

Project Part 4

PART 1 Unit Test: Home Page Rendering Test

The screenshot shows the VS Code interface with the following details:

- File Explorer:** Shows the project structure under "COMP229Project_NutritionTracker". The "frontend" folder contains "src" (with "components", "pages", "utils", "hooks", "constants", "types", and "index.ts"), "tests" (with "Home.test.ts"), and "api.js".
- Editor:** Displays a Jest test file "Home.test.ts" for the "Home" component. The test checks if the heading "Your Nutrition Tracker" is rendered correctly.
- Terminal:** Shows the command "npm test" being run in the root directory of the project. The output shows the test passed, indicating the heading was rendered correctly.
- Output:** Shows the results of the test, including the number of tests passed (1), snapshots (0), and time taken (6.444 s).

For Part I of the assignment, I wrote a unit test using **Jest** and **React Testing Library** to verify that the Home page renders the main heading correctly. The test checks whether the text “**Your Nutrition Tracker**” appears when the Home component is rendered inside a React Router environment.

The test file is located in:

src/_tests_/Home.test.js

The test uses:

- render() from React Testing Library
 - MemoryRouter to simulate routing
 - A mock API module to avoid real network requests

When running the command:

npm test

The results show:

- **Test Suites: 1 passed**
 - **Tests: 1 passed**
 - No failures or warnings
 - Jest successfully executed the unit test and confirmed that the component renders the expected heading.

This confirms that the Home page UI loads correctly, and the component behavior works as intended.

Project Part 4

Deployment using render: BEFORE I CLICKED DEPLOY

The screenshot shows the Render interface for creating a new web service. The top navigation bar includes 'My Workspace', 'New Web Service', 'Search', '+ New', 'Upgrade', and a user icon. The main section is titled 'New Web Service'.

Source Code: A recent commit from 'Mushfiq1 / MyNutritionTracker' is listed, made 2h ago.

Name: MyNutritionTracker

Language: Node

Branch: main

Region: Virginia (US East)

Root Directory (Optional): backend

Build Command: backend/ \$ npm install

Start Command: backend/ \$ npm start

Instance Type: Options include Pro (\$85/month), Pro Plus (\$175/month), Pro Max (\$225/month), and Pro Ultra (\$450/month). A note states: "Need a custom instance type? We support up to 512 GB RAM and 64 CPUs."

Environment Variables: Fields for MONGO_URI, JWT_SECRET, and PORT are shown, each with a redacted value. Buttons for '+ Add Environment Variable' and 'Add from env' are available.

Advanced: A collapsed section.

Deploy Web Service: The primary button at the bottom.

