**Philippine Copyright 2024**

by the Authors’ Names

and the

Polytechnic University of the Philippines

All rights reserved. Portions of this manuscript may be reproduced with proper referencing and due acknowledgment of the author.

**VILLA SALUD CATERING SERVICES INFORMATION SYSTEM: A COMPREHENSIVE SOLUTION FOR EVENT MANAGEMENT**

A Capstone Project

Presented to the Faculty of the College of Computer and Information Sciences

Polytechnic University of the Philippines

Sta. Mesa, Manila

In Partial Fulfilment of the Requirements for the Degree

Bachelor of Science in Information Technology

.

**Gasta, Marc Oliver L.**

**Rosario, Angelica G.**

**Armedilla, Alliyah Samantha D.**

**Santos, Akisha Gelsey L.**

**January 2024**

**APPROVAL SHEET**

This capstone project **TITLE OF CAPSTONE PROJECT (IN ALL CAPS AND IN BOLD FACE)** prepared and submitted by NAME OF RESEARCHERS (FN MI LN; Alphabetical Order) in partial fulfilment of the requirements for the degree, BACHELOR OF SCIENCE IN INFORMATION TECHNOLOGY has been:

Recommended for Approval and Acceptance

Date:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Mr. John Dustin D. Santos, MIT**

Faculty in-Charge

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Mr. John Dustin D. Santos, MIT**

Adviser (omit if same as FIC or none)

Approved by the Capstone Project Defense Panel:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Assoc. Prof. Name, MSIT**

Chair

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Asst. Prof. Name, MIS**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Prof. Name, DIT**

Accepted for the Department of Information Technology:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Asst. Prof. Lydinar D. Dastas, MSCS**

Chairperson

Accepted for the College of Computer and Information Sciences:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Prof. Benilda Eleonor V. Comendador, DIT**

Dean

**ACKNOWLEDGMENTS**

*SAMPLE ONLY. YOU MAY CREATE YOUR OWN*

The researchers would like to express their sincerest gratitude to their advisor, \_\_\_\_\_\_\_ [Advisor Name], for their invaluable guidance and support throughout their master’s program. Their expertise and encouragement helped the researchers to complete this research and write this thesis.

They would also like to thank \_\_\_\_\_\_\_ [Name] and \_\_\_\_\_\_\_ [Name] for serving on their thesis committee and providing helpful feedback and suggestions.

**CERTIFICATION OF ORIGINALITY**

This is to certify that the research work presented in this capstone project, COMPLETE TITLE OF THE CAPSTONE PROJECT for the degree Bachelor of Science in Information Technology at the Polytechnic University of the Philippines embodies the result of original and scholarly work carried out by the undersigned. This capstone project does not contain words or ideas taken from published sources or written works that have been accepted as basis for the award of a degree from any other higher education institution, except where proper referencing and acknowledgement were made.

|  |  |
| --- | --- |
|  | (Wet Signature)  **NAME OF RESEARCHERS**  Researchers  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Date Signed (date, month, year) |

**ABSTRACT**

Title : Title of Capstone Project

Researcher : LN, FN MI

Degree : Bachelor of Science in Information Technology

Institution : Polytechnic University of the Philippines

Year : 2024

Adviser : Name of Adviser

(Note: Abstract should be double-spaced, and not more than 250 words)

Keywords : At least five (5) keywords must be given (To include “Polytechnic University of the Philippines”)

**EXECUTIVE SUMMARY**

SAMPLE ONLY. The project is about scheduling and monitoring truck bodies in production. The main problem stems from late delivery of finished units due to factors that comprises it. The main objective of the study is to develop a system that will improve the current manual process regarding their monitoring and scheduling of the stages of works that is done and delays of the units that must be delivered on time. The plant manager and the admin will manage this system, which has the capability of creating job order/s, assign work order/s, update and monitor stage and status of the current job order. The system will generate reports and display status.

