Sri Lanka Institute of Information Technology

Project Report: Assignment 01 - Micro-Kernel Architecture
Software Architecture – SE3030

3rd Year – 1st Semester (2019) – Weekend Batch

Submitted to:

Mr. Udara Samarathunga

31st of March, 2019

Abstract

For this project we had chosen a Food ordering system in a restaurant. We have named it as Olive Garden Restaurant. This system was selected as there are some functions which we can implement as plug-in modules.

Moreover, the procedure we have to follow can be easily illustrated by implementing the food ordering system. We have considered two main system users when designed the solution. They are administrator and the customers of the restaurant. Features which are used by those two users were divided into some isolated modules considering the dependencies between the functions.

In order to eliminate the maintenance and the testing, and to cope up with higher performance, microkernel architecture pattern is the best solution to implement this kind of system. This project consists of three main bundles which some of them act as the publishers and some of them act as the consumers.

The source code was created in the Eclipse IDE using OSGi. The whole procedure was designed and the developed by the team members and all the team members contribute to the source codes of all bundles in design and development.

Overall mission of this project is to illustrate the best use of microkernel architecture pattern in software development.

Acknowledgement

We would first like to thank Mr. Udara Samarathunga, the lecturer in charge of the Software Architecture module, for making himself available and taking his valuable time to rectify all our doubts we had regarding the module and the assignment.

We would also like to thank lab instructors for all the guidance and support lent to carry out this assignment. We would not have done this much if it was not for them taking their valuable time to rectify the shortcomings in this assignment.

Declaration

We declare that this project report or any part of it was not copied from any other person's or group's work (published or unpublished), and has not previously submitted for assessment either at Sri Lanka Institute of Information Technology or elsewhere else.

Project Details:

Project Title	Food Ordering system using the microkernel architecture pattern

Group Details:

	Registration Number	Name
1	IT17121170	Ranasinghe N.K
2	IT17092548	Kamburugamuwa K.L.A.P.T
3	IT17120012	T. A. S. Shashiprabha
4	IT17119122	Liyanage I.M

Contents

Introduction

1. Microkernel Architecture

The microkernel architecture pattern mainly consists of two types of components. They are core system and plug-in modules. Application functionalities are divided between independent bundles as known as plug-in modules and the basic core system.

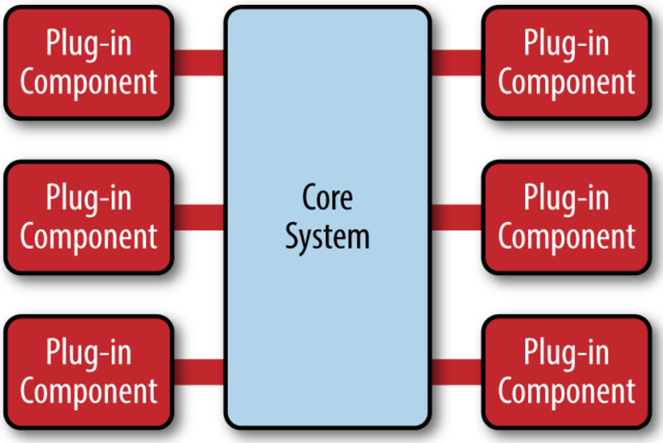


Figure 1- The microkernel architecture pattern

Plug-in modules can be connected to the core system through a variety of ways, including OSGi (open service gateway initiative), messaging, web services. The type of connection depends on the type of application building (small product or large application) and user's and developer's specific needs.

