Caraga State University
College of Computing and Information Sciences
Hiraya Hall, CSU-Main Campus, Ampayon, Butuan City
Competence Service Uprightness
YOUR BEST OPTION TO SUCCESS

Phone 09177078764
09177078713
09177078769
Loc. 123
Email Address: ccis@carsu.edu.ph
URL: https://ccis@carsu.edu.ph

INFORMATION TECHNOLOGY DEPARTMENT

1st Semester 2023-2024

PROJECT 2

IT 110 – Web Systems and Technologies

System Analyst: _____ Section: _____

Frontend Developer: _____ Section: _____

Backend Developer: _____ Section: _____

TITLE:

Multitier Marvel: Crafting a Full Stack Web Application with UI Framework (Bootstrap or Tailwind), JS Framework (React or Vue) or VanillaJS, and Laravel REST APIs.

PURPOSE:

The overarching purpose of this project is to create a feature-rich web application with a responsive and visually appealing UI, leveraging the latest technologies and best practices in web development. By adopting a Multitier Architecture, the project aims to achieve scalability and maintainability, while the chosen technologies and deployment strategies are intended to ensure a seamless user experience and efficient development and testing processes. Ultimately, this project seeks to deliver a high-quality Full Stack Web Application that meets the needs and expectations of its target audience.

- ✓ **Multitier Architecture**: Implement a clear separation of concerns by dividing the application into distinct layers, including a presentation layer (Frontend) and a data processing layer (Backend). This approach enhances modularity, maintainability, and scalability.
- ✓ **User Interface (UI) Framework**: Utilize a UI framework, such as Bootstrap or Tailwind, to ensure a visually appealing and responsive design. A well-designed user interface is critical for delivering an exceptional user experience.
- ✓ **JavaScript Framework**: Choose between React, Vue, or Vanilla JavaScript to build the Frontend. These frameworks offer varying levels of functionality and performance, catering to different project requirements and developer preferences.
- ✓ **Laravel Backend REST APIs**: Develop a Backend layer using Laravel, a powerful PHP framework for creating RESTful APIs. These APIs will be responsible for handling data processing, storage, and interaction with the Frontend.
- ✓ **Deployment on Vercel or Netlify**: Deploy the Frontend component on platforms like Vercel or Netlify. These services offer easy deployment and hosting, ensuring high availability and fast load times for the application.
- ✓ **Backend Deployment on ngrok**: Utilize ngrok for the Backend deployment. ngrok provides a secure and easy-to-use tunneling service that allows external access to your local server, making it suitable for testing and development purposes.

In addition, students will gain further knowledge of the following:

- **Git,** is a free and open source distributed version control system designed to handle everything from small to very large projects.

- **Github,** is a platform and cloud-based service for software development and version control using Git, allowing developers to store and manage their code.
- **HTTP client Fetch API,** is a modern web standard that provides a JavaScript interface for making network requests to retrieve resources from the web and handle the response data, enabling asynchronous communication between a web application and a server.
- **Laravel as a backend REST API,** is a PHP framework that allows developers to build robust and scalable web services by providing a structured, efficient, and developer-friendly way to handle data processing and communication with client applications via RESTful endpoints.
- **Vercel or Netlify** are cloud-based hosting and deployment platforms that simplify the process of deploying and managing web applications, offering features like automatic scaling and continuous integration to streamline web development workflows.
- **Ngrok** is a secure tunneling service that provides an external, public-accessible URL to your locally hosted server, facilitating testing, development, and sharing of web applications without the need for complex network configurations.

INSTRUCTIONS:

➢ **Create a team with three members** (Cross section is allowed), and within the team, designate one member as the Frontend developer, another as the Backend developer, and the third member as the System Analyst. **Generate a web app idea** where you can apply advanced database concepts, whether it's an existing or entirely new concept.

✓ **The System Analyst's Responsibilities:**
- ✧ **Design** the web app's advanced **database schema or ERD**.
- ✧ **Create API documentation**, provide technical descriptions for the Postman API collection, which will be generated by the Backend Developer and made available online using Postman. The Frontend Developer will then use this API documentation.
- ✧ Responsibilities include **showcasing both the database schema (ERD) and the API documentation, as well as delivering a comprehensive technical demonstration** of all the web application's features in a classroom setting.

✓ **The Frontend Developer's Responsibilities:**
- ✧ **Build the user interface** (following the schema designed by the System Analyst).
- ✧ **Implement frontend features**, including JavaScript functions (e.g. DOM manipulation and etc) and HTTP requests to Backend REST APIs deployed via Ngrok.
- ✧ **Store the source code or project folder in their own GitHub account**.
- ✧ **Link the Github Repository to either Vercel or Netlify for web deployment**.

