

KMITL

Project plan

Bangkok, Thailand

Jarne Dirken
Kobe Vandendijck
Sohaib Ibenhajene

Table of contents

1. COMPANY	4
2. PROBLEM	5
3. OBJECTIVE	6
4. BUSINESS CASE	7
4.1. Added value	7
4.2. Stake holders	7
5. PROJECT TIMELINE	8
5.1. The steps that are going to be taken	8
5.1.1. Initialization phase	8
5.1.2. Realization phase	8
6. COMMUNICATION AND RESPONSIBILITIES	9
7. PROJECT SCOPE	10
8. INSPECTION OF THE CURRENT APPLICATION	12
8.1. Register form	12
8.2. Login form	13
8.3. First impression	13
8.4. Borrow page	14
8.5. Return page	14
8.6. Borrowing history page	15
8.7. Approve borrow (admin)	15
8.8. Report borrow (admin)	16
8.9. Repair history (admin)	16
8.10. Equipment (admin)	17
8.10.1. Add equipment	17
8.10.2. Edit equipment	18
8.11. Location (admin)	18
8.11.1. Add location	19
8.11.2. Edit location	19
8.12. Users (admin)	20
8.12.1. Add user/register page	20
8.12.2. Password setting	21
8.12.3. Edit register	21

8.13. General	22
9. RESEARCH	23
9.1. Front-end	23
9.2. Javascript VS Typescript	23
9.3. Back-end	24
9.4. Database	25
9.5. UI	25
9.6. Real time updating	26
9.7. Push alerts	26
9.8. Authentication	27
9.9. Infrastructure	28
9.9.1. Hosting	28
9.9.2. Containers	29
9.10. Security	29
10. STORYBOARD	30
10.1. Desktop	30
10.1.1. General:	30
10.1.2. Student login:	32
10.1.3. Supervisor login:	35
10.1.4. Admin login:	42
10.2. Mobile	46
10.2.1. General:	46
10.2.2. Student login:	47
10.2.3. Supervisor login:	48
10.2.4. Admin login:	50
10.3. Links to screens	52
11. CONCEPTUALIZATION OF THE APPLICATION	52
11.1. Use case diagram	52
11.2. Datamodel	54
11.3. General	54
12. REFERENCES	55

1. Company

KMITL or King Mongkut's Institute of Technology Ladkrabang is a university located in Ladkrabang, Bangkok, Thailand.

There are two other universities called the King Mongkut's Institute of Technology Thonburi, and King Mongkut's Institute of Technology North Bangkok. This organization is dedicated to the exploration, analysis, enhancement, and provision of services in technology, science, and technical education. Its mission also encompasses the preservation and promotion of the nation's art and culture. The school counts more than 20,000 students in all major and educational levels. (kmitl, n.d.)

KMITL comprises 11 distinct schools, which are:

1. School of Engineering
2. School of Architecture, Art, and Design
3. School of Science
4. School of Agricultural Technology
5. School of Industrial Education and Technology
6. School of Food Industry
7. School of Information Technology
8. KMITL Business School
9. School of Liberal Arts
10. Faculty of Medicine
11. School of Dentistry



KMITL
สถาบันเทคโนโลยี
พระจอมเกล้า
เจ้าคุณทหารลาดกระบัง

2. Problem

KMITL has a lot of hardware components that students can borrow if they want to. That's why the school decided to make an inventory managing system. This way students can borrow items, return them and everything can be monitored by the application so no components should get lost.

However, the existing system, while effective in its primary goal, exhibits several limitations that require attention. Our task is to address these shortcomings to enhance the system's functionality and user experience. The identified issues include:

- The problem is that the current system doesn't have a good authentication system. Users can just create an account with whatever student number, email, name, etc...
- The absence of an approval process for borrowing items enables individuals to potentially borrow an entire stock without oversight, leading to potential abuse and inventory management challenges.
- The system's design is not fully optimized for mobile devices.
- The organization of the inventory table falls short of expectations, making it difficult for users to navigate and locate specific items efficiently.
- The overall design of the system is overly simplistic, which, while minimalist, may not engage users effectively or provide an intuitive user interface.
- User-friendliness is compromised, indicating that navigating the system and performing actions like borrowing or returning items could be more intuitive and less cumbersome for users.

Our involvement will focus on developing solutions to these issues, aiming to create a more secure, accessible, and user-friendly system that meets the needs of both the institution and its students. By addressing these challenges, we will ensure that the inventory management system not only functions effectively but also enhance the user experience for all stakeholders involved.

3. Objective

Dealing with borrowing components at university can really be a headache for students. University staff don't enjoy chasing after items they've lent to students either. We're spending our internship trying to fix this problem.

Our plan is to make a web application specifically for KMITL students and staff. It's going to be very straightforward and easy for everyone to use, but it will also have plenty of useful features. This web app is all about making sure students and staff can work together smoothly, so nobody must wonder where something is or who has borrowed it.

For building this app, we're using Next.js as our main technology and PostgreSQL for handling all the data. We want to make this app a lot better than the current one. Some new things we're adding include a system where staff can approve loans easily and a feature for checking items in and out with QR codes. The webserver should run on a local server in KMITL itself that we have access to. We have to keep in mind that the system will be used by a minimum of 600 students so security, performance and scalability will be key factors in our project.

With these updates, borrowing components at the university will become a whole lot easier and more organized.

4. Business case

4.1. Added value

Our project will offer significant value propositions with broad-reaching benefits. We stand to enhance the current borrowing system with the help of new technology and features.

With the help of our project, we reduce the risk of lost or stolen items through better tracking, monitoring, and authentication. Furthermore, we will be facilitating a smoother borrowing experience, saving time and reducing frustration for both students and staff. We mustn't forget to make our solution scalable that meets the growing needs of the KMITL community.

4.2. Stake holders

KMITL Staff: Our primary stakeholder is the staff of KMITL. They will benefit from improved inventory management, better clarity and role management.

KMITL Students: Students will benefit from this system by having a better user experience while borrowing or returning an item.

5. Project timeline

5.1. The steps that are going to be taken

Our project consists of two phases. The initialization phase and the realization phase.

5.1.1. Initialization phase

In this phase we will be focusing on our project plan (this document). This phase consists of research only. These are the steps we are going to take:

1. Talk to someone who uses or knows the current system and ask for an opinion.
2. Inspect the current system and try to look for improvements.
3. Research the tools we are going to use for the new system.
4. Make the screens in Figma.

The initialization phase should only last 3 weeks. **4/03 – 24/03**

9	04/3	Initiation phase	Internship 1
10	11/3		Internship 2: kick-off meeting
11	18/3		Internship 3

5.1.2. Realization phase

In this phase of our project, we will be developing a new system. We have done all the research so now it's time to put it into practice. These are the steps we are going to take:

1. Make a basic front-end and back-end with dummy data and make sure they can communicate with each other with the help of an API.
2. Improve the application so it looks like our Figma board with all the requirements.
3. Make everything work as needed, remove bugs and put in some real data.

The realization phase should last all the other weeks (10): **25/03 – 24/05**

12	25/3	Realization phase	Internship 4: 1st meeting at school
13	01/4		Internship 5
14	08/4		Internship 6: intermediate internship evaluation
15	15/4		Internship 7
16	22/4		Internship 8: 2nd meeting at school
17	29/4		Internship 9
18	06/5		Internship 10
19	13/5		Internship 11
20	20/5		Internship 12: submit internship evidence documents for review

6. Communication and responsibilities

Our supervisors will check up on us every week. If they don't have time to check up on us, they will send over a senior student who uses the current system and he will look at what we have done and report everything back to the supervisors.

Because we will be working on a web application, we can host the website and present it live. They can follow up our work remotely by looking at the website.

Kobe will be doing all the infrastructure and security of this website. He will set up a self hosted GitLab server and secure the pipeline with SAST,DAST and WAF. For hosting the website he will use the things already in place from the old website. He will reuse the SSL/TLS certificates and the API calls will be secured with API keys.

Because we will be working with two full stack developers on this project we are going to divide the work. Jarne will mainly work on the student and the login pages. Sohaib will mainly work on the supervisor pages. We'll both be developing the admin pages as well as any extra features that should be included such as alerting, shopping cart, ...

7. Project scope

Here our MoSCoW of the project.

The MoSCoW method is a prioritization tool in project management, dividing tasks into:

- Must have (M): Essential for success.
- Should have (S): Important but not critical.
- Could have (C): Nice to have, if possible.
- Won't have (W): Not included in this phase.

This helps focus on what's crucial for the project's outcome.

M

- ust have:
- Web application
 - Managing inventory
 - Borrowing items
 - Returning items
 - Approval based borrowing & returning
 - Creating account
 - QR code generating
 - Transaction history
 - Repair history
 - Basic infrastructure
 - TLS/SSL certificates
 - API keys
 - Domain name

S

- hould have:
- Item's transaction history
 - User transaction history
 - Responsive to fit mobile screens
 - Account summary
 - Add push notification 4 conditions: loan, return, almost expired(StSV), expired (StSV)
 - Filter on specific attributes
 - Urgent borrowing process
 - Excel import
 - Improve repair flow- icon addition: expired
 - Backups (snapshots vSphere)
 - Advanced borrow system
 - Multi loan / delete

C

ould have:

- Export item history
- Export user history
- Export repair history
- Advance booking features (reserve item)
- Add message when returning borrow
- Supervisor / admin change borrow status
- Add message on repair item
Repair history in item history
- Log of all activities (admin only)
- Low stock option alert
- Automatic number appending

W

on't have:

- Mobile application
- Any AI driven features
- Any analytics
- Importing all old data
- Placing QR code on all hardware components

8. Inspection of the current application

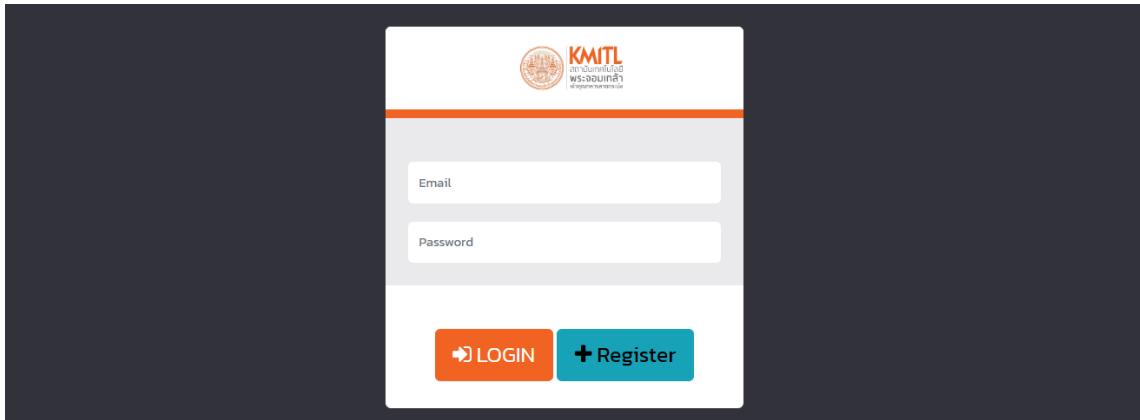
8.1. Register form

The screenshot shows a registration form titled "Register E-Borrow". It is divided into two main sections: "Information" and "การเข้าสู่ระบบ" (Login System). The "Information" section contains fields for "Student Code", "Firstname" (left) and "Lastname" (right), and "Tel.". The "การเข้าสู่ระบบ" section contains fields for "E-mail (For Login)" and "Password" (left) and "Confirm to Password" (right). A large orange "Confirm" button is located at the bottom center.

When we first saw the application, we had to register to use the application. The first thing we noticed is that we can just enter anything anywhere. Meaning if we want to fill in "1" everywhere, we can. There is no validation, which is not good.

Also, we can see that there is no "back" button. If we accidentally click on "register", there is no way for the user to go back.

8.2. Login form



When filling in nothing and pressing “login” you don’t see any error messages. This is not user friendly. Only when you type something and try to log in, you can see an error message (in Thai, so not in the language of the browser).

Additionally, we are concerned that this system, with its custom registration page storing data in a local database without adequate verification, coupled with a login page that fails to validate input values properly, could pose a significant security risk.

8.3. First impression

A screenshot of the E-Borrow landing page. On the left is a navigation bar with buttons for "+ Borrow", "Return", and "Borrow History". The main area has a search bar at the top labeled "Search". Below it is a table with columns: "#", "No.", "Name", "Model", and "Brand". The table contains 7 rows of data. A search bar "Search:" is also present in the main area.

That is what the landing page looks like when you first login, there is a lot going on. On the left side is the navigation bar with all the pages, and the rest of the page is a view of the database with all the available items. The navigation bar looks very good and user friendly, it is simple and easy to use and folds in if you want it to. The main part also has a search function, where you can search for anything by name, no brand or model. We noticed that if there are multiple items that are the same, they all show up on this screen. The last thing we noticed is that the database is not shown in full on the page, on the bottom there is a slider for showing the rest. These last 2 things make the first page a bit overwhelming and confusing.

8.4. Borrow page

The screenshot shows the 'Borrow' page of the KMUTNB Library Management System. The left sidebar has a blue header bar with '+ Borrow' highlighted. Other options include 'Search', 'Return', and 'Borrow History'. The main area has a teal header bar with 'Borrow'. Below it are fields for 'Borrow Date' (auto-filled with '05/03/2567 13:44:29'), 'Borrower' (auto-filled with '1 1'), 'Return Date' (with a calendar icon), and 'Equipment List' (a dropdown menu showing '-- select--'). At the bottom are 'Confirm' and 'Back' buttons.

The borrow page still has the clean navigation bar. The rest of the page looks good as well, and it is easy to use. The best part of the page is that it fills in the date automatically, so you don't have to go look for it. There is also a button that sends you back to the home page. The only thing on this page that bothered us is that the search bar gives you again all the items with the same name, and that makes it a bit chaotic.

