Daryl Blancaflor
Raymond Chin
Benjamin Guerrero
Carlos Lomeli

# Software Project Plan

# Introduction

## Project Scope

Cerberus is an all-purpose security package for a user's mobile device for Android. The software is designed to protect the user's sensitive data on the device using

The software will consist of the following inputs:
- User's first and last name
- User's email address
- User's address
- A scan of the user's fingerprint
- A password for user's application
- Permission to track the user's phone
- Permission to access the user's files
- Security questions for that specific user
- Permission to receive SMS
- Permission to connect to other apps

The software will consist of the following outputs:
- A confirmation email
- A user account
- SMS message (for 2-way verification)
- Database entry of the user

Any outputs that will be created will be reflected in the GUI. Every output that occurs should be in line with the input that the user has done.

# Major Software Functions

## *Process and Control Functions*

GUI Interface - The interface that user will interact with. Access to the functions of the interface will be locked behind a login screen. It gathers all necessary data from the user, as well as interacting with the access databases. This subsystem contains the screen representing the security application along with the various screens and their attributes.
Java engine - This subsystem performs the main functions of the system.

## *User Interface Processing*

Input Wizard – There are a number of wizards provided to guide the novice user through the necessary steps for creating an account. The wizards interact directly with the user interface as pop-up screens.

## *Input Processing*

File Structure - Cerberus will have all the files that will be necessary for the application to run stored in the same directory. This way, all the files will be stored in a single location. The individual files will be accessed by the code and by the user interface.

Database – Cerberus utilizes a Firebase database to store information on the users. The database will store general user information as well as data related to security for user authentication. The databases are accessed by the user interface.

## **Output Processing**

Data files – files containing information specified by the user that are read by the Java code. The files are generated by the user interface

The finished project will produce a deliverable .JAR file.

## **Performance and Behaviour Issues**

Cerebus is designed to be compatible with mobile devices that have fingerprint scanning and geolocation capabilities. Both the selected mobile device and desktop must have the storage capacity to download the application.

Fingerprint scanner will not work for all mobile devices, so a workaround will be implemented for those devices that do not have the hardware for this feature.

# Management and Technical Constraints

Our team will be using the Unified Process Model during design and implementation.
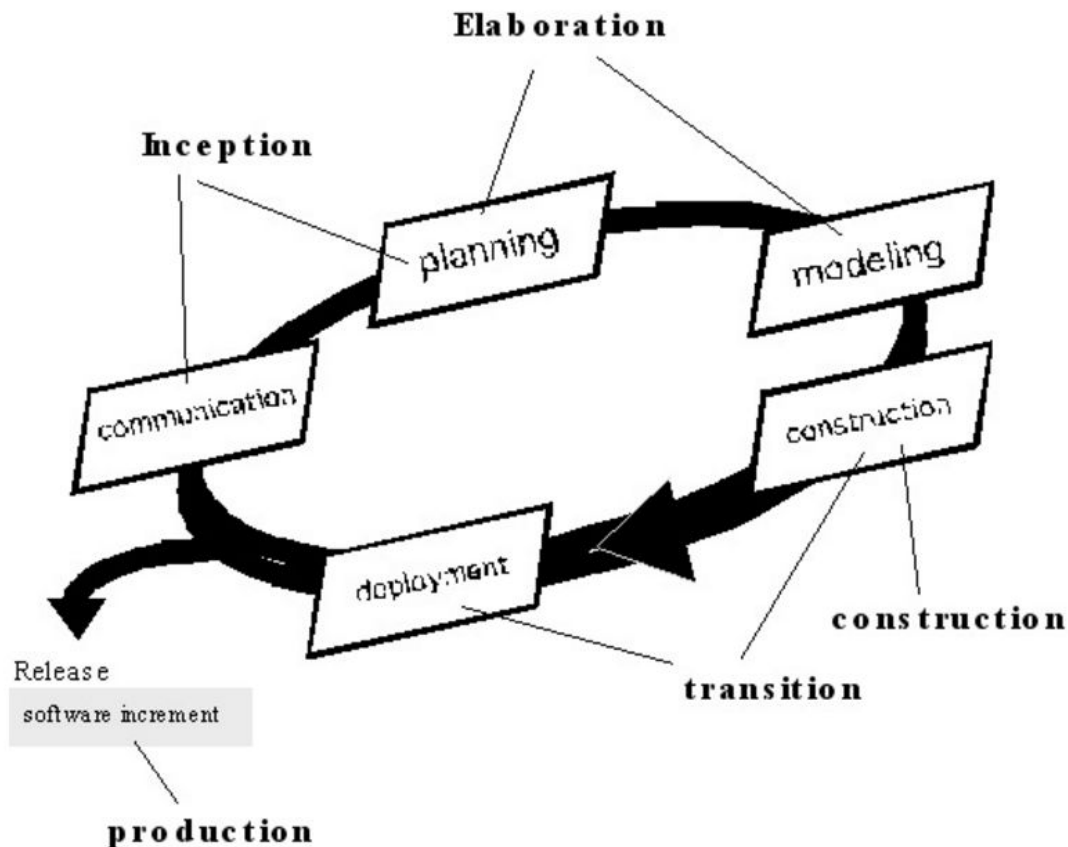


