CS207 PROJECT REPORT

To-Do Application

Group 17

Khushi Ladha B20013
Arjun Mehra B20149
Ayush Aggarwal B20151
Sourav Samant B20170
Kishan Sharma B20294

Index

Acknowledgement

A lot of effort has been put into this project. Completing this project would have not been possible without the guidance and support from some.
We would like to express our gratitude to Dr.Varun Dutt,CS207 ADP Instructor, for his valuable lessons and giving us an opportunity to learn more about the ADP course. We would also like to give our thanks to head TA Aadhar Gupta sir,our mentor Sagar Paliwal and other TAs for their valuable guidance.
Without the cooperation of each group member,completion of this project would have not been possible ahead of schedule.
Our thanks and gratitude goes to everyone for their support with this project.

Abstract

TODO - It is a multifunctional todo list web app which can be used to add, update and delete tasks with time. It can store different lists of tasks for different users. Moreover the program also sends mail to the registered email address with the remaining tasks for the day automatically at a specific time thereby reminding you of important tasks.

Languages such as CSS and Javascript have been used to design the frontend of the

system whereas PHP, Javascript and MySQL have been used to design and connect the system to the backend of the system. Moreover bootstrap library has been used to design the frontend elements.

The system is fully working and stores the input provided by the user to the database (The specific tables created to store this data). It also displays the required data on clicking the corresponding buttons to execute required queries.

Introduction

In this project,the aim is to build a web application to store a list of tasks that would be entered,updated and edited by the user. This project is presented with the aim of solving the issue of high workload of individual tasks and increasing the work organization. Hence,in this report we provide information on the website and its infrastructure and how each module plays an important role in this successful project.

Methodology

When you open the webpage, you're greeted with the login page which can be used to login for already registered users or you can also go to the registration page, where new users can register for this application. After successfully logging in, users can add tasks which can be later modified. After completion of a task, the user can delete the task.

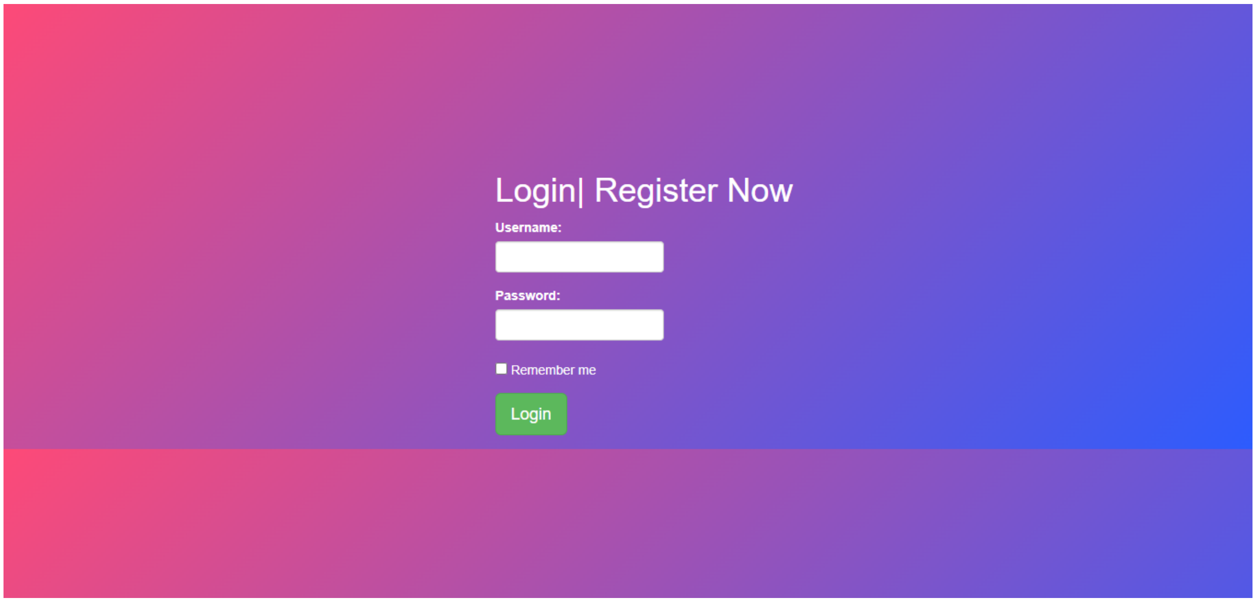
The landing page of the website is the index.php which checks if the user is logged in. If the user is logged in then index.php page loads otherwise login.php loads. For the first visit we can say that the user is not logged in so the login page will open corresponding to login.php where there are 2 input fields which are username and passwords and 2 redirect links one which leads to registration page and other is the login button. The login button checks username and password against the one stored in the database and if it is found then the session starts and the user logs in. If the user does not have an username and password he has to click on Register Now which will redirect to the Registration page corresponding to register.php. It has 4 input fields which are username,email address, password,confirm password. It has a google reCAPTCHA for verification and a register button. On successful registration You will be redirected to the login page where you can login.

