# REQUIREMENTS DOCUMENT

A Requirements document provides an in-depth and comprehensive understanding of what the product specifications and user requirements are and how the software would accomplish it.

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

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

1. 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 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

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.

This section gives a high-level overview of the software application to be built, sets the tone for the project, defines what the long-term objectives and goals of the project are and gives all the team members working on the project absolute clarity.

Key items to be addressed:

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
7. Are there any constraints that may influence your deliverables and schedule?
8. Describe why the software (or upgrade) is being developed.
9. List the most important features and capabilities.
10. BURN UP CHART

Provide the most recent Burn Up Chart *with estimated trajectory* for the project.

1. SYSTEM ENVIRONMENT: USE CASE DIAGRAM

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

1. SYSTEM REQUIREMENTS AND FUNCTIONAL REQUIREMENTS

*Any requirement which specifies what the system should do and describes the system is a functional requirement.*

The functional requirements or the overall description documents include the product perspective and features, operating system and operating environment, graphics requirements, design constraints and user documentation.

The appropriation of requirements and implementation constraints gives the general overview of the project in-regards-to what the areas of strength and deficit are and how to tackle them.

Key items to be addressed:

1. Describe system functionality and features. Describe what the system must do and how it does it.
2. Describe system workflows.

|  |  |
| --- | --- |
| Functional Requirement | Conditions [User Story] |
|  |  |

1. EXTERNAL INTERFACE REQUIREMENTS

Interface requirements consist of the hardware and the software interfaces along with user and communication interfaces.

1. User interfaces consist of the style guides, screen layout, buttons, functions.
2. The software interfaces consist of the platform, database system, front end and the backend framework, operating systems, tools and libraries.
3. Hardware interfaces includes details of the hardware components like the list of supported devices, nature of data and the hardware-software interactions.
4. Communications interfaces are the network server communications protocols. The requirements determine the communication standards to be utilized.
5. SYSTEM, DESIGN, AND DEVELOPMENT CONSTRAINTS

*Constraints are conditions outside the control of the project that limit the design alternatives.*

Describe any high-level items that limit the developer's options for designing the software such as:

1. Standards (including hardware and software) Imposed on the Solution
2. Schedule
3. Budget
4. Preferred Software Programming Language(s)
5. Required Development Tools
6. Handling of Security Requirements (if any)
7. Potential Risks
8. Policy and Regulation

1. NON-FUNCTIONAL REQUIREMENTS

*A non-functional requirement describes how a system should behave and what limits are on its functionality. Non-functional requirements affect the user experience by defining a system’s behavior, features, and general characteristics.*

The non-functional requirements constitute the following:

1. Performance requirements. How quickly does the system respond to users’ actions, or how long does a user wait for a specific operation to happen? Examples might include:
   * Response time
   * Transaction performance
2. Capacity. What are your system’s storage requirements, current and future? How will your system scale up for increasing volume demands?
3. Resources such as memory, storage, communications
4. Centralized or distributed processing.
5. Security requirements. Does your product store or transmit sensitive information? Data or drive encryption? Authentication or authorization requirements? Asset security? (Laptops, physical equipment) External partner requirements, especially those around data access and data security. Include policies, standards as well as methods for controlling access like Citrix, Firewalls, VPN
6. Compatibility. What are the minimum hardware requirements? What operating systems and their versions must be supported?
7. Environmental. What types of environments will the system be expected to perform within?
8. Scalability
9. Availability
10. Reliability
11. Interoperability
12. Usability
13. Portability

|  |  |
| --- | --- |
| Non-Functional Requirement | Description |
|  |  |

1. DATA MANAGEMENT

Describe the data management requirements for the system, including the primary data sources and repositories. Also describe the data retention, archival, and warehousing.

1. CHANGE CONTROL

Describe how changes to the project scope are controlled, and who approves these.