**Project Resources**
Required Staff
- Lead progammer
- UI/UX developer
- Back-end developer/ Database manager
- Software Tester
- Web developer

Required Hardware
- Windows desktop
- Android device

Required Software
- ● Firebase Realtime Database
- ● Firebase Cloud Messaging
- ● Firebase Hosting
- ● Firebase Authentication
- ● Android Studio

# Project Estimates

Using data from our different experiences, the Project Cerberus team was able to conclude that this project is doable if we keep up with deadlines and use spare time wisely. Based on the time it took to complete a calendar project in CECS 343 , we agreed that we should try to complete documentation in a quick manner and get coding as soon as possible as many of us have not worked on a full scale programming project like this before. However, due to lack of experience with relevant projects, we are not able to provide exact estimates at this time besides a tentative schedule that will be shown later on in the project plan.

Costs: While we do not plan on spending money to build this project, we did allocate two hundred dollars to the project in the case we needed to pay to use a service (ex. IDE, Lastpass, framework, etc) in order to complete development. The cost will mostly be in person months as we expect this project to take up a lot of time over the next two semesters. Based on this, we will say that the project will take two hundred dollars to complete plus however many person months it take to build.

If this was a real industry job the estimate would most likely be as follows :

Average Pay Security Software Engineer =  $69,131
Total Team Members =          4
Total Payment Owed =  $276,524

The cost of the project itself would most likely be another 250,000 dollars meaning that the total cost for this project would be over half a million dollars.

*Note the cost of this fake project is scaled to a 4 man team. Software projects are many times more expensive*

# Risk Management

**Project Risks**

Major risks for this software is as follows:
- Equipment Failure
- Failure to meet requirements
- Change of requirements
- Change in design
- Technology does not meet expectations

**Risk Table**

Impact will be on a scale from 1-5, 1 being low consequences and 5 being severe when one of the risks below occurs.

| Risks | Probability | Impact |
|---|---|---|
| Equipment Failure | 30 % | 2 |
| Deadline | 50 % | 3 |
| Change of Requirements | 20 % | 4 |
| Change in design | 20 % | 4 |
| Technology is unsuitable | 15 % | 4 |