The Truck Body Production Scheduling and Monitoring System consists of 5 users, the plant manager, agent, admin officer, production head, and the quality assurance. The plant manager and the admin officer has the capability of monitoring everything that is happening in the production. They can also add/create/update job orders. The agent is capable only for adding job order. The production head is in charge of updating production stages and statuses, and manage reports. The quality assurance is responsible for inspecting the final product and must ensure that it observes the quality standards.

The project team was able to gather data needed through the help of previews research documents/ projects. The team also conducted client interviews and consultations to adviser and faculty-in-charge for proper guidance in the project development. Moreover, the project team constructed survey questionnaires to be answered by the target users of the project. As a result of that, the team gathered essential response that is valuable in improving the project or in conducting revisions.

Based on the overall result of the survey, the system meets the clients’ needs and give them a useful system that helps their job more productive and more organized. This will give more focus on the innovation of the system. However, there are some things that can be improved for similar projects in the future.

**Table of Contents**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | |  | | | | **Page** | |
| **Title Page** | | | | | | i | |
| **Certification and Approval Sheet** | | | | | | ii | |
| **Acknowledgments** | | | | | | iii | |
| **Certification of Originality** | | | | | | iv | |
| **Abstract** | | | | | | v | |
| **Executive Summary** | | | | | |  | |
| **Table of Contents** | | | | | |  | |
| **List of Tables** | | | | | |  | |
| **List of Figures** | | | | | |  | |
| **CHAPTER 1 – INTRODUCTION** | | | | | |  | |
| **1.1** | | **Project Context** | | | | 1 | |
| **1.2** | | **Technical Background** | | | | 3 | |
| 1.2.1 | | | Equipment/Hardware | | |  | |
| 1.2.2 | | | Software | | |  | |
| 1.2.3 | | | Peopleware/Manpower | | |  | |
| 1.2.4 | | | Network Infrastructure/Architecture | | |  | |
| 1.2.5 | | | Storage, Backup and Recovery Procedure | | |  | |
| 1.2.6 | | | Security Procedures | | |  | |
| 1.2.7 | | | Policies and Procedures | | |  | |
| **1.3** | | **Problem Analysis** | | | |  | |
| 1.3.1 | | | Fishbone Diagram | | |  | |
| 1.3.2 | | | Problem and Solution Statement | | |  | |
| 1.3.3 | | | Problem-Requirements Matrix | | |  | |
| **1.4** | | **Purpose and Description** | | | |  | |
| **1.5** | | **Specific Objectives** | | | |  | |
| **1.6** | | **Scope and Limitations** | | | |  | |
| **CHAPTER 2 – review of related literature, Studies, and systems** | | | | | | | |
|  | Topic 1 | | | |  | |
| **chapter 3 – methodology** | | | | | |  | |
| **3.1** | | **Requirements Analysis** | | | |  | |
| 3.1.1 | | | Requirements – Features Matrix | | |  | |
| 3.1.2 | | | Use Case Diagram | | |  | |
| 3.1.3 | | | Use Case Report | | |  | |
| **3.2** | | **Design Specifications** | | | |  | |
| 3.2.1 | | | Activity Diagram | | |  | |
| 3.2.2 | | | GUI Design | | |  | |
| 3.2.3 | | | Database Schema | | |  | |
| 3.2.4 | | | Data Dictionary | | |  | |
| **3.3** | | **Development Methodology** | | | |  | |
| 3.3.1 | | | Process Model | | |  | |
| 3.3.2 | | | Development Tools | | |  | |
| **3.4** | | **Test Methodology/Procedures** | | | |  | |
| **3.5** | | **System Requirements** | | | |  | |
| **3.6** | | **Quality Plan** | | | |  | |
| **3.7** | | **Evaluation Plan** | | | |  | |
| **CHAPTER 4 – RESULTS AND DISCUSSION** | | | | | |  | |
| **CHAPTER 5 – SUMMARY OF FINDINGS, CONCLUSION AND RECOMMENDATION** | | | | | | | |
| **REFERENCES** | | | | | |  | |
| **APPENDICES** | | | | | |  | |
| Appendix A | | | | Data Gathering Instrument | |  | |
| Appendix B | | | | Client Forms and Reports | |  | |
| Appendix C | | | | Evaluation Tools, Test Documents and Test Results | |  | |
| Appendix D | | | | User’s Manual | |  | |
| Appendix E | | | | Sample Generated Outputs | |  | |
| Appendix F | | | | Certification of Originality Check/Turnitin Results | |  | |
| Appendix G | | | | Certification of Editing | |  | |
| Appendix H | | | | Implementation Report | |  | |
| Appendix I | | | | Capstone Project Revision Matrices | |  | |
| Appendix J | | | | Ethics Clearance and Terminal Report | |  | |
| Appendix K | | | | Biographical Statements | |  | |

