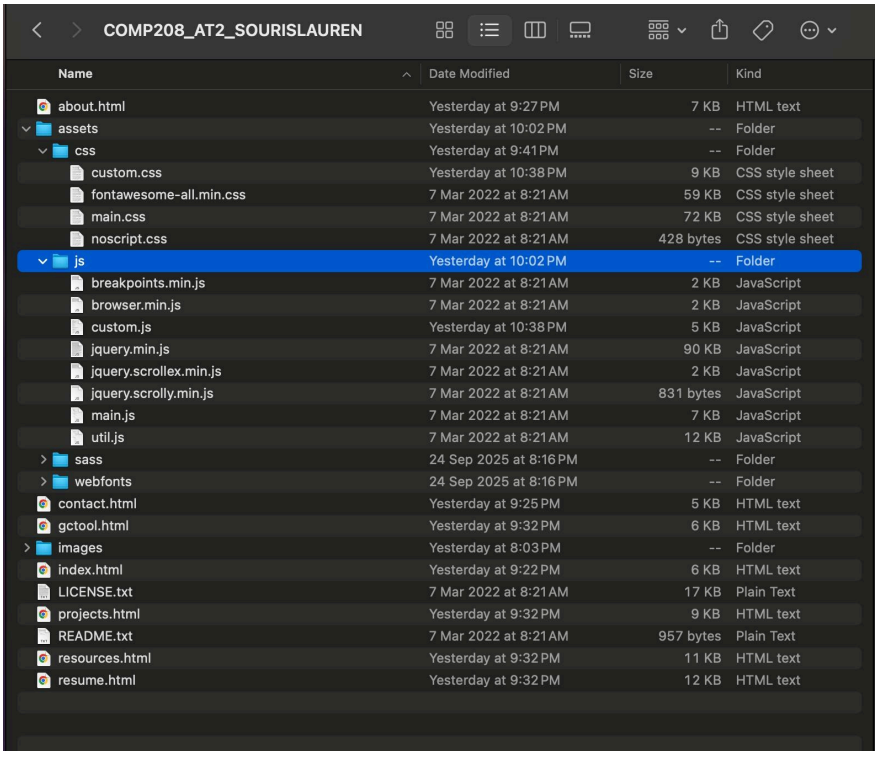
User Documentation: Website Overview

The website is a **responsive portfolio** for school leaders and educators to explore my professional identity and teaching resources, I have created throughout my university degree.
It demonstrates competence in HTML, CSS, and JavaScript through:

- Logical layout and consistent navigation.
- Accessible colour contrast and alt text for images.
- Three working JavaScript features that enhance user experience.

File and Folder Structure

The project follows a clear and logical structure with separate folders for HTML pages, CSS styling, and JavaScript functionality. This layout supports maintainability, readability, and professional version control practices.



Folder/File	Description
`index.html`	Home page featuring the **Quote Generator** interactive element.
`about.html`	Personal introduction and teaching philosophy section.
`projects.html`	Gallery of major TAS and Food Tech projects.
`resources.html`	Teaching materials and link to the **Grade Converter Tool**.
`gctool.html`	JavaScript-driven grade converter.
`resume.html`	Education, experience, and work history.
`contact.html`	Contact form with validation and popup confirmation.
`assets/css/custom.css`	Custom dark-theme design, colour variables, and layout styling.
`assets/js/custom.js`	Handles all three interactive JavaScript features.
`images/`	Contains all page images, banners, and resource previews.

HTML, CSS, and JavaScript Integration

Each page links the same CSS and JavaScript files:

```
<link rel="stylesheet" href="assets/css/custom.css">
<script src="assets/js/custom.js"></script>
```

The website demonstrates full integration between structure (HTML), design (CSS), and functionality (JavaScript). HTML defines the page layout and interactive elements, CSS controls visual styling and responsive layout, and JS enables dynamic features such as the Quote Generator, Grade Converter, and Contact Form popup. Together, they create a cohesive and user-centred web experience.

Below is a screenshot of 'About.html' with each HTML page linking to the same CSS and

```
<!DOCTYPE HTML>
<!--
  Forty by HTML5 UP
  Edited by Lauren Souris
-->
<html lang="en">
  <head>
    <title>About - Lauren Souris</title>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1, user-scalable=no" />

    <!-- Styles -->
    <link rel="stylesheet" href="assets/css/main.css" />
    <link rel="stylesheet" href="assets/css/custom.css" />
    <noscript><link rel="stylesheet" href="assets/css/noscript.css" /></noscript>
    <script defer src="assets/js/custom.js"></script>
  </head>
```

JavaScript files:

- **HTML** provides the semantic structure (sections, forms, buttons, headings).
- **CSS** defines the theme, typography, spacing, responsiveness, and animation.
- **JavaScript** adds interactivity and real-time feedback through event listeners and DOM manipulation.

