

Colin Quinn, Kuwar Nagpal, Kaylee Moore, Elena Bucciarelli, Michael Weger
Software Engineering
Dr. Farmer

Project Documentation

Use Cases and User Stories:

Job status
in list [User Stories](#)

LABELS
Priority 2 +

Description [Edit](#)
As a mechanic, I want to be able to mark a job as begun, in-progress, or blocked

Activity [Show details](#)
Write a comment...

SUGGESTED [Join](#)
ADD TO CARD [Members](#)
[Labels](#)
 [Checklist](#)
[Due date](#)
[Attachment](#)
[Cover](#)



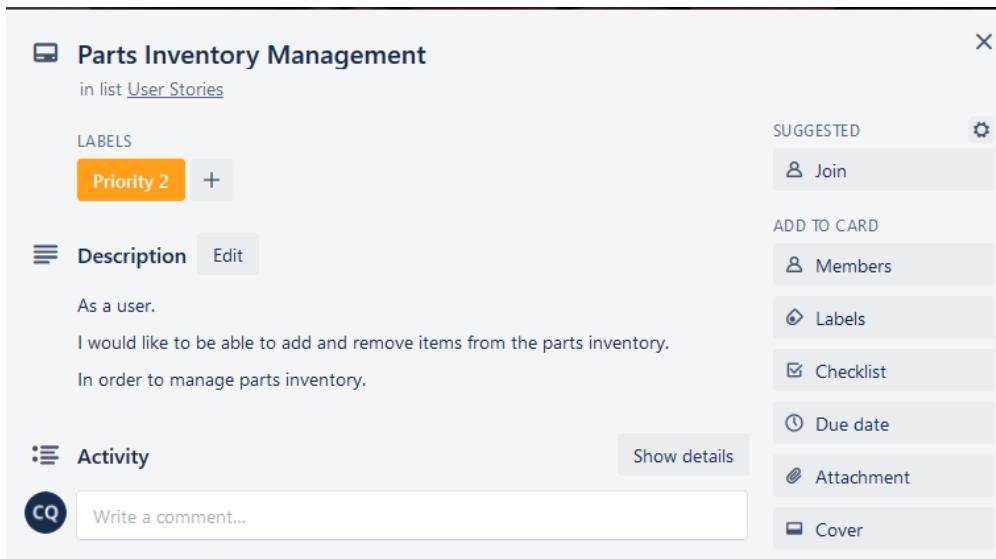
Parts Inventory Management
in list [User Stories](#)

LABELS
Priority 2 +

Description [Edit](#)
As a user.
I would like to be able to add and remove items from the parts inventory.
In order to manage parts inventory.

Activity [Show details](#)
Write a comment...

SUGGESTED [Join](#)
ADD TO CARD [Members](#)
 [Labels](#)
 [Checklist](#)
[Due date](#)
[Attachment](#)
[Cover](#)



Submit Maintenance Job

in list [User Stories](#)

LABELS

Priority 2 +

Description Edit

As a mechanic
I want to be able to submit a maintenance job
To track the job progress and have a listing of active jobs to improve productivity.

Activity Show details

cq Write a comment...

SUGGESTED

Join

ADD TO CARD

Members

Labels

Checklist

Due date

Attachment

Cover

Order processing

in list [User Stories](#)

LABELS

Priority 3 +

Description Edit

As a cashier
I want to be able to process customer orders.
In order to complete and track a sale.

Activity Show details

cq Write a comment...

SUGGESTED

Join

ADD TO CARD

Members

Labels

Checklist

Due date

Attachment

Cover

Index Search

in list [User Stories](#)

LABELS

Priority 3 +

Description Edit

As a user
I want to be able to quickly reference services and items for sale
In order to quickly find what I want.

Activity Show details

cq Write a comment...

SUGGESTED

Join

ADD TO CARD

Members

Labels

Checklist

Due date

Attachment

Cover

Transaction Types

in list [User Stories](#)

LABELS

Priority 5 +

Description Edit

As a cashier
I want to be able to handle multiple types of transactions (cash, credit, gift card)
In order to allow customers to complete quick checkouts.

Activity Show details

cq Write a comment...

SUGGESTED

Join

ADD TO CARD

Members

Labels

Checklist

Due date

Attachment

Cover

Track Transactions

in list [User Stories](#)

LABELS

Priority 5 +

Description Edit

As a user
I want to be able to track transactions
In order to reference a purchase history for any particular user

Activity Show details

cq Write a comment...

SUGGESTED

Join

ADD TO CARD

Members

Labels

Checklist

Due date

Attachment

Cover

Order Parts

in list [User Stories](#)

LABELS

Priority 5 +

Description Edit

As a mechanic
I want to be able to order parts from a supplier
In order to complete service jobs

Activity Show details

cq Write a comment...

SUGGESTED

Join

ADD TO CARD

Members

Labels

Checklist

Due date

Attachment

Cover

Spending tracking

in list [User Stories](#)

LABELS

Priority 4 +

Description Edit

As the accountant, I want to be able to see how much money has been spent on parts so we can keep accurate books for taxes

Activity Show details

cq Write a comment...

SUGGESTED

Join

ADD TO CARD

Members

Labels

Checklist

Due date

Attachment

Cover

Acceptance Criteria:

Log Employees in and Out	
1. Employee clicks on login option	Login fields appear
2. Employee enters login credentials and clicks login	System logs in (shows the main page)
3. Employee clicks on logout option	System returns to login page
Submit Maintenance Job	

1. Employee opens maintenance tab/section	System displays all currently scheduled jobs
2. Employee clicks 'add job' option	Form appears with text boxes
3. Employee enters job information and clicks submit	Form disappears and job appears in the list

Search for Parts in Inventory	
1. Employee opens parts index	System displays current parts in inventory
2. Employee enters into search field	System displays parts that match the search terms

Add item to order	
1. Cashier clicks add to order on an item	System gives confirmation window
2. Cashier clicks accept	System saves item to the current order and removes item from inventory list

Accept payments	
1. Cashier clicks on 'checkout' option on current order	System brings up all items in order and price total
2. Cashier selects 'pay' option	Transaction options appear (cash,card,etc)
3. Cashier chooses payment option	Relevant fields appear
4. Cashier fills in fields and clicks confirm option	System saves transaction, payment info sent to processing service

Add item to inventory list	
1. Employee selects 'inventory' section	System displays all current items
2. Employee selects 'add inventory' option	System displays fields for item information and price
3. Employee fills out fields and selects confirm	System adds items to list and displays list again

Update Work Order	
1. Employee selects Work Order section	System displays all current work orders
2. Employee selects work order in question	System displays information about the work order
3. Employee selects one of the three options for 'Awaiting parts', 'begun', or 'complete'	System marks work order as selected option

Remove Item from Inventory	
1. Employee selects inventory section	System displays list of inventory
2. Employee clicks on item	System displays item information
3. Employee selects 'delete' option	Item is removed from displayed list

Order Parts	
1. Employee opens parts ordering section	System displays all currently ordered parts
2. Employee selects 'add order' option	System displays fields and options for parts
3. Employee selects parts to order from list and clicks submit	System displays confirmation, saves order details, and sends order to secondary system

Spending Tracking	
1. Employee selects parts order section	System displays all ordered parts
2. Employee selects option for time frame	System displays all ordered parts within that time frame
3. Employee selects 'view total' option	System displays total spent cost

ilities Table:

Imp orta nce Fac tor (1-3)	Met by this arc hite Met	Why is it important?	How does the architecture meet it?