**LIST OF TABLES**

|  |  |  |
| --- | --- | --- |
| **Table No.** | **Description** | **Page** |
| 1 | Table Description 1 | 3 |
| 2 | Table Description 2 | 4 |

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **Figure No.** | **Description** | **Page** |
| 1 | Figure Description 1 | 3 |
| 2 | Figure Description 2 | 6 |

**Chapter 1**

**INTRODUCTION**

* 1. **PROJECT CONTEXT**

The content should use Arial typeface with 11pt font size. The alignment should be justified and the spacing should be 2.0 or double spacing.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla at lectus felis. Phasellus ullamcorper massa sed ex porttitor lacinia. Duis tempus, elit vel placerat vestibulum, velit diam rutrum tortor, quis aliquet urna nibh ut libero. Proin vel imperdiet ligula. Maecenas aliquam faucibus viverra. Praesent turpis dolor, eleifend ut pharetra at, sollicitudin et orci. Donec sit amet nisi elit. Ut facilisis justo sed gravida pretium. Aliquam bibendum urna ac nisl pellentesque, eu volutpat ipsum volutpat. Maecenas leo nisl, vehicula et risus egestas, sollicitudin malesuada mauris. In convallis varius ex, vel blandit nunc placerat a. Maecenas a interdum nulla, quis dictum nisl. Proin scelerisque, nibh sed mattis interdum, nulla erat gravida augue, ac semper dolor orci at nisl.

Proin a justo id metus laoreet lacinia. Integer tincidunt enim eu ex iaculis, sed hendrerit mauris condimentum. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia curae; Sed lobortis dapibus dolor, a mollis massa tempus ut. Suspendisse sodales massa nulla, tincidunt congue dolor rhoncus in. Vestibulum rutrum felis ut lacinia gravida. Aenean ac nibh aliquet, porttitor risus vitae, congue nisl. Integer auctor lorem rutrum convallis vulputate.

* 1. **TECHNICAL BACKGROUND**
     1. **Equipment/Hardware**

Table 1

**Company’s Existing Equipment**

|  |  |
| --- | --- |
| **Equipment** | **Quantity** |
| Dell Personal Computers | 2 |
| Laptops | 5 |

If the table exceeds one page, split the table and put “Continuation to Table #”. Maecenas facilisis consequat sem a vulputate. Vestibulum vulputate ipsum sit amet ex posuere, suscipit tristique orci convallis. Aliquam id tincidunt nunc, vel posuere enim. Pellentesque eu felis mauris. Morbi non consectetur nunc. Nulla sit amet erat vitae augue scelerisque posuere non ut dui. Praesent at vestibulum turpis. Donec laoreet mi eros, nec aliquet quam efficitur in.

