CALIFORNIA STATE UNIVERSITY
FULLERTON™

Department of Computer Science

This project has been satisfactorily demonstrated and is of suitable form.

This project report is acceptable in partial completion of the requirements for the Master of Science degree in Software Engineering.

A Tea Online Shopping Website - Teamee
Project Title

Holly Do
Student Name

Ning Chen, Ph.D.
Advisor's Name

Advisor's signature Date

Reviewer's Name

Reviewer's signature Date

Abstract

This project is an online tea and coffee store that provides users with information about the Teamee company, and allow the user to place orders on the website. Ecommerce is a blooming industry that encompasses various goods and services.[3] It is necessary in today's world for any company that sells products or services to have a website. The Teamee website is highly customizable and uniquely different from the typical Wordpress website.

This is a developmental project; thus, the core of the report would be within all my documentations of my project. There will be a total of three documents: the SRS, design, and test plan. The SRS document provide a detailed description of the Teamee system. The design document contains an outline of how the Teamee system will be implemented to satisfy all the requirements listed in the SRS. Finally, the test plan document will contain the methods that were used to ensure that the Teamee system is held to the highest quality. However, the primary document for my project would be the SRS (software requirement specification). It contains all the information necessary to describe what my project is about and a full description of the Teamee system.

The creation of the Teamee website will help increase its potential customer base. Anyone with internet access can view the website. Also, the Teamee online store provides users a convenient way to skip the line and procure goods from Teamee to satisfy their craving. The convenience is extended to employees since the online store will also be beneficial for Teamee's employees. The Teamee system will provide a way for admin to track and fulfill orders. Additionally, by providing information about the stores and products provided by Teamee, this will reduce questions that customers will have.

Table of Contents

1.0 Introduction

1.1 Background

As Amazon profit has shown, the ecommerce field is blooming. More and more consumers are using the internet to make purchases. Shopping online have many benefits for consumers which include:

1. Full catalog
2. Reviews posted by other customers
3. Shipped to home

In addition, companies also benefit from creating an ecommerce website. Expanding their business online would widen their customer base. Data can be collected from traffic through the website. Using the data collected, companies can determine market trends and customize their service towards their customer needs.

Generally, when ecommerce sites are mentioned, people think of Amazon, eBay, or other big name retailors. All kinds of products are sold on these websites. Then the food industry began to enter the internet space. In recent years, an emergence of websites and apps appear that allows consumers to order food or drinks online and have them delivered or ready to be picked up. Consumers have been shifting away from placing food or drink orders using the phone; and instead, they are using the internet to place orders.

1.2 Problem Statement

Online shopping has increased steadily throughout the years. Consumers increasingly looking towards online websites to procure goods and services. Thus, companies need to have their own website to attract potential customers. According to IBIS world, more people are connected online through their phones and online retail stores are a billion-dollar industry.[2] In the future, any company will need a website to provide information about itself or sell products or services. Therefore, it is necessary to learn how to develop attractive and responsive website.

In addition, the food industry has been penetrating the online markets. For instance, Starbucks have a well-developed website and app to attract and retain their fans. Even Uber had created an app to deliver food. Dominos is another prime example of this trend since it allows the user to create and customize their pizza order. The website is easy to use and convenient to operate since it provides a full menu of Dominos' selection. These flourishing examples demonstrated how services and goods can be rendered successfully online.

Food trends are constantly shifting, especially in regards to dessert. In the past, there were boba craze, ice cream, cupcakes, and other delectable treats that materialized. The US have a sweet tooth and it has proven to be insatiable.

1.3 Motivation for Project

My motivation for this project stems from the fact that I love tea. I hope to one day open a tea place for people to congregate and relax. I've always enjoyed gathering at a tea place with my friends; thus, it is my wish to own one in the future. This project had provided me with an opportunity to create a website and web app for my prospective store. Additionally, creating the Teamee web app had provided me with a project to show prospective employers. By building an ecommerce application, I had demonstrated a highly marketable skill as the ecommerce field is blooming. Furthermore, this project will help me demonstrate my ability to formulate a whole project from start to end.

2.0 Objectives

The objective is to create a functioning website and web app with an easy to use UI. The creation of the website and web app serves the following purpose:

2.1 Improve programming skills

1.1.1 Improve upon existing programming languages

1.1.2 Learn new languages such as SQL to implement the backend of the project.

2.2 Demonstrate to future employers the capability to complete a whole project

1.1.3 Create an effective software process to develop the Teamee website and web app

1.1.4 Provide evidence of software process

3.0 Resources

3.1 Hardware

The hardware that was used to create this project is a Lenovo Thinkpad E570 with an Intel i7 core. The laptop is running with Microsoft Windows 10 Pro.

3.2 Software

4.2.1 Frameworks: Bootstrap v3.3.6, Materialize, Laravel

4.2.2 Microsoft Office 2016

4.2.3 Microsoft Visio 2013

4.2.4 GitHub

4.2.5 Adobe Photoshop 2017

I chose Bootstrap and Materialize as my frameworks for the UI because it is one that I am closely familiar with. I have used it to develop webpages before so I am confident that I can use it to create the UI for my website. Laravel was also chosen based upon previous work experience.

GitHub will be used as my source control and repository. GitHub was chosen because it is one of the most popular Git repository hosting service and I have used it before for work. As a result, it would be one less thing for me to learn.

Adobe Photoshop was used to design the logo and other photo manipulations. The main use of Photoshop was to crop pictures, logo placement in pictures, and modification of pictures. For a website to be attractive to users, it was necessary to produce quality pictures.

3.3 Environment

4.3.1 Visual Studio 2015

4.3.2 Brackets

Visual Studio was chosen as my IDE because I have used it before and I am most familiar with its layout. I also listed Brackets because I intend to use it for my front-end development. Again, the reason why I chose Brackets is because I am familiar with it.

3.4 Learning Resources

There are numerous websites that are available to use as a resource for learning new languages. I will use w3school.com to review HTML, Javascript, and CSS. Also, I will use w3school.com to learn SQL because I am already familiar with the website and the tutorials that I've found on w3school was constructive toward my understanding of HTML, Javascript, and CSS. I will use PHP for backend development to create my database since Laravel is a PHP framework.

4.0 Planning and Staging

4.1 Timeline

- Iteration 1: Create a basic Teamee website

  Create layout of the website. Basic website with homepage, about us, contact, location, and menu pages.

- Iteration 2: Improve Functionality and Connect to database

  Implement backend of website. Create database and connect database to the website.

- Iteration 3: Satisfy business value and finalize the Teamee website

  All functional requirements are completed. The website is thoroughly tested in this iteration.

4.2 Work Products

There are four work products associated with Scrum. The four work products are:

1. Product Increment
2. Sprint Backlog
3. Product Backlog
4. Burndown Chart

4.2.1 Product Increment

The product increment is the product that is released after each sprint. The product should be a potentially shippable product that the customer can use and provide feedback upon. The product should contain all the use stories included in the sprint backlog. In addition, the product should be tested and documented.

4.2.2 Product Backlog

The product backlog contains all the items that are necessary to complete the product. It represents what will be built. The items in the product backlog are flexible to change. Items in the product backlog can include features, use cases, enhancements, defects, and technologies.

4.2.3 Sprint Backlog

The sprint backlog contains items from the product backlog. Items placed in the sprint backlog are considered the goals for that sprint. Items on the sprint backlog are delivered at the end of the sprint. At the end of each sprint, items for the next sprint are put into the sprint backlog. The categories for the sprint backlog are “Pending”, “Working On”, and “Completed”. All items began at pending, and are moved to working on when it is being implemented. Then after the item is finished and tested, it is moved to the completed section.

4.2.4 Burndown Chart

The burndown chart is demonstrated the amount of work remaining across time. This chart is also used to keep track of the progress made throughout each sprint. Moreover, it is also used to reflect the progress made against the product backlog. The burndown chart is updated throughout the length of the sprint.

4.3 Major Features

FR-1 Register for account

FR-2 Add item to cart

FR-3 Checkout items

FR-4 Search for the closest location

FR-5 View all locations

FR-6 Contact Teamee

FR-7 Sign up for promotions and news

FR-8 Obtain a receipt of the order

FR-9 Check all orders

FR-10 Complete an order

FR-11 View user’s contact information

FR-12 View website

FR-13 View items sold by Teamee

4.4 Product Backlog

Name	Description	Story Points
View website	As the user, I want to be able to view the Teamee website on my internet browser	8
Register for an Account	As the user, I want to be able to register for an account on Teamee so that I can view my orders.	5
Add item to cart	As the user, I want to be able to add an item to cart so that I can checkout my orders after I am done shopping.	5
Checkout items	As the user, I want to be able to checkout my items in my shopping cart so that I can pay and pick up my order at the store.	21
Search for the closest location	As the user, I want to be able to search for the closest Teamee store so that I can know which store to select to pick up my order.	13
View all locations	As the user, I want to be able to view all of Teamee's locations so that I can plan a trip to any Teamee location.	8
Contact Teamee	As the user, I want to be able to contact Teamee so that I can ask a question or leave a comment.	3
Sign up for promotions and news	As the user, I want to be able to sign up for promotions and news from Teamee because I am a fan of Teamee and I want to get any savings from Teamee.	3
Obtain a receipt of the order	As the user, I want to receive a receipt of my order in case something goes wrong or for my spending records.	8
View Items sold by Teamee	As the user, I want to be able to view all items sold by Teamee so that I can order the items that I want.	5
Check all orders	As the admin, I want to be able to look up all the current pending orders so that I can fulfill them and satisfy my customers.	8
Complete an order	As the admin, I want to be able to complete an order so that it can go to the completed section and I can keep track of my orders properly.	3
View user's contact information	As the admin, I want to be able to view the user's contact information so that I can contact them if something is wrong.	5

Figure 1: product backlog contains all the features with a user story as the description for each feature.