8.5. Return page

The screenshot shows the 'Borrow list' page. The left sidebar has a blue header bar with '+ Borrow' highlighted. Other options include 'Search', 'Return', and 'Borrow History'. The main area has a teal header bar with 'Borrow list'. It includes a search bar, a table header with columns for '#', 'Borrow Date', 'Return Date', 'Count Days', 'Equipment', 'Type', and 'Status', and a button '+ Add Borrow'. A message 'No data available in table' is displayed. Navigation buttons 'Previous' and 'Next' are at the bottom.

On this page is shown what you have borrowed from the university that you have not brought back yet. We like this concept very much, as well as the add borrow button that send you to the borrow page.

8.6. Borrowing history page

The screenshot shows the 'Borrow history' section of the E-Borrow application. On the left, there's a sidebar with links: 'Search', '+ Borrow', 'Return', and 'Borrow History' (which is highlighted in blue). The main area has a header 'Borrow history' with a search bar. Below it is a table with columns: '#', 'Borrow Date', 'Return Date', 'Equipment', 'Status', and 'Status Approve'. A message 'No data available in table' is displayed. At the bottom right of the table area are 'Previous' and 'Next' buttons.

The borrow history page shows you everything you have borrowed in the past and have brought back. This is also a nice addition to the inventory system.

8.7. Approve borrow (admin)

The screenshot shows the 'Approve Borrow' section of the E-Borrow application. On the left, there's a sidebar with links: 'Search', '+ Borrow', 'Return', 'Borrow History', 'Approve borrow' (which is highlighted in blue), 'Report borrow', 'Repair history', 'Equipment', 'Location', and 'Users'. The main area has a header 'Approve Borrow' with a search bar. Below it is a table with columns: '#', 'Borrow Date', 'Borrower', 'Return Date', 'Count Days', 'Equipment', 'Type', and two buttons 'Approve' and 'Reject'. A single row is shown in the table: '1' (row number), '07/03/2567 12:12:50', 'Dio Tony', '08/03/2567', '1', 'Extension Cord 3m 5 way with USB', 'Return', and two buttons 'Approve' (with a checkmark) and 'Reject'. At the bottom right of the table area are 'Previous' and 'Next' buttons.

Now let's look at the website from an administrator perspective. The first pages stay the same as before, but certain pages have been added. Starting with Approve borrow, on this page there is a list shown of all the request to borrow something as well as all the requests to return something. This would be quite chaotic if the system was in full use and there were more requests. For each request there is a button to approve or reject the request and each button triggers a notification pop-up.

8.8. Report borrow (admin)

#	Borrow Date	Borrower	Equipment	Deadline	Count Days	Return Date	Return Status	Approve Status	Approver
71	31/01/2567 18:24:25	Dio Tony	HDMI Cable 1m (01)	28/02/2568	394	-	borrowing	Approved	Dio Tony
72	31/01/2567 18:13:57	Dio Tony	MSI GF65 Thin	01/02/2567	1	31/01/2567 18:14:15	returned	Approved	Dio Tony
73	31/01/2567 18:13:15	Dio Tony	MSI GF62 8RE (Win Only)	01/02/2567	1	31/01/2567 18:14:18	returned	Approved	Dio Tony
74	31/01/2567 18:12:59	Dio Tony	ACER Laptop (401-1)	01/02/2567	1	31/01/2567 18:14:20	returned	Approved	Dio Tony
75	31/01/2567 12:45:17	Thanakorn Kriangudom	ACER Laptop (401-03)	05/02/2567	5	04/03/2567 14:37:21	returned	Approved	Dio Tony
76	30/01/2567 20:43:16	Dio Tony	Portable Speaker with Microphone	01/07/2567	153	-	borrowing	Approved	Dio Tony
77	29/01/2567 15:12:30	Pyae Phyoe Min	MSI GF62 8RE (101-01)	12/02/2567	14	13/02/2567 15:12:50	returned	Approved	Dio Tony
78	28/01/2567 22:59:07	Praiyanka Joshi	ACER Laptop	31/01/2567	3	01/02/2567 11:07:16	returned	Approved	Dio Tony
79	28/01/2567 16:55:54	Dio Tony	Mini-Display port to HDMI Converter	30/06/2567	154	01/02/2567 12:41:48	returned	Approved	Dio Tony
80	28/01/2567 16:55:15	Dio Tony	Display to HDMI (Female) Converter	30/06/2567	154	01/02/2567 12:41:44	returned	Approved	Dio Tony

This page is very good, you can filter on dates, so that you don't have too many entry's. You can also export as pdf with a button for printing out as well as a search bar and the sorting is handy on this kind of page.

8.9. Repair history (admin)

#	Equipment	Repair Date	Return Date	Status
1	HP Laptop	07/03/2567 14:39:30	07/03/2567 14:42:09	Returned
2	Digital Multimeter UT89XD (01)	-	06/03/2567 10:28:03	Returned
3	Digital Multimeter UT89XD (01)	-	06/03/2567 10:27:54	Broken
4	Digital Multimeter UT89XD (01)	-	06/03/2567 10:27:26	Returned
5	Digital Multimeter UT89XD (01)	06/03/2567 10:24:14	06/03/2567 10:26:18	Broken
6	DOBOT Magician	18/01/2567 17:53:54	18/01/2567 17:53:58	Broken
7	175mm 3D Printing Filament (ABS)	-	15/01/2567 16:32:12	Returned
8	175mm 3D Printing Filament (ABS)	-	11/01/2567 14:33:46	Broken
9	MSI GF62 8RE (101-01)	08/01/2567 12:43:13	10/01/2567 12:57:24	Returned
10	HP Laptop	08/01/2567 12:43:05	10/01/2567 13:01:32	Returned
11	HP Laptop	08/01/2567 12:42:59	10/01/2567 13:00:30	Returned
12	DOBOT Magician	-	08/01/2567 12:15:00	Broken
13	HP Laptop	-	04/01/2567 20:06:56	Broken
14	HP Laptop	-	04/01/2567 20:06:47	Returned
15	HP Laptop	-	04/01/2567 19:37:18	Broken
16	HP Laptop	-	04/01/2567 19:37:09	Returned

This is also quite a good and clear page, the few columns make it an easy to understand page without confusion. Maybe adding a date filtering would be good if the application was bigger and more items were being used.

8.10. Equipment (admin)

The screenshot shows a table of equipment items with the following columns: #, No., Name, Model, Brand, Location, Status, and Actions (Edit, Delete). The items listed include Extension Cord 20m (Roll), Tello Drone (Black), Extension Cord 3m 5 way with USB, Portable Speaker with 2 Microphones, Driller GSB 18 VE-EC Professional, Ipad Stand, AC-DC Adapter 9V, 2A (36), AC-DC Adapter 9V, 2A (35), and AC-DC Adapter 9V, 2A (34). The status column indicates some items have been borrowed.

#	No.	Name	Model	Brand	Location	Status	Action
394		Extension Cord 20m (Roll)		Toshino	HE2 - HM Inventory Zone E Storage 2	has been borrowed	Edit Delete
393		Extension Cord 20m (Roll)		Toshino	HE2 - HM Inventory Zone E Storage 2	Open	Edit Delete
392	RAI-MOBI-DRO-101-02	Tello Drone (Black)		Ryze	HA2 - HM Inventory Zone A Storage 2	Open	Edit Delete
391		Extension Cord 3m 5 way with USB		Toshino	HE2 - HM Inventory Zone E Storage 2	has been borrowed	Edit Delete
390		Portable Speaker with 2 Microphones		Soundmilen	HA1 - HM Inventory Zone A Storage 1	Open	Edit Delete
389	RAI-ELEC-SPA-101-02	Driller GSB 18 VE-EC Professional	SOLDERING GUNS	BOSCH	HC4 - HM Inventory Zone C Storage 4	Open	Edit Delete
388	RAI-COMP-ACES-136-01	Ipad Stand			201	has been borrowed	Edit Delete
387	RAI-ELEC-POW-401-36	AC-DC Adapter 9V, 2A (36)	Power Supplies	Venus Supply	HA3 - HM Inventory Zone A Storage 3	Open	Edit Delete
386	RAI-ELEC-POW-401-35	AC-DC Adapter 9V, 2A (35)	Power Supplies	Venus Supply	HA3 - HM Inventory Zone A Storage 3	Open	Edit Delete
385	RAI-ELEC-POW-401-34	AC-DC Adapter 9V, 2A (34)	Power Supplies	Venus Supply	HA3 - HM Inventory Zone A Storage 3	Open	Edit Delete

This page displays all the equipment that can be borrowed from the university. This also gives the administrator some options, you can add, edit or delete the items. Different to the search and borrow pages where every single item is shown, what we do not like. On this page it is actually really helpful to have. There is also a button on the screen that allows you to print out the QR-codes.

8.10.1. Add equipment

The form has fields for Picture (with a placeholder 'click here to upload'), No., Name, Model, Brand, Location (dropdown menu with 'กรุณาเลือก--'), Status (radio buttons for Open, Close), Status Repairing (radio buttons for Normal, Repair, Broken), and Save/Back buttons.

The “add” button on the equipment page leads you to another page. On this page you can choose to add an item to the list. Here you can enter its basic information, as well as a picture. The bad thing we noticed here is that the only required field is the location, this enables people to add empty items to the list.

8.10.2. Edit equipment

The screenshot shows the 'edit equipment' form. On the left is a sidebar with navigation links: Search, + Borrow, Return, Borrow History, Approve borrow, Report borrow, Repair history, Equipment, Location, and Users. The main area has a title 'edit equipment'. It includes fields for Picture (with a placeholder 'click here to upload'), No. (empty), Name ('Extension Cord 20m (Roll)'), Model (empty), Brand ('Toshino'), Location ('HE2 - HM Inventory Zone E Storage 2'), Status (radio buttons for Open and Close, with Open selected), Status Repairing (radio buttons for Normal, Repair, and Broken, with Normal selected), and two buttons at the bottom: 'Save' (blue) and 'Back' (yellow).

If you push the edit button you go back to the add equipment page, with the only difference being that the known information is filled out in the form and the title is different.

8.11. Location (admin)

The screenshot shows the 'Location' management page. The sidebar on the left includes the same navigation links as the previous page. The main area has a title 'Location' and a table with columns '#', 'Location', and 'Actions'. The table contains 10 entries:

#	Location		
1	HF2 - HM Inventory Zone F Storage 2	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
2	HF1 - HM Inventory Zone F Storage 1	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
3	HF3 - HM Inventory Zone F Storage 3	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
4	HE3 - HM Inventory Zone E Storage 3	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
5	HE3 - HM Inventory Zone E Storage 3	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
6	HE2 - HM Inventory Zone E Storage 2	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
7	HE1 - HM Inventory Zone E Storage 1	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
8	HD2 - HM Inventory Zone D Storage 2	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
9	HD1 - HM Inventory Zone D Storage 1	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
10	HB1 - HM Inventory Zone B Storage 1	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>

At the top right are buttons for '+ Add Data' and 'Search'. At the bottom right are buttons for 'Previous', page numbers (1, 2, 3), and 'Next'.

This page is again quite good, it shows all the locations where there is storage for equipment. On this page you can add, edit or delete locations. You can again search for one as well. On this page we noticed that if you have less entry's than the amount specified on the top, the button for going to another page jumps with it to the top, staying just underneath the last piece of data.

8.11.1. Add location

The screenshot shows a web-based application interface titled 'E-Borrow'. On the left, there is a sidebar with various menu items: Search, Borrow, Return, Borrow History, Approve borrow, Report borrow, Repair history, Equipment, Location, and Users. The main content area is titled 'Add Location' and contains a single input field labeled 'Location' with a placeholder text 'HF2 - HM Inventory Zone F Storage 2'. Below the input field are two buttons: 'Save' (blue) and 'Back' (yellow). The top right corner of the window shows standard window controls (minimize, maximize, close).

On this page you can enter a new location, otherwise it is empty. There is no basic format in which a location can be entered. This could be done in order to make sure that no one can just enter anything.

8.11.2. Edit location

The screenshot shows the same 'E-Borrow' application interface. The sidebar and overall layout are identical to the 'Add Location' page. The main content area is titled 'Edit Location' and displays the same input field and 'Save/Back' buttons. The input field now contains the text 'HF2 - HM Inventory Zone F Storage 2', indicating that the location has been edited.

On this page you can edit the location data, it is again empty other than the one form element.

8.12. Users (admin)

#	Student Code	Firstname	Lastname	Tel	Email	Level	Status	Action	Action	Action
1	8	8	8	8	8	Admin	Open			
2	2	2	2	2	2	Student	Open			
3	65011402	Natwasa	Manomaiwiboon	0900060151	65011402@kmit.ac.th	Student	Open			
4	64011444	Krit	Hoonpong simanont	0805925565	64011444@kmit.ac.th	Student	Open			
5	66011066	Myet Noe	Oo	0651184001	66011066@kmit.ac.th	Student	Open			
6	66011056	Lilith	Eempikul	0823695060	66011056@kmit.ac.th	Student	Open			
7	65011563	Tanakorn	Youngmeesuk	0972356319	65011563@kmit.ac.th	Student	Open			
8	65011598	Thitiphan	Chennukmatupoom	0969323625	65011598@kmit.ac.th	Student	Open			
9	63011322	Suphason	Suttinon	0843593545	63011322@kmit.ac.th	Student	Open			
10	65011415	Om Prakash	Acharja	+650638698495	65011415@kmit.ac.th	Student	Open			

On this page we can see all the registered users of our web application, you can see basic information as well as level clearance and status. You also have a couple of buttons that you can use: add user, password, edit and delete.

8.12.1. Add user/register page

Student Code:

Firstname:

Lastname:

Tel:

E-mail:

Role:

Status:

If you click on add user, you are sent to a page called register. On this page you can add basic information of a user, as well as set the role they have and the status of the account. You can not enter a password on this page so that is some extra work you will have to put in to create a user.