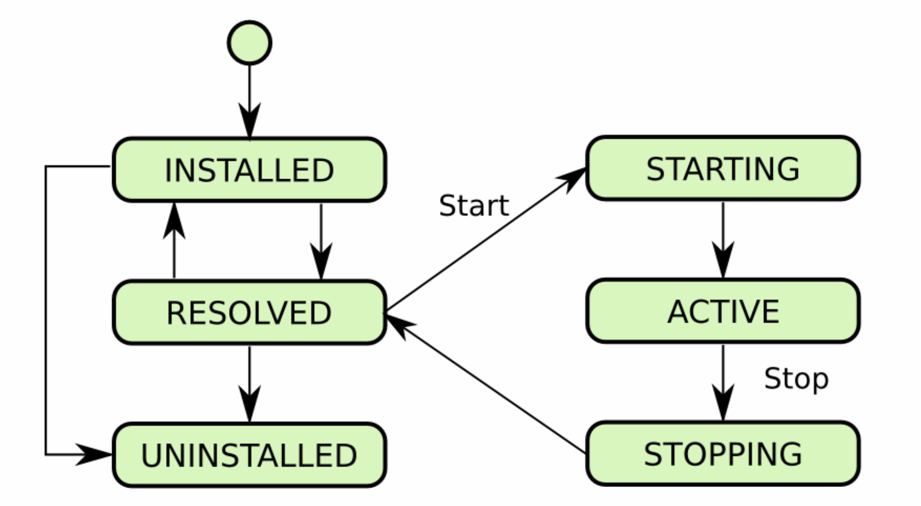


Figure 2 - Life cycle of OSGi bundle

2. Application

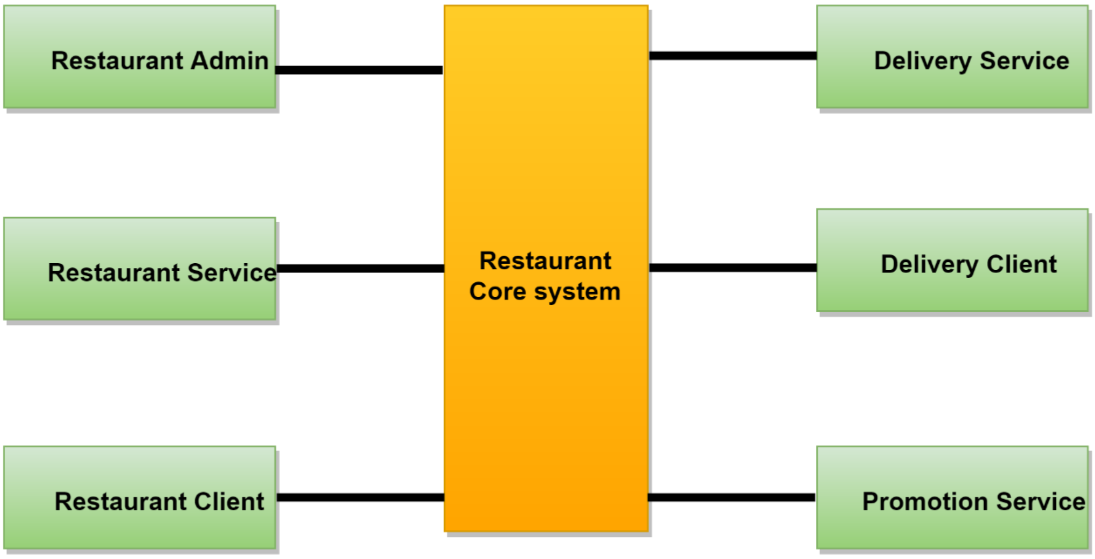


Figure 3 - Architecture of the Restaurant Application

This application is based on the basic functionalities of a restaurant. Microkernel architecture pattern is used to implement the solution. This application enables some basic features for customers and the administrator of the system such as;

Administrator

- Add menu items
- Add Prices
- Deallocate reserved tables

Customer

- Order food
- Choose delivery options
- Book a table
- Display total charges
- Use promotion code

The above-mentioned features/functions are implemented in separated bundles using OSGi for some future beneficial reasons.

3. Key benefits of using microkernel architecture pattern

- Overall agility

Overall agility means the ability of responding quickly to changes. Changes in a system can be easily done by modifying the loosely-coupled modules.

As an example, in the food ordering system, if new feature is needed to add to the system, developers don't need to change the whole source code. They just need to change or modify the relevant source if it is available as a module or else, they just have to add the new feature as a module to the core system.

- Ease of deployment

According to the implementation of the pattern and consideration of the dependencies between the modules, the plug-in modules can be dynamically added to the core system whenever the requirements arrive.

As an example, In the food ordering system, if the system owners don't need any delivery function to cater the customers, they can use the system without using/enabling the delivery module.

- Testability

The plug-in modules can be tested independently. This would be helpful for quality assurance engineers and the system testers to check and test the quality of the product easily.

As in the food ordering system, the available plug-in modules can be tested separately if they have minimum dependencies between them. When one team tests the administrator's functionalities, other team can test the customers' functionality without any doubts and conflicts.

- Performance

When a system is designed and implemented using microkernel architecture pattern, system users can decide what they need in system, they don't need to use and consume the all functionalities in the system. This will help to maintain a higher server performance by without using the unnecessary functionalities.

Methodology

Analysis

We have identified the users who are going to interact with the system. Then the functions which cater different types of users were listed down and explained in a use case diagram.

