

# ACADEMIC ANALYTICS SYSTEM

## Software Requirements Specification

V1.0

04/24/2023

TEAM-03

Instructor: **Muthumari Nammalwar**  
Spring 2023

## Revision History

Date	Description	Author	Comments
04/24/2023	Version 1.0	Team-03	First Revision

## Document Approval

The following Software Requirements Specification has been accepted and approved by the following:

Signature	Printed Name	Title	Date

# Table of Contents

<b>REVISION HISTORY .....</b>	<b>II</b>
<b>DOCUMENT APPROVAL .....</b>	<b>II</b>
<b>1. INTRODUCTION .....</b>	<b>1</b>
<p>THE ACADEMIC ANALYTICS SYSTEM IS A WEB-BASED APPLICATION DESIGNED TO PROVIDE ACADEMIC INSTITUTIONS WITH A COMPREHENSIVE PLATFORM FOR MONITORING AND ANALYZING STUDENT AND FACULTY DATA. THE SYSTEM IS INTENDED TO HELP ADMINISTRATORS AND FACULTY MEMBERS MAKE INFORMED DECISIONS ABOUT ACADEMIC PROGRAMS, POLICIES, AND STRATEGIES BY PROVIDING THEM WITH ACCESS TO A WIDE RANGE OF DATA AND ANALYTICS TOOLS. THE SYSTEM IS INTENDED TO BE USER-FRIENDLY, SCALABLE, AND SECURE, WITH A FLEXIBLE ARCHITECTURE THAT CAN ACCOMMODATE FUTURE GROWTH AND DEVELOPMENT. THIS SOFTWARE REQUIREMENTS SPECIFICATION (SRS) DOCUMENT OUTLINES THE FUNCTIONAL AND NON-FUNCTIONAL REQUIREMENTS OF THE SYSTEM, AS WELL AS ANY CONSTRAINTS, ASSUMPTIONS, AND DEPENDENCIES THAT MAY IMPACT ITS DEVELOPMENT. THE DOCUMENT IS INTENDED TO PROVIDE A CLEAR AND COMPREHENSIVE DESCRIPTION OF THE SYSTEM'S REQUIREMENTS, WHICH WILL SERVE AS A BASIS FOR ITS DESIGN, DEVELOPMENT, TESTING, AND DEPLOYMENT. ....</p>	
1.1 PURPOSE .....	1
<p>THE PURPOSE OF THIS SRS DOCUMENT IS TO DEFINE THE REQUIREMENTS FOR THE DEVELOPMENT OF AN ACADEMIC ANALYTICS SYSTEM. THE SYSTEM WILL BE DESIGNED TO HELP ACADEMIC INSTITUTIONS MONITOR AND ANALYZE STUDENT AND FACULTY DATA IN ORDER TO MAKE INFORMED DECISIONS. ....</p>	
1.2 SCOPE .....	1
1.3 DEFINITIONS, ACRONYMS, AND ABBREVIATIONS .....	1
1.4 REFERENCES .....	2
<ul style="list-style-type: none"> <li>NATIONAL CENTER FOR EDUCATION STATISTICS. (2021). DIGEST OF EDUCATION STATISTICS, 2020 (NCES 2021-038). U.S. DEPARTMENT OF EDUCATION, INSTITUTE OF EDUCATION SCIENCES. <a href="https://nces.ed.gov/pubsearch/pubsinfo.asp?pubid=2021038">https://nces.ed.gov/pubsearch/pubsinfo.asp?pubid=2021038</a> .....</li> <li>EDUCAUSE. (2019). THE TOP 10 IT ISSUES, 2019: THE DRIVE TO DIGITAL TRANSFORMATION BEGINS. <a href="https://library.educause.edu/resources/2019/1/the-top-10-it-issues-2019-the-drive-to-digital-transformation-begins">https://library.educause.edu/resources/2019/1/the-top-10-it-issues-2019-the-drive-to-digital-transformation-begins</a> .....</li> </ul>	2
1.5 OVERVIEW .....	2
<p>THE SYSTEM WILL SUPPORT DATA VISUALIZATION, PREDICTIVE ANALYTICS, AND REPORTING TOOLS, AND WILL PROVIDE USERS WITH A USER-FRIENDLY, INTUITIVE INTERFACE THAT ALLOWS FOR EASY NAVIGATION AND EXPLORATION OF DATA. THE AAS WILL BE DESIGNED TO BE SCALABLE, SECURE, AND FLEXIBLE, WITH A MODULAR ARCHITECTURE THAT CAN BE CUSTOMIZED TO MEET THE NEEDS OF INDIVIDUAL INSTITUTIONS.....</p>	
<b>2. GENERAL DESCRIPTION .....</b>	<b>2</b>
<p>THE ACADEMIC ANALYTICS SYSTEM (AAS) IS A SOFTWARE SYSTEM DESIGNED TO PROVIDE ACADEMIC INSTITUTIONS WITH A PLATFORM FOR MONITORING AND ANALYZING STUDENT AND FACULTY DATA. THE SYSTEM IS INTENDED TO ASSIST ACADEMIC ADMINISTRATORS AND FACULTY MEMBERS IN MAKING DATA-DRIVEN DECISIONS ABOUT ACADEMIC PROGRAMS, POLICIES, AND STRATEGIES. ....</p>	
2.1 PRODUCT PERSPECTIVE .....	2
<p>THE AAS WILL BE INTEGRATED WITH A VARIETY OF EXTERNAL DATA SOURCES, INCLUDING STUDENT INFORMATION SYSTEMS, LEARNING MANAGEMENT SYSTEMS, AND INSTITUTIONAL RESEARCH DATABASES. THE SYSTEM WILL ALSO INCLUDE A VARIETY OF DATA ANALYTICS AND REPORTING TOOLS THAT WILL ALLOW USERS TO GENERATE CUSTOMIZED REPORTS AND VISUALIZATIONS BASED ON THE DATA IN THE SYSTEM. ....</p>	
2.2 PRODUCT FUNCTIONS .....	2
<p>THE AAS WILL BE DEVELOPED AS A MODULAR SYSTEM, WITH A VARIETY OF DIFFERENT MODULES THAT CAN BE CUSTOMIZED AND INTEGRATED TO MEET THE SPECIFIC NEEDS OF DIFFERENT ACADEMIC INSTITUTIONS. EACH</p>	