8.12.2. Password setting

The screenshot shows a web-based application interface titled 'E-Borrow'. On the left, there is a sidebar with various navigation links: Search, Borrow, Return, Borrow History, Approve borrow, Report borrow, Repair history, Equipment, Location, and Users. The main content area has a title 'Password Setting ឯ៉េ-សមាស្រោះ : test test'. It contains two input fields: 'Oldpassword' and 'Newpassword'. Below these fields are two buttons: a green 'Save' button and a yellow 'Cancel' button.

This page only contains a form that lets you add and confirm a password of the user you have selected. You can change the password of a known user as well, it does not matter what role they have. The users don't even get a message to let them know their password has been changed. This could be a severe security breach if someone ever got in. You could even change an admin password if you wanted to.

8.12.3. Edit register

The screenshot shows the 'Edit Register' page of the E-Borrow application. The sidebar on the left includes the same set of navigation links as the previous page. The main page has a title 'Edit Register'. It features a form with several input fields: 'Student Code' (containing 'test'), 'Firstname' (containing 'test'), 'Lastname' (containing 'test'), 'Tel' (containing 'test'), 'E-mail' (containing 'test'), 'Role' (a dropdown menu set to 'Student'), and 'Status' (a dropdown menu set to 'Open'). At the bottom of the form are two buttons: a green 'Save' button and a yellow 'Back' button.

This page allows you to edit the basic information of a user, you can take away/give admin rights to/from the users. You can also change the status of an account to closed.

8.13. General

To give a general overview of the current application. It looks good and is very fast. There are things that should be fixed like a user profile page. The f11 button is kind off useless. The sidebar isn't properly aligned. The table is too big and doesn't look good on mobile. There is no option to filter on multiple criteria.

Also, we think it might be best to combine the return and history page together into one page. This could make it easier to understand.

The security also lacks some things that we will need to change on our own application. And we think that merging some of the extra pages (i.e. add, edit) would be beneficial as well.

Overall, a good start but it feels like a half-finished product. We will do our best to implement everything they asked for so that the new system is everything they need, and they can step away from the google forms.

9. Research

9.1. Front-end



Next.js is widely acclaimed for its exceptional capabilities in the development of modern web applications, especially as a frontend framework. Utilizing React, it offers an enhanced experience for building user interfaces, incorporating server-side rendering (SSR), static site generation (SSG), and client-side rendering methods to optimize performance and user experience across the web.

Next.js includes automatic image optimization with the `next/image` component, which serves optimized images in the formats supported by modern browsers, resizing images on demand. This feature significantly improves loading times and performance by reducing the size of images without compromising quality.

Built-in internationalization (i18n) support in Next.js allows for the creation of multi-language web applications with ease. It provides automatic language detection, URL routing, and efficient loading of localized content, making it simpler to develop global applications. Which would be very helpful here so we can just detect the language of the browser and give the error messages in that language.

9.2. Javascript VS Typescript

For next.js we can choose whether we want to build our app with JavaScript or typescript. So, which one is better? Well, they both have their pros and cons. JavaScript is easier to learn and more forgiving but that also means it's more prone to errors and really difficult to scale for larger projects. Typescript on the other hand is a superset of JavaScript so all valid JavaScript code will also work in typescript but the extra features that It's less forgiving so it can catch errors faster.

Ultimately, the best language to use to build Next.js applications is the one that you are most comfortable with and that best meets the needs of your project. If you are new to programming, I recommend starting with JavaScript. Once you have a good understanding of JavaScript, you can then decide if you want to learn TypeScript. In case you like to build production level projects with huge code base without a doubt typescript will be the best option for you. (Salman, 2023)

Next.js is built with TypeScript under the hood, so you get better IntelliSense and type definitions in your editor by default with just JavaScript. But when you couple that with TypeScript, you can get an even better developer experience — including instant feedback when your component expects props, but you didn't pass any. Well, this is something most IDEs understand even if you don't use

TypeScript. But as you work with Next.js, you will build many components with optional props, as well as many utility functions which can be enhanced with TypeScript for better bug spotting, documentation, IntelliSense.

You're also able to build with Next's exported types and define your own to build with across your applications. These types help give your code better structure by dictating what your objects, arrays, etc., look like ahead of time. That way, you, your code editor, and any developer after you knows how to reference your code. (Chaudhari, 2023)

So, in the end what does it all comes down to? Are we working on a bigger project, using typescript for more documentation, better bug spotting and better intellisense. That's why for this project we are going to use Typescript.

9.3. Back-end

As for our back-end we should also use Next.js. This is convenient because we don't need to change languages, we can just use Next.js for front- and back end.



Here we will write API-calls to our database to show the data on our website. One of the core features that facilitate Next.js's use as a backend solution is its API routes. These routes allow you to create RESTful APIs directly within your Next.js application. By placing any file under the app/api directory, you can create an endpoint that behaves as an API. This setup simplifies the development process as you can manage both your frontend and backend code in a single project, making it easier to develop, test, and deploy your application.

API routes in Next.js are essentially serverless functions that run on demand, scaling automatically with the number of requests. This serverless approach reduces the overhead of server management, ensuring that your application can scale effortlessly without the need to manage infrastructure. It's particularly beneficial for applications with fluctuating traffic, as it can dynamically allocate resources to meet demand.

Handling authentication in Next.js apps can be managed through API routes, utilizing libraries such as NextAuth.js. This library simplifies the implementation of authentication systems, supporting various authentication providers and strategies, including email, social login, and JWT tokens. Moreover, Next.js apps can be secured using standard web security practices, such as secure headers, HTTPS, and data validation/sanitization to prevent common web vulnerabilities.

9.4. Database

PostgreSQL



PostgreSQL is a powerful, open-source object-relational database system known for its robustness, scalability, and adherence to SQL standards. It has become the go-to database for a wide range of applications across various industries, from startups to large enterprises, due to its advanced features and reliability. PostgreSQL offers a sophisticated yet practical solution for managing data,

regardless of the complexity or volume, making it an ideal choice for both traditional and modern, web-facing applications.

Data integrity and reliability are paramount in PostgreSQL, with features such as atomicity, consistency, isolation, durability (ACID) properties, sophisticated locking mechanisms, and foreign keys ensuring that the database remains consistent and robust under various conditions. It also supports savepoints and point-in-time recovery, enhancing data protection and allowing administrators to restore data to a specific moment in case of an error or system failure. This is very useful when working with a lot of data that changes constantly.

In order to create the database and make a connection with it we are going to use Prisma.io. The purpose is that Prisma simplifies database access by providing an easy-to-use API for querying and managing our database. Prisma is used on the backend to make sure our database is created with schemas and for data fetching. Or as Prisma like to say it:



"Prisma makes working with data easy! It offers a type safe Node.js & TypeScript ORM, global database caching, connection pooling, and real-time database events." (Rauch, n.d.)

9.5. UI

For our UI design framework, we are incorporating both TailwindCSS and MUI to leverage the unique strengths of each. TailwindCSS has been chosen for its efficient UI construction and component management. Its performance is notably superior, outperforming traditional CSS by 48%, as highlighted by Austin in 2023. This significant improvement makes it an optimal choice for crafting responsive and visually appealing interfaces with less effort. (Austin, 2023)



MUI complements our design strategy by providing a robust set of React components. This integration is essential for achieving a sleek and attractive UI design, facilitating the development of complex UI features with simplicity and elegance.



By combining TailwindCSS's efficiency and MUI's comprehensive component library, we aim to create a user interface that is not only beautiful and user-friendly but also highly performant and scalable. This dual-framework approach allows us to harness the best of both worlds, ensuring a smooth and responsive user experience across our application.

9.6. Real time updating

For implementing real-time updates, we're planning to integrate WebSockets into our system. This technology enables instant communication between the user's browser and our server, ensuring that whenever a new request hits the database, the user on our website will receive an immediate notification via a popup.

Using WebSockets might be a bit more sophisticated than other methods for achieving real-time updates, but I've successfully utilized it in a previous project. This experience gives me confidence that incorporating it into our current web application will significantly enhance the user experience by keeping everyone informed with the latest updates as they happen.



Another approach might be to use React Querry. I've only heard about this method once and there is not a lot of documentation available, but this might also be a valid path to achieve the same result. Real Time updating.

9.7. Push alerts

To implement push alerts in our application, we're planning to focus on two main channels: email and LINE push notifications. There are others but these two seem to be the two most useful.

For sending email notifications, there are several tools and services we can utilize. These tools enable us to automate the process of sending out email alerts whenever specific actions occur within our application, such as new requests or updates.

For LINE notifications, we'll need to integrate with the LINE Messaging API. This requires setting up a LINE developer account, creating a LINE bot, and configuring it to send notifications. With this setup, we can send real-time alerts directly to a user's LINE app, informing them of new developments or updates relevant to their interests or actions on our platform.



By leveraging both email and LINE for push notifications, we aim to offer our users a flexible and efficient way to stay informed about important events and updates, enhancing their overall experience with our application.

9.8. Authentication

In evaluating our options for implementing authentication, we've narrowed down our choices to two primary candidates: Firebase and Auth0.



For our project, customization is a key requirement, especially the ability to incorporate our own fields for login and registration processes. After careful consideration, Firebase emerges as the more suitable choice due to its superior customization capabilities. It allows for the integration of custom user fields directly within the authentication flow, enabling us to tailor the login and registration experience to our specific needs. This flexibility extends to using Firebase's other services, such as Cloud Firestore or Realtime Database, to store and manage additional user data beyond the standard authentication parameters.

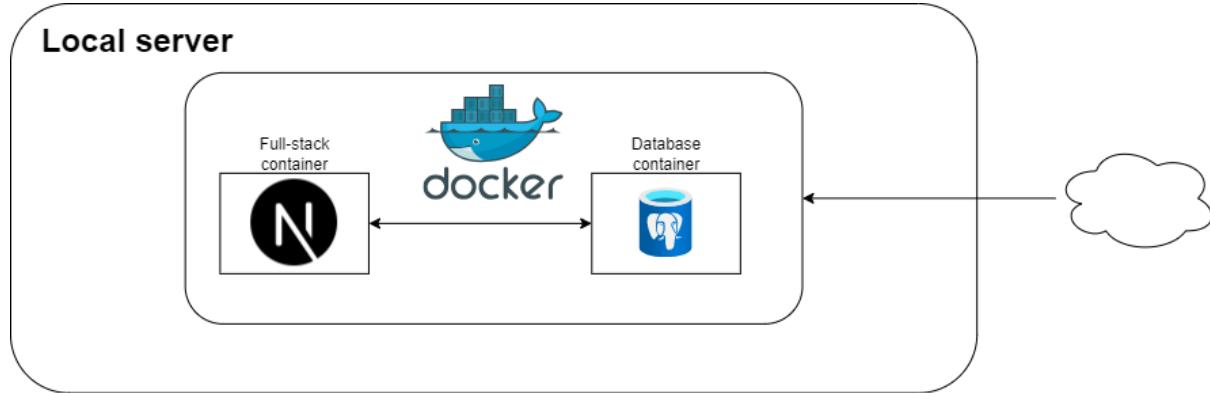
While Auth0 also offers a high degree of customization and a user-friendly interface for managing authentication flows, it seems that Firebase provides a more granular level of control that aligns better with our project's requirements. This includes the ability to create completely custom user interfaces for authentication, without the need for redirecting users to external pages, and the option to handle custom data more seamlessly within our application's infrastructure.

Moreover, Firebase's integration with other Google Cloud Platform services could offer additional advantages in terms of scalability, data processing, and analytics, which may prove beneficial as our project evolves.

In summary, although Auth0 is an exceptional choice with strong features for managing authentication flows, Firebase's extensive customization options and seamless integration with custom fields and additional user data make it the preferred choice for our project's specific needs. This decision is guided by our priority to deliver a tailored and integrated user experience throughout the authentication process.

9.9. Infrastructure

Here is what our infrastructure will look like:



The infrastructure for this application will be very simple because does it not require a big and complicated infrastructure.

The infrastructure for this application will look as the picture above suggests. On the local server there is a vSphere hypervisor installed, on this hypervisor I will create a Linux server. On this server I will install docker, and with this I will create 2 containers, 1 for the Next.js application itself and another for the PostgreSQL database. This way the application is easily scalable in the future if that is what the university wants.

9.9.1. Hosting



vSphere is a software suite developed by VMware, offering virtualization and cloud computing solutions for businesses. It facilitates the creation, management, and optimization of virtualized infrastructure and cloud environments. Key components include the ESXi Hypervisor, which partitions physical servers into multiple virtual machines (VMs); vCenter Server for centralized management and monitoring; the vSphere Client for web-based interface access; vSphere High Availability (HA) for automated failover protection; vSphere Distributed Resource Scheduler (DRS) for workload balancing; and vSphere Storage, providing virtualized storage solutions. Overall, vSphere simplifies the management and optimization of IT resources, granting businesses increased flexibility, scalability, and efficiency in their operations.

web-based interface access; vSphere High Availability (HA) for automated failover protection; vSphere Distributed Resource Scheduler (DRS) for workload balancing; and vSphere Storage, providing virtualized storage solutions. Overall, vSphere simplifies the management and optimization of IT resources, granting businesses increased flexibility, scalability, and efficiency in their operations.

9.9.2. Containers



Docker is a widely used platform for developing, shipping, and running applications within containers. Containers are lightweight, portable, and isolated environments that bundle an application with its dependencies. Docker simplifies the process of creating, deploying, and

managing containers through its core components: Docker Engine, Dockerfile, Docker Image, Docker Container, Docker Hub, and Docker Compose. It allows developers to package applications into portable units called Docker images, which can be easily shared and run on any platform that supports Docker. Docker has significantly impacted software development by providing a standardized and efficient way to package, distribute, and run applications across different environments.

