

Project Design Phase-II
Solution Requirements (Functional & Non-functional)

| | |
|---------------|------------------------------------|
| Date | 11 November 2022 |
| Team ID | PNT2022TMID05871 |
| Project Name | Project - Personal Expense Tracker |
| 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 | User Registration | Registration through Email/SignUp Registration through Gmail |
| FR-2 | User Confirmation | Confirmation via Email Confirmation via OTP |
| FR-3 | Add expenses | Enter the everyday expenses Split it into categories(example : food, petrol,movies) |
| FR-4 | Reminder mail | Sending reminder mail on target (for ex : if user wants a reminder when his/her balance reaches some amount(5000)) Sending reminder mail to the user if he/she has not filled that day's expenses. |
| FR-5 | Creating Graphs | Graphs showing everyday and weekly expenses. Categorical graphs on expenditure. |
| FR-6 | Add salary | Users must enter the salary at the start of the month. |
| FR-7 | Export CSV | User can export the raw data of their expenditure as CSV |

Non-functional Requirements:

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

| FR No. | Non-Functional Requirement | Description |
|--------|----------------------------|---|
| NFR-1 | Usability | A simple web application which is accessible across devices |
| NFR-2 | Security | The OAuth Google sign in and email login are secure with hashed and salted secure storage of credentials. |
| NFR-3 | Reliability | Containerized service ensures that new instance can kick up when there is a failure |
| NFR-4 | Performance | The load is managed through the load balancer used with docker. Thus ensuring good performance |
| NFR-5 | Availability | With load balancing and multiple container instances, the service is always available. |
| NFR-6 | Scalability | Docker and Kubernetes are designed to accommodate scaling based on need |