CIS 422 Project 1:
Ducks on Deck
Project Plan

Ellie Kobak (erk), Kalyn Koyanagi (kek), Liza Richards (ljr), and Kelly Schombert (ks)
January 30, 2022 – v2.0

Table of Contents

1. Project Plan Revision History

Date	Author	Description
1-8-2022	ks	Created the initial document and wrote first draft to be proposed at second group meeting.
1-10-2022	ks	Revised Organization and Roles section and Build plan section
1-20-2022	ljr	Added section 7 (group meeting records)
1-21-2022	ljr	Made changes to section 3 (Work Breakdown Schedule)
1-24-2022	ljr	Made changes to section 7
1-26-2022	ljr	Made changes to section 4 (Monitoring and Reporting)
1-28-2022	ljr	Made changes to section 7.
1-29-2022	ljr	Final draft of project plan is finished.

2. Management Plan

2.1. Organization and Roles

Each group member is assigned the following role which includes the outlined responsibilities:

- Writing lead: Liza Richards
    - The writing lead will verify the quality and completion of the writing in project documents such as the SRS and SDS.
    - The writing lead will assign writing related work to other members while monitoring progress.
- Record keeping: Ellie Kobak
    - The record keeper will ensure that task assignment and completion is being recorded in a complete and timely fashion.
    - The record keeper will summarize any decisions made or issues brought up during group meetings.
    - The record keeper will make sure that any revisions to records or documents are properly noted and archived.
- GUI Lead: Kalyn Koyanagi
    - The GUI lead will monitor progress and verify completion of assigned implementation tasks pertaining to visual and user interface components
    - The GUI lead will routinely check the quality of the GUI code such as performance, style, maintainability etc.
    - The GUI lead will be referred to when any minor design decisions pertaining to visual and user interface components need to be made.
- Data management lead: Kelly Schombert
    - The data management lead will monitor progress and verify completion of assigned implementation tasks.
    - The data management lead will routinely check the quality of the backend code such as performance, style, maintainability etc.
    - The data management lead will be referred to when any minor design decisions pertaining to visual and user interface components need to be made.
- Test lead: Luying Cai
    - The test lead will develop a thorough and robust set of program test cases for the debugging stage. The test lead will assist in developing component test cases where needed.
    - The test lead will check throughout the development process that the currently implemented components are being robustly tested by group members.
    - The test lead will make sure that any issues or bugs that have been raised are corrected in a timely fashion and assign debugging tasks.

The type and number of responsibilities for each member will be flexible to accommodate any issues or imbalances in workload or unforeseen circumstances that may arise. Every member will participate in routine meetings and communicate with each other outside of meetings to make decisions, be assigned tasks, and track the progress as the project goes on.

When a large decision (any decision that may impact other parts of the development process) must be made, the relevant lead will decide on a course of action individually and then propose their verdict to the group. The group will discuss if there are any alternatives that may be preferred or if there are concerns with the decision. If there are no concerns or alternatives, the lead's original decision is carried out. Otherwise, a group decision is made and carried out instead. Smaller decisions that will not impact other pieces to the development process will be left up to the appropriate lead.

2.2. Meetings and Communication

Group members will attend regular formal meetings at the following times:

- Sundays @ 4:30 pm on Zoom
- Fridays @ 12:00 pm on Zoom
- After class meetings on Mondays and Wednesdays in person

- Record of meetings is located here
- *Additional meetings will be scheduled as needed*

Meetings will take place either over Zoom or in the Knight Library depending on the health status of each individual. Each meeting will last 60-80 minutes depending on agenda items and as time permits for individual members.

Group members will attend regular informal check-ins after each CIS 422 class period.

Check-ins will take place in the Price Science Commons with whichever members are available and present for a check-in.

Group members will have discussions and report their progress outside of meetings via the following communication methods:

- Group SMS messaging
- Discord

Members are expected to read such messages to stay up to date on any updates or issues that might arise during development.

