**Java project proposal**

**Laundry Service system**

This is laundry service website called **Arial Cleaners** in the city of Chicago where we offer different kind of laundry services to our customer. This will be a Web based project with different kinds of services offered such as (signature wash/dry/fold, Corporate services etc), also the customer has on option of pick and drop the laundry at the location or there will be a pick up and drop option available for minimal cost.

Our project will mainly have the below features:

**Two login Panels:**

1. **Front End Login**: For member customers and regular customers. Login credentials will be as follows:
2. Username(Email).
3. Password.
4. **Back End Login**: For Admin user and delivery agent. Login credentials will be as follows:
5. Username
6. password

It is a role based project which makes the process efficient yet simple for the users and the admin to use. Below are different kinds of roles with description of the role.

**Front End Roles:**

1. **Customer who is not a member or just a regular User**: The regular customer will not get any member customer discounts.

2. **Customer who is a member**: This customer will get discounts every time he places an order and the discount amount will be automatically reduced from the total amount at the billing page.

**Back End Roles**

1.**Admin user**: He will be main person who will update the site at the backend. ex: adding a new service to the website, update the order status (received, processing, completed and ready for pickup or drop).

2. **Delivery agent:** Once the order is completed and will be ready for pick up, the delivery boy picks up the order and delivered it to the appropriate customer.

**Process flow of the project:**

**Front End Application for customer:**

1. User will sign up by giving the personal information and also opt for a membership (yearly on monthly).

Once the user is signed up, the user can log in by using the username and password.

1. Both the member and the regular customer will see the same page where they can place

their order. This page will give them the below options:

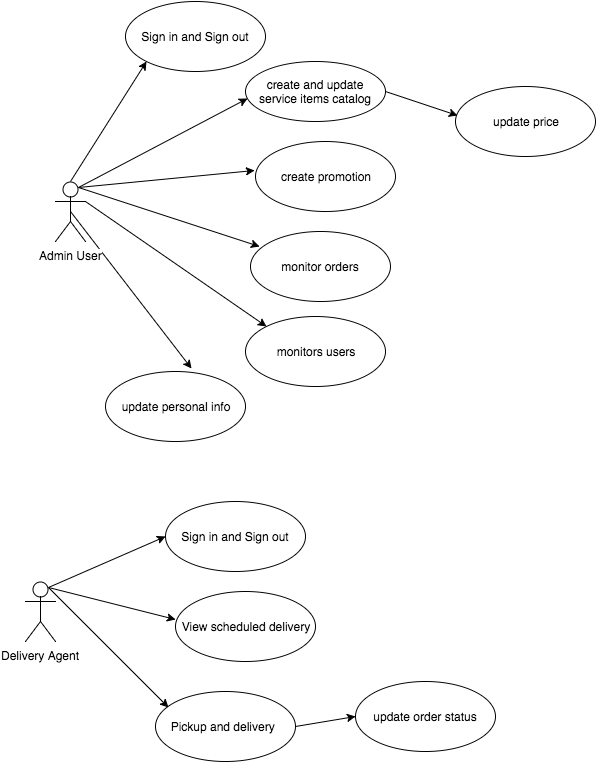
1. Select the service type like dry clean or wash/fold etc (drop down option to select the services will be provided)
2. User will select the service Items such as a shirt, bottom, blazer so on and he will be charged on the service items.
3. Next the user will decide the quantity, depending on the service item and quantity the system will show the price, tax and delivery charges.
4. The customer will then place the order by selecting if he needs the delivery options. Our website will charge $10 for one time pick up and drop.
5. Additionally, a text will be provided to add comments.

**Back End:**

Using the login credentials only admin and delivery agent can login to the back end portal.

1. The admin can carry out the following tasks:
2. Add, delete or update the service type, service items and assign price for each service item.
3. Admin can also add or delete promotions for the members, such as free delivery, additional monetary discounts etc.
4. He can also see the order history on daily, monthly or yearly basis.
5. The delivery agent logs in with his credentials and check the order details such as address, number of service items and pick up schedules. He assigns himself to pick up and drop a particular order and update the order status.

The below two UML use case diagrams which briefly explains the roles of each User.



../Downloads/use_case.png

**Technologies used: -**

**Front end:** HTML, CSS .

**Back end:** JAVA, MySQL, jsp, Servlet.

**CURD Operations:**

The CURD Operations are carried throughout the project to store, retrieve, update and delete the data.

**Inheritance:**

The below UML diagram explains the inheritance in this project. Class User is a superclass which has the common attributes and behaviors for other users in this project such as Admin user, customer and delivery agent. Also the Delivery\_Address class inherits the properties of the Address class.

**UML Diagram**: class_dig.png

**ERD:**

erd_dig.png