Palengk-E: A Farmer's Mobile Marketplace Application

By

Diñoso, Neilson P.

Mariscotes, Reinier

Pangilinan, Dylan Ranze Z.

San Juan, Marc Anthony

B.S Computer Engineering
Technological Institute of the Philippines

A Final Documentation Submitted to the Department of Computer Engineering, in Partial Fulfillment of the Requirements of Software Design Course

Month Year

TECHNOLOGICAL INSTITUTE OF THE PHILIPPINES

1338 Arlegui St., Quiapo, Manila

DEPARTMENT OF COMPUTER ENGINEERING

APPROVAL SHEET

The proposed project entitled **Palengk-E: A Farmer's Mobile Marketplace Application**

which was presented on **Date of Presentation** by the proponents:

Neilson P. Dionoso
Reineir Mariscotes
Dylan Ranze Z. Pangilinan
Marc Anthony M. San Juan

is hereby APPROVED by the following members of the committee:

Firstname M. Lastname Panel Member	**Firstname M. Lastname** Panel Member	**Firstname M. Lastname** Panel Member

Engr. Rufo I. Marasigan Jr.
Class Adviser

Engr. Jennifer B. Enriquez
Chair of Computer Engineering

TECHNOLOGICAL INSTITUTE OF THE PHILIPPINES

1338 Arlegui St., Quiapo, Manila

DEPARTMENT OF COMPUTER ENGINEERING

ACCEPTANCE SHEET

The project entitled entitled **Palengk-E: A Farmer's Mobile Marketplace Application** has been prepared and submitted by the proponents:

Neilson P. Dionoso
Reineir Mariscotes
Dylan Ranze Z. Pangilinan
Marc Anthony M. San Juan

for approval to the committee for the *Software Design.*

After a thorough review and evaluation of the proposed system, the committee has accepted the presented proposed design based on the required criteria.

The acceptance is valid for the information being presented. Accepted this **Date of Presentation, 2nd semester, SY-2022-2023.**

Firstname M. Lastname Panel Member	**Firstname M. Lastname** Panel Member	**Firstname M. Lastname** Panel Member

Dennis S. Nava, MIT
Class Adviser

Engr. Jennifer B. Enriquez
Chair of Computer Engineering

Acknowledgement

This research project would not have been possible without the support of many people. The author wishes to express his gratitude to his Project Adviser, Engr. Juan Dela Cruz was abundantly helpful and offered invaluable assistance, support, and guidance.

My deepest gratitude is also due to the members of the panel committee, Engr. Your Panel 1, Engr. Your Panel 2 and Engr. Your Panel 3, without their knowledge and assistance this study would not have been successful.

The author would also like to convey thanks to Engr. Jennifer Enriquez and TIP CpE-Manila Faculty Members for providing valuable information as the guidance of the project.

The author wishes to express his love and gratitude to his beloved families; for their understanding & endless love, throughout the duration of his studies.

For this achievement, I give back all the glory and praises to the omnipotent **Father Almighty.**

Rufo I. Marasigan Jr.

Table of Contents

…

List of Tables

List of Figures

Abstract

Abstract – The abstract gives the reader an overview of the study, based on information from the other sections of the report. The information given in the abstract is usually the basis of many readers as to whether they will read the entire report or not. The abstract is one paragraph of about 100-200 words, single-spaced. The typical information elements included in an abstract are as follows: (1) Some background or general information on the study; (2) The main topic (or purpose) of the study and its scope; (3) Some information on how the study was conducted (or the methodology used in the study); (4) The most important findings of the study; and (5) A statement of conclusion (justified based on the data presented).

Keywords: *keyword1, keyword2, keyword3, keyword4, keyword5*

Introduction

Since smartphone computing power continues to improve and mobile applications (apps) continue to dominate digital engagement, apps have become new frontiers for propelling field experiment methodology (Zhang, J.,et al.(2018)). Farming is becoming a more time-sensitive and information-intensive industry. After harvesting a crop or product, the farmer has limited time to locate the nearest market, check current stock levels, and determine which market will be more profitable for his crop. It takes a long time to analyze the market situation. Farmers in traditional marketing schemes had few options for selling their crops/products, making it challenging to maximize crop profit (M. Bhende. et al.). Farmers work tirelessly throughout the season to increase their harvests; they face numerous challenges, such as natural disasters, unpredictable rain, and the unavailability of water resources. Farmers require more resources to address the issues. If they can resolve all the problems and produce their product, the second major issue is that they will not receive the market price. They need help with money, and most of the time, they will not receive the price for their product as per their expectations due to market strategy (Shakeel-Ul-Rehman, et al). Since the intermediary entities, the farmer needs to receive fair prices for their product. The intermediary entities take much of the farmer's income while doing nothing. A farmer is unfamiliar with the various sales that are taking place in various locations or the best deals for his product.

Mendoza et al. (2021) studied the influence of Farmer's Mobile Marketplace applications on smallholder farmers' income in the Philippines. According to the study, using mobile applications boosted farmers' market access and allowed them to sell their products at greater prices, resulting in a rise in their income. The study also emphasized the necessity of trust and communication between farmers and Costumerss in assuring the mobile marketplace application's success.Cruz et al. (2020) conducted a study on the factors influencing the adoption of Farmer's Mobile Marketplace applications among farmers in the Philippines, identifying perceived usefulness, perceived ease of use, and perceived credibility as significant factors in the adoption of mobile applications by farmers. The study emphasized the importance of user-friendly interfaces, reliable information, and transparent pricing mechanisms in promoting the adoption of mobile applications by farmers. Finally, the introduction of Farmer's Mobile Marketplace applications in the Philippines has created a viable route for smallholder farmers to gain access to markets and boost their revenue. Trust, communication, user-friendliness, and reliable information are all important

components in the success of these applications. Mobile technology has the ability to enhance sustainable agriculture practices in the Philippines as well. Future research may look on the influence of Farmer's Mobile Marketplace applications on the agricultural sector as a whole, as well as potential for further development and improvement.

The following questions determine the possible setbacks in developing a mobile app to connect farmers to their customers and suppliers. How can this app help the digital divide between local farmers and Customers? How can this app make farmer products available to local markets? How do buyers search for and identify the farmers they want to connect with using the mobile application? This platform can encourage farmers to get into technology and e-commerce to reach a larger number of Customers with the help of the application. Due to the advancements in technology, it will be a disadvantage if they could not utilize the use of online marketing applications. Through the app, farmers can advertise and sell their goods online, which market owners or suppliers can buy and then sell to the local market to remain available for urgent purchases of Customers. This application can help promote economic sustainability and increase the income of farmers in the Philippines by facilitating direct market connections and reducing their reliance on intermediaries and middlemen.

This study will help connect farmers, suppliers, and Customers for them to purchase goods by using modern technologies and online shops. The app aims to make purchasing goods from farmers easier, faster, and more reliable. Moreover, it is chosen to distribute goods that come from farmers using mobile applications. The app will be used primarily by farmers and market owners or suppliers.

The Farmer's Mobile Marketplace application's goal is to provide a platform for farmers to connect with customers directly and sell their products. Farmers will be able to use the app to list their products, manage their inventory, set prices, and contact potential Customers. Customers can use the program to browse available products, place orders, and make payments. Product Restrictions: The application will mostly sell fresh products such as fruits, vegetables, and grains. It will exclude cattle and processed foods. Marketing Restrictions: The application will not conduct direct marketing campaigns or advertising for any of the products listed. Technical Restrictions: To

utilize the functionality, the application will require an internet connection and a suitable device. The program may not work properly on devices that have outdated software or hardware.

This mobile app will help market owners and suppliers in purchasing fresh goods from the farmers at a more reasonable price. Moreover, this app will be beneficial to The farmers since it can help them sell their goods in a fast and easy way. The mobile application will show the goods in stock, which Customers can buy straight from them. Prices are also reasonable since the goods come from the farmer or farmers themselves, without the intermediaries and middlemen; thus, their wage increases. Customers can now buy goods that have a reasonable price since the markets have bought the products at a much lower price since there will be no intermediaries and middlemen to hike up the price. Lastly, market owners because this platform will ease the connection between farmers and the market since this will serve as a direct linkage between them and the farmers using this application.

Software Needs Validations

Shakeel-Ul-Rehman stated that there is a need to change the agricultural business's marketing strategy, it is time to implement technology for buying and selling agricultural products (Shakeel-Ul-Rehman). Shakeel also indicated various problems and challenges for agriculture business marketing: lack of market knowledge and lack of agricultural knowledge. The study of (Abdul Razaque, et al) indicated that the mobile phone is playing a significant role in agriculture development, the use of mobile phones in developing nations is a crucial part of the upgrade of farming business to farming. Lately, communication via mobile phones has been regarded as critical in improving ranchers' access to comprehend rural market circumstances better. In developing countries, data correspondence innovations are expanding for the advancement of various individuals, for example, educators, specialists, and farmers. Ranchers are one of the large groups in developing countries that need offices nearby to increase their items and pay. Cell phones are becoming more popular among farmers but are also widely available among businesses, customers, and ranchers (Surabhi Mittal, Gaurav Tripati). Surabhi Mittal and Gaurav Tripati state that cell phones have made it possible for ranchers to obtain information about marketing and climate. Through this critical innovation, they can easily stay in touch with advertising personnel and the offers they develop at reasonable prices. Using mobile phones also

keeps them aware of climate forecasts for farming information applications like compost and pesticides, which may be influenced by unanticipated disasters, as reported by the meteorological office. This device has provided a new route and method for ranchers to communicate openly and share recent advancements. The studies revealed that cell phones saved agriculturists' vitality and time, ultimately increasing their pay.

