Applied Information Technology And Computer Science Vol. 2 No. 2 (2021) 1158-1174

AITCS

Homepage: http://publisher.uthm.edu.my/periodicals/index.php/aitcs
e-ISSN :2773-5141

The Development of Techna: Web-Based Online Computer Repairing Service Platform

Fikri Arif Azman, Rozanawati Darman*

Faculty of Computer Science & Information Technology, Universiti Tun Hussein Onn Malaysia, Parit Raja, Batu Pahat, Johor, 86400, MALAYSIA

DOI: https://doi.org/10.30880/aitcs.2021.02.02.072
Received 02 August 2021; Accepted 12 November 2021; Available online 30 November 2021

Abstract: Web-based online computer repairing service platform is a website for any technicians to make their business computer repairing online. The problem faced by traditional computer repairing is that it requires a physical appointment to make the repairs. Hence, the distance and time is needed to be considered here since the customer needs to approach the technician's place first for the approval from the technician's skill set regarding their computer issues. The objective of this project is to develop a website that acts as a platform for technicians and customers to do their online appointments instantly based on their nearby location. There are two users who are using the client website and that is the technicians and customers who seek computer repairing service. This website was built using HTML, CSS, JavaScript and PHP language. The method that this project uses is called prototype methodology. We inject a Geolocation API to the website to get precise location of the user and the technicians, helping the coding to get precise results. The outcome of this project hopefully helps users to seek computer repairs using an online platform.

Keywords: Online platform, Computer Repair Service, nearby location, Web-Based System, Geolocation API

1. Introduction

Computer repairing is a type of service that offers fixation like install, repair, maintenance and clean-up of any computer equipment. The standard procedure for computer repairing is people who seek computer service, find the nearest computer repair shops, and make an appointment about their damaged laptop to the technician. If the technician approves and can repair the damaged laptop, the customer will wait for a duration of time until the technician has done their work [1].

In hindsight, the requirements that determine the efficiency of the traditional repair procedure is distance, skills and time. A satisfied computer repairing procedure is having a short distance, large coverage of skill sets and punctuality. Hence, an online computer repairing service exists.

An online computer repairing service defines the same as before but with a little twist. The twist here is customers will make an appointment online by searching the web on the nearest online computer repairing service. With that having been said, online computer repairing service tackles the distance

*Corresponding author: zana@uthm.edu.my

requirement, but not the skills. An online platform is required to organize this. Hence, the foundation of Techna was born.

An online computer repairing service defines the same as before but with a little twist. The twist here is customers will make an appointment online by searching the web on the nearest online computer repairing service. With that having been said, online computer repairing service tackles the distance requirement, but not the skills. An online platform is required to organize this. Hence, the foundation of Techna was born.

Techna is an online platform for computer repairing, where users can request repair from nearby technicians. The general idea of the functions of Techna is that users can register to the platform and make an appointment with the technicians that are available at their nearby locations.

The aim of this project is to develop an online platform for computer repairing service for public citizens to use in Malaysia. Objectives has been made to realize the project aim: -

1. To design a virtual marketplace for customers who seek computer repair service and technicians who seek available customers.
2. To develop a site that centralized the computer repair services social and business network that is available online.
3. To test the functionality of conducting an online platform that offers computer repair service.

Section 2 will get in more on how online platforms work, the e-commerce transaction system strengths and flaws to abide by, and the geolocation APIs on how it works and the implementation of all of these web services. A brief explanation and comparison of 3 existing systems which are Techfix, Mobiklinik and Syno's online repairing service. Section 3 discusses the approach in building this web-based platform and how the full procedure works and the analysis and design. Section 4 will explain in detail on the results and discussions as an insight outcome for this project. Finally, Section 5 summarizes the Techna's platform and future visions in developing this platform.

2. Related Works

Highlights of the study are done to give ideas and direction of system development. Next, this section also discusses all the terms and definitions related to this system. This section also discusses and analyzes the comparison of existing systems with systems developed.

2.1 Introduction to Online Platform

Based on the Organisation for Economic Co-operation and Development [2], online platform is a term describing various services that can be done online. This includes marketplaces, search engines, social media, app stores, outlets, payment systems, just to name a few. Usually, online platforms involve two or more separate users that interact with each other. To clarify more to that, the platform provider is the party that provides such a platform like a website, app, software or a broadcast network. Then people who use the platform are considered as users. Examples of users are buyer and seller, uploader and downloader, client and customers. As mentioned before, Techna is the platform provider while the users are technicians and customers.

Additionally, the term "users" can be interpreted in a reasonably broad manner. It can be small to medium businesses, governments, employees, not just an individual consumer [2].

There are a lot of benefits when approaching an online platform. With online platforms, everything can be a lot faster. From buying products with a couple of mouse clicks, fast delivery and fast in terms of accessibility. Speaking of accessibility, an online platform makes every service available everywhere, with just an internet requirement. Finally, online platforms can change the industry, or be

revolutionary. Take this for example, Uber, a popular ride sharing service has undoubtedly changed the taxi game, by lowering the price, and expanding their service like UberEats, UberX and UberBOAT [3].

And to that, computer repairing service is able to enter the online platform market, making customers can access to the platform whenever needed. But before creating the platform, the typology of online platforms needs to be explained.

2.2 Geolocation API

The Geolocation API characterizes a significant level interface to area data related distinctly with the gadget facilitating the usage, for example, latitude and longitude. The API itself is a freethinker of the basic area data sources. Regular wellsprings of area data incorporate Global Positioning System (GPS) and area induced from network signals, for example, IP address, RFID, WiFi and Bluetooth MAC locations, and GSM/CDMA cell IDs, just as client input. No assurance is given that the API restores the gadget's genuine area [4].

To simplify the statement above, the Geolocation API returns a location and accuracy radius based on information about cell towers and Wi-Fi nodes that the mobile client can detect. This document describes the protocol used to send this data to the server and to return a response to the client.

2.3 System Comparison

The existing online repairing sites are rare to find, with only few of them existing such as Techfix, Mobiklinik, and Syno. They have some similar parts with each other, which is to make an online appointment for users who seek computer repairs. But, there are some to compare with Techna and can be seen in Table 1. Table 1 displays the comparison of the existing related system and Techna.

