Polytechnic University of Puerto Rico
Electrical & Computer Engineering & Computer Science Department
Hato Rey, PR

Software Project Management Plan

Lambda Solutions Group Presents:

Name	Student ID	Program	Signature
Emilio Acosta Ortiz	86144	Computer Engineering	
Kevin Medina Santiago	85540	Computer Engineering	
Gabrielys Rivera Flores	89456	Computer Engineering	
Manuel Seda Batista	120758	Computer Engineering	
Ricardo Vilá Palacios	113924	Computer Engineering	

Changelog

Name	Date	Changes Performed	Version
Manuel Seda	2/3/2019	Began LaTeX template	1.00
Manuel Seda	2/3/2019	Began working on Section 3	1.01
Emilio Acosta	2/4/2019	Began working on Section 4	1.02
Rafael Santiago	2/5/2019	Began working on Section 1	1.03
Raúl Viruet	2/7/2019	Began working on Section 5	1.04
Gabrielys Rivera	2/7/2019	Began working on Section 5	1.05
Michael Quiles	2/8/2019	Began working on Section 2	1.06

Contents

List of Figures

List of Tables

1 Introduction

1.1 Project Overview

The group's objective is to develop a steganography desktop application for Linux distributions. This application will allow cybersecurity Capture the Flag competitors in steganography assignments. The project requirements will define, in general terms, the setup of the application, topics for available information concerning Software Project Management, and user activities (if any). Other official requirements are specified in the Software Requirements Specification (SRS).

1.2 Project Deliverables

The Lambda Solutions Group will develop this web-based app by following the Software Design Description (SDD). As part of this development, here's a list the following management documents and software products as deliverables:

Deliverable Item	Date	Location
Software Requirements Specification (SRS)	February 14th, 2019	Uploaded to group SharePoint
Software Design Description (SDD)	February 14th, 2019	Uploaded to group SharePoint
System Test Documentation (STD)	February 14th, 2019	Uploaded to group SharePoint
Project Prototype Presentation	?	Uploaded to group SharePoint, Physical Presentation
Software Builds	February 14th, 2019	GitHub Repository
Status Reports	Weekly during the duration of the project	Uploaded to group SharePoint
Final Presentation	?	Uploaded to group SharePoint, Physical Presentation
Project Poster	?	Uploaded to group SharePoint, Printed Poster

Table 1.1: Deliverable Items Summary

1.3 Evolution of the SPMP

Every updates or modifications to the document SPMP would be discussed with the team, as well as with the client. The procedure for the updates or modifications is to be determine at the moment of the update or modification.

1.4 Reference Materials

The following references use the IEEE citation format.

1.5 Definitions & Acronyms

Term	Definitions
SRS	Software Requirements Specification[?]
SDD	Software Design Description[?]
STD	System Test Documentation[?]
SPMP	Software Project Management Plan[?]
TBD	To Be Determined
User	Person that would be using the application.
Resident User	A user that lives in Vieques or Culebra[?]
Tourist User	A user that doesn't live in Vieques or Culebra[?]
Client	Organization or company that the application is delivered to. May also refer to a piece of computer hardware or software that accesses a service made available by a server[?].
Server	Where the application would be hosted[?].
Hosting	Store (a website or other data) on a server or other computer so that it can be accessed over the Internet[?].
Terminal	A device or program that enables you to communicate with a computer[?].
Database	A structured set of data held in a computer, especially one that is accessible in various ways[?].
GUI	Graphical User Interface
Query	A query is a request for data or information from a database table or combination of tables[?].
Web App	A web application is a computer program that utilizes web browsers and web technology to perform tasks over the Internet[?].
HTML	Hyper Text Markup Language[?]

Table 1.2: Terms & Definitions

2 Project Organization

This section specifies the process model for the project, describes the organizational structure of the project, identifies organizational boundaries and interfaces, and defines individual responsibilities for the different elements of the project.

2.1 Process Model

The process used for this project will be a Waterfall Model such it is the most used model for small projects. In this model, each phase is executed completely before the beginning of the next phase and a feedback is taken after each phase to ensure that the project is on the right path. The different phases of the waterfall model are shown in Figure 2.1.

