Lab Report: Productivity Nexus

1. Introduction:

- **Project Title:** Productivity Nexus
- **Brief Overview**: This report details the development of a web-based Productivity Nexus, a comprehensive tool designed to help users track their progress and enhance their productivity.

2. Motivation

- **Personal Need:** As a student, I frequently struggled to manage my time effectively and stay organized amidst a busy schedule. I needed a tool that could track my daily progress, help me stay on top of deadlines, and provide insights into my overall productivity.
- **Targeted Audience:** The Productivity Nexus is primarily aimed at students who are looking for a comprehensive tool to help them improve their time management and achieve their academic goals.
- **Existing Tools:** Existing productivity tools like Google Calendar, Trello, and Todoist provided inspiration for this project. While these tools are useful, they often lacked the specific features I desired, such as a detailed productivity score, mindfulness exercises, and integrated resources like weather information.

3. Objectives:

The Productivity Nexus aims to:

1. Provide a comprehensive, user-friendly platform for managing productivity.
2. Offer a range of tools to support students in their time management and organization.
3. Enable secure user authentication and account management.
4. Implement an admin panel to manage user accounts and potentially other system features.

4. Methodology:

4.1 Development Environment:

The Productivity Nexus dashboard was developed using a combination of front-end and back-end technologies:

Front-End:

- **HTML:** Used for the structure and layout of the dashboard's web pages.
- **CSS:** Utilized Tailwind CSS framework to style the dashboard's visual elements and provide a consistent look and feel.
- **JavaScript:** Implemented to handle user interactions, dynamic content updates, and data manipulation (including tasks, reminders, events, and user authentication).

Back-End:

- **PHP:** Chosen as the server-side language to process user data, handle database interactions, and manage user authentication.
- **MySQL:** Employed as a relational database to store user information, task lists, reminders, events, notes, and book tracking data.

4.2 Design Process:
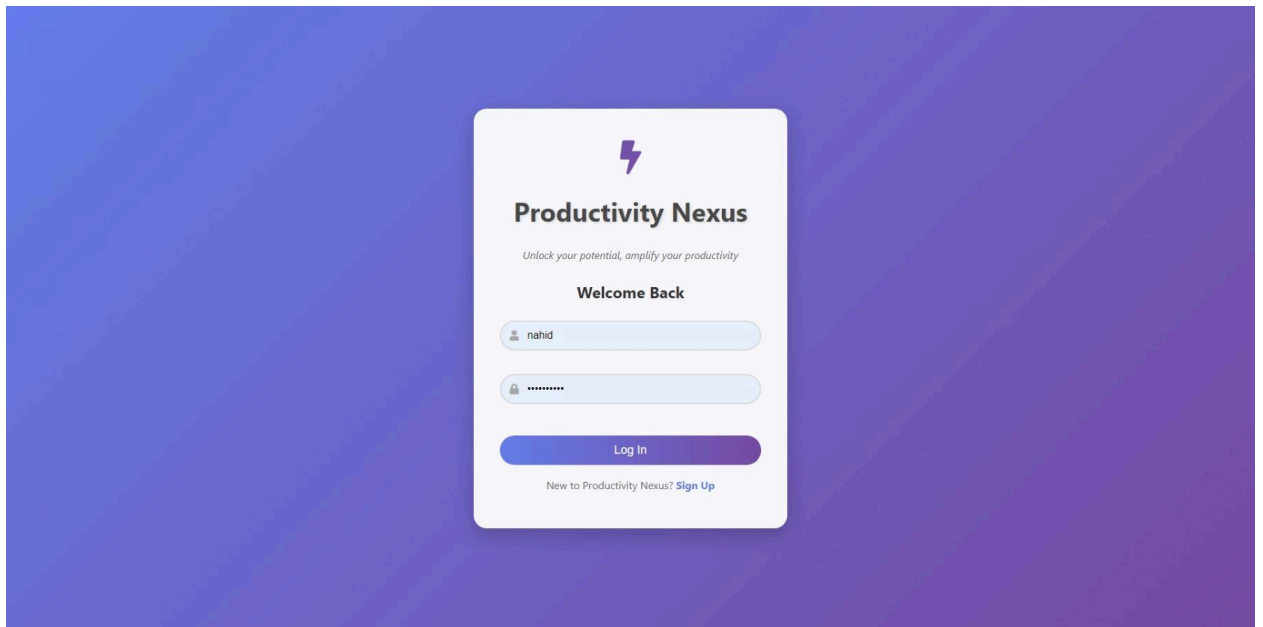
4.2.1 Iterative Design:

The design process for the Productivity Nexus dashboard was iterative, focusing on continuous improvement:

1. **Initial Prototypes:** Low-fidelity wireframes were created using [Figma] to visualize the basic layout and structure of the dashboard.
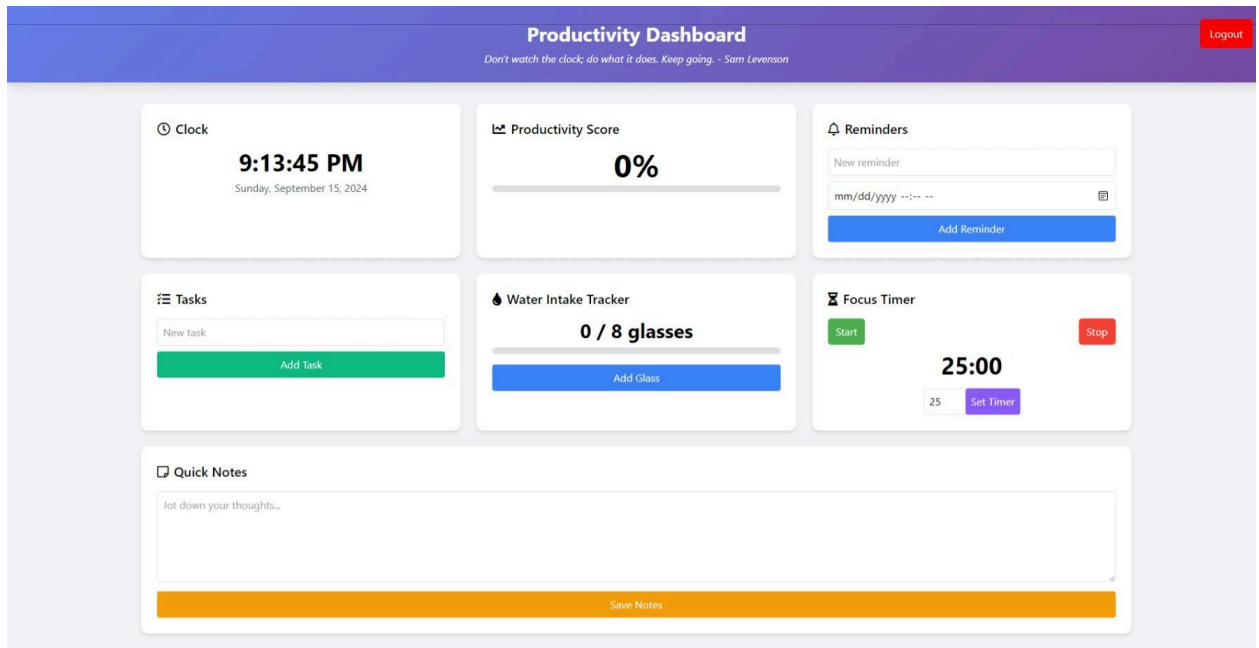
   Initial:

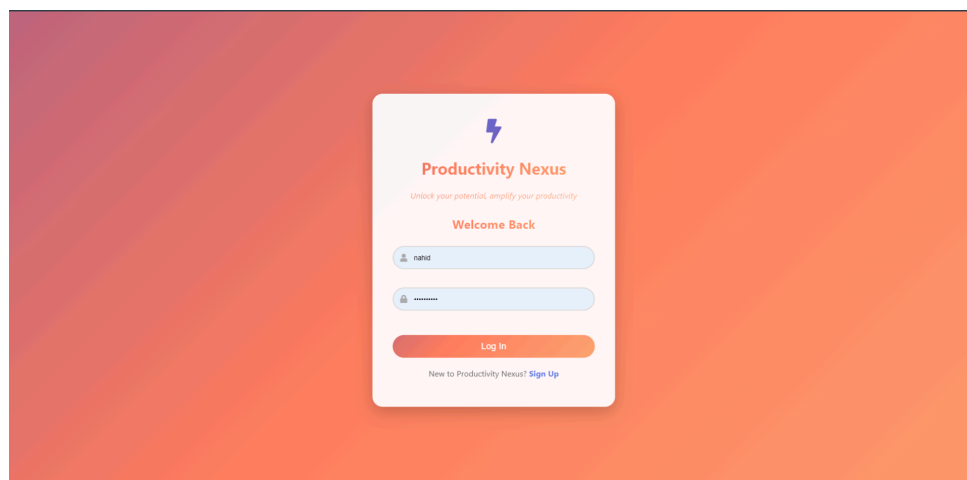   **(High Quality Image is uploaded on Github)**
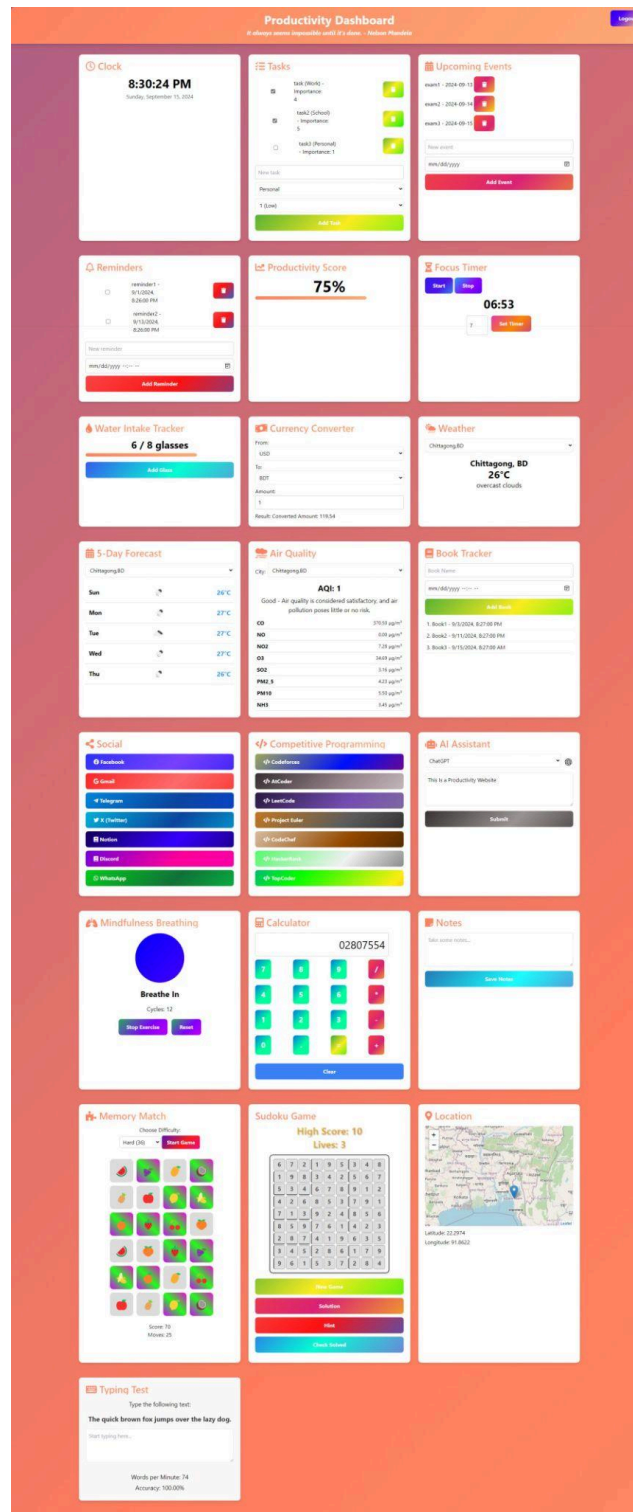
   Signup Page:

   Dashboard Page:

2. Feedback Incorporation: These wireframes were shared with potential users (friends, classmates) to gather feedback on the usability and organization of the dashboard.
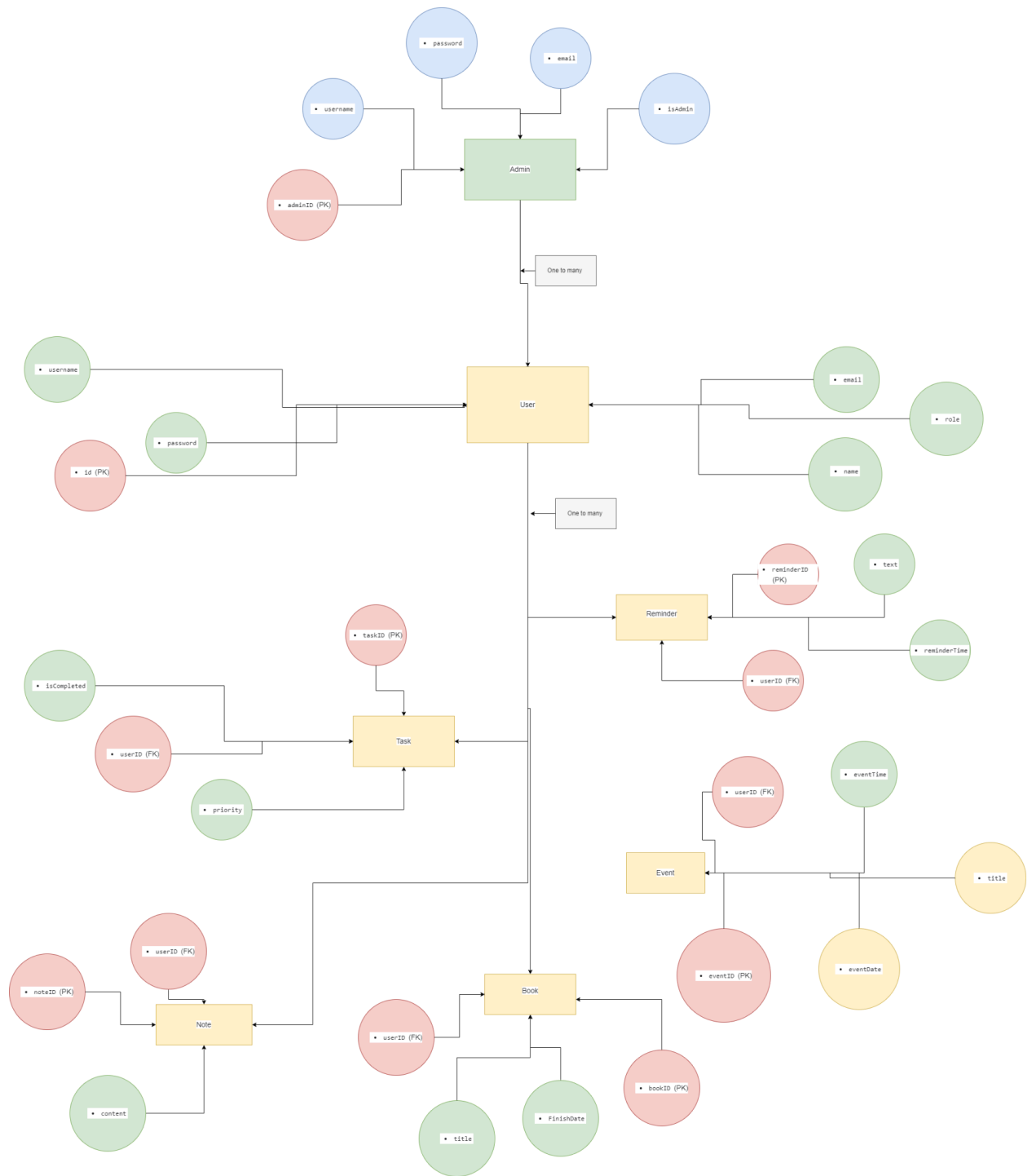
After Feedback:

Signup Page:

Dashboard Page:

Productivity Dashboard
It always seems impossible until it's done. - Nelson Mandela

Logout

Clock
8:30:24 PM
Sunday, September 15, 2024

Tasks
- task (Work) - Importance: 4
- task2 (School) - Importance: 5
- task3 (Personal) - Importance: 1

New task
Personal
1 (Low)
Add Task

Upcoming Events
exam1 - 2024-09-13
exam2 - 2024-09-14
exam3 - 2024-09-15
New event
mm/dd/yyyy
Add Event

Reminders
reminder1 - 9/1/2024, 8:26:00 PM
reminder2 - 9/15/2024, 8:26:00 PM
New reminder
mm/dd/yyyy
Add Reminder

Productivity Score
75%

Focus Timer
Start Stop
06:53
Set Timer

Water Intake Tracker
6 / 8 glasses
Add Glass

Currency Converter
From:
USD
To:
BDT
Amount:
1
Result: Converted Amount: 119.54

Weather
Chittagong,BD
Chittagong, BD
26°C
overcast clouds

5-Day Forecast
Chittagong,BD
Sun 26°C
Mon 27°C
Tue 27°C
Wed 27°C
Thu 26°C

Air Quality
City: Chittagong,BD
AQI: 1
Good - Air quality is considered satisfactory, and air pollution poses little or no risk.
CO   379.59 µg/m³
NO   0.04 µg/m³
NO2   7.28 µg/m³
O3   24.69 µg/m³
SO2   5.96 µg/m³
PM2_5   4.23 µg/m³
PM10   5.93 µg/m³
NH3   3.45 µg/m³

Book Tracker
Book Name
mm/dd/yyyy
Add Book
1. Book1 - 9/3/2024, 8:27:00 PM
2. Book2 - 9/11/2024, 8:27:00 PM
3. Book3 - 9/15/2024, 8:27:00 AM