The agricultural sector is a significant pillar of the economy, and as a business sector, it meets the world's food needs. In any case, the advancement of mobile apps in agriculture is limited when compared to other business sectors (S. Karetsos, M. Ntaliani, et al). Adoption of mobile-based agricultural extension services (AES) can be used as a tool for overall advancement in provincial farmland (Verma, P., et al.).To capitalize on the rapid growth of mobile phones in developing countries, businesses, government offices, and non-governmental organizations are increasingly focusing on the delivery of services via mobile phones in areas such as health, education, and agriculture (Baumüller, H.) Agricultural farmers have frequently criticized forecast developers for a lack of forecast skill and usability, as well as a lack of understanding of user needs. (Freebairn, J. W., Klemm, T., & McPherson, R. A.).

The paper by Manisha Bhende and her team planned and implemented a system for farmers that could be operated via their mobile phones. The system was created for a victimization service that is destined for design (SOA) by combining spatial knowledge and mental objects. The mental object is preserved within the framework of ontologies. The system bridges the communication gap between farmers and agricultural consultants. The system provides platforms such as an Android app and a website app through which farmers can sell their crop products to different layers of the marketing chain (market, merchant, or end-user) with multiple options. The platform will assist farmers in locating nearby markets, current stock details, and demand for a specific product in less time and with less effort.

The agricultural industry has, however, been plagued by enduring problems that have led to poor farm incomes, low rural employment, a lack of food security, and weak agricultural competitiveness. Also, two main issues present in the modern era occurred after examining the emerging situations of the local farmers, claims the article The Plight of Filipino Farmers. The difficulty in selling their goods at the neighborhood marketplaces comes first, followed by the low purchase cost.

Mendoza et al. (2021) researched the impact of Farmer's Mobile Marketplace applications on the revenue of smallholder farmers in the Philippines. The study found that employing mobile applications increased farmers' market access and allowed them to sell their products at higher prices, resulting in an increase in their revenue. Cruz et al. (2020) conducted a study on the factors influencing the adoption of Farmer's Mobile Marketplace applications among farmers in the Philippines, identifying perceived usefulness, perceived ease of use, and perceived credibility as significant factors in the adoption of mobile applications by farmers.

Several studies have confirmed that due to information transfer to farmers via services, many farmer participants are willing and able to diversify their farm practices (Mekbib et al., 2019). Farmers' attitudes and productivity have improved due to mobile phone extension programs in Sub-Saharan Africa. The most remote, subsistence smallholder farmers farthest away from markets experience the most significant improvements in access to information among the various participating groups' using services (Mekbib et al., 2019). The study highlighted the necessity of user-friendly interfaces, trustworthy information, and transparent pricing methods in encouraging farmers to utilize mobile applications. In addition, Panganiban et al. (2020) investigated the role of mobile technology in encouraging sustainable agriculture practices in the Philippines. According to the study, mobile applications have the ability to disseminate knowledge on sustainable agricultural techniques, promote access to organic inputs, and facilitate direct marketing between farmers and Customers.

Software Requirement Specifications

A. External Interface Requirements

1. User Interfaces

The logical characteristics of each interface between the software product and the users in our Android application are designed to be intuitive, user-friendly, and familiar to users of other Android applications. We follow standard GUI (Graphical User Interface) principles and design patterns, including utilizing common Android UI components such as ImageView, TextView, LinearLayout, and NestedScrollView.

A. ***ImageView****: We use ImageView to display images throughout the application. This includes product images, profile pictures, and other visual content that enhances the user*

experience. Images are displayed in a visually appealing and responsive manner, ensuring they are properly sized and positioned on different screen sizes.

B. ***TextView****: TextView is used for displaying text content in the application, including product descriptions, user information, and error messages. We adhere to Android typography guidelines, ensuring that text is legible, properly formatted, and visually appealing. We also provide appropriate text styles and sizes for different types of content, such as headings, body text, and labels.*

C. ***LinearLayout****: We utilize LinearLayouts to automatically adjust the content of the application to fit different screen sizes and orientations. This ensures that the UI remains responsive and visually appealing, regardless of the device or screen size used by the user. We follow Android layout constraints and guidelines, such as using weights and gravity to properly align and distribute UI components within LinearLayouts.*

D. ***NestedScrollView:*** *The NestedScrollView is used to enable scrolling of the application content without losing visibility of other important UI elements, such as the navigation bar. This allows users to easily navigate through the application, even if the content exceeds the screen height.*

The GUI standards and design guidelines we follow in our Android application are based on Material Design, which is a widely accepted design language for Android applications developed by Google. This includes using standard buttons and icons provided by Material Design, following color and typography guidelines, and maintaining consistent spacing and layout throughout the application.

In terms of standard buttons and functions, our navigation bar is a consistent element that appears on every screen of the application. It includes buttons for "Home", "Add Product" for farmers, "Buy Products" for buyers, and a "Profile" page for viewing user details. This ensures easy access to different sections of the application and provides a consistent navigation experience for users.

Error message display standards are implemented to provide clear and meaningful error messages to users in case of any issues or errors. This includes using concise and descriptive error messages that clearly indicate the nature of the error and provide suggestions for resolving it. We also provide appropriate error message styles and positioning to ensure they are easily noticeable and do not disrupt the overall user experience.

Detailed specifications of the user interface design, including sample screen images, GUI standards, layout constraints, error message and display standards, are documented in a separate user interface specification. Our focus is on creating a user-friendly and intuitive interface that provides a seamless experience for our Android application users, while adhering to industry standards and best practices for user interface design on the Android platform.

2. Hardware Interfaces

The interfaces between the software product, "Palengk-e," and the hardware components of the system are critical for ensuring seamless communication and efficient operation between the software and the hardware, as well as providing a smooth user experience. Currently, "Palengk-e" is designed to support Android devices, and the interfaces are tailored to cater to the specific requirements of the Android platform.

The logical and physical characteristics of the interfaces are designed to facilitate real-time data and control interactions between the software and the hardware components. For instance, the "Palengk-e" application allows farmers (sellers) and local vendors (buyers) to exchange information, such as product details, pricing, quantity, and availability, in real-time. The interfaces are designed to ensure accurate and reliable data transmission, synchronization, and updates between the software and the hardware.

Communication protocols are also an essential aspect of interfaces. "Palengk-e" utilizes modern communication protocols, such as HTTPS, for secure data transmission over the internet, ensuring data integrity and confidentiality. Additionally, APIs (Application Programming Interfaces) are used to facilitate seamless integration with other systems or platforms, allowing for smooth communication between "Palengk-e" and other relevant applications or services.

As "Palengk-e" is still in development, the interfaces are designed to be scalable and adaptable to accommodate future changes and updates. The application's interfaces are designed with a user-friendly approach, ensuring ease of use and intuitive navigation for both farmers and local vendors using Android devices. The interfaces are designed to provide a seamless and efficient user experience, enabling users to easily browse and select bulk orders of crops, manage their inventory, and track their transactions within the application.

In conclusion, the logical and physical characteristics of the interfaces between the software product, "Palengk-e," and the hardware components of the system are carefully designed to support Android devices, facilitate real-time data and control interactions, utilize modern communication protocols, and provide a user-friendly experience.

3. Software Interfaces

The "Palengk-e" mobile application interacts with various software components to ensure seamless operation and functionality. These components include:

A. ***Node.js:*** *"Palengk-e" utilizes Node.js for building the RESTful API. Node.js is a popular JavaScript runtime environment that enables server-side JavaScript execution, allowing for efficient and scalable server-side development.*

B. ***MySQL and Apache (XAMPP):*** *"Palengk-e" uses MySQL, a widely used relational database management system, along with Apache in the XAMPP stack for the backend database operations. XAMPP provides a local development environment for hosting the database server and web server, facilitating efficient database operations.*

The data stored in the database, such as user details, can be accessed and retrieved by the app when needed, for example, when a user wants to view their profile. The communication between the app and the database is facilitated through the RESTful API, which serves as an intermediary for processing data requests and responses. Detailed application programming interface (API) protocols are documented to ensure seamless communication and data sharing between different software components.

In conclusion, "Palengk-e" interacts with various software components, including Android Studio, Node.js, MySQL, Apache (XAMPP), and Postman, to enable the registration process, store and retrieve data in the database, and facilitate communication between different software components. Detailed API protocols and data-sharing mechanisms are documented to ensure smooth integration and functionality of the application.

4. Communications Interfaces

The communications functions associated with our software product include email, web browser, network server communications protocols, and electronic forms. These functions are crucial for enabling seamless communication between the software product and external systems or users.

To ensure the security of communication, we have implemented JSON Web Tokens (JWT) as the authentication and authorization mechanism for transmitting data between our application and the database. JWT has become the industry standard for securely transmitting sensitive information in modern web applications that utilize APIs.

One of the key advantages of JWT is its capability to offer a secure method of transmitting data using digital signatures. This ensures the integrity and authenticity of the data being transmitted, eliminating the risk of data tampering or unauthorized access. This level of security establishes trust between the communicating parties and safeguards sensitive information.

In addition to its security features, JWT also implements a stateless authentication mechanism. This means that servers can operate independently without the need to store session state for individual users. This eliminates the need for server-side storage of user session data, allowing for easy scalability and distribution of workloads across multiple servers, without performance degradation or infrastructure constraints.

Furthermore, JWT is cross-platform compatible, making it seamless to use across different programming languages and platforms. This allows developers to design APIs that can be consumed by various clients without worrying about the underlying infrastructure or software stack. This flexibility enables interoperability and smooth communication between different systems and platforms.

Another benefit of JWT is its compactness, which plays a crucial role in improving the overall performance of web applications. Being in a portable format, it reduces latency and enhances response time, resulting in a seamless user experience. This efficient data transmission over the network contributes to the overall performance and responsiveness of our software product.

In terms of communication standards, we are utilizing common protocols such as FTP (File Transfer Protocol) and HTTP (Hypertext Transfer Protocol) for data transfer between systems. These widely used standards ensure compatibility and interoperability with other systems and applications.

