# Solution Requirements [Functional & Non-functional]

| | |
|---|---|
| Date | 03 February 2026 |
| Team ID | LTVIP2026TMIDS74755 |
| Project Name | Rising waters: a machine learning approach to flood prediction |
| Maximum Marks | 4 Marks |

## Functional Requirements:

*Following are the functional requirements of the proposed solution.*

| FR No. | Functional Requirement [Epic] | Sub-Requirement [Story / Sub-Task] |
|---|---|---|
| FR-1 | Home Page | Displaying project overview and navigation options |
| FR-2 | User Input | Accepting environmental parameters such as temperature, humidity, cloud cover, and rainfall data. |
| FR-3 | Data Preprocessing | Applying scaling using StandardScaler before prediction |
| FR-4 | Prediction | Providing flood prediction result using trained XGBoost model. |
| FR-5 | Result Display | Showing “Flood Chance” or “No Flood Chance” in user interface. |
| FR-6 | Model Comparison | Comparing multiple ML models and selecting best based on accuracy. |
| FR-7 | Model Storage | Saving trained model and scaler using joblib. |
| FR-8 | Web Integration | Integrating ML model with Flask web application |

## Non-functional Requirements:

*Following are the non-functional requirements of the proposed solution.*

| FR No. | Non-Functional Requirement | Description |
|---|---|---|
| NFR-1 | Usability | The system should provide a simple and user-friendly interface without requiring technical knowledge. |
| NFR-2 | Security | The application should ensure safe handling of user inputs and prevent unauthorized access |
| NFR-3 | Reliability | The model should provide consistent prediction results for repeated inputs. |
| NFR-4 | Performance | The system should generate predictions quickly with minimal delay. |
| NFR-5 | Availability | The web application should be accessible anytime when deployed. |
| NFR-6 | Scalability | The system should allow addition of new features or larger datasets without affecting performance. |