

# Full Stack Development Practical -

## Experiment 1

---

Aarya Santosh Thorat (Roll No. 71)

Sanket Patil (Roll No. 57)

### Aim:

Building Responsive UI with Tailwind CSS

### 1. Prerequisite

Before starting this experiment, the following tools and technologies are required:

- Visual Studio Code (VS Code)
- Node.js installed with npm
- Tailwind CSS setup
- Basic knowledge of HTML, CSS, and JavaScript
- Live Server extension in VS Code to preview project output

### 2. Theory

Tailwind CSS is a utility-first CSS framework that provides predefined classes to rapidly design responsive UIs. Unlike Bootstrap, which comes with prebuilt components, Tailwind allows developers to create custom designs by combining small utility classes directly in HTML. It is highly customizable and supports responsive design, dark mode, animations, and Just-In-Time compilation.

In this experiment, we designed a responsive wellness application called 'Serenity'. The application provides a calming UI with sections like Breathing Exercises, Mood Tracker, Wellness Dashboard, and Wellness Activities. The project showcases how Tailwind can be used to create an aesthetic and functional UI.

**30% Extra:** We also integrated animations such as breathing circle effects, floating emojis, progress dashboards, and dynamic quotes to make the project more interactive and visually appealing. In addition to building a **responsive user interface** with Tailwind CSS, we implemented several **advanced UI/UX features** to make the project interactive, visually appealing, and engaging for users. These enhancements include:

1. **Breathing Circle Animation**

- A circular element expands and contracts smoothly using CSS keyframe animations.
- It visually guides the user through breathing exercises by syncing the animation with inhale/hold/exhale phases.
- Tailwind classes were combined with custom CSS (@keyframes breathe) to achieve this.
- This creates a **calm, meditative effect**, improving user engagement in the wellness activity.

## 2. Floating Emojis Effect

- Emojis (🧘, 🌿, 🌙) in the **Wellness Activities Grid** gently float up and down.
- Implemented using a **float keyframe animation** that shifts elements vertically.
- Adds a **playful and soothing touch** to the UI, making it feel more alive instead of static.

## 3. Progress Dashboard with Animated Counters

- The **Weekly Progress Dashboard** shows:
  - Meditation streak (days)
  - Mindful minutes
  - Mood average
  - Wellness score
- Progress rings and counters animate dynamically when the page loads.
- For example, `animateCounter()` gradually increases numbers (e.g., 0 → 156 mindful minutes), simulating real progress tracking.
- This creates an illusion of a **live, data-driven dashboard**, motivating users to track their wellness journey.

## 4. Dynamic Daily Quotes

- Each time the page loads, a **random inspirational quote** is displayed.
- Quotes are stored in an array, and JavaScript picks one using `Math.random()`.

- This ensures the app feels **fresh and personalized** on each visit, reinforcing mindfulness.

## 5. Ripple Effect on Buttons

- Buttons such as *Start Session* and *Stop* include a **ripple animation effect** when hovered.
- Adds tactile feedback, enhancing interactivity and making the UI more polished.

## 6. Glassmorphism Design

- Certain sections (Breathing Exercise, Mood Tracker, Activity Cards) use a **glass effect** (blurred, translucent background).
- Achieved using backdrop-filter: blur(10px); and semi-transparent borders.
- This provides a **modern, aesthetic UI style** commonly seen in professional apps.

## 3. Code

Below is the screenshot of the VS Code project folder:

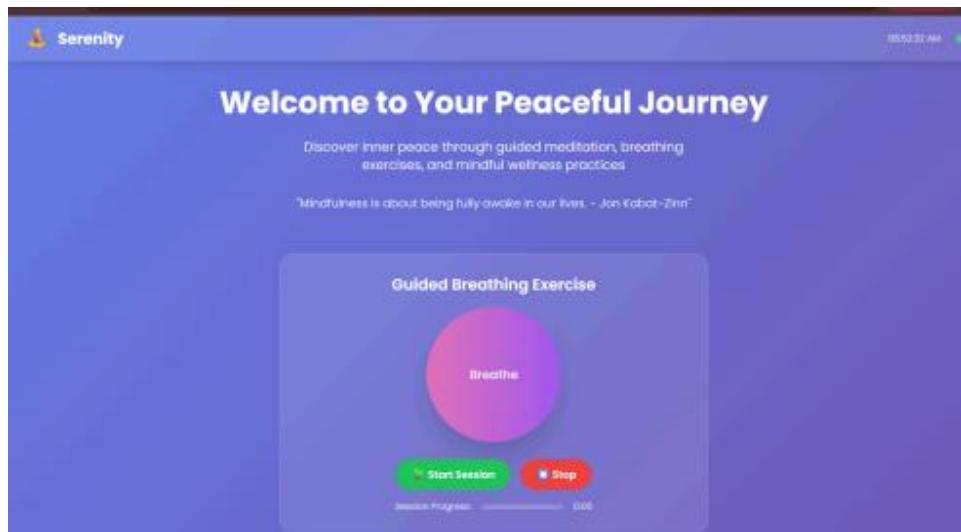


Fig 1: VS Code Project Folder Structure

The main HTML code for the project is shown below:

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8" />
<meta name="viewport" content="width=device-width, initial-scale=1.0" />
<title>Serenity - Find Your Inner Peace</title>
<script src="https://cdn.tailwindcss.com"></script>
<link href="https://fonts.googleapis.com/css2?family=Poppins:wght@300,400,500,600,700&display=swap"
      rel="stylesheet">
<style>
.font-primary {
  font-family: 'Poppins', sans-serif;
}

.breathing-circle {
  animation: breathe 4s ease-in-out infinite;
}

@keyframes breathe {
  0%, 100% {
    transform: scale(1);
    opacity: 0.8;
  }
  50% {
    transform: scale(1.2);
    opacity: 1;
  }
}

.floating {
  animation: float 6s ease-in-out infinite;
}

@keyframes float {
  0%, 100% { transform: translateY(0px); }
  50% { transform: translateY(-20px); }
}

.ripple-effect {
  position: relative;
  overflow: hidden;
}

.ripple-effect::before {
  content: '';
  position: absolute;
  top: 50%;
  left: 50%;
  width: 0;
  height: 0;
  background: rgba(255, 255, 255, 0.3);
  border-radius: 50%;
  transform: translate(-50%, -50%);
  transition: width 0.6s, height 0.6s;
}

.ripple-effect:hover::before {
  width: 300px;
  height: 300px;
}

.nature-bg {
  background: linear-gradient(135deg, #667eea 0%, #764ba2 100%);
}

.glass-effect {
  background: rgba(255, 255, 255, 0.1);
  backdrop-filter: blur(10px);
}
```

```

border: 1px solid rgba(255, 255, 255, 0.2);
}
.mood-tracker {
transition: all 0.3s ease;
}
.mood-tracker:hover {
transform: scale(1.1);
filter: brightness(1.2);
}
.wave-animation {
animation: wave 3s ease-in-out infinite;
}
@keyframes wave {
0%, 100% { transform: rotate(0deg); }
25% { transform: rotate(5deg); }
75% { transform: rotate(-5deg); }
}
.progress-ring {
transition: stroke-dasharray 0.5s ease-in-out;
}
</style>
</head>
<body class="font-primary nature-bg min-h-screen text-white">
<!-- Header --&gt;
&lt;header class="glass-effect shadow-lg"&gt;
&lt;nav class="container mx-auto px-6 py-4"&gt;
&lt;div class="flex items-center justify-between"&gt;
&lt;div class="flex items-center space-x-3"&gt;
&lt;div class="text-3xl wave-animation">□□□□□Serenity</h1>
</div>
<div class="flex items-center space-x-6">
<span class="text-sm opacity-80" id="currentTime"></span>
<div class="w-3 h-3 bg-green-400 rounded-full breathing-circle"></div>
</div>
</div>
</nav>
</header>
<main class="container mx-auto px-6 py-12">
<!-- Welcome Section --&gt;
&lt;section class="text-center mb-16"&gt;
&lt;h2 class="text-5xl font-bold mb-6 floating"&gt;
Welcome to Your Peaceful Journey
&lt;/h2&gt;
&lt;p class="text-xl opacity-90 mb-8 max-w-2xl mx-auto"&gt;
Discover inner peace through guided meditation, breathing exercises, and mindful
wellness
practices
&lt;/p&gt;
&lt;div class="text-lg opacity-80" id="dailyQuote"&gt;&lt;/div&gt;
&lt;/section&gt;
<!-- Interactive Breathing Exercise --&gt;
&lt;section class="max-w-2xl mx-auto mb-16 text-center"&gt;
&lt;div class="glass-effect rounded-3xl p-8 mb-8"&gt;
&lt;h3 class="text-2xl font-semibold mb-6"&gt;Guided Breathing Exercise&lt;/h3&gt;
<!-- Breathing Circle --&gt;
&lt;div class="relative mx-auto mb-8" style="width: 200px; height: 200px;"&gt;
&lt;div id="breathingCircle"
class="w-full h-full rounded-full bg-gradient-to-r from-pink-400 to-purple-500
breathing-circle flex items-center justify-center shadow-2xl"&gt;
&lt;span id="breathingText" class="text-white font-semibold text-lg"&gt;Breathe&lt;/span&gt;
&lt;/div&gt;
</pre>

```

