

---

**Analysis and Design of User Interface Private Tutor Mobile Application with Agile System Development Methodologies****Jansen Wiratama<sup>1</sup>, Michelle Melody<sup>2</sup>, Obie Krisnanto<sup>3</sup>, Christian Johan<sup>4</sup>**[Jansen.wiratama@umn.ac.id](mailto:Jansen.wiratama@umn.ac.id), [michelle.melody@student.umn.ac.id](mailto:michelle.melody@student.umn.ac.id),[obie.krisnanto@student.umn.ac.id](mailto:obie.krisnanto@student.umn.ac.id), [christian.johan@student.umn.ac.id](mailto:christian.johan@student.umn.ac.id)<sup>1, 2, 3, 4</sup> Universitas Multimedia Nusantara

---

**Article Information**

Submitted : 30 Nov 2023

Reviewed: 6 Des 2023

Accepted : 30 Des 2023

---

**Keywords**Agile system  
development, Mobile  
application, Private  
tutor, UML, User  
interface.

---

**Abstract**

Wijaya Learning Center in Gading Serpong faced challenges with low course participation due to an inefficient registration system, and to overcome these challenges, a UI analysis and design were conducted, leading to the development of a mobile application. This application allows clients to easily access class information, register online, and make payments. The backend, designed for the owner, supports schedule checks, transaction monitoring, and teacher management tasks. Class schedules, entered by the owner, remain consistent with changes permitted once a month. Clients can select their preferred schedule through the user interface. The study utilized Agile system development methodologies, known for their success in various fields. The research concludes that Agile methodologies significantly streamline the application design process, resulting in a functional mobile application that meets user requirements. Rigorous testing confirms the smooth usability of the application's four main functions, enhancing the private tutoring experience at Wijaya Learning Center.

---

## **A. Introduction**

Private tutoring is a learning activity carried out by students outside academic hours to improve their understanding of academic and non-academic material [1]. Private tutors can work in various fields, such as music courses, school lessons, sports, driving, languages, and cooking [2], [3]. Wijaya Learning Center is a course institution that aims to help students or the general public who want to study, especially in music, science, social sciences and languages in the Gading Serpong area, Tangerang. Wijaya Learning Center provides music courses for various instruments, from piano, violin, cello, and guitar to drums. Apart from that, there is also study guidance (bimble) for Elementary Schools (SD), Middle Schools (SMP), and High Schools (SMA), which is divided into Mathematics and Social Sciences tutoring, English language courses, and Mandarin language courses.

Even though it is pretty popular in the area, there are not many participants at the Wijaya Learning Center; one of the reasons is that there is no media available to register for private tutors, resulting in potential participants (clients) who are interested in registering to have difficulty accessing the registration form and viewing what classes are available. To overcome the obstacles related to the registration process at Wijaya Learning Center, a UI analysis and design were carried out to make it easier for clients to see the available classes and register online without having to come directly to the course. Clients can view and select available schedules through the application. After that, the client can also continue to make payments to confirm the registration in the application. This application also has a back-end display specifically for the Wijaya Learning Center owner to make it easier for managers. Wijaya Learning Center is managed by the owner, where the owner can monitor the registration process. Processes that the owner can carry out include checking schedules, monitoring transactions, creating, updating and deleting teachers, and printing total transaction reports per month.

The owner of the Wijaya Learning Center can input class schedules according to the duration of teaching and learning activities that the owner and teachers have determined. This schedule can only be changed once a month to maintain schedule consistency. The schedule that the employee has input will appear on the User Interface (UI) of the client, who is the application user who will register for the course. After that, the client can choose the available schedule according to his wishes. Based on several previous studies, quite good results were obtained in applying system design methods [4]. Several studies show that Agile system development can be used to design website-based, Android-based, and desktop-based applications. Besides, Agile system development has been successfully implemented in several fields [5], [6], such as health [7], education [8], [9], e-commerce [10], and MSMEs [11], [12]. Therefore, this research will use Agile system development as the methodology.

## **B. Research Method**

The foundation of agile system development is on an incremental and iterative methodology [13]. Agile techniques diverge from extensive initial preparation by embracing adaptable requirements and promoting continuous input from end users. Cross-functional teams collaborate on successive product versions within a specific time frame. This work is structured into a prioritized

backlog, with the order determined by its business or consumer significance [14]. The objective of each iteration is to produce a functional product. Collaboration between business stakeholders and developers is necessary to ensure the product aligns with consumer requirements and corporate objectives [15], [16]. Because this research only focuses on analysis and design, applying the Agile System Development method is carried out until the development stage. So, the release, tracking, and monitoring stages will be carried out in future research.