)	Y/N		
Performance	1	Y	Needs to be able to handle the multiple processes happening from employees in different sections of the store	Database can be queried from multiple sources
Security	3	N	System will be handling both inventory and monetary transactions, both of which need to be kept secure	
Safety	2	N	As the software controls car repair, it's important that repairs are done properly and facilitated by this software	
Availability	3	Y	If the application isn't available the user is getting no use of it. It's important the app is always available.	Database is consistent/doesn't 'turn off'
Maintainability	2	Y	Software will contain inventory and other information that needs to be kept consistent	Database is consistent/maintains data
Reliability	3	Y	The software is responsible for a critical component of the shop, it's important that it is not down often.	Database is stored in one location, always available
Scalability	1	Y	The software will only run in one location for now, so the initial load is all we need to plan for.	The database can be queried by multiple users. One central database is all that is necessary to support more locations.
Portability	1	Y	It is a relatively simple webapp, so portability is a given with the basic design	The database can be queried across many platforms.
Usability	3	Y	Application should be easy to use. Minimal training time should be required for teaching. Time is money.	Websites are generally self explanatory, so any training is to make sure things are filled in properly
Supportability	2	Y	Adding new features is unlikely, so it should be simple if needed	New features would be additions to the webapp and database, both of which are quite simple to update.
Reconfigurability	1	N	Not much to configure. Modifying the data only requires interacting with the database.	
Architecture Chosen:			Data-centric/Database	

Stereotypes:

CRC Cards/Class List Stereotypes:

Class: Cashier (Actor - People)

Responsibilities: (Knows) Transaction details (Does) Enter transactions, view inventory information

Collaborators: Inventory Manager, Transaction System, Log-in

Class: Inventory Manager (Business - Event)

Responsibilities: (knows) inventory status/information (Does) updates / views inventory

Collaborators: Transaction System, Merchandise, Part, Motorcycle

Class: Transaction System (Business - Event)

Responsibilities: (Knows) Payment types/information (Does) Saves transactions, sends changes to inventory manager, sends payment amounts to accounting system

Collaborators: Inventory Manager, Cashier, Accounting System, Merchandise, Part, Motorcycle

Class: Mechanic (Business - Thing)

Responsibilities: (Knows) Work Order details (Does) Completes Work Orders, Punch-In, Punch-Out

Collaborators: Manager, Work Order System, Log-in

Class: Merchandise (Business - Thing)

Responsibilities: (Knows) Item ID, name, description, price. (Does) information holding class

Collaborators:

Class: Part (Business - Thing)

Responsibilities: (Knows) Part ID, name, description. (Does) information holding class

Collaborators:

Class: Motorcycle (Business - Thing)

Responsibilities: (Knows) Bike ID, name, make, model, year, price, description. (Does) information holding class

Collaborators:

Class: Accounting System (Report - Electronic)

Responsibilities: (Knows) transaction details, view inventory (does) logs transaction data

Collaborators: Cashier System

Class: Work Order System (Report - Electronic)

Responsibilities: (Knows) Current jobs, current part orders (Does) Mark jobs as in progress/complete/not started, Sort jobs, Assign mechanics to jobs, Order needed parts

Collaborators: Cashier, Mechanic, Inventory Manager

Class: Manager (Actor - Person)

Responsibilities: (Knows) Schedule, employee information (Does) Manages schedules, administrates the various systems, keeps track of spending

Collaborators: Cashier, Mechanic, Accounting System, Log-in

Class: Log-in (Business - Event)

Responsibilities: (Knows) all employee info including manager, cashiers, and mechanics (Does) opens system depending on user scope

Collaborators: Manager, Cashier, Mechanic, Employee Database, Password Database

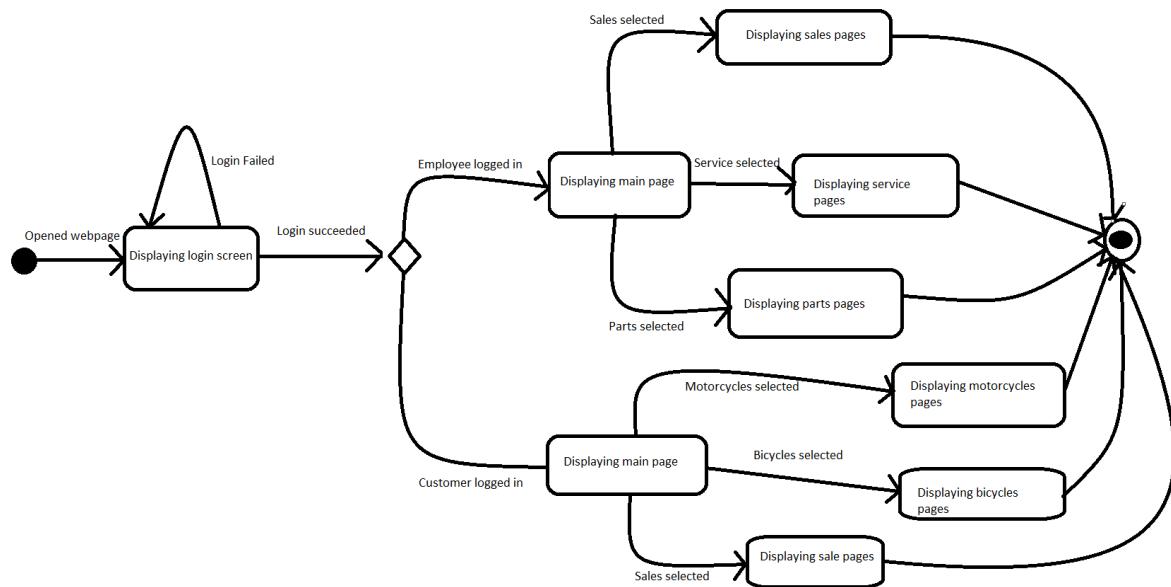
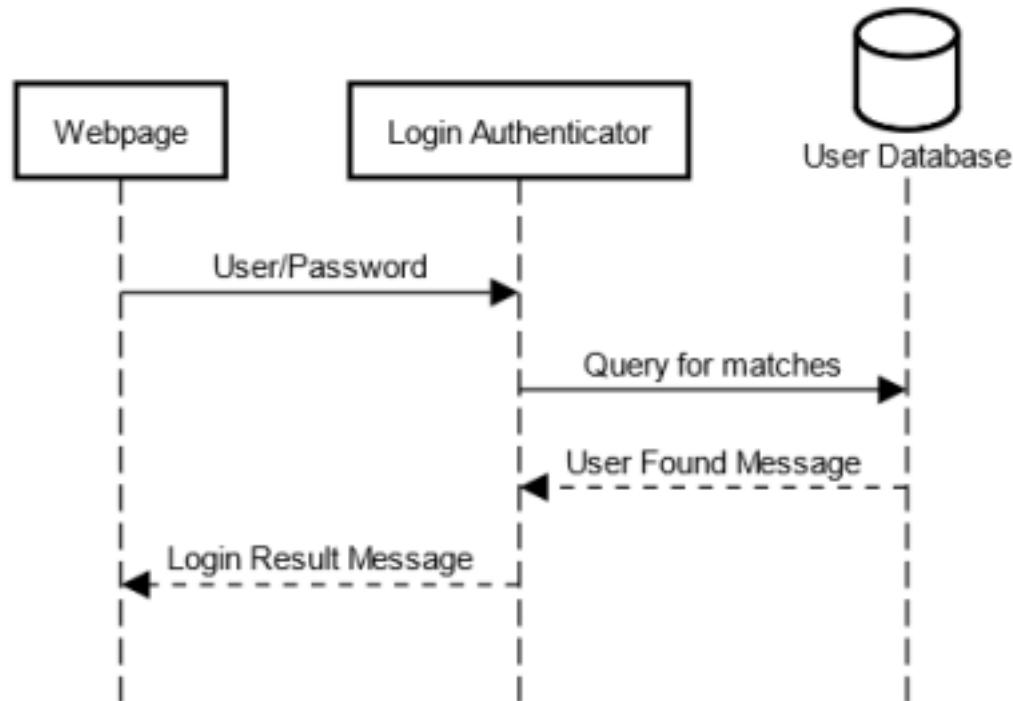
Merchandise	
<p>Responsibilities:</p> <ul style="list-style-type: none"> ● Knows <ul style="list-style-type: none"> ○ Name ○ Description ○ Price ○ Quantity ● Does <ul style="list-style-type: none"> ○ None, information holding class 	<p>Collaborators:</p> <ul style="list-style-type: none"> ○ None