```
</div>
<!-- Breathing Controls -->
<div class="space-y-4">
<div class="flex justify-center space-x-4">
<button id="startBreathing"
class="bg-green-500 hover:bg-green-600 px-6 py-3 rounded-full transition-all duration-300 ripple-effect font-semibold">
Start Session
</button>
<button id="stopBreathing"
class="bg-red-500 hover:bg-red-600 px-6 py-3 rounded-full transition-all duration-300 ripple-effect font-semibold">
Stop
</button>
</div>
<!-- Session Progress -->
<div class="flex items-center justify-center space-x-4">
<span class="text-sm opacity-80">Session Progress:</span>
<div class="w-32 h-2 bg-white bg-opacity-20 rounded-full">
<div id="sessionProgress" class="h-full bg-gradient-to-r from-green-400 to-blue-500 rounded-full transition-all duration-1000" style="width: 0%"></div>
</div>
<span id="sessionTime" class="text-sm opacity-80">0:00</span>
</div>
</div>
</div>
</section>
<!-- Daily Mood Tracker -->
<section class="mb-16">
<div class="glass-effect rounded-3xl p-8">
<h3 class="text-2xl font-semibold mb-6 text-center">How are you feeling today?</h3>
<div class="grid grid-cols-5 gap-4 max-w-lg mx-auto">
<div class="mood-tracker text-center cursor-pointer p-4 rounded-2xl" data-mood="amazing"
onclick="selectMood(this, 'amazing')">
<div class="text-4xl mb-2">😊</div>
<span class="text-sm font-medium">Amazing</span>
</div>
<div class="mood-tracker text-center cursor-pointer p-4 rounded-2xl" data-mood="happy"
onclick="selectMood(this, 'happy')">
<div class="text-4xl mb-2">😊</div>
<span class="text-sm font-medium">Happy</span>
</div>
<div class="mood-tracker text-center cursor-pointer p-4 rounded-2xl" data-mood="neutral"
onclick="selectMood(this, 'neutral')">
<div class="text-4xl mb-2">😐</div>
<span class="text-sm font-medium">Neutral</span>
</div>
<div class="mood-tracker text-center cursor-pointer p-4 rounded-2xl" data-mood="sad"
onclick="selectMood(this, 'sad')">
<div class="text-4xl mb-2">😢</div>
<span class="text-sm font-medium">Sad</span>
</div>
<div class="mood-tracker text-center cursor-pointer p-4 rounded-2xl" data-mood="stressed"
onclick="selectMood(this, 'stressed')">
<div class="text-4xl mb-2">😢</div>
<span class="text-sm font-medium">Stressed</span>
</div>
</div>
<div id="moodFeedback" class="mt-6 text-center opacity-0 transition-all duration-
```