Social
Facebook
Gmail
Telegram
X (Twitter)
Notion
Discord
WhatsApp

Competitive Programming
Codeforces
AtCoder
LeetCode
Project Euler
CodeChef
HackerRank
TopCoder

AI Assistant
ChatGPT
This is a Productivity Website
Submit

Mindfulness Breathing
Breathe In
Cycles: 12
Stop Exercise Reset

Calculator
02807554
7 8 9 /
4 5 6 *
1 2 3 -
0 . +
Clear

Notes
Take some notes...
Save Notes

Memory Match
Choose Difficulty:
Hard (36) Start Game
Score: 70
Moves: 25

Sudoku Game
High Score: 10
Lives: 3
New Game
Solution
Hint
Check Solved

Location
Latitude: 22.2974
Longitude: 91.8612

Typing Test
Type the following text:
The quick brown fox jumps over the lazy dog.
Start typing here...
Words per Minute: 74
Accuracy: 100.00%

3. Refinement and Development: Based on the feedback, the design was iteratively refined and implemented in HTML and CSS. The initial wireframes were then transitioned into high-fidelity mockups.
4. Functionality Development: Alongside design, the JavaScript functionality for user interactions and data manipulation was progressively implemented.
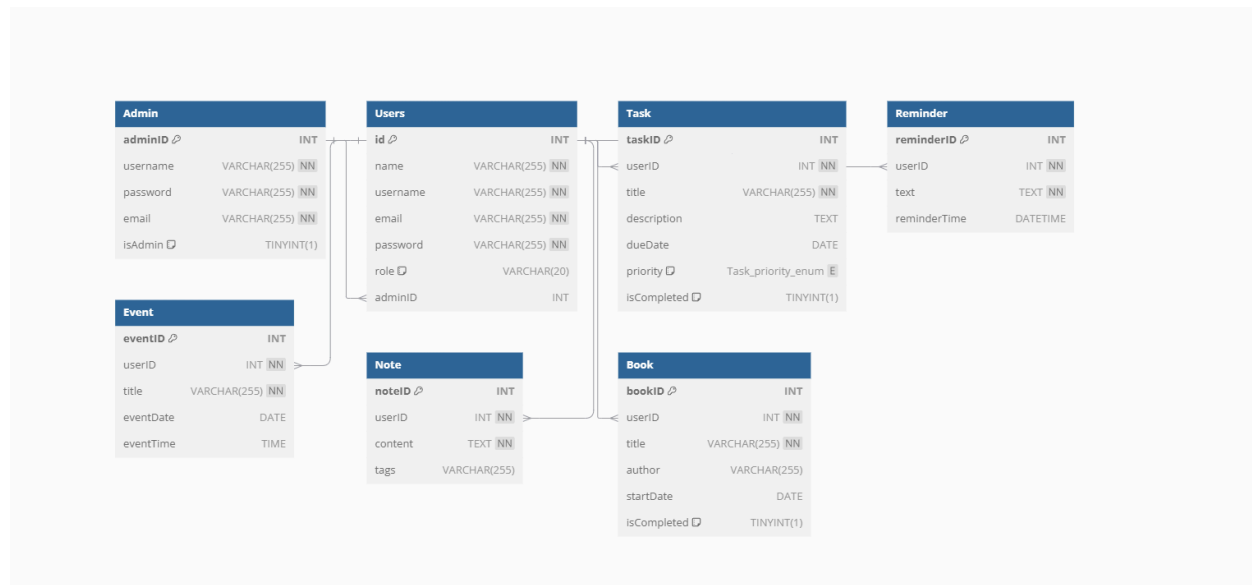
4.3 Database Design:

(High Quality Image is uploaded on Github)

4.3.1 Entity-Relationship Diagram (ER Diagram):

Admin entity attributes:
- password
- email
- username
- isAdmin
- adminID (PK)

One to many

User entity attributes:
- username
- password
- id (PK)
- email
- role
- name

One to many

Reminder entity attributes:
- reminderID (PK)
- text
- reminderTime
- userID (FK)

Task entity attributes:
- taskID (PK)
- isCompleted
- userID (FK)
- priority

Event entity attributes:
- userID (FK)
- eventTime
- title
- eventID (PK)
- eventDate

Note entity attributes:
- userID (FK)
- noteID (PK)
- content

Book entity attributes:
- userID (FK)
- title
- FinishDate
- bookID (PK)

4.3.2 Class Diagram :

Explanation of ER Diagram:

The ER diagram provides a visual representation of the database schema for the Productivity Nexus application. It illustrates the primary entities involved and their relationships, showcasing how data is structured and connected:

- **User:** This entity represents individual users of the application. It stores essential user information such as:
    - id (INT, PRIMARY KEY, AUTO_INCREMENT): Unique identifier for each user.
    - name (VARCHAR(255)): User's full name (optional).
    - username (VARCHAR(255), UNIQUE): User's unique username used for login.
    - email (VARCHAR(255), UNIQUE): User's email address for registration and contact.
    - password (VARCHAR(255)): User's encrypted password.
    - role (VARCHAR(20), DEFAULT 'user'): User's role, either 'user' or 'admin'.

- **Admin:** This entity represents administrators who have elevated privileges within the application. It stores information specific to admins, including:
    - adminID (INT, PRIMARY KEY, AUTO_INCREMENT): Unique identifier for each administrator.
    - username (VARCHAR(255), UNIQUE): Admin's username used for login.
    - password (VARCHAR(255)): Admin's encrypted password.
    - email (VARCHAR(255), UNIQUE): Admin's email address.
    - isAdmin (TINYINT(1), DEFAULT 1): Indicates that this is an administrator account (boolean value).

- **Task:** This entity represents tasks assigned or created by users. Each task contains the following attributes:
    - taskID (INT, PRIMARY KEY, AUTO_INCREMENT): Unique identifier for each task.
    - userID (INT, FOREIGN KEY, REFERENCES Users(id)): The user associated with the task.
    - title (VARCHAR(255)): Title of the task.
    - description (TEXT): Description of the task.
    - dueDate (DATE): Due date for the task.
    - priority (ENUM('low', 'medium', 'high')): Priority level for the task.
    - isCompleted (TINYINT(1), DEFAULT 0): Indicates whether the task is completed (boolean value).

- **Reminder:** This entity stores reminders set by users. It includes the following information:

- ○ reminderID (INT, PRIMARY KEY, AUTO_INCREMENT): Unique identifier for each reminder.
- ○ userID (INT, FOREIGN KEY, REFERENCES Users(id)): The user associated with the reminder.
- ○ text (TEXT): The reminder text.
- ○ reminderTime (DATETIME): Time and date when the reminder should be triggered.

- **Event:** This entity stores information about events created by users, containing:
  - ○ eventID (INT, PRIMARY KEY, AUTO_INCREMENT): Unique identifier for each event.
  - ○ userID (INT, FOREIGN KEY, REFERENCES Users(id)): The user associated with the event.
  - ○ title (VARCHAR(255)): Title of the event.
  - ○ eventDate (DATE): Date of the event.
  - ○ eventTime (TIME): Time of the event.

- **Note:** This entity represents notes taken by users, with the following attributes:
  - ○ noteID (INT, PRIMARY KEY, AUTO_INCREMENT): Unique identifier for each note.
  - ○ userID (INT, FOREIGN KEY, REFERENCES Users(id)): The user associated with the note.
  - ○ content (TEXT): The content of the note.
  - ○ tags (VARCHAR(255)): Optional tags associated with the note.

- **Book:** This entity tracks books being read by users, with these attributes:
  - bookID (INT, PRIMARY KEY, AUTO_INCREMENT): Unique identifier for each book.
  - userID (INT, FOREIGN KEY, REFERENCES Users(id)): The user associated with the book.
  - title (VARCHAR(255)): Title of the book.
  - author (VARCHAR(255)): Author of the book.
  - startDate (DATE): Date when the user started reading the book.
  - isCompleted (TINYINT(1), DEFAULT 0): Indicates whether the book is completed (boolean value).

Relationships:

The relationships between these entities represent how the data is connected and how different entities interact:

- Users can have many tasks, reminders, events, notes, and books. Each user can associate multiple tasks, reminders, events, notes, and books with their account.
- Tasks, reminders, events, notes, and books are associated with a specific user. Each task, reminder, event, note, or book belongs to a single user.
- An Admin can access the details of multiple users. Admins have a broader view and can manage information related to multiple users within the application.
- Each User belongs to only one Admin. This hierarchical relationship ensures that each user is assigned to a specific administrator for administrative purposes.

4.3.3 Data Storage:

The Productivity Nexus application utilizes a relational database (MySQL) managed through XAMPP. XAMPP provides a local development environment that includes Apache (web server), MySQL (database), and other components, which enables the database to be set up and accessed locally. The database schema includes separate tables for Users and Admins, with a foreign key relationship linking the two tables. This allows admins to easily access and manage user data.

