

Group Name: Horizon Craft

Group Number: 8

Student number	Name & Surname	Qualification Code
22115677	Durand Timothy Pillay	BINCT1
22106896	Yuvashna Dwarika	BINCT1
22383051	Andile Mikel Kayanja Lwanga	ADICT1
21547089	Samukelo Shozi	ADICT1
21818119	Buthanani Buqahqawuli Siyephu	ADICT1
22121080	Savashen Navanathan Govender	BINCT1
22114664	Vairaag Harry Jagunundan	BINCT1
22122564	Kiveshan Naidoo	BINCT1

Table of Contents

Process Model	3
Company description	3
Problem description	3
Process Model	3
REQUIREMENTS ELICITATIONS	4
User requirements	4
System requirements	4
Functional requirements	5
Non-functional requirements	5
Use Case Diagram	6
PROJECT MANAGEMENT	6
Project technology	6
Risks	6
Schedule/timeline	7
Costs	9
TOOLS AND ENVIRONMENT	9
Web/Mobile platform	9
Programming languages	10
Database	10
Operating environment	10
IDEs	10
SOFTWARE DESIGN	11
Flow diagram	11
ERD	12
Sequence diagram	13
Wireframe	14
SOFTWARE CONSTRUCTION/ DEVELOPMENT	14
Back End	14
Database	17
GUI	18
SOFTWARE VERIFICATION AND VALIDATION/TESTING	20
Correct Input:	20
Wrong Input:	20
Testing Types:	20

Laptop Tracking at Durban University of Technology: An Effective Asset Management System

Process Model

Company description

HorizonCraft is a company that specializes in software development of asset management solutions. Located in Durban, our focus is on leveraging sustainable and futuristic technologies to solve challenges in an institutional environment. We aim to deliver high-quality, user-friendly software that enhances student and staff experience within an institutional setting.

Our team comprises a total of eight individuals who have immense skills in identifying business issues, providing high-quality solutions and implementation of after-sales support at reasonable prices. We aim to satisfy our customers best to our ability and provide technological solutions that can enhance overall efficiency within the institutions we assist.

Problem description

The Durban University of Technology situated in Durban has two libraries (which are based at Steve Biko and ML Sultan Campus respectively) which issues students' laptops for a period of 2 hours. The student is required to go to the main library desk and request the librarian for a laptop for the time required.

The student's card is then scanned followed by the laptop to be logged to the user. The student is required to use the laptop within those hours in the library only. After the time has elapsed – the student is required to go back to the library main desk and return the laptop.

The issues experienced by this method is it is not only time-consuming but also prone to errors such as incorrect data entry and loss of records. Additionally, there is no effective way to track the location and status of the borrowed laptops, leading to issues like late returns and asset mismanagement.

The institution is also aware of long queues been reported for booking of laptops the tiresome task for librarians as they have additional tasks to do above booking of laptops. Our team is committed to developing a Laptop Tracking System for Durban University of Technology.

The laptops loaned are valuable assets that require an effective management system to ensure they are utilized efficiently and responsibly whilst assisting library staff to utilize their time efficiently.

Process Model

We will be utilizing a component-based process model for executing the proposed solution. The library already utilizes a system that assists in loaning of the laptops to students. Our proposed system will provide students a platform that they will utilize to advance book a laptop for the required duration. This approach will assist in mitigating the long queues and need for librarians to loan the laptops to students. The component-based process will be done as follows and figure 1.1 provides a basic overview:

☒ Component Analysis

- Identify potential components from current system that the proposed system can utilize.
- Determine the functionality the components will be providing to the system

- ☑ Requirements Modification
 - Using the information provided by the library staff and components of the existing system – HorizonCraft will modify and determine the requirements needed for the proposed solution.
 - Modifications will be made based on the components ability to integrate with the technologies chosen on the new system.
- ☑ System design with reuse
 - Development of the framework based on the system proposed considering the components selected.
- ☑ Development and Integration
 - Development of the proposed system using components, tools and required technologies.
 - Integration of the proposed system with the current system's infrastructure

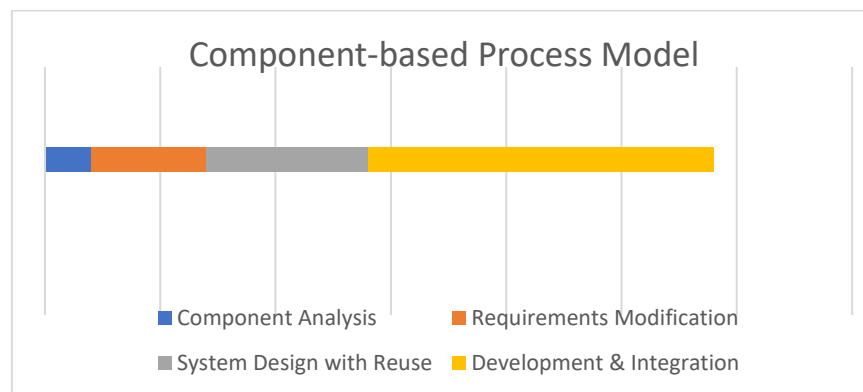


Figure 1.1: Overview of Component-based Process Model

REQUIREMENTS ELICITATIONS

User requirements

- Requesting Laptops:
 - Ability to request laptop loans remotely through an online platform.
- Booking and Return:
 - Access to view and manage current laptop bookings.
 - Notifications/reminders for laptop return upon approaching the 2-hour limit.
- User Authentication:
 - Secure login system to protect student data.
- Inventory Management:
 - A digital catalog of available laptops with current status (booked, available, overdue).
- Notification System:
 - Alerts for late returns, system issues, or inventory inconsistencies.

System requirements

- Implementation of secure login functionalities for administrators and students to access their respective portal features.

