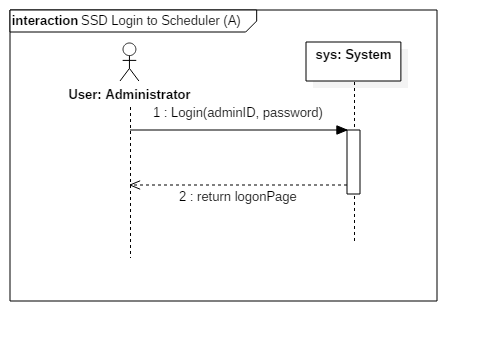
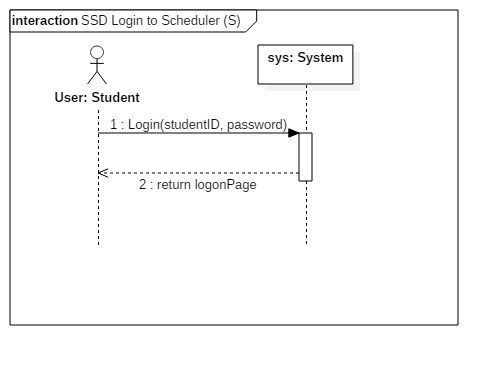
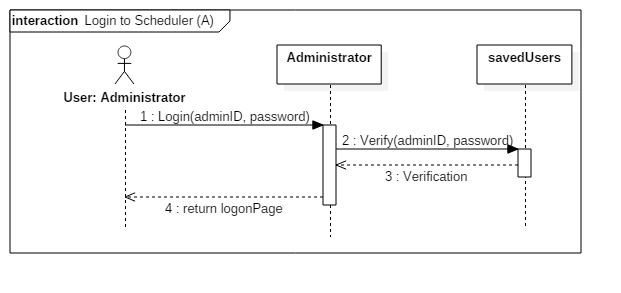
|  |  |  |
| --- | --- | --- |
| Use Case ID: | UC1 | |
| Use Case Name: | Login to Scheduler | |
| Created By: | Marc-Andre Leclair | Last Updated By: Claudia Della Serra |
| Date Created: | January 25th, 2016 | Last Revision Date: February 8, 2016 |
| Actor(s): | Student, Administrator | |
| Goal/Actor Goals: | A student or Administrator wants to Login in their account | |
| Description/Summary: | The Student or Administrator wants to login into their account. The initial webpage contains a login box where the he or she can enter their username and password. The request is sent and it is check in the database whether or not the account exist or if the password is correct | |
| Preconditions: | * The user has an internet connection. * The user has valid login credentials. | |
| Post-conditions: | The user is authenticated and logged in. | |
| Minimum Guarantee: | User will not log in | |
| Basic Flow: | 1. Student enters its login information 2. Server verifies if it is in the database 3. Student is logged in. | |
| Risk assessment: | Low | |
| Importance assessment: | 5/5 | |



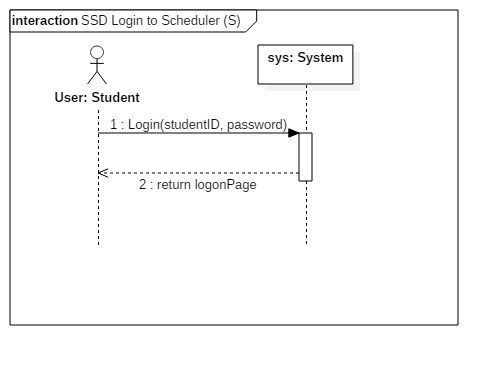


|  |  |  |
| --- | --- | --- |
| Contract 1.1 | Login | |
| Created By: | Ryan Lee | Last Updated By: Ryan Lee |
| Date Created: | March 17, 2016 | Last Revision Date: March 17, 2016 |
| Operation: | login(adminID: long) | |
| Cross-reference: | UC1 | |
| Preconditions: | * The user has an internet connection. * The user has valid login credentials. | |
| Post-conditions: | * The user is logged into their account. (instance creation) * A connection to the servers is established. (instance creation) | |

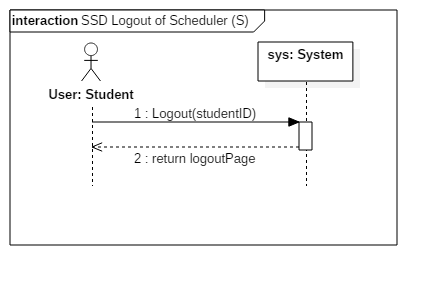
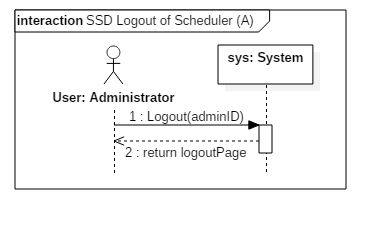


