Use Case: Validate Identity

Purpose: To validate the details and privileges of user before allowing access to system

Requirements: Have an account on the system

Event Flow: 1. User inputs username and password

2. Details checked against database

3. If details in database, log in with correct privileges

4. Show what can be viewed or changed by user

Success: Be able to view what they need to view

Use Case: Display Changes

Purpose: To view changes to the database after they have been committed

Requirements: Changes to be made to data base

Event Flow: 1. Changes submitted

2. The system will show all the current changes

3. Allow for more changes to be made if incorrect

Success: Correct changes made

Use Case: Log Out

Purpose: To log out of the system and prevent unwanted access

Requirements: Be logged in

Event Flow: 1. User click on log out

2. System updates that they have logged out

3. User sent to login page

Success: User on log in screen

Student:

The student only has one possible action on the system; to view their own details. The information displayed is dependent on their progression through their degree programme. The system should display grades achieved for each level of their programme, as well as their degree if they have completed it. The student cannot update any information, details displayed are read only.

Use Case: View Status

Purpose: Allow the student to view their grades for each completed level of their degree programme

Requirements: The student is logged in to the system

Event Flow: 1. Validate Identity use case

2. System displays student's status

3. Log out use case

Success: The student is logged out and has seen their status

Teacher:

The teacher should only have access to change and calculate the grades for students under them. This includes the progression or resits of modules. Details displayed are read and write.

Use case: Edit Grades

Purpose: Allow Teachers to add or update initial grades for each student taking a module at a given period and level of study

Requirements: Teacher to be logged in and have the correct privileges

Event Flow: 1. Validate Identity use case

2. System displays different students, under the teacher

3. Teacher chooses a student to view their grades

4. Teacher views the current and previous grades

5. Teacher can modify current grades

6. Teacher saves changes and checks if changes have been made

7. Log Out use case

Success: Teacher logged out

Use Case: Progression

Purpose: Allow Teachers to decide whether students can progress past, fail or resit the current year.

Requirements: Student has completed some work

Event Flow: 1. Validate Identity use case

2. System displays different students, under the teacher

3. Teacher chooses a student to calculate their progression

4. Check Progression use case

5. Students progression is set for the next year

6. Log Out use case

Success: Progression set and teacher Logged out

Use Case: Check Progression

Purpose: Allow teacher to determine what progression students have

Requirements: Student has completed all work

Event Flow: 1. Calculate Weighted Grades use case

2. If fail Student can Resit the year, if not already on resit

3. If anything else, progression to next year or graduation

4. Progression set to resit or progress

Success: Progression set

Use Case: Calculating Weighted Grades

Purpose: To allow teacher to decide what percentage is a fail, pass or resit

Requirements: Student has completed all work

Event Flow: 1. Add up module grades and with weight of credits

2. If weighted grades are bellow 39.5% Fail for all degrees

3. If weighted grades are between 39.5 and 44.4% Fail for 1-year MSc, MComp, MEng degrees and a Pass for BSc, BEng degrees

4. If weighted grades are between 44.5 and 49.4% Fail for 1-year MSc, MComp, MEng degrees and a Third-class hon for BSc, BEng degrees

5. If weighted grades are between 49.5 and 59.4% Pass for 1-year MSc degrees, Lower Second-Class hon for MComp, Meng, BSc, BEng degrees

6. If weighted grades are between 59.5 and 69.4% for Merit for 1-year MSc degrees, Upper Second-Class hon for MComp, Meng, BSc, BEng degrees

7. If weighted grades are above 69.5% for Distinction for 1-year MSc degrees, First-Class hon for MComp, Meng, BSc, BEng degrees

Success: Progression Type calculated

Registrars

Registrars add or remove students form the system, the adding of students involves choosing their degree and corresponding modules both compulsory and optional. The Registrar need to be able to check that the changes have gone through correctly.

