Table of Contents

[**USE CASES (UNIVERSITY ADMISSION SYSTEM):** 3](#_Toc131544524)

[Brief Use Cases: 3](#_Toc131544525)

[Registration: 3](#_Toc131544526)

[Login: 3](#_Toc131544527)

[Academic Information Intake: 4](#_Toc131544528)

[View Eligibility Criteria: 4](#_Toc131544529)

[Apply for Admission: 5](#_Toc131544530)

[Apply for Test: 5](#_Toc131544531)

[Enter Personal Info: 6](#_Toc131544532)

[Enter Academic Info: 6](#_Toc131544533)

[Verify Documents: 7](#_Toc131544534)

[Notify (Advertisement): 7](#_Toc131544535)

[Announcement: 8](#_Toc131544536)

[Scheduling Test: 8](#_Toc131544537)

[Print Challan: 9](#_Toc131544538)

[Make Test Payment: 9](#_Toc131544539)

[Take Test: 10](#_Toc131544540)

[Publish merit list: 10](#_Toc131544541)

[Check merit list: 11](#_Toc131544542)

[Make Admission Payment: 11](#_Toc131544543)

[Admission Confirmation: 12](#_Toc131544544)

[Admin view: 12](#_Toc131544545)

[Logout: 13](#_Toc131544546)

[Fully Dressed Use Cases: 13](#_Toc131544547)

[Login: 13](#_Toc131544548)

[Apply for Admission: 14](#_Toc131544549)

[Apply for Test: 16](#_Toc131544550)

[Enter Personal Info: 17](#_Toc131544551)

[Enter Academic Info: 19](#_Toc131544552)

[Verify Document: 19](#_Toc131544553)

[Announcement: 20](#_Toc131544554)

[Notify (Advertisement): 22](#_Toc131544555)

[Make Test Payment: 23](#_Toc131544556)

[Scheduling Test: 24](#_Toc131544557)

[Take Test: 26](#_Toc131544558)

[Print Challan: 27](#_Toc131544559)

[Make Admission Payment: 29](#_Toc131544560)

[Admission Confirmation: 30](#_Toc131544561)

[Publish Merit List: 31](#_Toc131544562)

[Check Merit List: 32](#_Toc131544563)

[Admin View: 33](#_Toc131544564)

[Log out: 35](#_Toc131544565)

**Assigned Use cases to Group Member:**

|  |  |  |
| --- | --- | --- |
| Name | Reg No | Assign Use cases |
| Hamza badar | Fa21-bse-055 | 1.View admin,  2.Published Merit list  3. Check Merit List |
| Mahad Wajid | Fa21-bse-057 | 1.Enter Personal info  2.Announcement  3.Scheduling test |
| Abdul Aziz | Fa21-bse-058 | 1.Take Test  2. Print Challan  3. Make Test Payment  4.Make Admission payment |
| Soman Ahmed | Fa21-bse-150 | 1.Logout,  2.Academic info  3.verify document |
| Danyal Nawaz | Fa21-bse-083 | 1.Login  2.Apply for test  3. Apply For Admission |
| Shah Hussain | Fa21-bse-172 | 1.Notify  2.Admission Confirmation |
| Ajwa Sardar | Fa21-bse-072 | 1.Registration  2.Arrange Test  3.Admission Criteria |

# **USE CASES (UNIVERSITY ADMISSION SYSTEM):**

## Brief Use Cases:

### Registration:

1. Capturing Basic Information: The registration page collects basic information about the prospective student, such as their name, date of birth, contact details, and educational background.
2. Application Management: The registration page also allows students to manage their application status, view their application progress, and upload supporting documents.
3. Payment Processing: Students may be required to pay an application fee during the registration process, and the registration page provides a secure payment gateway to process the payment.
4. Communication: The registration page may also allow students to communicate with the admission office and receive updates on their application status.
5. Admissions Criteria: The registration page may provide information on the admission criteria, including the required academic qualifications, standardized test scores, and any other requirements necessary for admission.

Application Deadline: The registration page may also display the application deadline, reminding prospective students to complete their application before the deadline

### Login:

1. User opens the admission system login page and enters their credentials (username and password).
2. The system verifies the user's credentials and checks if the user is authorized to access the admission system based on their role and access level.
3. If the credentials are correct and the user is authorized, the system logs the user into the admission system and displays the appropriate dashboard or landing page based on the user's role.
4. If the credentials are incorrect, the system displays an error message and prompts the user to enter their credentials again or reset their password if necessary.
5. Once logged in, the user can perform various actions such as submitting applications, reviewing applications, managing student records, etc. based on their role and level of access.
6. If the user logs out of the system or if their session times out, the system logs the user out and redirects them to the login page to re-authenticate.

### Academic Information Intake:

1. Collecting Academic Information: The academic information intake form collects details of the prospective student's academic qualifications, including their GPA, standardized test scores, and transcripts.
2. Course Preferences: The page may also ask the student to provide their course preferences and select the program they are interested in pursuing.
3. Additional Information: The academic information intake form may also collect additional information, such as extracurricular activities, work experience, and personal statements.
4. Admission Eligibility: Based on the information provided, the system can determine the student's eligibility for admission to the program they have applied for.
5. Academic Evaluation: The information collected on this page can be used to evaluate the academic profile of the student and compare it with other applicants.
6. Application Processing: The academic information intake form provides the admission office with the necessary information to process the application and make a decision on admission.
7. Scholarship and Financial Aid: The page may also ask the student to provide details of any scholarship or financial aid they may require to pursue their studies.

### View Eligibility Criteria:

1. Clear Understanding of Admission Requirements: The eligibility criteria page provides a clear understanding of the academic qualifications and other requirements necessary to be eligible for admission to a particular program.
2. Standardized Tests: The page may specify the standardized tests required for admission, such as SAT, ACT, GRE, or GMAT, and the minimum score required.
3. Course Prerequisites: The eligibility criteria page may also specify the course prerequisites that the prospective student must have completed in high school or college.
4. Language Proficiency: The page may specify the language proficiency requirements for international students, such as TOEFL, IELTS, or other language proficiency tests.
5. Residency Requirements: The eligibility criteria page may specify the residency requirements for admission, such as the state or country of residence, citizenship, or visa status.
6. Age Limit: The eligibility criteria page may specify the age limit for admission, especially for programs that have an age limit.
7. Program Specific Criteria: The eligibility criteria page may also provide specific criteria for certain programs, such as work experience, portfolio, or auditions.
8. Avoiding Wasted Applications: The eligibility criteria page helps prospective students avoid wasting time and effort on applications if they do not meet the admission requirements.