```

500"></div>
</div>
</section>
<!-- Wellness Activities Grid -->
<section class="grid md:grid-cols-3 gap-8 mb-16">
<!-- Meditation -->
<div class="glass-effect rounded-3xl p-8 text-center hover:scale-105 transition-all duration-300 cursor-pointer" onclick="startActivity('meditation')">
<div class="text-6xl mb-4 floating">🧘 </div>
<h3 class="text-xl font-semibold mb-4">Guided Meditation</h3>
<p class="opacity-80 mb-4">Find your center with our peaceful meditation sessions</p>
<div class="bg-white bg-opacity-20 rounded-full px-4 py-2 text-sm">
5-30 minutes
</div>
</div>
<!-- Nature Sounds -->
<div class="glass-effect rounded-3xl p-8 text-center hover:scale-105 transition-all duration-300 cursor-pointer" onclick="startActivity('nature')">
<div class="text-6xl mb-4 floating" style="animation-delay: 1s;">🌿 </div>
<h3 class="text-xl font-semibold mb-4">Nature Sounds</h3>
<p class="opacity-80 mb-4">Relax with soothing sounds from nature</p>
<div class="bg-white bg-opacity-20 rounded-full px-4 py-2 text-sm">
Ambient Audio
</div>
</div>
<!-- Sleep Stories -->
<div class="glass-effect rounded-3xl p-8 text-center hover:scale-105 transition-all duration-300 cursor-pointer" onclick="startActivity('sleep')">
<div class="text-6xl mb-4 floating" style="animation-delay: 2s;">🌙 </div>
<h3 class="text-xl font-semibold mb-4">Sleep Stories</h3>
<p class="opacity-80 mb-4">Drift off with calming bedtime stories</p>
<div class="bg-white bg-opacity-20 rounded-full px-4 py-2 text-sm">
15-45 minutes
</div>
</div>
</section>
<!-- Weekly Progress Dashboard -->
<section class="glass-effect rounded-3xl p-8">
<h3 class="text-2xl font-semibold mb-8 text-center">Your Wellness Journey</h3>
<div class="grid md:grid-cols-4 gap-8">
<!-- Meditation Streak -->
<div class="text-center">
<div class="relative w-24 h-24 mx-auto mb-4">
<svg class="transform -rotate-90 w-24 h-24">
<circle cx="12" cy="12" r="10" stroke="rgba(255,255,255,0.2)" stroke-width="2" fill="none" class="w-full h-full"></circle>
<circle id="meditationProgress" cx="12" cy="12" r="10" stroke="#10B981" stroke-width="2" fill="none" stroke-dasharray="0 63" class="progress-ring w-full h-full"></circle>
</svg>
<div class="absolute inset-0 flex items-center justify-center">
<span class="text-xl font-bold" id="meditationDays">7</span>
</div>
</div>
<div class="text-sm opacity-80">Days Streak</div>
<div class="text-lg font-semibold text-green-400">Meditation</div>
</div>
<!-- Mindful Minutes -->

```

```

<div class="text-center">
<div class="text-3xl font-bold text-blue-400 mb-2" id="mindfulMinutes">156</div>
<div class="text-sm opacity-80">This Week</div>
<div class="text-lg font-semibold text-blue-400">Mindful Minutes</div>
</div>
<!-- Mood Average -->
<div class="text-center">
<div class="text-3xl mb-2">😊 </div>
<div class="text-sm opacity-80">Average Mood</div>
<div class="text-lg font-semibold text-yellow-400">Happy</div>
</div>
<!-- Wellness Score -->
<div class="text-center">
<div class="text-3xl font-bold text-purple-400 mb-2" id="wellnessScore">85</div>
<div class="text-sm opacity-80">Wellness Score</div>
<div class="text-lg font-semibold text-purple-400">Excellent</div>
</div>
</div>
</section>
</main>
<script>
// Update current time
function updateTime() {
const now = new Date();
document.getElementById('currentTime').textContent = now.toLocaleTimeString('en-US', {
hour: '2-digit',
minute: '2-digit',
second: '2-digit'
});
}
updateTime();
setInterval(updateTime, 1000);
// Daily inspirational quotes
const quotes = [
"Peace comes from within. Do not seek it without. - Buddha",
"The mind is everything. What you think you become. - Buddha",
"Wherever you are, be there totally. - Eckhart Tolle",
"The present moment is the only time over which we have dominion. - Thich Nhất Hạnh",
"Mindfulness is about being fully awake in our lives. - Jon Kabat-Zinn"
];
document.getElementById('dailyQuote').textContent = ''' +
quotes[Math.floor(Math.random() *
quotes.length)] + ''';
// Breathing Exercise Logic
let breathingSession = null;
let sessionStartTime = null;
let sessionDuration = 120; // 2 minutes in seconds
const breathingCircle = document.getElementById('breathingCircle');
const breathingText = document.getElementById('breathingText');
const startBtn = document.getElementById('startBreathing');
const stopBtn = document.getElementById('stopBreathing');
const sessionProgress = document.getElementById('sessionProgress');
const sessionTime = document.getElementById('sessionTime');
function startBreathingSession() {
sessionStartTime = Date.now();
let phase = 0; // 0: inhale, 1: hold, 2: exhale, 3: hold
let phaseTime = 0;
const phaseDurations = [4000, 1000, 6000, 1000]; // inhale, hold, exhale, hold
const phaseTexts = ['Inhale...', 'Hold...', 'Exhale...', 'Hold...'];
const phaseColors = [
'from-green-400 to-blue-500',
'from-yellow-400 to-orange-500',

```