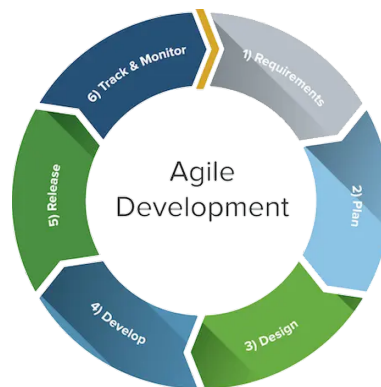


Figure 1. Agile System Development Methodology

#### **A. Requirements**

This phase involves many meetings with managers, stakeholders, and users to identify business requirements. The team needs to gather information like who will use the product and how they will use it. These requirements must be quantifiable, relevant, and detailed.

#### **B. Planning**

Once an idea is deemed viable and feasible, the project team comes together and works to identify features. This phase aims to break down the idea into smaller pieces of work (the features), theorize each feature and assign it to an iteration.

#### **C. Design**

The system and software design is prepared from the requirements identified in the previous phase. The team needs to think about what the product or solution will look like. The test team also develops a test strategy or plan to proceed.

#### **D. Develop**

This phase involves creating and testing features and scheduling deployment iteration (following the iterative and incremental development approach [IID]). The development phase starts with iteration 0 because no features are being delivered. This iteration lays down the foundation for development, with tasks like finalizing contracts, preparing the environments, and funding.

#### **E. Release**

Once the code has been developed, it is tested against the requirements to ensure the product solves customer needs and matches user stories. Unit testing, integration testing, system testing, and acceptance testing are done during this phase.

#### **F. Track and Monitor**

After testing, the product is delivered to customers for them to use. However, deployment isn't the end of the project. Once customers start using the product, they may encounter new problems that the project team must address.

### C. Result and Discussion

User requirements explain the requirements or end-user requirements for a system. The requirements definition is done with users knowing how to use the system created. Most user needs or requirements deal with how the user interacts with the system and what the user expects. The following are the user requirements for the Wijaya Learning Center.

**Table 1.** User requirements list's

No	User Requirements		
	Client	Owner	Instructor
1	Register and log in to your user account.	Manage instructor data.	Determine the teaching schedule.
2	View the course schedule.	Monitoring registration and available classes.	View the teaching schedule.
3	Choose a course class.	View transaction reports.	
4	Select the instructor and course schedule.		
5	Register for the selected course.		
6	Attach proof of payment.		

After the user requirements have been successfully created in Table 1, a use case diagram is created using UML [4]. The draw.io tools are used in creating use case diagrams. Use case diagrams are displayed with symbols by the object-oriented analysis and design approach.

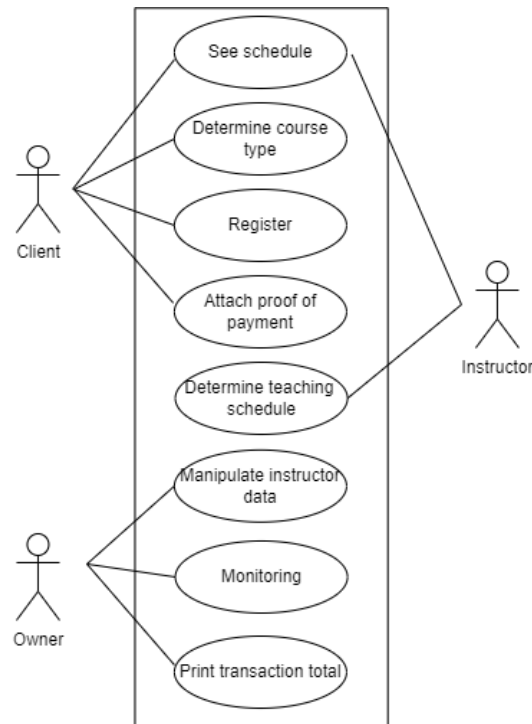


Figure 2. Private tutor use case diagram

Figure 2 contains a comprehensive use case diagram illustrating the fundamental functionalities of the envisioned Wijaya Learning Center application. The diagram intricately outlines fundamental interactions and roles within the system. The diagram encapsulates a user-friendly experience for clients, depicting their ability to view the schedule seamlessly, identify various course offerings, and engage in the registration process [17]. This registration process encompasses essential details such as selecting preferred time slots, schedules, and instructors, providing a streamlined and personalized enrollment journey. Clients are also empowered to submit proof of payment through this intuitive system, enhancing the efficiency of transactional processes.