Use case diagram

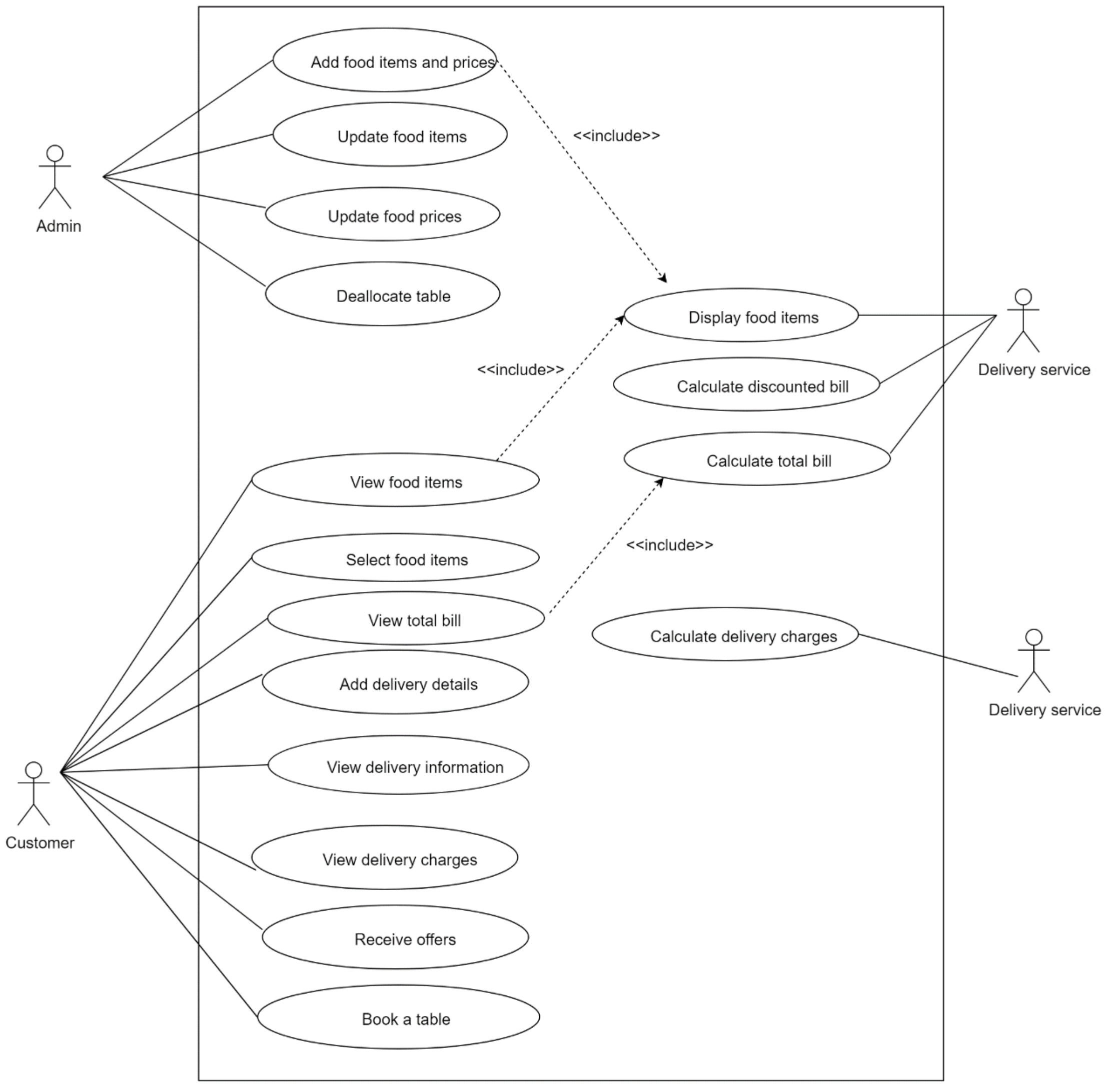


Figure 4 – Use case diagram

Design

There are three publisher bundles and three consumer bundles. The loosely coupled (have minimum dependencies between the bundles) Export bundles of three publishers are illustrated in below table and import bundles of each bundle are illustrated in the figure-5 respectively.

Publisher	Consumer
Restaurant Service	• Restaurant Admin • Restaurant Client
Delivery Service	• Delivery Client
Promotion Service	• Delivery Client • Restaurant Client

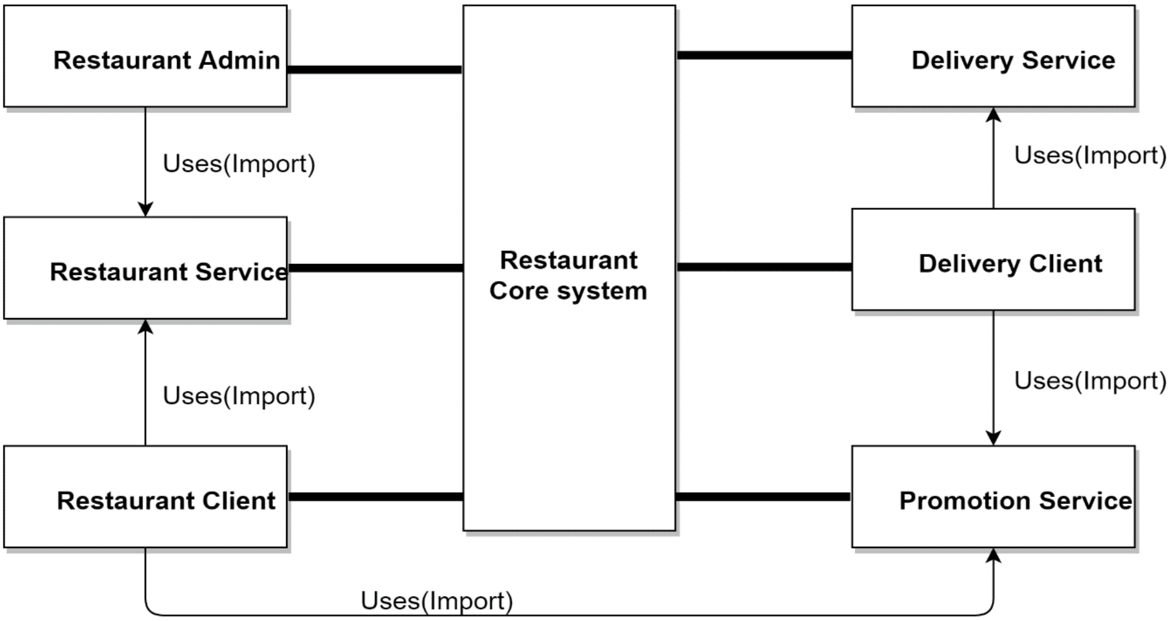


Figure 5 – Overview of the application architecture

Development

In this system we used OSGi, which allow users to distribute the application in a form of bundles. A bundle as known as module consists of Java classes and other resources that provides functions to owners, as well as providing services and packages to other bundles.