Project Part 4

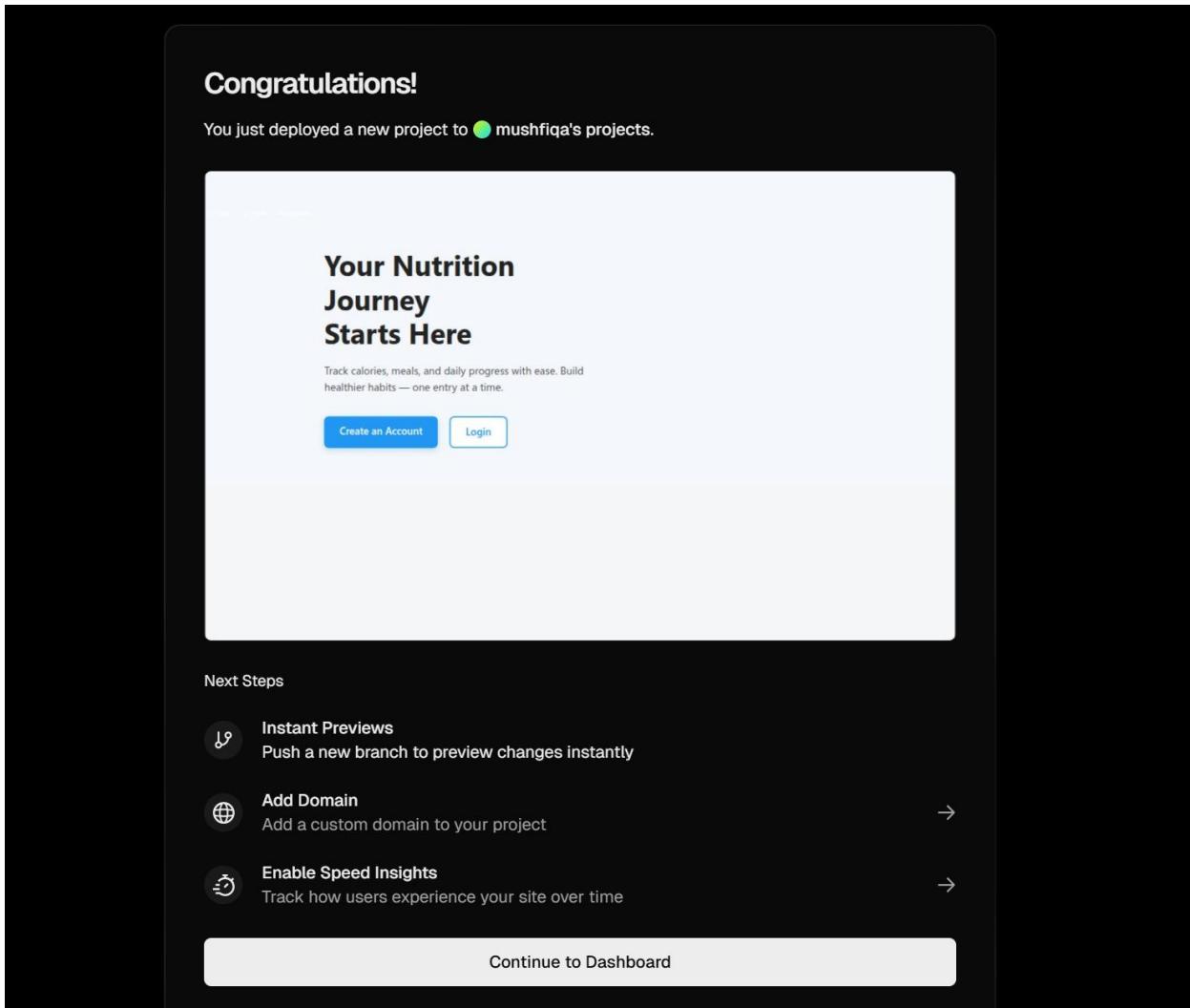
Confirmation of successful backend deployment on RENDER:

The screenshot shows the Render interface for the 'MyNutritionTracker' service. The service is listed as a 'WEB SERVICE' running on Node.js and is currently 'Free'. It has a 'main' instance with the URL <https://mynutritiontracker.onrender.com>. A message indicates that the free instance will spin down if inactive for 50 seconds. The log stream shows the deployment process starting at December 5, 2025, at 6:51 PM, with logs including 'Deploying...', 'Running 'npm start'', and 'Your service is live'. The logs also mention MongoDB connection and the primary URL.

Deployment of frontend on versal:

The screenshot shows the Vercel 'New Project' interface. It starts with an 'Importing from GitHub' step, showing the repository 'Mushfiqal/MyNutritionTracker' with the main branch selected. The next step is to choose a project name, which is set to 'my-nutrition-tracker'. Under 'Framework Preset', 'Create React App' is selected. The 'Root Directory' is set to 'frontend/mynutritiontracker-client'. In the 'Build and Output Settings' section, there is a note about environment variables. The 'Environment Variables' section shows a single entry for 'REACT_APP_API_URL' with the value 'https://mynutritiontracker.onrender.com'. A button to 'Import .env' is available. At the bottom, a progress bar indicates the deployment status.

Project Part 4



Link to frontend deployment: <https://my-nutrition-tracker.vercel.app/>

Project Part 4

TESTING SCREENSHOTS:

Create Your Account

Start tracking your meals, calories, and progress today.

