e-ISSN: 2614-8404
p-ISSN:2776-3234

Smart-Working and Hot Desking Application Development using Agile and Extreme Programming Method at Xtra Cowork

Daeng Ahmad Nurdin[1*)], Dodik Firmansah[2], Akrom Hafifi[3], Muhammad Darwis[4]
[1,2,3,4]Program Studi Teknik Informatika, Fakultas Ilmu Rekayasa, Universitas Paramadina
Email: [1]daeng.nurdin@students.paramadina.ac.id, [2]dodik.firmansah@students.paramadina.ac.id,
[3]akrom.hafifi@students.paramadina.ac.id, [4]muhammad.darwis@paramadina.ac.id

Abstract – The current condition of working desks often faces challenges such as poor management and inefficiency in space utilization, which complicates the adoption of Smart-Working culture at Xtra Cowork. To address these issues, the writer aims to facilitate the transition for offices adopting Smart-Working culture by designing a hot desking supporting mobile app. Through observation, the research identifies the need for a reliable system to support hot desking practices. To complete this research, the writer uses agile and extreme programming methods. For app designing, the writer uses UML. The main tools used for developing the app are Flutter with Dart language. The developed mobile app successfully fulfills these requirements. The results of the application testing using the black box method has passed 100% of the test cases. It shows that all functions have worked as designed and could be used by the users as intended. Future researchers can build upon this work to address other Smart-Working culture needs.

Keywords – mobile apps, extreme programming, hot desk apps, black-box method

I. INTRODUCTION

Xtra Cowork is an initial of the real office that the writer takes this research on, is adopting Smart-working culture. For almost two years, the company already adopt this culture, but recently this company is facing a new problem on its plan to resize the maximum work seat capacity to only 60%. The problem the company faces is how to maximize the spread of work seat utilization, therefore in this research, writer provide an idea to solve this issue.

Smart-Working by its definition is a framework that gives the employees the freedom to choose the time, place on the way they work [1]. The culture is based on ethos, process and technologies to aims the improve on work environments, productivity and also cost savings [2]. Claimed by Yolyntseva there are several advantages for an office to implement this revolutionizing culture such as enhanced energy efficiency, improved productivity, and seamless connectivity [3]. The aforementioned benefit of the Smart-Working culture has been experienced by the writer as someone who currently work in an office.

The adoption of Smart-Working at Xtra Cowork been followed by the emergence of hot desking concept. Hot desking by definition is a system in a work office where each space is available for each worker, rather than reserved for specific person, so it possible for the same space to be occupied by different person along the day or week [4]. The "Desk" refers to a working spot or space that usually has table in it which shared by multiple persons in the office, therefore the "Desk" word is used. In some case, the initial motivation of hot desking implementation is due to the cost reduction through space saving [5]. As mentioned earlier, the Xtra Cowork also has the plan to reduce its working space. To support the hot desking practices, a well-tailored mobile application would be needed.

A mobile application, commonly referred to as an "app," is a software program specifically designed to operate on mobile devices such as smartphones or tablets [6]. As a statistic data that was brought by the Statista predict that by 2024, the number of smartphone user in Indonesia will reach 194 million users [7]. With that much smart-phone user in Indonesia, the development of a mobile application hot desking support is chosen by the writer. Another reason writer decides to develop a mobile application for the hot desking support is because of its ease-of-use nature that could be accessed anytime and anywhere by the user.

In developing a software, a development method is needed to ensure the software will be well-planed and well-developed to achieve its critical function. The Agile project management methodology will be used in this research as it's adaptive nature that would easily evolves the requirement and the software target as new ideas emerged. Agile project management

e-ISSN: 2614-8404
p-ISSN : 2776-3234

methodology is an iterative approach that emphasizes flexibility, collaboration, and responsiveness to change throughout the project lifecycle [8]. It is a popular alternative to traditional project management methodologies like the waterfall approach, offering a more adaptive and customer-centric way of delivering projects [9]. To be specific, in this research, writer using the Xtreme Programming methodology.

