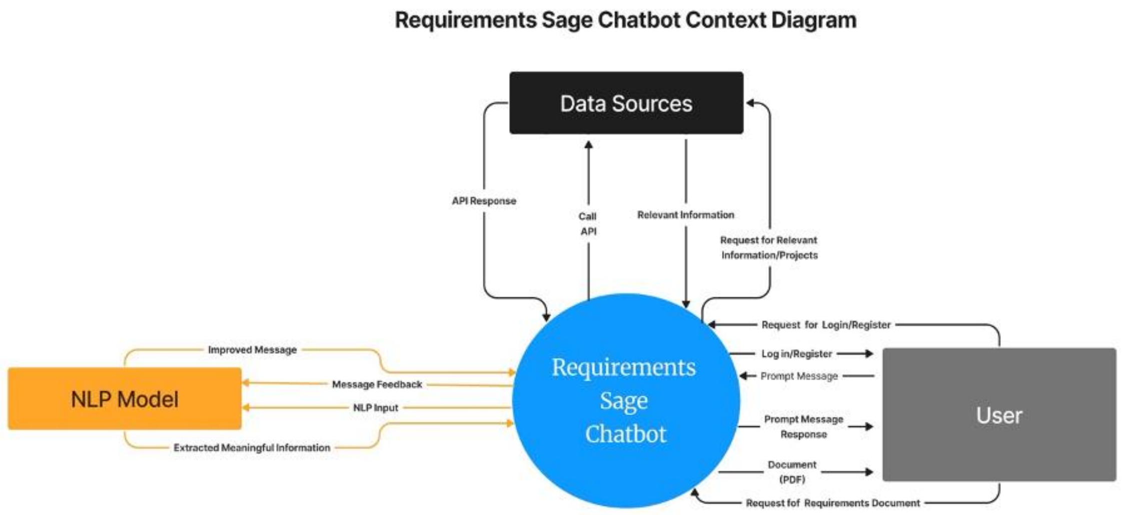
4. Swim Lane Diagram


User
Chat Bot
Database
Start
No
Enter Login details
Received Login Details
Valid login Details?
Yes
Enter the domain of the project
Analyze the domain of the user's project
Checks if its dataset available?
No
Identify the domain of the user's project
Loads the Data Set
Provide project Information
Asks Details about the project
Identify the key requirements
Provide Feedback
Ask Feedback about the identified requirements
Update the requirements
Provide Additional Information
Asks for further information
Analyze the Information provided
Generate proposal document
Provides Data for generation of document
Review the document
Include Scope Features and Identified requirements
Gather Feed Back and make necessary adjustments
No
Approved?
Finalize the proposal document
Yes
Stop
Save the proposal document