Explanation:
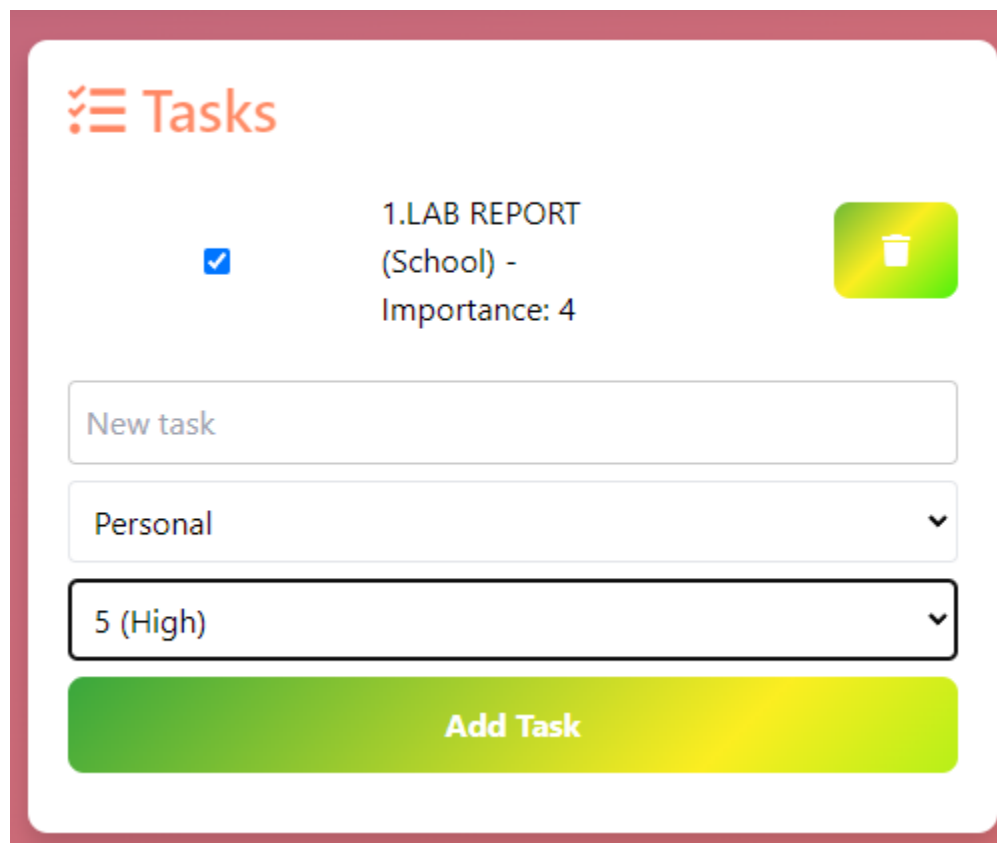
- **XAMPP:** XAMPP provides a convenient and self-contained development environment that includes the necessary components (Apache, MySQL, PHP) for running the web application and managing the database.
- **Structured Data:** MySQL's relational model excels at storing data in a structured format, which aligns well with the nature of the data in this application (user information, task lists, reminders, etc.).
- **Data Integrity:** The relational model's use of primary keys, foreign keys, and constraints helps to ensure the accuracy and consistency of the data.
- **Efficient Querying:** SQL (Structured Query Language) is a powerful language used for querying data in relational databases, providing flexibility and efficiency for retrieving and managing information.

5. Features:
5.1 Detailed Feature Descriptions:

5.1.1 Task Management:

- **Creation:** Users can create tasks by entering a title, description (optional), due date, priority level (low, medium, high), and a checkbox to mark the task as complete.
- **Editing:** Tasks can be edited to modify any of the aforementioned details, including changing priority, adding a due date, or marking a task as complete.
- **Management:** Users can view their task list in a visually organized format. Tasks can be sorted by priority, due date, or completion status. The task list includes checkboxes to mark tasks as complete, allowing for easy tracking of progress.

5.1.2 Event Scheduling:

- **Adding Events:** Users can add events to their calendar by entering a title, date, and time. They can also add an optional description for the event.
- **Calendar Integration:** The event scheduling feature is integrated with a visual calendar, allowing users to easily view their scheduled events within a monthly, weekly, or daily view. This visual representation enhances user experience and helps with planning.
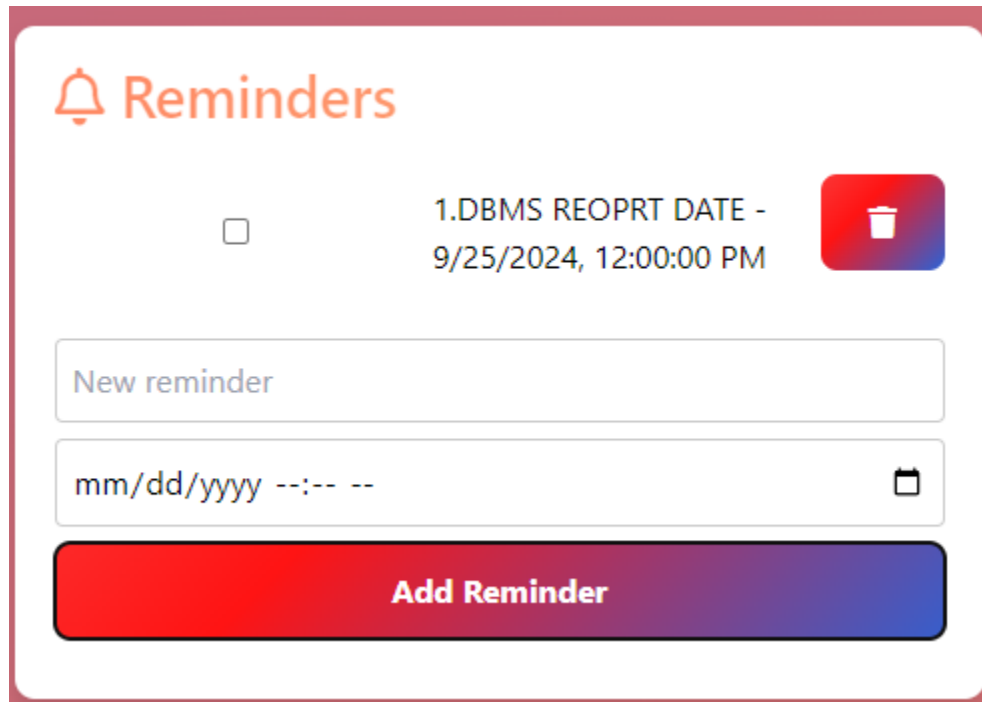
5.1.3 Reminders:

- **Setting Reminders:** Users can set reminders by entering the reminder text and specifying the time and date when the reminder should be triggered.
- **Reminder Delivery:** The application delivers reminders to users through a pop-up notification or alert at the designated time.

5.1.4 Productivity Score:

- **Calculation:** The productivity score is calculated based on a weighted average of task completion and reminder adherence.
  - Task Completion: Completed tasks contribute a score based on their priority level (high priority tasks contribute more).
  - Reminder Adherence: Completed reminders contribute a score based on their urgency (recent reminders contribute more).
  - Weighting: The score is calculated as a weighted average, giving more importance to task completion.

- **Display:** The productivity score is displayed visually as a percentage and a progress bar.
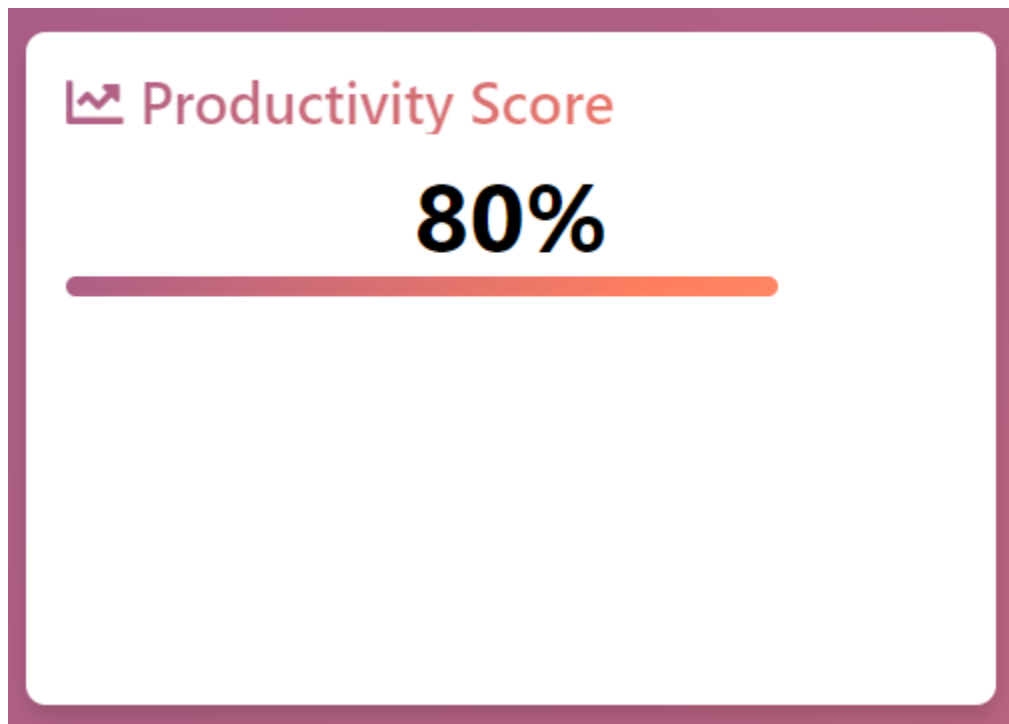
5.1.5 Water Intake Tracker:

- **Logging:** Users can log their water intake by clicking a button to indicate that they have consumed a glass of water.
- **Visual Tracking:** The water intake is tracked visually with a progress bar that fills as the user logs glasses of water.
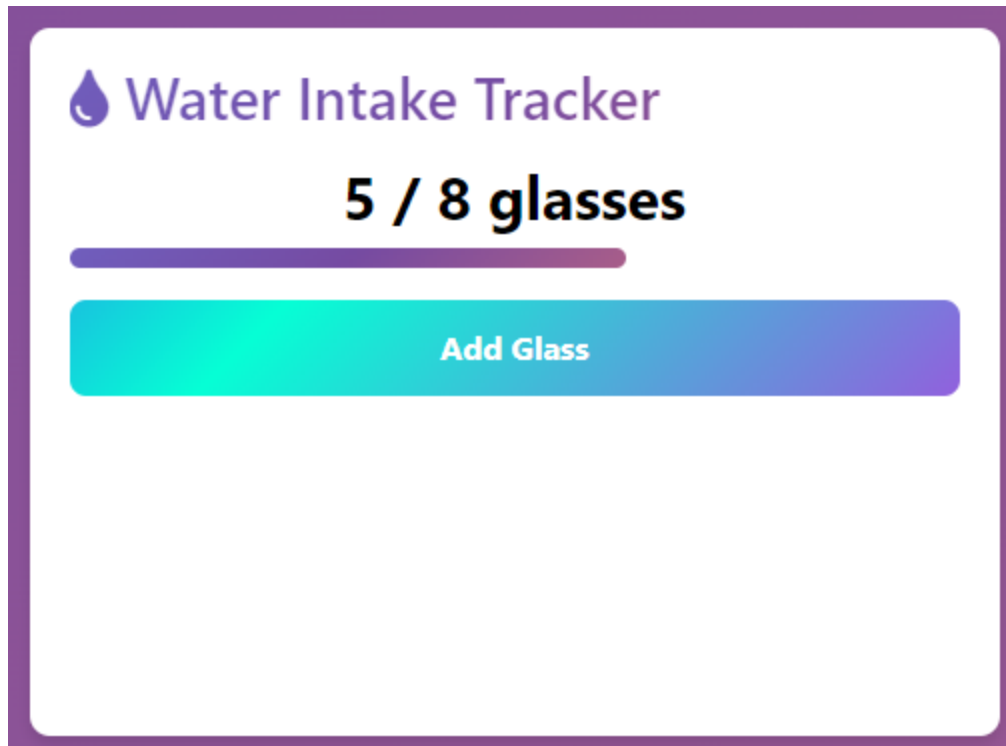
5.1.6 Focus Timer:

- **Pomodoro-Style:** The focus timer uses a pomodoro-style approach, alternating between focused work sessions and short breaks.
- **Customizable Intervals:** Users can customize the length of the work sessions and break intervals to suit their preferences.
- **Notifications:** The timer provides audio or visual notifications when a session is complete, reminding the user to take a break.
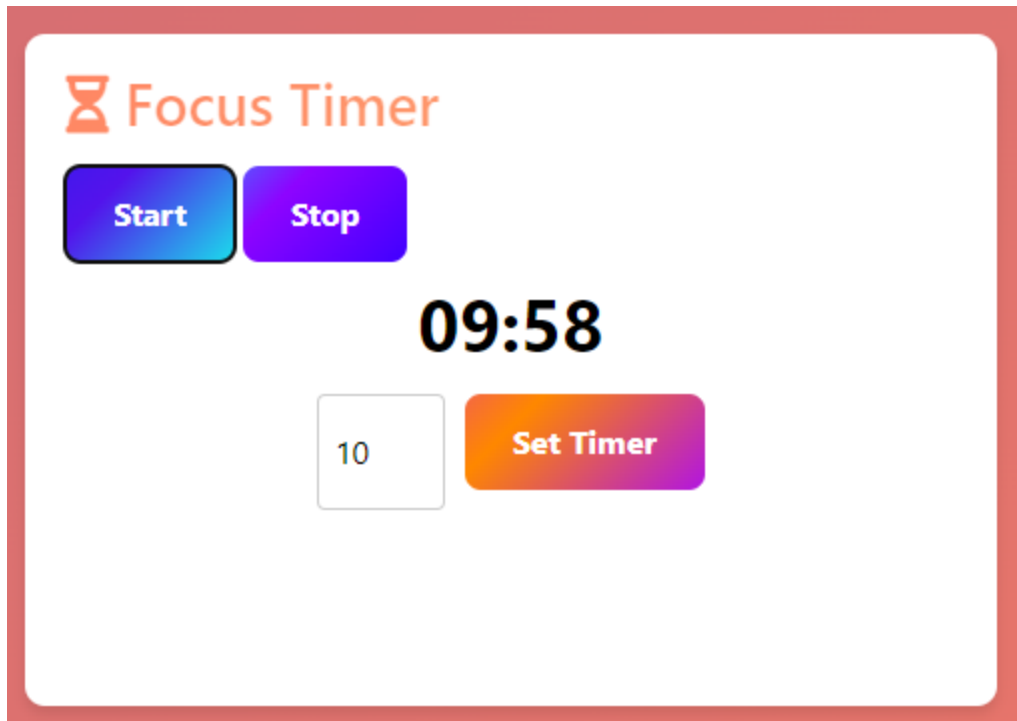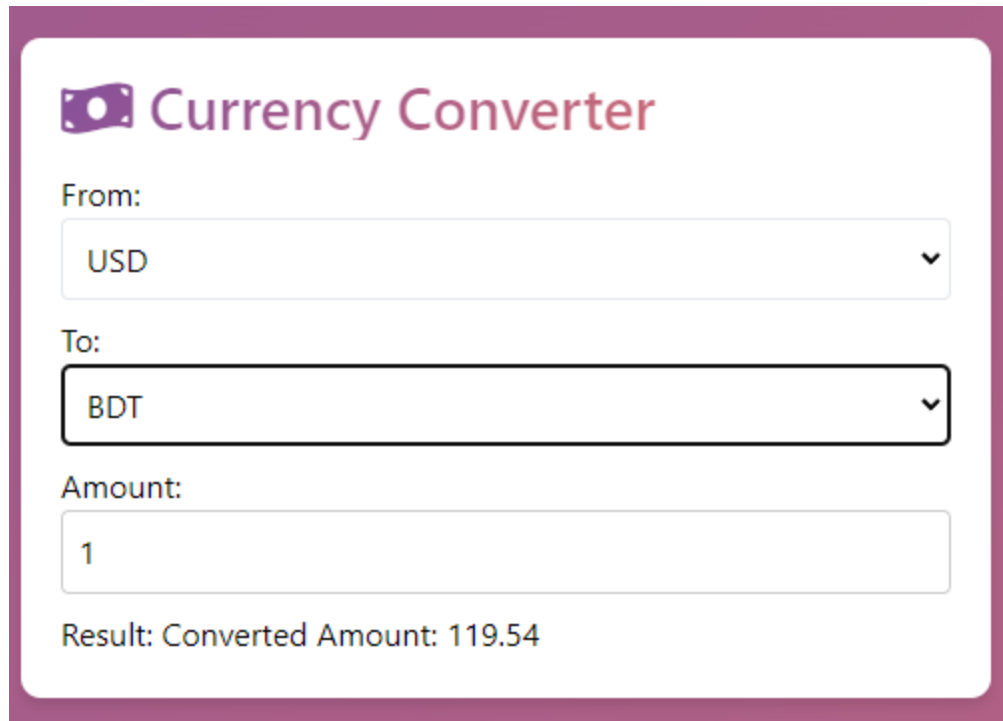
5.1.7 Mindfulness Exercises:

- **Breathing Exercise**: The widget offers a guided breathing exercise.
    - **Visuals:** A breathing circle visually guides the user through the inhale, hold, and exhale phases of each breath cycle.
    - **Cycle Tracking:** The widget tracks the number of completed breath cycles.
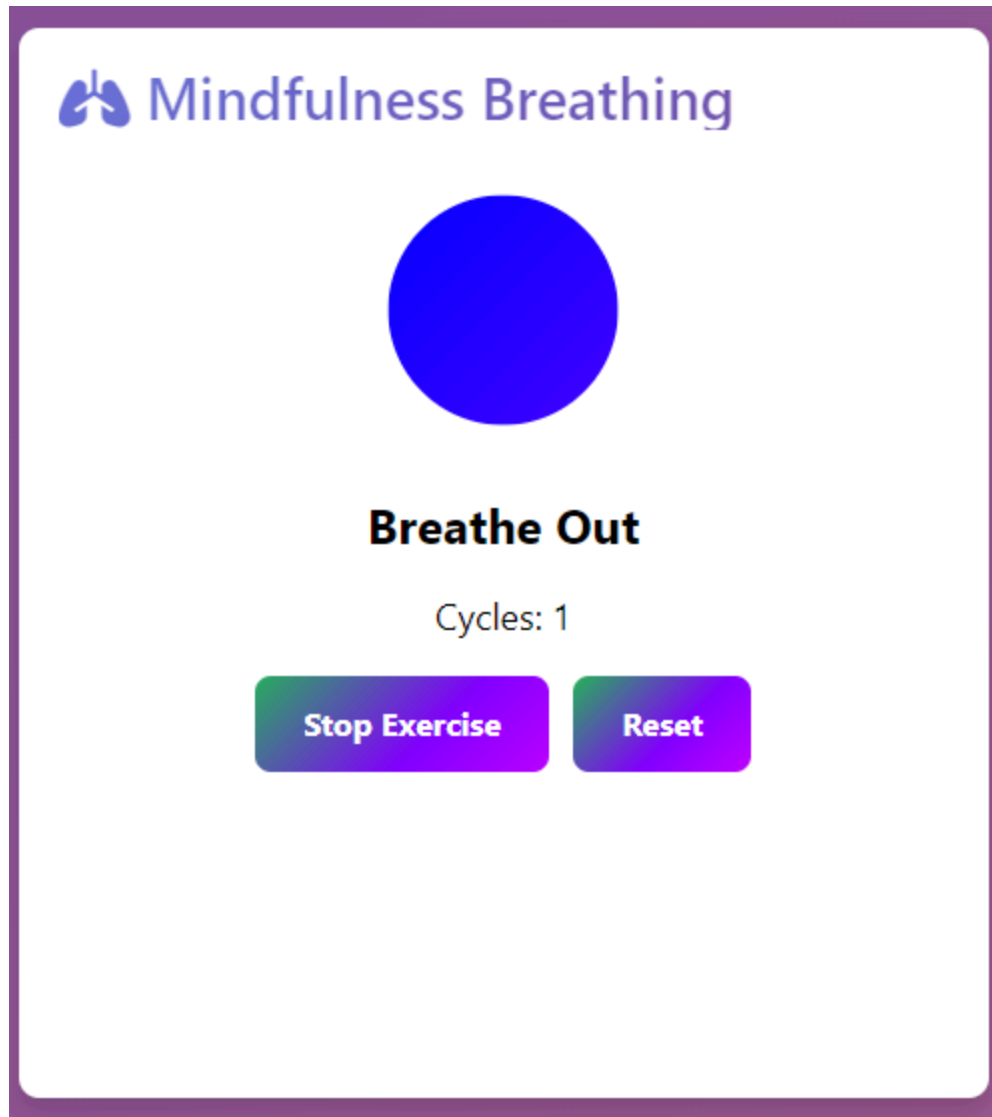
5.1.8 Currency Converter:

- **Dropdowns:** Users select the "from" and "to" currencies from dropdown menus.
- **Input Field:** Users enter the amount they want to convert into the input field.
- **Conversion:** The application displays the converted amount in real-time based on the latest exchange rates retrieved from an external API (e.g., Fixer.io).

5.1.9 Weather & Air Pollution:

- **Data Sources:** The weather and air pollution data is retrieved from OpenWeatherMap's API.
- **User Interaction:** Users can select their city from a dropdown menu.
- Display: The application displays the current weather conditions (temperature, description, and icon) and the Air Quality Index (AQI) for the selected city.
- **Pollutant Information:** It displays a list of pollutants and their current levels in the air (e.g., PM2.5, Ozone).

⛅ Weather

Chittagong,BD ⌄

Chittagong, BD
26°C
light rain

📅 5-Day Forecast

Chittagong,BD ⌄

Wed	🌧️	**26°C**
Thu	🌧️	**25°C**
Fri	🌧️	**25°C**
Sat	🌧️	**25°C**
Sun	🌧️	**26°C**
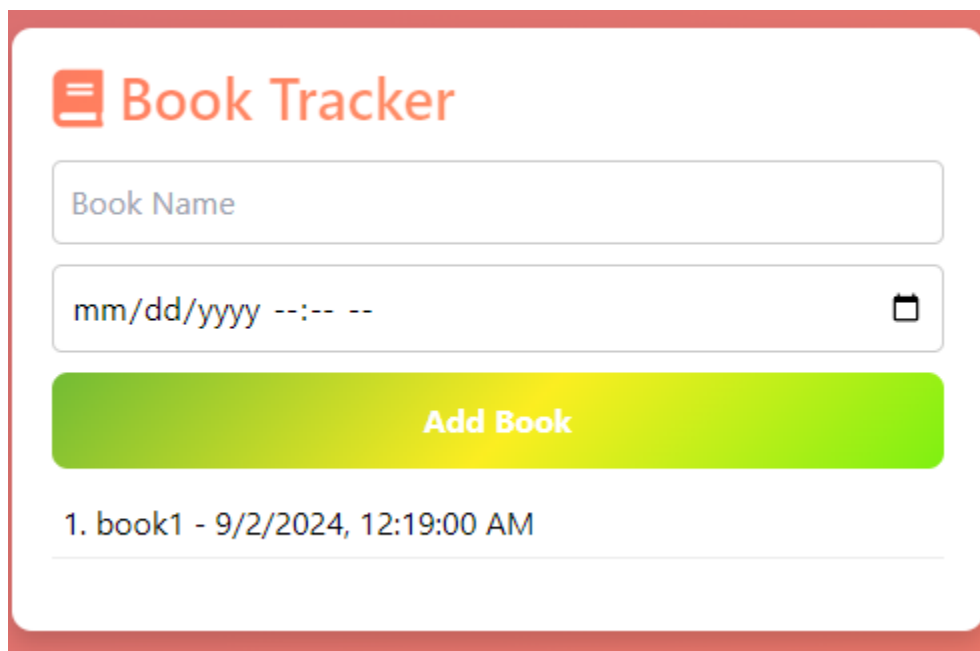
5.1.10 Book Tracker:

- **Adding Books:** Users can add books they are reading by entering the title and author. They can also specify the start date.
- **Tracking:** The application allows users to mark books as complete when they finish reading them.
- **Book List:** The book tracker presents a list of books the user is currently reading, with the option to mark books as complete.

5.1.11 Notes:

- **Creating Notes:** Users can create new notes by typing in the note content.
- **Editing Notes:** Notes can be edited to modify their content.
- **Storing Notes:** Notes are saved in the database, allowing users to access them later.

5.1.12 Games:

- **Sudoku:** The Sudoku game provides a classic Sudoku puzzle.
    - **Difficulty:** You could offer multiple difficulty levels.
    - **Hints:** Users can request hints to help them solve the puzzle.
    - **Checking Solutions:** Users can check their solution against the correct answer.

Sudoku Game

6	7	2	1				4	8
		4			8	9		
		8	3			5		7
7	1	3	9	2	4			6
8	5	9	7			4	2	3
4	2						9	
2		7	4		9		3	
9	6		5	3	7	2	8	4
3		5		8		1		9

New Game

Solution

Hint

Check Solved

- **Memory Match:** The Memory Match game presents a board of cards with matching pairs.
    - **Difficulty:** You can offer different board sizes and numbers of cards for varying difficulty levels.
    - **Scorekeeping:** The game tracks the player's score and moves.