All the functional requirements were placed into the product backlog with a description set as the user story. A story point is assigned to each functional requirement based upon the Fibonacci sequence. The Fibonacci sequence was used because it helped provided a relative estimate of each user story to create a more accurate estimate. Register for an account was used as the benchmark to compare the other user stories.

4.5 Release Plan

A release plan was created after the product backlog was completed. The release plan was created based upon the set timeline with 3 iterations. Each iteration was timeboxed to two weeks due to time constraint.

Iteration 1	Iteration 2	Iteration 3
Register for an account (5)	Add an item to cart (5)	Check all orders (8)
View all locations (8)	Checkout items (21)	Complete an order (3)
Search for closest location (13)	View Items Sold by Teamee (5)	View user's contact information (5)
Contact Teamee (3)		Signup for promotions and news (3)
View Website (8)		Obtain a receipt of the order (8)
37	31	27

Figure 2: A chart listing all the features to be completed per each iteration.

Iteration 1 contained the functional requirements that were deemed the most critical while the last iteration have the least number of story points. I wanted to keep risk at the lowest near the end of the project.

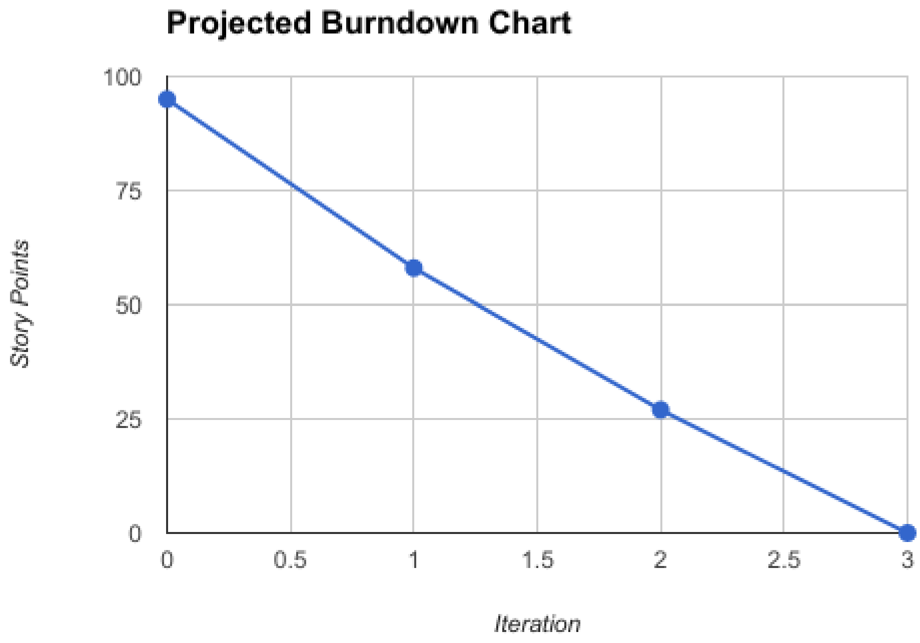


Figure 3: The projected burndown graph using all the story points for each iteration.

Based upon the product backlog chart that was produced, a projected burndown chart is created. This projected burndown chart will be used to measure the developmental progress. The total number of story points was 95 and after the first iteration it was expected that 37 story points should be completed. Then after the second iteration, another 31 points should be completed leaving only 27 story points for the last iteration.

5.0 Software Requirement Specification (SRS) Document

An Online Shopping Website - Teamee

Software Requirements Specification

1.0 Introduction

1.1. Purpose

The purpose of this Software Requirement Specification (SRS) document is to provide a thorough description of the software requirements for the Teamee web application. This document can be used to foster a better understanding of the Teamee web app and elucidate stakeholders of the software requirements. The requirements stated in this document are subjected to changes, and any changes shall be updated within this document. This documented is intended for its stockholders such as its software developers, project managers, and users.

1.2. Document conventions

This SRS document will be explained by textual descriptions of the concepts, diagrams to illustrate relationships, and tables to present information.

1.3. Product Scope

Teamee web app is tended to provide customers and potential customers information about Teamee which includes its products, locations, and services. This web app will be available for anyone with an internet access. Users can place their orders online and pickup in store. Administrators will be able to view all recent and past orders.

1.4. Definitions, acronyms, and abbreviations

Term	Definition
Users	People who access the website
Software Requirements Specification (SRS)	SRS a description of the software to be built
Administrators (Admin)	People who manages the website
Entity Relationship Diagram (ERD)	A diagram that shows the relationship between entities in a database
Website Application (web app)	A software that is created to run using an internet browser

Figure 4: Definitions chart

1.5. References

Wiegers, Karl, *Software Requirements (3rd Edition*), Microsoft Press 2013.

1.6. Overview

The remaining section of this SRS describes the functional requirements for Teamee website.

2.0 Overall description

2.1. Product Perspective

The end product will be a fully functional website that any users can access through the internet.

2.1.1. System Interfaces

The interface for this software is a web browser

2.1.2. User Interfaces

2.1.2.1 Home Page

- The navbar shall be transparent with the company’s logo on the left

- When the user hovers over a menu option with a dropdown, the dropdown will be displayed
- The Home Page shall contain video
- There shall be a carousel with images about Teamee to entice users
- The footer shall contain social media link, contact information, copyright, and terms of use

2.1.2.2 About Us

- There shall be a description about the company
- There shall be images about the company
- Information about where the tea sold by Teamee is cultivated from

2.1.2.3 Locations

- Displays a map with all Teamee's location

2.1.2.4 Contact

- A form that allow the user to send an email to the administrators of Teamee

2.1.3. Hardware interfaces

Mouse and keyboard are required.

2.1.4. Software interfaces

The project software interface will be viewed on a web browser

2.1.5. Communication interfaces

The project shall use email and phone number.

2.1.6. Memory constraints

N/A.

2.1.7. Operations

Operation will be when the website is being accessed until the users exited.

2.1.8. Site adaptation requirements

The website shall work on any web browser proportionately and correctly.

2.2. Product functions

Teamee provides the following functionality:

- **ABOUT US**: Provides information about the company.
- **LOCATIONS**: Provides the nearest store locations based on zip code
- **MENU**: Provides a list of all the products that Teamee sells.
- **CONTACT US**: Allows users to send an email message to Teamee.

2.3. User classes and characteristics

Teamee is made for different types of users:

- **Users** are people who accessed the website. Users are permitted to access the entire website, but must be logged in to place an order. A free account is provided to any user who wishes to make one.
- **Administrators** are responsible to take care of the Teamee website and messages. Administrators must login to gain access to the administrator portal. The administrator will be able to view users' account information, but not the user's password. The administrator can view current, past, and future orders made by users.

2.4. Constraints

CO-1 All HTML codes shall follow the HTML 5.0 standard.

CO-2 The Navigation menu tool bar shall be available on all Teamee's web pages.

CO-3 Online payments shall be made through valid credit card companies which includes Visa, Discover, and Mastercard.

2.5. Assumptions and dependencies

Assumption:

AS-1 Users have the plugins that are required to watch videos

AS-2 Users have internet connection to access the web app

AS-3 Users' internet connection speed is sufficient to watch videos on the web app

Dependencies:

DE-1 Users should have the basic knowledge of computers, so they can easily access the product by using the user manual provided to them with the product.

3.0 System Features

3.1 About Us

This webpage will contain a description of the company.

3.2 Locations

Users can find the nearest Teamee location from the zip code they provided.

3.3 Menu

Contains 3 tabs which allows the user to navigate to the following three options:

Tea

- Images of all the tea drinks available from Teamee
- Description of the tea

Coffee

- Images of all the coffee drinks sold by Teamee
- Description of the type of coffee

Snacks

- Images of all the snacks sold by Teamee
- Description of each snack product

3.4 Contact

Users can find Teamee's contact information, address, and send an email message.

4.0 Quality Attributes

4.1 Usability

USE-1 Teamee home page shall be navigable
USE-2 There shall be a navigation bar on every page with an option to go back to Home
USE-3 There shall be a Contact option for users to contact the administrators with any questions about the website

4.2 Performance requirements

PER-1 Homepage shall load under 10 seconds

PER-2 Any videos shall load under 5 seconds

PER-3 Items are added to cart within 3 seconds

PER-4 All error prompts shall be displayed within 2 seconds

PER-5 The notification for successful signup shall be displayed under 3 seconds

5.0 Functional requirements

Functional requirements are requirements that are created stakeholders and should follow the user stories. These requirements are essential to make the system work.

FR-1 Register for account

FR-2 Add item to cart

FR-3 Checkout items

FR-4 Search for the closest location

FR-5 View all locations

FR-6 Contact Teamee

FR-7 Sign up for promotions and news

FR-8 Obtain a receipt of the order

FR-9 View website

FR-10 View all items sold by Teamee

FR-11 Check all orders

FR-12 Complete an order

FR-13 View user's contact information

5.1 FR-1: Register for an account

Description

This functional requirement will allow the user to register for a new account at the Teamee website.

Input

- The user shall input his name, phone number, address, and email
- The user may enter his gender and date of birth
- The user shall click the submit button once he is done

Processing

- The system shall verify that all required fields were filled out
- The system shall store the user's information into the database
- The system shall send a confirmation email to the user's account

Output

The user’s account is temporary created. To complete registration, the user need to click on the link in the confirmation email.

Error Handling

- If the user entered an email address that already existed in the database, an error message will display informing the user that the email already existed.

- The system shall check to see if the user filled out all the required fields once the user clicked on the submit button. If the user failed to fill out all the required fields, then an error message shall be displayed.

5.2 FR-2: Add item to cart

Description

The user shall be able to add items into the shopping cart. Multiple items can be added to the cart.

