Requirement Elicitation Document

Sathwika Valija
Business Analyst and Requirements Engineering
5/12/25

REQUIREMENT ELICITATION DOCUMENT

IARE UNOFFICIAL BOT

1.1 Project Overview

The Web Scraper Bot for IARE CMS is a Python-based Telegram bot designed to securely scrape attendance and biometric data from the samvidha portal and streamline student activities such as uploading lab records and monitoring attendance. This automation improves visibility, security, and efficiency for students and admins.

1.2 Introduction

The IARE CMS Web Scraper Bot project was initiated to address key limitations faced by students while interacting with the Samvidha portal of the Institute of Aeronautical Engineering (IARE). The primary goal of this elicitation process is to gather, analyse, and document the requirements of the bot from all stakeholders involved-including students, developers, and administrators. This bot aims to automate and streamline attendance tracking, biometric data visibility, lab uploads, and issue reporting through an efficient and user-friendly interface. The elicitation process involved multiple techniques such as observations, and feedback collection to ensure the system addresses real user problems effectively.

1.3 Purpose of Elicitation

The purpose of this elicitation process is to gather, understand, and document the functional and non-functional requirements for the development of a Web Scraper Bot designed for the IARE CMS (Institute of Aeronautical Engineering Content Management System). This bot aims to automate the retrieval of student biometric data, lab attendance records, and facilitate PDF compression, feedback collection, and automated uploads using a user-friendly interface.

Through this elicitation, we aim to identify:

- The scope of automation required in scraping and processing CMS data.
- The usability and interface expectations for efficient usage.
- Any technical or security constraints related to CMS access.

1.4 Scope of the Document

This document outlines the scope of requirement elicitation for the development of the IARE CMS Web Scraper Bot. The primary focus is to gather, analyse, and define the functional and non-functional requirements necessary for building a secure and efficient automation tool that interacts with the IARE CMS (Samvidha portal). The scope includes:

- Scraping of **biometric** and lab, **attendance** data from the CMS.
- Enabling PDF compression for **optimized** upload of lab records.
- Facilitating a **user-friendly, button-based** interface that supports easy access to reports and file uploads.

- Addressing security and login protocols, ensuring data **confidentiality** without persistent credentials.
- Ensuring **scalability** to handle multiple users and concurrent requests.
- Identifying any **technical** or **security limitations** tied to CMS architecture or server deployment.

This scope ensures that the elicited requirements directly support the core objective: Enhancing usability, speed, and reliability in managing student attendance data and associated tasks through automation.

1.5 Stakeholders

- **Students:** End-users of the bot for viewing attendance, uploading the files (influence – High).
- **Admins:** Maintain developer domain and control access (influence – High).
- **Developers:** Build and maintain the bot and responds to user feedback/issues (influence – Medium).

1.6 Sources of Requirements

- Feedback from students about attendance visibility and upload difficulty.
- Observations on server migration and user performance.
- Reviewed the current CMS structure, existing attendance workflows, and student feedback mechanisms to understand data flow and integration needs.
- Discussions with developers and open-source contributors.
- Identifying requirements based on available technologies, infrastructure, and compatibility needs.

1.7 Functional Requirements

- Scraping of biometric/attendance data from Samvidha.
- Secure login without storing credentials post logout.
- Feedback system (report/respond mechanism).
- Categorization of feedback.
- Upload lab files using week number and subject.
- Compress lab PDF files to <1MB for easy upload.
- Button-based user interface.
- Real-time response for multiple users.
- Regex-based index automation.

1.8 Non-Functional Requirements

- **Scalable** (multiple users simultaneously)

- **Hosted on Heroku** (migrated from a slower server)
- **Cost-effective** (approx. $12/month for server)
- **Mobile** and **Desktop** compatibility
- **Optimized** for concurrency
- **Secure** (no credential storage after post logout)
- **Reliable scraping** even after HTML changes (resilient to structural changes)
- **Efficient** compression through API's PDF compression
- **Open-source**

1.9 Constraints and Assumptions

Constraints:

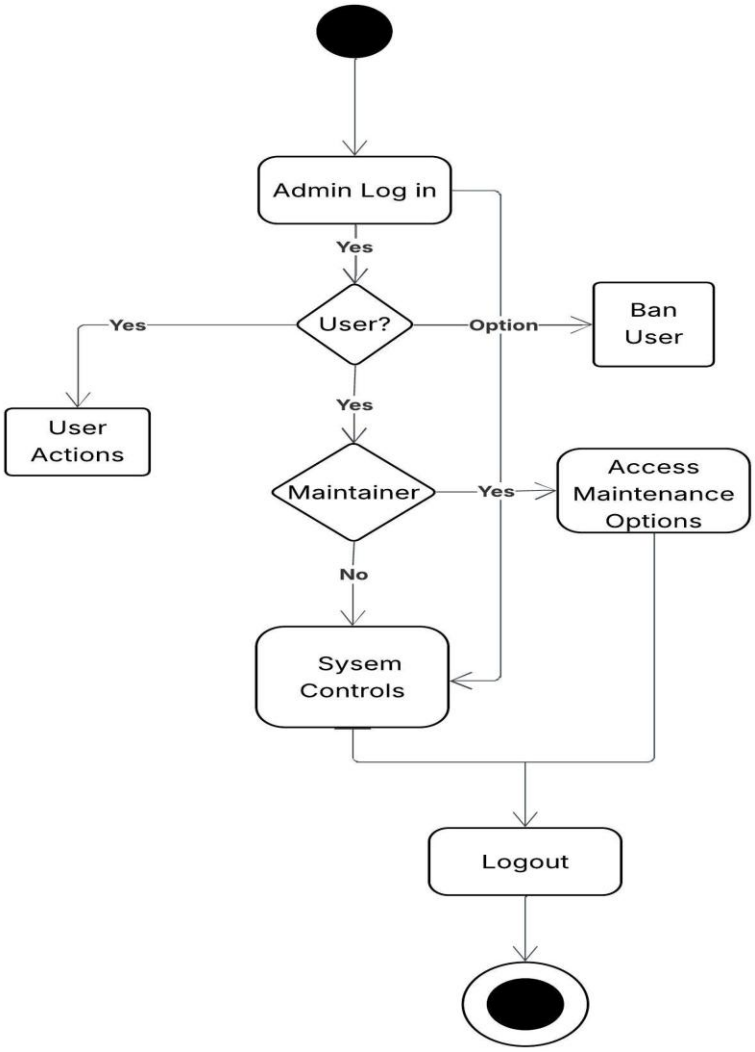
- The bot must comply with CMS security policies and avoid violating institutional access protocols.
- It can only operate within the limits of the CMS user interface, as no API is officially provided.
- Execution must work in Windows OS, which is the default environment at IARE.
- Limited to Python and Pyrogram libraries for Bot automation.

