High Level Design

ORGANIZE IT

version 1.4

30.11.2023

Document history

Version	Status	Date	Responsible person	Reason for change
1.0	Created	30.11.2023	Colic, Mezgolits, Nassar	Meeting
1.1	Edited	3.12.2023	Colic	Introduction, Server Overview
1.2	Edited	5.12.2023	Colic	Table of content format
1.3	Edited	10.12.2023	Nassar	Activity diagram added
1.4	Edited	12.12.2023	Team	Added all things needed for the presentation

Abbreviations

Table of contents

1.   Introduction

1.1.   Purpose

The purpose of "ORGANIZE IT" is to have a 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.
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.

1.2.   System Overview

The three main components of ORGANIZE IT will be the front- and backend and a database. Front- and backend will be done in Kotlin.

1.3.   Operating Conditions

ORGANIZE IT is a multiplatform software, which means that it will be compatible for mobile devices (iOS, Android) and desktop devices (Windows, MacOS, Linux).
The specific platforms are dependent on the development status of "Kotlin Multiplatform".

1.4.   Document Overview

This High Level Document contains following subjects:
- Introduction including the purpose of ORGANIZE IT and a short summary of the system
- An overview of various components with their role and functionality
- A class diagram, an activity diagram and a sequence diagram

2.   *Server* Overview

2.1.   Role of *Server* in the Overall System

The Server is used for the communication between the database and the users. Without a Server, ORGANIZE IT would not be synchronizable, meaning that users could not work together.

2.2.   Functionality

[Recherche notwendig was wir letztendlich als "Server" benutzen]

3.   Detailed Design
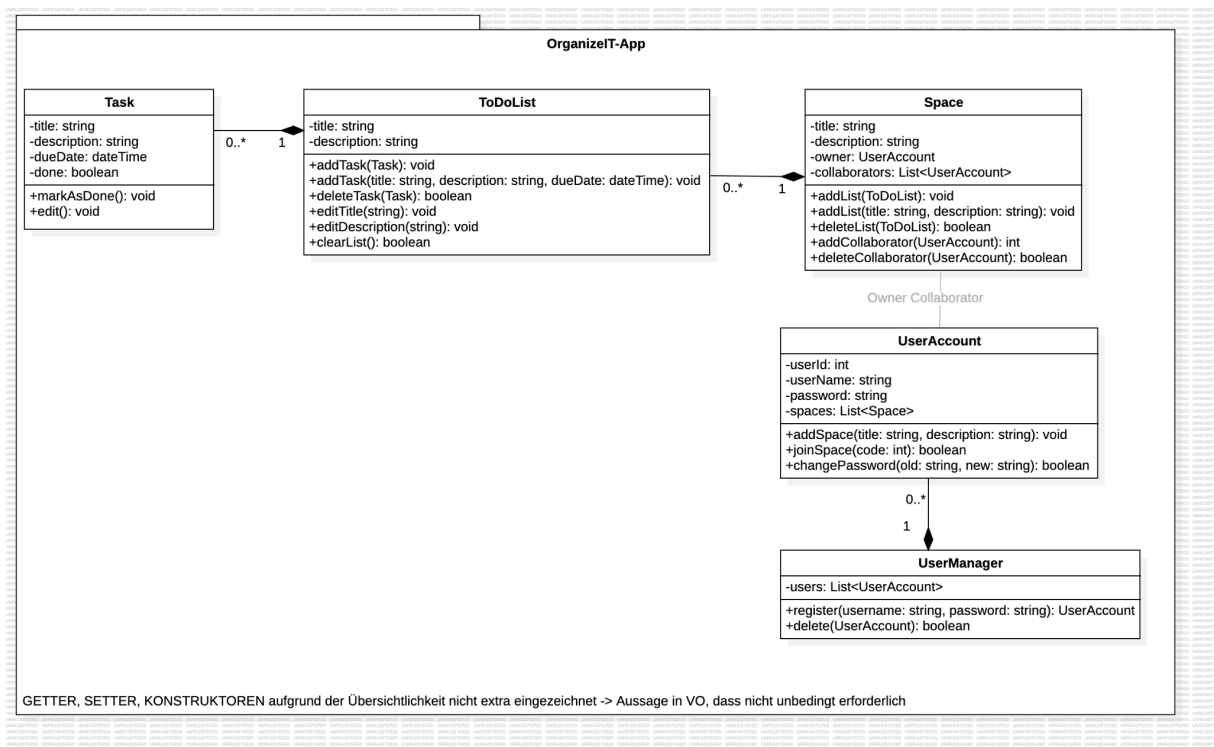
3.1.   Architecture

The project consists of three components:

- The client(s) - multiplatform
- The server
- The database

The exact technologies for the server and the database still have to be evaluated in the next exercises. The product "Ktor" looks very promising for the connection between client and server. The choice of technology for the connection whether HTTP endpoints or WebSockets still has to be determined after further tests.

3.2.   UML Diagrams

3.2.1. Class Diagram

3.2.2. Activity Diagram

3.2.3. Sequence Diagram

sd Edit ToDo-List

Lifeline1: User

Space

ToDoList

1 : editToDoList

2 : editToDoList

sd Login

sd Register