SE 3XA3: Development Plan
Title of Project

Team 28, Tuples1
Suhavi Sandhu, sandhs11
Shabana Dhayanath, dhayanas
Joseph Lu, luy89

December 6, 2017

Contents

List of Tables

List of Figures

Table 1: **Revision History**

Date	**Version**	**Notes**
2017-09-29	0.0	Initialization
2017-12-06	1.0	Revision 1

Development Plan of our product Password Protection Program.

1 Team Meeting Plan

The team will be meeting ~~twice~~ once a week, every week outside of labs, on Fridays at Thode library. ~~We have decided to rotate the roles every other week to split responsibilities equally.~~

Shabana - Researcher, Encryption
Suhavi - Team Leader, GUI
Joseph - Scribe, Database

The meeting agenda will be filled out by the scribe after every meeting and uploaded to the git repository for all members to access. Below is a template for the meeting minutes. It notes down the details of the meeting and also provides a guideline for the meetings progression. Ideally, we will start by updating the rest of the team with what we have accomplished since the last meeting, followed by items that need to be addressed (problems, opinions, questions), and at the end, what needs to be done by the next meeting. There will also be a written statement on decisions made during the meeting. Meeting Minutes Template

2 Team Communication Plan

The team will primarily use the phone to communicate any issues, changes and progress made on the project. For minor announcements, there will be a Facebook group where we can schedule and confirm meeting times. If there is an important issue that has to do with the code or design of the project, we will be using git issues. When working on the reports, we will be using Google Docs to allow everyone to contribute at the same time and then the scribe will format them into latex for uploading.

Table 2: **Meeting Minutes Template**

Meeting Minutes			
Meeting no	**Location**	**Meeting Date**	**Attendees**
Update on what happened since last meeting:			
Issues discussed this meeting:			
What to do until next meeting:			
Decisions made:			

3 Team Member Roles

Team Leader - Schedules meetings and delegates work to the other members as well as him/herself. Ideally, this member will make sure the git repo has no conflicts and looks clean.

Scribe - Will be in charge of writing meeting minutes as well as writing reports using Latex. This person ideally has experience using Latex.

Researcher - Will actively learn about the technology we will be implementing in our project. In our case, this will heavily focus on cryptography and documentation from the original product.

Programmer - All team members will play the role of the programmer as there will be a lot to code.

4 Git Workflow Plan

Branching Workflow

- ~~master - branch to host our stable code~~
- ~~develop - branch to develop our program~~
- each ~~additional~~ branch is used to implement specific parts of the program

Tags will be use when major milestones are reached or there is a major issue that needs attention.
Milestones will be used to divide our workload and allow the users to work in parallel.

5 Proof of Concept Demonstration Plan

5.1 Will implementation be difficult?

The hardest part of the program for us to implement is probably going to be integrating the database into our Python GUI program as our developers have not formally programed in ~~MySQL~~ SQLite. As we are currently taking a databases course, we will take advantage of those resources and ask the teaching assistants and instructor for clarification when needed.

5.2 Will testing be difficult?

~~Testing the encryption on our program may be difficult as we are not hackers and do not really have experience or knowledge in trying to break through encryptions.~~ The testing of the Python cryptography library that we will be using for the project is out of our scope as we do not have the necessary knowledge or techniques to do so. Instead, we will be assuming that as it is a Python library, Python has conducted sufficient testing to ensure the security of the library. Therefore, we will mainly focus on testing the user interface. ~~We will also be researching the Python cryptography library that we intend to use to ensure its credibility.~~

5.3 Is a required library difficult to install?

The required libraries should be easy to install with the python package manager pip. The libraries that will need to be installed for the project are cryptography, peewee, pyperclip and tkinter.

5.4 Will portability be a concern?

Portability should not be a concern as long as all the dependencies are downloaded.

