# Authors: Demir Doruk Dilek

|  |  |  |  |
| --- | --- | --- | --- |
| Task | Demir Doruk Dilek | Tuna Yılmaz | Ahmet Kaan Tırhış |
| Functional Requirements | X | X |  |
| Non-Functional Requirements | X |  | X |

# Requirements Document

## Content:

1. Functional Requirements
2. Non-Functional Requirements

## Functional Requirements

1. User can add tasks.  
2. User can edit tasks.  
3. User can remove tasks.  
4. User can marks task as completed or custom tag.  
5. Task can have deadline according to date.  
6. User can activate notification and set reminders to remind user to do task.  
7. User can login or sign in to an old account.  
8. User can manage profile and remove their account.  
9. App can filter and sort the list by tag or date.  
10. App can handle backup and import data  
11. Users can have access to same to do list.  
12. To handle the joint to do list there must be a database to store these lists.  
13. There should be an account recovery system for users that forget them password.  
14. App can remind tasks according to location of user.

## Non-Functional Requirements

1. Application should have a fast response time, especially for basic operations such as adding, deleting, or editing tasks.
2. The application should maintain performance even when many tasks are added.
3. The interface should be intuitive and user-friendly, allowing users to easily add, edit, and delete tasks.
4. The application should provide consistent user experience across different devices.
5. User data should be encrypted.
6. Database security should be ensured, preventing unauthorized access to user data.
7. The application should securely store user passwords and have a secure authentication process.
8. The application should not experience performance degradation as more users add and manage tasks.
9. The infrastructure should be easily scalable if the number of users increases.
10. The application should not lose data in the event of a system failure and should handle errors properly.
11. The app should continue functioning even when there are network connection issues or database connection problems.
12. The application should back up user data regularly and allow users to recover their data in case of data loss.
13. The application should log user actions so that the history of actions can be traced if needed.
14. The application should be energy-efficient and not consume excessive system resources.

## Effort Estimations

• Requirements analysis: 10 hours  
• Design: 15 hours  
• Development: 50 hour  
• Test: 20 hours  
• Final analysis and publish: 10 hours  
Total Estimated time to done: 105 hours

To estimate the effort estimation we use method of expert judgment

## Task Matrix:

|  |  |  |  |
| --- | --- | --- | --- |
| Task | Demir Doruk Dilek | Tuna Yılmaz | Ahmet Kaan Tırhış |
| add tasks | X | X | X |
| Edit tasks | X | X |  |
| Remove tasks | X | X | X |
| Mark Tasks |  |  | X |
| Add Death Line | X |  | X |
| Notification and reminder |  | X |  |
| Login and sign in |  | X | X |
| Manage account |  |  | X |
| Filter and sort |  |  |  |
| Backup and importing |  | X |  |
| Joint list implementation | X |  | X |
| Joint list database implementation | X |  | X |
| account recovery | X | X |  |
| Location notifications | X | X |  |