Part	
<p>Responsibilities:</p> <ul style="list-style-type: none"> ● Knows <ul style="list-style-type: none"> ○ Name ○ Description ○ Quantity ○ Location ● Does <ul style="list-style-type: none"> ○ None, this is an information holding class. 	<p>Collaborators:</p> <ul style="list-style-type: none"> ○ None

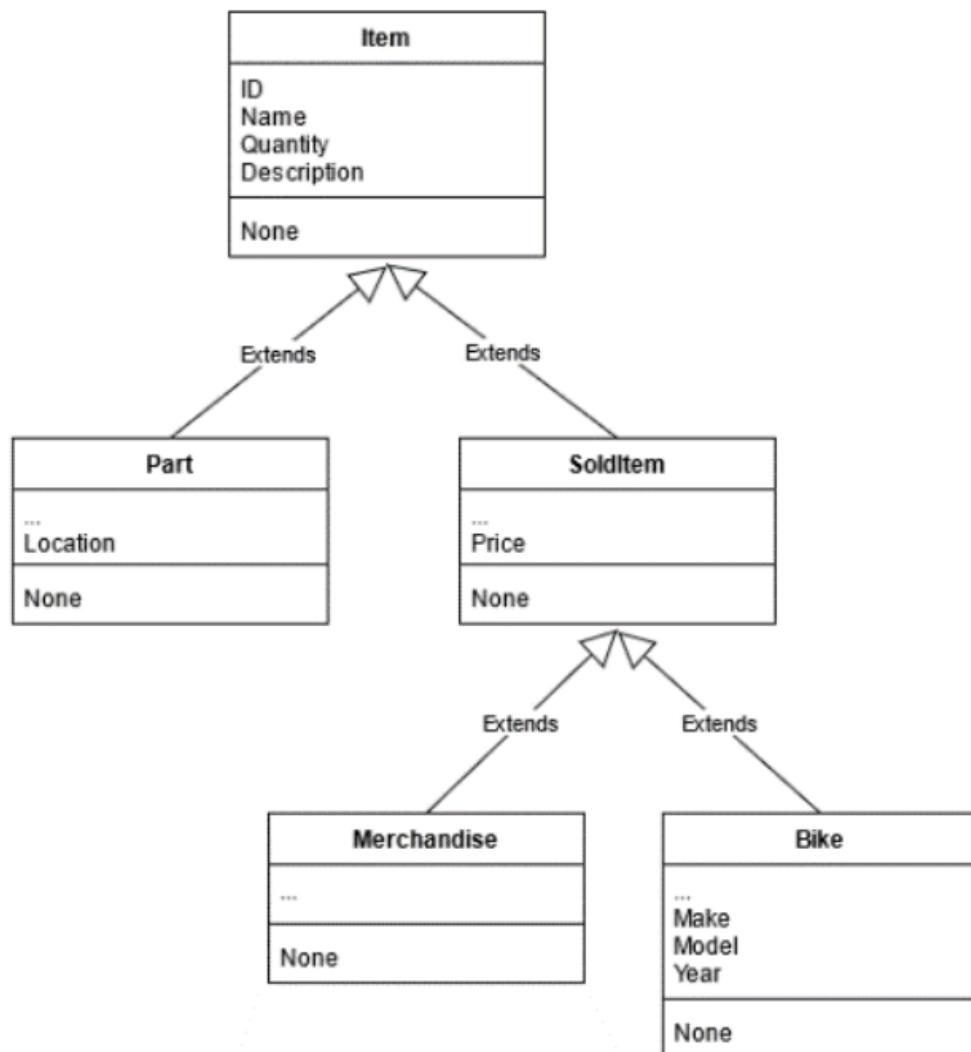
Bike	
<p>Responsibilities:</p> <ul style="list-style-type: none"> ● Knows <ul style="list-style-type: none"> ○ Name ○ Description ○ Price ○ Quantity ○ Make ○ Model ○ Year ● Does <ul style="list-style-type: none"> ○ None, this is an information holding class. 	<p>Collaborators:</p> <ul style="list-style-type: none"> ○ None

