Thesis e-borrowing web application

University Thomas More Geel

Jarne Dirken: r0882161

Sohaib Ibenhajene: r0883629

Kobe Vandendijck: r0902368

CAMPUS

Geel

Technology
Elektronics-ICT / Applied informatics

Internship

Course unit: Internship

Educational activity: Internship

Academic year 2023-2024

Introduction

Before embarking on this project, we want to express our gratitude for this incredible opportunity. The prospect of an international internship abroad is something we never imagined, and we are truly grateful for the chance to immerse ourselves in such a transformative experience.

Our primary goal is to leave a lasting positive impression of the students from Thomas More Geel at the University KMITL.

One of the key aspects of our internship journey is the hands-on approach. Here, they understand that true learning and growth comes from actively engaging in tasks and projects. Our aim is to gather as much experience as possible while honing our skills and craftsmanship.

Being in Thailand presents us with an honor: the opportunity to learn from the talented individuals here and witness their unique problem-solving approaches. We expand our horizons by enhancing our problem-solving abilities, sharpening our critical thinking skills, and refine our communication prowess.

We are excited about this journey and look forward to all the experiences and knowledge that await us.

Table of contents

Contents

Responsibilities

Introduction: Jarne Dirken

What: Text from our project plan

Who: Text from our project plan

Where: Sohaib Ibenhajene

When: Text from our project plan

Why: Jarne Dirken

Preparation: Jarne Dirken

Architecture and design: Jarne Dirken

Development process: Sohaib Ibenhajene

Visuals and interface: Jarne Dirken

Hosting: Kobe Vandendijck

Walkthrough: Kobe Vandendijck

Conclusion: Jarne Dirken

List of used abbreviations and symbols

Abbreviation	**Definition**
KMITL	King Mongkut's Institute of Technology Ladkrabang
ORM	Object Relational Mapping
API	Application Programming Interface

List of used images

1 What

1.1 Problem

KMITL has a wide variety of hardware components that are open for students to borrow. These items can range from small resistors all the way up to a laptop. Because of the number of items, the school decided to make an inventory management system. This system can be used by students to borrow different types of items that they need for a presentation, research, or personal use. All of these borrows can be monitored by the application, so in case of any missing or damaged items, you'll always know who last borrowed it.

Over the years their inventory has kept on growing which made it very difficult to keep managing everything with their very outdated web application. The existing system, while effective in its primary goal, exhibits several limitations that require attention. Some examples of why the current system isn't effective:

- 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 minimalistic, 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.

1.2 Our goal

After researching the current situation and system, we were able to completely understand the situation. Instead of only looking at it from the perspective of a student, we decided that it was best to research both sides of the situation.

For students like me, using the current system can be frustrating. For one thing, it doesn't show when an item that's already borrowed will be available again. Plus, the whole app isn’t mobile-friendly at all, which is a major hassle since most of us use our phones for pretty much everything. On top of that, the app just feels old and slow. It’s like everything takes a long time to load, which really isn’t ideal when you’re trying to get things done quickly.

From a teacher's viewpoint, the existing system falls short in several crucial areas. Firstly, it lacks the functionality to mark items as broken and to indicate that they are under repair. This missing feature complicates the management of resources. Additionally, the system does not support a comprehensive transaction flow. This means there’s no seamless way to track each step of an item's borrowing process, making it difficult to pinpoint responsibility in cases of defects. Moreover, the system does not allow for the bulk import of data from Excel sheets, forcing us to enter each item individually. This is not only time-consuming but also prone to errors, hindering effective resource management.

During this internship, our main goal is to give both the students and the staff ease of mind when they want to borrow an item. Knowing that the web application will take care of all planning and monitoring. Our plan is to make a web application specifically for KMITL students and staff. We’re striving to create a platform that will be very straightforward and easy to use for all kinds of students and staff. It’ll have a lot of extra features to make the user experience of the application that much more enjoyable. So, this web app is all about ensuring seamless collaboration between students and staff, while also taking over the more time-consuming and stressful aspects of the inventory management system.

