Project Requirements

Project Requirement 1 - Conceptual Model

Build a conceptual (analysis) model showing either the state of concepts that your program uses or the data it uses as input or output. You should use UML class diagram notation to illustrate your model and describe it in text. However, as this model is focused on requirements, you should not describe a model of the actual or planned classes in your program (except by coincidence).

Your conceptual model should include classes with attributes (but not necessarily operations), and associations between classes annotated with appropriate multiplicities. Use aggregation instead of association to represent part-whole relationships. If there are concepts with a kind-of relationship, you should use inheritance. Your model should have 8-10 classes, including aggregation or inheritance, in addition to associations.

Project Requirement 2 - Software Architecture Document

Develop a Software Architecture Document that fulfills the functional and non-functional requirements of your software product. Include all the needed information for proceeding with detailed design. The structure of SAD should include the following:

1. Product Overview — Product vision, stakeholders, target market, etc.
2. Architectural Models — Specification using various models, both static and dynamic
3. Mapping Between Models — Tables and text relating models
4. Architectural Design Rationale — Explanation of difficult, crucial, and hard-to-change design decisions

Architectural models should be specified for at least two modules (e.g., Irrigation Layer and User Interface Layer). List all possible scenarios for evaluating your SAD and place them in the utility tree, describing at least one scenario in detail.

Project Requirement 3 - Detailed Design Document

Develop a Detailed Design Document (DDD) containing the following suggested structure:

1. Mid-Level Design Models
2. Low-Level Design Models
3. Mapping Between Models
4. Detailed Design Rationale

Draft a mid-level static model using the class diagram and describe DeSCRIPTR models for at least one module (e.g., Irrigation Layer). Include low-level design models with packages and describe operation specifications for a class.

Project Requirement 4 - Implementation

In this phase, implement the project based on the models developed in Requirements 1, 2, and 3.

Technical requirements:

- The code should use OO programming
- Include at least one architecture style
- Implement at least two mid-level patterns
- Initialize a Git repository
- Make commits for each part of the code
- Push to GitHub or Bitbucket

Note:

- The authentication module is not considered a module. - A simple CRUD for a model is not considered a module.