Extreme Programming (XP) is an agile software development methodology that originated from the Chrysler C3 project in 1996 Dybå & Dingsøyr (2008) [10]. XP focuses on twelve core practices for software development, including planning games, short releases, system metaphors, simple design, testing, refactoring, pair programming, collective ownership, continuous integration, maintaining a 40-hour workweek, having on-site customers, and adhering to coding standards [11]. One of the benefits of this approach by using the Agile Development methodology with Extreme Programming is that in the development cycle each team will be enabled to correct each other out, with the development time considered shorter and also quickly adapting with changes of any form without reducing the quality of the software being built [12].

The goal of this research is to develop a well hot desking supporting mobile application that could be used by the employee as user and also for the administrator to have the control over the hot desking regulation.

II. RESEARCH METHODOLOGY

Since the goal of this research is to develop a well-built mobile application that could support the hot desking at Xtra Cowork, the writer uses the research methodology of this paper is by reading the already existing literature and finding additional literature that could support this research by giving the understanding of the theory and concept of a well-built mobile application.

Development methodology that will be used to develop this mobile application is using the Agile Development methodology with Extreme Programming that classified as light software development methodology.

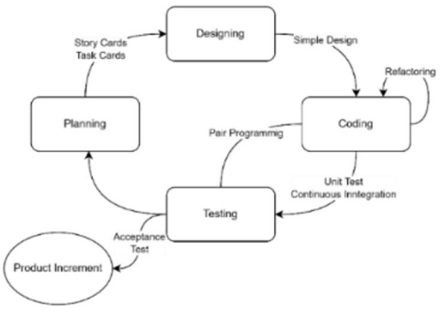

Figure. 1 Extreme Programming Model

As explained by Figure 1, the extreme programming model has it steps that needed to be implemented to ensure a software being developed is well-built and well-planed. The steps of extreme programming development process that will be done in this research are as followings:

A. Planning

In this step, user story creation is being done to ensure the functional requirement is align with the goals of supporting the hot desking process.

B. Designing

In the designing phase, the system flow and database flow will be drawn in UML to ease the coding steps, then the evaluation will be done for the flow to recheck the requirement functionality goals.

C. Coding

In this step, the previously generate flow UML will be implemented into real solution. In this research, we will use the Flutter framework as the base application code, and the SQL Server for the database.

D. Testing

The mobile application that has been coded will be tested using Black-Box testing methods, this method of testing is more focused on functionality specification of a software, tester could define several input conditions in which the test will be done and the expected results of the software.

III. RESULTS AND DISCUSSION

In this section, writer will discuss the implementation of the SDLC methodology of agile development model with extreme programming in the hot desking supporting mobile application design.

3.1 Analysis

The writer done the analysis and data collection using observation and interview methods. The analysis and data collection were being done in order to identify the requirement of this particular mobile app. In this stage, the writer will be presenting the software requirement specification of the mobile app.

The need for software design and development in the hot desking supporting mobile app are as the following:

1. Employee Booking:
   - Employees can easily book a desk for today or tomorrow.
   - Select the desired date and department.
   - View available desks on a floor map and confirm bookings.
2. Admin Dashboard:
   - Admins can upload floor maps and map desks to buttons.

e-ISSN: 2614-8404
p-ISSN : 2776-3234

- Set up departments and associate desks with specific departments.

3. Login:

- Admins and employee need to be authorized before doing actions in the app.

To ensure the employee's needs of hot desking, the app must fulfill the needs based on the first point as shown above. While the second point was aimed to fulfill the admin needs to manage the content change also administration. Lastly the third point was required in order to each user could be authorized and the app variable could adjust based on their roles and user information.

3.2 Planning

For designing, Unified Modelling Language (or UML for short) is being used by writer to describe the system design, flow and its interaction between user and also the system itself. UML is a general-purpose visual modeling language that standardize visualization of system design [13]. In this stage, writer will present how the app works by using the Use Case Diagram and Activity diagram.

1. Use Case Diagram

Use case diagram explain about the interaction between the system and the user(s) [14]. The use case diagram of hot desking supporting mobile app shown below was designed to fulfill its software requirement specification.

