Requirement	Test inputs	Expected outputs	Description	Valid/invalid
Sign up	Register with valid inputs Full Name: “Tineprah” Email: “tine@example.com” Password: “Strong@Pass1” Confirm: “Strong@Pass1”	Account is created, user is redirected to homepage, and logged in	Confirms successful registration flow with valid and secure inputs and correct matching password confirmation.	v
	Passwords do not match: Password: “Strong@Pass1” , Confirm: “Strong@Pass2”	Flash message: “Both passwords do not match”	Tests confirmation logic for verifying both password fields match exactly before account creation.	i
	Weak password (too short): Password: “123abc”	Flash message: “Passwords must be at least 8 characters…”	Ensures the password strength policy is enforced (e.g., minimum length, uppercase, non-letter characters).	i
	Weak password (missing special chars): Password: “Password1”	Flash message about missing non-alphabetic characters	Checks that the password meets the requirement for 2 non-letter characters.	i
	Reuse of existing email or username: Email or username already in use	Flash: “email or username is already in use”	Validates that the system rejects registration when either the email or username already exists in the database.	i
	Missing required field (email): Leave email blank	Browser validation triggers error No request sent	Ensures required HTML5 validations prevent form submission when required fields are empty.	i
	Enter invalid email format: Email: “tine-at-email.com”	Browser triggers native validation (HTML5)	Checks front-end HTML5 validation for correct email structure (@ and domain suffix).	i
	Successful redirect after registration: Submit valid details	User redirected to /homepage	Confirms post-registration navigation flow and login session activation.	v
	Already logged-in user visits register: Authenticated user accesses /register	Redirected to homepage	Prevents access to registration page when already logged in.	v
	Link to Login Page works: Click “Already have an account? Sign In”	Redirects to /login	Verifies navigation to the login page is available for returning users.	v
Sign In	Sign in with valid credentials: Email: tine@example.com Password: Correct@Pass1	Redirected to homepage, login session begins	Tests the standard successful login flow where credentials are correct and account exists.	v
	Incorrect email or password: Email exists, password is wrong OR vice versa	Flash: “Invalid credentials!”, page remains on login	Verifies login protection by rejecting incorrect combinations and notifying the user.	i
	Missing email input: Email: (blank) Password: “SomePass1!”	Browser validation prevents form submission	Ensures front-end HTML5 validation blocks submission of empty required fields.	i
	Missing password input: Email: “user@example.com” Password: (blank)	Browser validation prevents submission	Tests the required password field works properly with required attribute.	i
	Invalid email format: Email: “user-at-email.com”	Browser validation error	Confirms HTML5 validation checks for valid email format before form submission.	i
	Session started after login: Submit valid credentials	User remains authenticated across protected pages	Validates session persistence and that the user remains logged in after successful login.	v
	Guest button behavior: Click “Continue as Guest” button	Redirected to homepage without authentication	Ensures guest access route works, skipping login validation but limiting access to protected features.	v
	Click “Create an Account” link: Click “Create an Account” below the form	Redirects to /register	Tests that users can easily switch to the registration page from login if they don’t have an account.	v
				v
Log Notes	Save a new note: Valid title + valid note body Title: “Workout Recap”, Note: “Did 30 mins of cardio today.”	Success Status: Note saved successfully, appears in “Saved Notes”	Verifies that notes with both a valid title and content are correctly stored and displayed to the user.	v
	Special characters input: Title: “ Mind Reset”, Note: “@today’s mood: relaxed #zen”	Success Status: Note is saved and displays emojis/symbols correctly	Tests that the system can handle and safely display special characters, emojis, and symbols without rendering issues or errors.	v
