



Vivekanand Education Society's Institute Of Technology
Department Of Information Technology

G-HEALTH

Empowering your journey to better health and wellness

Presented by: Rushikesh Yeole
Roll no: 72
Div: D15B

UNDER THE GUIDANCE OF
DR.RAVITA MISHRA

Table of Contents

- 01** Introduction
- 02** Problem Statement
- 03** Objective
- 04** Requirements
- 05** Hardware Requirements
- 06** Implementation

Introduction

This Flutter app offers a variety of health and wellness tools, including a BMI calculator, exercise tracker, nutrition tracker, sleep tracker, and meditation timer. Users can easily monitor their fitness, dietary habits, sleep patterns, and mindfulness practices. Additionally, the app features a sign-up form for personalized experiences. Designed to support users in achieving their health goals, it provides a convenient platform for tracking and managing their overall well-being.



Problem Statement

Develop a Flutter-based health and wellness app that serves as a comprehensive dashboard for users to monitor and manage various aspects of their health

Objective

The goal is to create a convenient platform that supports users in achieving their fitness milestones, maintaining a balanced diet, improving sleep quality, and incorporating meditation into their daily routine, ultimately promoting better health and well-being.

Requirements

- Platform: I chose Flutter for its cross-platform compatibility, allowing me to develop for both iOS and Android devices simultaneously.
- Database: Firebase was selected for its robust authentication and real-time data storage capabilities, ensuring secure user registration and efficient data management.



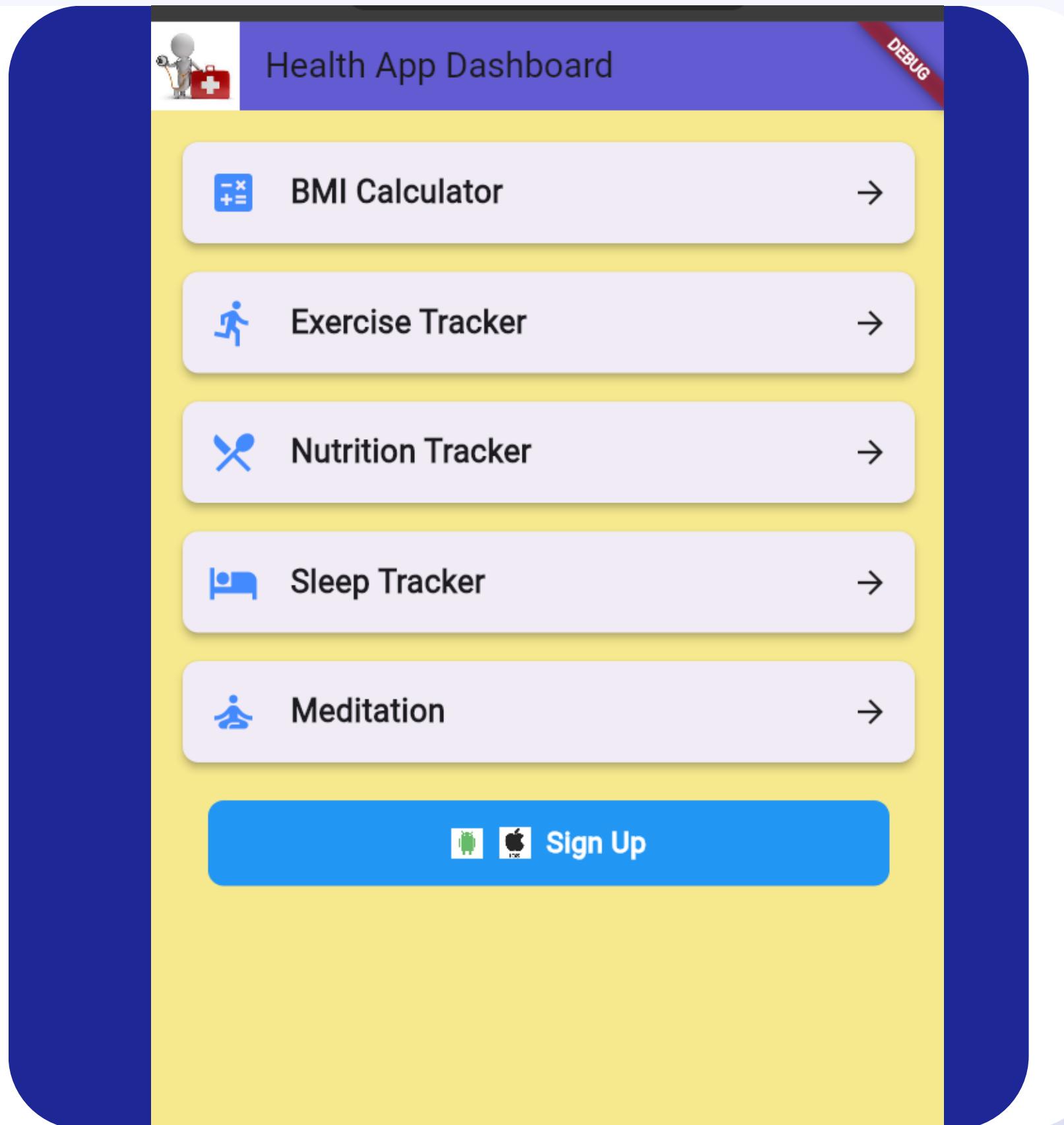
Hardware Requirements

- Processor: Intel Core i5
- RAM: 8 GB
- 10GB Storage
- Stable internet connection
- Mobile Devices
- Screen Resolution