9.10. Security

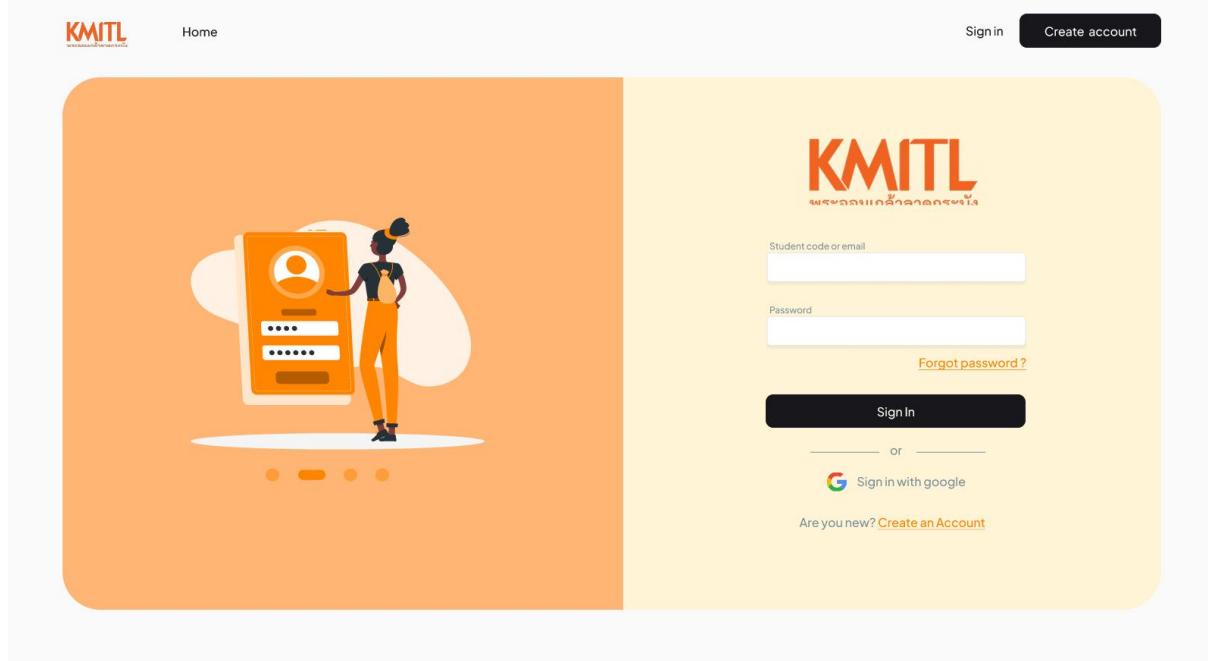
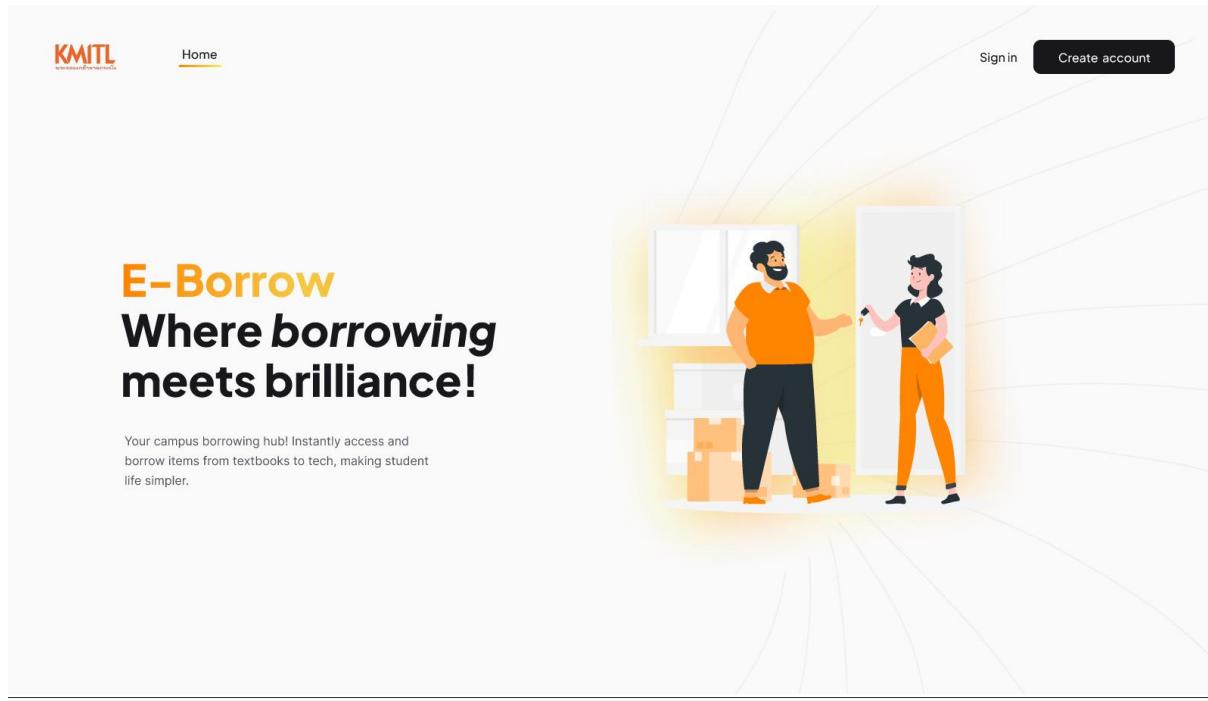
How the security measures will be implemented is mentioned in detail in the other file (Project_Plan_Security.dockx).

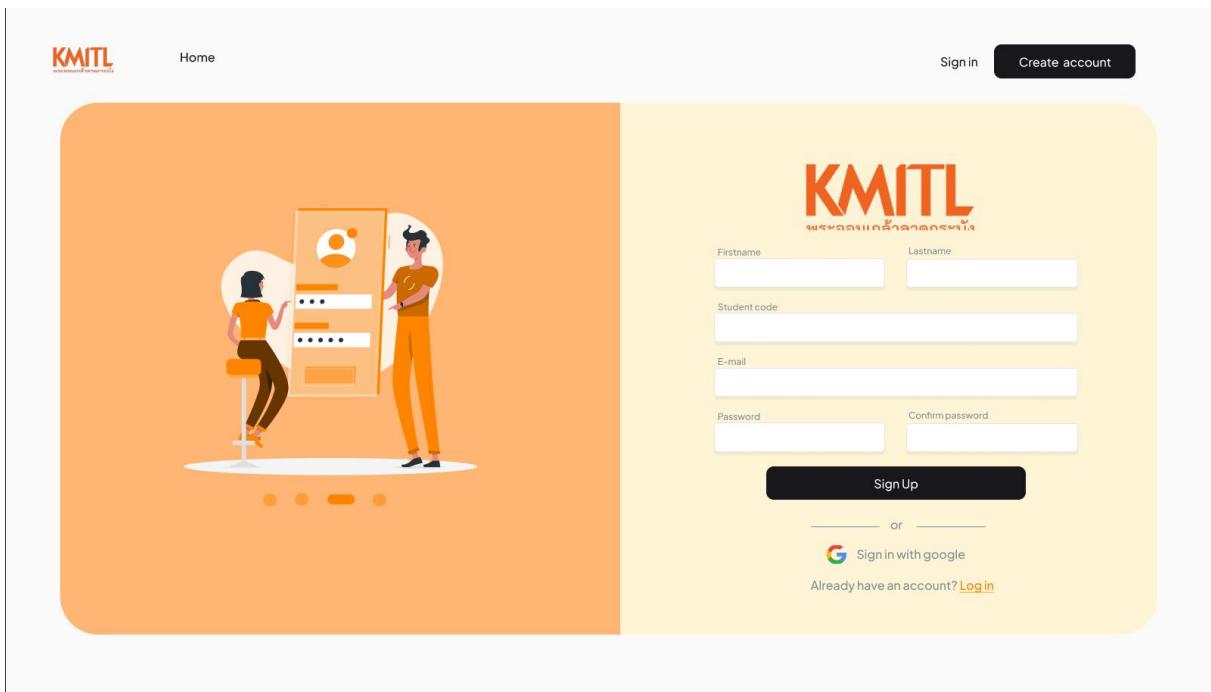
In simple terms the CCS student will play the role of an evil developer on the developer platform. He will clone the repository weekly and check for weaknesses on the platform while also trying to exploit the application. The information gotten from this will be used to secure the application.

10. Storyboard

10.1. Desktop

10.1.1. General:





The user profile page for John Doe. The top navigation bar includes links for Borrow, Return, and History. The main content area shows a placeholder photo with options to upload a new one or remove it. Below this is a 'Personal Info' section containing fields for Full Name (John Doe), Mail (john doe@example.com), Level (Student), Tel (6622249773), Student Number (45872530), and Password (*****). An 'Edit' button is located in the top right corner of this section. At the bottom left is a 'Log out' button.

10.1.2. Student login:

The screenshot shows the KMITL library system interface for student users. At the top right, there is a user profile for "John Doe" with the email "johndoe@kmitl.com". A bell icon indicates notifications. On the left, there are navigation links: "Borrow" (highlighted), "Return", and "History". Below these are search and filter fields for "Name", "Model", "Brand", and "Location", along with a "Sort by" dropdown and applied filters ("Location: RCC building, Main...").

The main area displays a list of products under the heading "PRODUCTS". Each item includes an icon, name, model, brand, location, and a "Borrow" button.

Name	Model	Brand	Location	Action
Digital Multimeter UT89XD (01)	MULTIMETER KITS	UNI-T	HA3 - HM Inventory Zone...	Borrow
Digital Multimeter UT89XD (01)	MULTIMETER KITS	UNI-T	HA3 - HM Inventory Zone...	Borrow
Digital Multimeter UT89XD (01)	MULTIMETER KITS	UNI-T	HA3 - HM Inventory Zone...	Borrow
Digital Multimeter UT89XD (01)	MULTIMETER KITS	UNI-T	HA3 - HM Inventory Zone...	Borrow

A "Logout" link is located at the bottom left of the main interface. A modal window titled "Borrow details" is open in the foreground, showing a digital multimeter icon and the following information:

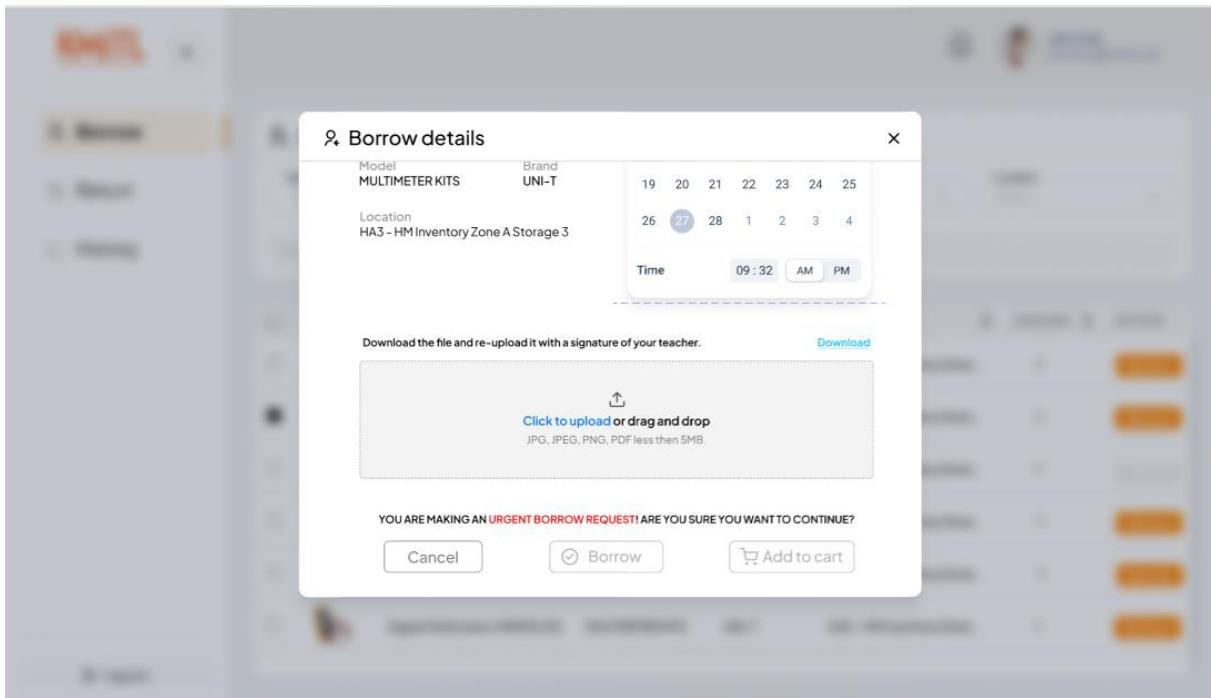
Product Details:

- Name: Digital Multimeter UT89XD (01)
- Model: MULTIMETER KITS
- Brand: UNI-T
- Location: HA3 - HM Inventory Zone A Storage 3

Borrow Options:

- Amount: Input (text field)
- Time: 09:32 AM
- Buttons: Cancel, Borrow (green), Add to cart (orange)

The modal also includes a calendar for selecting a pickup date, with February 2023 shown, and a "Select a pickup time" header.



The screenshot shows the main "Borrow" page. On the left, there are navigation links: "Borrow" (highlighted), "Return", and "History". The top right corner shows a user profile for "John Doe" (john.doe@kmitl.com) with a notification bell icon. The main content area has a search bar with dropdowns for "Name", "Model", "Brand", and "Location". It also includes sorting options ("Sort by Name") and applied filters ("Filters Applied: Location: RCC building, Main..."). Below this, there are two tabs: "PRODUCTS" and "PENDING BORROWS" (which is currently selected). A single pending borrow request is listed, showing a digital multimeter (UT89XD) from the MULTIMETER KITS brand at location HA3 - HM Inventory Zone. The status is "Pending" from 12 March 2024 to 20 March 2024. Buttons for "Cancel" and "View" are shown next to the item details.