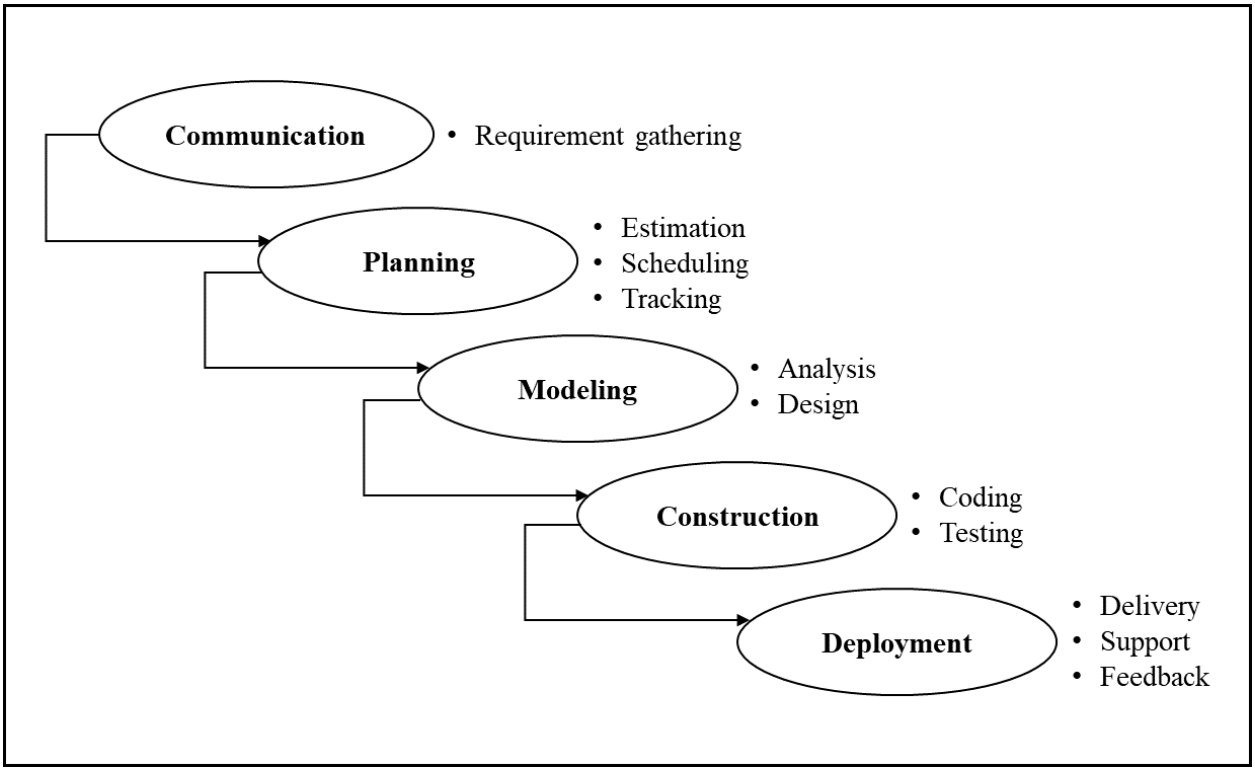

Figure 2.1: Waterfall Model

2.2 Organizational Structure

A diagram representing the organizational structure of the project is shown below:

Lambda Solutions' Development Team is comprised of the following members:

- Emilio Acosta Ortiz
- Kevin Medina Santiago

- Gabrielys Rivera Flores
- Manuel Seda Batista
- Ricardo Vila Palacios

Figure 2.2 shows the project development timeline.

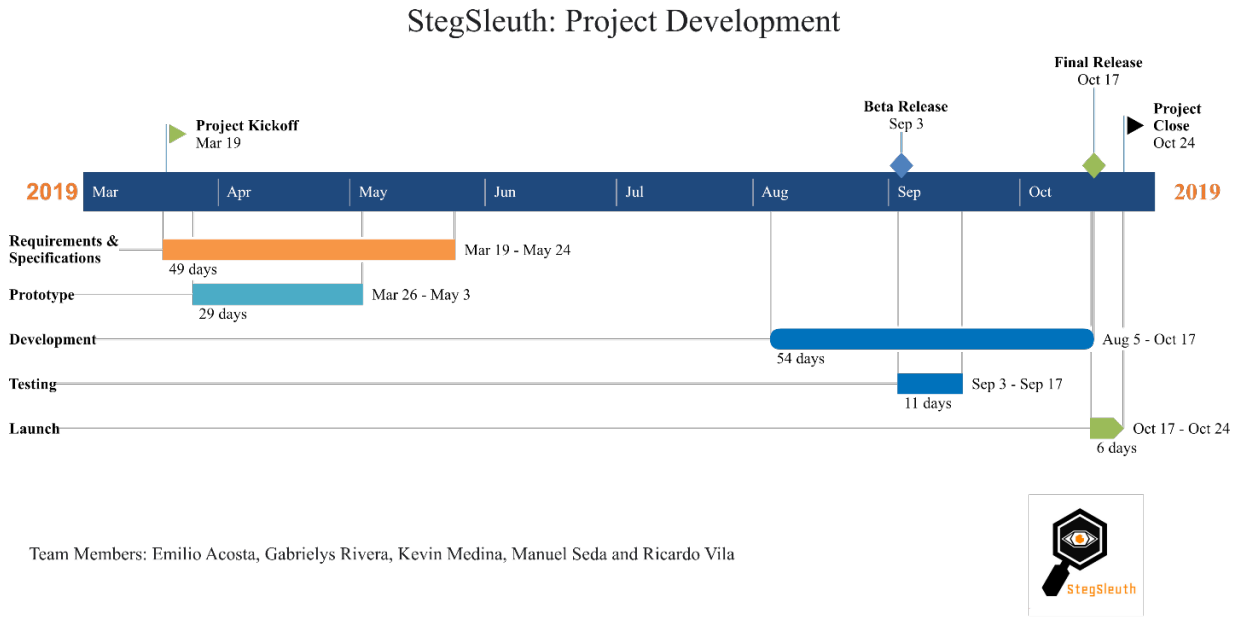

Figure 2.2: Project Timeline

The following table shows the roles of each member of the development team:

Name	Role
Manuel Seda Batista	Project Manager
Rafael Sanstiago Solivan	Database Administrator
Michael Quiles Melendez	GUI Designer
Gabrielys Rivera Flores	GUI Developer
Raúl Viruet Román	Database Designer
Emilio Acosta Ortiz	Client Designer
Oscar Navarro Estarrelas	Client Developer

Table 2.1: LSG Members Roles

2.3 Organizational Boundaries & Interfaces

Team members throughout the development of each phase will be responsible for coordinating meetings, updates, communications and equipment deliverables.