On the instructor's front, the use case diagram elucidates their role in viewing schedules and determining their teaching schedules. This functionality ensures instructors have clear visibility into their responsibilities and commitments, fostering effective time management. Simultaneously, the owner is equipped with elevated administrative capabilities. The diagram illustrates the owner's ability to manipulate teacher data, streamlining the process of managing instructor information. Furthermore, the owner possesses the capacity to monitor and generate comprehensive reports on total registration transactions. This strategic oversight not only facilitates data-driven decision-making but also enables the owner to assess the overall performance and growth of the learning centre. Including a printing feature further emphasizes the importance of tangible documentation in the monitoring and record-keeping, ensuring a robust and accountable management system for the Wijaya Learning Center.

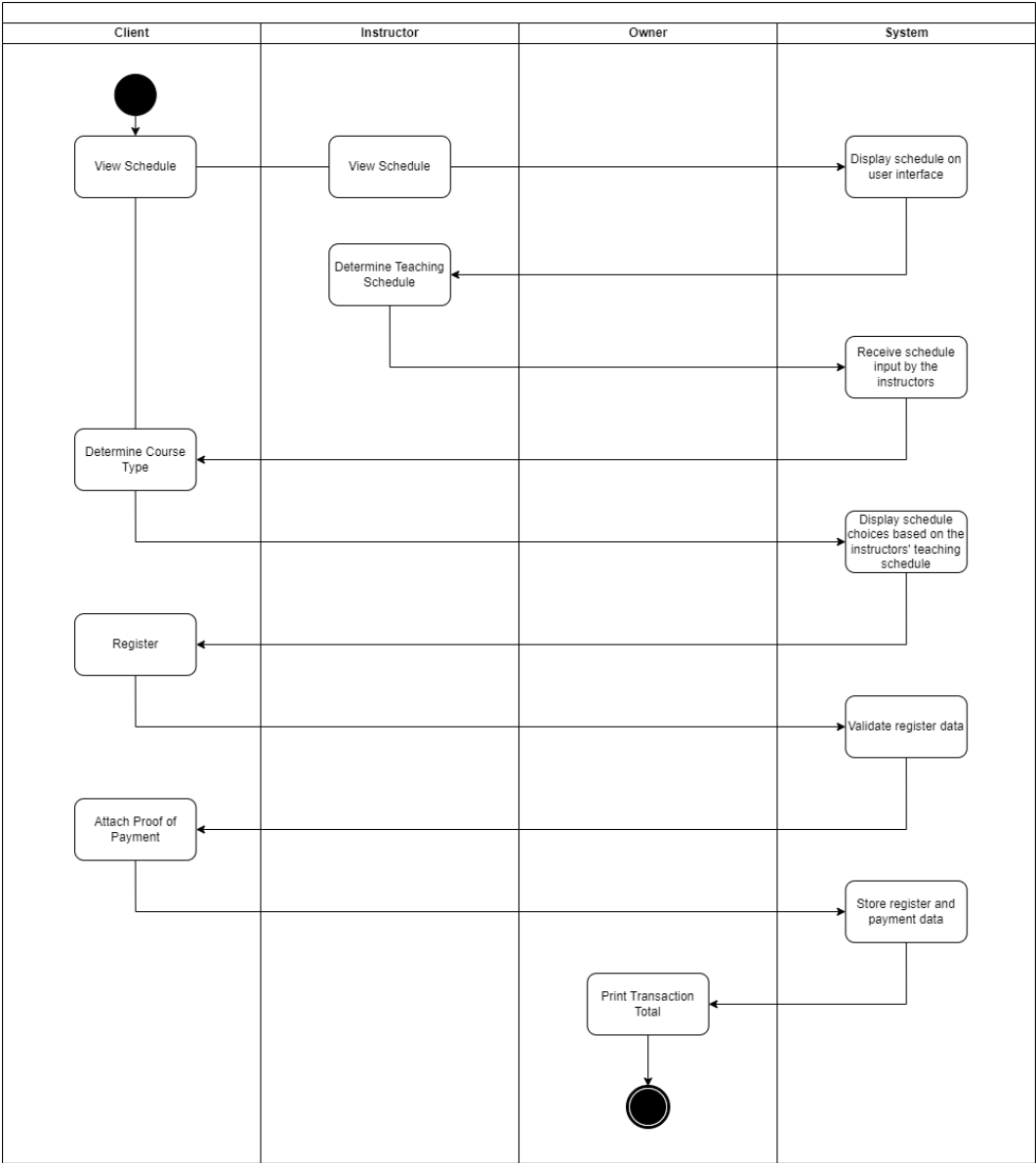


Figure 3. Private tutor activity diagram

