

High Level Design (HLD)

Personal Portfolio Website

Revision Number: 2.0

Last date of revision: 6 Feb 2023

Shubham Saptasagare

Jayesh Rajput

Document Version Control

Date Issued	Version	Description	Author
18/01/2023	1.0	Added Introduction and General description	Shubham Saptasagare
21/01/2023	1.1	Added Design details and abstract	Shubham Saptasagare
22/01/2023	1.2	Created and organized whole document	Shubham Saptasagare
25/01/2023	1.3	Updated Design Details	Shubham Saptasagare
27/01/2023	1.4	Checked and added Performance	Jayesh Rajput
28/01/2023	1.5	Added Conclusion and References	Jayesh Rajput
06/02/2023	2.0	Updated diagrams in Design Details	Jayesh Rajput

Table of Content

Document Version Control	2
Abstract	4
Introduction	5
1. Why this High-Level Document?	5
2. Scope	5
3. Definition	5
General Description	6
1. Product Perspective	6
2. Problem Statement	6
3. Problem Solution	6
4. Proposed Methodology	6
5. Further Improvements	6
6. Data Required	6
7. Tools Used	7
8. Constraints	7
9. Assumptions	7
Design Details	8
1. Process Workflow	8
2. Error Handling	9
Performance	10
1. Reusability	10
2. Application compatibility	10
3. Resources Utilization	10
4. Deployment	10
Conclusion	12
Reference	13

Abstract

A personal portfolio is a collection of documents, materials, and information that showcases a person's skills, experiences, achievements, and qualifications. An abstract of a personal portfolio provides a brief overview of its contents, purpose, and main highlights. It serves as an introduction to the individual's background and capabilities, and is typically included at the beginning of the portfolio. The abstract should be concise and clearly convey the key aspects of the person's background and qualifications.

Introduction

1. Why this High-Level Design Document?

A High-Level Design Document provides a broad overview of a system, including its overall architecture, components, modules, and interfaces. It serves as a blueprint for the development of the system and helps stakeholders understand the system's structure and functions. Additionally, it can also be used to communicate the design decisions and constraints to the development team, facilitate discussions and collaboration among team members, and provide a basis for detailed design and implementation.

The HLD will:

- Present all the design aspects and define them in detail
- Describe the user interface being implemented
- Describe the hardware and software interfaces
- Describe the performance requirements
- Include design features and the architecture of the project

2. Scope

The HLD documentation presents the structure of the system, such as the database architecture, application architecture (layers), application flow (Navigation), and technology architecture. The HLD uses non-technical to mildly technical terms which should be understandable to the administrators of the system.

3. Definition

The terms used in the projects are:

- Portfolio- All personal technical information will be stored here.
- About me- These talks about user profile.
- Services- The technology used by user. These are some skills of the user.
- Contact- The all contact information will be kept on this page.

General Description

1. Product Perspective

From a product perspective, a portfolio refers to a collection of products or services offered by an organization, typically categorized according to their stage in the product lifecycle (i.e. development, introduction, growth, maturity, and decline). The purpose of a product portfolio is to provide a comprehensive overview of the organization's offerings and to ensure a balance between products that generate revenue and those that are being developed for future growth. The portfolio helps organizations make informed decisions regarding resource allocation, product positioning, and investment strategy.

2. Problem Statement

To create the personal portfolio website.

3. Problem Solution

Develop a website for the portfolio. Add all the information of ourselves. Projects, internships, achievements, experience, education etc things we should add in this website. So all are getting our personal information easily.

4. Further Improvement

The project can be implemented by full stack development. We can use PHP, DBMS systems etc for storing the data provided by all. All project link can be given. We should make separate websites for this.

5. Data Required

For making this website below data is required

1. Personal Information: Name, title, contact information, brief introduction, and profile picture.
2. Skills and Expertise: A list of skills, technical abilities, and areas of expertise.
3. Work Experience: A summary of past work experience, including company names, job titles, dates of employment, and key responsibilities and accomplishments.

