
Software Requirements Specifications

for

BUP After-Admission Portal

Version 1.0 approved

Prepared by
2054901011 Anika Raisa
2054901037 Mahmuda Akter Metu
2054901042 Jannatul Ferdous Mirza
2054901056 Arin Rahman

Group-06
21-08-2022

Contents

| | | |
|----------|---|-----------|
| 1 | Introduction | 4 |
| 1.1 | Purpose | 4 |
| 1.2 | Document Convention | 4 |
| 1.3 | Intended Audience and Reading Suggestions | 5 |
| 1.4 | Product Scope | 5 |
| 1.5 | References | 6 |
| 2 | Overall Description | 6 |
| 2.1 | Product Perspective | 6 |
| 2.2 | Product Functions | 7 |
| 2.3 | User Classes and Characteristics | 7 |
| 2.4 | Operating Environment | 7 |
| 2.4.1 | Software Required: | 7 |
| 2.4.2 | Hardware Required: | 8 |
| 2.5 | Design and Implementation Constraints | 8 |
| 2.6 | User Documentation | 8 |
| 2.7 | Assumptions and Dependencies | 9 |
| 3 | External Interface Requirements | 9 |
| 3.1 | User interfaces: | 9 |
| 3.2 | Hardware Interfaces | 10 |
| 3.3 | Software Interfaces | 10 |
| 3.4 | Communication Interfaces | 11 |
| 4 | System Features | 11 |
| 4.1 | Marks Input Prior Viva | 11 |
| 4.1.1 | Description and Priority | 11 |
| 4.1.2 | Stimulus/Response Sequences | 12 |
| 4.1.3 | Functional Requirements | 12 |
| 4.2 | Marks Input Post Viva | 12 |
| 4.2.1 | Description and Priority | 12 |
| 4.2.2 | Stimulus/Response Sequences | 12 |
| 4.2.3 | Functional Requirements | 13 |
| 4.3 | Access Control (Administration) | 13 |
| 4.3.1 | Description and Priority | 13 |
| 4.3.2 | Stimulus/Response Sequences | 13 |

| | | |
|----------|---|-----------|
| 4.3.3 | Functional Requirements | 13 |
| 4.4 | Access Control (Student) | 14 |
| 4.4.1 | Description and Priority | 14 |
| 4.4.2 | Stimulus/Response Sequences | 14 |
| 4.4.3 | Functional Requirements | 14 |
| 4.5 | Subject Choice | 15 |
| 4.6 | Migration | 15 |
| 4.7 | Admin Auto Acknowledgement | 16 |
| 4.7.1 | Description and Priority | 16 |
| 4.7.2 | Response Sequence | 16 |
| 4.7.3 | Functional Requirements | 16 |
| 4.8 | Admission | 17 |
| 4.8.1 | Description and Priority | 17 |
| 4.8.2 | Response Sequence | 17 |
| 4.8.3 | Functional Requirements | 17 |
| 5 | Other Nonfunctional Requirements | 18 |
| 5.1 | Performance Requirements | 18 |
| 5.2 | Safety Requirements | 19 |
| 5.3 | Security Requirements | 19 |
| 5.4 | Software Quality attributes | 20 |
| 5.5 | Business Rules | 22 |

1 Introduction

1.1 Purpose

The students enter into an educational institute in the belief that they will be taught the requirements that need to be fulfilled when they want to get a job. For taking admission students, educational institutes should have a proper process of taking students on board. And that's where the role of the admission portal comes into the picture. The pandemic has taught us the importance of automation processes instead of the traditional system. The generation demands technology. Nowadays, most universities have mainly the Before-Admission portal which helps the students to apply. But the purpose of this After-Admission portal is the automation of admitting the students in order of the vacant seats. It will automate the process of any student canceling admission and will automatically notify the administration the seat is vacant, and then if the administration wants it can send a notification to the student that he/she can come and be admitted. So the purpose of this whole automation is the ease of the administration and students in the hassle-some time of admission.