MODULE WILL BE DESIGNED TO BE INDEPENDENT AND FLEXIBLE, SO THAT IT CAN BE MODIFIED OR REPLACED WITHOUT AFFECTING THE FUNCTIONALITY OF THE OVERALL SYSTEM.....	2
2.3 USER CHARACTERISTICS.....	3
2.4 GENERAL CONSTRAINTS.....	3
THE ACADEMIC ANALYTICS SYSTEM (AAS) MUST OPERATE WITHIN CERTAIN CONSTRAINTS IN ORDER TO ENSURE ITS SUCCESSFUL DEVELOPMENT AND IMPLEMENTATION. THE FOLLOWING CONSTRAINTS HAVE BEEN IDENTIFIED:...	3
• TECHNICAL CONSTRAINTS: THE AAS MUST BE DESIGNED AND DEVELOPED USING INDUSTRY-STANDARD TECHNOLOGIES AND BEST PRACTICES TO ENSURE THAT IT IS RELIABLE, SCALABLE, AND MAINTAINABLE. THE SYSTEM SHOULD BE COMPATIBLE WITH COMMON WEB BROWSERS AND OPERATING SYSTEMS.....	3
• TIME CONSTRAINTS: THE AAS MUST BE DEVELOPED AND IMPLEMENTED WITHIN A SPECIFIC TIMEFRAME IN ORDER TO MEET THE NEEDS OF ACADEMIC INSTITUTIONS. THE DEVELOPMENT SCHEDULE SHOULD ALLOW FOR ADEQUATE TESTING AND QUALITY ASSURANCE.....	4
• BUDGET CONSTRAINTS: THE DEVELOPMENT AND IMPLEMENTATION OF THE AAS MUST BE COMPLETED WITHIN A SPECIFIC BUDGET. THE SYSTEM SHOULD BE DESIGNED WITH COST-EFFECTIVENESS IN MIND, AND SHOULD NOT REQUIRE SIGNIFICANT ONGOING MAINTENANCE OR SUPPORT COSTS.....	4
• DATA CONSTRAINTS: THE AAS MUST ADHERE TO DATA PRIVACY REGULATIONS AND GUIDELINES, AND SHOULD BE DESIGNED TO PROTECT SENSITIVE ACADEMIC DATA. THE SYSTEM SHOULD ALSO BE DESIGNED TO ENSURE DATA ACCURACY, COMPLETENESS, AND CONSISTENCY.....	4
• RESOURCE CONSTRAINTS: THE AAS DEVELOPMENT TEAM WILL HAVE LIMITED RESOURCES, INCLUDING PERSONNEL, HARDWARE, AND SOFTWARE. THE SYSTEM SHOULD BE DESIGNED TO MAKE THE MOST EFFICIENT USE OF AVAILABLE RESOURCES, AND SHOULD BE DESIGNED TO FACILITATE FUTURE DEVELOPMENT AND GROWTH. ....	4
• LEGAL CONSTRAINTS: THE AAS MUST COMPLY WITH ALL RELEVANT LAWS AND REGULATIONS, INCLUDING THOSE RELATED TO DATA PRIVACY, INTELLECTUAL PROPERTY, AND ACCESSIBILITY. THE SYSTEM SHOULD ALSO BE DESIGNED TO ENSURE THAT IT IS ACCESSIBLE TO USERS WITH DISABILITIES. ....	4
.2.5 ASSUMPTIONS AND DEPENDENCIES .....	4
THE SUCCESSFUL DEVELOPMENT AND IMPLEMENTATION OF THE ACADEMIC ANALYTICS SYSTEM (AAS) IS DEPENDENT ON A NUMBER OF ASSUMPTIONS AND DEPENDENCIES. THESE INCLUDE:.....	4
• <b>AVAILABILITY OF DATA SOURCES:</b> THE AAS IS DEPENDENT ON THE AVAILABILITY OF RELEVANT DATA SOURCES, INCLUDING STUDENT INFORMATION SYSTEMS, LEARNING MANAGEMENT SYSTEMS, AND INSTITUTIONAL RESEARCH DATABASES. THE SYSTEM WILL NOT BE ABLE TO FUNCTION WITHOUT ACCESS TO THESE DATA SOURCES.....	4
• <b>ACCESS TO HARDWARE AND SOFTWARE RESOURCES:</b> THE AAS DEVELOPMENT TEAM WILL REQUIRE ACCESS TO HARDWARE AND SOFTWARE RESOURCES, INCLUDING DEVELOPMENT AND TESTING ENVIRONMENTS, DATABASES, AND SERVERS.....	5
• <b>AVAILABILITY OF DEVELOPMENT RESOURCES:</b> THE SUCCESSFUL DEVELOPMENT OF THE AAS IS DEPENDENT ON THE AVAILABILITY OF SKILLED DEVELOPERS, TESTERS, AND PROJECT MANAGERS. ....	5
• <b>USER PARTICIPATION:</b> THE AAS IS DESIGNED TO BE A USER-DRIVEN SYSTEM, AND IS DEPENDENT ON THE ACTIVE PARTICIPATION OF ACADEMIC ADMINISTRATORS AND FACULTY MEMBERS. THE SYSTEM WILL NOT BE EFFECTIVE WITHOUT USER INPUT AND ENGAGEMENT.....	5
• <b>DATA PRIVACY REGULATIONS:</b> THE AAS IS DESIGNED TO HANDLE SENSITIVE ACADEMIC DATA, AND IS DEPENDENT ON COMPLIANCE WITH RELEVANT DATA PRIVACY REGULATIONS AND GUIDELINES. ....	5
• <b>AVAILABILITY OF FUNDING:</b> THE DEVELOPMENT AND IMPLEMENTATION OF THE AAS IS DEPENDENT ON THE AVAILABILITY OF FUNDING, BOTH FOR DEVELOPMENT COSTS AND ONGOING MAINTENANCE AND SUPPORT COSTS. ....	5
• <b>TECHNICAL DEPENDENCIES:</b> THE AAS IS DEPENDENT ON A NUMBER OF TECHNICAL COMPONENTS, INCLUDING WEB SERVERS, DATABASES, AND ANALYTICS TOOLS. THESE COMPONENTS MUST BE COMPATIBLE AND INTEGRATE SEAMLESSLY IN ORDER FOR THE SYSTEM TO FUNCTION PROPERLY. ....	5

