**Chapter 2 – Analysis**

2.1 - Chapter IntroductionThe process of identifying the underlying problem and comprehending the problem domain from a jumble of facts and figures is known as analysis. The purpose of the analysis is to provide a complete, consistent, and unambiguous image of the system. Also, what should the system supply to meet all user requirements. Requirements Engineering is another name for this procedure. The system can be designed with the help of analysis.

2.2 - Fact Finding Techniques

Fact-gathering procedures are information-gathering strategies used in system analysis to properly define and comprehend system requirements. The following strategies were primarily employed:

1. Interviews
2. Observation
3. Existing Documents Analyzing

2.2.1 – Interviews

The initial technique utilized to collect and establish system requirements, as well as clarify and confirm those needs, was an interview. In order to effectively address and weigh the inputs of each interviewee, the interviewer must first comprehend the perspective of each interviewee.

The following is the interview schedule that was used.

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Interviewee** | **Position** | **Interview Duration** |
| 23/10/2021 | Mr. Samudu Kannangara | Owner | 40 minutes |
| 23/10/2021 | Mr. Upali Kannangara | Stock Keeper | 30 minutes |
| 23/10/2021 | Mr. Thamara Kannangara | Technician | 40 minutes |
| 25/10/2021 | Mr. Susith Sewikrama | Delivery Manager | 20 minutes |
| 25/10/2021 | Mr. Chamara Perera | Technician | 30 minutes |
| 27/10/2021 | Miss. Kalpani Dinusha | Customer | 20 minutes |

*Table 2.1 – Stakeholders Interview Schedule*

2.2.2 – Observation

Observation is a fact-finding technique in which system analysts observe how individuals perform tasks and activities during site visits. This is an excellent approach to learn what end users go through in their day-to-day processes, and it gives you a lot of insight into the business process.

2.2.3 – Existing Documents Analyzing

Analyzing existing papers is a key strategy for acquiring requirements. When building a user-friendly system, evaluating the papers and reports of an existing system can help. Solid information and facts are usually collected by studying existing papers, which helps to corroborate and validate the requirements gathered through other methods.

2.3 - Existing System

From the beginning, U-Star Digital has used a semi-paper-based manual approach. During the system analysis phase, it was discovered that the current system can only perform a limited number of functions.

Customers used to come into the store and convey their needs to the technician who built their computer under the old system. Customers arrived at the shop, handed over the defective item or computer to the hardware professionals, and then waited in their restroom until the job was completed. To take online orders, they currently utilize a phone-based and WhatsApp-based order system. There are no other options than bank transfers, and the customer must send legal transaction paperwork to confirm the order. Customers are not given accurate delivery information due to a lack of appropriate delivery information.

2.4 - Existing System Use Case Diagrams

There were numerous downsides to the paper-based technique.

* It was discovered that data was repeating.
* Data that is unidentifiable owing to illegible handwriting.
* Data was lost because of misplacing the recording materials.
* It takes a long time.
* Manual computations are required.
* Data security is lacking.
* More manpower is required.

The following are the existing system's High-Level Use Case Diagrams for the modules listed.

Diagram

Description automatically generated

*Figure 2.1 – Use Case Diagram for the Existing System.*

2.5 - Requirements Analysis

2.5.1 - Functional Requirements

Calculations, data insertion, manipulation, and processing, as well as data presentation, are all defined by functional requirements. The system's required features are listed below.

* Computer Hardware Assistant
  + Customer selects the items begins with Processor.
  + Assist the customer with selected items matching item specifications.
  + Allow customer to make order end of the assisting process.
* Troubleshooting Assistant
  + Customers can enter the description of the defective item.
  + Detect the keywords of the entered text and suggest the solutions.
  + Customers can’t find the solutions they can make an appointment.
* Appointment Booking
  + Customers can select the date and time.
  + Customers need to fill brief of the appointment.
  + Check the item warranty to take free service.
  + The technician can pick up an appointment.
  + Generate job card for the appoiment.
  + The technician can update the status of the job card.
  + Notify the Technician in a new order.
* Shopping Cart
  + Customer search and select items to add to the cart.
  + Customers can make the order in the selected items in the cart.
  + Calculates selected items' totals and discounts.
  + Orders are recorded in the Customer Dashboard Orders section and Shop Managers Orders section.
  + Able to update order status.
  + Notify the shop manager of a new order.
* Payment
  + Customers can select the payment method.
  + Carry the order totals to the payment method.
  + Proceed to checkout with total payment.
* Inventory Management
  + Insert, update and delete the Items, Brands, Manufacturers, Models, and Item Specifications.
  + Notify with low stock and out-of-stock Items.
  + Able to create a low stock threshold.
  + Able to create discounts with a selected number of date periods.
  + Use the first in first out (FIFO) mechanism to manage inventory.
* Warranty Management
  + Notify the customer about the item's remaining warranty period.
  + Able to set the warranty durations of the items.
  + Able to create warranty types.
* Delivery Management
  + Able to create courier companies.
  + Customers can notify with courier company details and order tracking numbers when order dispatch in the warehouse.
  + Customers can view their order details by giving the courier company a tracking URL using the tracking number.
  + Customers were able to update the courier process when receiving the item.
* Invoice Generation
  + Calculate order totals and courier payment and generate an invoice for the customer.
  + Invoice can filter with the selected date range.
  + Invoices can export and have the ability to print.
* Backup and System Log
  + Enables the backup of system data and extracts the system log.
  + Track all the system behavior.
  + Ability to export the log.
* User Management
  + Able to manage all of the users of the system.
  + Aids to create user roles and permission.
* Report Management
  + This module will make it possible to generate, preview, and print a variety of reports.
  + Able to export and filter the reports within the selected time frame.