On successful login, The tasks page will open corresponding to index.php. Here we have a logout button which will end the session and redirect to login.php. To add tasks we click on + icon and a form appears where we have to fill in the details and click on add task which will create an entry in the database with a unique event id corresponding to user id. For adding tasks, the transport.js is taking the inputs from index.php and passing it to insert.php. We can now delete the task by clicking on the cross button or update it by clicking on the update button. Transport.js is responsible for making the forms visible or hidden along with connecting the php files together.

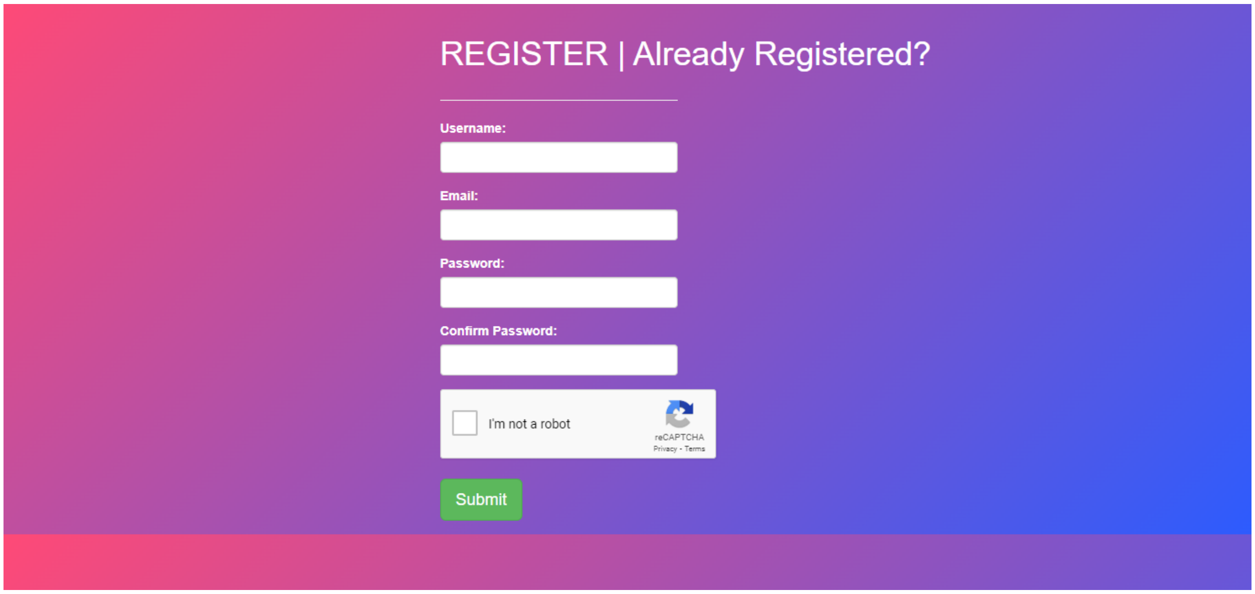
Transport.js connects index.php to update.php for updation and to delete.php for deletion.

The CSS of login and registration are in login.php and register.php while the CSS of index.php is style.css. Moreover bootstrap is included and used as necessary.

Output



Login Page for existing users



REGISTER | Already Registered?