2.4 Project Responsibilities

To know each phase of the project, consult Section 2.2 of the document. The entire team is responsible for the entire development of the project and the successful delivery of the product. The tasks of team members can be delivered according to the experience and the phases below:

1. Requirements and Specifications
2. Prototype
3. Development
4. Testing
5. Launch

3 Managerial Process

This section specifies management objectives and priorities to be met during the elaboration of this software project.

3.1 Management Objectives & Priorities

Lambda Solutions's philosophy is to provide software solutions that will satisfy our clients and users' needs and beyond. The project's main objective is to create a system that will benefit cybersecurity. The development team will compile the following documents, explained in section 3.4, to assist in logging their progress to themselves, the program manager and the client:

- Status Report
- Project Review
- Progress Report
- Change Requests
- Meeting Logs

To help visualize the budget allocation, the table below is provided:

Project Element	Allocation
Database Design, Development & Maintenance (Including Hardware and Staff)	35%
GUI Design & Development (Includes Staff)	15%
Back-end Design & Development (Includes Staff)	25%
System Deployment (Includes Hardware and Staff)	20%
Other Costs	5%

Table 3.1: Budget Allocation

The following table maps the flexibility of each of the project's dimensions.

Project Dimension	Fixed	Constrained	Flexible
Cost		**X**	
Schedule	**X**		
Functionality			**X**
Staff Requirements		**X**	
Technical Process			**X**
Project Deliverables	**X**		

Table 3.2: Project Dimension Flexibility Matrix

3.2 Assumptions, Dependencies & Constraints

3.2.1 Assumptions

The project's design revolves around the following assumptions:

- The system will be deployed on Linux distributions
- The schedule is accurate
- All dependencies are satisfied
- All constraints are dealt with
- Only users with a high degree of computer experience will interact with our program
- Users know how to resolve a dependency chain and compile from source
- Provided files are within the project's foreseen scope
- Keys used for decryption operations are correct
- Payload to be encoded into the input file is within the project's scope

3.2.2 Dependencies

The project's design revolves around the following dependencies:

- **A supported Linux distribution** - the software will be created using tools readily available for most Linux distributions. Other POSIX compliant Operating Systems may be able to build and run the software, but the stability and reliability of said software will be unknown
- **Python Encryption libraries** - files used as input in the decryption module of our software will be processed by functions from existing Python Encryption libraries.
- **Python qt5 bindings** - the software's GUI will be crafted using the qt5 framework.
- **Python Steganography libraries** - steganographic encoding & decoding will be programmed using python steganographic libraries

3.2.3 Constraints

The project's design revolves around the following constraints:

- **Regulatory Policies** - the project as a whole must comply with local, state and federal regulatory policies
- **Interfaces to Operating System** - this project's software will depend on many operating system elements such as the file system and random number generator.
- **Reliability Requirements** - because the project will be used as a tool for competitions, the software must perform with speed, accuracy and precision

3.3 Risk Management

The risks this project can (and will for some) incur are, but not limited to:

Risk Factor	Assessment	Risk Rating	Contingencies
Team Stability	The likelihood of team members leaving or not getting along could greatly impact the project's completion	Low	Team-building exercises, have a pool of employee candidates ready to fill any voids in the team
User Experience	Users not having the experience necessary to operate the system	Low	Have tutorials, documentation and staff readily available to help users with any difficulties operating the system
User Acceptance	Users being dissatisfied with the system when compared to the current one, causing the customer to terminate the contract	Very Low	Have a section of the software dedicated to collecting user feedback
Technology	New technologies being used in the project could cause difficulties in the project's completion	Medium	Undergo trainings and use all resources available to ensure the team's proficiency with the technologies involved in making the project
Schedule Conflicts or Workload Conflicts	Team members are full times students with full or part-time employments. University or employer workflow & scheduling conflicts can and will arise	Very High	Organized execution of the project's tasks will minimize the dangers of scheduling conflicts

Table 3.3: Risk Management Analysis

3.4 Monitoring & Controlling Mechanisms

The project manager will have bi-weekly meetings to check on the development team's progress and to guide the team's efforts. Weekly status reports must be submitted to the project manager to monitor the team's progress. Monthly project reviews will be conducted on the existing project modules and phases. Whenever the Project Manager or the client requests a change, it must be documented. Every meeting must have a log. The following table summarizes these mechanisms:

Report Type	From	To	Time Period
Status Report	Development Team	Project Manager	Weekly
Project Review	Project Manager	Development Team	Bi-weekly
Progress Report	Lambda Solutions	Client	Monthly
Change Request	Project Manager and/or Client	Development Team	Whenever Needed
Meeting Logs	Development Team & Project Manager	Lambda Solutions	Whenever a Meeting Occurs

Table 3.4: Reports

3.5 Staffing Plan

To be able to complete the project, talented and dedicated staff members are needed. These staff members should be broken down into three main development teams. These teams coincide with the number of main modules and their responsibilities should stem from the team they're on. Each team will have a team leader, and these leaders will report to a project manager. The following table will show the type of staff needed.