Assumptions:

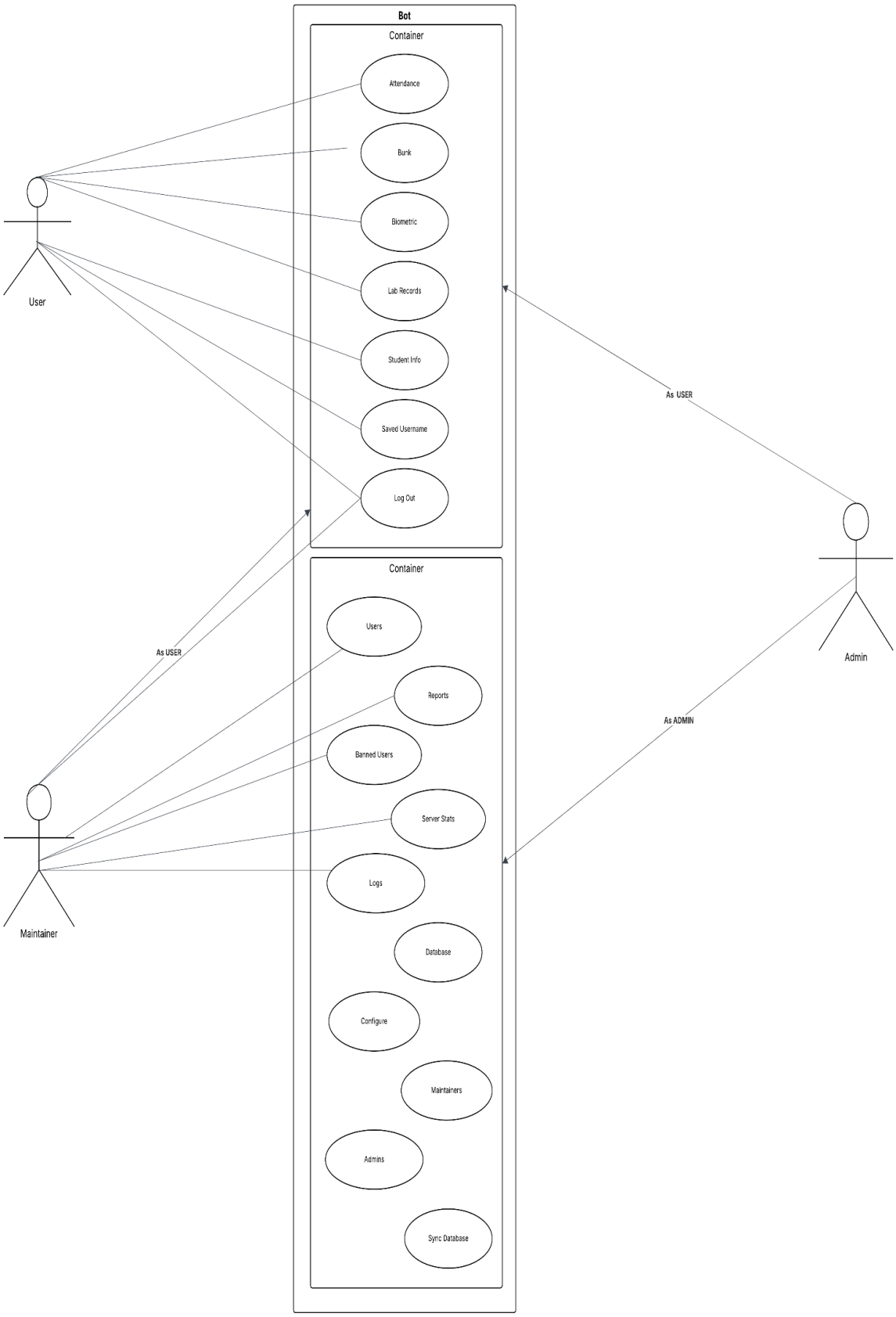
- Internet connectivity is stable during bot execution.
- The CMS interface will not change significantly during the bot's deployment.
- Required software dependencies (e.g., Python 3, libraries) will be installed on the host system.

1.10 Diagrams

- **Activity diagram**



- **Use Case Diagram**



1.11 Conclusion

The requirement elicitation process successfully captured the functional and non-functional needs of the stakeholders involved in using the IARE CMS Bot Unofficial. By analysing feedback and observing current system limitation, a comprehensive understanding of the desired improvements was achieved. These insights from the foundation for building a secure, scalable, and efficient bot that enhances student experience and administrative ease. The elicited requirements will guide the development process, ensuring the final solution is both practical and impactful.