

Flow of Activities for scenario of: **Input Degree**

Main Flow:

1. System starts in waiting state
2. Student enters degree code or title
3. System returns core courses
4. Student enters major
5. System returns major course options
6. Student selects electives
7. Student clicks save button
8. System saves schedule
9. Student clicks exit
10. System exits

Exceptions:

1. Student enters invalid degree code or title
 - a. No core courses are selected
 - b. System returns to waiting state
2. Student enters invalid major
 - a. No major courses are assigned
 - b. System returns to waiting state

Name	Input Degree					
Brief Description	The system verifies degree selection and assigns core, major and elective courses					
Actors	Student					
Related Use Cases	Select study length					
Entry Condition	Waiting state in which the system requests a degree					
Exit Condition	Next button is clicked					
Flow of events	<table><tr><th>Actors</th><th>System</th></tr><tr><td>1. Student enter a degree code or title. 2. Student enter a major 3. Student selects electives from list 4. Student selects save button 5. Student selects next button</td><td>1.1 System returns degree matching code or title 2.1 System returns major courses matching major title 3.1 System assigns electives in valid positions 4.1 System saves current configuration 5.1 System progress to Select study length</td></tr></table>		Actors	System	1. Student enter a degree code or title. 2. Student enter a major 3. Student selects electives from list 4. Student selects save button 5. Student selects next button	1.1 System returns degree matching code or title 2.1 System returns major courses matching major title 3.1 System assigns electives in valid positions 4.1 System saves current configuration 5.1 System progress to Select study length
Actors	System					
1. Student enter a degree code or title. 2. Student enter a major 3. Student selects electives from list 4. Student selects save button 5. Student selects next button	1.1 System returns degree matching code or title 2.1 System returns major courses matching major title 3.1 System assigns electives in valid positions 4.1 System saves current configuration 5.1 System progress to Select study length					
Exception Condition	1. If the student enters a degree code or title that does not match the database. A. The message “You must select a valid degree title or code” is displayed B. The system returns to wait state 2. If the student enters a major that does not match the database A. The message “You must select a valid major” is displayed B. The system returns to wait state					

Flow of Activities for scenario of: **Select Study Length**

Main Flow:

1. System enter wait state
2. Student enters numerical study length
3. System returns adjusted schedule
4. Student clicks next
5. System exits

Exceptions:

1. Student enters study length greater than maximum allowed period
 - a. System does not adjust schedule
 - b. System enters wait state

Name	Select Study Length					
Brief Description	Student can select a study length and the system will adjust the number of semesters and courses per semester accordingly					
Actors	Student					
Related Use Cases	Input Degree, Return Schedule					
Entry Condition	Input Degree is completed					
Exit Condition	Student saves study length					
Flow of events	<table><tr><th>Actors</th><th>System</th></tr><tr><td>1. Student enters numerical number of years to study 2. Student clicks next</td><td>1.1 System returns adjusted schedule 2.1 System progresses to Return Schedule</td></tr></table>		Actors	System	1. Student enters numerical number of years to study 2. Student clicks next	1.1 System returns adjusted schedule 2.1 System progresses to Return Schedule
Actors	System					
1. Student enters numerical number of years to study 2. Student clicks next	1.1 System returns adjusted schedule 2.1 System progresses to Return Schedule					
Exception Condition	<div>1. If the student enters number greater than the maximum allowed study period<div>a. The message “You must select a study period between ‘minNumYears’ and ‘maxNumYears’”.</div><div>b. System enters wait state</div></div>					

Flow of Activities for scenario of: View Schedule	
Main Flow:	<ol style="list-style-type: none"> 1. Student selects to view a complete schedule 2. System returns an existing schedule either in PDF or a HTML table 3. Student selects exit 4. System exits
Exceptions:	<ol style="list-style-type: none"> 1. Student tries to view incomplete table (ie Input degree has NOT been completed by student) <ol style="list-style-type: none"> a. System informs student to complete a schedule b. System exits

Name	View Schedule					
Brief Description	Use case that allows a student to view a completed schedule					
Actors	Student					
Related Use Cases	Select study length, Modify schedule					
Entry Condition	Student modifies schedule Student selects to view existing schedule Student completes advisory process for the first time					
Exit Condition	Student selects exit option					
Flow of events	<table><tr><th>Actors</th><th>System</th></tr><tr><td>1. Student elects to view schedule or Student completes advisory process. 2. Student selects exit</td><td>1.1 System return a non-editable version of schedule 2.1 System exits</td></tr></table>		Actors	System	1. Student elects to view schedule or Student completes advisory process. 2. Student selects exit	1.1 System return a non-editable version of schedule 2.1 System exits
Actors	System					
1. Student elects to view schedule or Student completes advisory process. 2. Student selects exit	1.1 System return a non-editable version of schedule 2.1 System exits					
Exception Condition	1. Student attempts to view schedule without first completing Input Degree a. System informs student that schedule is incomplete b. System exits					

Flow of Activities for scenario of: **Modify Schedule**

Main Flow:

1. System starts in waiting state
2. Student drags courses to valid positions
3. System allocates courses to valid positions
4. Student clicks save
5. System saves configuration

Exceptions:

1. Student drags course to invalid position (for example, placing a 3000 level course in the first year)
 - a. A message is displayed "Course position is invalid"
 - b. System enters wait state
2. Student moves course with an assumed knowledge before the preceding course
 - a. A message is displayed "WARNING. This course assumes students have complete 'assumedCourse', it is advised to complete 'assumedCourse' first".
 - b. System allocates change but highlights course with assumed knowledge

Name	Modify Schedule					
Brief Description	Stage of schedule construction that allows students to move courses by dragging them into ‘slots’					
Actors	Student					
Related Use Cases	Return Schedule					
Entry Condition	Student elects to modify existing shedule					
Exit Condition	Student selects exit					
Flow of events	<table><tr><th>Actors</th><th>System</th></tr><tr><td>1. Student elects to modify existing schedule 2. Student drags a course from slot A to slot B 3. Student clicks save</td><td>1.1 System returns schedule 2.1 Course swaps two courses from slot A to slot B 3.1 System saves new schedule</td></tr></table>		Actors	System	1. Student elects to modify existing schedule 2. Student drags a course from slot A to slot B 3. Student clicks save	1.1 System returns schedule 2.1 Course swaps two courses from slot A to slot B 3.1 System saves new schedule
Actors	System					
1. Student elects to modify existing schedule 2. Student drags a course from slot A to slot B 3. Student clicks save	1.1 System returns schedule 2.1 Course swaps two courses from slot A to slot B 3.1 System saves new schedule					
Exception Condition	2. Student moves 3000 level course to first year A. A message is displayed “Cannot enroll in 3000 level course in first year” B. System enters wait state 2. Student moves a course with assumed knowledge to slot before required course A. A message is displayed “WARNING. ‘CourseA’ assumes students have completed ‘CourseB’.” B. System assigns course swap but highlights problem course					

Flow of Activities for scenario of: Modify Degree	
Main Flow:	<ol style="list-style-type: none"> 1. System starts in wait state 2. Faculty staff submits changes to a degree 3. System updates database
Exceptions:	<ol style="list-style-type: none"> 1. Faculty member submits course with duplicant code <ol style="list-style-type: none"> a. System informs Faculty member about conflict b. Database is not updated

Name	Modify Degree					
Brief Description	Faculty member makes changes to degree (ie removal or addition of courses or majors)					
Actors	Faculty Member					
Related Use Cases	Inform enrolled students					
Entry Condition	Faculty member adds new subclass (such as course or major) to degree					
Exit Condition	Faculty member selects exit					
Flow of events	<table><tr><th>Actors</th><th>System</th></tr><tr><td><div>1. Faculty member submits change to degree</div><div>2. Faculty member save changes</div><div>3. Faculty member selects exit</div></td><td><div>1.1 System checks for conflicts in codes or course names</div><div>2.1 System updates database</div><div>3.1 System exits</div></td></tr></table>		Actors	System	<div>1. Faculty member submits change to degree</div> <div>2. Faculty member save changes</div> <div>3. Faculty member selects exit</div>	<div>1.1 System checks for conflicts in codes or course names</div> <div>2.1 System updates database</div> <div>3.1 System exits</div>
Actors	System					
<div>1. Faculty member submits change to degree</div> <div>2. Faculty member save changes</div> <div>3. Faculty member selects exit</div>	<div>1.1 System checks for conflicts in codes or course names</div> <div>2.1 System updates database</div> <div>3.1 System exits</div>					
Exception Condition	<div>1.1 Conflict in course codes or names</div> <div>A. System informs faculty member of conflict</div> <div>B. Database does not update</div>					

Flow of Activities for scenario of: Inform Students enrolled	
Main Flow:	<ol style="list-style-type: none"> 1. Faculty member modifies degree 2. System informs students enrolled in degrees or courses that have been modified 3. System exits
Exceptions:	<ol style="list-style-type: none"> 1. No students are enrolled in modified courses / degrees

Name	Inform students enrolled					
Brief Description	Informs students enrolled in changed course or degrees of changes					
Actors	Faculty Members					
Related Use Cases	Modify Degree					
Entry Condition	Faculty member saves changes to degree					
Exit Condition	Students are informed of changes					
Flow of events	<table><tr><th>Actors</th><th>System</th></tr><tr><td>1. Faculty member submits changes 2. Faculty member selects exit</td><td>1.1 System sends informs students enrolled in courses and / or degree 1.2 System exits</td></tr></table>		Actors	System	1. Faculty member submits changes 2. Faculty member selects exit	1.1 System sends informs students enrolled in courses and / or degree 1.2 System exits
Actors	System					
1. Faculty member submits changes 2. Faculty member selects exit	1.1 System sends informs students enrolled in courses and / or degree 1.2 System exits					
Exception Condition	1.1 No students are enrolled in modified courses.					