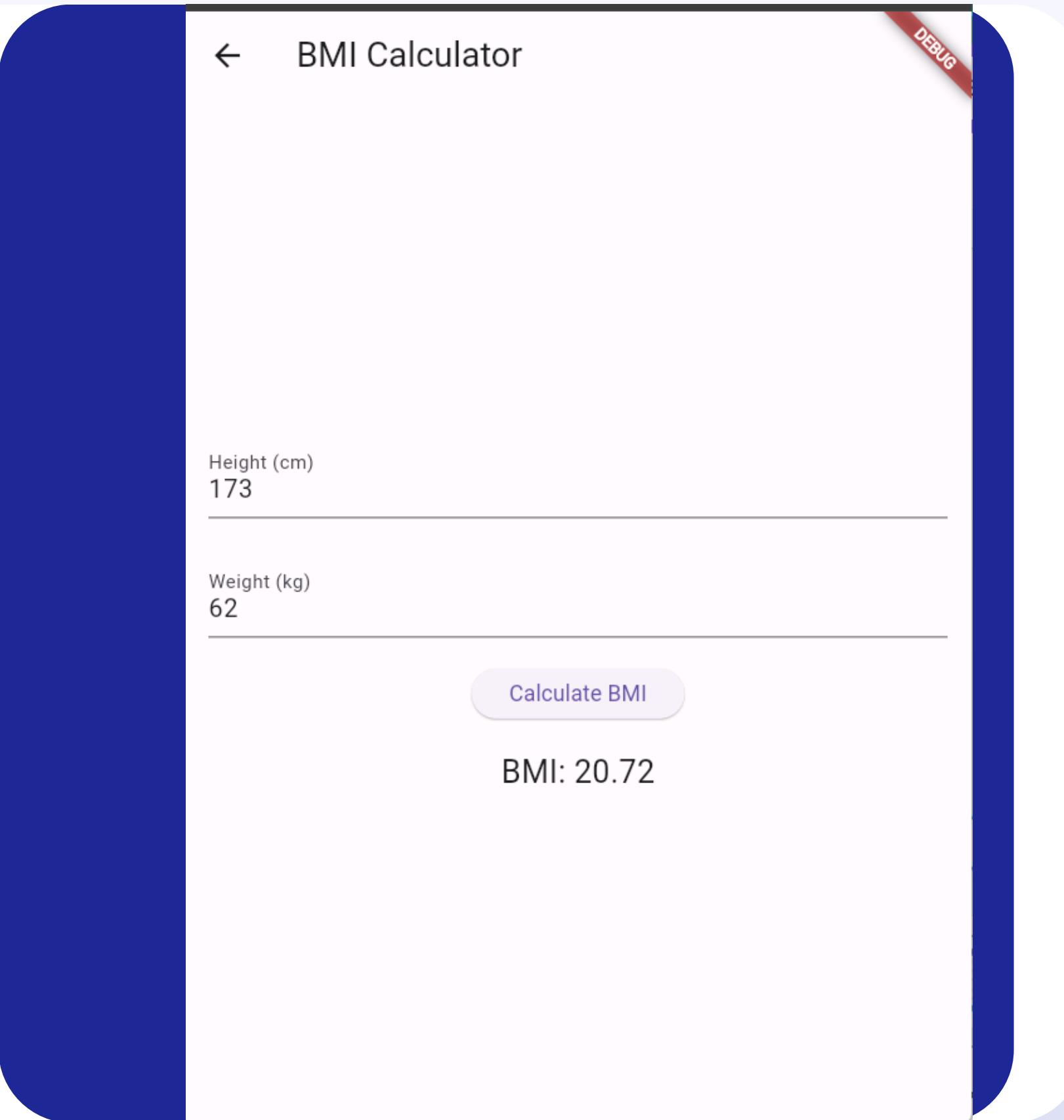
Dashboard

- Buttons for navigating to different sections: BMI Calculator, Exercise Tracker, Nutrition Tracker, Sleep Tracker, Meditation.
- SignUp Option.



