# Aluminto: Alumni Management System

## 1. Introduction

### 1.1 Purpose

The purpose of Aluminto is to provide an efficient platform for managing alumni across an institute. It aims to streamline the process of alumni registration, communication, and data management for both alumni and administrators.

### 1.2 Scope

Aluminto will include features such as alumni registration, profile management, event management, communication tools, and administrative functionalities for managing alumni data. It will utilize a MySQL database to store alumni information and interaction details.

### 1.3 Definitions, Acronyms, and Abbreviations

SRS: Software Requirements Specification
CLI: Command Line Interface
DBMS: Database Management System
API: Application Programming Interface

## 2. Overall Description

### 2.1 Product Perspective

Aluminto will be a command-line application written in Python, accessible via terminals or command prompts. It will interact with a MySQL database for storing and retrieving alumni data. The application will focus on providing essential functionalities for managing alumni interactions and events.

### 2.2 Product Features

◦ Alumni registration and profile management
◦ Event management for alumni gatherings and activities
◦ Communication tools for sending announcements and updates to alumni
◦ Administrative panel for managing alumni data and events

### 2.3 User Classes and Characteristics Alumni Users:

Can register, update their profiles, view events, and receive communication from the institute. Administrators: Have access to administrative functionalities such as adding/editing alumni data, managing events, and sending announcements.

## 3. Specific Requirements

### 3.1 External Interface Requirements

### 3.1.1 User Interfaces

The CLI shall provide a user-friendly interface with clear prompts and instructions for users to navigate through the application.

### 3.1.2 Hardware Interfaces

The system shall operate on standard computing hardware compatible with Python and MySQL.

### 3.1.3 Software Interfaces

Aluminto shall interact with the MySQL database for storing and retrieving alumni data. It may integrate with external APIs for email communication or event management services.

## 3.2 Functional Requirements

3.2.1 Alumni Management Users shall be able to register as alumni by providing necessary details such as name, graduation year, contact information, etc. Registered alumni shall be able to update their profiles as needed.

3.2.2 Event Management Administrators shall be able to create, edit, and delete events.

3.2.3 Administrative Panel Administrators shall have access to functionalities for managing alumni data, events, and communication settings.

## 3.3 Non-Functional Requirements

### 3.3.1 Performance

The system shall respond to user inputs promptly, with minimal latency in data retrieval and processing.

### 3.3.2 Security

Alumni data shall be stored securely in the database with proper encryption and access control measures. Authentication mechanisms shall be implemented to ensure that only authorized users can access and modify alumni information.

### 3.3.3 Reliability

The system shall be reliable, with minimal downtime and robust error handling mechanisms in place to handle unexpected errors or failures gracefully.