1.2 Document Convention

1. Entire document is justified
2. Convention for Sections
 - Font Face: CM Roman
 - Font style: Bold
 - Font size: 16
3. Convention for subtitle
 - Font face: CM Roman
 - Font style: Bold
 - Font size: 14
4. Convention for running text
 - Font face: CM Roman
 - Font size: 12

1.3 Intended Audience and Reading Suggestions

Section 1 states the reason for using the software and the future scopes, which will attract the client. The overall description of the product is reviewed in section 2. It includes the functions and the designs of the software. In section 3, the hardware and software requirements are overviewed. In section 4 the most important part of the document is the features of the software that are briefly discussed. In sections 5 and 6, the functional and non-functional requirements are stated broadly to develop the software. Below are some guidelines for the readers of this document.

- **Project Manager:** This document is important for the project managers to understand the requirements of the client. Section 2,3,4,5 and 6 is important to review before getting on board with the project.
- **Developers:** Developers can be enlightened on what features the client wants, and develop it that wise. Sections 3,4,5 are a must for the core developers who are working directly on the project.
- **Marketing Staff:** The marketing staff can highlight the beneficial parts of the software and attract clients who might be interested in buying that software by going through subsections 1.1, 1.4, and section 4.
- **Users:** The users can have a clear guideline on how to use the software by reviewing subsections 2.2, 2.3, 2.4, 2.6, and section 4.
- **Testers:** Testers will test the software in terms of the requirements of the client and if the software fulfills the requirements or not. Testers can assure the validity of the software by going through subsections 2.2, 2.3, 2.4, and sections 3 and 4.
- **Documentation Writers:** Documentation writers can have a clear idea, visualize the product, and write a document about it clearly by going through subsections 1.1, 1.4, 2.1, 2.4, and sections 3 and 4.

1.4 Product Scope

A good admission portal and higher enrollment rates are directly related. An admission portal helps educational institutions make the enrollment process seamless and easy for the students to know the updates easily.

Out of the countless number of educational institutions out there, institutions need to have a system that is enough attractive and easy for a student to apply and migrate less complicatedly. Our after-admission portal helps to gain that attraction.

The mapping of the whole admission journey after giving the examination gets easy by using an after-admission portal. And by automation, it becomes so much easy for the administration to give admission and migrate the students tirelessly.

The after-admission portal creates an admin-friendly environment as it automates the whole process of sending notifications to the students if he/she is eligible to admit or migrate.

By using this software, students who want to be admitted can pay the admission fees online without standing in a long line waiting when their time will come.

1.5 References

E. Wieggers, K., 1999. SRS IEEE Template. 1st ed. [ebook] Available at: <https://classroom.google.com/c/NDk3MTYzMjU2MTk3/m/NTM3OTAwNjg1MzU2/details> [Accessed 20 August 2022].

2 Overall Description

2.1 Product Perspective

- The software is developed to aid the process of post-admission exams and the subject allocation of a student. The main aim of developing the software is to reduce the manual power required for the work of merit list preparation, subject allocation, and proper admission in each department according to respective seats.
- To reduce complexity in the system of subject allocation of a student from both administration and student sides.
- To manage student details in an orderly manner for proper allocation of seats according to merit, various criteria like merit and subject preference. Having all information in a sorted manner aids in reducing the time required for seat allocation which is crucial for all parties involved.

2.2 Product Functions

1. Stores the marks of individual sections from the admission exam to generate a merit list for viva.
2. Stores viva marks to generate a merit list for admission.
3. Contains student profiles of students who passed viva with their allocated subject.
4. Contains a migration system for a student to check if they are eligible to be migrated to a more demanded subject.
5. Provides facile and user interactive GUI.

2.3 User Classes and Characteristics

1. Administration

Administration-level users will input marks in their respective places to store them in the database for later processing. This class of users is also able to see the students that are eligible for the seats.

2. Student

Students who have passed the viva are able to log in to their user profiles using their admission roll number and see their allocated subject and apply for migration if needed. Finalizing admission can be done using the 'Get Admitted' option by paying the admission fees online.