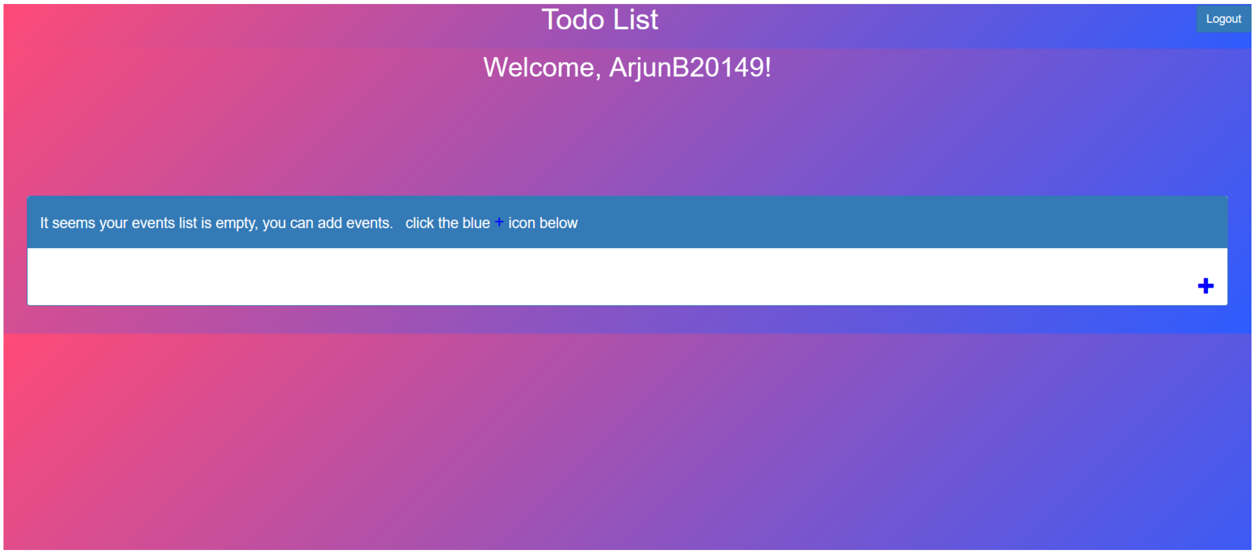
Username:

Email:

Password:

Confirm Password:

Registration page for new users,with Human verification system

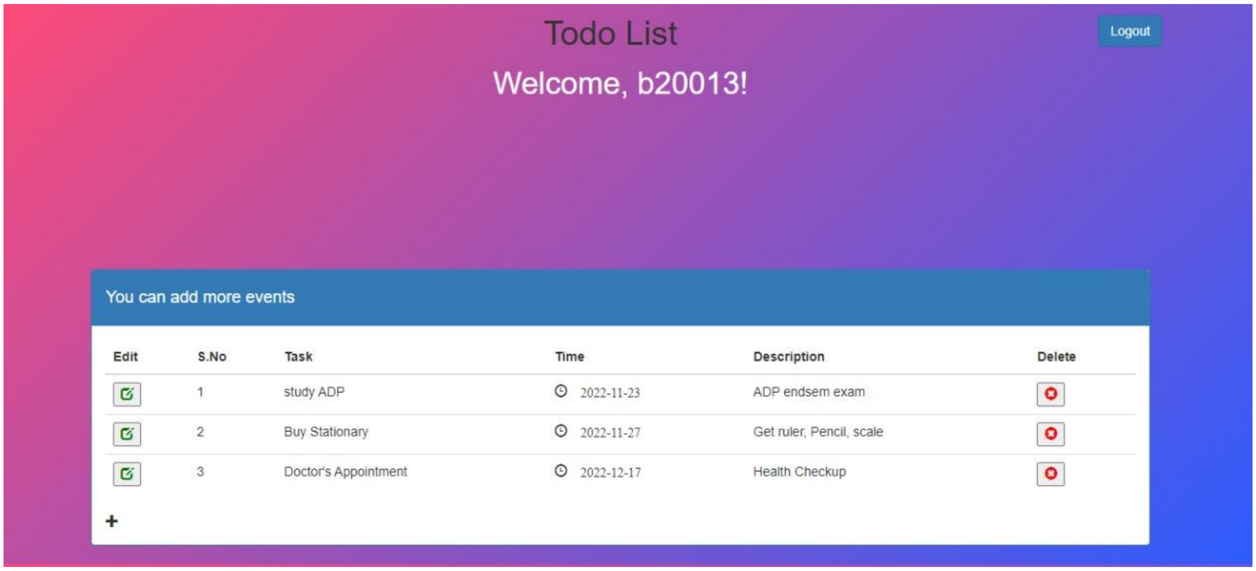


Todo List

Welcome, ArjunB20149!