Contract logon for student

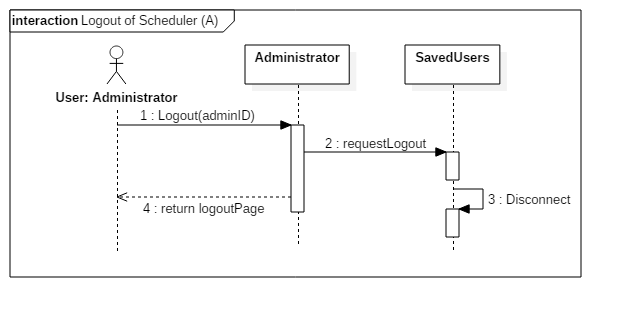
|  |  |  |
| --- | --- | --- |
| Contract 1.2 | Login | |
| Created By: | Ryan Lee | Last Updated By: Ryan Lee |
| Date Created: | March 17, 2016 | Last Revision Date: March 17, 2016 |
| Operation: | login(studentID: long) | |
| Cross-reference: | UC1 | |
| Preconditions: | * The user has an internet connection. * The user has valid login credentials. | |
| Post-conditions: | * The user is logged in. (instance creation) * A connection to the servers is established. (instance creation) | |



|  |  |  |
| --- | --- | --- |
| Use Case ID: | UC2 | |
| Use Case Name: | Logout of Scheduler | |
| Created By: | Marc-Andre Leclair | Last Updated By: Ryan Lee |
| Date Created: | January 25th, 2016 | Last Revision Date: March 17, 2016 |
| Actor(s): | Student, Administrator | |
| Goal/Actor Goals: | A student or Administrator wants to Logout of their account. | |
| Description/Summary: | The Student or Administrator wants to logout of their account. He or she sends a request to the server to close the connection between themselves and the Time Turner. | |
| Preconditions: | The user is logged in their account | |
| Post-conditions: | The user is disconnected from the server and logged out. | |
| Minimum Guarantee: | User will remain logged in | |
| Basic Flow: | 1. The user is already logged in his account. 2. The user indicates that they wish to logout of the system. 3. The server terminates the connection. 4. The user logs out through the webpage. | |
| Risk assessment: | Low | |
| Importance assessment: | 5/5 | |

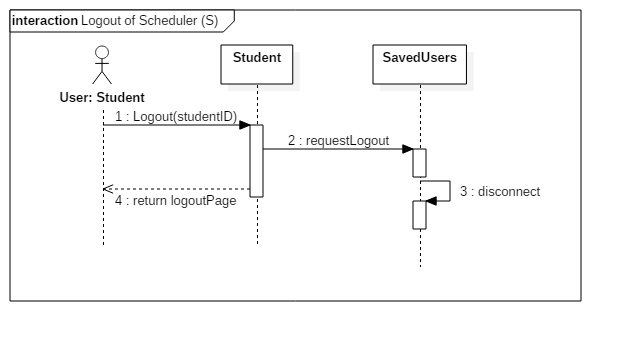


|  |  |  |
| --- | --- | --- |
| Contract 2.1 | Logout | |
| Created By: | Ryan Lee | Last Updated By: Ryan Lee |
| Date Created: | March 17, 2016 | Last Revision Date: March 17, 2016 |
| Operation: | logout(adminID: long) | |
| Cross-reference: | UC2 | |
| Preconditions: | * The user is logged into their account. | |
| Post-conditions: | * The user is disconnected from the server and logged out. (instance deletion) | |

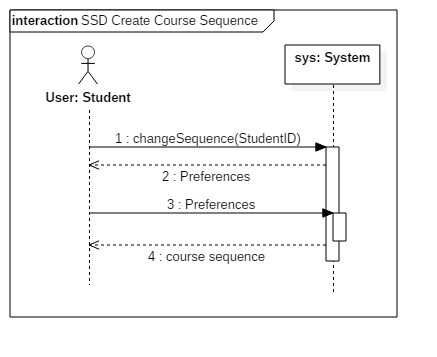


Contract logout scheduler student

|  |  |  |
| --- | --- | --- |
| Contract 2.2 | Logout | |
| Created By: | Ryan Lee | Last Updated By: Ryan Lee |
| Date Created: | March 17, 2016 | Last Revision Date: March 17, 2016 |
| Operation: | logout(studentID: long) | |
| Cross-reference: | UC2 | |
| Preconditions: | * The user is logged into their account. | |
| Post-conditions: | * The user is disconnected from the server and logged out. (instance deletion) | |