Input

- The user shall select the item that he wished to put into his cart and include the quantity to be added to cart
- The user shall click on the add to cart button

Processing

- The system shall verify that a quantity is selected

- The system shall add the item into the cart and update the item count

Output

The system shall display the number of items in the cart

Error Handling

- If the user did not select the quantity, the system shall display an error message prompting the user to select a quantity.

5.3 FR-3: Checkout items

Description

After the user had placed items into the cart, the user can check out the items and place an order.

Input

- User input his name, email, phone number, and credit card
- User select the store to pick up the order
- User click place order button

Processing

- The system shall verify the user’s payment through Stripes
- The system shall confirm that the user had input all the required fields
- The system shall store the user’s information into the database
- The system shall create an order number
- The system shall send a confirmation email to the user’s email

Output

A confirmation page is displayed to inform the user that the order was placed successfully. An email confirmation shall be sent to the user's email address.

Error Handling

- Stripes was unable to verify the user's payment. The system shall display and error page to inform the user that the order was not placed.

5.4 FR-4: Search for the closest location

Description

The user shall be able to search for the closest Teamee location using either an address or zip code.

Input

- User shall input an address or zip code in the search bar
- User shall click the search button

Processing

- System shall search for the nearest Teamee location using Google map API

Output

The system shall zoom the user to the nearest Teamee location and an info window for the store shall be displayed. The info window shall contain the store's address, picture, and phone number.

5.5 FR-5: View all locations

Description

When the user enters the location page, a map with all the Teamee's location shall be displayed with markers showing each store's location.

Input

- User click on the location tab

Processing

- System shall display a google map
- System shall show markers of all the Teamee locations

Output

A google map zoomed into the Teamee locations. The user shall be able to view all the Teamee locations at once.

5.6 FR-6: Contact Teamee

Description

A contact page shall allow the user to fill out a form and send it to Teamee's staff.

Input

- User shall click on the contact link
- User shall fill out the contact form
- User shall click the submit button

Processing

- System shall verify that all required fields are filled out
- System shall send the message along with the user's contact information to info@Teamee.com

Output

The user's message shall be sent to info@Teamee.com. A success message shall be displayed for the user if all the required fields were filled out.

Error Handling

- If the user did not fill out all the required filled, an error prompt shall be displayed.

5.7 FR-7: Sign up for promotions and news

Description

The signup bar shall allow the user to input his/her email address and be eligible

Input

- User shall enter in his email address
- User shall click on the submit button

Processing

- System shall verify that the user had not left the field blank
- System shall verify that the user had entered an email address

Output

The user's email shall be stored in Teamee's database.

Error Handling

- If the user had left the field blank and clicked the submit button, the system shall display an error prompt to inform the user that this field cannot be blank.

5.8 FR-8: Obtain a receipt of the order

Description

The user is given a receipt every time an order is placed. Teamee shall send a receipt in form of an email confirmation.

Input

- Order is submitted

Processing

- System shall retrieve the user's email address
- System shall send a receipt email confirmation to the user's address

Output

The user shall receive a receipt which contains the items and price that the user had paid

5.9 FR-9: View website

Description

The website shall have an attractive and easy to use UI for the user. There shall be a homepage, contact, locations, menu, and about us page.

Input

- User shall open an internet browser
- User shall enter in a Teamee URL

Processing

- System shall detect the URL that was entered
- System shall detect that the route for the URL entered exists

Output

System shall display the page on the browser if the correct URL is entered

Error Handling

- If the wrong URL is entered, an error page shall be displayed

5.10 FR-10: View all items sold by Teamee

Description

The user shall be able to view all the items sold by Teamee. Items shall be listed within their respective category.

Input

- User hovers over the menu tab in the nav bar
- User click on a category of item
- User can click on an item to get a detailed description

Processing

- System shall detect that user had mouse hovered over menu
- System shall detect the category that was clicked on

Output

System shall display the correct item page

5.11 FR-11: Check all order

Description

The admin shall be able to check on all orders that are made by the users

Input

- Admin input password and email address
- Users had placed some orders

Processing

- System shall check for all orders
- System shall retrieve all orders

Output

A list of pending orders is displayed to the admin

5.12 FR-12: Complete an order

Description

The admin can complete and order in the pending list.

Input

- Admin shall click on the order
- Admin shall click complete order once the order has been fulfilled

Processing

- System shall verify that order exist
- System shall change the status of the order from pending to completed

Output

The order is moved from under the pending list to the completed section. The order is stored in the database.

5.13 FR-13: View user's contact information

Description

The admin shall be able to view the user's contact information in case the admin need to contact the user about an order that was placed. Also, the admin need the user's contact information in order to send the user's news or promotions.

Input

- User shall input his contact information such as name, phone number, and email address
- Admin shall be logged in to view the user's contact information
- Admin click on the user's name to view a specific user's info

Processing

- System shall query for the user's contact information from the database

Output

A popup window shall display the user's contact information to the admin

6.0 Logical database requirements

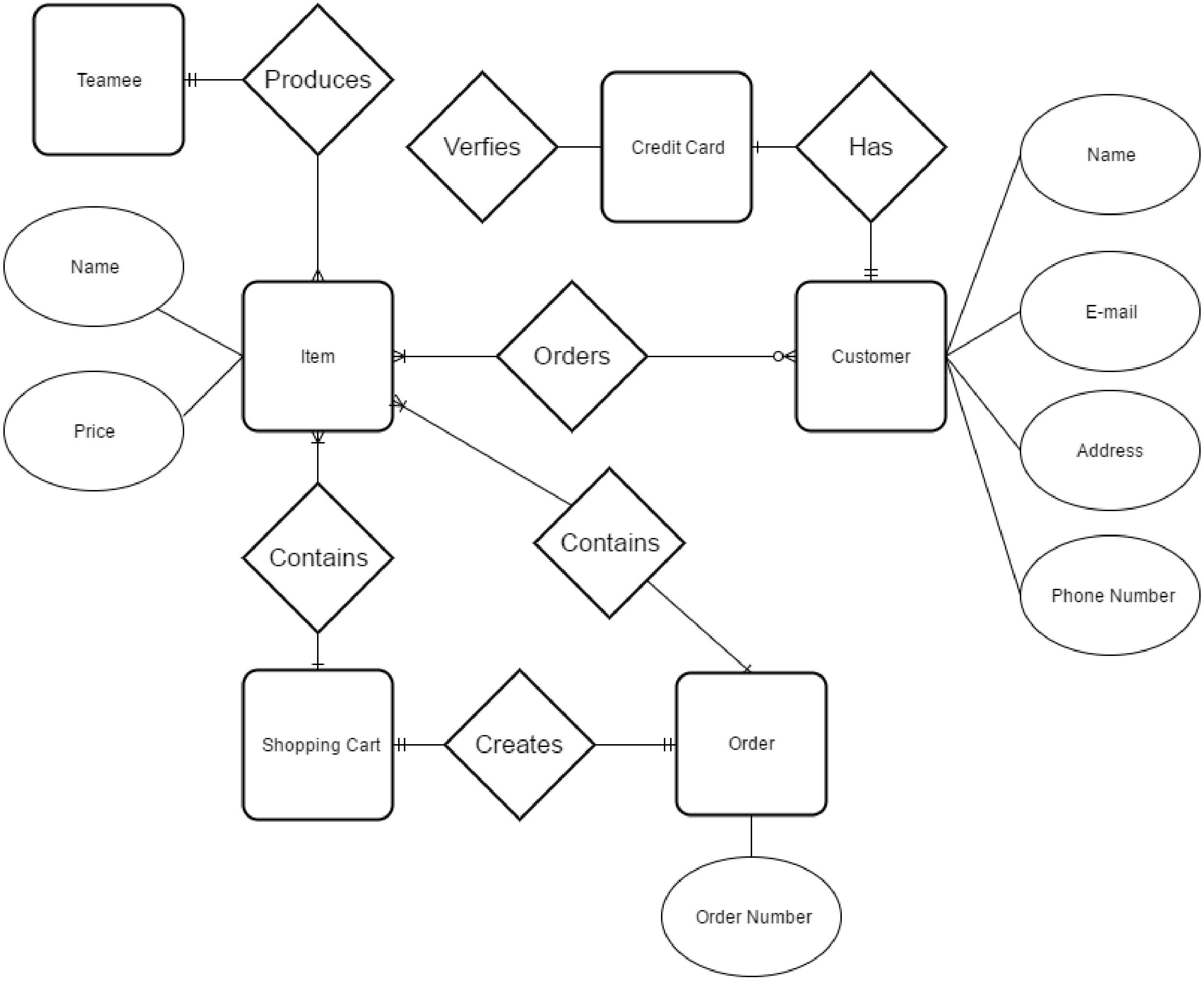


Figure 5: Entity Relationship Diagram

The ERD shows the relationship between entities within the Teamee system in the database.

6.1 Design constraints

DC-1 The system shall use Model-View-Controller

DC-2 The system shall use the Laravel framework

7.0 Software system attributes

7.1 Reliability

RE-1 The website shall be functional when being accessed

RE-2 Website shall not crash while user is accessing it

7.2 Availability

AVA-1 The Teamee web app shall be available 99% of the time every day except for maintenance windows. Maintenance window shall be within the time of 12AM-2AM local time.

7.3 Security

Only users with the correct authorization are allowed access.

7.4 Maintainability

MAIN-1 The code will be well documented

MAIN-2 Documents shall be updated when changes are made.