<b>3. SPECIFIC REQUIREMENTS</b>	<b>5</b>
THE ACADEMIC ANALYTICS SYSTEM (AAS) MUST MEET THE FOLLOWING SPECIFIC REQUIREMENTS:	5
1. USER MANAGEMENT: THE SYSTEM MUST HAVE A USER MANAGEMENT MODULE TO MANAGE USER ACCOUNTS, ROLES, AND PERMISSIONS. THE MODULE MUST SUPPORT THE CREATION, MODIFICATION, AND DELETION OF USER ACCOUNTS.	5
2. DATA COLLECTION: THE SYSTEM MUST BE ABLE TO COLLECT DATA FROM MULTIPLE SOURCES, INCLUDING STUDENT INFORMATION SYSTEMS, LEARNING MANAGEMENT SYSTEMS, AND INSTITUTIONAL RESEARCH DATABASES. THE DATA COLLECTION PROCESS MUST BE AUTOMATED AND SECURE.	5
3. DATA PROCESSING: THE SYSTEM MUST BE ABLE TO PROCESS COLLECTED DATA TO GENERATE MEANINGFUL ANALYTICS AND REPORTS. THE PROCESSING ALGORITHMS MUST BE ACCURATE, EFFICIENT, AND SCALABLE.	5
4. ANALYTICS AND REPORTS: THE SYSTEM MUST BE ABLE TO GENERATE A WIDE RANGE OF ANALYTICS AND REPORTS BASED ON PROCESSED DATA. THE ANALYTICS AND REPORTS MUST BE CUSTOMIZABLE, AND MUST SUPPORT MULTIPLE DATA VISUALIZATIONS.	5
5. ALERTING: THE SYSTEM MUST HAVE AN ALERTING MODULE TO NOTIFY USERS OF IMPORTANT EVENTS OR CHANGES IN DATA. THE ALERTING MODULE MUST BE CUSTOMIZABLE, AND MUST SUPPORT MULTIPLE COMMUNICATION CHANNELS.	5
6. SECURITY: THE SYSTEM MUST BE DESIGNED TO ENSURE THE SECURITY OF SENSITIVE ACADEMIC DATA. THE SYSTEM MUST ADHERE TO RELEVANT DATA PRIVACY REGULATIONS AND GUIDELINES, AND MUST IMPLEMENT APPROPRIATE ACCESS CONTROLS AND ENCRYPTION.	6
7. ACCESSIBILITY: THE SYSTEM MUST BE DESIGNED TO ENSURE ACCESSIBILITY FOR ALL USERS, INCLUDING THOSE WITH DISABILITIES. THE SYSTEM MUST ADHERE TO RELEVANT ACCESSIBILITY STANDARDS AND GUIDELINES.	6
8. PERFORMANCE: THE SYSTEM MUST BE DESIGNED TO HANDLE LARGE VOLUMES OF DATA AND USERS, AND MUST BE SCALABLE AND PERFORMANT. THE SYSTEM MUST BE ABLE TO HANDLE MULTIPLE CONCURRENT REQUESTS WITHOUT SIGNIFICANT PERFORMANCE DEGRADATION.	6
3.1 EXTERNAL INTERFACE REQUIREMENTS	6
3.2 FUNCTIONAL REQUIREMENTS	6
3.3 USE CASES	7
3.3.1 Use Case #1 & #2	7
3.3.2 Use Case #3	8
3.4 NON-FUNCTIONAL REQUIREMENTS	12
<i>Non-functional requirements are those that specify the qualities or attributes that the system must have, but that are not related to its functionality. Here are some possible non-functional requirements for an Academic Analytics System:</i>	12
1. Usability: The system should be easy to use and navigate, with clear and intuitive interfaces, minimal errors and confusion, and consistent feedback and help features.	12
2. Performance: The system should be responsive and fast, with quick loading times, efficient processing of data, and minimal delays or downtime.	12
3. Security: The system should be secure and reliable, with strong authentication and authorization measures, data encryption and backup, and protection against unauthorized access, theft, or damage.	12
4. Scalability: The system should be able to handle increasing volumes of data, users, and transactions, without significant loss of performance or functionality.	12
5. Compatibility: The system should be compatible with a range of hardware, software, and network configurations, and should be able to integrate with other systems or tools used by the academic institution.	12
6. Accessibility: The system should be accessible to users with disabilities or special needs, with support for assistive technologies, alternative input/output methods, and compliance with relevant accessibility standards.	12
7. Reliability: The system should be reliable and accurate, with minimal errors, bugs, or data inconsistencies, and with proper monitoring, testing, and debugging procedures.	12
8. Maintainability: The system should be easy to maintain and update, with clear documentation, version control, and modular design, and with support for troubleshooting, bug fixing, and enhancement requests.	12
9. Legal and ethical compliance: The system should comply with relevant laws, regulations, and ethical standards related to data privacy, security, and use, as well as with the policies and guidelines of the academic institution.	12
10. Performance metrics: The system should be able to track and measure its own performance and usage, with relevant metrics and indicators, such as response time, usage patterns, and resource utilization.	12