To ensure data security during communication, we have implemented encryption mechanisms, such as SSL (Secure Sockets Layer) or TLS (Transport Layer Security), to protect data from unauthorized access or interception. This ensures that data transmitted between the software product and external systems or users is encrypted and secured.

Additionally, we have defined data transfer rates and synchronization mechanisms to ensure efficient and synchronized communication between different components of our software product. These specifications are documented in the communication requirements section of our software documentation, ensuring that all communication processes are standardized and adhered to.

In conclusion, the use of JWT for authentication and authorization, along with standard communication protocols, encryption mechanisms, data transfer rates, and synchronization mechanisms, ensures secure, efficient, and reliable communication between our software product and external systems or users. These communication requirements are documented in detail in our software documentation to ensure adherence to industry standards and best practices.(Copyright © 1999 by Karl E. Wiegers)

B. Non-Functional Requirements

1. Performance Requirement

Performance requirements for our Android application are crucial to ensure optimal performance under various circumstances. These requirements are intended to guide the developers in understanding the intent and making suitable design choices, considering that the application has been tested on only one emulator and may not be applicable to all screen sizes.

***Responsiveness**: The application should provide a responsive user experience with minimal latency or delays in user interactions, such as tapping buttons, scrolling, and navigating between screens. The application should be designed to respond promptly to user actions, ensuring smooth and seamless interactions without noticeable lags or delays.*
Rationale: Responsiveness is important to provide a positive user experience, as users expect applications to be quick and responsive to their inputs. A laggy or unresponsive

application can result in frustration, decreased user satisfaction, and potential abandonment of the application.

Load Time: *The application should load quickly upon launch, minimizing waiting times for the user. The initial load time of the application should be optimized to ensure that users can access the application's features and functionalities as quickly as possible, without experiencing prolonged loading times.*
Rationale: Fast load times are essential to provide a smooth onboarding experience for users. If the application takes too long to load, users may lose interest and abandon the application.

Resource Utilization: *The application should be optimized in terms of resource utilization, including CPU, memory, and battery usage. It should not consume excessive system resources, as this can result in degraded performance, reduced battery life, and potential crashes. Efficient resource utilization is crucial in ensuring smooth and efficient operation of the application, considering that it is designed to run on most Android devices.*
Rationale: Optimized resource utilization is important to ensure that the application does not cause unnecessary strain on the device's resources, resulting in efficient performance and extended battery life. Overconsumption of resources can lead to slow performance, lags, and crashes, which can negatively impact the user experience.

Compatibility: *The application should be compatible with a wide range of Android devices, including different screen sizes, resolutions, and hardware configurations. It should be designed to adapt to varying device specifications to ensure consistent performance across different devices, considering that it has been tested on a limited emulator. Compatibility testing should be conducted to identify and address any potential issues related to device variations, ensuring that the application performs optimally on all supported devices.*

Rationale: *With the diverse range of Android devices available in the market, ensuring compatibility is crucial to ensure that the application can cater to a wide user base. Inconsistent performance across different devices can result in a fragmented user experience and potential loss of users.*

Real-time System Timing Relationships: *If the application includes real-time system functionalities, such as sensor data processing or time-critical operations, specific timing relationships should be defined and adhered to. For example, if the application relies on real-time data updates, the timing of data retrieval, processing, and presentation should be carefully managed to ensure accurate and timely results.*

Rationale: *Real-time system functionalities require precise timing relationships to ensure that the application can provide accurate and up-to-date information. Deviation from the timing requirements can result in inaccurate data or missed deadlines, which can impact the overall functionality and usability of the application.*

By specifying these performance requirements, the developers will have a clear understanding of the expectations and design choices needed for the application. These requirements are important to ensure that the application performs optimally, provides a positive user experience, and meets the needs of the target audience. Regular performance testing and optimization should be conducted throughout the development process to ensure that the application meets these requirements under different circumstances and devices.
(Copyright © 1999 by Karl E. Wiegers)

2. Safety Requirement

One potential performance requirement for the product is the usability of the User Interface (UI) in terms of color accessibility. As the current UI uses bright colors and lacks alternative options, it may cause discomfort or harm to some users. To mitigate this risk, the development team should conduct accessibility testing and consider incorporating features such as dark mode or alternative color options that are easier on the eyes, while still maintaining the overall theme of the application.

Additionally, compliance with relevant external policies or regulations, such as data privacy regulations or industry-specific guidelines, should be ensured. Safeguards, such as encryption of user data and secure authentication, should be implemented to protect user information. If the product requires safety certifications, the necessary design, development, and testing practices should be in place to obtain and maintain such certifications.

By considering these performance requirements related to potential loss, damage, or harm, and implementing appropriate safeguards or actions, the development team can ensure the safe use of the product, while complying with relevant policies, regulations, and safety certifications. Regular review and updates of these requirements should be conducted to ensure ongoing compliance with safety standards throughout the product's lifecycle.
(Copyright © 1999 by Karl E. Wiegers)

3. Security Requirement

Regarding data protection, it is essential to implement appropriate security measures to safeguard the data used or created by the product. This includes securing the MySQL database that stores registered data. Currently, using xampp and localhost to save data may pose security risks, as there are no additional security features such as passwords, limited session, or encryption in place. It is imperative to implement robust security measures, such as strong passwords, encrypted connections, and limited access to the database, to protect against unauthorized access or data breaches.

By addressing these security and privacy requirements, and implementing appropriate measures to protect data and authenticate users, the product can ensure the

confidentiality, integrity, and availability of data, while complying with relevant policies, regulations, and certifications. Regular monitoring and updates should be conducted to maintain the security and privacy of the product throughout its lifecycle.

4. Software Quality Attributes

Future feature additions, such as Push Notifications, Email Confirmation, and One-Time Pin integration, can be easily incorporated into our development process.

A. ***Maintainability:*** *The use of Android Studio, a widely used and popular Integrated Development Environment (IDE) for Android app development, can contribute to the maintainability of the product. Android Studio provides various tools and features that facilitate code organization, documentation, and version control, making it easier for developers to maintain and update the codebase in the future as new features or requirements arise. The statement indicates that future feature additions can be easily incorporated into the development process, which suggests that the product is being designed with maintainability in mind.*

B. ***Adaptability:*** *It implies that the product is being developed in a way that allows for easy integration of additional features in the future, such as push notifications, email confirmation, and one-time pin. This indicates that the product is being designed with adaptability in mind, allowing for changes or updates to be incorporated without major disruptions to the existing codebase. This aligns with the quality characteristic of adaptability, which involves designing a product to easily adapt to changing requirements or environments.*

C. ***Reusability:*** *It also suggests that the codebase being developed can potentially be reused for adding more features in the future, such as push notifications, email confirmation, and one-time pin. This implies that the code is being designed and coded in a way that promotes reusability, reducing redundancy and improving development efficiency. This aligns with the quality characteristic of reusability, which involves designing and coding a product in a way that promotes code reuse in different parts of the product or in future projects.*

Considering these quality characteristics, including maintainability, adaptability, and reusability, during the development process can contribute to the overall quality of the product. Ensuring that the product is designed and developed in a way that allows for easy incorporation of future feature additions can enhance its maintainability, adaptability, and reusability, resulting in a high-quality software product.(Copyright © 1999 by Karl E. Wiegers)

Software Design Evaluation

A. Abstract

This will contain the abstract about the software design evaluation. Refer to the abstract format in the preceding pages.

C. Introduction

This contains the introduction for the software design evaluation. It is written similarly to the general introduction in the documentation. In this part, the introduction should focus on the discussions on the gaps and possibilities of the algorithms to be used in the development of the system.

One of the most significant shortcomings of search engines is their inability to understand natural language queries. Most search engines use keyword matching to deliver results, which might result in inaccurate or irrelevant results if the search query is poorly formulated or the terms are ambiguous. Furthermore, searching engines may struggle with sophisticated queries that necessitate contextual knowledge, such as questions involving several concepts or allusions to specific events or people. Despite these limitations, there are numerous opportunities to improve and enhance search algorithms. Advances in machine learning and natural language processing, for example, have resulted in the development of more complex search engines capable of understanding the meaning and context of search queries. Furthermore, new technologies like as semantic search and graph databases provide new methods to organize and access information, potentially alleviating some of the restrictions of classic keyword-based search algorithms.

D. Review of Related Literature

This part of the documentation discusses the different related literature that can be used in the evaluation of the algorithm to be used in the study. It includes discussions of the process involved in the algorithm, its advantages, and disadvantages. It is written with subheadings.

Efficiency: Search algorithms may retrieve information rapidly and efficiently in enormous databases, saving time and effort (Hearst). Search algorithms can be tailored to meet specific requirements, such as ranking search results based on relevance or filtering results based on specified criteria (Manning et al.).

Scalability: Because search algorithms can handle very big datasets, they are suitable for applications such as web search engines or scientific data processing (Dean and Ghemawat,).

Accessibility: Search algorithms can be accessed via a variety of interfaces, such as web-based search engines, mobile applications, and desktop software, making them widely accessible to users with varying degrees of technical competence (Manning et al.,).