KMITL
www.kmitl.ac.th

[Borrow](#)

[Return](#)

[History](#)

Return

🔍

📅

📅

↑↓ Sort by Name >

Filters Applied: X Name: Digital Multimeter UT89... X Borrow Date: 11/03/2024

CURRENT BORROWS

	🕒 2 days remaining Name: Digital Multimeter UT89XD (01) Model: MULTIMETER KITS	Brand: UNI-T Borrow Date: 11/03/2024 - 11:24 Return Date: 13/03/2024 - 11:24	Return
	🕒 1 day late Name: Digital Multimeter UT89XD (01) Model: MULTIMETER KITS	Brand: UNI-T Borrow Date: 14/03/2024 - 9:24 Return Date: 15/03/2024 - 9:24	Return

[Log out](#)

KMITL
www.kmitl.ac.th

[Borrow](#)

[Return](#)

[History](#)

History

🔍

📅

📅

Filters Applied: X Name: Digital Multimeter UT89... X Borrow Date: 11/03/2024

ALL BORROWS

	🕒 On Time Name: Digital Multimeter UT89XD (01) Model: MULTIMETER KITS	Brand: UNI-T Borrow Date: 11/03/2024 - 11:24 Return Date: 13/03/2024 - 11:24	View
	🕒 1 day late Name: Digital Multimeter UT89XD (01) Model: MULTIMETER KITS	Brand: UNI-T Borrow Date: 14/03/2024 - 9:24 Return Date: 15/03/2024 - 9:24	View

[Log out](#)

10.1.3. Supervisor login:

The screenshot shows the KMITL library supervisor login interface. The top navigation bar includes the KMITL logo, a bell icon, and the user information "John Doe" and "johndoe@kmitl.com". On the left, a sidebar menu for "Supervisor" roles lists "Borrow", "Return", "History", "Requests" (which is selected and highlighted in orange), "Repairs", and "Lendings". Below the sidebar is a "Log out" button. The main content area is titled "Requests" and displays four pending borrow requests for Digital Multimeters. Each request card includes the item name, model, date range, status (Pending), year, brand, requester, location, and three buttons: "Reject", "Approve", and "View".

PENDING BORROWS	URGENT BORROWS	REQUESTED BORROWS
Name: Digital Multimeter UT89XD (01) Model: MULTIMETER KITS ⌚ 12 March 2024 - 20 March 2024	Pending Year: 2023 Brand: UNI-T	Requestor: John Doe Location: HA3 - HM Inventory Zone... [Reject] [Approve] [View]
Name: Digital Multimeter UT89XD (01) Model: MULTIMETER KITS ⌚ 2 March 2024 - 5 March 2024	Pending Year: 2019 Brand: UNI-T	Requestor: John Doe Location: HA3 - HM Inventory Zone... [Reject] [Approve] [View]
Name: Digital Multimeter UT89XD (01) Model: MULTIMETER KITS ⌚ 4 March 2024 - 6 March 2024	Pending Year: 2021 Brand: UNI-T	Requestor: John Doe Location: HA3 - HM Inventory Zone... [Reject] [Approve] [View]
Name: Digital Multimeter UT89XD (01) Model: MULTIMETER KITS ⌚ 5 March 2024 - 7 March 2024	Pending Year: 2019	Requestor: John Doe Location: HA3 - HM Inventory Zone... [Reject] [Approve] [View]

KMITL  

 John Doe
johndoe@kmitl.com

 Borrow

 Return

 History

Supervisor

 Requests

 Repairs

 Lendings



Requests

PENDING BORROWS URGENT BORROWS REQUESTED BORROWS



 Pending
⌚ 12/03/2024 - 20/03/2024
Name: Digital Multimeter UT89XD (01)
Requestor: John Doe
Location: HA3 - HM Inventory Zone...
Year: 2018

Reject Approve



 Pending
⌚ 12/03/2024 - 20/03/2024
Name: Digital Multimeter UT89XD (01)
Requestor: John Doe
Location: HA3 - HM Inventory Zone...
Year: 2021

Reject Approve



 Pending
⌚ 12/03/2024 - 20/03/2024
Name: Digital Multimeter UT89XD (01)
Requestor: John Doe
Location: HA3 - HM Inventory Zone...
Year: 2015

Reject Approve

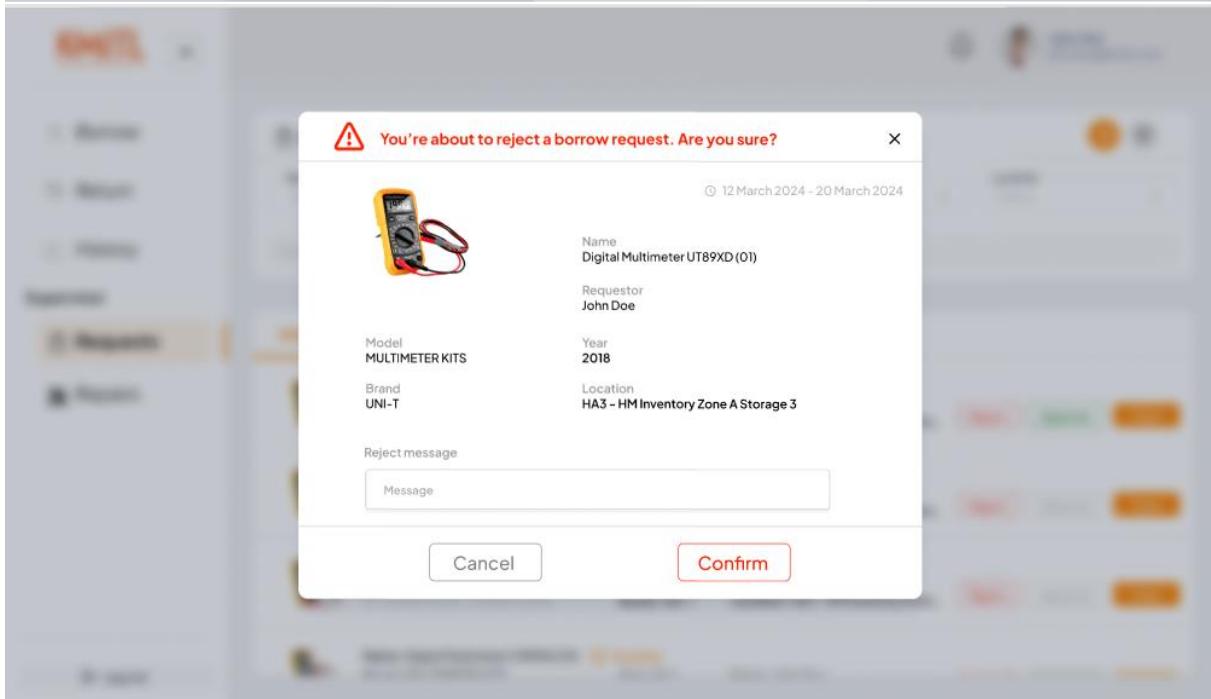
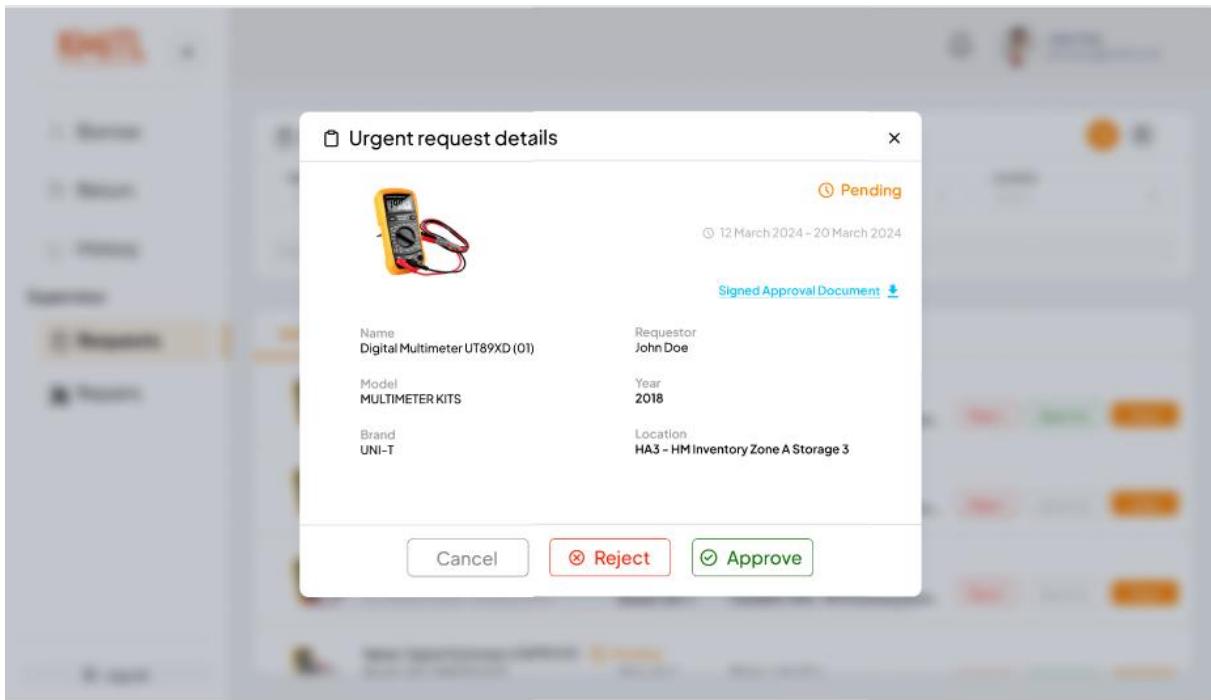
 Request details ×

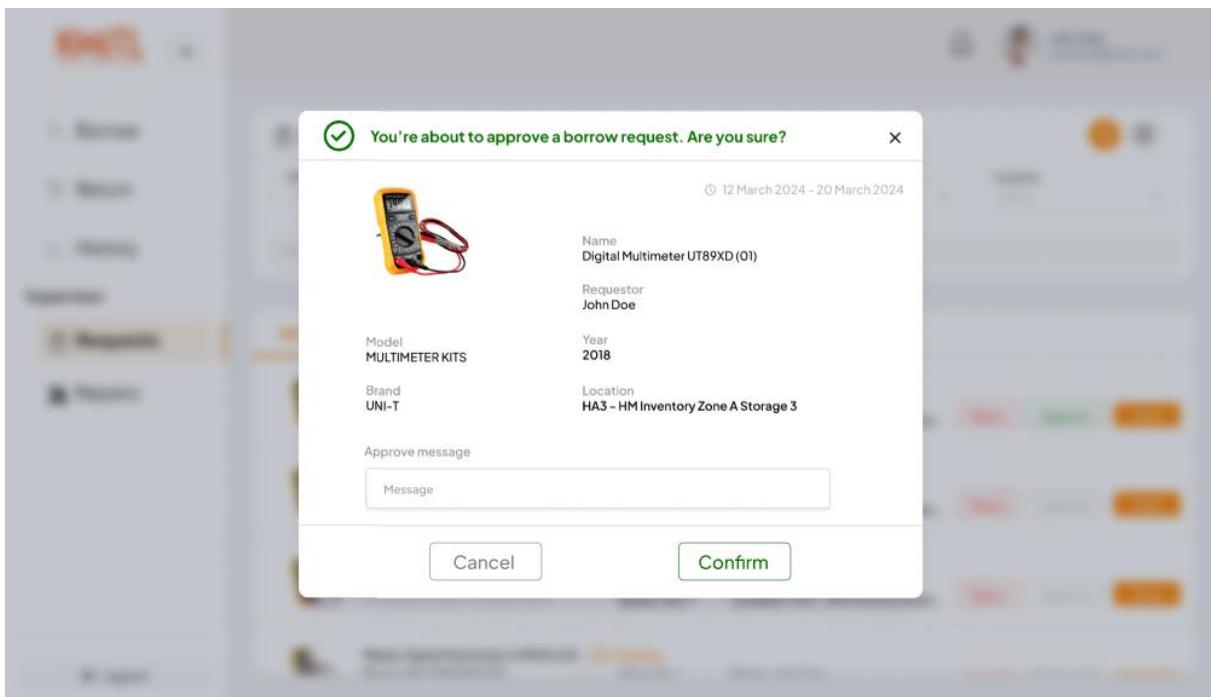


 Pending
⌚ 12 March 2024 - 20 March 2024

Name: Digital Multimeter UT89XD (01)	Requestor: John Doe
Model: MULTIMETER KITS	Year: 2018
Brand: UNI-T	Location: HA3 - HM Inventory Zone A Storage 3

Cancel Reject Approve





KMITL

Borrow

Return

History

Supervisor

Requests

Repairs

Log out

Repairs

Name: Search Repair Date: Select

Sort by: Year Filters Applied: Name: Digital Multimeter

IN REPAIR	REPAIR HISTORY
Name: Digital Multimeter UT89XD (01) Model: MULTIMETER KITS ① 12 March 2024 - /	Year: 2019 Brand: UNI-T Requestor: John Doe Location: HA3 - HM Inventory Zone... View
Name: Digital Multimeter UT89XD (01) Model: MULTIMETER KITS ① 2 March 2024 - /	Year: 2019 Brand: UNI-T Requestor: John Doe Location: HA3 - HM Inventory Zone... View
Name: Digital Multimeter UT89XD (01) Model: MULTIMETER KITS ① 4 March 2024 - /	Year: 2019 Brand: UNI-T Requestor: John Doe Location: HA3 - HM Inventory Zone... View
Name: Digital Multimeter UT89XD (01) Model: MULTIMETER KITS	Year: 2019 Requestor: John Doe Location: HA3 - HM Inventory Zone... View

KMITL  

-  Borrow
-  Return
-  History

Supervisor

-  Requests
-  Repairs

 Log out

Repairs

Name: Search  Repair Date:  Return Date:  Status: 

↑↓ Sort by Year >

Filters Applied:  Status: Returned, in repair

IN REPAIR		REPAIR HISTORY	
	Name: Digital Multimeter UT89XD (01) Model: MULTIMETER KITS  12 March 2024 - 20 March 2024	Year: 2019 Brand: UNI-T	Requestor: John Doe Location: HA3 - HM Inventory Zone...
	Name: Digital Multimeter UT89XD (01) Model: MULTIMETER KITS  2 March 2024 - 5 March 2024	Year: 2019 Brand: UNI-T	Requestor: John Doe Location: HA3 - HM Inventory Zone...
	Name: Digital Multimeter UT89XD (01) Model: MULTIMETER KITS  4 March 2024 - 6 March 2024	Year: 2019 Brand: UNI-T	Requestor: John Doe Location: HA3 - HM Inventory Zone...
	Name: Digital Multimeter UT89XD (01) Model: MULTIMETER KITS	Year: 2019	Requestor: John Doe Location: HA3 - HM Inventory Zone...