Table 1: Comparison Existing System and Techna

Features\System	Techfix	Mobiklinik	Syno	Techna
1. Offers various device repairs	Yes	No	Yes	Yes
2. Real-time appointment	No	No	No	Yes
3. 24/7 availability	No	No	No	Yes
4. Geolocation Implemented	No	Yes	No	Yes
5. Available in whole Malaysia	No	No	No	Yes
6. Allow member/account registration	Yes	Yes	No	Yes

3. Methodology

The prototype model is used in developing this system. Based on Figure 1, the critical part is where the iteration begins, since the early stages and later are simplified. The fundamentals of the development here are to always find a room for improvement of the system.

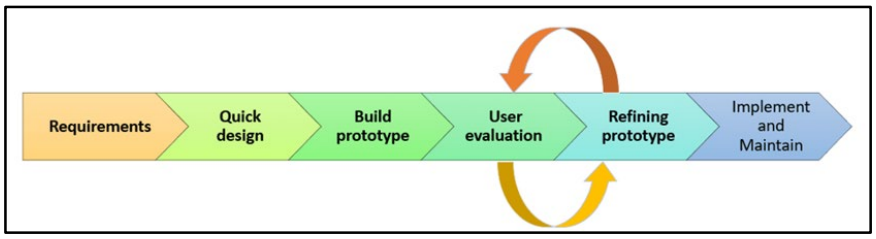


Figure 1: Prototype Model [5]

Extreme prototyping is used mostly for web-based applications [6]. It is a type of prototype that is straightforward to the needs of the system. In Techna, services and modules of the system are critical. So, this type of prototype is considered to be the best choice among others based on execution-wise.

The iteration part for the extreme prototype is as follows: -

1. Create a basic HTML format consisting of all existing pages.
2. Simulate the data flow using a prototype service layer.
3. Service implementation and integration.

In general, the phases will be broken down into 6 parts, the requirement gathering phase, quick design phase, building the prototype (the phase that will be the iteration part), user/client feedback, refining the prototype and implementing it to the final product.

3.1 Analysis and design

In this section, a lot of diagrams will be explained such as the Entity Relationship Diagrams (ERD) of this system, the Data Flow Diagrams (DFD), Flow Charts and the user interface during the analysis and design stage of Techna.

There are two phases of work done in this section, the first part is analysis of the requirements and flows, the second part is designing the system.

3.2 Requirement Analysis

In this phase, only the basic requirements are gathered and will be expanded during the development, defining this model. The basic requirements are functional and non-functional requirements. The table 2 (Refer Appendix) shows the functional requirements of the system, which already briefly explained in section 1.

Non-functional requirements will be classified into 4 basic aspects which are security, accessibility, availability and confidentiality. The list below defines the aspects gathered: -

1. Security – the system must be safeguarded against errors and cyber criminals.
2. Accessibility – the system can be universally used by anyone.
3. Availability – the system needs to ensure their functionality during operational hours.
4. Confidentiality – the privacy and concern in details about sensitive data.

Table 3 (Refer Appendix A) are the examples for the basic non-functional requirements gathered during the planning of making the system.

3.3 Flow chart

Chart streams have been used to analyze, design, document or manage processes in various fields. The five main elements of a flow chart are start or end terminals, flow lines, inputs and results, processes and options.

The implementation of flowchart in this system are shown in figure 13 and 14. Figure 13 shows the user's flowchart when using the request repair module, while figure 14 shows the flowchart for the technician seeking nearby clients.

3.4 Data Flow Diagrams

The context diagram shows the overall functionality of the system, in other words, the 'big picture' of the system. A context diagram must be drawn in order to clarify the boundaries of this system. It identifies the flows of information between the system and external entities [8]. Figure 2 shows Techna's context diagram. In general, three types of users are heavily involved here, where each has their own role in stabilizing this system.

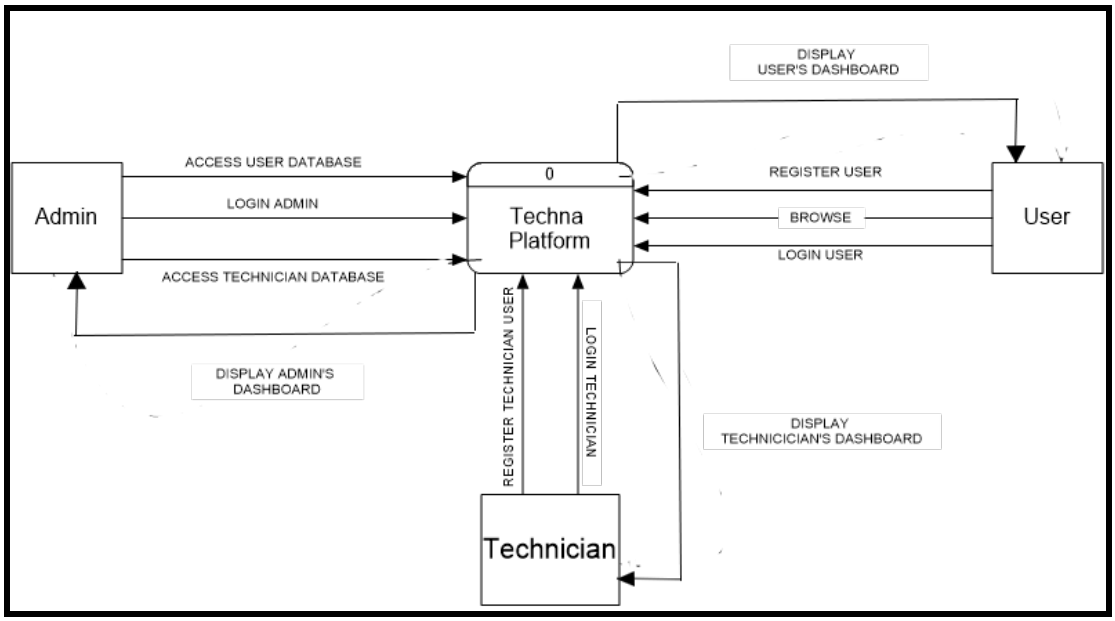


Figure 2: Techna's Context Diagram

The context diagram will then be expanded into Data Flow Diagram Level 0 and Data Flow Diagram Level 1.

This data flow diagram level 0 breaks down the general flow, function and modules from the context diagram. The results of the data flow diagram creates a new entity that will store the data inside this system.

Figure 3 shows the system's diagram. The process is now divided into 5 new modules, which are the registration module, login module, request repair module, appointment module and transaction module. The database or storage involved in this diagram has been classified into 6 tables which are the Admin database, the User database, the Technician database, the Manufacturers database, the Transaction Due/Request database and the Transaction Receipt Database.