3. Work Breakdown Schedule

- Week 1 (1/5 – 1/8)
    - Create build plan
    - Create project plan
    - Create working drafts of the SRS and SDS
    - (Initial project documents due on Tuesday, Jan. 11)
- Week 2 (1/9 – 1/15)
    - Gather sample data and create test files
    - Assign implementation tasks to all members
    - Build non-GUI components such as structures for holding student data, randomizing algorithms, and systems of reading to log files or from saved files
    - Build queue structure that holds student data
    - Successfully parsing names into queue structure from test file imports.
    - Randomizing system for queue with student data
    - Build GUI components dealing with visual output and user interface
    - Finish visual windows (main menu, on deck)
    - Successfully have a visual window displayed on the user screen from the start of run time.
    - Successfully importing randomized names into the “On Deck” window.
    - Finishing exporting functions in fileWriter
    - Incorporate working keystrokes into the visuals
    - Combine all components to have an initial executable system ready for testing
- Week 3 (1/16 – 1/22)
    - Meet with Professor Anthony Hornof for discussion of project design, requirements, and tasks.
    - Keystrokes usage connected to writing to record keeping files
    - User has ability to export and access record keeping files generated during usage
    - Begin debugging phase on individual files
    - Make final changes to visual aesthetics
    - Combine all components to have new executable system ready for testing
    - If on or ahead of schedule, consider implementation of ‘nice to have’ requirements or improvement to required implementation
    - Review and update documentation

- Week 4 (1/23 - 1/30)
    - Complete file reader and file writer files (non-GUI components)
    - Complete buildQueue and visual components
    - Combine all components to have an executable system ready for testing
    - Continue testing using the test files created
    - Practice demo for presentation
    - Have documentation fully updated
    - Have a stable program to release by Friday, Jan. 28.
    - (Project due on Sunday, Jan. 30)

The responsibility for monitoring progress of each milestone will be in accordance with the roles outlined under the **Management Plan** section. Members will likely have roughly equal distribution of implementation tasks.

4. Monitoring and Reporting

Tasks that have a completion that can be objectively verified will be recorded to a shared spreadsheet. This spreadsheet will also note the group member in charge of completing this task and by what date the task should be completed by. The spreadsheet will be frequently updated to show every self-reported task completion and verification by another group member. Ideally, verification will be done by the relevant lead member or the record keeper.

5. Build Plan

5.1. Plan Details

As further outlined in the SDS, the program will consist of 5 components implemented through 5 files. These files are: buildQueue.py, fileReader.py, fileWriter.py, and DucksOnDeck.py.

The first files to write are fileReader.py and buildQueue.py. fileReader.py reads an imported tab-delimited file containing the class roster from the instructor and stores any student data necessary for the remaining program execution. buildQueue.py uses the class data obtained by fileReader.py to create a randomized queue of students that determines which students will be “on deck”.

After the components concerned with the organization and storing of data are implemented, then work begins on the visual, user interface, and data writing components. These components are written through the DucksOnDeck.py and fileWriter.py files. DucksOnDeck.py handles the display of “on-deck” names as well as the key controls used by the instructor. fileWriter.py responds to key controls from DucksonDeck.py and writes the relevant student data to log files on the system.

Once each component can execute its individual functionalities and have been tested for appropriate requirements, the final component to link all other components together will be DucksOnDeck.py. This file links the components concerned with receiving and organizing data to the visual and data writing components that need information stored in the backend.

After reaching this buildable state, any remaining ‘must-have’ requirements will be implemented and the possibility of implementing remaining ‘should-have’ or ‘nice-to-have’ requirements will be evaluated.

All resources should be focusing on testing and debugging to reach a stable release candidate by the project deadline unless optional requirements and functionality are being added.

5.2. Rationale

The rationale behind breaking down the system into four components is to separate the most prominent “must-have” requirements into manageable pieces. When the components are separated, the group is also able to ensure that the pieces work individually before attempting to combine the system. This modularity will also give the best environment to attempt incorporating “nice-to-have” requirements.