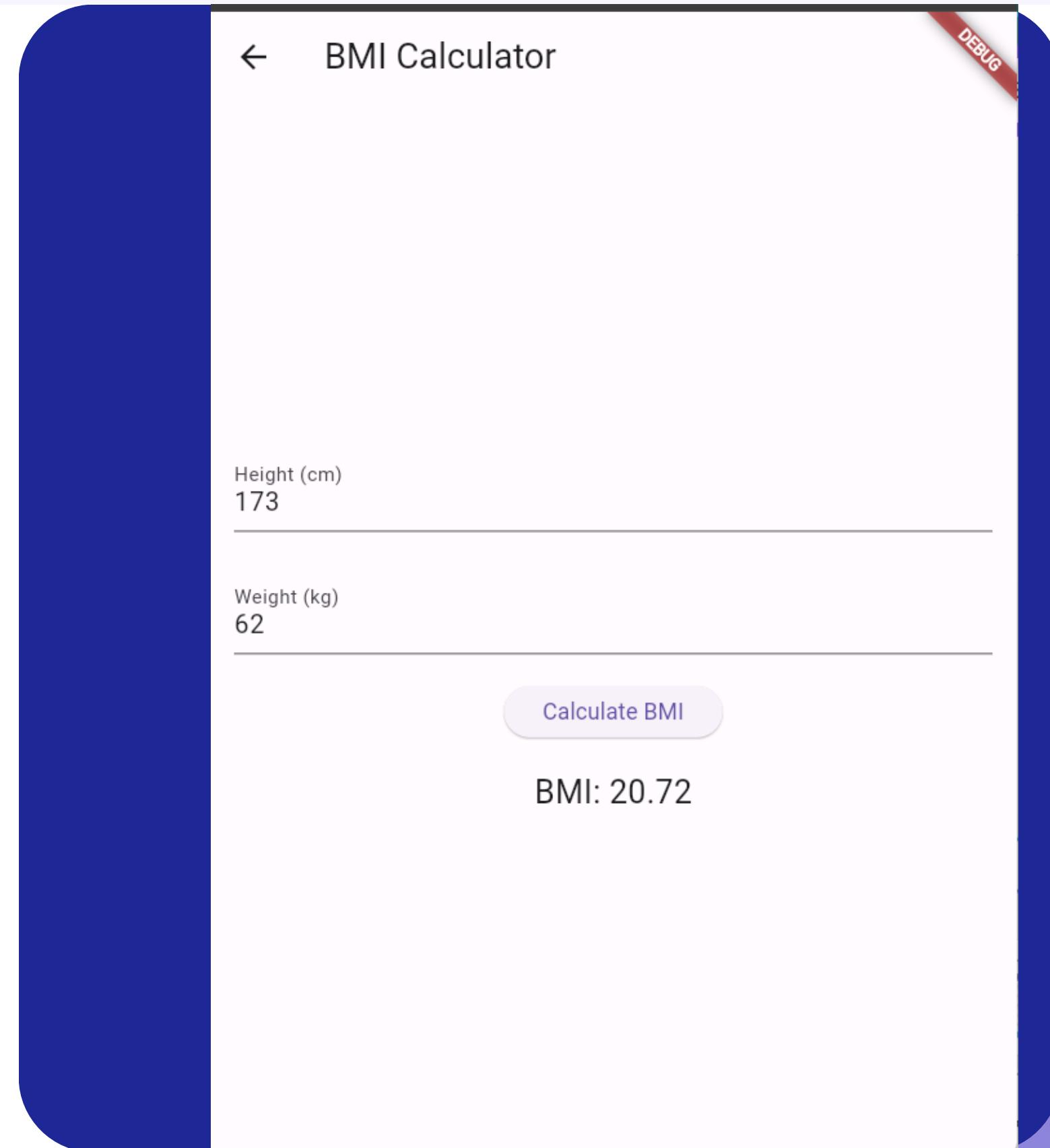
Sign Up

- Intuitive design with fields for name,username, email, and password.
- Efficient authentication mechanisms with error feedback using a snack bar.
- Option to navigate back to the login page for existing users.



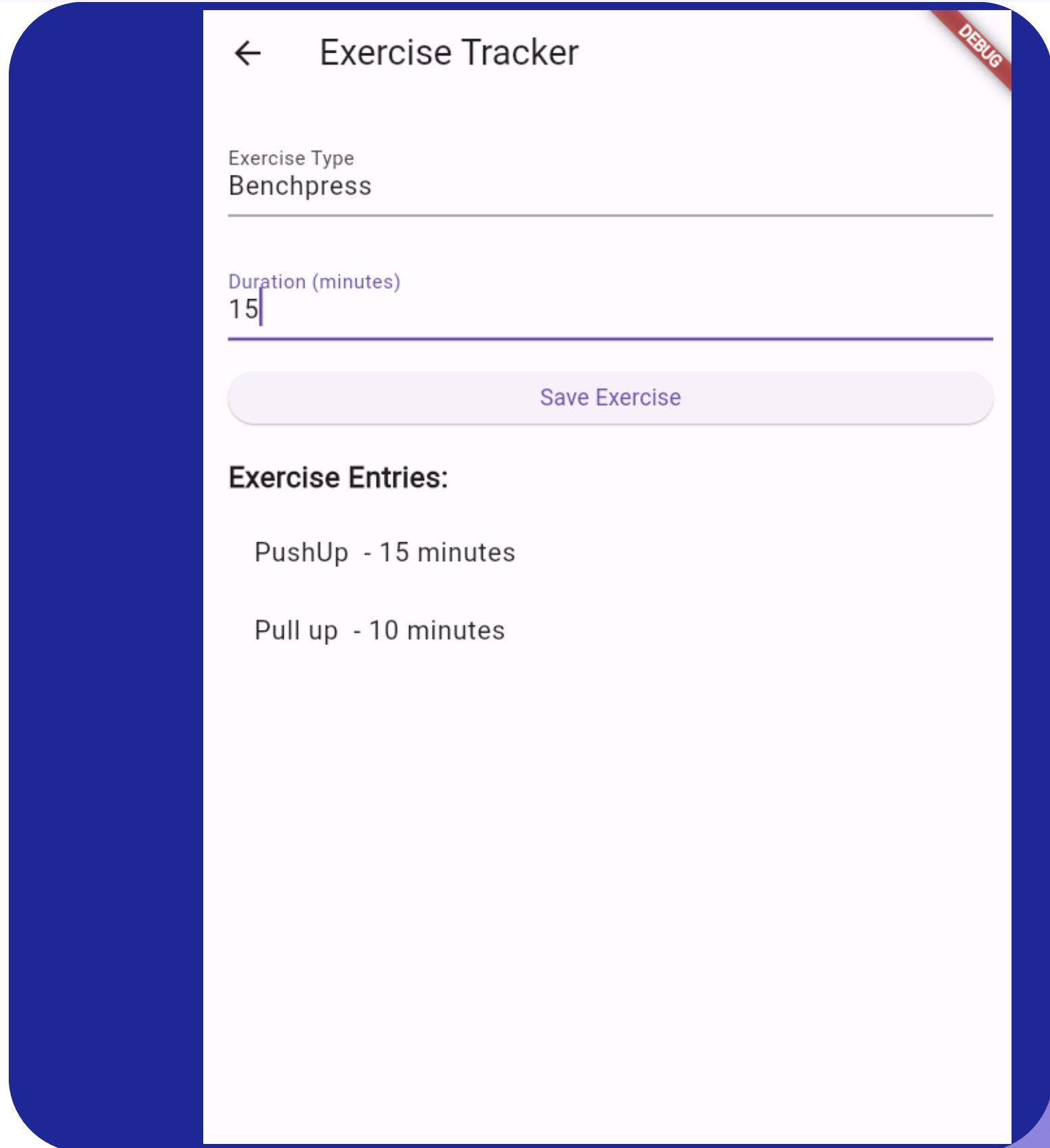
BMI Calculator

- The BMI calculator provides users with a personalized assessment of their body mass index based on input data such as height and weight.



