AI Bio Lab Website – Multi-Page Responsive Lab Portal

This project is a modern, multi-page research lab website built entirely with Vanilla HTML, CSS, and JavaScript, designed to showcase the lab's mission, research areas, team members, and publications. It features a responsive layout using Flexbox and CSS Grid, interactive cards with hover transitions, and dynamic content powered by API calls to fetch research papers. Visitors can use the contact form, which saves submissions locally using localStorage and displays animated success messages. Shared components like a navbar and dynamic footer year unify the pages.

Page	Purpose	Main Features
Home (`index.html`)	Landing page; lab intro, research areas, team	Responsive Grid, Flexbox navbar, hover transitions, dynamic footer year
Projects (`projects.html`)	Displays research publications dynamically	API calls (CrossRef), dynamic JS content, Grid layout, responsive design
Contact (`contact.html`)	Collects visitor messages	LocalStorage, JS form handling, animated success message, responsive form

1️⃣Home Page – `index.html`

Purpose:
Introduces the research lab, its mission, research areas, and team members. Serves as the “landing page” for the website.

Sections:

1. Navbar
    - Contains lab name and links to all three pages.
    - Responsive using Flexbox.
    - Highlights the current page (`Home`) using `.active`.
2. Mission Statement
    - A short paragraph describing the lab’s vision and objectives.
    - Styled with CSS for spacing and typography.
3. Research Areas
    - A grid of 4 cards:
        - Cancer Genomics
        - AI Drug Discovery
        - Single-Cell Analysis
        - Translational Bioinformatics
    - Each card has:
        - An image representing the research area (local image or placeholder)
        - A title (`<h3>`)
        - A description paragraph
    - CSS Grid makes the layout responsive.
    - Cards have hover transitions (lift + shadow).
4. Team Members
    - A grid of 4 cards for lab members.
    - Each card includes:
        - Photo (`<img>` local or placeholder)
        - Name and role
    - Responsive layout using CSS Grid.
    - Cards include hover animations.
5. Footer
    - Shows the lab name and current year.
    - Year is dynamically set via JS.

Key Features Implemented Here:

- Flexbox (navbar)
- CSS Grid (research & team cards)
- CSS transitions on hover (cards)
- Responsive design (grid and navbar)
- CSS functions (`clamp()` for font sizes)
- Dynamic footer year (JS)

2️⃣ Projects Page – `projects.html`

Purpose:
Displays lab’s research outputs / publications dynamically using an API. Students can learn how to fetch data from a live source.

Sections:

1. Navbar
    - Same as Home page, highlights `Projects`.
2. Publication Search
    - Input field + search button to query papers by keyword.
    - Default search (`AI`) loads on page load.
3. Results Grid
    - Dynamically generated cards for each paper fetched from the CrossRef API:
        - Paper title (`<h3>`)
        - Author(s)
        - Year of publication
        - Clickable DOI link to the paper
    - CSS Grid layout for responsive display.
    - Cards can be styled with hover effects.

Key Features Implemented Here:

- API call using fetch() + async/await
- Responsive layout (CSS Grid)
- Dynamic JS content generation (createElement + innerHTML)
- Hover transitions on cards
- Navbar active state
- Footer year dynamic via JS

3️⃣ Contact Page – `contact.html`

Purpose:
Provides a form for visitors to contact the lab and demonstrates local storage and JS animations.

Sections:

1. Navbar
   - Same as other pages, highlights `Contact`.
2. Contact Form
   - Fields: Name, Email, Message
   - Checkbox: Subscribe to Newsletter
   - Submit button
3. Success Message
   - Animates in with CSS transitions when form is submitted
   - Message disappears automatically after a few seconds.
4. Local Storage
   - Saves form submissions to `localStorage` as JSON object (`key:value`)
   - Loads previous submission automatically if it exists, so user sees their last input.

Key Features Implemented Here:

- Form handling with JS
- Local storage (`localStorage.setItem` / `getItem`)
- CSS animation for success message (fade-in/out)
- Responsive layout (form adapts to smaller screens)
- Navbar active state

- Footer year dynamic via JS

Feature	Implementation (File / Section)	How it's implemented
Vanilla HTML/CSS/JS	All files (`index.html`, `projects.html`, `contact.html`, `style.css`, `script.js`)	No frameworks or libraries used; all layout, styling, and dynamic behavior handled with pure HTML, CSS, and JS.
Responsive Design	`style.css` (media queries)	Navbar and grid layout respond to screen width using: - `@media (max-width:768px)` for stacking navbar items - `grid-template-columns: repeat(auto-fit, minmax(220px,1fr))` for cards
Flexbox	`style.css` / `nav`	Navbar uses `display: flex; justify-content: space-between; align-items: center;` for proper alignment of logo + links
CSS Grid	`style.css` / `.grid`	Research areas and team members sections use `display: grid; grid-template-columns: repeat(auto-fit, minmax(220px,1fr));` for responsive cards layout
CSS Functions	`style.css` / headings	Headings use `font-size: clamp(1.5rem, 3vw, 2rem);` to scale font size based on screen width
Media Queries	`style.css` / `@media` block	Adjust navbar layout and spacing for screens smaller than 768px
API Call	`projects.html` / `<script>`	Fetch publications from **CrossRef API** using `fetch()` and

		`async/await`: `https://api.crossref.org/works?query=<search>&rows=6`
Local Storage (key:value)	`contact.html` / `<script>`	Contact form saves submission in `localStorage` with: `localStorage.setItem("contactFormData", JSON.stringify(contactData));` Loads previous submission with `localStorage.getItem("contactFormData")`
JS Animation (transition)	`style.css` + `contact.html` / success message	Success message fades in/out using: `.success-message { opacity: 0; transform: translateY(-10px); transition: all 0.4s ease; } .success-message.show { opacity: 1; transform: translateY(0); }`
CSS Transition	`style.css` / `.card`	Cards animate on hover using: `.card { transition: transform 0.3s ease, box-shadow 0.3s ease; } .card:hover { transform: translateY(-8px); box-shadow: 0 12px 25px rgba(0,0,0,0.1); }`
Dynamic Footer Year	`script.js` / all pages	JS sets current year dynamically: `document.getElementById("year").textContent = new`

		`Date().getFullYear();`
Default API load / search	`projects.html`	On page load, `searchPapers()` is called to display default papers for topic “AI”
Navbar Active Link Highlight	`index.html/projects.html/contact.html`	`class="active"` is added to the link of the current page
Team Members with Images	`index.html` / `.grid .card img`	Team members section uses `<img src="...">` with unique images and descriptions
Research Area Cards	`index.html` / `.grid .card`	Cards display image + title + descriptive paragraph; modern hover effect with transition