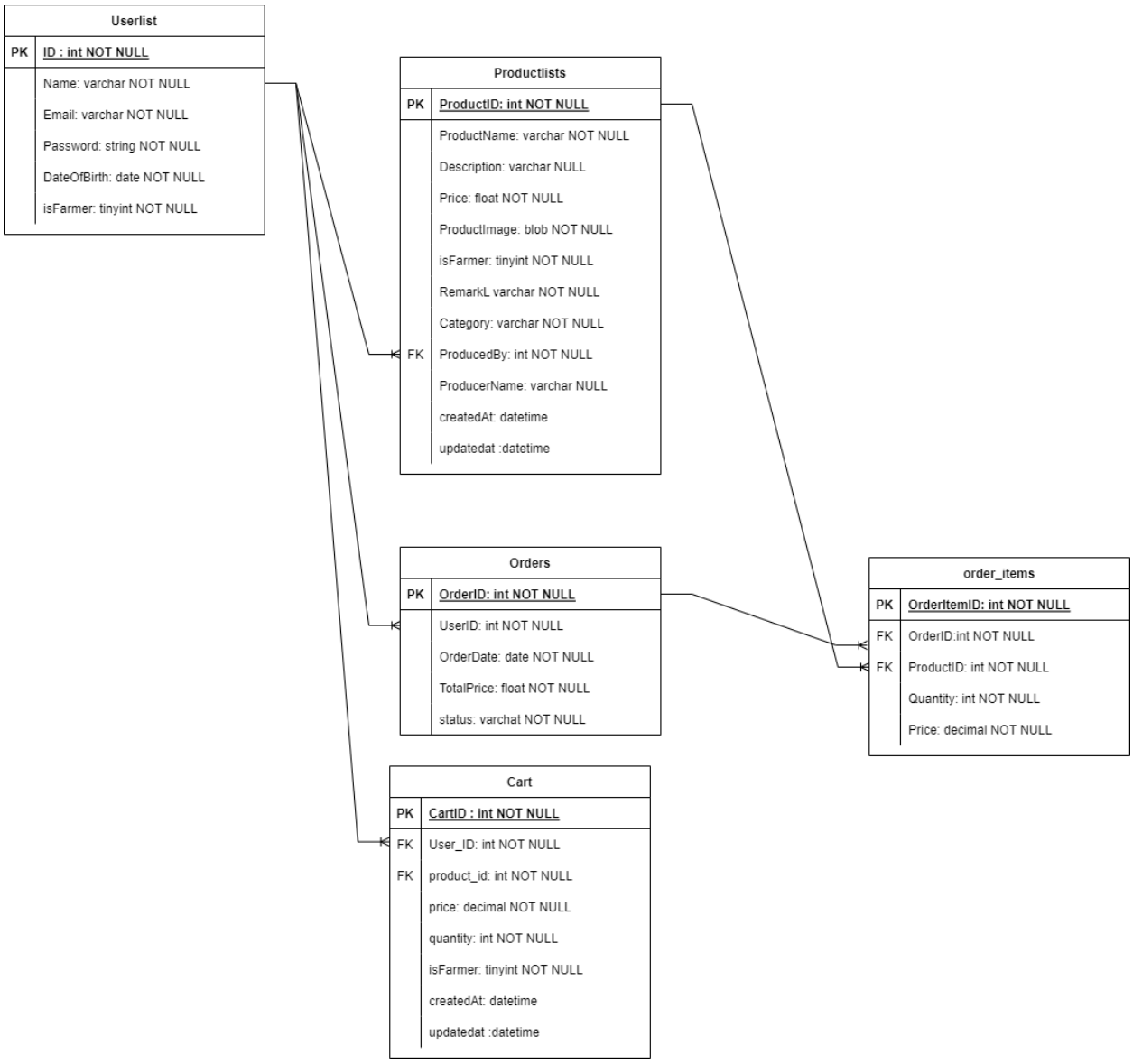
Keyword reliance: The quality of search results can be influenced by the terms used in the search query, which might result in irrelevant or incomplete results (Hearst, 2009).

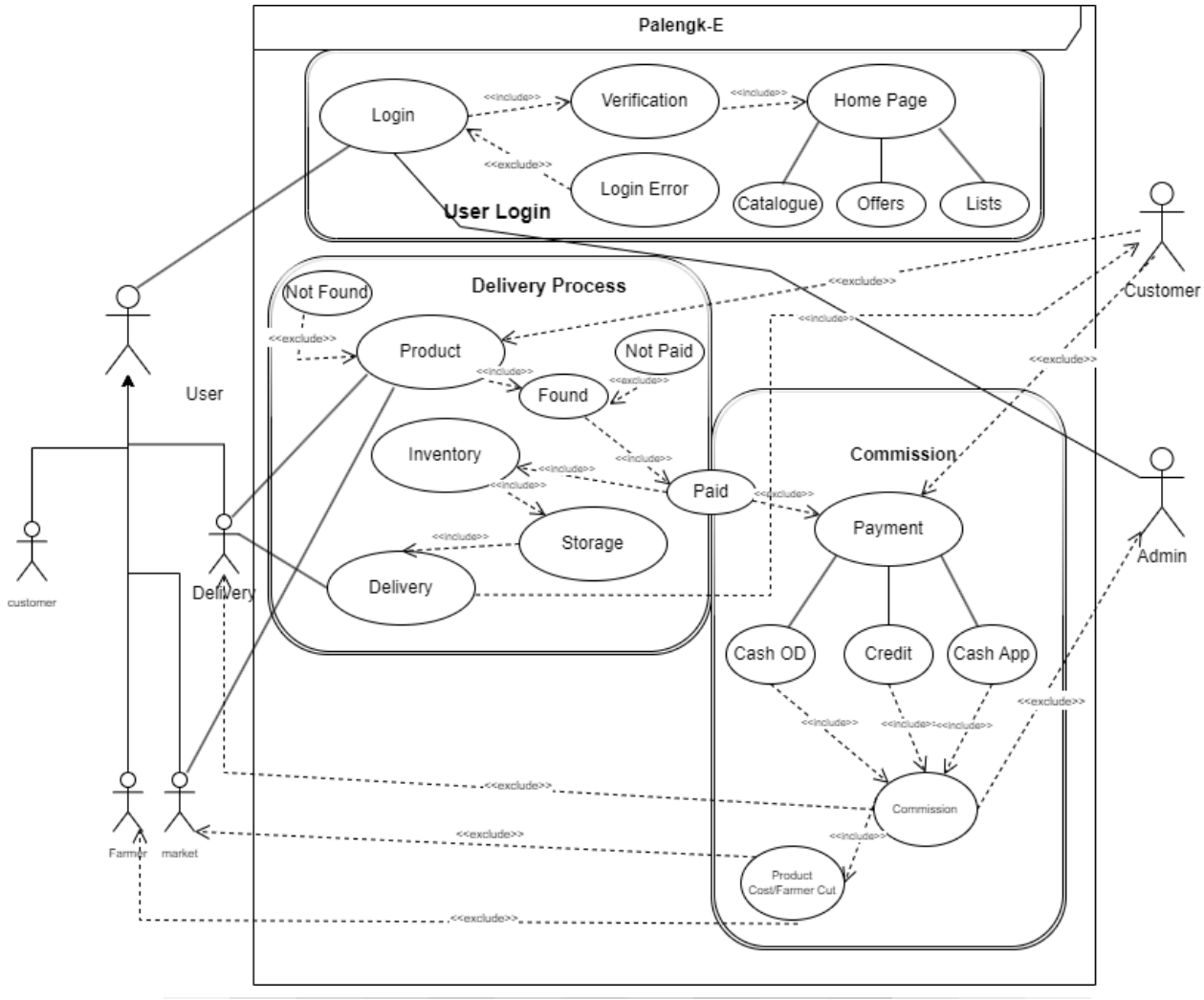
Computational constraints: When working with particularly big or complicated datasets, reliance on computational resources might make search algorithms slower and less efficient, limiting their accessibility to users with limited computational resources or experience (Dean and Ghemawat, 2008).

Limitations of natural language processing: Search engines may be unable to detect and interpret natural language inquiries, limiting their usefulness in specific circumstances (Manning et al., 2008).

E. Methodology

This part contains the discussion of the overview of the flow of the algorithm as general in reference to the objectives. This includes the algorithm process diagram (DFD Level1), and individual process flowchart and its discussions.


Userlist
PK
ID : int NOT NULL
Name: varchar NOT NULL
Email: varchar NOT NULL
Password: string NOT NULL
DateOfBirth: date NOT NULL
isFarmer: tinyint NOT NULL
Productlists
PK
ProductID: int NOT NULL
ProductName: varchar NOT NULL
Description: varchar NULL
Price: float NOT NULL
ProductImage: blob NOT NULL
isFarmer: tinyint NOT NULL
RemarkL varchar NOT NULL
Category: varchar NOT NULL
FK
ProducedBy: int NOT NULL
ProducerName: varchar NULL
createdAt: datetime
updatedat :datetime
Orders
PK
OrderID: int NOT NULL
UserID: int NOT NULL
OrderDate: date NOT NULL
TotalPrice: float NOT NULL
status: varchat NOT NULL
order_items
PK
OrderItemID: int NOT NULL
FK
OrderID:int NOT NULL
FK
ProductID: int NOT NULL
Quantity: int NOT NULL
Price: decimal NOT NULL
Cart
PK
CartID : int NOT NULL
FK
User_ID: int NOT NULL
FK
product_id: int NOT NULL
price: decimal NOT NULL
quantity: int NOT NULL
isFarmer: tinyint NOT NULL
createdAt: datetime
updatedat :datetime



F. Algorithm Testing and Evaluations

This part discusses the simulations, verifications, and testing of 3 algorithms identified or different hyperspectral parameters needed in the implementations of the system later on in Software Engineering.

G. Results and Discussions

This part discusses the results of the ***Algorithm Testing and Evaluations.***

H. Conclusions and Recommendations

This part discusses the conclusion of the testing made and recommendations for the improvement of the algorithm.

Software Design Descriptions

A. System Architecture

Describe and illustrate the System Architecture and topology.

