Brendan Gibbons

Applied Programming Concepts

Marisha Rawlins

Assignment 2

**Waterfall Model**

1. **Requirements Definition (4-5 Days)**
   1. The user will design a scheduling system for a university, such as LeopardWeb. This system will allow students, faculty, and administrators to add courses, search for courses, print schedules, and more.
   2. The requirements for this assignment are as follows:
      1. Database of users
         1. 100 students
         2. 10 instructors
         3. 1 administrator
      2. Database of courses, Will include:
         1. CRN
         2. Course name
         3. Times
         4. Instructors
      3. Types of users
         1. Student
            1. Able to register.
            2. Able to see available courses.
            3. Able to see own schedule.
         2. Instructor
            1. Able to see available courses.
            2. Able to see their course roster.
         3. Administrator
            1. Able to see everything listed above.
            2. Able to edit courses, users, and schedules.
      4. Should include multiple semesters, schedule printouts, and scheduling preferences.
      5. The system is required to be thoroughly tested.
2. **System and Software Design (2-3 Weeks)**
   1. The program will be constructed of a superclass for users, subclasses for students, instructors, and administrators, and a main class which contains the interface for the program (Add/remove users, print courses, etc)
   2. The classes included are listed as follows, with all criteria for the classes listed below:
      1. Base user class
         1. Attribute requirements:
            1. First name
            2. Last name
            3. ID
         2. Function requirements:
            1. Set functions for attributes.
            2. Print all functions.
      2. Student user class
         1. Attribute requirements:
            1. All attributes in user
            2. Major
            3. School
            4. GPA
         2. Function requirements:
            1. Set functions for attributes
            2. Search courses
            3. Add/drop courses
            4. Print schedule
      3. Instructor user class
         1. Attribute requirements:
            1. All attributes in user
            2. Department
            3. School
            4. Hire date
         2. Function requirements:
            1. Set functions for attributes
            2. Print schedule
            3. Print class lists
            4. Search for courses
      4. Administrator user class
         1. Attribute requirements:
            1. All attributes in user
            2. Division
            3. Hire date
         2. Function requirements:
            1. Set functions
            2. Add courses
            3. Remove courses
            4. Add/remove users
            5. Add/remove students
            6. Search and print rosters/courses
3. **Implementation and Unit Testing (1 Week)**
   1. The program will be tested with test cases which implement all the data for all 3 subclasses, and the functions will be tested to ensure they are behaving as expected to populate and store data within the database.
4. **Integration and System Testing (1 Week)**
   1. Implement the different classes to work together
   2. Connect the database for data storage
   3. Test system with sample user data and course information
5. **Operation and Maintenance (1 Week – Timeline of software usage)**
   1. Install the system for a university to use, regularly check for bugs, provide documentation for common issues people run in to, and push updates