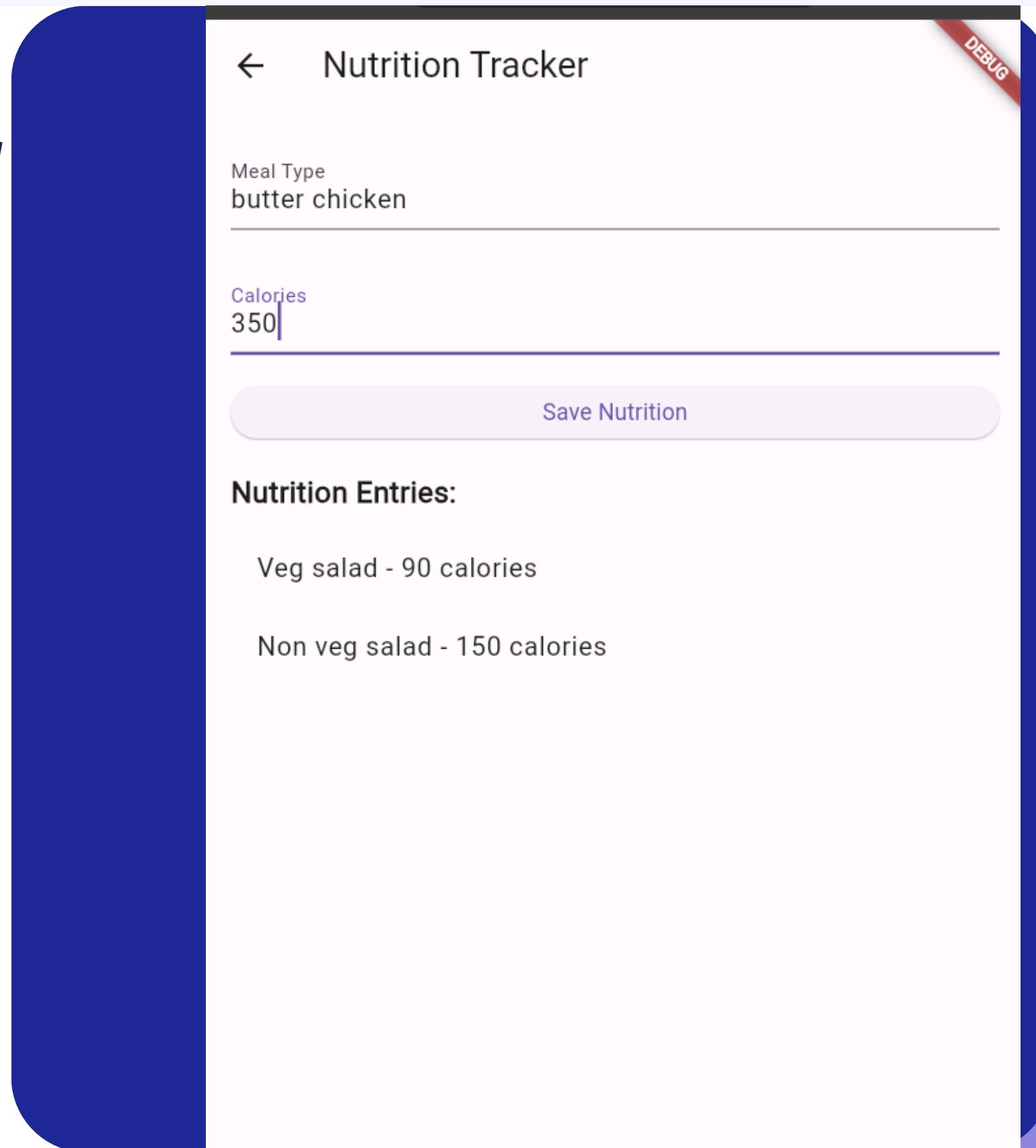
Exercise Tracker

- The exercise tracker should enable users to log various physical activities they engage in, such as running, cycling or yoga.
- Users can input details like exercisetype,duration, intensity, and any additional notes to keep a comprehensive record of their workouts.