Create the bundles

In Eclipse,

1. click on **File --> New --> Project**

In the New Project dialog,

2. selected Plug-in Project and clicked Next

In the Plug-in Project dialog,

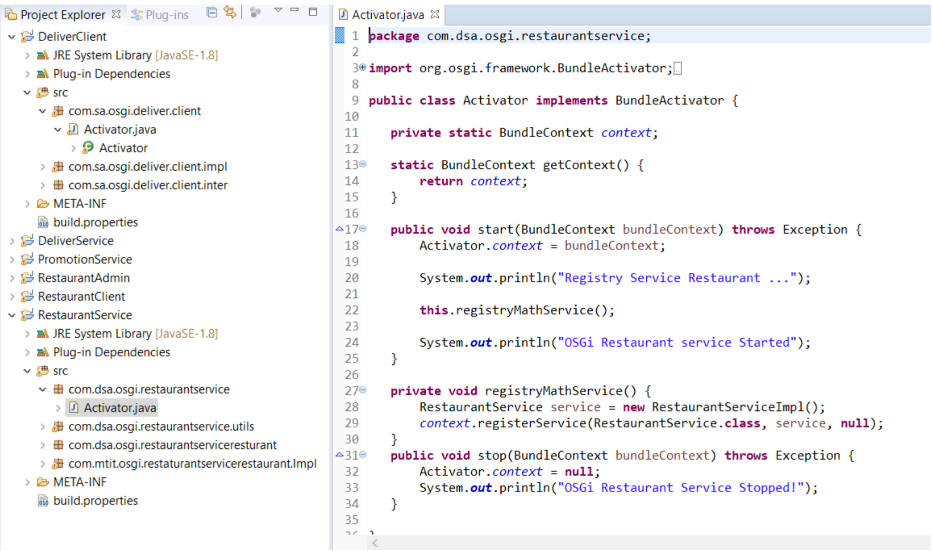
3. Project Name: RestaurantService
4. Target Platform: **OSGi framework --> Standard**

Used default values for the remaining input and click Next

Eclipse create two files:

Activator.java and **MANIFEST.MF**

- **Activator.java**



```
DeliverClient
    JRE System Library [JavaSE-1.8]
    Plug-in Dependencies
    src
        com.sa.osgi.deliver.client
            Activator.java
                Activator
        com.sa.osgi.deliver.client.impl
        com.sa.osgi.deliver.client.inter
    META-INF
    build.properties
DeliverService
PromotionService
RestaurantAdmin
RestaurantClient
RestaurantService
    JRE System Library [JavaSE-1.8]
    Plug-in Dependencies
    src
        com.dsa.osgi.restaurantservice
            Activator.java
        com.dsa.osgi.restaurantservice.utils
        com.dsa.osgi.restaurantserviceresturant
        com.mtit.osgi.restaturantservicerestaurant.Impl
    META-INF
    build.properties
```

```java
package com.dsa.osgi.restaurantservice;

import org.osgi.framework.BundleActivator;

public class Activator implements BundleActivator {

    private static BundleContext context;

    static BundleContext getContext() {
        return context;
    }

    public void start(BundleContext bundleContext) throws Exception {
        Activator.context = bundleContext;

        System.out.println("Registry Service Restaurant ...");

        this.registryMathService();

        System.out.println("OSGi Restaurant service Started");
    }

    private void registryMathService() {
        RestaurantService service = new RestaurantServiceImpl();
        context.registerService(RestaurantService.class, service, null);
    }
    public void stop(BundleContext bundleContext) throws Exception {
        Activator.context = null;
        System.out.println("OSGi Restaurant Service Stopped!");
    }

}
```

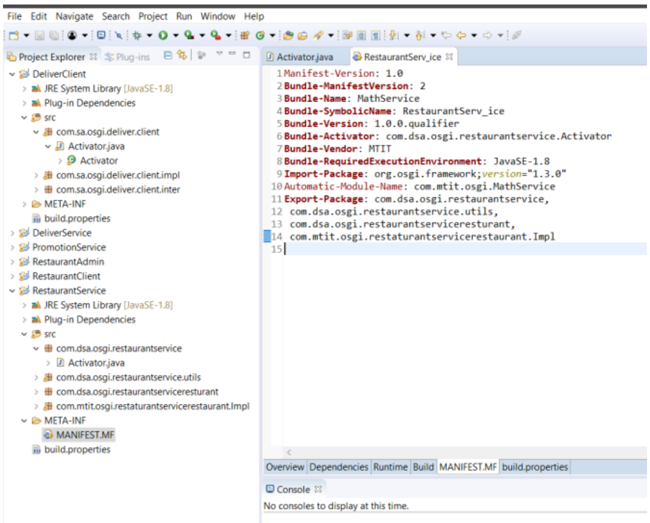
Figure 6 – Activator.java

Notifications which display at the time of the bundle startup or shutdown, were created in this Activator class by implementing the BundleActivator interface.