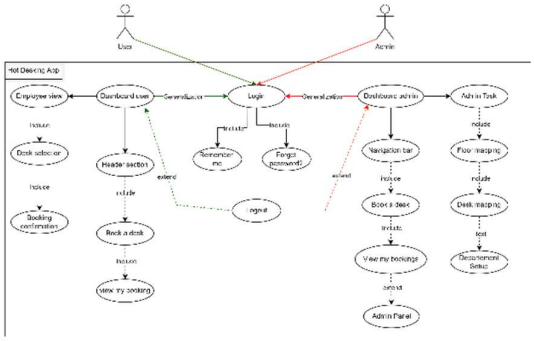

Figure. 2 Use Case Diagram of hot desking supporting mobile app

In figure 2, the admin and user both shared several features (that shown as generalization on the diagram), including the login for authentication and logout to delete the app session. The login features itself was included as it was included in the software requirement specification in the analysis section earlier in the research. Both the user and the admin have their own separated function that does not cross each other as their function was tailored based on their needs and task that each user could do and authorized to do.

The user side functions of the app have to fulfill the needs as described in analysis section, the user must be able to book a desk for the date they choose, the user also able to View available desks on a floor map and confirm bookings.

While the admin's side of function that presented on the figure 2 also has functions that could fulfill its needs based on the software specification requirement on the analysis section earlier, admins can upload floor maps and map desks to buttons. Admin also need to be able to set up departments and associate desks with specific departments.

2. Activity Diagram

Activity diagram was used in the research to explain the flow of the user's interaction with the system, it also based on the unified modelling language which was not constrained by any programming language and measured to could transfer the technical idea [15]. In this research the activity diagram that will be presented will be grouped based on their actor, the user and also the admin.

a. Activity diagram User

There are two activity diagrams drawn to explain the user interaction with system that will be shown in figure 3 and also figure 4. Each of the diagram will present different activity, namely the first one will present the flow of user interaction to book a desk, while the second one will present the flow of how user interact with system to cancel a booking.

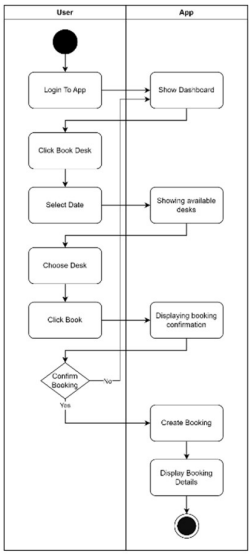

Figure. 3 Activity Diagram User of Hot desking supporting app 1

Figure 3 explaining the flow interaction between user and also the system on how the user will interacting when the user wants to create a desk booking or reservation via the mobile app. The user first must login then initiates the booking by clicking on the book desk menu, the system will response by displaying the right information regarding the user interaction.

e-ISSN: 2614-8404
p-ISSN : 2776-3234

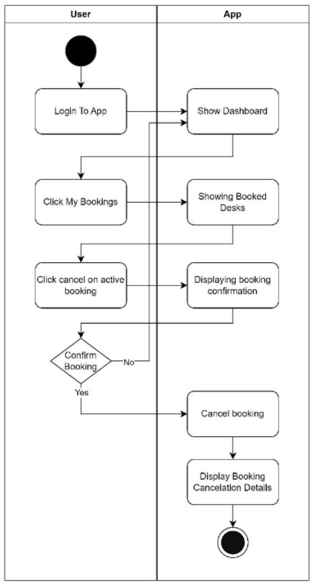

Figure. 4 Activity Diagram User of Hot desking supporting app 2

In figure 4, the user interacting with the system in activity to cancel the booking that has been made earlier by itself. The user initiates the activity by opening the "*my bookings*" menu to then proceed to select the booking he/she want to cancel. The system then will send the confirmation prompt to the user, this step will also prevent unintended input of the system to cancel the bookings.

b. Activity Diagram Admin

Same as the activity diagram prepared for user, the admin's activity diagram also consists of two activity diagram that representing different activity of admin on figure 6 and figure 7. The first diagram will explain the admin interaction with system on activity create new floor layout to be used by users. While the second one aimed to explain the interaction on activity of admin updating the user department setting.

