Software Requirements Specification

for

Nudge - Task Manager

Version 1.0 approved

Prepared by: Team members:

Rithvik Jayaram - PES2UG20CS583

Sunkuru Sanjivani Patra - PES2UG20CS563

M Vamshidhar Reddy - PES2UG20CS585

Ayushi Kumari - PES2UG20CS587

Of PES University

Created on 11/09/2022

PES UNIVERSITY, BANGALORE
Department of Computer Science and Engineering

TABLE OF CONTENTS

Revision History

Name	Date	Reason For Changes	Version

Introduction

Purpose

The purpose of this document is to present a detailed description of the application. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, and the constraints under which it must operate. This document is intended for both the stakeholders and the developers of the system.

Intended Audience

This document is intended for the:

- Developers of the project
- The professors who would review the document
- Client, who wants to make their life more organized

Product Scope

Nudge is a simple, but useful task manager. It allows you to keep track of your projects and manage all your tasks. With Nudge, you'll never again worry about missing deadlines. You can add due dates, set up reminders, and build productive habits. Create new tasks, quickly pull up your daily to-dos, keep focused on your priorities and see yourself grow with a personalized dashboard

References

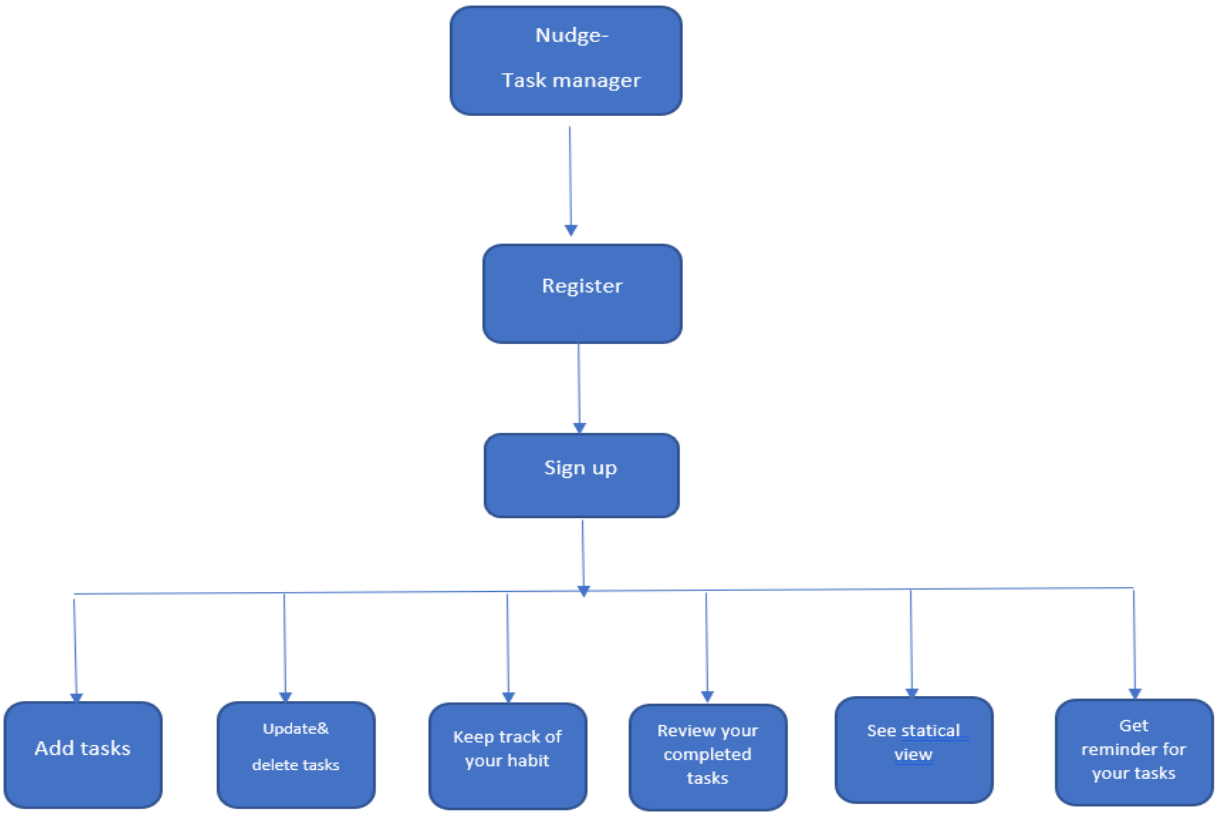
https://docs.leantime.io/#/

Overall Description

Product Perspective

Nudge is a new, task managing software aimed at providing the user a tool that lets you plan, organize and prioritize all your tasks so that you can finish them within the limited time frame and in the most efficient manner. The software aims to ensure that you rule out the possibility of missing out on your deadlines and get your work-life organized.