# Project Schedule

**Project Task Set**

*Process Model*
Our team has decided to use the Unified Process Model during design and implementation
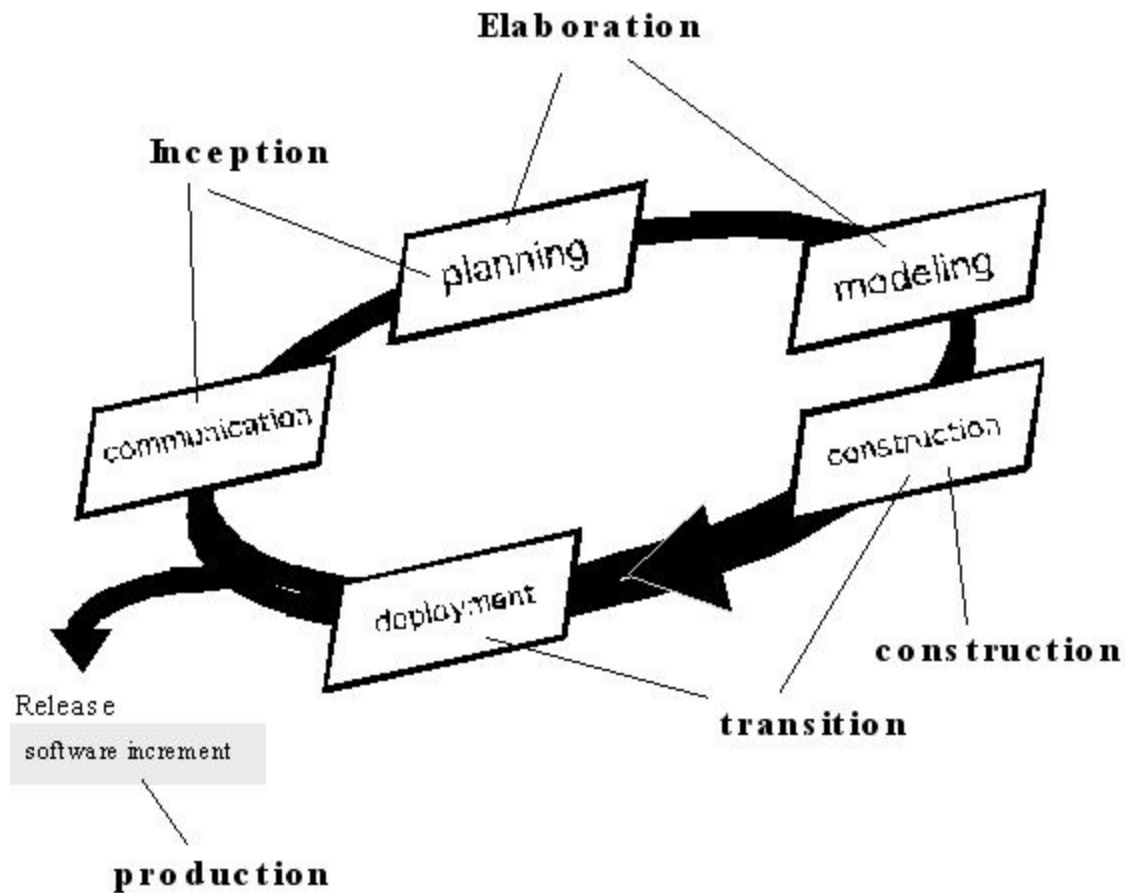


*Framework Activities*
- Risk Analysis
- Programming
- Testing
- Design
- Customer Communication
- Scheduling
- Customer Evaluation on Software's design


*Task Set*
- Requirements specification

- Basic UI interface
- Getting permissions from phone
- Hashing Passwords and Encryption of Phone
- Connecting app and website
- Help Construction
- Testing

# List of Deliverables

Documentation
    Vision Document
    Project Plan
    Use Cases
    Test Cases
    UI Sketch
    User Guide
    UML and Flowchart
Code
    Database
    Complete Interface
    Fingerprint Functionality
    Completed Website
    Complete Product

# Functional Decomposition

*Interface Task Breakdown*
- User Options
- Main Menu
- Password Options
  - View Passwords
  - Edit Passwords
- Database construction

*Website Breakdown*
- UI
  - Main Menu
  - User Options
  - Help

*App Breakdown*
- Input Handler construction
- Object Handler construction
- Image Handler construction
- Menu interactions
- Connection to website

