Software Requirements Specification
SRS

“ORGANIZE IT”

Document history

Version	Status	Date	Responsible person	Reason for change
1.0	Created	20 Nov 2023	Team	Start of the project
1.1	Edited	23 Nov 2023	Team	Meeting for the SRS
1.2	Edited	24 Nov 2023	Nour Nassar	Added 4 Use Case descriptions
1.3	Edited	25 Nov 2023	Marcel Colic	Added 4 Use Case descriptions
1.4	Edited	25 Nov 2023	Michael Mezgolits	Added 4 Use Case descriptions
1.5	Edited	26 Nov 2012	Arik Kofranek	Added 2 Use Case descriptions and some NFRs

Glossary

spaces	One user’s or multiple different user’s synchronized work area encompassing tasks (within ToDo-lists) and Notes.
ToDo-list	Each ToDo-list belongs to a space and can contain multiple tasks.
tasks	Tasks can be created by users. Each task belongs to a ToDo-List.

Table of contents

1. Introduction **1**
1.1. Purpose, Goals and Background 1
2. Product scope **2**
2.1. Functional requirements 2
2.1.1. Must-criteria 2
2.1.2. Should-criteria 2
2.1.3. Could-criteria 2
2.1.4. Won't-criteria 2
2.2. Use case diagram 4
2.2.1. Use case descriptions 5
2.3. Non functional requirements 11
3. General overview **12**
3.1. Product application 12
3.1.1. Areas of application 12
3.1.2. Target groups, qualification level 12
3.1.3. Assumptions and dependencies 12

1. Introduction

1.1. Purpose, Goals and Background

"ORGANIZE IT" is a new multiplatform software designed to streamline task management in today's dynamic digital landscape. With the uprising in remote work and the need for seamless multi- and inter-platform communication, “ORGANIZE IT” addresses the challenges of staying organized and synchronized across various engagements.

This productivity tool offers work spaces for individuals, teams, students, flat share members and other collaborators. Its primary goal is to enhance productivity by providing a unified hub where tasks and notes can be managed, shared and synchronized effortlessly among multiple parties. “ORGANIZE IT” aims to facilitate a smoother and more efficient workflow and allows users to adapt their organizational framework to their unique needs and preferences.

2. Product scope

2.1. Functional requirements

2.1.1. Must-criteria

- The system must provide the user with the ability to create new spaces.
- If a new user is registered, the system must create a default space for that user.
- The system must provide the user with the ability to edit the title of a space.
- The system must provide the user with the ability to delete spaces.
- The system must provide the user with the ability to close and archive spaces.
- The system must provide the user with the ability to create to-do lists.
- The system must provide the user with the ability to edit to-do lists.
- The system must provide the user with the ability to delete to-do lists.
- After clicking on a specific to-do-list the system must provide the user with the ability to create tasks.
- After clicking on a specific to-do-list the system must provide the user with the ability to edit tasks.
- After clicking on a specific to-do-list the system must provide the user with the ability to delete tasks.
- After clicking on a specific to-do-list the system must provide the user with the ability to mark tasks as finished.

2.1.2. Should-criteria

- The system should provide the user with the ability to register to the service.
- Upon having registered to the service the system should provide the user with the ability to log in with an existing account.
- Upon having registered to the service the system should provide the user with the ability to log into their account on various other devices in order to access their own spaces.
- Upon having registered to the service the system should provide the user to stay logged in on various of their devices at the same time.
- After logging in on a device the system should be able to synchronize the user's spaces including to-do-lists and notes between logged in devices.

