**Chapter 1**

**INTRODUCTION**

The fast-paced growth of technology triggered the development of different information systems all over the world. These systems aid a great deal of establishments in various fields such as business, education and more. This leads to changes in the traditional approach performed in every industry.

In this day and age, government establishments are one that is greatly influenced by the changes that the technology has yielded. One of them are the barangays, which is a basic political unit that serves as the primary planning and implementing unit of government policies, plans, programs, projects and activities. Manual operations seem to become less and less ineffective these days. Thus, several local barangays are switching from manual to automated as the need for faster service intensifies.

Many are unable to keep up with the changes that are constant in the field of information technology but still, various computer-based systems are being made similar to this project. Local barangays will finally be able to catch up with the continuous evolution of technology.

**Project Context**

This project entitled, “Web-based Barangay Information System” intends to aid local barangays in updating, retrieving and overall maintenance of their documents, records and other data. Basically, this system aims to replace the barangay’s manual processes to a fully automated one. Its main features include the documentation of the

residents’ profile, creating announcements for the dissemination of information and the blotter reporting module.

The usual problems that occur in this establishment are the ones that are made by humans; e.g. typographical errors, orthographical errors, etc. Troubles in manual labor such as these will gradually decrease as the project’s been implemented. Furthermore, issues regarding lost or missing documents and records will be resolved since all of the data in the barangay is being automated and backed up. This means that data will be easily retrieved but not eradicated. Consuming excessive time will not be a problem seeing that this project promotes efficiency.

With this, the need for large staff in a barangay will be eliminated and undoubtedly benefit their day-to-day operations.

**Purpose and Description**

The proponents analyze the findings in system evaluation to improve and apply the suggestions of the respondents to the proposed system.

**Purpose.** The core function of this study is to make an electronic storage for the barangay so that retrieving, issuing and updating of permits, documents, certificates and residents’ records will be easier. With this, the space allocated for paper-based documents, records and logs will not be necessary anymore. Furthermore, issuance of

permits and certificates is included in this system where the user can edit and print it if desired.

**Description.** This system facilitates the management of information in the barangay e.g. residents’ records, permits, certificates, documents, etc., by maintaining it as complete and up-to-date as possible. It will serve as the central data source of a resident’s detailed records, latest announcements, economic development and population growth as well as the information about the barangay itself.

**Objectives**

Local barangays struggle to catch up in this digital era where almost everything is being automated. Safe keeping of data becomes crucial seeing that their records are all paper-based.

**General Objective.** To update, retrieve, produceand manage records, documents and other data efficiently and accurately in a barangay.

**Specific Objectives.** The following are the specific objectives that the system must accomplish.

* 1. To understand how the existing (present) system operates;
  2. To determine the problems encountered in the existing (present) system;
  3. To develop a software application that would be:

1. secure all data kept in the barangay and residents’ record within its jurisdiction by making backups and user type restriction access
2. produce or issue various documents such as certificates, permits and other related records
3. inform the citizens of the latest announcements, assemblies, events, programs, projects and activities that takes place in their barangay
4. reduce paper-based logs, records, documents and other data
5. provides detailed information about the residents based on the census data the barangay kept
6. easily manage residents’ information, barangay documents, issuance of permits and certificates and other related records
7. eliminate the manual approaches of performing tasks and efficiently manage all data in the barangay
   1. To determine whether the proposed project technically, operationally, scheduled and economically feasible to implement; and
   2. To endow with an implementation plan for the proposed software/ system.

**Scope and Limitations**

Back then, local barangays have been enduring the troubles of handling documents and records manually. This section presents the scope and limitations of the project.

**Scope.** This capstone project provides an efficient way of performing tasks and overall maintenance of the barangay’s records and data. Retrieving files will not be much of a problem unlike searching for certain documents by hand. This system allows the users to print permits and certifications such as Community Tax Certificate and Barangay Clearance if needed. It should be noted that in this project, each user type (super administrator, administrator, resident user, non-resident user) have different limits in terms of accessing the system’s functions and features. Evidently, the super admin (super administrator) has the full capability to access the system and add as many admins (administrator) as he/she wants. On the other hand, the admin can operate all the system’s functions with the exception of creating an account for an admin. The resident users’ accessible features include security related functions such as blotter and incident reporting. Other functions will all be available once the resident registered an account and logged it in. A non-resident user though can only view the system but not entirely utilized its features.

**Limitations.** For all the advantages it offers, however, this system is limited only within the jurisdiction of a barangay. This means that citizens from other town won’t be able to access the system’s many features.

**Chapter 2**

**REVIEW OF RELATED LITERATURE AND STUDIES**

This chapter will serve as a guide to the proponents and future researchers in acquainting themselves regarding the previous studies and literature that is related in the proposed system. The works and studies accomplished by other researchers that will be seen in this chapter are selected and reviewed by the proponents for the purpose of providing the readers information concerning this project.

Conducting this step is being performed in hopes of having a major impact in producing satisfactory results to this project.

**Foreign Literature**

The writings included below came from foreign researchers who tackled subjects concerning the implementation of information systems in local government.

A journal written byZeb et al. (2013) regarding a “Survey on Information Technology Use for Municipal Infrastructure Management” was conducted. In here, they discussed the increasing dependence of municipalities on computer-based systems and data management. Nowadays, information that was traditionally exchanged through human-communications is now being replaced through computer-to-computer data exchange. The survey comprised of respondents that were selected to be experts from local and regional governments. And the same as the municipalities in the Philippines, theirs were also categorized into four types based on the size of area and population: city, district, town and village. They concluded that while municipalities do not commonly use these types of systems, they routinely use them for a range of infrastructure

management functions to provide the necessary environment for more advanced and efficient communications to take place.

Urbańczyk and Wojsczyk (2013) on their “Analysis of Information System in Local Government Unit” mentioned that an information system should have adequate flow of information to ensure the effective management of the unit. Various methodologies were used for thorough diagnosis of the information systems used by the local government unit. They determined that the information system in local government unit must be subjected to continuous analysis and evaluation of its performance. This way, it would provide the ability to expand service functionalities thus, the system and needs of the user would see eye to eye.

Yigitcanlar (2014) on his “Bridging the Gap between the Citizens and Local Authorities via e-Government” believed that e-government is becoming a significant tool at the municipal, regional and national government levels. He added that the vast majority of users of within these government see benefits from being able to access different services online. It combines electronic information-based services for citizens and with their participation comes with the successful development of government online services. He concluded that though the implementation of various information systems to the local government posed many difficulties, they have the potential to improve management of information and the participation of citizens when used correctly.

Breitmeyer (2015) wrote an article regarding the disadvantages faced in a manual system. He believed that manual system put pressure on people to be correct in all details of their work at all times. The level of service needed in a manual system is dependent on individuals and this puts a requirement to run continuous training for staff

to keep them motivated and ensure they are following the correct procedures. He stated that it takes more effort and physical space to keep track of paper documents, to find information and to keep details secure. In a manual transaction, mistakes are often completely redone rather than just updated. It is appears to be lacking in terms of security and is proven time consuming which means it is costly. This raises the question if automated system has such benefits, why are not all businesses and other establishments run fully integrated systems.

Gurung et al. (2015) wrote “E-governance for an Effective and Efficient Service” which is concerned with the concept of implementing e-governance on land administration services in Southeast Asia. They believed that e-government can increase efficiency in service delivery, availability and accessibility of services and information to the users as well as government organizations. Land is considered as a major asset in most countries thus, land administration is regarded that as well. This is why they introduced the Land Information System (LIS) that supports disaster security measures. LIS assists in establishing tenure security, resolving land disputes in post disaster scenario. They concluded that despite of the challenges that enter with implementing it, it is necessary to incorporate ICT in land administration to achieve e-governance goals.

**Local Literature**

The articles and journals cited in this segment are in relation to the implementation and impact of information systems particularly within the local government units.

Sino Cruz (2014) wrote “An Assessment of the Current Status of e-LGU in Pangasinan Municipalities,” where he declared that the idea of incorporating electronic governance to the local government units (LGUs) was first initiated by the collaboration of two organizations that was a part of Support Program for Electronic Governance

which are the Department of Science and Technology (DOST) and the National Computer Center (NCC). Some of their project also included the utilization of computers in public schools. The purpose of these projects is to produce competitive human resources that can easily adopt to the changing needs of Global Information Economy and prepare the LGUs for the computerization of revenue generating and administrative functions. In his conclusion, he stated that e-LGU in the Philippines is still on its infant stage. However, he also believed that the government should pay attention to the needs of the LGUs in order to address the concerns in implementing information systems which would posed greater efficiency and productivity on its processes and transactions when integrated properly.

An article by Bautista (2015) entitled, “Promoting Digital Empowerment through Implementation of Barangay Management System” started off by stating that the Philippines is composed of local government units (LGUs) that are classified as province, city, municipality and barangay. She discussed the smallest administrative unit of the country which is the barangay. She added that it serves as the foundation from which the country’s success emanates and its ability to lead the society into a stronger, more resilient and prosperous than before. Motivated by the vision of empowering this so-called barangay, this study determines the advantages of ICT particularly an office management system that hasten the transactions performed by the barangays. She deduced that otherthan performing various tasks with greater ease and less time

consuming, among ICT’s advantages are upgraded security, improved data integrity and better data consistency.

Solteo et al. (2016) wrote a journal about the readiness of the municipal local government units in Biliran where they determined that the employees were computer literates, moderately agreed that e-governance are effective on their respective jobs and also had a moderately high level of readiness to implement it. Garcia and Pacis (2017) believed that the use of ICTs in governance is expected to empower and equip citizens to participate in the government’s various programs. It improves the knowledge and awareness of citizens through ICT-mediated information sharing that lead to better public services. Palmones et al. (2017), on the other hand, wrote an article that took account of the development of municipal e-governance systems in Camarines Sur. They stated that a proactive program in Nabua and Bula is being implemented to transform local government services into a one-stop web portal solution. It involves automating level operations across departments and sub-offices to provide faster and better service leading and overall good governance to its people. These papers highly believed and supports that e-government in the Philippines provide a citizen-centered serviced for the prosperity of the local units such as the barangay and eventually the country.

**Foreign Studies**

The studies made by researchers overseas are mentioned below for further knowledge regarding the application of various information systems used by their local government.

A study called “Local Government Records Management” created by Magee (2014) was conducted to examine the situation of a local government’s records program and the different departments’ knowledge of records management requirements and existing recordkeeping practices. She stated that records management has a vital and specific role in government, specifically the local government. It provides its citizens with a way to hold its government accountable for the action they take on our behalf. After the case study, the importance of clear records, management principles, records retention, file and document management and the storage of records is clearer than ever. She believed that with the increase of digitally created records, the records coordinator and IT must collaborate on implementing electronic records management.

Skelton (2014) wrote a study named, “The Implementation and Education of Geographic Information Systems in a Local Government for Municipal Planning” that talked about the identification of local needs that could benefit from the use of information systems and the collection and utilization of local data. The researchers of the project gained essential insight and knowledge with regards to the processes that affect the implementation of Geographic Information Systems (GIS) in a town in a developing country. Studies such as the one written by Razzak et al. (2011) implemented GIS for mapping road traffic injuries. On the other hand, Naves (2015) made a paper where GIS was researched as a new tool to evaluate epidemiology.

A study by Manikas (2015) entitled, “Records Management and Electronic Records Management” stated that records management (RM) allow people to manage records electronically while the electronic records management (ERM) focuses on both the electronic management of paper records and electronic records. The researcher determined that the RM and ERM adoption proved to be beneficial as it offers increased efficiency in the organizational environment and faster and higher quality services to the users. He observed that the basic factors that characterized perceived efficiency is time and effort and the implementation of the RM affect these factors since the system gives them the opportunity to have and easy access to the needed information.

**Local Studies**

The local studies cited in here tackle the different ideas, concept and understanding that are related in the study of the development of management systems within barangays.

A project named “Barangay Resident Record Management and Certificate Issuance System” by Algara (2015) focuses on managing the resident records of the barangay. The aim of their study is to offer detailed reliable and secured keeping of all data that can enhance the way of managing, issuance of certificates and safekeeping of all the residents’ confidential records. They proposed a safer storage of financial records, fast retrieval of files that maintains reliability and accuracy. It turned up to be very useful to the staff, from hand written records and manual issuing of barangay clearance and certificates, as well as the income and expenses reports became an automated processing system.

Another study named, “Barangay 1-A Incident Recording, Document Request, Profiling, and Project/Program Monitoring System” by Javellana (2015) was made for a barangay in Davao City. They introduced a system that promotes document management and keeping track of records in a modern approach. Similar to other barangays in the Philippines, their benefactor heavily relies on papers and most of their documents located in their physical file cabinets. The main purpose of their system is to organize barangay records, requests, projects and reduce paper dependency. They prioritized client requests transactions thus, decreasing the waiting time given by the clients. Moreover, an incident recording system was included in their project where the complaints of the client are kept. They believed it will effectively manage projects and programs ultimately facilitating the daily happenings in the barangay.

Researchers of the project entitled “Barangay Office Management System” Pulumbarit and Suarez (2017) believed that information technology plays a crucial role in assisting people in organizing, storing and retrieval of a huge amount of information. It enables us to do more tasks, thus making it a vital commodity in working environments such as barangays. They stressed that as the Philippine population increases, the number of households in each barangay rises as well. As a result, more and more people are seeking the services of the barangay council and that it will act as the unit of local government near the people. An increasing number of residents mean the same for the workload of the barangay council members and staff which consequently lead to the development of the system.

According to the project, “Computerized Information System” by Lado et al. (2017), they mentioned that computerization is a control system that manages processes in the industrial workplace. It reduces human errors and processing time, thus

boosting productivity that results in a high quality production. They stated that their project is appropriate for the use of barangay employees, who can access to the profile information of residents within the barangay. The system will give safer storage, fast retrieval of files, maintains reliability and accuracy and swift tracking of files.

Imus, et al. (2018) in a study on “Barangay Management Information System” wanted to address a solution to the difficulties that has been occurring in a barangay. They’re program contains features that records and manages information and at the same time send documents from barangay hall to the city hall. They made a system that can operate online and offline to keep, process, retrieve and update the barangay and city hall’s information through a secure network interface. Similar to the proposed project it handles information relating to the barangay profile, barangay clearances and barangay disputes/cases.

**Chapter 3**

**TECHNICAL BACKGROUND**

In this chapter, the proponents analyze the design of the proposed study in order to choose the appropriate methodology needed for its development. This includes the research methodology, analyzation of gathered data, software, hardware and system requirements. This discussion will lead the proponents to fully grasp the individual processes undergoing within the system.

**Research Approach**

The proponents used the methods stated in this chapter to attain precise and current information about the system being developed. There are various types of methodologies out there, however, the proponents decided to use the descriptive method which is considered the most basic form of research.

Descriptive research is a research methodology that is used to “describe” a situation, subject, behavior, or phenomenon. It is often described as studies that are concerned with finding out “what is” and it attempts to gather proven information that can be used to analyze a target audience or a particular subject. In this case, the barangay which is the proposed project’s beneficiary is studied to primarily describe the problems that exist and the situation within the establishment.

The researchers have gathered information by conducting interviews, giving questionnaires and surveys to the respondents which include the barangay staffs, officials and local citizens under the barangay’s supervision. This method aided the

researchers in determining the specific problems encountered in the existing system. They were also able to identify the advantages and disadvantages of the existing software and used it to provide basis in defining adjectives for the researchers. The usage of this methodology allowed the proponents to obtain more accurate and detailed information needed to accomplish the proposed project.

**Methodology**

In this section, the method used to develop this project is being determined and discussed. This will enable the proponents to fully understand the individual process behind the development of this study.

The key model needed to understand this project is the SDLC or the Software Development Life Cycle is a process that produces software with the highest quality and lowest cost in the shortest time. SDLC includes a detailed plan for how to develop, alter, maintain, and replace a software system.SDLC involves several distinct stages, including system planning, system design, system coding, system testing, and system deployment.

Figure 1. Software Development Life Cycle(SDLC)

There are various kinds of model under the SDLC. Waterfall model is one of the most popular SDLC models that illustrate the software development process in a linear sequential flow. This means that any phase in the development process begins only if the previous phase is complete. The waterfall model has distinct goals for each phase and the first is the planning which involves the understanding what needs to perform initially, identify the project’s functions and purpose. The design phase specifies the software, hardware and system requirements and also defines the overall architecture of the system. Afterwards, the development of the project started which then followed by the testing phase. By this time, the system goes through constant software testing to uncover bugs, flaws or errors. Modifications arise to improve the system’s performance, correct problems or address the user requirements. Lastly, the system must be

assessed to be aware if the finished software product is operational and functioning as intended.

Figure 2. Waterfall Model

**Requirements Analysis**

The proponents undergo in executing different methods to gather the necessary information for the advancement of the system. This segment will help in discovering the boundaries of the proposed system and how it will aid the beneficiary in their work environment.

**Observation.** As the name suggest, is the way of collecting data by means of watching or observing the subject. It is also known as job shadowing as it is the direct way of viewing individuals in their environment and how they perform tasks assigned to them. Moreover, it enables the proponents to examine how the barangay carry out the processes and get a hold of how they typically fulfill their jobs. The proponents went to

the barangay office to conduct several research methodologies including the observation of their employees’ work practices. The main problem of the establishment is their dependency on paper. Announcements and other notices are posted near the entrance as a means of disseminating loads of information. It is also observed that some barangay staffs are seemed to be running a few errands since they are not in their workplace.

**Survey.** It is the most common method of data gathering. Survey is a list of questions aimed at extracting specific data from a particular group of people. This will measure the opinions and experiences of the chosen respondents which in this case, are the citizens of the local barangay and some of its staff. It may be conducted by phone, mail, via the internet, and sometimes face-to-face on busy street corners or in malls. The researchers decided to use Google Forms to conduct their survey. The link was posted in the barangay’s Facebook group so that the residents who are also members of this group can view and answer the survey. This method will make it easier and less complicated for both the parties involved in the project.

**Questionnaire.** Unlike surveys, the questions here have options. This is a quick way to collect information, less intrusive and more objective unlike the other methods. Questionnaires are classified as quantitative and qualitative but in this case, the proponents relied on its qualitative method seeing that this approach is concerned with gaining insights on underlying motivations and reasons. This way, the focus will be the barangay staffs’ and officials’ experiences, thoughts and opinions regarding the aspect of their environment that the researchers will be able to dive deeper into the problems that they struggle with. Similar to the method of conducting the survey, the researchers

of the project used Google Forms. With this, the questionnaires are tallied and will aid the proponents in knowing the results immediately.

**Interview.** This is usually done in order to explore topics in considerable depth with the respondents and serves as a suggestive reference during the ongoing interview. This will aid the proponents in focusing attention on relevant points relating to the study and securing comparable data in different data gathering methods. During the interview, the researchers discovered that the barangay office consists mainly of three departments which are the Administration, the Treasury and the Barangay Public Information Office or BPIO. The administration is responsible for the management of their personnel as well as the overall supervision of the barangay. Treasury, as the name implies, is in charge of the finances or the flow of funds in the establishment while the BPIO is the authority in terms of distributing information to the public or even the barangay. According to the administrative head, the Barangay Guyong comprises of almost eighteen thousand people and nearly eight thousand of them are qualified voters. They were uncertain of this as the disorganization of their documents and other records implicated their work. Furthermore, the barangay chairman said that the higher ups required every barangay in the municipality to have a website of their own. This coincides with what we’re aiming for.

**System Requirements**

The stability, accuracy, efficiency and reliability that this system offers will be attained by identifying and evaluating different specifications and requirements for it to function as intended.

**Software.** It refers to the intangible parts of a computer system**.** It is the collection of instructions that enable the user to interact with a computer, its hardware, or perform tasks. It will command the user how it will operate and function. In order for the system to work, user must follow the requirements shown on the table below.

Table 1. Software Specifications

|  |  |
| --- | --- |
| **Component** | **Requirement** |
| **Web Browser** | Google Chrome or Mozilla Firefox |
| **Text Editor** | Sublime Text Editor |
| **Database** | MySQL |
| **Server** | XAMPP |

**Hardware.** These are considered as the tangible parts of a computer system. These are the physical components that make up a computer and without this, the computer would not exist, and software could not be used that is why hardware is one of the main components in a computer system. In order for the system to work, user must follow the requirements shown on the table below.

Table 2. Hardware Specifications

|  |  |  |
| --- | --- | --- |
| **Component** | **Minimum Requirements** | **Recommended Requirements** |
| **Operating System** | Microsoft® Windows® XP with Service Pack 3 | Microsoft® Windows® 7/Vista/8/8.1/10 |
| **Processor** | Intel® Core® 2 Duo E4400 or AMD Athlon® 64 X2 Dual Core processor; 2GHz processor | Intel® Pentium® 4 or Intel® Core® i-Series; 3.0GHz processor or higher |
| **Memory** | 1GB of RAM DDR2 | 4GB of RAM DDR3 or higher |
| **Storage** | 3GB of available hard-disk space | 10GB of available hard-disk space or more |

**Peopleware.**Peopleware refers to the human role in a system. They are the responsible and the one that puts their dedication and efforts to make this project possible. And to do that, the human resources have unique responsibilities delegated to them in order to fully accomplish the development of the project.

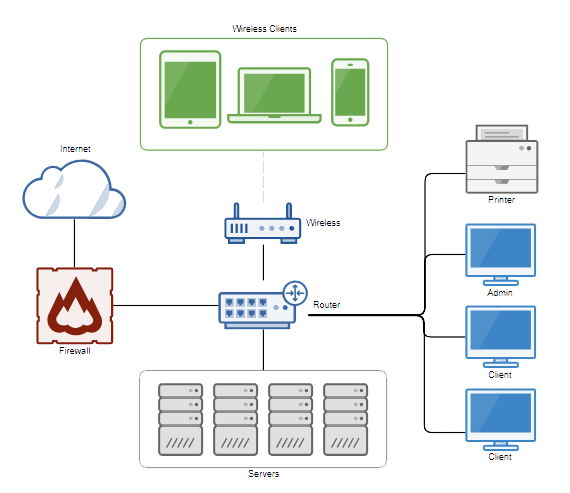
Table3. Organizational Chart

|  |  |
| --- | --- |
| **Job Title** | **Job Description** |
| **Project Manager** | In charge in overall planning and execution of a particular segment in the project. |
| **Programmer** | Responsible in coding, testing and debugging during the system development. Also in charge of the maintenance of the system. |
| **System Designer** | In charge of the design and appearance of the system’s user interface. |
| **System Analyst** | Analyze user requirements, procedures, and problems to automate or improve existing systems and review computer system capabilities, workflow, and scheduling limitations. |
| **Technical Writer** | Manages the documents needed in the system development process. |

The table above shows the job titles and the job description of the proponents in the project team. Job titles are the proponent’s position, and the job descriptions are the responsibilities or functions of the proponents during the system development. Each job titles have different tasks assigned to them and are accountable in what is going to transpire to it. These are the human resources needed to successfully develop and deploy the system.

**Network Infrastructure.** The infrastructure of network is necessary to include in developing a software project because it can lead to improved security, accessibility, usability, speed and scalability of the proposed system’s network connectivity.

Figure 4. Network Infrastructure



The connection between clients and servers is being shown in the figure above. The router is the center where the other entities are connected as it allows the connectivity to one or more computers generate a network. The right side workstations serve as the one who mainly utilize the system when implemented. To put it simply, these are the barangay staffs be it a user or an administrator, who will manage the system and data established within the servers. The sharing of information and resources among numerous clients is made possible through the servers. On the other hand, the wireless that is included in this network design is related to the devices the wireless clients used to connect to the internet and therefore, be able to operate the web-based barangay management system which is the proposed project. The firewall however, is concern with blocking unauthorized access within the network. Lastly, the wireless clients mentioned is without a doubt the citizens that are under the barangay’s jurisdiction.

**Chapter 5**

**Conclusions and recommendation**

The proponents conducted a research using different approaches from various establishments including universities but primarily the barangay office with the intention of gathering information concerning a barangay’s operations and work practices.

The need to catch up with the technological growth of the country instigated the making of this capstone project. Local government units (LGUs) especially the local barangays are now being insisted to have at least a functioning website of their own for a better communication that will lead to a better community. E-government was encouraged not only in the country but all over the world as well.