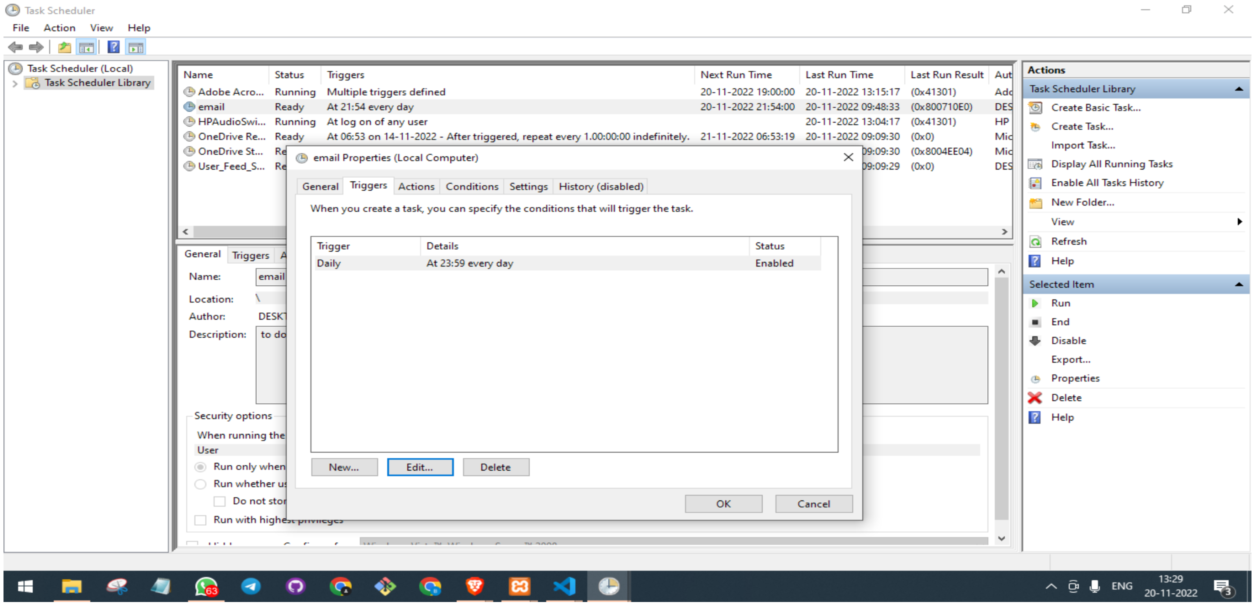
It seems your events list is empty, you can add events.   click the blue + icon below

Tasks page in which Users can create new tasks in chronological order



Edit	S.No	Task	Time	Description	Delete
	1	study ADP	2022-11-23	ADP endsem exam	
	2	Buy Stationary	2022-11-27	Get ruler, Pencil, scale	
	3	Doctor's Appointment	2022-12-17	Health Checkup	

Tasks arranged in chronological order



Name	Status	Triggers	Next Run Time	Last Run Time	Last Run Result	Aut
Adobe Acro...	Running	Multiple triggers defined	20-11-2022 19:00:00	20-11-2022 13:15:17	(0x41301)	Add
email	Ready	At 21:54 every day	20-11-2022 21:54:00	20-11-2022 09:48:33	(0x800710E0)	DES
HPAudioSwi...	Running	At log on of any user		20-11-2022 13:04:17	(0x41301)	HP
OneDrive Re...	Ready	At 06:53 on 14-11-2022 - After triggered, repeat every 1.00:00:00 indefinitely.	21-11-2022 06:53:19	20-11-2022 09:09:30	(0x0)	Mic
OneDrive St...	Re			09:09:30	(0x8004EE04)	Mic
User_Feed_S...	Re			09:09:29	(0x0)	DES

General Triggers A

Name: email

Location: \

Author: DESK

Description: to do

Security options

When running the

User

○ Run only when

○ Run whether u

□ Do not stor

□ Run with highest privileges

email Properties (Local Computer)

General | Triggers | Actions | Conditions | Settings | History (disabled)

