

# CSC309 A4P2: FastLine Report

## High-level view (Sections of the system)

The app can be broken down into three sections, the customer section, the seller section, and also the admin section.

- \*The customer can order food from sellers, and modify/view their own information.
- \*The seller can receive orders from customers, and modify/view their own information.
- \*The admin can clear the database.

## Description and interaction of sections

Let's describe each section, first, we will start describing the customer part.

### (Customer)

#### Log-in:

The image displays two side-by-side screenshots of the FastLine web application interface. Both screenshots feature a 'Welcome to Fastline!' header with a food truck icon. The left screenshot shows two registration forms: 'Sign up as customer' and 'Sign up as food seller'. The 'Sign up as customer' form includes fields for 'Username(Enter an email address)', 'Your email address', and 'Password(At least 6 characters)', with a 'Sign up!' button. The 'Sign up as food seller' form includes fields for 'Username(Enter an email address)', 'Your email address', 'Password(At least 6 characters)', 'Your Password', 'Food Truck Name', 'Name of the truck', 'Address', 'Hours of Operation', 'Phone', and 'Google Map URL', with a 'Sign up!' button. The right screenshot shows the 'Log-in to your account' form, which includes fields for 'E-mail address' and 'Password', a 'Login' button, a 'Log in with Facebook' button, and a 'New to us? Sign up' link.

You can log-in with Facebook, or register and then log-in with your newly created account.

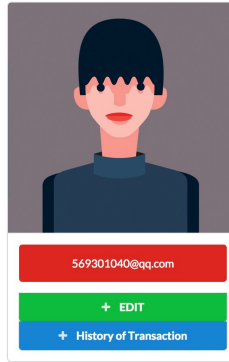
#### HomePage

At home page, you can view your profile, search food/type, and also start ordering food.

The image shows a screenshot of the FastLine homepage. The header is dark blue and contains navigation links 'Home' and 'Profile', a search bar with the text 'Search food or type', a 'Search' button, and a user welcome message 'Welcome! 569301040@qq.com' with a 'Log out' button. The main content area is white and features the FastLine logo (a food truck icon inside a circle), the text 'FASTLINE', the tagline 'Save time, No more line.', and a prominent blue 'Order Now' button with a right arrow.

## Profile

<a href="#">Home</a>	<a href="#">Order Now!</a>	<input type="text" value="Search food or type"/>	<a href="#">Search</a>		Welcome! 569301040@qq.com	<a href="#">Log out</a>
----------------------	----------------------------	--	------------------------	--	---------------------------	-------------------------



In your profile, you can edit your password, and also view your history of order.

## History of order

<a href="#">Home</a>	<a href="#">Profile</a>	<a href="#">Order now!</a>	<input type="text" value="Search food or type"/>	<a href="#">Search</a>		Welcome! 569301040@qq.com	<a href="#">Log out</a>
----------------------	-------------------------	----------------------------	--	------------------------	--	---------------------------	-------------------------

### ORDER# 1

Wed Apr 06 2016 13:31:23 GMT-0400 (EDT)

Order ref 1

Order at White

Your Order	Price	Quantity
beef	\$6	1
Total \$6		

For example, this user ordered once from seller White. The order is showing the date of order, the order number, where he orders, and also the order details.  
(This section will show all the orders history a user has. In this case, one order is recorded.)

## Ordering Process

1) Choose a truck to order from order now page.

In our database now, there's only one truck available to order as shown below, you can add more truck by registering more seller later on.


<a href="#">Home</a>	<a href="#">Profile</a>	<input type="text" value="Search food or type"/>	<a href="#">Search</a>	<a href="#">Welcome! 569301040@qq.com</a>	<a href="#">Log out</a>
----------------------	-------------------------	--	------------------------	---	-------------------------

Choose your truck to order!



2) After you choose this truck, you can view the menu and start ordering food.

<a href="#">Home</a>	<a href="#">Profile</a>	<a href="#">Order now!</a>	<input type="text" value="Search food or type"/>	<a href="#">Search</a>	<a href="#">Welcome! 569301040@qq.com</a>	<a href="#">Log out</a>
----------------------	-------------------------	----------------------------	--	------------------------	---	-------------------------




**White**

Location: 16 idehill cres.  
Hours: 9Am  
Phone: 6479872422  
Type: Chinese

**1**  
POPULARITY

[Reviews](#)

 **Menu**

Quantity

Dish

Price

**beef** **6**

Please Choose a Payment method :)

**Pay by Cash:**

[Place Order](#)

**Pay by Credit Card:**

Billing Information:

In this truck, the menu only has one item, you can add more menu item as a seller. The customer can also make a comment here beside ordering food.

This is the biggest interaction between customer and seller as customer can make comment and order to the seller. The seller can receive comment and orders from customer.

## (Seller)

### Log-in

Seller can only register their account at register page, the Facebook log-in is not available for seller.