3.5 INVERSE REQUIREMENTS .....	12
INVERSE REQUIREMENTS ARE SPECIFICATIONS THAT DESCRIBE WHAT THE SYSTEM SHOULD NOT DO OR HAVE, RATHER THAN WHAT IT SHOULD DO OR HAVE. THEY ARE ALSO KNOWN AS "NEGATIVE REQUIREMENTS": .....	12
1. THE SYSTEM SHOULD NOT VIOLATE STUDENTS' PRIVACY RIGHTS, SUCH AS BY COLLECTING OR SHARING SENSITIVE OR PERSONAL DATA WITHOUT THEIR CONSENT OR INAPPROPRIATELY .....	12
2. THE SYSTEM SHOULD NOT GENERATE OR DISPLAY INACCURATE OR MISLEADING DATA OR INSIGHTS, SUCH AS BY USING FAULTY OR BIASED ALGORITHMS, INCOMPLETE OR OUTDATED DATA, OR INAPPROPRIATE ASSUMPTIONS OR MODELS. ....	12
3. THE SYSTEM SHOULD NOT CAUSE HARM OR DAMAGE TO USERS, DATA, OR SYSTEMS, SUCH AS BY INTRODUCING VIRUSES, BUGS, OR SECURITY BREACHES, OR BY OVERLOADING OR DISRUPTING NETWORK OR COMPUTING RESOURCES. ....	13
4. THE SYSTEM SHOULD NOT DISCRIMINATE OR DISADVANTAGE ANY STUDENTS OR GROUPS BASED ON THEIR GENDER, RACE, ETHNICITY, RELIGION, SEXUAL ORIENTATION, OR OTHER PERSONAL CHARACTERISTICS, SUCH AS BY APPLYING UNFAIR OR BIASED CRITERIA OR EVALUATIONS. ....	13
5. THE SYSTEM SHOULD NOT BE USED FOR ANY ILLEGAL OR UNETHICAL PURPOSES, SUCH AS BY FACILITATING CHEATING, PLAGIARISM, OR ACADEMIC MISCONDUCT, OR BY VIOLATING COPYRIGHT OR INTELLECTUAL PROPERTY RIGHTS. ....	13
6. THE SYSTEM SHOULD NOT BE OVERLY COMPLEX OR CONFUSING, SUCH AS BY INCLUDING UNNECESSARY OR REDUNDANT FEATURES, OR BY REQUIRING EXTENSIVE TRAINING OR TECHNICAL SKILLS TO USE.....	13
7. THE SYSTEM SHOULD NOT BE INCOMPATIBLE OR INCONSISTENT WITH OTHER ACADEMIC OR ADMINISTRATIVE SYSTEMS USED BY THE INSTITUTION, SUCH AS BY CREATING CONFLICTS OR ERRORS IN DATA OR PROCESSES. ....	13
8. THE SYSTEM SHOULD NOT COMPROMISE OR HINDER ACADEMIC FREEDOM, SUCH AS BY LIMITING OR CENSORING ACADEMIC CONTENT, OR BY RESTRICTING ACCESS OR EXPRESSION BASED ON POLITICAL OR IDEOLOGICAL BIASES. ....	13
9. THE SYSTEM SHOULD NOT BE VULNERABLE TO HACKING, PHISHING, OR OTHER CYBER ATTACKS, SUCH AS BY USING WEAK OR OUTDATED SECURITY MEASURES, OR BY EXPOSING CRITICAL OR SENSITIVE DATA TO UNAUTHORIZED USERS OR SYSTEMS. ....	13
10. THE SYSTEM SHOULD NOT VIOLATE ANY APPLICABLE LAWS OR REGULATIONS, SUCH AS BY FAILING TO COMPLY WITH DATA PROTECTION, PRIVACY, OR ACCESSIBILITY REQUIREMENTS, OR BY ENGAGING IN FRAUDULENT OR DECEPTIVE PRACTICES. ....	13
3.6 DESIGN CONSTRAINTS .....	13
3.7 LOGICAL DATABASE REQUIREMENTS .....	14
3.8 OTHER REQUIREMENTS.....	14
<b>4. CHANGE MANAGEMENT PROCESS .....</b>	<b>15</b>

# 1. Introduction

The Academic Analytics System is a web-based application designed to provide academic institutions with a comprehensive platform for monitoring and analyzing student and faculty data. The system is intended to help administrators and faculty members make informed decisions about academic programs, policies, and strategies by providing them with access to a wide range of data and analytics tools. The system is intended to be user-friendly, scalable, and secure, with a flexible architecture that can accommodate future growth and development. This Software Requirements Specification (SRS) document outlines the functional and non-functional requirements of the system, as well as any constraints, assumptions, and dependencies that may impact its development. The document is intended to provide a clear and comprehensive description of the system's requirements, which will serve as a basis for its design, development, testing, and deployment.

## 1.1 Purpose

The purpose of this SRS document is to define the requirements for the development of an Academic Analytics System. The system will be designed to help academic institutions monitor and analyze student and faculty data in order to make informed decisions.

## 1.2 Scope

The Academic Analytics System will be a web-based application that will provide administrators and faculty members with access to various academic data and analytics tools. The system will include features such as data visualization, predictive analytics, and reporting tools.

## 1.3 Definitions, Acronyms, and Abbreviations

<b>AAS</b>	– Academic Analytics System
<b>FR</b>	- Functional Requirements
<b>NFR</b>	- Non-Functional Requirements
<b>UC</b>	- USE-CASE

## 1.4 References

- National Center for Education Statistics. (2021). Digest of Education Statistics, 2020 (NCES 2021-038). U.S. Department of Education, Institute of Education Sciences. <https://nces.ed.gov/pubsearch/pubsinfo.asp?pubid=2021038>
- Educause. (2019). The Top 10 IT Issues, 2019: The Drive to Digital Transformation Begins. <https://library.educause.edu/resources/2019/1/the-top-10-it-issues-2019-the-drive-to-digital-transformation-begins>

## 1.5 Overview

The system will support data visualization, predictive analytics, and reporting tools, and will provide users with a user-friendly, intuitive interface that allows for easy navigation and exploration of data. The AAS will be designed to be scalable, secure, and flexible, with a modular architecture that can be customized to meet the needs of individual institutions.

The following sections of this document will describe in detail the functional and non-functional requirements of the AAS, as well as any constraints, assumptions, and dependencies that may impact its development.

## 2. General Description

The Academic Analytics System (AAS) is a software system designed to provide academic institutions with a platform for monitoring and analyzing student and faculty data. The system is intended to assist academic administrators and faculty members in making data-driven decisions about academic programs, policies, and strategies.

### 2.1 Product Perspective

The AAS will be integrated with a variety of external data sources, including student information systems, learning management systems, and institutional research databases. The system will also include a variety of data analytics and reporting tools that will allow users to generate customized reports and visualizations based on the data in the system.

### 2.2 Product Functions

The AAS will be developed as a modular system, with a variety of different modules that can be customized and integrated to meet the specific needs of different academic institutions. Each

module will be designed to be independent and flexible, so that it can be modified or replaced without affecting the functionality of the overall system.

### 2.3 User Characteristics

The users of the Academic Analytics System (AAS) are primarily academic administrators and faculty members in higher education institutions. These users will have varying levels of technical expertise and will come from a variety of academic backgrounds, including but not limited to:

- Students
- Academic advisors
- Department chairs
- Deans
- Provosts
- Institutional researchers
- Faculty members

Some users may have experience with data analysis and reporting tools, while others may not. Therefore, the AAS should be designed to be user-friendly and intuitive, with a clear and intuitive user interface that allows users to easily access and interact with data and analytics tools.