As investigated in the study of Nullam (2018), it was argued that imperdiet iaculis dui et rhoncus. Curabitur et enim ipsum. Ut accumsan tellus a suscipit elementum. Etiam pharetra volutpat consectetur. Nunc nibh enim, varius id sodales sit amet, auctor et erat. Curabitur ultrices a tellus eget vehicula. Nulla blandit, dolor non porttitor tincidunt, nulla erat faucibus libero, eu aliquet nunc felis sit amet nibh. Integer nibh nisl, ultrices eu lacus vitae, gravida ultricies diam. Fusce sollicitudin purus odio, non blandit velit malesuada id. Phasellus aliquet velit sit amet purus condimentum bibendum. Sed velit nibh, malesuada egestas pretium in, porta sed nibh. Fusce at tortor convallis, dignissim quam ac, egestas eros. Etiam a velit elit. Cras mattis a lectus et accumsan. Mauris vitae fermentum velit. Sed aliquam neque est, sit amet pharetra justo laoreet eu.

Figure 1. Name of Figure



Figure 1 shows that praesent convallis enim non ante venenatis, a pretium elit tincidunt. Cras nec ligula dui. Nulla quam justo, dignissim iaculis euismod nec, dictum eget tortor. Nulla lectus ipsum, lacinia id rhoncus ut, lacinia id dolor. Phasellus metus eros, auctor sed justo nec, rhoncus convallis libero. Aliquam tristique magna non vehicula euismod. Morbi eros felis, sollicitudin ut placerat vitae, finibus in magna. Praesent fermentum ex risus, vitae aliquam sapien finibus eu. Nullam lacinia neque at tortor egestas eleifend. Nam dapibus varius tortor nec semper. Phasellus id imperdiet eros. Aenean id ornare odio. Aenean aliquet nec tellus at placerat.

**CHAPTER 2**

**REVIEW OF RELATED LITERATURE**

**CHAPTER 3**

**METHODOLOGY**

**3.1 REQUIREMENT ANALYSIS**

**3.1.1. Features Matrix**

Table #

|  |  |  |
| --- | --- | --- |
| **Feature** | **Description** | **Requirements** |
| Inquiry Management | Allows admins to see all inquiries made by customers. Allows inquire for a possible reservation or booking | Admin can select inquiries and review them. |
| Admin can send email confirmation |
| Admin can say yes or no to inquiries before pushing as a reservation |
| Admin Dashboard | Enables admins to view booked and available dates and generate reports. | Admin can view an overview of reservation status (e.g., booked, available). |
| Admin can generate reports for specific months or years. |

This table represents all the main features our system has. This explains how system works, what it means and what it’s called. This also shows the requirements needed to meet the feature or the capabilities of the system. It can also show who is involved in the feature.

**3.1.2. Use Case Diagram**

Figure #. Admin Side

****

Figure # shows the use case diagram for the admin side of the system. This shows the interaction of the admin into the system. It shows the capabilities or the abilities of an admin into the system. It also represents the behavior of the admin towards the system.

Figure #. Patron Side

****

Figure # shows the use case diagram for patron (customer). This diagram shows the behavior, interaction and, the capabilities of the patron to the system.

**3.1.3. Use Case Report**

Table #

|  |  |
| --- | --- |
| **Use Case Name** | **Manage Reservations** |
| Actors | Admin |
| Description | The admin can manage customer reservations, including updating the reservation status (confirmed, canceled, rescheduled), and checking available and booked dates. |
| Preconditions | Admin must log in to the system. |
| Main Flow | 1. Admin logs into the system. |
| 2. Admin navigates to the reservation management dashboard. |
| 3. Admin views booked and available dates. |
| 4. Admin updates reservation statuses as needed. |
| Alternative Flow | 1. If the reservation does not exist, the system displays an error message. |

Continuation to Table #

|  |  |
| --- | --- |
| Postconditions | Updated reservation details are reflected in the system. |