First Name

Last Name

Email Address

Password

[Create Account](#)

Already have an account? [Login here](#)

Logging in after signing up:

The screenshot shows a web browser window with the URL "my-nutrition-tracker.vercel.app/home". The title bar says "MyNutritionTracker". The page has a blue header with "MyNutritionTracker" and navigation links for "Tracker", "Profile", and "Log". Below the header, there's a section titled "Your Nutrition Tracker" with a "+ Add Entry" button. Three cards show statistics: "Total Calories Today" (0), "Meals Logged" (0), and "Avg Calories / Meal" (0). A section titled "Your Entries" with the message "No entries yet. Start by adding a meal!" is also visible.

This screenshot shows the same "Your Nutrition Tracker" page as the previous one, but with different content. It displays a success message: "Account created successfully! You can now log in with the email and password you provided." Below this, it says "Welcome, JerryPhillip" and provides a "Log In" link. The rest of the page, including the stats cards and entry section, is identical to the previous screenshot.

Project Part 4

Adding entry:

Food Name
Chicken Breast

Calories
400

Protein (g) Carbs (g) Fats (g)
30 22 8

+ Add Entry

Entry is added to Dashboard:

Total Calories Today
400

Meals Logged
1

Avg Calories / Meal
400

Your Entries

Chicken Breast
Calories: 400
Protein: g
Carbs: 22g
Fats: g
12/5/2025

Edit Delete

Updating entry:

Project Part 4

Edit Nutrition Entry

Food Name: Chicken LEG

Calories: 400

Protein (g): 15

Carbs (g): 22

Fats (g):

Update Entry

Updated entry:

MyNutritionTracker

Tracker Profile Log Out

Your Nutrition Tracker

+ Add Entry

Total Calories Today: 400

Meals Logged: 1

Avg Calories / Meal: 400

Your Entries

Chicken leg

Calories: 400

Protein: 15g

Carbs: 22g

Fats: g

12/5/2025

Edit **Delete**

Project Part 4

E2e TEST:

The screenshot shows the Cypress E2e test runner interface. On the left, the test file `e2e/login.cy.js` is open, displaying a single passing test named `passes`. The test body contains the command `visit https://example.cypress.io`. On the right, a browser window shows the `cypress.io` website with the "Kitchen Sink" example application loaded. The test has passed, indicated by a green checkmark icon in the test runner's status bar.

E2e test on deployed app:

The screenshot shows the Cypress E2e test runner interface. On the left, the test file `login.cy.js` is open, showing a test named `should load the login page`. The test body includes code to visit the login page, assert the presence of the "Welcome Back" title, scroll into view, and assert the visibility of the login button. On the right, a browser window shows the deployed `my-nutrition-tracker` application's login page. The page features a "Welcome Back" header, an email input field, a password input field, and a "Login" button. A link for creating a new account is visible at the bottom.

Project Part 4

LIGHTHOUSE PREFORMANCE LEVELS: BEFORE

Your Nutrition Journey Starts Here

Track calories, meals, and daily progress with ease. Build healthier habits — one entry at a time.

Create an Account Login

Performance: 91 Accessibility: 89 Best Practices: 100 SEO: 100

91
Performance

FCP
LCP
TBT
SI
CLS

Values are estimated and may vary. The performance score is calculated directly from these metrics. See calculator.

Console All assistance What's new

No messages No user messages No errors No warnings No info No verbose

Performance Improvement Suggestions:

1. Removed unused console logs

Reduced unnecessary console statements in frontend components to speed up JavaScript execution.

2. Minimized image sizes

Ensured the logo and UI images are optimized for faster loading.

3. Enabled React lazy loading for pages

Used React.lazy() to load non-critical pages only when needed, improving initial load speed.