4. Projects: Detailed information on past and current projects, including project descriptions, images, videos, and links to live demos or code repositories.
5. Education: Information on academic background, including degrees obtained, universities attended, and relevant coursework.

6. Tools Used

- For making whole code VS Code is used.
- Chrome is used for display the website.
- Web hosting tool

7. Constraints

User Experience: The portfolio website should provide a seamless user experience with easy navigation and clear information presentation.

Device Compatibility: The website should be responsive and accessible on different devices such as laptops, smartphones, and tablets.

Loading Speed: The website should have a fast loading speed to prevent users from losing interest and leaving the site.

Content Limitations: The portfolio website should present only relevant information and not overload the user with too much content.

Design Constraints: The design should be visually appealing and professional, while also reflecting the individual's personal brand and style.

8. Assumptions

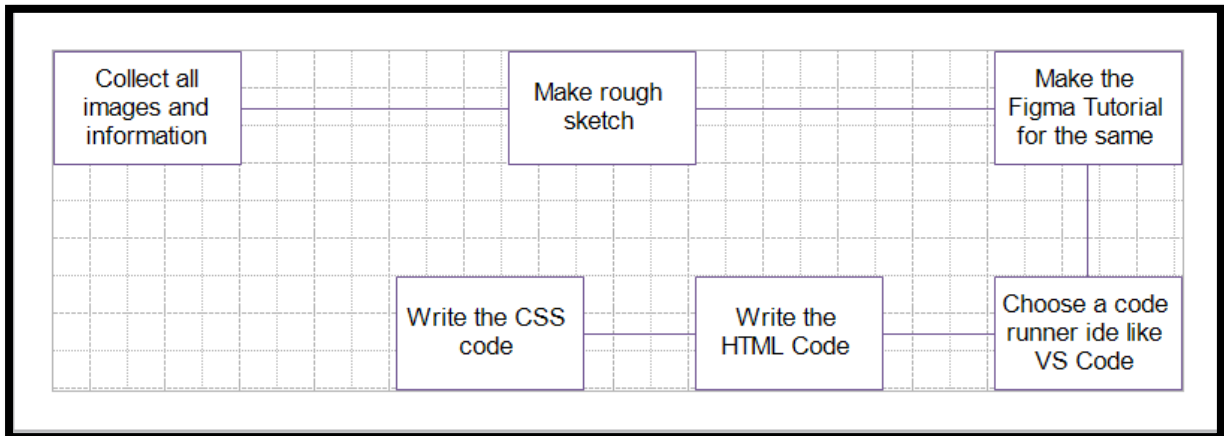
- The website represents the creator's professional brand and showcases their skills, achievements, and experience.
- Visitors to the site are primarily potential clients, employers, or collaborators who are interested in learning more about the creator's work.
- The website should be visually appealing and easy to navigate, allowing visitors to quickly find the information they are looking for.

- The website should be mobile-friendly and optimized for various screen sizes and devices.
- The creator will regularly update their portfolio to keep it current and relevant. The website should accurately reflect the creator's personal style, aesthetic, and professional goals.
- The website should provide easy access to relevant contact information, such as an email address or social media profiles.