Users of the AAS will need to have access to sensitive academic data, and will therefore need to be authorized by their respective institutions to use the system. The AAS should include access controls and audit logging features to ensure that only authorized users are able to access and modify data in the system.

Finally, the AAS should be designed to be scalable, so that it can accommodate the needs of institutions of different sizes and types. The system should be able to handle varying amounts of data and users, and should be designed to facilitate future growth and development.

### 2.4 General Constraints

The Academic Analytics System (AAS) must operate within certain constraints in order to ensure its successful development and implementation. The following constraints have been identified:

- **Technical constraints:** The AAS must be designed and developed using industry-standard technologies and best practices to ensure that it is reliable, scalable, and

maintainable. The system should be compatible with common web browsers and operating systems.

- **Time constraints:** The AAS must be developed and implemented within a specific timeframe in order to meet the needs of academic institutions. The development schedule should allow for adequate testing and quality assurance.
- **Budget constraints:** The development and implementation of the AAS must be completed within a specific budget. The system should be designed with cost-effectiveness in mind, and should not require significant ongoing maintenance or support costs.
- **Data constraints:** The AAS must adhere to data privacy regulations and guidelines, and should be designed to protect sensitive academic data. The system should also be designed to ensure data accuracy, completeness, and consistency.
- **Resource constraints:** The AAS development team will have limited resources, including personnel, hardware, and software. The system should be designed to make the most efficient use of available resources, and should be designed to facilitate future development and growth.
- **Legal constraints:** The AAS must comply with all relevant laws and regulations, including those related to data privacy, intellectual property, and accessibility. The system should also be designed to ensure that it is accessible to users with disabilities.

### 2.5 Assumptions and Dependencies

The successful development and implementation of the Academic Analytics System (AAS) is dependent on a number of assumptions and dependencies. These include:

- **Availability of data sources:** The AAS is dependent on the availability of relevant data sources, including student information systems, learning management systems, and

institutional research databases. The system will not be able to function without access to these data sources.

- **Access to hardware and software resources:** The AAS development team will require access to hardware and software resources, including development and testing environments, databases, and servers.
- **Availability of development resources:** The successful development of the AAS is dependent on the availability of skilled developers, testers, and project managers.
- **User participation:** The AAS is designed to be a user-driven system, and is dependent on the active participation of academic administrators and faculty members. The system will not be effective without user input and engagement.
- **Data privacy regulations:** The AAS is designed to handle sensitive academic data, and is dependent on compliance with relevant data privacy regulations and guidelines.
- **Availability of funding:** The development and implementation of the AAS is dependent on the availability of funding, both for development costs and ongoing maintenance and support costs.
- **Technical dependencies:** The AAS is dependent on a number of technical components, including web servers, databases, and analytics tools. These components must be compatible and integrate seamlessly in order for the system to function properly.

### 3. Specific Requirements

The Academic Analytics System (AAS) must meet the following specific requirements:

- 1. User Management:** The system must have a user management module to manage user accounts, roles, and permissions. The module must support the creation, modification, and deletion of user accounts.
- 2. Data Collection:** The system must be able to collect data from multiple sources, including student information systems, learning management systems, and institutional research databases. The data collection process must be automated and secure.
- 3. Data Processing:** The system must be able to process collected data to generate meaningful analytics and reports. The processing algorithms must be accurate, efficient, and scalable.
- 4. Analytics and Reports:** The system must be able to generate a wide range of analytics and reports based on processed data. The analytics and reports must be customizable, and must support multiple data visualizations.
- 5. Alerting:** The system must have an alerting module to notify users of important events or changes in data. The alerting module must be customizable, and must support multiple communication channels.

**6. Security:** The system must be designed to ensure the security of sensitive academic data. The system must adhere to relevant data privacy regulations and guidelines, and must implement appropriate access controls and encryption.

**7. Accessibility:** The system must be designed to ensure accessibility for all users, including those with disabilities. The system must adhere to relevant accessibility standards and guidelines.

**8. Performance:** The system must be designed to handle large volumes of data and users, and must be scalable and performant. The system must be able to handle multiple concurrent requests without significant performance degradation.

### 3.1 External Interface Requirements

The Academic Analytics System (AAS) must meet the following external interface requirements:

1. **User Interface:** The system must have a user interface that is easy to use, intuitive, and aesthetically pleasing. The user interface must be responsive, and must support multiple devices and screen sizes.
2. **Data Sources:** The system must be able to integrate with multiple data sources, including student information systems, learning management systems, and institutional research databases. The system must support standard integration protocols and APIs.
3. **Reporting Tools:** The system must be able to generate reports and analytics using third-party reporting tools. The reporting tools must be compatible with the system's data processing algorithms and data visualization requirements.
4. **Communication Channels:** The system must support multiple communication channels for alerts and notifications, including email, SMS, and push notifications.
5. **Accessibility Tools:** The system must support accessibility tools for users with disabilities, including screen readers, text-to-speech software, and keyboard shortcuts.
6. **Security Protocols:** The system must adhere to relevant security protocols for external interfaces, including encryption and authentication protocols.
7. **External Users:** The system must support external users, including students, parents, and other stakeholders. The system must adhere to relevant data privacy regulations and guidelines for external user data.

### 3.2 Functional Requirements

The Academic Analytics System (AAS) must meet the following functional requirements:

#### 3.2.1 User Management:

- 1.1. The system must allow administrators to create and manage user accounts with different roles and permissions.
- 1.2. The system must allow users to reset their passwords and update their account information.
- 1.3. The system must provide user activity logs for administrators to track user actions.

#### 3.2.2 Data Collection:

- 2.1. The system must be able to automatically collect data from multiple sources, including student information systems, learning management systems, and institutional research databases.

2.2. The system must support manual data entry and data imports from spreadsheets and other file formats. 2.3. The system must validate data to ensure accuracy and completeness.

### **3.2.3 Data Processing:**

- 3.1. The system must be able to process collected data using predefined algorithms and custom rules.
- 3.2. The system must be able to handle missing and incomplete data.
- 3.3. The system must generate error reports for incorrect data inputs.
- 3.4. The system must have the ability to normalize data across different sources.