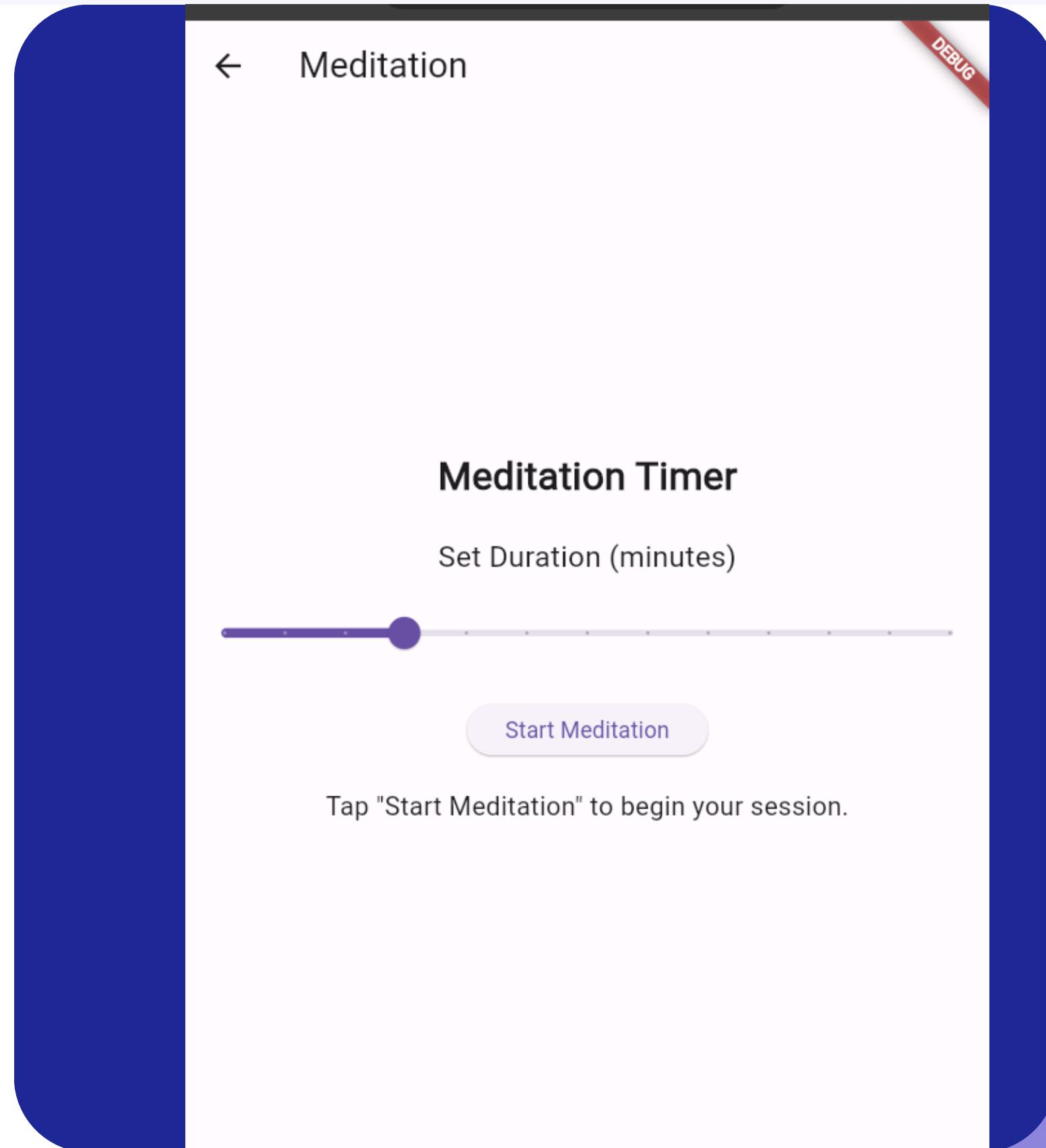
Nutrition Tracker

- The nutrition tracker allows users to log their daily meals and snacks, including details such as meal type and nutritional content.
- Users can input information about the food they consume, including calories.



Meditation Timer

- The meditation timer offers users the flexibility to customize their meditation sessions according to their preferences and goals.



Authentication

The screenshot shows the Firebase Authentication console interface. On the left, a sidebar navigation bar includes links for Project Overview, Authentication (which is selected and highlighted in blue), Realtime Database, Storage, App Check, Firestore Database, What's new, Extensions (NEW), and Release Monit... (NEW). Below these are sections for Product categories (Build, Release & Monitor, Analytics) and a note about Spark (No-cost \$0/month) with an Upgrade button.

The main content area is titled "Authentication" and displays a table of users. The table has columns for Identifier, Providers, Created, Signed In, and User UID. It lists two entries:

Identifier	Providers	Created	Signed In	User UID
rushikesh@gmail.com	✉️	Apr 4, 2024	Apr 4, 2024	ivMe0mlKumhAE6PUcL5L4Qd...
test@gmail.com	✉️	Mar 3, 2024	Mar 12, 2024	qG1UYc7E9UUzV0c70bfnh08I...

At the top of the main area, there is a search bar labeled "Search by email address, phone number, or user UID", an "Add user" button, and a refresh/circular arrow icon. Below the table, there are controls for "Rows per page" (set to 50), page navigation (1 - 2 of 2), and a three-dot menu icon.

LITERATURE SURVEY

Author	Title	link
Philip Olla, Caley Shimskey	a literature survey of mobile health applications	https://www.researchgate.net/publication/276099168_mHealth_taxonomy_a_literature_survey_of_mobile_health_applications
Anu Grace Sam , K. Nithya& R. Murali Krishnan	A Study on The Usage of M-Health Apps	https://www.sdnbvc.edu.in/wp-content/uploads/2023/01/PAPERID-1.pdf
<u>Julian Wangler</u> and <u>Michael Jansky</u>	The use of health apps in primary care	https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8057994/

References

1. <https://mdanderson.libguides.com/appsguide/medrefapps>
2. <https://www.ncbi.nlm.nih.gov/books/NBK580938/4>
3. <https://telehealthtechnology.org/toolkit/mhealth-app-selection-references-and-literature-review/>

Conclusion

The goal is to create a convenient platform that supports users in achieving their fitness milestones, maintaining a balanced diet, improving sleep quality, and incorporating meditation into their daily routine, ultimately promoting better health and well-being.



Vivekanand Education Society's Institute Of Technology
Department Of Information Technology

ECOMMERCE PWA

Presented by: Rushikesh Yeole
Roll no: 72
Div: D15B

UNDER THE GUIDANCE OF
DR.RAVITA MISHRA

Introduction

Progressive Web Apps (PWA) are web apps built and enhanced with modern APIs to provide enhanced capabilities while still reaching any web user on any device with a single codebase. They combine the broad reach of web apps with the rich capabilities of platform-specific apps to enhance the user experience.



FEATURES IMPLEMENTED

- CREATING METADATA
- REGISTER A SERVICE WORKER
- SERVICE WORKER EVENTS SUCH AS PUSH,SYNC ,FETCH
- GOOGLE LIGHTHOUSE REPORT

Implementation

WATCH STORE

Home Shop About Contact

Featured Products



Rolex
\$200

Add to Cart



Rolex
\$250

Add to Cart



Jacob & Co
\$450

Add to Cart



Jacob & Co
\$550

Add to Cart

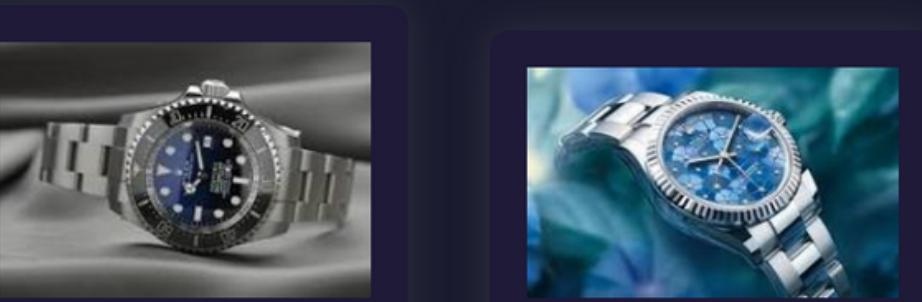
© 2024 Watch Store. All rights reserved.