Role	**Team**	**Experience & Education Required**
GUI Developer	GUI Development	Bachelors in Computer Science or 2-3 years front end development experience
Systems Developer	Functionality Development	Bachelors in Computer Science or 2-3 years systems development experience
Linux Developer	Packaging & Deployment	Bachelors in Computer Science or at least one year of experience

Table 3.5: Staff Roles, Teams and Requirements

4 Technical Process

This section of the SPMP shall specify the technical methods, tools and techniques to be used on the project. In addition, the plan for software documentation shall be specified and also describes the plans for project support functions such as quality assurance, configuration management, verification and validation may be specified.

4.1 Methods, Tools & Techniques

This section details the computer systems, the development methodologies, the structure of the equipment, the programming languages, the tools, the techniques and the methods that will be used to specify, design, build, test, integrate, document, deliver, modify and maintain project standards.

4.1.1 Methods

- **Testing Methods**
  - White-Box Testing: Each software unit will be tested individually to produce the desired results.
  - Black-Box Testing: The software will be tested after completion to test its overall functionality.
- **Development Methods**
  - Programming Style: Developers will be using the indent style and naming conventions.
  - Programming Languages: For our web page, the developers must be fluent in HTML, Java and SQL.

4.1.2 Tools

- **Documentation Tools**
  - Microsoft Word: Software used for the creation of every required document as well as for simultaneous collaboration of the team members.
  - Microsoft Power Point: Software used for creation of the required presentations.
  - Google Drive: Cloud Storage for saving all required documents.
  - LaTeX: software used for the final creation of each required document, as well as for the simultaneous collaboration of team members.
- **Development Tools**
  - MySQL Workbench: Software used for the database creation and management of data.
  - Eclipse IDE: Software used to create the website's GUI.
  - Notepad++ : Software used for general purpose languages.
  - Computers: Computers: equipment that will be used by team members to complete all tasks. This equipment, when connected to the Internet, will also be used to test the page in a web browser.

4.1.3 Techniques

- Smooth transition: this method will ensure that each phase is completely completed, reviewed, tested and approved by the client before making major changes.
- Commitment of the administrator: this technique will leave a positive impact that will ensure a functional service.
- Quality of the service: the quality of the service is an important strategy to execute a functional and reliable system that the user can feel safe and comfortable to use.

4.2 Software Documentation

The software documentation to be made is composed of the following IEEE (Institute of Electrical and Electronics Engineers, 1963) Standards:

Document	Description	Format
SPMP	Software Project Management Plan: Document management refers to the planning of the project, its main stages and its organization.	IEEE 1058-1998
SDD	Software Design Description: Document describing the product design.	IEEE1016-1998
SRS	Software Requirement Specifications: Document defining the requirements and details requested by the client.	IEEE830-1998
STD	Software Test Documentation: Document defining each test and the tester.	IEEE829-1998

Table 4.1: Software Documentation

Other documentation:

- Presentations: The team needs to offer two different presentations with the purpose of describing the system in a detailed manner.
- Personal Log: Each team member must possess a personal log file in which they will write what was discussed in each meeting as well as their work done in each objective of the product.
- Team Log File: Each team meeting and decisions will be noted in the team logbook. The materials used and the objectives that were discussed will be written in the logbook as well.

4.3 Project Support Functions

4.3.1 Configuration Management

The purpose of the software configuration management is to organize, control, plan and coordinate every part of the software suite to be developed. This will be done in parallel to the development process of the software.

- Software versions will increment 0.1 each time someone performs a change on a document.
- Individual testing for each unit will be conducted as the development process progresses.

- The project shall be stored and updated by all members.
- Milestones, identified by 1.0 version increments, for each software unit will be established. The following changes count as milestones: Initial functional web page, fixes, optimizations and requests & final fixes and adjustments.
- Individual testing for each unit will be conducted as the development process progresses.

4.3.2 Verification & Validation

The activities to be performed in this function are designed to verify the software and make sure that it meets the stated system specifications in the SRS document. This will be done through intense testing detailed in the STD document.

4.3.3 Quality Assurance

Testing by the client will be conducted for this function of the project. The purpose of this testing is to make sure that the client is satisfied with the developed Software and meets the system requirements stated in the SRS document.

5 Work Packages, Schedule & Budget

In this section, there will be a description of work packages, alongside with a schedule on how the project should be developed and description of the dependencies and requirements between elements in the project.

5.1 Work Packages

Employee allocation
Describes which group of people performs which task, and also specifies the amount of employees allocated to a particular task.