	Keyword search functionality: Enter “cardio” in the search field	Success Status: Displays notes that contain “cardio” in title or content	Validates partial matching logic in the search bar and ensures that the filtered notes correspond to the search term.	v
	Search with empty query: Leave the search field blank, click “Search”	Success Status: Returns all notes or does nothing	Confirms whether the app supports listing all saved notes when the search field is empty, useful for reset or catch-all queries.	v
	Show all notes button: Click “Show All Notes” without search filters	Success Status: All saved notes are displayed	Ensures the “Show All Notes” button successfully triggers a full list of saved notes regardless of any prior filter or search.	v
	Title missing Title: (empty), Note: “Feeling better after meditation.”	Error message displayed: “Title cannot be empty.”	Checks frontend validation and backend constraints to ensure the title is a required field before saving a note.	i
	Note content missing Title: “Morning Log”, Note: (empty)	Error message displayed: “Note cannot be empty.”	Ensures that the note body cannot be left blank and must be filled before submission is allowed.	i
	Title & note missing Title: (empty), Note: (empty)	Two error messages: “Title cannot be empty.”, “Note cannot be empty.”	Confirms that both input fields are validated individually and together when empty.	i
	Delete a saved note: Click on a saved note entry	Note is removed from the Saved Notes section	Confirms the user can delete a previously saved note, and the note disappears from the visible list.	v
Set Reminder	Add valid reminder: Activity: "Stretching", Time: "07:30"	Success Status: Reminder is saved and appears in the list	Tests that a valid activity with a selected time is stored properly in local storage and rendered in the UI.	v
	Reminder triggers popup: Time matches a saved reminder	Success Status: Popup and sound appear (also system notification)	Ensures real-time checks successfully detect time match and trigger the reminder alert, sound, and popup.	v
	Reminder triggers sound: Time matches a saved reminder	Success Status: Sound plays once popup appears	Checks if reminder sound effect loads, plays, and does not repeat unnecessarily.	v
	Reminder triggers system notification: Browser has permission, Reminder matches time	Success Status: System-level push notification appears	Confirms that the Notification API works and respects granted permissions.	v
	Deny notification access: Browser notification permission: "denied"	Success Status: No system notification shown	Ensures app fails gracefully and continues to work if user declines push notification permission.	v
	Input special characters: Activity: " Yoga & Stretch !", Time: "18:00"	Success Status: Reminder saved and displayed properly	Verifies encoding and rendering of emoji, special characters, and punctuation in user inputs.	v

	Missing activity input: Activity: (blank), Time: "10:00"	Alert message: “Please enter an activity…”	Ensures the app blocks saving when the activity field is empty.	i
	Missing time input: Activity: "Jogging", Time: (blank)	Alert message: “Please select a time…”	Tests that the time field is mandatory before allowing a reminder to be created.	i
	No input at all Activity: (blank),Time: (blank)	Alert message triggered	Validates that both fields are required for submission.	i
	Delete a reminder: Click next to a listed reminder	Reminder is removed from list and localStorage	Ensures reminders can be removed from both the UI and stored data.	v
Log Exercise	Log predefined exercise: Type: Cardio Duration: 40	Exercise is saved and appears in “Logged Exercises” section Bar chart updates with new entry Personalized recommendation (e.g., “Great for heart health!”, "Try to reach 30 minutes") appears below	Tests complete success flow: log entry saved, chart updated with duration, and relevant motivational advice shown.	v
	Log custom exercise: Type: Custom Custom: “Boxing” Duration: 20	Exercise saved in “Logged Exercises” Bar chart updates Personalized message shown ("Great custom session!")	Confirms that user-defined custom entries are processed and visualized with general motivational feedback.	v