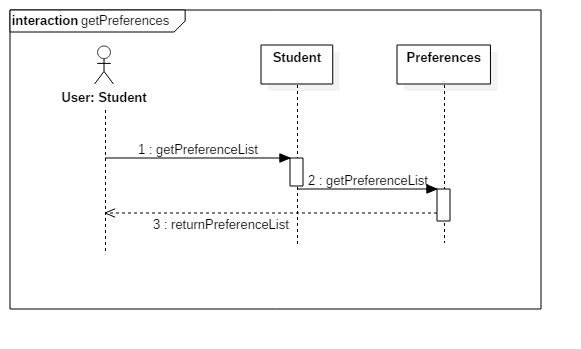


|  |  |  |
| --- | --- | --- |
| Use Case ID: | UC3 | |
| Use Case Name: | Create Course Sequence | |
| Created By: | Marc-Andre Leclair | Last Updated By: Ryan Lee |
| Date Created: | January 25th, 2016 | Last Revision Date: March 9, 2016 |
| Actor(s): | Student | |
| Goal/Actor Goals: | A student wants to create a course sequence | |
| Description/Summary: | The Student wants to create his or her course sequence. The system must provide the user with the option to add their own preferences in order to generate a sequence that is personalized to these preferences. | |
| Preconditions: | * The user is logged into the system * The user is registered in a program. | |
| Post-conditions: | The user has a personalized course sequence for the future | |
| Minimum Guarantee: | No course sequence will be created | |
| Basic Flow: | 1. Student requests to create a sequence. 2. The system must prompt the student to add his or her preferences to the sequence. 3. The student selects zero, one, or more preferences. 4. The system prompts the student to confirm their preferences 5. The student confirms his choices. 6. The course sequence is created and shown to the student. | |
| Risk assessment: | High | |
| Importance assessment: | 5/5 | |



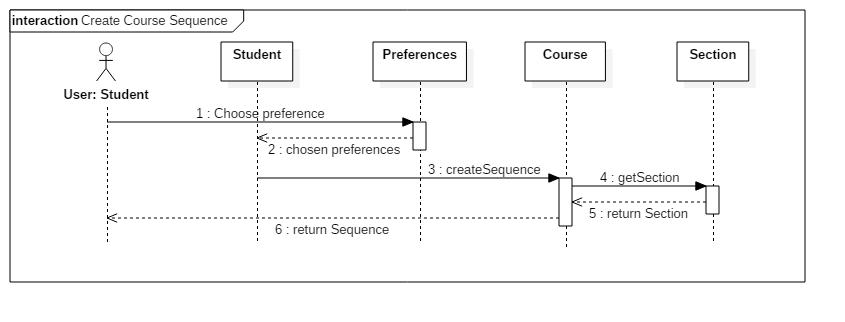
Contract for getPreference

|  |  |  |
| --- | --- | --- |
| Contract 3.1 | getPreference | |
| Created By: | Ryan Lee | Last Updated By: Ryan Lee |
| Date Created: | March 17, 2016 | Last Revision Date: March 17, 2016 |
| Operation: | getPreferences( ) | |
| Cross-reference: | UC3 | |
| Preconditions: | * The user is logged into the system * The user is registered in a program. | |
| Post-conditions: | none | |

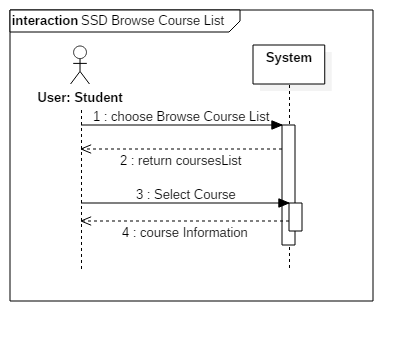


Contract create course sequence

|  |  |  |
| --- | --- | --- |
| Contract 3.2 | createCourseSequence | |
| Created By: | Ryan Lee | Last Updated By: Ryan Lee |
| Date Created: | March 17, 2016 | Last Revision Date: March 17, 2016 |
| Operation: | createCourseSequence(studentID: long) | |
| Cross-reference: | UC3 | |
| Preconditions: | * The user is logged into the system * The user is registered in a program. | |
| Post-conditions: | * The user has a personal sequence created for them. (attribute modification, association creation) | |

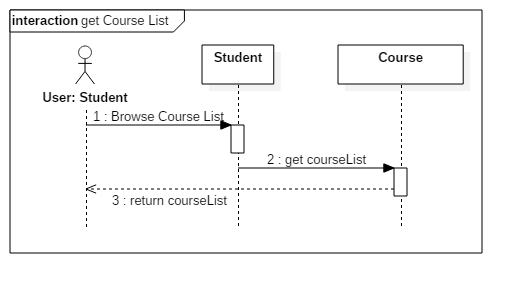


|  |  |  |
| --- | --- | --- |
| Use Case ID: | UC4 | |
| Use Case Name: | Browse Course List | |
| Created By: | Ideawin-Bunthy Koun | Last Updated By: Ryan Lee |
| Date Created: | January 25th, 2016 | Last Revision Date: March 9, 2016 |
| Actor(s): | Student | |
| Goal/Actor Goals: | Browse the list of courses from the course calendar. | |
| Description/Summary: | The user wishes to view available courses and access information pertaining to them. The system must display such information, including sections, times, and locations. | |
| Preconditions: | * User must be logged in as a Student. * Course list must be available. | |
| Post-conditions: | A list of courses is displayed. | |
| Minimum Guarantee: | The system fails to display a list of courses and displays an error message to the user. | |
| Basic Flow: | 1. Student selects the ‘Browse Course List” feature. 2. System retrieves list of courses. 3. System displays list of courses. 4. User selects a course from that list. 5. System displays information about the selected course. | |
| Risk assessment: | Medium | |
| Importance assessment: | 3/5 | |



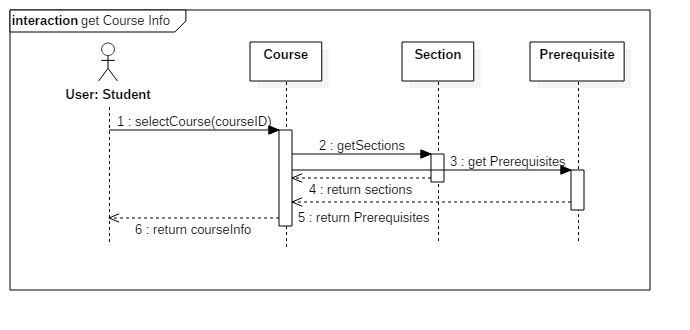
Contract get course list

|  |  |  |
| --- | --- | --- |
| Contract 4.1 | getCourseList | |
| Created By: | Ryan Lee | Last Updated By: Ryan Lee |
| Date Created: | March 17, 2016 | Last Revision Date: March 17, 2016 |
| Operation: | getCourseList( ) | |
| Cross-reference: | UC4 | |
| Preconditions: | * User must be logged in as a Student. * Course list must be available. | |
| Post-conditions: | None | |

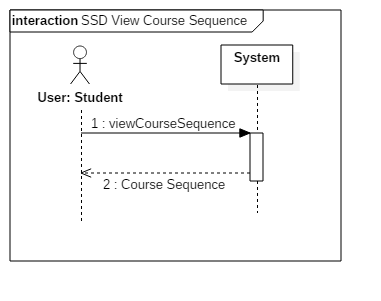


Contract get course information

|  |  |  |
| --- | --- | --- |
| Contract 4.2 | getCourseInfo | |
| Created By: | Ryan Lee | Last Updated By: Ryan Lee |
| Date Created: | March 17, 2016 | Last Revision Date: March 17, 2016 |
| Operation: | getCourseInfo(courseID: long) | |
| Cross-reference: | UC4 | |
| Preconditions: | * User must be logged in as a Student. * Course list must be available. | |
| Post-conditions: | None | |

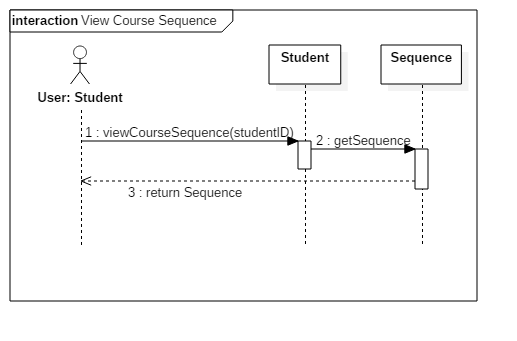


|  |  |  |
| --- | --- | --- |
| Use Case ID: | UC5 | |
| Use Case Name: | View Course Sequence | |
| Created By: | Ideawin-Bunthy Koun | Last Updated By: Ryan Lee |
| Date Created: | January 25th, 2016 | Last Revision Date: March 9, 2016 |
| Actor(s): | Student | |
| Goal/Actor Goals: | Allows the user to view their expected progression in the program. | |
| Description/Summary: | User can view a complete course sequence of the program the student is enrolled in. The Sequence will show the courses the Student is expected to take, and the order he or she is expected to take them in. | |
| Preconditions: | * User must be logged in as a Student * The system has access to a course sequence. | |
| Post-conditions: | The system displays a course sequence. | |
| Minimum Guarantee: | The system fails to display a course sequence and displays an error message to the user. | |
| Basic Flow: | 1. Student selects the option to view his or her course sequence. 2. System retrieves the sequence. 3. Course sequence is displayed to the student. | |
| Risk assessment: | Medium | |
| Importance assessment: | 4/5 | |