- Development of a booking module enabling students to reserve laptops remotely thereby ensuring availability status is accurately reflected in real-time on the platform.
- Management by administrators to monitor the status and health of the laptop inventory by allowing them to add, edit, and remove laptop details and status.
- Implement automated notifications and alerts for students and administrators related to booking timings and status (for a 2-hour duration of the loan).
- Logging of all actions (booking, return, alerts) and associating them with respective user data.
- Ensure data storage, transmission, and processing adhere to security best practices.
- Leverage Power Platform capabilities for workflow automation, app development, and data analytics.
- Ensure system compatibility across various devices and browsers for accessibility.

Functional requirements

- Development of an online system that allows students to request laptops electronically.
- Creation of a digital inventory management system for administrators to track and manage laptop assets.
- Simplifying the process of borrowing and returning laptops for students by reducing waiting time.
- Providing relevant notifications to keep users informed about the booking and when the system is due.

Non-functional requirements

- To implement capabilities that ensure administrators can actively view which students/staff have borrowed laptops.
- Improve data accuracy and reliability in laptop tracking and management.
- Facilitating responsible and efficient utilization of laptops by students.
 - Enabling administrators to identify underutilized or unused assets for potential optimization.
- Enhance accountability by tracking the history of laptop assignments and returns.
 - Provide a system-generated audit trail for better monitoring.
- Provide training and support to users (both students and administrators) for effective system utilization.
 - Ensure that all stakeholders can make the most of the new tracking system.
- Implement robust security measures to protect sensitive data, including student information and laptop details. (Includes complying with data privacy regulations to safeguard user data.)

Use Case Diagram

The actors involved in our system is the user (registered DUT student) and the admin (Library Staff/ Tech division of DUT). The user has access to login, requesting for booking a laptop, filing a return of a laptop (this option is only available if there is an outstanding device). The admin is has the ability to add and remove laptops from the inventory as well as view laptops that have been borrowed by students.

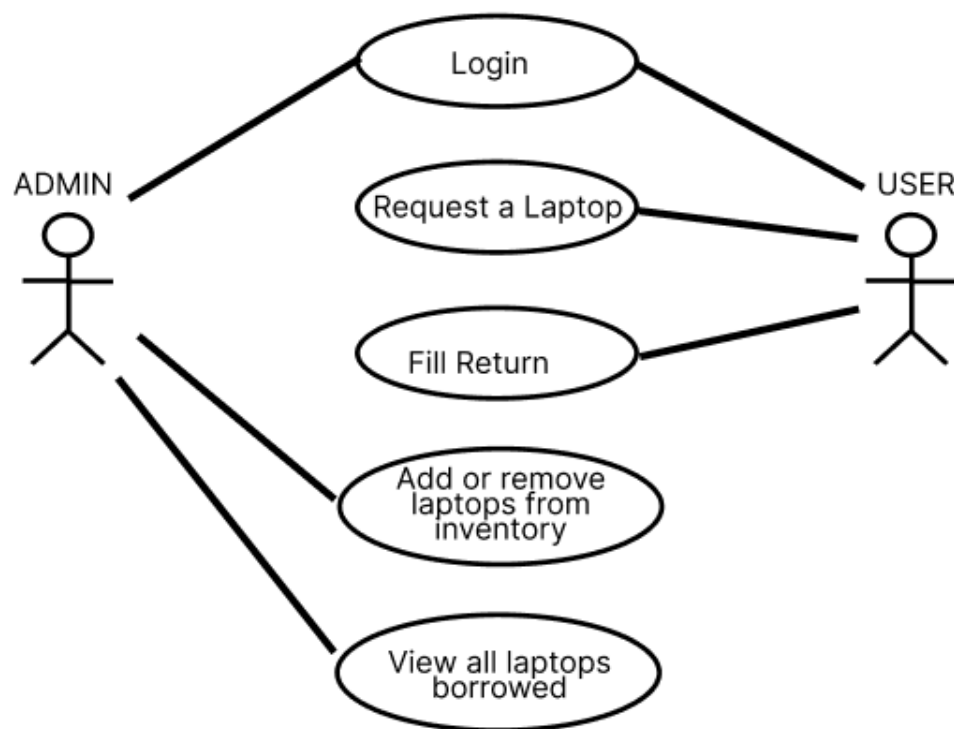


Figure 1.2: Use case diagram of the proposed system

PROJECT MANAGEMENT

Project technology

The HorizonCraft team has carefully selected Microsoft Power Platforms as its main technology due to its versality and the integrational advantage it had over other technologies. The proposed system will utilize mainly Power Apps and Power BI for the management tools needed to carefully track the laptops being loaned to students with the library setting.

Power Apps will assist in the development of the overall application in conjunction with the cloud storage needed whilst Power BI abets the automation of the confirmation email.

Risks

Risks can be described as unforeseen circumstances or events that affect the project schedule, or the quality of the system being produced. Figure 1.3 tabulates the different risk types and the potential risks the project may encounter whilst being developed.

Risk Type	Possible Risks
Technology	<ul style="list-style-type: none"> Errors or data loss during the migration of existing manual records to the new system could lead to inaccuracies and operational disruptions.
People	<ul style="list-style-type: none"> User's resistance to change and inadequate training may result in low user adoption rates and dissatisfaction with the system. Ignoring user feedback or failing to iterate on the system based on user needs and experiences may result in a less effective solution.
Organizational	<ul style="list-style-type: none"> Data breaches or privacy violations could occur if security measures are not adequately implemented, leading to legal and reputational consequences. Failure to comply with relevant regulations, such as data protection laws, could result in penalties and reputation damage.
Tools	<ul style="list-style-type: none"> Difficulty integrating the new system with existing university systems and databases may disrupt operations.
Requirements	<ul style="list-style-type: none"> Expanding the project scope beyond the original objectives could lead to increased costs, delays, and resource constraints.
Estimation	<ul style="list-style-type: none"> Insufficient budget, staffing, or technical expertise may hinder the project's progress.