8.0 Use Case

8.1 Use Case Descriptions

Primary Actor	Use Cases
User	UC-1 Register for account UC-2 Add item to cart UC-3 Checkout items UC-4 Search for the closest location UC-5 View all locations UC-6 Contact Teamee UC-7 Sign up for promotions and news UC-8 Obtain a receipt of the order UC-9 View website UC-10 View all items sold by Teamee
Administrator	UC-11 Check all orders UC-12 Complete an order UC-13 View user's contact information

Figure 6: List of Use Cases

ID and Name:	**UC-1 Register for account**
Primary Actor:	User
Description:	The user shall register for an account on Teamee by clicking of the Register link. The user must enter his/her email information, username, and password to register.
Trigger:	User accessed Teamee's website and clicked on the Register link to register for an account.
Preconditions:	PRE-1 User must have a valid email address PRE-2 Username must be available PRE-3 Password must be at least 6 characters long
Postconditions:	POST-1 Teamee shall send a verification email to the user's email POST-2 The user shall click on the verification link in the email

	POST-3 The user’s information shall be stored in the Teamee’s database
Normal Flow:	1.0 Register for an Account 1. User click on the Register link on the Teamee web app 2. User enter a valid email address 3. User enter a username that is available 4. User enter a password that is at least 6 characters long 5. User click submit button 6. System shall display a confirmation screen that tell the user to check his/her email for verification
Alternative Flow:	1.1 Sign-in to a Registered Account 1. User with an existing account shall sign into his/her account 2. User sign in using his/her email with the correct password
Exceptions:	E1 Invalid email address E2 User name is taken E3 Password is not long enough E4 Password does not match confirm password
Priority:	High
Frequency of Use:	Dozens of users a day.
Other Information	N/A
Assumptions:	AS-1 User have an email address AS-2 Peak usage when web app first become available

ID and Name:	**UC-2 Add Item to Cart**
Primary Actor:	User
Description:	The user can browser drinks and products sold by Teamee and select products to be added to his/her shopping cart.
Trigger:	User accessed Teamee’s website and clicked on the Menu link located in the navigation bar. In the Menu page, the user can click on a product and click on the Add to Cart button.
Preconditions:	PRE-1 User must be in the Menu page PRE-2 User set quantity
Postconditions:	POST-1 Item is added to chart
Normal Flow:	2.0 Add Item to Cart 1. User click on a product in the Menu page 2. User select the quantity to be added 3. User click Add to Cart button 4. Item is added to cart 5. Number of items in cart is updated
Alternative Flow:	2.1 Item is Unavailable 1. An alert will display if the user attempts to add an item that is unavailable to cart 2. Item is not added into the cart
Exceptions:	N/A

Priority:	High
Frequency of Use:	Hundreds of instances per day
Other Information	N/A
Assumptions:	AS-1 User have internet connection to access the Menu page

ID and Name:	**UC-3 Checkout Items**
Primary Actor:	User
Description:	Checkout items allow the user to place an order.
Trigger:	User clicked on the shopping cart and proceeded to checkout.
Preconditions:	PRE-1 User must have a valid email address PRE-2 User must provide a shipping and billing address PRE-3 User must provide payment information such as credit card information
Postconditions:	POST-1 Teamee shall send an order confirmation to the user's email POST-2 User's order is stored into Teamee's database POST-3 Administrator shall be able to view the user's order
Normal Flow:	3.0 Checkout Items 1. User enters in email address, shipping, billing, and payment information 2. User click submit button 3. Teamee system verifies payment information 4. Teamee system stores user's order 5. Teamee system send order confirmation email to the user's email
Alternative Flow:	3.1 Sign-in to a Registered Account and perform checkout 1. User enter payment, shipping, and billing information 2. Teamee system saves user's payment, shipping, and billing information into the user's account. 3. Continue normal flow at 3.
Exceptions:	E1 Invalid email address, payment, shipping, or billing information
Priority:	Normal
Frequency of Use:	hundreds of users a day.
Other Information	Users do not need to have an account to checkout items.
Assumptions:	AS-1 User have an email address, billing, shipping, and payment information AS-2 Higher than normal usage during the weekend

ID and Name:	**UC-4 Search for the closest location**
Primary Actor:	User
Description:	The user shall be able to search for the closest location to himself.

Trigger:	User clicked on the search icon.
Preconditions:	PRE-1 User must be in the location page PRE-2 User must provide an address or zip code
Postconditions:	POST-1 A popup window will show the closest location to the user.
Normal Flow:	4.0 Search for the closest location 1. User type in a zip code or address in the search bar 2. An autofill will display based on the user's input 3. User clicks on the search icon 4. System shall display a popup window of the closest location to the address or zip code inputted
Alternative Flow:	N/A
Exceptions:	E1 Invalid address or zip code
Priority:	Normal
Frequency of Use:	hundreds of users a day.
Other Information	N/A
Assumptions:	AS-1 User have an address or zip code to input

ID and Name:	**UC-5 View all locations**
Primary Actor:	User
Description:	Users shall be able to view all the locations of Teamee on a map.
Trigger:	User clicked on the location link.
Preconditions:	PRE-1 User must have clicked on the location link.
Postconditions:	POST-1 A map with markers shall display all of Teamee's locations.
Normal Flow:	5.0 View all locations 1. User clicks on the location link 2. A map is generated using Google map API 3. Markers displayed all of Teamee's location 4. Users can click on a location marker and a information window shall be displayed
Alternative Flow:	N/A
Exceptions:	N/A
Priority:	Normal
Frequency of Use:	hundreds of users a day.
Other Information	Map is zoomed in to show all locations when page is loaded
Assumptions:	AS-1 There are at least 1 Teamee location

ID and Name:	**UC-6 Contact Teamee**
Primary Actor:	User
Description:	Users can contact Teamee by clicking on the contact link and filling out the contact form.

Trigger:	User clicked the submit button
Preconditions:	PRE-1 User must fill out all the required fields
Postconditions:	POST-1 An email is sent to info@Teamee.com with the user's input message and contact information
Normal Flow:	6.0 Contact Teamee through contact form on the website 1. User clicked on the contact link 2. User fill out the contact form 3. User clicked on the submit button 4. A success or error message is displayed 5. An email is sent to the info@Teamee.com with the message from the user
Alternative Flow:	N/A
Exceptions:	E1 info@Teamee.com inbox is full E2 User did not fill out all the required fields
Priority:	Normal
Frequency of Use:	hundreds of users a day.
Other Information	Users do not need to have an account to contact Teamee
Assumptions:	AS-1 User have an email address

ID and Name:	**UC-7 Sign up for promotions and news**
Primary Actor:	User
Description:	Users can sign up to receive news and promotions from Teamee by entering in their email address.
Trigger:	User typed in his email address and clicked the submit button
Preconditions:	PRE-1 User must have an email address
Postconditions:	POST-1 Teamee shall store the user's email address in the database
Normal Flow:	7.0 Sign up for promotions and news 1. User type in his email address 2. User clicked submit button 3. The user's email address is stored within the Teamee's database
Alternative Flow:	N/A
Exceptions:	E1 Invalid email address
Priority:	Normal
Frequency of Use:	dozens of users a day.
Other Information	Users do not need to have an account to sign up for promotions and news
Assumptions:	AS-1 User have an email address

ID and Name:	**UC-8 Obtain a receipt of the order**

Primary Actor:	User
Description:	After the user placed an order, a receipt of the order is sent to the user’s email address.
Trigger:	User clicked on Place Order button and Stripe has cleared the payment.
Preconditions:	PRE-1 User must have a valid email address PRE-2 Stripe cleared the payment
Postconditions:	POST-1 Teamee shall send an order confirmation to the user’s email
Normal Flow:	8.0 Obtain a receipt of the order 1. The system shall send a receipt to the user’s email address
Alternative Flow:	N/A
Exceptions:	E1 Invalid email address E2 Payment did not clear
Priority:	Normal
Frequency of Use:	hundreds of users a day.
Other Information	The receipt shall follow the format specified in the document layout section.
Assumptions:	AS-1 User have an email address AS-2 User’s inbox is not full AS-3 User’s payment successfully cleared

ID and Name:	**UC-9 View website**
Primary Actor:	User
Description:	The user shall be able to view all the web pages from Teamee
Trigger:	User enters in Teamee url
Preconditions:	PRE-1 User must have a browser open PRE-2 User must enter in Teamee url
Postconditions:	POST-1 Teamee website shall be displayed
Normal Flow:	9.0 View website 1. User opens website browser 2. User enters in Teamee url 3. System shall display the Teamee homepage
Alternative Flow:	N/A
Exceptions:	E1 Invalid URL
Priority:	High
Frequency of Use:	hundreds of users a day.
Other Information	Web pages: 1. Homepage 2. Contact 3. About Us 4. Location
Assumptions:	AS-1 User is connected to the internet AS-2 User have a compatible web browser

ID and Name:	**UC-10 View all items sold by Teamee**
Primary Actor:	User
Description:	The user shall be able to view all items sold by Teamee.
Trigger:	User clicked on a menu item in the nav bar.
Preconditions:	PRE-1 User must have an internet browser open PRE-2 User must have an internet connection
Postconditions:	POST-1 Teamee shall display the items that are sold by Teamee
Normal Flow:	10.0 View all items sold by Teamee 1. User click on a menu item 2. System shall display all the menu items sold by Teamee
Alternative Flow:	N/A
Exceptions:	N/A
Priority:	Normal
Frequency of Use:	hundreds of users a day.
Other Information	The user shall be able to click on each menu item and view a more detailed description of each menu item.
Assumptions:	AS-1 User is connected to the internet AS-2 Teamee have items to display

ID and Name:	**UC-11 Check All Orders**
Primary Actor:	Administrator
Description:	The Administrator can check all orders made by the users. Orders are separated by pending, completed and future orders.
Trigger:	Administrator logged into her account.
Preconditions:	PRE-1 Administrator must be logged in
Postconditions:	POST-1 Administrator can edit orders POST-2 Teamee system shall save an edits made by the administrator
Normal Flow:	11.0 Check All Orders 1. Administrator’s portal open to the Orders page 2. Orders are displayed with newest orders on top 3. Administrator can search through orders 4. Teamee system saves any changes made to orders
Alternative Flow:	N/A
Exceptions:	E1 Order was deleted from the system
Priority:	Normal
Frequency of Use:	Daily when the store is open
Other Information	Each store has 1 administrator account
Assumptions:	AS-1 Administrator account was set up

ID and Name:	**UC-12 Complete an Order**
Primary Actor:	Administrator
Description:	When an order is completed, the administrator will mark the order as completed.