### Apply for Admission:

1. Prospective student opens the admission system application page and starts filling out the application form.
2. The system prompts the student to enter their personal information, educational background, test scores, and other required information.
3. The system validates the student's application and prompts them to review and verify the accuracy of the information entered.
4. Once the student has reviewed and verified their application, they submit the application to the university.
5. The system sends a confirmation email to the student, acknowledging receipt of the application and providing them with further instructions and updates about the application process.
6. The university admission office receives the application and reviews it to determine the student's eligibility for admission.
7. The admission office may request additional information or documents from the student to complete the application process.
8. The admission office makes a decision on the application, and the system sends an email notification to the student about the decision.
9. If the student is accepted, the system provides them with further instructions on how to enroll in the university.
10. If the student is rejected, the system provides them with information on the reasons for the rejection and any available options for appeal or reapplication.

### Apply for Test:

1. Prospective student logs in to the admission system and opens the apply for test page.
2. The system prompts the student to select the type of test they wish to take (e.g., SAT, ACT, GRE, etc.).
3. The system displays the available test dates, locations, and times for the selected test.
4. The student selects their preferred test date, location, and time.
5. The system prompts the student to pay the test registration fee.
6. The student pays the fee through the admission system's integrated payment gateway.
7. The system confirms the student's test registration and sends them an email confirmation with the test details and instructions on what to bring to the test center.
8. On the day of the test, the student arrives at the test center and presents the necessary identification and materials as instructed by the admission system.
9. The student takes the test and receives their score report through the admission system.
10. The university admission office receives the student's test scores and uses them as part of the application evaluation process.

### Enter Personal Info:

1. **Application processing**: Personal information is required to process and evaluate applications for admission. This includes basic information such as name, address, contact details, and academic records.
2. **Communication:** Personal information such as email address and phone number is needed to communicate with student about their application status, interviews, and other related matters.
3. **Financial aid:** Personal information is used to determine eligibility for financial aid, scholarships, and other forms of financial assistance. This includes information about income, assets, and family background.
4. **Accommodation**: Personal information is used to determine accommodation options for students, including preferences for living arrangements and any special needs.
5. **Health and safety:** Personal information is required for health and safety purposes, such as medical history, emergency contacts, and insurance information.
6. **Diversity and inclusion**: Personal information is used to promote diversity and inclusion in the university community, such as gathering information about an applicant's race, ethnicity, or socioeconomic background.

### Enter Academic Info:

1. The user logs in to the university admission system using their username and password.
2. The user navigates to the "Academic Info" section of the system.
3. The system displays the user's academic record, which may include information such as courses taken, grades received, and degree progress.
4. The user may have the ability to update their academic information or view additional details by clicking on links or buttons within the system.
5. The user may also be able to access information related to academic requirements, such as degree program requirements, course prerequisites, or graduation requirements.
6. The user may be able to download or print their academic information for their own records or to provide to others.
7. If the user encounters any issues or discrepancies in their academic information, they may be able to request assistance or submit a support ticket to the university administration.

### Verify Documents:

1. Prospective students submit their application online and upload scanned copies of their academic transcripts, certificates, and other relevant documents.
2. The admission system receives the documents and sends them to a document verification service for validation.
3. The document verification service checks the documents against its database of known valid documents and employs various security measures such as image analysis and watermark detection to ensure their authenticity.
4. The verification service then generates a report indicating whether each document is verified or not and the level of confidence in the verification.
5. The admission system receives the verification report and uses it to determine the eligibility of the prospective students.
6. If a document is found to be invalid or inaccurate, the admission system can reject the application or request the applicant to provide additional supporting documents.

### Notify (Advertisement):

1. **Admission Offer Notifications:** The university admission system can notify applicants of their admission offer, which will contain information about the program they have been accepted into, the start date, and any other important information.
2. **Rejection Notifications:** If an applicant's admission is rejected, the university admission system can notify them with an explanation of why they were not accepted and any other helpful information about next steps.
3. **Waitlist Notifications**: If an applicant is placed on a waitlist, the university admission system can notify them of their status and provide them with any additional information they may need to know.
4. **Deadline Reminders**: The university admission system can also send reminders to applicants about upcoming deadlines, such as the deadline to accept an offer of admission or submit any required documents.
5. **Scholarship Notifications**: If a scholarship is awarded, the university admission system can notify the applicant of the award amount and any additional information they may need to know.
6. **Orientation Information:** The university admission system can also notify admitted students of upcoming orientation dates and any other relevant information that will help them prepare for their first day of classes.

### Announcement:

1. **Application deadlines:** Universities may use announcements to inform prospective students of application deadlines, including early decision or early action deadlines, regular decision deadlines, and any applicable extension deadlines.
2. **Admission decision notifications:** Universities may use announcements to inform applicants of admission decisions, including acceptance, rejection, or waitlist status. These notifications may include instructions on next steps for admitted students, such as submitting a deposit or registering for classes.
3. **Scholarship and financial aid announcements:** Universities may use announcements to inform students of scholarship and financial aid opportunities, including deadlines for submitting applications and instructions for how to apply.
4. **Open house and campus visit announcements:** Universities may use announcements to promote campus visits and open house events, providing information on dates, times, and registration details.
5. **Program updates and changes:** Universities may use announcements to inform students of changes or updates to academic programs or requirements, as well as new program offerings or changes to admission requirements.
6. **Important dates and deadlines**: Universities may use announcements to highlight important dates and deadlines related to the admission process, such as when transcripts or test scores are due, or when students can expect to receive their admission decision.

### Scheduling Test:

1. **Standardized testing:** Many universities require applicants to take standardized tests such as the SAT, ACT, GRE, or GMAT. Test schedules are used to inform applicants of upcoming test dates, registration deadlines, and other important information related to the testing process.
2. **Subject-specific testing:** Some universities may require students to take subject-specific tests in addition to standardized tests. Test schedules are used to inform applicants of the specific tests required, the dates and locations of the tests, and any registration deadlines.
3. **Language proficiency testing:** For international students, universities may require proof of English proficiency through tests such as TOEFL or IELTS. Test schedules are used to inform students of the testing process, including available test dates, registration deadlines, and test fees.
4. **Accommodation for test takers:** Test schedules are used to ensure that appropriate accommodations are provided for test takers with disabilities or other special needs, such as extended testing time or special testing conditions.
5. **Test score submission deadlines:** Universities may have specific deadlines for receiving test scores from applicants. Test schedules are used to inform students of these deadlines, allowing them to plan accordingly and ensure that their scores are submitted on time.
6. **Alternate testing arrangements:** In some cases, universities may offer alternate testing arrangements for students who are unable to take the test on the scheduled date, such as makeup exams or remote testing options. Test schedules are used to inform students of these options and the associated deadlines and requirements.

### Print Challan:

1. The Student or Guest logs into the university admission system using their credentials.
2. The system calculates the fee based on the Student or Guest program and any applicable discounts or scholarships.
3. The system generates a unique challan number and associates it with the Student or Guest account.
4. The system presents the challan to the Student or Guest, including the fee amount, the due date, and the payment instructions.
5. The Student or Guest can choose to print the challan or save it as a PDF.
6. If the Student or Guest chooses to print the challan, the system generates a formatted document with the necessary information, including the challan number and a barcode for tracking.
7. The Student or Guest then proceeds to pay the fee using the specified payment method.