Product Functions



User Classes and Characteristics

There is only one user class in this software, that is, User.

USER:

- Register and create an account
- Add tasks as per user priority and can update the list whenever you want
- We can mark the task as completed when done with it
- Review the statistics of all complete tasks
- Keep track of your habit

Operating Environment

The software is designed to work on the latest versions of windows, Linux, and platforms.

The software is completely web-based so that it can work on all popular web browsers.

Design and Implementation Constraints

• The user should have sufficient knowledge of computers.

• The users must know the English language, as the user interface will be provided in the mentioned language.

• User needs a standard web browser to support MERN Stack and a good network connection.

Assumptions and Dependencies

The browsers which the user is using are either Google Chrome 10.0 and above or Mozilla Firefox 4.0 and above. Issues with the internet can affect the working of the software.

External Interface Requirements

User Interfaces

The user interface will be in the form of a website where users can interact with the various components present on the screen. The home page will display all the available services. There will be tabs that will direct users to pages where they can enter daily or monthly goals and track them. Users can also view weekly/monthly statistics of their goals achieved and yet to-do tasks. There is also a timeline feature available for users who are interested in tracking their progress over a custom time range.

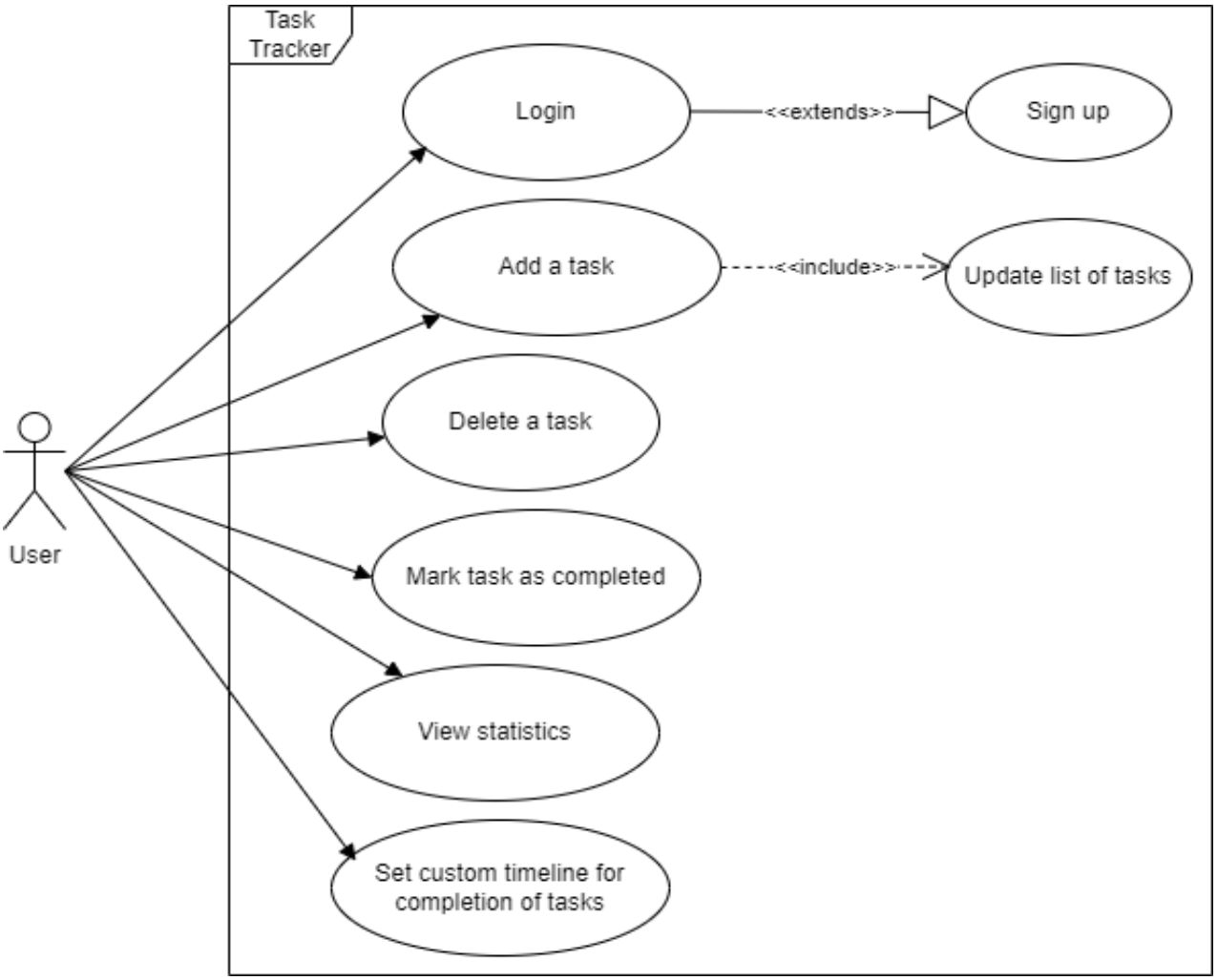
Software Interfaces

The services of the product are accessed via a web application that uses the following tools:
MongoDB - Database to store information about the tasks, goals, timelines, and statistics.
ExpressJS - Backend tool to route between pages/different services.
ReactJS - Develop the front-end part of the webpage with a dynamic touch.
NodeJS - Runtime environment to run all the Javascript for backend services

Communication Interfaces

The website will require a suitable web browser that meets the standards of the tech stack used and retrieving and sending data will be done through HTTP requests. There is no additional layer of security apart from the ones provided by the tools used.

Analysis model



System Features

This Web-based App is catering to specific users who want a seamless experience in becoming productive and managing tasks and habits. These are the features provided:

LOGIN / REGISTRATION PAGE

Description and Priority
The user will be able to log in and register to keep track of their daily to-dos and habits.
Priority- High Priority

Stimulus/Response Sequences
A database will be maintained for all the registered users to keep a track of all their tasks and habits. The users will be able to login into the app after registering and adding tasks and habits.

Functional Requirements
Requirement 1: The software must maintain a database consisting of all the mentioned details.
Requirement 2: Software must be user-friendly and efficient.

TRACK TASKS FOR EACH DAY

Description and Priority
The user will be able to add tasks with information like due date, priority, etc, and mark them as completed once they're done with them.