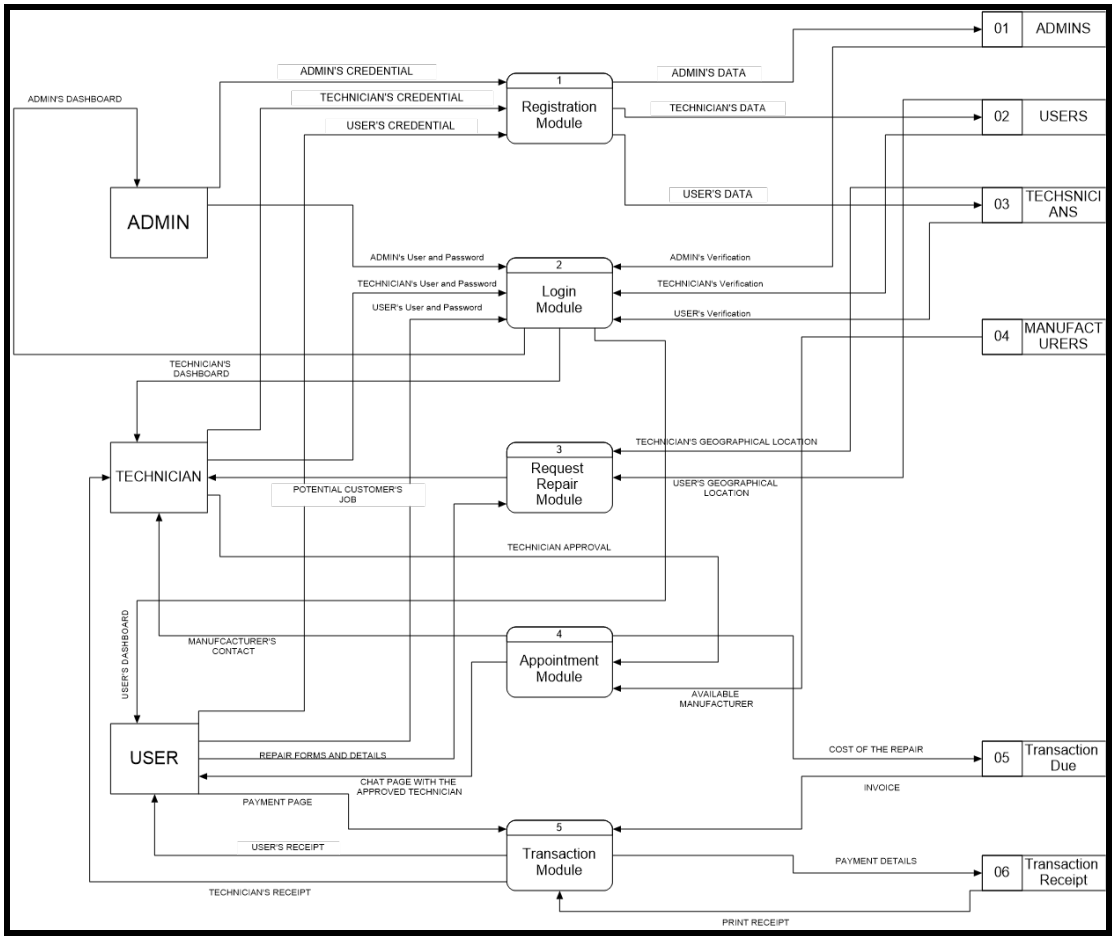


Figure 3: Techna's Data Flow Diagram Level 0

The diagram shows the movement of data for each user when corresponding to those modules. Based on the diagram, the modules are still wide and not specific. For example, the transaction module can be divided into more submodules such as wage payment module, manufacturer payment module and customer payment modules. Another example is the request repair module, which is not divided into submodules but rather the specific process of the modules such as request repair service, finding nearest available technicians and repair approval from technician. Hence, Data Flow Diagram Level 1 is created.

The results of this data flow diagram level 1 are created based on what is needed in order to achieve a more procedural process. The request repair module is broken down into sub processes such as request repair service by the user, finding the nearest available technician and repair approval by the technician.

The flows are realized at Figure 4, which shows the users involved, process flow, the movement of data from entities to processes.

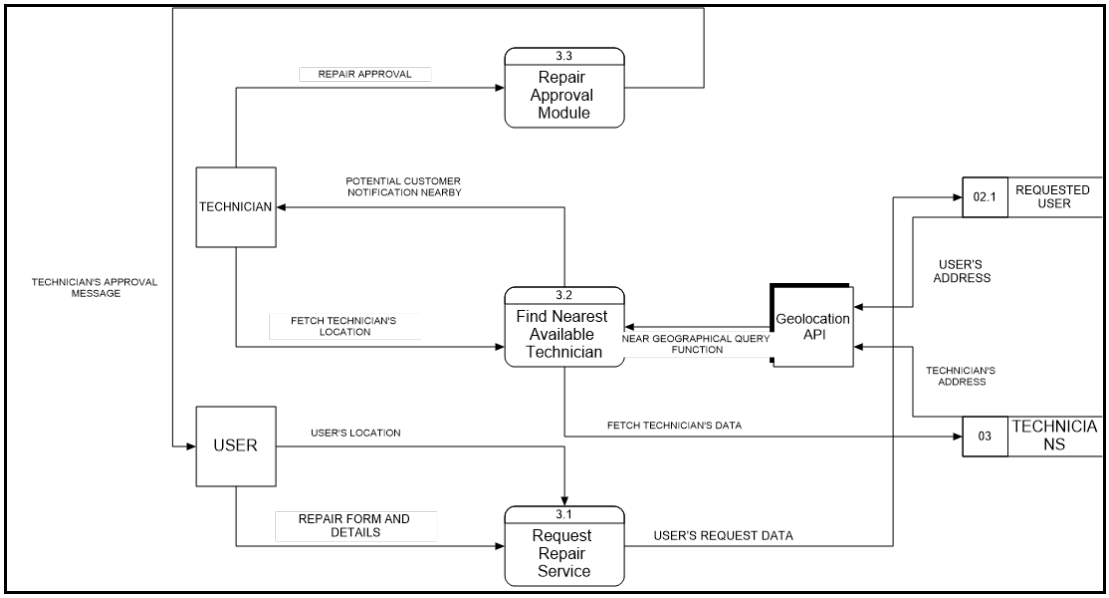


Figure 4: Techna's Data Flow Diagram Level 1 for Request Repair Module

The user started the process by requesting repair service by filling their repair form and details. Then, the requested data will be stored in the database named requested user database, a database that rapidly adds and removes data due to constant updating. The data will then display to the available technicians if there is any. The process of finding the nearest available technicians are as follows: -

1. Requested User's location data will be retrieved by the Geolocation API.
2. The Geolocation API will automatically fetch the pre-registered location of online technicians that is near 10 km radius from the user's requested location.
3. If there are available technicians nearby, the details of issues the user had will be displayed at the technicians dashboard, waiting for the approval.