Figure 1.3: Tabulation of the risks expected whilst developing the system

Overall, HorizonCraft's team will try their utmost to achieve the objectives needed to make this solution a success. In addition, the team of developers and system analysts have been actively contributing to developing contingency plans, should the need arise.

Schedule/timeline

➤ **Phase 1: Project Initiation**

Define project objectives and scope.

Assemble the project team and allocate responsibilities.

Conduct a stakeholder analysis.

Develop a project charter and obtain approval.

➤ **Phase 2: Planning and Requirements Gathering**

Conduct a detailed requirements analysis with stakeholders.

Create a project plan, including a timeline, budget, and resource allocation.

Identify risks and develop a risk management plan.

Select the technology stack (e.g., Microsoft Power Platform).

Design the system architecture and develop a database schema.

Create wireframes and prototypes of the user interface.

- **Phase 3: Development and Testing**
Development of the Laptop Tracking System.
Conduct unit testing and system integration testing.
Perform user acceptance testing (UAT)
- **Phase 5: Deployment**
Deploy the Laptop Tracking System in a staging environment.
Train administrators and support staff.
Conduct a soft launch with a limited user group for initial feedback.
- **Phase 6: Post-Deployment and Evaluation (scheduled for after deployment)**
Monitor system performance and user feedback.
Address any issues or bugs that arise.
Conduct a comprehensive evaluation of the system's functionality.
Make necessary adjustments and improvements based on user feedback.

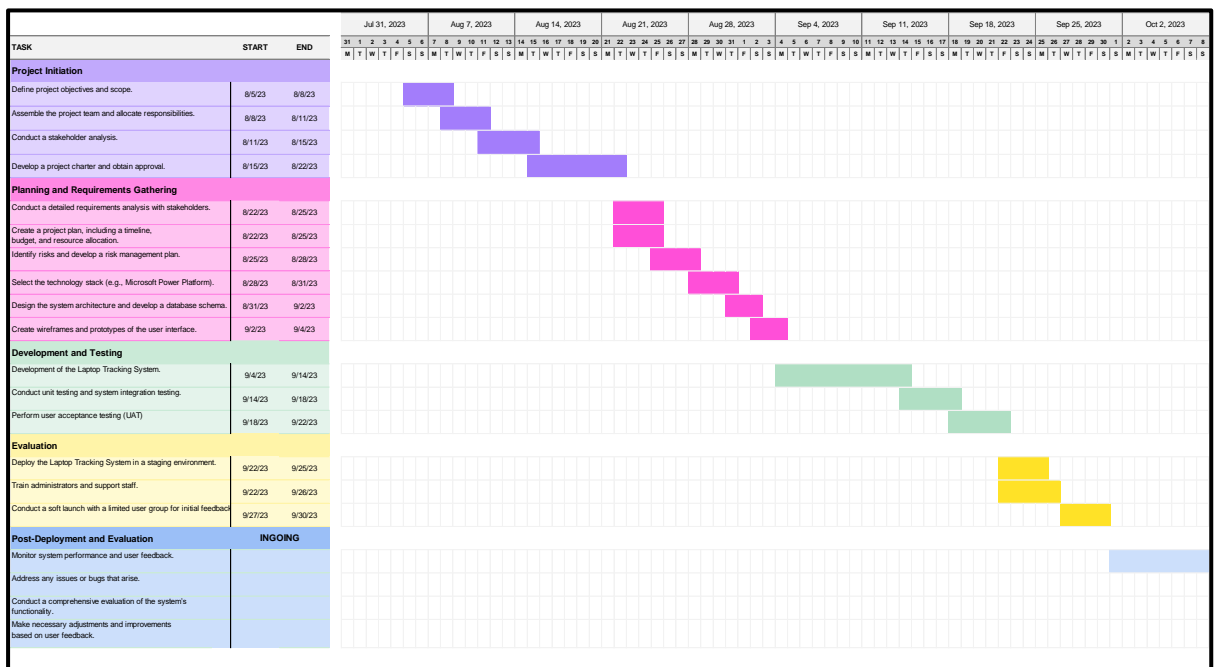


Figure 1.4: Gantt Chart displaying the schedule and estimated timeline.

Costs

Our costing incorporates hardware, software, staff, maintenance and risks costs. Figure 1.5 shows an overall breakdown of what each amount is entitled for.

HorizonCraft Costing		
Description	Price	
Hardware and Infrastructure	R	25,000.00
Software and Licensing	R	15,000.00
Team/Personnel Costs	R	130,000.00
Overtime (including travel and allowance)	R	80,000.00
Maintenance	R	200,000.00
Risks	R	350,000.00
GRAND TOTAL	R	800,000.00

Figure 1.5: Costing Table with amount for each sector.

Overall, the project will cost an amount of R 800 000.00, and will cater for various sectors that will be utilized during development and completion of the project.

TOOLS AND ENVIRONMENT

Web/Mobile platform

The system platform for our project will be the Microsoft Power Platform. Microsoft Power Platform is a suite of business application development tools and services provided by Microsoft.

It is designed to enable organizations to create custom solutions for various business needs, including data analysis, process automation, and application development. The power platform consists of several key components and two of these components will be leveraged on to build the Laptop Tracking System:

- **Power Apps:** Power Apps is a low-code/no-code application development platform. It allows you to create custom apps with a user-friendly interface. In the context of the Laptop Tracking System, power apps can be used to develop the user interface for students and an administrator to request laptops and manage inventory digitally.
- **Power Automate:** Power Automate is an automation and workflow tool. It can be used to automate processes related to laptop requests, notifications, and approval workflows. For example, you can create workflows to route laptop requests to the appropriate administrators for approval.

Programming languages

- **Power Apps (Canvas Apps):** Power apps use their own formula language for creating logic and formulas within the application.
- **Power Automate:** Power Automate Expressions allows you to use expressions to manipulate data and create conditional logic in your workflows.

Database

Microsoft power platform is often integrated with the common data service, which is a cloud-based data storage and modeling service provided by Microsoft. The CDS can serve as the primary database for the laptop tracking system.