B. Constraints

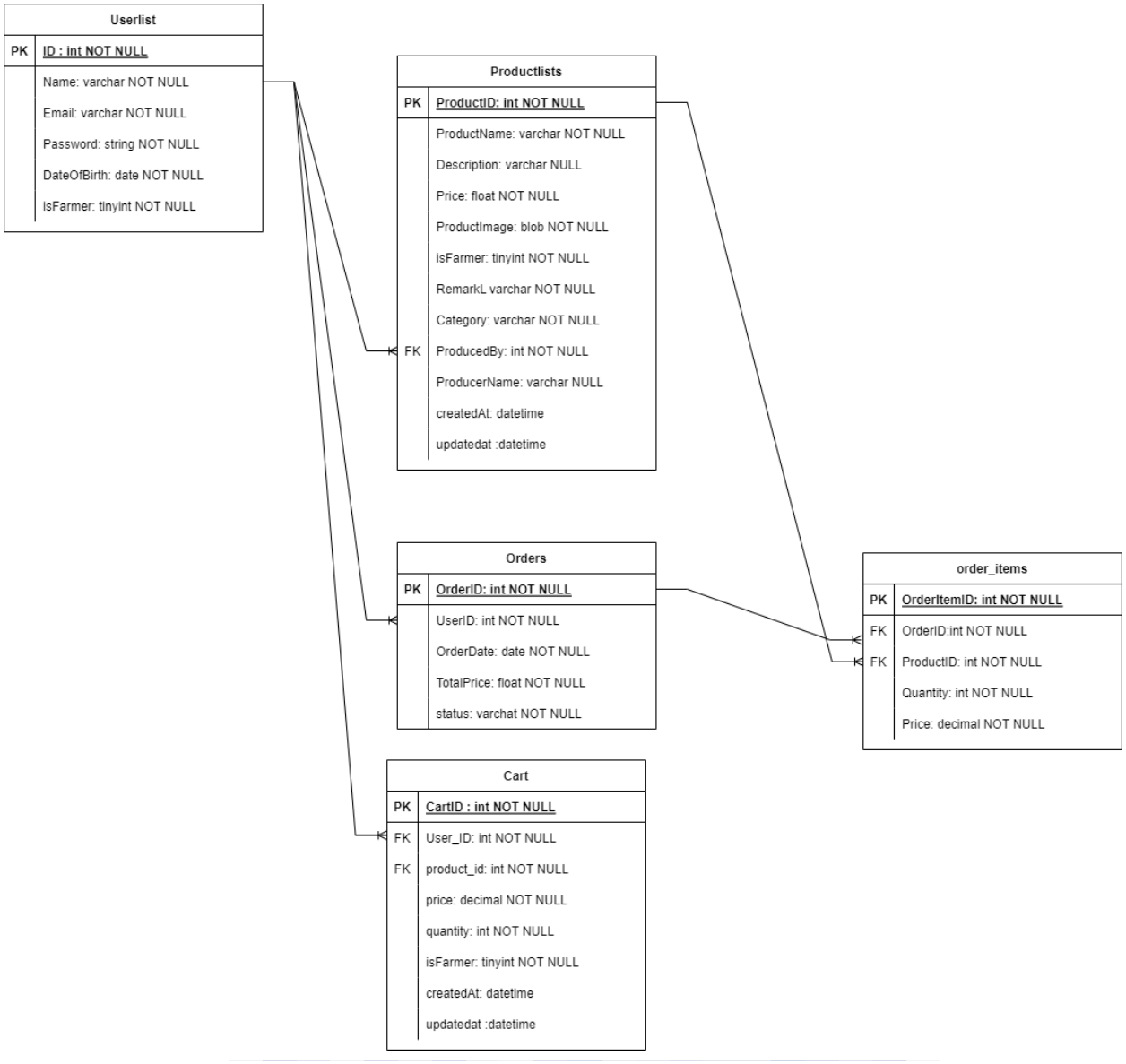
Cost	Each developer's fees and resources would be included in the cost.
Time	After facing delays, the time given for the project is ideal and short.
Resources	Only four developers are working on the project. Developers must have access to programming and word processing applications, as well as a laptop or personal computer running Windows.
Quality	The majority of the time, the systems would be operational. When an issue arises, there will be monitoring and countermeasures. The system is secure and resistant to intruders.
Risks	Risks will be reduced by doing regular backups of the website and database. Security and preventive measures will be implemented as well.

Table 5. 1 Constraints

C. Functional View

*Describ*e and illustrates the *Functional View of the System*

D. Data Flow Diagram (DFD)



E. System Flow Chart

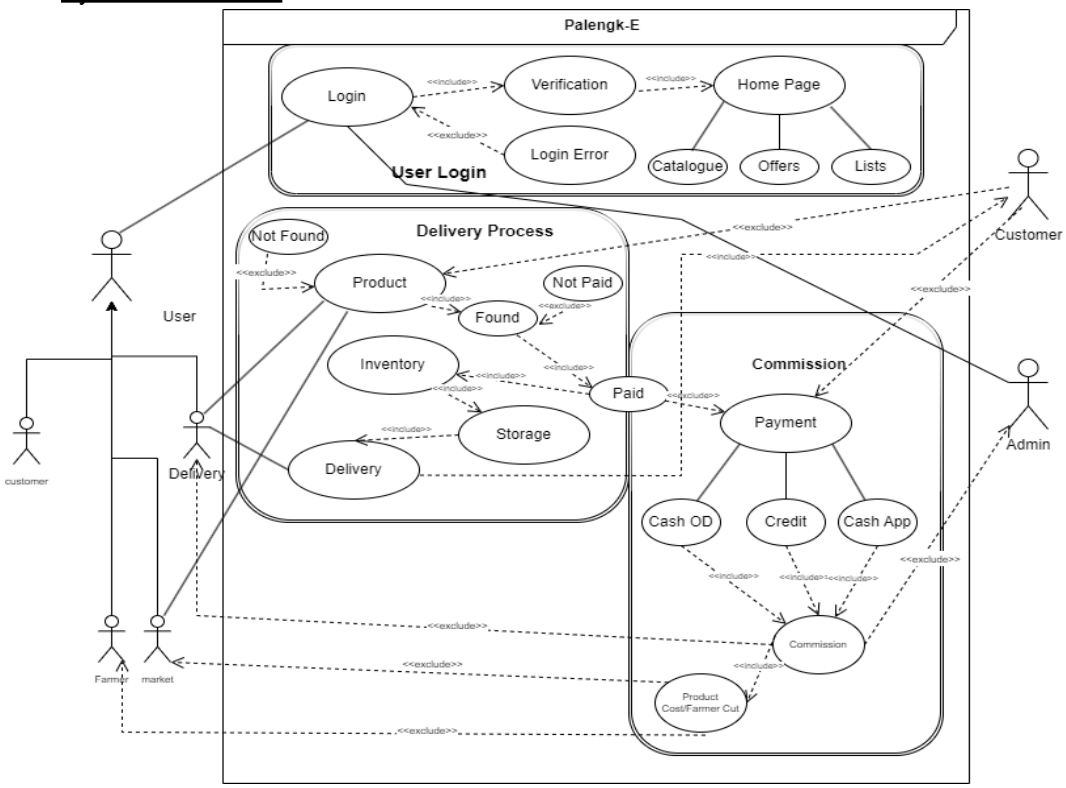


Figure 5. 2 Entity Relation Diagram

F. Data Dictionary

PK/FK/UK	Field Name	Data Type	Length	Input Format	Output Format	Default Value	Continuous / Discrete	Base or Derived
YES	ID	Int	255	9(255)	9(255)	User ID	Continuous	Base
NO	Name	varchar	50	X(50)	X(50)	User Name	Continuous	Base
YES	Email	varchar	100	X(100)	X(100)	User Email	Continuous	Base
NO	Password	text				Password	Continuous	Base
NO	DateOfBIrth	date				Date of Birth	Continuous	Base

NO	isFarmer	tinyint	1	9(1)	9(1)	User Identifier	Discrete	Base
NO	Address	varchar	150	X(150)	X(150)	Address	Continuous	Base

Table 5. 2 User List Table

PK/FK/UK	Field Name	Data Type	Length	Input Format	Output Format	Default Value	Continuous/ Discrete	Base or Derived
YES	ProductID	Int	11	9(11)	9(11)	Product Id	Continuous	Base
NO	ProductName	Varchar	50	X(50)	X(50)	Product Name	Continuous	Base
NO	Description	Varchar	50	X(50)	X(50)	Product Description	Continuous	Base
NO	Price	Varchar	25	X(25)	X(25)	Product Price	Continuous	Derived
NO	ProductImage	Varchar	25	X(25)	X(25)	Product Image	Continuous	Base
NO	Remark	varchar	255	X(255)	X(255)	Customer Middle Initial	Continuous	Base
NO	Category	varchar	255	X(255)	X(255)	Product Category	Continuous	Base
YES	ProduceBy	int	11	9(11)	9(11)	Producer Identifier	Continuous	Base
NO	ProducerName	varchar	255	X(255)	X(255)	Producer Name	Continuous	Base
NO	Location	varchar	100	X(100)	X(100)	Product location	Discrete	Base
NO	createdAt	datetime				Customer Display Picture	Continuous	Base
NO	updatedAt	datetime				Address Details	Continuous	Base

Table 5. 3 Products Lists Table

PK/FK/UK	Field Name	Data Type	Length	Input Format	Output Format	Default Value	Continuous / Discrete	Base or Derived
YES	RegisterID	Int	255	9(255)	9(255)	Registered ID	Discrete	Base
NO	Name	varchar	50	X(50)	X(50)	User Name	Continuous	Base
YES	Email	varchar	100	X(50)	X(50)	User Email	Continuous	Base
NO	Password	tinytext				User Password	Continuous	Base
NO	DateofBirth	date				User Birthdate	Continuous	Base
NO	Address	varchar	150	X(150)	X(150)	User Address	Discrete	Base
NO	isFarmer	tinyint	4	9(4)	9(4)	User Identifier	Continuous	Base
NO	Status	tinyint	4	9(4)	9(4)	User Status	Discrete	Base

Table 5. 4 Pending Users Table

PK/FK/UK	Field Name	Data Type	Length	Input Format	Output Format	Default Value	Continuous / Discrete	Base or Derived
YES	OrderID	Int	11	9(11)	9(11)	Order ID	Continuous	Base
YES	UserID	Int	11	9(11)	9(11)	User ID	Continuous	Base
NO	OrderDate	date				Date of Order	Continuous	Base
NO	TotalPrice	float				Order Price	Continuous	Derived