### **3.2.4 Analytics and Reports:**

- 4.1. The system must be able to generate predefined analytics and reports on student performance, course effectiveness, and institutional effectiveness.
- 4.2. The system must support ad hoc analytics and reports based on user queries.
- 4.3. The system must provide a range of data visualizations, including charts, graphs, and tables.
- 4.4. The system must allow users to customize reports by selecting data and filtering options.

### **3.2.5 Alerting:**

- 5.1. The system must be able to send alerts to users based on predefined rules and thresholds.
- 5.2. The system must support different types of alerts, including email, SMS, and push notifications.
- 5.3. The system must allow users to configure alert preferences and frequency.

### **3.2.6 Security:**

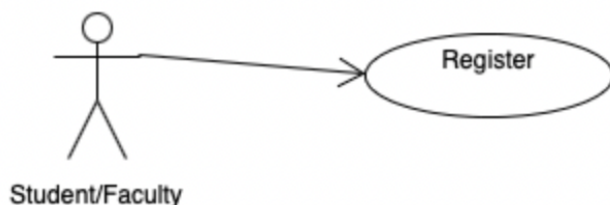
- 6.1. The system must enforce role-based access controls to ensure that users can only access data and functionality that are appropriate for their roles.
- 6.2. The system must support data encryption at rest and in transit.
- 6.3. The system must have backup and disaster recovery procedures in place.

### **3.2.7 Accessibility:**

- 7.1. The system must adhere to accessibility standards and guidelines, including Section 508 of the Rehabilitation Act and the Web Content Accessibility Guidelines (WCAG).
- 7.2. The system must provide alternative text descriptions for non-text content, and support keyboard navigation and screen reader compatibility.
- 7.3. The system must have a user interface that is easy to navigate and use for users with disabilities.

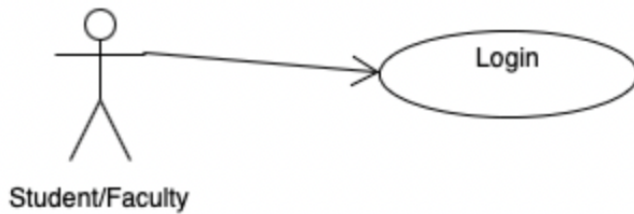
## **3.3 Use Cases**

### **3.3.1 Use Case #1 & #2**



**UC1&UC2.Student/Faculty registering for system:**

### 3.3.2 Use Case #3



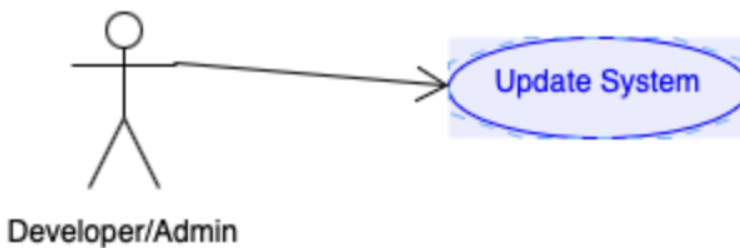
**UC3. Students/Faculty trying to login:**

### 3.3.3 Use Case #4



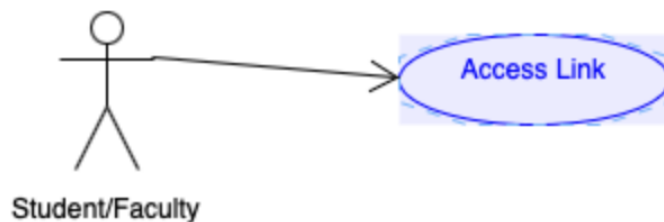
**UC4. Students/Faculty searching database for information:**

### 3.3.4 Use Case #5



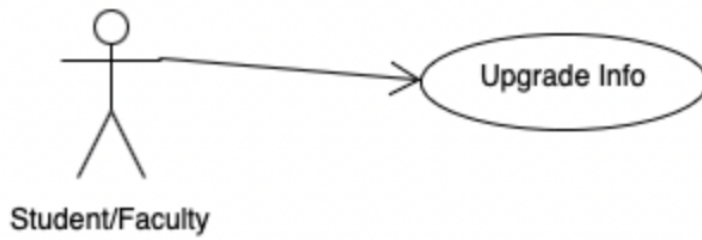
**UC-05: Developers/Admin trying to update the system:**