Table #

|  |  |
| --- | --- |
| **Use Case Name** | **Handle Inquiries** |
| Actors | Admin |
| Description | The admin can review and answer inquiries from customers. |
| Preconditions | Admin must log in to the system. |
| Main Flow | 1. Admin logs into the system. |
| 2. Admin navigates to the inquiries section. |
| 3. Admin reviews inquiries submitted by patrons. |
| 4. Admin sends responses to the inquiries. |
| Alternative Flow | 1. If there are no inquiries, the system displays a "No Inquiries Available" message. |

Continuation to Table #

|  |  |
| --- | --- |
| Postconditions | Customer inquiries are marked as reviewed/responded to. |

Table #

|  |  |
| --- | --- |
| **Use Case Name** | **Send Email Confirmations** |
| Actors | Admin |
| Description | The admin sends email confirmations for reservations to patrons. |
| Preconditions | Reservation details must be complete. |
| Main Flow | 1. Admin logs into the system. |
| 2. Admin selects a reservation and sends an email confirmation. |
| Postconditions | Customer inquiries are marked as reviewed/responded to. |

Table #

|  |  |
| --- | --- |
| **Use Case Name** | **Book Reservation** |
| Actors | Patron |
| Description | The patron can book a reservation for a specific date, selecting the food package and customizing details. |
| Preconditions | The customer must access the system through the interface. |
| Main Flow | 1. Patron accesses the reservation interface. |
| 2. Patron views available dates for reservations. |
| 3. Patron selects a date, food package, and additional details. |
| 4. Patron submits the booking request. |
| Postconditions | Reservation is successfully saved in the system. |

Table #

|  |  |
| --- | --- |
| **Use Case Name** | **View Reservation Summary** |
| Actors | Patron |
| Description | The patron can view the details of their reservation. |
| Preconditions | A reservation must exist in the system. |
| Main Flow | 1. Patron navigates to the reservation summary section. |
| 2. Patron reviews the details of their reservation. |
| Alterative Flow | If no reservation exists, the system displays a "No Reservation Found" message. |
| Postconditions | Reservation details are displayed. |

Table #

|  |  |
| --- | --- |
| **Use Case Name** | **Cancel/Reschedule Reservation** |
| Actors | Patron |
| Description | The patron can cancel or reschedule their reservation through the system. |
| Preconditions | A reservation must exist in the system. |
| Main Flow | 1. Patron navigates to their reservation. |
| 2. Patron selects the option to cancel or reschedule the reservation. |
|  | 3. System processes the cancellation or rescheduling. |
| Postconditions | Reservation is either canceled or rescheduled, and the system updates its status. |

Table #

|  |  |
| --- | --- |
| **Use Case Name** | **Receive Notifications** |
| Actors | Patron |
| Description | The patron receives email notifications for booking confirmations or updates. |
| Preconditions | Email address must be registered in the system. |
| Main Flow | 1. Patron submits a booking or updates a reservation. |
| 2. System sends an email notification to the patron. |
| Postconditions | Notification is successfully received by the patron. |

**3.2 DESIGN SPECIFICATIONS**

**3.2.1. Activity Diagram**

Patron’s processes. In the following figures, it will show the processes available in patron’s side. This shows systematic approach of the patron to the system. The process will be shown in a flowchart to better present and to easily understand the process.

Figure #. Check Availability.



Figure #. Book a Reservation.



Figure #. Modify Reservation



Figure #. View Reservation Summary



Admin’s processes. The following figure will represent the process for the admin. It will be shown in a flowchart method to easily present the systematic process of the admin in the system.

