VIETNAM NATIONAL UNIVERSITY OF HOCHIMINH CITY
THE INTERNATIONAL UNIVERSITY
SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

SOFTWARE ENGINEERING
IT076IU

MIDTERM REPORT

Topic: Web portal for motor servicing at home

By Group 03 – Member List

No.	Full name – Student name	Student ID
1	Trịnh Thuỷ Tiên	ITITIU21328
2	Trần Bách Tùng	ITITIU21340
3	Đặng Nguyễn Trường Huy	ITITIU21010
4	Nguyễn Hoàng Minh Khôi	ITITIU21229
5	Bùi Phương Thanh	ITITIU21311
6	Nguyễn Trần Gia Huy	ITITIU21054
7	Vũ Văn Đô	ITITIU21125
8	Bạch Huy Hoàng	ITITWE21113
9	Nguyễn Trần Quốc Thịnh	ITITIU20312

Instructor: Assoc. Prof. Nguyen Thi Thuy Loan

April 2024

1. Introduction:

a. Overview & Objectives:

Modern households face time limits and inconveniences with traditional vehicle care, creating a need for accessible, trustworthy, and efficient motor services.

Scheduling complexity and ensuring service provider competence add to the challenge. There is a compelling need for a solution that delivers trustworthy automotive care directly to clients' homes, addressing key pain areas promptly and effectively.

b. Briefly summarize:

In this project , as a team of 9 members, we aim to implement these features into our project:

- User Authentication and Profiles
- Service Booking
- Payment Integration
- Continuous Improvement

Using various tools and programming languages:

- IDE: Visual Studio Code and IntelliJ IDEA.
- Language programming: HTML, CSS, JavaScript.
- Framework: PHP, CSS.
- Frontend: HTML, CSS.
- Backend: JavaScript.
- Tool: VSC, IntelliJ IDEA, GitHub, Figma... IDEs: Visual Studio Code, IntelliJ IDEA Collaboration Tools: Microsoft Team, Discord Code Repositories and Hosting Platforms, manage source code: GitHub g. Library: Express, Axios.

And our objectives for this project are: Convenience , Accessibility , Transparency , Customer Satisfaction , Efficiency.

Our implementation of front end design is completed at this moment.

2. Implementation Details:

a. The technologies, programming languages, frameworks, and tools used

- **Technologies and Tools (addition):**
  - + **Vue.js:** Vue.js is a progressive JavaScript framework used for building user interfaces. It's known for its simplicity and flexibility**.**
  - + **Vue Router:** Vue Router is the official router for Vue.js. It provides a way to manage application-level routing in Vue applications**.**
  - + **PrimeVue:** PrimeVue is a UI component library for Vue.js. It provides a set of rich and customizable components for building user interfaces**.**
  - + **Sass:** Sass is a preprocessor scripting language that is interpreted or compiled into CSS. It adds features like variables, mixins, and nesting to CSS.
- **Framework:** PHP, CSS.

  They offer benefits in terms of efficiency, maintainability and scalability. They enable the teams to reach for the requirements of the direction our teams are aiming.
- **Frontend:** HTML, CSS.

These are the two languages that the frontend team applied in designing the web-user graphic interface. The frontend team had done well in creating a website interface, which users can view and interact with, using HTML and CSS.

- **Backend:** JavaScript.

  The backend team agreed on using one of the most common programming languages to create the pages with JavaScript. The language is familiar yet efficient in building a multipage website.

b. Design Patterns or Software Engineering Principles:

- **Component-Based Architecture:**

  Vue.js encourages a component-based architecture, where the user interface is broken down into reusable and composable components. This improves code maintainability and encourages reusability.
- **Single Responsibility Principle (SRP):**

  Components and modules likely adhere to the SRP, meaning each component or module should have a single responsibility and reason to change. This makes the codebase more maintainable and easier to understand.
- **Modular Development:**

  The project seems to follow a modular development approach, with dependencies managed through npm. This allows for better organization, reuse, and collaboration among team members.
- **Test-Driven Development (TDD):**

  The inclusion of Vitest and unit test scripts (test:unit) suggests a commitment to test-driven development practices. Writing tests before implementing features ensures code correctness and helps prevent regressions.
- **Linting and Formatting:**

  The project uses ESLint and Prettier to enforce coding standards and formatting conventions. Consistent code style across the project improves readability and maintainability.