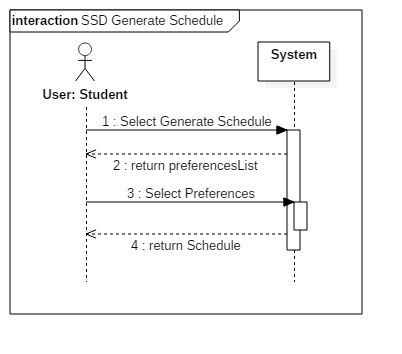


Contract course sequence

|  |  |  |
| --- | --- | --- |
| Contract 5.1 | viewCourseSequence | |
| Created By: | Ryan Lee | Last Updated By: Ryan Lee |
| Date Created: | March 17, 2016 | Last Revision Date: March 17, 2016 |
| Operation: | viewCourseSequence(studentID: long) | |
| Cross-reference: | UC5 | |
| Preconditions: | * User must be logged in as a Student * The system has access to a course sequence. | |
| Post-conditions: | None | |

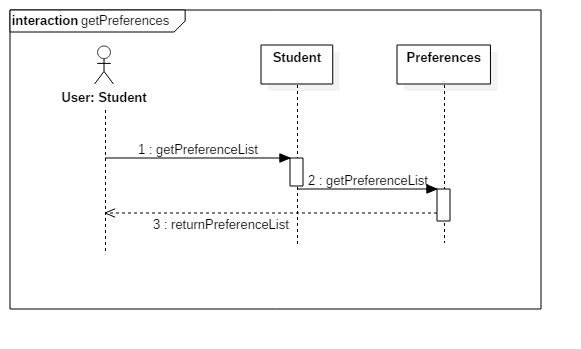


|  |  |  |
| --- | --- | --- |
| Use Case ID: | UC6 | |
| Use Case Name: | Generate Schedule | |
| Created By: | Ideawin-Bunthy Koun | Last Updated By: Ryan Lee |
| Date Created: | January 25th, 2016 | Last Revision Date: March 9, 2016 |
| Actor(s): | Student | |
| Goal/Actor Goals: | Obtain a suggested schedule for the next four years. | |
| Description/Summary: | The system can generate a 4-year schedule for the user based on the student’s preferences and constraints and on prerequisite and co-requisite policy. | |
| Preconditions: | * User must be logged in as a Student. * The system must have access to course database. * The system must have access to the student’s records. | |
| Post-conditions: | The system generates a schedule with no overlaps. | |
| Minimum Guarantee: | The system fails to generate a schedule and displays an error message to the user. | |
| Basic Flow: | 1. User selects the ‘Generate Schedule’ feature. 2. System retrieves list of preferences. 3. User selects his or her options and preferences. 4. System prompts user to confirm preferences. 5. System responds by displaying a schedule that meets the user’s preferences and constraints. | |
| Risk assessment: | High | |
| Importance assessment: | 5/5 | |



Contract Get preferences

|  |  |  |
| --- | --- | --- |
| Contract 6.1 | getPreferences | |
| Created By: | Ryan Lee | Last Updated By: Ryan Lee |
| Date Created: | March 17, 2016 | Last Revision Date: March 17, 2016 |
| Operation: | getPreferenceList | |
| Cross-reference: | UC6 | |
| Preconditions: | * User must be logged in as a Student. * The system must have access to course database. * The system must have access to the student’s records. | |
| Post-conditions: | None | |



Contract generate schedule

|  |  |  |
| --- | --- | --- |
| Contract 6.2 | generateSchedule | |
| Created By: | Ryan Lee | Last Updated By: Ryan Lee |
| Date Created: | March 17, 2016 | Last Revision Date: March 17, 2016 |
| Operation: | generateSchedule(studentID: long) | |
| Cross-reference: | UC6 | |
| Preconditions: | * User must be logged in as a Student. * The system must have access to course database. * The system must have access to the student’s records. | |
| Post-conditions: | * Schedule will be added to student’s saved schedules. (attribute modification, association creation) | |