5. Data Flow Diagrams

5.1

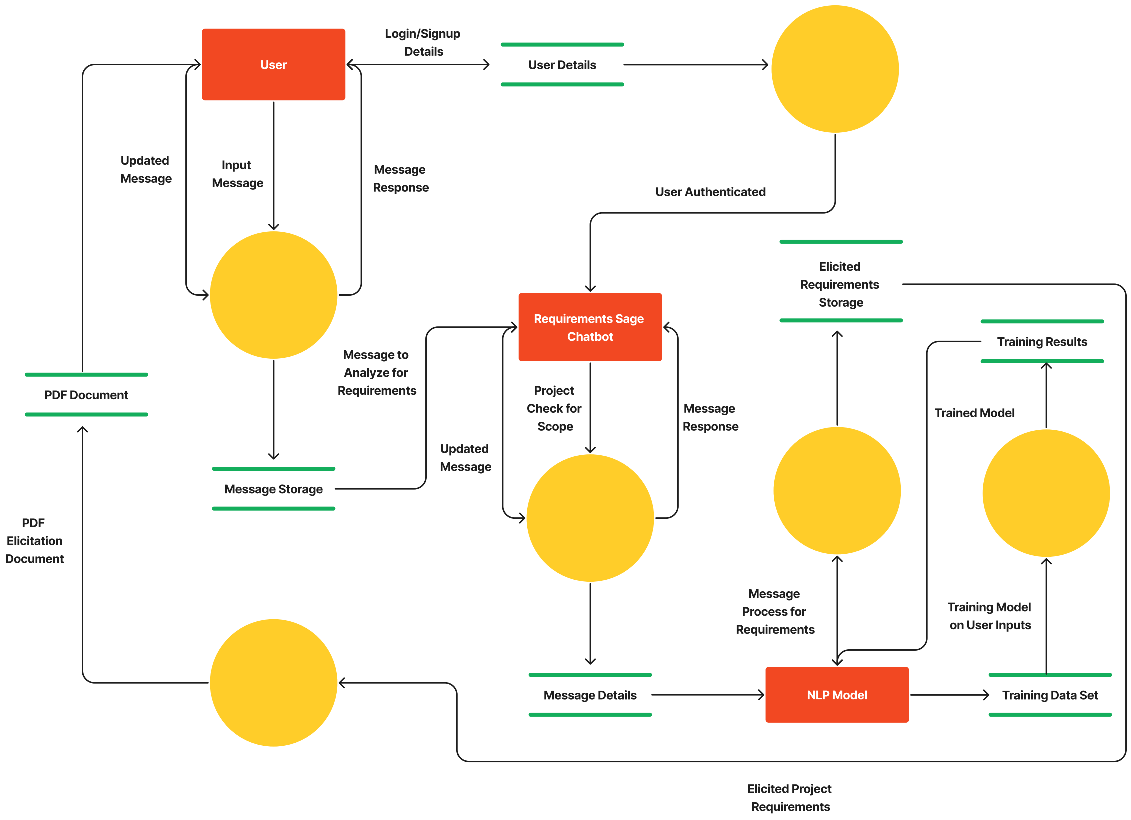
Level-0 DFD



Requirements Sage Chatbot Context Diagram

5.2

Level-1 DFD



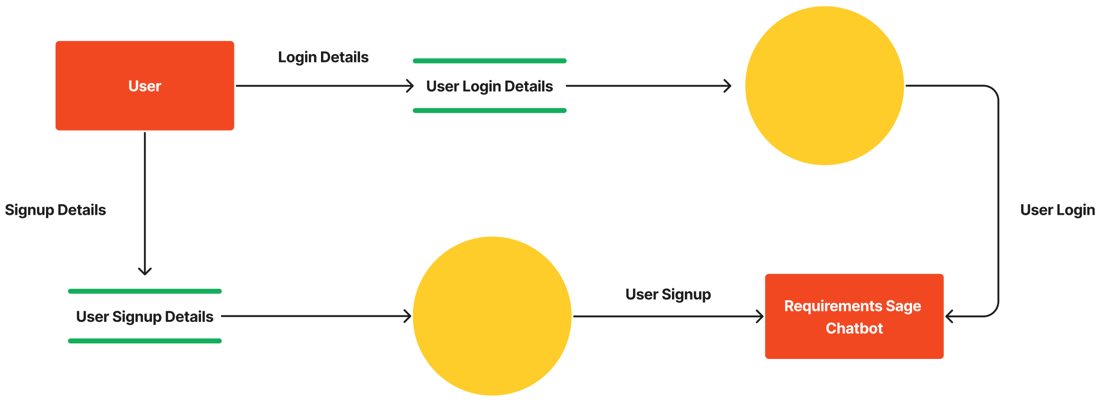
Main Processes:

1. User Authentication
2. User Input/Output
3. Project Scope Check
4. Model Training
5. Requirements Elicitation
6. Requirements Elicitation Document Generation

We need another abstraction of DFD for some processes that are non-primitive such as User Authentication, Model Training, and Requirements Elicitation.

