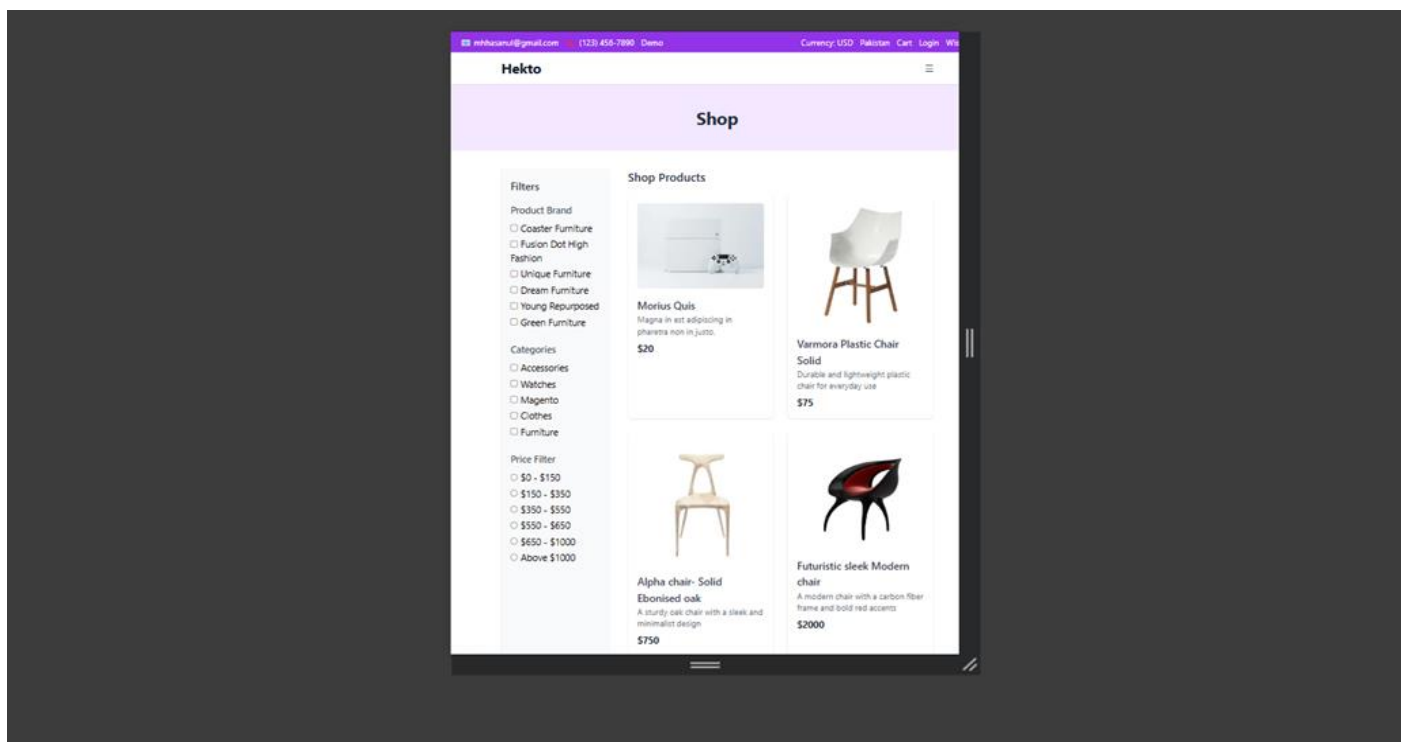
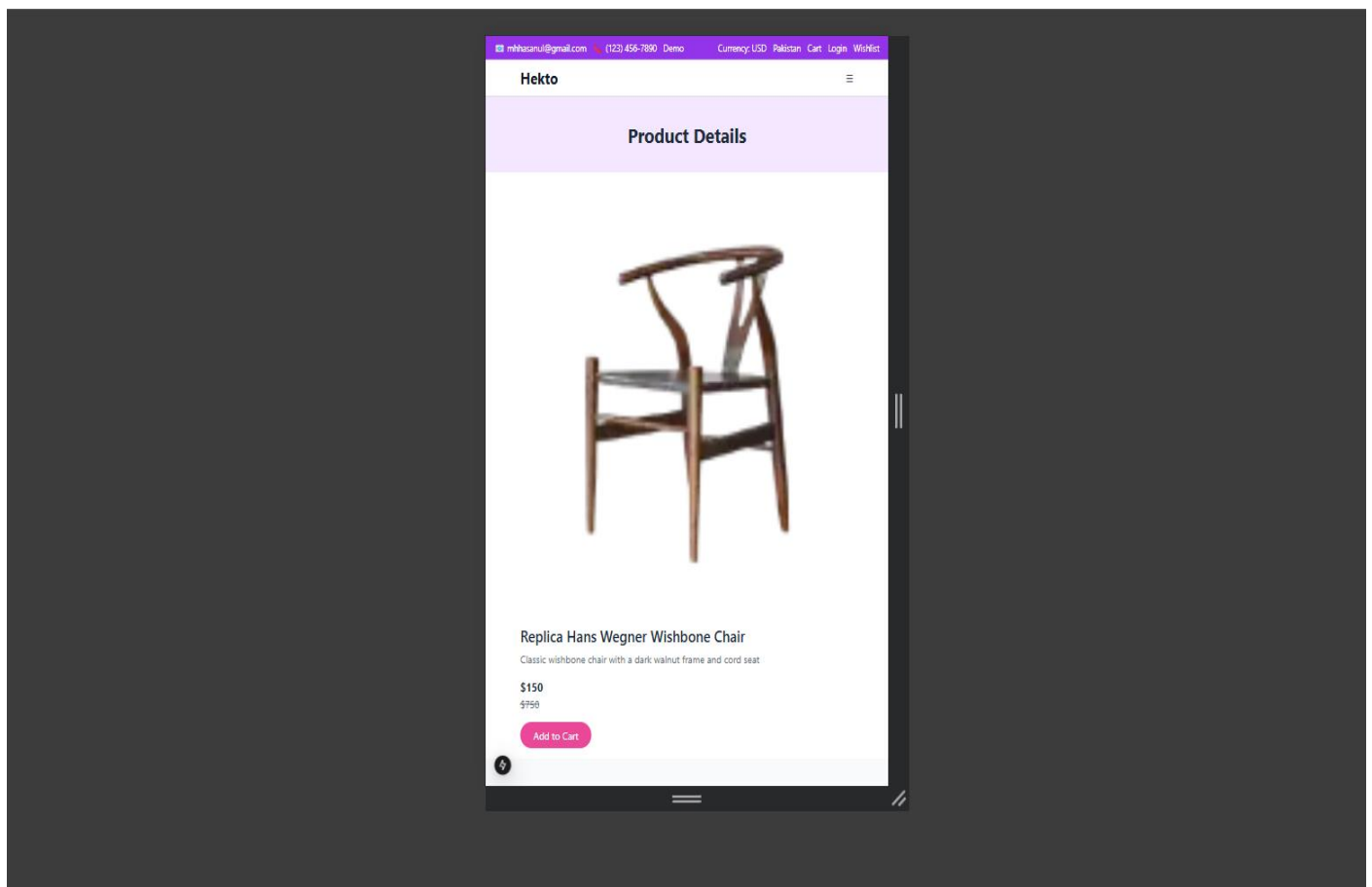
