Isabella DeCubellis

05/19/2025

Waterfall Model

Problem Statement:

Making a system to allow different types of users (student, instructor, and admin) to complete certain tasks specific to them.

Requirements Engineering:

1. Feasible Study
   - The technology needed does exist to create the system.
   - The technology does fit within the budget.

   Conclusion – The project is feasible, and we can continue.

2. Requirements elicitation
   - We have examined past systems such as the LeopardWeb application we already use. Some requirements are,
     - Allow users to login
     - Allow certain users to do certain tasks specific to that type of user.
   - Specifications from users
     - Will have a unique user ID to login with
     - Once logged in, they then have options to complete certain tasks
3. Requirements specification
   - All users
     - Have a unique ID specific to them
     - Will have first and last name
   - Students
     - Search courses
     - Add/drop courses
     - Print their schedule
   - Instructors
     - Print their schedule
     - Print their class list
     - Search for courses
   - Admin
     - Add/remove courses from the system
     - Add/remove users
     - Add/remove student from a course

        - Search/print rosters and courses
4. Requirements validation
    - Have other group members check that the requirements specified match the document

Design and Implementation:

1. Architectural design – high-level components: classes and objects, functions, database, and GUI.
2. Interface design – Have the separate components in separate files, then use them all in one main file and create a GUI.
3. Component design – Student class, instructor class, admin class derived from the base class user. The GUI will most likely be text-based.
4. Database design –

   User Table – First name, last name, ID number, and their status (student, instructor or admin)

   Student Table – ID, first name, last name, their major, and courses they are signed up for.

   Instructor Table – ID, first name, last name, program name, courses they are teaching

   Admin Table – ID, first name, last name, program name, students they manage, instructors they manage.

   Course Table – Name of course, CRN, instructor for the class, who is in the class, time class starts and ends, where the classroom is located.

Software Validation:

1. Component testing (unit testing) – test each component as we go so we can make sure each part is working properly.
2. System testing – after we integrate everything, test the overall system making sure it works together.
3. Acceptance testing – test the system with actual data

Software Evolution:

Modify the system as needed. If errors occur, fix them. If the user has an issue, tailor it to their needs.