It offers data security, scalability, and seamless integration with power apps and power automate. Regular backup and recovery are crucial in ensuring data is not compromised in any manner.

The implementation of monitoring tools and practices to track database performance and optimize queries for efficient data retrieval is also of importance especially with regards to assisting developers in advising towards the customer's needs.

Operating environment

In our journey to develop the laptop tracking system for Durban University of Technology, creating a total operating environment is paramount as accessible people like to share our approach to establishing this environment to ensure the smooth operation of the system.

We have therefore chosen Microsoft Power Pages as a cloud platform, and Windows Server will be writing a system of transfers hosting our system. Azure provides the scalability and reliability we need, while Windows Server offers a familiar environment for our team.

IDEs

- **Power Apps Studio:**
Power App Studio is an integral part of our development process for building canvas apps within Microsoft Power Apps. It provides an intuitive visual interface for creating apps screens and logic. Since Power Apps is central to our project, this tool is crucial for app development.
- **Visual Studio Code (VS Code):**
VS Code serves as our primary code editor. HorizonCraft leverages it for JavaScript, Python, and other scripting tasks as well as its integrated Git functionality. The extensive library of extensions allows us to customize our development environment to suit our needs.
- **Power Automate (Flow):**
Power Automate comes with a built-in expression editor that will be used for creating custom expressions and logic. It simplifies the automation of processes and workflows within our Laptop Tracking System by providing automated emails.

SOFTWARE DESIGN

Flow diagram

This graphical image in figure represents a basic flow of events that the user will undertake based on their desired goal.

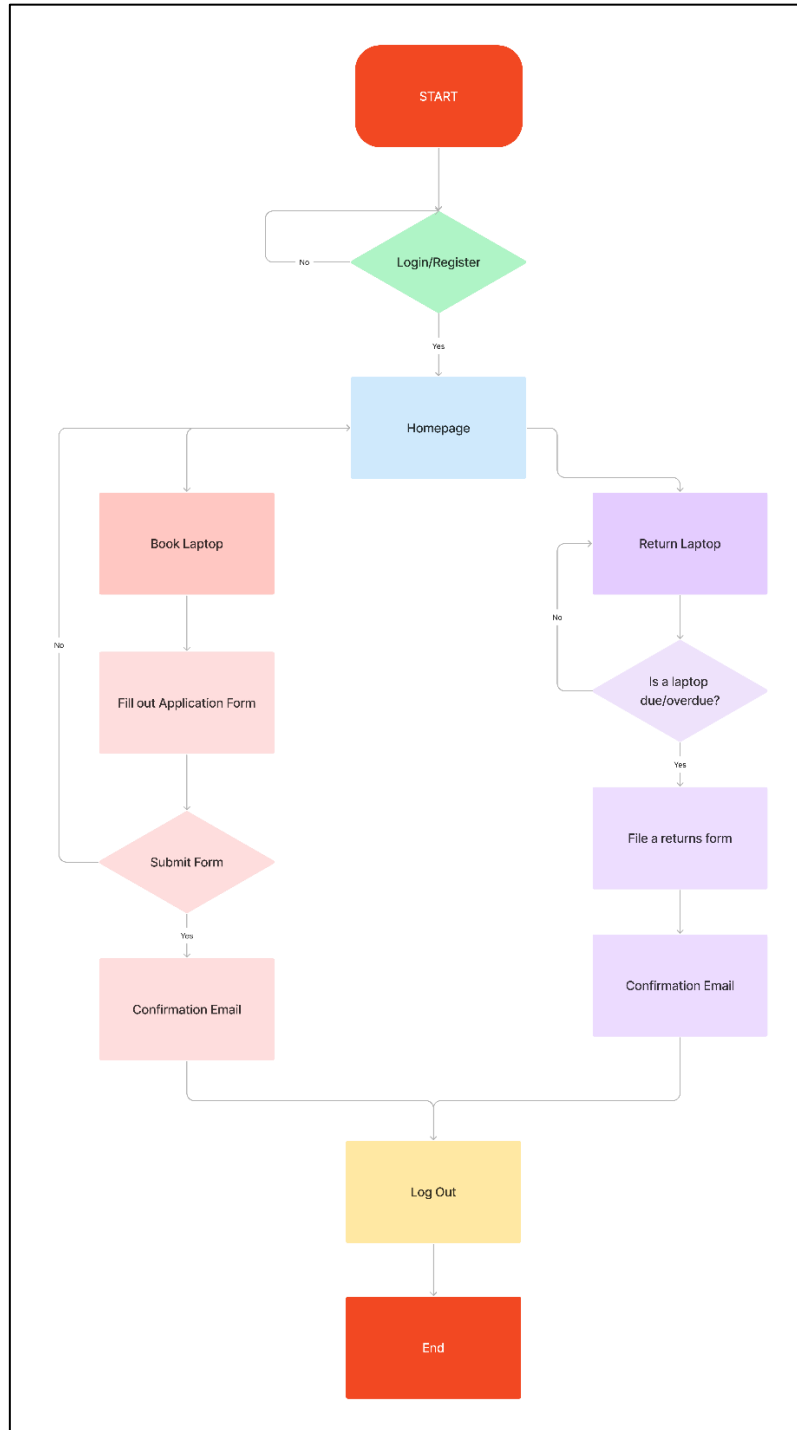


Figure 1.6: Flow Diagram of the proposed system

Overall, the user will be required to sign-in in order to be able to book or return laptops thereby ensuring assets are safeguarded and data is not compromised.

ERD

Figure 1.7 depicts the Entity-Relationship Diagram (ERD) that the system's database will be utilizing. It shows the relationships between the tables and the corresponding attributes/features.