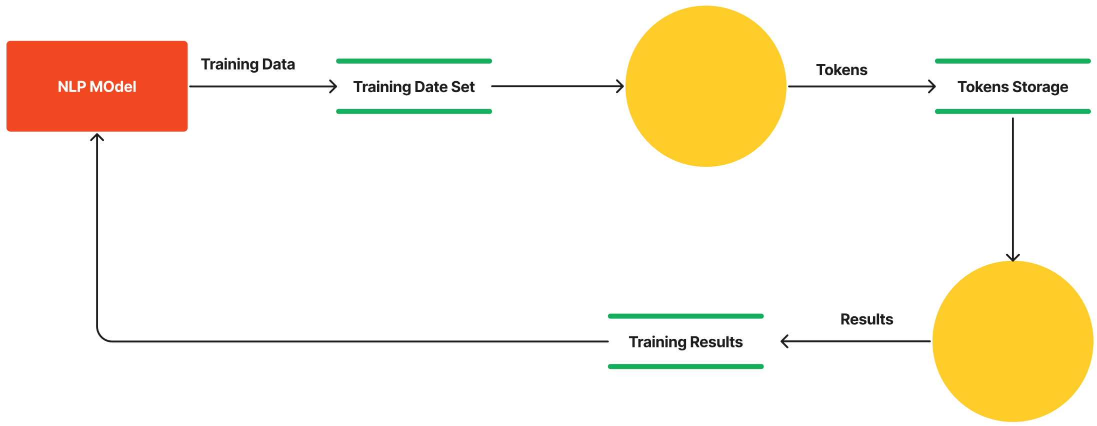
5.3

Level-2 DFD (User Authentication)



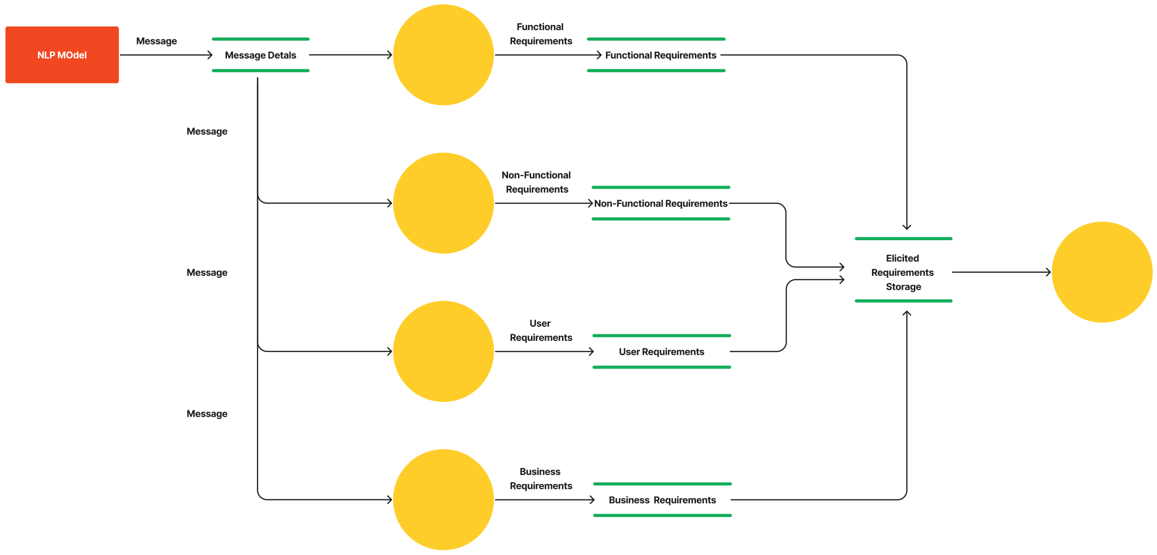
5.4

Level-2 DFD (Model Training)



<u>5.5</u>

Level-2 DFD (Requirement Elicitation)



Primitive Processes

The processes which are found to be primitive are

- 1.1 Login
- 1.2 Signup
- 3. Project Scope Check
- 4.1 Tokenization
- 4.2 Reinforcement Learning
- 5.1 Elicit User Requirements
- 5.2 Elicit Business Requirements
- 5.3 Elicit Functional Requirements
- 5.4 Elicit Non-Functional Requirements
- 6. Requirements Elicitation Document Generation

These processes are likely to not break down into sub processes but still we shall make another abstraction for these processes if needed.