Project Instructions.

You have been assigned to a group and a project. Your project work will be centered around the specification, design and implementation of a software product.

You may choose either of the projects below: You do not need to inform me of your choice – just get with your group and select one of these.

1. (Fitness App). Design and implement a mobile fitness app that will both guide a person to increase their fitness level and will track a persons effort surrounding diet and exercise. Exercises for muscle mass, fat control and hearth health should all be considered. App should be designed for Adult males and females over 18 or older . Can restrict to under age 70 if desired. User should be able to choose the desired set of Exercise from a list of appropriate exercises that fit the goals of the user given the users provided profile details. App should provide an exercise program and diet based on the exercises chosen Then app should allow for tracking of diet and exercise each day and week and modify users diet and exercise program elements based on weekly progress. Ease of Use and ability of game to adapt to users fluctuations in adhering to provided diet and exercise plan. Good Interface design is an essential consideration. Any choice of App platform, language is allowed for implementation.

2. (Fitness Game). Design and implement a game for young people in a selected range that will demonstrate (play out scenarios) of what is healthy behavior to engage in daily or weekly, how much , and what can happen if a person overeats, or under eats, never works out , works out too much etc. The game should let the user choose the behaviors and exercises that align with the games suggestions or that violate the games suggestions for optimum health. The age selection should guide the design of the Game. Choices of age ranges are: 5-8 , 10-12, 15-18. Could also restrict the game to only girls or only boys. Creativity and game engagement are essential elements of consideration. In other words, visually appealing, fun to play yet learn good habits and consequences of bad habits. Any implementation language or environment is allowed.

In the Modules section there are two Sample Projects that can be reviewed to help you with the creation of the life cycle documents surrounding Requirements – Design .

The project is a group project and is graded as such. However, the presentations and oral reports are graded individually.

Submission of project documents: You will submit 2 versions of your project –A Project Draft and a Final project. The draft will be a single pdf document that contains the required project deliverables for the draft as described below. Only one submission is needed per group. For the Final you will submit – the source code and executable in a .zip and a pdf of the life cycle document sections 1-5 described below and a video of a demo of the implementation as describe in section 6 below.

Capstone Project Draft and Final.

Failure to follow one or more of the below general requirements will result in grade of no better than a C.

General Requirements

- Project will be submitted in 2 Phases: project draft and final project.
- Document Type and Format. All project parts listed below must be submitted within a single MS Word Document that contains the sections named and numbered and formatted as shown below under.
    - Format: Font: Times New Roman, Single Spaced, 12 pt type, 1 column, and must include the sections named and numbered as detailed below in sections named Project Draft and Project Final.
- Refer to the sample projects provided to see examples of sections required. Note that the samples provide Many MORE sections and subsection than you are required to do. You are to ONLY do what is specified. Do not copy/provide sections or subsections not called for as doing so will reduce your score.
- Do not copy narrative sections from the project samples provided. This means DO NOT FOLLOW the project samples examples of including narrative sections that explain phases of the life cycle. Most likely the only narrative you may need to add may be within areas of the user manual and in your overall system description. For any narrative, you must follow proper use of paragraphs, must be spellchecked, and make use of complete and properly formed sentences. Avoid the use of lists of bullet items unless they directly represent aspects of the user interface.

  .

Project Draft:

The draft must include the following sections.

1. **User Manual** that defines the operation of the system for the user – see the user manual example in microwave oven sample provided.
    a. As part of the User Manual provide a set of user-interaction scenarios that illustrate and describe key interactions that will be implemented as part of your app or game.

2. **Requirements Analysis and Specification** – Analyze (create system models) and Specify all Requirements (Specify means write all requirements as unambiguous, complete, consistent and verifiable requirements statements).
    a. Overall System Description : Similar to what I provided but when some detail added to refine the scope , depth and breadth of the overall functionality of the system that you will then specify clearly in the models and system and user requirements in parts b. and c. of this section.
    b. Produce the Requirements Analysis: This means you must produce a minimum of 3 analysis models that model how your system will work. You must pick your system model types from this list of model types: Use Case Diagram, Inheritance Model, State chart, Structural Models, Data Flow Models, Semantic Data Models, Service Usage Models) . The Sample projects posted have examples of these types of models. You need to review these model within both sample projects to refresh how they work and

what they represent. You can also look for sample usage of these models on the web. There are a lot of samples to be found to illustrate each of these types of models.

c. Specify all system and user Requirements. Your requirements must be written as to be unambiguous, verifiable and non-amalgamated and should be close to complete to receive the best grade. You should review the requirements definition section in the sample models provided as an example of writing requirements.

Project Final.

The final project submission must include:

You must add the Architecture Design and Detailed Design and Implementation to sections 1-2 to form your final report. In other words you submit the entire project (sections 1-5) for the project final. Sections 3 – 5 must contain:

3. *Architecture Design. You must include the overall structural view of the architecture of the system including the major components and subcomponents of the architecture. You must also include the corresponding data dictionaries for this view of the architecture. There are examples of the structural view of an architecture given in the sample projects projected. You should review those.
4. *Detailed Design : Produce a set of Design models for the system. Produce 3-5 design models with a corresponding data dictionary for each model. These models should include at least the Control – Flow diagrams for the major control components of the software, a Sequence Diagram, and Collaboration diagram that details the major object interaction (also called object interaction diagrams) for the overall system and key subsystems. There are samples of these types of models in the sample projects posted in the Modules in Canvas. You need to review the detailed design sections of the provided sample projects.
5. Implementation : You will develop either a prototype or a full implementation depending on project depth and breadth.
6. Create a video demo of about 5 min that shows your software in use and demos the main features. Load the mp4 as a separate submission file or put the file onto YouTube and provide the link to the video in YouTube, or use echo360 in the class to create the video and provide a link to it. Link must work and video must play to get credit.

*Note: You may either model the entirety of the software in detail (all requirements taken through to detailed design) and provide the implementation of a prototype of the game consisting of the **main** functional requirements or you may instead fully implement all requirements so that the project is deep in implementation and shallow in design. If you choose the full implementation you may reduce Design portion in lieu of the time you spend on a full implementation.