Development of the backend (non-GUI) and GUI components have been separated to allow group members to work in areas that within their expertise or that they find interesting.

We do expect team members to have more difficulty in creating and integrating the GUI elements as no members have had experience with GUIs. We will have to assess the difficulties that arise with implementing the visual and user interface components early on and reallocate team members accordingly. In addition, even members who are not part of the initial development of GUI components should be required to make themselves familiar with the relevant libraries.

6. Acknowledgements

This template was given to the UO CIS 422 class by Anthony Hornof. This template is similar to a document produced by Stuart Faulk in 2017, and uses publications cited within the document, such as IEEE Std 1362-1998 and ISO/IEC/IEEE Intl Std 29148:2018.

7. Record of Group Meetings

Whole Team Meeting on Zoom 1/06/2022:

1. Agenda:
    a. Discuss build plan
    b. Plan different steps of completing build plan, prioritize some over others
    c. Divide work of finishing project plan, SRS, and SDS between group members
2. Notes
    a. Started at 6:00 pm
    b. Liza created shared drive with all docs
    c. Start with small and focus on getting the queue working with the terminal, then focus on visuals
    d. Make list of everything that needs to be done
    e. Next meeting Sunday 01/09 at 10 am
    f. Make sure we take as much from existing SRS and SDS but tailored to our project
    g. For SRS: delete sections we do not want to include
    h. Ended at 7:00 pm

Whole Team Meeting on Zoom 1/09/2022:

1. Agenda:
    a. Go over progress on initial project plan
    b. Make plans for finishing project plan
    c. Finalize meetings times and roles
    d. Discuss build plan
2. Notes:
    a. Started at 10:00 am
    b. Make build plan - brainstorm individually and then meet and go over ideas and then decide roles

c. Make project name - Duck pun
d. Meeting times
    i. Also establish work times that are not mandatory for everyone
    ii. Short meetings 20-30 minute after lecture
    iii. Stay in PSC after lecture
e. Set 4 names on deck for now, maybe change to 1-9 option later
f. Edge case for smaller classes
    i. If there are 4 students in the class and 4 on deck, all 4 students automatically on deck
    ii. Or make queue auto set to students - 1 of total class if 4 or less

g. Dict might be too complicated with keeping track of names, lists better?
h. Have two lists one to be called on and one that was called on
i. Create a random number for each name?- may also be too complicated
j. Use cases - take from packet of given use cases
k. Build plan
l. Brainstorm
    i. Luying - include all of user case
    ii. Small demo processor logs when
    iii. We need to keep record of students of what students who need to be called and more info stored on how who has been called on
    iv. For emailing a student, instructor needs access to email address
    v. Kelly - for the GUI use tkinter as part of python standard library,
    vi. Hard part will be out it interacts with backend
    vii. One of more py files and scripts that corresponds with tkinter data
    viii. Keep in file and only access when necessary?
    ix. Holds info on its own
    x. Another python script pulls name to display
    xi. Need to think what DS needed?
    xii. Do we need to make a roster file? Is it on the instructor to make TDF?
    xiii. ASSUME File is provided in right format
    xiv. Backend works with queue
    xv. Key mapping - tkinter has key mapping? Can tkinter receive signals?
    xvi. Need to be able to flag and unflag student - flag only seen in log files
    xvii. Liza - need to make sure how files are overwritten
    xviii. Figure out random number generator - shrink list? Can create a lot of inefficiencies
    xix. Mentioned - a modal dialogue box for errors
    xx. Queue needs n% of queue - choosing the right students
    xxi. Keep names in a queue for picking?
    xxii. Have large list of same queue but only choose deck from top n% of queue
    xxiii. Maintain first original
    xxiv. Modules
        1. Main view
        2. Roster
        3. Instructor controls
        4. Exported data
    xxv. Casual informal meetings 20-30 minutes after class
    xxvi. Formal meetings with agenda Sunday and Fridays over zoom or in person
    xxvii. Rotate moderators every term,

xxviii. Have meeting jobs and project jobs

m. Ended at 11:45 pm

Liza and Luying Meeting on Zoom 1/09/2022:

1. Agenda:
    a. Discuss SRS
    b. Divide and conquer SRS work
2. Notes:
    a. Started at 2:00 pm
    b. Asked and resolved questions regarding the different SRS sections.
        i. clarifying questions on how to answer questions in sections 2.1 to 2.5.
    c. Figured out how to divide the SRS between them evenly.
        i. Liza does sections 2.6 and all of 3
        ii. Luying does sections 2.1 to 2.5
    d. Ended at 2:30 pm

Whole Team Meeting in person 1/10/2022:

1. Agenda:
    a. Progress check on initial project plan
    b. Review what needs to be done on each document and divide that work
    c. Choose tasks for each group member
2. Notes:
    a. Started at 8:00 pm
    b. Each team member goes over what work they have done, what work they have left, and what they are held back on
        i. Kelly finished project plan
        ii. Liza and Luying finish SRS
        iii. Kalyn and Ellie working on SDS, need help to finish
    c. Divided up leftover work to do on documents.
        i. Liza helps on SDS
        ii. Kelly makes diagrams for SDS
    d. Each group member mentioned what task they would prefer to do and from this, tasks were assigned to each group member
        i. Liza - writing lead, Ellie - record keeping lead, Kalyn - GUI lead, Luying - testing lead, Kelly - data management lead
    e. Ended at 8:30 pm

Whole Team Meeting in person 1/12/2022:

1. Agenda:
    a. Progress Check
    b. Make sure everyone knows what tasks they are doing and how they can do them
    c. Review which tasks to prioritize over others
2. Notes:
    a. Started at 11:25 pm
    b. Confirmed responsibilities of each team member

i. Ellie - working on file reader, first prioritizing parsing imported files into queue structure
ii. Liza - updating SRS, SDS, project plan first but will get started on presentation and other documentation needed, research how to connect keystrokes to GUI
iii. Kelly - forming test files and end of term summary files that will be written do during system runtime
iv. Kalyn - first prioritize getting a generic display window to pop up, then focus on getting buttons to work on display.
v. Luying - forming tests and writing main.py

c. Ended at 12:00 pm

Whole Team Meeting on Zoom 1/14/2022:

1. Agenda:
    a. Very briefly discuss tasks and progress
2. Notes:
    a. Start meeting at 12:30
    b. Project repository built on github and shared with all team members
    c. Make sure everyone knows what task they are doing
    d. Everyone states what they are specifically working on regarding their task
        i. Ellie: Starting with parsing test data into a queue
        ii. Liza: Starting to work on keystrokes
        iii. Kalyn: Starting to work GUI and having basic window display using Tkinter
        iv. Kelly: Working on forming test data and file writer mechanics
        v. Luying: Working on how to write tests for program
    e. End meeting at 12:45

Whole Team Meeting on Zoom 1/16/2022:

1. Agenda:
    a. Progress check on all tasks
    b. Discuss what is left to be done on each task and the order in which remaining portions will be completed
    c. Ask if anyone needs clarification on any of their tasks
2. Notes:
    a. Started at 4:00 pm
    b. Ellie is still forming the queue parsing system for the files
    c. Liza is working on editing the SRS, SDS, and project plan for final submission
    d. Kelly has formed the test files and began working on file writing system
    e. Kalyn is still looking into how to use Tkinter and has basic window display running
    f. Clarification among the whole group about how to connect all the python files for them to work together.
    g. So far, no one needs help with tasks and are still working on how to make their different portions of the project work.
    h. Ended at 5:00 pm

Kalyn and Liza Meeting on Zoom 1/20/2022:

1. Agenda:
    a. Discuss GUI, specifically keystrokes
    b. Discuss how keystrokes will correspond to GUI and other files
2. Notes:
    a. Started meeting at 4pm
    b. Discussed how the files interact with one another
        i. Specifically how the keystrokes will lead to changes in the files to be exported
    c. Discussed Tkinter mechanics and potential options for how to implement keystrokes using Tkinter

d. Ended meeting at 4:30 pm

Instructor Meeting on Zoom 1/20/2022:

1. Agenda:
    a. Five minute overview of project
        i. Project name
        ii. Team members
        iii. Problem that the system solves
        iv. How does the system solve the problem
        v. Overview of software design
        vi. List of technologies being used
        vii. Up-to-date project timeline
        viii. Who is working on what
        ix. What is done so far
        x. What is being worked on now
        xi. What is preventing progress
    b. Each team member discussed what they have done, what they are working on, what is left to work on
2. Notes:
    a. Started at 5:00 pm
    b. Liza tells instructor her tasks, progress on tasks, and what is left to be done
        i. writing lead, working on all the documents, started on presentation.
    c. Ellie tells instructor her tasks, progress on tasks, and what is left to be done
        i. record keeper, using spreadsheet, making progress on the fileReader and buildQueue, not working yet
    d. Kalyn tells instructor her tasks, progress on tasks, and what is left to be done
        i. GUI, has basic window displayed but needs to make progress on the details like keystrokes and such
    e. Kelly tells instructor her tasks, progress on tasks, and what is left to be done
        i. File writer making progress, debugged together
    f. Luying tells instructor her tasks, progress on tasks, and what is left to be done
        i. testing lead, going to contribute to main.py
    g. Ended at 6:00 pm

Whole Team Meeting in person 1/20/2022:

1. Agenda:
    a. Progress Check
    b. Debrief from instructor meeting
    c. Check in with all team members
2. Notes:
    a. Started at 6:10 pm
    b. Checked in with each team member to ask how they felt about the instructor meeting
    c. Clarified how we should proceed with the given instructor feedback

d. Divide work to give more tasks to Luying since she is the testing lead and couldn't do much until most of the system is complete
    i. Luying helps with main.py more with Kelly
e. Ended at 6:20 pm

Whole Team Meeting on Zoom 1/23/2022:

3. Agenda:
    a. Check in with each team member
    b. Progress check on each member's tasks
    c. Clarification on software mechanics
    d. Discuss how to divide up work of lost team member
    e. Answer questions
4. Notes:
    a. Started at 4:15 pm
    b. Ellie is making progress with the file reader, most of it is done but she is working on getting it to parse more accurately still
    c. Kelly asks clarification questions on how the different portions of the system interact with one another
        i. importing file from teacher then the file reader goes through and parses this input into a queue, gives the queue to visuals, responses in visuals then go into file writer to update the files that will be exported
    d. Discussing the loss of a team member, how we should proceed and divide up her work
        i. Decided that Kelly is taking the lead on main.py
        ii. Test cases aren't necessarily needed.
    e. Ended at 5:45 pm

Whole Team Meeting in person 1/24/2022:

1. Agenda:
    a. Progress report on tasks
    b. Discuss what each member is stuck on and how they need help
    c. Discuss mechanics of program
    d. Discuss what is left to be done and when it will be done by
2. Notes:
    a. Start meeting after class at 11:25
    b. Ellie confirms with us the structure of her file parsing system, shows team how it works and team provides clarification for system mechanics
    c. Kelly and Kalyn clarify how file writing, file reading, and visuals should all work together
    d. Liza asks clarifying questions on documents from instructor feedback and updates team members with progress on all documents.
    e. End meeting at 12:40

Whole Team Meeting in person 1/26/2022:

1. Agenda:
    a. Progress report on tasks
    b. Discuss what each member is stuck on and how they need help
    c. Discuss mechanics of program
    d. Discuss what is left to be done and when it will be done by