### Make Test Payment:

1. The Guest logs into the university admission system using their credentials and selects the option to initiate a test payment.
2. The system presents the available test payment methods, such as test credit card or test bank transfer.
3. The Guest selects the preferred test payment method and enters the required test payment information, such as test card number or test bank account number.
4. The system simulates the payment processing and verification and provides the Guest with a response, such as success or failure, along with any error codes or messages.
5. The Guest can then analyze the response and make any necessary changes to the payment processing and verification functionalities in the code.
6. Once all payment functionalities are completed, the system shows notification message to the guest as a confirmation.

### Take Test:

1. The Guest logs into the university admission system using their credentials.
2. The system presents the available tests and the Guest selects the one they want to take.
3. The system checks if the Guest meets the prerequisites for the selected test, such as Fsc marks, and documents verification.
4. If the Guest meets the prerequisites, the system presents the test schedule to the Guest.
5. The Guest takes the test, answering the questions presented to them.
6. The Guest submits the completed test to the test proctor and waiting for results.

### Publish merit list:

1. **Generate Merit List:** This use case involves generating the merit list of admitted students based on the evaluation results. The system sorts the applications based on the criteria specified by the university and creates a list of students who have been admitted to the university.
2. **Notify Admitted Students:** This use case involves notifying the admitted students about their acceptance to the university. The system sends out notifications to the email addresses or phone numbers provided by the students, informing them of their admission status and any further steps they need to take.
3. **Publish Merit List:** This use case involves publishing the merit list on the university website. The system creates a webpage or PDF document containing the list of admitted students and makes it accessible to the public.
4. **Update Student Records:** This use case involves updating the student records in the university's student information system. The system adds the admitted students to the database, assigns them a student ID, and updates their personal and academic information.
5. **Manage Waitlist:** This use case involves managing the waitlist of students who have not yet been admitted. The system adds the remaining students to a waitlist and notifies them of their waitlist status. As spots become available, the system evaluates the students on the waitlist and admits them if they meet the admission criteria.
6. **Review Merit List:** This use case involves reviewing the merit list for accuracy and completeness. The system allows the admin to review the list and make any necessary corrections or adjustments.

### Check merit list:

1. **View Merit List:** This use case involves allowing users to view the published merit list of the admitted students. The system displays the merit list on the university website or provides a link to download the list in a readable format.
2. **Search Merit List:** This use case involves allowing users to search the merit list for specific students or criteria. The system allows users to search the merit list based on student name, program, or other relevant criteria.
3. **Check Application Status:** This use case involves allowing users to check the status of their application. The system retrieves the user's application status from the admission database and displays it to the user.
4. **Notify Waitlisted Students:** This use case involves notifying waitlisted students of any changes in their status. The system sends out notifications to the email addresses or phone numbers provided by the students, informing them of any updates to their waitlist status.
5. **Send Acceptance Letter:** This use case involves sending an acceptance letter to the admitted students. The system generates an acceptance letter with the relevant details and sends it to the admitted student's email or physical address.
6. **Contact Admission Office:** This use case involves allowing users to contact the admission office for any questions or concerns regarding their application or admission status. The system provides contact information for the admission office, such as phone numbers and email addresses.

### Make Admission Payment:

1. The Student or Guest logs into the university admission system using their credentials.
2. The system presents the Student or Guest with the total amount due for admission fees, along with any applicable taxes or fees.
3. The Student or Guest selects the preferred payment method, such as credit card, bank transfer, or cash.
4. The Student or Guest enters the required payment information, such as card details or bank account number.
5. The system verifies the payment information and processes the payment.
6. If the payment is successful, the system generates a payment receipt and associates it with the Student or Student or Guest account.
7. The system updates the Student or Guest account to indicate that the payment has been received and processed and that the Student or Guest has been officially admitted.
8. If the payment is unsuccessful, the system presents an error message and allows the Student or Guest to try again or select a different payment method.

### Admission Confirmation:

1. **Acceptance of admission offer:** The system could allow newly admitted students to confirm their acceptance of the admission offer through an online portal. This would help the university to accurately estimate the incoming class size and plan for enrollment.
2. **Confirmation of enrollment status**: The system could allow students to check their enrollment status and confirm their enrollment in the courses they plan to take.
3. **Confirmation of financial aid and scholarship status**: The system could allow students to confirm their financial aid and scholarship status and accept or decline any offers they have received.
4. **Confirmation of attendance at orientation events:** The system could allow students to confirm their attendance at orientation events, which are typically held prior to the start of the semester to help new students acclimate to campus life.
5. **Acknowledgment of university policies:** The system could require students to confirm their acknowledgment of university policies, such as academic integrity and conduct standards, before they are fully enrolled.

### Admin view:

1. **Manage Applicants:** This use case involves managing the applications submitted by students. The admin can view, edit, and update the status of each application. They can also add new applications to the system and delete outdated or incomplete applications.
2. **Evaluate Applications:** This use case involves evaluating the applications based on various criteria such as academic achievements, extracurricular activities, and personal statements. The admin can view the applications that need to be evaluated and enter the evaluation results into the system.
3. **Publish Merit List:** This use case involves publishing the merit list of admitted students. The admin can generate the merit list based on the evaluation results and sort the list according to various criteria such as academic achievements and program preferences.
4. **Manage Programs:** This use case involves managing the different programs offered by the university. The admin can add new programs, edit existing programs, and delete outdated programs. They can also view the details of each program, such as admission requirements and program outcomes.
5. **Manage Users:** This use case involves managing the users who have access to the admission system. The admin can add new users, edit existing users, and delete users who no longer need access. They can also assign different levels of access to each user based on their role in the system.

### Logout:

1. The user logs in to the university admission system using their username and password.
2. After the user has completed their tasks within the system, they want to end their session and log out to prevent unauthorized access to their account.
3. The user clicks on the "log out" button, which is typically located in the upper right corner of the screen.
4. The system displays a confirmation message asking the user if they are sure they want to log out.
5. The user confirms their intention to log out by clicking on the "yes" button.
6. The system ends the user's session and returns them to the login screen.
7. The user is no longer able to access any of the features or functionality of the system until they log in again.

