|  |  |  |
| --- | --- | --- |
| **Tech Saksham**  Final Project Report  **Track Name** |  |  |

**“TODO LIST”**

**“Avanthi’s Scientific Technology And Research Academy”**

|  |  |
| --- | --- |
| **ROLL NO** | **NAME** |
| 19PT1A0535 | A. Lavanya |
| 19PT1A0519 | K. Deepika |
| 19PT1A0531 | P. Jahnavi Priyashma |
| 19PT1A0516 | K. Akhila |

|  |  |
| --- | --- |
| Umamaheshwari | Trainer Name |

**ABSTRACT**

* A to-do list is a list of tasks that need to be completed, typically organized in order of priority.
* It is one of the simplest solutions for task management and provides a minimal and elegant way for managing tasks a person wishes to accomplish.
* Our aim is to design a simple and elegant website for people to keep a track of the status of their tasks.
* Making a to-do list is an easy and important task that everyone should do.
* The immense satisfaction that one gets when completing the task and marking it on the list are incomparable.
* Moreover, creating a list of tasks ensure you don’t miss out on anything.
* It’s a scientific fact that when you write the tasks that you need to complete, you are even more motivated to complete it.
* With this in mind, we come to build a platform which will help people create their own task list.
* With the help of modern tools and technologies, we strive to build a minimal and efficient to-do list which minimizes distractions and helps people achieve task management with ease and without hassle.

**INDEX**

|  |  |  |
| --- | --- | --- |
| **Sr. No.** | **Table of Contents** | **Page No.** |
| 1 | Chapter 1: Introduction | 1-3 |
| 2 | Chapter 2: Services and Tools Required | 4-6 |
| 4 | Chapter 3: Architecture Detail Working | 7-10 |
| 5 | Conclusion | 11 |
| 6 | References | 12 |
| 7 | Code | 13 |

**CHAPTER 1**

**INTRODUCTION**

**1.1Overview:** A to-do list is a simple prioritized list of the tasks a person must complete. People make a list of everything they need to do, ranked according to priority from the most critical task at the top to the least critical task at the bottom A few of the features of a good to-do list application include:

Plan and execute simple actions.

Prioritize, manage, and reason about tasks.

Record notes, action items and ideas.

To-dos are the tasks or the atomic entities that make up a to-do list. To-dos are made quickly, the bulk of them do not specify the work; instead, they are typically just comprehensive enough to serve as a valuable indicator.

There are clear immediate implications to adding a to-do list to a person’s productivity system. The functionalities provided by a good to-do list application/system help declutter the user’s mind as their pending tasks are recorded safely and they won’t be forgotten.

The To-do list project is a user-friendly website which helps them to keep a track of their tasks. It is a simple site which requires no sign-in/log-in or any personal details but still records your task, mark the completed tasks, and stores them even if you visit the site after a few days.

**1.2 Feature:**

* Reorder Tasks
* Prioritisation
* Quick Entry/Capture
* Simple UI
* Picture on Tasks
* Today List
* Sync
* Future Work
* Fun to Use
  1. **Advantages:**
* It can help the project team to see all the things that they need to do so that they can be aware of the phase that they need to be in if they want to finish the project in time without compromising quality.
* One of the key challenges is to choose appropriate storage solution, that will allow to maintain its biggest advantages:

Simplicity

Speed

low recourses demand

* A project to-do list can serve as a reference of the things that have already been done and the procedures that are needed to be implemented the soonest possible time.
  1. **Scope:**
* To-do list has it’s following functions: organization of a proper work with an accurate planning of every task with it’s short description.
* Scope is a simple to-do list that will helps u in your daily routine. you can easily create lists and tasks, manage them and receive reminders on time.
  1. **Future Work:**
* The future work for a to-do list involves improving its functionalities, user experience, and integration with other productivity tools, as well as expanding its accessibility across different platforms and devices.
* Some potential areas for improvement include adding more customizable features, incorporating artificial intelligence and machine learning algorithms, and enhancing collaboration and sharing capabilities.
* Additionally, there may be opportunities to integrate to-do lists with other productivity tools, such as calendars, email, and project management software, to provide a more comprehensive solution for managing tasks and workflows.
* Finally, as technology continues to evolve, there may be new and innovative ways to use to-do lists to enhance productivity and efficiency in both personal and professional settings.