The proponents began by coming up with a beneficiary–the Barangay Guyong of Santa Maria Bulacan–and gathering the necessary information from them. The residents’ information details, their household information, Barangay clearance and other documents were examined if it has to be included in the project. Thoughts and opinions of the staffs and the residents within the barangay were then asked several questions for data gathering and feasibility studies.

The development of the software project was in progress so were the documents that were vital to its foundation. Every meeting, a presentation or a demo was conducted by the thesis adviser to check the system’s development progress. Afterwards, it was then presented in the company of three professionals for further assessment. Their critiques were beneficial and appreciated as it pointed out the problems, setbacks and also the solution to them. Then finishing up paperworks and conducting evaluations were the remaining tasks for the proponents to see through.

**Conclusions**

After performing different research methodologies, the proponents came across several difficulties regarding the procedures that the barangay used that can be resolved by the proposed system. These mainly include the excessive paper dependency within the establishment, time consumed pertaining to the handling of tasks manually and data loss as a result of relying on paper. This project is deemed to prove itself useful due to the following reasons:

1. That the proposed system can reduce the paper dependency within the barangay thus, decreased circumstances like missing documents and data transmission problems.
2. That the software project intends to substantially reduce if not eliminate the manual approaches of performing tasks in the barangay to promote efficient management of their data.
3. That this project would be a great improvement to the barangay’s way of informing its residents of the latest announcements, events, assemblies and programs.
4. That this software project would ensure the safe keeping of residents’ records and documents of the barangay thus, modifying and retrieval of these data will be easier.
5. That the proposed system would greatly lower the time consumed in terms of the client request waiting time as it aims to make the documents and other data electronic.

**Recommendations**

Three respectable individuals from different field of information technology are invited to review and criticize the proposed system. The researchers listed all the advices and suggestions made by these peoplefor the betterment of the software project. Below are the said remarks:

1. That the software project should have the ability to import or mass uploading files (e.g. Excel spreadsheet) in order to lessen the strain on the administrator’s side pertaining to the insertion of large number of barangay residents’ records.
2. That the software project must contain verification particularly an e-mail verification for the system’s resident user account so as to have an authentication and also to have a contact to between the system and the user regarding the blotter report module.
3. That the proposed system ought to generate reports and print them anytime should the barangay ever have or need to.
4. That the software project should give the resident user an ability to request when an information in his/her record is in need of modification.
5. That the proposed project needs a notification module should the user have the need to be informed and reminded of the progress concerning the blotter that he/she sent.
6. That this system’s announcements component must be evident to the users’ eye and has an automatic deletion feature if it were old and not that important anymore. In addition, the announcements should be arranged in order of significance.
7. That the downloadable forms component of the software project needs a list or logs of the number of users that downloaded the documents provided in the system.

APPENDICES

**APPENDIX A.**

**ENGLISH CRITIC’S CERTIFICATION**

**APPENDIX B.**

**STATISTICIAN’S CERTIFICATION**

**APPENDIX C.**

**PROJECT PROPOSAL**

**Web Based Barangay**

**Information System**

**for Barangay Guyong**



Program

**Bachelor of Science in Information Technology**

**CHRISTIAN JOSEPH A. DUCUT**

**DANIEL C. GULMATICO**

**KATHLEEN DC. HERNANDEZ**

**MA. JULIE ANNE C. LORCA**

Proponents

**MR. JAYSON R. HERMOGENES**

Adviser

1. **Project Overview and Problem Statement**

The fast-paced growth of technology triggered the development of varied information systems all over the world. These systems aid a great deal of establishments, but of course not everyone is able to keep up with the changes that are constant in the industry of information technology. One of them is the local barangays in every town of the Philippines. It is known that there are transactions occurring daily in a barangay, be it an issuance of papers such as permits, clearances, certifications and more. However, most of these tasks are performed manually which is why the proponents decided to create a web-based management system in order to resolve such cases. Because the objective of this project is to efficiently manage the issuance of different documents and keep all the data secure.

The usual problems encountered in a traditional manner of working are the ones that are made by humans (e.g. typographical errors, etc.). That is why this system is capable of generating documents, thus, manual process will soon be out of the picture. Issues regarding lost or missing documents and records will be resolved because all of the data in the barangay is being automated and backed up. This means that data will be easily retrieved but not eradicated. Furthermore, this system has security or a peace-keeping function that operates if a resident of the barangay has information to relay like an incident report, blotter or even a minor complaint. This will eliminate the need for a large staff in a barangay.

To conclude, the proponents decided to see this project in its entirety to assist the local barangays’ issues stated above. With this, the barangay will be able to maintain their resident’s records based on the census data that they kept. All this will enable them to finally catch up with the continuous evolution of technology and greatly benefit their day-to-day operations.

1. **Project Objectives**

To accomplish the goal of producing accurate reports and proper management of automated files efficiently, the following will serve as the objectives of this project:

1. To understand how the existing (present) system operates;
2. To determine the problems encountered in the existing (present) system;
3. To develop a software application that would be:
   1. To secure all data kept in the barangay and residents’ record within its jurisdiction by making backups and user type restriction access.
   2. To produce or issue various documents such as certificates, permits and other related records.
   3. To provide a detailed information of a resident and easily retrieve it.
   4. To eliminate the manual approaches of performing tasks and efficiently manage all data in the barangay.
   5. To inform the citizens of the latest announcements, assemblies, events and activities that takes place in their barangay.
4. To determine whether the proposed project technically, operationally, scheduled and economically feasible to implement; and
5. To endow with an implementation plan for the proposed software/system.
6. **Project Benefits**

Stated below are the benefits that will be gain from this project which will be provided to the users and to the developers:

**To the users:**

1. This project will ensure the safe keeping of records, documents and other data of the barangay.
2. This study promotes an efficient way of updating and retrieving of records and overall maintenance of the barangay’s database.
3. Users will be informed in the happenings in their area since this system provides announcements and other notices.

**To the proponents:**

1. By creating this project, the proponents will develop their knowledge in terms of creating web-based applications. This will familiarize them in other programming languages.
2. Developing this project in a group-based environment will enhance the proponents’ ability to cooperate and communicate effectively.
3. **High-level Components of the Software Product**

**General Components (Administrator, Officials, Citizen)**

**I. Register/Registration.** To access the application, the user must register.

**II. Log in.** After the registration, logging in is required to enter this application and gain access to its many benefits. User type is divided into 3 categories which is the administrator, officials and citizen.

**III. Blotter.** Once the user logged in, he /she is capable of filing a blotter that will be assessed by the administrator.

**VI. Certificates and Permits.** This will enable the users to sign forms and print them if needed.

**V. Homepage.** Recent announcements in the barangay are being shown here such as assemblies, activities and events.

**Specific Components**

**Officials**

**I. Attendance.** Barangay officials’ attendance is being monitored through this. Time in time out, absences and late are included in this module.

**Citizen**

**I. Report.** In this module, citizens within the barangay can send information about whether they have an incident to report or other related statement.

1. **List of Optional Functional Units of the Software Product**

**I. Log out.** This will change the user of the application. It lets the user switch account if needed.

**II. Barangay Officials.** It shows the list of the officials in the barangay.

**III. Forgot Password.** In case the user forgot his/her password, this will let them retrieve it provided that he/she has verification saying that their account is without a doubt their own.

1. **Tools and Technologies Used**

The development of this project will be achievable with the use of the following tools and technologies:

* Hypertext Preprocessor (PHP) – an open-source, server side HTML embedded scripting language used to create dynamic web pages.
* MySQL – a powerful open-source database server built based on a relational database management system (RDBMS) and is capable of handling a large concurrent database connection.
* Bootstrap – a free and open-source front-end framework for designing websites and web applications.
* jQuery – a concise and fast JavaScript library that can be used to simplify event handling, HTML document traversing, Ajax interactions and animation for speedy website development.
* Apache Cordova – an open-source mobile development framework that will allow us to use standard web technologies such as HTML5, CSS3 and JavaScript for cross-platform development.

1. **Similarity With Any Previous Project**

These are the references that have similar functions and components to this application.

Lado, M. J., Et al. (2017). "Computerized Information System". *Unpublished Undergraduate Thesis*, Colegio de San Antonio de Padua – Guinsay, Danao City

Magleo, E., Et al. (2018). "Barangay Management Information System". *Unpublished Undergraduate Thesis*, San Carlos College – San Carlos, Pangasinan

Gonzales, M., Et al. (2018). "Barangay Management System". *Unpublished Undergraduate Thesis*, De La Salle Lipa – Lipa, Batangas

**APPENDIX D.**

**COMMUNICATION LETTERS**

**APPENDIX E.**

**INTERVIEW GUIDE QUESTIONS**

**Interview Guide Questions**

PURPOSE: Your answer to the questions below will provide us with information about your establishment’s situation in carrying out daily activities using your existing system or operations. Please take as much space to answer each question as you need.

1. Does your establishment use any software product/s (e.g. attendance monitoring, payroll system, etc.)?
   1. If not, how long have you been operating manually?
   2. What are the usual problems that you encountered when performing your day-to-day activities?
   3. To these stated problems, what specific problem you encountered most?
   4. Does it affect your job? How?
2. Are there any instances that you encountered a problem wherein there’s a missing document or any information for a resident within the barangay?
   1. What actions you usually do when you encountered this kind of situation?
3. Are you familiar in utilizing various websites (e.g. Facebook, Twitter, etc.)?
   1. If yes, please state your insight/s about these software products.
4. What is your personal view in automating the manual recording, retrieving and producing of different types of document (e.g. residents' information, clearances, permits, etc.)?
   1. What are the possible features that you expect from this automated system aside from the ones stated above?
5. Do you think that this kind of system is needed in your establishment? Why?
6. Are you in favor of replacing the manual labor into an automated management system?
7. If this project is somehow implemented, what are the possible benefits that you expect from this automated system to have?
8. What might possible problems do you see if the system is implemented?

**APPENDIX F.**

**SURVEY QUESTIONNAIRES**

SURVEY QUESTIONNAIRE #1

Name:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Designation: Local citizen Gender: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**OBJECTIVE:** The goal of this questionnaire is to reach your feelings and opinion regarding the usage of an online barangay management system**.** Your response to these questions will be essential to our capstone project. Do answer the questions to the best of your ability.

INSTRUCTION: Please shade the circle that corresponds to your answer. See the legend below.

**\*SA** – Strongly Agree **\*A** – Agree **\*D** – Disagree **\*SD** – Strongly Disagree

1. I am oftentimes in need to go to the barangay to accomplish paperwork requirements such as permits, clearances and other documents.

**O SA O A O D O SD**

1. I find it time-consuming when going to the local barangay establishment to manage paperworks.

**O SA O A O D O SD**

1. I have encountered difficulties concerning matters in the barangay (e.g. long queuing, unattended by staff, etc.)

**O SA O A O D O SD**

1. There are situations when I want to report to the barangay officials regarding complaints, blotter, incidents, etc.

**O SA O A O D O SD**

1. I can easily file a report or a complaint in our local barangay.

**O SA O A O D O SD**

1. I always know the happenings in our barangay regarding events, programs, occasions and incidents.

**O SA O A O D O SD**

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

Signature

SURVEY QUESTIONNAIRE #2

Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_ Date: \_\_\_\_\_\_\_\_\_\_\_\_\_

Designation: Local citizen Gender: \_\_\_\_\_\_\_\_\_\_\_

**OBJECTIVE:**The goal of this questionnaire is to reach your feelings and opinion about the possible use of an online barangay management system**.** Your response to these questions will be essential to our capstone project. Do answer the following questions to the best of your ability.

1. Does your local barangay provides some kind of software product for the local citizens to use (e.g. website of the barangay)?

**O** Yes **O** No

1. As a citizen of your town, is keeping in touch with the barangay important? State your reason.

**O**Yes **O**No

**Reason:**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

1. If implemented, do you see any possible problem/s regarding this online barangay management system? State your reason.

**O**Yes **O**No

**Reason:**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

1. Do you think this systemwill benefit you? If yes, please do specify its advantage/s.

**O**Yes **O**No

**Benefits**:

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

1. Do you own a device, a mobile phone, personal computer or a laptop that is able to connect to the internet and access different websites (e.g. Facebook, Twitter, Instagram, etc.)?

**O**Yes **O**No

1. Do you have a stable internet connection in your home?

**O**Yes **O**No

1. Do you ever have a complaint, a protest or any kind of report that you want to convey to the barangay in your town? If yes, what did you do? Please state the situation.

**O**Yes **O**No

**Situation:**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

1. Have you ever encountered and/or used an existing software product/s that helps you file a report, be informed in the happenings within your barangay or manage necessary paperworks? If yes, name that software/s.

**O**Yes **O**No

**Name of the Software/s:**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

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

Signature

**APPENDIX G.**

**FORMS AND DOCUMENTS**

**APPENDIX H.**

**SYSTEM EVALUATION**

**APPENDIX I.**

**SOFTWARE DEVELOPMENT COST**

**SOFTWARE DEVELOPMENT COST**

1. **Hardware Depreciation Cost**

HP Pavilion 15 Laptop Php 31,600.00

ASUS X441SA Laptop 19,500.00

Canon Printer Pixma MP237 3,465.00

**Total Hardware Cost Php 54,565.00**

Hardware Cost : Php 54,565.00

Life Expectancy : 3 years

Development Period : 10 months

Salvage Value [(Hardware Cost / Life Expectancy) \* Development Period] / 12

= [(54,565.00 / 10) \* 3] / 12

**= Php 1,364.13**

Annual Depreciation Cost = (Hardware Cost – Salvage Value) / Life Expectancy

= (54,565.00 – 1,364.13)/ 3

**= Php 17,733.62**

Monthly Depreciation Cost = Annual Depreciation Cost / 12

= 17,733.62/ 12

**= Php 1,477.81**

**Hardware Depreciation Cost** = Monthly Depreciation Cost \* Development Period

= 1,477.81\* 10

**= Php 14,778.10**

1. **Software Depreciation Cost**

Microsoft Windows 10 Php 7,000.00

Microsoft Office 2010 Professional Edition 3,000.00

Sublime Text Editor 1,500.00

**Total Software Cost Php 11, 500.00**

Software Cost : Php 11, 500.00

Life Expectancy : 2 years

Development Period : 10 months

Salvage Value [(Software Cost / Life Expectancy) \* Development Period] / 12

= [(11, 500.00 / 2) \* 10] / 12

**=Php 4,791.67**

Annual Depreciation Cost = (Software Cost – Salvage Value) / Life Expectancy

= (11, 500.00 –4,791.67) / 2

**=Php3, 354.17**

Monthly Depreciation Cost = Annual Depreciation Cost / 12

= 3,354.17/ 12

**= Php 279.51**

**Software Depreciation Cost** = Monthly Depreciation Cost \* Development Period

= 279.51 \* 10

**=Php 2,795.10**

1. **Labor Cost**

Programmer/System Analysts (per Head) : Php 28,500.00/Mo

Project Manager 23,250.00

Technical Writer 20,500.00

System Designer 21,100.00

**Total Salary 93,350.00/Mo**

Development Period : 10 Months

**LaborCost**  = Developer’s Fee \* 10

= 93,350.00 \* 10

**= Php 933,500.00**

1. **Miscellaneous Cost**

**Web Hosting Php 3,000.00**

Web Hosting Plan - Personal 1,500.00

Domain name registration fee/year 1,500.00

2 ream Letter-sized bond paper (160.00 per ream) Php 320.00

Canon Ink refill 379.00

Others (Transportation, food, etc) 1,200.00

**Total Miscellaneous Cost Php 4,899.00**

1. **Overhead Cost**

1st Month - Electric Bill Php 250.00

2nd Month- Electric Bill 250.00

3rd Month - Electric Bill 250.00

4thMonth- Electric Bill 250.00

5th Month- Electric Bill 250.00

6th Month- Electric Bill 250.00

7th Month- Electric Bill 250.00

8th Month- Electric Bill 250.00

9th Month- Electric Bill 250.00

10th Month- Electric Bill 250.00

**Total Overhead Cost Php 2,500.00**

1. **Summary**

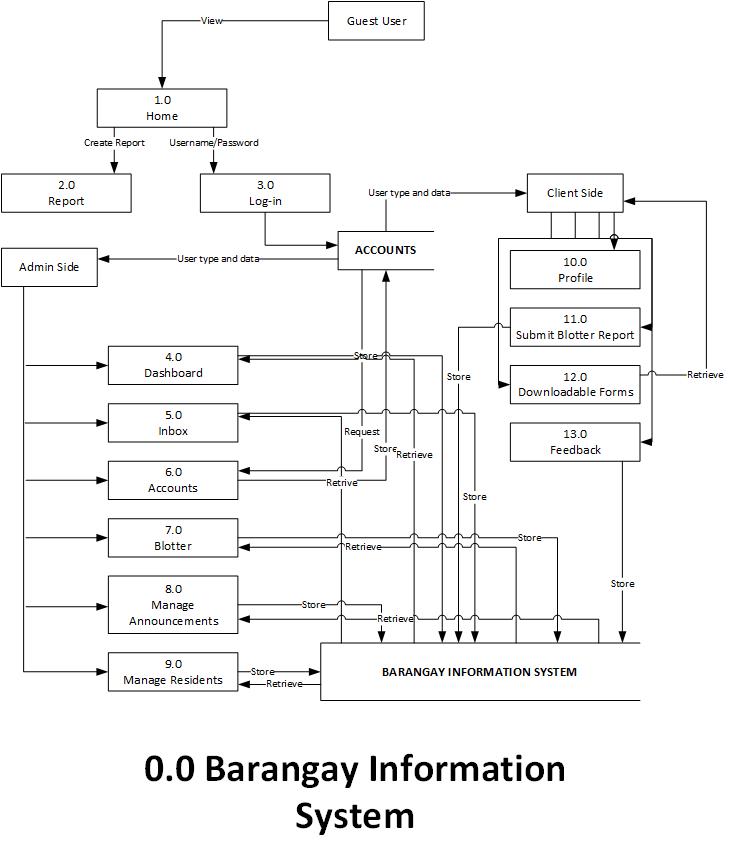
|  |  |
| --- | --- |
| **Category** | **Cost** |
| Hardware Depreciation Cost | Php 14,778.10 |
| Software Depreciation Cost | 2,795.10 |
| Labor Cost | 933,500.00 |
| Miscellaneous Cost | 4,899.00 |
| Overhead Cost | 2,500.00 |
| **Total Development Cost** | **Php 956,472.20** |

