# DESIGN AND ARCHITECTURE DOCUMENT

Software architectural design is the process of applying various techniques and principle for the purpose of defining a module, a process, or a system in sufficient detail to permit its physical coding. The conventional approach to the software design process focuses on partitioning a problem and its solution into detailed pieces up front before proceeding to the construction phase. These up-front software architecture efforts are critical and leave no room to accommodate changing requirements later in the development cycle.

UML Architectural Modeling is one approach for performing modeling activities. It gives developers a guideline of how to build models-using an agile philosophy as its backbone-that resolve design problems and support documentation purposes but not “over-build” these models.

1. PROJECT INFORMATION
   1. Project Name
   2. Date of document submission
   3. Version of document (if any changes)
   4. Client (name of client company)
2. PROJECT TEAM

Describe how the project will be organized and managed. Identify reporting lines and outline specific roles that will be filled. You need to be clear about staff roles so that you don't duplicate responsibilities, and so that everyone is clear about what's expected of them. If this is a long-term project, you may even consider developing job descriptions for team members.

|  |  |  |
| --- | --- | --- |
| Team Member | Role | Document Responsibilities |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Also state which team member is responsible for writing which section in this document.

1. PROJECT SCOPE AND DEFINITION

Outline the strategic vision, goals & objectives and ideally include a high-level mission statement. This will help align the team on the approach and keep these goals in mind during solutioning. It will also help in defining additional work and potential project enhancements as the team keep this context in mind.

1. Define the project. What is the project about?
2. What is the scope of the project?
3. What is the problem to be solved?
4. What specific outcomes will be achieved? How will they be measured?
5. What are the deliverables?
6. What are the milestones for the deliverables?
   1. If possible, identify projected increments by Sprint (each sprint)
7. Are there any constraints that may influence your deliverables and schedule?
8. COMMON TERMS AND DEFINITIONS
9. ARCHITECTURAL AND ENVIRONMENTAL REQUIREMENTS
10. Hardware inventory, specifications, and locations
11. Inputs and outputs
12. Input and output devices
13. Servers with descriptions
14. Hosting
15. Development and production environments
16. Interfaces or integrations with other hardware
17. WATERFALL OR VISUAL MODEL

Recreate a waterfall model (not a use case, not sequence diagram, just our own diagram, however we want it to be), note how the diagrams changed?

1. USE CASE DIAGRAM VIEW

Include a Use-Case diagram here with a brief explanation of all the interactions.

HAVE INCLUDE AND EXCLUDES ARROWS.

1. BEHAVIORAL VIEW: SEQUENCE DIAGRAMS

Include at least 2 Sequence diagrams here with a brief explanation of all the interactions.

Returning something is dotted

1. DESIGN AND STRUCTURAL VIEW: CLASS DIAGRAM

Include a Class diagram here with a brief explanation of all the interactions.

Notice the relationships

Available upon request