It consists of two main methods Start and Stop, which are executed at the time when called the start <bundled> or stop < bundled> in the console.

- **MANIFEST.MF**

The MANIFEST.MF file is the deployment descriptor for the bundle. This file contains the details of export and import packages of a particular bundle. And the name, version, manifest version, activator, required execution environment are also mentioned in the MANIFEST.MF file.



```
Manifest-Version: 1.0
Bundle-ManifestVersion: 2
Bundle-Name: MathService
Bundle-SymbolicName: RestaurantServ_ice
Bundle-Version: 1.0.0.qualifier
Bundle-Activator: com.dsa.osgi.restaurantservice.Activator
Bundle-Vendor: MTIT
Bundle-RequiredExecutionEnvironment: JavaSE-1.8
Import-Package: org.osgi.framework;version="1.3.0"
Automatic-Module-Name: com.mtit.osgi.MathService
Export-Package: com.dsa.osgi.restaurantservice,
 com.dsa.osgi.restaurantservice.utils,
 com.dsa.osgi.restaurantserviceresturant,
 com.mtit.osgi.restaturantservicerestaurant.Impl
```

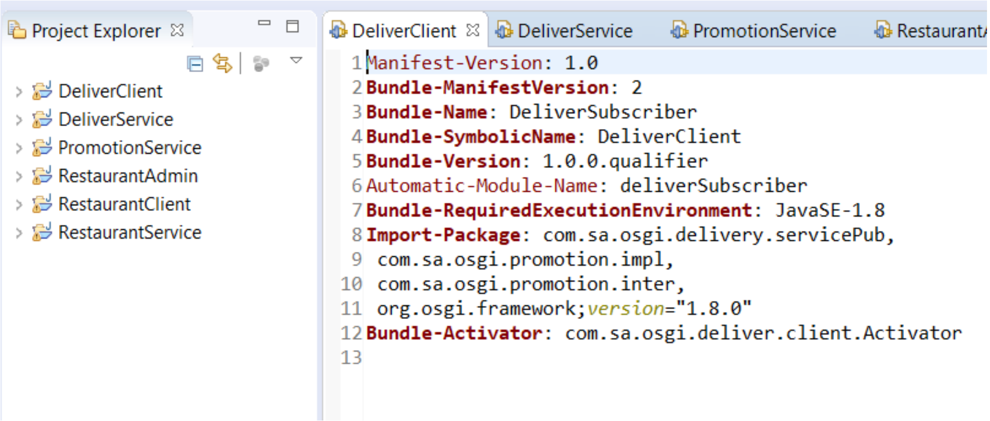
No consoles to display at this time.

Figure 7 – MANIFEST.MF File

✓ Above-mentioned steps were followed when creating all the six bundles with the relevant project names.

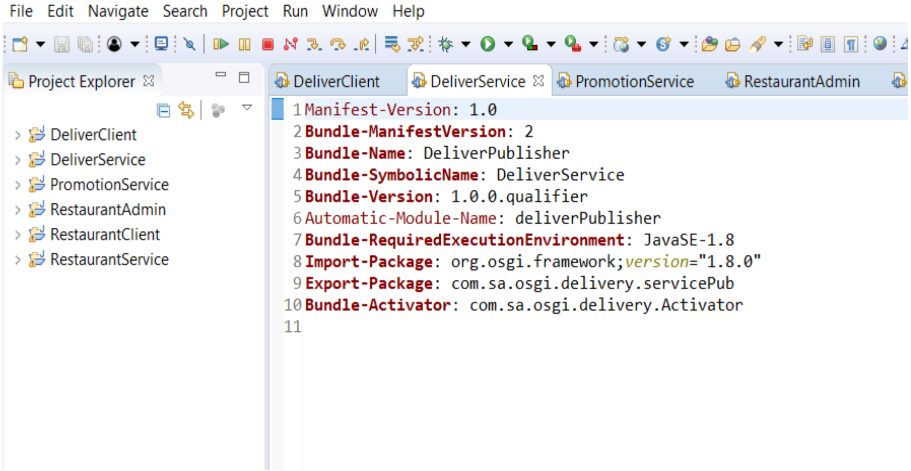
Manifest Implementation

Deliver Client (Consumer)