6 Technology

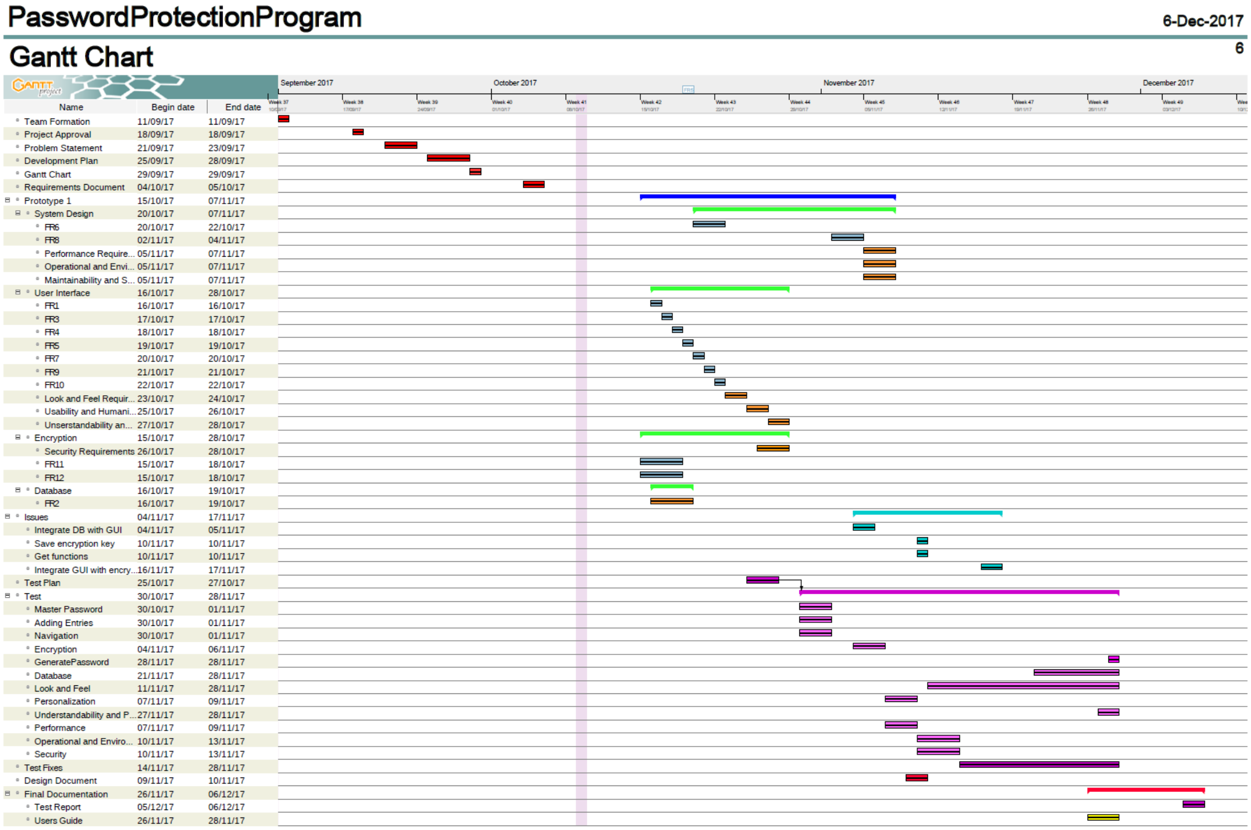
We will be using Python as the main language to create our application. Our main library to create the GUI is the Python library tkinter. The encryption method we will use is provided by the python library cryptography. The primary IDE we will use is IDLE and the primary form of testing will be done on pyunit. Our documentation generation will be done with Doxygen and our reports will be written in latex. The database that will store the usernames and passwords will be ~~MariaDB~~ SQLite via peewee.

7 Coding Style

We are planning to following the Python coding style
Python Coding Syle

8 Project Schedule

Gantt chart



PasswordProtectionProgram

6-Dec-2017

Gantt Chart

6

Name	Begin date	End date
Team Formation	11/09/17	11/09/17
Project Approval	18/09/17	18/09/17
Problem Statement	21/09/17	23/09/17
Development Plan	25/09/17	28/09/17
Gantt Chart	29/09/17	29/09/17
Requirements Document	04/10/17	05/10/17
⊟ Prototype 1	15/10/17	07/11/17
⊟ System Design	20/10/17	07/11/17
FR6	20/10/17	22/10/17
FR8	02/11/17	04/11/17
Performance Require...	05/11/17	07/11/17
Operational and Envi...	05/11/17	07/11/17
Maintainability and S...	05/11/17	07/11/17
⊟ User Interface	16/10/17	28/10/17
FR1	16/10/17	16/10/17
FR3	17/10/17	17/10/17
FR4	18/10/17	18/10/17
FR5	19/10/17	19/10/17
FR7	20/10/17	20/10/17
FR9	21/10/17	21/10/17
FR10	22/10/17	22/10/17
Look and Feel Requir...	23/10/17	24/10/17
Usability and Humani...	25/10/17	26/10/17
Unserstandability an...	27/10/17	28/10/17
⊟ Encryption	15/10/17	28/10/17
Security Requirements	26/10/17	28/10/17
FR11	15/10/17	18/10/17
FR12	15/10/17	18/10/17
⊟ Database	16/10/17	19/10/17
FR2	16/10/17	19/10/17
⊟ Issues	04/11/17	17/11/17
Integrate DB with GUI	04/11/17	05/11/17
Save encryption key	10/11/17	10/11/17
Get functions	10/11/17	10/11/17
Integrate GUI with encry...	16/11/17	17/11/17
Test Plan	25/10/17	27/10/17
⊟ Test	30/10/17	28/11/17
Master Password	30/10/17	01/11/17
Adding Entries	30/10/17	01/11/17
Navigation	30/10/17	01/11/17
Encryption	04/11/17	06/11/17
GeneratePassword	28/11/17	28/11/17
Database	21/11/17	28/11/17
Look and Feel	11/11/17	28/11/17
Personalization	07/11/17	09/11/17
Understandability and P...	27/11/17	28/11/17
Performance	07/11/17	09/11/17
Operational and Enviro...	10/11/17	13/11/17
Security	10/11/17	13/11/17
Test Fixes	14/11/17	28/11/17
Design Document	09/11/17	10/11/17
⊟ Final Documentation	26/11/17	06/12/17
Test Report	05/12/17	06/12/17
Users Guide	26/11/17	28/11/17

Figure 1: Gantt Chart

9 Project Review

When looking back on the development plan after finishing the project, we see that we may have been a bit ambitious in the organization of roles and work. Although we had initially intended to rotate the member roles every week, we were quick to realize that this wasn't really feasible and hard to keep track of in the long run. Therefore, the major roles were left as initially assigned throughout the project.

Also, we had hoped to have 2 meetings a week, we could rarely get a significant amout of work done during the lab periods so instead we often met of Fridays so that we could work on a large portion of the project at once.

The team communicated with each other as intended and used separate git workflow branches for the different parts of the code which really facilitated staying on track with the implementation even though we were all working on separate parts.