

Software Requirements Specification

Task planner

Version:

1.0

Summary:

A desktop app tasked with helping the users manage their time spent on certain tasks and assignments. This is achieved using the in-app calendar and to-do list as well as a time manager, which lets the user decide how much time they wish/need to spend on a task.

Date created:

30. 5. 2022

Author:

Ron Studený

Contents

Contents.....	2
1.0 Introduction.....	3
1.1 Purpose of this document	3
1.2 Document convention.....	3
1.3 Who is this document for?	3
1.4 Contacts.....	3
2. Overall description.....	4
2.1 Product as a whole.....	4
2.2 Program's functionality	4
2.3 Targeted user group.....	4
2.4 Software environment	4
2.5 User environment.....	4
2.6 Dependencies and assumptions	4
3. Interface requirements.....	5
3.1 User interface	5
3.2 Hardware interface	5
3.2 Software interface	5
5. System properties.....	6
5.1 Calendar	6
5.1.1 Importance and description	6

1.0 Introduction

1.1 Purpose of this document

To introduce the reader to the point of this project and all the functions and utility associated with it.

1.2 Document convention

Title describes the topic, and the underlying text provides additional detail. Names and titles of people are highlighted using **bold** font.

1.3 Who is this document for?

This document is dedicated to my supervising teacher **Ing. Pavel Švec**.

1.4 Contacts

Email: ronstudeny@gmail.com

Phone number: +420 606 338 678

2. Overall description

2.1 Product as a whole

The product will be a desktop app. The app will have the following features. Main functionality of the app will be assured by a calendar, in which a user can define any number of tasks. Each task can have a time interval, in which the user wishes to complete the said task. Once the defined time, dedicated to the task, expires a popup will ask the user, if they have completed the task and the answer will be recorded into a statistic.

2.2 Program's functionality

Main premise of the app will be the calendar, in which the user will be able to define a task and a time interval, in which they wish to complete the said task. Each task will be expandable to show details of what the task consists of.

Otherwise, the program will be running in the background and checking, if any of the deadlines have passed. If so, it will send a notification to the user, asking whether they have finished the task or not, if not, it will proceed to ask if it should prolong the deadline, or note the task as not complete. The result will be recorded into a database.

2.3 Targeted user group

The software is to be used by anyone, who wishes to enhance their time management abilities or simply wants to have an overview of their obligations.

2.4 Software environment

Software is designed to run on windows 8.1 or later.

2.5 User environment

The app itself will be as simplistic as possible, though some level of complexity will have to be employed, as not skip any functionality of the app.

2.6 Dependencies and assumptions

As stated, the app is dependent on the fact, that it is running on a windows machine. It is assumed that the user has a basic grasp of working with windows OS and apps associated with it.

3. Interface requirements

3.1 User interface

The UI should be as simple and user-friendly as possible since the whole premise of the app is to save and organize one's time. However, some level of complexity is required in order to fulfill all the software requirements.

3.2 Hardware interface

A standard keyboard and a computer mouse, or any valid alternative to such devices will be required to operate this software.

3.2 Software interface

The software will be realized using the WPF.

4. System properties

4.1 Calendar

4.1.1 Importance and description

The calendar is the main feature of the application. It provides the overview of all the tasks the user has set, the title and a deadline of the said task.

4.1.2 Inputs and outputs

The calendar provides the required information and is waiting for input from the user in the form of a mouse click.

4.1.3 Functional requirements

When a day in the calendar is clicked, it will show the tasks that are scheduled for the day. Clicking on the individual task will show a detailed description and deadline.

4.2 Task addition

4.1.1 Importance and description

This part of the app allows the user to add a desired task, assign a time limit and provide detailed description, that is later shown to the user, when the task is expanded.

4.1.2 Inputs and outputs

The user is expected to provide data necessary to create a task with all of its properties. It then provides a confirmation and records the task into the calendar

4.1.3 Functional requirements

Provides an easy way to add new and edit already existing tasks within the system.

5. Non-functional requirements

5.1 Performance

The application will not be resource heavy, most of the times it will run in the background. When the user is actively using the app, it won't take up much more of the computer's resources.

5.2 Security

Since the application runs locally on the user's machine and doesn't send any data to a database, there aren't many security risks.

5.3 Reliability

The application will use the system's time coordination, to ensure the program doesn't go out of sync with the actual time.