

BETSOL-TODO

Design Document

Version 1.0

03/02/2023

**Authors (Alphabetical Order)**

| Name | Email |
| --- | --- |
| **Pratheek J** | **pratheek.rj@gmail.com** |
|  |  |
|  |  |
|  |  |

**Revision History**

| Version | Comment | Date | Author |
| --- | --- | --- | --- |
| 1.0 | TODO Design | 21/04/2023 | Pratheek J |

**Contents**

[Introduction 4](#_gjdgxs)

[DESCRIPTION 4](#_30j0zll)

[DETAILS 4](#_1fob9te)

[Tech Stack that will be used 4](#_3znysh7)

[Data Model 4](#_2et92p0)

[Sequence diagram: 5](#_tyjcwt)

[API DESCRIPTION: 5](#_3dy6vkm)

[Figma file: 6](#_1t3h5sf)

# Introduction

A paper to-do list is good way to track your task but switching to a digital one has advantages. If you like to write by hand, mark jobs off in ink or pencil, and draw arrows to indicate when priorities and deadlines change, paper is fine. However, the to-do app makes it easier for you to write down, change the status of your tasks. It will manage and track all the to-dos a person needs to complete. Additionally, you can edit that task name which is already created.

# DESCRIPTION

In this version of the Todo list, the user will be getting four options: Create (add) a new task or add a new ToDo in the ToDo List App. View every task or View every ToDo that has been added to the application. Remove any item from the list of ToDo’s.

Outcomes

* A single location where employees can track to-do items.
* Lucid display of tasks.
* Improve productivity.

# DETAILS

## Tech Stack that will be used

Python (Flask), PostgreSQL, Bootstrap, HTML, CSS, JavaScript

## Data Model

* + **Database: todo**
    - Tasks table
* **id**: this is a primary key, this Indicates id of a task
* **task**: name of the task
* **status:** Indicates the progress for the to-do item.

Graphical user interface, text, application, chat or text message

Description automatically generated

## Sequence diagram:

Diagram

Description automatically generated

## API DESCRIPTION:

| API | Get |
| --- | --- |
| Request Object | {  “description”: string  } |
| Response Object | {  “response”:” done ”  “success”: true  } |
| API Description | This API is used for creating a task in the todo app. |
| Business logic | If a user wants to add a task to the todo application, they can use create api and give in the task name. |

| API | Delete |
| --- | --- |
| Request Object | NA |
| Response Object | {  “response”: “Removed Task”  “success”: true  } |
| API Description | This API is used for deleting a task in the todo application. |
| Business logic | If a user wants to delete a task in the todo application, they can use delete api and delete the task. |

//To be done

| API | Edit |
| --- | --- |
| Request Object | {  “description”: string  } |
| Response Object | {  'success': True,  'response': 'Done'  } |
| API Description | This API is used for editing a task in the todo application. |
| Business logic | If a user wants to edit a task in the todo application, they can use the edit API to edit and update the task description. |

## Figma file:

[Link to Figma File](https://www.figma.com/file/Q7AEF3op1SIAubAFP0Add9/ToDO?node-id=0%3A1&t=FXG3ejF8nC9c8hAG-0)