Trigger:	Administrator clicked completed button.
Preconditions:	PRE-1 Administrator is logged into her account PRE-2 Order to be marked completed existed
Postconditions:	POST-1 Teamee system shall move the completed order from pending to the completed section
Normal Flow:	12.0 Complete an Order 1. Admin opened a pending order 2. Admin clicked completed button 3. Teamee system move the pending order into the completed order section
Alternative Flow:	N/A
Exceptions:	E1 Order does not exist
Priority:	Low
Frequency of Use:	Hundreds of uses per day
Other Information	N/A
Assumptions:	AS-1 Admin is logged in AS-2 Order was not cancelled while in pending

ID and Name:	**UC-13 View user's information**
Primary Actor:	Administrator
Description:	Admin can look for a user's contact information such as name and phone number.
Trigger:	Admin is in the admin portal and clicked on view profile on a user's name
Preconditions:	PRE-1 Admin must be logged in PRE-2 User must exist
Postconditions:	POST-1 A popup window shall display the user's contact information
Normal Flow:	13.0 View user's information 1. Admin logged in 2. Admin clicked a user's name 3. System shall display a popup window with user's contact information 4. Admin click the close button to close the popup window
Alternative Flow:	N/A
Exceptions:	E1 User does not exist
Priority:	Normal
Frequency of Use:	Hundreds of times a day
Other Information	User's contact information includes: 1. Name 2. Phone number 3. Payment type
Assumptions:	AS-1 User had previously placed an order

6.0 Software Design Document

1.0 INTRODUCTION

1.1 Purpose

This software design document describes the architecture and system design of the Teamee web application.

1.2 Scope

This document will focus on the GUI of the Teamee application. It will also contain designs of the website's software architecture.

1.3 Overview

This Design Document will contain a:

1. Purpose
2. Scope
3. Reference
4. System Overview
5. System Architecture
6. Data Design
7. Component Design
8. Appendix

1.4 Reference Material

Wiegers, Karl, *Software Requirements (3rd Edition*), Microsoft Press 2013.

2.0 SYSTEM OVERVIEW

The framework that shall be used is Laravel 5.4. Laravel is a PHP framework that contains its own template and supports PHP directly in its pages.

Front End Framework

1. Materialize
2. Bootstrap
3. Animate.css

3.0 SYSTEM ARCHITECTURE

3.1 Architectural Design

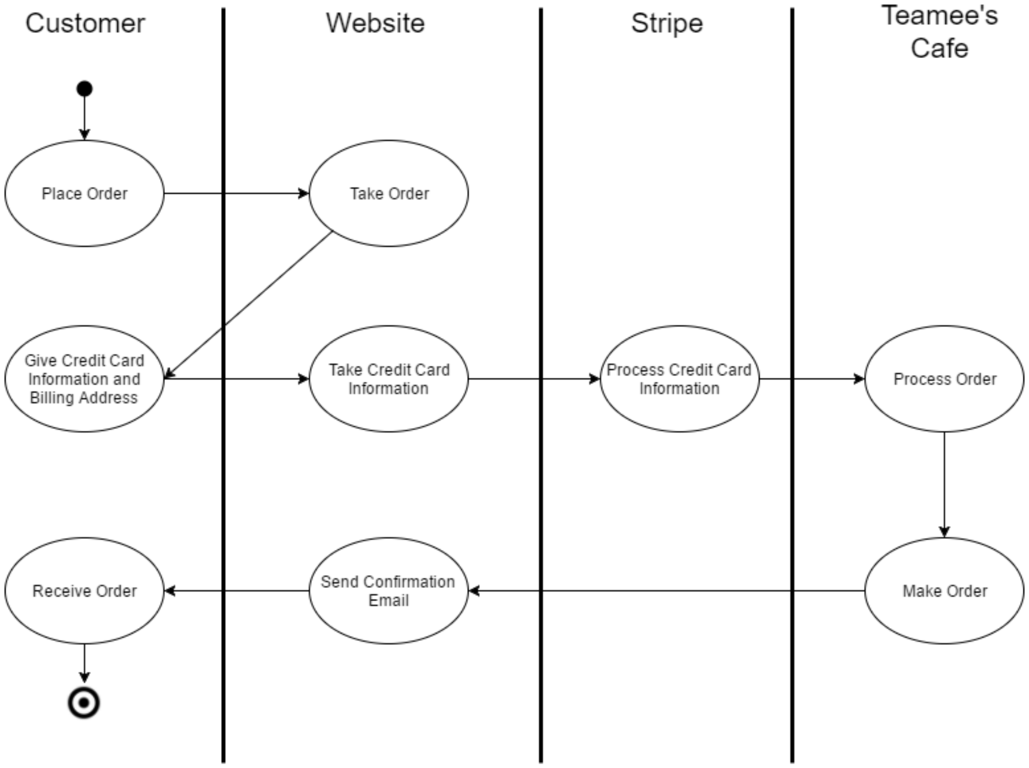


Figure 7: Swimline diagram

The swimline diagram showed a flow when a customer began to place an order on the website. This diagram helps shows the entities that are involved and the activities that take place. The flow began when customers placed items into a chart, they can then go to checkout. During checkout, customers are asked for some personal information such as name, billing address, and credit card information. The information is transferred to Stripe, a company that helps in the billing process. After Stripe verified that the credit card is valid and payment is processed, the order go to the store selected. Employees at the store received the order and fulfill the order. The order is hold until the customer come to the store and picks up the order.

3.2 Decomposition Description

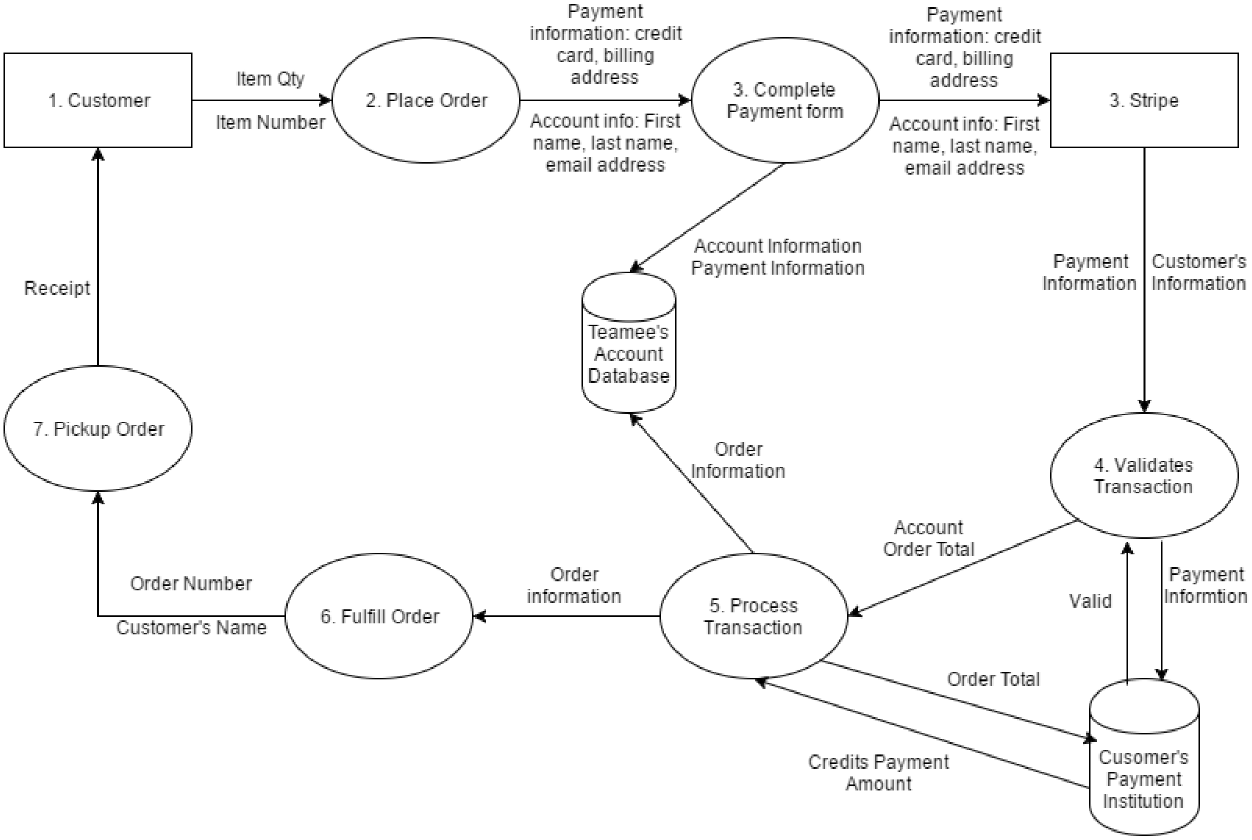


Figure 8: DFD for placing an order

The data flow diagram shows the flow of data when a customer placed an order. This diagram should be used in conjunction with Figure 1. The DFD shows a more detailed view of all the data that occur during an order placement.

3.3 Design Rationale

3.3.1 Trade Study

Quality Attribute	MySQL	MongoDB
Performance	7	7
Ease of Use	6	9
Compatibility	10	2
Supportability	8	5
Scalable	6	9
Reliability	8	5
Total	45	37

Figure 9: A chart of a trade study done to determine which type of DBMS to use.