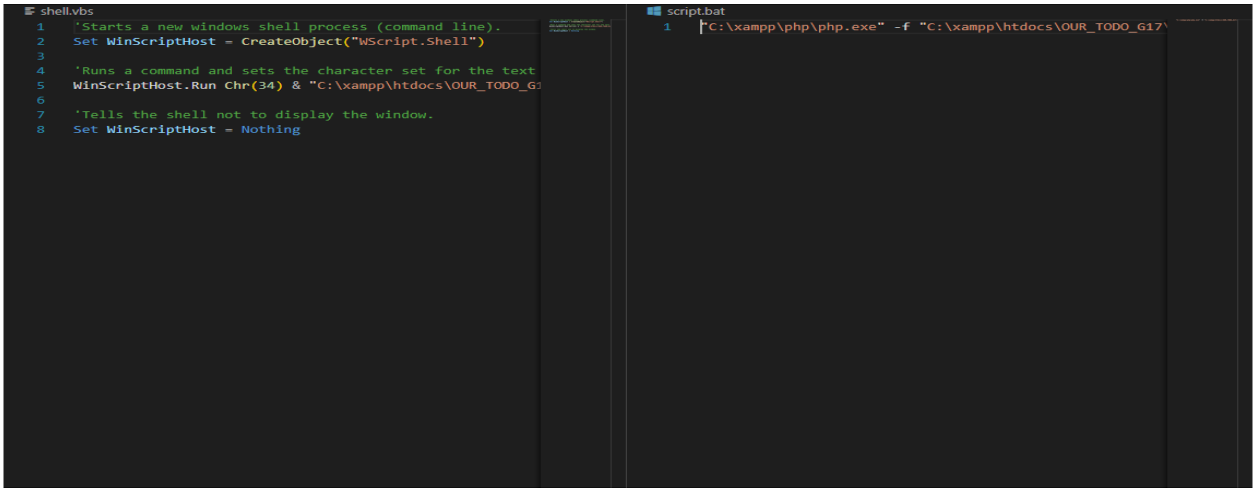
When you create a task, you can specify the conditions that will trigger the task.

Trigger	Details	Status
Daily	At 23:59 every day	Enabled

New... Edit... Delete

OK Cancel

Task Scheduler



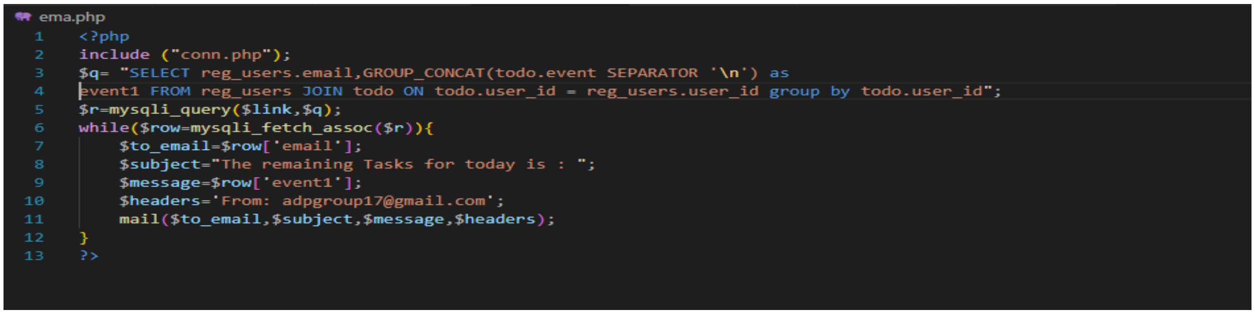
```
'Starts a new windows shell process (command line).
Set WinScriptHost = CreateObject("WScript.Shell")

'Runs a command and sets the character set for the text
WinScriptHost.Run Chr(34) & "C:\xampp\htdocs\OUR_TODO_G1

'Tells the shell not to display the window.
Set WinScriptHost = Nothing
```