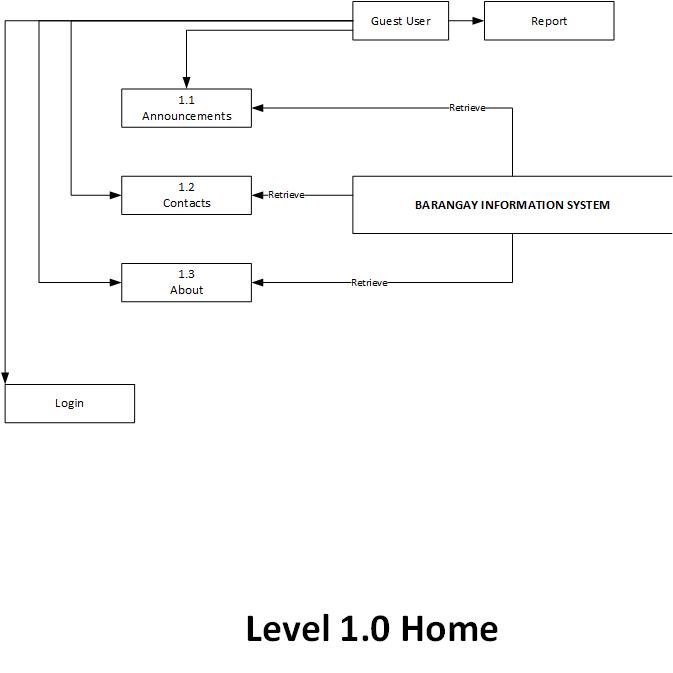
**APPENDIX J.**

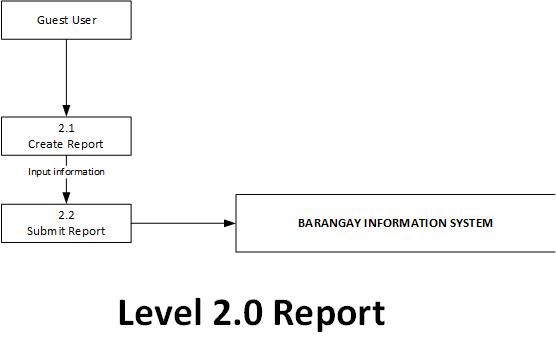
**COST BENEFIT ANALYSIS**

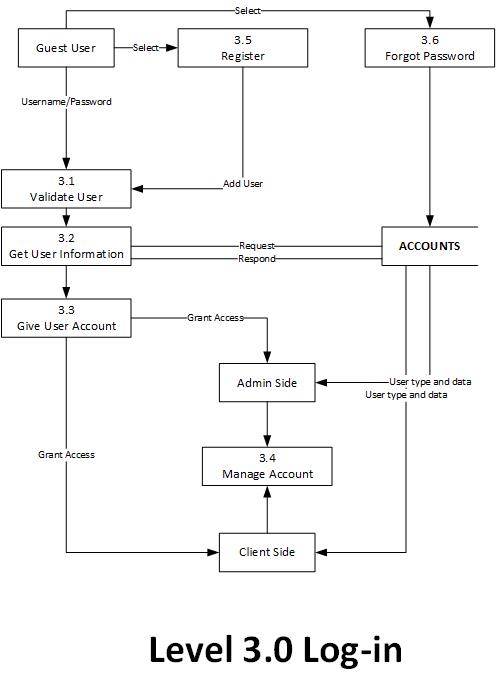
**APPENDIX K.**

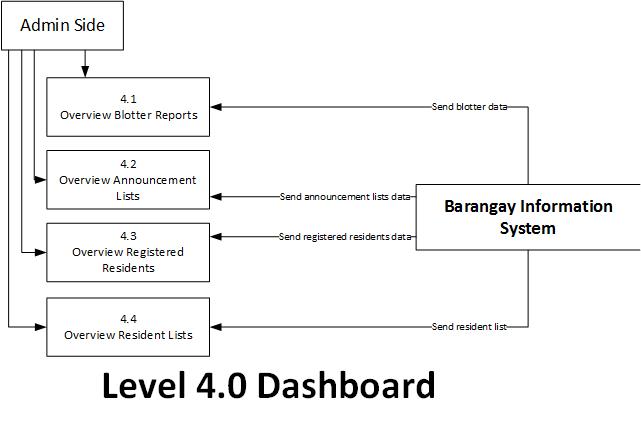
**DATA FLOW DIAGRAM**

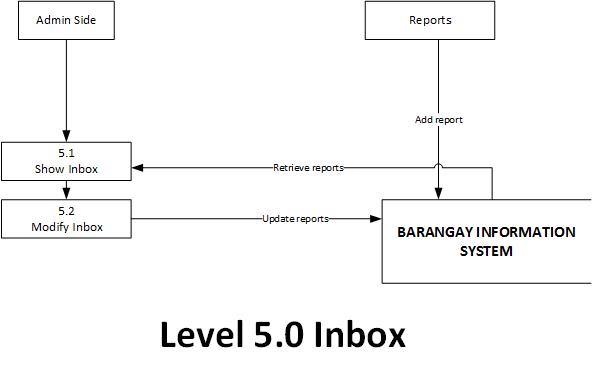


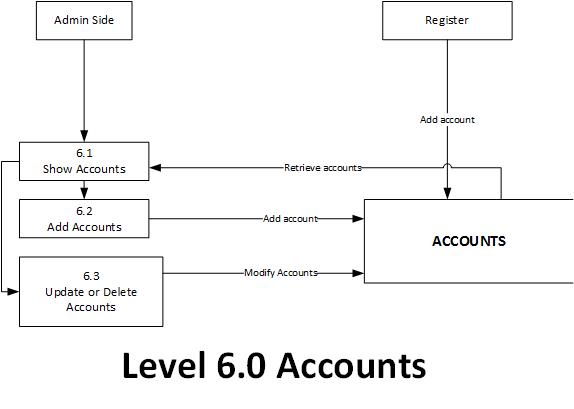


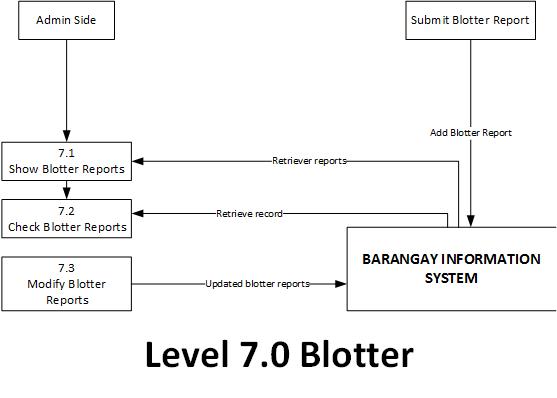


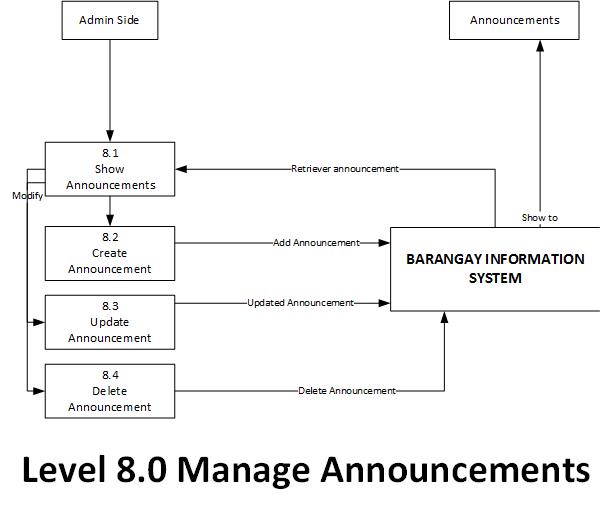




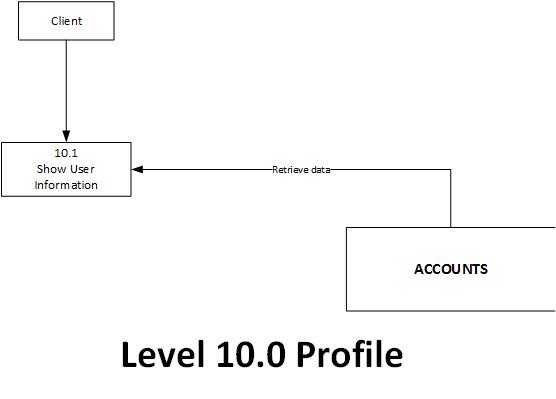


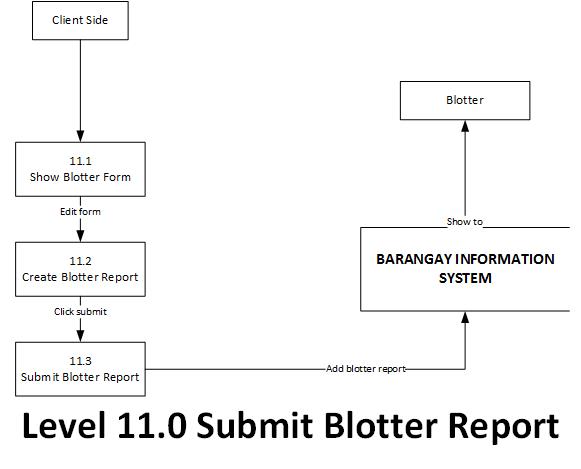


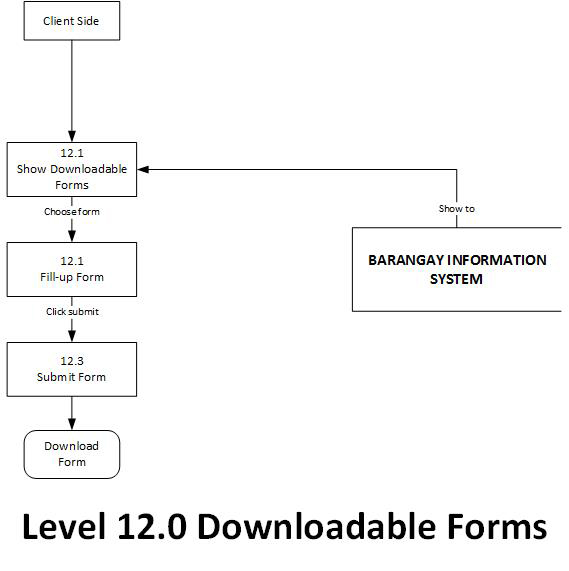


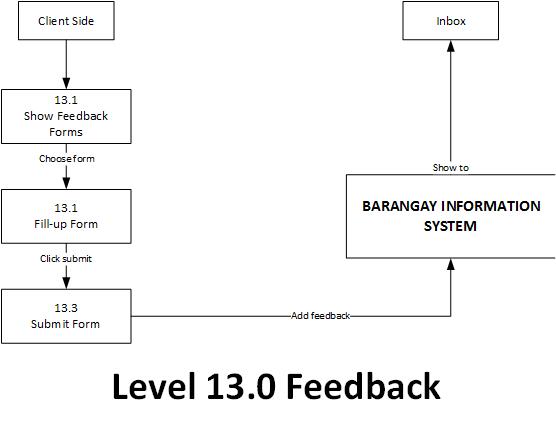








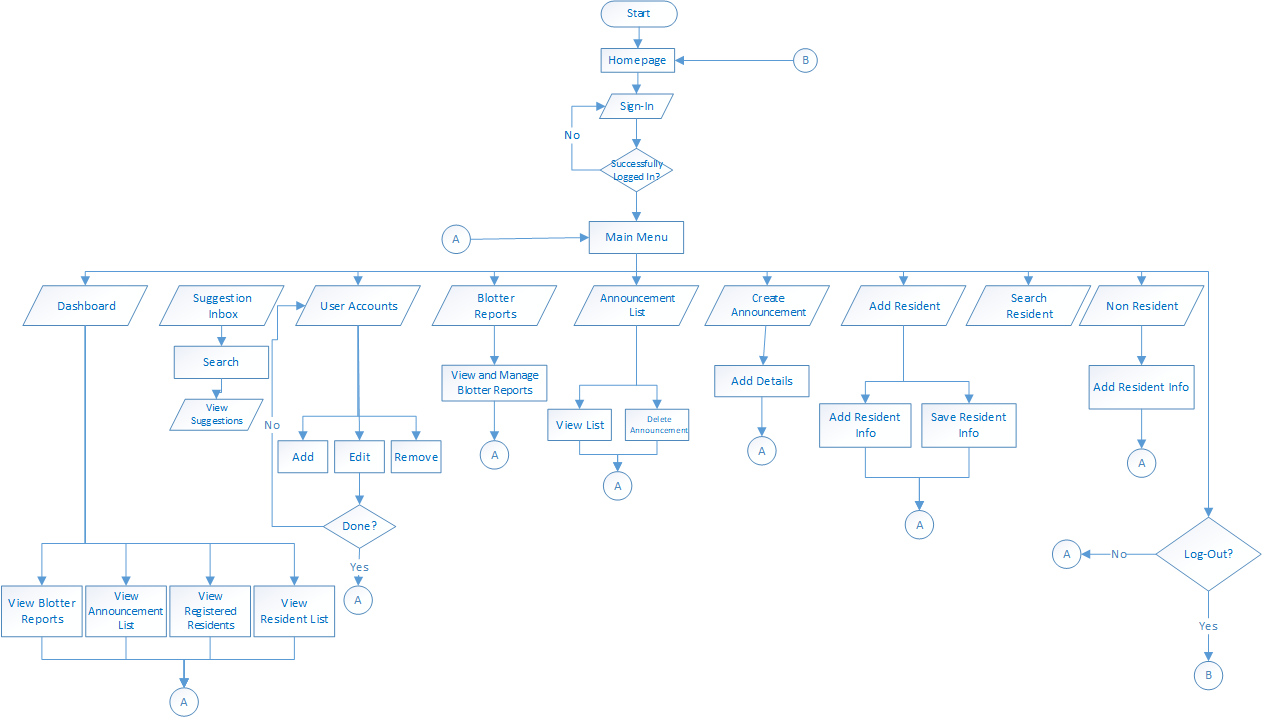




**APPENDIX L.**

**SYSTEM FLOWCHART**

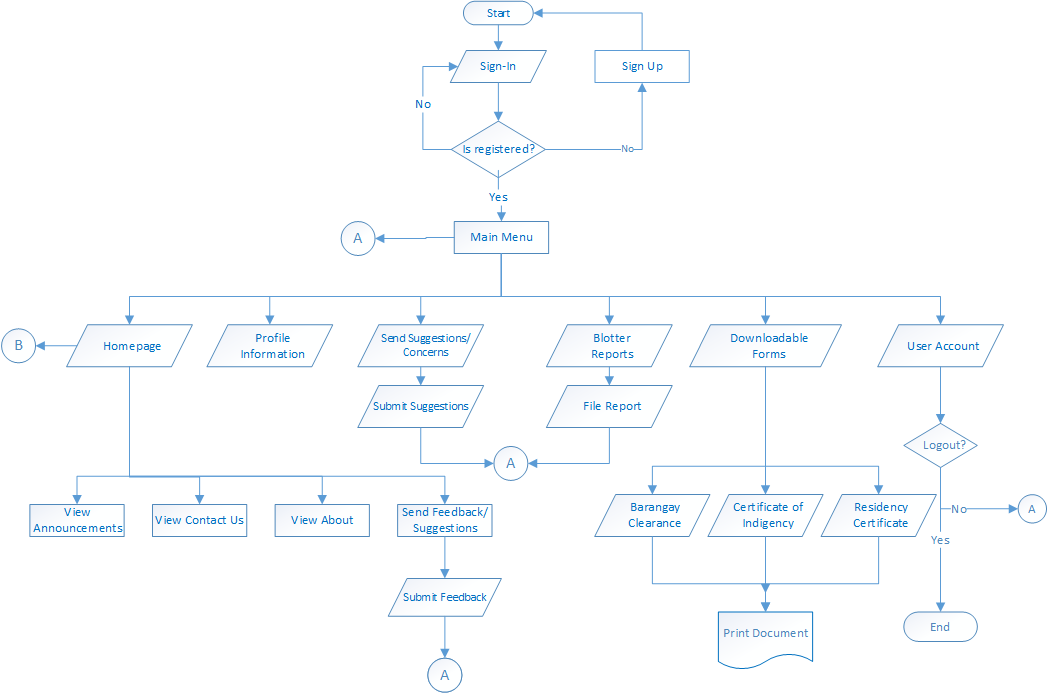
ADMIN SIDE



NON- RESIDENT SIDE



RESIDENT SIDE



**APPENDIX M.**

**DATA DICTIONARY**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Table Name** | **Column Name** | **Data Type** | **Constraints** | **Unique** |
| Announcements | id (Primary) | int(10) | Primary Key | Yes |
|  | Title | varchar(191) | Not Null | No |
|  | Description | text | Not Null | No |
|  | remember\_token | varchar(100) | Not Null | No |
|  | created\_at | timestamp | Not Null | No |
|  | updatedat | timestamp | Not Null | No |
|  |  |  |  |  |
| barangayclearance | gender | varchar(191) | Not Null | No |
|  | name | varchar(191) | Not Null | No |
|  | street | varchar(191) | Not Null | No |
|  | civil | varchar(191) | Not Null | No |
|  | purpose | varchar(191) | Not Null | No |
|  | dateissued | varchar(191) | Not Null | No |
|  | remember token | varchar(100) | Not Null | No |
|  | created\_at | timestamp | Not Null | No |
|  | updatedat | timestamp | Not Null | No |
|  |  |  |  |  |
| Create\_blotters | id (Primary) | int(10) | Primary Key | Yes |
|  | controlno | varchar(191) | Not Null | No |
|  | accused | varchar(191) | Not Null | No |
|  | contactnum | varchar(191) | Not Null | No |
|  | complainant | varchar(191) | Not Null | No |
|  | datetime | varchar(191) | Not Null | No |
|  | caseofincident | varchar(191) | Not Null | No |
|  | detail | text | Not Null | No |
|  | status | varchar(191) | Not Null | No |
|  | remember token | varchar(100) | Not Null | No |
|  | createdat | timestamp | Not Null | No |
|  | updated\_at | timestamp | Not Null | No |
|  |  |  |  |  |
| Feedback | id | int(11) | Primary Key | Yes |
|  | name | varchar(191) | Not Null | No |
|  | concerntitle | varchar(191) | Not Null | No |
|  | concerndetail | varchar(191) | Not Null | No |
|  | contactnum | varchar(191) | Not Null | No |

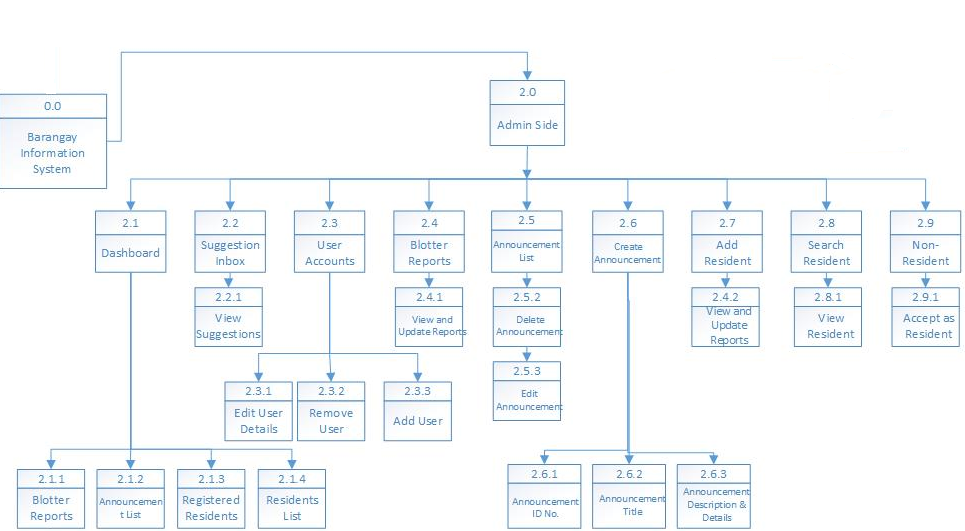
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Forms | gendertitle | varchar(191) | Not Null | No |
|  | Name | varchar(191) | Not Null | No |
|  | Street | varchar(191) | Not Null | No |
|  | Civil | varchar(191) | Not Null | No |
|  | purpose | varchar(191) | Not Null | No |
|  | Reason | varchar(191) | Not Null | No |
|  | Reason | varchar(191) | Not Null | No |
|  | Age | varchar(191) | Not Null | No |
|  | dateissued | varchar(191) | Not Null | No |
|  | created at | timestamp | Not Null | No |
|  | updated\_at | timestamp | Not Null | No |
|  |  |  |  |  |
| Non-resident | id (Primary) | int(10) | Primary Key | Yes |
|  | lastname | varchar(191) | Not Null | No |
|  | firstname | varchar(191) | Not Null | No |
|  | middlename | varchar(191) | Not Null | No |
|  | email | varchar(191) | Not Null | No |
|  | image | varchar(191) | Not Null | No |
|  | gender | varchar(191) | Not Null | No |
|  | civil | varchar(191) | Not Null | No |
|  | age | varchar(191) | Not Null | No |
|  | birthdate | varchar(191) | Not Null | No |
|  | votersno | varchar(191) | Not Null | No |
|  | yearsres | varchar(191) | Not Null | No |
|  | street | varchar(191) | Not Null | No |
|  | barangay | varchar(191) | Not Null | No |
|  | cityprovince | varchar(191) | Not Null | No |
|  | province | varchar(191) | Not Null | No |
|  | zipcode | varchar(191) | Not Null | No |
|  | sitio | varchar(191) | Not Null | No |
|  | mlast | varchar(191) | Not Null | No |
|  | mfirst | varchar(191) | Not Null | No |
|  | mmiddle | varchar(191) | Not Null | No |
|  | flast | varchar(191) | Not Null | No |
|  | ffirst | varchar(191) | Not Null | No |
|  | fmiddle | varchar(191) | Not Null | No |
|  | citizenship | varchar(191) | Not Null | No |
|  | religion | varchar(191) | Not Null | No |
|  | hno | varchar(191) | Not Null | No |
|  | hrel | varchar(191) | Not Null | No |
| **Table Name** | **Column Name** | **Data Type** | **Constraints** | **Unique** |
|  | sib | varchar(191) | Not Null | No |
|  | created\_at | timestamp | Not Null | No |
|  | updatedat | timestamp | Not Null | No |
|  |  |  |  |  |
| Resident | id (Primary) | int(10) | Primary Key | Yes |
|  | lastname | varchar(191) | Not Null | No |
|  | firstname | varchar(191) | Not Null | No |
|  | middlename | varchar(191) | Not Null | No |
|  | email | varchar(191) | Not Null | No |
|  | image | varchar(191) | Not Null | No |
|  | gender | varchar(191) | Not Null | No |
|  | civil | varchar(191) | Not Null | No |
|  | age | varchar(191) | Not Null | No |
|  | birthdate | varchar(191) | Not Null | No |
|  | votersno | varchar(191) | Not Null | No |
|  | yearsres | varchar(191) | Not Null | No |
|  | street | varchar(191) | Not Null | No |
|  | barangay | varchar(191) | Not Null | No |
|  | cityprovince | varchar(191) | Not Null | No |
|  | province | varchar(191) | Not Null | No |
|  | zipcode | varchar(191) | Not Null | No |
|  | sitio | varchar(191) | Not Null | No |
|  | mlast | varchar(191) | Not Null | No |
|  | mfirst | varchar(191) | Not Null | No |
|  | mmiddle | varchar(191) | Not Null | No |
|  | flast | varchar(191) | Not Null | No |
|  | ffirst | varchar(191) | Not Null | No |
|  | fmiddle | varchar(191) | Not Null | No |
|  | citizenship | varchar(191) | Not Null | No |
|  | religion | varchar(191) | Not Null | No |
|  | hno | varchar(191) | Not Null | No |
|  | hrel | varchar(191) | Not Null | No |
|  | sib | varchar(191) | Not Null | No |
|  | remember\_token | varchar(100) | Not Null | No |
|  | createdat | timestamp | Not Null | No |
|  | updated\_at | timestamp | Not Null | No |
|  |  |  |  |  |
| Roles | id (Primary) | int(10) | Primary Key | Yes |
|  | name | varchar(191) | Not Null | No |
|  | display\_name | varchar(191) | Not Null | No |
|  | description | varchar(191) | Not Null | No |
| **Table Name** | **Column Name** | **Data Type** | **Constraints** | **Unique** |
| Role\_user | Role\_id | Int(10) | Primary Key | Yes |
|  | userid (Primary) | int(10) | Primary Key | Yes |
|  | User\_type | Varchar(191) | Not Null | No |
|  |  |  |  |  |
| users | Id | int(10) | Primary Key | Yes |
|  | lastname | varchar(191) | Not Null | No |
|  | firstname | varchar(191) | Not Null | No |
|  | middlename | varchar(191) | Not Null | No |
|  | Image | varchar(191) | Not Null | No |
|  | Email | varchar(191) | Not Null | No |
|  | username | varchar(191) | Not Null | No |
|  | password | varchar(191) | Not Null | No |
|  | Role | varchar(191) | Not Null | No |
|  | Gender | varchar(191) | Not Null | No |
|  | civil | varchar(191) | Not Null | No |
|  | age | varchar(191) | Not Null | No |
|  | birthdate | varchar(191) | Not Null | No |
|  | votersno | varchar(191) | Not Null | No |
|  | Yearsres | Varchar(191) | Not Null | No |
|  | street | varchar(191) | Not Null | No |
|  | barangay | varchar(191) | Not Null | No |
|  | cityprovince | varchar(191) | Not Null | No |
|  | province | varchar(191) | Not Null | No |
|  | zipcode | varchar(191) | Not Null | No |
|  | sitio | varchar(191) | Not Null | No |
|  | mlast | varchar(191) | Not Null | No |
|  | mfirst | varchar(191) | Not Null | No |
|  | mmiddle | varchar(191) | Not Null | No |
|  | ftast | varchar(191) | Not Null | No |
|  | thirst | varchar(191) | Not Null | No |
|  | fmiddle | varchar(191) | Not Null | No |
|  | citizenship | varchar(191) | Not Null | No |
|  | religion | varchar(191) | Not Null | No |
|  | hno | varchar(191) | Not Null | No |
|  | hrel | varchar(191) | Not Null | No |
|  | sib | varchar(191) | Not Null | No |
|  | is super admin | tinyint(3) | Not Null | No |
|  | remember\_token | varchar(100) | Not Null | No |
|  | created at | timestamp | Not Null | No |
|  | updated\_at | timestamp | Not Null | No |

