# Homework 1 Team Presentation – 10 Points

Please remember Your Homework 1 continuation will be Homework 2, and Homework 3. Finally, you will combine all your Homework 1, Homework 2, and Homework 3, to prepare your final written report. So, you will continue to work with the same teams for the entire semester. Select what project you want to work on as a team. You can choose any project related to health, welfare, global, cultural, social, environmental, and economic.

**Learning Objective**

The learning objectives for this HW-1 are to give you real-time exposure to using Agile or Waterfall methodology, the requirement phase from SDLC, and risk management.

**Deliverables**

* PPT.

**Video Presentation and Slides (Switch on your cameras for Video presentation):**

Each team will have a 12- 15 minute video presentation (if the time limit is more than 15 minutes, that is fine) that includes slides showing the following:

1. Title slide, including:
   1. Project title
   2. Date
   3. List of all team member names
2. Brief overview/description of the project
3. Requirements
   1. This is a list of requirements of the project. It should provide a list of *what* the software project should do without going into detail of *how* the software does it and also the priority of the implementations of the requirement. Format of the requirements should be same as mine

For example:

Functional requirement 1.1

ID: FR1

TITLE: Download mobile application  
DESC: A user should be able to download the mobile application through either an application store or similar service on the mobile phone. The application should be free to download.  
DEP: None

Functional requirement 1.2

ID: FR2

TITLE: Download and notify users of new releases  
DESC: When a new/updated version or release of the software is released, the user should check for these manually. The download of the new release should be done through the mobile phone in the same way as downloading the mobile application.  
DEP: FR1

Non-Functional Requirement 2.1

ID: NFR1

TITLE: Hardware needed  
DESC: Number of required hardware needed for implementing the functionality  
DEP: none

**Priority of Requirements**

|  |  |  |
| --- | --- | --- |
| Requirement ID | Title | Requirement Type |
| NFR3 | Hardware needed | Non - Functional |
| FR1 | Download mobile application | Functional |
| FR2 | Download and notify users of new releases | Functional |

1. Initial Development Plan
   1. Mention which methodology you will follow (Waterfall or Agile)
      1. If you are using a water fall model:

Create an implementation milestone like below showing how much time you will spend each task. (Requirement analysis, design, implementation, testing, final documentation), end date should be the final documentation submission date.

A screenshot of a computer

Description automatically generated

* + 1. If you are using a agile model:

Create an implementation milestone like below showing how much time you will spend each task. (Requirement analysis, design, implementation, testing, final documentation), end date should be the final documentation submission date.

A screenshot of a computer

Description automatically generated

1. Process Management:
   1. Understanding current similar projects that are already available, Reviewing and analyzing their practices.
2. Team Assignments and Responsibilities

* Mention what each member of the team did for their homework 1,
  + For Example, Person A: played the role of a project manager and he monitored everything was going on track
  + Person B: Researched regarding requirements.
  + Person C: Researched regarding requirements and also made the PPT for Homework -1.

1. Recording Link of your presentation

**Coding Requirements for Software Engineering Project**

**Core Requirements (Must-Have)**

These are the minimum features your project must implement to be considered complete and meet grading expectations.

**Basic Functionality Implementation**

* Implement the primary features of your application as defined by your project goals.
* Follow proper software design principles: modularity, separation of concerns, and clean code practices.
* Write well-structured code with clear comments and meaningful variable/function names.

**User Interface (UI)**

* Build a functional and user-friendly interface (web, desktop, or mobile) that allows users to interact with the core features.

**Optional Features or out of scope**

These add extra value and complexity, so they could be out of scope for this semester but if you want to try it, you can

**Advanced Features or Algorithms**

* Implement additional algorithms or optimizations beyond the core logic.
* Add predictive models, machine learning, or data analytics if relevant.

**Enhanced User Interface**

* Improve UI responsiveness, add animations, or create multiple views (e.g., admin vs. user).
* Make the UI mobile-friendly or implement a dedicated mobile app.

**Deployment & CI/CD**

* Deploy the application on a cloud platform (AWS, Azure, Heroku).
* Set up Continuous Integration/Continuous Deployment pipelines for automated testing and deployment.

**Advanced Security**

* Add multi-factor authentication, encryption, or role-based access control.

**Rubrics Scale:**

1. Title slide – 0.5 point
2. Brief overview/description of the project – 0.5 point
3. Requirements (including Priority of Requirements) – 3 points.
4. Initial Development Plan – 2 points
5. Process Management – 0.5 point.
6. Team Assignments and Responsibilities – 0.5 point
7. Recording Link – 3 points