```

'from-pink-400 to-purple-500',
'from-indigo-400 to-purple-600'
];
breathingSession = setInterval(() => {
const elapsed = Date.now() - sessionStartTime;
const progress = Math.min((elapsed / (sessionDuration * 1000)) * 100, 100);
// Update progress bar
sessionProgress.style.width = progress + '%';
sessionTime.textContent = Math.floor(elapsed / 1000 / 60) + ':' +
String(Math.floor((elapsed /
1000) % 60)).padStart(2, '0');
// Update breathing phase
phaseTime += 100;
if (phaseTime >= phaseDurations[phase]) {
phase = (phase + 1) % 4;
phaseTime = 0;
breathingText.textContent = phaseTexts[phase];
breathingCircle.className = `w-full h-full rounded-full bg-gradient-to-r
${phaseColors[phase]} breathing-circle flex items-center justify-center shadow-2xl`;
}
// End session after duration
if (elapsed >= sessionDuration * 1000) {
stopBreathingSession();
alert('🎉 Congratulations! You completed your breathing session. How do you feel?');
}
}, 100);
startBtn.disabled = true;
startBtn.classList.add('opacity-50');
}
function stopBreathingSession() {
if (breathingSession) {
clearInterval(breathingSession);
breathingSession = null;
sessionStartTime = null;
breathingText.textContent = 'Breathe';
breathingCircle.className = 'w-full h-full rounded-full bg-gradient-to-r from-pink-400
to-purple-500 breathing-circle flex items-center justify-center shadow-2xl';
sessionProgress.style.width = '0%';
sessionTime.textContent = '0:00';
startBtn.disabled = false;
startBtn.classList.remove('opacity-50');
}
}
startBtn.addEventListener('click', startBreathingSession);
stopBtn.addEventListener('click', stopBreathingSession);
// Mood Tracker
function selectMood(element, mood) {
// Remove previous selection
document.querySelectorAll('.mood-tracker').forEach(el => {
el.classList.remove('bg-white', 'bg-opacity-30', 'scale-110');
});
// Highlight selected mood
element.classList.add('bg-white', 'bg-opacity-30', 'scale-110');
// Show feedback
const feedback = document.getElementById('moodFeedback');
const responses = {
amazing: "🌟 That's wonderful! Keep riding this positive wave!",
happy: "😊 Great to hear! Happiness is contagious!",
neutral: "🌿 That's perfectly okay. Every day is a new opportunity.",
sad: "❤️ We're here for you. Consider trying a meditation session.",
stressed: "🌸 Take a deep breath. You've got this! Try our breathing exercise."
}

```

```

};