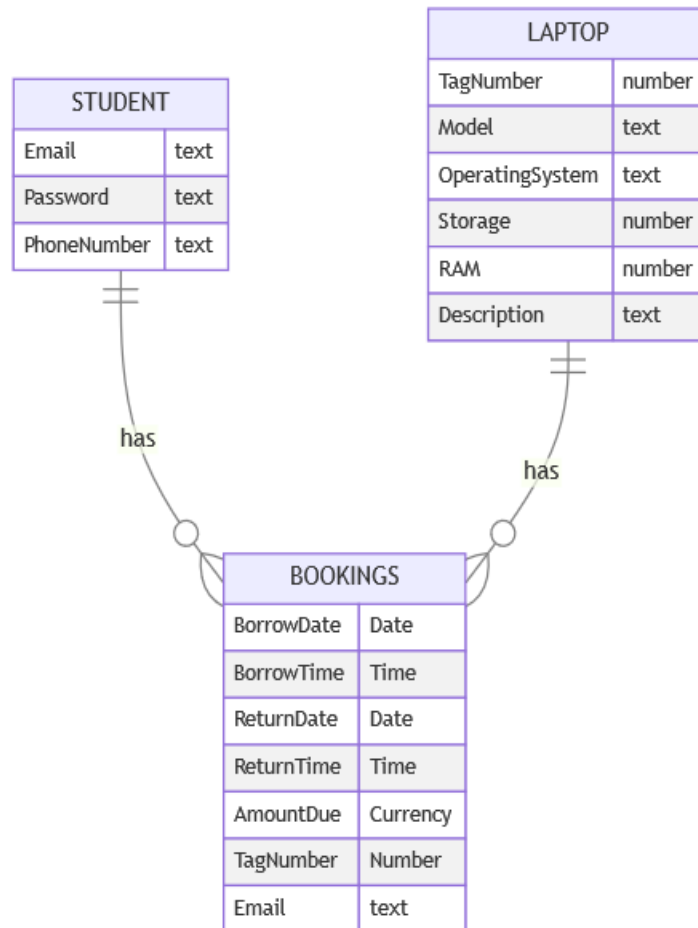


Figure 1.7: ERD Diagram of the cloud database

In summary, the ERD showcases that a student can book multiple laptops (one-to-many relationship between Student and Bookings), and a laptop can be booked multiple times (one-to-many relationship between Laptop and Bookings).

The Bookings entity holds foreign keys from both the Student and Laptop table which establishes these relationships and ensures referential integrity within the database.

Sequence diagram

Figure 1.8 showcases how the user will interact with the system when they want to loan a laptop from the library. The user will interact with the platform whilst in the background, Power Apps and Power BI will be utilized to complete automated tasks and handling of the data.

Figure 1.9 depicts how the action of filing for a return will only be completed if there is an outstanding laptop that the student owes to the library. If the student does not have any laptops overdue – then they will not be sent to the filing return page.