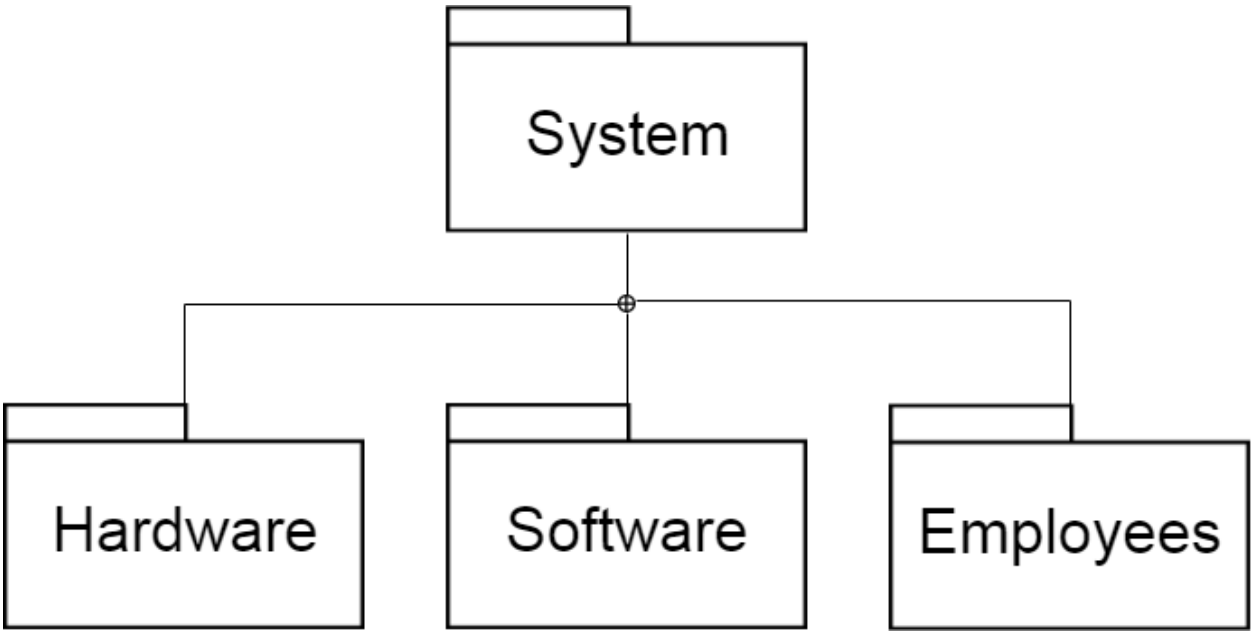

Figure 5.1: Package Diagram - System Top View

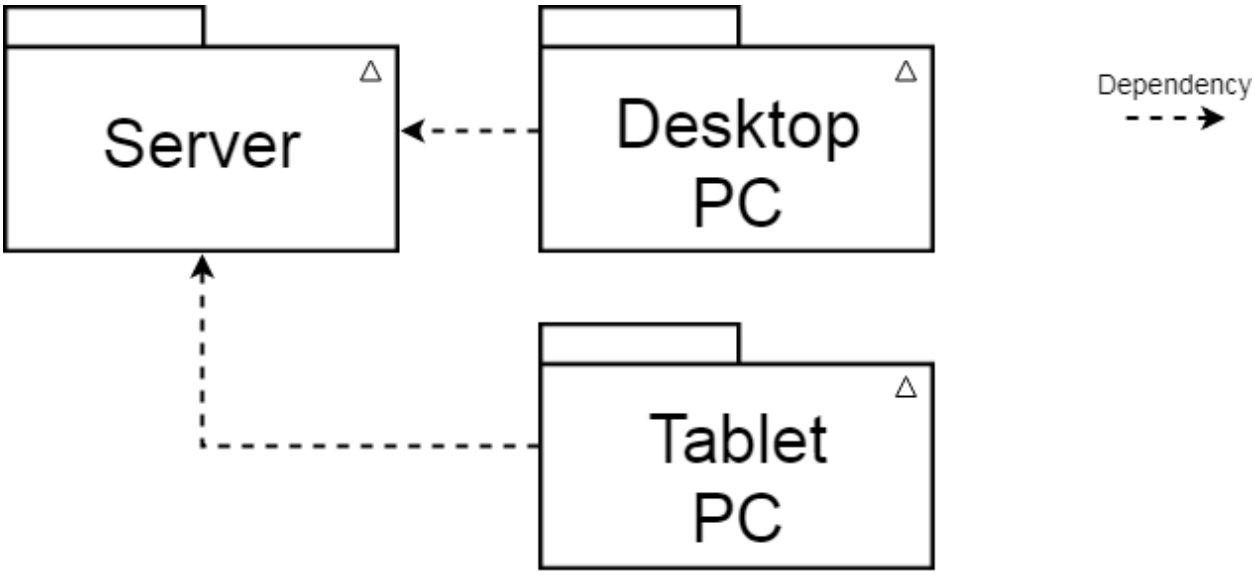

Figure 5.2: System Dependencies

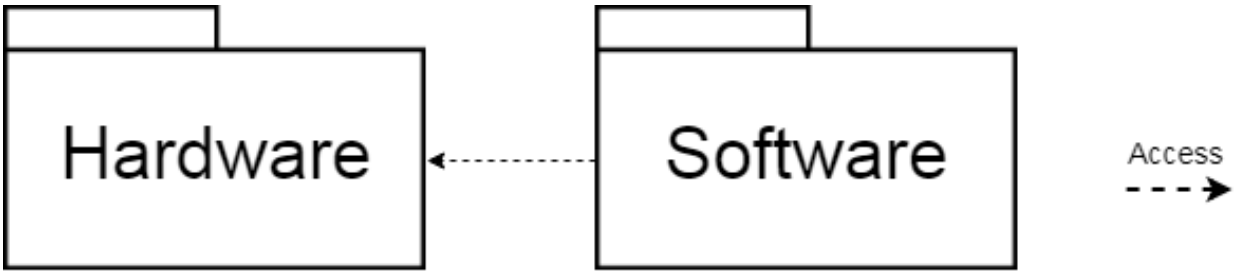

Figure 5.3: Dependencies between packages

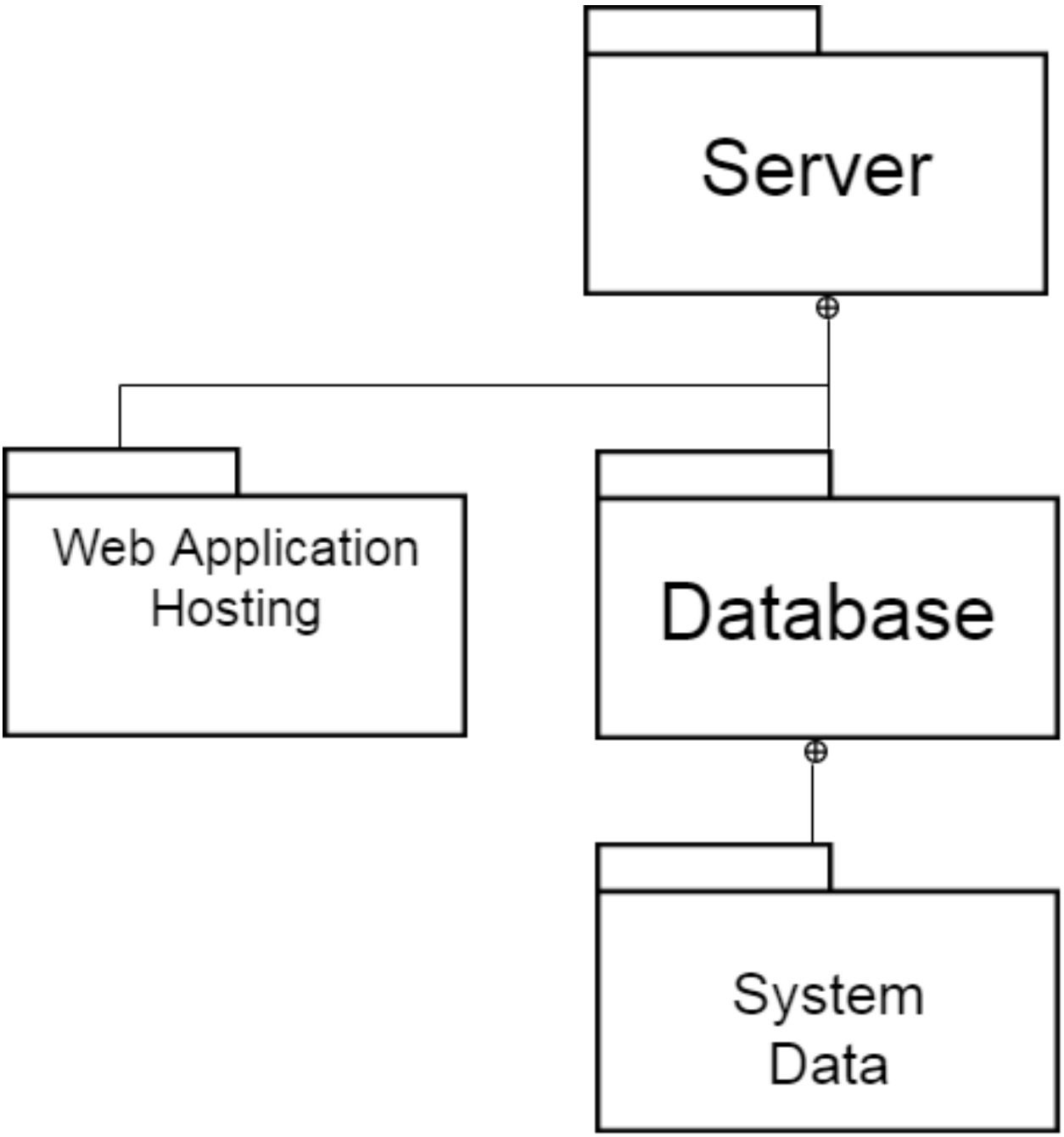

Figure 5.4: Package Diagram - Server

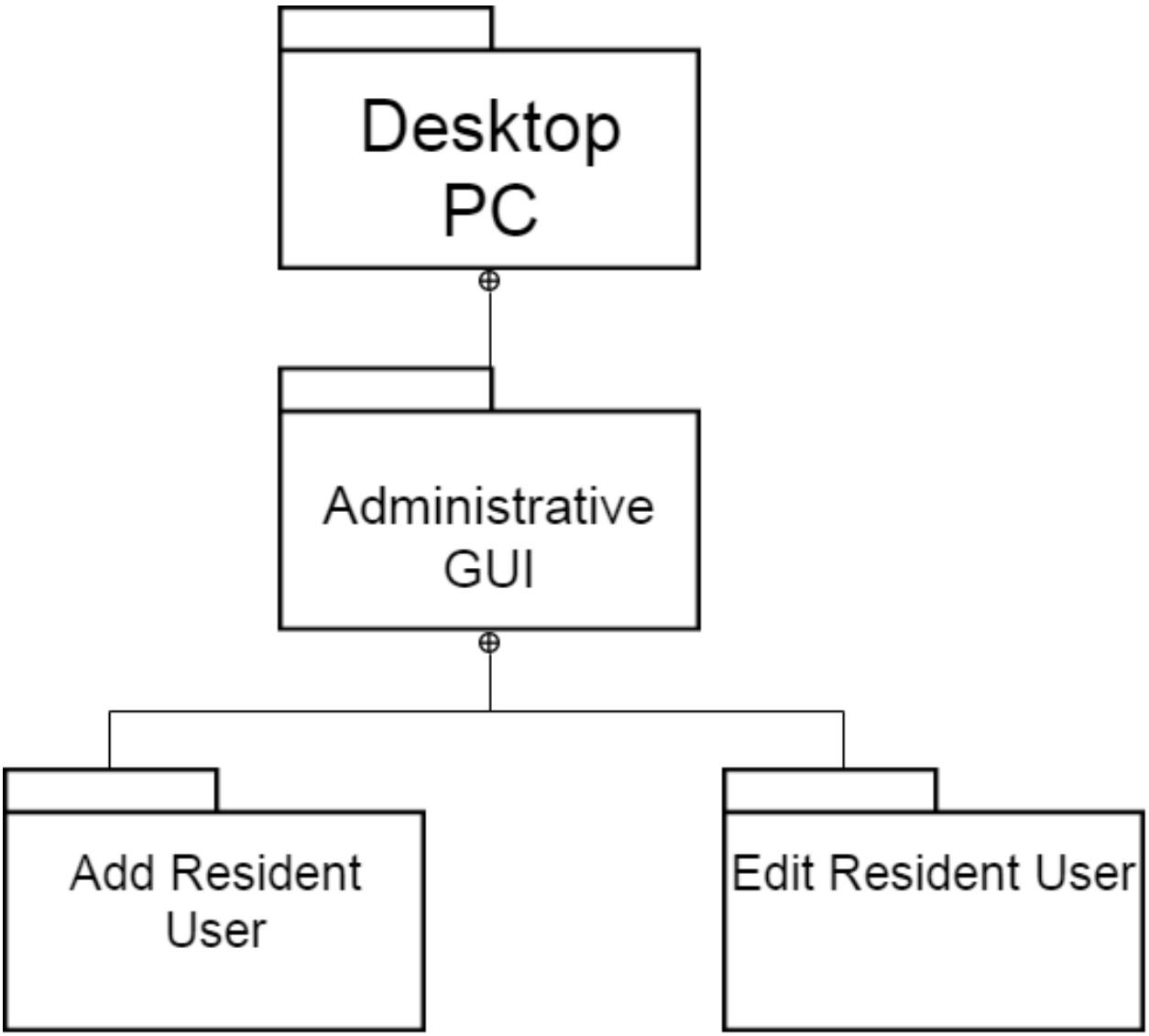

Figure 5.5: Package Diagram - Desktop PC Component

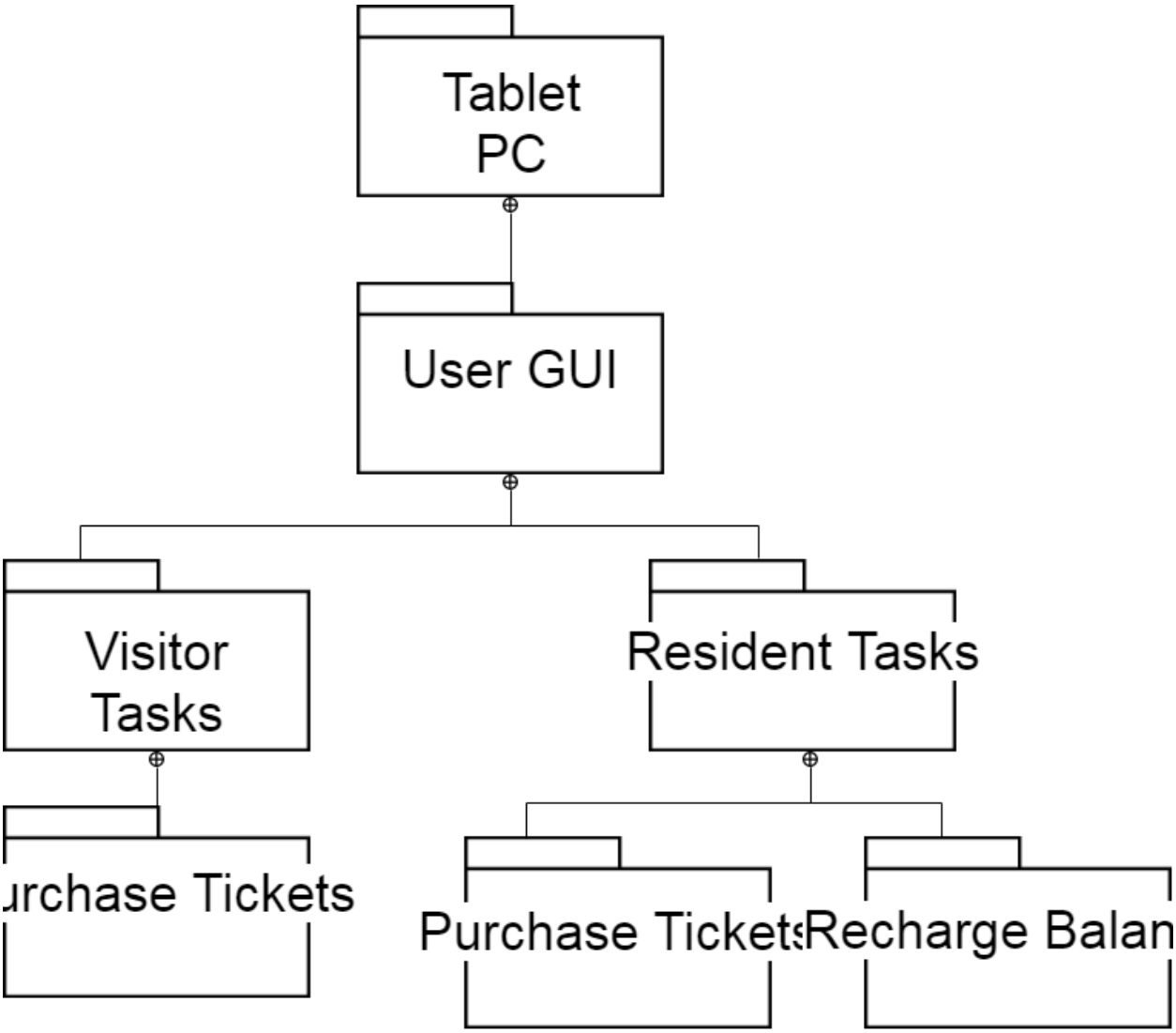

Figure 5.6: Package Diagram - Tablet PC Component

Group	Task	Number of Employees
A	Install Vieques terminals and configure them with the system	3
B	Install Ceiba terminals and configure them with the system	3
C	Install Culebra terminals and configure them with the system	3
O	Install terminals and server equipment in customer service main office, and configure these devices.	5