Memory Match

Choose Difficulty:

Medium (25) ▼ **Start Game**
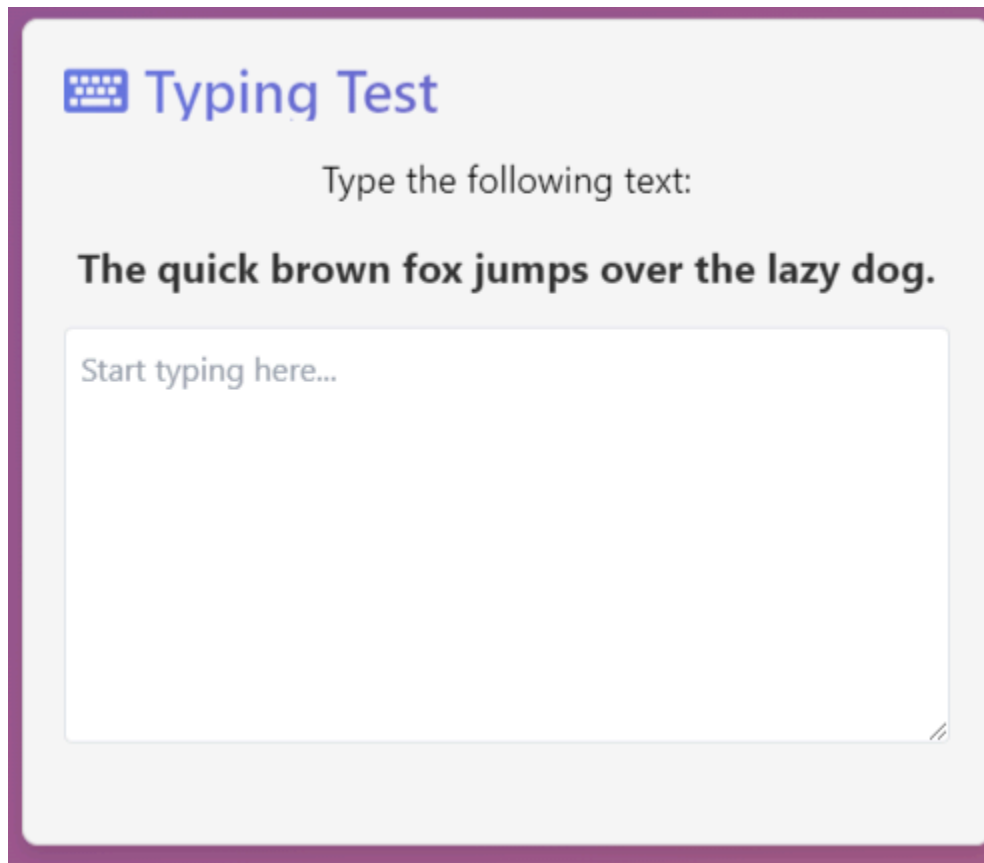
Score: 40

Moves: 17

5.1.13 Typing Test:

- **Text to Type:** The typing test displays a sample paragraph of text for users to type.
- **Time Tracking:** The application tracks the time it takes the user to type the text.
- **Results:** The typing test calculates and displays the user's words per minute (WPM) and accuracy.

5.1.14 Mapping:

- **User Interaction:** Users can click on the map to place markers.
- **Coordinates:** When a marker is placed, the application displays the latitude and longitude coordinates of the selected location.
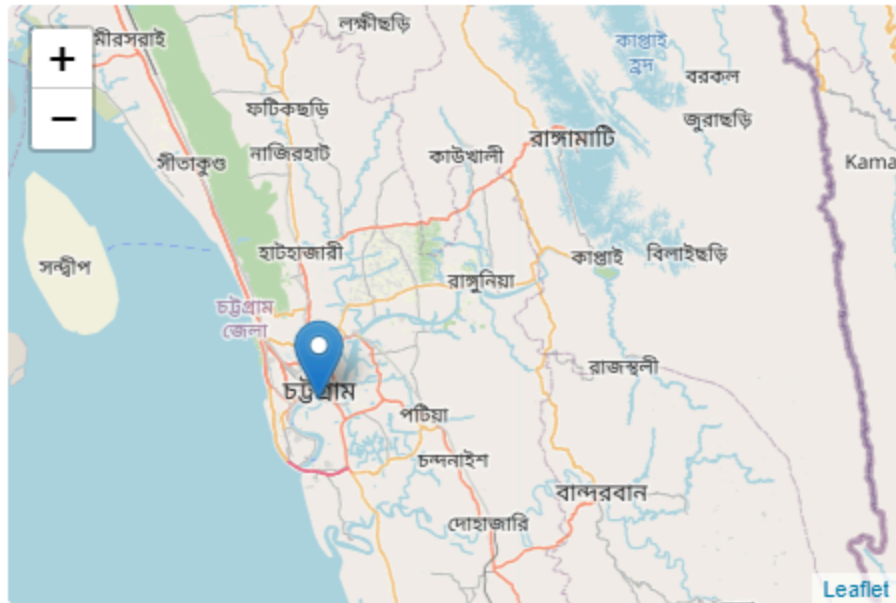
5.1.15 Social:

- Facebook
- Gmail
- Telegram
- Twitter
- Notion
- Discord
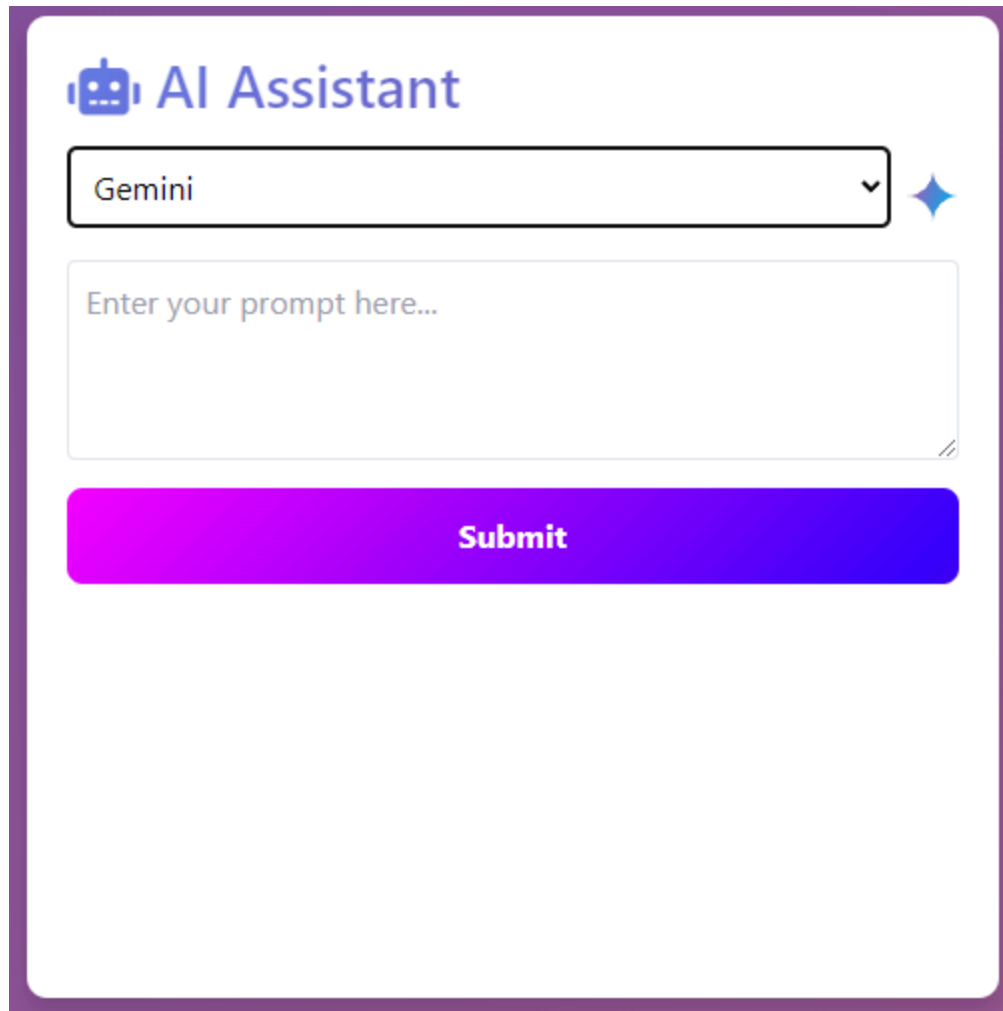- WhatsApp
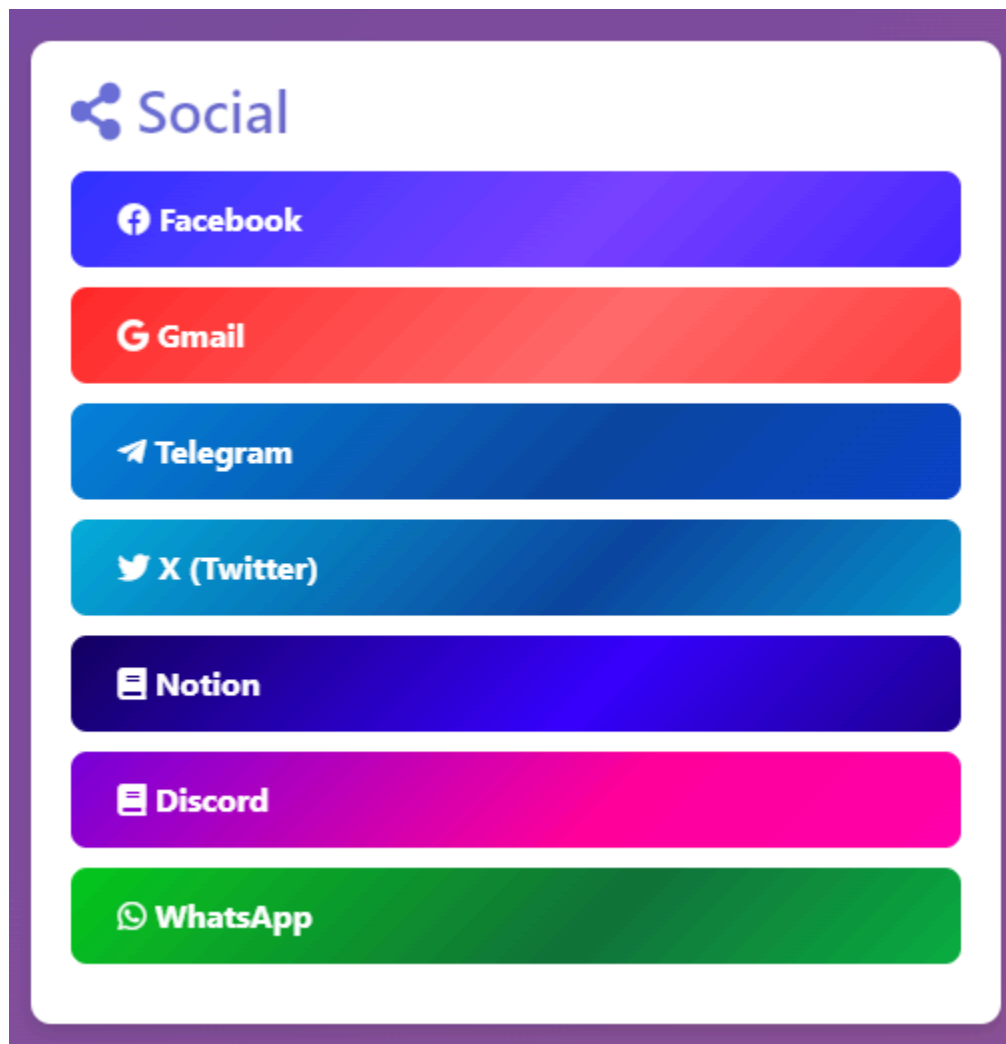
5.1.16 Competitive Programming:

- Codeforces
- AtCoder

- LeetCode
- Project Euler
- CodeChef
- HackerRank
- TopCoder

5.1.17 AI Assistant:

- Gemini
- Ai Studio
- Chatgpt
- Perplexity
- Claude

5.2 User Interface (UI) and User Experience (UX):

Signup Page:

Dashboard Page:

Logout

🕐 Clock

8:30:24 PM
Sunday, September 15, 2024

☰ Tasks

- ☑ task (Work) – Importance: 4
- ☑ task2 (School) - Importance: 5
- ☐ task3 (Personal) - Importance: 1

New task
Personal
1 (Low)
Add Task

📅 Upcoming Events

exam1 - 2024-09-13
exam2 - 2024-09-14
exam3 - 2024-09-15

New event
mm/dd/yyyy
Add Event

🔔 Reminders

- ☐ reminder1 – 9/1/2024, 8:26:00 PM
- ☐ reminder2 – 9/13/2024, 8:26:00 PM

New reminder
mm/dd/yyyy --:-- --
Add Reminder

📈 Productivity Score

75%

⏳ Focus Timer

Start Stop
06:53
3 Set Timer

💧 Water Intake Tracker

6 / 8 glasses
Add Glass

📹 Currency Converter

From:
USD
To:
BDT
Amount:
1
Result: Converted Amount: 119.54

🌤 Weather

Chittagong,BD
Chittagong, BD
26°C
overcast clouds

📅 5-Day Forecast

Chittagong,BD

Sun		26°C
Mon		27°C
Tue		27°C
Wed		27°C
Thu		26°C

🌫 Air Quality

City: Chittagong,BD
AQI: 1
Good - Air quality is considered satisfactory, and air pollution poses little or no risk.

CO	370.50 μg/m³
NO	0.00 μg/m³
NO2	7.28 μg/m³
O3	34.49 μg/m³
SO2	3.16 μg/m³
PM2.5	4.23 μg/m³
PM10	5.50 μg/m³
NH3	3.45 μg/m³

📕 Book Tracker

Book Name
mm/dd/yyyy --:-- --
Add Book
1. Book1 - 9/3/2024, 8:27:00 PM
2. Book2 - 9/11/2024, 8:27:00 PM
3. Book3 - 9/15/2024, 8:27:00 AM

⌗ Social

- 🅕 Facebook
- Ⓖ Gmail
- ✈ Telegram
- 🐦 X (Twitter)
- 🅝 Notion
- 🅓 Discord
- ⓦ WhatsApp

</> Competitive Programming

- </> Codeforces
- </> AtCoder
- </> LeetCode
- </> Project Euler
- </> CodeChef
- </> HackerRank
- </> TopCoder

🖥 AI Assistant

ChatGPT
This is a Productivity Website
Submit

🧘 Mindfulness Breathing

Breathe In
Cycles: 12
Stop Exercise Reset

🖩 Calculator

02807554

7	8	9	/
4	5	6	*
1	2	3	-
0	.	=	+

Clear

📝 Notes

Take some notes...
Save Notes

🎴 Memory Match

Choose Difficulty:
Hard (36) Start Game

Score: 70
Moves: 25

Sudoku Game

High Score: 10
Lives: 3

6	7	2	1	9	5	3	4	8
1	9	8	3	4	2	5	6	7
5	3	4	6	7	8	9	1	2
4	2	6	8	5	3	7	9	1
7	1	3	9	2	4	8	5	6
8	5	9	7	6	1	4	2	3
2	8	7	4	1	9	6	3	5
3	4	5	2	8	6	1	7	9
9	6	1	5	3	7	2	8	4

New Game
Solution
Hint
Check Solved

📍 Location

Latitude: 22.2974
Longitude: 91.8622

⌨ Typing Test

Type the following text:
The quick brown fox jumps over the lazy dog.

Start typing here...

Words per Minute: 74
Accuracy: 100.00%

- **Color Scheme:** Explain your choice of colors (consistent with the overall theme, good contrast for readability, and pleasing to the eye).
- **Layout:** Describe the layout of the dashboard (organized, easy to navigate, logically grouped features).
- **Typography:** Mention the fonts you used (clear, legible, consistent with the brand).
- **Accessibility:** Discuss any steps taken to make the UI accessible (alt text for images, keyboard navigation).
- **User Testing:** Mention any user testing you conducted to gather feedback on UI/UX and how you incorporated those suggestions.

5.3 Technology Stack:

- **HTML:** 5.2
- **CSS: 4.0** (Tailwind CSS)
- **JavaScript:** ES6
- **Backend Language:** PHP 7.4
- **Database:** MySQL 5.7
- **API:** OpenWeatherMap API (for weather & air pollution),World Map ,Foreign Exchange and currency converter.
- **Libraries/Frameworks:** Any additional libraries or frameworks you used (e.g., for form validation, charts, etc.).

6. Results & Analysis
6.1 Evaluation:

- **User Testing:** Describe how you conducted user testing (e.g., recruiting participants, tasks they performed, feedback collected).
- **Self-Evaluation:** Explain how you tested the application's functionality and usability yourself.

- **Comparison to Other Tools:** Did you compare the Productivity Nexus to other productivity tools? Discuss your findings.

6.2 Key Findings:

- **Strengths:** Highlight the strengths of your application based on the evaluation.
    - Did it meet the objectives you set out?
    - What did users like about the app?

- **Weaknesses:** Discuss any weaknesses or areas where improvements can be made.
    - What challenges did you encounter?
    - What user feedback was less positive?

6.3 Data:

- **Usage Statistics:** Include any data you collected about application usage (e.g., number of users, frequency of use, most popular features).
- **User Feedback:** Share any qualitative feedback you collected from users (quotes, general impressions).

7. Conclusion:

7.1 Summary of Contributions:

Summarize the key contributions of your Productivity Nexus project. Highlight the following:

- **Functionality:** What specific features did you develop and implement?

- **Design:** What were your key design decisions for the UI and UX?
- **Innovation:** Did your project introduce any new or innovative approaches to productivity?

7.2 Future Work:

Discuss potential areas for future development and improvement of your project:

- **New Features:** What new features could you add to enhance the user experience?
- **UI Enhancements:** How could you further refine the UI to make it more user-friendly or visually appealing?
- **Mobile Responsiveness:** Consider making the application more responsive for mobile devices.
- **Data Visualization:** Could you improve data visualization with charts or graphs to make the productivity score more engaging?

9.Source:

- **Github :** Nahid_Ibna_Harun
- **Live Link of Application :** Productivity_Nexus

8.API:

- **OpenWeatherMap:**

  1. **Current Weather :** Current Weather

  2.**3-hour Forecast 5 days:** 3-hour Forecast 5 days

  3. **Air Pollution API :** Air Pollution API

  4.**Documentation :** OpenWeatherMap

- **Foreign Exchange and currency converter :** **Foreign Exchange**

  1.Documentation : **Fixer.com**

- **World Map :** **Leaflet**

  1.Documentation : **Leaflet_Documentation**