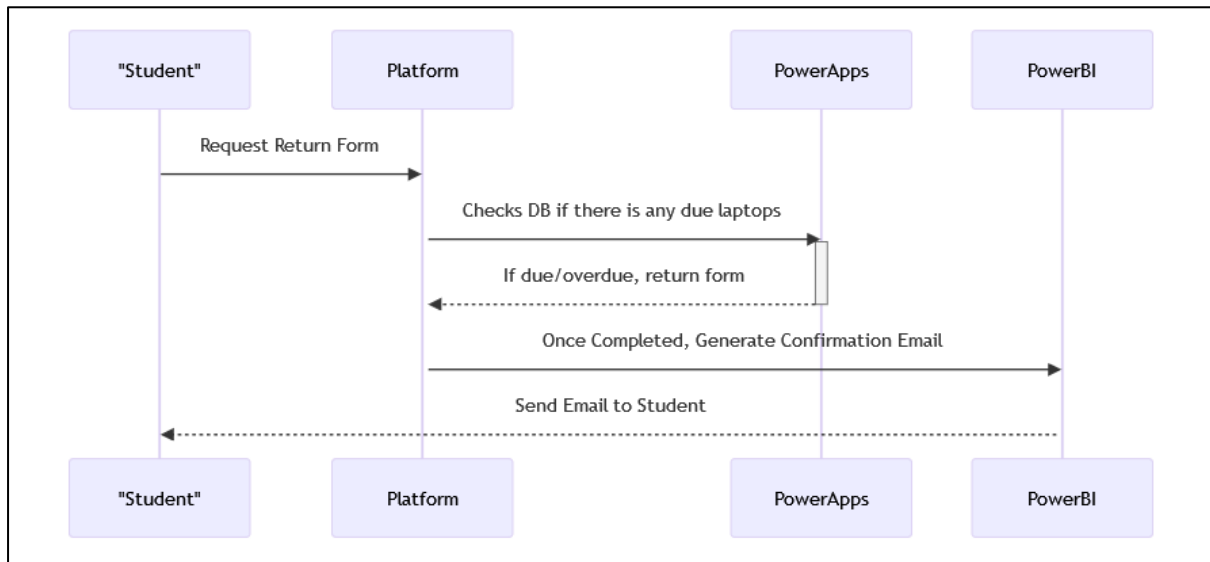


Figure 1.8: Sequence diagram for loaning a laptop

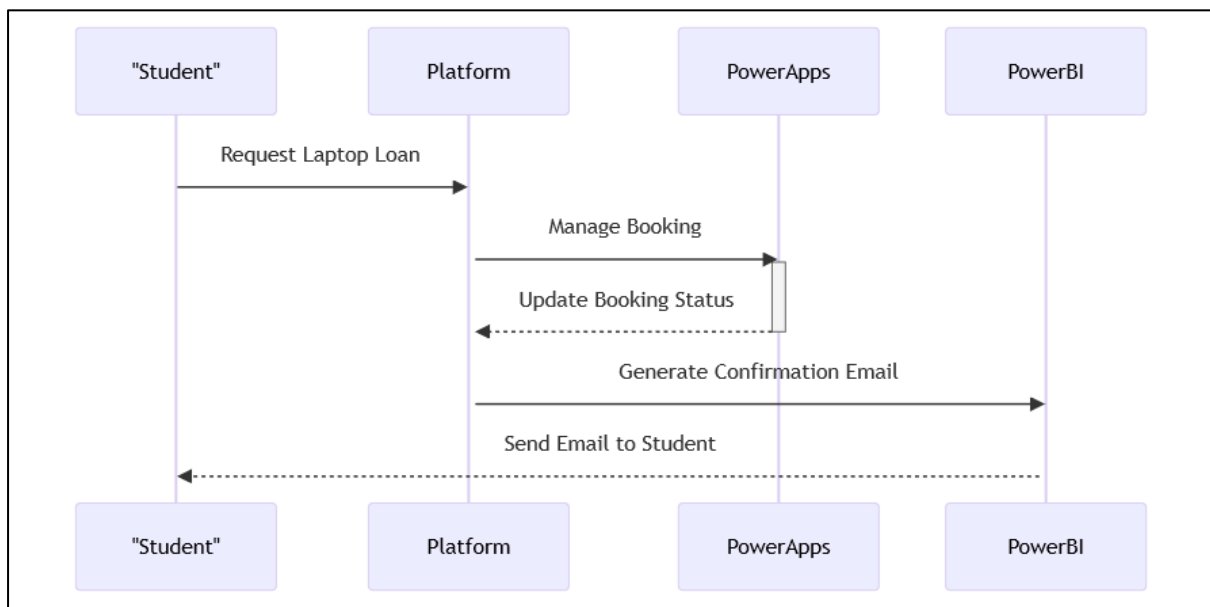


Figure 1.9: Sequence diagram for filing a return of laptop

Wireframe

The figures 1.10 and 1.11 showcase the wireframes that were taken into account to create an intuitive and user-friendly application that both students and admins can utilize and manage their time more efficiently.

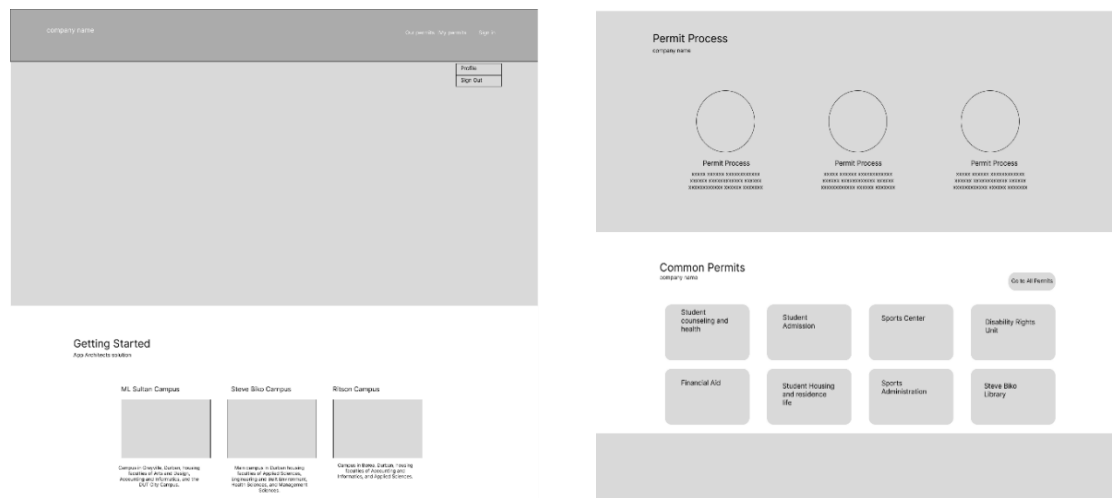


Figure 1.10: Wireframe which consists of how the dashboard will be structured.



Figure 1.11: Wireframe of the application process and the success message

SOFTWARE CONSTRUCTION/ DEVELOPMENT

Back End

The backend comprises of several cloud functions, this JavaScript code snippet is designed to enhance the user interface by dynamically updating the display of laptop application statuses within a table grid. Utilizing jQuery, the code is triggered once the entire webpage is fully loaded. It specifically listens for a "loaded" event on the table grid, identified by the classes "entitylist" and "entity-grid".

Once this grid is loaded, the code iterates through each table cell tagged with a data-attribute of 'bp_laptop_status'. For each such cell, the code retrieves its associated status value and sets

its display style to "flex" for better visual presentation. A switch statement is then employed to map these numerical status values to more user-friendly, readable statuses such as "Submitted", "Under Review", "Denied", or "Approved".

These readable statuses are inserted into the table cells, each wrapped in a span element styled with a specific CSS class that corresponds to the status. This ensures that both students and administrators can easily comprehend the current status of laptop applications, thereby enhancing the system's usability.



```
1 $(document).ready(function () {
2   $(".entitylist.entity-grid").on("loaded", function () {
3     $(this).children(".view-grid").find("td[data-attribute='bp_laptop_status']").each(function (i, e) {
4       var value = $(this).data(value);
5       $(this).css({
6         "display": "flex"
7       });
8       // When the user has submitted a laptop application, these are the statuses that will be displayed
9       switch(value.value.Value) {
10        case 121260000:
11          $(this).html("<span class='table-status-cell submitted'>Submitted</span>");
12          break;
13        case 121260003:
14          $(this).html("<span class='table-status-cell review'>Under Review</span>");
15          break;
16        case 121260002:
17          $(this).html("<span class='table-status-cell denied'>Denied</span>");
18          break;
19        case 121260001:
20          $(this).html("<span class='table-status-cell approved'>Approved</span>");
21          break;
22        default:
23          $(this).css({
24            "color": "#000000"
25          });
26      }
27    });
28  });
29 });
```

Figure 1.12: Code snippet of the Microsoft Azure cloud functions in JavaScript

In the Laptop Management System at Durban University of Technology, the given JavaScript code snippet is focused on enhancing the user interface of a login page. Utilizing jQuery, the code is executed once the webpage is fully loaded. It targets various HTML elements on the page and applies specific CSS styles to them to improve their appearance and usability.