2.4 Operating Environment

2.4.1 Software Required:

- (i) **Operating System:** Windows
- (ii) **Browser:** any updated browser
- (iii) **Front End tools:** Creative Tim, Envato HTML, Sublime Text, jQuery
- (iv) **Back End tools:** PHP, PHPStrom, Laravel
- (v) **Database:** SQL server

2.4.2 Hardware Required:

- (i) **CPU:** Intel Pentium VI processor
- (ii) **RAM:** minimum 512 MB
- (iii) **HDD:** At least 40 GB free space
- (iv) **Peripherals:** Keyboard, Monitor, Mouse, Printer

2.5 Design and Implementation Constraints

1. Students will only be able to migrate once.
2. Subject choice given by the student at first will be considered final.
3. Only administration users will be able to edit information and lists
4. Online payment for admission will be similar to the online semester fee payment system
5. Student profiles will only be available for students who have passed viva and accessible by a password sent to their respective phone numbers.

2.6 User Documentation

- **Login:** Administrator profiles will be pre-existing for login. Student profiles will only be available for login to the respective students after the publishing of viva marks and will be active till the admission process is finished. Admin profiles will always be existing.
- **Results:** The results of the written exam and viva-voce exam showing the eligible candidates for admission will be given on the home page.
- **Subject Choice:** eligible students after viva will be able to login to their profiles using a password that is sent to their respective phone numbers. In their profile, they will be allowed to give a subject list with the available subjects in a faculty according to their choice only once. This list will determine the demand of the subjects for each admission period.

- **Migration:** any student eligible for admission will be assigned a subject according to their merit. The subject given is dependent on the demand of that subject on that given admission period. When the migration option is on the student will be allocated to a subject with higher priority in their subject choice list if available. Migration checking is done only once.
- **Admission:** After subject allocation is done, a student can get admitted by paying admission fees online through selected agents like- tap, trust bank, bkaash etc.

2.7 Assumptions and Dependencies

1. Administrator is already created.
2. Student accounts will only be created after viva result publication and will be valid up to the completion of the whole admission process.
3. Roles and responsibilities are already established.

3 External Interface Requirements

3.1 User interfaces:

The following screens will be provided:

1. A login screen for entering a username, and password will be provided. Access to different screens will be based on the user.
2. There will be a screen for displaying information regarding entries by the institute.
3. There is a screen for displaying the name, date of birth, selecting programs, and grades in order for the student to choose their desired program/ to be able to migrate to their desired program.
4. There is a screen for submitted status for the students.
5. There is a screen for the admin at the server end displaying the information regarding entering the merit list for students who appeared for the admission test for allocating a subject to the student.

3.2 Hardware Interfaces

No additional hardware is required for the system. Basic hardware and resources for data communication are used, which include network connection at the server hosting site, network server, and network management tools. The admin will require standard hardware devices like a CPU(Central Processing Unit), 8 GB RAM (Random Access Memory), 40 GB HDD, monitor, keyboard, and a mouse to access the system. Moreover, a scanner is required to scan documents such as Statement of Results, Certificates, and documents supporting the benefit of the quota system for further verification and validation against the data that has been given as input manually. Whereas on the client side, a screen resolution of 800 X 600 is required or of better resolution to be able to view the web application.

3.3 Software Interfaces

On the server side, an operating system Windows 9x/XP or Windows ME or above is used.

Front-end software:

- Creative Tim

Creative Tim is used to incorporating Bootstrap-based design elements that will make the designing of the outlook of the web-based application feasible. It enables the user to get started with one of the pre-built pages making web designing less time-consuming, befitting a business model, etc. The admin templates are easy to use. The admin at the server end can save abundant time with the help of the admin dashboard. The use of pre-made sections enables the delivery of the web application faster.

- Envato HTML Templates

Envato HTML Templates include pre-made templates in accordance with Bootstrap, Laravel, etc. It provides a multiple file upload system particularly required for uploading multiple documents for each student signed up in the system such as Statement of Results, Grade Certificates, proof of Quota, etc. It also provides the developer with optimized CSS and Java Script codes, making coding less time-consuming.

- Sublime Text

This front-end development tool is used for simultaneous editing in multiple areas of the front-end code.