Figure #. Check Availability



Figure #. Manage Reservations



Figure #. Respond to Inquiries



**3.2.2. GUI Design**

**Figure #. Landing Page**

****

**Figure #. Admin Account Creation**

****

**Figure #. Admin Log in**

****

**Figure #. Inquiry Page**

****

**Figure #. Dates Page**



**Figure #. Patron’s Homepage**



**Figure #. Patron: View Reservation**



Figure #. Creation of Reservation



**Figure #. Initial Inquiry Questions**



**3.2.3. Database Schema**



**3.2.4. Data Dictionary**

Table #. Admin Table

|  |  |  |  |
| --- | --- | --- | --- |
| **ADMIN** | | | |
| Field Name | Data Type | Field Size | Description |
| id | INT |  | Description ID |
| admin\_id | VARCHAR | 255 | Unique admin’s ID |
| email | VARCHAR | 255 | Admin’s email address |
| name | VARCHAR | 255 | Admin’s last name |
| phone\_num | INT |  | Admin’s phone number |
| time\_created | TIMESTAMP |  | Time for when the table has created |
| time\_updated | TIMESTAMP |  | Time for when the table has been updated |

Table #. Patron’s Table

|  |  |  |  |
| --- | --- | --- | --- |
| **PATRON (CUSTOMER)** | | | |
| Field Name | Data Type | Field Size | Description |
| id | INT |  | Description ID |
| patron\_id | VARCHAR | 255 | Unique patron’s ID |

Continuation of Table #.

|  |  |  |  |
| --- | --- | --- | --- |
| Email | VARCHAR | 255 | Patron’s email address |
| name | VARCHAR | 255 | Patron’s last name |
| address | VARCHAR | 255 | Patron’s address |
| phone\_num | INT |  | Patron’s phone number |
| time\_created | TIMESTAMP |  | Time for when the table has created |

Table #. Reservation Table

|  |  |  |  |
| --- | --- | --- | --- |
| **RESERVATION** | | | |
| Field Name | Data Type | Field Size | Description |
| id | INT |  | Description ID |
| reserve\_id | VARCHAR | 255 | Unique reservation’s ID |
| patron’s id | VARCHAR | 255 | Patron’s unique id |
| num\_pax | INT | 255 | Number of persons that will attend |
| food\_package | ENUM |  | What type of food package the customer will avail |

Continuation of Table #.

|  |  |  |  |
| --- | --- | --- | --- |
| time\_created | TIMESTAMP |  | Time for when the table has created |
| time\_updated | TIMESTAMP |  | Time for when the table has been updated |

Table #. Inquiries Table

|  |  |  |  |
| --- | --- | --- | --- |
| **INQUIRIES** | | | |
| Field Name | Data Type | Field Size | Description |
| id | INT |  | Description ID |
| inquiry\_id | VARCHAR | 255 | Unique event’s ID |
| patron\_id | VARCHAR | 255 | Unique patron’s ID |
| reserve\_id | VARCHAR | 255 | Unique reservation’s ID |
| time\_date | VARCHAR | 255 | Time and date of the event |
| place | VARCHAR | 255 | Place of the event |
| theme | VARCHAR | 255 | The theme of the event |
| motif | VARCHAR | 255 | Motif of the event |
| time\_created | TIMESTAMP |  | time for when the table has created |
| time\_updated | TIMESTAMP |  | time for when the table has been updated |