The activity diagram in Figure 3, derived from Figure 2's use case diagram, visually represents the step-by-step flow of activities within the Wijaya Learning Center application. This diagram simplifies the understanding of how users (clients, instructors, and the owner) interact with the system by graphically outlining each component's operations. It serves as a clear guide, detailing the sequential processes involved in using various features of the application. The activity diagram acts as a visual roadmap, making it easier for both developers and users to comprehend the user-system interactions, from initial engagement to administrative tasks, enhancing transparency in the application's functionality.

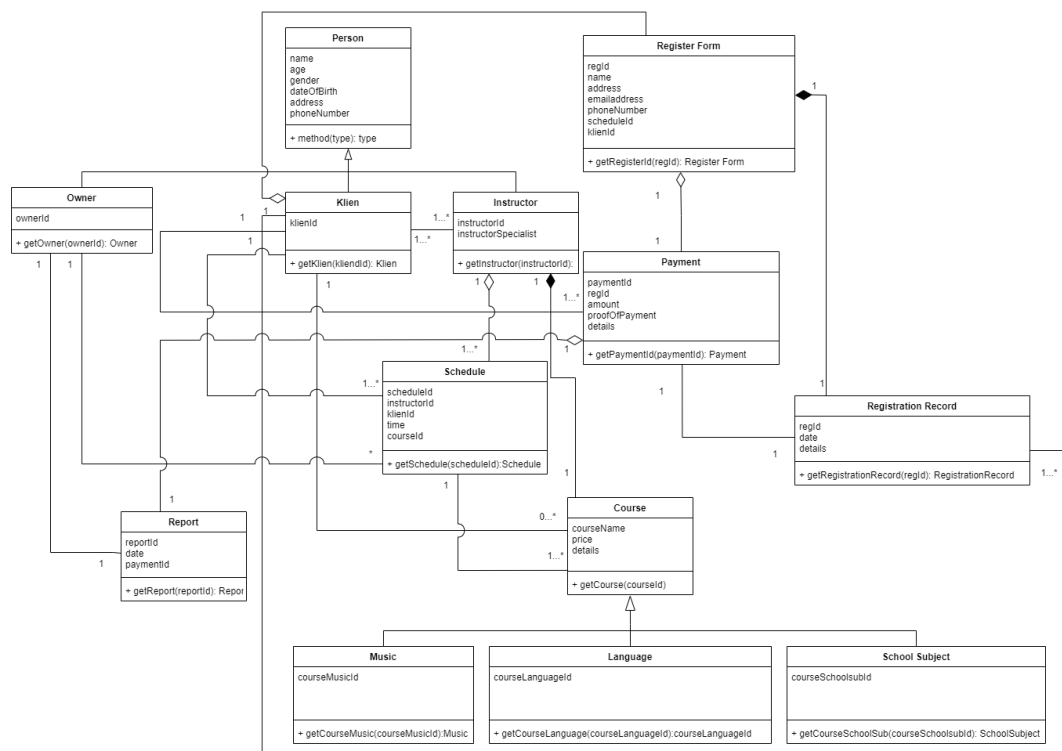


Figure 4. Private tutor class diagram

Apart from creating a use case diagram in Figure 2 and an activity diagram in Figure 3, a class diagram was also created to identify the classes in the designed application. The accuracy of the analysis can be seen in the Class Diagram in Figure 4, which shows each class in the application. The difference between class diagrams before and after system design lies in the value of each class attribute, which is more straightforward thanks to the support of system use. The following is a class diagram for the Wijaya Learning Center after the system was designed.