	Track progress (Logging mutiple entries): Log several valid exercises ("Running" 30, "Cycling" 20, etc.)	All exercises visible Bar chart reflects multiple entries Suggestions tailored per type (e.g., "Builds muscle")	Verifies ability to handle multiple logs, update the chart with each, and adapt suggestions dynamically.	v
	Show personalized tips: Type: HIIT Duration: 20	“Great fat burner!” displayed below	Confirms personalized suggestions change dynamically based on logged workout type.	v
	Custom name is missing: Type: Custom, Custom: (empty), Duration: 20	Alert: “Please complete all required fields.”	Ensures users are prompted when choosing “Custom” but leaving the name blank.	i
	Custom selected + "Push,Ups"	Alert: "Please enter a valid exercise name"	Ensuring all symbols are rejected	i
	Custom selected + empty field	Alert: "Please complete all required fields"	Validating error handling when custom is selected but left blank	i
	Custom selected + "123"	Alert: "Please enter a valid exercise name"	Ensuring purely numeric input is rejected in custom exercise type	i
	Custom selected + "Kickboxing123"	Alert: "Please enter a valid exercise name"	Testing alphanumeric rejection if only letters are allowed	i
	Custom selected + "Lift$"	Alert: "Please enter a valid exercise name"	Ensuring special characters are rejected	i
	Missing duration input: Type: Stretching, Duration: (empty)	Alert: “Enter a correct time.”	Confirms validation for the required duration field	i
	Zero duration: Type: Strength, Duration: 0	Alert: “Enter a correct time.”	Rejects entries with zero time, ensuring positive number constraint.	i
	Negative duration: Type: Pilates, Duration: -5	Alert: “Enter a correct time.”	Prevents logging of exercises with negative durations.	i
	Non-numeric duration: Type: Yoga, Duration: “thirty”	Alert: “Enter a correct time.”	Ensures that duration must be a number.	i
	Press submit with no inputs	Message "Please fill in this field."	Ensuring user prompts when fields are not filled	i
	Track progress on valid logs: Submit multiple logs with valid durations	Bar chart appears with new entries	Verifies dynamic updates to the Chart.js progress chart as exercises are logged.	v
	Delete an entry manually: Click on any Logged Exercise	Exercise entry is removed from UI	Confirms UI deletion works as expected. (No database persistence tested in this frontend setup.)	v
Log Stress	Log valid stress entry: Stress Level: 3, Cause: “Exams”, Notes: “Busy week ahead.”	Entry saved and displayed under Logged Stress Entries Personalized tips shown	Confirms a full and valid submission results in a saved entry, form reset, suggestions update, and smooth scroll to entry list.	v
	Submit without selecting level: No stress level selected Cause filled	Alert: “Please select a stress level.”	Tests required field validation for selecting stress level using the emoji scale.	i
	Submit with empty cause: Stress level selected Cause field left blank	Alert: “Please enter a stress cause.”	Ensures that users must provide a cause to contextualize their stress level.	i
	Submit with invalid cause: Stress level: 4, Cause: “123@#$”	Alert: “Please enter a valid stress cause (letters only).”	Tests input validation for alphabetic-only causes (no numbers/symbols).	i
	Submit with only optional notes: Notes field filled No stress level or cause	Alert: “Please select a stress level.”	Ensures that optional fields alone are not sufficient for submission.	i
	Form resets after submission: Submit valid log	Form fields clear after submission	Confirms UI resets form after successful logging.	v
	View logged entries: Log 2–3 entries	Entries appear in chronological order under “Logged Stress Entries”	Verifies stress entries are stored, rendered, and displayed correctly	v
	Display relaxation tips: Submit level 4 stress	Breathe Flow link + 2 tips shown (e.g., “Try a breathing exercise”)	Confirms recommendation engine delivers accurate, dynamic suggestions based on latest stress level.	v
	Breathe Flow link appears (level ≥ 3): Submit level 3, 4, or 5	One tip includes link to Breathe Flow	Tests the conditional logic that triggers a relaxation activity link only for moderate to high stress levels.	v
	Personalized tips update on resubmit: Submit stress level 2, then later 5	Old tips replaced with new tips for level 5	Verifies the relaxation recommendation section refreshes based on most recent entry.	v