**APPENDIX N.**

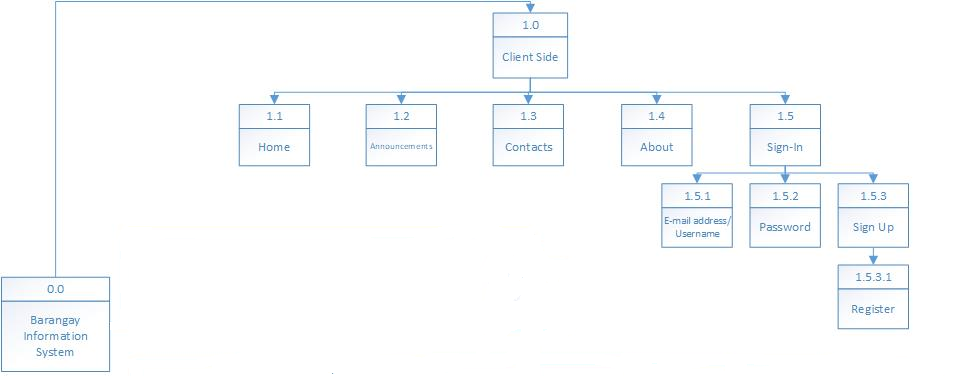
**ENTITY RELATIONSHIP DIAGRAM (ERD)**

**APPENDIX O.**

**VISUAL TABLE OF CONTENTS (VTOC)**

ADMIN SIDE

CLIENT SIDE



**APPENDIX P.**

**INPUT- PROCESS- OUTPUT**

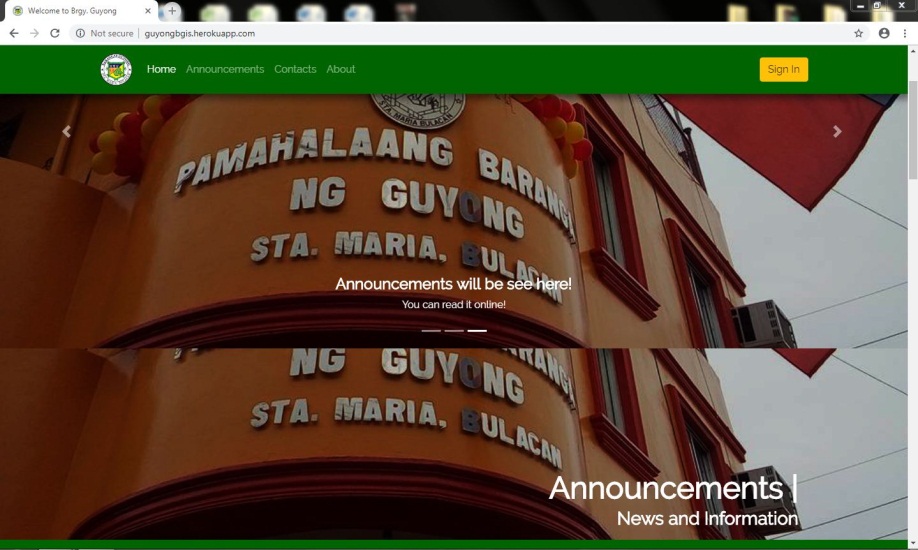
**APPENDIX Q.**

**PROGRAM FLOWCHART**

**APPENDIX R.**

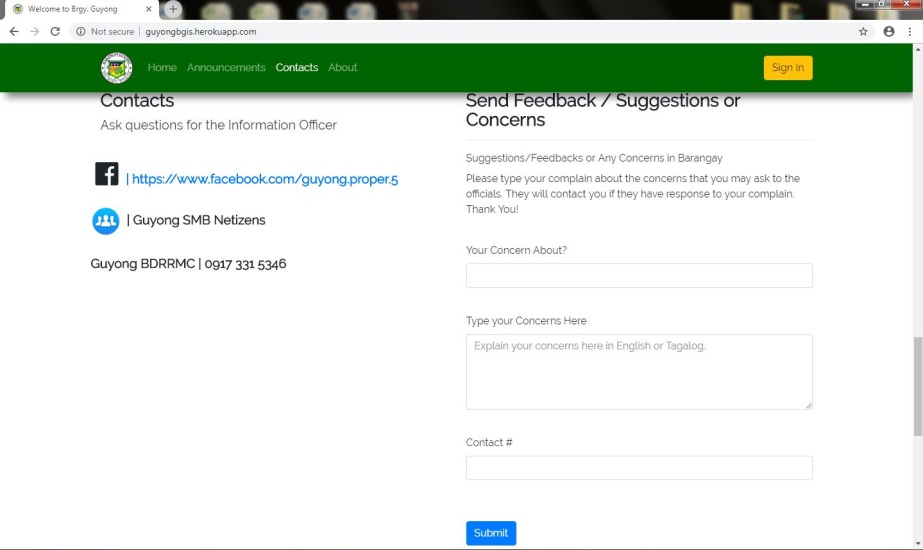
**USER GUIDE**

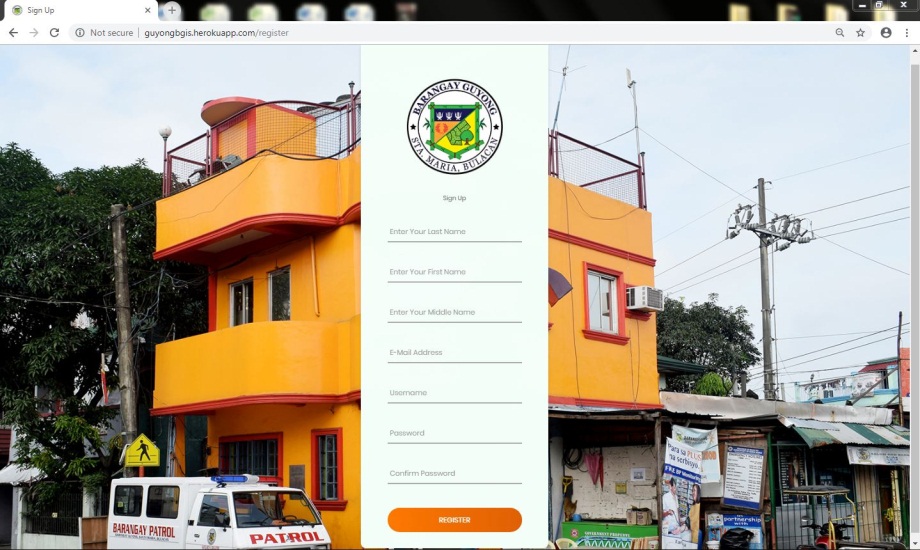
**Client Side**

****

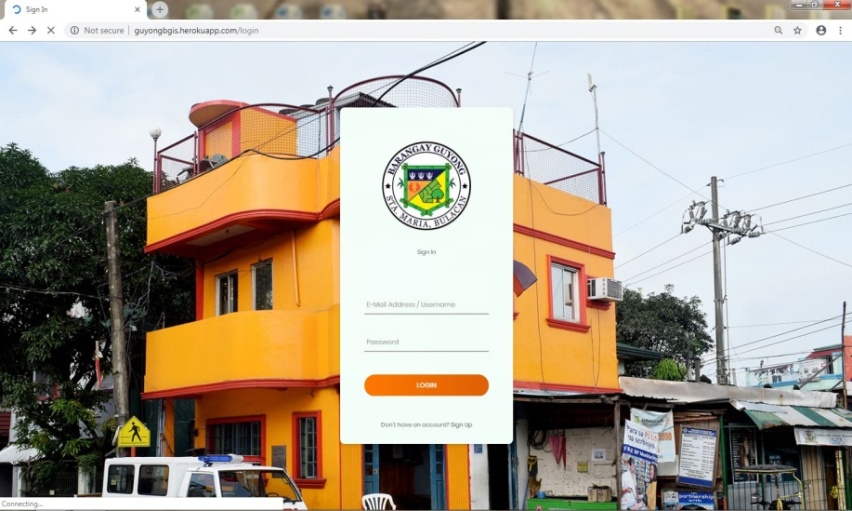
**Homepage** – The face of the website where Announcements, Contacts, About and Sign In can be seen.

**Contacts** –Serves as a way to communicate with the barangay.

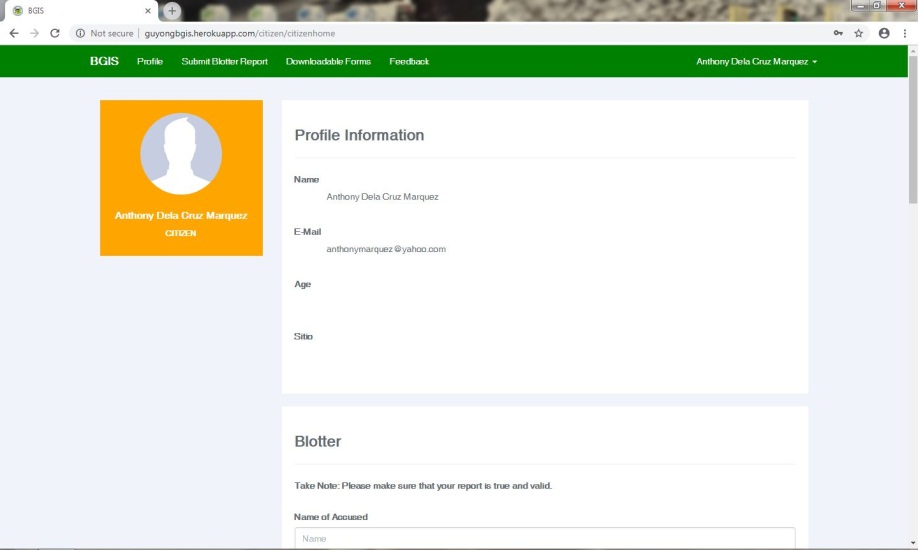
**Feedback** –A user can submit his/her concerns and opinions on any matter to the barangay.

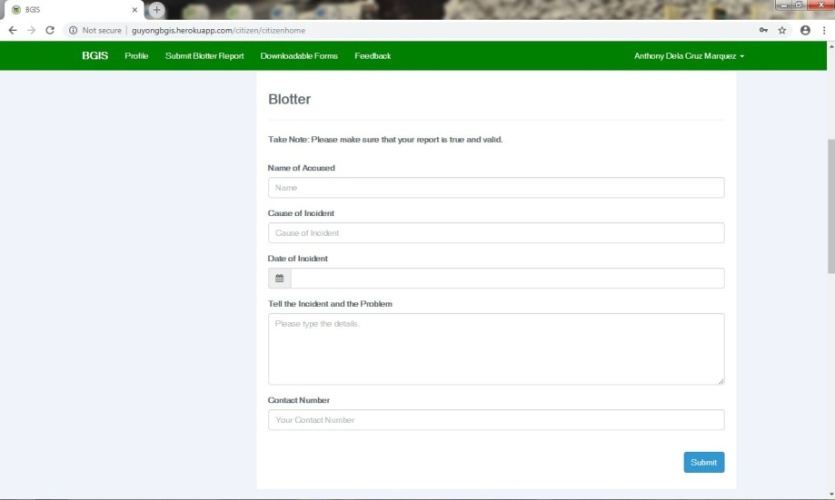
****

**Register** –This will enable the user to create an account and benefit from the website’s features. He/she needs to provide the necessary information for a successful registration.

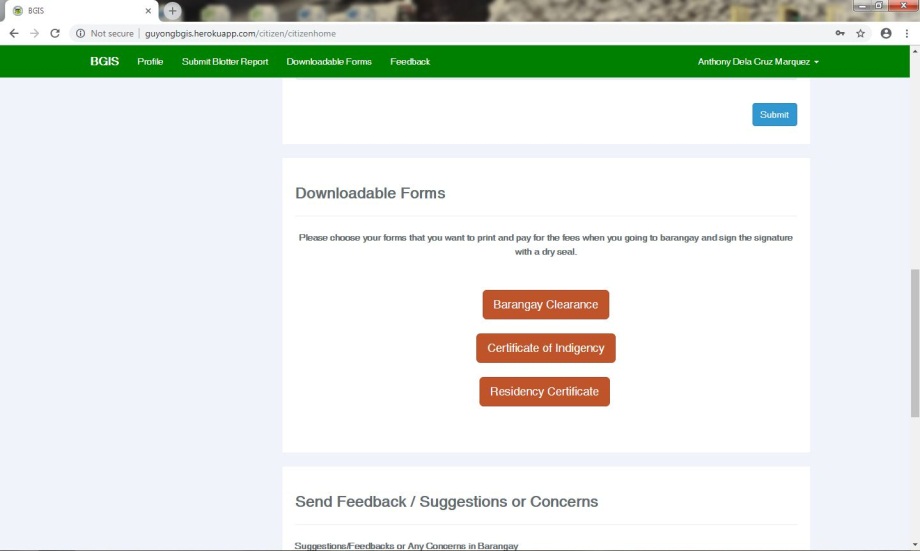
****

**Sign in** –A user will be able to access his/her registered account through here.

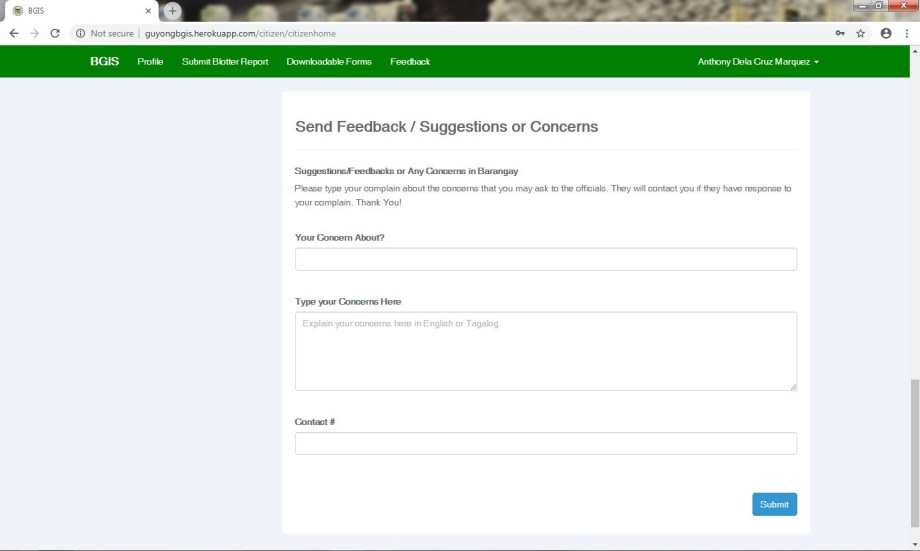
**Profile** – After logging in, the profile and other features are shown to the user. It includes Blotter Report, Downloadable Forms, Feedback and Log out.

****

**Blotter Report** –The registered user can deliver his/her concerns regarding an argument between other parties.

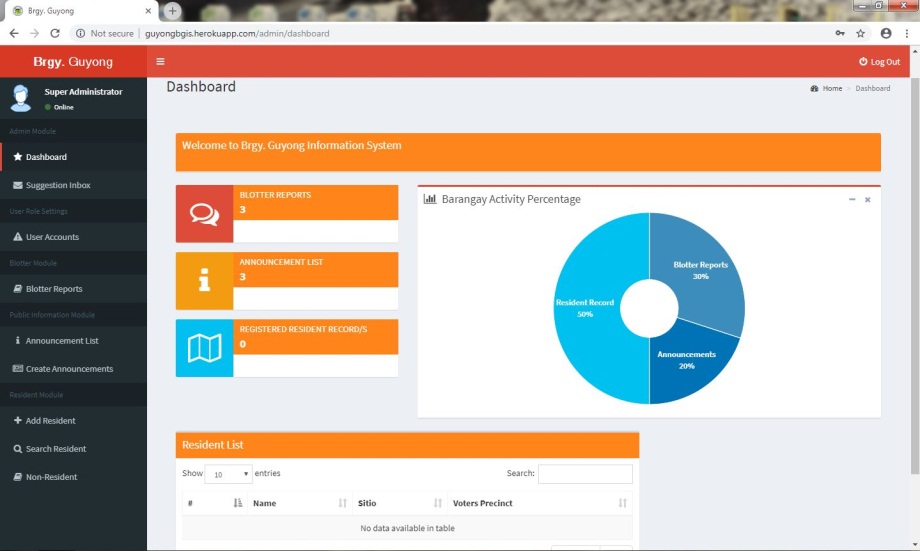
****

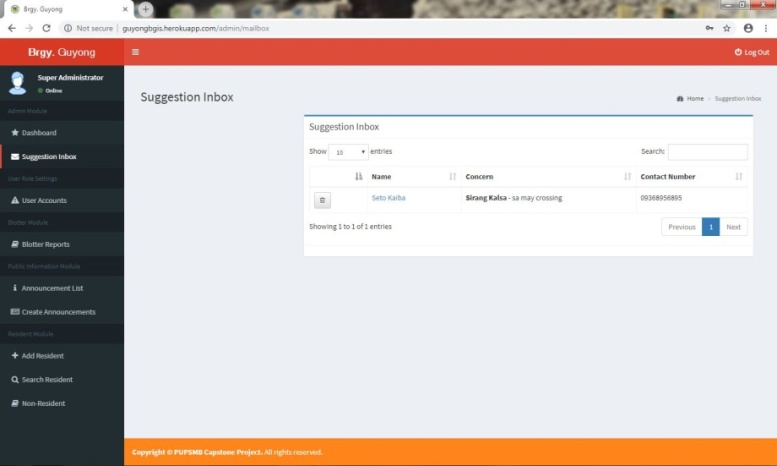
**Downloadable Forms**–Forms such as Barangay clearance, Barangay certification papers and Residency certificate are downloadable in the site.



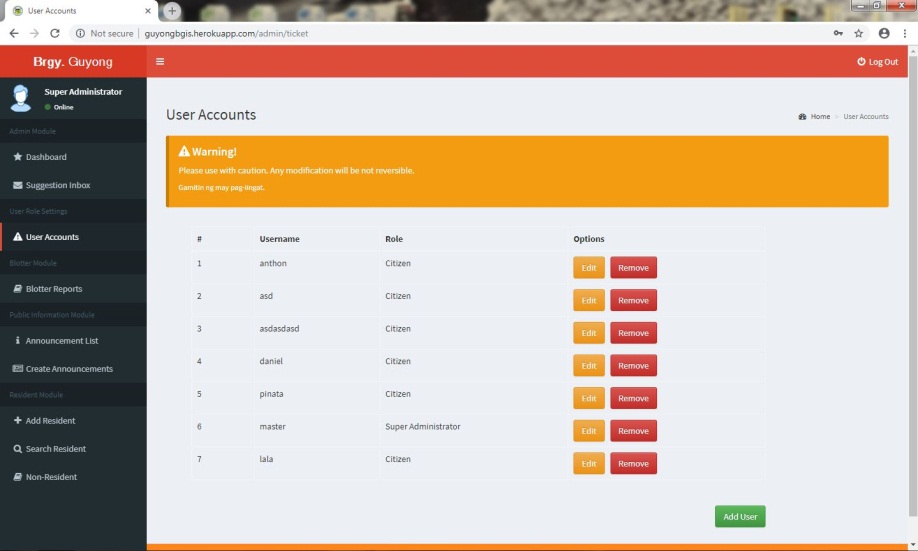
**Feedback** –A resident user can inform the barangay of their qualms about any matter that it can help resolve.

**Admin Side**

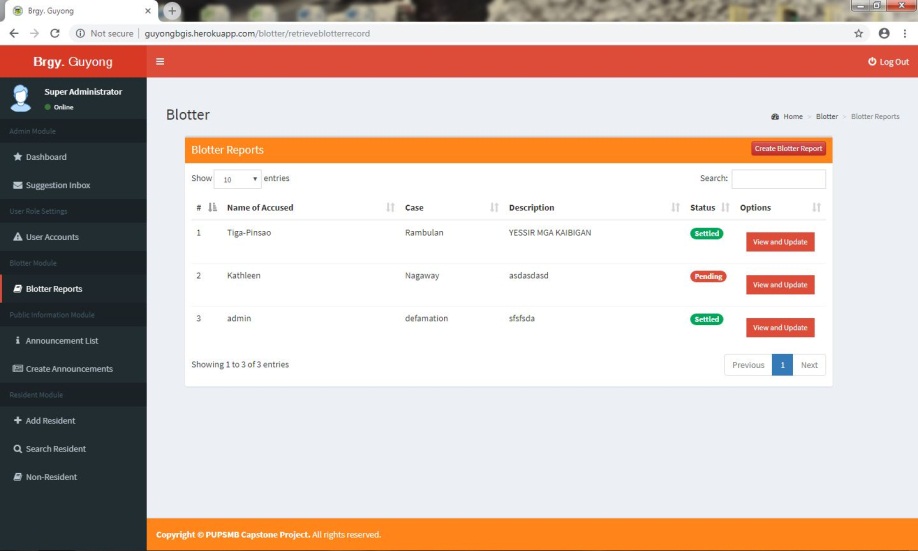
**Dashboard** –Once the admin logged in his/her account, he/she will now be able to use the system entirely.This shows an overview of the reports submitted by the users.

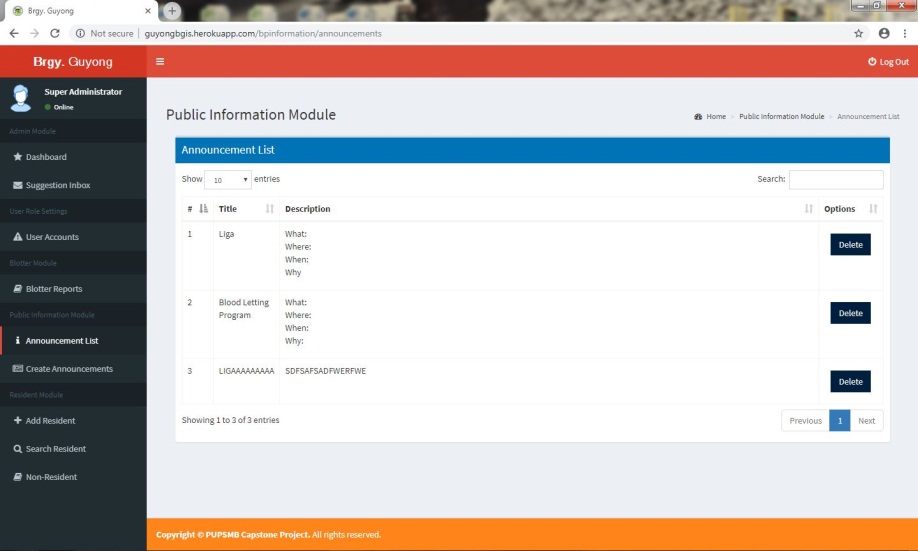


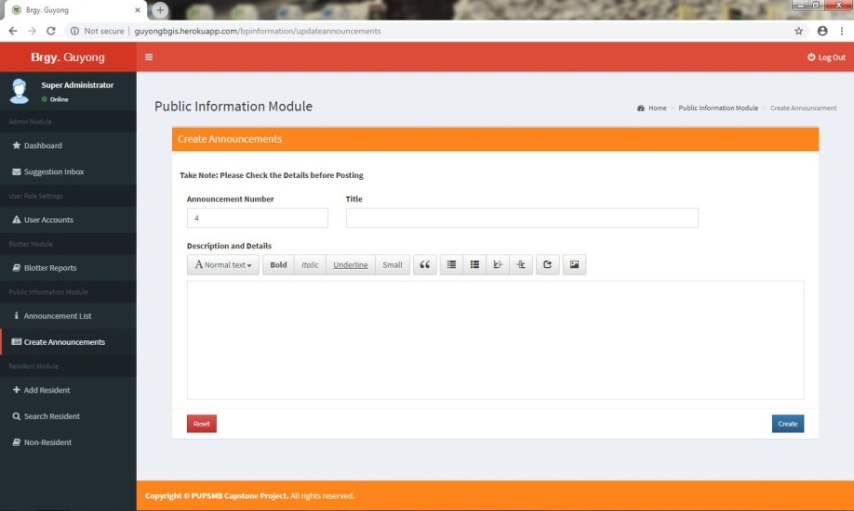
**Inbox** –Suggestions, concerns and comments submitted by the users are stored in here.



**User Accounts** –This is where the super admin can add and remove the other admins that can access the system.This is the user account management module.

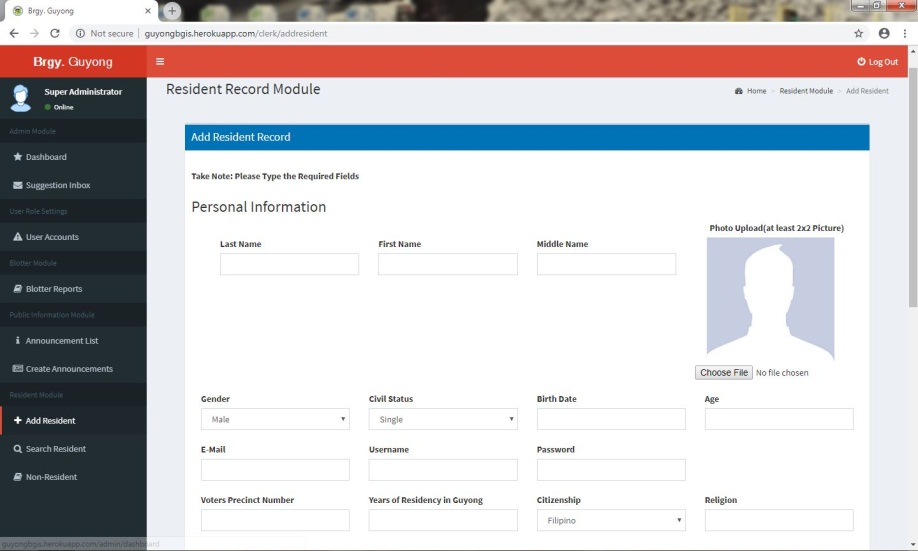


**Blotter Reports** – Complaints conveyed to the barangay are shown here sothat the admin can view and handle them easily.

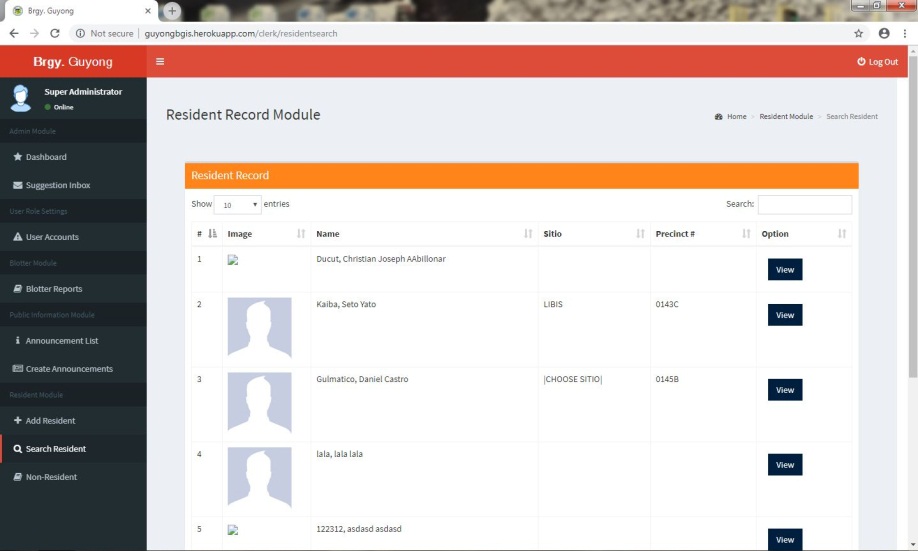


**Create Announcement**– The admin can announce any information the barangay needs to disseminate to its people.

**Announcement List**–This is where the created announcements can be seen and removed if needed.



**Add Resident**–This serves as the resident records of the barangay. Data such as personal and household information of every resident within the Barangay Guyong are inserted in here.