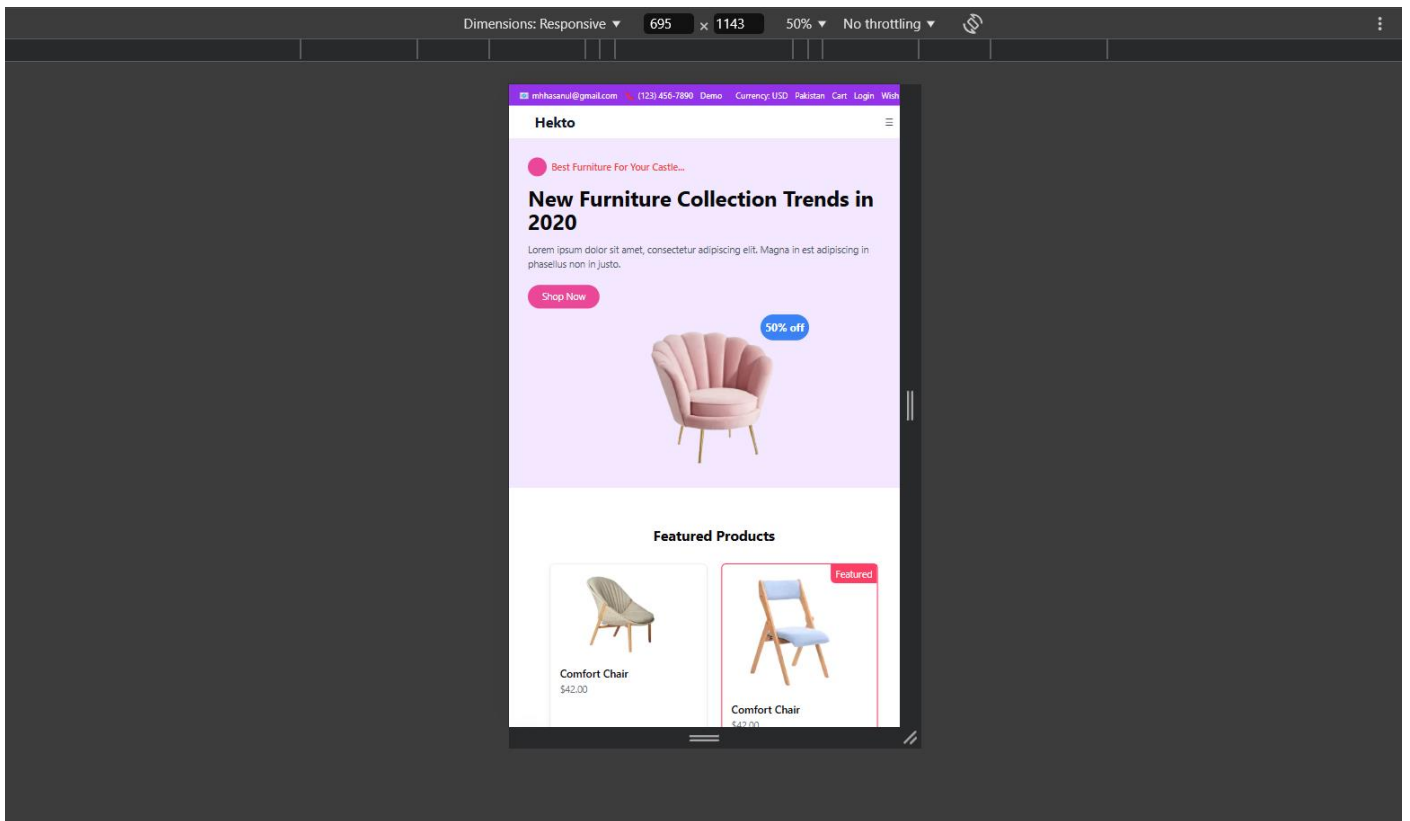