NO	Status	varchar	20	X(20)	X(20)	Order Status	Discrete	Base

Table 5. 5 Orders Table

PK/FK/UK	Field Name	Data Type	Length	Input Format	Output Format	Default Value	Continuous / Discrete	Base or Derived
YES	OrderItemID	int	11	9(11)	9(11)	Item ID	Continuous	Base
YES	OrderID	int	11	9(11)	9(11)	Order ID	Continuous	Base
YES	ProductID	unt	11	9(11)	9(11)	Product ID	Discrete	Base
NO	Quantity	int	1	9(1)	9(1)	Order Quantity	Continuous	Derived
NO	Price	decimal	10, 2	9(10,2)	9(10,2)	Order Price	Continuous	Derived

Table 5. 6 Orders Item Table

PK/FK/UK	Field Name	Data Type	Length	Input Format	Output Format	Default Value	Continuous/ Discrete	Base or Derived
YES	CartID	int	11	9(11)	9(11)	Cart ID	Continuous	Base
YES	user_id	int	11	9(11)	9(11)	User Id	Continuous	Base
YES	product_id	int	11	9(11)	9(11)	Product ID	Continuous	Base
NO	product_image	longblob				Product Image	Continuous	Base
NO	price	decimal	10, 2	9(10,2)	9(10,2)	Product Price	Continuous	Derived
NO	quantity	int	11	9(11)	9(11)	Product Quantity	Continuous	Derived

NO	created_at	timestamp				User Identifier	Discrete	Base
NO	updated_at	timestamp				User Identifier	Disrete	Base

Table 5. 7 Cart Table

Software Testing's and Evaluations

A. Test Approach

This part includes the discussions of the testing approach to evaluate if the objectives and the system attributes are met. It measures both qualitative and quantitative approaches for functional and non-functional requirements of system

B. Test Plan

- User Management:

- Add User
- Delete User
- Pending User
- Registered User
- New Farmer
- Current Farmer

- Login Page:

- Login Page with correct and incorrect username and password

Farmers POV:

- My Products

- Delete Product
- Restock Product
- View Products
- View Profile

Customers:

- Buyers POV

- Homepage (View Products)
- My Cart
- My Orders
- Profile Page

- Sign in Page

- Log in user
- Register User
- Login with correct and incorrect credentials
- Login with incomplete fields

- My Cart

- Add quantity
- Subtract Quantity
    - Checkout
- Checkout Product
    - Edit Profile
- Edit user information

Test Cases

Test Case Name	Test Case ID
Login Page	TCA01
Login with correct and incorrect username and password	TCA02
Sign in Page	TCB01
Log in user	TCB02
Register user	TCB03
Login with correct and incorrect credentials	TCB04
Login with incomplete fields	TCB05
Farmers POV	TCC01
View products	TCC02
Delete products	TCC03
Restock products	TCC04
View profile	TCC05
Buyers POV	TCD01
Homepage (View products)	TCD02
My cart	TCD03
My orders	TCD04
Profile page	TCD05
My cart	TCE01

Add quantity	TCE02
Subtract quantity	TCE03
Checkout	TCF01
Checkout product	TCF02
Edit profile	TCG01
Edit user information	TCG02

Table 6. 1 Table For Test Cases

c. Testing

1. Admin's Test

The functionality of each field in the admin's shown in table 6.2:

Test Condition/Scenarios	User's Name	Expected Output	Actual Output	Remarks
LOGIN PAGE				
Logging in		Successful login	Successful login	Passed
Logging out		Successful logout	Successful logout	Passed
LANDING PAGE				
View Homepage		Admins shall be able to view homepage	Admins were able to view homepage	Passed
USER MANAGEMENT				
Add Users		Admins shall be able to add users	Admins were able to add users	Passed
Delete Users		Admins shall be able to delete users	Admins were able to delete users	Passed

Table 6. 2 Admin's Test.

2. Farmer's Test

The functionality of each field in the merchant shown in table 6.3:

Test Condition/Scenarios	User's Name	Expected Output	Actual Output	Remarks
REGISTRATION				
Register to the site		Success registration of the Farmer	Successful registered	Passed
LOGIN PAGE				
Logging in		Successful login	Successfully Login	Passed
Logging out		Successful logout	Successfully Logout	Passed
HOMEPAGE				
View Homepage		Farmers should be able to view homepage	Farmers will be able to view homepage	Passed
Add Products		Farmers should be able to add products	Farmers successfully added products	Passed
Delete Products		Farmers should be able to delete products	Farmers successfully deleted products	Passed
Edit Product Listings		Farmers should be able to edit their product's information	Farmers can update their products	Passed

Table 6. 3 Farmer's Test.

3. Customer's Test

The functionality of each field in the customer shown in table 6.4:

Test Condition/ Scenarios	User's Name	Expected Output	Actual Output	Remarks
REGISTRATION				
Register to the site		Success registration of the customers	Successful registered	Good
LOGIN PAGE				
Logging in		Successful login	Successfully Login	Excellent
Logging out		Successful logout	Successfully Logout	Excellent
HOMEPAGE				
View Homepage		Customers shall be able to view homepage	Customers were able to view home page	Excellent
View Products		Customers shall be able to view products listed	Customers can view products	Excellent
View Farmer's Information		Customers may view Farmer's information	Customers can view Farmer's information	Very Good
Make Orders		Customers shall be able to add or delete orders and check out products	Customers successfully made orders	Good
Add to Cart		Customers shall be able to add to cart products	Customers can add to cart products	Excellent

Table 6. 4 Customer's Test

4. User's Acceptability Test

Below is the questionnaire that will be distributed to 5 respondents.:

Attributes	Description	1 (Poor)	2 (Fair)	3 (Good)	4 (Very Good)
Accessibility	The website is accessible 24 hours a day, seven days a week.				
Reliability	The system follows the defined performance specifications				
Maintainability	The website can perform successful repair within a given time.				
Maintainability	The website can perform successful repair within a given time.				
Security	The website ensured the protection of user's personal information				

Table 6. 5 System Software Attributes

In analyzing the result of the test, the Likert scale is being used. Below shows the scale and its

description. If the series of questions that when combined describe a trait or a performance-use means and standard deviations to describe the scale.

$$WM = (f4x4) + (f3x3) + (f2x2) + (f1x1)/N$$

Where: WM = Weighted mean

N = Total No. of Cases

Likert's scale is the total sum of all Likert's items. Likert items refer to the question or statement on the given questionnaire. The respondent will evaluate each Likert item by giving each item a score based on how the item meets the respondent's standard or satisfaction. The scores will then be combined to determine the weighted mean and its corresponding adjectival rating.

Numerical Scale Rating Scale	**Adjectival Rating**
4	**Excellent**
3.0-3.9	**Good**
2.0-2.9	**Fair**
1.0-1.9	**Poor**

Table 6. 6 Likert Scale.

Testing, Test cases and Results

A. Test Suite A - Website Pages

Test Suite ID	
Test Case ID	
Test Case	
Summary	
Test Procedure	
Expected Result	
Actual Result	
Status	
Created By	
Date of Creation	
Executed By	
Date of Execution	
Test Environment	

Table 6. 7 TCA01 Website Home Page

B. Test Suite B - Login System

Test Suite ID	
Test Case ID	
Test Case Summary	
Prerequisites	
Test Procedure	1.
Expected Result	
Actual Result	
Status	
Created By	
Date of Creation	
Executed By	
Date of Execution	
Test Environment	

Table 6. 8 TCB01 - Login with incomplete credential

Test Suite ID	
Test Case ID	
Test Case Summary	
Prerequisites	
Test Procedure	
Expected Result	
Actual Result	
Status	
Created By	
Date of Creation	
Executed By	
Date of Execution	
Test Environment	

Table 6. 9 TCB02 - Login with correct Username/Password

c. Test Suite C - Registration Page

Test Suite ID	
Test Case ID	
Test Case Summary	
Prerequisites	
Test Procedure	1.
Expected Result	
Actual Result	
Status	
Created By	
Date of Creation	
Executed By	
Date of Execution	
Test Environment	

Table 6. 10 TCC01 - Register/Create a Customer Account

Test Suite ID	
Test Case ID	
Test Case Summary	
Prerequisites	
Test Procedure	1.
Expected Result	
Actual Result	
Status	
Created By	
Date of Creation	
Executed By	
Date of Execution	
Test Environment	

Table 6. 11 TCC02 - View Registered User Account as Admin

Test Suite ID	
Test Case ID	
Test Case Summary	
Prerequisites	
Test Procedure	
Expected Result	
Actual Result	
Status	
Created By	
Date of Creation	
Executed By	
Date of Execution	
Test Environment	

Table 6. 12 TCC03 – Delete Registered User Account as Admin

D. Test Suite D – Merchant Account

Test Suite ID	
Test Case ID	
Test Case Summary	
Prerequisites	
Test Procedure	
Expected Result	
Actual Result	
Status	
Created By	
Date of Creation	
Executed By	
Date of Execution	
Test Environment	

Table 6. 13 TCD01 – Register/Create Merchant Account

Test Suite ID	
Test Case ID	
Test Case Summary	
Prerequisites	
Test Procedure	
Expected Result	
Actual Result	
Status	
Created By	
Date of Creation	
Executed By	
Date of Execution	
Test Environment	

Table 6. 14 TCD02 – View Merchant Account

E. Test Suite E – Merchant Product

Test Suite ID	
Test Case ID	
Test Case Summary	
Prerequisites	
Test Procedure	
Expected Result	
Actual Result	
Status	
Created By	
Date of Creation	
Executed By	
Date of Execution	
Test Environment	