2. Notes:
    a. Start team meeting after class 11:25
    b. Talked about how we wanted our visual display window to look, what color background, buttons, and text
    c. Helped Ellie debug the parser, not splitting in the right spots
    d. Got the file to split in the right spot (made it split on commas instead of spaces)
    e. New line characters were not being stripped from the lines
    f. Figured out how to get new line characters stripped
    g. Working on user inputting the file name?
        i. file names vary among users
    h. Ended at 12:40

Whole Team Meeting on Zoom 1/28/2022

1. Agenda:
    a. Progress report on tasks
    b. Go over what tasks are left
    c. Work to debug errors in each file
    d. Review documents as a group and take comments
2. Notes:
    a. Start team meeting at 12:10 pm
    b. Went through how the system works right now
        i. Current problem is when you move between names using arrow keys, the previous name does not de-highlight
    c. Need to finish SDS, programmers documentation, and users documentation
    d. Revises how we run the program, can run from visual.py so we rename that file to DucksOnDeck.py
    e. get rid of main, just run directly from visual
    f. Ellie helping on SDS, Kelly redoing SDS diagrams
    g. Add documentation to each python file
    h. went over how each of the files interacts with one another again
    i. drew diagrams
    j. DucksOnDeck.py fixed
    k. End team meeting at 2:45

Whole Team Meeting on Zoom 1/29/2022

1. Agenda:
    a. Progress report on tasks
    b. Go over what tasks are left
    c. Go through all the installation steps as a group
    d. Review documents as a group and take comments
2. Notes:
    a. Start team meeting at 2:30 pm
    b. Went through how the system works right now
        i. Working system, problem with exporting the daily log/performance summary?
        ii. Creates a text file in the working directory which can be opened by the user
    c. Most of the work needs to be done on the SDS, need new diagrams
    d. Went through installation instructions
        i. did a walkthrough of what it would look like for a user to use the system, did a practice demo
        ii. wrote down each step for the installation instructions and had every group member confirm the path of steps
    e. Ended meeting at 3:15

Whole Team Meeting in person 1/30/2022

1. Agenda:
    a. Progress report on tasks
    b. Go over what is left to finish the project
    c. Divide what tasks are left amongst the group
    d. Make final submission of project on canvas.
2. Notes:
    a. Start meeting at 2:00
    b. Most of the work left is on the SDS
        i. revision to diagrams and writing
    c. Everyone goes through the programmer and user documentations and make revisions as needed
    d. Made changes to directory structure
    e. End meeting at 5:00

8. Monitoring and Reporting Spreadsheet

Date	Name	Time (min)	Task
Week 1: (Jan. 2-8)			
1/6/2022	Whole Team	60	Meeting
1/6/2022	Liza	20	Created shared drive with all the docs
1/6/2022	Ellie and Kalyn	145	Created initial SDS document and began writing first draft
1/8/2022	Liza and Luying	120	Created initial SRS document and began writing first draft to propose at next meeting
1/8/2022	Kelly	200	Create initial project plan document and began first draft of project plan
Week 2: (Jan. 9-15)			
1/9/2022	Whole Team	75	Meeting 10am
1/9/2022	Luying and Liza	30	Meeting at 2pm
1/9/2022	Kalyn and Ellie	125	Meeting at 3pm
1/9/2022	Whole Team	105	Meeting
1/9/2022	Kalyn	175	Added sections to SDS
1/9/2022	Ellie	190	Added UML diagrams to SDS
1/9/2022	Liza and Luying	135	Updated sections of the SRS