✓ **The Backend Developer's Responsibilities:**
- ✧ **Create the Backend REST APIs** (following the schema designed by the System Analyst).
- ✧ **Store the source code or project folder in their own GitHub account**.
- ✧ **Setup Ngrok in their own workstation** to deploy Backend project.
- ✧ **Provide Postman REST API collection,** share the Postman REST API collection with both the System Analyst and the Frontend Developer's Postman accounts. The System Analyst will use this collection to create technical descriptions for API documentation, which the Frontend Developer will then employ.

➢ **The instructor will offer clear instructions on deploying, setting up Ngrok, sharing the Postman Collection, and making it accessible online.** Recordings of these instructions will also be available.

SUBMISSION AND PRESENTATION:

DATE:

- ✓ **December 3rd Week, All Sections** (Lab Schedules)

ORDER OF PRESENTATION:

- ✓ Group evaluations will take place exclusively in the Laboratory Room based on Lab Schedule. The use of the Wheel of Names site will be utilized for the Order of presentation.

TYPE OF PRESENTATION:

- ✓ **The instructor will conduct group evaluation or project defense.** Additionally, students will engage in brief discussions to explain their project's development process.

RATING OF PRESENTATION:

- ✓ The Instructor will utilize the Project Rubrics.

PRE-REQUISITES:

- ➢ In your group, **the Frontend Developer and Backend Developer should each have their own workstation (laptop)**. **The Frontend Developer's laptop is necessary for the project demonstration and presentation**, while the **Backend Developer's laptop will be used to check if any HTTP requests from the frontend application directly affect the locally hosted database (via Ngrok)**. Students without access to a laptop can borrow one for their presentation. It's vital to optimize the workstations (laptops) to prevent any problems during both development and the final presentation.

APPLICATION INFORMATION:

- ➢ Prior to the project defense, the **System Analyst is required to complete the following information.**

Name of Application: _____

Frontend:

Frontend URL: _____

Frontend Github Repo: _____

Backend:

Ngrok (with API path) URL: _____

Backend Github Repo: _____

API Documentation:

Postman API URL: _____

PROJECT RUBRICS:

Core Criteria	Criteria Description	Performance Rating				Rating
		Exceeds Expectations	Meets Expectations	Needs Improvement	No Implementation	
Frontend Requirements	**Criteria No. 1:** The website has an appealing and user-friendly design with a consistent and attractive color scheme across all pages, ensuring a visually cohesive experience. It is responsive, working well on both web and mobile devices, and features consistent font styles and legible text. Navigation links are clear, consistently positioned, and provide easy access to related content, guiding users effectively.	**4**	**3**	**2**	**1**	
	Criteria No. 2: The frontend application effectively notifies users of any activities or actions, ensuring correct outputs, error-free displays, and appropriate alert messages when actions are taken. It also introduces additional application features, demonstrating creative thinking and innovation.	**4**	**3**	**2**	**1**	
	Criteria No. 3: The user interface forms are aligned with the established database schema. The source code or project folder is efficiently uploaded to the Frontend Developer's GitHub account, and the GitHub repository is connected and successfully deployed on the chosen platform. Ensure that Git commits are either pushed to or merged into the main branch.	**4**	**3**	**2**	**1**	

Backend Requirements	**Criteria No. 4:** Each frontend application's HTTP request updates the locally hosted database, implementing server-side validation and establishing Laravel Sanctum authentication security on necessary API endpoints.	**4**	**3**	**2**	**1**	
	Criteria No. 5: The Laravel database migrations align with the database schema designed by the system analyst, and the source code or project folder has been stored in the backend developer's GitHub account. The backend REST API is accessible via ngrok and has been configured correctly. Ensure that Git commits are either pushed to or merged into the main branch.	**4**	**3**	**2**	**1**	
System Presentation	**Criteria No. 6:** The database schema is properly normalized, and the API documentation is well-prepared, with each API endpoint thoroughly explained and accurately described.	**4**	**3**	**2**	**1**	
	Criteria No. 7: All frontend application features, ranging from authentication to the core functionality of the web application, are demonstrated clearly and comprehensively explained.	**4**	**3**	**2**	**1**	
		Total Rating (out of 28)				

Overall Comment and Suggestion:

Phone 09177078764
09177078713
09177078769
Loc. 123
Email Address: ccis@carsu.edu.ph
URL: https://ccis@carsu.edu.ph

Prepared by:

JABEZ IAN CHRIS D. PENALVER
Instructor, IT Department

Checked/Reviewed by:

MARIA BESA JOY ORTUYO, MSc.
IT CHAIRPERSON

Approved by:

VICENTE PITOGO, DIT
CCIS DEAN