Table 6. 15 TCE01 – Create/Add Product as Merchant

Test Suite ID	
Test Case ID	
Test Case Summary	
Prerequisites	
Test Procedure	
Expected Result	
Actual Result	
Status	
Created By	
Date of Creation	
Executed By	
Date of Execution	
Test Environment	

Table 6. 16 TCE02 – Remove Product as Merchant

F. Test Suite F- View Products

Test Suite ID	
Test Case ID	
Test Case Summary	
Prerequisites	
Test Procedure	
Expected Result	
Actual Result	
Status	
Created By	
Date of Creation	
Executed By	
Date of Execution	
Test Environment	

Table 6. 17 TCF01 – View Products as Customer

Test Suite ID	
Test Case ID	
Test Case Summary	
Prerequisites	
Test Procedure	
Expected Result	
Actual Result	
Status	
Created By	
Date of Creation	
Executed By	
Date of Execution	
Test Environment	

Table 6. 18 TCF02 – Add to Cart Product as Customer

Test Suite ID	
Test Case ID	
Test Case Summary	
Prerequisites	
Test Procedure	
Expected Result	
Actual Result	
Status	
Created By	
Date of Creation	
Executed By	
Date of Execution	
Test Environment	

Table 6. 19 TCF03 – Remove Cart Product as Customer

G. Test Suite G- Check Out Products

Test Suite ID	
Test Case ID	
Test Case Summary	
Prerequisites	
Test Procedure	
Expected Result	
Actual Result	
Status	
Created By	
Date of Creation	
Executed By	
Date of Execution	
Test Environment	

Table 6. 20 TCG01 – Check out Product.

H. Test Suite H- Choose Payment Option

Test Suite ID	
Test Case ID	
Test Case Summary	
Prerequisites	
Test Procedure	1.
Expected Result	
Actual Result	
Status	
Created By	
Date of Creation	
Executed By	
Date of Execution	
Test Environment	

Table 6. 21TCG02 – Choose Payment Option as Customer

Test Suite ID	Test Case ID	Test Case Description	No. of Times Tested	Passed %	Failed %	Required %	Remarks
		Website Home Page					Passed
		Login with Incomplete Credential					Passed
		Login with correct username/password					Passed
		Register/Create a Customer Account					Passed
		View Registered Account as Admin					Passed
		Delete Registered Account as Admin					Passed
		Register/create Merchant Account					Passed
		View Merchant Account					Passed
		Create/add Product as Merchant					Passed
		Remove Product as Merchant					Passed
		View Products as Customer					Passed
		Add to Cart Product as Customer					Passed
		Remove Cart Product as					Passed

		Customer					
		Check Out Product					Passed
		Choose Payment Option as Customer					Passed

Table 6. 22 Summary of the Result of Test Cases

The table above shows the result of the tested features. Based on the result, all unit modules passed the test. Each module is integrated to work as a whole. All the features passed 100%. Based on Table 6.21, the group met the functional specifications of the system. After doing each test case, the performer of the test records the Actual Result, and these results must be matched with the Expected Result. As the Actual Results matches with the Expected Result, a test case will have a Passed mark.

Test Condition/Scenarios	**User's Name**	**Expected Output**	**Actual Output**	**Remarks**
LOGIN PAGE				
Logging in		Successful login	Successfully Login	Passed
Logging out		Successful logout	Successfully Logout	Passed
LANDING PAGE				
View Homepage		Admins shall be able to view homepage	Admins can view homepage	Passed
USER MANAGEMENT				
Add Users		Admins shall be able to add users	Admins successfully added users	Passed

Delete Users		Admins shall be able to delete users	Admins successfully deleted users	Passed
BOOST MANAGEMENT				
Boost Products and Farmers		Admins shall be able to approve Farmers' boost request	Admins can successfully boost products and Farmers	

Table 6. 23 Admin's Test Result.

Test Description	No. of Times Tested	Passed %	Failed %	Passed %	Remarks
Logging in					Passed
Logging out					Passed
View Homepage					Passed
Add Users					Passed
Delete Users					Passed
Boost Products and Farmers					Passed

Table 6. 24 Summary of Admin's Test Result

Test Condition/Scenarios	User's Name	Expected Output	Actual Output	Remarks
REGISTRATION				
Register to the site	.	Success registration of the merchant	Successful registered	Passed
LOGIN PAGE				
Logging in		Successful login	Successfully Login	Passed
Logging out		Successful logout	Successfully Logout	Passed
LANDING PAGE				
View Homepage		Farmers shall be able to view homepage	Farmers shall be able to view homepage	Passed
FARMER'S PAGE				
Add Products		Farmers shall be able to add products	Farmers successfully added products	Passed
Delete Products		Farmers shall be able to delete products	Farmers successfully deleted products	Passed
Process Orders		Farmers shall be able to process orders and ship them	Farmers successfully processed orders	Passed

Edit Product Listings		Farmers shall be able to edit their product's information	Farmers can update their products	Passed

Table 6. 25 Farmer's Test Result

Test Description	No. of Times Tested	Passed %	Failed %	Passed %	Remarks
Register to the Site					Passed
Logging in					Passed
Logging out					Passed
View Homepage					Passed
View Products					Passed
View Farmer's Information					Passed
Make Orders					Passed
Add to Cart					Passed
Send Feedback					Passed
Add Users					Passed
Delete Users					Passed
Add Products					Passed
Delete Products					Passed
Process Orders					Passed
Edit Products Listing					Passed

Table 6. 26 Summary of Farmer's Test Result

Test Condition/ Scenarios	User's Name	Expected Output	Actual Output	Remarks
REGISTRATION				
Register to the site		Success registration of the customers	Successful registered	Good
LOGIN PAGE				
Logging in		Successful login	Successfully Login	Excellent
Logging out		Successful logout	Successfully Logout	Excellent
LANDING PAGE				
View Homepage		Customers shall be able to view homepage	Customers were able to view home page	Excellent
CUSTOMER'S PAGE				
View Products		Customers shall be able to view products listed	Customers can view products	Excellent
View Farmer's Information		Customers may view Farmer's information	Customers can view Farmer's information	Very Good

Make Orders		Customers shall be able to add or delete orders and check out products	Customers successfully made orders	Good
Add to Cart		Customers shall be able to add to cart products	Customers can add to cart products	Excellent
Send Feedback		Customers may send feedback	Customers can send feedback	Good

Table 6. 27 Customer's Test Result

Test Description	No. of Times Tested	Passed %	Failed %	Passed %	Remarks
Register to the site					Passed
Logging in					Passed
Logging out					Passed
View Homepage					Passed
View Products					Passed
View Farmer's Information					Passed
Make Orders					Passed
Add to Cart					Passed
Send Feedback					Passed

Table 6. 28 Summary of Customer's Test Result

Attributes	Description	1 (Poor)	2 (Fair)	3 (Good)	4 (Very Good)	Weighted Mean
Accessibility	The website is accessible 24 hours a day, seven days a					**4**

	week.					
Reliability	The system follows the defined performance specifications					**4**
Maintainability	The website can perform successful repair within a given time.					**4**
Usability	The website functionalities met the needs of clients					**4**
Security	The website ensured the protection of user's personal information					**4**

Table 6. 29 System Software Attribute Test Result.

Attributes	Weighted Mean	Description
Accessibility	87.5%	**PASSED**
Reliability	87.5%	**PASSED**
Maintainability	87.5%	**PASSED**
Usability	87.5%	**PASSED**
Security	87.5%	**PASSED**

Table 6. 30 Summary of System Software Attributes Test Result

The result on the table signifies the tested attributes of the system. Respondents were asked to rate each attributes the performance of the attributes. After getting the weighted mean, the group was able to determine the performance of each attribute. Based on Table 6.29, by using the Likert's scale, it shows that the system attributes had good performance.

To verify if the objectives and the requirements of this website have been met, we used the Technology Acceptance Model (TAM) theory. Twenty-five respondents were asked to participate in the survey that includes Perceived Usefulness, Perceived Ease of Use, User Satisfaction, and Attribute of Usability which aims to evaluate the effectiveness of the website.

The results of the survey can be found in the table below.

Category	Lowest Rate	Highest Rate	Mean Rating
Perceived Usefulness	2	4	87.5%
Perceived Ease of Use	2	4	87.5%
User Satisfaction	2	4	87.5%
Attribute of Usability	2	5	87.5%

Table 6. 31 Survey Results

A. Functional
Example:
-Testing for accuracy and precision of the system
-MSE
-RMSE
-Testing for the functionality of the system
-Test cases

B. Non-Functional
-Testing for attributes of the system
-Survey result using Likert scale…

Conclusions and Recommendations

Appendices

Appendix A: Software Project Management Plan

A. Software Process Model
B. Roles and Responsibilities
C. Time Table

PHASE	ACTIVITIES	TARGET DATE	RESPONSIBILITY	DELIVERABLES
Planning	• Brainstorming • Discussion of Project	January 2023	Diñoso Pangilinan Mariscotes	Initial idea/Concept of the system

			San Juan	
	● Research about current farmer mobile marketplace	January 2023	Pangilinan	Background of the study/Review of related literature
	● Proposal Presentation	February 2023	Diñoso Pangilinan Mariscotes San Juan	Approved proposal of the idea
	• Documentation • (Chapter 1,2,3)	January - February 2023	Diñoso Pangilinan Mariscotes	Chapters 1 2 3 of the documentation
Requirement Analysis	• Data Flow Diagram (Contextual)	January 2023	Diñoso	DFD
	• Entity Relationship Diagram	January 2023	Diñoso	ERD
	• Drafting Graphical User Interface	January 2023	Mariscotes	Initial GUI
Design	• Defining Software Design Descriptions	February - Apr 2023	Mariscotes	GUI layout and design
	• Database Drafting	Mar 2023		Working Database
Coding	• System Development • Landing Page • Registration Page • User Dashboard • Check out Page • About Us Page • Contact Us Page	February - May, 2023	Diñoso Mariscotes	Working web application
Unit Testing	• Model Training	April - May 2023	Pangilinan San Juan	Project Prototype