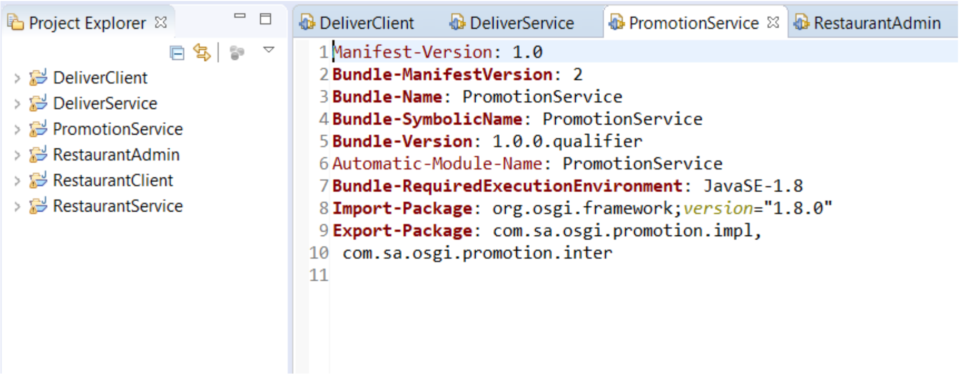
```
Manifest-Version: 1.0
Bundle-ManifestVersion: 2
Bundle-Name: DeliverSubscriber
Bundle-SymbolicName: DeliverClient
Bundle-Version: 1.0.0.qualifier
Automatic-Module-Name: deliverSubscriber
Bundle-RequiredExecutionEnvironment: JavaSE-1.8
Import-Package: com.sa.osgi.delivery.servicePub,
 com.sa.osgi.promotion.impl,
 com.sa.osgi.promotion.inter,
 org.osgi.framework;version="1.8.0"
Bundle-Activator: com.sa.osgi.deliver.client.Activator
```

Deliver Producer (Publisher)



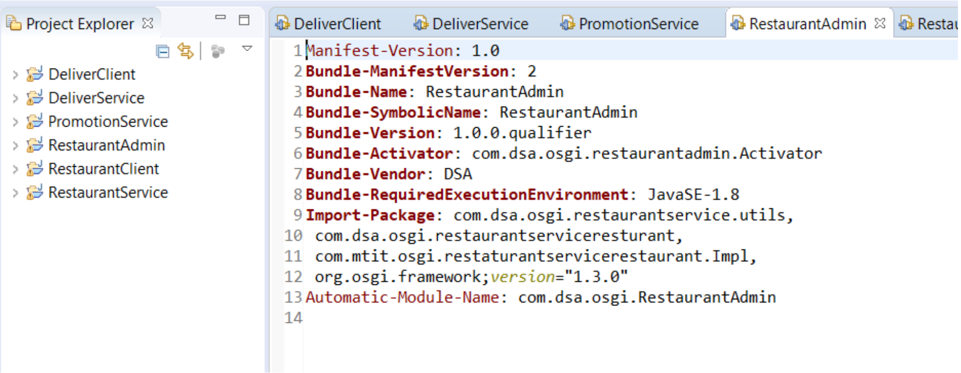
```
Manifest-Version: 1.0
Bundle-ManifestVersion: 2
Bundle-Name: DeliverPublisher
Bundle-SymbolicName: DeliverService
Bundle-Version: 1.0.0.qualifier
Automatic-Module-Name: deliverPublisher
Bundle-RequiredExecutionEnvironment: JavaSE-1.8
Import-Package: org.osgi.framework;version="1.8.0"
Export-Package: com.sa.osgi.delivery.servicePub
Bundle-Activator: com.sa.osgi.delivery.Activator

```

Promotion Producer (Publisher)



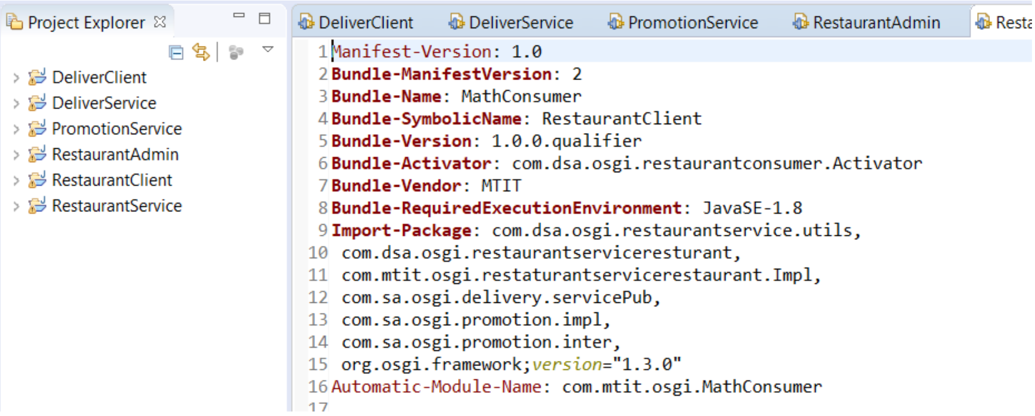
```
Manifest-Version: 1.0
Bundle-ManifestVersion: 2
Bundle-Name: PromotionService
Bundle-SymbolicName: PromotionService
Bundle-Version: 1.0.0.qualifier
Automatic-Module-Name: PromotionService
Bundle-RequiredExecutionEnvironment: JavaSE-1.8
Import-Package: org.osgi.framework;version="1.8.0"
Export-Package: com.sa.osgi.promotion.impl,
 com.sa.osgi.promotion.inter