Hackathon Day 5

Screenshots: Responsiveness





Documentation

1. Text Cases Executed and Their Results:

All are Given in the Excel File.

2. Performance Optimization Steps Taken:

2.1 Code Optimization

- Minified CSS and Typescript files: Reduced file sizes to improve load times.
- Removed unused code: Cleaned up redundant scripts and styles.

2.2 Image Optimization

- Compressed images using modern tools without quality loss.
- Implemented lazy loading for images below the fold.

2.3 Caching Mechanisms

- Enabled browser caching for static resources.
- Configured server-side caching to reduce load on the database.

3. Security measures implemented:

Stored All the Important Data which Including sanity dataset and Project Id in the .env file for avoiding data breach.

4. Challenges Faced and Resolutions Applied:

4.1 Challenge: Slow Page Load Times

- **Cause:** Large images and unoptimized scripts.
- **Resolution:** Compressed images, reduced unused code, and used lazy loading.

4.2 Challenge: Poor Mobile Experience

- **Cause:** Website wasn't properly designed for mobile screens.
- **Resolution:** Improved mobile responsiveness using CSS and tested with different devices.

4.3 Challenge: Slow API Calls

- **Cause:** Inefficient API handling.
- **Resolution:** Optimized API requests and used Reacts Next.js built-in fetching methods, Which Includes async Function and await, And also used sanity Client to get data from sanity studio.