### 3.3.5 Use Case #6

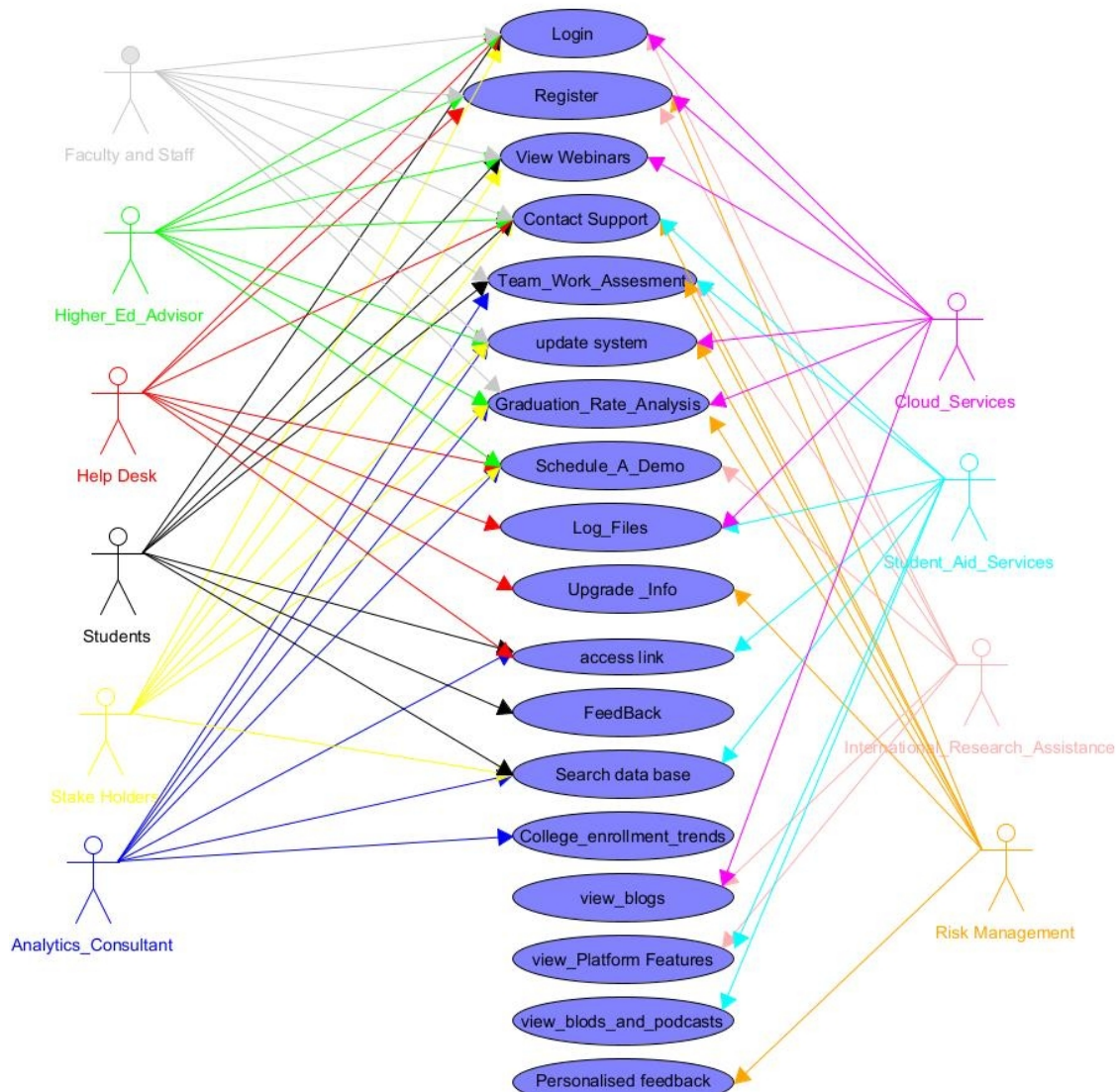


**UC-06: Students/Faculty trying to access a link:**

### 3.3.6 Use Case #7



**UC7.Faculty/students trying to upgrade personal info.**



**SAMPLE use-case diagram representing actors and functions in the AAS.**

**Actors: Below are some of the actors we have identified for our application.**

- Faculty and Staff
- Higher\_Ed\_Advisor
- Help Desk
- Students
- Stake Holders
- Analytics\_Consultant
- Cloud\_Services
- Student\_Aid\_Services
- International\_Research\_Assistance
- Risk Management

**Functions: Below are some of the functions done by the actors identified.**

- Login
- Graduate rate analysis
- View academic growth
- Prospect training plan
- Schedule a demo
- View webinars
- Contact support
- View platform features
- View blog and podcasts
- Track college enrollment Trends
- Teamwork Assessment
- Log files
- Personalized Feedback
- View student activity record
- Show past academic record

### **3.3.7 Implementing Traceability Matrix :**

#### **IDENTIFIED USE CASES FOR THE SYSTEM:**

- UC1.Students registering for system.
- UC2.Faculty registering for system.
- UC3.Students/Faculty trying to login.
- UC4.Students/Faculty searching database for information.
- UC5.Developers/Admin trying to update the system.
- UC6.Students/Faculty trying to access a link.
- UC7.Faculty/students trying to upgrade personal info.

#### **Functional Requirements**

- **FR-01 Access the Data:** Shall allow the students and faculty to retrieve the data from Distributed database using website.
- **FR-02 Upload the Data:** Shall allow users to upload the data to back up in distributed database using replication.

- **FR-03 Restore the Data:** Shall Allow users to back up the data to any storage as it is possible in dbms.
- **FR-04 Scan through Data:** Shall Allow the users to scan through the data.
- **FR-05 Update the Data:** Shall allow the users to update their personal information.

	UC-01	UC-02	UC-03	UC-04	UC-05	UC-06	UC-07
FR-01			#	#		#	#
FR-02	#	#					
FR-03					#		
FR-04				#			
FR-05					#		#

**For use case 01/02:**

Basic Flow – New student/faculty trying to register

Alternate Flow 2 – Already registered

Alternate Flow 3 – Invalid email id/password

<b>Scenario 1-successful registration</b>	<b>Basic flow</b>	
<b>Scenario 2- If he is already registered</b>	<b>Basic flow</b>	<b>Alternate flow 2</b>
<b>Scenario 3-If entered mail id/password is wrong</b>	<b>Basic flow</b>	<b>Alternate flow 3</b>

### 3.4 Non-Functional Requirements

Non-functional requirements are those that specify the qualities or attributes that the system must have, but that are not related to its functionality. Here are some possible non-functional requirements for an Academic Analytics System:

- 1. Usability:** The system should be easy to use and navigate, with clear and intuitive interfaces, minimal errors and confusion, and consistent feedback and help features.
- 2. Performance:** The system should be responsive and fast, with quick loading times, efficient processing of data, and minimal delays or downtime.
- 3. Security:** The system should be secure and reliable, with strong authentication and authorization measures, data encryption and backup, and protection against unauthorized access, theft, or damage.
- 4. Scalability:** The system should be able to handle increasing volumes of data, users, and transactions, without significant loss of performance or functionality.
- 5. Compatibility:** The system should be compatible with a range of hardware, software, and network configurations, and should be able to integrate with other systems or tools used by the academic institution.
- 6. Accessibility:** The system should be accessible to users with disabilities or special needs, with support for assistive technologies, alternative input/output methods, and compliance with relevant accessibility standards.
- 7. Reliability:** The system should be reliable and accurate, with minimal errors, bugs, or data inconsistencies, and with proper monitoring, testing, and debugging procedures.
- 8. Maintainability:** The system should be easy to maintain and update, with clear documentation, version control, and modular design, and with support for troubleshooting, bug fixing, and enhancement requests.
- 9. Legal and ethical compliance:** The system should comply with relevant laws, regulations, and ethical standards related to data privacy, security, and use, as well as with the policies and guidelines of the academic institution.
- 10. Performance metrics:** The system should be able to track and measure its own performance and usage, with relevant metrics and indicators, such as response time, usage patterns, and resource utilization.