Implementation

WATCH STORE

Home Shop About Contact

Featured Products



Rolex
\$200 **Add to Cart**

Rolex
\$250 **Add to Cart**

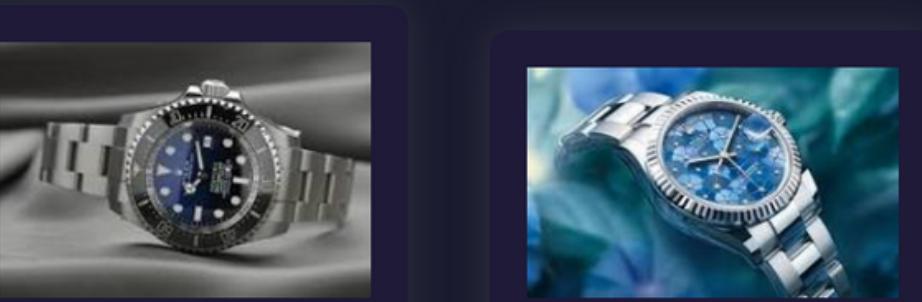
Elements Console Sources Network Performance Memory Application > Default levels ▾ No Issues

```
ServiceWorker registration successful with scope: http://127.0.0.1:5500/ index.html:17
```

WATCH STORE

Home Shop About Contact

Featured Products



Rolex
\$200 **Add to Cart**

Rolex
\$250 **Add to Cart**

Elements Console Sources Network Performance Memory Application > Default levels ▾ No Issues

```
Fetch request: http://127.0.0.1:5500/script.js serviceworker.js:34
Fetch request: http://127.0.0.1:5500/image/watch1.png serviceworker.js:34
Fetch request: http://127.0.0.1:5500/image/watch2.png serviceworker.js:34
Response from cache: http://127.0.0.1:5500/style.css serviceworker.js:39
Response from cache: http://127.0.0.1:5500/script.js serviceworker.js:39
Response from cache: http://127.0.0.1:5500/image/watch1.png serviceworker.js:39
Response from cache: http://127.0.0.1:5500/image/watch2.png serviceworker.js:39
Fetch request: http://127.0.0.1:5500/manifest.json serviceworker.js:34
Response from cache: http://127.0.0.1:5500/manifest.json serviceworker.js:39
ServiceWorker registration successful with scope: http://127.0.0.1:5500/ index.html:17
Fetch request: http://127.0.0.1:5500/image/watch1.png serviceworker.js:34
Response from cache: http://127.0.0.1:5500/image/watch1.png serviceworker.js:39
```

Implementation

The screenshot shows a web browser window displaying a "WATCH STORE" website. The site features a navigation bar with links for Home, Shop, About, and Contact. Below this is a section titled "Featured Products" featuring two watch images: a dark-colored watch and a silver-toned Rolex. The Rolex watch is labeled with the brand name "Rolex" and a price of "\$250". A purple "Add to Cart" button is visible below the watch image. In the bottom left corner of the page, there is a small "Cart" icon.

The main content area of the browser shows the "Application" tab of the developer tools. Under the "Service workers" section, it displays information for a service worker at the URL `http://127.0.0.1:5500/`. The source file is listed as `serviceworker.js`, and the status is shown as "#1582 activated and is running". There are buttons for "Push", "Sync", and "Periodic Sync" with their respective test tags: "Test push message from DevTools.", "test-tag-from-devtools", and "test-tag-from-devtools". The "Update Cycle" section shows the timeline for the current service worker version #1582, with steps for "Install", "Wait", and "Activate".

At the bottom of the developer tools interface, the "Console" tab is active, showing a log entry: "Push event received: PushEvent {isTrusted: true, data: PushMessageData, type: 'push', target: ServiceWorkerGlobalScope e, currentTarget: ServiceWorkerGlobalScope, ...}" with a file path of "serviceworker.js:61".

Implementation

WATCH STORE

Home Shop About Contact

Featured Products



Rolex
\$200

Add to Cart



Rolex
\$250

Add to Cart

© 2024 Watch Store. All rights reserved.

Application | Elements | Console | Sources | Network | Performance | Memory | Application | 1 | ⚙️ | ⋮

Service workers

http://127.0.0.1:5500/ [Network requests](#) [Update](#) [Unregister](#)

Source: [serviceworker.js](#)
Received 4/4/2024, 5:52:30 PM

Status: #1582 activated and is running [stop](#)

Push: [Test push message from DevTools.](#) [Push](#)

Sync: [test-tag-from-devtools](#) [Sync](#)

Periodic Sync: [test-tag-from-devtools](#) [Periodic Sync](#)

Background services

- Back/forward cache
- Background fetch
- Background sync
- Bounce tracking mitigation
- Notifications

Update Cycle

Version	Update Activity	Timeline
#1582	Install	
#1582	Wait	
#1582	Activate	██████████