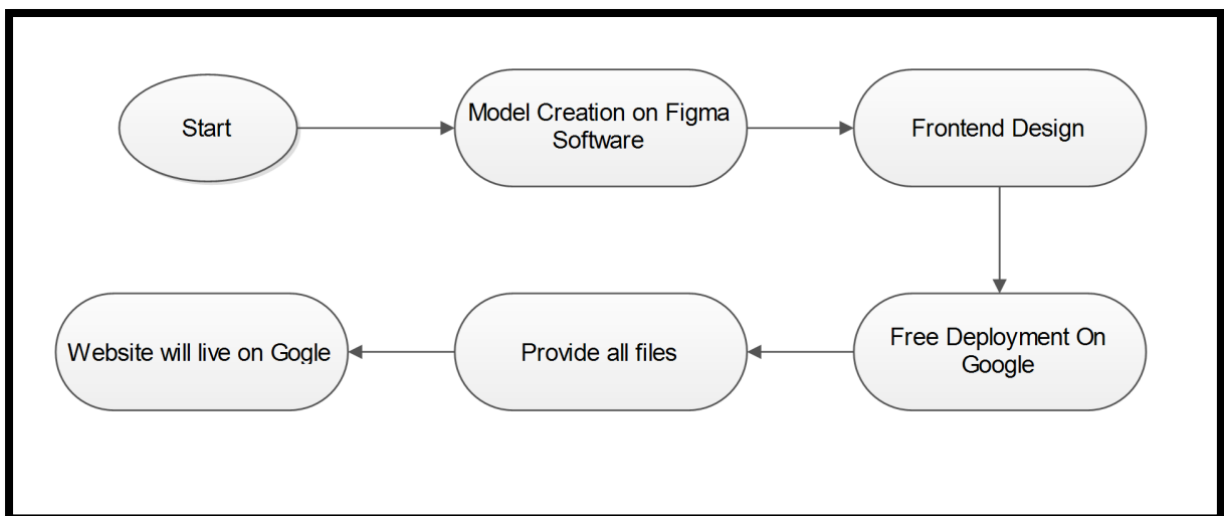
Design Details

1. Process Workflow

Below is the process flow diagram.



Deployment Process



2. Error Handling

When making a portfolio website, error handling refers to the process of anticipating and responding to potential errors or problems that may arise. Here are some best practices for error handling in a portfolio website:

1. **Validate user inputs:** Ensure that all forms and inputs have proper validation to prevent incorrect data from being submitted.
2. **Use try-catch blocks:** Surround critical code with try-catch blocks to catch and handle exceptions that may occur.
3. **Log errors:** Log all errors with detailed information such as the error message, stack trace, and date/time to help with debugging and problem solving.
4. **Display user-friendly error messages:** Show user-friendly error messages to help users understand what went wrong and what they can do to fix it.

Performance

1. Reusability

Use a responsive design that adapts to different screen sizes, allowing the website to be viewed on a variety of devices.

Use a modular design that allows you to reuse components, such as header and footer, on multiple pages.

Use a CSS framework like Bootstrap or Foundation to help with the overall design and responsiveness of the website.

2. Application compatibility

The compatibility of a portfolio website depends on various factors such as the technology used to build the website, the target audience and devices, and the web browsers supported. To ensure maximum compatibility, it's best to build a responsive website that adjusts to different screen sizes and device types, and test the website on popular web browsers (e.g. Google Chrome, Mozilla Firefox, Apple Safari, Microsoft Edge) to ensure it works well on each. Additionally, using up-to-date technologies, such as HTML5 and CSS3, can help ensure the website works well on modern devices and browsers.

3. Resource utilization

Resource utilization of a portfolio website refers to the amount of computing resources, such as CPU, memory, and network bandwidth, used by the website to run and deliver its content to users. Factors that impact resource utilization include the size and complexity of the website, the number of visitors, the type of content served (e.g. text, images, videos), and the infrastructure hosting the website (e.g. shared hosting, dedicated server, cloud-based solutions).

4. Deployment

The code is deployed in GitHub. The whole system is live and is hosted on netlify app. This provides free web hosting.

Conclusion

The conclusion of making a portfolio website is the realization of a well-designed and functional platform that showcases your skills, achievements, and experiences to potential clients, employers, and other interested parties. A portfolio website is a crucial tool for anyone seeking to establish an online presence and present themselves professionally in their chosen field. The successful completion of a portfolio website requires careful planning, attention to detail, and a thorough understanding of design, web development, and digital marketing principles. Ultimately, a well-made portfolio website can help you stand out in a competitive job market, build your personal brand, and advance your career.

References

1. <https://themeforest.net/search/web%20developer%20portfolio>
2. https://dribbble.com/tags/web_developer_portfolio
3. <https://plainenglish.io/blog/40-portfolio-templates-free-for-web-design>
4. <https://www.sanity.io/guides/best-developer-portfolio-templates-2021>
5. <https://brainstation.io/career-guides/how-to-build-a-web-developer-portfolio>