**CHAPTER 2**

**SERVICES AND TOOLS REQUIRED**

**2.1 Services Used:**

There are many to-do list services available, both free and paid, that can be used to manage tasks and prioritize daily activities. Some of the most popular to-do list services include:

Todolist: a feature-rich to-do list app that offers advanced task management, collaboration, and productivity features.

Trello: a highly visual to-do list app that uses boards, lists, and cards to organize tasks and workflows.

Google Tasks: a simple to-do list app that integrates seamlessly with other Google apps, such as Gmail and Google Calendar.

Microsoft To Do: a free to-do list app that offers integration with Microsoft 365 and Outlook.

Any.do: a to-do list app that features a minimalist design and intuitive interface, with advanced features like voice recognition and calendar integration.

Wunderlist: a popular to-do list app that was acquired by Microsoft and replaced by Microsoft To Do, but still available for use.

Remember The Milk: a comprehensive to-do list app that offers features like smart lists, location-based reminders, and integration with other apps and services.

Overall, there are many different to-do list services available, each with their own strengths and weaknesses, so it is important to choose one that best fits your needs and preferences

**2.2 Tools and Softwares used:**

Work requests often get sent in many different forms, such as email, verbally, or even a sticky note left on your desk!. With no centralization and task management software, it's no wonder work instructions get missed.

Ensure compliance steps are adhered to.

Ensure deadlines are clear with automated reminders and early warning indicator.

Give complete management oversight across departments and vendors.

**2.2.1 NodeJS:**

A To-do list is a beginner application that many programming students make. It is a list of tasks that you need to do in a day. You can add tasks to it and even delete them when done with them. Often it can be seen that many tutorials leave the to-do list application on the frontend part only. The reader never gets to know how to integrate it with the backend, such that a proper database is used and data is not lost on refresh/restart. This article seeks to do teach you how to make a complete To-do List project.

This article is targeted at the audience that has gained basic knowledge of web development but has not built any projects. Many students know the syntax but are unable to make applications. Just knowing certain technology such as Nodejs is not enough, being able to use it with other technologies and build an application using programming logic is also required. This article is made to help such people sharpen their knowledge by building projects.

**2.2.2 HTML:**

Here is a basic example of a to-do list using HTML:

<!DOCTYPE html>

<html>

<head>

<title>To-Do List</title>

</head>

<body>

<h1>To-Do List</h1>

<ul>

<li>Buy groceries</li>

<li>Pay bills</li>

<li>Finish work project</li>

</ul>

</body>

</html>

This code creates a simple to-do list with three tasks: buying groceries, paying bills, and finishing a work project. The tasks are listed in an unordered list (<ul>) with each task represented as a list item (<li>). Additional tasks can be added to the list by adding new list items.

This is just a basic example and a more complex to-do list system would require additional HTML, CSS, and JavaScript to create a more robust and user-friendly interface.

**CHAPTER 3**

**ARCHITECTURE DETAIL WORKING**

To-Do List project is an application specially built to keep track of errands or tasks that need to be done. This application will be like a task keeper where the user would be able to enter the tasks that they need to do. Once they are done with their tasks they can also remove them from the list. Let us see what is there in the application:

You can add the tasks that are to be done in a descriptive way.

You will be able to add as many tasks as you have.

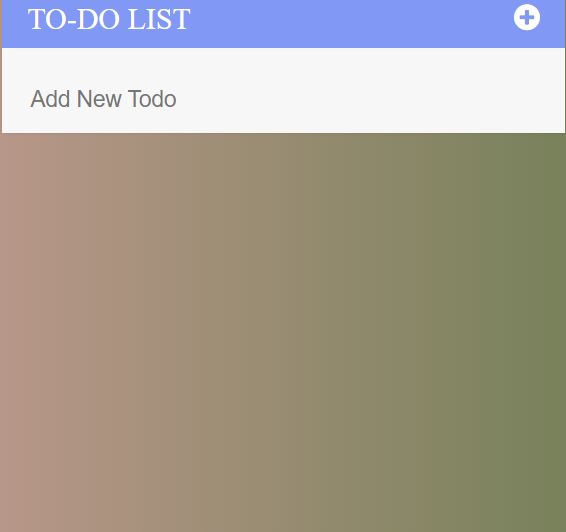
Once the task is completed, you will be seen cross mark on it by clicking on task.

Once you are ready with importing your code you can run it on the web page.

