CS 200	**Project 3**	**100 points**
Fall 2017	**Design**	Nov 1, 2017 @ 11:59pm

Practical information

- Ask your questions on Slack or make an appointment to meet with the TA.
- Teams: As Assigned.
- Each team submits only *one full solution*.
- The submission should consist of a simple **HTML file** (called report.html) and any additional file required (.py, .txt, .jpg and whichever other files you want to submit). It should have explicit references to *all submitted files* as well as inline inclusion of images. In the header of the report file, you must specify: the *name* of the student, student's *CWID*, student's *email* address and the *time spent* on the assignment (in hours, for statistical purposes). Your solution should be entirely included in the body of the report file. Specify any extra assumptions or your own clarifications if needed. Then for each task, describe your solution in the form of a report.
- The **report.html** must be a complete report. Failure to producing this report will result in a penalty of 5% on your total grade.
- All submitted files should be referred to from within the report file, otherwise they will not be accessible during grading! Create a ZIP file archiving all your files. The archive should have only a folder as its root. The folder and the archive must have the same name. Submit the ZIP file in Blackboard.
- Only one team member (the submitter) shall submit the full solution (ZIP file). Mention who the submitter is in the **report.html**. Additionally, the **report.html must include a *Task Distribution* section**. This section should outline all the tasks that were performed and what percentage each team member contributed. If the balance is not evenly spread among all, the grades for each team member may differ. Each team member must submit the report.html, but with only the header, not the solution, and the Task Distribution section, affirming that you all agree on it.
- The submission medium is Blackboard.
- You may re-submit as many times as you wish before the deadline is passed.

Goals

This assignment will make you familiar *with the design of a software system and UML class & sequence diagrams*.

The grading scheme is as follows:

- Image & datafiles of your class diagram using VPP (40%)
- Image & datafiles of your sequence diagrams using VPP (45%)
- Discussion of your design (10%)
- Bitbucket statistics (5%)

Upload all images, datafiles, source files and result files to Blackboard and provide links to all of them from your report.html file.

Also include any additional information the corrector might require to correct the assignment.

Your submission should consist of one directory called: Design.

Assignments
Given the analysis of the requirements you have performed for the Chocoholics Anonymous (ChocAn) project, in this assignment, you will perform the design of the system.

Task 1
Draw the complete UML class diagram of the ChocAn system by **specifying all classes of the system and their relationships**. This will typically include more classes than you had previously identified during the analysis phase. For each class, show **all the attributes with their type and visibility** modifier. You should also show **all methods that you identified necessary with their full signature (with parameters and return types) and their visibility modifiers**. All associations must be clearly marked with **name, multiplicities, role names, and directions**. Each use case must be initiated by a distinct public method in your class diagram, which may invoke other private ones. Note that if the class diagram is too big, you may use aliases and packages to simplify the reading of the diagram. Write any assumption that you feel will help understand your design decisions.

Task 2
Draw one UML sequence diagram per use case you had identified. Remember that a sequence diagram corresponding to a use case is initiated by an actor who triggers the corresponding public method. Also, recall that a sequence diagram is not a full implementation, but only shows an overview of the messages sent between objects. **Make sure the sequence diagrams are consistent with your class diagram, meaning any message you send to an object should be a method of the receiver object.** Note that you may need to revise the use cases previously produced and revised during assignments 1 and 2 as your design decisions should be based on them.

Task 3
To verify your design, **examine the coupling and cohesion issues**. Discuss the coupling and cohesion of your design. Argument in a paragraph of **150 words (+/-10%) why your design is "good"**.

Don't forget to use your bitbucket repository. Commits page will be checked for 5%. Remember to commit/push all your files!

Additional Resources
The UML tool to use is Visual Paradigm (VPP).