

Front-End UI/UX Mini Project

Project Submission Template — Completed for “Weatherly: Responsive Weather Dashboard”

Project Title: Weatherly — Responsive Weather Dashboard (HTML & CSS Only)

Submitted By:

- **Team Member:** Arya Ramachandran (2460338)

Alex Jilson(2460321)

Joshua Thomas(2460386)

- **College E-mail ID:** arya.ramachandran@btech.christuniversity.in

alex.jilson@btech.christuniversity.in

Joshua.thomas@btech.christuniversity.in

Course: UI/UX Design Fundamentals

Instructor Name: Ms. Nagaveena

Institution: Christ (Deemed to be) University

Date of Submission: 13/08/2025

2. Abstract

This mini project designs and develops **Weatherly**, a responsive weather dashboard created **only with HTML and CSS**. The interface showcases multiple city cards (e.g., New York, Rome, Sydney) with temperature, conditions, hourly and daily forecasts, and supporting stats like humidity and wind. The key goal is to practice clean UI, consistent visual hierarchy, and mobile-first responsiveness without JavaScript. Core technologies include **HTML5 semantic structure, CSS3 (Flexbox, Grid-like layouts, media queries, and variables)**. The final outcome is a polished, accessible front-end that adapts across desktop, tablet, and mobile, demonstrating thoughtful layout, branding, and readability suitable for extension with real data later.

3. Objectives

- Design a **user-friendly interface** applying modern UI principles.
- Develop a **fully responsive** layout using **only HTML and CSS**.
- Implement **semantic HTML5** elements for structure and accessibility.
- Apply **CSS variables, Flexbox, and media queries** for theming and layout.
- Ensure **readability, contrast, and accessibility** across devices.

4. Scope of the Project

- Focused on **front-end UI** only; **no JavaScript** or server-side logic.
- Targets **desktop, tablet, and mobile** viewports (mobile-first).
- Uses **pure HTML/CSS** and open-source tooling (no UI libraries).
- Demonstrates **multi-city weather cards**, stats blocks, and a sidebar panel.
- **Out of scope:** live APIs, form submission, and data persistence.

5. Tools & Technologies Used

Tool/Technology	Purpose
HTML5	Markup, semantics, content structure
CSS3	Styling, layout, responsiveness, theming
VS Code	Authoring and project organization
Chrome DevTools	Responsive testing & debugging

6. HTML Structure Overview

- Semantic tags used: `<header>`, `<nav>`, `<main>`, `<article>`, `<section>`, `<aside>`, `<footer>`.
- Reusable content blocks: **City Weather Card**, **Stats Row**, **Hourly List**, **Daily List**, **Right-column Metric Panels**.
- Navigation as an unordered list within `<nav>`.
- Accessibility touches: `role="main"`, `aria-label` groups, labelled icon SVGs.
- **Branding**: inline logo within `.brand` and `.site-title`.

Observed structure highlights (based on provided code):

- **Header** with brand logo and simple nav.
- **Main hero** split into a left column (city cards + tips) and a right column (barometer/indoor panel).
- Each city is an `<article class="card">` containing icon, temperature row, stats, and forecast lists.
- **Footer** with attribution and privacy link.

7. CSS Styling Strategy

- External CSS file (e.g., `css/style.css`) with **CSS Variables** for colors, spacing, and typography.
- **Layout**: Flexbox for header, hero columns, card internals; simple column stacking via media queries.
- **Responsiveness**: Breakpoints at `~980px`, `480px`, etc., to reflow columns and scale type.

- **Visual Design:** Glassy panels, soft shadows, accent color for temperature units, muted secondary text.
- **Micro-interactions:** (Optional) hover emphasis on cards/links; transitions can be added if needed.

8. Key Features

Feature	Description
Responsive Design	Layout adapts from multi-column desktop to stacked mobile views.
Smooth Navigation	Simple top nav; anchors ready for section linking.
Project (City) Cards	Consistent cards with icon, temp, stats, hourly and daily forecasts.
Accessible Typography	High contrast palette, readable hierarchy.
Theme Variables	Color and spacing controlled via :root CSS variables.
Right-Column Panels	Barometer and indoor widgets as modular panels.