- **jQuery**

jQuery is a JavaScript Library that provides ease with HTML and Ajax. It can be used for free, making development cost-effective and its theme mechanisms help to build interactive web applications.

Back-end software:

The server-side scripting language used to build the web application in PHP. PHP offers detailed documentation, frameworks that are supported by multiple web browsers, and several readymade scripting codes.

Among the many PHP development tools, PHPStorm is chosen especially because it is befitting with the framework Laravel which is extensively used in this web application. The features of PHPStorm include code completion, code re-arranger, and support from multiple platforms such as Windows, Linux, MAC, etc.

Database: SQL server is used for the storage of data in tables.

3.4 Communication Interfaces

Students and the admin at the server end must connect to the internet to be able to access the website. A dial-up modem of standard capacity, broadband internet, and dial-up or broadband connection with a trusted Internet Service Provider(ISP) is required.

4 System Features

4.1 Marks Input Prior Viva

4.1.1 Description and Priority

This feature includes the process of inputting marks of each section i.e., English, General Knowledge, Math, Biology and the appropriate percentage of marks from HSC and SSC to prepare a merit list of students eligible for the viva. This feature is a high priority feature as the admission selection process starts from these marks.

4.1.2 Stimulus/Response Sequences

The administrators will input the marks in their respective sections and extra information fields like quota. The database will sort the list to determine a primary list of students eligible for attending the viva voce exam. This list will be published on the home page of the website.

4.1.3 Functional Requirements

- **REQ-01: Correctness of Information**

Information regarding marks must be correctly input in the system

- **REQ-02: Proper Software**

Must have access to SQL server and basic knowledge to use the server

- **REQ-03: Fill All Fields**

All number fields must be filled. For sections that are optional and left unanswered, -1 must be input. If any field is left empty, an error will be shown.

4.2 Marks Input Post Viva

4.2.1 Description and Priority

This feature will input viva marks to the prior data to sort the data. In this sorting the seats allocated for students with quota will be filled up first from a list sorted for only students with quota. Later the remaining seats will be filled by sorting all of the students. This is also a high priority feature as this feature is main point in selecting the students who are eligible for admission and allocating their subjects. The students will be allocated to their first-choice subject if seat is available, if not will be allocated to their second-choice subject and so on.

4.2.2 Stimulus/Response Sequences

The administrators will input the marks in the viva sections. The database will first only select and sort among students with quota and fill-up quota reserved seats in each department. After this part is done, all students will

be sorted regardless of their quota status and allocated seats in their first-choice department if there are seats available. If seats are not available, they will be allocated to their second-choice department and so on.

4.2.3 Functional Requirements

- **REQ-01: Correctness of Information**

Information regarding marks must be correctly input in the system

- **REQ-02: Proper Software**

Must have access to SQL server and basic knowledge to use the server

- **REQ-03: Fill All Fields**

All number fields must be filled. For sections that are optional and left unanswered, -1 must be input. If any field is left empty, an error will be shown.

4.3 Access Control (Administration)

4.3.1 Description and Priority

This feature separates the two user classes- administration and students. Administration accounts will be pre-existing for the admins to manage and oversee the input of marks, information, and results of each phase. This is a high priority feature as it separates between the two users and sensitive information.

4.3.2 Stimulus/Response Sequences

The administrators will input the marks in the defined sections. The result lists will be determined from these lists. The merit lists are seen and editable by the administration abiding by some restrictions.

4.3.3 Functional Requirements

- **REQ-01:Pre-existing account:**

Administration accounts will be created beforehand and will be accessible by admins all year long.

- **REQ-02: Ethical use of software:**

Admin users must not disclose their account information to ensure that only administrators have access to sensitive information

4.4 Access Control (Student)

4.4.1 Description and Priority

This feature is available only to students who are eligible for admission after passing the viva. The students can access their account using their respective ID no given for the admission exam of that faculty, passwords will be sent to their respective phone numbers provided during form fill-up. After logging in, students will see the subject they are allocated and have the option to apply for migration. After subject allocation is final, the option for admission will be given through which students will be able to make the payment for admission online by selected agents. This is also a high-priority feature as this allows the student to see, apply for a change in their subject, and get admitted online.