*Testing Task Breakdown*
- Pair programming
- In-house testing

# Timeline Chart

| | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| R=Raymond, D = Daryl, B = Benjamin, C = Carlos | 18-Sept | 25-Sept | 2-Octo | 9-Oct | 16-Oct | 23-Oct | 30-Oct | 6-Nov | 13-Nov | 20-Nov | 27-Nov | 4-Dec | 11-Dec | | |
| Documentation | | | | | | | | | | | | | | | |
| Vision Document | R, D, B, C | | | | | | | | | | | | | | |
| Project Plan | | R, D, B, C | | | | | | | | | | | | | |
| Use Cases | | | R, D, B, C | | | | | | | | | | | | |
| Test Cases | | | | | R, D, B, C | | | | | | | | | | |
| UI Sketch | | | | | | | R, D, B, C | | | | | | | | |
| User Manual | | | | | | | | | R, D, B, C | | | | | | |
| UML and Flowchard | | | | | | | | | | | R, D, B, C | | | | |
| | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | |
| | 22-Jan | 29-Jan | 5-Feb | 12-Feb | 19-Feb | 26-Feb | 5-Mar | 12-Mar | 19-Mar | 26-Mar | 2-Apr | 9-Apr | 16-Apr | | |
| Database | R,C | | | | | | | | | | | | | | |
| User Interface | | | D,B | | | | | | | | | | | | |
| Fingerprint Functionality | | | | | D,B,R | | | | | | | | | | |
| Completed Website | | | | | | | | R,D | | | | | | | |
| Complete Product | | | | | | | | | | R,D,B,C | | | | | |
| Testing | | B, D, C | | | | R,B,D | | | | | | | | | |

# Staff Organization

Team Structure:

Project Cerberus uses a democratic model while making most decisions but final decision goes to the elected team leader. The team will share responsibilities amongst themselves while also assigning a lead in specific areas. The team leader will go to one of these heads if they have a question for a specific part of the project. However, each team member will work on all facets of the project, as we prefer to work in a fluid environment rather than follow a rigid structure. This allows us to work democratically and support each other in the various facets of the project.

**<u>Team Roles:</u>**

Daryl Blancaflor
● Team Leader - Daryl is the team's elected leader and responsible for talking to the class advisor. Communication between advisor and team will be facilitated through the team leader
● Lead Programmer - Daryl is also the team's lead programmer and in charge of making sure team is keeping up with the programming  aspect of the project
● Software Tester - Daryl will help coordinate software testing
● Documentation - Responsible for keeping documentation up to date

Carlos Lomeli
● Lead UI/UX programmer - Carlos is in charge of the user interface for Project Cerberus
● Back-end developer - Carlos is also in charge of some of the back-end development as he has experience with database design
● Documentation - Responsible for keeping documentation up to date

Ray Chin
● Lead Back-end developer - Ray will be in charge of most of the database design
● Web Developer - Ray is also part of the web development team for Project Cerberus
● Software Tester - Ray will assist in Software Testing to make sure final product is ready

- Documentation - Responsible for keeping documentation up to date

Benjamin Guerrero
- Lead Software Tester - Ben is lead software tester due to his intensive interest in making things work correctly
- UI/UX Programmer - Ben will help design and implement the UI/UX for the project and play a key role in making the app accessible to a majority of users
- Documentation - Responsible for keeping documentation up to date

# Management Reporting and Communication

*Mechanisms for Progress Reporting*

Progress of the project will be communicated through e-mail and Trello. All files shared or sent between team members will be done via Trello or Google Drive. These communications will be done informally, unless special or formal documentations is required. Also a test log will be kept for error tracking.

*Mechanisms for Inter/Intra Team Communication*

Our team will conduct meetings three times a week to update each other on their progress. They will also share any concerns or issues that cannot be communicated through electronic communication. However all other communication will be done electronically. Most of this communication will be through Trello or email however can (especially if crucial) be done through phone or text messaging.

Our team will contact our client through email. Although we will also meet with our client twice a week.

# Quality Assurance and Control

Scope and Intent of SQA Activities

The role of the team is to not veer off track from the original product idea. Testing will be done by the team throughout implementation to make sure progress is being made.

### SQA Organizational Role

The role is to review the code and find any errors associated with the application. The team will then discuss the errors and implement ideas on how to fix them. Everyone can be a part of this process, not just the SQA team, and everyone can help one another when someone gets stuck.

# Change Management and Control

### Scope and Intent of SQA Activities

The role of the team is if any new ideas are introduced the job of the team is to make sure that the idea adds better functionality for our application and not just something that will take extra time from our group. Any additions should be made sure that they can be implemented by the deadline and doable by the group.

### SCM Organizational Role

Any new changes or features agreed upon will be discussed to the group leader so that he can communicate with the client to check that those additions are suitable for the application.  Once those changes have been agreed upon they will be able to be implemented into our groups project.