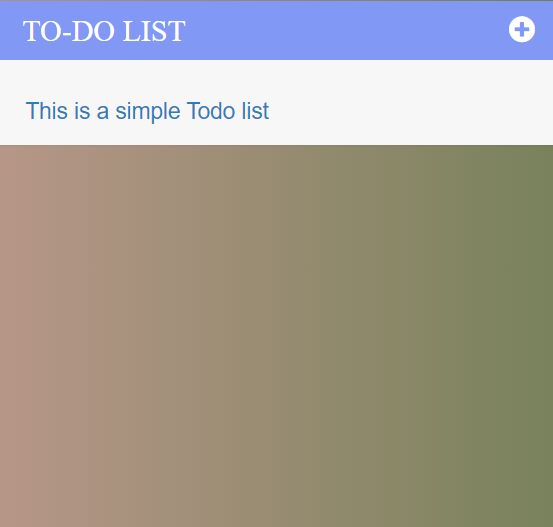
1. After running the application, following:



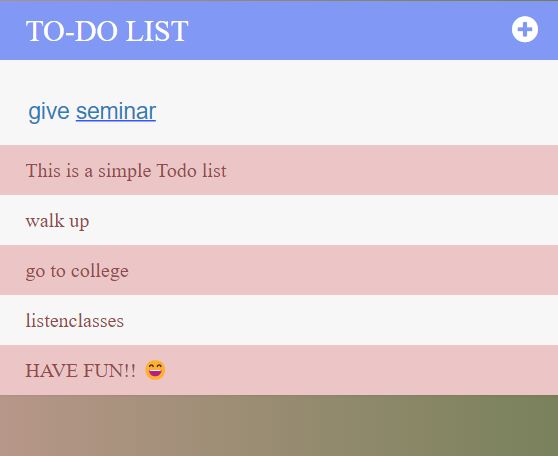
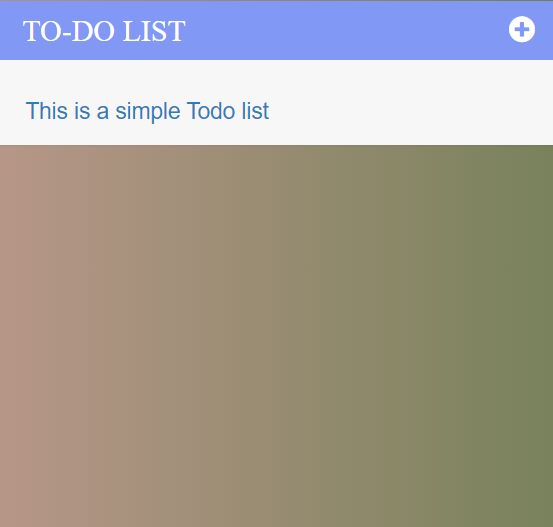
2.Now begin working on the Web page,Click on the”+”button on the top right:



3. Now, here you can type the task that you want to add:



4. You can write the task as follows and add then click on add.Similarly add few more:

  
5.Now ,to check the completed task from the task list,simply click on the task.





6.Now, to remove the task from the task list, simply click on the delete button.



7.To delete more, Similarly click on delete button.



**CONCLUSION**

* To-do-list-web page performs very efficient comparing to competitor.
* There is a space for further developments with regards of keeping app small and quick.
* It would be optimal to use dedicated CSS and continuing using vanilla JavaScript.
* To-do-web page can be developed as a sole application as well as a very efficient module to be combined in a larger project.

**REFERENCES**

1.Staff, C. A. C. M. (2016). React: Facebook's functional turn on writing Javascript.Communications of the ACM, 59(12), 56-62.

2. Kvalheim, C. (2015). The Little Mongo DB Schema Design Book. The Blue Print Series.

3. Khedkar, S., Thube, S., Estate, W. I., & Naka, C. (2017). Real time databases for applications. International Research Journal of Engineering and Technology (IRJET), 4(06), 2078-2082.

4. Tilkov, S., & Vinoski, S. (2010). Node. js: Using JavaScript to build high-performance network programs. IEEE Internet Computing, 14(6), 80-83.

5. Syed, B. A., & Bean, M. (2014). Beginning Node. js (pp. 181-182). New York City: Apress.

6. Morales-Morell, A. (2001). Usability aspects of a location-aware TODO list application. University of Puerto Rico, Mayaguez (Puerto Rico).

7. Gaspar, F., Taniça, L., Tomás, P., Ilic, A., & Sousa, L. (2015). A framework for application-guided task management on heterogeneous embedded systems. ACM Transactions on Architecture and Code Optimization (TACO), 12(4), 1-25.

**CODE**

**Please Provide Code through Git Hub Repo Link**