Acceptance Testing	• Finalization of the Whole Documentation	May 2023	Diñoso Pangilinan Mariscotes San Juan	Finalized Documentation
	• Final Defense	May 22, 2023		Panel's approval
	• Documentation Final Submission	May 2023	Diñoso Pangilinan Mariscotes San Juan	

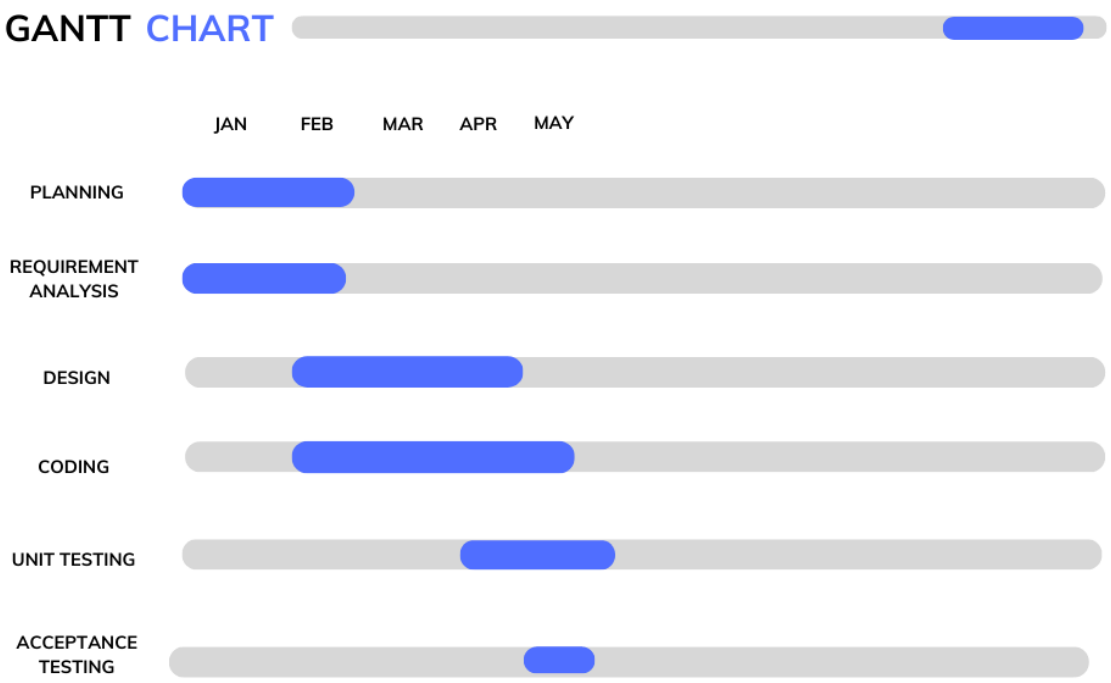
D. Task Dependencies

Tasks	Assigned Members
Chapter 1 **Introduction**	PANGILINAN
Chapter 2 **Software Needs Validation**	PANGILINAN
Chapter 3 **Software Design Evaluation** A. External Interface Requirements B. Non Functional Requirement	PANGILINAN MARISCOTES DIÑOSO
Chapter 4 **Technology and Techniques Evaluation** A. Abstract B. Introduction C. RRL	PANGILINAN SAN JUAN

D. Methodology E. Results and Discussion F. Conclusion and Recommendation	
Chapter 5 **Software Design Description** A. System Architecture B. Constraints C. Functional View D. DFD E. System Flowchart F. ERD G. Data Structure	PANGILINAN SAN JUAN
Chapter 6 **Software Testing and Evaluation** A. Test Approach B. Test Plan C. Testing	PANGILINAN SAN JUAN
Chapter 7 **Conclusion and Recommendation**	PANGILINAN SAN JUAN MARISCOTES DIÑOSO
Appendix A. Software Process Model	PANGILINAN

B. Roles And Responsibilities C. Time Table D. Task Dependencies E. Resources Needed F. Gantt Chart	SAN JUAN MARISCOTES DIÑOSO

E. Gantt Chart



Appendix B: User's Manual

How to:	Instructions:
Sign Up	Run the system > Registration
Log In	Run the system > Input credentials > Login
View Products	Run the system > Login as User > Go to "Shop"

Add to Cart	Run the system > Login as User > Go to “Shop” > Select the Product that you want to add to cart > Click “Add to Cart”
Check out Products	Run the system > Login as User > Go to “Cart” > Click “Check out”
Cash on Delivery as Payment Method	Run the system > Login as User > Select “Cash on delivery” in Check out menu

Table b. 1 User Manual

How to:	**Instructions:**
Sign Up	Run the system > Registration
Log In	Run the system > Input credentials > Login
Add Products	Run the system > Login as Farmer> Go to “Add Products” > Upload the picture of your product and fill in products information
Delete Products	Run the system > Login as Farmer> Go to “My Products” > Select the product you want to delete > Click “Delete”
Update Order Status	Run the system > Login as Farmer > Go to “Notifications” > Click “Delivery Status”

Table b. 2 User's Manual (Farmers)

How to:	**Instructions:**
Log In	Run the system > Input credentials > Admin Login
Add User	Run the system > Login as Admin > Go to “User Management” > Click “Add Users” > Fill in necessary information
Delete Buyer	Run the system > Login as Admin > Go to “User Management” > Click “Delete” to the user you want to delete
Add Buyer	Run the system > Login as Admin > Go to “Farmer Management” > Click “Add Users” > Fill in necessary information

Table b. 3 Admin's Manual

Appendix C: Picture of Actual Testing

Appendix D: System Source Code
Appendix E: Algorithm Source Code

Appendix F: Curriculum Vitae

Pangilinan, Dylan Ranze Z.

Bachelor in Science and Computer Engineer

Address: *Pag-asa Strt. Barangay Tonsuya Malabon City*

Email: dylanpangilinan30@gmail.com

Date of Birth: November 30, 2002

Marital Status: Single

Nationality: Filipino

EDUCATIONAL BACKGROUND:

College:
Technological Institute of the Philippines
Senior High School:
Immaculate Conception Parochial School ---------- Batch 2021
High School:
Immaculate Conception Parochial School ---------- Batch 2019

Elementary:
Immaculate Conception Parochial School

SKILLS:

- Basic computer hardware assembly and disassembly
- Knowledgeable in the programming language C++, Python
- Troubleshooting common computer errors
- Experience in team leadership

ORGANIZATION:

- C.O.C.C. (Cadet Officaer Candidate Course) Batch 2017
- Graduated to C.A.T. (Citizenship Advancement Training) S.Y. 2017 - 2019
- Graduated from C.A.T. with a Cadet Major rank Batch 2019

San Juan, Marc Anthony M.

Bachelor in Science and Computer Engineer

Address: *Quintos Sr. St. Sampaloc Manila*

Email: mmasanjuan@tip.edu.ph

Date of Birth: July 23, 2002

Marital Status: Single

Nationality: Filipino

EDUCATIONAL BACKGROUND:

College:		
Technological Institute of the Philippines	----------	Current Year
Senior High School:		
Arellano University Juan Sumulong Campus	----------	Batch 2020
High School:		
Ramon Magsaysay High School - Espanya	----------	Batch 2018

Elementary:

Immaculate Conception Parochial School

SKILLS:

- Basic computer hardware assembly and disassembly
- Knowledgeable in the programming language C++, Python
- Troubleshooting common computer errors
- Experience in team leadership
- Microsoft Imagine Cup 2023 – Finalist
- The Blokc Hackaton 2023 – Qualifier

ORGANIZATION:

- Computer Engineering Student Society — SY 2022 - 2023
- Tempest Gaming Society - Manila — SY 2022- 2023

General Format

Font: Arial Narrow
Font Style: Regular
Font Size: 12

Body Content Spacing:
Spacing: Before, 0 pt and After, 0 pt
Line Spacing: Single

Margin: Top (1”), Bottom (1”), Left (1”), and Right (1”)
Orientation: Portrait
Paper Size: 8.5” x 11”
Page Number: Bottom of Page, Plain Number 3

Document Mapping/Navigation: View, Show, Click Document Map/Navigation Pane

Modify Styles:
Heading 1 Font: Arial Narrow, Font Style: Bold, Font Size: 12, Font Color Automatic
Alignment: Center, Indentation: Left, 0 pt and Right, 0 pt, Special: None

Spacing: Before, 0 pt and After, 0 pt

Heading 2 Font: Arial Narrow, Font Style: Bold, Font Size: 12, Font Color Automatic
Alignment: Center, Indentation: Left, 0 pt and Right, 0 pt, Special: None
Spacing: Before, 0 pt and After, 0 pt

Heading 3 Font: Arial Narrow, Font Style: Bold, Font Size: 12, Font Color Automatic
Alignment: Justified, Indentation: Left, 0 pt and Right, 0 pt, Special: None
Spacing: Before, 0 pt and After, 0 pt

Etc.

Tables & Figures: References, Insert Caption, Label: Figure, Numbering: Arabic (1, 2, 3,…),
Click “Include chapter number”,
Chapter starts with style: Heading 2,
Use separator: – dot, Paragraph: Center.

Table Content Format:
Spacing: Before, 0 pt and After, 0 pt
Line Spacing: Single
Font Size: 10

Note: table caption is placed before the table.
figure caption is placed below the figure.

Citation and Bibliography: References, Manage Sources, New (Create Source), Type of Source (select Book, Book Section, Journal Article, etc.)

Equation Editor: Insert, Symbols, Equation, Insert New Equation
Equation Design: Structures could be Fraction, Script, Radical, Integral, etc., as applicable

https://www.stepacademic.net/ijcsr/article/view/397