Then the technicians can choose whether to approve their availability or not. If so, the technician and the user will be connected to the appointment page for further process.

3.5 Database Design

In this part, after the flows and process are mapped in the diagram, the data involved will be seen clearly. Firstly, there are 6 known big databases that had been identified earlier during the creation of Data Flow Diagram Level 0 which are the admin database, the technician database, the user database, the manufacturer database, the transaction due/request database and the transaction receipt database. All of these databases need to correlate with each other for easier understanding of data entered for each database, reducing any data redundancy and a smoother transfer for each data.

For example, Figure 5 shows the Entity-Relationship Diagram for User Database, Technician Database and Manufacturer Database. By creating the ERD, a new smaller database appeared in order to stabilize the economy of the data transfer.

In detail, the ERD below shows the relationship of databases when requesting the repairing service. Each user can request many repair services, but not one at a time. Whereby the requested repair will be distributed to all nearby technicians that are available. Then, the technician will choose the desired user

as their customer, resulting in a new database called "Appointment" database, which inherits data from the related database, in this case the user database and the technician database. The highlighted blue text are the foreign keys retrieved from the related entity, whereby the italic style in the ERD are the primary key of the database.

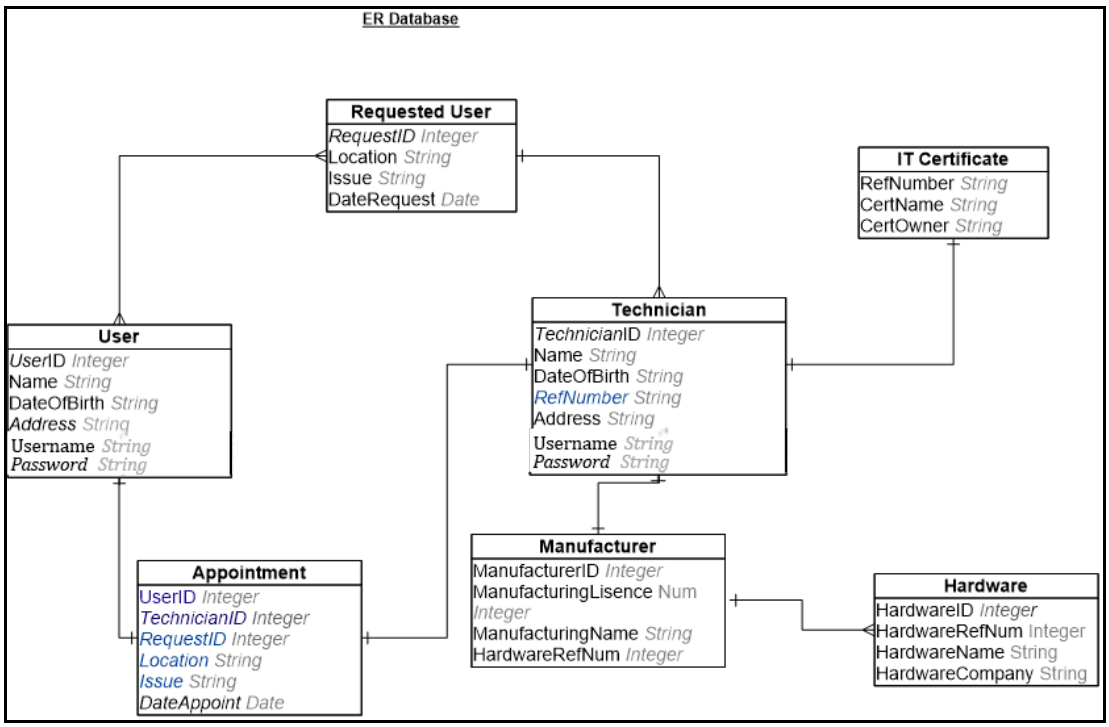


Figure 5: The ERD for request repair service

4. Result and Discussion

Techna was developed using three editors, first is the "What You See Is What You Get" (WYSIWYG) editor for styling and designing the layout, Brackets Text Editor for building the hypertext markup language (HTML), Javascript and PHP of this website.

4.1 User Interface

During the previous section, the interface design had been discussed on how Techna approached their design in the web-system prototype. Figure 6, Figure 7 and Figure 8 shows the final user interface on the system.