9. Challenges Faced & Solutions

Challenge	Solution
Overlapping/overflow on small screens	Introduced media queries to stack columns and reduce font sizes.
Alignment using older techniques	Standardized on Flexbox for alignment and spacing.
Typography scaling issues	Switched to relative units and responsive font sizes.
Duplicate CSS rule blocks for .card	Consolidated repeated .card definitions into one source-of-truth.
Repeated id="cityname" across multiple cards	Converted duplicate IDs to classes or unique IDs to maintain valid HTML.
Local absolute image path for logo (C:\...)	Replaced with project-relative asset path (e.g., assets/logo.jpg).

10. Outcome

- Achieved a **clean, consistent, and engaging** front-end layout.
- Verified **responsive behavior** across major breakpoints.
- Strengthened understanding of **semantic HTML, Flexbox,** and **mobile-first** design.
- Codebase ready for **future enhancement** with JavaScript and live data.

11. Future Enhancements

- Add JavaScript for **API integration, form validation,** and **dynamic units** toggle.
- Animate icons and transitions; introduce **reduced-motion** support.
- Add **dark/light** theme toggler via CSS variables + prefers-color-scheme.
- Improve **a11y** with keyboard focus styles and ARIA live regions for updates.
- Componentize repeated markup into **includes** (if templating) or **web components** later.

12. Sample Code

A) Header & Navigation (HTML)

```
<header>

  <div class="brand">

    <span class="logo" aria-hidden="true">

    </span>

    <div class="site-title">Weatherly</div>

  </div>

  <nav aria-label="Primary">

    <ul>

      <li><a href="#home">Home</a></li>

      <li><a href="#forecast">Forecast</a></li>

      <li><a href="#maps">Maps</a></li>

      <li><a href="#settings">Settings</a></li>

    </ul>

  </nav>

</header>
```

B) City Weather Card (HTML)

```
<article class="card" aria-labelledby="city-newyork">

  <svg class="icon" viewBox="0 0 64 64" aria-hidden="true">

    <circle cx="32" cy="28" r="10"></circle>

  </svg>

  <div class="info">

    <h2 id="city-newyork" class="city">NEW YORK</h2>

    <p class="time">Sunday • 12 Dec 2024 • 04:32 PM</p>

    <div class="temp-row" role="group" aria-label="current temperature">

      <div class="temp-value">23</div>
```

```

    <div class="temp-unit">°C</div>

</div>

<p class="desc">Clear sky</p>

<div class="stats" aria-label="current stats">

    <div class="stat"><span class="muted">Precip</span><span> 0%</span></div>

    <div class="stat"><span class="muted">Humidity</span><span> 81%</span></div>

    <div class="stat"><span class="muted">Wind</span><span> 5 km/h</span></div>

</div>

</div>

</article>