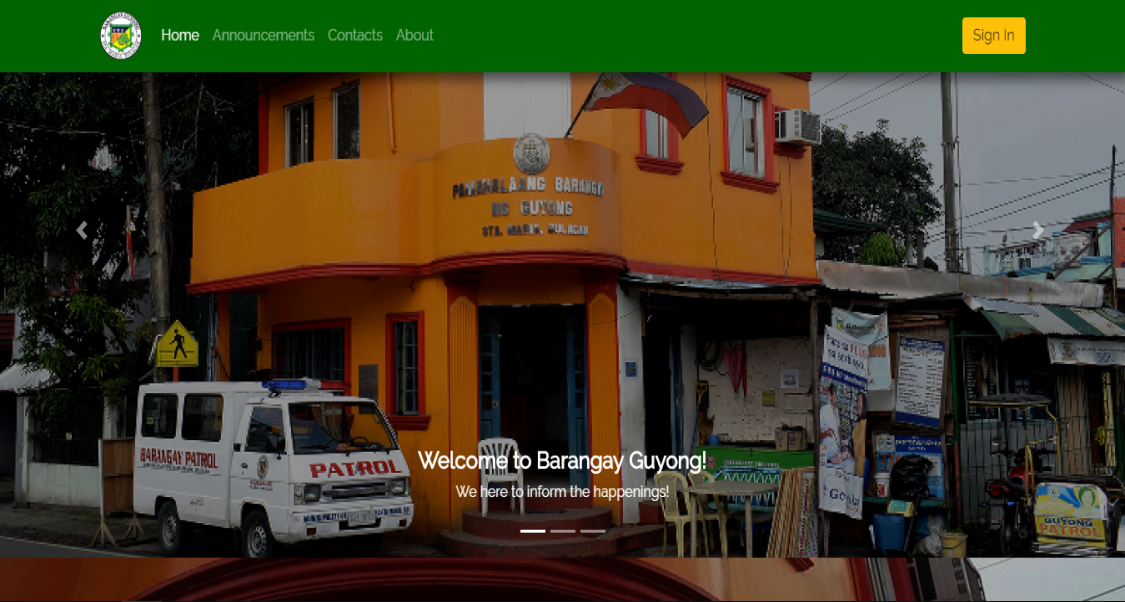


**Search Resident** –The residents’ information added in the above module are displayed here. The admin can search and view it.

