# PROJECT PLAN

Internship PLS

*Professor Adkisson*

*SWE 3313 | Section 02*

# Table of Contents

# Scope

## Customer

I. Add new customer info
    a. Accept full name
    b. Accept phone number
    c. Generate GUID
II. View existing customer info
    a. Display customer list
    b. View reward points

## Orders

I. Add drink
II. Add modifications
III. Remove Modifications
IV. Remove drink
V. Remove all drinks
VI. Change drink quantity
VII. Cancel order
VIII. Review order
    a. Display list containing:
        i. Items
        ii. Modifications
        iii. Prices
        iv. Taxes
        v. Subtotal
        vi. Total
IX. Complete order
    a. Order confirmation
    b. Receipt
        i. Items
        ii. Modifications
        iii. Prices
        iv. Taxes
        v. Subtotal
        vi. Total
        vii. Payment Method
        viii. Date and Time

## Payment

I. Anonymous Customer
    a. No reward points
    b. Only pay by card
II. Rewards Customer
    a. Payment by reward points
        i. Must cover entire cost
        ii. Cannot split with card
        iii. Reduce reward points
        iv. Display new reward point total
    b. Payment by card
        i. Add reward points
        ii. Display new reward point total
III. Accept credit card info
IV. Validate credit card info
V. Cancel payment
VI. Notify of success/failure

## Data Management

I. Read JSON file
    a. Menus
    b. Configuration data
    c. Drink offerings
    d. Modification offerings
    e. Customer data
II. Write JSON file
    a. Customer data
    b. Sales data

## User Interface

I. Main Screen
    a. Order button
        i. Starts order as anonymous customer
    b. Customer List button
    c. Management Tools button
II. Customer List Screen
    a. Customer List
        i. Anonymous customer first
        ii. Sorted by last name, first name, phone number

iii. Displays reward points
iv. Order button for each customer
b. Add New Customer button
c. Return to Main Screen button

III. Add Customer Screen
a. Required Fields
i. First name
ii. Last name
iii. Phone number
1. Must be unique
b. Submit button
i. Add the customer to Customer List
ii. Continue to Order Screen
c. Cancel button
i. Returns to Main Screen

IV. Order Screen
a. Drink List to add to order
i. Price info
b. Modification List to add to drink
i. Price info
c. Order Summary
i. List of drinks in order
ii. List of modifications for each drink
iii. Subtotal
iv. Total
v. Tax
d. Proceed to Payment Button
i. Unclickable until at least one drink is ordered
e. Cancel Order Button
i. Returns to Main Screen

V. Payment Screen
a. Display reward points
b. Pay with rewards button
c. Pay with card button
i. Credit card number field
ii. Cardholder name field
iii. Card expiration date field
iv. Security code field
d. Submit Button
i. Continues to Receipt Screen

e. Cancel Order Button
    i. Returns to Main Screen

VI. Receipt Screen
a. List of ordered drinks
b. Modifications to each drink listed under drink
c. Quantity of each drink/modification
d. Price of each drink
e. Subtotal
f. Tax
g. Total
h. Payment Method
    i. Credit card
        1. Display the last four digits
    ii. Reward Points
        1. Display the number of redeemed points
i. Display remaining reward points for rewards customers
j. Continue Button
    i. Returns to Main Screen

VII. Management Screen
a. Generate Sale Report button
b. Return to Main Screen button

## Schedule

| Sprint # | Task # | Description | Due Date |
|---|---|---|---|
| 0 | 01.0 | Project Plan | Oct. 09 |
| | 02.0 | Requirements | Oct. 12 |
| | 03.0 | UI Design (No functionalities, only buttons) | Oct. 24 |
| | 03.1 | Build Main menu | Oct. 15 |
| | 03.2 | Build Management menu | Oct. 15 |
| | 03.3 | Build Customer List Screen | Oct. 15 |
| | 03.4 | Build Order Screen | Oct. 15 |
| | 03.5 | Finalize Color Scheme | Oct. 22 |
| | 03.6 | Finalize Fonts | Oct. 22 |
| | 03.7 | Finalize Button Design | Oct. 22 |
| 1 | 04.0 | Implement Point System | Nov. 07 |
| | 05.0 | Finalize Menu Items/Customizations | Nov. 07 |
| | 05.1 | Price changes per customization | Oct. 30 |
| | 06.0 | Implement loading from JSON | Nov. 05 |
| | 07.0 | Implement Items/Customizations | Nov. 05 |
| 2 | 08.0 | Display Item/total cost on ordering menu screen | Nov. 13 |
| | 09.0 | Display Customer List | Nov. 13 |
| | 10.0 | Implement system to add customers | Nov. 13 |
| | 10.1 | Input verification | Nov. 13 |
| | 10.2 | Store customers in JSON file | Nov. 13 |
| | 11.0 | Add card payment | Nov. 13 |
| | 11.1 | Verify card validity | Nov. 13 |
| | 12.0 | Display point totals | Nov. 20 |
| | 12.1 | In customer list | Nov. 20 |
| | 12.2 | In receipt | Nov. 20 |
| | 13.0 | Implement Receipt Display | Nov. 20 |
| | 14.0 | Implement CSV Compatibility | Nov. 26 |
| | 14.1 | Add to CSV list every sale | Nov. 26 |
| | 14.2 | Generate CSV Report in management screen | Nov. 26 |
| | 15.0 | Final Polish | Nov. 28 |

## Deliverables

**Project Plan – Oct 9**
Details the schedule and basic planning for completing the application

**Requirements – Oct 14**
Finalizes plans for fulfilling requirements, use cases, and use case diagrams

**Prototype – Oct 30**
First iteration of the application, not fully functional, for feedback purposes

**UI Design – Oct 24**
Final UI design of the application

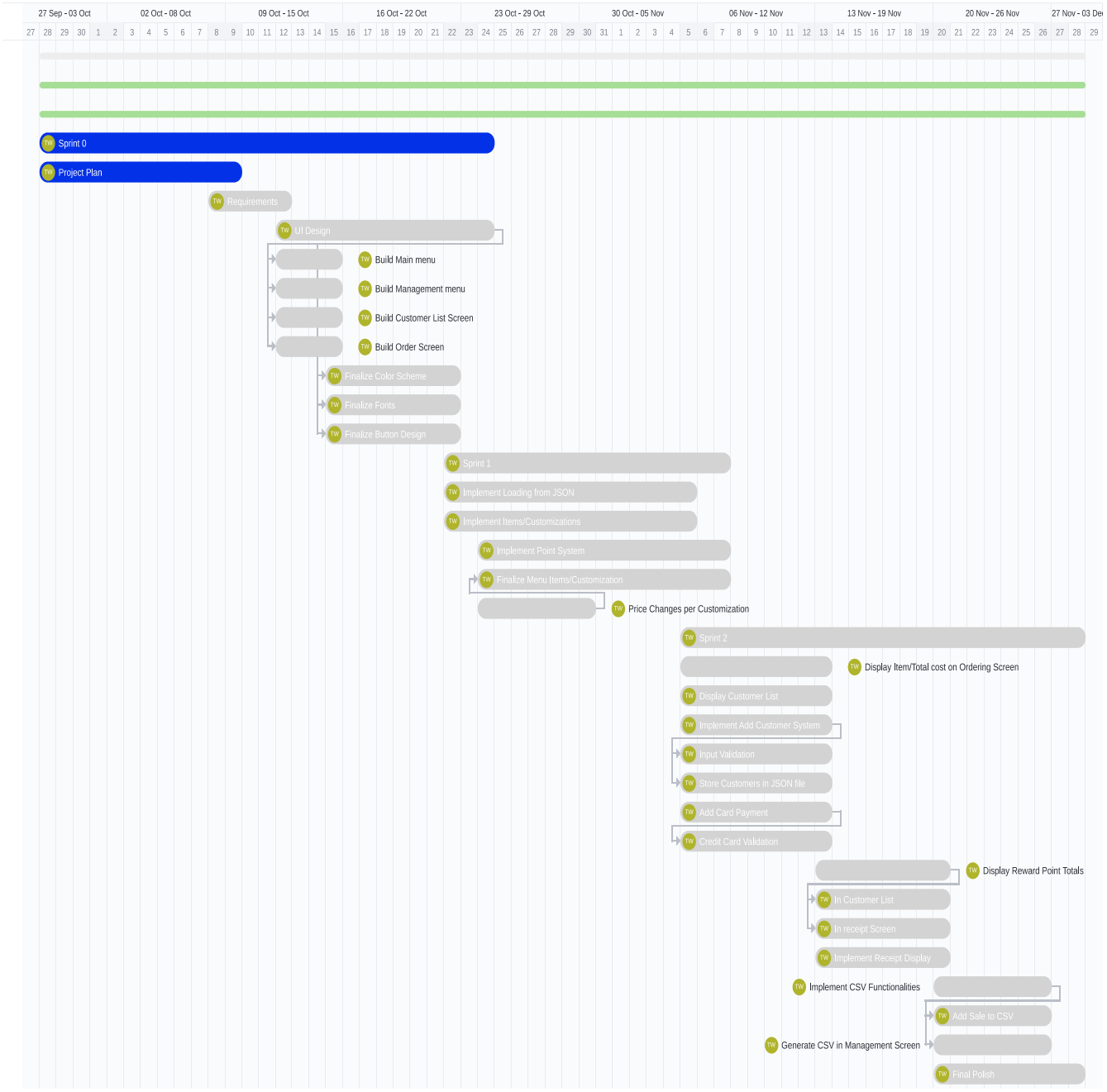
**Technical Design – Nov 7**
Details of the programming aspects of the application

**Final Product – Nov 28**
Fully operational point-of-sale system with all requirements fulfilled

**Presentation – Nov 28**
Present the completed product to the customer

**Peer Evaluation – Nov 30**
Evaluation of team members and their contributions to the team

# Gantt Chart


27 Sep - 03 Oct
02 Oct - 08 Oct
09 Oct - 15 Oct
16 Oct - 22 Oct
23 Oct - 29 Oct
30 Oct - 05 Nov
06 Nov - 12 Nov
13 Nov - 19 Nov
20 Nov - 26 Nov
27 Nov - 03 Dec
Sprint 0
Project Plan
Requirements
UI Design
Build Main menu
Build Management menu
Build Customer List Screen
Build Order Screen
Finalize Color Scheme
Finalize Fonts
Finalize Button Design
Sprint 1
Implement Loading from JSON
Implement Items/Customizations
Implement Point System
Finalize Menu Items/Customization
Price Changes per Customization
Sprint 2
Display Item/Total cost on Ordering Screen
Display Customer List
Implement Add Customer System
Input Validation
Store Customers in JSON file
Add Card Payment
Credit Card Validation
Display Reward Point Totals
In Customer List
In receipt Screen
Implement Receipt Display
Implement CSV Functionalities
Add Sale to CSV
Generate CSV in Management Screen
Final Polish

## Team Members

**Tagan Williamson**
Team Organizer | Programmer

**Richard Thomas**
Lead Programmer | UI Designer

**Jacob Germana-McCray**
Lead Writer | Programmer

**Sarah Crosby**
Graphic Designer | Programmer

# Tagan Williamson

## Software Engineer Intern

800 Peachtree Street NE
Atlanta, GA 30308

770.570.7191
taganw516@gmail.com
www.linkedin.com/in/taganwilliamson

## Summary

Hard-working college student with a passion for Software Engineering. Possesses strong interpersonal and problem-solving skills, communication, and attention to detail. Brings 3 years of technical work experience in Civil Engineering. Eager and quick learner with a constant desire to grow and add value to a team. Enjoys asking questions and receiving feedback.
Self-motivated individual that enthusiastically welcomes challenges.

## Skills

**Programming |** Python • Java • OOP

**Interpersonal |** Problem-solving • Communication • Time Management

## Experience

**Contour Engineering /** Engineering Technician
July 2018 - January 2022, Atlanta, Ga

- Performed detailed inspections on a wide range of construction tasks to ensure proper implementation of construction methods and structural elements
- Problem solved with contractors, engineers, and other construction personnel to find solutions to complex issues in the field
- Wrote daily detailed reports documenting the solutions and other findings of inspections for engineers, developers, and contractors

## Education

**Kennesaw State University /** BS, Software Engineering
August 2019 - December 2023, Kennesaw, Ga
Relevant courses:

- Currently taking Data Structures
- Programming & Problem Solving I & II (Java)

## Awards

- Dean's List
- Earned Python skill badge on LinkedIn by scoring in the top 30% of over two million people on the skill assessment. (2021)

# Richard Thomas

(404) 933-5799 | Rthomas1137@gmail.com | Github

## EDUCATION

**Kennesaw State University** — **Kennesaw/Marietta, GA**

*Bachelor of Science in Computer Science*

Expected graduation date: December 2024

- *3.78 GPA*
- *Honors Student*

**Relevant courses:**
*Programming and Problem Solving I and II, Computer Organization and Architecture, Introduction to Software Engineering, Technical Writing.*

## Projects

**Typing Test:**
*JavaScript* | **link:** Website

- Built a website that allows users to test their typing speed and accuracy using random quotes.
- Implemented a login system that allows users to login in through google and compare their typing speeds with one another
- stored user information using FireBase and generated quotes using Type.fit API.

**Weather App:**
*JavaScript* | **link:** Website

- Developed a web app that gives accurate temperature and weather conditions for over 200,000 cities worldwide.
- Utilized Weather API from OpenWeather and IP geolocation API from Abstract API.

**BlackJack:**
*Java* | *JavaFX*

- Built a Java application that allows the user to play a hand of Blackjack against a computer.
- Implemented an interactive GUI made with JavaFX.

**Sudoku Solver:**
*Java*

- Created a program that solves any solvable sudoku problem.
- Utilized backtracking.

## Activities And Interests

**In His Image Ministry**
*Volunteer responsibilities:*
Event planning and decorating, serving food, operating technical equipment, assisting with fundraising activities.

## SKILLS

| | |
|---|---|
| Programming | Java, C#, JavaScript, Python |
| Web & Database | HTML, CSS, MySQL, FireBase |
| Methodologies | Object Oriented Programming and Agile development |
| Tech | Git, Visual Studio, VsCode, IntelliJ, Eclipse, Figma, MS Office |

**Jacob Germana-McCray**
#236 Royal Palms Ave.
Kingsland, GA 31548

912-322-9380
jacobgermanamccray@gmail.com
https://github.com/JGM01

## Skills

**Languages:** Rust, C/C++, Java, C#, Python

**Operating Systems:** Linux (Debian/Arch), UNIX, MacOS X, Windows 7/10/11

## Education

- **Kennesaw State University** — Marietta, GA
  *Bachelor of Science, Computer Science (GPA: 3.1)* — *Aug. 2021 - May. 2024*
  - Relevant courses: Data Structures, Computer Organization & Architecture, Intro to Software Engineering, Into to Databases

## Experience

- **Research Assistant** — Marietta, GA
  *Quantum Key Distribution* — *Sep. 2022*
  - Implemented quantum entanglement algorithms using python with the Qiskit package.
  - Programmed quantum circuits for demonstration purposes and simulated their results using the Aer quantum computer simulator.
- **Camden County Board of Education** — Kingsland, GA
  *Information Technology Intern* — *Aug. 2020 - May. 2021*
  - Repaired laptops for hundreds of students.
  - Organized and created a system to help improve efficiency.
  - Logged and kept track of hundreds of issues.

## Projects

- **Shopping Bot**
  *Python, Selenium* — *https://github.com/JGM01/fumobot*
  - Automated purchasing on the website amiami.com and was capable of purchasing an item automatically in 12 seconds even when the website was under heavy stress.
  - Implements multiprocessing to run several parallel bots at the same time to increase chance of success.
- **Chip_8 Emulator**
  *Rust* — *https://github.com/JGM01/chip_8_emulator*
  - Built a CPU, RAM, and Display emulator to run the CHIP_8 instruction set.
  - Implemented Computer Organization & Architecture principles such as creating a bus to communicate between structures.
  - Due to Rust safety features, there are no memory leakage risks.
- **Wii Calculator**
  *C, CMake* — *https://github.com/JGM01/wiicalculator*
  - Modified the Nintendo Wii to accept input from the Wiimote and used it to do simple addition
  - Uses CMake to build the executable to run on the Wii

**Sarah Crosby**
218 Lakeview Trl.
Hiram, GA 30141

678-492-0913
Scrosb10@students.kennesaw.edu

## Skills

- Languages: Java, C#, SQL
- Experience with UI Design
- Experience with technical communication

## Education

- Kennesaw State University
  - Computer Game Design and Development Undergraduate
  - Data Structures
  - Principles of Programming
  - Game Design and Development

## Experience

- Database and Inventory Management
- Experience working with Point-of-Sale systems

## Projects

- Third Person Shooter – Unity/C#
  - Designed and programmed a third person shooter and designed UI and graphical elements
- Chatroom – Java/JavaFX
  - Designed and programmed a chat room using servers

## Technical Description

The Coffee Company point of sale system is designed to:

- Track customers via phone number
- Take drink orders and allow each drink to be customized
- Calculate order cost (including customizations and tax)
- Accept card payment or point redemption
- Track customer points
- Track sales and generate CSV report

The system will utilize the following technologies:

- C# (Programming Language)
- JSON (Storage of items/customizations, configuration, customer, and sales data)
- C# WinForms (UI Design)
- Newtonsoft (Convert between .NET and JSON)
- CSVHelper (Create CSV files)
- CreditCardValidator
- GitHub

The application will be coded in C# and will be able to run on most operating systems and modern devices. The device the application is installed on will need to have enough storage space for expected sales, menu, and customer data.

Some of the major project risks are:

- Application features taking longer to complete than expected
- Errors in price calculation
- Lack of communication between developers and/or customers
- Lack of clear documentation

Risk will be mitigated by:

- Frequent communication between team members and the client
- Time management and assignment delegation/prioritization
- Consistent documentation
- Ongoing testing throughout development

The UI of the application will be organized and intuitive. The application will open to the Main Menu screen that will contain:

- Order Now button that jumps directly to the Order screen as an anonymous customer.
- Customer List button that navigates to the Customer List screen.
- Management Tools button that directs the user to the Management screen.

Customer List screen:

- Display an exhaustive list of rewards customers
- The list will contain the full name, phone number, and reward point totals for all customers.
- The list will be sorted by last name, first name, and lastly, phone number except for the anonymous customer which will be at the top of the list.
- Each customer will have an order button that will begin an order under their name.
- Add Customer button that will move to the Add Customer screen

Add Customer screen:

- Required entry fields for first name, last name, and phone number
- Submit button that adds the customer to the list and continues to the order screen under their name

Order Screen:

- Will have to panes one of the left and one on the right.
- The left pane will allow for drink adding and modification.
- The right pane will show drinks and modifications currently in the order, and pricing information.
- Proceed to Payment button that will only be clickable once at least one drink is ordered.

Payment Screen:

- Display reward points for registered customers
- Pay with rewards selection that can only be used if the user is registered and has enough points to fully pay for the order
- Pay with card selection that is the default that will require a card number, cardholder name, expiration date, and security code.
- Submit button that will notify of completed/failed payment. Upon successful payment, will continue to Receipt screen.

Receipt Screen:

- Display list of ordered drinks and modifications.
- Pricing information of each drink
- Subtotal, tax, and final total for order
- Payment method with the last four digits of the card displayed when a credit card is used, or the number of redeemed reward points otherwise.
- Continue button that returns to the Main screen.

Management Screen:

- Generate Sales Report button that generates a CSV file in excel with relevant sales data formatted to client specifications.

**Each screen will also have a Return to Main Menu button or a Cancel button that will immediately return to the Main Menu.**

## Data Management

Data will be stored in one of three JSON files:

Customer List Data
- Customer ID
- Customer phone
- Customer First and Last Name
- Reward points

Sales Data
- Customer ID
- Transaction Time and Date
- Subtotal
- Total
- Tax
- Payment method
- Items in order (Name, customizations, price)

Configuration Data
- Menu items
- Customizations
- Prices
- Tax rate
- Reward points per dollar

The configuration data will be loaded upon the application starting. The JSON file cannot be edited through the application, users would have to edit the file directly to change the settings.

Registered User Data
- Last Name, First Name. (at least ten characters total)
- Sorted by last name in customer list
- Reward points (starts at zero when first initialized).
- Phone number (must be unique and valid).
- A unique id will be generated.

Anonymous User Data
- Always at the top of customer list
- Phone number set to 000-000-0000
- Anonymous (first/last name)
- Reward points are not tracked

Purchase data for logged in User

- After a purchase, rewards are given to the user (1 point per $10 spent).
- Payment method is recorded
- Sales data is generated
- Data entry events will update customer and sales data with the latest information.

Purchase data for Anonymous User

- Payment method is recorded
- Sales data is generated
- Data entry events will update customer and sales data with the latest information.

Drink menu

- Menu and drink customization data is loaded from a JSON file

Generating sales data

- CSV file of all sales is generated

## Testing

The application and its features will be tested during each stage of development for functionality.

Highest Priority Tasks:
- Buttons navigate to the proper screen
- Menu items and prices are displayed properly
- Order pane updates so the customer can view full order
- Total cost is properly calculated
- Credit card verification works properly
- Customer data is displayed and saved properly
- CSV report is generated properly with accurate information

Other Tasks:
- UI Design
- Ease of Use
- Code Cleanup

Testing Documentation:
- Any errors encountered will be documented along with how they interacted with the program before the error occurred.
- These errors will also be documented along with their severity to prioritize maintenance.
- Once error reports are received developers will use reports to track and fix issues in order of severity.
- Once corrections are made, the testing process will be performed again to ensure that the error is fixed.

In addition to testing for errors, the testing process will note the program's overall efficiency. If the program has noticeable delays when loading data or changing screens, the developers will make changes to improve efficiency.

Testing Methods:
- Most of the testing will be done with White Box testing to help the development team find issues in the code.
- Black box testing will be done at the end of each sprint to ensure all requirements are met.
- In addition to testing, communication with the customer will be frequent in case there are any changes to the requirements that need to be made.