## Fully Dressed Use Cases:

### Login:

|  |  |
| --- | --- |
| **USE CASE NAME:** | **LOGIN** |
| **Scope:** | University admission system |
| **Primary Actor(s):** | Student, Faculty, Administrator |
| **Stakeholders and Interests:** | Student: Wants to access their application status, view admission requirements, and submit their application.  Faculty: Wants to access course materials, view student records, and communicate with students.  Administrator: Wants to manage the admission process, access university data, and maintain the system. |
| **Preconditions:** | The user has a valid username and password. |
| **Main Success Scenario:** | The user opens the admission system login page and enters their credentials.  The system validates the user's credentials and checks their access level.  The system logs the user into the admission system and displays their appropriate dashboard or landing page. |
| **Success Guarantee:** | The user is logged into the admission system with the appropriate level of access. |
| **Extensions:** | If the user enters invalid credentials, the system displays an error message and prompts them to re-enter their credentials or reset their password.  If the user's access level is insufficient, the system displays an error message and denies access to the admission system.  If the user's session times out, the system logs the user out and redirects them to the login page. |
| **Frequency of Occurrence:** | High |
| **Technology and Data Variation List:** | The admission system may use different authentication technologies, such as LDAP, OAuth, or custom authentication.The system may store user login data in a database, file system, or other storage technologies. |
| **Prototype:** | A prototype of the login feature can be created using web application frameworks such as Ruby on Rails, Django, or Laravel. |
| **Level:** | User goal |
| **Use Case Section:** | Core |
| **Special Requirements:** | The system should ensure that user passwords are secure and stored using appropriate encryption techniques. |
| **Miscellaneous:** | None |

### Apply for Admission:

|  |  |
| --- | --- |
| **USE CASE NAME:** | **APPLY FOR ADMISSION** |
| **Scope:** | University admission system |
| **Primary Actor(s):** | Student |
| **Stakeholders and Interests:** | Student: Wants to apply for admission to the university, view admission requirements, and submit their application. |
| **Preconditions:** | The user has a valid login and has accessed the admission system. |
| **Main Success Scenario:** | The user selects the option to apply for admission.  The system presents the admission application form to the user.  The user fills out the application form, including personal information, academic records, and supporting documents.  The user submits the application form.  The system validates the application form and checks for any missing information or errors.  The system confirms the successful submission of the application and provides the user with a confirmation message. |
| **Success Guarantee:** | The user has successfully submitted their admission application. |
| **Extensions:** | If the user fails to fill out the application form correctly, the system displays an error message and prompts them to correct the errors.  If the user fails to provide all necessary supporting documents, the system displays a warning message and prompts them to upload the missing documents.  If the user encounters technical issues while filling out the application form, the system provides a support contact for assistance. |
| **Frequency of Occurrence:** | Moderate |
| **Technology and Data Variation List:** | The admission system may use different application form formats, such as online forms or downloadable PDF forms.  The system may store application data in a database, file system, or other storage technologies. |
| **Prototype:** | A prototype of the apply for admission feature can be created using web application frameworks such as Ruby on Rails, Django, or Laravel. |
| **Level:** | User goal |
| **Use Case Section:** | Core |
| **Special Requirements:** | The system should ensure that the admission application form is secure and that user data is stored using appropriate encryption techniques. |
| **Miscellaneous:** | None |

### Apply for Test:

|  |  |
| --- | --- |
| **Use Case Name:** | **APPLY FOR TEST** |
| **Scope:** | University admission system |
| **Primary Actor(s):** | Student |
| **Stakeholders and Interests:** | Student: Wants to apply for a university admission test, view test requirements, and receive confirmation of their test registration. |
| **Preconditions:** | The user has a valid login and has accessed the admission system. |
| **Main Success Scenario:** | The user selects the option to apply for a test.  The system presents the available tests to the user.  The user selects the desired test and confirms their choice.  The system validates the user's eligibility for the test, based on their admission status and other relevant criteria.  If the user is eligible, the system presents the test registration form to the user.  The user fills out the test registration form, including personal information and payment details.  The user submits the test registration form.  The system validates the test registration form and checks for any missing information or errors.  The system confirms the successful registration of the test and provides the user with a confirmation message. |
| **Success Guarantee:** | The user has successfully registered for the university admission test. |
| **Extensions:** | If the user is not eligible for the test, the system displays an error message and prompts them to check their admission status or contact support.  If the user fails to fill out the test registration form correctly, the system displays an error message and prompts them to correct the errors.  If the user fails to provide valid payment details, the system displays a warning message and prompts them to update their payment information.  If the user encounters technical issues while registering for the test, the system provides a support contact for assistance. |
| **Frequency of Occurrence:** | Moderate |
| **Technology and Data Variation List:** | The admission system may use different payment gateways or payment processing systems to handle test registration fees.  The system may store test registration data in a database, file system, or other storage technologies. |
| **Prototype:** | A prototype of the apply for test feature can be created using web application frameworks such as Ruby on Rails, Django, or Laravel. |
| **Level:** | User goal |
| **Use Case Section:** | Core |
| **Special Requirements:** | The system should ensure that test registration data is secure and that payment information is stored using appropriate encryption techniques. |
| **Miscellaneous:** | None |

### Enter Personal Info:

|  |  |
| --- | --- |
| **Use case name** | Enter Personal Info |
| **Scope** | The Personal Info use case describes the functionality of the university admission system for gathering and managing the personal information of prospective students. |
| **Main Success Scenario** | The prospective student logs in to the university admission system.  The system displays a form for the student to enter their personal information, such as their name, date of birth, address, contact information, and academic history.  The student fills out the form with their personal information.  The system validates the entered data and checks for any errors or missing information.  If there are any errors or missing information, the system displays an error message and prompts the student to correct the mistakes.  Once all the information is complete and error-free, the system saves the information in the database and displays a confirmation message to the student. |
| **Pre-Condition** | The prospective student must have applied for admission to the university. |
| **Primary Actors** | Student |
| **Level** | Level 1: Basic Level: The student enters their personal information into the university admission system.  Level 2: Intermediate Level: The system validates the entered data and checks for any errors or missing information.  Level 3: Advanced Level: The system saves the information in the database and displays a confirmation message to the student. |
| **Stake Holders** | Prospective students: The primary stakeholders who benefit from the Personal Info use case by having their personal information accurately recorded and stored in the university admission system.  University administrators: The secondary stakeholders who benefit from the Personal Info use case by having access to accurate and up-to-date personal information for each prospective student. |
| **Success Guarantee** | The personal information of the prospective student is accurately recorded and stored in the university admission system. |
| **Exceptions** | The university admission system should be user-friendly and easy to navigate.  The system should validate the entered data and check for any errors or missing information.  The personal information of the student should be securely stored in the database.  The system should provide clear and concise feedback to the student on the status of their personal information. |
| **Special Requirements** | The university admission system must comply with all applicable data privacy laws and regulations. |
| **Technology and data variation list** | Technology: The university admission system is a web-based application that requires a modern web browser and an internet connection.  Data Variation: The personal information entered by the student may vary in format and content, depending on the student's background and cultural norms. |
| **Frequency of occurrences** | This use case occurs for each new prospective student who applies for admission to the university. |
| **Miscellaneous** | The Personal Info use case is one of several use cases in the university admission system, including Application Submission, Transcript Submission, Test Scores Submission, and Admissions Decision. |
| **Prototyping** | A prototype of the university admission system with the Personal Info use case can be developed and tested with a small group of prospective students before the system is deployed for the entire applicant pool. |