Priority- High Priority

Stimulus/Response Sequences
A database will be maintained for all the tasks mentioned with all the details of the tasks.

Functional Requirements
Requirement 1: The software must maintain a database consisting of the tasks.
Requirement 2: Software must be able to move the tasks to the correct table based on status like completed, to be done, etc.

Requirement 3: TBD

TRACK HABITS

Description and Priority
The user will be able to add habits which are like recurring tasks and mark them as done for the day and get a graphical progress report based on how consistent they are in following their routines.
Priority- High Priority

Stimulus/Response Sequences
A database will be maintained for all the registered users to keep a track of all their habits. All the previous data will be maintained to draw inferences from this data.

Functional Requirements
Requirement 1: The software must maintain a database consisting of all the habits.
Requirement 2: TBD

VIEW WEEKLY/MONTHLY STATISTICS

Description and Priority
The user will be able to view graphical progress charts for their habits.
Priority- Medium priority

Stimulus/Response Sequences
The database of the history of habits will be used for drawing inferences using different analysis techniques.

Functional Requirements
Requirement 1: The software must use the habits database and display graphs and charts showing progress.
Requirement 2: TBD

ADD A TIMELINE

Description and Priority
The user will be able to see their tasks for a particular day in a calendar.
Priority- Low Priority

Stimulus/Response Sequences
All the tasks for a particular day are extracted from the database and displayed in the calendar.

Functional Requirements
Requirement 1: The software must be able to extract the tasks from the database for any day and display it in the calendar.
Requirement 2: TBD

Other Nonfunctional Requirements

Performance Requirements

- Tasks added are reflected under 5 seconds.
- Statistics related to tasks are calculated accurately.

Safety and Security Requirements

- Only users who log in with suitable credentials can use the product. This is done to prevent anonymity and illegal use of the software.
- Any data stored in the database is encrypted to prevent data from being stolen and used for criminal purposes.

Software Quality Attributes

- The website can be used on any mainstream browser.
- The website is built using the MERN technology stack which is an open source that's constantly being improved upon by tech experts from around the world. Therefore, there is a lot of support and documentation available for a developer.
- Open-source technology also comes with no vendor lock-in, so if you decide to move on or change something later on down the line, there won't be any extra things to be taken care of.

Other Requirements

Technical Feasibility

For the implementation of the nudge, the technical resources needed were estimated.

- The current solution to the software was decided based on:
- The complexity of the technical resources needed.
- The manpower needed to implement the project.
- Team member's prior experience with the technology.
- The stack which provides a complete set of tools required to facilitate agile methodology.