```

Restaurant Admin (Consumer)



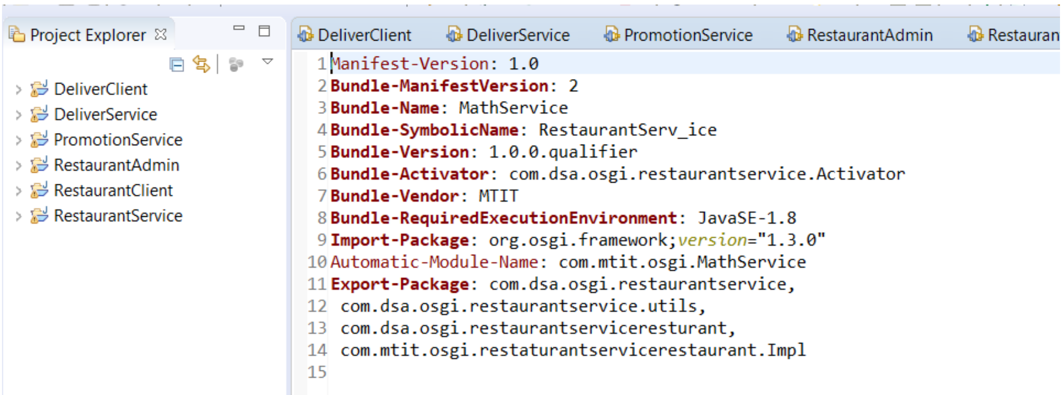
```
Manifest-Version: 1.0
Bundle-ManifestVersion: 2
Bundle-Name: RestaurantAdmin
Bundle-SymbolicName: RestaurantAdmin
Bundle-Version: 1.0.0.qualifier
Bundle-Activator: com.dsa.osgi.restaurantadmin.Activator
Bundle-Vendor: DSA
Bundle-RequiredExecutionEnvironment: JavaSE-1.8
Import-Package: com.dsa.osgi.restaurantservice.utils,
 com.dsa.osgi.restaurantserviceresturant,
 com.mtit.osgi.restaturantservicerestaurant.Impl,
 org.osgi.framework;version="1.3.0"
Automatic-Module-Name: com.dsa.osgi.RestaurantAdmin

```

Restaurant Client (Consumer)



```
Manifest-Version: 1.0
Bundle-ManifestVersion: 2
Bundle-Name: MathConsumer
Bundle-SymbolicName: RestaurantClient
Bundle-Version: 1.0.0.qualifier
Bundle-Activator: com.dsa.osgi.restaurantconsumer.Activator
Bundle-Vendor: MTIT
Bundle-RequiredExecutionEnvironment: JavaSE-1.8
Import-Package: com.dsa.osgi.restaurantservice.utils,
 com.dsa.osgi.restaurantserviceresturant,
 com.mtit.osgi.restaturantservicerestaurant.Impl,
 com.sa.osgi.delivery.servicePub,
 com.sa.osgi.promotion.impl,
 com.sa.osgi.promotion.inter,
 org.osgi.framework;version="1.3.0"
Automatic-Module-Name: com.mtit.osgi.MathConsumer
```

Restaurant Service (Publisher)



```
Manifest-Version: 1.0
Bundle-ManifestVersion: 2
Bundle-Name: MathService
Bundle-SymbolicName: RestaurantServ_ice
Bundle-Version: 1.0.0.qualifier
Bundle-Activator: com.dsa.osgi.restaurantservice.Activator
Bundle-Vendor: MTIT
Bundle-RequiredExecutionEnvironment: JavaSE-1.8
Import-Package: org.osgi.framework;version="1.3.0"
Automatic-Module-Name: com.mtit.osgi.MathService
Export-Package: com.dsa.osgi.restaurantservice,
 com.dsa.osgi.restaurantservice.utils,
 com.dsa.osgi.restaurantserviceresturant,
 com.mtit.osgi.restaturantservicerestaurant.Impl
```