MySQL and MongoDB were both assigned a 7 for performance since they are about equal because this project is relatively small; thus, it is not expected that there would be huge stores of data. For ease of use, MongoDB was assigned a higher value because it uses dynamic schemas so that it is unnecessary to first define the structure when creating records. Also, one can simply change the structure of records by deleting and adding new fields to the existing ones. MySQL was given a significantly higher rating for compatibility because MySQL is supported by Laravel. In addition, MySQL is better suited as a ecommerce website because it can handle transactions exceptionally well compared to MongoDB. For supportability, MySQL received a higher rating than MongoDB because MySQL is an older and more established DBMS. As a result, it will have more documentations and skilled experts to provide additional assistance. Both DBMS are scalable; however, it is easier to scale using MongoDB. MySQL was given a higher rating for reliability than MongDB because can use ACID which stands for atomicity, consistency, isolation, and durability. These are the four properties of database transactions which are satisfied by MySQL.

3.4 System Context Diagram

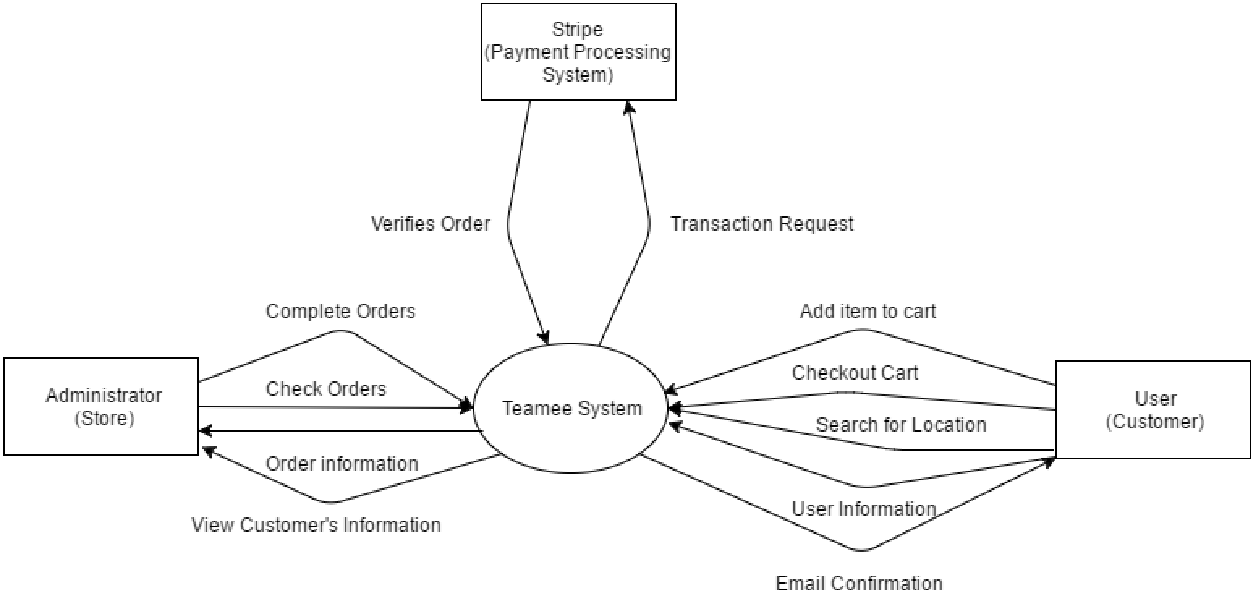


Figure 10 System Context diagram used to understand the interactions of various entities with the Teamee system

4.0 DATA DESIGN

4.1 Data Description

MySQL shall be used as the database.

4.2 Data Dictionary

Data Element	Description	Composition/Data Type	Length
First Name	The user’s first name	String	50
Last Name	The user’s last name	String	50
User’s email address	The user’s email address	String	50
Order number	Auto incremental order number for each order placed by the user	Primary key Integer	10000000
Item Number	Each item sold by Teamee is assigned an item number	Integer	1000
Payment Method		Alphabetic	16

Transaction Number		Integer	12

Figure 11 data dictionary

5.0 COMPONENT DESIGN

5.1 Login Pseudocode

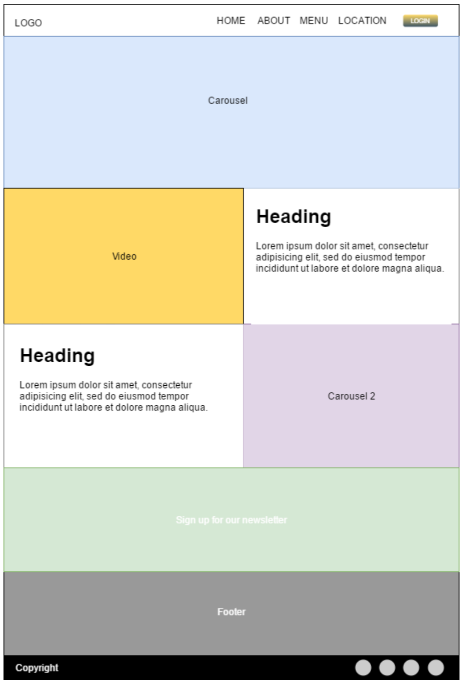
```
header.php
check for cookie
if exist set control panel with options and logout
if none exist check session,
if exist set control panel with options,
if none exist
exit

Ask for login and password with remember option (will set cookie)
user inputs details, database check to see if user exists
if exist register session and setcookie if remember option true
else
show signup so user can register
```

6.0 HUMAN INTERFACE DESIGN

6.1 Overview of User Interface

The user interface for the system will allow the user to easily place and order and view information about Teamee.

6.2 Screen Images



LOGO
HOME
ABOUT
MENU
LOCATION
LOGIN
Carousel
Video
Heading
Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.
Heading
Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.
Carousel 2
Sign up for our newsletter
Footer
Copyright

Figure 12: Mockup of the homepage

7. QUALITY ATTRIBUTES

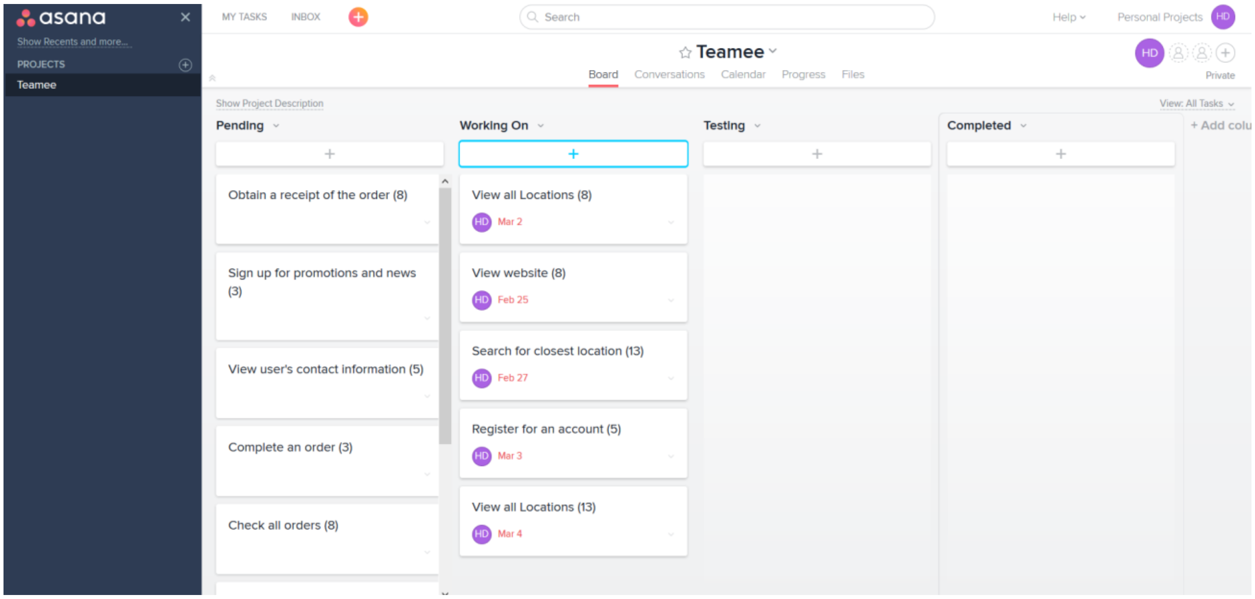
7.1 Quality Attribute Scenarios

7.0 Development

7.1 Iteration 1

7.1.1 Sprint Planning

During the sprint planning for the first iteration, a set of use cases were selected and moved to the "Working On" section. These cases were chosen based upon their high risk and priority. View website was chosen because a basic UI layout was needed in order for the users to use the website. View all locations and Search for the closest location was chosen because they were deemed at a high risk due to my unfamiliarity with using Google map API. Register for an account was also chosen because it would be one of the first function that a new user would utilize. The figure below showed the backlog for Iteration 1 with the five use cases to be implemented.