Console What's new

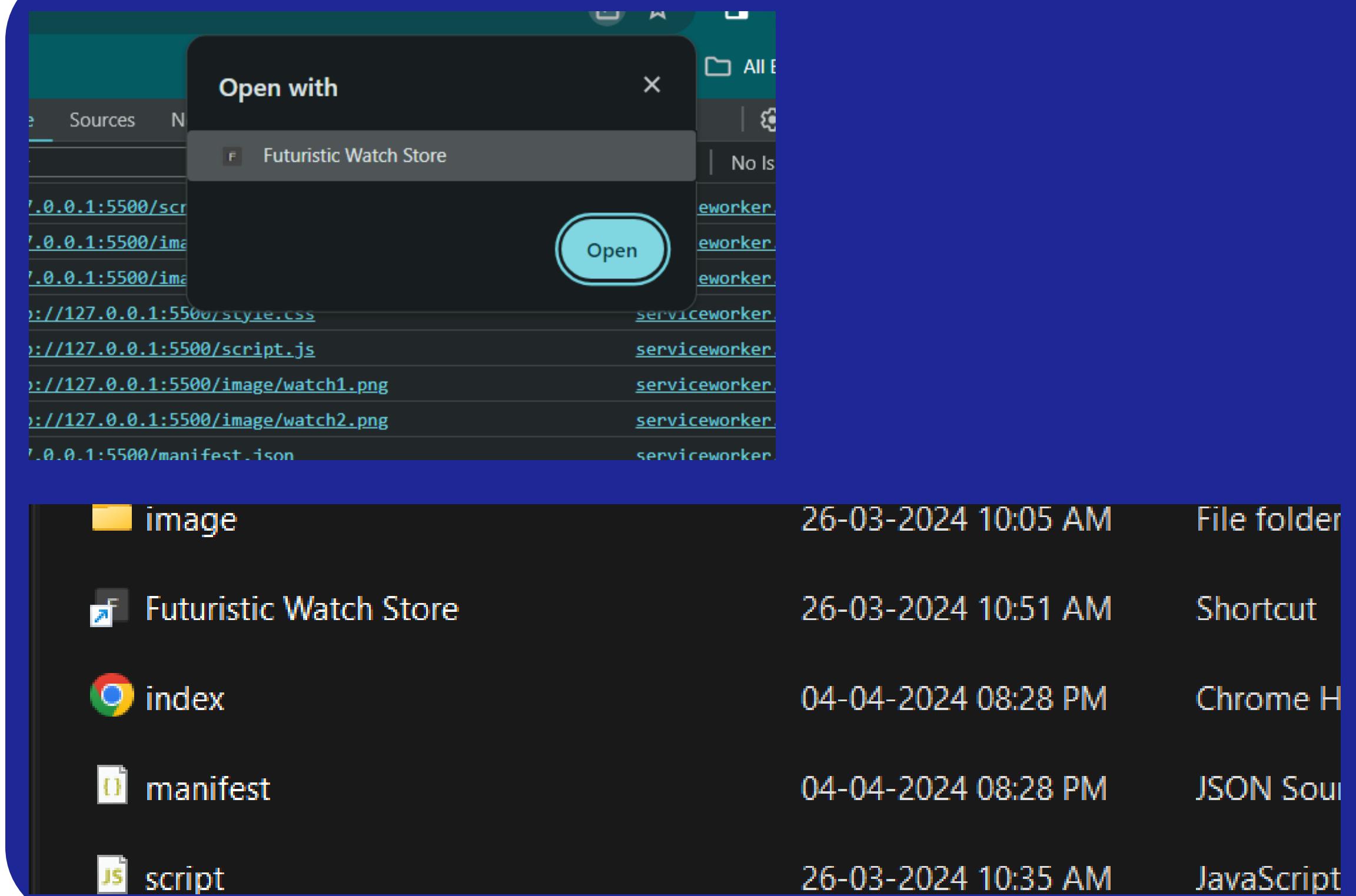
Default levels ▾ | No Issues

PushEvent {isTrusted: true, data: PushMessageData, type: 'push', target: ServiceWorkerGlobalScope, currentTarget: ServiceWorkerGlobalScope, ...}

