Software Requirements Specification

for

Orderly

Version 1.0 approved

Prepared by :

Himadri Pal, 15IT119

Information Technology Department, NITK

6th January 2017

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
Himadri Pal	06.01.17	Initial Draft	1.0

1. Introduction

1.1 Purpose

The main purpose of this document is to present a brief description of the application called Orderly. This document lists the constraints of the system along with all the features of the application. The 'Orderly', application works as a time management tool and allows you to schedule your work efficiently. The document ensures that the software requirements are properly understood. Also, the details of this document provide guidelines for future use and development of the product.

1.2 Document Conventions

Term	Definition
Orderly	Name of the application

1.3 Intended Audience and Reading Suggestions

This document is intended for a wide range of people from project managers who need to maintain a list of deadlines to homeowners who need to manage the resources of the house. Basically, this application targets any individual who needs to manage a number of tasks and schedule reminders. Further, it lists the functional and non-functional requirements of the product. Sequential viewing of the SRS is recommended.

1.4 Product Scope

Since the product is a resource management and job scheduling tool, it can be used in many environments ranging from corporate offices to schools. It's aimed at both personal and business markets. Furthermore, as the product develops, it would provide an option to integrate your business routines with personal or private jobs.

1.5 References

1.5.1 Django Documentation
https://docs.djangoproject.com/en/1.10/

2.   Overall Description

2.1   Product Perspective

Orderly is an independent software which aims to reduce the workload on individuals and companies by providing a workflow management or resource management tool. Simply stated, it's a to-do list with reminders and alarms which is intended to manage your work schedule. It has a graphical UI.

2.2   Product Functions

- Creation of accounts based on two classes - Group or Individual
- Group accounts has two different accesses - Admin and Regular
- The Admin or the Individual has the power to make changes and edit the schedules and deadlines in real time. (Admin has edit access in groups)
- Reminders and alarms can be added by the individuals or admins.
- Users can add tasks with deadlines. The task would be shown according to the deadline which closest deadline first.

2.3   User Classes and Characteristics

As mentioned earlier, the users will broadly classify into two classes:

- Group Owned

    UC-G1: Admin
    > He/she is the administrator of the portal. He/she has the ability to add projects, extend deadlines and add reminders.
    UC-G2: Regular
    > He/she has the ability to view the deadlines and the list of jobs which need to be completed.

- Individual Owned

    UC-I1: Individual

    > The characteristics of this user class is same as administrator of group owned class.

2.4   Operating Environment

- OE -1:   Orderly shall operate with the following Web browsers: Mozilla Firefox v9, Google Chrome v8, Microsoft Edge, all ideally in their latest versions.
- OE-2:   The server hosting the system should have Python 3 and Django 1.10.3 installed.

2.5 Design and Implementation Constraints

CO-1: The system shall use the current corporate standard sqllite database engine.
CO-2: All HTML code shall conform to the HTML 5.0 standard.
CO-3: All scripts shall be written in Python's Django Framework.

2.6 User Documentation

What all should be provided:

- Help option in the Application.
- "How to use" guide at the time of account creation and set-up.

2.7 Assumptions and Dependencies

In case of desktops and laptops behaving as servers, it is assumed that the user is running on a Windows OS or one of the variants of Linux.

3. External Interface Requirements

3.1 User Interfaces

- ❖ Welcome Screen:
  - ➢ For first-time users only (admin, regular or individual)
  - ➢ Prompts the user to enter account information to be stored on the server
  - ➢ Notifies the user if their information is invalid
  - ➢ Provides a quick tutorial which explains the different menus and options
- ❖ Error Screen:
  - ➢ In case of invalid username or password message is displayed regarding the specific issue.
- ❖ TO-DO List Screen:
  - ➢ Shows the tasks along with their deadlines. Buttons for adding or deleting tasks according to the user privileges.

3.2 Hardware Interfaces

The product can be hosted on all desktops, laptops with working internet connection.

3.3 Software Interfaces

The product is developed using Python 3 and the Django web development package. It requires SQLite database.

