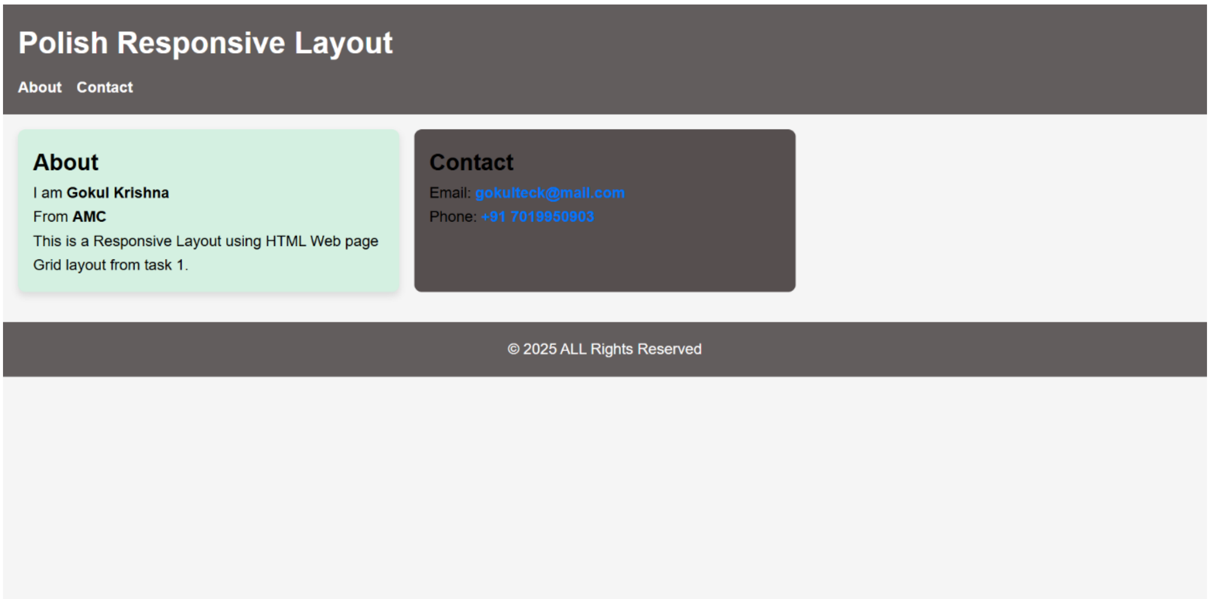
AeroAspire - SDE Intern
Gokul Krishna S

Week 1 – Day 5 (September 27)

Bonus task :

Reflect on week;
bonus tasks:
hover effects, mobile friendliness;

Polish Responsive Layout with Hover effect :



Polish Responsive Layout

About

I am **Gokul Krishna**

From **AMC**

This is a Responsive Layout using HTML Web page Grid layout from task 1.

Contact

Email: gokulteck@mail.com

Phone: +91 7019950903

© 2025 ALL Rights Reserved

Note :

- Click on the image ↑ to see the **Hover Effect**.

Hover Effect :

- Added smooth hover color and underline transitions on nav links and menu icon for better interactivity.
- Added subtle hover effect with background color and shadow on content sections.
- These improvements make the site feel polished, interactive, and mobile-friendly in line.

Questions/Reflections :

1. List top 3 challenges you faced this week and how you overcame them.

   Challenge 1: Structuring and Styling the Layout

   - At first, it was tricky to decide whether to rely more on Flexbox or Grid for positioning different sections. But in the end I decided to do both Flexbox and Grid Style Web page.

   Challenge 2: Adding JavaScript Interactivity

   - When I first tried creating a time-based greeting and a toggle button, the logic didn't always run as expected. I broke the problem into smaller pieces and used console.log() to debug step by step and took some help using chatgpt

   Challenge 3: Responsive design

   - Making the site look good on both desktop and mobile was frustrating, but media queries step by step helped.

2. If you were building this for a client, what extra polish or refinements would you add?

   If building this project for a client, I would:

   - Make sure it felt really polished and professional. So, I'd focus on adding a few extra touches that make a big difference in how people experience the site. For example, I'd add smooth animations and subtle transitions
   - I'd also spend extra time making sure the form is super user-friendly, with clear error messages and helpful hints when people type their info .
   - On the design side, I'd pick colors and fonts that really match the client's brand personality and create richer visuals, maybe turning the photo section into a nice gallery or slideshow.
   - Finally, I'd link the contact form to a backend service to save messages or send email notifications, instead of just showing data on the page. And before handing it off, I'd write up clear instructions so the client can update their site easily in the future.

3. Describe entire flow of creating this site: design → code → test → deploy.

   Design :

- To start, I sketched out a rough plan of what the site should look like and what sections I wanted—like About, Photo, and Contact.
- Chose design tools and approaches such as Flexbox and Grid for layout, and decided whether to rely more on Flexbox or Grid for positioning different sections
- Then I picked tools Flexbox and Grid to help organize things neatly, imagining how the site would look on different screen sizes.

Code :

- Once I had a clear picture, I jumped into writing the HTML to build the page's skeleton. It was like framing a house—making sure all the rooms (sections) were in the right place .
- Then I used CSS to decorate the site, controlling colors, spacing, and how everything lined up.

Test :

- After the coding was done, I played around with the site on different devices and browsers, kind of like trying on clothes to see if they fit right everywhere.
- Also tested the contact form's validation to make sure only the right inputs were accepted, and all submitted data showed up neatly without needing to refresh the page.

Deploy :

- When it was time to put the site out there for everyone to see, I started by setting up Git to keep track of all my changes.
- I made sure to commit my progress regularly and even created a separate branch when trying out new features, so the main site stayed safe and stable.
- I pushed the final version to GitHub and used GitHub Pages to publish the site online.
- I saw my site live on the internet, which was really exciting because it meant my work was accessible to anyone, anytime.
- Finally, I wrote up a README to explain how things were set up and how someone else (or me in the future) could keep working on the project without confusion