4.4.2 Stimulus/Response Sequences

The student will be required to log in to their account using their exam ID no and provided password to see their profile. Students will have the option to select the order of the subjects in the faculty according to their choice, but this can be done only once. If they turn on the migration option, they will not be able to use the ‘get admission’ feature and have to wait for the next subject allocation. After the finalization of the subject allocation, the ‘get admission’ button will be available and students will be able to pay the fees online to get admitted.

4.4.3 Functional Requirements

- **REQ-01:Account Access:**

Student accounts will only be created after the viva and only for students who have passed the viva. These accounts will be valid till the finalization of the admission process.

- **REQ-02: Proper Information:**

As students will be only given the chance to provide their order of subject choice, they are advised to do it with much caution.

4.5 Subject Choice

- **User End** After getting the final marks including viva and all, selected students will have to give subject choice according to their own priority. The most preferred subject will be allotted to the student after checking the vacancy of seats.
- **Admin End** The admins will be able to check the choice list given by the students and allow them in their respective departments according to priority. There will be two constraints for completing the procedure. One is checking the marks of students and another one is 'Quota'. After merging these two constraints, admin will finally provide subjects to each student.

4.6 Migration

- **Description and Priority** This feature enables the student to make a choice of their preferred subject despite being allocated with a subject prior to access to this feature. The student can only access this feature after successfully being allotted with a subject under the faculty. The individual will be able to make a choice of subject depending upon the topics answered in the admission test. The decision depends on the availability of seats, merit and quota basis. This feature has a priority of 9 on a scale of 10.
- **Response Sequence** The admin checks the student's choice against the topics answered in the admission test. For instance, a student who answers Mathematics instead of Biology is more likely to get allotted either ICE or CSE depending on the availability of seats, merit and quota. Once a migration request is made it can not be cancelled. The migration request can only be made for a specific amount of time right after the publication of results. No student is allowed to request for migration at any other time of the year.
- **Functional Requirements**

REQ-01: Internet Connection:

Internet connection is required for receiving updates.

REQ-02: Valid ID and Password:

For logging in to the software, the students must have their own ID and password that will be provided to them when they applied for the exam. The wrong ID and password will show "Invalid Login Attempt".

REQ-03:-Admin security key:

The admin side must have a security key by which they can be logged in and observe the process and edit information if needed.

4.7 Admin Auto Acknowledgement

4.7.1 Description and Priority

This feature introduces the automation of the admitting, migrating, and admission cancellation processes. If any student who is already admitted, cancels the admission, the admin panel will receive the notification that a seat is vacant and they can call the next student on the waiting list. This feature has a priority of 8 on a scale of 10.

4.7.2 Response Sequence

When a seat is vacant, the student on the 1st waiting list will get notified and be asked if he/she wants to be admitted or not. If that student accepts the admission he/she will be redirected to the next process. And if the student cancels the process, a notification will be sent to the next eligible student.

4.7.3 Functional Requirements

- **REQ-01: Internet Connection**

Internet connection is required for sending and receiving updates.

- **REQ-02: Valid ID and Password**

For logging in to the software, the students must have their own ID and password that will be provided to them when they applied for the exam. The wrong ID and password will show "Invalid Login Attempt".

- **REQ-03: Fill All Fields**

The admin side must have a security key by which they can be logged in and observe the process and edit information if needed.

4.8 Admission

4.8.1 Description and Priority

If the eligible candidate wants to continue the admission process, he/she will be directed to the payment page. And if the student wants to migrate to an upper subject he/she can keep the migration process on. And if any student wants to cancel the admission procedure, the notice of the vacant seat will be sent to the next eligible student.

4.8.2 Response Sequence

When the student clicks on the admit button, he/she will be directed to the payment page. On the payment page, there will be options given on which payment method the student wants to pay. There will be mobile banking and net banking options through which the student can pay and confirm admission. If any student wants to keep the migration on, he/she will have to click the migrate option, then a confirmation text will appear that the migration is turned on. And if someone cancels the admission, the seat will be vacant and notification will be sent to the next eligible student.