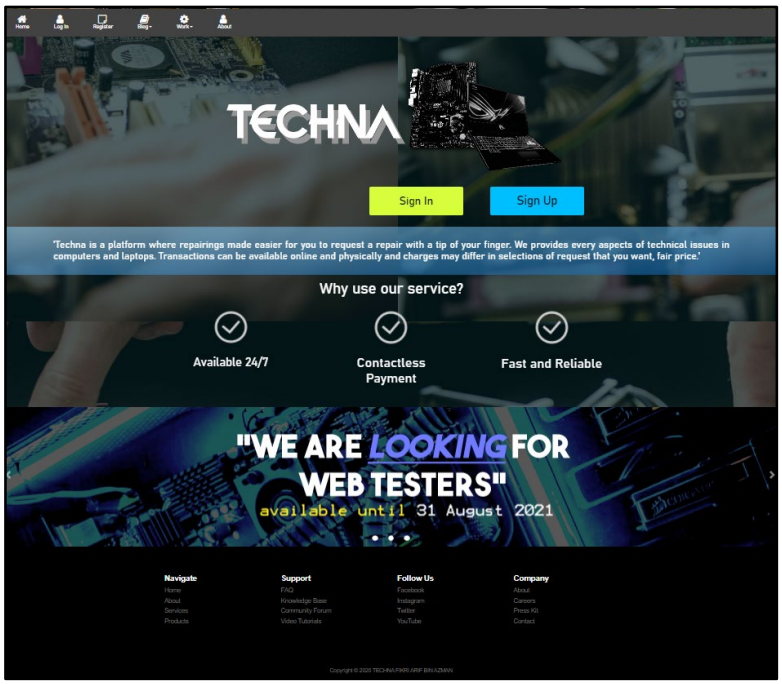


Figure 6: Home page of the website

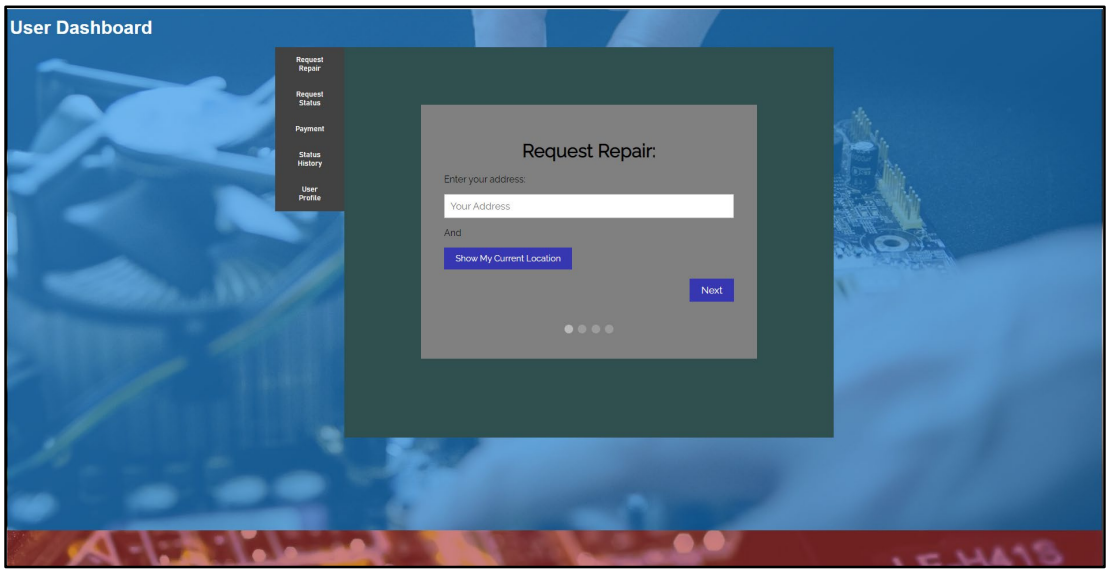


Figure 7: The user's dashboard

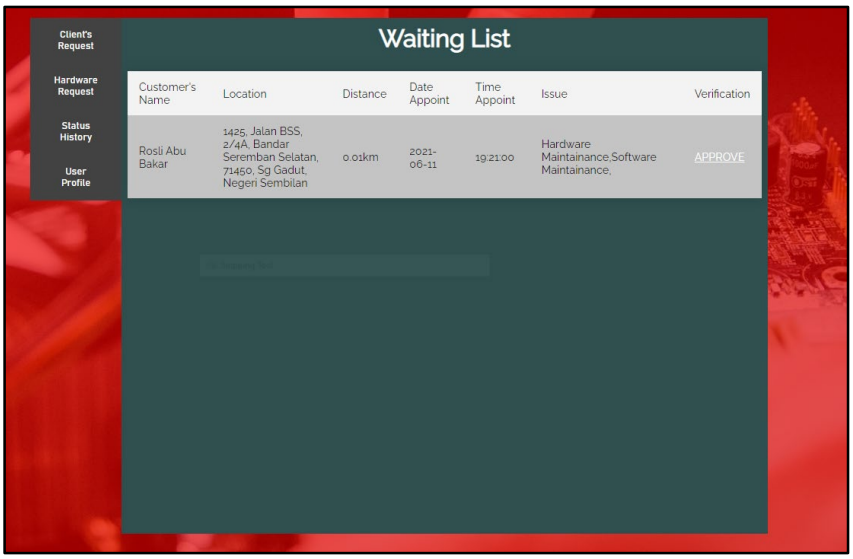


Figure 8: The technician's dashboard

In the backend of this website, there is a calculation of the distance between the user and the technician. The calculation had been done in the PHP language. Figure 9 shows the coding snippets of the formula. The formula is created as a function and will be called right after the technician logs in. Users who request the repair and are under 5 km near the technician will be displayed in the front-end of the technician's dashboard.

```php
//Distance Check
$sqlfind1 = "SELECT * FROM requestrepair WHERE status = 'pending'";
$result1 = mysqli_query($conn,$sqlfind1) or die( mysqli_error($conn));
$row1 = mysqli_fetch_field($result1);
      //THE DISTANCE FUNCTION
function distance($lat1, $lon1, $lat2, $lon2, $unit) {
  if (($lat1 == $lat2) && ($lon1 == $lon2)) {
    return 0;
  }
  else {
    $theta = $lon1 - $lon2;
    $dist = sin(deg2rad($lat1)) * sin(deg2rad($lat2)) +  cos(deg2rad($lat1)) * cos(deg2rad($lat2)) * cos(deg2rad($theta));
    $dist = acos($dist);
    $dist = rad2deg($dist);
    $miles = $dist * 60 * 1.1515;
    $unit = strtoupper($unit);

    if ($unit == "K") {
      return (round($miles * 1.609344, 2));
    } else if ($unit == "N") {
      return ($miles * 0.8684);
    } else {
      return $miles+100000;
    }
  }
}
if (!empty($result1) && $result1->num_rows > 0) {

    while($row1 = $result1->fetch_assoc()) {
$lalala = distance($lat_check, $long_check, $row1['latitude'], $row1['longitude'], "K");
if ($lalala < 5){
    echo "<tr><td value='".$row1['user_name']."'>" .$row1['user_name']. "</td><td>".$row1['address']. "</td><td>" .$lalala. "km</td>
```

Figure 9: The snippet coding of calculating distance between the user and the technician