### Sign up as food seller

Please provide your information below.

**Username(Enter an email address)**

**Password(At least 6 characters)**

**Food Truck Name**

**Address**

**Hours of Operation**

**Phone**


**Google Map URL**

**Cuisine Style**

As a seller, you need to provide these information in order to create an account.

## Home

[Home](#) [Your Page](#)  [Search](#) Welcome! chrislu0126@hotmail.com [Log out](#)



[chrislu0126@hotmail.com](#)  
[+ Edit Profile/Menu](#)  
[+ History of Transaction](#)

**ORDER1**  
Wed Apr 06 2016 13:31:23 GMT-0400 (EDT)  
Order ref 1

Your Order	Price	Quantity
beef	\$6	1
Total \$6		

This web is deigned by Chris, Lei, Josh and Jason.

The seller home page is different from customer's. The seller can view their profile and could direct to edit their page under the name card. The seller can also view the order directly from the right side of page. The latest order will show up on top. For example, if we have three orders now, the third order will show up on top as it's the latest order.

## Edit profile/menu

### Edit Your Profile

chrislu0126@hotmail.com

**Account Info**  
[Edit](#)

**Truck Info**  
[Edit](#)

Food Truck Name

White

Address

16 idehill cres.

Hours of Operation

9Am

Phone

6479872422

Google Map URL

http://google.com

Type of food

**Menu**  
[Edit](#)

[Add a dish](#)

**Current Menu:**

beef

6


detele

[Save](#)

The seller can edit their information like name, address. And for sure, they can add/delete the menu item.(food name and price.)

## SellerPage

[Home](#) [Edit your Profile](#)  [Search](#) Welcome! chrislu0126@hotmail.com [Log out](#)



**White**  
Location: 16 idehill cres.  
Hours: 9Am  
Phone: 6479872422  
Type: Chinese

1  
POPULARITY

**Your Reviews** [Make a comment of yourself](#) [Address](#)

**White** Wed Apr 06 2016 14:26:07 GMT-0400 (EDT)  
I am a food truck

**White** Wed Apr 06 2016 14:37:02 GMT-0400 (EDT)  
hi

**Your Menu**

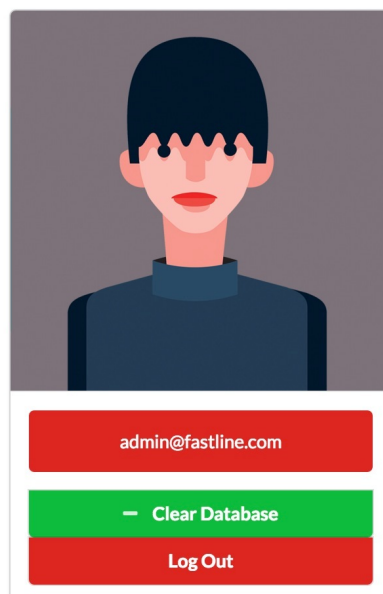
[Quantity](#) [Dish](#) [Price](#)

beef 6

On the seller page, the seller can add a comment to itself and also view the menu and its popularity, which is the total number of order.

## (Admin)

Admin of this app can clear the database. Just log-in with [admin@fastline.com](mailto:admin@fastline.com) and the pwd is adminfastline



## Common usage-Search

In the menu bar, we implemented the search function.

Basically, you can search two things. The dish name and the cuisine style.

If you type a dish name, and search, the result will show any food truck that contain the dish ranked by the popularity.

If you type a cuisine style word, e.g. Chinese, Japanese, the result will show any truck matched this style ranked by the popularity as well.

<a href="#">Home</a>	<a href="#">Profile</a>	<input type="text" value="Search food or type"/>	<a href="#">Search</a>		Welcome! 569301040@qq.com	<a href="#">Log out</a>
----------------------	-------------------------	--	------------------------	--	---------------------------	-------------------------

Search result



White

For example, if I just search Chinese in the box, this is the result since White is of cuisine style Chinese.

## Security vulnerability

For the security section, we did not have a real security framework to secure the app. However, we pay attention to the user data security. The password of user is saved in the database with the encryption of node module Bcrypt. The password is generated into hash.

Second of all, user log-in is required to continue viewing the page. If a user log out in the middle, by typing the url, he/she will not see the app content before he/she log-in again. The makes sure the protection of app data.

As our app is quite big and we spent a lot of time to implement what we expected before, we are unable to look at the security section a bit more deeper. However, we all know the importance of this section and we will continue building our app after the deadline.

## Testing

We have test each function/section one by one. We spent a lot of time testing what we want from the app response. Although we did not use mocha, but we use the node logs, as well as the google chrome check function to debug our app.

For example, we spent a lot of time testing the database queries. Also, the user authentication/modification is a big part to test as well, we need to test whether a user a created properly with fields filled, and also checking if the user's info is updated.

