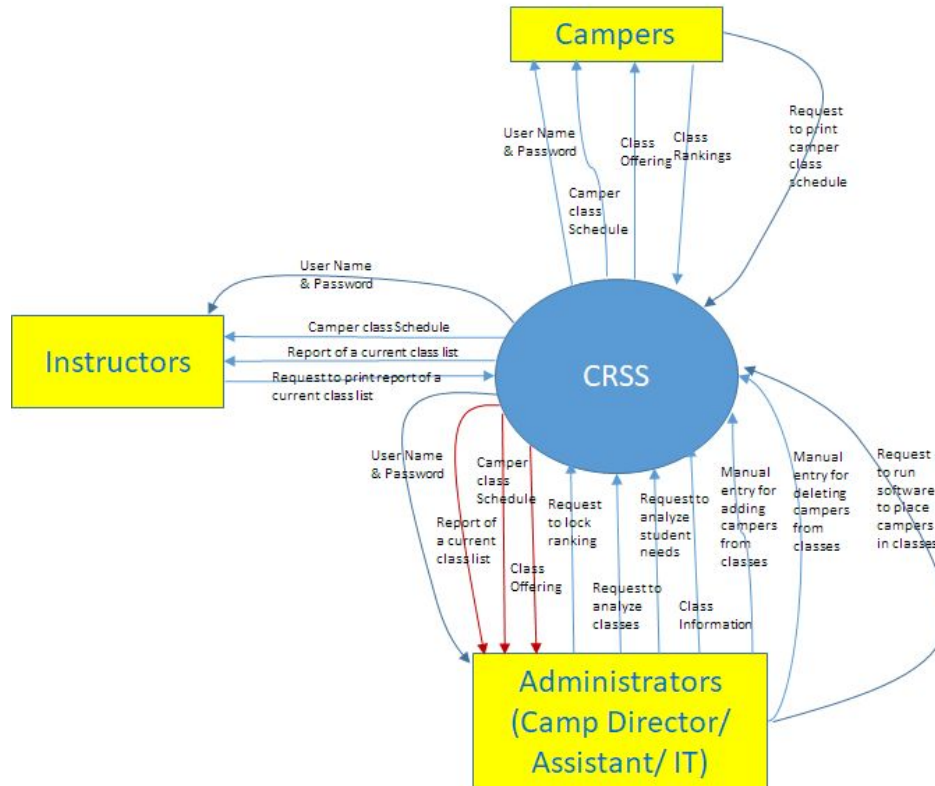


The Context Diagram



Question:

Who provides user information to the system?
(Not specified in given documents)

Here are some use cases that I think we may have (bryan) (in order of spec doc - Josh) (added a few more use cases based on context diagram and notes from class - Hailin):

- Campers rank classes - UC_1
- User logs in - UC_7
- Admins enroll students in classes - UC_5
- View camp schedules - UC_2
- >>>Print camper class schedule - NEED Colin UC_11
- Admins analyze class enrollment - UC_3
- Manually add/remove students to classes - UC_6
- Admins adjust class offerings and class info - UC_8
- Admins analyze student needs - UC_9
- Admins create class offering schedule - UC_4
 - Set blocks
 - Set camp sessions
- Instructors view class enrollment - UC_10

- >>>Admin lock ranking - Bryan UC_12
- >>>Print report of a current class list UC_13
- >>>System generates username and password (Hailin) UC_14

UC_1:

| | |
|-------------------------|--|
| Name | Campers rank classes |
| ID | UC_1 |
| Description | Camper rank 10 different classes based on their preferences |
| Actors | Campers |
| Organizational Benefits | Allow the system to take camper's preference into enrollment consideration. |
| Frequency of Use | 1 per camper or 1 per lock down <Question: Can campers change ranking before lock down?> |
| Triggers | Camper save or submit their ranking |
| Preconditions | Camper logged in. Ranking is locked. Camper has not submitted ranking before <This condition depends on the answer to the previous question> |
| Postconditions | A camper has a ranked class list satisfying following conditions: <ul style="list-style-type: none"> • 10 classes • No 2 classes have the same ranking • No class has more than 1 ranking |
| Main Course | <ol style="list-style-type: none"> 1. Camper successfully logged in 2. Camper select classes and choose rankings 3. Camper save or submit selections |
| Alternate Courses | AC-1: Camper is logged out before completing ranking <ol style="list-style-type: none"> 1. Camper is sent to login screen 2. Go back to main course step 1 AC-2: Camper selected less or more than 10 courses <ol style="list-style-type: none"> 1. Prompt camper to select more courses 2. Go to main course step 2 AC-3: Camper ranked two classes with the same ranking <ol style="list-style-type: none"> 1. Prompt camper to change rankings of those courses 2. Go to main course step 2 |
| Exceptions | Course information has not been put in the system Camper ranked two courses offered at the same time Camper ranked the same class with more than one ranking |

| | |
|--|---|
| | Camper selected same courses more than once |
|--|---|

UC_2:

| | |
|-------------------------|---|
| Name | View camper class schedule |
| ID | UC_2 |
| Description | Camper schedule will show the time blocks, the class enrolled in that block, the location of the class, any special instructions for the class and the instructor(s) names |
| Actors | Campers, instructors, administrators |
| Organizational Benefits | Camper needs to be able to see his/her schedule Instructors need to be able to see a camper's schedule Administrators need to be able to see all enrolled camper's schedules |
| Frequency of Use | Once every few days to multiple times a day |
| Triggers | Camper runs view camper class schedule Instructor, or system administrator run view camper schedule after selecting a camper. |
| Preconditions | A camper must log in successfully. An instructor must log in successfully. System administrator must log in successfully. Class assignments have been run. |
| Postconditions | Specific camper's class schedule displays on requester's screen |
| Main Course | MC-1: Camper view class schedule 1. Camper logged in 2. Camper click on view class schedule 3. Camper class schedule displays MC-2: Instructor/administrator view a camper's class schedule 1. Instructor/administrator logged in 2. Instructor/administrator select camper 3. Instructor/administrator click on view class schedule 4. Chosen camper's class schedule displays |
| Alternate Courses | AC1 - Camper/Instructor/administrator is not logged in 1. Camper/Instructor/administrator sent to login screen 2. Go to main course MC-1/2 step 1 AC2 - Instructor/Administrator did not choose camper 1. Prompt Instructor/Administrator to choose camper before view class schedule 2. Go to main course MC-2 step 2 |
| Exceptions | EX1 - Camper is not enrolled |

| | |
|--|---|
| | <ul style="list-style-type: none">1. No schedule is displayed <p>EX2 - Camper has not been assigned classes</p> <ul style="list-style-type: none">1. No schedule is displayed |
|--|---|

UC_3:

| | |
|-------------------------|---|
| Name | Administrators analyze class enrollment |
| ID | UC_3 |
| Description | Administrators analyze class enrollment to get statistics of enrollment (low/high) |
| Actors | Administrators |
| Organizational Benefits | Administrators need to be able to see how many people are enrolled in each class, so that they can make adjustments to offerings in the next session |
| Frequency of Use | Once after enrollment ends. Then run as needed. |
| Triggers | Administrators run 'analyze class enrollment' program |
| Preconditions | Administrators logged in with administrator rights. Class assignments have been run. |
| Postconditions | Record access. Report generation. |
| Main Course | <ol style="list-style-type: none"> 1. Administrator logged in (authentication) 2. Administrator select session 3. Administrator runs 'analyze class enrollment' |
| Alternate Courses | AC_1: Administrator does not select session <ol style="list-style-type: none"> 1. Pop up window alert 'No session was selected' 2. Go to step 2 in main course AC_2: Administrator did not log in or active session expired <ol style="list-style-type: none"> 1. Pop up window alert 'Need to log in as an administrator' 2. Go to step 1 in main course |
| Exceptions | EX_1: Class assignments has not been run <ol style="list-style-type: none"> 1. Pop up window alert 'Enrollment is not complete' |

