Benson Chen

Prof. Rawlins

Assignment 2

5/29/22

**Incremental Model Diagram**

Initial version

Outline Description

Intermediate versions

Final version

**Description of each step (Incremental Model)**

**Outline Description:** The initial outline of the project is created describing the goal.

**Specification:** Specific details and requirements are included here and what can be done is determined.

**Design & Development:** The code is created based on specifications.

**Testing:** This phase works on user debugging to test whether it is suitable.

**Validation:**  User feedback is taken and is noted for future iterations of the code if needed.

**Versions:** Each version aims to add more functionality based on previous testing and feedback to provide a viable final product.

**Initial Version**

**Specification**

The LeopardWeb project has the goal of creating a scheduling system that will allow students, faculty, and admin to add courses, search for courses, print schedules and more.

• Database of users: the system should work for 100 students, 10 instructors, and 1 admin, however, we will test with fewer.

• Database of courses: this will contain information such as the CRN, course name, times, and instructor.

• Three types of users: o student – can register, can see available courses and their own schedule.

o instructor – can see available courses and their own course roster.

o admin – can see everything, can edit courses/users/schedules.

• The system should include multiple semesters, print-out of schedule, scheduling preferences.

• The system as a whole and all components must be tested thoroughly.

The base class of the system is user with:

• Attributes: first name, last name, ID.

• Methods: set function for each attribute, and a function to print all info for the object.

Requirement Validation: These requirements make sense, and they also match with what the user wants the system to do.

**Design & Development:**

This phase requires the creation and build of the code and databases. Three derived classes from user with unique functions would be created for each.

There will be three derived classes:

• All derived classes must contain any additional attributes and appropriate set/get functions.

• student – the student class will have functions that allow them to search courses, add/drop courses, print their schedule.

• instructor – the instructor class will have functions that allow them to print their schedule, print their class list, and search for courses.

• admin – the admin class will have functions that allow them to add courses to the system, remove courses from

**Testing:**

After initial completion of all components of the code, these components can be combined and tested. These components include specific user classes and functions.

**Validation:**

* This stage involves the latest version of the program and to see if it satisfies the specifications. If not, we trace back to development for future versions.