4.8.3 Functional Requirements

- **REQ-01: Internet Connection**

Internet connection is required for receiving updates.

- **REQ-02: Payment account**

For confirming the admission process the student must pay online for which, the student must have an account consisting of sufficient balance to transfer it to the university account.

- **REQ-03: Valid ID and Password**

For logging in to the software, the students must have their own ID and password that will be provided to them when they applied for the exam. The wrong ID and password will show "Invalid Login Attempt".

- **REQ-04: Admin security key**

The admin side must have a security key by which they can be logged in and observe the process and edit information if needed.

5 Other Nonfunctional Requirements

5.1 Performance Requirements

The execution of computing technologies and the accessibility of sufficient hardware resources are prerequisites for software performance. Performance bottlenecks can develop when access to shared resources is disputed and cause delays in function execution or increases in function execution time. The mix of functions must deliver the performance necessary to meet the operational needs. The operational performance parameters set by the operational requirements must therefore be satisfied by the functional solution. The performance requirements given for the software product must be satisfied by the functional timing and resource use of the functional solution. Response time, throughput, and resource utilization levels that satisfy predetermined performance targets are the goals of software performance engineering. The performance of the software is influenced by the features of the computing environment. The following, among others, are elements of the computing environment that must be taken into account while defining software architecture:

1. Execution time
2. Memory utilization
 - a. Primary memory (random-access memory, RAM) consumption
 - b. Virtual memory (secondary storage) consumption
3. Swap time (virtual memory management read and write latency)
4. Data storage latency (the time it takes to access a particular location in storage)
5. Data storage throughput (the rate at which information can be read from or written to the storage)
6. Interrupt latency (the time between the generation of an interrupt by a device and the servicing of the device)

5.2 Safety Requirements

Identification of protective needs that make sure that system failures do not result in harm, death, or environmental damage is the aim of safety requirements. Safety standards may be "must not" requirements, which means that they outline circumstances and occurrences that are never acceptable. The terms "checking and recovery features that should be included in a system" and "features that provide protection against system failures and external attacks" are used to define functional safety standards.

5.3 Security Requirements

Identification of protective needs that make sure that system failures do not result in harm, death, or environmental damage is the aim of safety requirements. Safety standards may be "must not" requirements, which means that they outline circumstances and occurrences that are never acceptable. The terms "checking and recovery features that should be included in a system" and "features that provide protection against system failures and external attacks" are used to define functional safety standards.

- **Hazard Identification:** Identify the hazards that may threaten the system. Hazard identification may be based on different types of hazards: physical, electrical, biological, service failure, etc.
- **Hazard Assessment:** The process is concerned with understanding the likelihood that a risk will arise and the potential consequences if an accident or incident should occur. Risks may be categorized as: intolerable (must never arise or result in an accident), as low as reasonably practical - ALARP (must minimize the possibility of risk given cost and schedule constraints), and acceptable (the consequences of the risk are acceptable and no extra costs should be incurred to reduce hazard probability). The acceptability of a risk is determined by human, social, and political considerations. In most societies, the boundaries between the regions are pushed upwards with time i.e. society is less willing to accept risk (e.g., the costs of cleaning up pollution may be less than the costs of preventing it but this may not be socially acceptable). Risk assessment is subjective. Hazard assessment process: for each identified hazard, assess hazard probability, accident severity, estimated risk, and acceptability.

- **Hazard Analysis:** Concerned with discovering the root causes of risks in a particular system. Techniques have been mostly derived from safety-critical systems and can be: inductive, bottom-up: start with a proposed system failure and assess the hazards that could arise from that failure; and deductive, top-down: start with a hazard and deduce the causes of this could be. The fault-tree analysis is a deductive top-down technique.: Put the risk or hazard at the root of the tree and identify the system states that could lead to that hazard. Where appropriate, link these with 'and' or 'or' conditions. A goal should be to minimize the number of single causes of system failure.
- **Risk Reduction:** The aim of this process is to identify dependability requirements that specify how the risks should be managed and ensure that accidents/incidents do not arise. Risk reduction strategies: hazard avoidance; hazard detection and removal; damage limitation.