1/10/2022	Whole Team	30	Meeting
1/10/2022	Liza and Luying	120	Updated sections of the SRS
1/10/2022	Kelly	80	Revised project plan sections
1/10/2022	Ellie	75	Added sections to SDS
1/11/2022	Ellie and Kalyn	200	Finished first working draft of SDS
1/11/2022	Liza	120	Finished first working draft of SRS
1/11/2022	Kelly	180	Finished first working draft of the project plan
1/11/2022	Liza	2	Submission of the initial project plan on canvas
1/12/2022	Whole Team	35	Meeting
1/14/2022	Whole Team	15	Meeting, divided tasks
Week 3 (Jan. 16-22)			
1/16/2022	Whole Team	60	Meeting, update on tasks
1/17/2022	Whole Team	35	Team meeting after class to discuss logistics of system and make sure everyone is making progress on tasks
1/19/2022	Whole Team	35	Team meeting after class to discuss logistics of system and ensure everyone knows their assigned tasks
1/20/2022	Whole Team and Instructor	60	Instructor meeting at 5pm
1/20/2022	Kalyn and Liza	30	Meeting, update on GUI components
1/20/2022	Whole Team	10	Meeting to debrief from instructor meeting
1/20/2022	Kelly	10	Created fileWriter.py
1/20/2022	Kalyn	5	Created fileReader file

1/20/2022	Kalyn	5	Created buildQueue file
1/20/2022	Kalyn	60	Created visual file, basic window displayed
1/21/2022	Kalyn	90	Added keystrokes to visual file
1/22/2022	Liza	60	Updated keystrokes in visual file
1/22/2022	Ellie	25	Modified initial buildQueue component and fileReader component.
1/22/2022	Liza	90	Created and made first draft of user documentation
1/22/2022	Kalyn	90	Modified initial visual component, added buttons to main menu visual
Week 4 (Jan. 23-30)			
1/23/2022	Ellie	30	Initial version of fileReader file created
1/23/2022	Kalyn	240	Finished menu and deck windows
1/23/2022	kelly	3	FileWriter.py moved to new repository
1/23/2022	Liza	120	Created and wrote the first draft of programmer documentation
1/23/2022	Liza	90	Created initial document and wrote first draft of installation instructions
1/23/2022	Whole Team	90	Meeting, debugging and working
1/24/2022	Liza	180	Revisions made to technical documents
1/24/2022	Liza	60	Wrote README
1/24/2022	Whole Team	75	Team meeting after class to discuss logistics of system and make sure everyone is making progress on tasks
1/24/2022	Kalyn	60	Made changes to main menu screen,
1/24/202	Liza	120	Finished project plan, made changes to technical

2			documents
1/25/2022	Kelly	120	Daily log functionality implemented in fileWriter.py
1/25/2022	Kalyn	60	Updated deck user info complete
1/26/2022	Ellie	20	Modified fileReader file
1/26/2022	Kelly	20	Export functions added to fileWriter.py
1/26/2022	Kelly	160	Export functions tested and removed, summary performance dictionaries implemented to fileWriter.py
1/26/2022	Kelly	35	fileWriter functions tested individually and with fileReader
1/26/2022	Kelly	170	Completed daily log and end of term summary files
1/26/2022	Whole Team	75	Team meeting after class to discuss logistics of system and make sure everyone is making progress on tasks
1/26/2022	Liza	75	Finished SRS, made changes to technical documents, installation instructions and README
1/27/2022	Ellie	150	Finalized a working fileReader file
1/27/2022	Kelly	65	fileWriter functionality implemented
1/27/2022	Kelly	45	fileWriter good documentation completed
1/27/2022	Ellie	30	Finished file reader and build queue
1/27/2022	Liza	75	Made changes to technical documents, installation instructions, and README
1/28/2022	Whole team	155	Went over working system, figured out what else needed to be done
1/28/2022	Kalyn	60	Finished functionality
1/29/2022	Whole Team	45	Meeting going over remaining tasks
1/29/2022	Kalyn	90	Finished final modifications, modified aesthetics of windows

1/30/2022	Whole Team	180	Meeting going over remaining tasks and finishing them
1/30/2022	Whole Team	2	Compeleted project submitted on canvas
	total time (in hours)	93.36666667	