feedback.textContent = responses[mood];
feedback.classList.remove('opacity-0');
feedback.classList.add('opacity-100');
}
// Activity Starter
function startActivity(type) {
const activities = {
meditation: "瞑想 SESSION 🕉️ Starting your meditation session... Find a comfortable position and close your eyes.",
nature: "🌿 Playing nature sounds... Let the peaceful sounds of nature wash over you.",
sleep: "🌙 Beginning sleep story... Get comfortable and prepare for a restful journey."
};
alert(activities[type]);
}
// Animate progress rings and counters on load
window.addEventListener('load', function() {
// Animate meditation progress ring (7 days = ~78% of 9 days for visual appeal)
setTimeout(() => {
document.getElementById('meditationProgress').style.strokeDasharray = '49 63';
}, 1000);
// Animate counters
animateCounter('mindfulMinutes', 156, 2000);
animateCounter('wellnessScore', 85, 2500);
});
function animateCounter(elementId, targetValue, duration) {
const element = document.getElementById(elementId);
const increment = targetValue / (duration / 50);
let currentValue = 0;
const timer = setInterval(() => {
currentValue += increment;
if (currentValue >= targetValue) {
element.textContent = targetValue;
clearInterval(timer);
} else {
element.textContent = Math.floor(currentValue);
}
}, 50);
}
</script>
</body>
</html>

```

## 4. Output

### Step 1: Homepage UI with Guided Breathing Exercise

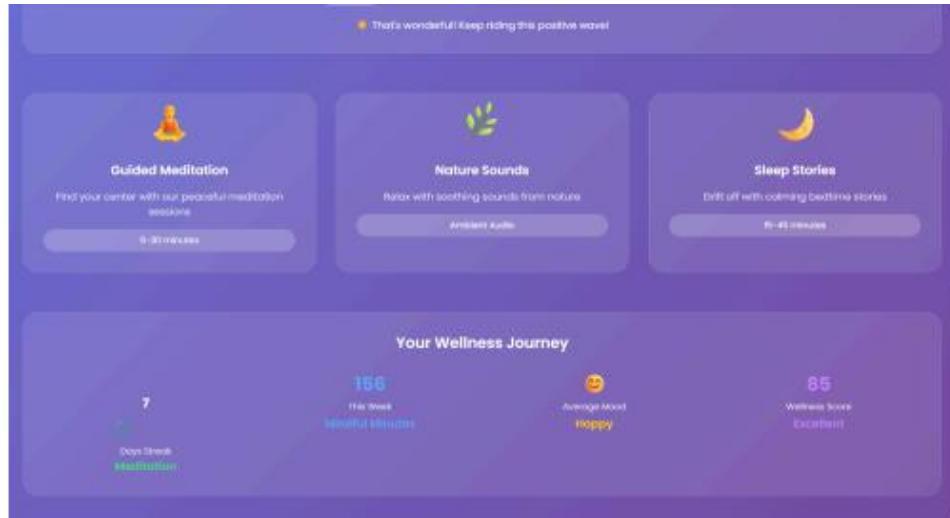


Fig 2: Serenity Homepage and Guided Breathing Exercise

### Step 2: Wellness Activities and Dashboard

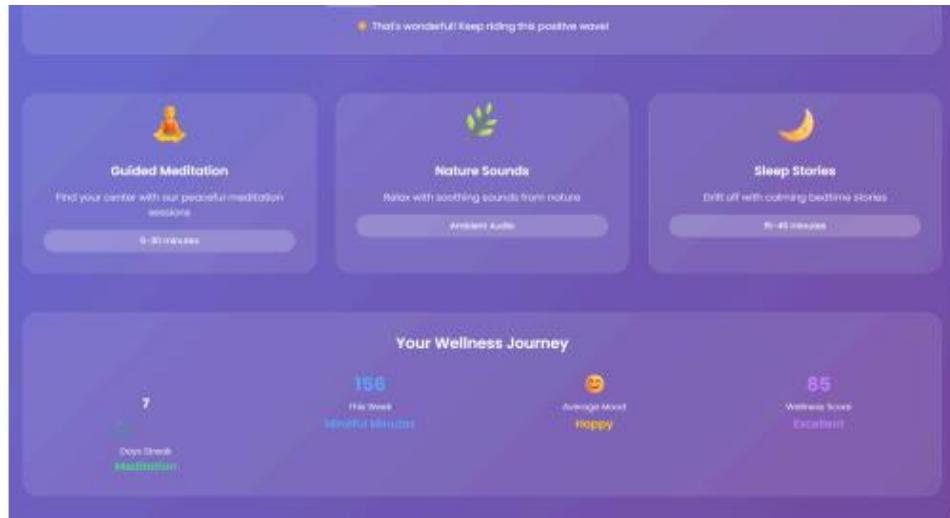


Fig 3: Serenity Wellness Activities and Dashboard

## 5. Conclusion

In this experiment, we successfully built a responsive and interactive UI using Tailwind CSS. The project 'Serenity' demonstrates how utility-first frameworks enable rapid prototyping and customization. By adding animations, dynamic quotes, and progress dashboards, we enhanced the user experience beyond the basic requirements, making the project more engaging and visually appealing.