### Enter Academic Info:

|  |  |
| --- | --- |
| **Use Case Name** | **ENTER ACADEMIC INFO** |
| **Scope** | This use case describes the process of managing academic information in the university admission system, which is used by students and staff members to manage the admission process. |
| **Primary Actors** | Student: A person who is applying for admission to the university<br>- Staff Member: A person who is responsible for managing the admission process |
| **Stakeholders** | Students: They want to be able to provide accurate academic information to be considered for admission.  Staff Members: They want to ensure that the academic information is accurate and up-to-date for the admission process. |
| **Precondition** | The user is logged in to the university admission system. |
| **Main Success Scenario** | 1. The user selects the "Academic Info" option in the system.  2. The system displays a form for the user to enter their academic information.  3. The user fills out the form with their academic information.  4. The system saves the academic information to the user's profile. |
| **Alternative Flow** | If the user cancels the form, the system returns the user to the previous page. |
| **Success Guarantee** | The user's academic information is saved to their profile in the system. |
| **Expectations** | The user expects that their academic information will be accurately recorded in the system for consideration in the admission process. |
| **Technology and Data Variation List** | 1. The university admission system is accessed via a web browser.  2. The user's academic information is stored securely in a database.  3. The system is built using object-oriented programming principles. |
| **Frequency of Occurrence** | This use case occurs whenever a user needs to enter or update their academic information in the university admission system. |
| **Use Case Section** | This use case is part of the "Application Management" section of the university admission system. |
| **Special Requirements** | The academic information entered by the user must be validated to ensure that it is accurate and complete. |
| **Prototype** | A prototype of the university admission system is available for testing. |
| **Miscellaneous** | None. |

### Verify Document:

|  |  |
| --- | --- |
| **Use Case Name** | **VERIFY DOCUMENT** |
| **Scope** | This use case describes the process of verifying documents in the university admission system, which is used by staff members to manage the admission process. |
| **Primary Actors** | Staff Member: A person who is responsible for verifying documents submitted by students for the admission process. |
| **Stakeholders** | Students: They want to ensure that their documents are accurately verified and processed for the admission process.  Staff Members: They want to ensure that the documents submitted by students are accurate and complete before being processed for the admission process. |
| **Precondition** | The staff member is logged in to the university admission system and has access to the submitted documents. |
| **Main Success Scenario** | 1. The staff member selects the "Verify Documents" option in the system.  2. The system displays a list of submitted documents for the staff member to verify.  3. The staff member reviews each document and verifies that it is accurate and complete.  4. The staff member updates the status of each document as verified or not verified in the system.  5. The system saves the updated status of the documents. |
| **Alternative Flow** | If the staff member identifies an error in a document, they can mark it as "rejected" and provide a reason for rejection. The student will be notified and given a chance to submit a corrected document. |
| **Success Guarantee** | The documents submitted by students are accurately verified and processed for the admission process. |
| **Expectations** | The staff member expects that the documents submitted by students are accurate and complete before being processed for the admission process. |
| **Technology and Data Variation List** | 1. The university admission system is accessed via a web browser.  2. The documents submitted by students are stored securely in a database.  3. The system is built using object-oriented programming principles. |
| **Frequency of Occurrence** | This use case occurs whenever a staff member needs to verify documents submitted by students for the admission process. |
| **Use Case Section** | This use case is part of the "Application Management" section of the university admission system. |
| **Special Requirements** | The staff member must have access to the submitted documents and must be able to verify their accuracy and completeness. |
| **Prototype** | A prototype of the university admission system is available for testing. |
| **Miscellaneous** | None. |

### Announcement:

|  |  |
| --- | --- |
| **Use Case Name:** | Announcement |
| **Scope:** | University Admission System |
| **Primary Actors:** | Admin, Students |
| **Stakeholders and Interests:** | Admission Officer: Need to inform prospective students about the admission process and deadlines. Prospective Students: Need to receive timely and accurate information about the admission process to make informed decisions. |
| **Precondition:** | The admission officer has access to the announcement section of the admission system. |
| **Success Guarantee:** | Prospective students receive timely and accurate information about the admission process and deadlines. |
| **Main Success Scenario:** | The admission officer logs into the admission system and navigates to the announcement section. The admission officer creates a new announcement with a title, description, and deadline. The admission officer publishes the announcement to the admission system. The prospective students log into the admission system and navigate to the announcement section. The prospective students read the announcement and take appropriate action. |
| **Extensions:** | If the admission officer fails to create a new announcement, the system displays an error message and the use case ends. If the admission officer fails to publish the announcement, the system displays an error message and the use case ends. If the prospective students fail to read the announcement, the system sends a reminder to their registered email address. |
| **Frequency of Occurrence:** | Multiple times a year, depending on the admission cycle. |
| **Technology and Data Variation List:** | The admission system should support multiple languages for the announcement text. The system should be accessible from different devices such as desktops, laptops, tablets, and smartphones. The system should be able to handle a large number of concurrent users during peak admission periods. |
| **Prototype:** | A prototype of the Announcement use case can be developed to demonstrate the following functionalities The ability to create and publish announcements. The ability to display announcements in a user-friendly format. The ability to send email reminders to students who have not read the announcements. |
| **Special Requirements:** | The announcement section should be easy to navigate and accessible to all prospective students. |
| **Level:** | User Goal |
| **Use Case Section:** | Main Flow, Extensions |
| **Miscellaneous:** | The admission officer should be able to edit or delete announcements if necessary. The system should have a mechanism to ensure that the announcements are displayed in chronological order. The system should be able to track the number of views and clicks on each announcement for administrative purposes. |

### Notify (Advertisement):

|  |  |
| --- | --- |
| **Use Case Name** | **Notify (Advertisement)** |
| Scope | This use case covers the process of notifying potential applicants of upcoming admissions for the university. |
| Primary Actors | University Administrators |
| Stakeholders | University Administrators, Prospective Applicants |
| Pre-Conditions | The university has upcoming admissions, and the university administrators have prepared the admission advertisement. |
| Main Success Scenario | 1. The university administrators access the admission system and select the Notify feature.  2. The administrators create the admission advertisement, including the required information, such as application deadlines, required documents, and admission criteria.  3. The administrators select the target audience for the notification based on various criteria, such as age, academic level, location, and interests.  4. The admission system generates the notification and sends it to the target audience via email or other communication channels.  5. Prospective applicants receive the notification and follow the instructions to apply for admission. |
| Alternative Scenarios | - If there are technical issues with the admission system, the administrators will contact the IT department to resolve the issue.  - If the notification fails to reach the target audience, the administrators will investigate the issue and take appropriate actions to resolve it. |
| Success Guarantee | The notification process will be successful, and the target audience will receive the admission advertisement. |
| Expectations | The university administrators expect to reach a wide audience of potential applicants and to receive a high number of applications as a result of the admission advertisement. |
| Frequency of Occurrence | This use case occurs periodically, depending on the admission cycles of the university. |
| Technology and Data Variation List | - Operating System: Windows, Mac OS, Linux  - Web Browser: Chrome, Firefox, Safari, Edge  - Communication Channels: Email, Social Media, SMS |
| Prototype | A prototype of the notify feature will be developed to test the use case. |
| Miscellaneous | None. |
| Use Case Section | Admission |
| Special Requirements | The admission advertisement should include all necessary information for prospective applicants to apply for admission, including any required documents and deadlines. The notification should also be personalized and relevant to the target audience to increase the chances of attracting qualified applicants. |