2 Who

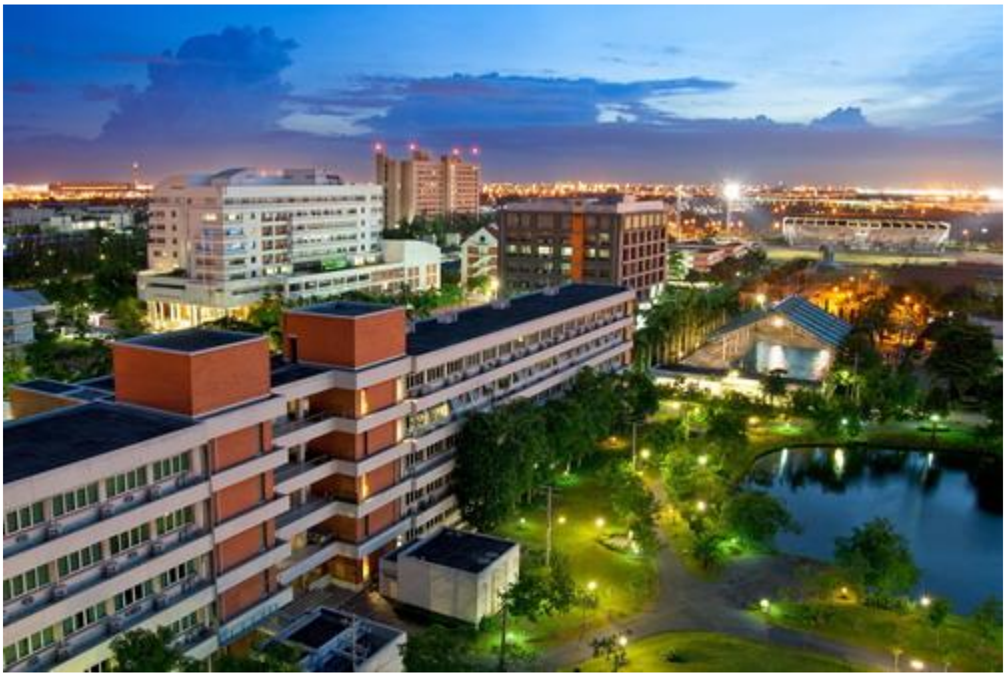
KMITL in other words, King Mongkut's Institute of Technology Ladkrabang.

The people that came up with the issue were the people that used the current application. They are the ones that realized that there were some important features missing and that the application is not at a level where it is supposed to be. We have also talked to some students about it, and they don’t see a big problem with the current system because some of them can abuse this system by borrowing items and never actually returning them. These kinds of things should not be possible, and we are going to solve this in our application.

Our core target public comprises both students and staff members at KMITL, who will be the primary users of our system. Students will get to see a way smoother process for borrowing items, making it tough for anyone to hang onto items they should’ve returned. As for the staff, they’ll find it way easier to keep tabs on everything, from what each student is currently borrowing to any repairs that have to be made to items. With the help of our application, 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.

3 Where

KMITL, positioned in Lat Krabang, Bangkok, Thailand. A university where we are doing our internship.



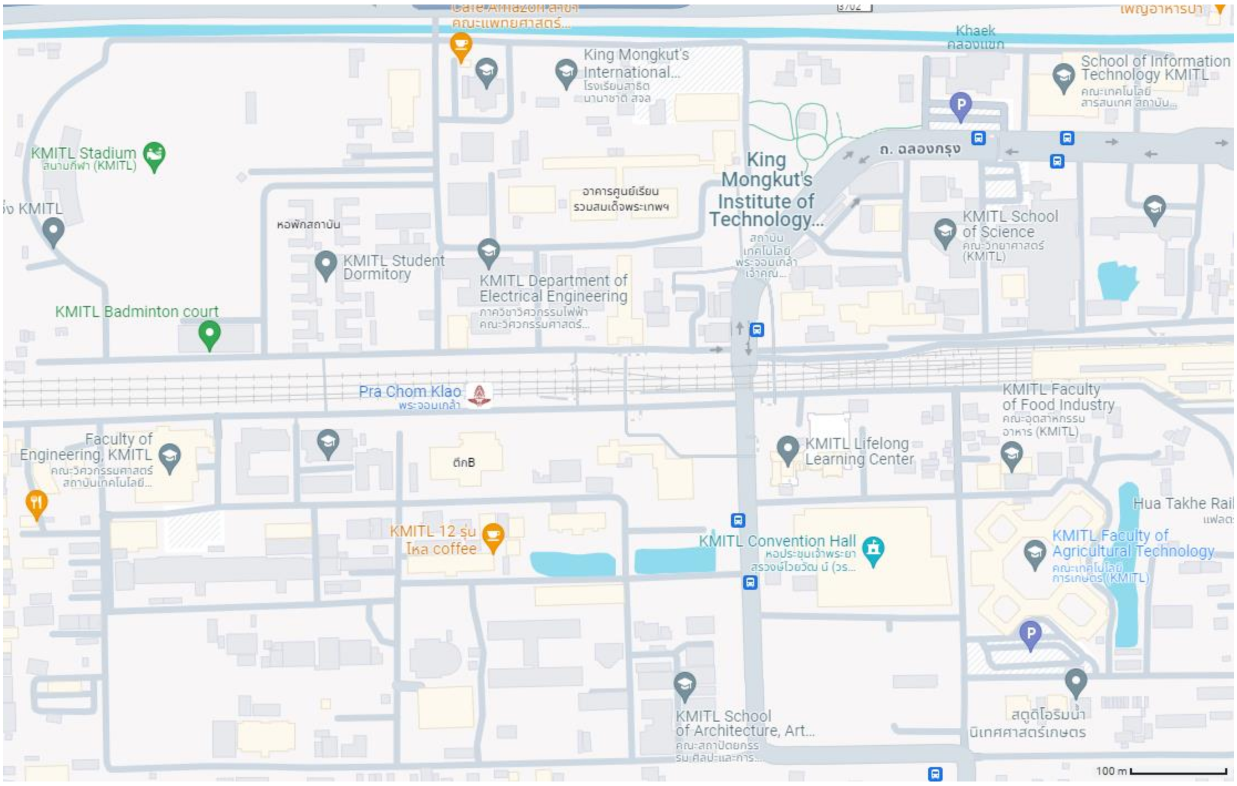
(KMITL, n.d.)

