Chaper 3

**METHODOLOGY**

This chapter discusses the methodology used in the development of the system titled **“Creative Tee: Custom Design Platform.”** It provides explanations of how the system was planned, developed, and tested. It also includes the system's technical specifications, tools that were used, and the adopted software development model. this chapter also includes a software cost estimation that helped guide the development process.

**Project Design**

The development of **Creative Tee: Custom Design Platform** focused on providing a user-friendly and interactive web-based system that grants users to create personalized apparel such as T-shirts, mugs, caps, etc. The platform has features like a design canvas, drag-and-drop elements, real-time product previews, and an integrated ordering system. Users can choose product templates, upload custom images or texts, and place orders for their unique designs.

The front end of the system was designed to be intuitive and responsive using **ReactJS**, **HTML**, **CSS**, and **JavaScript**, while the back-end was developed using **NodeJS** for handling application logic and **MongoDB** as the database for storing user data, design files, and order transactions.

The goal was to create a seamless experience from designing to ordering, ensuring that the system is both creative and functional for users of all skill levels.

Fishbone Diagram

Figure 1

Fish Diagram

Figure 2

Use case Diagram

Figure 3

Data flow diagram

**Technical Requirements**

**Hardware**

Hardware includes the tangible part of the computer,

It also consist of internal and external parts that will be used in connecting these Devices

For storing data and pther important task.

**Hardware Requirements**

The system was developed and tested using the following hardware specifications:

* **Laptop Model:** Huawei Matebook D15
* **Operating System:** Windows 11 Home
* **Processor:** 11th Gen Intel(R) Core(TM) i5-1135G7 @ 2.50GHz
* **Installed RAM:** 16.0 GB (15.8 GB usable)
* **System Type:** 64-bit Operating System, x64-based processor

**Software Requirements**

The following software tools were utilized in the system development:

* **Visual Studio Code** – used as the main Integrated Development Environment (IDE) for writing and editing the codebase.
* **MySQL** – is an open-source relational database management system (RDBMS) that uses **Structured Query Language (SQL)** to manage and manipulate data.
* **XAMPP** – is a free, open-source software package that provides a local server environment for developing and testing web applications.
* **HTML** – is the standard language used to create and structure content on the web.
* **CSS** – is a language used to style and design HTML content. It controls the layout, colors, fonts, spacing, and overall visual appearance of web pages.
* **JavaScript** – is a programming language used to make web pages interactive and dynamic. It allows you to add features like clickable buttons, form validations, animations, and real-time content updates.
* **Web Browsers** – such as Google Chrome and Microsoft Edge, used for testing the platform’s compatibility.

Software Development Models and Procedures

The researchers adopted the **Modified Waterfall Model** in developing the Creative Tee system. We used Modified waater fall model because its not complicated very straight through structure and logical sequence, allowing us both to work systematically from requirements to deployment. It also Grants the researchers to return to previous phases when necessary for correction and improvements.

The phases followed in the development are outlined below:

**1. Requirements Gathering and Analysis**

This phase focused on identifying the needs of users who want to create personalized apparel. Data was gathered through interviews, and surveys. The researchers identified essential features such as product selection, design canvas, image/text upload, real-time preview, and order placement.

**2. System Design**

In this phase, the structure and flow of the system were planned. The researchers designed the user interface for both customers and administrators. Data models and system architecture diagrams were created. The user interface was designed to be simple, visually appealing, and easy to navigate.

**3. Implementation (Coding)**

The actual development of the system took place in this phase. The front end was built using ReactJS to allow real-time updates to the product designs, while the backend was implemented using NodeJS to handle user accounts, file uploads, product customization, and order processing. All data were stored in MongoDB, ensuring flexibility and scalability.

**4. Testing**

Testing was done to ensure that all features of the system worked correctly and met the identified requirements. The system underwent unit testing for each module, integration testing to ensure that modules worked together, and user acceptance testing (UAT) to verify the system’s usability and performance.

**5. Deployment and Maintenance**

After testing, the system was deployed to a live server for public use. Monitoring tools were used to track performance and security. User feedback was continuously collected to identify areas for improvement. Maintenance included fixing bugs, updating content, and ensuring system security.

**Software Cost Estimation**

the Software cost estimation is a huge challenge for web Developers in the Recent years cost of developing the system, either way the most used mthod in software cost estimation is the **Constructive Cost Model (COCOMO)**.

the **Basic COCOMO** was used to provide an overview of effort and time based on estimated lines of code (LOC). As development progressed, the **Intermediate COCOMO Model** was applied, which considered additional factors such as product reliability, system complexity, team experience, and development tools.

This estimation helped the researchers plan and allocate resources efficiently throughout the project lifecycle.