For instance, the code modifies the active navigation tabs by setting their border, font size, and color. It also hides any Font Awesome icons within a container. The text and styles of specific list items in the content area are also updated, including changing one to "Sign Up" and styling it differently.

The code pays special attention to the login form. It modifies the layout of the username and password fields, setting their borders and placeholders. It also changes the appearance of the "Remember Me" checkbox and the "Sign In" button, including their font sizes and colors. Additionally, it adjusts the layout of the form's sections to make them more visually appealing and user-friendly.

The use of CSS variables like `var(--portalThemeColor1)` allows for easy theming, making it simpler to adapt the UI to different visual themes. Overall, this code snippet is instrumental in ensuring that the login page is not only functional but also visually appealing and user-friendly.

```

content-snippets > account-signin-pagecopy > Account-SignIn-PageCopy.contentsnippet.value.html > script
1 <script>
2 $(document).ready(function() {
3     $(".nav-tabs>li.active>a, .nav-tabs>li.active>a: hover, .nav-tabs>li.active>a: focus").css({
4         "border": "2px solid var(--portalThemeColor1)",
5         "border-radius": "20px",
6         "font-size": "24px",
7         "font-weight": "300",
8         "color": "var(--portalThemeColor1)"
9     });
10
11     $(".container .fa").css({
12         "display": "none"
13     });
14
15     $("#content > ul > li:nth-child(2) > a").css({
16         "color": "#8F8F8F",
17         "font-size": "24px"
18     });
19
20     $("#content > ul > li:nth-child(2) > a").text("Sign Up");
21
22     $("#content > ul > li:nth-child(3) > a").css({
23         "color": "var(--portalThemeColor1)",
24         "font-size": "24px",
25         "text-decoration": "underline"
26     });
27
28     $("#content > ul").css({

```

```

content-snippets > account-signin-pagecopy > Account-SignIn-PageCopy.contentsnippet.value.html > script
28     $("#content > ul").css({
29         "margin-bottom": "50px"
30     })
31
32
33     $(".nav-tabs").css({
34         "border-bottom": "none"
35     });
36
37     $("#local-login-heading > span > span").text("welcome, please log in to continue.");
38     $("#external-login-heading > span > span").text("External Sign In");
39
40     $(".login-heading-section").css({
41         "border-bottom": "none"
42     });
43
44     $("#mainContent > div.row > div:nth-child(1) > form > div > div:nth-child(3)").css({
45         "display": "flex",
46         "flex-direction": "column"
47     });
48
49     $("#mainContent > div.row > div:nth-child(1) > form > div > div:nth-child(3) > label").css({
50         "margin-left": "-83px",
51         "margin-bottom": "10px",
52         "color": "var(--portalThemeColor1)"
53     })

```

```

content-snippets > account-signin-pagecopy > Account-SignIn-PageCopy.contentsnippet.value.html > script > ready() callback
53 })
54
55     $("#mainContent > div.row > div:nth-child(1) > form > div > div:nth-child(3) > div").css({
56         "width": "100%"
57     });
58
59     $("#Username").css({
60         "border": "0",
61         "border-bottom": "2px solid #181818",
62         "-webkit-backface-visibility": "hidden"
63     })
64
65     $("#Username").prop("placeholder", "Enter your username")
66
67     $("#Username[placeholder]").css({
68         "color": "#8F8F8F",
69         "font-size": "22px"
70     })
71
72     $("#mainContent > div.row > div:nth-child(1) > form > div > div:nth-child(4)").css({
73         "display": "flex",
74         "flex-direction": "column"
75     });
76
77     $("#mainContent > div.row > div:nth-child(1) > form > div > div:nth-child(4) > label").css({
78         "margin-left": "-83px",
79         "margin-bottom": "10px",
80         "color": "var(--portalThemeColor1)"
81     })
82
83     $("#mainContent > div.row > div:nth-child(1) > form > div > div:nth-child(4) > div").css({
84         "width": "100%"
85     });
86
87     $("#PasswordValue").css({
88         "border": "0",
89         "border-bottom": "2px solid #000",
90         "-webkit-backface-visibility": "hidden"
91     });

```


Figure 1.13: Code snippet of the Microsoft Azure cloud functions in JavaScript

Database

A cloud-based data storage and modeling service is provided by Microsoft and this will be used in storing the data. Below the tables highlight how each table will be established in the CDS:

Column Names	Data Type	Description Of Column
Email <PK>	Text (format: Email)	Consists of the student email. The student is required to use their dut4life email accounts
Password	Text	User's desired password
Phone Number	Text (format: Phone number)	(Optional) The phone number of the user.

Figure 1.14: Student Table in the CDS

Column Names	Data Type	Description Of Column
TagNumber <PK>	Number	The tag or asset number of the laptop. Allows for easier tracking.
Model	Text	Provides the model/make of the laptop.
OperatingSystem	Text	Details the type of operating system loaded on the laptop.
RAM	Number	Amount of RAM loaded onto the system
Storage	Number	Amount of storage space available on the laptop.
Description	Rich Text	Other valuable information on the laptop. Details what is expected of the laptop. Helps the user get an overall description of the device.

Figure 1.15: Laptop Table in the CDS

Column Names	Data Type	Description Of Column
TagNumber <FK>	Number	The tag or asset number of the laptop. Allows for easier tracking.
Email <FK>	Text (format: Email)	Consists of the student email. The student is required to use their dut4life email accounts

Borrow_date	Date Time (format: Date only)	Captures which proposed date the user wishes to borrow the laptop.
Borrow_time	Date Time	Captures which proposed time the user wishes to borrow the laptop.
Return_date	Date Time (format: Date only)	Captures which proposed date the user wishes to return the laptop.
Return_time	Date Time	Captures which proposed time the user wishes to return the laptop.
Amount_due	Currency	Amount due by the student. This is applied once the deadline has passed and students have not returned the laptop.