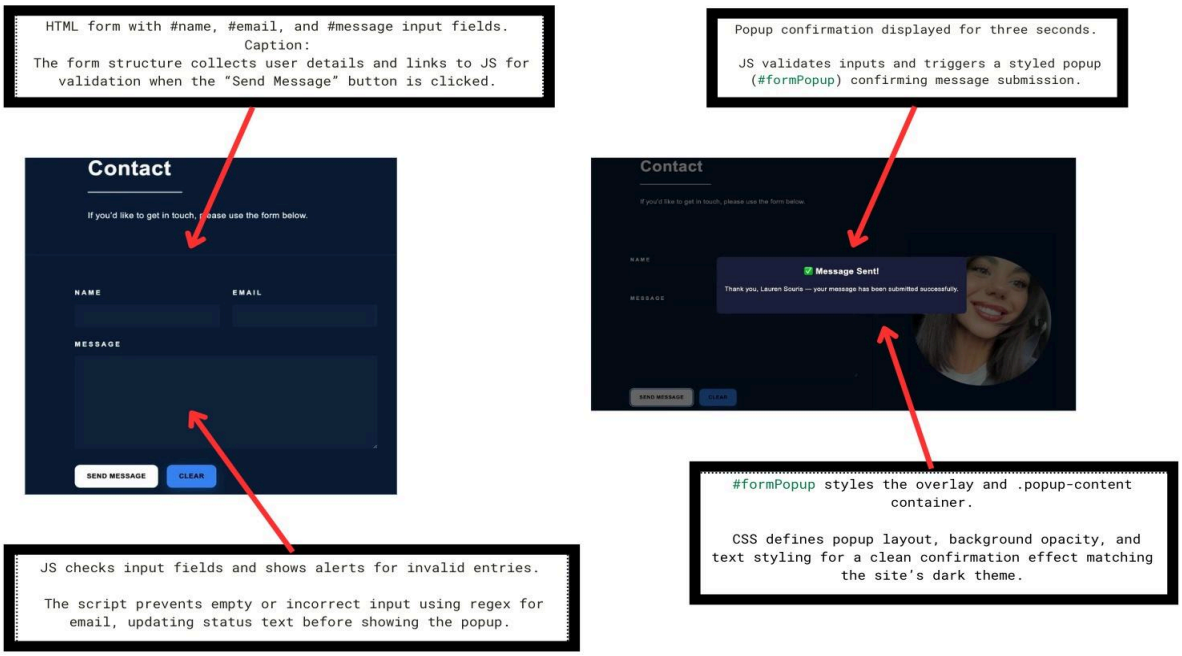
Interactive JS Feature 1: Contact Form with Popup Confirmation

Files: `contact.html`, `custom.css`, `custom.js`
Function: Validates input (name, email, message) and displays a popup confirmation for 3 seconds upon successful submission.

How it Works:

1. JavaScript listens for the form's `submit` event.
2. Checks for empty fields and valid email format.
3. If valid, shows `#formPopup` overlay (styled in CSS).
4. Popup disappears automatically or on click.

The Contact Form integrates HTML for structure, JavaScript for validation and user feedback, and CSS for a smooth popup confirmation. It enhances accessibility and ensures professional interaction consistency across the site.



Below Screeshoshot is the Code snippet in VS Code showing validation function in [custom.js]

```
// Basic validation
if (name.length < 2) return fail("Please enter your full name.");
if (!emailOK.test(email)) return fail("Please enter a valid email address.");
if (message.length < 10) return fail("Message should be at least 10 characters.");

if (status) status.textContent = "Message sent successfully ✅";
```

Integration Points:

HTML → Form elements

CSS → Popup styling (`#formPopup`, `.popup-content`)

JS → Event listener, validation logic, form reset

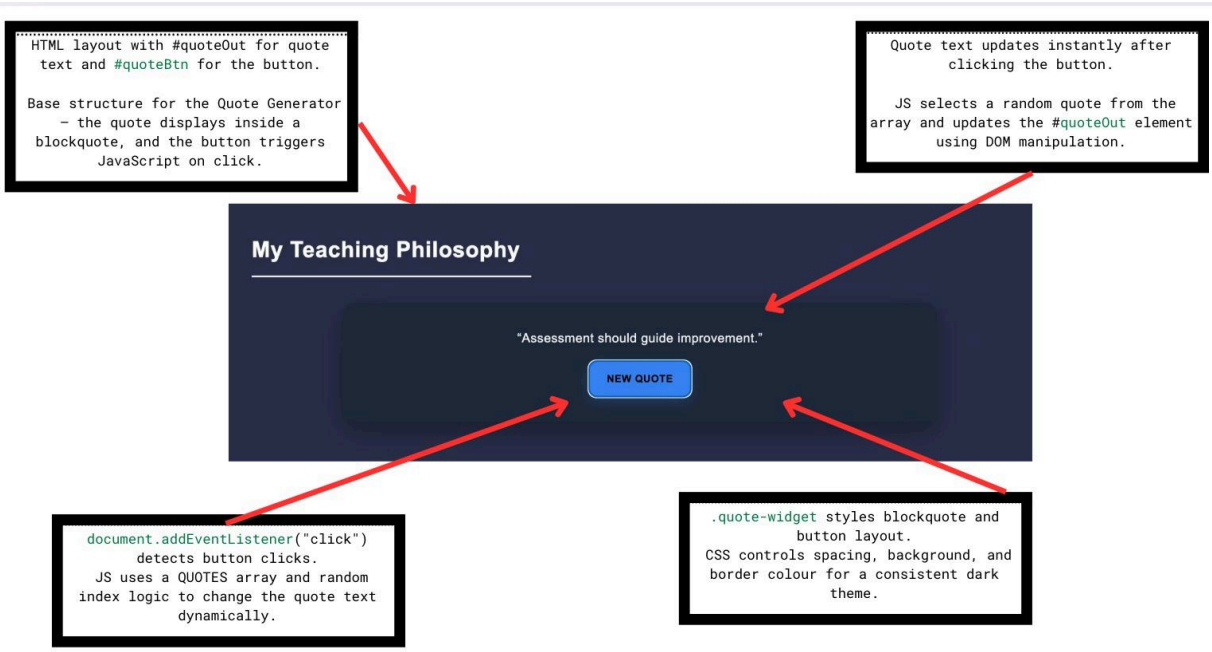
Interactive Feature 2: Quote Generator (Home Page)