### Make Test Payment:

|  |  |
| --- | --- |
| Use case name | Make Test Payment |
| Scope | This use case describes the interactions between the University Admission System and the Guest who wants to pay the admission test fee. |
| Main success scenario | The Guest logs in to the University Admission System.  The system presents the Guest with an option to pay the admission test fee.  The Guest selects the option and enters the required information, such as name, program, and fee amount.  The system presents the Guest with a list of available payment methods.  The Guest selects a payment method and submits the payment.  The system confirms the payment and updates the application status to indicate that the test fee has been paid. |
| Preconditions | The Guest has submitted an application for admission.  The admission test fee has been determined by the university.  The admission test has been scheduled by the university. |
| Primary actors | Guest |
| Level | Level 1: Enter payment information and select payment method  Level 2: Confirm payment |
| Stakeholders | Guest: Wants to pay the admission test fee.  University: Wants to ensure that the admission test fee is paid by the Guests. |
| Success guarantee | The University Admission System will allow the Guest to easily pay the admission test fee and the system will update the application status accordingly. |
| Exceptions | If the test payment processor encounters any issues with processing test payments through the admission system, it can contact the system administrators or finance department for assistance. |
| Special requirements | The payment must be processed securely and efficiently.  The system must update the application status once the payment has been completed. |
| Technology and data variation list | The payment may be processed through different payment gateways.  The test fee may vary based on the program and the number of tests the Guest is taking. |
| Frequency of occurences | This use case will occur multiple times per day during the admission process. |
| Miscellaneous | The University Admission System may provide a list of payment methods available for the Guest to select.  The payment may have an expiry date to ensure that it is completed within a specified timeframe |
| Prototyping | A prototype of the payment feature should be developed and tested with Guests to ensure that the system meets their needs and is easy to use. |

### Scheduling Test:

|  |  |
| --- | --- |
| **Use case name** | Scheduling Test |
| **Scope** | The Test Schedule use case describes the functionality of the university admission system for managing and scheduling entrance exams for prospective students. |
| **Main Success Scenario** | The admission staff member logs in to the university admission system.  The system displays a list of eligible students who have completed all the necessary requirements to take the entrance exam.  The admission staff member selects a student from the list and schedules the entrance exam for the student.  The system displays a list of available dates and times for the exam.  The admission staff member selects a date and time for the exam that is convenient for both the student and the university.  The system confirms the date and time of the exam and notifies the student via email or other means.  The student arrives at the designated location and takes the exam at the scheduled time. |
| **Pre-Condition** | The prospective student must have applied for admission to the university and have completed all the necessary requirements to be eligible to take the entrance exam. |
| **Primary Actors** | Admin, students |
| **Level** | Level 1: Basic Level: The admission staff member schedules the entrance exam for the prospective student.  Level 2: Intermediate Level: The system displays a list of available dates and times for the exam.  Level 3: Advanced Level: The system confirms the date and time of the exam and notifies the student via email or other means. |
| **Stake Holders** | Admission staff - responsible for scheduling and managing entrance exams for prospective students.  Prospective students - responsible for attending the entrance exam as per the scheduled date and time. |
| **Success Guarantee** | The entrance exam is scheduled for the student on a convenient date and time, and the student is notified of the exam details. |
| **Exceptions** | If the admission staff member encounters any technical difficulties while scheduling the exam, they can contact the university helpdesk for assistance.  If the student is unable to attend the scheduled exam, they can contact the university to reschedule for a different date and time.  If there is an error with the database or system, the system will display an error message and prompt the staff member to try again later. |
| **Special Requirements** | The system should allow the staff member to view the scheduled exam details and make changes if necessary.  The system should be able to generate reports on the exam scheduling and attendance for administrative purposes. |
| **Technology and data variation list** | Technology: The university admission system is a web-based application that requires a modern web browser and an internet connection.  Data Variation: The list of available exam dates and times may vary based on the availability of exam proctors and facilities. |
| **Frequency of occurrences** | This use case occurs for each eligible student who has completed all the necessary admission requirements to take the entrance exam. |
| **Miscellaneous** | The university admission system should have a feature to automatically reschedule an exam if a student fails to attend the scheduled exam.  The system should have an option to cancel an exam and notify the student if the exam cannot be held due to unforeseen circumstances such as natural disasters or system outages.  The system should have a mechanism to prevent overbooking of exam dates and times to ensure the smooth conduct of exams. |

### Take Test:

|  |  |
| --- | --- |
| Use case name | Take Test |
| Scope | This use case describes the interactions between the University Admission System and the Guest who wants to take the admission test. |
| Main success scenario | The Guest arrives at the test location at the scheduled time.  The Guest presents their identification and admission details to the test proctor.  The test proctor verifies the Guest's identity and admission details and provides them with the test materials.  The Guest completes the test within the specified time.  The Guest submits the completed test to the test proctor.  The test proctor confirms that the test has been completed and sends it for evaluation.  The University Admission System updates the application status to indicate that the admission test has been completed. |
| Preconditions | The Guest has applied for admission to the university.  The admission test has been scheduled by the university.  The test location and time have been communicated to the Guest. |
| Primary actors | Guest |
| Level | Level 1: Present identification and admission details  Level 2: Complete test within specified time  Level 3: Submit completed test to test proctor  Level 4: Confirm test completion and send it for evaluation |
| Stakeholders | Guest: Wants to take the admission test.  University: Wants to evaluate the Guest's knowledge and skills to determine their eligibility for admission |
| Success guarantee | The University Admission System will ensure that the Guest can take the admission test, the test is evaluated correctly, and the application status is updated accordingly. |
| Exceptions | If the Guest is unable to attend their scheduled testing date due to unforeseen circumstances, they must contact the testing center to reschedule.  If the Guest is found to be cheating during the test, their test results are nullified and they may be subject to disciplinary action. |
| Special requirements | The test must be taken securely and efficiently.  The system must update the application status once the test has been completed and evaluated.  The test materials and format must be appropriate for the program and the Guest's field of study. |
| Technology and data variation list | The test materials may vary based on the program and the Guest's field of study.  The test format may vary based on the program and the Guest's field of study. |
| Frequency of occurences | This use case will occur multiple times per year during the admission process. |
| Miscellaneous | The University Admission System may provide study materials to the Guest to prepare for the test.  The test may have a passing score that the Guest must achieve to be considered for admission.  The system may generate a report of the test results for the Guest and the university. |
| Prototyping | A prototype of the test-taking process should be developed and tested with Guests to ensure that the system meets their needs and is easy to use. |