2.5.1 - Non-Functional Requirements

Non-functional requirements had to be met in order for a system to be valuable and successful.

* Usability

The system is directly connected to all of the shop's consumers. Users of the system can access all reports and appointments over the internet. Giving clients better service means making data easier to access and sending notifications about existing purchases.

* Maintainability  
  It's simple to correct a bug, add new features, make changes to support new operating systems, and make the software easier to maintain for others. It also improves the contentment of members.
* Performance

The system should be quick and finish the task in the shortest time possible. Data should be retrieved as soon as feasible, and the response time should be minimal.

* Security

Considering client personal information such as email addresses, phone numbers, and physical addresses is saved in the system, security is a top priority for the system.

* Reliability and accuracy

The system's major features include the generation of progress reports and workout programs. As a result, the system's accuracy and dependability are critical.

2.6 - Similar Systems and Literature Reviews

It was required to examine the existing system before designing the web-based Computer Hardware Purchasing and Troubleshooting Assistant management system. Because the current system is semi-paper-based, a literature review and related system analysis were also decided. The systems listed below were evaluated in order to have a thorough understanding of the situation.

Graphical user interface

Description automatically generated2.6.1 – Microcenter

*Figure 2.2 – Interface Design of Microcenter Similar System*

Micro Center offers more computers, electronics, networking, and communication devices (more than 30,000 items in stock) than any other company. Micro Center is deeply passionate about providing information technology products and technology support services. We have offered in-store pickup of online orders since 2010.[1]

2.6.2 – Newegg

Graphical user interface, website

Description automatically generated

*Figure 2.3 – Interface Design of NewEgg Similar System*

Newegg began selling PC components and helped popularize the PC-building movement, and in doing so, developed a cult following among PC and DIY enthusiasts. [2]

2.6.3 - Comparison of the Proposed System to a Comparable System

The results of the comparison as below (table 2.2)

|  |  |  |  |
| --- | --- | --- | --- |
| **Functionality** | **Proposed System** | **Microcenter** | **NewEgg** |
| Computer Hardware Assistant | Yes | Yes | No |
| Troubleshooting Assistant | Yes | No | No |
| Appointment Booking | Yes | Yes | No |
| Shopping Cart | Yes | Yes | Yes |
| Payment Management | Yes | Yes | Yes |
| Inventory Management | Yes | Yes | Yes |
| Warranty Management | Yes | No | Yes |
| Delivery Management | Yes | Yes | Yes |
| Mobile App | No | No | Yes |

*Table 2.2 – Comparison between Proposed System and Similar Systems*

2.7 - Model of the Proposed System's Process

The Software Development Process, also known as the Software Development Lifecycle, is the division of software development activity into phases in order to better design, project management, and product management.

Scrum, Extreme Programming (XP), RUP, and other process models exist for developing software systems.

2.7.1 – Scrum

“The Scrum process organizes development into a sequence of sprints, each of which results in a potentially usable product with an added increment of function. The tasks for each sprint are set, in consultation with a stakeholder representative, during a sprint planning meeting and cannot be added to during the sprint. Each task is typically expressed as a user story. Each sprint is time boxed: the end date of the sprint does not change. Tasks that cannot be accomplished in time are returned by the team to the backlog for future consideration.” [3]

2.7.2 – Extreme Programming (XP)

“Extreme programming (XP) is a software development methodology which is intended to improve software quality and responsiveness to changing customer requirements. As a type of agile software development, it advocates frequent "releases" in short development cycles, which is intended to improve productivity and introduce checkpoints at which new customer requirements can be adopted.”

2.7.3 – Rational Unified Process (RUP)

“Stands for "Rational Unified Process." RUP is a software development process from Rational, a division of IBM. It divides the development process into four distinct phases that each involve business modelling, analysis and design, implementation, testing, and deployment. The four phases are:

1. **Inception -** The idea for the project is stated. The development team determines

if the project is worth pursuing and what resources will be needed.

1. **Elaboration -** The project's architecture and required resources are further

evaluated. Developers consider possible applications of the software and costs

associated with the development.

1. **Construction -** The project is developed and completed. The software is

designed, written, and tested.

1. **Transition -** The software is released to the public. Final adjustments or updates

are made based on feedback from end users.

The RUP development methodology provides a structured way for companies to envision create software programs. Since it provides a specific plan for each step of the development process, it helps prevent resources from being wasted and reduces unexpected development costs” [5]

Based on the considerations outlined in Table 2.3, the Rational Unified Process (RUP) was chosen as the best process model for the proposed system over other approaches.

|  |  |  |
| --- | --- | --- |
| RUP | Scrum | XP |
| A good option for a huge, long-term project. | There is no set termination date. At the end of the current iteration, the next iteration plan is determined. | Release regularly. |
| The planning process is driven by the end date and includes intermediate milestones. | Scrum employs a project backlog instead of scope. | A shorter time limit |
| Requirements are fixed | It's ideal for rapid upgrade tasks that don't have to be completed by a certain date. | Expecting adjustments in requirements. |
| The scope of the project is predetermined. | Documentation is severely limited. | There is a lack of general design. |
| Documentation is required. |  | There is little to no documentation. |

*Table 2.3 – Comparison between Different Process Models*