Figure 1: Picture of the school KMITL.

We've been set up in the E12 building at the Faculty of Engineering over at KMITL, that's our home lab for this project. It's not just us in there, the lab is shared with other students from different years who are studying in the same field. This setup is very handy because it means we can easily bounce ideas off each other and ask questions when we're stuck. We're all here working away on different projects, but it's great to have that kind of brainpower around when you need it. So, yeah, that's where you'll find us most of the time, deep in our work and helping each other out.

As you can see, KMITL isn't just a typical school with a few buildings, it's more like a small town. The campus spreads over a vast area, complete with its own network of streets connecting various buildings and faculties. It's so large that it even has train tracks running through it. But it's not all about studies, campus life is vibrant and full of options. You'll find multiple canteens offering a wide range of food choices, along with several 7/11 stores for quick snacks. For those looking to stay active, there's no shortage of sports activities—you can play badminton, volleyball, basketball, football, futsal, go running, or even play pétanque. With all these amenities, it's hard to get bored on this campus.

Under this text we have provided an image from google maps just to give you an idea of how big the entire school is.



คณะแพทยศาสตร์...
King Mongkut's International...
โรงเรียนสาธิตนานาชาติ สจล
Khaek
คลองแขก
School of Information Technology KMITL
คณะเทคโนโลยีสารสนเทศ สถาบัน...
KMITL Stadium
สนามกีฬา (KMITL)
KMITL
อาคารศูนย์เรียนรวมสมเด็จพระเทพฯ
King Mongkut's Institute of Technology...
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณ...
ถ. ฉลองกรุง
หอพักสถาบัน
KMITL Student Dormitory
KMITL Department of Electrical Engineering
ภาควิชาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์...
KMITL School of Science
คณะวิทยาศาสตร์ (KMITL)
KMITL Badminton court
Pra Chom Klao
พระจอมเกล้า
KMITL Faculty of Food Industry
คณะอุตสาหกรรมอาหาร (KMITL)
Faculty of Engineering, KMITL
คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยี...
ตึกB
KMITL Lifelong Learning Center
Hua Takhe Rail
แฟลต
KMITL 12 รุ่น
ไหล coffee
KMITL Convention Hall
หอประชุมเจ้าพระยาสุรวงษ์ไวยวัฒ น์ (วร...
KMITL Faculty of Agricultural Technology
คณะเทคโนโลยีการเกษตร (KMITL)
KMITL School of Architecture, Art...
คณะสถาปัตยกรรม ศิลปะและการ...
สตูดิโอริมน้ำ นิเทศศาสตร์เกษตร
100 m

Figure 2: Grounds of KMITL in google maps.

As you can see in the image the entire school ground is huge. It's built around roads and rivers which makes it such a unique place.

4 When

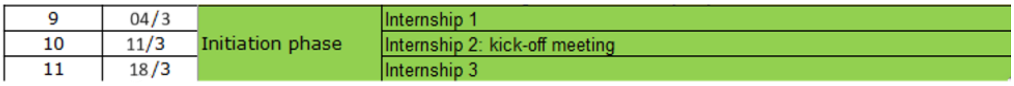
Our project consists of two phases. The initialization phase and the realization phase.

4.1 Initialisation phase

In this phase we will be focusing on our project plan. 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

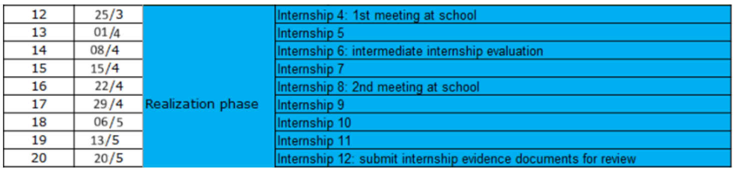
Figure 3: Picture of the first three work weeks (initialization phase).