Use Case: Remove Student

Purpose: To remove students if they fail or want to leave the university

Requirements: Student account

Event Flow: 1. Validate Identity use case

2. System displays the options available to the registrars

3. Registrar chooses Delete Student

4. Registrar inputs student id number and full name, clicks delete

5. Student is removed form databases

6. Display Changes use case

7. Log Out use case

Success: Student removed, registrar logged out

‘

Use Case: Add Students

Purpose: To allow student to study at the university and have access to their grades

Requirements: A new student

Event Flow: 1. Validate Identity use case

2. System displays the options available to the registrars

3. Registrar chooses Add Student

4. Registrar inputs personal data for student

5. Choose degree

5. Module Choice use case

6. Student added

7. Display Changes use case

Success: Student added

Use Case: Module Choice

Purpose: to allow the Registrar to select what optional modules on top of the compulsory modules

Requirements: A Student account and a degree

Event Flow: 1. See what modules are compulsory for the degree chosen, and how many credits are available

2. See the optional modules that fulfil the credits needed

3. Choose the modules wanted on top of compulsory

4. Check Degree use case

5. Modules chosen

Success: Modules chosen

Use Case: Check Degree

Purpose: To check what level of the degree the student is on

Requirements: Student account, degree chosen

Event Flow: 1. Check whether the student is doing an Undergraduate or a Postgraduate

2. If undergraduate use the Check Credits for Undergraduate degree use case

3. If Postgraduate degree use the Check Credits for Postgraduate degree use case

Success: Correct module credits selected for student

Use Case: Check Credits for Undergraduate degree

Purpose: To make sure all students do 120 credits a year

Requirements: A Student account, a degree and modules chosen

Event Flow: 1. Add up the credits for every module chosen

2. If 120, modules are fine

3. If over 120, send back to options page and state the modules that are the cause for over the 120 mark

4. If under 120, send back to module page and state how much they are under

5. Credits add up to 120

Success: Credits add up to 120

Use Case: Check Credits for Postgraduate degree

Purpose: To make sure all students do 180 credits a year

Requirements: A Student account, a degree and modules chosen

Event Flow: 1. Add up the credits for every module chosen

2. If 180, modules are fine

3. If over 180, send back to options page and state the modules that are the cause for over the 180 mark

4. If under 180, send back to module page and state how much they are under

5. Credits add up to 180

Success: Credits add up to 180

Admin

The Admin will create or remove accounts, i.e. the names and passwords. As well as departments, degrees and modules

Use Case: Accounts

Purpose: To create new accounts, such as teacher, registrars or students

Requirements: Admin Privilages

Event Flow: 1. Extend Validate Identity use case

2. Select create new account

3. Select type

4. Input name

5. Auto generate a password

6. save changes

7. Include Display Changes use case

7. Include Log Out Use Case

Use Case: Edit Departments

Purpose: Allow the Admin to create or delete departments within the university

Requirements: Admin Privileges

Event Flow: 1. Validate Identity use case

2. System displays the options available to the Admin

3. Admin chooses Edit Departments

4. Admin either creates a new department or deletes form the list

5. Display Changes use case

6. Log Out use case

Success: Department created or deleted

Use Case: Edit Degrees

Purpose: Allow the Admin to create or delete degrees within the universities departments

Requirements: Admin Privileges, Department created

Event Flow: 1. Validate Identity use case

2. System displays the options available to the Admin

3. Admin chooses Edit Departments Degrees

4. Admin either creates new degrees for the department or deletes current ones

5. Display Changes use case

6. Log Out use case

Success: Degrees created or deleted

Use Case: Edit Modules

Purpose: Allow the Admin to create or delete modules within the universities department specific degrees

Requirements: Admin Privileges, Department created, Degree created

Event Flow: 1. Validate Identity use case

2. System displays the options available to the Admin

3. Admin chooses Edit Departments Degree modules

4. Select what department they wat to change

5. Select what degree they want to change

4. Admin either creates new modules for the degree or deletes current ones

5. Display Changes use case

6. Log Out use case

Success: Modules created or deleted