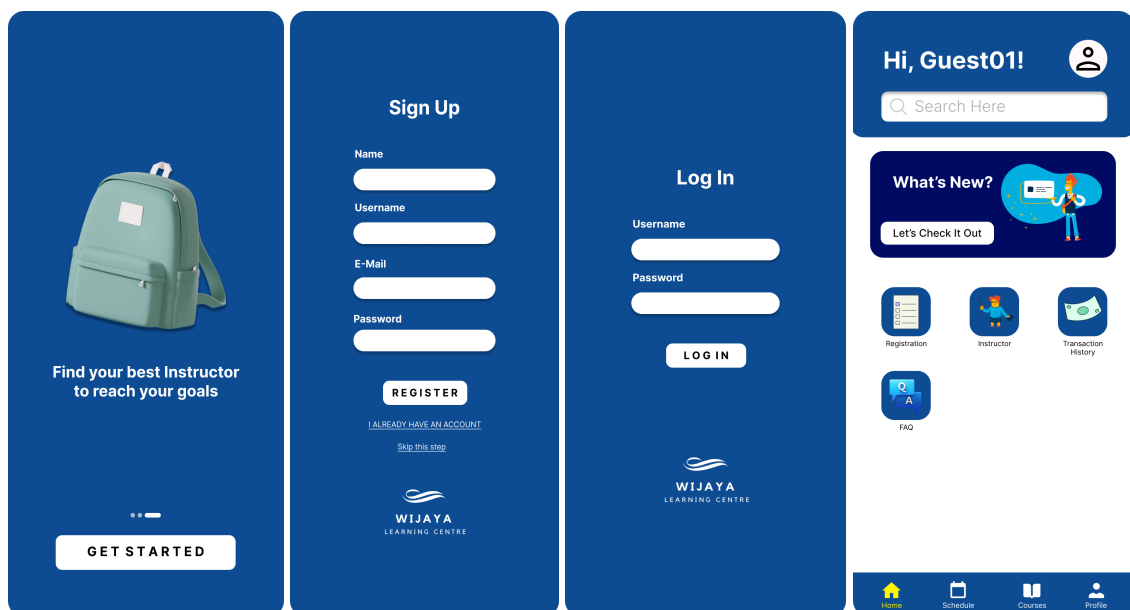


Figure 5. Landing page, register, login, and dashboard User Interface

At the design stage in the Agile system development methodology, the application user interface was designed using FIGMA tools. This user interface design is a prototype for application coding so that programmers can visually see the user interface display, thus making the coding stage easier. The coding stage is carried out using the Flutter programming language.

Figure 5 is the landing page or welcome page for the Wijaya Learning Center application. The UI in Figure 5 is the first display on the user's screen when opening the application. Figure 5 displays four UIs regarding the landing page, sign-up page, login page, and user dashboard. After selecting "Get Started", the user will be directed to the sign page. If the user already has an account, the user can press "I already have an account" to be directed to the login page. Four features can be accessed via the dashboard: course registration, viewing available teachers, viewing transaction history or course registration, and FAQ. There are four icons on the navigation bar (bottom part): home, schedule, course, and profile.

