Name: Andy Le

Team Members: Manreet Kaur, Gabriel Melara-Paz

Date: 5/17/25

Waterfall Method

Requirements Definition: The goal is to design a scheduling system software that provides a service where students, instructors, and administrators are able to add and remove courses, search courses, print schedules, and many other functionalities which are allowed for each user type. Some constraints include a hierarchy of user types where user cannot perform actions that an instructor can perform unless their actions overlap and an instructor cannot perform actions that an admin can perform unless given permission.

System and Software Design: The system must be able to allow user to enter their personal information such as first and last name, and ID. Then the program prompts must ask if the user is a student, instructor, or administrator. Then within each user type, depending on the action performed by the user, the program must have a database to access where it can grab the corresponding student or class to add or drop. Then the program must update the schedule of that user with the changes made and let's say if someone requests a class list of a class, that list must also be updated with the current students.

Implementation and Unit Testing: Code will be and has been written abstractly with the main functionalities working and so the next step will be to then implement the database to draw the relevant data to use and introduce code into the methods of each user type class that will save changes onto database.

Integration and System Testing: So far, the logic behind each method for every class has been defined, so now the relevant information must be drawn when those methods are called.

Operation and Maintenance: The program will be run various times through all possible selections for each user alongside testing the database for any inconsistent saves for each change. Bugs will be checked with proper debug techniques and the program will be tested beyond its capabilities such as case sensitivity for strings, invalid variable types for inputs, etc. Each phase of the waterfall model will then be revisited if needed after testing.

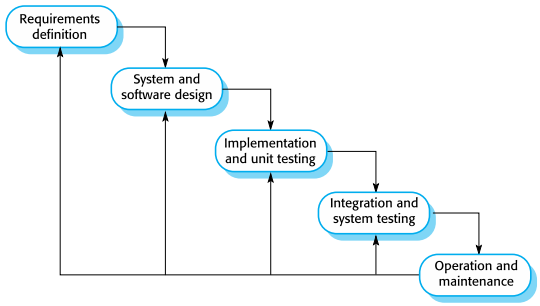

Figure L1-1: Diagram of each phase of Waterfall Model