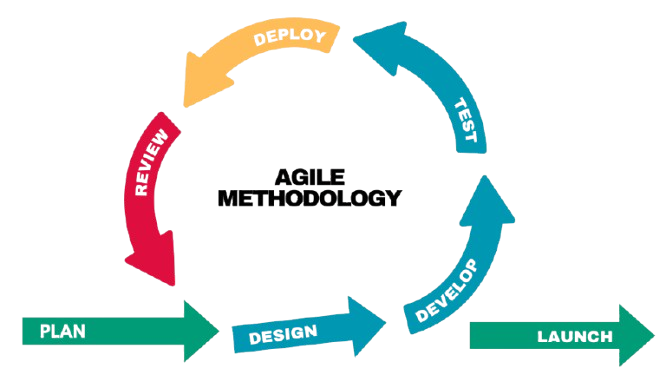
Table #. Payment Table

|  |  |  |  |
| --- | --- | --- | --- |
| **PAYMENT** | | | |
| Field Name | Data Type | Field Size | Description |
| id | INT |  | Description ID |
| pay\_id | VARCHAR | 255 | Unique payment’s ID |
| reserve\_id | VARCHAR | 255 | Unique reservation’s ID |
| patron\_id | VARCHAR | 255 | Unique patron’s ID |
| amount | VARCHAR | 255 | Amount to be paid by the customer |
| date | DATE |  | Date the order payment has been issued |
| time | TIME |  | Time the order payment has been issued |
| payment\_date | DATETIME |  | Date of the possible settlement of date of the customer |
| payment\_type | ENUM |  | What type of payment the customer will do; full or down payment |
| time\_created | TIMESTAMP |  | time for when the table has created |
| time\_updated | TIMESTAMP |  | time for when the table has been updated |

**3.3. DEVELOPMENT TOOLS**

**3.3.1. Process Model**

Figure #. Agile Methodology



We applied the Agile methodology in developing the system because it allows us to adjust to changes in the project requirements more effectively. This approach ensures our efforts are focused and efficient while also making it easier to improve or modify the system’s design when needed. If there are updates or revisions to the design, Agile helps us track the system’s progress more effectively.

By using Agile, we adopted a flexible and collaborative approach to system development. This method allows us to handle changes smoothly, simplify updates to the system’s structure, and monitor its progress closely. With Agile principles, our goal is to create a reliable, high-quality system that automates and replaces manual processes, improving efficiency and overall service quality for Villa Salud.

**Planning**

In Agile methodology, everything begins with the Planning Phase. Here, we focused on gathering all the information needed to develop the system. We clearly defined the project’s scope and limitations, created an initial plan for implementation, and assessed how feasible and credible the project would be. To better understand what the client needed, we also conducted an interview to gather their requirements and ensure those needs would be implemented effectively in the system.

**Design**

In this phase, we created a layout based on the client’s requirements. We designed the interface to be simple and easy to use, aiming to ensure user satisfaction. To achieve this, we used a minimal color scheme, made the design responsive, and prioritized a user-friendly approach. We used Figma to visualize the system, allowing the client to see a preview of the partial outcome and provide feedback for improvements.

**Develop**

In the development stage, the developer is assigned to work on the system’s integrity, design, layout, and functionality. The developer gathers all the necessary information from the client and ensures the system is properly integrated. The features to be implemented in the system include managing event reservations, organizing menu packages, automating scheduling, generating reports, creating announcements. When writing the code, the system's design, specs, and the customer's specific requirements are all carefully considered. By taking these aspects carefully, we will make sure that the end product satisfies the system's expectations and is in line with the planned goals.

**Test**

In the testing phase, the Quality Assurance (QA) team checks how well the system works, how it performs, and its overall design. This includes performing functionality, integrity, and unit testing. In addition to testing the system's responsiveness and design, any faults or errors are found and fixed. Before the system is fully implemented at Villa Salud, the researchers will test it with the client, a small group of users, and employees to make sure it is functional.

**Deploy**

After the testing phase, the system will be deployed and checked by the client. During this deployment stage, the client will use the system and test its functionalities and integrations to ensure everything works as expected.

**Review**

The review stage is when the client evaluates their experience using the system and provides feedback. During this stage, the client can also suggest improvements to the system’s functionalities or requirements to enhance its performance.

**Launch**

The launch stage is when the developers officially launch the system and deploy it for public use. During this stage, the system becomes fully operational and accessible to the intended users. The researchers ensure that everything is in place for the system’s smooth functioning, and it is made available for the public or the target audience to begin using.

**3.3.2. Development Tools**

|  |  |
| --- | --- |
| Programming Languages | JavaScript, PHP |
| Supporting Tools | HTML, CSS, GITHUB |
| Relational Database Management  System | PHP, MySQL |
| Integrated Development Environment | Visual Studio Code, GitHub |
| Server | Google |
| Web Browser | Google Chrome, Microsoft Edge, Safari |