 Repaired 

 Broken 

 Pending 

KMITL  

-  Borrow
-  Return
-  History

Supervisor

-  Requests
-  Repairs
-  Lendings

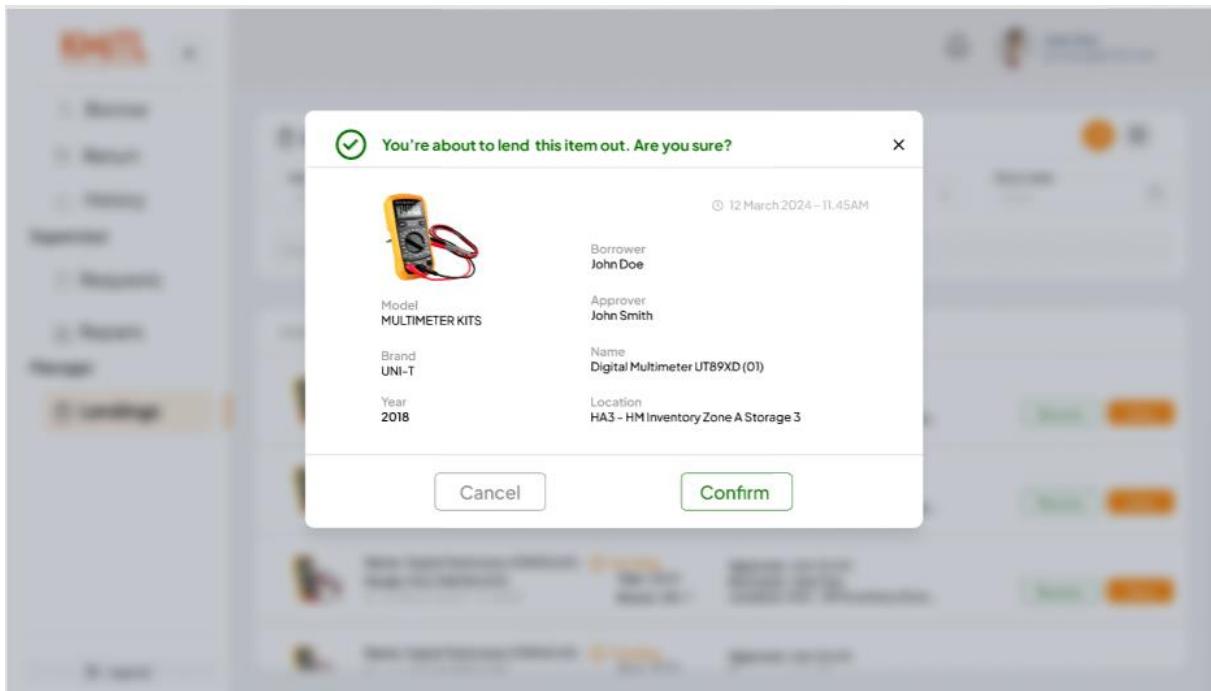
 Log out

Lendings

Name: Search  Borrower: Search  Approver:  Return date: 

Filters Applied:  Location: RCC building, Main...  Brand: This is a long text for ex...

PENDING BORROWS		PENDING RETURNS		HISTORY
	Name: Digital Multimeter UT89XD (01)  Pending Model: MULTIMETER KITS 	Year: 2023 Brand: UNI-T	Approver: John Smith Borrower: John Doe Location: HA3 - HM Inventory Zone...	 Hand Over 
	Name: Digital Multimeter UT89XD (01)  Pending Model: MULTIMETER KITS 	Year: 2023 Brand: UNI-T	Approver: John Smith Borrower: John Doe Location: HA3 - HM Inventory Zone...	 Hand Over 
	Name: Digital Multimeter UT89XD (01)  Pending Model: MULTIMETER KITS 	Year: 2023 Brand: UNI-T	Approver: John Smith Borrower: John Doe Location: HA3 - HM Inventory Zone...	 Hand Over 
	Name: Digital Multimeter UT89XD (01)  Pending Model: MULTIMETER KITS 	Year: 2023	Approver: John Smith Borrower: John Doe Location: HA3 - HM Inventory Zone...	 Hand Over 



KMITL
www.kmitl.ac.th

Borrow Return History Requests Repairs Lendings

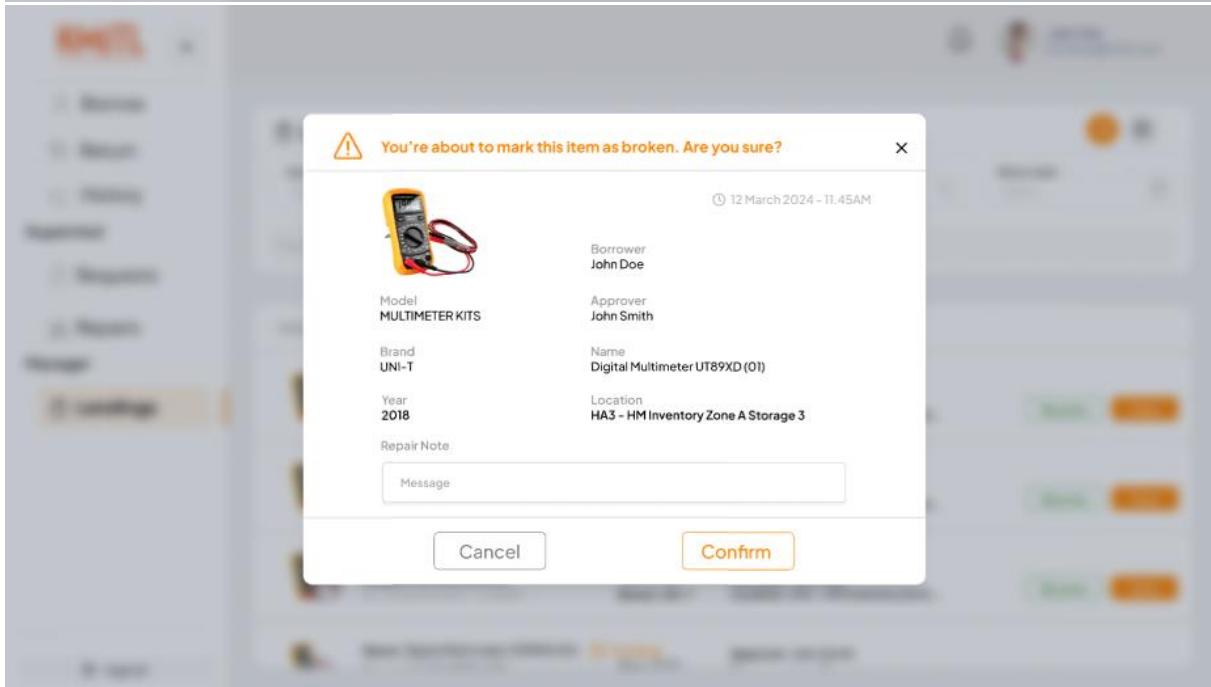
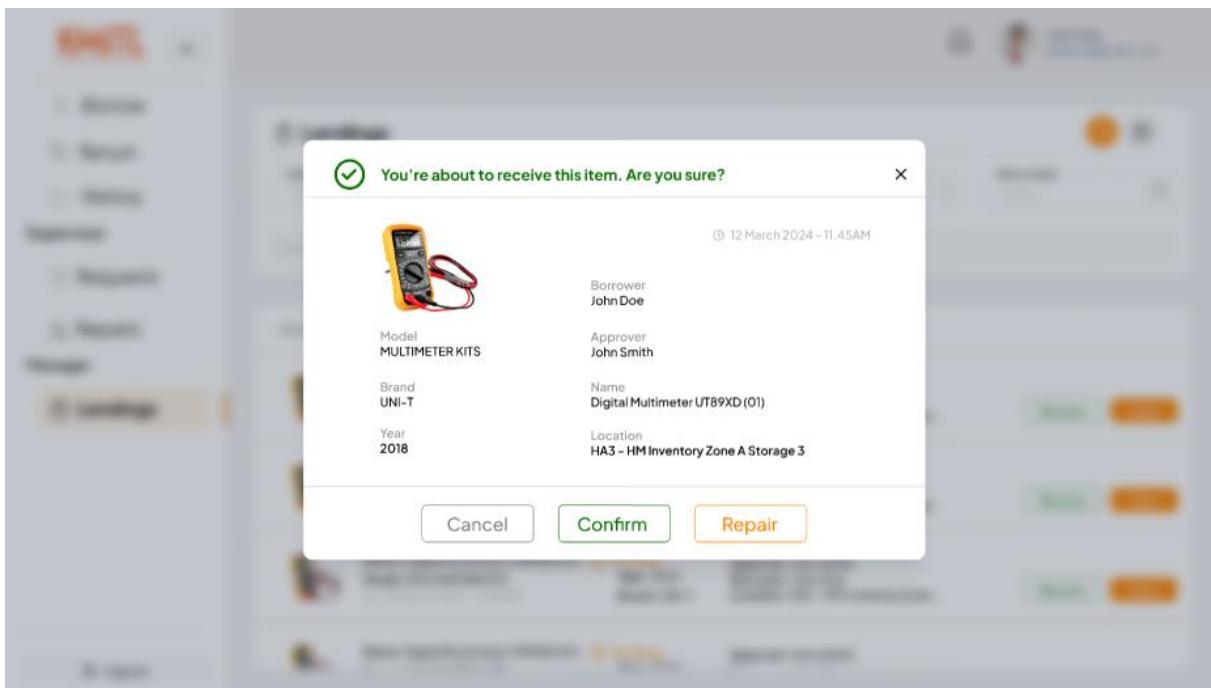
John Doe
john.doe@kmitl.com

Lendings

PENDING BORROWS PENDING RETURNS HISTORY

Name: Digital Multimeter UT89XD (01)	Model: MULTIMETER KITS	⌚ Pending	Year: 2023	Brand: UNI-T	Approver: John Smith	Borrower: John Doe	Location: HA3 - HM Inventory Zone...	Receive	View
Name: Digital Multimeter UT89XD (01)	Model: MULTIMETER KITS	⌚ Pending	Year: 2023	Brand: UNI-T	Approver: John Smith	Borrower: John Doe	Location: HA3 - HM Inventory Zone...	Receive	View
Name: Digital Multimeter UT89XD (01)	Model: MULTIMETER KITS	⌚ Pending	Year: 2023	Brand: UNI-T	Approver: John Smith	Borrower: John Doe	Location: HA3 - HM Inventory Zone...	Receive	View
Name: Digital Multimeter UT89XD (01)	Model: MULTIMETER KITS	⌚ Pending	Year: 2023	Brand: UNI-T	Approver: John Smith	Borrower: John Doe	Location: HA3 - HM Inventory Zone...	Receive	View

Log out



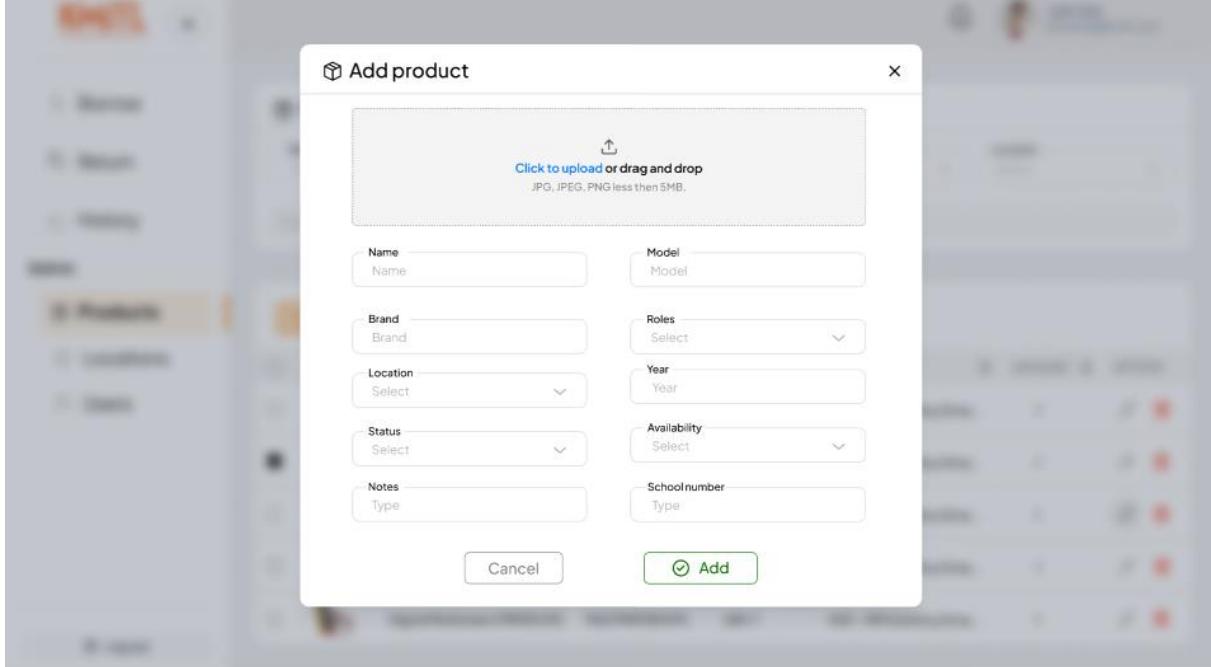
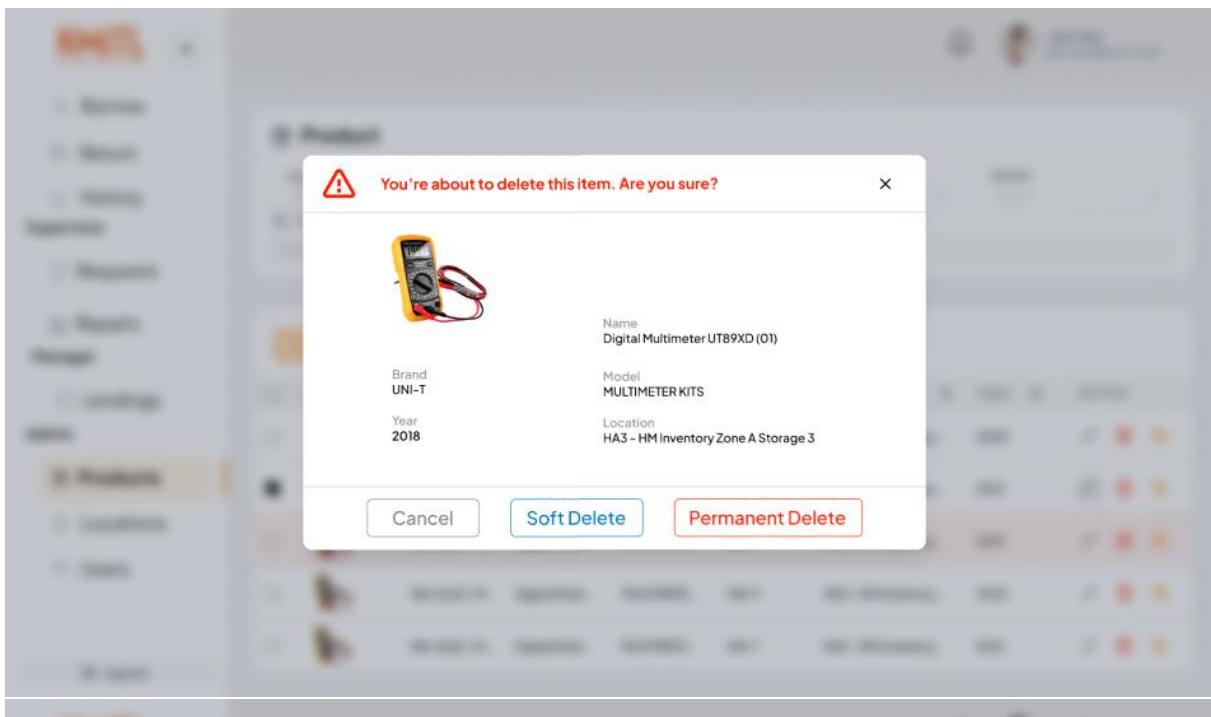
Lendings

PENDING BORROWS	PENDING RETURNS	HISTORY
Name: Digital Multimeter UT89XD (01) Model: MULTIMETER KITS ⌚ 12 March 2024 - 18 March 2024	Location: HA3 - HM Inventory Zone A Storage 3 Year: 2023 Approver: John Smith Brand: UNI-T Borrower: John Doe	<button>View</button>
Name: Digital Multimeter UT89XD (01) Model: MULTIMETER KITS ⌚ 12 March 2024 - 18 March 2024	Location: HA3 - HM Inventory Zone A Storage 3 Year: 2023 Approver: John Smith Brand: UNI-T Borrower: John Doe	<button>View</button>
Name: Digital Multimeter UT89XD (01) Model: MULTIMETER KITS ⌚ 12 March 2024 - 18 March 2024	Location: HA3 - HM Inventory Zone A Storage 3 Year: 2023 Approver: John Smith Brand: UNI-T Borrower: John Doe	<button>View</button>
Name: Digital Multimeter UT89XD (01) Model: MULTIMETER KITS ⌚ 12 March 2024 - 18 March 2024	Location: HA3 - HM Inventory Zone A Storage 3 Year: 2023 Approver: John Smith Brand: UNI-T Borrower: John Doe	<button>View</button>

10.1.4. Admin login:

Product

IMAGE	NO.	MODEL	BRAND	LOCATION	YEAR	ACTION
	RAI-ELEC-M...	Digital Multi...	MULTIMETE...	HA3 - HM Inventory...	2020	<button>Edit</button> <button>Delete</button> <button>Details</button>
<input checked="" type="checkbox"/>	RAI-ELEC-M...	Digital Multi...	MULTIMETE...	HA3 - HM Inventory...	2015	<button>Edit</button> <button>Delete</button> <button>Details</button>
	RAI-ELEC-M...	Digital Multi...	MULTIMETE...	HA3 - HM Inventory...	2019	<button>Edit</button> <button>Delete</button> <button>Details</button>
	RAI-ELEC-M...	Digital Multi...	MULTIMETE...	HA3 - HM Inventory...	2023	<button>Edit</button> <button>Delete</button> <button>Details</button>
	RAI-ELEC-M...	Digital Multi...	MULTIMETE...	HA3 - HM Inventory...	2021	<button>Edit</button> <button>Delete</button> <button>Details</button>



[Borrow](#)
[Return](#)
[History](#)

Supervisor

[Requests](#)
[Repairs](#)

Admin

[Products](#)
Locations
[Users](#)

[Log out](#)

Locations

Name	Search	Sort by	Name >
Filters Applied: X Location: RCC building, Main...			
+ ADD DELETE			
NAME			ACTION
HF2 - HM Inventory Zone F Storage 2	<input type="checkbox"/>	/	trash
HF2 - HM Inventory Zone F Storage 2	<input checked="" type="checkbox"/>	/	trash
HF2 - HM Inventory Zone F Storage 2	<input type="checkbox"/>	/	trash
HF2 - HM Inventory Zone F Storage 2	<input type="checkbox"/>	/	trash
HF2 - HM Inventory Zone F Storage 2	<input type="checkbox"/>	/	trash

Users

Name	Search	Student Code	Search	E-mail	Search	Level	Select					
Filters Applied: X Location: RCC building, Main...												
+ ADD DELETE Export Excel												
FIRST NAME		LAST NAME		STUDENT CODE		TELEPHONE		EMAIL		ROLE		ACTION
Myat Noe	Hoonpong simanont	65011402		0900060151		suparat.pu@kmitl...		STUDENT		/	trash	refresh
Lilinth	Eiampikul	64011444		0805925565		sarucha.ya@kmitl...		ADMIN		/	trash	refresh
Tanakorn	Youngmeesuk	66011066		0651184001		atchariya.bo@km...		SUPERV...		/	trash	refresh
Thitiphan	Chenrukmatupoom	66011056		0823695060		66110044@kmitl...		ADMIN		/	trash	refresh
Suphason	Suttinon	65011563		0972356319		66016100@kmitl...		STUDENT		/	trash	refresh

KMITL 

- Borrow
- Return
- History

Supervisor

- Requests
- Repairs

Admin

- Products
- Locations
- Users**



History: John Doe

Name	Approver	Request Date	Location
<input type="text"/> Search	<input type="text"/> Search	<input type="text"/> Select	<input type="text"/> Select
<input type="button"/> Sort by Date borrowed <input type="button"/>			
Filters Applied: <input type="checkbox"/> Location: RCC building, Main... <input type="checkbox"/> Brand: This is a long text for ex...			

REQUESTED BORROWS

	Name: Digital Multimeter UT89XD (01) Model: MULTIMETER KITS <small>① 12 March 2024 - 20 March 2024</small>	⚠ Rejected Year: 2023 Brand: UNI-T	Rejected by: John Doe Location: HA3 - HM Inventory Zone... 
	Name: Digital Multimeter UT89XD (01) Model: MULTIMETER KITS <small>② 2 March 2024 - 5 March 2024</small>	✓ Approved Year: 2019 Brand: UNI-T	Approved by: John Doe Location: HA3 - HM Inventory Zone... 
	Name: Digital Multimeter UT89XD (01) Model: MULTIMETER KITS <small>③ 4 March 2024 - 6 March 2024</small>	✓ Approved Year: 2021 Brand: UNI-T	Approved by: John Doe Location: HA3 - HM Inventory Zone... 
	Name: Digital Multimeter UT89XD (01) Model: MULTIMETER KITS <small>④ 10 March 2024 - 15 March 2024</small>	⚠ Rejected Year: 2019 Brand: UNI-T	Rejected by: John Doe 

KMITL 

- Borrow
- Return
- History

Supervisor

- Requests
- Repairs

Admin

- Products**
- Locations
- Users



History: Digital Multimeter UT89XD (01)

Name	Email	Student code
<input type="text"/> Search	<input type="text"/> Search	<input type="text"/> Search
<input type="button"/> Sort by Date borrowed <input type="button"/>		
Filters Applied: <input type="checkbox"/> Location: RCC building, Main... <input type="checkbox"/> Brand: This is a long text for ex...		

HISTORY

	Name: John Doe Student code: 05478642 <small>① 12 March 2024 - 20 March 2024</small>	Telephone: 0933263356 E-mail: 63011164@kmitl.ac.th ⚠ 2 Days late 
	Reparation <small>② 6 March 2024 - 11 March 2024</small>	✓ Repaired 
	Name: John Doe Student code: 05478642 <small>③ 2 March 2024 - 5 March 2024</small>	Telephone: 0933263356 E-mail: 63011164@kmitl.ac.th ✓ On time 
	Name: John Doe Student code: 05478642 <small>④ 10 March 2024 - 15 March 2024</small>	Telephone: 0933263356 

10.2. Mobile

10.2.1. General:

The mobile application wireframes for E-Borrow are displayed across three screens:

- Landing Page:** Features the KMITL logo at the top right. The main title "E-Borrow" is in orange, followed by "Where borrowing meets brilliance!" in black. Below the title is a description: "Your campus borrowing hub! Instantly access and borrow items from textbooks to tech, making student life simpler." A "Get started" button is at the bottom.
- Login/Signup Screen:** Displays the KMITL logo at the top right. It includes fields for "Student code or email" and "Password". Below these are "Forgot password?" and "Sign In" buttons. A "Sign in with google" button is also present. At the bottom, it says "Are you new? [Create an Account](#)".
- User Profile Screen:** Shows the KMITL logo at the top right. The title "User profile" is at the top left. It features a circular profile picture of a woman with a red rose. Buttons for "Upload new photo" and "Remove" are below the photo. A note states "At least 800 x 800 px recommended. JPG or PNG is allowed". The "Personal Info" section includes fields for "Full Name" (John Doe), "Student Number" (45872530), "Tel" (6622249773), "Level" (Student), and "Mail" (johndoe@example.com). An "Edit" button is located next to the personal info fields.

10.2.2. Student login:

Borrow Details (Main Screen):

- Name: Select
- Model: Select
- Brand: Select
- Location: Select

Digital Multimeter UT89XD (01) Product Details:

- Model: MULTIMETER KITS
- Brand: UNI-T
- Location: HA3 - HM Inventory Zone the sec...

Borrow Details (Modal):

- Name: Digital Multimeter UT89XD (01)
- Model: MULTIMETER KITS
- Brand: UNI-T
- Location: HA3 - HM Inventory Zone the sec...
- Select a pickup time

Pickup Date Calendar:

SUN	MON	TUE	WED	THU	FRI	SAT
29	30	31	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	1	2	3	4

Borrow Details (Date Picker Modal):

Time: 09:32 AM

Borrow Details (Urgent Request Modal):

YOU ARE MAKING AN URGENT BORROW REQUEST!
ARE YOU SURE YOU WANT TO CONTINUE?

Pending Borrows (Bottom Section):

- Digital Multimeter UT89XD (01) - Pending (12/03/24 - 20/03/24)
 - Model: MULTIMETER KITS
 - Brand: UNI-T
 - Location: HA3 - HM Inventory Zone the sec...
- Digital Multimeter UT89XD (01) - Pending (12/03/24 - 20/03/24)
 - Model: MULTIMETER KITS
 - Brand: UNI-T
 - Location: HA3 - HM Inventory Zone the sec...

10.2.3. Supervisor login:

Left Screenshot: Approve Borrow Request

You're about to approve a borrow request. Are you sure?

Name: Digital Multimeter UT89XD (01)
Amount: 1
Requestor: John Doe

Approve message: Message [Text Input]

Buttons: Cancel, Confirm

Middle Screenshot: Repairs - IN REPAIR

Repair ID: John Doe (In repair) 12/03/24 - /

Item Details: Name: Digital Multimeter UT89XD (01), Model: MULTIMETER KITS, Brand: UNI-T, Location: HA3 - HM Inventory Zone..., Year: 2023

Right Screenshot: Repairs - REPAIR HISTORY

Repair ID: John Doe (Repaired) 12/03/24 - 20/03/24

Item Details: Name: Digital Multimeter UT89XD (01), Model: MULTIMETER KITS, Brand: UNI-T, Location: HA3 - HM Inventory Zone..., Year: 2023

Repair ID: John Doe (Broken) 12/03/24 - 20/03/24

Item Details: Name: Digital Multimeter UT89XD (01), Model: MULTIMETER KITS, Brand: UNI-T, Location: HA3 - HM Inventory Zone..., Year: 2023

Left Screenshot: Lendings - PENDING BORROWS

You're about to hand over this item. Are you sure?