3. Testing and Evaluation:

There are types of tests:

- **Unit Tests**: focus on testing individual units or components of the code in isolation. In the context of a Vue.js application, this could mean testing Vue components, Vuex store actions, mutations, and getters, as well as any utility functions.
- **Integration Tests**: verify that different parts of the system work together correctly. In the context of a Vue.js application, integration tests may focus on testing how Vue components interact with each other, how they interact with external services, and how they handle state management (using Vuex or other state management solutions).
- **System Tests**: verify that the entire system, including the frontend, backend, and any external services, behaves as expected from the user's perspective. These tests simulate real user interactions with the application.

4. Progress and Challenges:

a. PROJECT TIMELINE

No.	Timeline	Process	Task	Responsibility	Note
1	22/02 - 03/03	Requirement & Analyze	Requirement Analysis	all of members	
2	03/03 - 06/03		Requirement Specification	all of members	
3	07/03 - 14/03		Writing the proposal	Trường Huy, Minh Khôi Phương Thanh, Gia Huy Văn Đô, Quốc Thịnh	Partner: Thuỷ Tiên Bách Tùng Huy Hoàng
4	15/03 - 17/03	Design	Analyze design	Bách Tùng Thủy Tiên	
5	18/03 - 20/03		Design Mockups Creation	Bách Tùng	Partner: Thủy Tiên
6	21/03 - 25/03		Architectural Design	Phương Thanh	(cont)
7	26/03 - 28/03		Object-oriented Design	Trường Huy	(cont)
8	25/03 - 31/03		Database Design	Văn Đô	(cont)
9	25/03 - 25/04		Website interface design	Thuỷ Tiên Bách Tùng Huy Hoàng	(cont)
10	01/04 - 14/04		Writing online progress report	all of members	
11	15/04 - 27/04		Midterm exam	all of members	
12	28/04 - 18/05	Implementation	Functions Coding	Trường Huy Minh Khôi Gia Huy	
13	15/05 - 20/05	Testing	Content Integration	Trường Huy	
14	21/05 - 23/06		Test cases	Phương Thanh	Partner: Quốc Thịnh
15	24/05 - 26/05		Test data	Huy Hoàng	
16	27/05 - 29/05		Integration Testing	Thủy Tiên, Quốc Thịnh	
17	30/05 - 01/06		System Testing	Gia Huy	
18	02/06 - 05/06		Risk Monitoring and Planning	Phương Thanh Quốc Thịnh	
19	06/06 - 10/06	Deploying		all of members	

20	25/05 - 04/06		Writing Final Report	all of members	
21	05/06 - 12/06		Prepare file presentation	all of members	
22	13/06 - 20/6		Review + demo	all of members	
23	11/06 - 30/06	Management		all of members	

b. **Challenges & Solution for implementation phase:**

No.	Challenges Title	Challenges Description	Solutions	Explain for solutions
1	Change Management	The project faces constraints related to budget, time, or skilled personnel.	Prioritizing essential tasks and using project management tools	Prioritizing project requirements and applying agile methodologies can help manage these limitations effectively. Utilizing project management tools to optimize resource allocation and streamline processes can also be beneficial
2		Changes in project scope or objectives can lead to project scope creep, increasing costs and timelines.	Documenting requirements clearly and enforcing strict change management protocols	Clear documentation of requirements and scope, along with strict change management procedures, can control scope creep. Regular reviews and approvals for any changes are essential to keep the project aligned with its original goals.
3	Stakeholder Alignment	Misalignment among stakeholders regarding project goals and visions can lead to conflicts and delays.	Maintaining regular communication and stakeholder engagement	Regular communication, transparent decision-making processes, and stakeholder management strategies are vital. Workshops and regular update meetings can also ensure everyone remains on the same page.
4	Regulatory Compliance	Projects, especially in industries like healthcare or finance, may face strict regulatory requirements.	Involving compliance experts early and conducting regular audits	Engaging compliance experts early in the project to guide the development process and ensure all requirements are met can prevent future issues. Continuous monitoring and audits during implementation can also help.
5	User Adoption	Resistance from users in adopting new systems or technologies is a common issue.	Providing thorough training and involving users early in the process	Providing comprehensive training sessions, user-friendly documentation, and responsive support can ease the transition for users. Involving users early in the development process through feedback loops and beta testing can also increase buy-in.
6	Quality	Ensuring the quality	Implementing	Implementing robust testing phases, including