5.4 Software Quality attributes

Software quality attributes, which include qualities like availability, interoperability, correctness, reliability, learnability, robustness, maintainability, readability, extensibility, testability, efficiency, and portability, make it easier for software testing experts to gauge how well a software product performs. Software architects can ensure that a software program will function in accordance with the client's demands thanks to high scores in the software quality attributes.

- **Availability:** This characteristic provides insight into how well an application will carry out the duties that are given to it. A few ideas related to software security, performance, integrity, reliability, dependability, and confidentiality are also included in the definition of availability. Additionally, top-notch availability means that any operating errors in a software-driven systems process aim service outage times do not go beyond a certain time limit.
- **Interoperability:** To complete some tasks, software-driven systems may need to collaborate and communicate with one another. The capacity of two systems to communicate information through specific interfaces is referred to as interoperability. The interoperability attribute

must therefore be examined in terms of both syntactic and semantic interoperability by Software Quality Assurance engineers.

- **Performance:** This characteristic relates to a software-driven system's capacity to adhere to timing specifications. It indicates that from the perspective of testing, software testing engineers must determine if the system reacts to various events within predetermined time frames. These occurrences could take the shape of clock events, process interruptions, messages, user requests, and more.
- **Testability:** Software testability measures how well a software-driven system enables testers to run tests by predetermined criteria. This characteristic evaluates how simple it is for software quality assurance engineers to create test specifications for a given system and all of its constituent parts. Engineers can evaluate a system's testability using a variety of methods, including encapsulation, interfaces, patterns, reduced coupling, and more.
- **Security:** This characteristic evaluates how well a system can detect and prevent nefarious or illegal acts that can endanger the system. The quality assumes significance since security refers to a system's capacity to safeguard data and defend information against unauthorized access. Techniques for authorization and authentication, defense against network intrusions, data encryption, and other dangers are all included in security. It is crucial for specialists and software testing companies to periodically perform updated security checks on systems.
- **Usability:** Every software-driven system is created to be simple to use to do specific tasks. Usability is a quality that describes how simple it is for users to complete activities on a system and what kind of user support the system offers. KISS is the most well-known tenet for this property (Keep It Simple Stupid). Additionally, software must be tested by software quality assurance engineers to see if it supports various forms of accessibility for users with impairments. Usability has a significant and long-standing impact on a software application's or package's commercial success.
- **Functionality:** This characteristic assesses how closely a software-driven system adheres to actual requirements and specifications. In

the early phases of software testing activities, most software testing professionals would advise performing tests that evaluate the expected functionality of a system because they see this quality as being vital and a top necessity of a modern application.

5.5 Business Rules

- **Strive for consistency**

Similar scenarios should call for the same set of actions, the same terminology should be used in prompts, menus, and help screens, and the same color scheme, layout, capitalization, typefaces, and other design elements should be utilized throughout. Exceptions should be understandable and few in number, such as requiring confirmation of the delete command or not echoing passwords.

- **Offer informative feedback**

There should be a UI feedback for each user activity. The response can be small for regular and minor acts, but it should be more substantial for major and infrequent actions. It is practical to depict changes explicitly in a visual representation of the things of interest.

- **Prevent errors**

Design the user interface as carefully as you can to prevent users from making major mistakes. For instance, menu options that are inappropriate should be hidden, and numeric entry areas shouldn't accept alphabetic letters. If users make a mistake, the interface should provide clear, helpful, and detailed recovery instructions. Users should be directed to fix only the problematic portion of a name-address form, for instance, rather than being forced to retype the entire form if they provide an erroneous zip code. When something goes wrong, the interface should either leave the state unaltered or provide advice on how to change it back.

- **Keep users in control**

Experienced users have a great desire for the feeling that they control the interface and that it reacts to their activities. They dislike shocks and unexpected changes in behavior, and they find laborious data entry processes, challenges locating relevant data, and failure to deliver intended results annoying.