## Performance

We did try using Jmeter to test the performance.

However, we didn't spend much time on this as we are familiar with it and we need time to finish the function of our app.

Thread Group.jmx (/Users/chris/Downloads/apache-jmeter-2.13/bin/Thread Group.jmx) - Apache JMeter (2.13 r1665067)

Sample test  
Thread Group  
HTTP Request Defaults  
HTTP Cookie Manager  
HTTP Request  
View Results in Table  
工作台

View Results in Table

Name: View Results in Table

Comments:

Write results to file / Read from file

Filename:  Browse... Log/Display Only: ☐ Errors ☐ Suc

Sample #	Start Time	Thread Name	Label	Sample Time(ms)	Status	Bytes	Late
1	01:10:34.749	Thread Group ...	HTTP Request	2	500	933	
2	01:10:34.749	Thread Group ...	HTTP Request	2	500	933	
3	01:10:34.807	Thread Group ...	HTTP Request	0	500	933	
4	01:10:35.009	Thread Group ...	HTTP Request	0	500	933	
5	01:10:35.213	Thread Group ...	HTTP Request	0	500	933	
6	01:10:35.412	Thread Group ...	HTTP Request	0	500	933	
7	01:10:35.616	Thread Group ...	HTTP Request	0	500	933	
8	01:10:35.818	Thread Group ...	HTTP Request	0	500	933	
9	01:10:36.022	Thread Group ...	HTTP Request	0	500	933	
10	01:10:36.223	Thread Group ...	HTTP Request	0	500	933	
11	01:10:36.427	Thread Group ...	HTTP Request	0	500	933	
12	01:10:36.628	Thread Group ...	HTTP Request	0	500	933	
13	01:10:36.831	Thread Group ...	HTTP Request	0	500	933	
14	01:10:37.037	Thread Group ...	HTTP Request	0	500	933	
15	01:10:37.239	Thread Group ...	HTTP Request	0	500	933	
16	01:10:37.440	Thread Group ...	HTTP Request	0	500	933	
17	01:10:37.641	Thread Group ...	HTTP Request	0	500	933	
18	01:10:37.846	Thread Group ...	HTTP Request	0	500	933	
19	01:10:38.044	Thread Group ...	HTTP Request	0	500	933	
20	01:10:38.247	Thread Group ...	HTTP Request	0	500	933	
21	01:10:38.459	Thread Group ...	HTTP Request	0	500	933	
22	01:10:38.650	Thread Group ...	HTTP Request	0	500	933	
23	01:10:38.851	Thread Group ...	HTTP Request	0	500	933	
24	01:10:39.054	Thread Group ...	HTTP Request	0	500	933	



Speaking of the performance of our app, it's quite stable as we did a lot of debug and testing on it.

One thing to mention is that we link the ui semantics file on top of each file, this might cause a slower loading speed than actually putting the css/js file in our repo.

The only concern is we cannot put it up to Heroku as some error happened. We will look into it later.

### **Demo Link and Git repo**

<https://www.dropbox.com/s/bsqlf4wvw8wii1q/fastline%20demo.mov?dl=0>

<https://github.com/imnxllet/FastLine>

### **Function Comparison with Expectation (the phase 1 report)**

#### **User authentication:**

We implemented this function perfectly. As user is required to log-in using a username and password.

#### **User profiling:**

We implemented this function perfectly. As a seller needs to provide details of their truck.

#### **User Interaction**

This section is well defined above. Users are interacting with each other.

#### **Social network:**

We implemented this. Customers and sellers form a social network within the website. They can socialize with each other using comment conversation. They can also share the food product or the website on other social network.

#### **Rating and commenting:**

Customers can leave a comment on the seller's profile regarding their product/service overall quality, their purchase experience, etc. The overall rating will be reflected on the seller's profile. Sellers are allowed to make a response to the comment.

#### **Admin:**

All users are allowed to modify their own information on the profile. For example, sellers can add/delete products on the menu, or they can change their address, etc.. Admin can also clear the db.

### **Map(Geo-location):**

We didn't quite achieve this section as we did not use google map API in our app. However, seller can include a google map URL of their address in their profile which allows customers to click and check their position.

### **Order List**

The seller can see the latest order in their homepage. The homepage is set to refresh each 5 second so that the list will keep updating automatically. The latest order is shown on top of the list.

### **History of transaction**

Well implemented.

Users are allowed to view their history of transaction. For example, a customer can look back into when and where they buy the food, as well as viewing the quantity and food items they bought. A seller also has such information, and this really helps them in doing financial management.

### **Payment**

We provided two options to pay. However, the card payment method is just for demonstration purpose, no real transaction is happening. Nevertheless, the customer is required to fulfill all fields in the payment form.

### **Searching: (names, types of food,)**

Well implemented.

Users can search food/drinks items by names and types on the navigation bar.