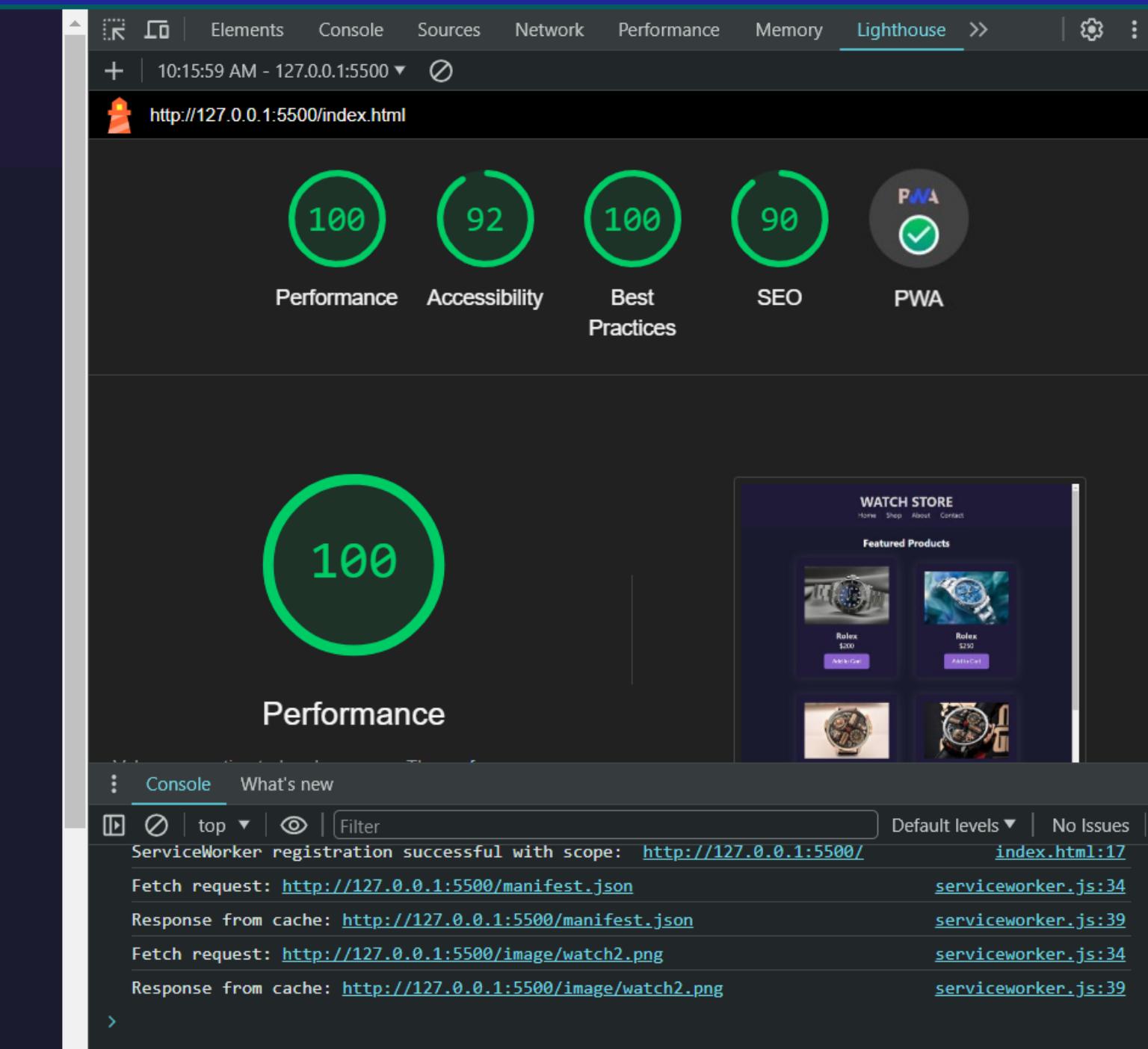
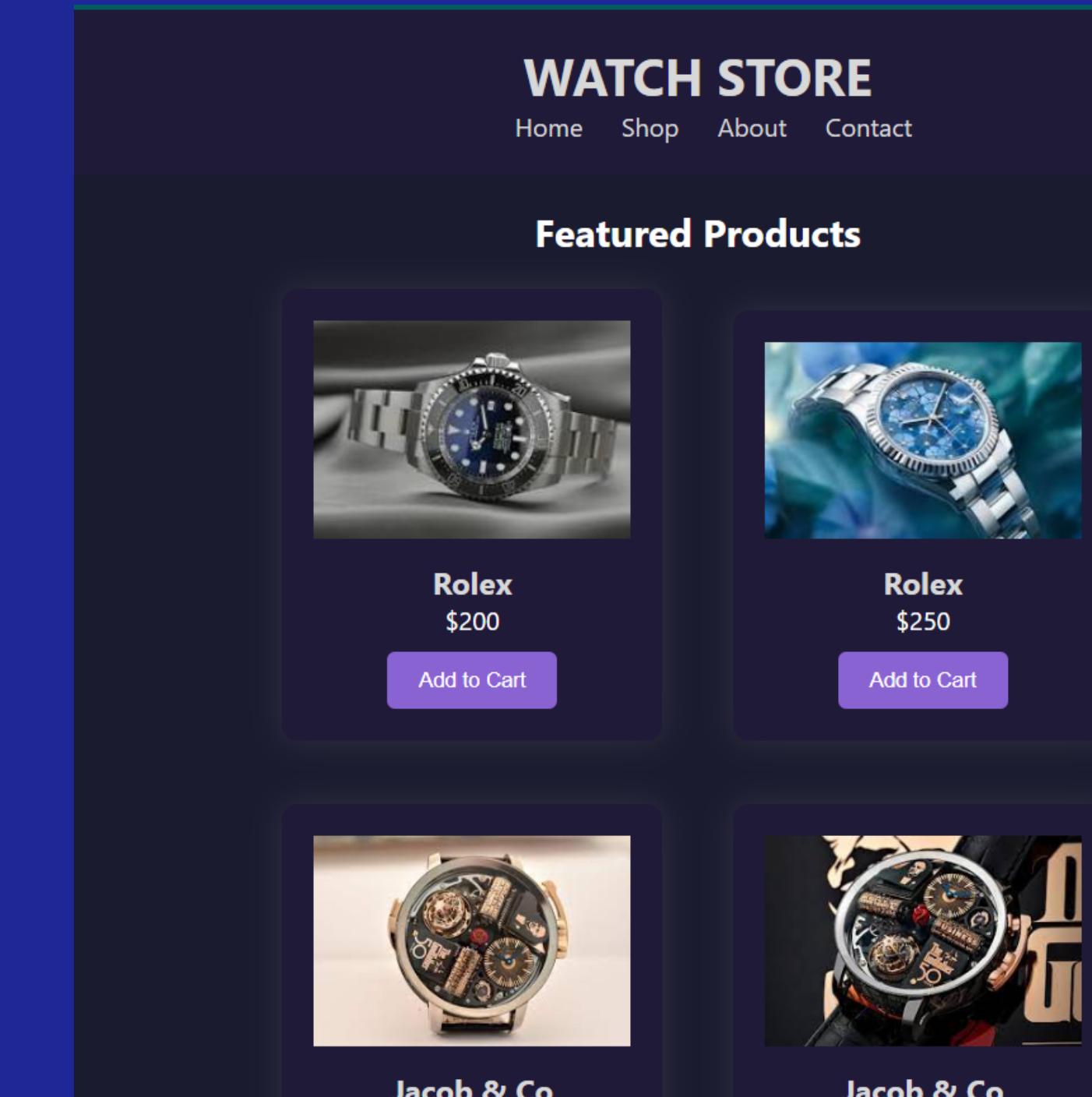
Uncaught (in promise) TypeError: Failed to execute 'showNotification' on 'ServiceWorkerRegistration': No notification permission has been granted for this origin.
at serviceworker.js:67:27

Sync event triggered [serviceworker.js:73](#)

Implementation



Implementation



Thank You