Files: `index.html`, `custom.js`, `custom.css`
Function: Displays a random motivational teaching quote each time the user clicks the button.

How it Works:

1. Array of 10 quotes stored in JavaScript.
2. Click event on `#quoteBtn` selects a random index and updates the content of `#quoteOut`.
3. CSS styles the quote widget using `.quote-widget` for centred layout.

The Quote Generator uses JavaScript to select and display random motivational quotes from an array when the button is clicked. HTML provides the structure, CSS styles the `.quote-widget` for consistent layout and theme, and JS handles the interactivity, updating the text dynamically through DOM manipulation.



Integration Points:

HTML → `<blockquote id="quoteOut">` for output

CSS → `.quote-widget` and `.button` styling

JS → Event handler + randomiser logic

Interactive Feature 3: Grade Converter Tool

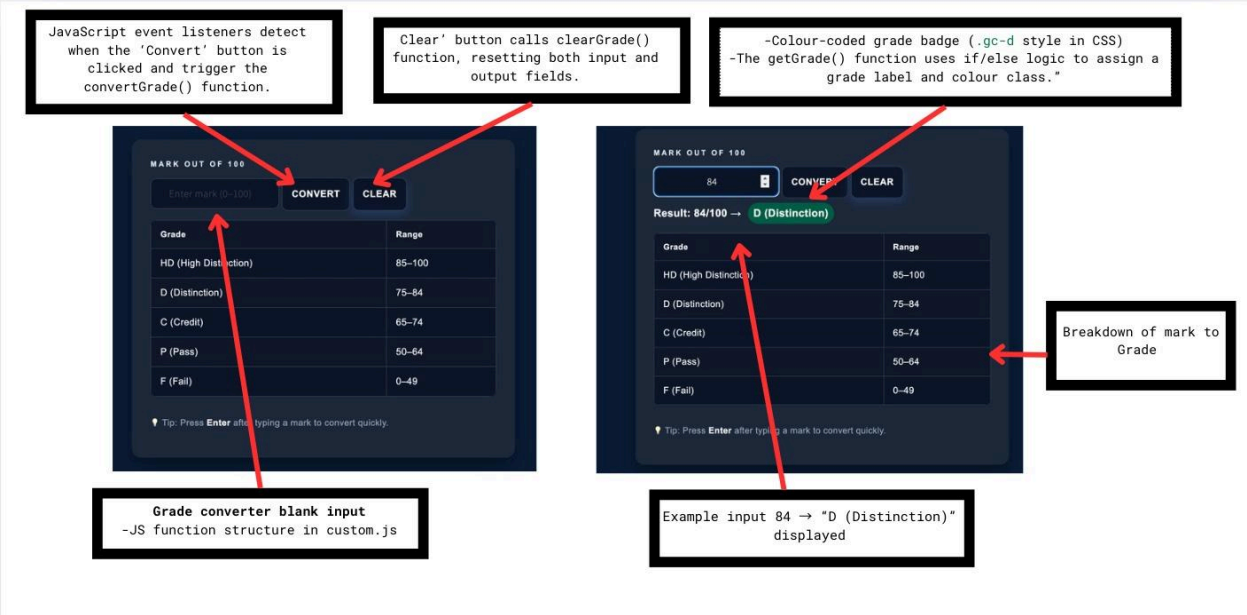
Files: `gctool.html`, `custom.js`, `custom.css`

Function: Converts a numeric mark (0 – 100) into an ACU-style grade (HD, D, C, P, F).

How it Works:

1. User enters mark into input field `#gcMark`.
2. Clicking "Convert" or pressing Enter triggers `convertGrade()` in JS.
3. Function determines grade range using `if` statements and displays formatted output in `#gcOut`.
4. "Clear" button resets input and output areas.

JavaScript listens for button clicks, validates user input, and updates the DOM with a dynamically generated grade result. CSS applies colour-coded badges for each grade category, ensuring clear visual feedback and consistency with the site's theme.



Integration Points:

HTML → Form inputs and buttons

CSS → `.gc-badge` classes (colour per grade)

JS → Conversion and event listener logic