	Assurance	of the final product meets expectations can be challenging, particularly with complex projects.	comprehensive testing phases and maintaining high standards	unit testing, integration testing, and user acceptance testing, is crucial. Automating tests and maintaining high standards throughout the development lifecycle can help maintain quality.
7	Technology Integration	Integrating new technologies with existing systems can be complex and fraught with technical difficulties.	Conducting compatibility tests and using middleware or adapters	Conducting thorough compatibility tests and incremental implementation can mitigate risks. Employing middleware or developing custom adapters can also bridge the gap between old and new systems.

5. Midterm Report and Feedback:

a. Summary:

1. **Introduction:**
   - overview and objectives of the project.
   - Briefly summarize.
2. **Implementation Details:**
   - The technologies, programming languages, frameworks, and tools used.
   - Design patterns or software engineering principles
3. **Testing and Evaluation:**
   -Types of tests conducted, such as unit tests, integration tests, and system tests.
4. **Progress and Challenges:**
   - Highlight any significant milestones achieved in the project timeline table.
   - Challenges and solutions during the implementation phase.
5. **Midterm Report and Feedback:**
   - Summarize the contents of the midterm report.
   - The key points presented to the project stakeholders during the presentation.
   - Feedback received from the stakeholders and adjustments made based on their suggestions.
   - Future work or improvements.
6. **References**

b. The key points presented to the project stakeholders:

The Midterm Report focuses on highlighting the implemented portion of each member as well as the many languages used in the project. The report also brings out the objectives that our developing team wants to showcase, namely: Convenience, Accessibility, Transparency, Customer Satisfaction, and Efficiency. It drives us further into implementation details which include the tools and technologies that are used and the design pattern. Lastly, testing and evaluation come along with progress and challenges explaining the issues we face and how long each portion is to be finished.

These are the key points attracting the customers that we have conducted:

- **Goal**: Provide convenient and trustworthy automotive services directly to customers' homes, addressing time constraints and inconveniences.
- **Function**: Personalized user profiles, easy service booking, seamless payment integration, continuous service improvement.

- **Advantages**: Customers enjoy doorstep service, hassle-free booking and payments, consistent quality improvements, and transparent, reliable experiences.
- **Special Features**: Tailored user experiences, smooth payment process, ongoing service enhancements, and a focus on accessibility and transparency for customer trust and satisfaction.

Remember the key points are calculated and considered from what our team has offered in the project.

c. Feedback:

Stakeholders appreciated the notion of doorstep service and the simplicity with which services could be selected. Furthermore, some people proposed updating the user interface to facilitate navigation. Changes included improving the user interface, adding a search option for faster service discovery, and streamlining the checkout process.

d. Future work or improvements:

We can enhance user experience (UX) enhancement. In addition, implement a personalized dashboard for users to track their service history, upcoming appointments, and recommendations.

6. References:

1. https://scoredigital.io/articles/why-is-vue-js-framework-still-a-growing-player-in-the-frontend-market
2. https://swimm.io/learn/system-design/the-top-7-software-design-patterns-you-should-know-about
3. https://www.linkedin.com/posts/milan-jovanovic_%3F%3F%3F%3F-%3F%3F%3F%3F%3F-%3F%3F%3F-%3F%3F%3F%3F%3F%3F%3F-activity-7175762992548958208-H6it
4. https://www.lambdatest.com/learning-hub/react-end-to-end-testing
5. https://www.hotjar.com/customer-satisfaction/challenges-and-solutions/
6. https://www.w3.org/WAI/planning/involving-users/