### 3.5 Inverse Requirements

Inverse requirements are specifications that describe what the system should not do or have, rather than what it should do or have. They are also known as "negative requirements":

1. The system should not violate students' privacy rights, such as by collecting or sharing sensitive or personal data without their consent or inappropriately.
2. The system should not generate or display inaccurate or misleading data or insights, such as by using faulty or biased algorithms, incomplete or outdated data, or inappropriate assumptions or models.

3. The system should not cause harm or damage to users, data, or systems, such as by introducing viruses, bugs, or security breaches, or by overloading or disrupting network or computing resources.
4. The system should not discriminate or disadvantage any students or groups based on their gender, race, ethnicity, religion, sexual orientation, or other personal characteristics, such as by applying unfair or biased criteria or evaluations.
5. The system should not be used for any illegal or unethical purposes, such as by facilitating cheating, plagiarism, or academic misconduct, or by violating copyright or intellectual property rights.
6. The system should not be overly complex or confusing, such as by including unnecessary or redundant features, or by requiring extensive training or technical skills to use.
7. The system should not be incompatible or inconsistent with other academic or administrative systems used by the institution, such as by creating conflicts or errors in data or processes.
8. The system should not compromise or hinder academic freedom, such as by limiting or censoring academic content, or by restricting access or expression based on political or ideological biases.
9. The system should not be vulnerable to hacking, phishing, or other cyber attacks, such as by using weak or outdated security measures, or by exposing critical or sensitive data to unauthorized users or systems.
10. The system should not violate any applicable laws or regulations, such as by failing to comply with data protection, privacy, or accessibility requirements, or by engaging in fraudulent or deceptive practices.

### 3.6 Design Constraints

1. **Technology constraints:** The system should be designed to work with the available hardware, software, and networking technologies that are used by the institution, such as servers, databases, web browsers, and mobile devices. It should also be compatible with any future upgrades or changes in technology.
2. **Time constraints:** The system should be designed within the **time frame of six months** and budget allocated for its development, testing, and deployment should not exceed **\$100000**. It should also meet any milestones or deadlines set by the stakeholders, such as the start of a new academic year or semester.
3. **User interface constraints:** The system should be designed to be user-friendly and accessible to a wide range of users, including students, faculty, administrators, and support staff. It should also be designed to comply with any accessibility standards and guidelines, such as those defined by the **Web Content Accessibility Guidelines (WCAG)**.
4. **Data constraints:** The system should be designed to handle large and complex data sets, including student profiles, academic records, performance metrics, and other relevant

data. It should also be designed to ensure the accuracy, consistency, and integrity of the data, and to protect the data against loss, corruption, or unauthorized access.

5. **Security constraints:** The system should be designed to ensure the confidentiality, integrity, and availability of the data and the system itself. It should also be designed to comply with any security policies and standards, such as those defined by the ISO/IEC 27001 or the NIST Cybersecurity Framework.
6. **Performance constraints:** The system should be designed to operate efficiently and reliably under normal and peak loads, and to handle any exceptions or errors gracefully. It should also be designed to provide fast and responsive feedback to users, and to minimize any latency or delays in data processing or retrieval.
7. **Scalability constraints:** The system should be designed to accommodate future growth and expansion in terms of users, data, and functionality. It should also be designed to handle any changes or modifications in the system requirements, such as the addition of new features or the integration with other systems.

### 3.7 Logical Database Requirements

1. **Data entities:** The system should define the main data entities that are relevant to academic analytics, such as students, courses, instructors, departments, programs, and assessments. Each entity should have a set of attributes that capture the relevant information about it, such as name, ID, grade, enrollment status, etc.
2. **Data relationships:** The system should define the relationships between the different data entities, such as the association between students and courses, instructors and departments, programs and assessments, etc. The relationships should be represented by keys and foreign keys that ensure referential integrity and consistency.
3. **Data storage:** The system should store the data in a reliable and scalable database management system that can handle the expected data volumes and performance requirements. The database management system should support the necessary database operations, such as querying, indexing, and backup and recovery.
4. **Data migration:** The system should provide a mechanism for migrating the data from any legacy systems or sources to the new system, and for ensuring the consistency and accuracy of the migrated data. The migration process should be documented and tested thoroughly, and should minimize any downtime or disruption to the system.

### 3.8 Other Requirements

1. **Performance requirements:** The system should be able to handle the expected workload and user concurrency levels without significant degradation in performance. The system should be designed to minimize response time, processing time, and resource utilization, and should be scalable to accommodate future growth.
2. **Compatibility requirements:** The system should be compatible with the existing hardware and software environment, and should support various operating systems, web browsers, and devices. The system should also be designed to accommodate future changes or upgrades in the environment, such as new hardware or software releases.
3. **Maintenance requirements:** The system should be easy to maintain and support, and should enable efficient troubleshooting, debugging, and error handling. The system should provide adequate documentation, training, and support resources to enable administrators and users to operate and maintain the system effectively.

## 4. Change Management Process

The change management process is a set of procedures and guidelines that govern how changes are requested, evaluated, approved, implemented, and monitored within a system or organization.

1. **Request:** A change request is initiated by a stakeholder who identifies a need for a change to the system, such as a new feature, bug fix, or performance improvement.
2. **Evaluation:** The change request is reviewed by a change management team or a designated change manager who assesses the impact, feasibility, risks, and benefits of the proposed change. This may involve analyzing the requirements, design, code, and testing artifacts that will be affected by the change.
3. **Approval:** The change request is approved or rejected based on the evaluation results and the predefined criteria for change approval. The approval may require authorization from various stakeholders, such as the project manager, product owner, or system architect.
4. **Implementation:** The change is implemented by the development team, who modifies the system code, configuration, or data to incorporate the change. The implementation may involve testing, documentation, and deployment activities.
5. **Monitoring:** The change is monitored for its effectiveness, quality, and impact on the system's performance, reliability, and security. The monitoring may involve measuring the system's key performance indicators, analyzing user feedback, and conducting regression testing.

Some key considerations for a change management process include:

- Defining clear roles and responsibilities for the change management team, stakeholders, and developers.
- Establishing a change control board or committee to oversee the change management process and ensure consistency and transparency in the decision-making.
- Creating a change management plan or policy that outlines the procedures, criteria, and tools for requesting, evaluating, approving, implementing, and monitoring changes.
- Incorporating a change tracking and reporting mechanism to provide visibility and accountability for the change management activities.
- Ensuring that the change management process aligns with the system's quality, security, and compliance requirements, as well as any relevant standards or regulations.