4. Optimized API calls

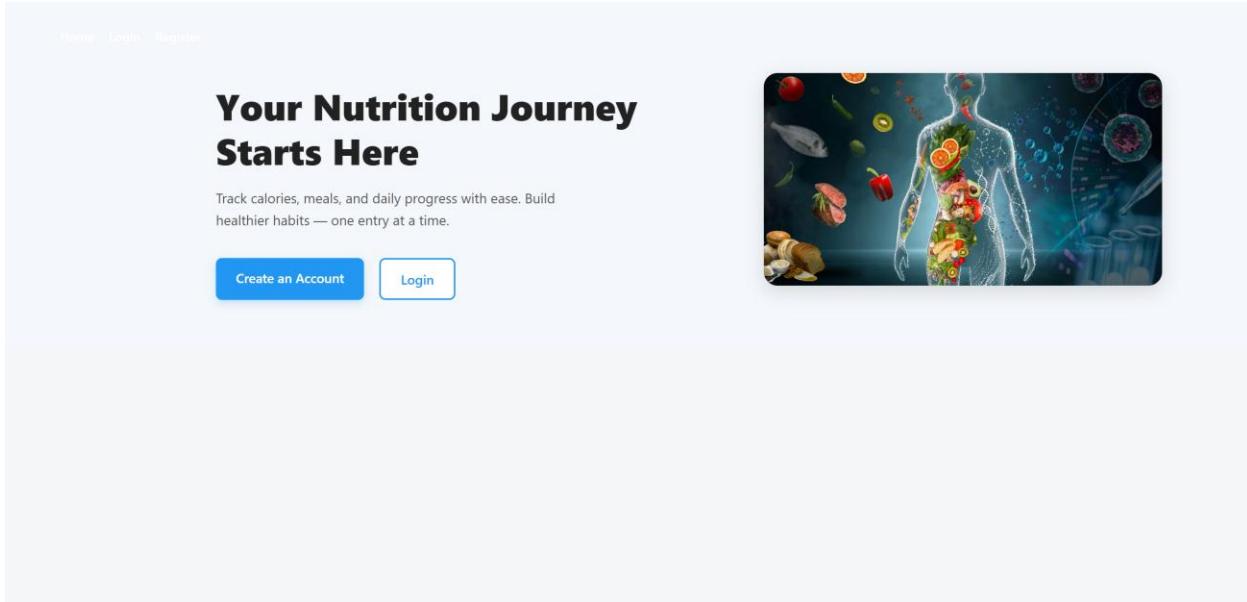
Made sure only required data is fetched on load and removed redundant API calls.

5. Minimized CSS and JS bundle size

Ensured unused imports were removed to reduce total bundle size.

BEFORE changes:

Project Part 4



AFTER changes: CI/CD Demonstration

A screenshot of a GitHub pull request interface for the repository "Mushfiqat / MyNutritionTracker". The title of the pull request is "CI/CD Demonstration Update". The description area contains the text: "Added small paragraph to Dashboard.js to demonstrate CI/CD pipeline." A Copilot review summary is visible on the right, stating: "Copilot can review pull requests. Request a review from Copilot to get fast, actionable feedback on your code, so you can start iterating before you receive a human review." The bottom of the screen shows standard GitHub commit and file change statistics: "1 commit", "34 files changed", and "1 contributor".

Project Part 4

Deployment Details

Created: 2m ago | Status: Ready | Duration: 43s | Environment: Preview

Domains: my-nutrition-tracker-git-update-ci-demo-mushfiqas-projects.vercel.app

Source: update-ci-demo

Deployment Settings: 3 Recommendations

Build Logs: 3 | 43s

Deployment Summary: Resources

Deployment Checks: Skipped

Runtime Logs: View and debug runtime logs & errors

Observability: Monitor app health & performance

Speed Insights: Not Enabled

Web Analytics: Not Enabled

Your Nutrition Journey Starts Here

Track calories, meals, and daily progress with ease. Build healthier habits — one entry at a time.

This project was updated as part of the CI/CD demonstration.

Create an Account | Login

CI/CD Demonstration Summary

To demonstrate CI/CD, I made a small update to my project by adding a new paragraph on the Dashboard page. I created a new branch (update-ci-demo), committed the change, and pushed it to GitHub. Then, I opened a Pull Request and merged it into the main branch.

Once the merge happened, Vercel automatically detected the update, rebuilt the project, and deployed a new production version without me needing to manually redeploy anything. After the deployment finished, I visited my live website and verified that the new paragraph appeared, confirming that the CI/CD pipeline was working successfully.

Project Part 4

LINKS:

LINK TO GITHUB: <https://github.com/Mushfiqa1/MyNutritionTracker.git>

LINK TO DEPLOYMENT ON VERSAL: <http://my-nutrition-tracker-e5i9rxrjb-mushfiqas-projects.vercel.app/>