4.2 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 remaining weeks (10): **25/03 – 27/05**



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

Figure 4: Picture of the remaining work weeks (realization phase).

5 Why

The old system is not doing its job, it's unclear and not well structured. It also has a lot of security vulnerabilities that students take advantage of. Also, the role management is not handled well. There are only 2 roles in the current system. There was an occasion where one of the administrators had a master student use the system, he accidentally deleted 20 records because there are no different roles other than admin and user.

The reason all of this is possible is because of the implementation of a not fully functioning program. In the current program they didn't put in enough thought for it to work properly. There are too many missing features that should have been included.

Currently the students and staff have been affected by plenty of shortcomings of the application such as not being able to use the app on mobile devices. Also, there's no template or easy way to handle urgent borrowing in the current system. Currently there isn't even a template for the approval document, as well as instead of having an upload button on the application, you must fill in a Google form with your own document.

The solution we can give the school is a fully functional application that has all the wants and needs the school can ask for. For this to happen, we need to come prepared with a solid plan. That's why we will spend the first 3 weeks preparing the project, talking to the teachers and students. That way we know what features they want and don't want. So that we can make a difference with our application.

6 How

6.1 Preparation

First, we prepare for our project. This all happens during the initialization phase. In this phase our goal is to research the current system and situation, to gain as much information as possible and come up with a good solution.

Our solution of course being as detailed as possible. From making a good project scope to inspecting the current application to doing research about which programming languages are best to use and making Figma screens. Everything we did during this phase of the project was to make sure that they understood what our solution would be. The most important part is a lot of communication. We have weekly checkups, so they know exactly what we are doing and whether they like our solution or not.

Everything we did during this phase of the project is documented in our "Project Plan" document. Of course, during this phase of the project we also made a Moscow which stated exactly what we were and were not going to do. (Jarne Dirken, 2024)

During the realization phase certain things in the hosting and security part of the project changed due to the circumstances. We had no access to the firewall and vSphere environment, and our Thai supervisors didn't have the time or access to help us with those problems.

6.2 Web application

We can divide our web application into three main parts: architecture & design, development process & visuals and interface.

6.2.1 Architecture and Design

As for the architecture, it remained consistent with the initial plan detailed on pages 53 and 54 of our "Project Plan." This design was chosen as it best met all the needs of our client, KMITL.

As figured below you can see our use case diagram. This provides an overview of the different user interactions that our system supports.