T	Train Customer service office (OA) employees how to use the system to add new Resident users to the system and modify existing users. Personnel will also perform customer service training. Train clerks for Vieques, Culebra and Ceiba (KV, KC, KF) on how to operate and troubleshoot their respective terminals.	4
KV	Vieques customer service clerk, in charge of trooubleshooting and provide customer assistance if required.	6

Group	Task	Number of Employees
KC	Culebra customer service clerk, in charge of trooubleshooting and provide customer assistance if required.	4
KF	Ceiba customer service clerk, in charge of trooubleshooting and provide customer assistance if required.	4
OA	In charge of adding new resident users to the database and enabling their acess to the system. They are also responsible of adding existen Resident user information if required.	4
SD	Software Developers (SD) are in charged of designing the system according to specifications	6
TST	System Testers, wil perform Unit Testing, Integration Testing, Sanity Testing and Interface Testing	15

Table 5.1: Employee Allocation to project realization

5.2 Dependencies

The following list describes the dependencies for each step of the development process.

1. Design GUI
2. Design App Logic
3. Develop Application
4. Tests I
5. Fix Issues I
6. Tests II
7. Fix Issues II
8. Fix Issues II
9. Package Application I
10. Tests III
11. Fix Issues III
12. Package Application II
13. Tests IV
14. Fix Issues IV
15. Write documentation

5.3 Resource Requirements & Allocation

Describes and specifies required equipment to be purchased and installed for the initial system realization.

Equipment	Amount Needed	Description
Personal Computers	5	Equipment needed for all programming and system testing
Personal USB Drives	5	For quick local file sharing, installation medium for GNU/Linux
Test Devices	2	Computers running Windows, GNU/Linux and macOS to test the system's compatibility and distributability

Table 5.2: Equipment Summary

Name	Position	Description of Responsibilities
Emilio Acosta Ortiz		
Kevin Medina Santiago		
Gabrielys Rivera Flores		
Manuel Seda Batista		
Ricardo Vilá Palacios		

Table 5.3: Staff Allocation

5.4 Schedule

A detailed look of project milestones and due dates.

Milestone	Start Date	Due Date	Notes
Prototype Development	March 26, 2019	May 3, 2019	Develop the prototype of the Graphical User Interface (GUI).
Desktop Application Development	August 5, 2019	October 17, 2019	Develop StegSleuth application for steganography and cryptography.
Testing Period I	September 3, 2019	September 6, 2019	Perform test to the developed application.
Application Readjustment Period I	September 6, 2019	September 14, 2019	Correct issues and bugs found on Test Period I.
Testing Period II	September 14, 2019	September 17, 2019	Perform test to the developed application.
Application Readjustment Period II	September 17, 2019	October 17, 2019	Correct issues and bugs found on Test Period II.
Desktop Application Launch	N/A	October 17, 2019	Application goes online.

Table 5.4: Software Development Milestones