### Print Challan:

|  |  |
| --- | --- |
| Use case name | Print Challan |
| Scope | This use case describes the interactions between the University Admission System and the applicant who wants to generate a challan to pay the admission fee. |
| Main success scenario | The Student or Guest logs in to the University Admission System.  The system presents the Student or Guest with an option to generate a challan for the admission fee.  The Student or Guest selects the option and enters the required information, such as name, program, and fee amount.  The system generates a challan with a unique identification number.  The Student or Guest can download and print the challan for payment.  The system updates the application status to indicate that the challan has been generated. |
| Preconditions | The Student or Guest has submitted an application for admission.  The admission fee has been determined by the university. |
| Primary actors | Student, Guest. |
| Level | Level 1: Generate challan.  Level 2: Download and print challan. |
| Stakeholders | Student or Guest: Wants to generate a challan to pay the admission fee.  University: Wants to ensure that the admission fee is paid by the Student or Guests before proceeding with the admission process. |
| Success guarantee | The University Admission System will allow the Student or Guest to easily generate a challan for the admission fee and the system will update the application status accordingly. |
| Exceptions | If the Student or Guest encounters an issue with the payment channels or payment confirmation, they can contact the finance department for assistance. |
| Special requirements | The challan must contain all required information, such as Student or Guest name, program, fee amount, and identification number.  The challan must be unique to prevent duplication or fraud. |
| Technology and data variation list | The challan may be generated in different formats, such as PDF or HTML.  The challan may have different designs or logos for different programs or departments. |
| Frequency of occurences | This use case will occur multiple times per day during the admission process. |
| Miscellaneous | The University Admission System may integrate with a payment gateway to enable online payment of the admission fee.  The system may provide the option to track the status of the challan and the payment.  The challan may have an expiry date to ensure that it is used within a specified timeframe. |
| Prototyping | A prototype of the challan generation feature should be developed and tested with Student or Guests to ensure that the system meets their needs and is easy to use. |

### Make Admission Payment:

|  |  |
| --- | --- |
| Use case name | Make Admission Payment |
| Scope | This use case describes the interactions between the University Admission System and the Student or Student or Guest who wants to pay the admission fee. |
| Main success scenario | The Student or Guest logs in to the University Admission System.  The system presents the Student or Guest with an option to pay the admission fee.  The Student or Guest selects the option and enters the required information, such as name, program, and fee amount.  The system presents the applicant with a list of available payment methods.  The applicant selects a payment method and submits the payment.  The system confirms the payment and updates the application status to indicate that the admission fee has been paid. |
| Preconditions | The applicant has been accepted for admission to the university.  The admission fee has been determined by the university.  The payment deadline has been specified by the university. |
| Primary actors | Student, Guest. |
| Level | Level 1: Enter payment information and select payment method  Level 2: Confirm payment |
| Stakeholders | Student or Guest: Wants to pay the admission fee.  University: Wants to ensure that the admission fee is paid by the Student or Guest. |
| Success guarantee | The University Admission System will allow the Student or Guest to easily pay the admission fee and the system will update the application status accordingly. |
| Exceptions | If the payment transaction fails or is declined by the payment gateway or processor, the Student or Guest is prompted to correct their payment information or select a different payment method.  If the payment transaction is successful but there is a delay or error in recording the payment transaction details, the Student, Guest or system administrators can contact the finance department for assistance. |
| Special requirements | The payment must be processed securely and efficiently.  The system must update the application status once the payment has been completed.  The payment deadline must be enforced to ensure that the Student or Guest completes the payment within the specified timeframe. |
| Technology and data variation list | The payment may be processed through different payment gateways.  The admission fee may vary based on the program and the Student or Guest's status (domestic/international). |
| Frequency of occurences | This use case will occur multiple times per year during the admission process. |
| Miscellaneous | The University Admission System may provide a list of payment methods available for the Student or Guest to select.  The payment may have an expiry date to ensure that it is completed within a specified timeframe.  The system may generate a receipt for the Student or Guest once the payment has been completed. |
| Prototyping | A prototype of the payment feature should be developed and tested with Student or Guests to ensure that the system meets their needs and is easy to use. |

### Admission Confirmation:

|  |  |
| --- | --- |
| **Use Case Name** | **Admission Confirmation** |
| Scope | This use case covers the process of confirming admission for accepted applicants in the university admission system. |
| Primary Actors | Accepted Applicants |
| Stakeholders | Accepted Applicants, University Administrators, Faculty Members, IT Department |
| Pre-Conditions | The applicant must have been accepted into the university through the admission system. |
| Main Success Scenario | 1. The accepted applicant logs into the admission system using their credentials.  2. The applicant views the status of their admission and any pending tasks required for enrollment.  3. The applicant confirms their intention to enroll in the university.  4. The admission system generates an admission confirmation letter for the applicant.  5. The applicant receives the admission confirmation letter and follows any instructions provided. |
| Alternative Scenarios | - If the applicant is unable to log in, they will contact the IT department for assistance.  - If there are any technical issues with the admission system, the applicant will contact the IT department to resolve the issue. |
| Success Guarantee | The admission confirmation process will be successful, and the applicant will receive confirmation of their admission to the university. |
| Expectations | The applicant expects to receive confirmation of their admission and to have clear instructions for the enrollment process. |
| Frequency of Occurrence | This use case occurs frequently during the enrollment period. |
| Technology and Data Variation List | - Operating System: Windows, Mac OS, Linux  - Web Browser: Chrome, Firefox, Safari, Edge  - Data Format: PDF, Email, Text Message |
| Prototype | A prototype of the admission confirmation process will be developed to test the use case. |
| Miscellaneous | None. |
| Use Case Section | Admission |
| Special Requirements | The admission confirmation letter should include all necessary information for the applicant to complete the enrollment process, including any required documents and deadlines. |