Approver: John Smith (Pending) 12 March 2024 - 11:45AM

Item Details: Name: Digital Multimeter UT89XD (01), Model: MULTIMETER KITS, Brand: UNI-T, Location: HA3 - HM Inventory Zone..., Year: 2023

Approver: John Smith

Borrower: John Doe

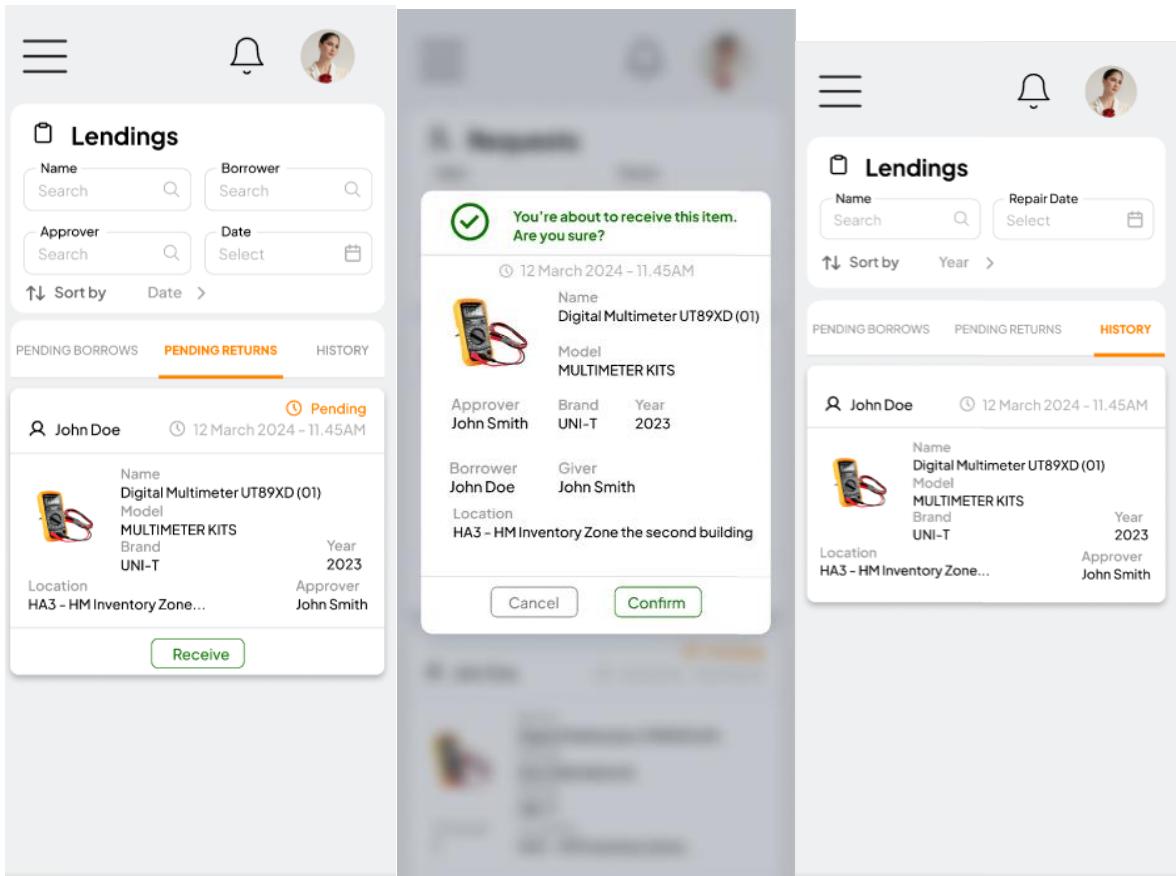
Location: HA3 - HM Inventory Zone the second building

Buttons: Hand over, Cancel, Confirm

Right Screenshot: Lendings - PENDING RETURNS

Pending borrows: John Doe (Pending) 12 March 2024 - 11:45AM

Item Details: Name: Digital Multimeter UT89XD (01), Model: MULTIMETER KITS, Approver: John Smith, Borrower: John Doe, Location: HA3 - HM Inventory Zone the second building, Year: 2023



10.2.4. Admin login:

The left screenshot shows the main navigation menu for 'KMITL'. It includes sections for 'Links' (Borrow, Return, History), 'Supervisor' (Requests, Repairs), and 'Admin' (Products, Location, Users). The 'Products' link is highlighted. At the bottom is a 'Log out' button.

The right screenshot shows the 'Products' screen. It has search and filter fields for Name, Model, Brand, and Location. Buttons for '+ ADD' and 'DELETE' are at the top. Below is a table for 'PRODUCTS' with two entries. Each entry includes a small image of a multimeter, brand (UNI-T), year (2020), and location (HA3 - HM Inventory...). Action icons (edit, delete, refresh) are to the right of each row.

A modal window titled 'Add product' is open on the right. It has a file upload area ('Click to upload or drag and drop'), fields for 'Name' (Name), 'Model' (Model), 'Brand' (Brand), and 'Roles' (Select), and 'Cancel' and 'Add' buttons.

Add product

Location: Select

Year: Select

Status: Select

Availability: Select

Notes: Notes

School number: Schoolnumber

Locations

Name: Select

Sort by: Name >

LOCATIONS		
<input type="checkbox"/> Name: HA3 - HM Inventory Zone A Storage 3	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
<input checked="" type="checkbox"/> Name: HA3 - HM Inventory Zone A Storage 3	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
<input type="checkbox"/> Name: HA3 - HM Inventory Zone A Storage 3	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
<input type="checkbox"/> Name: HA3 - HM Inventory Zone A Storage 3	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
<input type="checkbox"/> Name: HA3 - HM Inventory Zone A Storage 3	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
<input type="checkbox"/> Name: HA3 - HM Inventory Zone A Storage 3	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>

Users

Name: Search Student Code: Search

E-mail: Search Level: Select

USERS	
First Name: Myat Noe Last Name: Hoonpong simanont Email: suparat.pug@kmitl.ac.th Level: STUDENT	Student Code: 65011402 Telephone: 0900060151
First Name: Myat Noe Last Name: Hoonpong simanont Email: suparat.pug@kmitl.ac.th Level: STUDENT	Student Code: 65011402 Telephone: 0900060151
First Name: Myat Noe	Student Code: 65011402

History: John Doe

Name: Search Approver: Search

Request Date: Select Location: Select

REQUESTED BORROWS

Rejected by: John Smith	Rejected: 12/03/24 - 20/03/24
Rejected by: John Smith	Rejected: 12/03/24 - 20/03/24
Name: Digital Multimeter UT89XD (01) Model: MULTIMETER KITS Brand: UNI-T Location: HA3 - HM Inventory Zone...	
Year: 2023	

Approved by: John Smith Approved: 2/03/24 - 5/03/24

Approved by: John Smith	Approved: 2/03/24 - 5/03/24
Name: Digital Multimeter UT89XD (01) Model: MULTIMETER KITS Brand: UNI-T Location: HA3 - HM Inventory Zone...	
Year: 2023	

History: Digital Multimeter UT89XD (01)

Name: Search Student code: Search

Email: Search Date: Sort by

REQUESTED BORROWERS

BORROWER	STATUS	DATE
John Doe	2 Days late	12/03/24 - 20/03/24
John Doe	On time	2/03/24 - 7/03/24
John Doe	On time	2/03/24 - 7/03/24

REQUESTED BORROWS

BORROW	STATUS	DATE
John Doe	2 Days late	12/03/24 - 20/03/24
John Doe	On time	2/03/24 - 7/03/24
John Doe	On time	2/03/24 - 7/03/24

10.3. Links to screens

Pc: <https://www.figma.com/file/rAbnR0flHkCGDIqVOcOKpa/Inventory-system-desktop?type=design&mode=design&t=vOCWCuKTRJqVi1gD-1>

Mobile: <https://www.figma.com/file/acxboRAx3vk0i3YaA4kCX8/Inventory-system-mobile?type=design&mode=design&t=vOCWCuKTRJqVi1gD-1>

11. Conceptualization of the Application

11.1. Deliverables

We discussed that at the end we must give a project handover. This will be two different files. One for the user on how to navigate and use the application and the other for developers to understand the code and the tools we used. Basically two manuals as project handover including our code.

11.2. Use case diagram

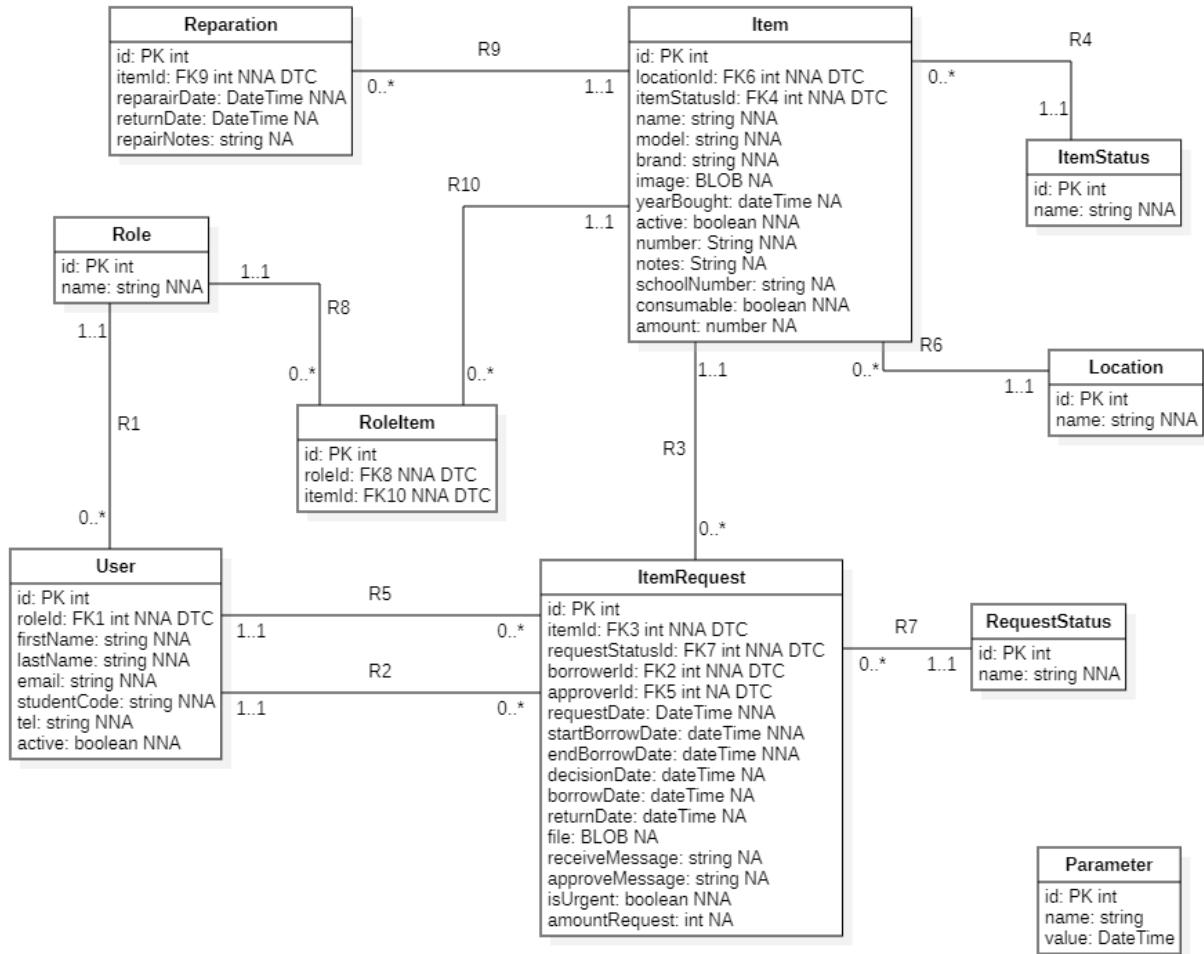
First, we will discuss our use case diagram. This is a visual representation of everything that a user should be able to do within our application.

Here you can see we have an actor called "user". This can be a student or lecturer. We also have an actor called "supervisor", he can do everything a user can and more. As last we have an actor called "admin", he can do everything a user and a supervisor can and more.



11.3. Datamodel

We will now be discussing what structure we have come up with for the database, using a data model.



11.4. General

As a font we'll be using: [Source Sans 3](#)

And as a color schema we chose some colors that match the school:

- Text: #130E01 (black)
- Background: #FFFFFF (white)
- Accent: #CF4307
- Primary: #FF8400
- Secondary: #FFF5D6 (light yellow-ish)

12. References

- Austin, D. D. (2023, October 30). *Medium*. Retrieved from Why Tailwind CSS Is 48% Better For Performance Than CSS-In-JS:
<https://medium.com/coding-at-dawn/why-tailwind-css-is-48-better-for-performance-than-css-in-js-93c3f9fd59b1#:~:text=It's%20an%20undeniable%20fact%20that,Block%2DElement%2DModifier>.
- Chaudhari, M. (2023, July 4). *Why should you use TypeScript for Next.js?* Retrieved from Medium: <https://mayank1513.medium.com/why-should-you-use-typescript-for-next-js-7f0698ed8b2>
- kmitl. (n.d.). *about kmitl*. Retrieved from kmitl: <https://www.kmitl.ac.th/>
- Osnat, R. (2020, 12 23). *Kubernetes Security Basics and 10 Essential Best Practices*. Retrieved from Cloud native academy:
<https://www.aquasec.com/cloud-native-academy/kubernetes-in-production/kubernetes-security-best-practices-10-steps-to-securing-k8s/>
- Rauch, G. (n.d.). *The easiest way to work*. Retrieved from Prisma.io:
<https://www.prisma.io/nextjs>
- Salman. (2023, October 25). *Innovating with Next.js: JavaScript vs. TypeScript – A Developer’s Guide*. Retrieved from Medium:
<https://medium.com/@salluarsh/whether-to-use-javascript-or-typescript-to-build-next-js-af6f8b7ad4f3#:~:text=I%20personally%20recommend%20using%20TypeScript,maintainable%20and%20bug%2Dfree%20code>
- Vulnerability Scanning Tools*. (n.d.). Retrieved from owasp:
https://owasp.org/www-community/Vulnerability_Scanning_Tools