2.1.3. Could-criteria

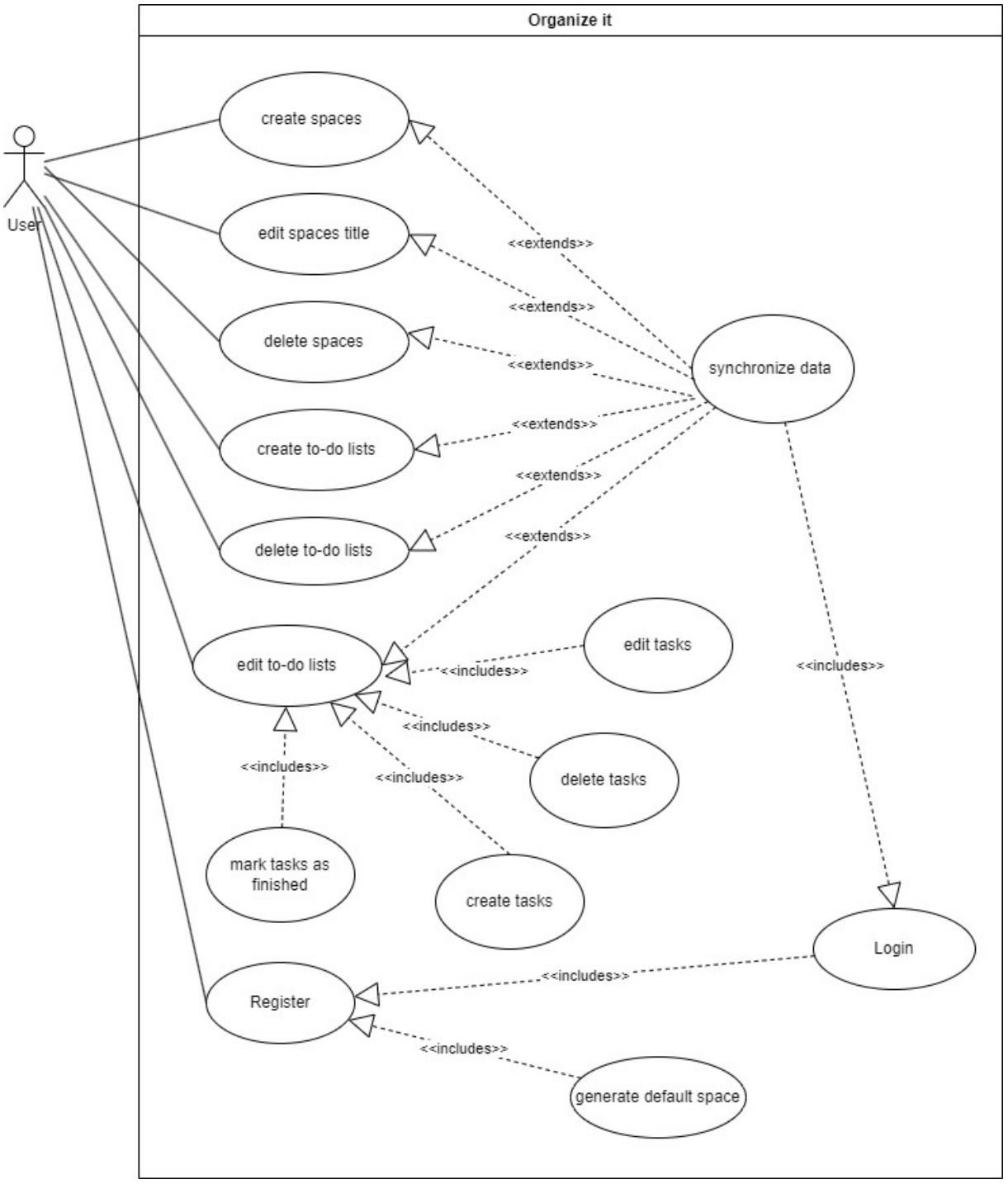
- The system could provide the user with the ability to add due-dates to tasks.
- The system could provide the user with the ability to create notes.
- The system could provide the user with the ability to edit notes.
- The system could provide the user with the ability to delete notes.
- After accessing the settings pertaining to a specific space the system could provide the user with the ability to invite other users to collaborate with them on that space.
- Upon joining a space the system could be able to synchronize the space's content across all of the collaborating user's various devices.

2.1.4. Won't-criteria

- The system won't provide the user with the ability to select AI generated task suggestions based on the title of a to-do-list.
- If a task has been marked as finished, the system won't automatically delete the task.

- The system won’t be able to send notifications.
- The system won’t provide the user with the possibility to customize spaces.
- The system won’t provide the user with the possibility to customize to-do-lists.
- The system won’t provide the user with the possibility to customize tasks.

2.2. Use case diagram



2.2.1. Use case descriptions

System	
Use case	Register
Actors	User
Description	The User initiates the process of creating a new account on the System to gain access to the platform's synchronization capabilities for spaces and to-do lists.
Stimulus	The stimulus for this use case arises when the user expresses the intention to register on the System. This may be driven by the desire to synchronize and manage spaces and to-do lists efficiently.
Response	Upon receiving the registration request, the System responds by creating a new user account. Simultaneously, a default space is automatically generated for the user. This default space serves as an initial organizational structure, ready for the user to manage and synchronize their spaces and to-do lists.
Comments	The registration process may involve the collection of necessary information from the user, such as username, email, and password, to establish a secure and unique account.

System	
Use case	Create to-do lists
Actors	User
Description	In this use case, the user intends to create a new to-do list within the system to effectively manage tasks.
Stimulus	The stimulus for this use case is the user's need to organize tasks by generating a new to-do list.
Response	The system responds by creating a new, editable to-do list for the user. This list serves as a blank canvas that the user can later modify, add tasks to, and delete as needed.
Comments	The system should provide an intuitive interface for users to easily create new to-do lists, possibly incorporating features such as naming the list.

System	
Use case	Delete to-do lists
Actors	User
Description	The user wishes to remove an existing to-do list from the system.
Stimulus	The stimulus occurs when the user decides to eliminate a to-do list, either because it is no longer relevant or as part of organizing their tasks.
Response	Upon receiving the deletion request, the system removes the selected to-do list, ensuring that it is no longer accessible within the user's account.
Comments	The system should confirm the user's intention to delete a to-do list to avoid accidental removal of important information.

