HW 3: Design

Team: *Yvonne Rogell, Stephanie Peters, Emily Desmond*

A list of use cases:

- Student logs in….
  - Searches for classes
  - Registers for a class if: 1) not full, 2) prerequisites are met, 3) the addition will not exceed maximum credits for the student's level (undergrad/grad)
  - If class is full: student can send email to advisor with their details
  - Drops a class if: it's before the deadline
- Student is logged off…
  - Can search for classes
- Advisor adds student to waitlist
- Space opens in class - student gets removed from waitlist and added to class

Using the noun extraction approach, underline all noun phrases in our problem description:

On SU Online, students are able to search and register for classes. In order to add a course, students need to meet the prerequisites and the class may not be full. Students may also drop a class they decide not to take before the registration deadline. Students can only take a maximum of 18 credits if undergrad (student) or 12 credits if grad student.

Students must be logged into SU Online before attempting to add or drop, but can search for classes either logged in or not. If the class is full, students must send an email to their advisor with their name, SUID, and the course they want to take; students will be put on a waitlist for the requested course. If space allows, students will be automatically enrolled in the order they are listed on the waitlist.

Analysis of "probably classes," "possibly classes," and "rejected as classes".

Probably	**Possibly**	**Rejected**
Student	UndergradStudent - enough unique to justify separate class? maybe….	SU Online - because it is the system itself
Class	GradStudent - see above	Prerequisites - attribute of course
RegistrationDeadline - similar to due date in library ICL		Credits - attribute of course

Email - has a responsibility to transmit message		Name - attribute of student
Advisor		SUID - attribute of student
Waitlist		Space - attribute of waitlist
		Order - attribute of waitlist

Hidden class: RegistrationSystem, responsible for collection of classes available for registration in the system.

Comment on any relationships identified between classes (i.e. dependency, aggregation, inheritance).

Waitlist has a list of Students waiting to be enrolled in a Class (aggregation)

Student has an Advisor / Advisor has many students (aggregation)

Advisor manipulates Waitlist (dependency)

RegistrationSystem has a list of Classes (aggregation)

RegistrationSystem	
• Know registration deadline Date • Hold a list of available classes	• Class • Date

Student	
• Find a Class • Register for a Class • Send an Email • Know their name • Know their SUID • Know their max num credits • Know their undergrad/grad status • Know previous courses taken	• Advisor • RegistrationSystem • Class • Email

Class	
• Know the prerequisites • Know the number of credits • Know their class ID • Know status (open or full) • Know capacity • Hold list of enrolled students	• RegistrationSystem • Student • Waitlist

Date	
• Check whether a registration deadline has been passed.	

Email	
• Transmit Waitlist request from Student to Advisor	• Student • Advisor

Waitlist	
• Add a Student • Enroll Student in Class • Remove from Waitlist (after Student has been enrolled) • Knowing the Class's availability	• Class • Student • Advisor

Advisor	
• Receive requests from Email • Add Student to Waitlist	• Email • Waitlist