Pending
- Obtain a receipt of the order (8)
- Sign up for promotions and news (3)
- View user's contact information (5)
- Complete an order (3)
- Check all orders (8)

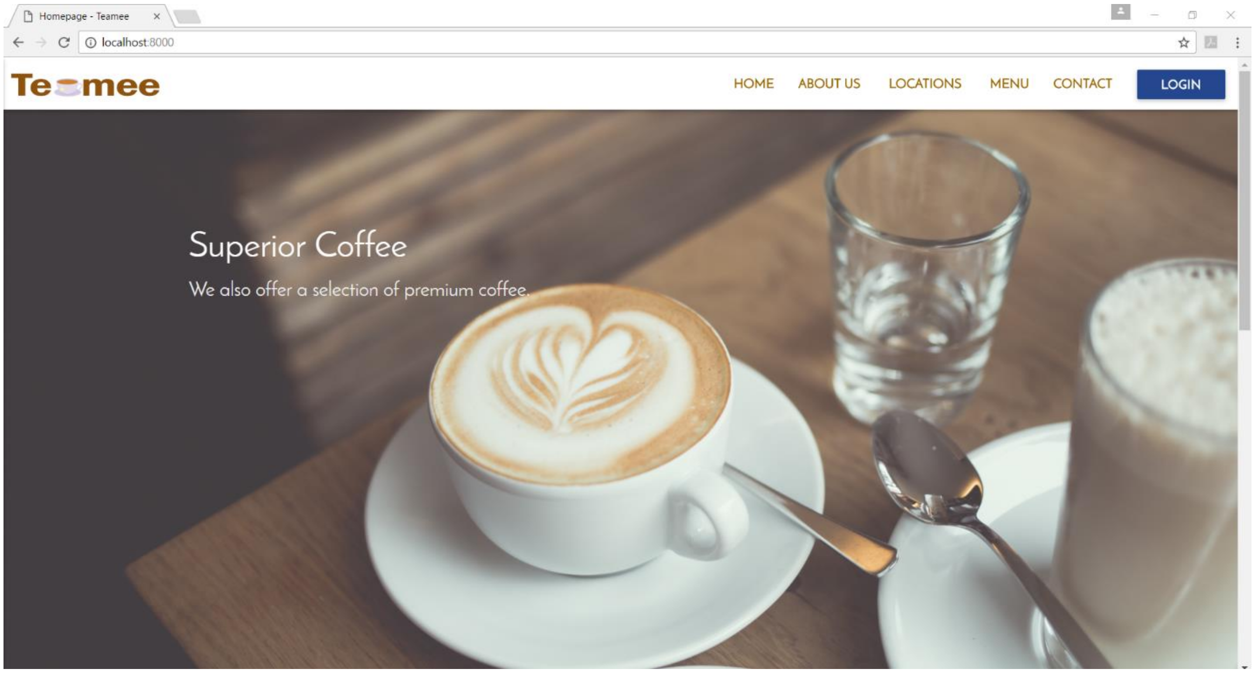
Working On
- View all Locations (8) HD Mar 2
- View website (8) HD Feb 25
- Search for closest location (13) HD Feb 27
- Register for an account (5) HD Mar 3
- View all Locations (13) HD Mar 4

Testing

Completed

Figure 13: Sprint backlog for iteration 1 using Asana board feature

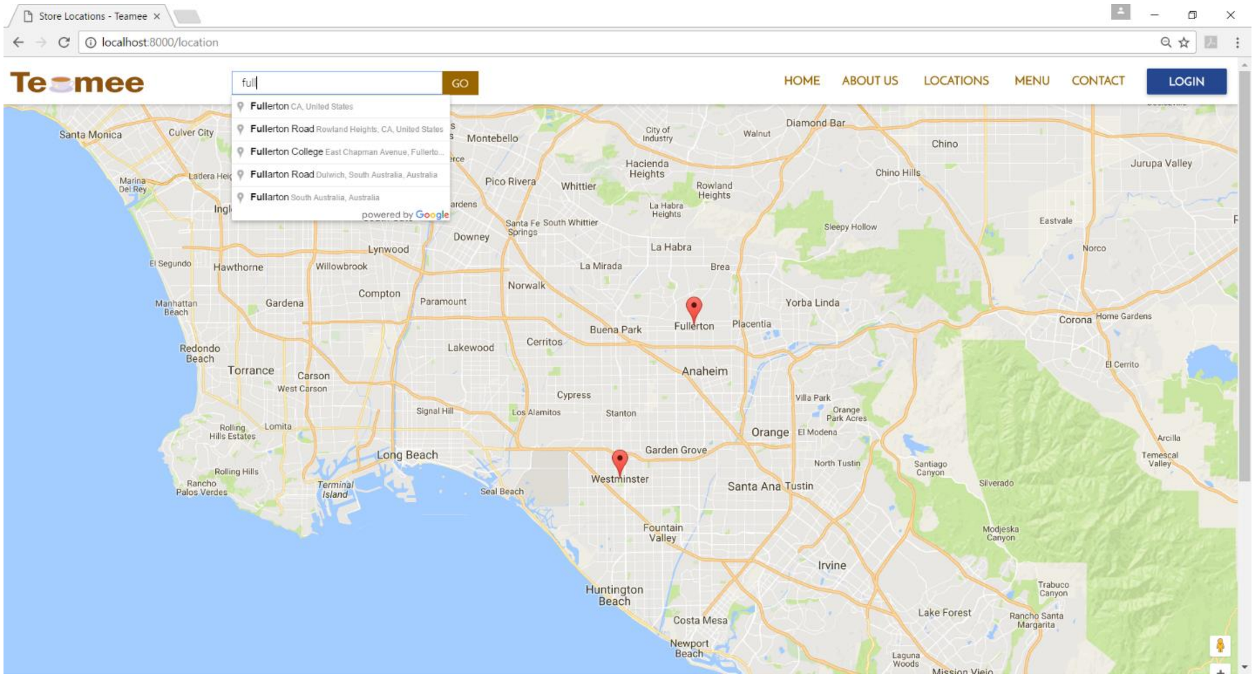
The four figures below showed screenshots of what the website currently looks like after Iteration 1.



Superior Coffee

We also offer a selection of premium coffee.

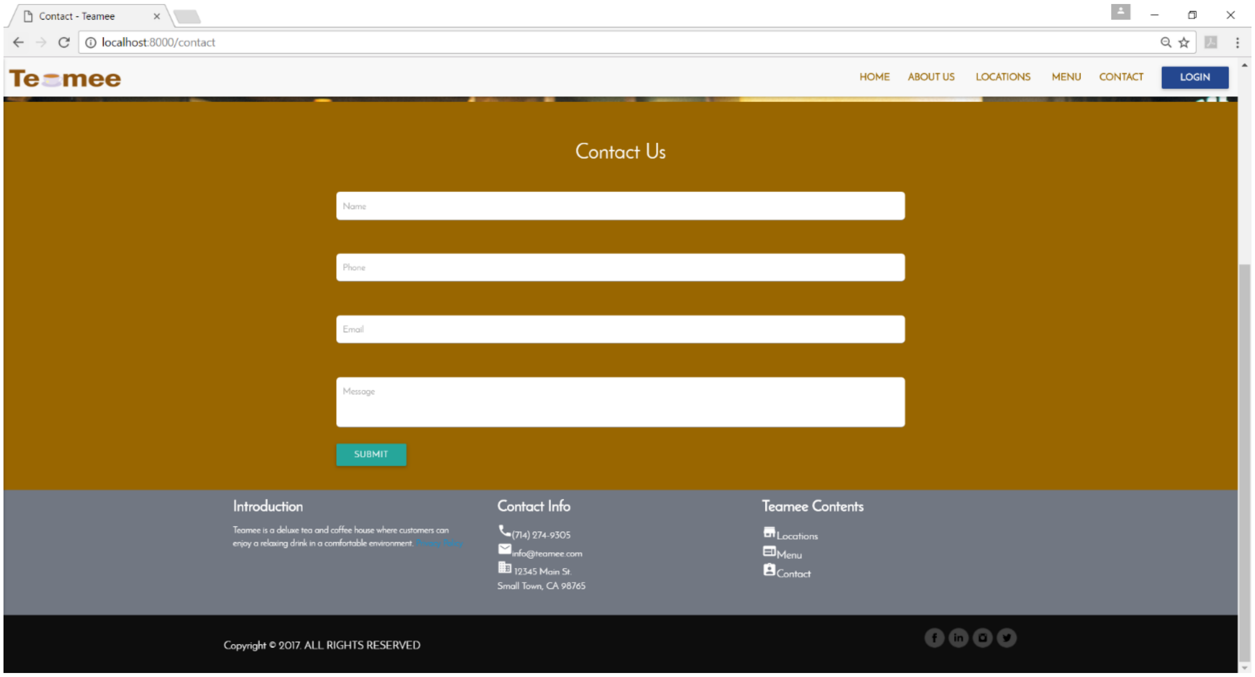
Figure 14 screenshot of the Teamee homepage



full

- Fullerton CA, United States
- Fullerton Road Rowland Heights, CA, United States
- Fullerton College East Chapman Avenue, Fullerto...
- Fullarton Road Dulwich, South Australia, Australia
- Fullarton South Australia, Australia

Figure 15 screenshot of the locations page with the two Teamee stores on Google map



Contact Us

Introduction

Teamee is a deluxe tea and coffee house where customers can enjoy a relaxing drink in a comfortable environment. Privacy Policy