Class Diagram / Sequence Chart / State Diagram

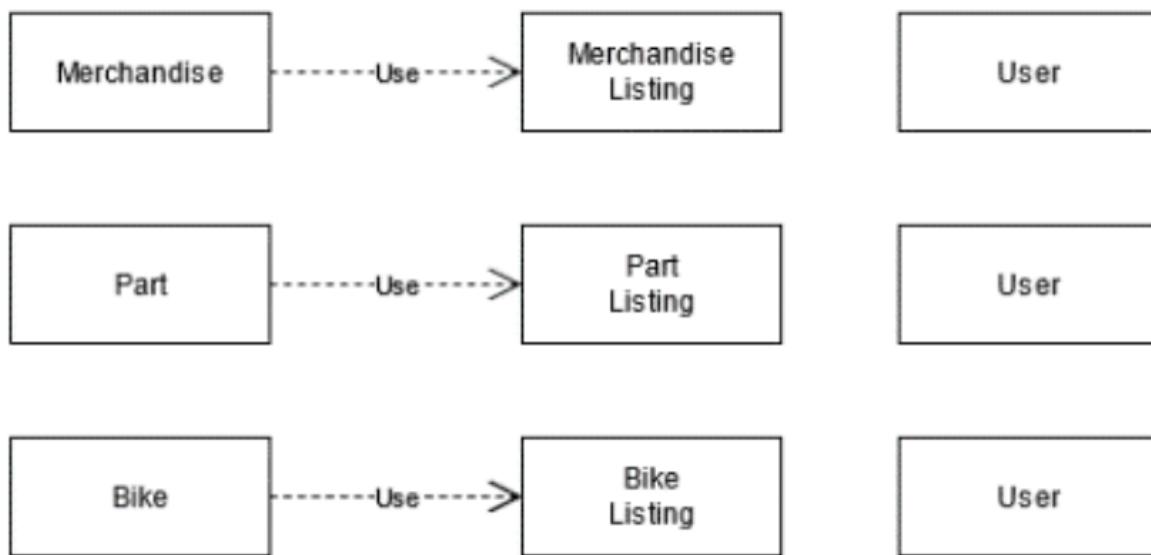
Log-in



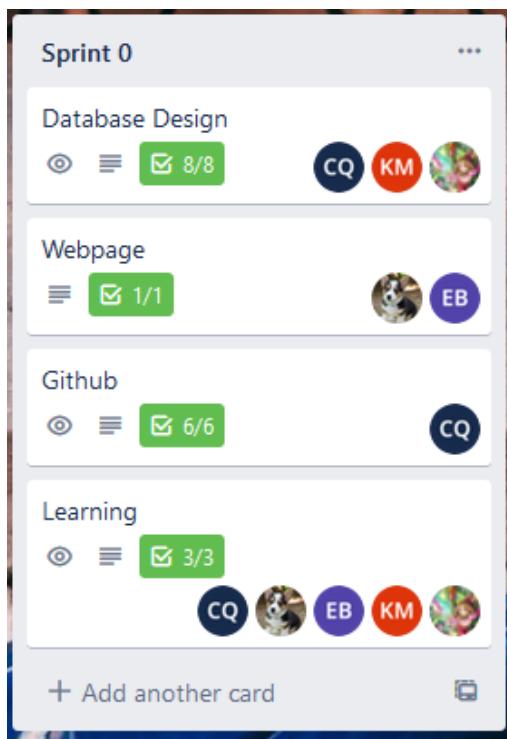
Item Inheritance



Item Listing Dependencies



SCRUM User Stories In Sprints



Database Design

in list Sprint 0

MEMBERS



X

Description

As a developer
I want a database set up
So I can begin working on everything else!
The database should have a table for each item in the checklist.

ADD TO CARD

Members

Labels

Checklist

Due date

Attachment

Cover

POWER-UPS

BUTLER NEW

ⓘ

Add button

ACTIONS

Move

Copy

Make template

Watch ✓

Archive

Share

Tables

Hide completed items

Delete

100%

- Users
- Parts Inventory
- Service Jobs
- Transactions
- Merchandise
- Bikes

Add an item

Format

Hide completed items

Delete

100%

- Excel Mockup
- SQLite

Add an item

Activity

Show details



Write a comment...

Webpage

in list Sprint 0

MEMBERS



SUGGESTED

Join

Description Edit

As a user
I want an interactive webpage
So that I don't have to use a terminal window
Make a pretty landing page for the application from which we can login

TODO

100% 

Pretty Page

Add an item

Hide completed items Delete

Activity

Show details

Write a comment...

POWER-UPS

+ Add Power-Ups

BUTLER NEW

+ Add button

Gitub

in list Sprint 0 [Edit](#)

MEMBERS



+

Description [Edit](#)

As a developer
I want a github repo set up
So that we can have a neatly organized repo for source control
Please make the .gitignore nice :)

TODO

100%

- Setup
- Readme
- Directory structure
- .gitignore
- Branch structure
- Define rules for making commits, merging etc

[Hide completed items](#)

[Delete](#)

[Add an item](#)

Activity



[Write a comment...](#)

[Show details](#)

ADD TO CARD

Members

Labels

Checklist

Due date

Attachment

Cover

POWER-UPS

Add Power-Ups

BUTLER NEW

(i)

Add button

ACTIONS

Move

Copy

Make template

Watch

(✓)

Archive

Share

Learning

in list Sprint 0

MEMBERS



Description

Edit

As a developer
I want to learn how to use html/css and node.js
So that I can create a cool buzzword filled webapp
Spend some time to learn this stuff

TODO

100%

- SQLite
- Nodejs
- HTML/CSS

Hide completed items

Delete

Add an item

Activity

Show details



Write a comment...

ADD TO CARD

Members

Labels

Checklist

Due date

Attachment

Cover

POWER-UPS

Add Power-Ups

BUTLER NEW



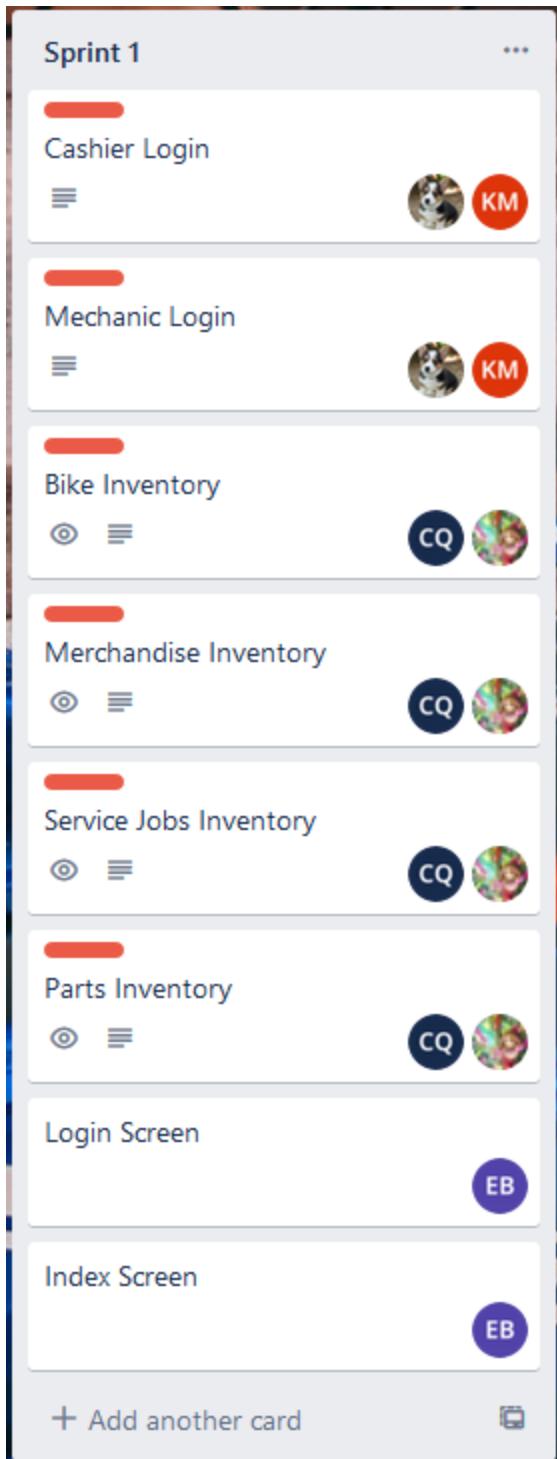
Add button

ACTIONS

Move

Copy

Make template



Many of these tasks were similar, so only a few screenshots will be shared as they have the same goals of allowing workers to login, manage inventory, and general design.

Cashier Login

in list Sprint 1

MEMBERS LABELS

 KM + Priority 1 +

SUGGESTED 

 Join

ADD TO CARD

 Members

 Labels

 Checklist

 Due date

 Attachment

 Cover

Description 

As a manager
I want cashiers to be able to log in and out at the start and end of their shift.
In order to track hours.

Activity 

 CQ Write a comment...

Bike Inventory

in list Sprint 1

MEMBERS LABELS

cq + Priority 1 +

Description Edit

As a manager,
I want to see what is in the bike inventory
So that I know what bikes are available for customers and when new ones will need to be ordered.

Activity Show details

cq Write a comment...