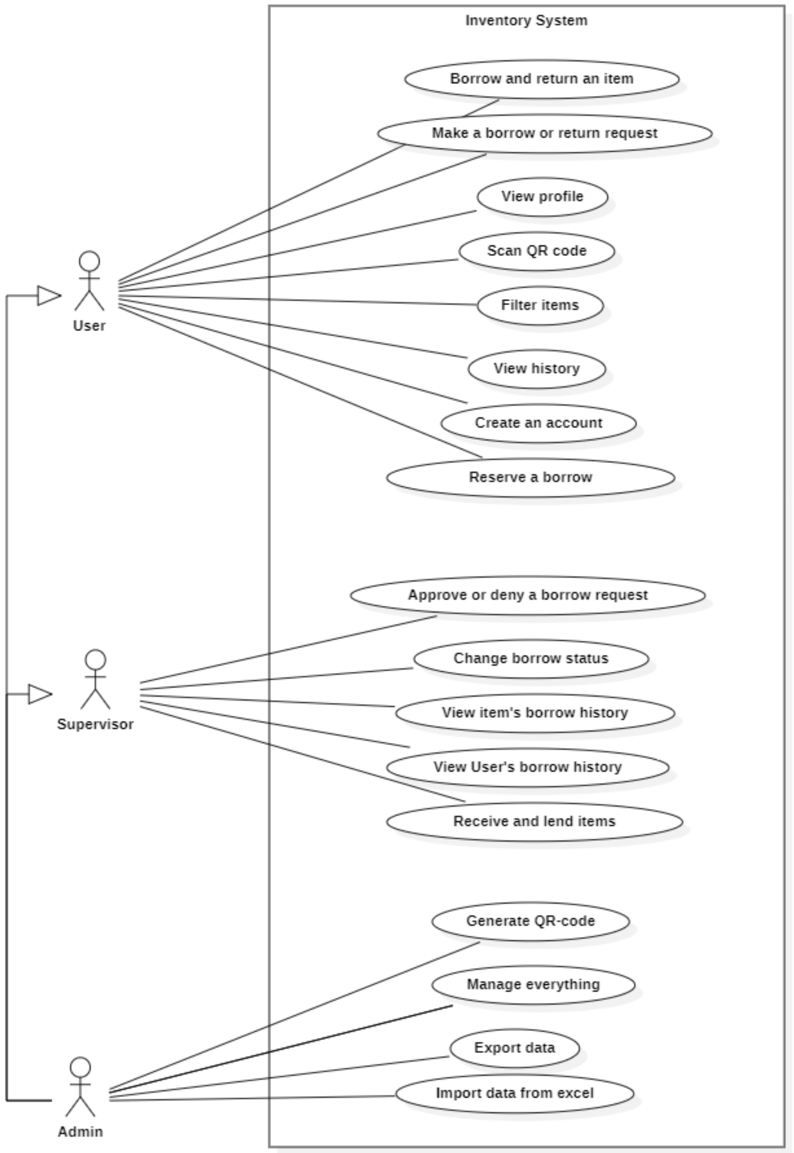


Figure 5: Use case diagram - Provides an overview of the different user interactions that our system supports.

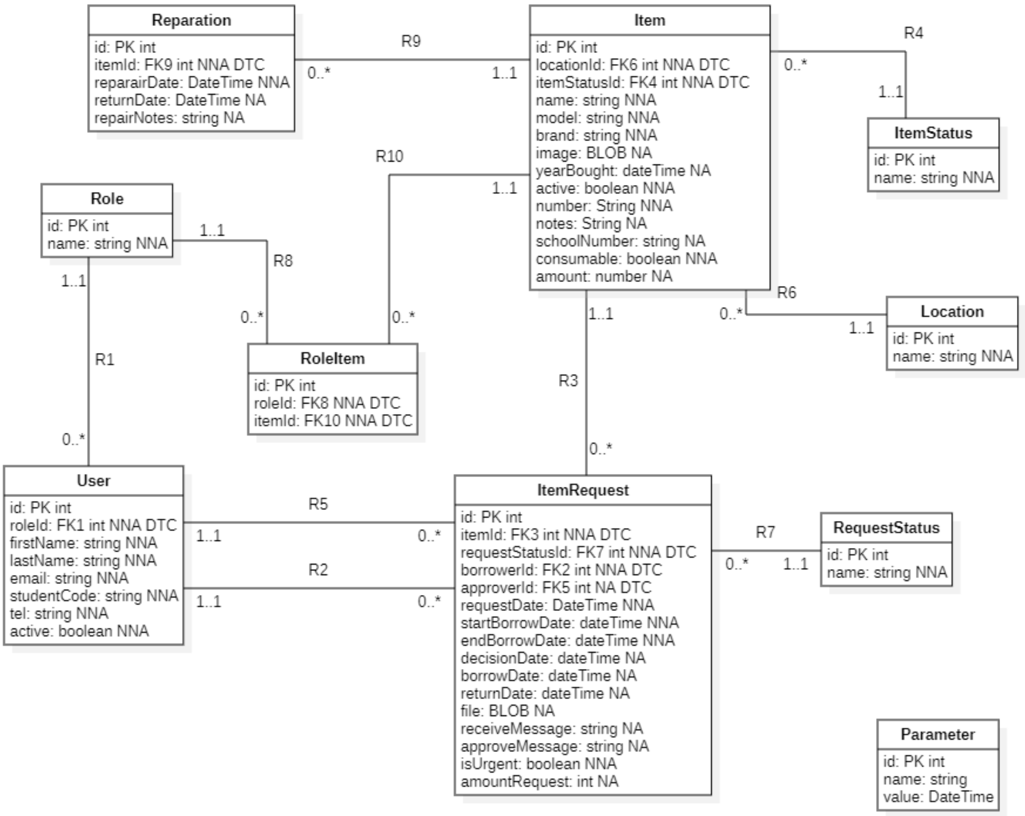


Figure 6: Data model - Outlines the structure of the database and the relationships between different data entities.

As you can see in the figure above, our data model. This outlines the structure of our database and the relationship between different data entities.

We went with this setup because it matched best with all the needs of our client.

6.2.2 Development process

Initial Setup: As we have stated in our project plan pages: 23 to 27. We will be using next.js 14 for our project because this was a requirement made by our customer. We started creating the environment and installing the necessary packages. The packages include:

- Prisma which is used for our ORM.
- Firebase which is used for our authentication as well as online file storage and notification storage.
- GitHub which we will use for storing our project. After we have set up the personal GitLab server we will of course use that with a dedicated pipeline.

Because next.js is a full stack framework there is no need to configure an API as well. As for our database we use PostgreSQL because this was also a requirement set by our client.

We started by dividing the work. Because we work on this project with 2 full stack developers it was easy to divide the workload. Jarne started working on the Prisma schema and dummy data. Sohaib started working on the structure of our application, making all the folders and files ready, when that was done, he started working on the home page and the template of the dashboard of the application. This way the home page and the dashboard template were done as well as the whole database at the same time. This way we could start working together with data on different pages at the same time without running into any conflicts.