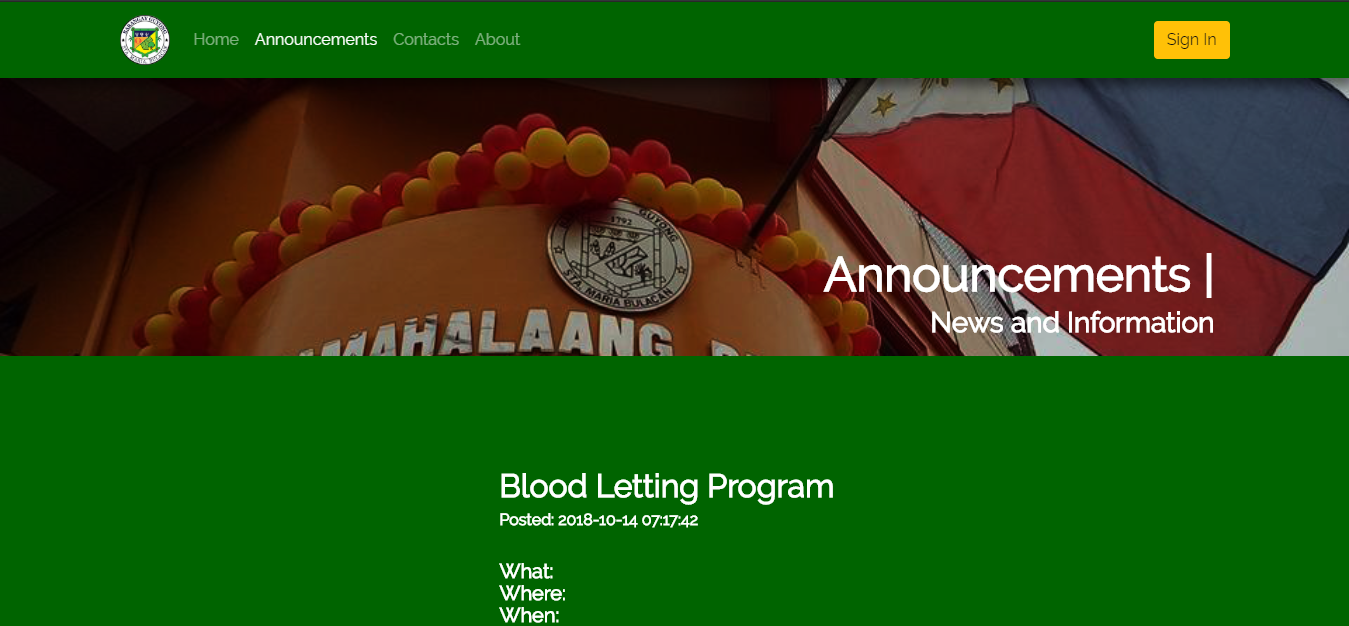
**APPENDIX S.**

**PROGRAM OUTPUT SNAPSHOTS**

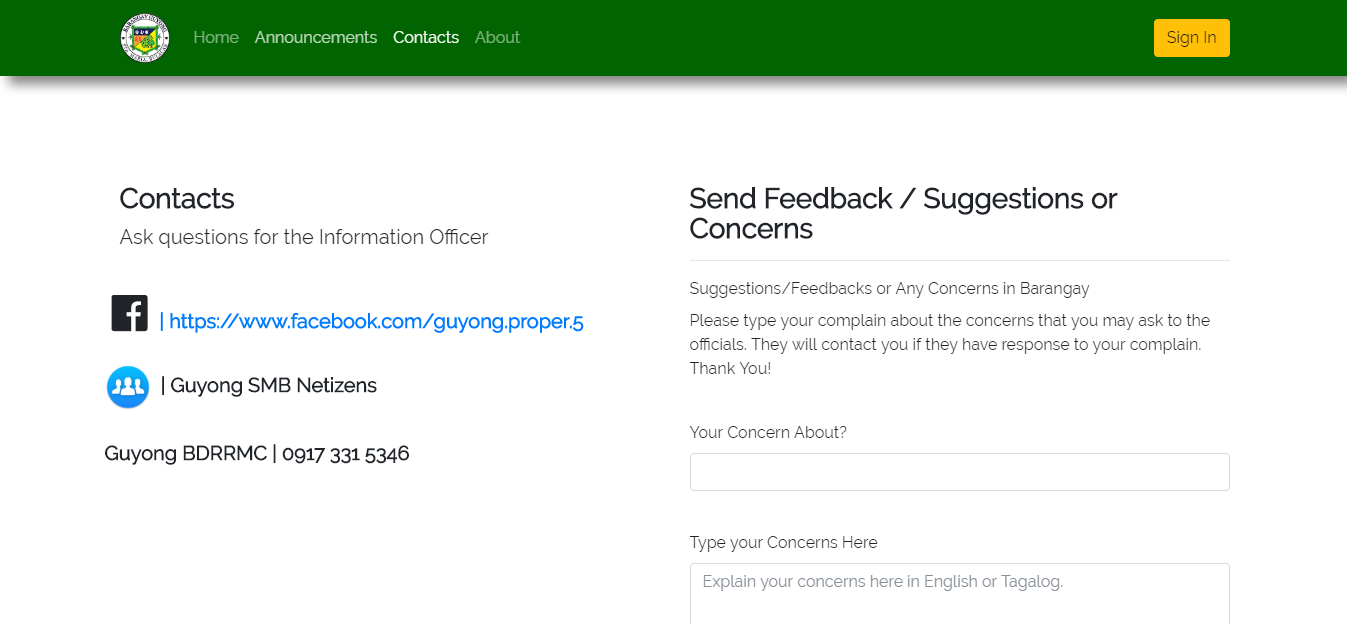
Homepage

****

Announcement List

****

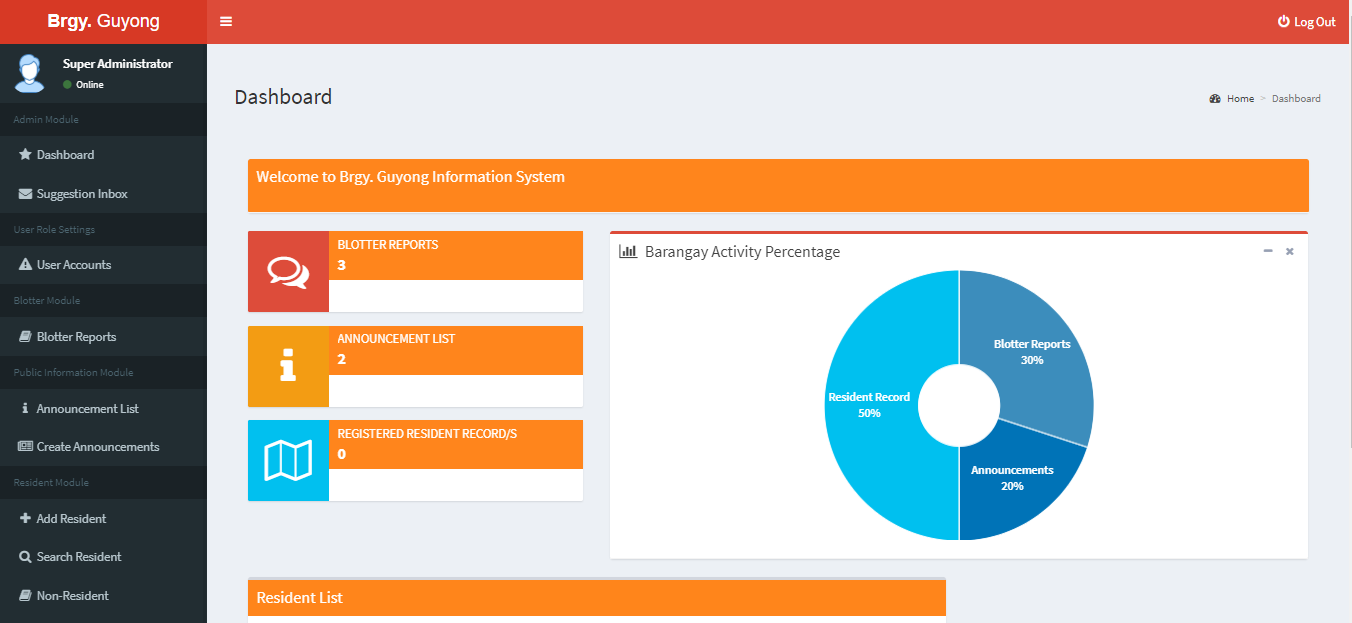
Contacts

****

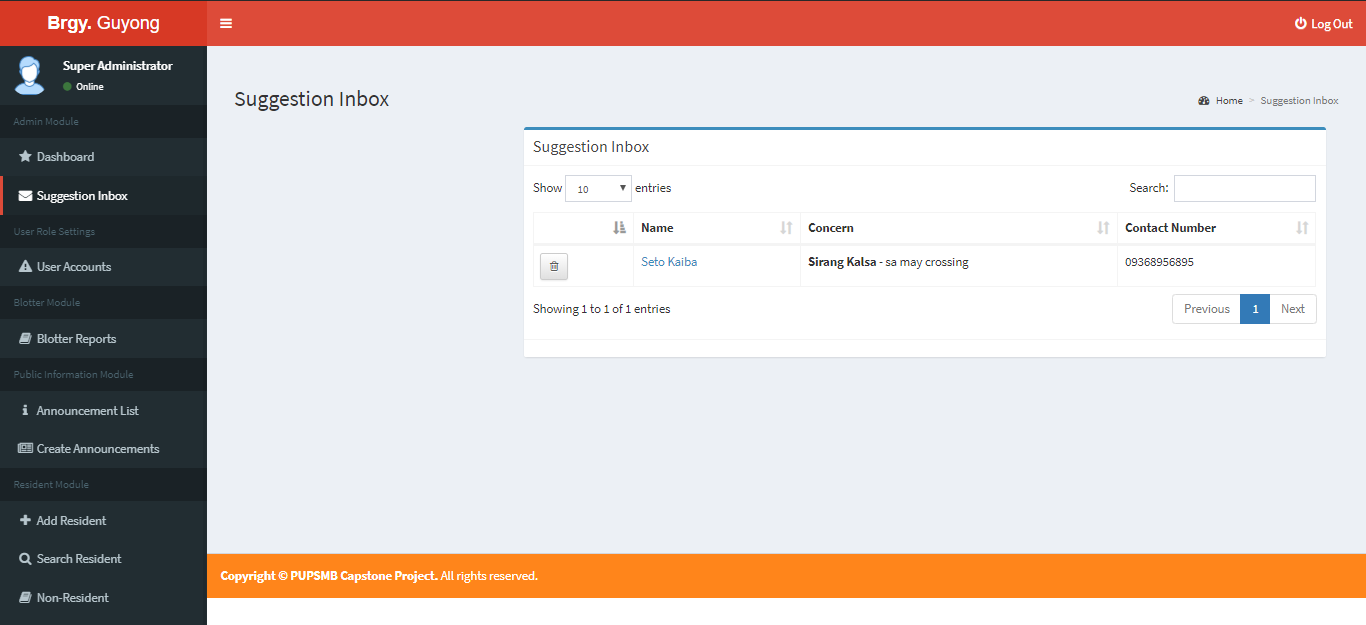
**About**

****

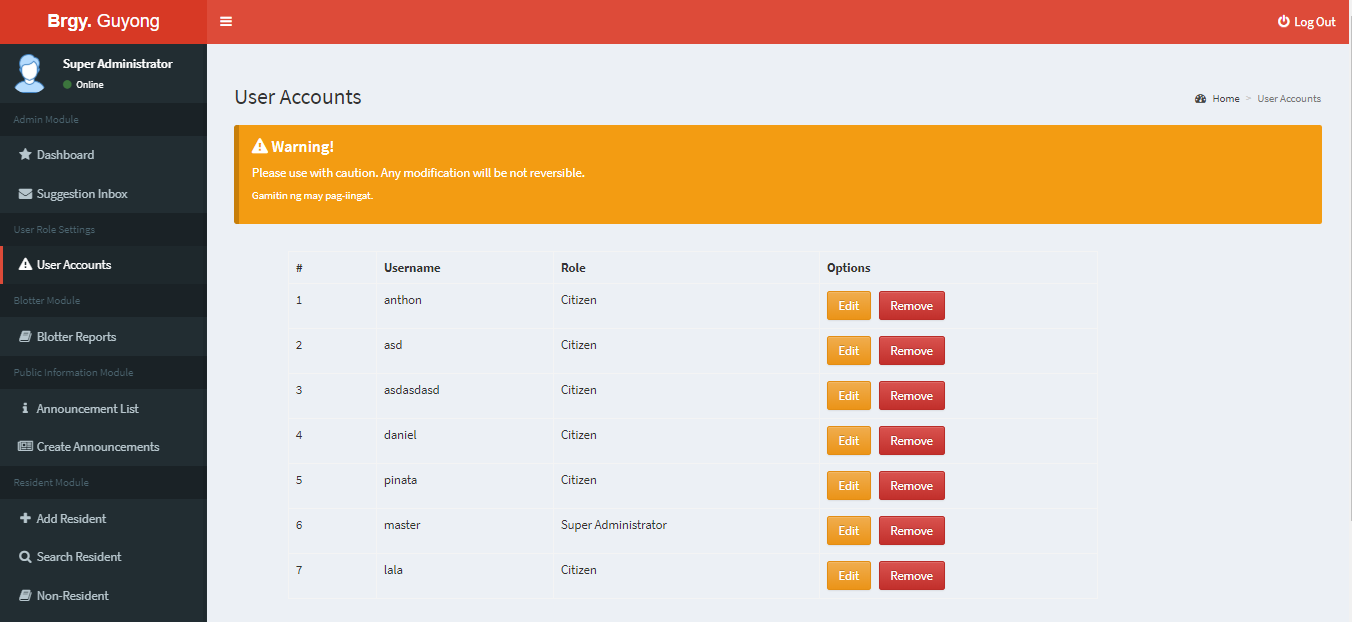
Dashboard



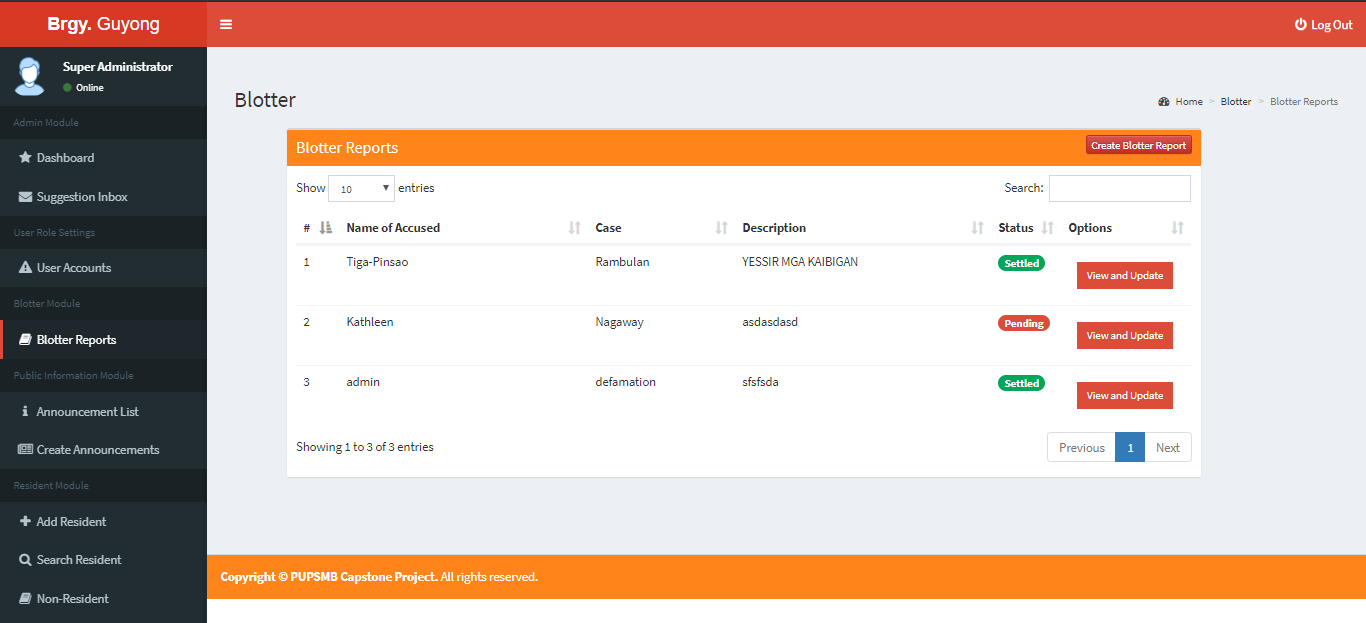
Suggestion Inbox

****

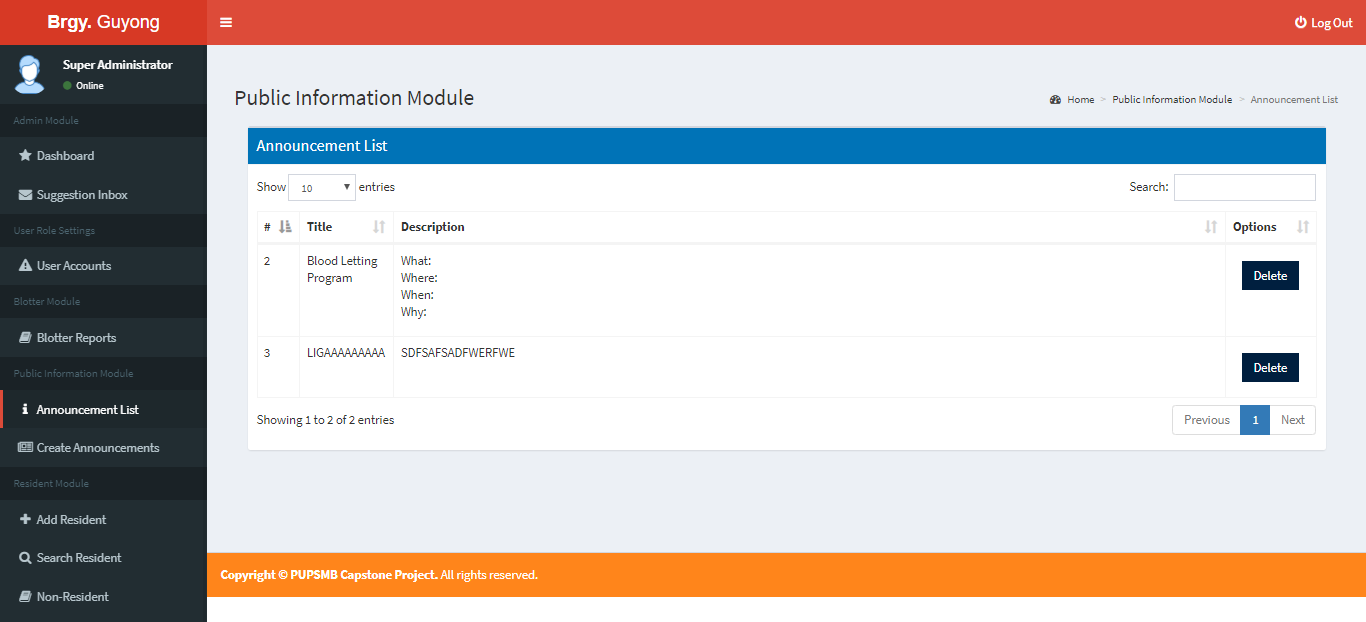
User Accounts



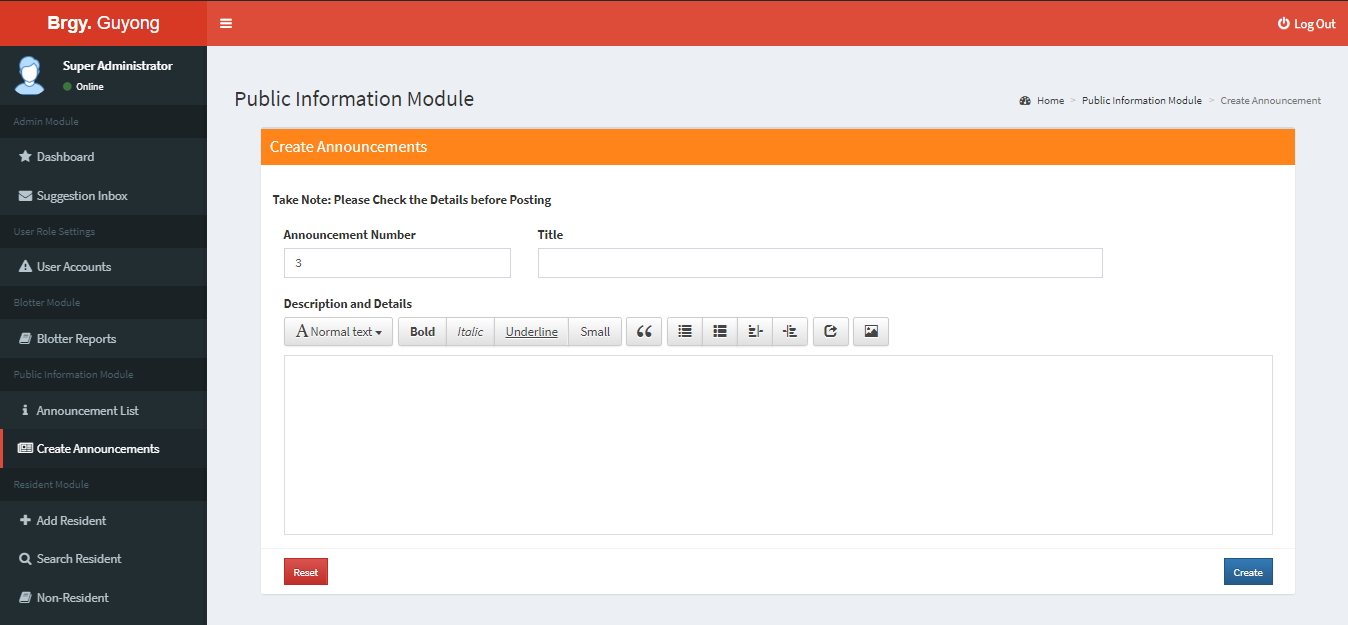
Blotter Reports



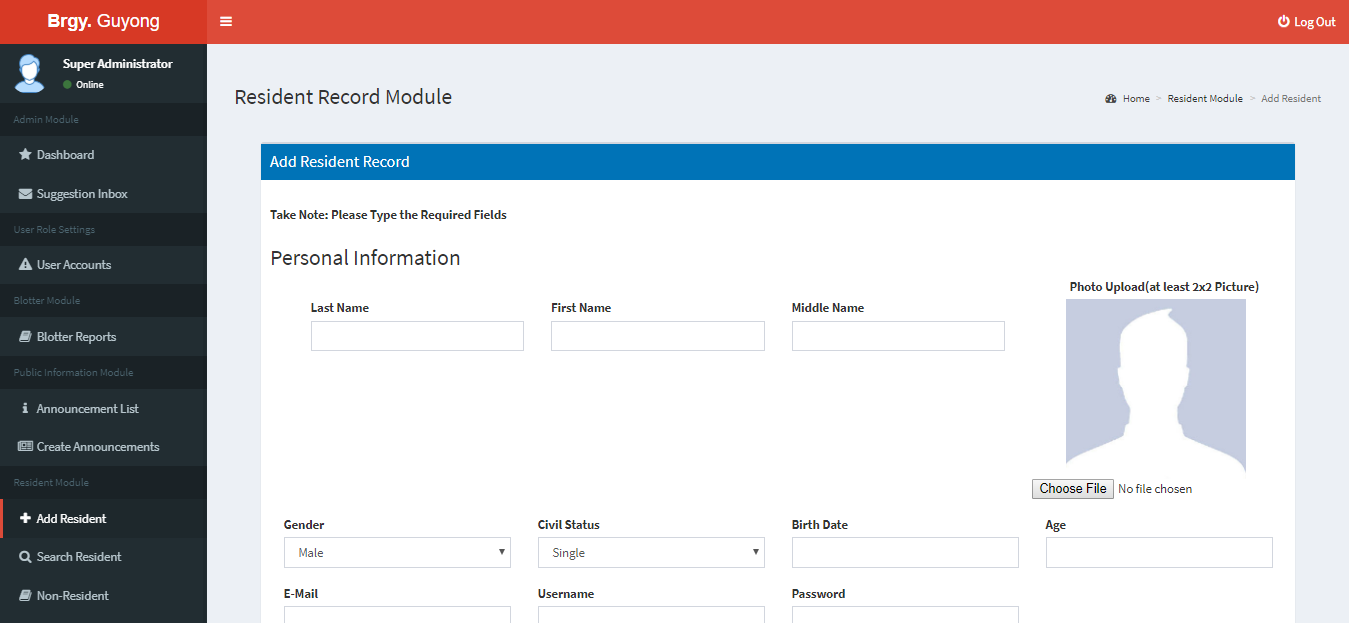
Announcement List

****

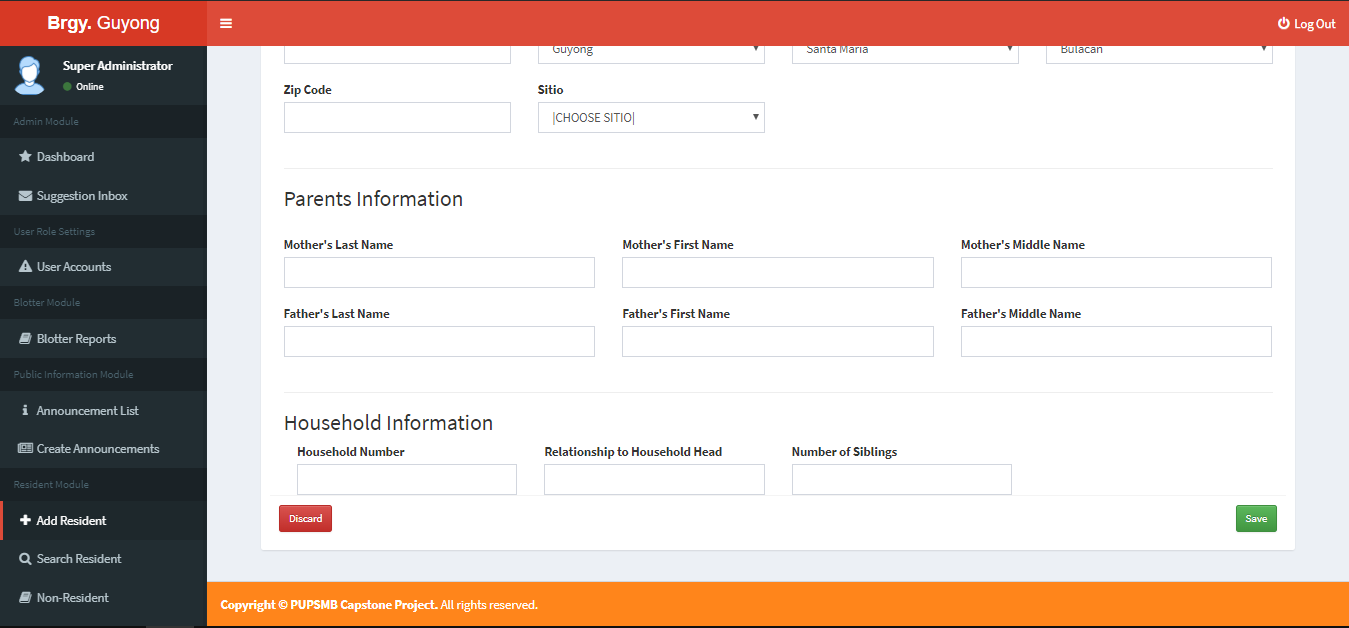
Create Announcement



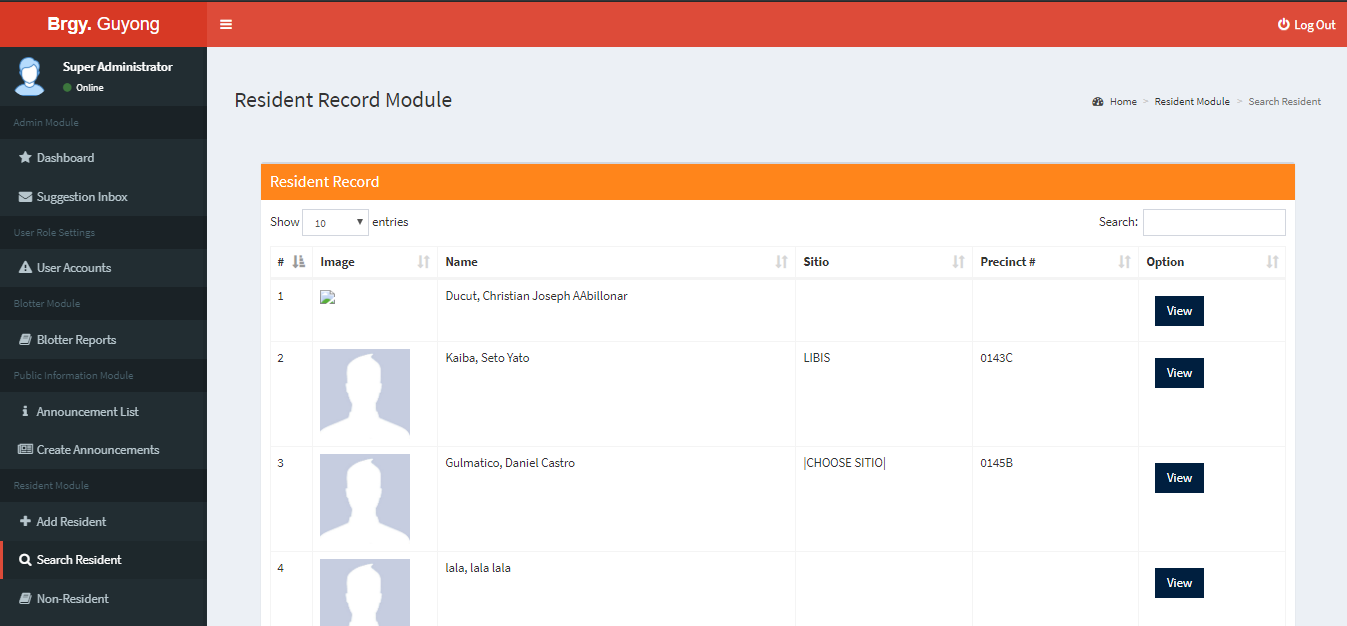
Resident Record Module



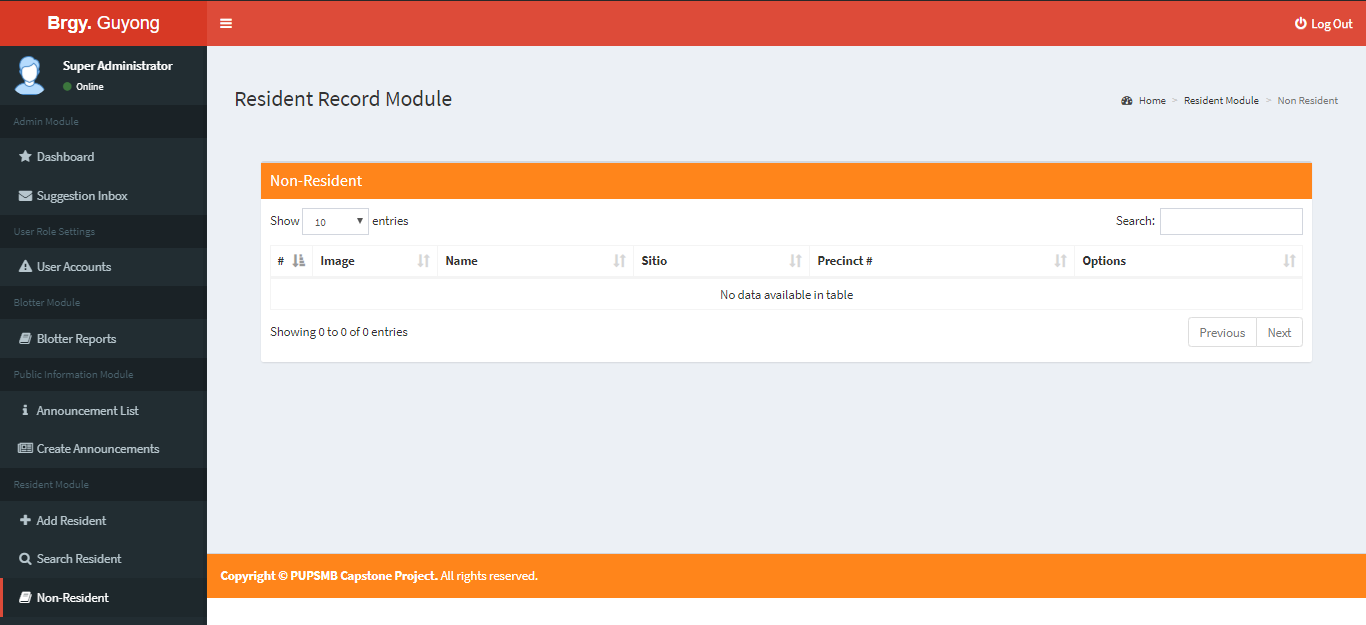




Search Resident



Non-resident



**APPENDIX T.**

**PROGRAM CODE LISTING**

**RegisterSuperAdminCommand.php**

<?php

namespace App\Console\Commands;

use App\User;

use Illuminate\Support\Facades\Hash;

use Illuminate\Console\Command;

class RegisterSuperAdminCommand extends Command

{

@var string

protected $signature = 'register:super-admin';

@var string

protected $description = 'Register super admin';

@return void

private $user;

public function \_\_construct(User $user)

{

parent::\_\_construct();

$this->user = $user;

}

@return mixed

public function handle()

{

$details = $this->getDetails();

$admin = $this->user->createSuperAdmin($details);

$this->display($admin);

}

@return array

private function getDetails() : array

{

$details['id'] = $this->ask('ID (Enter it #1)');

$details['lastname'] = $this->ask('Last Name');

$details['firstname'] = $this->ask('First Name');

$details['middlename'] = $this->ask('Middle Name');

$details['username'] = $this->ask('Username');

$details['email'] = $this->ask('Email');

$details['password'] = $this->secret('Password');

$details['confirm\_password'] = $this->secret('Confirm password');

$details['role'] = $this->ask('Role');

while (! $this->isValidPassword($details['password'], $details['confirm\_password'])) {

if (! $this->isRequiredLength($details['password'])) {

$this->error('Password must be more that six characters');

}

if (! $this->isMatch($details['password'], $details['confirm\_password'])) {

$this->error('Password and Confirm password do not match');

}

$details['password'] = $this->secret('Password');

$details['confirm\_password'] = $this->secret('Confirm password');

}

return $details;

}

@param array $admin

@return void

private function display(User $admin) : void

{

$headers = ['Name', 'Email', 'Super admin'];

$fields = [

'email' => $admin->email,

'admin' => $admin->isSuperAdmin()

];

$this->info('Super admin created');

$this->table($headers, [$fields]);

}

@param string $password

@param string $confirmPassword

@return boolean

private function isValidPassword(string $password, string $confirmPassword) : bool

{

return $this->isRequiredLength($password) &&

$this->isMatch($password, $confirmPassword);

}

@param string $password

@param string $confirmPassword

@return bool

private function isMatch(string $password, string $confirmPassword) : bool

{

return $password === $confirmPassword;

}

Checks if password is longer than six characters.

@param string $password

@return bool

private function isRequiredLength(string $password) : bool

{

return strlen($password) > 5;

}

}

**LoginController.php**

use AuthenticatesUsers;

Where to redirect users after login.

@var string

protected $redirectTo = '/home';

Login username to be used by the controller.

@var string

protected $username;

/\*\*

\* Create a new controller instance.

\*/

public function \_\_construct()

{

$this->middleware('guest')->except('logout');

$this->username = $this->findUsername();

}

/\*\*

\* Get the login username to be used by the controller.

@return string

\*/

public function findUsername()

{

$login = request()->input('login');

$fieldType = filter\_var($login, FILTER\_VALIDATE\_EMAIL) ? 'email' : 'username';

request()->merge([$fieldType => $login]);

return $fieldType;

}

Get username property.

@return string

public function username()

{

return $this->username;

}

}

**RegisterController.php**

<?php

namespace App\Http\Controllers\Auth;

use App\User;

use App\Resident;

use App\Http\Controllers\Controller;

use Illuminate\Support\Facades\Hash;

use Illuminate\Support\Facades\Validator;

use Illuminate\Http\Request;

use DB;

use Illuminate\Support\Facades\Auth;

use Illuminate\Auth\Events\Registered;

use Illuminate\Foundation\Auth\RegistersUsers;

class RegisterController extends Controller

{

use RegistersUsers;

@var string

protected $redirectTo = '/home';

@return void

public function \_\_construct()

{

$this->middleware('guest');

}

@param array $data

@return \Illuminate\Contracts\Validation\Validator

protected function validator(array $data)

{

return Validator::make($data, [

'id' => 'required|integer|max:25',

'lastname' => 'required|string|max:255',

'firstname' => 'required|string|max:255',

'middlename' => 'required|string|max:255',

'email' => 'required|string|email|max:255|unique:users',

'username' => 'required|string|max:255|unique:users',

'password' => 'required|string|min:6|confirmed',

'role' => 'required|string|max:255',

]);

}

@param array $data

@return \App\User

public function register(Request $request)

{

if ($request['role'] == "Super Administrator")

{

$notification = array(

'message' => 'Please try again later!',

'alert-type' => 'error');

return back()->with($notification);

}

$this->validator($request->all())->validate();

event(new Registered($user = $this->create($request->all())));

$this->guard()->login($user);

return $this->registered($request, $user)

?: redirect($this->redirectPath());

}

protected function create(array $data)

{

$pic = "images/steve.jpg";

return User::create([

'id' => $data['id'],

'lastname' => $data['lastname'],

'firstname' => $data['firstname'],

'middlename' => $data['middlename'],

'image' => $pic,

'email' => $data['email'],

'username' => $data['username'],

'password' => Hash::make($data['password']),

'role' => $data['role'],

]);

}

protected function guard()

{

return Auth::guard();

}

public function showRegistrationForm()

{

return view('auth.register');

}

public function index()

{

$user = User::count('id');

return view('auth.register',compact('user'));

}

}

**ResetPasswordController.php**

use ResetsPasswords;

@var string

protected $redirectTo = '/home'

@return void

public function \_\_construct()

{

$this->middleware('guest');

}

}

**AddResident.php**

<?php

use Illuminate\Http\Request;

use App\User;

use Illuminate\Support\Facades\Validator;

use Gate;

use Illuminate\Support\Facades\Hash;

use Illuminate\Validation\Rule;

use DB;

class AddResident extends Controller

{

public function \_\_construct()

{

$this->middleware('auth');

}

protected function validator(array $request)

{

return Validator::make($request, [

'lastname' => 'required|string|max:255',

'firstname' => 'required|string|max:255',

'middlename' => 'required|string|max:255',

'email' => 'required|string|max:255',

'gender' => 'required|string|max:255',

'civil' => 'required|string|max:255',

'image' => 'image|mimes:jpeg,png,jpg,svg',

'age' => 'required|string|max:255',

'birthdate' => 'required|string|max:255',

'votersno' => 'required|string|max:255',

'yearsres' => 'required|string|max:255',

'street' => 'required|string|max:255',

'barangay' => 'required|string|max:255',

'cityprovince' => 'required|string|max:255',

'province' => 'required|string|max:255',

'zipcode' => 'required|string|max:255',

'sitio' => 'required|string|max:255',

'mlast' => 'required|string|max:255',

'mfirst' => 'required|string|max:255',

'mmiddle' => 'required|string|max:255',

'flast' => 'required|string|max:255',

'ffirst' => 'required|string|max:255',

'fmiddle' => 'required|string|max:255',

'citizenship' => 'required|string|max:255',

'religion' => 'required|string|max:255',

'hno' => 'required|string|max:255',

'hrel' =>'required|string|max:255',

'sib' => 'required|string|max:255',

'username' => 'required|string|max:255|unique:users',

'password' => 'required|string|min:6|confirmed',

]);

}

public function index()

{

$resident=User::count('id');

return view('clerk.addresident',compact('resident'));

}

public function store(Request $request)

{

$role = "Citizen";

$file = $request->file('image');

$pic = "";

if($file == '' || $file == null){

$pic = "images/steve.jpg";

}

else

{

$date = date("Ymdhis");

$extension = $request->file('image')->getClientOriginalExtension();

$pic = "images/".$date.'.'.$extension;

$request->file('image')->move("images",$pic);

}

// $request->file('photo')->move(public\_path("/uploads"), $newfilename);

User::create([

'id' => $request['id'],

'lastname' => $request['lastname'],

'firstname' => $request['firstname'],

'middlename' => $request['middlename'],

'email' => $request['email'],

'image' => $pic,

'username' => $request['username'],

'password' => Hash::make($request['password']),

'gender' => $request['gender'],

'civil' => $request['civil'],

'age' => $request['age'],

'role' => $role,

'birthdate' => $request['birthdate'],

'votersno' => $request['votersno'],

'yearsres' => $request['yearsres'],

'street' => $request['street'],

'barangay' => $request['barangay'],

'cityprovince' => $request['cityprovince'],

'province' => $request['province'],

'zipcode' => $request['zipcode'],

'sitio' => $request['sitio'],

'mlast' => $request['mlast'],

'mfirst' => $request['mfirst'],

'mmiddle' => $request['mmiddle'],

'flast' => $request['flast'],

'ffirst' => $request['ffirst'],

'fmiddle' => $request['fmiddle'],

'citizenship' => $request['citizenship'],

'religion' => $request['religion'],

'hno' => $request['hno'],

'hrel' => $request['hrel'],

'sib' => $request['sib'],

]);

return back();

}

}

**AnnouncementList.php**

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;

use Gate;

use App\Announcements;

class AnnouncementList extends Controller

{

public function \_\_construct()

{

$this->middleware('auth');

}

protected function validator(array $request)

{

return Validator::make($request, [

'title' => 'required|string|max:255',

'description' => 'required|max:4098',

]);

}

public function index()

{

$announcementnum = Announcements::count('id');

$announcements = Announcements::all();

return view('announcements.announcements',compact('announcements','announcementnum'));

}

public function destroy(Request $request)

{

$announcements = Announcements::findorFail($request->id);

$announcements->delete([

'id' => $request['id'],

'title' => $request['title'],

'description' => $request['description'],

]);

return back();

}

}

**CitizenHome.php**

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;

use App\Feedback;

use App\CreateBlotter;

class CitizenHome extends Controller

{

public function \_\_construct()

{

$this->middleware('auth');

}

protected function validator(array $request)

{

return Validator::make($request, [

'name' => 'required|string|max:255',

'concerntitle' => 'required|string|max:255',

'concerndetail' => 'required|string|max:255',

'contactnum' => 'required|string|max:255',

]);

}

public function index()

{

toastr()->success('Successfully Logged In!');

$cid = Feedback::count('id');

$bid = CreateBlotter::count('id');

return view('citizen.citizenhome',compact('cid','bid'));

}

public function sendFeedback(Request $request)

{

Feedback::create([

'id' => $request['id'],

'name' => $request['name'],

'concerntitle' => $request['concerntitle'],

'concerndetail' => $request['concerndetail'],

'contactnum' => $request['contactnum'],

]);

toastr()->success('Success!');

return back();

}

public function blottercreate(Request $request)

{

CreateBlotter::create([

'id' => $request['id'],

'controlno' => $request['controlno'],

'accused' => $request['accused'],

'complainant' => $request['complainant'],

'contactnum' => $request['contactnum'],

'caseofincident' => $request['caseofincident'],

'datetime' => $request['datetime'],

'detail' => $request['detail'],

'status' => $request['status'],

]);

toastr()->success('Success!');

return back();

}

}

**CreateBlotterController.php**

class CreateBlotterController extends Controller

{

public function \_\_construct()

{

$this->middleware('auth');

}

protected function validator(array $request)

{

return Validator::make($request, [

'controlno' => 'required|string|max:255',

'accused' => 'required|string|max:255',

'complainant' => 'required|string|max:255',

'contactnum' => 'required|string|max:255',

'caseofincident' => 'required|string|max:255',

'datetime' => 'required|string|max:255',

'detail' => 'required|max:4098',

'status' => 'required|string|max:255',

]);

}

public function index()

{

$id = CreateBlotter::count('id');

$createblotter = CreateBlotter::all();

return view('blotter.createblotter',compact('createblotter','id'));

}

public function store(Request $request)

{

CreateBlotter::create([

'id' => $request['id'],

'controlno' => $request['controlno'],

'accused' => $request['accused'],

'complainant' => $request['complainant'],

'contactnum' => $request['contactnum'],

'caseofincident' => $request['caseofincident'],

'datetime' => $request['datetime'],

'detail' => $request['detail'],

'status' => $request['status'],

]);

toastr()->success('Success!');

$blotter = CreateBlotter::all();

return view('blotter.retrieve',compact('blotter'));

}

public function update(Request $request)

{

$blotter = CreateBlotter::all();

$blotter = CreateBlotter::findorFail($request->id);

$blotter->update([

'id' => $request['id'],

'controlno' => $request['controlno'],

'accused' => $request['accused'],

'caseofincident' => $request['caseofincident'],

'detail' => $request['detail'],

'status' => $request['status'],

]);

return back();

}

}

**HomeController.php**

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;

use Illuminate\Routing\Redirector;

use App\CreateBlotter;

use App\Resident;

use App\Announcements;

use Gate;

use Illuminate\Support\Facades\Auth;

class HomeController extends Controller

{

@return void

public function \_\_construct()

{

$this->middleware('auth');

}

@return \Illuminate\Http\Response

public function index()

{

if (Auth::user()->role == "Citizen")

{

return redirect('citizen/citizenhome');

}

toastr()->success('Successfully Logged In!');

$announcement = Announcements::count('id');

$blotter = CreateBlotter::count('id');

$resident = Resident::count('id');

$allres = Resident::all();

return view('admin.dashboard',compact('blotter','resident','announcement','allres'));

}

}

DatBackupController.php

<?php

class DataBackupController extends Controller

{

public function \_\_construct()

{

$this->middleware('auth');

}

public function index()

{

return view('errors.notallowed');

}

}

**NonResidentController.php**

class NonResidentController extends Controller

{

public function \_\_construct()

{

$this->middleware('auth');

}

public function index()

{

$nonre = NonResident::count('id');