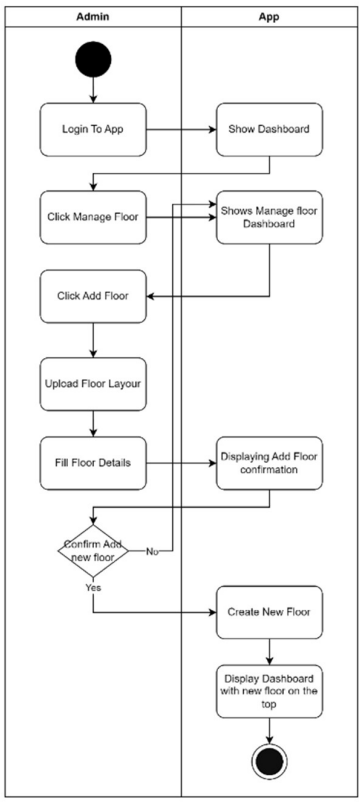

Figure. 5 Activity Diagram Admin of Hot desking supporting app 1

Activity diagram that shown on figure 5 explaining the interaction of admin and system on activity of admin setting up a new floor layout to be used by the users. As shown above, the admin initiates the activity by clicking on *manage floor* menu that exclusively accessed by the admin. As the interaction of the system, the app displaying information as the admin's input on the system, lastly it also notifies the admin about the result of its action.

e-ISSN: 2614-8404
p-ISSN : 2776-3234

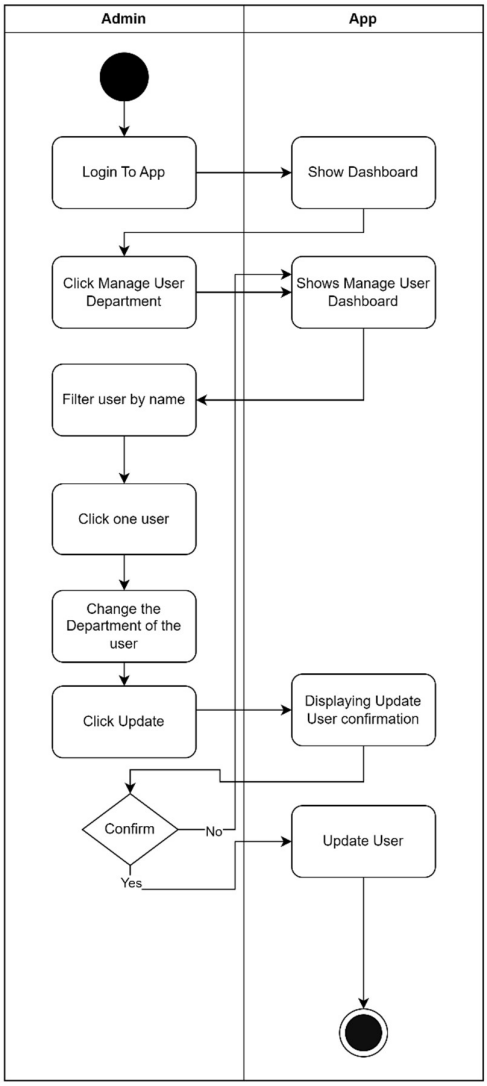

Figure. 6 Activity Diagram Admin of Hot desking supporting app 2

The activity diagram in figure 6 explaining the interaction between admin and system on the activity of admin want to change the department of a user. The activity was initiated by the admin and ended by the system updating the user department information.

3.3 Designing

In the designing step, the writer will explain on how the system will be built based on its classes and how it interacts with each other. To achieve such goal, the writer will be presenting it by using class diagram that also based on unified modelling language. A class diagram could describe the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects [16]. As class diagram function is a fit to explain the system design, therefore the writer will use this diagram for explaining the system design.

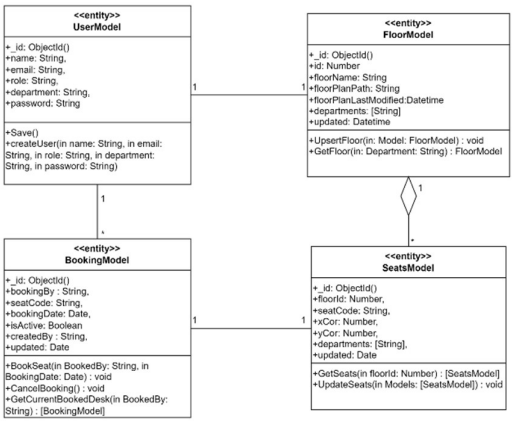

Figure. 7 Class Diagram of Hot desking supporting app

As shown by the figure 7 above, the hot desking supporting mobile app will be consisted of 4 main classes that interact with each other based on their relationship.