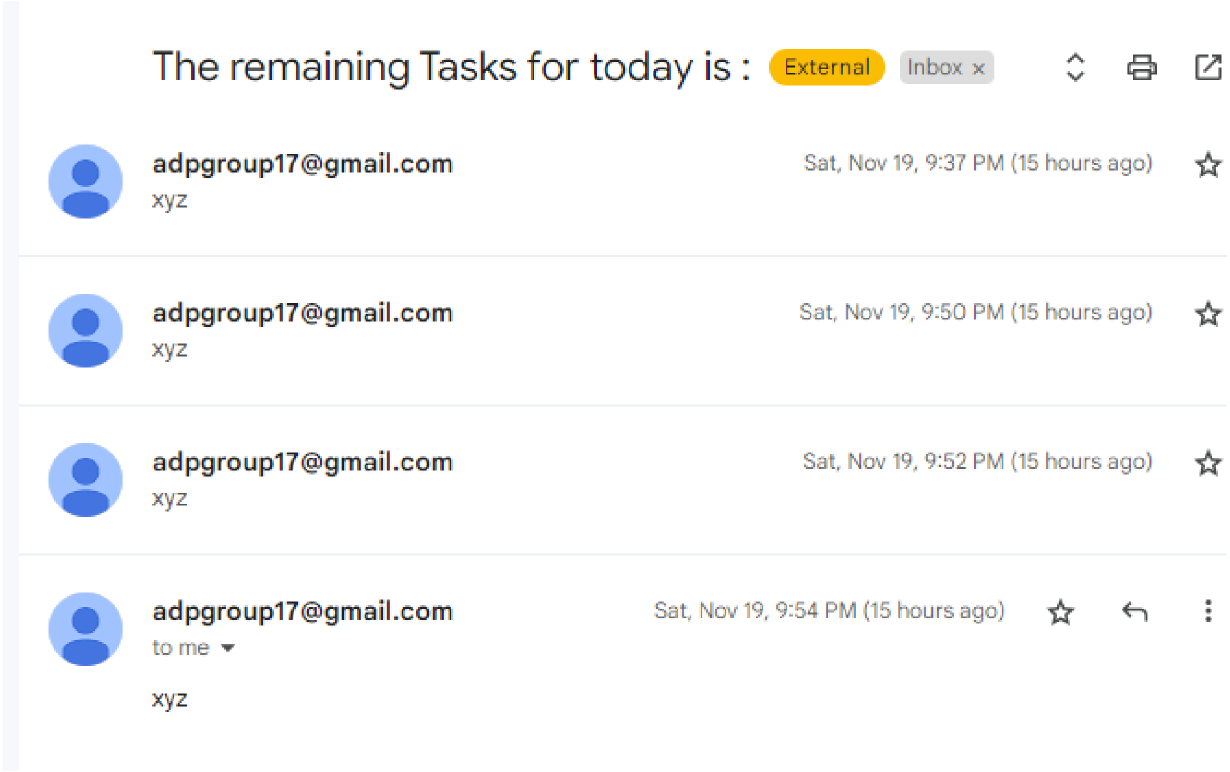
```
"C:\xampp\php\php.exe" -f "C:\xampp\htdocs\OUR_TODO_G17\
```

Shell.vbs and script.bat file



```
<?php
include ("conn.php");
$q= "SELECT reg_users.email,GROUP_CONCAT(todo.event SEPARATOR '\n') as
event1 FROM reg_users JOIN todo ON todo.user_id = reg_users.user_id group by todo.user_id";
$r=mysqli_query($link,$q);
while($row=mysqli_fetch_assoc($r)){
    $to_email=$row['email'];
    $subject="The remaining Tasks for today is : ";
    $message=$row['event1'];
    $headers='From: adpgroup17@gmail.com';
    mail($to_email,$subject,$message,$headers);
}
?>
```

Ema.php file(to send email)



The remaining Tasks for today is :

adpgroup17@gmail.com Sat, Nov 19, 9:37 PM (15 hours ago)
xyz

adpgroup17@gmail.com Sat, Nov 19, 9:50 PM (15 hours ago)
xyz

adpgroup17@gmail.com Sat, Nov 19, 9:52 PM (15 hours ago)
xyz

adpgroup17@gmail.com Sat, Nov 19, 9:54 PM (15 hours ago)
to me

xyz

Screenshot of mail received

CONCLUSION

With this web application built and ready for online deployment,it has the potential to include more features to be implemented in the future,which would increase the efficiency. By using this web app,one can reduce his/her own stress by arranging multiple tasks as per his/her own schedule and reduce the burden in order to stay organized and fulfill tasks faster.