UC_4:

| | |
|-------------------------|--|
| Name | Create class offering schedule |
| ID | UC_4 |
| Description | This item would place classes in a timeslot each day. <Q - Are there constraints on when the classes can be held? For example if astronomy is a class can it be held in the middle of the day?> |
| Actors | Instructors, Administrators |
| Organizational Benefits | The course schedule must be set up to provide classes and times for campers to choose from and to rank their preferences. |
| Frequency of Use | 1 time per summer session at most, if all classes are established prior to the summer for all sessions then may need to run only one time. <Q - Will classes need to be set up for each summer session independently or will all session schedules be set up at once?> |
| Triggers | Administrator validates that all course information has been entered into the system. |
| Preconditions | Instructors / administrators enter class information into the system. |
| Postconditions | The class schedules will be created for a given summer session or for all summer sessions. |
| Main Course | 1. An administrator is logged into the system. 2. The administrator validates that all class information is entered into the system. <Q - How do they know? Maybe class information needs to be tied to instructor and the administrator can verify that each instructor has uploaded information> 3. The system creates the class offering schedule. 4. The class offering schedule is saved. 5. The administrator is free to log out or perform additional tasks. |
| Alternate Courses | AC01 Instructor is double booked on time specific class. 1. Select the class with the least strict time requirement from the double booked instructor. 2. Place it in an open time slot (see EX01) AC02 Instructor is double booked on non time specific class. 1. Select one non time constraint class by the instructor. 2. Place class into an open slot. (see EX01) |

| | |
|------------|--|
| Exceptions | <p>EX01 No slot exists to place class</p> <ol style="list-style-type: none">1. Classes with conflicts are highlighted.2. Administrator manually adjusts the class schedule. <p>EX02 Not all classes have been entered into the system</p> <ol style="list-style-type: none">1. The administrator must manually input the class information and schedule the class. <p>EX03 The class offering schedule can not be saved.</p> <ol style="list-style-type: none">1. Administrator must fix the issue.2. Begin this use case again. |
|------------|--|

UC_5:

| | |
|-------------------------|---|
| Name | Enroll campers in classes |
| ID | UC_5 |
| Description | Places campers in classes according to the ranks created by the campers. |
| Actors | Administrators |
| Organizational Benefits | Campers will be placed into their classes by order of their ranks and in the order the campers submit their ranks. This helps campers stay happy about their choices and sets up the camp experience. |
| Frequency of Use | 1 time per summer session at most, if all classes are established prior to the summer for all sessions then may need to run only one time. <Q - Will this be ran once per summer session?> |
| Triggers | <p>Administrators will run the enrollment program. <Q - How will administrators know? Set up deadline for rank placements? Random otherwise? We should ask.></p> <p>Do we have a menu system? Do we write that? - SRS , prototypes of screenshots - system from scratch or using preexisting - NOT from command prompt</p> <p>ASSUME is some menuing system w/ windows - in UI can do protoyping / handdraw and take pic - "to do login user goes to menu and do login" -doesn't have to be pretty, just to determine what going on</p> |
| Preconditions | Class time schedules are established and camper rankings have been completed. |
| Postconditions | Campers are enrolled in unique classes in non-overlapping time slots based on input from each camper's ranking. Instructors can see their schedules. |
| Main Course | <ol style="list-style-type: none"> 1. An administrator is logged into the system. 2. An administrator determines it is ok to enroll campers into classes. 3. The system places campers into classes using class ranking input from the campers. 4. All camper schedules are saved. 5. The administrator is free to log out or perform additional tasks. |
| Alternate Courses | <p>AC01 A campers choice is not available.</p> <ol style="list-style-type: none"> 1. This camper gets placed in the front of the list for their next choice. (see EX01) |

| | |
|------------|--|
| | <p>2. Enrollment proceeds.</p> <p>AC02 A camper does not have any more rank preferences</p> <p>1. The next round of enrollment proceeds by placing students in random classes based on order students entered their information. (see EX01)</p> |
| Exceptions | <p>EX01 There are no open classes to complete a student's enrollment</p> <p>1. Students with incomplete schedules are highlighted.</p> <p>2. Administrator manually adjusts the class schedule.</p> <p>EX02 Not all campers have been enrolled in classes</p> <p>1. The administrator must manually add the non-enrolled campers to classes.</p> <p>EX03 The camper schedules can not be saved.</p> <p>1. Administrator must fix the issue.</p> <p>2. Begin this use case again.</p> |