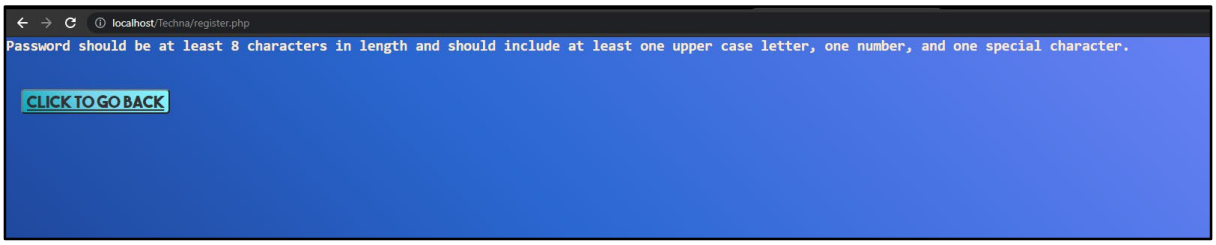
During the registration page, the user was prompted to enter a strong password that includes an uppercase, lowercase, a number and a special character. The minimum of the password length must be above 8. If the criteria did not meet, an error will occur. Figure 10 shows the snippet code of the validation process and Figure 11 is an error message if the user did not meet those criteria.

```php
$username = filter_input(INPUT_POST, 'username');
$password = filter_input(INPUT_POST, 'psw');
$dob = filter_input(INPUT_POST, 'dob');
$name = filter_input(INPUT_POST, 'user_name');
$uppercase = preg_match('@[A-Z]@', $password);
$lowercase = preg_match('@[a-z]@', $password);
$number    = preg_match('@[0-9]@', $password);
$specialChars = preg_match('@[^\w]@', $password);
$sql = "INSERT INTO usertechna (username,password,user_name,user_dateofbirth)
values ('$username',MD5('$password'),'$name','$dob')";
$sql2 = "SELECT concat(left(max(id),3),right(max(id),3) +1) from usertechna";
if(!$uppercase || !$lowercase || !$number || !$specialChars || strlen($password) < 8) {
     echo '<h2>Password should be at least 8 characters in length and should include at least one upper case letter, one number, and
     one special character.</h2>';
     echo "<br><br><button style='font-size : 22px; text-align : center;float: left; margin-left: 20px'><a style='color:#333;' href =
     'register.html' style='float:left'>Click To Go Back</a></button>";
}else {
if ($conn->query($sql)){

echo "You have been registered!";
echo "<br><br><button style='font-size : 22px; text-align : center;float: left;'><a style='color:#333;' href = 'user_login.html'
style='float:left'>Click Here To Login</a></button>";
}

    else {
         $error = "REGISTER FAIL!";
        echo $error;
      }
```

Figure 10: The snippet coding of checking if the password had met the criteria



5. Conclusion

In conclusion, Techna was planned and developed to create a platform for customers who seek computer repairs and technicians who seek help. The more the clients, the more efficiency this platform gets. This is the place for any new technicians who want to build their career as an online computer technician.

Unfortunately, the project is developed in a rough and unfamiliar condition, the project starts developing during a global pandemic. This alone creates a lot of limitations in this project. The limitations include: -

1. Users cannot input other geolocations other than their own current location.
2. Data displayed are not in real-time, making the user need to refresh to update the data.
3. An average standalone payment system developed by Techna.

However, these limitations will be recovered within time. Firstly, the geolocation API will be implemented as whole, making the users can input other geolocations so they can help request repairs for others. Secondly, the web socket will be implemented in the system, making the data transfers in real-time and connecting the system with secured payment gateway such as FPX and Ipay88. Finally, the system will still be in development in terms of interface, sub-modules such as creating an e-wallet and making the connected manufacturers their own platform.

Acknowledgement

The authors would like to thank the Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia for its support and encouragement throughout the process of conducting this study.

Appendix

Table 2: Functional Requirements

No.	USER TYPE	MODULES
1.	Admin	1. User, Technician and Manufacture Database: admin can update, delete and modify the data of the database. 2. Hardware dan Software Database page: List of hardware/software that need to be delivered to technician (if request any). 3. Transaction receipt database: List of successful or failed transactions made by users and technicians. 4. Wage payment page: A page for wage payment if the repair service is successfully conducted. 5. Technician verification page: Admin can check and verify registered technicians. 6. Hardware/Software Orders page: List of pending orders by technicians that request hardware/software products.
2.	Technicians	1. Register Technician page (Must have a User account first): A page for existing users to sign up as a Techna technician. 2. Page for finding customers that are in range with their location: A page that listed all customers that needed computer repairing service nearby. 3. Hardware/Software Request Order page: page for technician to order hardware/software products if any customers require a replacement or renewal. 4. User Setting and Management page: Profile updates, Banners, Profile Pictures, Addresses or any user info related can be modified here. 5. Financial Statement page: A page that displays technician's wage daily, weekly, monthly and withdrawal request.
3.	Manufacturer	1. Invoice receiver from the company/technician's email, requesting the hardware or software needed.
4.	User	1. Login/Register page (User): A page for user to login/signup as a Techna user 2. Top-down computer problem analysis page: A page that narrows down the user's computer problem 3. The transaction page: User will pay the price here.

Table 2: (cont.)

No.	USER TYPE	MODULES
		4. The Confirmation page: This page requires the user to confirm their appointment at the approved technician and choose whether to go to the technician place or just let the technician repair their home. 5. User Setting and Management page: Profile updates, Banners, Profile Pictures, Addresses or any user info related can be modified here. 6. Live chat page: A page that lets user to ask questions about any inquiries related to website

Table 3: Non-Functional Requirements

No.	ASPECTS	EXAMPLES
1.	SECURITY	1. The access permissions for data management may only be changed by the administrator of the website. 2. Passwords shall never be viewable at the point of entry or at any other time. 3. The payment system shall be accessed only by authorized admins. 4. A third-party firewall will be implemented. 5. All the sensitive data will be not exposed to each unauthorized user.
2.	ACCESSIBILITY	1. The web-system will be built in the common English language. 2. The user interface will only implement one module at a time. 3. Available for users with any computer devices, including mobile computer devices. 4. The 3-colour block rule will be used in the user interface.
3.	AVAILABILITY	5. The system will be up for 24/7. 6. Ensure a fast connection between clients and users to communicate. 7. The payment system for the technicians will operate within 3 to 5 days when requested.