Contact Info

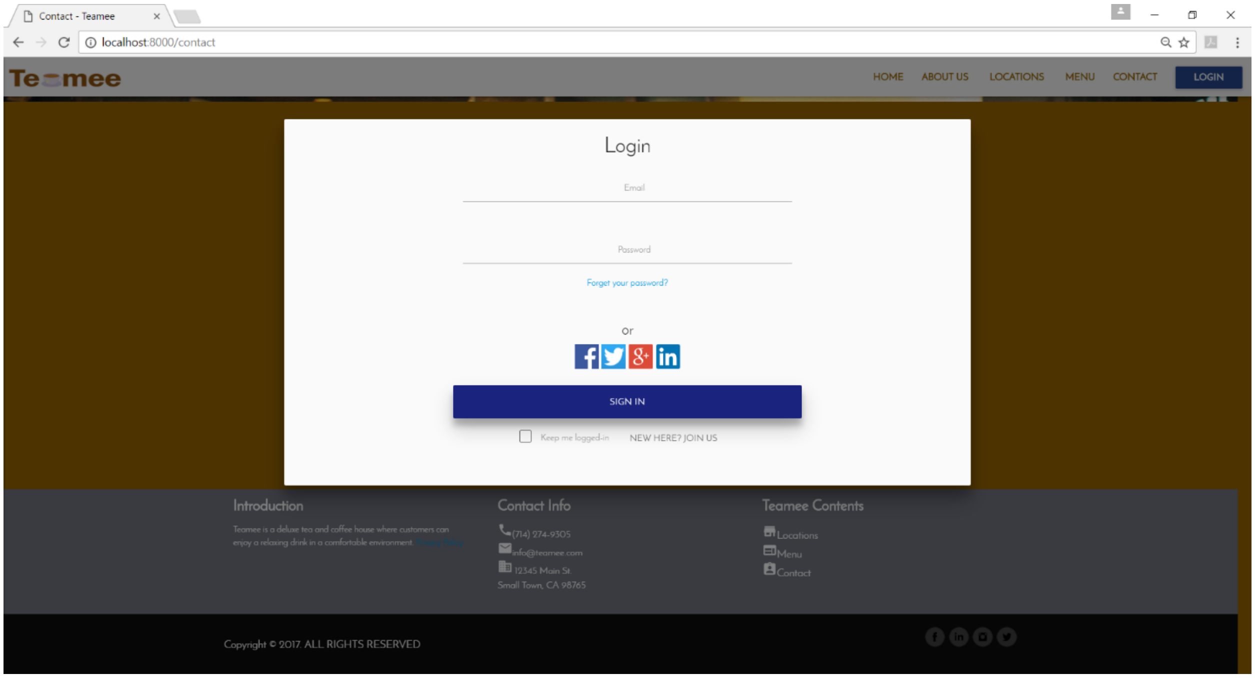
☎(714) 274-9305
info@teamee.com
12345 Main St.
Small Town, CA 98765

Teamee Contents

Locations
Menu
Contact

Copyright © 2017. ALL RIGHTS RESERVED

Figure 16: Screenshot of the contact page



Login

Introduction

Teamee is a deluxe tea and coffee house where customers can enjoy a relaxing drink in a comfortable environment. Privacy Policy

Contact Info

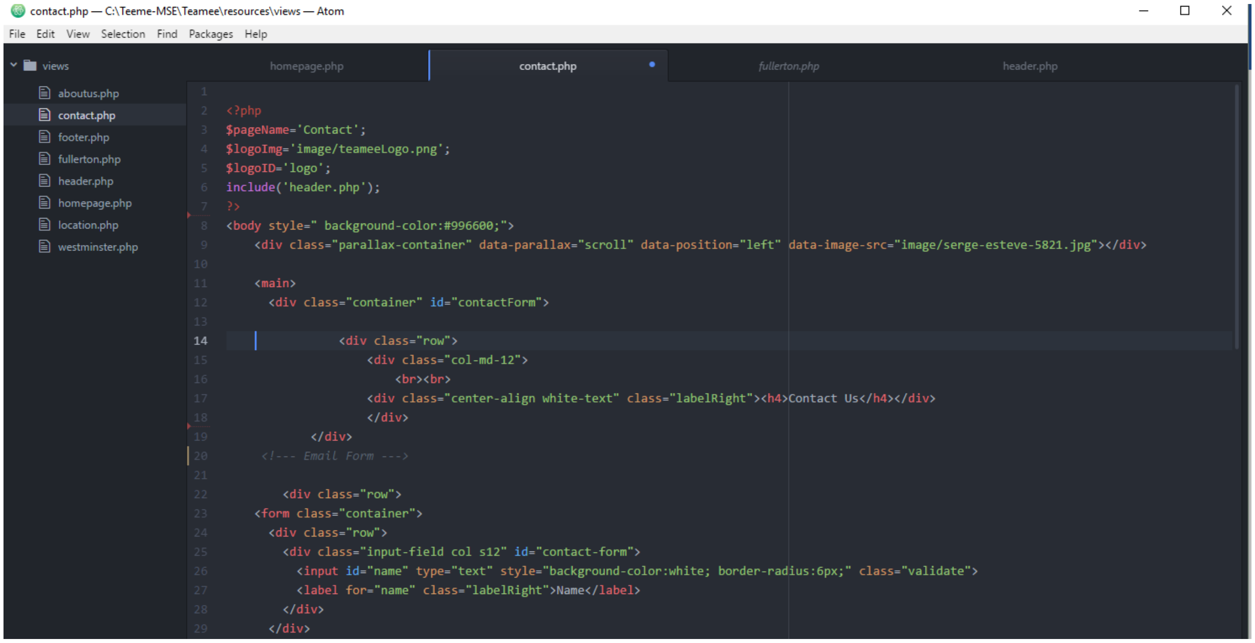
☎(714) 274-9305
info@teamee.com
12345 Main St.
Small Town, CA 98765

Teamee Contents

Locations
Menu
Contact

Copyright © 2017. ALL RIGHTS RESERVED

Figure 17 shows a modal for users to login



```
<?php
$pageName='Contact';
$logoImg='image/teameeLogo.png';
$logoID='logo';
include('header.php');
?>
<body style=" background-color:#996600;">
    <div class="parallax-container" data-parallax="scroll" data-position="left" data-image-src="image/serge-esteve-5821.jpg"></div>

    <main>
      <div class="container" id="contactForm">

                <div class="row">
                    <div class="col-md-12">
                        <br><br>
                    <div class="center-align white-text" class="labelRight"><h4>Contact Us</h4></div>
                    </div>
            </div>
     <!--- Email Form --->

        <div class="row">
    <form class="container">
      <div class="row">
        <div class="input-field col s12" id="contact-form">
          <input id="name" type="text" style="background-color:white; border-radius:6px;" class="validate">
          <label for="name" class="labelRight">Name</label>
        </div>
      </div>
```

Figure 18 contact page code

Figure 18 shows a snippet code sample of the contact page. Atom was used as the code editor. On the left-hand side there's a quick link to the various pages that are within the views folder.

Conclusion

The global and digital world makes it necessary for any business to have a website. Internet access is widely available now which makes it easier for businesses to promote their product online.[9] The convenience of online shopping has made ecommerce a blooming industry. [7] Ecommerce stores are beginning to rival brick and mortar stores as more users are discovering the ease of online shopping.[10] Fast shipping and enticement such as skipping the line, also encourages more users to order online.

An attractive and user friendly website could become a company's face in the global market.[6] A website can contain information about the company and its products to help promote its services. Having an exemplary website is like having an excellent salesman, who can convince customers to buy its products. Thus, it is essential to develop a website that would be appealing and informational.

The Teamee website makes Teamee's products accessible for users almost anywhere. Customers can place their orders online and pickup their drinks instore. A description of each drinks and snacks are also available on the website. The company can post limited time coupons or promotions at any time. Stripe was used as the payment processor so that payment made are secure and validated. Stripe's API supports PHP which is convenient since Laravel is a PHP framework. Laravel supported MySQL which made connection to the database straightforward. The company can collect data obtained from online orders to improve their services and provide better catering to their customers. Additionally, the website allowed users to submit their email address which Teamee can use to send coupons, news, and other information. This feature presents Teamee with a cheap avenue for advertisement.

Teamee's website was created using a PHP framework called Laravel. Laravel allow the developer to use either its blade template or write PHP right into the page. Scrum was used as the developmental process. Scrum was chosen because it advocates incremental releases. The market for cafes is constantly changing with new stores and fads that it is imperative that development can handle changing requirements.[3] Moreover, Scrum is flexible permitting the developer freedom of implementation. Asana was used to host the Sprint Backlog board which assisted in organization and task tracking. In total, there were three sprints that were performed in the development of the website. During each sprint, a sprint planning and sprint review were executed. The sprint planning involved reordering of the Sprint Backlog and the sprint review was used to track the project's progress.

The website is well documented with an SRS, design document, and test plan. The SRS provided a description of the website with all the functional and non-functional requirements. It was updated throughout the whole developmental project. The design document provided the developer information about the architecture of the website. The trade study conducted demonstrated that a relational database was the appropriate choice. The test plan showed that the website has been thoroughly tested and client can be assured of its quality. This will help maintenance personnel understand the code and logic behind the construction of the website. As a result, updating the website will be painless and smoother. Updates to the website will be necessary when new locations are added, promotions, or features.

References

1. Bass, Len, Paul Clements, and Rick Kazman. *Software Architecture in Practice, 2nd ed*. Addison-Wesley Professional, 2003.

2. Cordova. Apache. https://cordova.apache.org/ Accessed January 13, 2017.

3. Daniel Simon Sanz and Ankur Agrawal, "Automated Menu Recommendation System Based on Past Preferences" International Journal of Advanced Computer Science and Applications(IJACSA), 5(7), 2014

4. E-Commerce & Online Shopping Auctions in the US. IBISWorld. June 2016. Online Report. Accessed January 13, 2017.

5. Kaplan, Jeff. Changing Channels: new ways to do business. Network World. March 5, 2007; 24, 9; ABI/INFORM Collection page 25. News article.

6. King, David Lee. Designing the Digital Experience: How to Use Experience Design Tools and Techniques to Build Websites Customers Love. Information Today, Inc. 2008. Print.

7. McPherson, Doug. Worldwide E-commerce Sales to Rise Nearly 20 Percent This Year.

8. Wiegers, Karl, *Software Requirements (3rd Edition*), Microsoft Press 2013.

9. Winter, S.J. Inf Syst E-Bus Manage (2012) 10: 279. doi:10.1007/s10257-011-0165-5.

10. WorldPay. The Global Rise in ‘Always-On’ Shoppers Driving eCommerce. Online report. April 25, 2012.