UC_6:

| | |
|-------------------------|---|
| Name | Manually add students to classes |
| ID | UC_6 |
| Description | An administrator can add students to a class after the schedule has already been generated |
| Actors | Camp administrators/directors |
| Organizational Benefits | Simplifies the process of adding a student to a class after the schedule has been generated |
| Frequency of Use | Used when necessary to add/change a camper's enrollment |
| Triggers | User selects an option to edit a courses enrollment |
| Preconditions | User must be logged in as a director/administrator Class assignments have been run. |
| Postconditions | The camper has been enrolled in the class |
| Main Course | <ol style="list-style-type: none"> 1. An administrator is logged into the system. 2. Administrator selects the class that they would like to modify the enrollment of 3. Administrator adds camper to the section |
| Alternate Courses | <p>AC1 - User wants to remove a student from a course</p> <ol style="list-style-type: none"> 2. User selects a student to remove from class 3. User sent to main course step 2 <p>AC2 - User is not logged in</p> <ol style="list-style-type: none"> 3. User sent to login screen 4. User sent to main course step 2 |
| Exceptions | <p>EX1 - User tries to add a camper to a section that is full</p> <ol style="list-style-type: none"> 1. Alert user that the class is full 2. User sent to main course step 2 <p>EX2 - User tries to add a camper to a class that isn't enrolled in the current session</p> <ol style="list-style-type: none"> 1. Alert user that the camper is not enrolled in the camp 2. User sent to main course step 2 <p>EX3 - User is not logged in</p> <ol style="list-style-type: none"> 1. Alert user that they are not logged in 2. Send user to login screen |

| | |
|--|---|
| | <p>EX4 - User tries to enroll a camper in a section that would create a schedule conflict for the student</p> <ol style="list-style-type: none">1. Alert user that the camper is currently enrolled in another class at that time, let the user know what the other class is2. User sent to main course step 2 |
|--|---|

UC_7:

| | |
|-------------------------|---|
| Name | User logs in |
| ID | UC_7 |
| Description | Allow users to access the Class Rank & Sort System |
| Actors | Administrators, Instructors, Campers |
| Organizational Benefits | Only authorized users can access the Class Rank & Sort System |
| Frequency of Use | Used whenever a user needs to access the system |
| Triggers | User navigates to login screen |
| Preconditions | User must have valid username and password |
| Postconditions | A class will be created that will be available for campers to sign up for |
| Main Course | <ol style="list-style-type: none">1. User enters username and password2. User submits credentials3. User is logged in |
| Alternate Courses | AC1 - User decides they no longer want to log in <ol style="list-style-type: none">1. User exits program |
| Exceptions | EX1 - User enters incorrect username or password <ol style="list-style-type: none">1. User is alerted that their username or password is incorrect2. User is sent back to main course step 1 |

UC_8:

| | |
|-------------------------|--|
| Name | Adjust class offerings and class info - SHOULD THIS BE 2? |
| ID | UC_8 |
| Description | Administrators are able to manually add or remove class offerings, as well as adding class information to the system |
| Actors | Admins |
| Organizational Benefits | Administrators are able to have full control over what classes are offered when, and able to edit availability as needs arise |
| Frequency of Use | At minimum as many times as there are different classes offered for each session in order to approve instructor listings, before campers have access to ranking system As often as needed before and toward beginning of session for last minute changes to instructor/class availability or to cancel classes for low enrollment |
| Triggers | Instructors have submitted class offering proposals and availability Instructor availability or other need necessitates changing class availability |
| Preconditions | User must be logged in as administrator |
| Postconditions | Course proposals are approved, canceled, or edited |
| Main Course | <ol style="list-style-type: none"> 1. Administrator logs in 2. Admin views list of course proposals from instructors 3. Admin determines course proposal is satisfying, approves 4. Course is added to camp session offering list |
| Alternate Courses | <p>AC_01: Admin determines course proposal is not adequate</p> <ol style="list-style-type: none"> 1. Admin rejects course approval 2. Course proposal marked as "denied" in list of admin course options and instructor proposal submission form <p>AC_02: Admin determines course proposal requires modifications</p> <ol style="list-style-type: none"> 1. Admin selects edit option 2. Admin edits class details 3. Admin notifies instructor of edits to review and resubmit when ready <p>AC_01: Approved course requires details changed</p> <ol style="list-style-type: none"> 1. Admin selects edit option 2. Admin edits class details 3. Admin saves new class details |

| | |
|------------|---|
| Exceptions | <p>EX_01: User is not logged in</p> <ol style="list-style-type: none">1. User is redirected to login screen <p>EX_02: Admin edits or approves class for so that instructor is teaching 2 classes at same time</p> <ol style="list-style-type: none">1. Admin is alerted edit cannot be made2. Details reverted to original |
|------------|---|

UC_9:

| | |
|-------------------------|---|
| Name | Analyze student needs |
| ID | UC_9 |
| Description | Administrators are shown all cases in which a student does not have a full and valid schedule, such as having empty blocks, double booked blocks, or duplicate classes in different blocks |
| Actors | Admins |
| Organizational Benefits | As often as needed - likely very often after running ranking system and before/at the beginning of each camp session |
| Frequency of Use | Several times before each camp session |
| Triggers | Camper schedules have been provisionally set by the class preference ranking system |
| Preconditions | User must be logged in as administrator |
| Postconditions | Student(s) have no logical errors in their schedules, 15 unique classes per week |
| Main Course | <ol style="list-style-type: none"> 1. Admin logs in 2. Admin views list of student schedule conflicts/errors 3. Admin manually edits student schedules (UC_6) to remove duplicate courses, add student into open courses in empty blocks 4. Admin saves schedule changes |
| Alternate Courses | AC_01: No open, non-duplicate classes remain to enroll student in to fill schedule with unique courses <ol style="list-style-type: none"> 1. Admin edits capacity details for otherwise valid class (UC_8) 2. Admin adds student to non-duplicate class |
| Exceptions | EX_01: User is not logged in <ol style="list-style-type: none"> 1. User is redirected to login screen EX_02: Admin adds class creating another logical error <ol style="list-style-type: none"> 1. Admin is alerted additional conflict created 2. Details reverted to original |

UC_10:

| | |
|-------------------------|---|
| Name | View class enrollment |
| ID | UC_10 |
| Description | Instructors or admins are able to view report of enrollment for each class |
| Actors | Instructors, Admins |
| Organizational Benefits | Instructors are able to easily verify who is registered for their course(s) Admins are able to oversee who should be in each course |
| Frequency of Use | Used as necessary to view class list, verify attendance |
| Triggers | Class ranking system has been run, classes are provisionally enrolled with campers |
| Preconditions | User is logged in a instructor or admin |
| Postconditions | User is shown report of class enrollments for classes they teach (instructor) or any class (admin) |
| Main Course | <ol style="list-style-type: none">1. Instructor, admin logs in2. User selects to show class list for given class they teach (instructor) or any class (admin)3. User is shown report of currently enrolled campers for given class selected |
| Alternate Courses | AC_01: Instructor has been removed from class <ol style="list-style-type: none">1. Instructor told no longer has access to class, cannot rerun report AC_02: Class enrollment has been manually updated by admin <ol style="list-style-type: none">1. Instructor told enrollment has been updated, refresh to view |
| Exceptions | EX_01: User is not logged in <ol style="list-style-type: none">1. User is redirected to login screen |