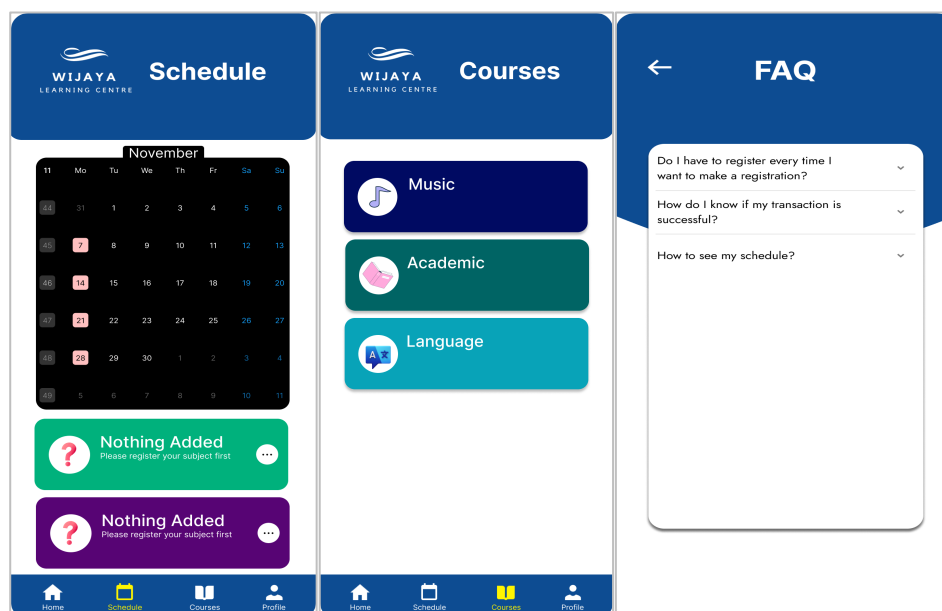


Figure 6. Schedule, Courses, and FAQ User Interface

Figure 6 presents a detailed glimpse into the user interface (UI) of the client schedule, course menu, and FAQ page, providing a comprehensive view of the application's functionality. The schedule section prominently features an interactive calendar, serving as a dynamic hub for course schedule information. Clients can seamlessly navigate through dates and gain insights into upcoming sessions, fostering an organized and user-friendly experience. Within the courses section, clients are empowered to choose from various courses, each elegantly presented with relevant details. This intuitive design not only facilitates easy course selection but also enhances the overall accessibility of the application. Additionally, the FAQ page is a valuable resource, offering a compiled list of Frequently Asked Questions tailored to clients. This section aims to address common queries and provide swift solutions, contributing to an enriched user experience by ensuring clients can readily access relevant information and support.



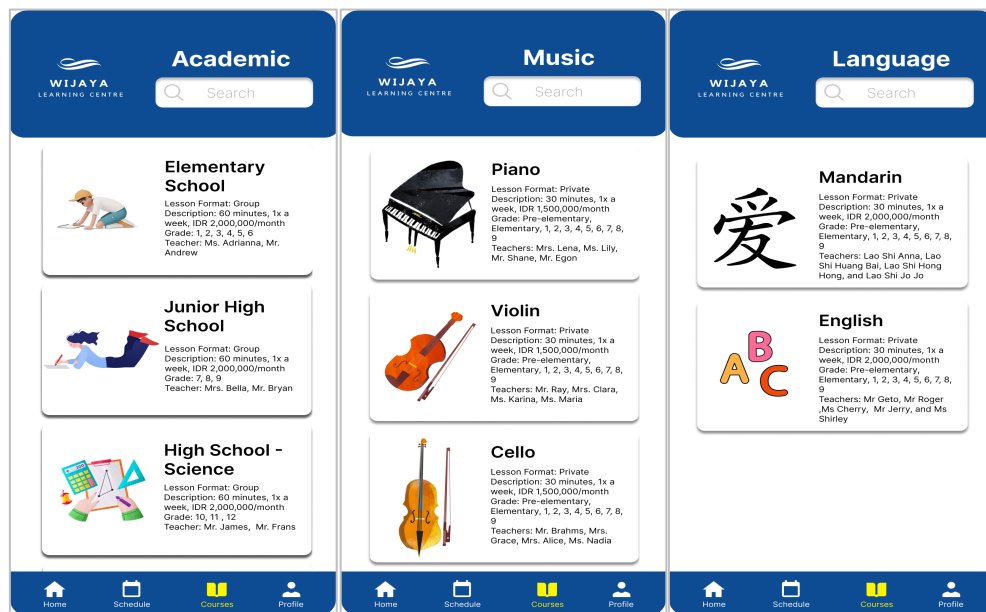


Figure 7. Courses list (Academic, Music, and Language) User Interface

Figure 7 displays the courses page in the Wijaya Learning Center application. There are three types of courses, namely music, academics and languages. The instruments available in music courses are piano, violin, guitar, cello and drums. Meanwhile, academic courses are divided into elementary school, middle school, high school (MIPA), and high school (Social); then, for language courses, English and Mandarin classes are available.

	Monday	Tuesday
13:00	Mr. Shane	Mr. Shane
13:30		
14:00	Mr. Shane	Mr. Shane
14:30		
15:00	Mr. Shane	Ms. Lily
15:30		
16:00	Mr. Shane	Ms. Lily
16:30		
17:00	Mr. Shane	Ms. Lily
17:30		
18:00		
18:30	Mrs. Lena	Mrs. Lena
19:00		
19:30	Mrs. Lena	Mrs. Lena
20:00		

Figure 8. Registration form, registration data, and instructor schedules User Interface

Figure 8 shows the course registration UI display, which starts with inputting data, selecting the type of course, selecting the schedule and instructors available for that type of course, and attaching proof of payment, which can be seen in Figure 9. After the entire registration process is successful, the schedule will be displayed. Set on the client calendar.

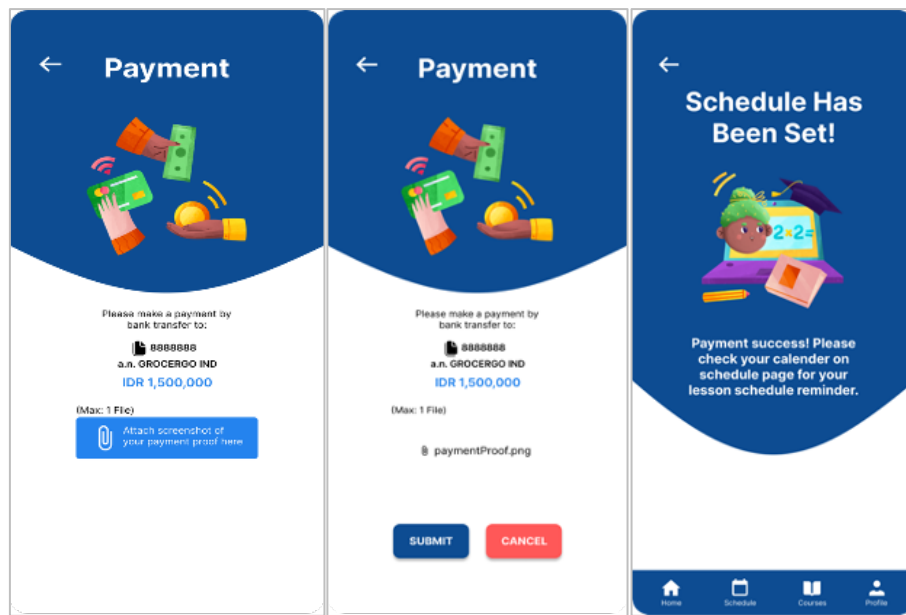


Figure 9. Payment submission and payment notification User Interface

In the final stage of this research, an evaluation was carried out using the Functional Testing to validate the function of the application that had been designed. This evaluation was conducted by conducting direct testing by application users, namely owner, instructor, and four client. Testing is carried out using an Android Smartphone.

**Table 1.** Functional testing

No	Features	Result
1	Register client and instructor account	Passed
2	Select courses	Passed
3	Make a payment	Passed
4	View schedule	Passed

The main features evaluated are log-in/log-out, select courses, make a payment, and view schedule. The results of the functional test using show that the four main functions of this application can be used according to their intended purpose. Even though the results of this test show that the four main functions of this application run smoothly, further experiments still need to be carried out by comparing with several other test methods so that the test results are more certain.

#### D. Conclusion

The conclusions obtained in this research indicate that applying Agile System Development Methodologies in designing the Wijaya Learning Center Application can simplify the application analysis and design process. The design result is a mobile application-based application that runs smoothly according to the user requirements obtained at the initial stage. The functional tests that have been

carried out validate that the four main functions of this application can be used without problems by users.

### E. Acknowledgment

We would like to acknowledge the invaluable support received from the Information Systems Department at Universitas Multimedia Nusantara. Their contribution played a significant role in the successful completion of this research. We extend our sincere gratitude to our colleagues from the LPPM Department, whose insightful input and expertise greatly assisted us throughout the research process. Their valuable contributions helped shape the direction of this study and enhance its overall quality.

### F. References

- [1] H.-A. Dang and F. H. Rogers, "The Growing Phenomenon of Private Tutoring: Does It Deepen Human Capital, Widen Inequalities, or Waste Resources?," *World Bank Res. Obs.*, vol. 23, no. 2, pp. 161–200, Oct. 2008, doi: 10.1093/wbro/lkn004.
- [2] O. Kozar, "The Face of Private Tutoring in Russia: Evidence from Online Marketing by Private Tutors," *Res. Comp. Int. Educ.*, vol. 8, no. 1, pp. 74–86, Jan. 2013, doi: 10.2304/rcie.2013.8.1.74.
- [3] X. Zheng, C. Wang, Z. Shen, and X. Fang, "Associations of private tutoring with Chinese students' academic achievement, emotional well-being, and parent-child relationship," *Child. Youth Serv. Rev.*, vol. 112, p. 104934, 2020, doi: <https://doi.org/10.1016/j.childyouth.2020.104934>.
- [4] D. Dennis, Alan; Wixom, Barbara; David; Tegarden, *Systems Analysis and Design: An Object-Oriented Approach with UML*, Sixth. United States: Wiley, 2020. [Online]. Available: <https://umnlibrary.vitalsource.com/books/9781119561217>
- [5] L. Cordeiro, R. Barreto, R. Barcelos, M. Oliveira, V. Lucena, and P. Maciel, "Agile Development Methodology for Embedded Systems: A Platform-Based Design Approach," in *14th Annual IEEE International Conference and Workshops on the Engineering of Computer-Based Systems (ECBS'07)*, 2007, pp. 195–202. doi: 10.1109/ECBS.2007.16.
- [6] T. G. Gonçalves, C. Kolski, K. M. de Oliveira, G. H. Travassos, and E. G.-L. Strugeon, "A Systematic Literature Review on Intelligent User Interfaces: Preliminary Results," in *Adjunct Proceedings of the 31st Conference on l'Interaction Homme-Machine*, 2019. doi: 10.1145/3366551.3370344.
- [7] G. Porruvecchio, G. Concas, D. Palmas, and R. Quaresima, "An Agile Approach for Integration of an Open Source Health Information System BT - Agile Processes in Software Engineering and Extreme Programming," 2007, pp. 213–218.
- [8] R. Bernhaupt, C. Martinie, P. A. Palanque, and G. Wallner, "Human-Centered Software Engineering: 8th IFIP WG 13.2 International Working Conference, HCSE 2020, Eindhoven, The Netherlands, November 30 – December 2, 2020, Proceedings," *Human-Centered Softw. Eng.*, 2020, doi: 10.1007/978-3-030-64266-2.
- [9] J. Wiratama, H. Santoso, and Clairence, "Developing a Class Scheduling

- Mobile Application for Private Campus in Tangerang with the Extreme Programming (XP) Model,” *G-Tech J. Teknol. Terap.*, vol. 7, no. 2, pp. 484–493, 2023, doi: <https://doi.org/10.33379/gtech.v7i2.2288>.
- [10] S. You-qun, “Agile development of online order management system based on Oracle HTML DB,” *Comput. Eng.*, 2007.
- [11] W. Prabowo and C. Wiguna, “Sistem Informasi UMKM Bengkel Berbasis Web Menggunakan Metode SCRUM,” *J. MEDIA Inform. BUDIDARMA*, 2021, doi: 10.30865/MIB.V5I1.2604.
- [12] T. A. D. P. Subianto, J. Wiratama, and F. A. Halim, “The Development of web-based Cashier and Inventory Information Systems using Prototyping Model on Micro, Small, and Medium Enterprise (MSMEs) in Indonesia,” *JOINS (Journal Inf. Syst.*, vol. 8, no. 1, pp. 80–89, 2023, doi: 10.33633/joins.v8i1.7983.
- [13] P. M. Institute, *Agile Practice Guide*. Project Management Institute, 2017. [Online]. Available: <https://books.google.co.id/books?id=Nzc2DwAAQBAJ>
- [14] I. G. Stamelos and P. Sfetsos, *Agile Software Development Quality Assurance*. Information Science Reference, 2007. [Online]. Available: <https://books.google.co.id/books?id=TWscMx09ENkC>
- [15] linkedin.com (Firm), *Agile Software Development: Extreme Programming*. linkedin.com, 2019. [Online]. Available: <https://books.google.co.id/books?id=oXoWzgEACAAJ>
- [16] S. F. Wijaya, H. Prabowo, R. R. Kosala, and M. Sicad, “Development of an Agile ERP Framework for Implementation: A Systematic Literature Review,” *Int. J. Control Autom.*, vol. 12, no. 5, pp. 1–12, 2019, doi: 10.33832/ijca.2019.12.5.01.
- [17] A. H. Makalalag, Y. A. Ekawardhani, and T. V. L. Gaol, “User Interface/User Experience Design for Mobile-Based Project Management Application Using Design Thinking Approach,” vol. 4, pp. 269–274, 2021, doi: 10.5281/ZENODO.5055189.