**Functional Requirements Template**

|  |  |
| --- | --- |
| **Target release** | 1.0 |
| **Title** | Financial Analysis System |
| **Document Status** | Draft |
| **Document Owner** | Solomon Kirwa |
| **Designer** | Rufus Githinji |
| **Developers** | Kevin Wambugu |
| **QA** | Bismarck Metet |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **Product description** | | | | | |
|  | Define the purpose, scope and general requirements to meet the goal of the product.   1. Automated financial analysis process and reporting. 2. A web-based solution 3. Access to NSE data | | | | | |
|  | **Requirements** | | | | | |
|  | **S/N** | **Title** | **Description** | **Priority** | | **Notes** |
|  | 1. | Licensee accounts data | A user wants to receive licensee’s data in excel format | Must have | | We need to train licensees on how to upload the data on our platform. |
|  | **System configurations** | | | | | |
|  | Outlines steps needed to configure a product, such as registering a user account before signing in, admin unlocks a newly created account before the account owner can be able to access the system | | | | | |
|  | **Workflow within the system** | | | | | |
|  | For example, how approvals work | | | | | |
|  | **Outputs** | | | | | |
|  | For example, expected reports | | | | | |
|  | **User interface/wireframes** | | | | | |
|  |  | | | | | |
|  | **External interfaces** | | | | | |
|  | These areinterfaces that the project has with external entities for example external APIs for example linking with NSE API | | | | | |
|  | **Questions** | | | | | |
|  | Below is a list of questions to be addressed because of this requirements document. | | | | | |
|  | Question | | | | Outcome | |
|  | Should we have a batch processing aspect for automated analysis? | | | | We think it is important but not for version one | |
|  | **Not doing** | | | | | |
|  | Business intelligence to analyze every financial document- out of scope | | | | | |
|  | **Non-functional requirements** | | | | | |
|  | 1. Performance-explains how long a user must wait before the target operation happens (the page renders, a transaction is processed, etc.) given the overall number of users at the moment. 2. **Reliability**specifies how likely the system would run without a failure for a given period of time under predefined conditions. 3. **Availability**describes how likely the system is accessible to a user at a given point in time. 4. Security – ensures data inside the system will be protected against malware attacks or unauthorized access. 5. Usability - addresses the How hard is it to use the product?   T be able to achieve these requirements we must make the measurable and testable | | | | | |