Add To Card

- Members
- Labels
- Checklist
- Due date
- Attachment
- Cover

POWER-UPS

+ Add Power-Ups

Login Screen

in list Sprint 1

MEMBERS

EB +

Description Edit

Get designs sketched out

Activity Show details

cq Write a comment...

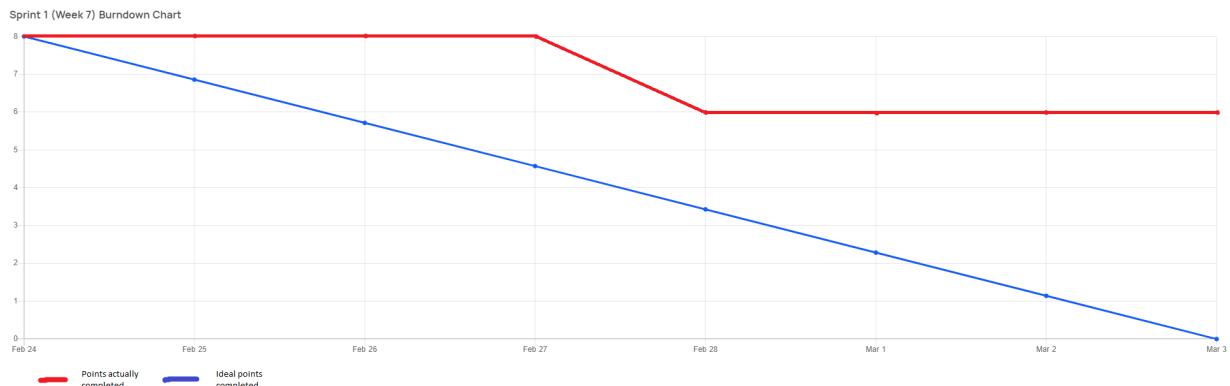
Suggested

- Join

Add To Card

- Members
- Labels
- Checklist
- Due date
- Attachment

SCRUM Burndown Chart:



Uncompleted tasks were pushed to the following sprint along with new documentation tasks.

SCRUM Meeting Sheets:

Week of: 4th week

Project Title: Cool (We're not at the level of cool yet Michael) Motorcycle Thing

Team Members: Colin, Elena, Kaylee, Kuwar, Michael

I. Tasks from Last Week That Were *Defined* to be Worked Upon:

1. User Stories
2. Priority for task assignment
3. Basic setup

II. *Detailed* Description of Work Accomplished this Week Towards all of these Tasks: (Be sure to define what *each* person specifically did each week. Repeated weeks of no work will result in failure of student.

1. Created user stories and they're really cool (everyone participated in this)
2. Assigned priorities to each user story, they're really rad, and well organized, we (and by we, I mean everyone participated) promise.
3. Created Github repo and branches, and a really gnarly Trello board, (Kuwar did Github and Michael organized the Trello).
4. N/A, everyone participated.

III. Unanticipated Tasks or Problems that Arose This Week and Required Addressing:

1. Forgetting about having to fill this out, we all participated.
2. Scheduling last minute Sprint Planning, we did it after hours because we are all really dedicated (we were all there, we all participated).

IV. Tasks to be Worked Upon During the *Upcoming* Week (this section becomes section 1 of the next week):

1. Database design
2. Webpage
3. Github

4. Learning the languages that we need to use

V. General Reflection on the Week:

1. What went well? We ALL actually showed up, and all participated. It was nice.
2. What did not go well? We all kind of forgot about this paper, and had to put together a last minute meeting to get it done.
3. What should we do more of? More meetings/discussion. Participating, of the MAXIMUM proportions.
4. What can we do less of? Forgetting, taking unnecessary (yet hilarious) risks with this document, while still somehow being serious.

Week of: 2/7/2021

Project Title: Motorcycle Dealership (but not that cool)

Team Members: Kaylee, Kuwar, Michael, Elena, Colin

I. Tasks from Last Week That Were *Defined* to be Worked Upon:

1. Learn how to use SQLite and js for this project. HTML/CSS for webpage design.
2. Create tables for database
3. Acceptance criteria for user stories
4. Further GitHub setup

II. *Detailed* Description of Work Accomplished this Week Towards all of these Tasks: (Be sure to define what *each* person specifically did each week. Repeated weeks of no work will result in failure of student.

1. Everyone took time this week to look into using each of the tools and languages they will use for their portion of the project.

Everyone

2. Discussion was had on tables for the database. Tables were mocked up in Excel and created in SQLite. Tables were approved by team members.

Michael created tables

Kaylee and Kuwar approved table

3. Team got together to discuss acceptance criteria. Acceptance criteria was drafted for user stories.

Michael, Kaylee, and Kuwar

4. Initial commits made to GitHub. Branch structure and rules.

Kuwar

III. Unanticipated Tasks or Problems that Arose This Week and Required Addressing:

1. Trello was down during some meetings. Unable to create cards until afterward.

IV. Tasks to be Worked Upon During the *Upcoming* Week (this section becomes section 1 of the next week):

1. Cashier/Mechanic login
2. Merch/Parts/Services/Bikes index
3. Website design

V. General Reflection on the Week:

1. What went well?
Worked efficiently on acceptance criteria. Meetings went smoothly and were planned.
2. What did not go well?
Not everyone attended each meeting. All members attended meetings.
3. What should we do more of?
Have better ways to notify members when a meeting will take place. Not all members use the same forms of communication constantly.
4. What can we do less of?
Lack of communication between sub-teams.

Week of: 6th

Project Title: Super Cool Motorcycle Store

Team Members: Elena, Kuwar, Michael, Colin, Kaylee

I. Tasks from Last Week That Were *Defined* to be Worked Upon:

1. Cashier / Mechanic / Index Login Screens
2. Bike / Merchandise / Service Jobs / Parts Inventory screens
3. Front end Design draft

II. *Detailed* Description of Work Accomplished this Week Towards all of these Tasks: (Be sure to define what *each* person specifically did each week. Repeated weeks of no work will result in failure of student.

1. Elena drafted a design template for the front end.

2. The group worked together on acceptance criteria and an ilities chart.

III. Unanticipated Tasks or Problems that Arose This Week and Required Addressing:

1. Overly busy with midterms, did not have much time to really plan other things out.
2. Tired.

IV. Tasks to be Worked Upon During the *Upcoming* Week (this section becomes section 1 of the next week):

1. Login screen creation
2. Inventory screens
- 3.
- 4.

V. General Reflection on the Week:

1. What went well? At least Elena did a task. MAXIMUM PARTICIPATION.
2. What did not go well? Not enough time for MAXIMUM participation throughout the group.
3. What should we do more of? Actually sitting down to work out some tasks as a group.
4. What can we do less of? Ideally, midterms.

Week of: 7th week

Project Title: Super Cool Motorcycle Project

Team Members: Elena, Michael, Colin, Kaylee, Kuwar

I. Tasks from Last Week That Were *Defined* to be Worked Upon:

1. Login screens
2. Inventory screens
3. CRC cards

II. *Detailed* Description of Work Accomplished this Week Towards all of these Tasks: (Be sure to define what *each* person specifically did each week. Repeated weeks of no work will result in failure of student.

