Collaborative Problem-Chit Chat

By Nazmul Hussain

- How can we make web applications better?

- How can we improve our existing projects?

Outline

- Design sense

- Feature implementation

- Explore packages and find the perfect package

- Handling unexpected error

Design Sense

Improvement Techniques & Resources

Improvement Techniques

1. Study Design Fundamentals:
   - layout, typography, color theory, and hierarchy.
   - basics of user experience (UX)

2. Read Design Books and Blogs:
   - Explore design books, blogs
   - Online resources (Behance, Dribbble, Awwwards)

3. Analyze Well-Designed Websites:
   - Understand how design elements are applied in practice
   - Whitespace, alignment, contrast, and visual hierarchy.

4. Use Design Tools:
   - Adobe XD, Figma, Sketch, Canva
   - Design mockups and prototypes

5. Practice Regularly:
   - Working on small design projects
   - Wireframes and mockups for web pages

Resources

- [70+ Web Design Resources](#)

Feature Implementation

Identify Features, Unique Feature Invention,
Parallel Feature Integration

Identify Features

1. Purpose and Functionality:
   - Beyond basic website content or navigation
   - Provides a service, tool, or capability to the user

2. User Interaction:
   - Through clicks, inputs, or other actions
   - Enable users to perform tasks, access information, or achieve specific goals.

3. Added Value:
   - Enhancing the UX, Providing utility
   - Contribute to the site's functionality

4. Distinctiveness:
   - Identified as separate from the standard content.
   - Offer unique capabilities

5. Complexity:
   - Relatively simple, like a search bar.
   - Complex, such as a real-time chat application.

Invention

1. Understand User Needs:
   - Pain points of target audience
   - Gather insights: Research and Surveys

2. Study the Competition:
   - Analyze other websites and platforms
   - Identify gaps or opportunities

3. Brainstorm Ideas:
   - Sessions with Team/Individually
   - No idea is too outlandish at this stage

4. Think Beyond the Obvious:
   - Avoid common and overused features
   - Offer unique capabilities

5. Cross-Pollination of Ideas
   - Borrowing ideas from unrelated industries
   - Creative solutions often come from unexpected sources.

Explore Package & Find the Perfect Package

Package for My Project?

1. Understand the Need.
2. Explore the "npm Trends" Website
3. Read Blogs and Tech News Sites

   (Dev.to, Smashing Magazine, CSS-Tricks)
4. Follow Developer Forums and Communities

1. Project Requirements
2. Popularity and Community Support
3. Documentation
4. Maintainer Activity, Version History
5. Dependencies, Security
6. Performance, Size
7. Scalability, Customizability

Handling Unexpected Errors.

Error: The Best Part!

1. Get Familiar With Error Types
2. Learn To Read & Understand Errors
3. Find Out exactly where it's happening
4. Isolate the Problem
5. Research and Documentation
6. Consult Colleagues and Online Communities.
7. Error Page
8. Preventative Measures
9. Learn from Errors

Resources

- [Comprehensive guide to error handling](#)
- [JavaScript error handling](#)

<QnA/>
"Ask Me"

Thank You👍