	Delete a stress entry: Click delete button on an entry	Entry removed from UI with smooth transition	Validates deletion flow with fade-out effect, fetch call, and UI update.	v
Play Breathe Flow Game	Start inhale when spacebar is pressed: Press and hold Spacebar	Orb grows, feedback shows “Inhale... Hold it...”	Verifies the inhale sequence starts correctly, orb animation triggers, and inhale feedback is displayed.	v
	Show exhale prompt after 10 seconds: Hold Spacebar for ≥10, but don’t release	Feedback updates to “Exhale now (release spacebar)”	Ensures exhale guidance is triggered only after the defined minimum hold time	v
	Earn calm points correctly: Hold Spacebar for ≥10s, then release	Calm Points increase by 10 Feedback: “ Great job!”	Confirms correct breathing earns points, updates bar fill, and gives proper textual reward.	v
	Reject short/early exhale: Press and release Spacebar before 10s	Feedback: “ Too quick! Wait for the exhale signal.” Bar reverses slightly	Tests that early releases are flagged and do not reward calm points, and the calm bar visually reverses progress.	v
	Track cumulative progress: Repeat valid breathing cycles	Calm Points increase in steps of 10 until 50	Ensures multiple correct rounds update progress and score cumulatively, up to a maximum.	v
	Trigger game completion: Calm Points reach 50	“ Great Job!” message appears Final score shown	Validates the “Game Over” screen appears at 50 points and displays the user’s final score.	v
	Restart the game: Click “Try Again” after Game Over	Score resets to 0, bar empties, instructions reset	Checks reset logic restores default game state for replay.	v
	Orb animation starts/stops correctly: Toggle Spacebar press and release	Orb visually grows and shrinks correctly	Confirms visual breathing orb is synced with user input (press = grow, release = shrink).	v
	Bar fill animates during inhale: Hold Spacebar for a full breath	Calm fill bar animates smoothly during inhale	Tests real-time animation of the calm progress bar while holding breath.	v
	Non-spacebar input does nothing: Press any key other than Spacebar	No visual or scoring reaction	Confirms that only the Spacebar activates gameplay, ensuring clean input handling.	i

register user name:	Logged-in user with username = “Tineprah”	username = “Tineprah”Greeting text shows: “Good Morning Tineprah”	Validates dynamic greeting with time-based message (“Morning”, “Afternoon”, “Evening”) + username display.	v
Log out	Click Logout button (user signs out), Flash popup appears	Message appears in top banner and fades after 3 seconds	Tests that feedback messages (e.g., success alerts) are shown to users and automatically dismissed to avoid clutter.	v
Toggle menu sidebar:	Click hamburger icon (☰ Menu)	Sidebar expands/collapses And display other pages to visit; Dashboard, Stress, Exercise, Breathe, Notes, etc.	Verifies that the sidebar menu toggles visibility as expected using JavaScript, improving navigation accessibility.	v
Navigate via dashboard card (exercise):	Click “Track Exercise” button	Redirects to /exercise page	Confirms that dashboard buttons are correctly linked to the Exercise Tracking feature.	v
Navigate via dashboard card (Stress Tracking)	Click “Track Stress” button	User is redirected to /stress page	Ensures the “Track Stress” dashboard card is functioning correctly and leads to the intended route.	v
Navigate via dashboard card (Breathe Flow)	Click “Breathe Flow” button	User is redirected to /breathe_flow page	Validates navigation to the breathing activity feature via the dashboard.	v
Navigate via dashboard card (My Note)	Click “My Note” button	User is redirected to /notes page	Tests correct redirection to the journaling/notes feature from the dashboard card.	v
Navigate footer links:	Load the page normally	Footer displays all app navigation links: Dashboard, Stress, Exercise, Breathe, Notes, etc.	Checks that all essential links are visible and correctly placed in the footer for quick access to other parts of the application.	v
Log in as a Guest	Load homepage with no user logged in	Greeting shows “Good Morning” without username	Ensures that the system gracefully handles the case when no user is logged in, by hiding personalized name while still greeting generically.	v