Development details:

When we started on the full development of the application, Jarne mostly focused on the user part of the application such as, the borrowing flow all the way from requesting a borrow up until returning it back and checking whether the item is in its original condition or not. While Sohaib mostly focused on the admin part of the application such as item management, creation, import and Qr-code generation to make it easier for the staff to manage all the different types of items.

During development we realized that some of the features that they wanted weren’t quite possible to accomplish with our current knowledge and timeline. Such as automatic number appending for each item and being able to reserve a borrow in advance. In this case we resolved the issue by organizing a meeting with our advisors and simply being clear in our communication and explaining the situation in a respectful manner. This way we were able to come to a conclusion and exclude this from our scope.

If any one of us had any issues during the development process, we could always ask each other for help. If the issue persisted and it was too difficult to resolve, we could always rely on the help of our seniors in our lab and on our advisors.

Testing: We have made a checklist document which was used by the professors and master student to check all of our features. After every check passed successfully, we officially rounded up our project.

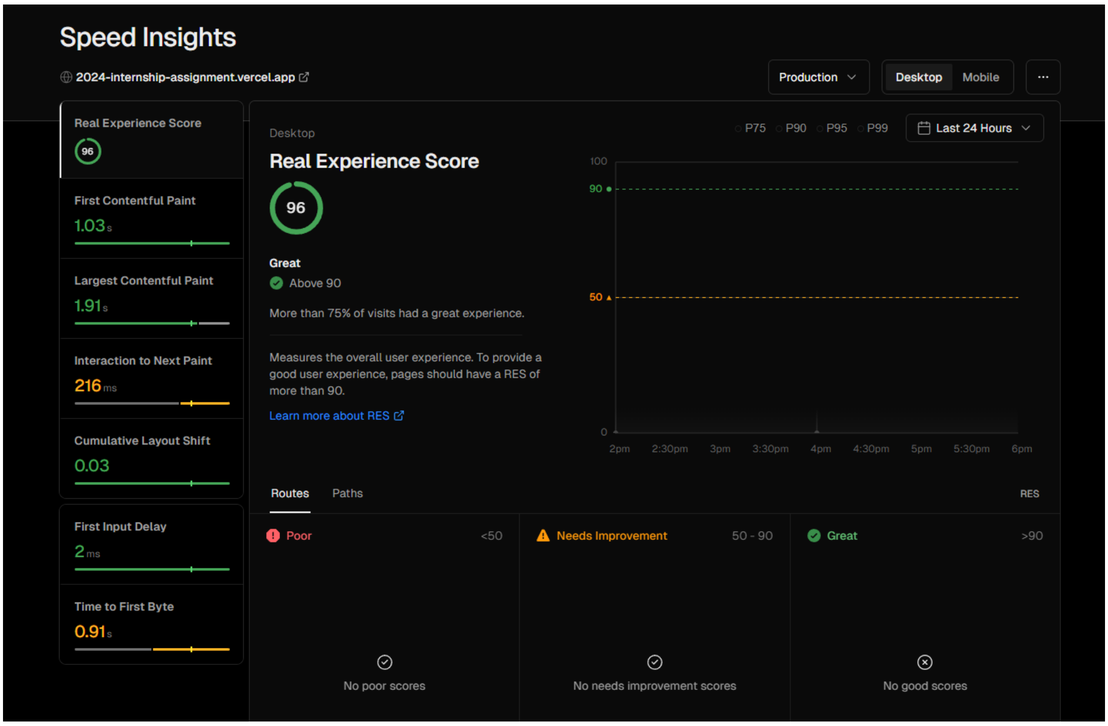
Integration: For our project we went for two types of integration.

One is made for personal use with dummy data hosted on Vercel. This way we can represent our project to the fullest in our portfolio. You can visit our demo website via this link. For email and password, you can visit the GitHub and check the readme for more information.

One is made on the server of KMITL itself. This has been made possible with an Ubuntu server running 2 containers, with a nginx reverse proxy.

All of our API calls are secured internally by who can make which calls and of course we integrated API keys for optimal security.

After integration we ran some speed insight checks for our application, these were the results:



Speed Insights

2024-internship-assignment.vercel.app

Real Experience Score 96

First Contentful Paint 1.03 s

Largest Contentful Paint 1.91 s

Interaction to Next Paint 216 ms

Cumulative Layout Shift 0.03

First Input Delay 2 ms

Time to First Byte 0.91 s

Desktop

Real Experience Score

96

Great

Above 90

More than 75% of visits had a great experience.

Measures the overall user experience. To provide a good user experience, pages should have a RES of more than 90.

Learn more about RES

Poor <50

No poor scores

Needs Improvement 50 - 90

No needs improvement scores

Great >90

No good scores

Figure 7: Image of speed insights on Vercel.

