**APPOINTMENT SCHEDULER APPLICATION PROJECT**

**Developer Meeting Report**

**Meeting Date:** 27/03/2024

**Participants**

|  |  |
| --- | --- |
| **Customer Team** | **Developer Team** |
| Hatice Sebla Karabunar | Abdel Aziz Mounther Qudeih |
| Ali Berke Devecioğlu | Ahmed Alidlbi |
| Tahsin Emre Telli | Mohamad Malek Alkhodary |

Haidar Kahla

**Objective**

In the meeting, we reviewed a portion of the UI/UX design, with ongoing evaluation of the remaining sections. Our development is based on the MERN Stack, which includes ReactJS, Node.js, Express.js, and MongoDB. Additionally, we have created two visualization diagrams.

**Key Discussion Points**

* **UI/UX DESIGN:**

The client has reviewed a portion of the UI/UX design, and we are currently evaluating the remaining sections. And provided the “Figma” link to view the design step by step.

* **TECHNOLOGY STACK:**

**MERN Stack**

* **React**: We use ReactJS for its efficient, flexible, and reusable components, which streamline the development of interactive user interfaces.
* **NodeJS**: We use Node.js because it enables scalable, server-side applications and efficient data-intensive real-time functionality across distributed devices.
* **ExpressJs**: Express.js is used for its minimalist structure that simplifies the development of robust web applications and APIs with optimal performance and routing capabilities.
* **MongoDB**: MongoDB is used for its flexible schema design, which allows for easy storage and retrieval of large volumes of structured and unstructured data in a highly scalable fashion.
* **Development:**

We have initiated development, initialized the repository, and designed the dashboard. Backend implementation is underway, with step-by-step progress. Additionally, we reviewed a portion of the UI/UX design, with ongoing evaluation of the remaining sections. Our development is based on the MERN Stack, which includes ReactJS, Node.js, Express.js, and MongoDB. Additionally, we have created two visualization diagrams.

* **DIAGRAMS:**

We create diagrams for our applications to visually represent system architecture and workflows, enhancing understanding and facilitating better communication among team members and stakeholders. These diagrams also aid in identifying potential issues and streamlining the development process.

1. **Use Case Diagram:**

Use case diagrams visually represent user interactions with a system, clarifying requirements and functionalities.

Une image contenant diagramme, croquis, dessin, ligne

Description générée automatiquement

Use case Diagram

1. **Activity diagram:**

Activity diagrams are used to model the workflow and sequence of operations in a system, showing the dynamic aspects of information flow and decision paths.

Une image contenant diagramme, texte, Dessin technique, Plan

Description générée automatiquement

Activity diagram

1. **Class diagram:**

Class diagrams are used to visually represent the structure of a system by illustrating its classes, attributes, operations, and the relationships between objects.

Une image contenant texte, capture d’écran, Police, conception

Description générée automatiquement

Class diagram

**Next meeting agenda: 7/5/2024**

1. Complete UI interface and review client
2. Specify requirements and interactivity in the page.
3. Check the features implemented in the app.