$resident = NonResident::all();

return view('clerk.nonresident',compact('resident','nonre'));

}

public function store(Request $request)

{

}

public function destroy(Request $request)

{

$non = NonResident::findorFail($request->id);

$file = $request->file('image');

$pic = "images/steve.jpg";

$non->delete([

'id' => $request['id'],

'lastname' => $request['lastname'],

'firstname' => $request['firstname'],

'middlename' => $request['middlename'],

'email' => $request['email'],

'image' => $pic,

'gender' => $request['gender'],

'civil' => $request['civil'],

'age' => $request['age'],

'birthdate' => $request['birthdate'],

'votersno' => $request['votersno'],

'yearsres' => $request['yearsres'],

'street' => $request['street'],

'barangay' => $request['barangay'],

'cityprovince' => $request['cityprovince'],

'province' => $request['province'],

'zipcode' => $request['zipcode'],

'sitio' => $request['sitio'],

'mlast' => $request['mlast'],

'mfirst' => $request['mfirst'],

'mmiddle' => $request['mmiddle'],

'flast' => $request['flast'],

'ffirst' => $request['ffirst'],

'fmiddle' => $request['fmiddle'],

'citizenship' => $request['citizenship'],

'religion' => $request['religion'],

'hno' => $request['hno'],

'hrel' => $request['hrel'],

'sib' => $request['sib'],

]);

Resident::create([

'id' => $request['id'],

'lastname' => $request['lastname'],

'firstname' => $request['firstname'],

'middlename' => $request['middlename'],

'email' => $request['email'],

'image' => $pic,

'gender' => $request['gender'],

'civil' => $request['civil'],

'age' => $request['age'],

'birthdate' => $request['birthdate'],

'votersno' => $request['votersno'],

'yearsres' => $request['yearsres'],

'street' => $request['street'],

'barangay' => $request['barangay'],

'cityprovince' => $request['cityprovince'],

'province' => $request['province'],

'zipcode' => $request['zipcode'],

'sitio' => $request['sitio'],

'mlast' => $request['mlast'],

'mfirst' => $request['mfirst'],

'mmiddle' => $request['mmiddle'],

'flast' => $request['flast'],

'ffirst' => $request['ffirst'],

'fmiddle' => $request['fmiddle'],

'citizenship' => $request['citizenship']

'religion' => $request['religion'],

'hno' => $request['hno'],

'hrel' => $request['hrel'],

'sib' => $request['sib'],

]);

toastr()->success('Success!');

return back();

}

}

**pdfController.php**

class PdfController extends Controller

{

public function index(Request $request)

{

$gender = DB::table('forms')->select('gendertitle')->value('gendertitle');

$name = DB::table('forms')->select('name')->value('name');

$street = DB::table('forms')->select('street')->value('street');

$civil = DB::table('forms')->select('civil')->value('civil');

$purpose = DB::table('forms')->select('purpose')->value('purpose');

$date = DB::table('forms')->select('dateissued')->value('dateissued');

$pdf = DomPDF::loadview('Forms.BarangayClearance',compact('gender','name','street','civil','purpose','date'));

$pdf->SetPaper('letter','portrait');

FormsModel::truncate();

return $pdf->stream();

}

public function resi(Request $request)

{

$gender = DB::table('forms')->select('gendertitle')->value('gendertitle');

$name = DB::table('forms')->select('name')->value('name');

$street = DB::table('forms')->select('street')->value('street');

$reason1 = DB::table('forms')->select('reason1')->value('reason1');

$reason2 = DB::table('forms')->select('reason2')->value('reason2');

$date = DB::table('forms')->select('dateissued')->value('dateissued');

$pdf = DomPDF::loadview('Forms.Indigency',compact('gender','name','street','reason1','reason2','date'));

$pdf->SetPaper('letter','portrait');

FormsModel::truncate();

return $pdf->stream();

}

public function resi2(Request $request)

{

$gender = DB::table('forms')->select('gendertitle')->value('gendertitle');

$name = DB::table('forms')->select('name')->value('name');

$age = DB::table('forms')->select('age')->value('age');

$street = DB::table('forms')->select('street')->value('street');

$civil = DB::table('forms')->select('civil')->value('civil');

$purpose = DB::table('forms')->select('purpose')->value('purpose');

$date = DB::table('forms')->select('dateissued')->value('dateissued');

$pdf = DomPDF::loadview('Forms.Residency',compact('gender','name','age','street','civil','purpose','date'));

$pdf->SetPaper('letter','portrait');

return $pdf->stream();

}

}

**RegNon.php**

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;

use App\NonResident;

use Illuminate\Support\Facades\Validator;

use Gate;

use Illuminate\Validation\Rule;

use DB;

class RegNon extends Controller

{

public function index()

{

$resident=NonResident::count('id');

return view('auth.regnon',compact('resident'));

}

public function store(Request $request)

{

$file = $request->file('image');

$pic = "";

if($file == '' || $file == null){

$pic = "images/steve.jpg";

}

else

{

$date = date("Ymdhis");

$extension = $request->file('image')->getClientOriginalExtension();

$pic = "images/".$date.'.'.$extension;

$request->file('image')->move("images",$pic);

}

NonResident::create([

'id' => $request['id'],

'lastname' => $request['lastname'],

'firstname' => $request['firstname'],

'middlename' => $request['middlename'],

'email' => $request['email'],

'image' => $pic,

'gender' => $request['gender'],

'civil' => $request['civil'],

'age' => $request['age'],

'birthdate' => $request['birthdate'],

'votersno' => $request['votersno'],

'yearsres' => $request['yearsres'],

'street' => $request['street'],

'barangay' => $request['barangay'],

'cityprovince' => $request['cityprovince'],

'province' => $request['province'],

'zipcode' => $request['zipcode'],

'sitio' => $request['sitio'],

'mlast' => $request['mlast'],

'mfirst' => $request['mfirst'],

'mmiddle' => $request['mmiddle'],

'flast' => $request['flast'],

'ffirst' => $request['ffirst'],

'fmiddle' => $request['fmiddle'],

'citizenship' => $request['citizenship'],

'religion' => $request['religion'],

'hno' => $request['hno'],

'hrel' => $request['hrel'],

'sib' => $request['sib'],

]);

return view('auth.login');

}

}

**ResidentSearch.php**

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;

use Gate;

use App\User;

class ResidentSearch extends Controller

{

public function \_\_construct()

{

$this->middleware('auth');

}

public function index()

{

$resident = User::all();

return view('clerk.residentsearch',compact('resident'));

}

}

**RetrieveBlotter.php**

{

public function \_\_construct()

{

$this->middleware('auth');

}

public function index()

{

$blotter = CreateBlotter::all();

return view('blotter.retrieve',compact('blotter'));

}

}

**TicketController.php**

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;

use \App\Ticket;

use \App\User;

use App\Http\Controllers\Controller;

use Illuminate\Support\Facades\Hash;

use Illuminate\Support\Facades\Validator;

use Gate;

class TicketController extends Controller

{

//

public function \_\_construct()

{

$this->middleware('auth');

}

protected function validator(array $request)

{

return Validator::make($request, [

'lastname' => 'required|string|max:255',

'firstname' => 'required|string|max:255',

'middlename' => 'required|string|max:255',

'email' => 'required|string|email|max:255|unique:users',

'username' => 'required|string|max:255|unique:users',

'password' => 'required|string|min:6|confirmed',

'role' => 'required|string|max:255',

]);

}

public function index()

{

if (!Gate::allows('isSAdmin')){

return view('errors.notallowed');

}

$ticket = User::all();

return view('admin.ticket',compact('ticket'));

}

public function create()

{

}

public function store (Request $request)

{

User::create([

'id' => $request['id'],

'email' => $request['email'],

'lastname' => $request['lastname'],

'firstname' => $request['firstname'],

'middlename' => $request['middlename'],

'username' => $request['username'],

'password' => Hash::make($request['password']),

'role' => $request['role'],

]);

return back();

}

public function update(Request $request)

{

$ticket = User::findorFail($request->edit\_id);

$ticket->update([

'id' => $request['id'],

'email' => $request['email'],

'lastname' => $request['name'],

'username' => $request['username'],

'password' => Hash::make($request['password']),

'role' => $request['role'],

]);

return back();

}

public function show($id)

{

}

public function destroy(Request $request)

{

$ticket = User::findorFail($request->edit\_id);

$ticket->delete([

'id' => $request['id'],

'email' => $request['email'],

'lastname' => $request['name'],

'username' => $request['username'],

'password' => Hash::make($request['password']),

'role' => $request['role'],

]);

return back();

}

}

**UpdateAnnouncement.php**

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;

use App\Announcements;

use Gate;

class UpdateAnnouncement extends Controller

{

public function \_\_construct()

{

$this->middleware('auth');

}

protected function validator(array $request)

{

return Validator::make($request, [

'title' => 'required|string|max:255',

'description' => 'required|max:4098',

]);

}

public function index()

{

$id = Announcements::count('id');

return view('announcements.updateannouncement',compact('id'));

}

public function store(Request $request)

{

Announcements::create([

'id' => $request['id'],

'title' => $request['title'],

'description' => $request['description']

]);

return back();

}

}

**Welcome.php**

<?php

class Welcome extends Controller

{

public function index()

{

$cid = Feedback::count('id');

$announcements = Announcements::all();

return view('welcome',compact('announcements','cid'));

}

public function send(Request $request)

{

Feedback::create([

'id' => $request['id'],

'name' => $request['name'],

'concerntitle' => $request['concerntitle'],

'concerndetail' => $request['concerndetail'],

'contactnum' => $request['contactnum'],

]);

toastr()->success('Success!');

return back();

}

}

Announcements.php

<?php

namespace App;

class Announcements extends Model

{

protected $table = 'announcements';

protected $fillable = ['id','title','description'];

}

BarangayClearanceModel.php

<?php

namespace App;

class BarangayClearanceModel extends Model

{

protected $table = 'barangayclearance';

protected $fillable=['gender','name','street', 'civil','purpose','dateissued'];

}

**CreateBlotter.php**

<?php

namespace App;

class CreateBlotter extends Model

{

protected $table = 'create\_blotters';

protected $fillable = ['id','controlno', 'accused', 'complainant' ,'contactnum','caseofincident','datetime','detail','status'];

}

**Feedback.php**

<?php

namespace App;

class Feedback extends Model

{

protected $table = 'feedback';

protected $fillable = ['id','name','contactnum','concerntitle','concerndetail'];

}

**FormsModel.php**

<?php

namespace App;

class FormsModel extends Model

{

protected $table = 'forms';

protected $fillable = ['gendertitle','name','street','civil','purpose','dateissued','reason1','reason2','age'];

}

**NonResident.php**

<?php

namespace App;

class NonResident extends Model

{

protected $table = 'nonresident';

protected $fillable = ['id','lastname','firstname','middlename','email','image','gender','civil','age','birthdate','votersno','yearsres','street','barangay','cityprovince','province','zipcode','sitio','mlast','mfirst','mmiddle','flast','ffirst','fmiddle','citizenship','religion','hno','hrel','sib'];

}

Resident.php

<?php

namespace App;

class Resident extends Model

{

protected $table = 'resident';

protected $fillable = ['id','lastname','firstname','middlename','email','image','gender','civil','age','birthdate','votersno','yearsres','street','barangay','cityprovince','province','zipcode','sitio','mlast','mfirst','mmiddle','flast','ffirst','fmiddle','citizenship','religion','hno','hrel','sib'];

}

Ticket.php

<?php

namespace App;

class Ticket extends Model

{

protected $fillable = [

'id','email','name', 'username', 'password','role',

];

@var array

protected $hidden = ['password', 'remember\_token',];

}

User.php

<?php

namespace App;

class User extends Authenticatable

{

protected $fillable = [

'id','email','username', 'password','role', 'lastname','firstname','middlename','email','image','gender','civil','age','birthdate','votersno','yearsres','street','barangay','

cityprovince','province','zipcode','sitio','mlast','mfirst','mmiddle','flast','ffirst','fmiddle','citizenship','religion','hno','hrel','sib',

];

protected $hidden = ['password', 'remember\_token',];

@return boolean

public function isSuperAdmin() : bool

{

return (bool) $this->is\_super\_admin;

}

public function createSuperAdmin(array $details) : self

{

$user = new self($details);

if (! $this->superAdminExists()) {

$user->is\_super\_admin = 1;

}

$user->password = Hash::make($details['password']);

$user->save();

return $user;

}

public function superAdminExists() : int

{

return self::where('is\_super\_admin', 1)->count();

}

}

Adminlte.php

<?php

return [

'title' => 'Brgy. Guyong',

'title\_prefix' => '',

'title\_postfix' => '',

'logo' => '<b>Brgy.</b> Guyong',

'logo\_mini' => '<b>B</b>G',

'layout' => 'fixed',

'collapse\_sidebar' => false,

'dashboard\_url' => 'admin/dashboard',

'logout\_url' => 'logout',

'logout\_method' => null,

'login\_url' => 'login',

'register\_url' => 'register',

'menu' => [

[

'header' => 'Admin Module',

],

[

'text' => 'Dashboard',

'url' => 'admin/dashboard',

'icon' => 'star',

],

[

'text' => 'Suggestion Inbox',

'url' => 'admin/mailbox',

'icon' => 'envelope',

],

[

'header' => 'User Role Settings',

'can' => 'isSAdmin',

],

[

'text' => 'User Accounts',

'url' => 'admin/ticket',

'icon' => 'warning',

'can' => 'isSAdmin',

],

[

'header' => 'Blotter Module',

],

[

'text' => 'Blotter Reports',

'url' => 'blotter/retrieveblotterrecord','blotter/createblotter','blotter/updateblotterrecord',

'icon' => 'book',

],

[

'text' => 'Database Backup/Restore',

'url' => 'admin/datasync',

'icon' => 'refresh',

'can' => 'isUser',

],

[

'header' => 'Public Information Module',

],

[

'text' => 'Announcement List',

'url' => 'bpinformation/announcements',

'icon' => 'info',

],

[

'text' => 'Create Announcements',

'url' => 'bpinformation/updateannouncements',

'icon' => 'newspaper-o',

],

[

'header' => 'Resident Module',

],

[

'text' => 'Add Resident',

'url' => 'clerk/addresident',

'icon' => 'plus',

],

[

'text' => 'Search Resident',

'url' => 'clerk/residentsearch',

'icon' => 'search',

],

[

'text' => 'Non-Resident',

'url' => 'clerk/nonresident',

'icon' => 'book',

],

],

'plugins' => [

'datatables' => true,

'select2' => true,

'chartjs' => true,

],

];

Auth.php

<?php

return [

'defaults' => [

'guard' => 'web',

'passwords' => 'users'],

'guards' => [

'web' => [

'driver' => 'session',

'provider' => 'users',],

'api' => [

'driver' => 'token',

'provider' => 'users',],

],

'providers' => [

'users' => [

'driver' => 'eloquent',

'model' => App\User::class,],

'users' => [

'driver' => 'database',

'table' => 'users',

],

],

'passwords' => [

'users' => [

'provider' => 'users',

'table' => 'password\_resets',

'expire' => 60,

],

],

];

Broadcasting.php

<?php

return [

'default' => env('BROADCAST\_DRIVER', 'null'),

'connections' => [

'pusher' => [

'driver' => 'pusher',

'key' => env('PUSHER\_APP\_KEY'),

'secret' => env('PUSHER\_APP\_SECRET'),

'app\_id' => env('PUSHER\_APP\_ID'),

'options' => [

'cluster' => env('PUSHER\_APP\_CLUSTER'),

'encrypted' => true,

]],

'redis' => [

'driver' => 'redis',

'connection' => 'default'],

'log' => [

'driver' => 'log',

],

'null' => [

'driver' => 'null',

]

]];

Cache.php

<?php

return [

/\*

'default' => env('CACHE\_DRIVER', 'file'),

'stores' => [

'apc' => [

'driver' => 'apc'],

'array' => [

'driver' => 'array'],

'database' => [

'driver' => 'database',

'table' => 'cache',

'connection' => null,],

'file' => [

'driver' => 'file',

'path' => storage\_path('framework/cache/data'),

],

'memcached' => [

'driver' => 'memcached',

'persistent\_id' => env('MEMCACHED\_PERSISTENT\_ID'),

'sasl' => [

env('MEMCACHED\_USERNAME'),

env('MEMCACHED\_PASSWORD')],

'options' => [

// Memcached::OPT\_CONNECT\_TIMEOUT => 2000,

],

'servers' => [

[

'host' => env('MEMCACHED\_HOST', '127.0.0.1'),

'port' => env('MEMCACHED\_PORT', 11211),

'weight' => 100,

]]],

'redis' => [

'driver' => 'redis',

'connection' => 'default',

]],

'prefix' => env(

'CACHE\_PREFIX',

str\_slug(env('APP\_NAME', 'laravel'), '\_').'\_cache)];

database.php

<?php

return [

'default' => env('DB\_CONNECTION', 'mysql'),

'connections' => [

'sqlite' => [

'driver' => 'sqlite',

'database' => env('DB\_DATABASE', database\_path('database.sqlite')),

'prefix' => '',

],

'mysql' => [

'driver' => 'mysql',

'host' => env('DB\_HOST', '127.0.0.1'),

'port' => env('DB\_PORT', '3306'),

'database' => env('DB\_DATABASE', 'forge'),

'username' => env('DB\_USERNAME', 'forge'),

'password' => env('DB\_PASSWORD', ''),

'unix\_socket' => env('DB\_SOCKET', ''),

'charset' => 'utf8mb4',

'collation' => 'utf8mb4\_unicode\_ci',

'prefix' => '',

'strict' => true,

'engine' => null],

'pgsql' => [

'driver' => 'pgsql',

'host' => env('DB\_HOST', '127.0.0.1'),

'port' => env('DB\_PORT', '5432'),

'database' => env('DB\_DATABASE', 'forge'),

'username' => env('DB\_USERNAME', 'forge'),

'password' => env('DB\_PASSWORD', ''),

'charset' => 'utf8',

'prefix' => '',

'schema' => 'public',

'sslmode' => 'prefer',

]

'sqlsrv' => [

'driver' => 'sqlsrv',

'host' => env('DB\_HOST', 'localhost'),

'port' => env('DB\_PORT', '1433'),

'database' => env('DB\_DATABASE', 'forge'),

'username' => env('DB\_USERNAME', 'forge'),

'password' => env('DB\_PASSWORD', ''),

'charset' => 'utf8',

'prefix' => '',

]],

'migrations' => 'migrations',

'redis' => [

'client' => 'predis',

'default' => [

'host' => env('REDIS\_HOST', '127.0.0.1'),

'password' => env('REDIS\_PASSWORD', null),

'port' => env('REDIS\_PORT', 6379),

'database' => 0,

]]];

Dompdf.php

<?php

return array(

'orientation' => 'portrait',

'defines' => array(

"font\_dir" => storage\_path('fonts/'),

"font\_cache" => storage\_path('fonts/'),

"temp\_dir" => sys\_get\_temp\_dir(),

"chroot" => realpath(base\_path()),

"enable\_font\_subsetting" => false,

"pdf\_backend" => "CPDF",

"default\_media\_type" => "screen",

"default\_paper\_size" => "a4",

"default\_font" => "serif",

"dpi" => 96,

"enable\_php" => false,

"enable\_javascript" => true,

"enable\_remote" => true,

"font\_height\_ratio" => 1.1,

"enable\_html5\_parser" => false,

));

Filesystems.php

<?php

return [

'default' => env('FILESYSTEM\_DRIVER', 'local'),

'cloud' => env('FILESYSTEM\_CLOUD', 's3'),

'disks' => [

'local' => [

'driver' => 'local',

'root' => storage\_path('app')],

'public' => [

'driver' => 'local',

'root' => storage\_path('app/public'),

'url' => env('APP\_URL').'/storage',

'visibility' => 'public'],

's3' => [

'driver' => 's3',

'key' => env('AWS\_ACCESS\_KEY\_ID'),

'secret' => env('AWS\_SECRET\_ACCESS\_KEY'),

'region' => env('AWS\_DEFAULT\_REGION'),

'bucket' => env('AWS\_BUCKET'),

'url' => env('AWS\_URL'),

]]];

Hashing.php

<?php

return [

'driver' => 'bcrypt',

'bcrypt' => [

'rounds' => env('BCRYPT\_ROUNDS', 10)],

'argon' => [

'memory' => 1024,

'threads' => 2,

'time' => 2,

]];

Logging.php

<?php

return [

'default' => env('LOG\_CHANNEL', 'stack'),

'channels' => [

'stack' => [

'driver' => 'stack',

'channels' => ['single']],

'single' => [

'driver' => 'single',

'path' => storage\_path('logs/laravel.log'),

'level' => 'debug'],

'daily' => [

'driver' => 'daily',

'path' => storage\_path('logs/laravel.log'),

'level' => 'debug',

'days' => 7],

'slack' => [

'driver' => 'slack',

'url' => env('LOG\_SLACK\_WEBHOOK\_URL'),

'username' => 'Laravel Log',

'emoji' => ':boom:',

'level' => 'critical',],

'stderr' => [

'driver' => 'monolog',

'handler' => StreamHandler::class,

'with' => [

'stream' => 'php://stderr]],

'syslog' => [

'driver' => 'syslog',

'level' => 'debug',],

'errorlog' => [

'driver' => 'errorlog',

'level' => 'debug',

Mail.php

<?php

return [

'driver' => env('MAIL\_DRIVER', 'smtp'),

'host' => env('MAIL\_HOST', 'smtp.mailgun.org'),

'port' => env('MAIL\_PORT', 587),

'from' => [

'address' => env('MAIL\_FROM\_ADDRESS', 'hello@example.com'),

'name' => env('MAIL\_FROM\_NAME', 'Example')],

'encryption' => env('MAIL\_ENCRYPTION', 'tls'),

'username' => env('MAIL\_USERNAME'),

'password' => env('MAIL\_PASSWORD'),

'sendmail' => '/usr/sbin/sendmail -bs',

'markdown' => [

'theme' => 'default',

'paths' => [

resource\_path('views/vendor/mail'),

]]];

Queue.php

<?php

return [

'default' => env('QUEUE\_DRIVER', 'sync'),

'connections' => [

'sync' => [

'driver' => 'sync',

],

'database' => [

'driver' => 'database',

'table' => 'jobs',

'queue' => 'default',

'retry\_after' => 90,

],

'beanstalkd' => [

'driver' => 'beanstalkd',

'host' => 'localhost',

'queue' => 'default',

'retry\_after' => 90],

'sqs' => [

'driver' => 'sqs',

'key' => env('SQS\_KEY', 'your-public-key'),

'secret' => env('SQS\_SECRET', 'your-secret-key'),

'prefix' => env('SQS\_PREFIX', 'https://sqs.us-east-1.amazonaws.com/your-account-id'),

'queue' => env('SQS\_QUEUE', 'your-queue-name'),

'region' => env('SQS\_REGION', 'us-east-1'),

],

'redis' => [

'driver' => 'redis',

'connection' => 'default',

'queue' => 'default',

'retry\_after' => 90,

'block\_for' => null,

]],

'failed' => [

'database' => env('DB\_CONNECTION', 'mysql'),

'table' => 'failed\_jobs',

]];

Services.php

<?php

return [

'mailgun' => [

'domain' => env('MAILGUN\_DOMAIN'),

'secret' => env('MAILGUN\_SECRET')],

'ses' => [

'key' => env('SES\_KEY'),

'secret' => env('SES\_SECRET'),

'region' => env('SES\_REGION', 'us-east-1')],

'sparkpost' => [

'secret' => env('SPARKPOST\_SECRET')],

'stripe' => [

'model' => App\User::class,

'key' => env('STRIPE\_KEY'),

'secret' => env('STRIPE\_SECRET'),

]];

Toastr.php

<?php

return [

'options' => [

'closeButton' => true,

'closeClass' => 'toast-close-button',

'closeDuration' => 300,

'closeEasing' => 'swing',

'closeHtml' => '<button><i class="icon-off"></i></button>',

'closeMethod' => 'fadeOut',

'closeOnHover' => true,

'containerId' => 'toast-container',

'debug' => false,

'escapeHtml' => false,

'extendedTimeOut' => 4000,

'hideDuration' => 1000,

'hideEasing' => 'linear',

'hideMethod' => 'fadeOut',

'iconClass' => 'toast-info',

'iconClasses' => [

'error' => 'toast-error',

'info' => 'toast-info',

'success' => 'toast-success',

'warning' => 'toast-warning',

],

'messageClass' => 'toast-message',

'newestOnTop' => false,

'onHidden' => null,

'onShown' => null,

'positionClass' => 'toast-top-right',

'preventDuplicates' => true,

'progressBar' => true,

'progressClass' => 'toast-progress',

'rtl' => false,

'showDuration' => 300,

'showEasing' => 'swing',

'showMethod' => 'fadeIn',

'tapToDismiss' => true,

'target' => 'body',

'timeOut' => 4000,

'titleClass' => 'toast-title',

'toastClass' => 'toast',

]];

Views.php

<?php

return [

'paths' => [

resource\_path('views'),

],

'compiled' => realpath(storage\_path('framework/views'))];

UserFactory.php

<?php

use Faker\Generator as Faker;

$factory->define(App\User::class, function (Faker $faker) {

return [

'name' => $faker->name,

'email' => $faker->unique()->safeEmail,

'password' => '$2y$10$TKh8H1.PfQx37YgCzwiKb.KjNyWgaHb9cbcoQgdIVFlYg7B77UdFm',

'remember\_token' => str\_random(10),

]

});

Create\_users\_table.php

class CreateUsersTable extends Migration

{

public function up()

{

Schema::create('users', function (Blueprint $table) {

$table->integer('id')->unsigned();

$table->string('lastname');

$table->string('firstname');

$table->string('middlename');

$table->string('image')->nullable();

$table->string('email')->unique();

$table->string('username')->unique();

$table->string('password');

$table->string('role')->nullable();$table->string('gender')->nullable();

$table->string('civil')->nullable();

$table->string('age')->nullable();

$table->string('birthdate')->nullable();

$table->string('votersno')->nullable();

$table->string('yearsres')->nullable();

$table->string('street')->nullable();

$table->string('barangay')->nullable();

$table->string('cityprovince')->nullable();

$table->string('province')->nullable();

$table->string('zipcode')->nullable();

$table->string('sitio')->nullable();

$table->string('mlast')->nullable();

$table->string('mfirst')->nullable();

$table->string('mmiddle')->nullable();

$table->string('flast')->nullable();

$table->string('ffirst')->nullable();

$table->string('fmiddle')->nullable();

$table->string('citizenship')->nullable();

$table->string('religion')->nullable();

$table->string('hno')->nullable();

$table->string('hrel')->nullable();

$table->string('sib')->nullable();

$table->unsignedTinyInteger('is\_super\_admin')->default(0);

$table->rememberToken();

$table->timestamps();

});

}

public function down()

{

Schema::dropIfExists('users');

}}

Create\_password\_resets\_table.php

<?php

class CreatePasswordResetsTable extends Migration

{

public function up()

{

Schema::create('password\_resets', function (Blueprint $table) {

$table->string('username');

$table->string('token');

$table->timestamp('created\_at')->nullable();

});

}

public function down()

{

Schema::dropIfExists('password\_resets');

}}

Laratrust\_setup\_tables.php

<?php

class LaratrustSetupTables extends Migration

{

public function up()

{

Schema::create('roles', function (Blueprint $table) {

$table->increments('id');

$table->string('name')->unique();

$table->string('display\_name')->nullable();

$table->string('description')->nullable();

$table->timestamps();

});

Schema::create('permissions', function (Blueprint $table) {

$table->increments('id');

$table->string('name')->unique();

$table->string('display\_name')->nullable();

$table->string('description')->nullable();

$table->timestamps();

});

Schema::create('role\_user', function (Blueprint $table) {

$table->unsignedInteger('role\_id');

$table->unsignedInteger('user\_id');

$table->string('user\_type');

$table->foreign('role\_id')->references('id')->on('roles')

->onUpdate('cascade')->onDelete('cascade');

$table->primary(['user\_id', 'role\_id', 'user\_type']);

});

Schema::create('permission\_user', function (Blueprint $table) {

$table->unsignedInteger('permission\_id');

$table->unsignedInteger('user\_id');

$table->string('user\_type');

$table->foreign('permission\_id')->references('id')->on('permissions')

->onUpdate('cascade')->onDelete('cascade');

$table->primary(['user\_id', 'permission\_id', 'user\_type']);

});

Schema::create('permission\_role', function (Blueprint $table) {

$table->unsignedInteger('permission\_id');

$table->unsignedInteger('role\_id');

$table->foreign('permission\_id')->references('id')->on('permissions')

->onUpdate('cascade')->onDelete('cascade');

$table->foreign('role\_id')->references('id')->on('roles')

->onUpdate('cascade')->onDelete('cascade');

$table->primary(['permission\_id', 'role\_id']);

});

}

public function down()

{

Schema::dropIfExists('permission\_user');

Schema::dropIfExists('permission\_role');

Schema::dropIfExists('permissions');

Schema::dropIfExists('role\_user');

Schema::dropIfExists('roles');

}}

CreateBlotter.php

<?php

class CreateBlotter extends Migration

{

public function up()

{

Schema::create('create\_blotters', function (Blueprint $table) {

$table->increments('id');

$table->string('controlno')->nullable();

$table->string('accused')->nullable();

$table->string('contactnum')->nullable();

$table->string('complainant');

$table->string('datetime');

$table->string('caseofincident');

$table->text('detail');

$table->string('status');

$table->rememberToken();

$table->timestamps();

});}

2018\_09\_18\_033651\_resident.php

<?php

class Resident extends Migration

{

public function up()

{

Schema::create('resident', function (Blueprint $table) {

$table->increments('id');

$table->string('lastname')->nullable();

$table->string('firstname')->nullable();

$table->string('middlename')->nullable();

$table->string('email')->nullable();

$table->string('image')->nullable();

$table->string('gender')->nullable();

$table->string('civil')->nullable();

$table->string('age')->nullable();

$table->string('birthdate')->nullable();

$table->string('votersno')->nullable();

$table->string('yearsres')->nullable();

$table->string('street')->nullable();

$table->string('barangay')->nullable();

$table->string('cityprovince')->nullable();

$table->string('province')->nullable();

$table->string('zipcode')->nullable();

$table->string('sitio')->nullable();

$table->string('mlast')->nullable();

$table->string('mfirst')->nullable();

$table->string('mmiddle')->nullable();

$table->string('flast')->nullable();

$table->string('ffirst')->nullable();

$table->string('fmiddle')->nullable();

$table->string('citizenship')->nullable();

$table->string('religion')->nullable();

$table->string('hno')->nullable();

$table->string('hrel')->nullable();

$table->string('sib')->nullable();

$table->rememberToken();

$table->timestamps();

});}

2018\_09\_18\_050304\_announcements.php

<?php

class Announcements extends Migration

public function up()

{

Schema::create('announcements', function (Blueprint $table) {

$table->increments('id');

$table->string('title');

$table->text('description');

$table->rememberToken();

$table->timestamps();

});}}

Barangayclearance.php

<?php

class Barangayclearance extends Migration

{ public function up()

{

Schema::create('barangayclearance', function (Blueprint $table) {

$table->string('gender');

$table->string('name');

$table->string('street');

$table->string('civil');

$table->string('purpose');

$table->string('dateissued');

$table->rememberToken();

$table->timestamps();

});

}}

Feedback.php

<?php

class Feedback extends Migration

{

public function up()

{

Schema::create('feedback', function (Blueprint $table) {

$table->integer('id');

$table->string('name')->nullable();

$table->string('concerntitle');

$table->string('concerndetail');

$table->string('contactnum');

$table->rememberToken();

$table->timestamps();

});

}}

Non-resident.php

<?php

class Nonresident extends Migration

{

public function up()

{

Schema::create('nonresident', function (Blueprint $table) {

$table->increments('id');

$table->string('lastname');

$table->string('firstname');

$table->string('middlename');

$table->string('email');

$table->string('image');

$table->string('gender');

$table->string('civil');

$table->string('age');

$table->string('birthdate');

$table->string('votersno');

$table->string('yearsres');

$table->string('street');

$table->string('barangay');

$table->string('cityprovince');

$table->string('province');

$table->string('zipcode');

$table->string('sitio');

$table->string('mlast');

$table->string('mfirst');

$table->string('mmiddle');

$table->string('flast');

$table->string('ffirst');

$table->string('fmiddle');

$table->string('citizenship');

$table->string('religion');

$table->string('hno');

$table->string('hrel');

$table->string('sib');

$table->rememberToken();

$table->timestamps();

}); }}

Forms.php

<?php

public function up()

{

Schema::create('forms', function (Blueprint $table) {

$table->string('gendertitle')->nullable();

$table->string('name')->nullable();

$table->string('street')->nullable();

$table->string('civil')->nullable();

$table->string('purpose')->nullable();

$table->string('reason1')->nullable();

$table->string('reason2')->nullable();

$table->string('age')->nullable();

$table->string('dateissued')->nullable();

$table->timestamps();

});

}}

DatabaseSeeder.php

<?php

class DatabaseSeeder extends Seeder

{

public function run()

{

// $this->call(UsersTableSeeder::class);

$this->call(RolesTableSeeder::class);

}}

RolesTableSeeder.php

<?php

class RolesTableSeeder extends Seeder

{

public function run()

{

//Create Admin Role

$sadmin = new Role();

$sadmin->name = "suadmin";

$sadmin->display\_name = "Super Administrator";

$sadmin->save();

$admin = new Role();

$admin->name = "admin";

$admin->display\_name = "Administrator";

$admin->save();

//Create Desk Officer Role

$desk = new Role();

$desk->name = "deskofficer";

$desk->display\_name = "Desk Officer";

$desk->save();

//Create Secretary Role

$secretary = new Role();

$secretary->name = "secretary";

$secretary->display\_name = "Secretary";

$secretary->save();

//Create Treasurer Role

$treasurer = new Role();

$treasurer->name = "treasurer";

$treasurer->display\_name = "Treasurer";

$treasurer->save();

//Create Clerk Role

$clerk = new Role();

$clerk->name = "clerk";

$clerk->display\_name = "Clerk";

$clerk->save();

$citizen = new Role();

$citizen->name = "citizen";

$citizen->display\_name = "Citizen";

$citizen->save();

//Attach the roles

$user1 = User::find(1);

$user1->detachrole($sadmin);

$user1->attachrole($sadmin);

$user2 = User::find(2);

$user2->detachrole($admin);

$user2->attachrole($admin);

$user3 = User::find(3);

$user3->detachrole($desk);

$user3->attachrole($desk);

$user4 = User::find(4);

$user4->detachrole($secretary);

$user4->attachrole($secretary);

$user5 = User::find(5);

$user5->detachrole($treasurer);

$user5->attachrole($treasurer);

$user6 = User::find(6);

$user6->detachrole($clerk);

$user6->attachrole($clerk);

$user6 = User::find(7);

$user6->detachrole($citizen);

$user6->attachrole($citizen);

}}

Index.php

<?php

define('LARAVEL\_START', microtime(true));

require \_\_DIR\_\_.'/../vendor/autoload.php';

$app = require\_once \_\_DIR\_\_.'/../bootstrap/app.php';

$kernel = $app->make(Illuminate\Contracts\Http\Kernel::class);

$response = $kernel->handle(

$request = Illuminate\Http\Request::capture()

);

$response->send();

$kernel->terminate($request, $response);

Api.php

<?php

Route::middleware('auth:api')->get('/user', function (Request $request) {

return $request->user();

});

Channels.php

<?php

Broadcast::channel('App.User.{id}', function ($user, $id) {

return (int) $user->id === (int) $id;

});

RoutesWeb.php

<?php

Route::get('/','Welcome@index');

Route::post('/send','Welcome@send')->name('welcome.send');

Auth::routes();

Route::get('/register','Auth\RegisterController@index')->name('register');

Route::get('/regnon','Regnon@index');

Route::resource('/regnon','RegNon');

//Admin

Route::get('/home','HomeController@index')->name('home');

Route::get('/admin/dashboard','AdminController@index')->name('dashboard');

Route::get('/admin/ticket','TicketController@index')->name('ticket');

Route::resource('admin/ticket','TicketController');

Route::get('admin/datasync','DataBackupController@index')->name('datasync');

Route::get('blotter/createblotter','CreateBlotterController@index');

Route::resource('blotter/createblotter','CreateBlotterController');

Route::get('bpinformation/announcements','AnnouncementList@index');

Route::resource('bpinformation/announcements','AnnouncementList');

Route::get('bpinformation/updateannouncements','UpdateAnnouncement@index');

Route::resource('bpinformation/updateannouncements','UpdateAnnouncement');

Route::get('clerk/addresident','AddResident@index');

Route::resource('clerk/addresident','AddResident');

Route::get('clerk/residentsearch','ResidentSearch@index');

Route::get('clerk/nonresident', 'NonResidentController@index');

Route::resource('clerk/nonresident','NonResidentController');

Route::get('blotter/retrieveblotterrecord','RetrieveBlotter@index');

Route::get('/barangayclearance','DownloadForms@index')->name('barangayclearance');

Route::get('barangayclearance/print','PdfController@index');

Route::get('indigency/print','PdfController@resi');

Route::get('residency/print','PdfController@resi2');

Route::get('/citizen/citizenhome','CitizenHome@index');

Route::post('/citizen/citizenhome/sendfeedback','CitizenHome@sendFeedback')->name('citizen.sendfeedback');

Route::post('/citizen/citizenhome/blottercreate','CitizenHome@blottercreate')->name('citizen.blottercreate');

Route::post('/citizen/citizenhome/clearance','Forms@clearance')->name('forms.clearance');

Route::post('/citizen/citizenhome/indigency','Forms@indigency')->name('forms.indigency');

Route::post('/citizen/citizenhome/residency','Forms@residency')->name('forms.residency');

Route::get('/admin/mailbox','Mailbox@index');

// Exception routes

Route::get('exception/index', 'ExceptionController@index');

\_masterblade.php

<!doctype html>

<html lang="{{ app()->getLocale() }}">

<head>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

<meta name="description" content="LogViewer">

<meta name="author" content="ARCANEDEV">

<title>LogViewer - Created by ARCANEDEV</title>

{{-- Styles --}}

<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css" integrity="sha384-Gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJlSAwiGgFAW/dAiS6JXm" crossorigin="anonymous">

<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/font-awesome/4.4.0/css/font-awesome.min.css">

<link href='https://fonts.googleapis.com/css?family=Montserrat:400,700|Source+Sans+Pro:400,600' rel='stylesheet' type='text/css'>

<style>

html {

position: relative;

min-height: 100%;

}

body {

font-size: .875rem;

margin-bottom: 60px;

}

.main-footer {

position: absolute;

bottom: 0;

width: 100%;

height: 60px;

line-height: 60px;

background-color: #E8EAF6;

}

- .main-footer p {

margin-bottom: 0;

}

.main-footer .fa.fa-heart {

color: #C62828;

}

.page-header {

border-bottom: 1px solid #8a8a8a;

}

.navbar-brand {

padding: .75rem 1rem;

font-size: 1rem;

}

.navbar-nav .nav-link {

padding-right: .5rem;

padding-left: .5rem;

}

.box {

display: block;

padding: 0;

min-height: 70px;

background: #fff;

width: 100%;

box-shadow: 0 1px 1px rgba(0,0,0,0.1);

border-radius: .25rem;

}

.box > .box-icon > i,

.box .box-content .box-text,

.box .box-content .box-number {

color: #FFF;

text-shadow: 0 1px 1px rgba(0, 0, 0, 0.3);

}

.box > .box-icon {

border-radius: 2px 0 0 2px;

display: block;

float: left;

height: 70px; width: 70px;

text-align: center;

font-size: 40px;

line-height: 70px;

background: rgba(0,0,0,0.2);

}

.box .box-content {

padding: 5px 10px;

margin-left: 70px;

}

.box .box-content .box-text {

display: block;

font-size: 1rem;

white-space: nowrap;

overflow: hidden;

text-overflow: ellipsis;

font-weight: 600;

}

.box .box-content .box-number {

display: block;

}

.box .box-content .progress {

background: rgba(0,0,0,0.2);

margin: 5px -10px 5px -10px;

}

.box .box-content .progress .progress-bar {

background-color: #FFF;

}

.log-menu .list-group-item.disabled {

cursor: not-allowed;

}

.log-menu .list-group-item.disabled .level-name {

color: #D1D1D1;

}

.stack-content {

color: #AE0E0E;

font-family: consolas, Menlo, Courier, monospace;

white-space: pre-line;

font-size: .8rem;

}}

.badge.badge-level-all,

.box.level-all {

background-color: {{ log\_styler()->color('all') }};

}

.badge.badge-level-emergency,

.box.level-emergency {

background-color: {{ log\_styler()->color('emergency') }};

}

.badge.badge-level-alert,

.box.level-alert {

background-color: {{ log\_styler()->color('alert') }};

}

.badge.badge-level-critical,

.box.level-critical {

background-color: {{ log\_styler()->color('critical') }};

}

.badge.badge-level-error,

.box.level-error {

background-color: {{ log\_styler()->color('error') }};

}

.badge.badge-level-warning,

.box.level-warning {

background-color: {{ log\_styler()->color('warning') }};

}

.badge.badge-level-notice,

.box.level-notice {

background-color: {{ log\_styler()->color('notice') }};

}

.badge.badge-level-info,

.box.level-info {

background-color: {{ log\_styler()->color('info') }};

}

.badge.badge-level-debug,

.box.level-debug {

background-color: {{ log\_styler()->color('debug') }};

}

.badge.empty,

.box.empty {

background-color: {{ log\_styler()->color('empty') }};

}

.badge.badge-env {

background-color: #6A1B9A;

}

</style>

</head>

<body>

<nav class="navbar navbar-expand-md navbar-dark sticky-top bg-dark p-0">

<a href="{{ route('log-viewer::dashboard') }}" class="navbar-brand mr-0">

<i class="fa fa-fw fa-book"></i> LogViewer

</a>

<button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false" aria-label="Toggle navigation">

<span class="navbar-toggler-icon"></span>

</button>

<div class="collapse navbar-collapse" id="navbarNav">

<ul class="navbar-nav mr-auto">

<li class="nav-item {{ Route::is('log-viewer::dashboard') ? 'active' : '' }}">

<a href="{{ route('log-viewer::dashboard') }}" class="nav-link">

<i class="fa fa-dashboard"></i> Dashboard

</a>

</li>

<li class="nav-item {{ Route::is('log-viewer::logs.list') ? 'active' : '' }}">

<a href="{{ route('log-viewer::logs.list') }}" class="nav-link">

<i class="fa fa-archive"></i> Logs

</a>

</li>

</ul>

</div>

</nav>

<div class="container-fluid">

<main role="main" class="pt-3">

@yield('content')

</main>

</div>

{{-- Footer --}}

<footer class="main-footer">

<div class="container-fluid">

<p class="text-muted pull-left">

LogViewer - <span class="badge badge-info">version {{ log\_viewer()->version() }}</span>

</p>

<p class="text-muted pull-right">

Created with <i class="fa fa-heart"></i> by ARCANEDEV <sup>&copy;</sup>

</p>

</div>

</footer>

{{-- Scripts --}}

<script src="https://code.jquery.com/jquery-3.2.1.min.js" crossorigin="anonymous"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.12.9/umd/popper.min.js" crossorigin="anonymous"></script>

<script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/js/bootstrap.min.js" crossorigin="anonymous"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/Chart.js/2.7.1/Chart.min.js"></script>

@yield('modals')

@yield('scripts')

</body>

</html>

Dashboardblade.php

@extends('log-viewer::bootstrap-4.\_master')

@section('content')

<div class="page-header mb-4">

<h1>Dashboard</h1>

</div>

<div class="row">

<div class="col-md-6 col-lg-3">

<canvas id="stats-doughnut-chart" height="300" class="mb-3"></canvas>

</div>

<div class="col-md-6 col-lg-9">

<div class="row">

@foreach($percents as $level => $item)

<div class="col-sm-6 col-md-12 col-lg-4 mb-3">

<div class="box level-{{ $level }} {{ $item['count'] === 0 ? 'empty' : '' }}">

<div class="box-icon">

{!! log\_styler()->icon($level) !!}

</div>

<div class="box-content">

<span class="box-text">{{ $item['name'] }}</span>

<span class="box-number">

{{ $item['count'] }} entries - {!! $item['percent'] !!} %

</span>

<div class="progress" style="height: 3px;">

<div class="progress-bar" style="width: {{ $item['percent'] }}%"></div>

</div>

</div>

</div>

</div>

@endforeach

</div>

</div>

</div>

@endsection

@section('scripts')

<script>

$(function() {

new Chart(document.getElementById("stats-doughnut-chart"), {

type: 'doughnut',

data: {!! $chartData !!},

options: {

legend: {

position: 'bottom'

}

}

});

});

</script>

@endsection

DownloadForms.php

<?php

namespace App\Http\Controllers;

class DownloadForms extends Controller

{

public function index()

{

return view('navi.downloadforms');

}}

Forms.php

<?php

namespace App\Http\Controllers;

class Forms extends Controller

{

public function clearance(Request $request)

{

FormsModel::create([

'gendertitle' => $request['gendertitle'],

'name' => $request['name'],

'street' => $request['street'],

'civil' => $request['civil'],

'purpose' => $request['purpose'],

'dateissued' => $request['dateissued'],

]);

return redirect('/barangayclearance/print');

}

public function indigency(Request $request)

{

FormsModel::create([

'gendertitle' => $request['gendertitle'],

'name' => $request['name'],

'street' => $request['street'],

'reason1' => $request['reason1'],

'reason2' => $request['reason2'],

'dateissued' => $request['dateissued2'],

]);

return redirect('/indigency/print');

}

public function residency(Request $request)

{

FormsModel::create([

'gendertitle' => $request['gendertitle'],

'name' => $request['name'],

'street' => $request['street'],

'civil' => $request['civil'],

'age' => $request['age'],

'purpose' => $request['purpose'],

'dateissued' => $request['dateissued3'],

]);

return redirect('/residency/print');

}}

Citizenmiddleware.php

<?php

namespace App\Http\Middleware;

class citizenMiddleware

{

public function handle($request, Closure $next, $guard = null)

{

if (Auth::guard($guard)->check()) {

if($request->user()->Role == 'Citizen'){

return redirect('citizen/citizenhome');

}

return $next($request);

}

if(Auth::guest())

{

return redirect('/Restricted');

}}}

Kernel.php

<?php

namespace App\Http;

class Kernel extends HttpKernel

{

protected $middleware = [

\App\Http\Middleware\CheckForMaintenanceMode::class,

\Illuminate\Foundation\Http\Middleware\ValidatePostSize::class,

\App\Http\Middleware\TrimStrings::class,

\Illuminate\Foundation\Http\Middleware\ConvertEmptyStringsToNull::class,

\App\Http\Middleware\TrustProxies::class,

];

protected $middlewareGroups = [

'web' => [

\App\Http\Middleware\EncryptCookies::class,

\Illuminate\Cookie\Middleware\AddQueuedCookiesToResponse::class,

\Illuminate\Session\Middleware\StartSession::class,

// \Illuminate\Session\Middleware\AuthenticateSession::class,

\Illuminate\View\Middleware\ShareErrorsFromSession::class,

\App\Http\Middleware\VerifyCsrfToken::class,

\Illuminate\Routing\Middleware\SubstituteBindings::class,

],

'api' => [

'throttle:60,1',

'bindings',

],7u

];

protected $routeMiddleware = [

'auth' => \Illuminate\Auth\Middleware\Authenticate::class,

'auth.basic' => \Illuminate\Auth\Middleware\AuthenticateWithBasicAuth::class,

'bindings' => \Illuminate\Routing\Middleware\SubstituteBindings::class,

'cache.headers' => \Illuminate\Http\Middleware\SetCacheHeaders::class,

'can' => \Illuminate\Auth\Middleware\Authorize::class,

'guest' => \App\Http\Middleware\RedirectIfAuthenticated::class,

'signed' => \Illuminate\Routing\Middleware\ValidateSignature::class,

'throttle' => \Illuminate\Routing\Middleware\ThrottleRequests::class,

'role' => \Laratrust\Middleware\LaratrustRole::class,

'permission' => \Laratrust\Middleware\LaratrustPermission::class,

'ability' => \Laratrust\Middleware\LaratrustAbility::class,

];

}

**APPENDIX U.**

**PHOTO DOCUMENTATION**

**APPENDIX V.**

**CURRICULUM VITAE**