3.4   Communications Interfaces

No communication interfaces have been specified.

4.   System Features

4.1   New User Registration

4.1.1   Description and Priority

Facilitates the registration of new users. This is a high priority feature.

4.1.2   Stimulus/Response Sequences

Stimulus: User clicks the register button.

Response: User is directed to the register page and is prompted to fill up the relevant details.

4.1.3   Functional Requirements

- Input – User name, email id, password of user's choice.
- Processing – Username is checked for duplicates. Validation is carried out and a new entry in the database is created if successful.
- Output – User is registered, if he/she is not already registered.

4.2   Tutorial for new users

4.2.1   Description and Priority

New Users would be guided and shown on how to use the app. This is a low priority feature.

4.2.2   Stimulus/Response Sequences

Stimulus: User keeps clicking the next button till the tutorial is finished.

Response: User is shown around the app.

4.2.3   Functional Requirements

No functional requirements have been specified.

4.3   Individual / Admin / Regular User Login

4.3.1   Description and Priority

Registered Users will be able to login to the system using this feature.

4.3.2   Stimulus/Response Sequences

Stimulus: The user enters the login details prompted by the system.

Response: The system processes the User information.

4.3.3   Functional Requirements

- Input – User email ID and password.
- Processing – The system authenticates the user email ID and password.
- Output – If authentication is successful, he/she is logged onto the system. If not, the User is requested to type the correct credentials.

4.4   Adding Tasks

4.4.1   Description and Priority

New Tasks are added using this feature. This feature can only be accessed by admin or individual users. It's a medium priority feature.

4.4.2   Stimulus/Response Sequences

Stimulus: The user enters the deadline and the task name add clicks the add button.

Response: The system adds the task to the database and it is displayed on the screen.

4.4.3   Functional Requirements

- Input – The type of task and it's deadline should be correct.
- Processing – A new database entry is created for the task.
- Output – The task is displayed on the dashboard.

4.5   Deleting Tasks

4.5.1   Description and Priority

User (with appropriate privileges) can delete tasks whose deadlines have or have not been crossed.

4.5.2   Stimulus/Response Sequences

Stimulus: The user selects the task to be deleted and clicks the delete button.

Response: The system removes the task from the database and it is removed from the dashboard.

4.5.3   Functional Requirements

- Input – The appropriate task should be selected.
- Processing – The database entry is removed.
- Output – The task is removed from the dashboard.

5.   Other Nonfunctional Requirements

5.1   Performance Requirements

PE-1: Responses to tasks shall take no longer than 7 seconds to load onto the screen after the user submits the task

PE-2: The system shall display confirmation messages to users within 4 seconds after the user submits information to the system

5.2   Safety Requirements

No Safety requirements have been specified.

5.3   Security Requirements

SE-1: Users shall be required to log in to "Orderly" for all operations.

SE-2: The system shall permit only administrators in the case of groups to add or delete tasks.

5.4   Software Quality Attributes

Availability: Orderly shall be available to users for as long as the administrators keep the servers running.

Interoperability: The system can be set up in both Windows and Linux OS.

Portability: As long as the servers are running, your tasks can be accessed and seen from any device having a browser support.

Maintainability: Since Orderly is built in modules, it is easy to integrate changes or add functionality without affecting the other components.

Reliability: Changes made to the database are always accompanied with the appropriate notification on the user screen.

Usability: The presence of the welcome screen / "How To Use" guide eases the learning process. The interactive and simple UI also facilitate easy learning.

5.5   Business Rules

In the case of groups in Orderly, only the administrator has the privileges to add,modify or delete the tasks or projects. Regular users have only viewing rights. In case of individual accounts, the individuals are permitted to add, modify and delete tasks.

6.   Other Requirements

No Other Requirements have been specified.

Appendix A: Glossary

Python: A high-level programming language
Django: A web development framework written in Python
Server: A machine hosting the web application
Individual: A user with all privileges
Admin: A user with special privileges
Regular: A regular user with only viewing privileges

Appendix B: Analysis Models

No Analysis Models have been given.