The *UserModel* class representing the user profile, it includes the user identifier, name, email, role, department, and also password. The user password will be protected by salting and hashing the password to ensure the secureness of the authorization function of the hot desking supporting mobile app. The *UserModel* could have as many *BookingModel* object, while it only could have 1 *FloorModel* object to determine its booking availability by the user's department.

BookingModel class included to represent the booking that will be made by the user. For each *BookingModel*, it will occupy a *DeskModel* for a timeslot. The timeslot was defined on the *BookingModel* as *bookingDate* property. Therefore, for each day, a desk could only be occupied by one booking.

DeskModel class in the figure 7 representing the bookable desk and also unavailable booking, it will draw itself based on coordination that included as class properties, namely *xCor* and *yCor* properties as shown above.

Lastly the *FloorModel* included in the diagram representing the floor layout and also its mapping based on the user departments.

3.4 Implementing

In this phase, the writer will show the UI of the hot desking supporting mobile app that has been built.

1. Start Page

e-ISSN: 2614-8404
p-ISSN : 2776-3234

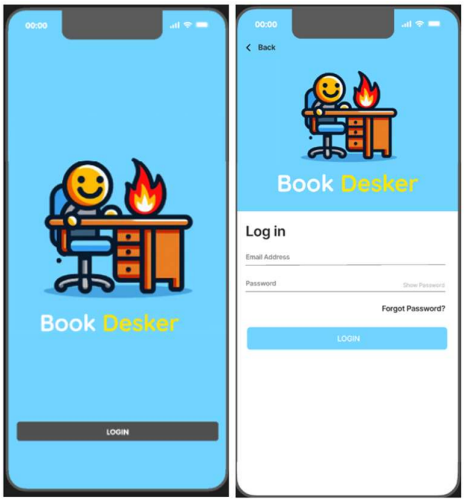

Figure. 8 Start Page UI

Figure 8 shows the start page of the hot desking supporting mobile app. The users will be redirected to this page first as the login page.

2. Home

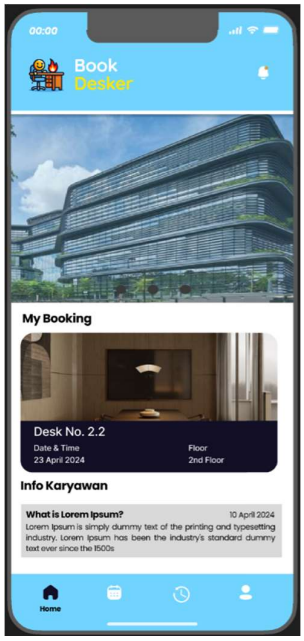

Figure. 9 Home Page UI

After logging in, the user will be redirected to this page as shown in figure 9. Here the user could see the latest information about the company, also it could see the active booking he/she made earlier.

3. Desk Book

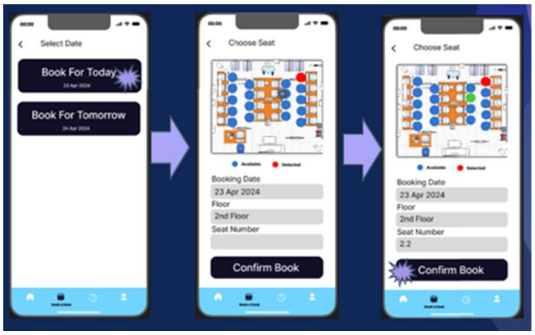

Figure. 10 Desk Book Menu

The figure 10 shows the interface of the hot desking supporting mobile app when a user wants to book a desk.

4. Booking History

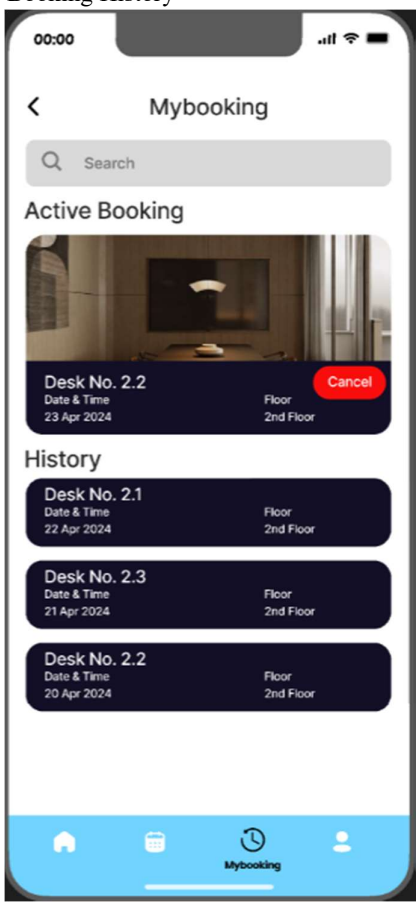

Figure. 11 My Booking Menu

Figure 11 show the UI of Booking History, or as it called in the app is *MyBooking page.*

5. Admin Dashboard

e-ISSN: 2614-8404
p-ISSN : 2776-3234

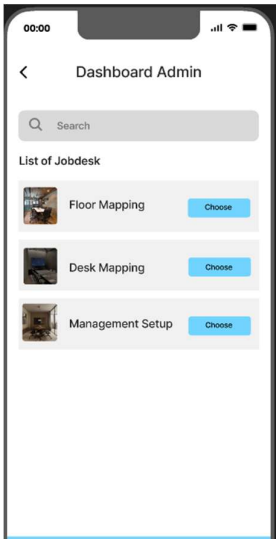

Figure. 12 Dashboard Admin UI

Figure 12 show the UI of admin's task dashboard. Here showed that the admin has privilege for updating maintaining the floor map, desk map, and also the management setup.

6. Floor Mapping

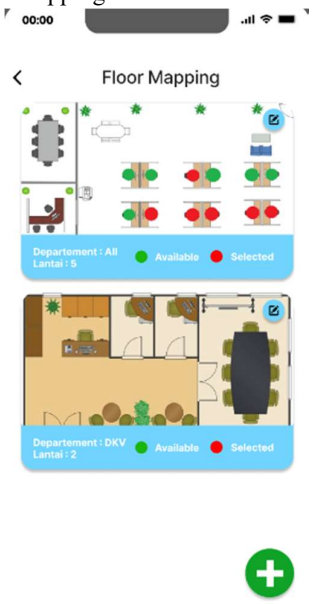

Figure. 13 Floor Mapping UI

The figure 13 contains the UI of floor mapping menu that could be accessed by the admin to manage the floors.

7. Desk Mapping

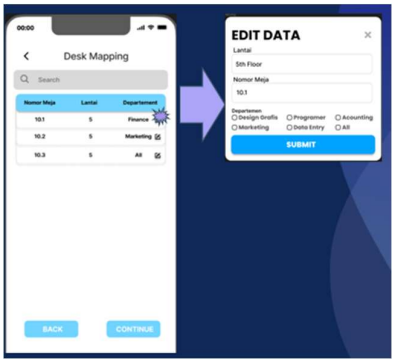

Figure. 14 Desk Mapping UI

Figure 14 shows the UI of and admin that want to edit a mapping of a desk by app.

8. Department Mapping

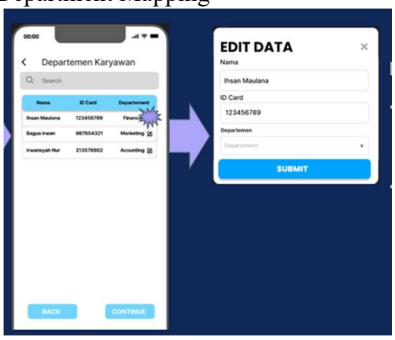

Figure. 15 User Department Mapping UI

Figure 15 shows the hot desking supporting mobile app UI when an admin wants to update a user department information.

3.5 Testing

The hot desking supporting mobile app will be tested by using Black Box testing method. Black Box testing is used as application testing method to obtain the system response and compare it to the expected response based on the system requirement [17]. The result of testing with Black Box Method are shown below in Table 1.

Table 1 Unit Test Results

Unit	System Testing	System Reaction	Test Result	Tested By
Login	1. Fill in the login form 2. Click Login	Redirect to home view	Succeed	User
Create Booking	1. Click Book Desk 2. Select Date 3. Choose desk 4. Click Book 5. Confirm	Booking created, booking details displayed	Succeed	User
Cancel Booking	1. Click My Booking menu 2. Choose a desk to be cancelled 3. Click Cancel 4. Confirm	Booking cancelled	Succeed	User
Create new floor map	1. Click Floor Mapping 2. Click Plus Icon 3. Upload Floor Layout 4. Fill in floor details 5. Click Confirm	New floor map registered	Succeed	Admin
Change a department of a desk	1. Click Desk Mapping 2. Choose a desk to be updated 3. Change department from IT to HR 4. Click Update 5. Click Confirm	Desk department ownership updated	Succeed	Admin

Based on the table 1 above from five test that prepared, the system passed 100% of the test cases. the hot desking supporting mobile app is already passed all the test, and it's ready to be used by masses.

e-ISSN: 2614-8404
p-ISSN : 2776-3234

IV. CONCLUSION

The writer is designing hot desking supporting mobile app in order to ease the transition for offices that want to implement the Smart-Working culture. Based on that reason, the writer in this research analyzing the needs of hot desking supporting mobile app by observation and interviews. One of which the needs come from the hot desking practice that being implemented. The needs of a reliable system to support the hot desking practice in smart working.

Based on the test results, the development of hot desking mobile app has fulfilled the needs that has been described in the analysis stage.

For the next researcher, the writer hopes could develop the mobile app to fulfill the other Smart-Working culture needs.

REFERENCES

[1] L. Marino and V. Capone, " Smart Working and Well-Being before and during the COVID-19 Pandemic: A Scoping Review," *European Journal of Investigation in Health,* vol. 11, pp. 1516-1536, 2021.

[2] A. M. McEwan, Smart Working Creating The Next Wave, London: Routledge, 2013.

[3] Y. Yolyntseva, "Future of Work: Transforming with Smart Offices," 25 Sep 2023. [Online]. Available: https://www.businesstechweekly.com/operational-efficiency/digital-transformation/smart-offices/.

[4] G. Golzen, *Cut the office in half without tears,* The Sunday Times, 1991.

[5] K. Peterson, "Why companies are doing away with assigned desks," 14 November 2014. [Online]. Available: https://www.cbsnews.com/news/why-companies-are-doing-away-with-assigned-desks/.

[6] A. M. a. P. S. Rosado Cruz, "Modern Software Engineering Methodologies for Mobile and Cloud Environments," 2016.

[7] M. Siahaan, "Smartphone users in Indonesia 2018-2028," 2 Nov 2023. [Online]. Available: https://www.statista.com/forecasts/266729/smartphone-users-in-indonesia.

[8] H. Wells and H. Smyth, *An Agile Approach to the Real Experience of Developing Research Methodology,* Routledge, 2024, pp. 181-192.

[9] H. Dong, N. Dacre, D. Baxter and S. Ceylan, "What is Agile Project Management? Developing a New Definition Following a Systematic Literature Review," *Project Management Journal,* 2024.

[10] E. Ermin, R. Soekarta and M. Insan Abdillah, "Utilising QR-Code of Covid-19 Vaccine for Attendance Using Extreme Programming Method," *Antivirus : Jurnal Ilmiah Teknik Informatika,* vol. 17, no. 2, pp. 193-202, 2023.

[11] D. Andriansyah and L. Nulhakim, "Extreme Programming Dalam Perancangan Sistem Informasi Jasa Fotografi," *ICIT Journal,* vol. 7, no. 1, pp. 10-19, 2021.

[12] M. S. a. Y. K. D. I. K. Juliany, "Perancangan Sistem Informasi E-Marketplace Bank Sampah Berbasis Web," *Semin. Nas. Teknol. Inf. dan Multimed,* p. 19–24, 2018.

[13] R. I. J. G. B. James, The unified modeling language reference manual, 1998.

[14] D. S. S. R. E. T. Reza Fauzan, "A Different Approach on Automated Use Case Diagram Semantic Assessment," *International Journal of Intelligent Engineering and Systems,* vol. 14, no. 1, pp. 496-505, 2021.

[15] R. S. M. a. A. C. Elmansouri, "A UML 2.0 activity diagrams/csp integrated approach for modeling and verification of software systems," *Computer Science,* vol. 22, 2021.

[16] H. M. E. R. a. L. e. M. N. Y. Achraf, "Transforming the business process diagram into a class diagram by model-driven architecture," *Indonesian Journal of Electrical Engineering and Computer Science,* vol. 29, no. 2, pp. 845-851, 2023.

[17] B. Beizer, Black-box testing: techniques for functional testing of software and systems, John Wiley & Sons, Inc, 1995.