System	
Use case	Edit to-do lists
Actors	User
Description	The user expresses the need to modify the content of an existing to-do list.
Stimulus	The stimulus arises when the user wants to add, remove, or edit tasks within a specific to-do list.
Response	The system responds by allowing the user to edit the content of the selected to-do list. New tasks can be added, existing tasks can be modified, and unnecessary tasks can be removed.
Comments	The editing functionality should be user-friendly, enabling users to make changes efficiently.

System	
Use case	Create spaces
Actors	User
Description	The user creates a space for a specific set of upcoming to-do lists.
Stimulus	The stimulus arises when the user thinks of a topic where the user wants to manage and track tasks.
Response	The system responds by allowing the user to create a new space.
Comments	The process to create a space should be made as easy as possible.

System	
Use case	Edit space title
Actors	User
Description	The user edits the name of a space.
Stimulus	The stimulus occurs when the user expresses the wish to re-name a space title.
Response	The system responds by allowing the user to edit the space title of a selected space..
Comments	The system should ask for confirmation after the user edited a space title..

System	
Use case	Delete spaces
Actors	User
Description	The user deletes the space
Stimulus	The stimulus occurs when the user thinks that a space is unneeded and that the space should be removed.
Response	The system responds by allowing the user to delete a selected space.
Comments	The system should confirm the user's intention to delete a space and warn the user that the tasks and to-do lists in this space will also be removed.

System	
Use case	Mark tasks as finished
Actors	User
Description	The user marks selected tasks as finished.
Stimulus	The stimulus occurs when the user finishes tasks and wants to mark them as finished.
Response	The system responds by marking selected tasks as finished.
Comments	The process to mark tasks as finished should be done in one single action.

System	
Use case	Create tasks
Actors	User
Description	The user creates a specific task within an to-do list.
Stimulus	The stimulus occurs when the user has a new task and adds it to the system.
Response	The system responds by creating the task and displays it in the user-interface.
Comments	The process to create a task should be as easy as possible.

System	
Use case	Edit tasks
Actors	User
Description	The user edits a specific task.
Stimulus	The stimulus occurs when the user wants to apply changes to a specific task.
Response	The system responds by changing the task and displays it in the user-interface.
Comments	The process to change a task should be as easy as possible.

System	
Use case	Delete tasks
Actors	User
Description	The user deletes a specific task.
Stimulus	The stimulus occurs when the user wants to delete a specific task from a to-do list.
Response	The system responds by deleting the task.
Comments	This should only be done, if a user presses “delete to-do” or deletes a whole list. It should **NOT** be done automatically, if a task is marked as completed.

System	
Use case	Generate default space
Actors	User, System
Description	A default space is created for the user.
Stimulus	The stimulus occurs when the registration of a user is successful.
Response	The system responds by creating a default space for the user.
Comments	This step is required so that the user can start using the app after registration.

System	
Use case	Synchronize Data
Actors	User, System
Description	A registered and logged in user, who uses the product on multiple devices, edits their content on one of their devices and intends to synchronize their changes to all other devices.
Stimulus	The stimulus occurs whenever the user chooses to synchronize their changes.
Response	The system uploads the user's changes from their local device to the backend components, updates the stored data and provides updates to be pushed to the user's other devices.
Comments	Logging in on devices is a prerequisite for synchronization of data across devices.

System	
Use case	Login
Actors	User, System
Description	A registered user wants to log in to their personal account on one of their devices.
Stimulus	The user wants to log in with their personal account.
Response	The system uploads the provided login credentials to the corresponding backend components and responds positively or negatively to them.
Comments	Registration in order to obtain a personal account is a prerequisite for Login.

2.3. Non functional requirements

2.3.1. Product requirements

2.3.1.1. Usability Requirements

- The app should run on multiple platforms namely: iOS, android, macOS and windows.

2.3.1.2. Performance Requirements

- The app's functionalities must respond within a reasonable timeframe (> 20 seconds) for 95% of all user requests made to the system.

2.3.2. Organizational Requirements

2.3.2.1. Development Requirements

- For software development Kotlin with Compose multiplatform is to be used.
- For development collaboration and code management github is to be used.
- Branches are to be used for individual feature development.
- Pushes are to be peer reviewed by at least 2 team members before merging a branch with the main branch is allowed.
- The system's codebase must be open source.

3. General overview

3.1. Product application

3.1.1. Areas of application

The areas of application are very broad: From professional work environments to flat shares, all sorts of organizations or private engagements can make use of this simple productivity tool.

3.1.2. Target groups, qualification level

“ORGANIZE IT” targets users of all qualification levels alike. The system is intended to be used by professionals in their projects as well as by private people in their daily lives. Since the “ORGANIZE IT” is planned as an multi-platform app (macOS, Windows, iOS, Android) most users of modern operating systems will be able to use the product.

3.1.3. Assumptions and dependencies

The only assumption pertaining to the use of Organize IT is that users own some end device (smartphone or desktop computer) with the respective rights and the knowledge in order to install and use the app.