Figure 1.16: Association table with Student and Laptop

Figure 1.14 shows the entities that exist in the student table and these include email, password and phone number. Figure 1.15 highlights the entities that exist in the laptop table which are, tag/asset number, model of laptop, operating system installed on the machine, followed by RAM and storage available.

The table also makes use of a description entity that provides an overall account of the features that the laptop has. Figure 1.16 is the association table that will contain each email of the student and the corresponding tag number of the laptop borrowed. This table will also contain the student's estimated borrow and return date and time to allow for effective tracking.

GUI

Figure 1.17 is what is displayed to the student once they have been logged into the system. The user has the option to view the application process of how to go about loaning a laptop from the library using the system.

Figure 1.18 depicts what occurs once the user selects a laptop that they wish to borrow. Once the user goes through filling in of the information required for the application – a success message is displayed.

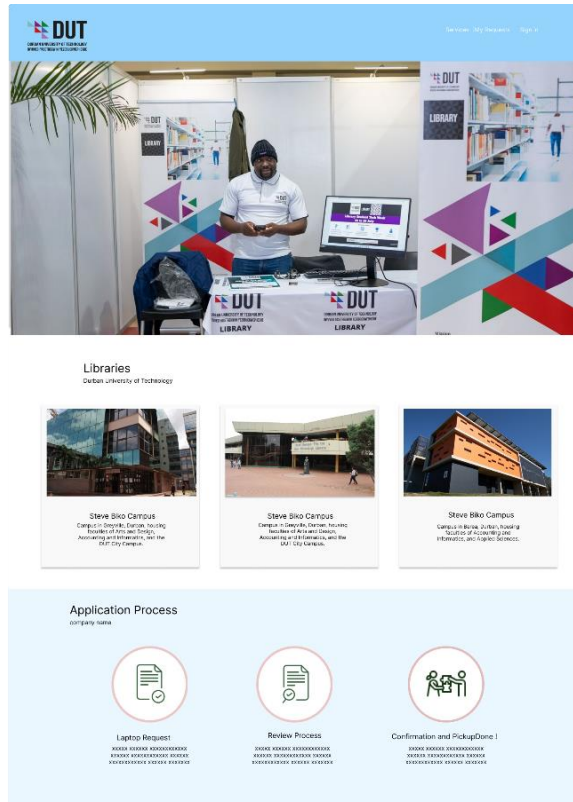


Figure 1.17: Dashboard of the system.

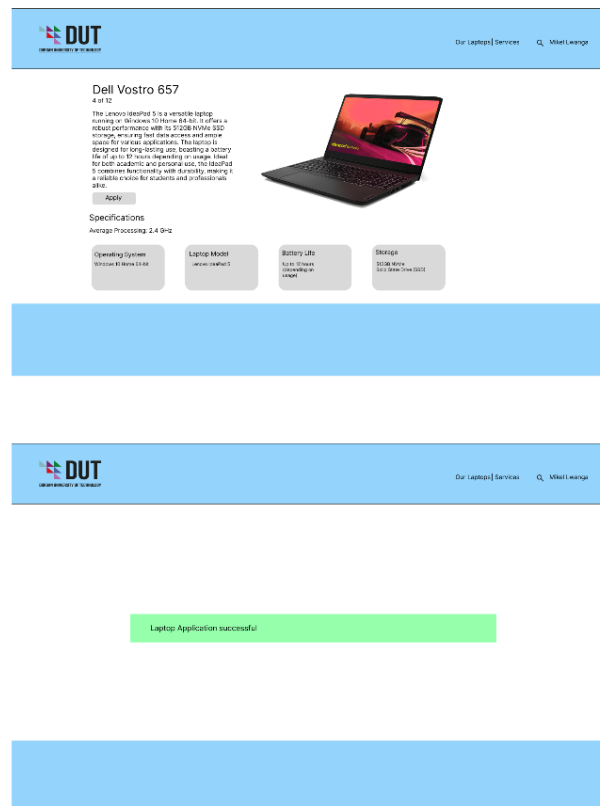


Figure 1.18: Laptop that is being selected from the inventory followed by the success message when the laptop is booked.

SOFTWARE VERIFICATION AND VALIDATION/TESTING

This section outlines the various methods and procedures followed to verify and validate the software components of the Laptop Management System, specifically focusing on Power Apps and Power Automate functionalities.

Correct Input:

- **Objective:** To ensure that the system handles correct inputs as expected within the Power Apps interface.
- **Methodology:** Unit testing using Power Apps' built-in testing features.
- **Tools Used:** Power Apps Test Studio
- **Results:** All tests passed, confirming that the system correctly processes valid inputs.

Wrong Input:

- **Objective:** To verify that the system gracefully handles incorrect or invalid inputs, especially in Power Automate flows.
- **Methodology:** Error handling tests in Power Automate.
- **Tools Used:** Power Automate's built-in error handling features.
- **Results:** The system displayed appropriate error messages and did not crash, confirming robustness against invalid inputs.

Testing Types:

- **Unit Testing:** Conducted to test individual components within Power Apps.
 - **Tools:** Power Apps Test Studio
 - **Results:** All unit tests passed.
- **Integration Testing:** Performed to test the data flow between Power Apps and Power Automate.
 - **Tools:** Power Automate's built-in testing features.
 - **Results:** Data integrity maintained between modules.
- **System Testing:** Carried out to validate the entire system, including Power Apps and Power Automate functionalities.
 - **Tools:** Power Platform Center of Excellence (CoE) Toolkit
 - **Results:** System met all specified requirements.
- **User Acceptance Testing (UAT):** Conducted with real users to ensure the system meets user needs.
 - **Methodology:** Alpha and Beta testing within the Power Platform environment.
 - **Results:** Positive feedback received, minor adjustments made in Power Apps Studio.