1. Kuwar completed the login screens that will load into the splash page, which will be the inventory screens.

2. Some CRC cards were made by everyone during the meeting.

III. Unanticipated Tasks or Problems that Arose This Week and Required Addressing:

1. Overall not much was unanticipated, just a normal Kettering week.
- 2.

IV. Tasks to be Worked Upon During the *Upcoming* Week (this section becomes section 1 of the next week):

1. Inventory screen once splash page is loaded
2. Inventory management stuff (search, new tabs, service jobs active)
3. Split views for sales and mechanic shops
4. Add state diagrams for work flow

V. General Reflection on the Week:

1. What went well? Login screen is done, meeting went as planned.
2. What did not go well? Forgetting <http://>
3. What should we do more of? Documentation, it always helps to know what others are up to during the week.
4. What can we do less of? Procrastination, it somehow always sneaks back into life.

Week of: 8th week

Project Title: Super Cool Motorcycle Project

Team Members: Elena, Michael, Colin, Kaylee, Kuwar

I. Tasks from Last Week That Were *Defined* to be Worked Upon:

1. Screen Design
2. Search Function
3. State and Sequence Diagrams

II. Detailed Description of Work Accomplished this Week Towards all of these Tasks: (Be sure to define what *each* person specifically did each week. Repeated weeks of no work will result in failure of student.

1. Elena got the layouts for most of the screens done in powerpoint, just need to finish with the code for each of them.

2. Everyone worked together on state and sequence diagrams.

III. Unanticipated Tasks or Problems that Arose This Week and Required Addressing:

1. The nice weather, it can be a large distraction when you haven't left the house in months.
2. Still somehow busy, it never ends.

IV. Tasks to be Worked Upon During the *Upcoming* Week (this section becomes section 1 of the next week):

1. Put together a mock burndown chart based off of these documents
2. Build out webpages, functionality can be next sprint
- 3.
- 4.

V. General Reflection on the Week:

1. What went well? Elena got the layouts done, they look great.
2. What did not go well? Search function did not get done
3. What should we do more of? Communication and preparation
4. What can we do less of? Still procrastinating things makes it hard to keep up.

Week of: 9th week

Project Title: Super Cool Motorcycle Project

Team Members: Elena, Michael, Colin, Kaylee, Kuwar

I. Tasks from Last Week That Were *Defined* to be Worked Upon:

1. Mock out a burndown chart
2. Building out basis of webpages

II. Detailed Description of Work Accomplished this Week Towards all of these Tasks: (Be sure to define what *each* person specifically did each week. Repeated weeks of no work will result in failure of student.)

1. A more formalized documentation style is in the google drive being used to keep documents. (might be a good idea to include in the git repo).

2. Everyone's machine is up and running with the git code base so we can all start to focus more on producing code.

III. Unanticipated Tasks or Problems that Arose This Week and Required Addressing:

1. Difficulty getting everyone's machine to compile the current code base.
2. Lull in communication this week.

IV. Tasks to be Worked Upon During the *Upcoming* Week (this section becomes section 1 of the next week):

1. Push burndown chart to this week
2. Building more webpages with the features included.
3. Database contacting with SQLite to get data from the database.
- 4.

V. General Reflection on the Week:

1. What went well? We all got a bit of progress towards working on the codebase together.
2. What did not go well? Getting code running on all machines
3. What should we do more of? Coding, documentation seems to be coming together well and we can now focus more on creating a solid product. Also, communication is always a good thing to have.
4. What can we do less of? Procrastination.

Week of: 10th week

Project Title: Super Cool Motorcycle Project

Team Members: Elena, Michael, Colin, Kaylee, Kuwar

I. Tasks from Last Week That Were *Defined* to be Worked Upon:

1. Burndown chart for at least 1 sprint
2. Building more webpages
3. Database contact with SQLite to get data from database

II. Detailed Description of Work Accomplished this Week Towards all of these Tasks: (Be sure to define what *each* person specifically did each week. Repeated weeks of no work will result in failure of student.

1. Michael and Kuwar got SQLite database contacting working, as well as a couple more screens.

2. Colin is confirming information of burndown chart.

3. Everyone worked together in planning a good way to submit our information.

III. Unanticipated Tasks or Problems that Arose This Week and Required Addressing:

1. For some reason profs are assigning/adding new things to the finals requirements which takes up more time randomly.
2. Confirming pull requests are hard when people are on different schedules.

IV. Tasks to be Worked Upon During the *Upcoming* Week (this section becomes section 1 of the next week):

1. Not cry
2. Test outputs against acceptance criteria
3. Make more class diagrams
4. Run through to make sure documentation is accurate and complete

V. General Reflection on the Week:

1. What went well? Things got done, procrastination time is no longer viable.
2. What did not go well? Trouble with node, getting slammed with many things at once.
3. What should we do more of? Maybe stressing because it helps not procrastinate, it sounds weird but it works. More progress, feels good.
4. What can we do less of? Not much, maybe just not filling this sheet out at the last second?

Test Cases / Test Outputs:

Seen above in the “Acceptance Criteria” section.

Login System	
Attempt to login with non-existent username and password combination when DB has no user entries	Fail to login

Non-existent username and password	Fail to login
Existing username incorrect password	Fail to login
Existing password incorrect username	Fail to login
Existing username and non-matching existing password	Fail to login
Matching username and password combination	Successful login

Item Listings	
View page with no data in DB	Blank table with headers
View page with one entry in DB	One entry in table with headers
View page with n entries in DB	Background box scaled to match table size, n entries in table with table headers.

Test Output Login:

employeeid	employeefirstname	employeelastname	username	password
Filter	Filter	Filter	Filter	Filter
1223	John	Bruh	johnbruh123	secretpassword
1225	Smith	Man	sman125	smithdude

Database table used

Please login

Login incorrect.

Employee username
noentries

Employee Password

Attempted login with empty employees table in the DB.

Please login

Login incorrect.

Employee username
random

Employee Password

Non-existent username and password output

Please login

Login incorrect.

Employee username
johnbruh123

Employee Password

Existing username with non-existent password

Please login

Login incorrect.

Employee username
random

Employee Password

Non-existent username with existing password

Please login

Login incorrect.

Employee username
johnbruh123

Employee Password

Existing username with non-matching existing password

Please login

Login successful. Redirecting now...

Employee username
johnbruh123

Employee Password

Matching username and password combination

Test Output Item Listing:

id	name	description	quantity	price
Filter	Filter	Filter	Filter	Filter
00000	Hat	Really Cool Hat!!!!1	37	5.05
00001	Shirt	Neat shirt I guess	4	25.0
00002	Du...	Dummy Desc	0	0.00
00003	Du...	Dummy Desc	0	0.00
00004	Du...	Dummy Desc	0	0.00

Table of merchandise items

Merchandise

Page viewed with no entries in DB. Failed to pass test as no table is displayed with headers only.

Merchandise

Id	Name	Description	Quantity	Price
00000	Hat	Really Cool Hat!!!!!!	137	5.05

Page viewed with one entry in DB.