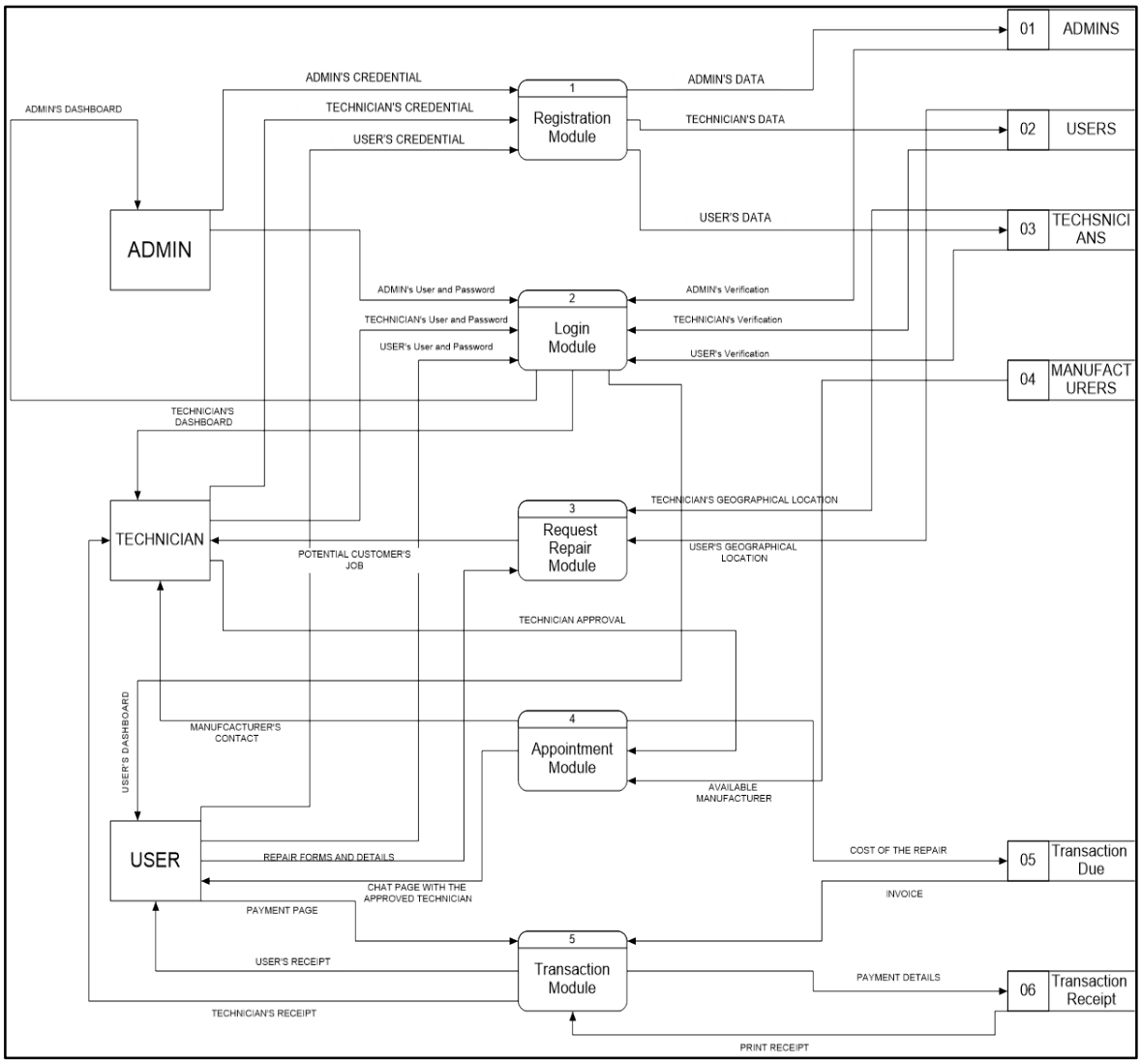


Figure 12: Data Flow Diagram Level 0

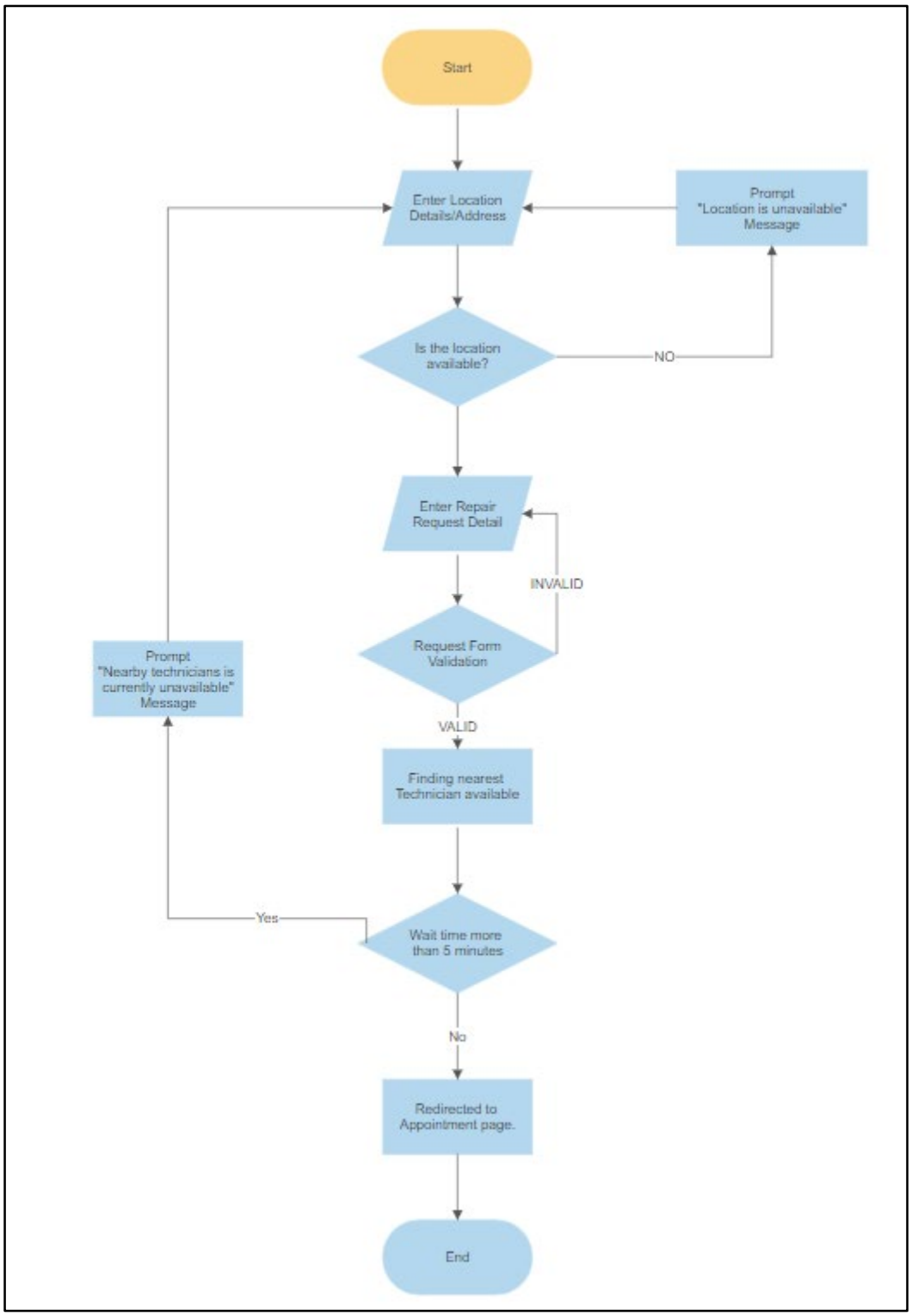


Figure 13: The Flowchart for the User's request repair module

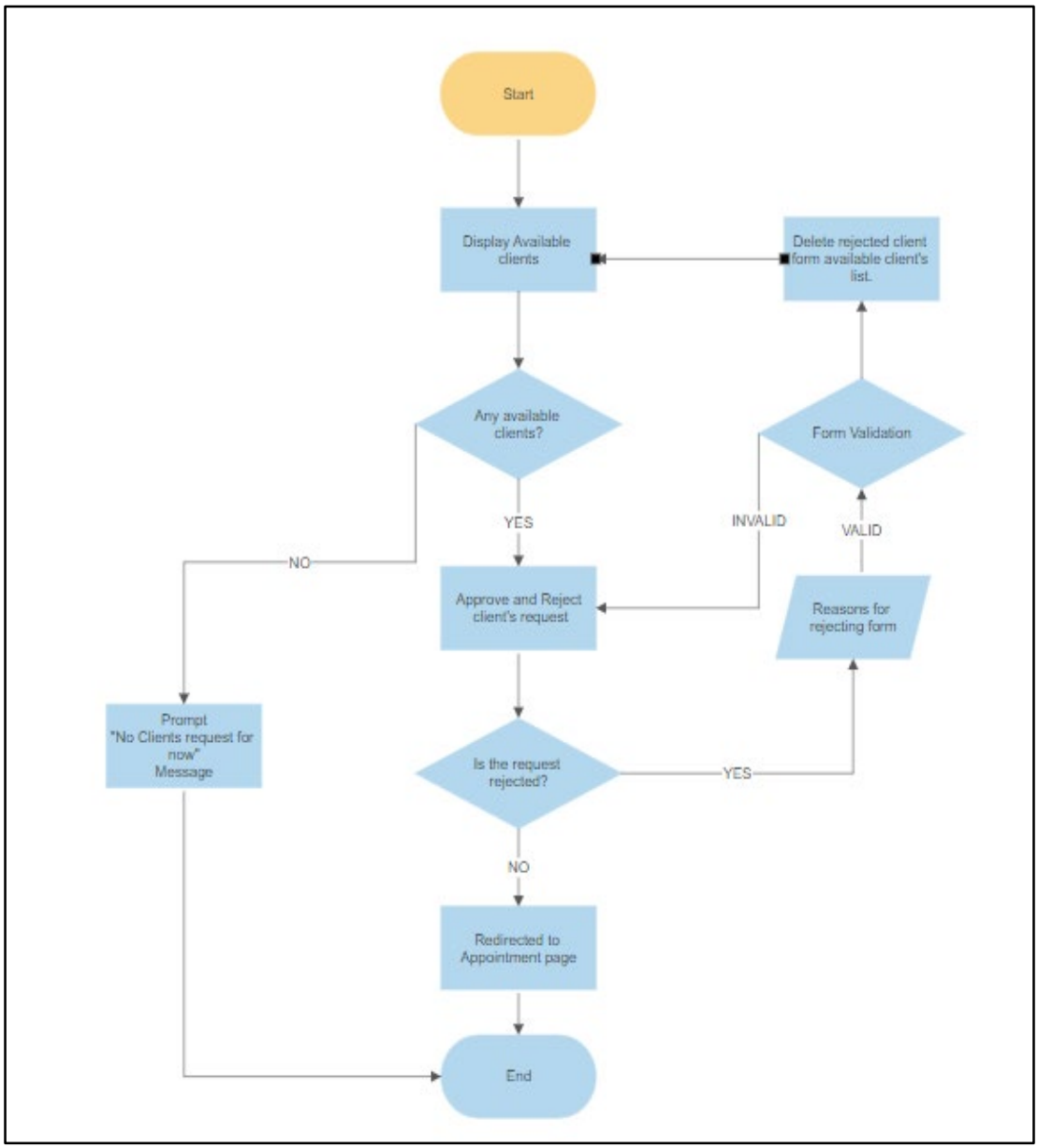


Figure 14: The Flowchart for the Technician's list nearby user's request module

References

[1] S. Goldheart, "General Computer Repair Procedures," March 2015, [Online]. Available: https://www.ifixit.com/Wiki/History/General_Computer_Repair_Procedures.

[2] OECD, "An Introduction to Online Platforms and Their Role in the Digital Transformation," May 2019, doi: https://doi.org/10.1787/53e5f593-en.

[3] L. Pepic, "The sharing economy: Uber and its effect on taxi companies," Acta Economica vol. 16 no. 28, December 2018, doi: https://doi.org/10.7251/ACE1828123P.

[4] M. Caceres, "Geolocation API Specification 2nd Edition," September 2020, [Online]. Available: https://www.w3.org/TR/geolocation-API/#introduction.

[5] M. Matthew, "Prototyping Model in Software Engineering: Methodology, Process, Approach," December 2020, [Online]. Available: https://www.guru99.com/software-engineering-prototyping-model.html.

[6] E. Quilliam, "What are the difference between software prototyping methods," February 2020, [Online]. Available: https://itenterprise.co.uk/software-prototyping-methods/.

[7] Computer Science Department, “Context Diagrams.” MSc-IT Study Material, chapter 6, January 2011, [Online]. Available: https://www.cs.uct.ac.za/mit_notes/software/htmls/ch06s06.html.
[8] USGS, “Data Dictionaries,” USGS Survey Manual, chapter 502, no. 7, 2008, [Online]. Available: https://www.usgs.gov/products/data-and-tools/data-management/data-dictionaries.