### Publish Merit List:

|  |  |
| --- | --- |
| **Use case name** | **Publish merit list** |
| **Scope** | This use case describes the actions and interactions involved in publishing the merit list of admitted students in the University Admission System. The merit list contains the names of students who have been admitted to the university, sorted by their academic achievements. |
| **Primary actors** | Admin |
| **Pre-condition** | The admin user has logged into the University Admission System.  All admission decisions have been made and recorded in the system. |
| **Main success scenario** | The admin user navigates to the dashboard and selects "Publish Merit List".  The system generates a list of all admitted students, sorted by their academic achievements.  The admin user reviews the list and approves it for publication.  The system publishes the merit list on the university website and notifies all admitted students.  If the admin user identifies errors in the merit list, they can correct them and re-approve the list for publication. |
| **Level** | User goal. |
| **Stakeholders** | **Admission Committee**: Needs to evaluate applications, calculate merit scores, and publish the merit list to make admission decisions.  **Applicants:** Expect timely and accurate updates on their application status and admission chances.  **University Staff:** Need to access student information for various administrative tasks, such as financial aid, registration, and housing. |
| **Success guarantee** | The merit list accurately reflects the admissions decisions and is published in a timely manner.  Admitted students receive timely notifications of their admission status. |
| **Exceptions** | The University Admission System is expected to accurately and securely store admission decisions.  The system is expected to generate a clear and organized merit list.  The system is expected to notify admitted students in a timely manner. |
| **Special requirement** | The University Admission System must be secure and protect student information from unauthorized access or disclosure.  The system must be able to handle a large volume of admission decisions and students. |
| **Technology and data variation list** | The University Admission System may use different technologies for publishing the merit list on the university website.  The system may need to convert admission decision data into a format that is suitable for the merit list. |
| **Frequency of occurrences** | The Publish Merit List use case occurs once per admission cycle, which may occur multiple times per year. |
| **Miscellaneous** | The merit list may need to comply with privacy laws and regulations in order to protect student information. |
| **Prototyping** | A prototype of the University Admission System has been developed and is undergoing testing. |

### Check Merit List:

|  |  |
| --- | --- |
| **Use case name** | **Check merit list** |
| **Scope** | This use case describes the actions and interactions involved in checking the merit list of admitted students in the University Admission System. The merit list contains the names of students who have been admitted to the university, sorted by their academic achievements. |
| **Primary actors** | Guest/student. |
| **Pre-condition** | The applicant has submitted an application to the university.  The university has published the merit list. |
| **Main success scenario** | The applicant navigates to the university website and selects "Check Merit List".  The system displays the merit list of admitted students, sorted by their academic achievements.  The applicant reviews the merit list to see if their name is listed.  If the applicant's name is on the merit list, they receive notification of their admission status.  If the applicant's name is not on the merit list, they receive notification that they have not been admitted. |
| **Level** | User goal |
| **Stakeholders** | **Faculty:** Professors and instructors who will be responsible for teaching and advising the admitted students.  **University Administration:** The higher-level administration who oversee the university as a whole. |
| **Success guarantee** | The merit list accurately reflects the admissions decisions and is published in a timely manner.  Applicants receive timely notifications of their admission status. |
| **Exceptions** | The University Admission System is expected to accurately and securely store admission decisions.  The system is expected to generate a clear and organized merit list.  The system is expected to notify applicants in a timely manner. |
| **Special requirement** | The University Admission System must be secure and protect student information from unauthorized access or disclosure.  The system must be able to handle a large volume of admission decisions and students. |
| **Technology and data variation list** | The University Admission System may use different technologies for displaying the merit list on the university website.  The system may need to convert admission decision data into a format that is suitable for the merit list. |
| **Frequency of occurrences** | The Check Merit List use case occurs once per admission cycle, which may occur multiple times per year. |
| **Miscellaneous** | The merit list may need to comply with privacy laws and regulations in order to protect student information. |
| **Prototyping** | A prototype of the University Admission System has been developed and is undergoing testing. |

### Admin View:

|  |  |
| --- | --- |
| **Use case name** | **Admin view** |
| **Scope** | This use case describes the actions and interactions of the admin user in the University Admission System. The admin user is responsible for managing the application process, reviewing and approving/denying applications, and generating reports. |
| **Primary actors** | Admin |
| **Pre-condition** | The admin user has logged into the University Admission System.  The system has access to the necessary data, including applicant information and admission requirements. |
| **Main success scenario** | The admin user navigates to the dashboard and selects "Applications".  The system displays a list of all current applications.  The admin user selects an application to review.  The system displays the application details, including the applicant's personal information, transcripts, test scores, and essay.  The admin user reviews the application and makes a decision to approve, deny, or request additional information.  If additional information is required, the admin user can request it through the system and notify the applicant.  The system updates the application status and notifies the applicant of the decision.  The admin user can generate reports on application status and other statistics as needed.  If the application is incomplete or missing required information, the admin user can deny the application and notify the applicant.  If the admin user cannot make a decision based on the available information, they can request additional review by faculty or other staff members. |
| **Level** | User goal and satisfaction of user. |
| **Stakeholders** | **Applicants:** Students who are applying to the university.  **Faculty:** Professors and instructors who are responsible for teaching and advising students.  **Staff:** Other staff members who may interact with the University Admission System.  University Administration: The higher-level administration who oversee the university as a whole. |
| **Success guarantee** | The admin user can efficiently manage the application process and ensure that all applications are reviewed in a timely manner.  Applicants receive timely notifications and updates on their application status. |
| **Exceptions** | The University Admission System is expected to accurately and securely store applicant information.  The system is expected to provide the admin user with clear and organized application details.  The system is expected to generate reports that accurately reflect application status and other statistics. |
| **Special requirement** | The University Admission System must be secure and protect applicant information from unauthorized access or disclosure.  The system must be able to handle a large volume of applications and users. |
| **Technology and data variation list** | The University Admission System may be accessed on different devices with different screen sizes and resolutions.  The system may store applicant information in different formats and from different sources.  The system may use different technologies for generating reports. |
| **Frequency of occurrences** | The admin user will use this use case frequently during the application review period, which may occur multiple times per year. |
| **Miscellaneous** | The University Admission System may need to comply with privacy laws and regulations in order to protect applicant information. |
| **Prototyping** | A prototype of the University Admission System has been developed and is undergoing testing. |

### Log out:

|  |  |
| --- | --- |
| **Use Case Name** | **LOG OUT** |
| **Scope** | This use case describes the process of logging out of the university admission system, which is used by students and staff members to manage the admission process. |
| **Primary Actors** | Student |
| **Stakeholders** | Students: They want to be able to securely log out of the system to protect their personal information.  Staff Members: They want to ensure that the system is secure and that students can log out without any issues. |
| **Precondition** | The user is logged in to the university admission system. |
| **Main Success Scenario** | 1. The user clicks on the "Log Out" button in the system.  2. The system prompts the user to confirm that they want to log out.  3. The user confirms that they want to log out.  4. The system logs the user out and redirects them to the login page. |
| **Alternative Flow** | If the user cancels the log out confirmation, the system remains on the current page. |
| **Success Guarantee** | The user is successfully logged out of the system and their session is terminated. |
| **Expectations** | The user expects that their personal information will be protected after logging out of the system. |
| **Technology and Data Variation List** | 1. The university admission system is accessed via a web browser  2. The user's personal information is stored securely in a database.  3. The system is built using object-oriented programming principles. |
| **Frequency of Occurrence** | This use case occurs whenever a user wants to log out of the university admission system. |
| **Use Case Section** | This use case is part of the "Account Management" section of the university admission system. |
| **Special Requirements** | The user's session must be terminated securely to protect their personal information. |
| **Prototype** | A prototype of the university admission system is available for testing. |
| **Miscellaneous** | None. |