Data is properly displayed

Merchandise

Id	Name	Description	Quantity	Price
00000	Hat	Really Cool Hat!!!!!!	137	5.05
00001	Shirt	Neat shirt I guess	4	25
00002	Dummy Item	Dummy Desc	0	0
00003	Dummy Item	Dummy Desc	0	0
00004	Dummy Item	Dummy Desc	0	0

Page viewed

with n=5 entries in DB. Background box scaled to match table size

SCRUM Work Completed:

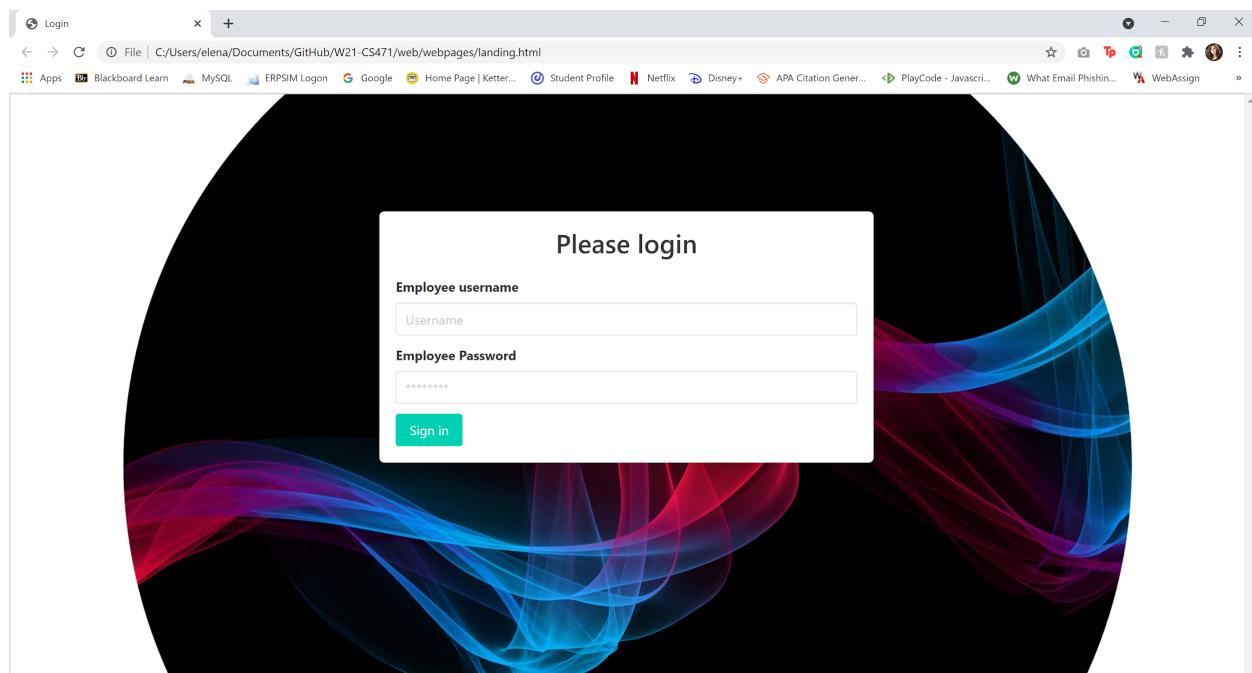
- All of Sprint 0 was completed as prerequisites to starting the programming.
 - A github repository was created.
 - All members researched the languages needed for the project.
 - A basic webpage was created.
 - A basic database was linked to that webpage.
- Most of Sprint 1 was completed.
 - Login systems work, however there is no distinction between cashier and mechanic.

- All inventory is accessible once logged in.
- Login screens are professionally designed.
- Sprint 2:
 - Inventory pages show each category now
 - Index screen is now designed
- Sprint 3:
 - Database screens are now available after login
 - Page design loads with the rest of the page
 - Search and management did not get completed.
- Documentation tasks were also added throughout sprints, however not added into the sprints board on Trello.
-

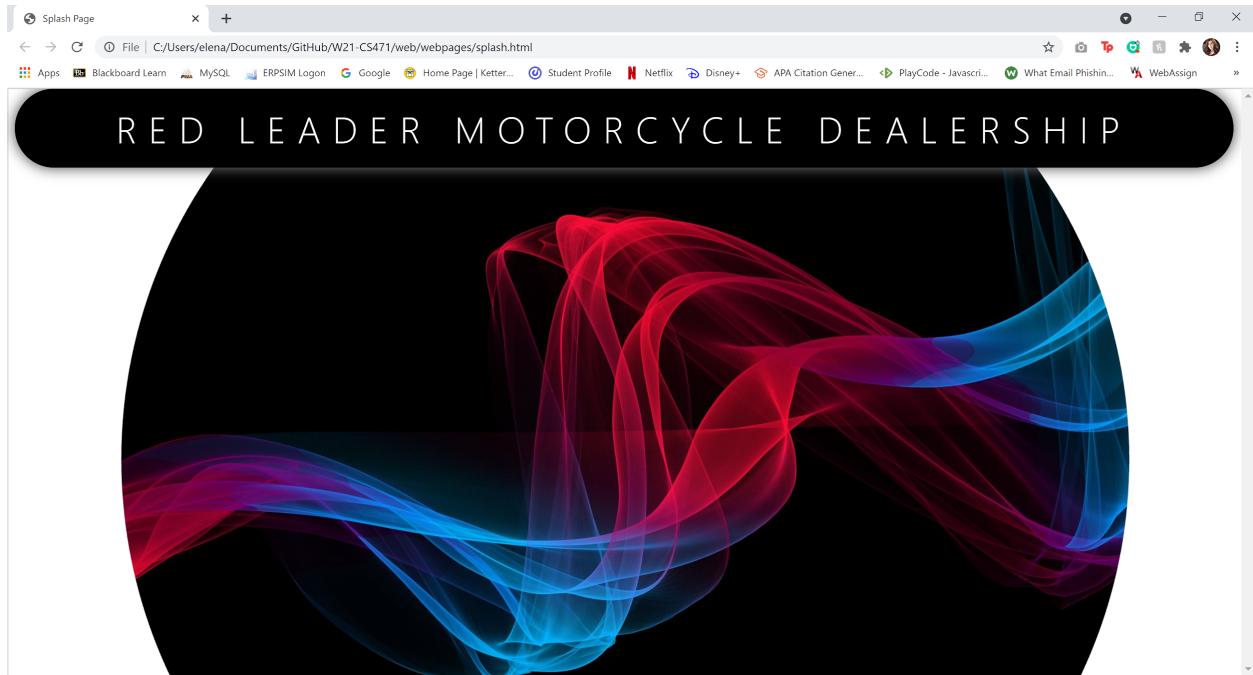
Additional Information:

Front-end Display

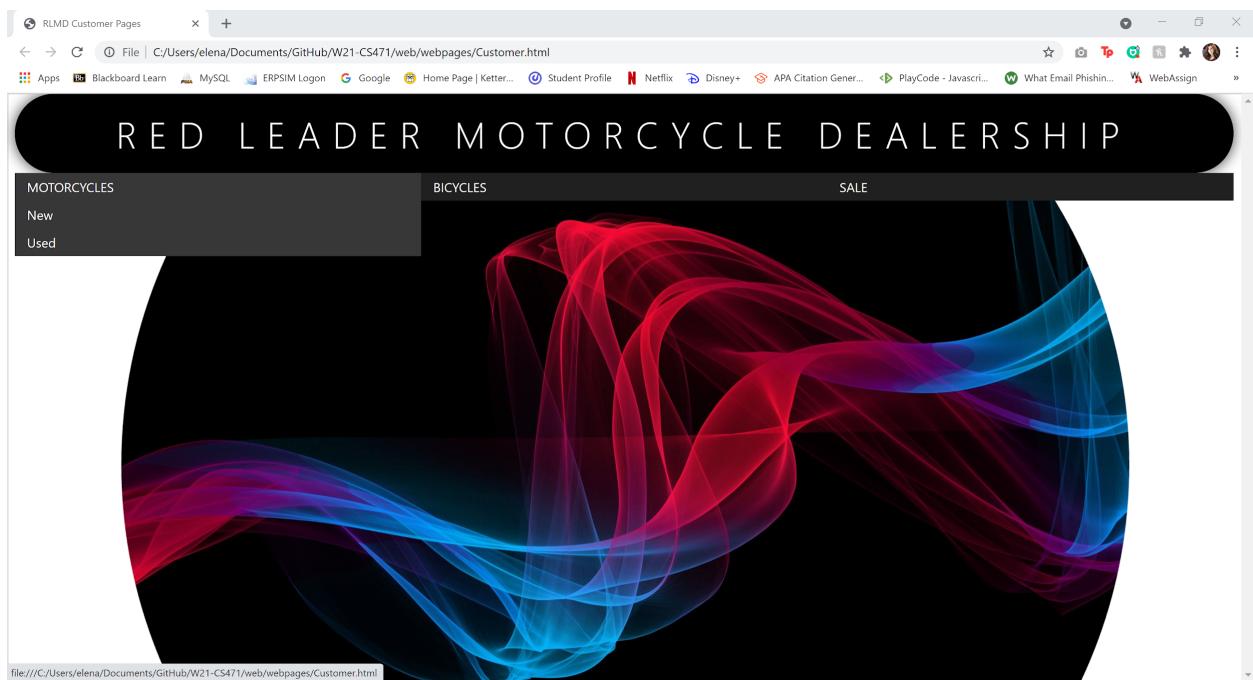
Landing Log-in Page

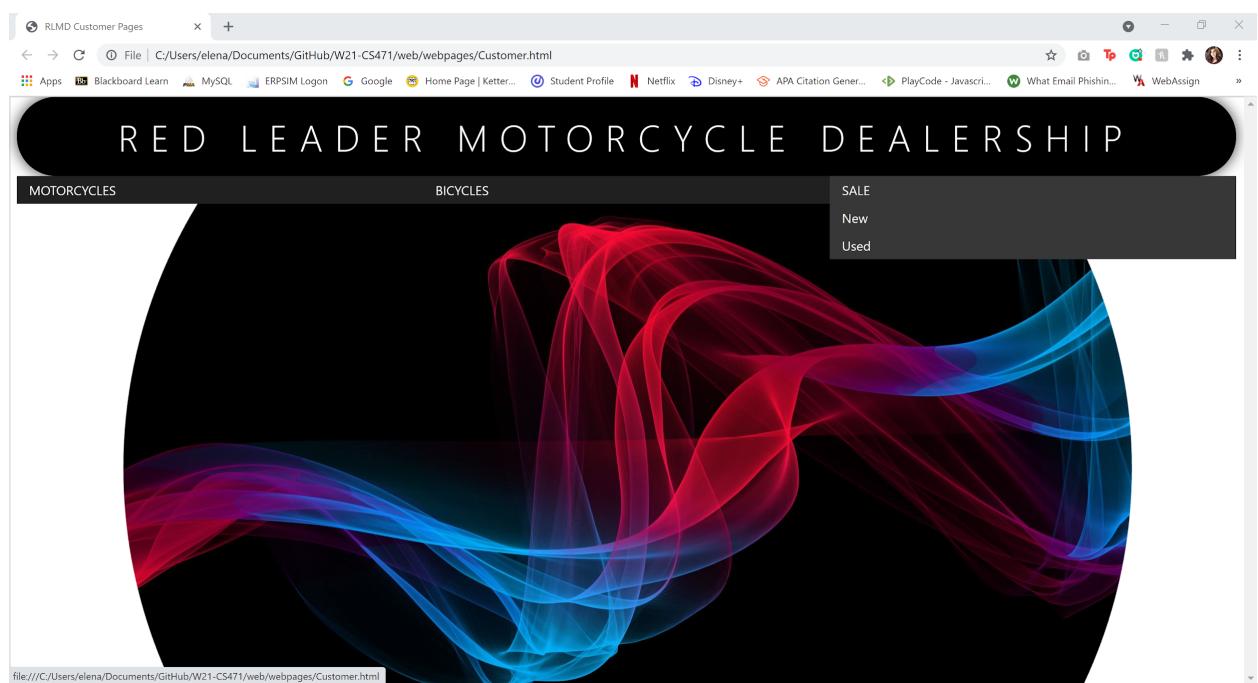
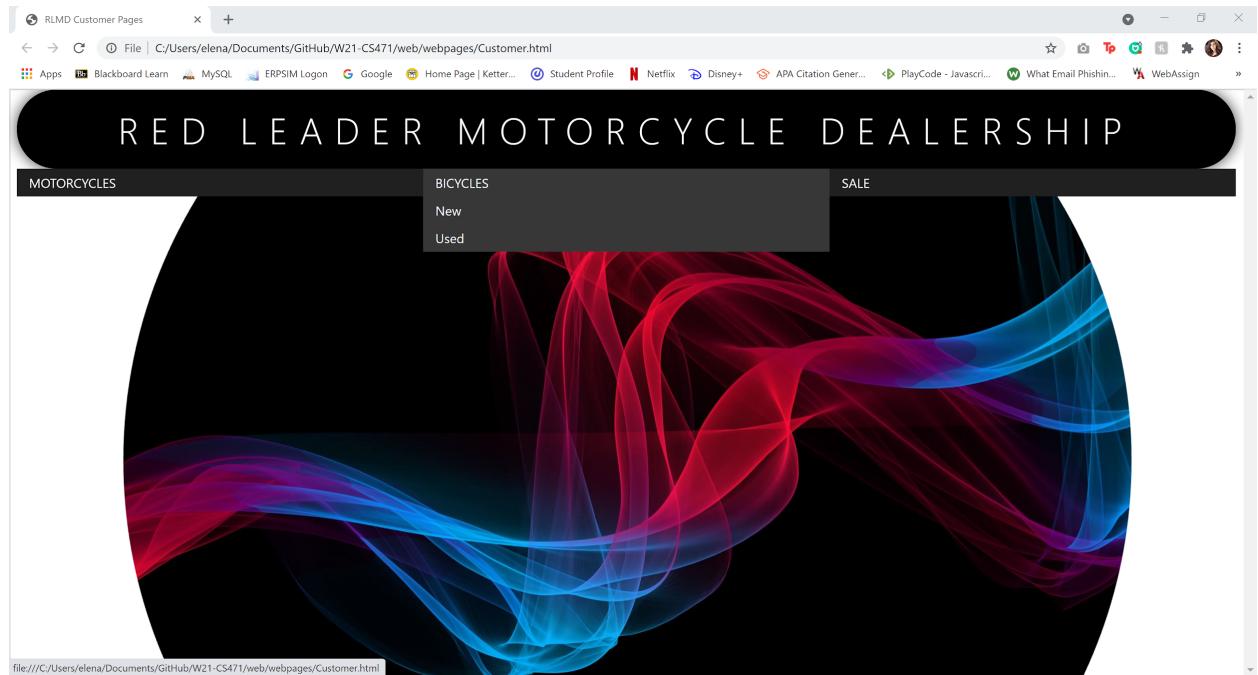


Splash Page



Customer Views:





Business Views:

