

Use Cases

1. Campers rank classes
2. Instructors offer classes
3. Administrator input classes' information to the system
4. System generates the schedule
5. Update classes' info
6. Manually add or delete a camper from classes
7. Authenticity management
8. Administrators add or remove class offering
9. Analyze courses for enrollment
10. Analyze students needs

1. Campers rank classes

Name	Camper rank class
ID	UC_001
Description	Campers rank courses they want, the course will be ranked by the number from height to low
Actor	Camper
Organizational Benefits	Well-organized structure of class ranking, easily identify which class is more popular
Frequency of Use	Once per section
Trigger	Campers press the "Rank Classes" button
Preconditions	All classes are set up to be ranked
Postcondition	Camper get a ranking list of classes that he desired
Main Courses	<ol style="list-style-type: none"> 1. System prompt user to log in 2. After logged in, show a list of available classes 3. User ranks classes 4. System stores the saved class ranking 5. User submit classes rank, and then system redirects the user to their saved ranking list to view the full list. 6. User confirms submission
Alternate Courses	<p>AC1: System determine user is not logged on</p> <ol style="list-style-type: none"> 1. Return to step 1 <p>AC2: User logs off again</p> <ol style="list-style-type: none"> 1. 2.System save current ranking 2. Return to step 1 <p>AC3: Ranking is not completed but submit</p> <ol style="list-style-type: none"> 1. Save current ranking 2. System prompts user "unfinished ranking list" 3. Return to step 3

	AC4: Rank more than the limit classes <ol style="list-style-type: none"> 1. System Prompt user “Over Limit” 2. Return to Step 3
Exception	User is not a camper <ol style="list-style-type: none"> 1. Return to step 1 System fails to registered class <ol style="list-style-type: none"> 1. System notifies user error occurs 2. Return to step 1

2. Instructors offer classes

Name	Instructors offer classes
ID	UC_002
Description	Instructors offer classes so that campers could be able to rank classes
Actor	Instructor
Organizational Benefits	Provide courses so that campers could be able to pick and rank which courses they want
Frequency of Use	Once for every section
Trigger	Instructors press “Offer Classes” button
Preconditions	System has been set up
Postcondition	System has a list of courses that instructors offer
Main Courses	<ol style="list-style-type: none"> 1. System prompt user to log in 2. After logging in,the user could be able to add or edit classes. 3. User adds a new class or edit a class 4. System stores the save classes 5. User submits and the system redirects the user to their saved items list to view their list of classes 6. User confirms submission

Alternate Courses	AC1: System determine user is not logged on 1. Return to step 1 AC2: User logs off again 1. System save current class but noted it as “unfinished” 2. Return to step 1 AC3: some courses are unfinished but submit 1. System prompts warning “unfinished class” 2. Return to step 3
Exception	User is not the instructor 1. Return to step 1 System fail to add class 1. System notifies user error occurs 2. Return to step 1

3. Administrator input classes’ information to the system

Name	Input information of classes
ID	UC_003
Description	Administrator input classes’s information like student number, section number and time to the system
Actor	Administrator
Organizational Benefits	Provide courses information so that the system could perform in a proper way.
Frequency of Use	Once for every section
Trigger	Administrators press “Input Class Info” button
Preconditions	Instructors have offered classes

Postcondition	System has a list of courses which instructors offer with certain information
Main Courses	<ol style="list-style-type: none"> 1. System prompts user to log on 2. After logging in, the user could be able to see the classes that were offered by instructors 3. User sets information for each class 4. System stores the save class information 5. User submits and the system redirects the user to their saved items list to view their list of classes with detailed information. 6. User confirms submission
Alternate Courses	<p>AC1: System determine user is not logged on Return to step 1</p> <p>AC2: User logs off again</p> <ol style="list-style-type: none"> 1. System save current class but noted it as “unfinished” 2. Return to step 1 <p>AC3: some courses are unfinished but submit</p> <ol style="list-style-type: none"> 1. System store the current information 2. Return to step 3
Exception	<p>User is not the instructor</p> <ol style="list-style-type: none"> 1. Return to step 1 <p>System fail to add class</p> <ol style="list-style-type: none"> 1. System notifies user error occurs 2. Return to step 1

4. Generate campers’ schedule

Name	Generate campers’ schedule
ID	UC_004
Description	After the system is locked down, an administrator runs the system to generate a

	schedule for each camper.
Actor	Administrator
Organizational Benefits	Increase system management and try to satisfy campers' needs
Frequency of Use	Once for every section
Trigger	Administrators press "Generate Schedule" button
Preconditions	System is locked down
Postcondition	System generate a schedule for each student
Main Courses	<ol style="list-style-type: none"> 1. System prompts user to log on 2. User confirms to generate schedule 3. System stores the schedule 4. System redirects the user to their saved items list to view their list of classes with detailed information. 5. System gives access of the schedule either by person or by course to instructors 6. System gives each camper access to his own schedule
Alternate Courses	AC1: System determine user is not logged on <ol style="list-style-type: none"> 1. Return to step 1 AC2: User logs off again <ol style="list-style-type: none"> 1. Return to step 1
Exception	User is not the administrator <ol style="list-style-type: none"> 1. Return to step 1 System fail to generate the schedule <ol style="list-style-type: none"> 1. System notifies user error occur 2. Return to step 1

5. Update classes's info

Name	Update classes's info
ID	UC_005
Description	Administrator can update/change classes' information.
Actor	Administrator
Organizational Benefits	Make the system more flexible.
Frequency of Use	Once every class information needs to be changed.
Trigger	Administrators press the "Update Class Info" button.
Preconditions	One or more classes need to be updated/changed, and this class already exists in the system.
Postcondition	The class information in the system has been updated.
Main Courses	<ol style="list-style-type: none"> 1. System prompt user log on After logging in, the User could be able to change the class information. 2. User updates/changes the class information that is already in the system. 3. System stores the new class information. 4. User submits and the system redirects the user to their saved items list to view their list of classes with detailed information. 5. User confirms submission
Alternate Courses	<p>AC1: System determine user is not logged on</p> <ol style="list-style-type: none"> 1. Return to step 1 <p>AC2: User logs off again</p> <ol style="list-style-type: none"> 1. System save current class but noted it as "unfinished"

	2. Return to step 1 AC3: some courses are unfinished but submit 1. System prompts warning “unfinished class” 2. Return to step 2
Exception	User is not the Administrator 1. Return to step 1 System fail to update/change class 1. System notifies user error occur 2. Return to step 2

6. Manually add or delete a camper from classes

Name	Add/Delete camper from classes
ID	UC_006
Description	User can add or delete camper information for any class
Actor	Administrator
Organizational Benefits	After the schedule has been created,the stored camper information can also be changed in case of emergency.
Frequency of Use	Once every camper information needs to be changed.
Trigger	User presses the”Add/Remove a camper” button.
Preconditions	The class schedule has already been stored in system
Postcondition	A camper is added to or removed from a class
Main Courses	1. System prompt user to log in

	<ol style="list-style-type: none"> 2. After logging in, the User could be able to change the class information. User adds or deletes the camper information form class that has already been stored . 3. User clicks the save button. 4. System stores the changed class information. 5. User submits and the system redirects user to review the new information page 6. User confirms submission
Alternate Courses	<p>AC1: System determine user is not logged on Return to step 1</p> <p>AC2: User logs off again System save new changed information status but noted it as "unfinished" 3. Return to step 1</p> <p>AC3: some courses are unfinished but submit 4. System prompts warning "unfinished class" 5. Return to step 2</p>
Exception	<p>User is not the Administrator 1. Return to step 1</p> <p>System fail to add or delete class 1. System notifies user error occur 2. Return to step 1</p>

7. Authenticity management

Name	Authenticity
ID	UC_007
Description	Campers, instructors and administrators can login in the system, new users can register an

	account.
Actor	Administrator, Instructors and Campers.
Organizational Benefits	Easy to identify the user identification and make the system more reliable and secure.
Frequency of Use	Every time user need to login/register an account
Trigger	Users press the “login” button or users press the “Sign in” button.
Preconditions	Does the database have the information?
Postcondition	Valid user login/register the account.
Main Courses	<ol style="list-style-type: none"> 1. System prompt users register an account. 2. User inputs his information. 3. System checks if it is a valid account 4. System confirms the information and stores the account information. 5. System returns to the login page . 6. User login. 7. System check if the account exists. 8. User enter system.
Alternate Courses	AC1: System find it is an invalid user <ol style="list-style-type: none"> 1. Return to step 1 AC2: User press login button <ol style="list-style-type: none"> 1. Return to step 5
Exception	

8. Administrator add or remove class offering

Name	Administrator add or remove class offering
ID	UC_008
Description	Administrator add or remove class offering

Actor	Administrator
Organizational Benefits	Administrator could edit a class offering in case of some emergency
Frequency of Use	Each time the class offering needs to be changed
Trigger	Administrators press “Add/remove Class offering” button
Preconditions	Is there a precondition necessary ?
Postcondition	A class offering is added or deleted
Main Courses	<ol style="list-style-type: none"> 1. System prompt user to log in 2. After logged in, show a list of available classes 3. User adds or delete a course offering 4. System stores the saved class list 5. Users submit, and the system redirects the user to their saved list to view the full list. 6. User confirms submission
Alternate Courses	<p>AC1: System determine user is not logged on</p> <ol style="list-style-type: none"> 1. Return to step 1 <p>AC2: User logs off again</p> <ol style="list-style-type: none"> 1. System save current ranking 2. Return to step 1 <p>AC3: add a course that already exists or delete a course that does not exist</p> <ol style="list-style-type: none"> 1. System prompts user “Conflict Action” 2. Return to step 3
Exception	<p>User is not an administrator</p> <ol style="list-style-type: none"> 1. Return to step 1 <p>System fails to add or delete course</p> <ol style="list-style-type: none"> 2. System notifies user error occur 3. Return to step 1

9. Analyze courses for enrollment

Name	Analyze courses for enrollment
ID	UC_009
Description	Administrator can analyze classes for enrollment such as status of low and high
Actor	Administrator
Organizational Benefits	According to the enrollment status of each class, the administrator can provide more popular class sessions and reduce the unpopular class sessions.
Frequency of Use	Once per section
Trigger	Administrators press “analyze enrollment” button
Preconditions	All campers have finished their enrollments
Postcondition	An analysis of classes enrollment status will be produced by the software.
Main Courses	<ol style="list-style-type: none"> 1. System prompt user to log in 2. User confirms to generate analy 3. System reads all enrollment data of all classes. 4. System sorts the enrollment from high to low. 5. System generates the enrollment analysis report to the administrator.
Alternate Courses	<p>AC1: System determine user is not logged on 1.Return to step 1</p> <p>AC2: User logs off again 1.System stops generating. 2. Return to step 1</p>
Exception	<p>User is not an administrator 1. Return to step 1</p> <p>System fail to generate analysis 1. System alert user with the error 2. System return to step 3</p>

10. Analyze students needs

Name	Analyze students needs
ID	UC_010
Description	Administrator can analyze students' needs
Actor	Administrator
Organizational Benefits	Analyze students' needs to better arrange time.
Frequency of Use	Each time after the administrator analyzing the enrollment.
Trigger	Administrators press "analyze-need" button
Preconditions	The enrollment analysis is done.
Postcondition	An analysis of students need will be generalized by software
Main Courses	<ol style="list-style-type: none"> 1. System prompt user log on 2. User confirms to generate analysis. 3. System reads the analysis of enrollment and students' needs. 4. System compare the enrollment data and students' needs 5. System generates the students' needs analysis report and new calendar to the administrator.
Alternate Courses	<p>AC1: System determine user is not logged on 1.Return to step 1</p> <p>AC2: User logs off again 1.System stops generating. 2. Return to step 1</p>
Exception	<p>System fail to generate analysis</p> <ol style="list-style-type: none"> 1. System alert user with the error 2. System return to step 3