```

C) Card & Layout (CSS)

```

:root{

  --bg:#041821; --panel:#0b2a33; --accent:#122f6f; --muted:rgba(255,255,255,.75);

  --glass:rgba(255,255,255,.03); --radius:12px; --maxw:1100px; --pad:18px;

}

.page{max-width:var(--maxw);margin:24px auto;padding:24px;display:flex;flex-
direction:column;gap:20px}

.hero{display:flex;gap:20px;align-items:stretch}

.left-col{flex:1;display:flex;flex-direction:column;gap:16px}

.right-col{flex:0 0 380px}

.card{background:linear-
gradient(180deg,rgba(255,255,255,.03),rgba(255,255,255,.02));

  padding:20px;border-radius:var(--radius);box-shadow:0 10px 30px rgba(0,0,0,.35);

  display:flex;gap:18px;align-items:center}

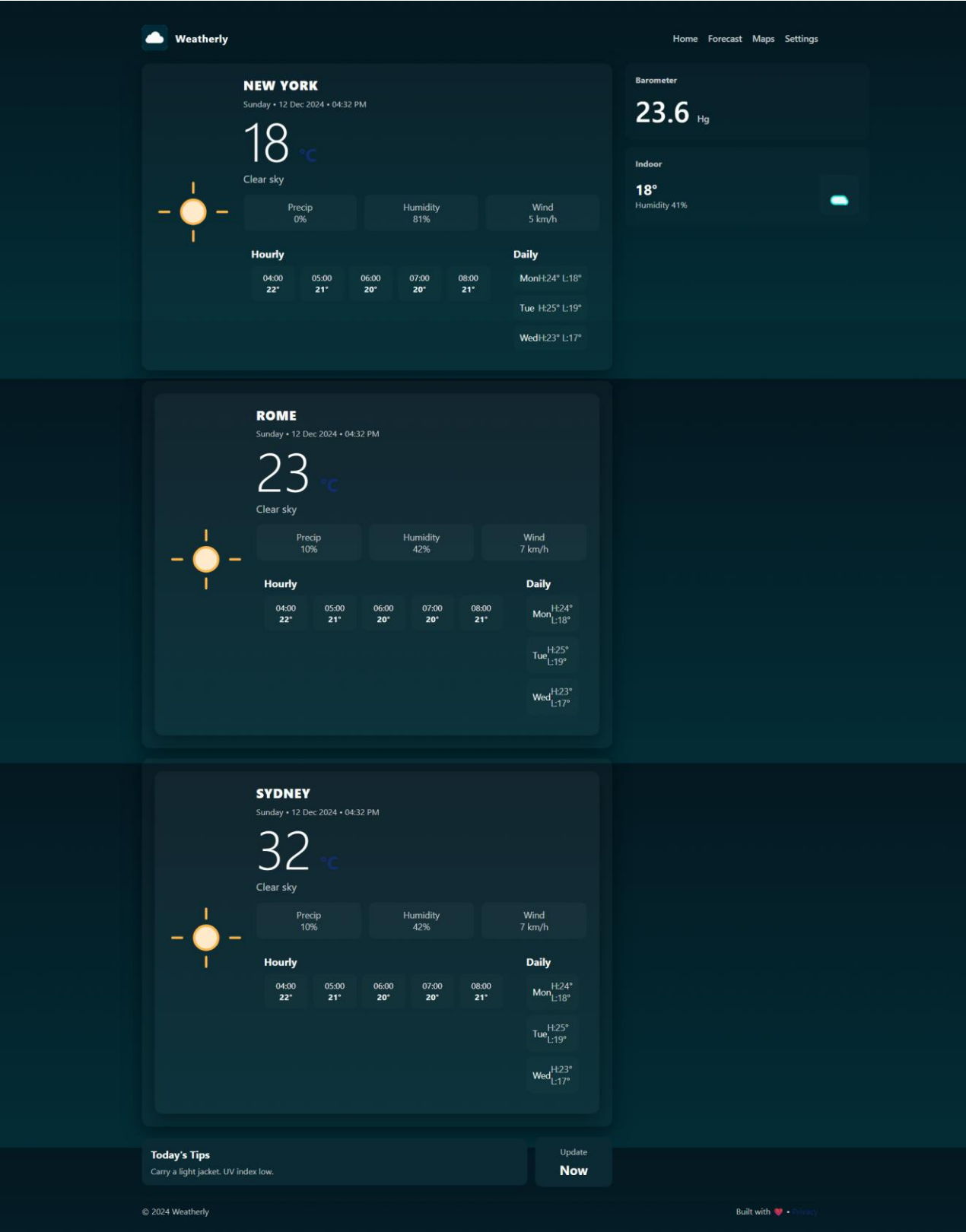
@media (max-width:980px){.hero{flex-direction:column}.right-col{flex:1}}

@media (max-width:480px){.card{flex-direction:column;align-items:flex-start}}

```

Note: Full project HTML and CSS have been provided separately; the above snippets illustrate structure and style organization for 2–3 screens.

13. Screenshots of Final Output



14. Conclusion

This **Weatherly** portfolio mini project showcases a responsive, semantic, and accessible front-end crafted solely with **HTML and CSS**. Building multi-city weather cards and modular panels strengthened practical skills in responsive layout, typographic hierarchy, and systemized theming with CSS variables. The deliverable is presentation-ready and primed for future enhancement with JavaScript, API data, and theme toggling.

15. References

- **L&T LMS:** <https://learn.intedutech.com/Landing/MyCourse>
- Mozilla Developer Network (MDN) — HTML & CSS references
- Web Content Accessibility Guidelines (WCAG) quick reference

Appendix: Validation & Code Quality Notes (Optional for Review)

- Consolidate duplicate .card CSS blocks into one.
- Replace absolute Windows image path with project-relative path.
- Avoid duplicate IDs (e.g., id="cityname"); use unique IDs or classes.
- Prefer semantic headings (<h1>–<h3>) inside cards for city names.
- Move inline styles to CSS for maintainability.
- Provide alt text for non-decorative SVGs or mark as aria-hidden="true" when decorative.
- Consider prefers-reduced-motion m