As we can see an overall score of 90+ is not bad, especially on Vercel which is a free hosting service. We have made this our website located in Germany and have only tested it in Thailand so the score should be more stable when you test it in Belgium.

We also provided KMITL with our project handover. Our project handover includes full access to the code we wrote during development, admin access to the firebase project and two documents. One user manual, which explains how everything works and one developer manual, which explains how the code and everything for the future developers works.

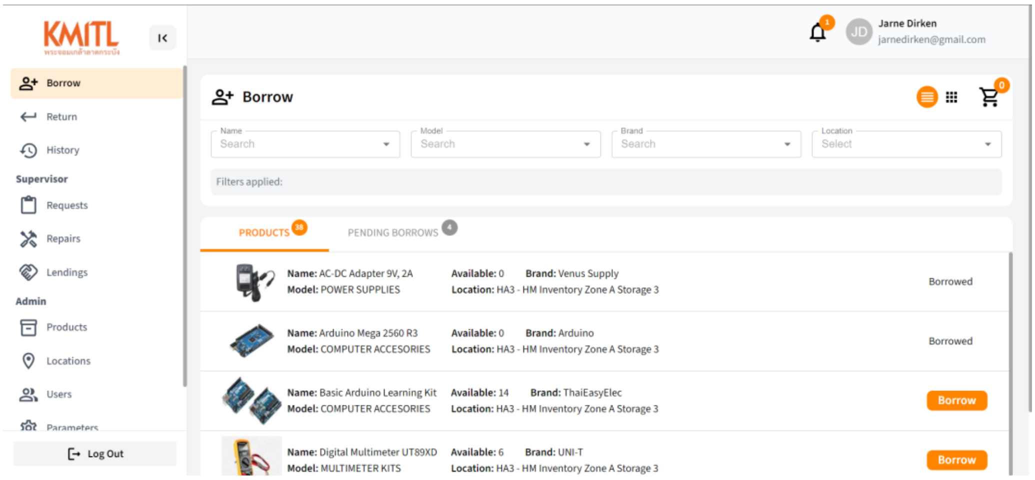
6.2.3 Visuals and interface

In terms of looks, we aligned the color scheme with KMITL's branding, opting for an orange-themed interface. All interface designs are accessible via the following Figma links for further detail:

- Desktop version
- Mobile version

These designs are also documented extensively from pages 30 to 51 in our project plan.

As you can see in the screenshot below, we have provided a clear overview of the core of our application. The dashboard page, this is where the magic happens.



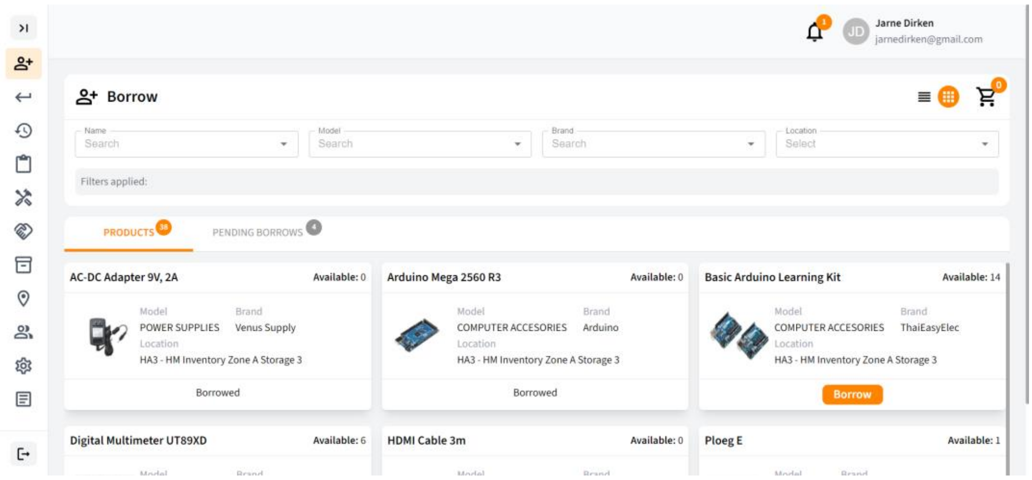
Name: AC-DC Adapter 9V, 2A **Model:** POWER SUPPLIES **Available:** 0 **Brand:** Venus Supply **Location:** HA3 - HM Inventory Zone A Storage 3 Borrowed

Name: Arduino Mega 2560 R3 **Model:** COMPUTER ACCESORIES **Available:** 0 **Brand:** Arduino **Location:** HA3 - HM Inventory Zone A Storage 3 Borrowed

Name: Basic Arduino Learning Kit **Model:** COMPUTER ACCESORIES **Available:** 14 **Brand:** ThaiEasyElec **Location:** HA3 - HM Inventory Zone A Storage 3 Borrow

Name: Digital Multimeter UT89XD **Model:** MULTIMETER KITS **Available:** 6 **Brand:** UNI-T **Location:** HA3 - HM Inventory Zone A Storage 3 Borrow

Figure 8: Screenshot of our web application dashboard overview.