- **Reduce short-term memory load**

Designers should steer clear of interfaces that require users to remember information from one display and then use that information on another display due to humans' limited capacity for information processing in short-term memory (the rule of thumb is that people can remember "seven plus or minus two chunks" of information). It means that lengthy forms should be condensed to suit a single display, phone numbers shouldn't need to be entered again for smartphones, and website locations should always be accessible

For each setting, these underlying concepts need to be adjusted, improved, and expanded. Although they have several drawbacks, they provide mobile, desktop, and online designers a decent place to start. The concepts discussed in the following sections are geared toward enhancing users' productivity by delivering clear displays, easy-to-understand interfaces, and quick informational feedback to boost user confidence in their ability to use the system.

- **Permit easy reversal of actions**

Actions should be as reversible as feasible. Because users are aware that mistakes may be corrected, this feature reduces fear and stimulates the investigation of novel choices. A single action, a data-entry task, or an entire series of actions, such as entering a name-address block, can all be considered units of reversibility.

- **Design dialog to yield closure**

Action sequences need to be grouped into units with a beginning, middle, and end. Users have a sense of success, relief, a signal to let go of backup plans, and an indication to get ready for the next group of tasks after receiving informative feedback after completing a set of actions. For instance, e-commerce websites guide consumers through the check-out process from choosing products to receiving a clear confirmation page that seals the deal.

- **Seek universal usability**

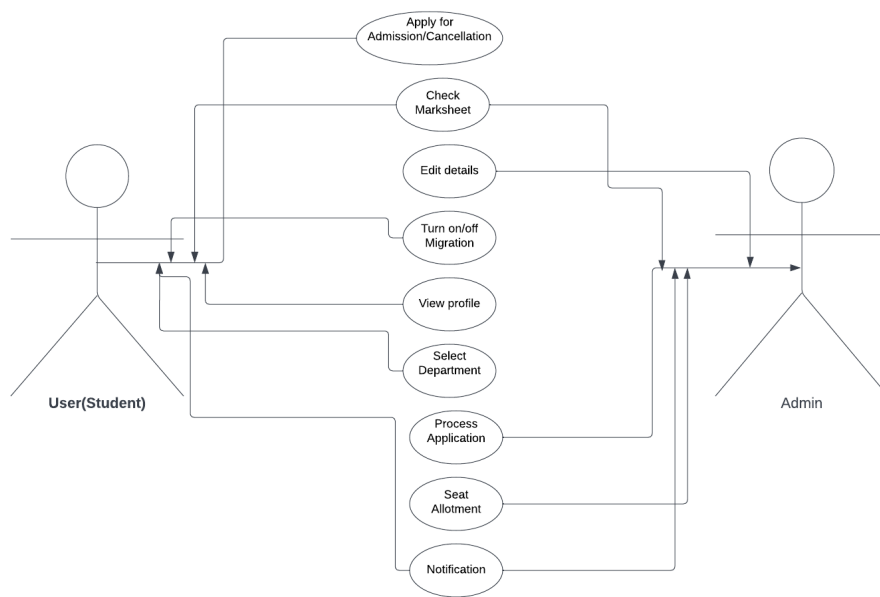
Recognize the requirements of various users and design with elasticity in mind to enable content change. Age ranges, disabilities, international disparities, and technological diversity all add to the breadth of requirements that serve as the basis for design. The interface design is

enhanced and its perceived quality is raised by including features for both beginners and professionals, such as explanations and shortcuts.

Appendix A: Glossary

- **Academic Program:** The student's area of academic interest falls under the umbrella term "academic program." Students are considered to be the "property" of the academic program they apply to, are accepted into, and eventually graduate from.
- **AP:** AP or Accounts Payable module within the People Soft Finance System. Will be used for issuing student refunds and third-party sponsor refund checks.
- **Department:** The students who will get selected finally, have to select a department out of their priority. According to their marks and quota preference, they will be allotted to distinct departments.

Appendix B: Analysis Models



**Figure 1:Use Case Diagram
for BUP Admission Portal**

Figure 1: Use Case Diagram