[Project Name]

*Your proposed project must have the following overall requirements:*

* *It must be intended for mobile, wearable, or embedded-style platforms.*
* *You may be required to use data persistence pending the project type.*
* *The project must have significant depth and appropriate complexity given the size of your team and the time allotted.*
* *The responsibility for any technical training your project may require rests upon you and your teammates.*

# Project Description

Write out the description of your project. Describe what the project is about, the basics of what the product will do, and the audience for whom the product is intended.

Some example project types include:

* Mobile e-commerce client
* Health tracking application
* Security and activity monitoring within a given physical space
* Gamified personal improvement application
* Habit breaker/developer

# Part 1: Non-Functional Requirements

Non functional requirements describe the overall operation of a system, not specific behaviors or operations. Important information such as the runtime platform, system compatability, and overall quality requirements are listed here.

1. The system is to be an Android mobile application
2. Minimum supported specs:
   1. Android OS 4.0.3 (Ice Cream)
   2. Screen Resolution: 720 x 1280
   3. Screen Size: 4.7in
   4. Bluetooth 2.1
   5. Mobile data plan
   6. GPS-enabled
   7. Available storage space of at least 750MB
3. The system will be thouroughly tested prior to delivery.
   1. The system must be shown to have functional errors in no more that 15% of delivered requirements.
   2. Frame rate should average no less than 30fps

# Part 2: Functional Requirments

Below are an example functional requirements. Your team must fully define what your system will do. Language and terminology must be appropriate for laypeople, as this document is for the client. **Note:** *The following requirements are not complete!* The requirements must be validated with your client through interviews, emails, or other communication.

## Glossary

Before formally defining the functional requirements, it is necessary to define the domain-specific terminology to avoid confusion or ambiguity. For example:

* “System” – the delivered software and any additional resources required to make it function.
* “Book” – books, magazines, media or any other item meant to be used at or borrowed from the library.
* “Patron” – any person who wishes to borrow a publication.
* “Librarian” – a person representing the library who helps patrons find and/or checkout publications.

## Priority

Each requirement has a priority level of [1], [2], or [3].

1. Must-have functionality critical to the problem solution.
2. Highly desirable feature that should be included.
3. Optional requirements that will be completed if time allows.

## Books

1. A librarian can add a book to the system – see UC5. [1]
   1. The system stores the following required information about a book:
      1. Publication Type [1]
      2. Title [1]
      3. Author [1]
      4. Publisher [1]
      5. Publication Date [1]
      6. ISBN [1]
      7. Replacement Price [2]
   2. The system stores the following optional information about a book:
      1. Keywords [1]
2. A librarian can checkout books for a patron – see UC10 [1].
   1. Etc., etc., etc. (LOTS MORE REQUIREMENTS HERE)

# Part 3: Basic Design

Before developing any major project, you should first have a design and a plan. For this section, consider your logical architecture and produce a reasonably complete design to express it. Most commonly, a UML diagram will suffice, but supplementing that with wireframe designs of the UI, flow chart diagrams, or other design documents is acceptable and encouraged.

Your design should encapsulate the logical architecture (classes, interfaces, data members, methods, etc) and a component model as needed (these show the relationships between things like a server, mobile device, local application, etc).

# Part 4: Project Plan

You must define milestones for each phase of development. There will be 4 phases of development followed by the final product presentation at the end of the sprint. Each phase begins on the first day of the school week and ends on the last day. On the first day, each team will deliver a document containing the goals of each member of the team. On the final day of the phase, the instructor will meet with each team and ask for a demo. Each person must be able to demonstrate what he/she contributed during the week.

For this proposal, outline the major goals your team will have collectively completed by the end of each phase.