AC-DC Adapter 9V, 2A Available: 0
Model: POWER SUPPLIES Brand: Venus Supply
Location: HA3 - HM Inventory Zone A Storage 3
Borrowed

Arduino Mega 2560 R3 Available: 0
Model: COMPUTER ACCESORIES Brand: Arduino
Location: HA3 - HM Inventory Zone A Storage 3
Borrowed

Basic Arduino Learning Kit Available: 14
Model: COMPUTER ACCESORIES Brand: ThaiEasyElec
Location: HA3 - HM Inventory Zone A Storage 3
Borrow

Digital Multimeter UT89XD Available: 6

HDMI Cable 3m Available: 0

Ploeg E Available: 1

Figure 9: Screenshot of our web application dashboard overview with sidebar collapsed and card view enabled.

We will briefly explain why we went with these exact layout choices.

On the left we see a sidebar for easy access to all the links inside our application. This sidebar is collapsable if the user wants more space.

On the top right we have the notification bell as well as the user profile icon with name and email address. This way they can easily navigate to their profile or view the notifications they have received.

Each page is marked with a clear icon and title so it’s clear to tell what page you’re currently on. We have made each page changeable from a list view to a card view. This way the user can choose which way they want to view a page. When the screen size is reduced, the interface automatically switches to a card layout to ensure that all data fits comfortably on the screen without the need for horizontal scrolling. This adjustment improves the readability and user experience on smaller devices and cannot be manually switched back to the list layout.

6.3 Hosting

The hosting process has been greatly simplified during the course of the internship. We originally wanted to do a CI/CD pipeline that would set up the application from a locally hosted Gitlab server. But because of the circumstances that was not possible. So we settled on an ubuntu server with 2 containers: one for the database and the other for the application.

The 2 containers are launched together through a docker-compose file. And the database is set up and seeded with Prisma during the process of creating the application image.

Because we got no access to the network and the services here, implementing security was tough. We implemented SSL certificates on the application website and tried to keep sensitive data out of the common files, to make it as secure as possible.

6.3.1 Walkthrough

The docker containers are set up together with a docker-compose file. This file first creates an empty PostgreSQL container from the basic PostgreSQL image found on Docker Hub.

Once this is done the custom image for the application is created, during which the database is deployed and seeded with some necessary information. To make this go smoothly we created a custom script to make sure that the PostgreSQL container is running and ready to accept connections.

The containers are connected by a bridged docker network “KMITL_inventory” and the database data is made persistent in a volume. The application container forwards to port 80, this then is forwarded to the nginx proxy manager. The nginx proxy manager connects the traffic from the FQDN to the application.

7 Conclusion

Our internship at King Mongkut's Institute of Technology Ladkrabang has been an incredibly enriching experience. Throughout this project, we faced numerous challenges and opportunities that have significantly contributed to our personal and professional growth. By immersing ourselves in this project, we were able to identify critical areas of improvement and implement effective solutions that enhance the user experience for both students and staff at KMITL.

One of the key achievements of our project was creating a robust web application that incorporates advanced authentication mechanisms, a streamlined borrowing process, and a user-friendly interface optimized for mobile devices. By integrating features such as item condition tracking, bulk data import, and QR code generation, we have greatly improved the efficiency and reliability of the inventory management system. Our commitment to detailed planning, constant communication with stakeholders, and iterative development ensured that we delivered a product that meets the needs and expectations of our users.

Furthermore, this internship has provided us with invaluable insights into real-world software development processes, from initial research and design to implementation and testing. We have honed our technical skills, particularly in using Next.js, Prisma, and Firebase, and have learned the importance of collaboration and adaptability in a dynamic work environment.

In conclusion, our project has not only solved the immediate problems faced by KMITL's inventory management system but has also laid a strong foundation for future enhancements. We are confident that the new system will provide lasting benefits to the KMITL community, fostering a more efficient and enjoyable borrowing experience. This internship has been a transformative journey, and we are proud of the work we have accomplished. We look forward to applying the knowledge and skills we have gained to future projects and continuing to grow as professionals in the field of information technology.

8 References

Jarne Dirken, S. I. (2024, May). *2024-InternshipAssignment.* Opgehaald van Github: https://github.com/JarneDirken/2024-InternshipAssignment/blob/main/files/Project_plan.pdf

KMITL. (n.d.). *History of KMITL*. Retrieved from King Monkut's Insitute of Technology LadKrabang: https://www.kmitl.ac.th/history-kmitl