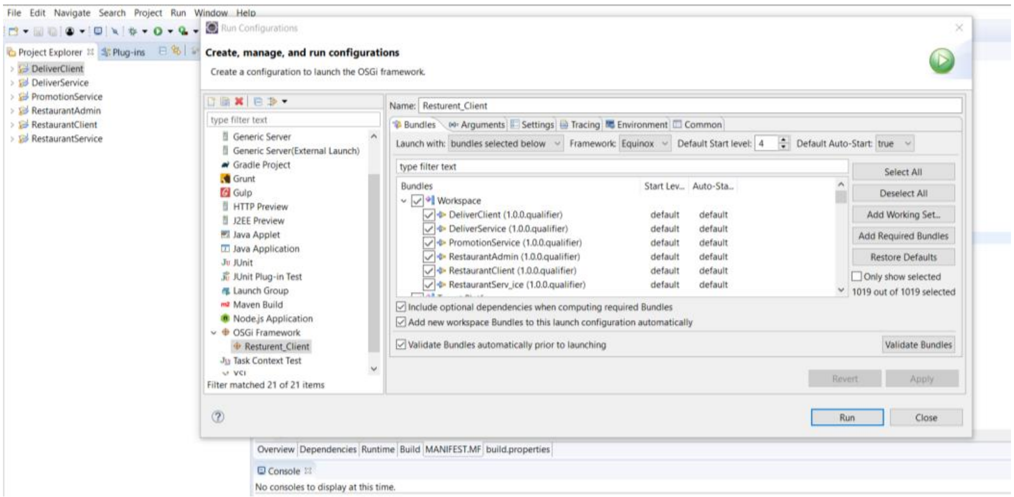
Execute the bundles

Click on **Run --> Run.**

The dialog box, "Create, manage and run configuration." Will be opened.

In that dialog,

1. double-click the Equinox OSGi Framework button
2. It will open a configuration dialog box as in the figure
3. Add required bundles and click on Run



Create, manage, and run configurations

Create a configuration to launch the OSGi framework.

Name: Resturent_Client

Launch with: bundles selected below Framework: Equinox Default Start level: 4 Default Auto-Start: true

Bundles	Start Lev...	Auto-Sta...
☑ Workspace		
☑ DeliverClient (1.0.0.qualifier)	default	default
☑ DeliverService (1.0.0.qualifier)	default	default
☑ PromotionService (1.0.0.qualifier)	default	default
☑ RestaurantAdmin (1.0.0.qualifier)	default	default
☑ RestaurantClient (1.0.0.qualifier)	default	default
☑ RestaurantServ_ice (1.0.0.qualifier)	default	default

□ Only show selected

1019 out of 1019 selected

☑ Include optional dependencies when computing required Bundles

☑ Add new workspace Bundles to this launch configuration automatically

☑ Validate Bundles automatically prior to launching

Filter matched 21 of 21 items

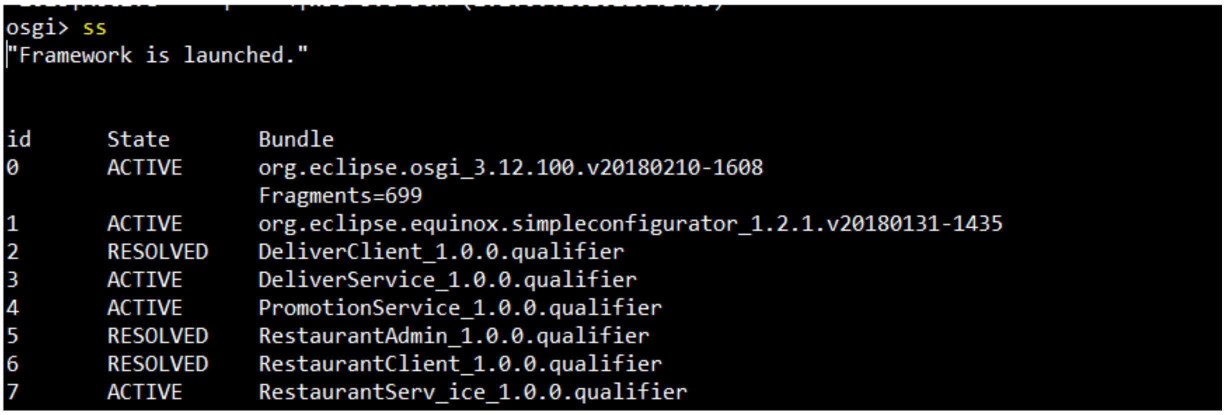
No consoles to display at this time.

Figure 8 – Create configurations

- OSGi console

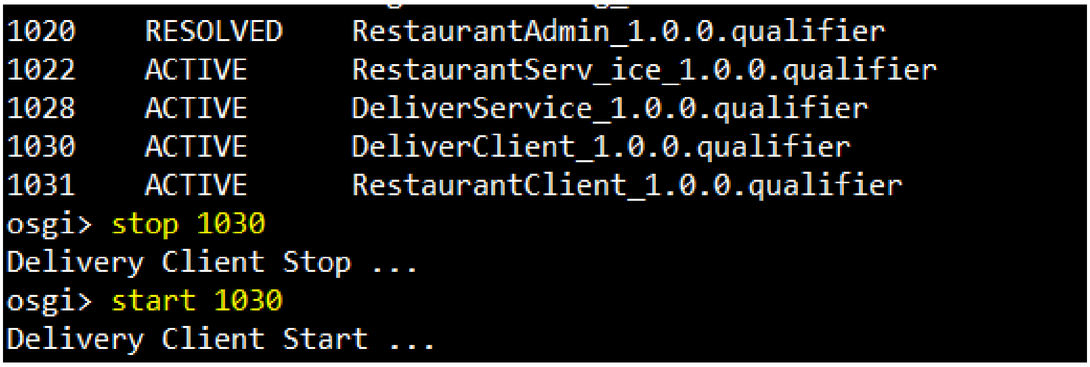
  OSGi Console commands

  - ss - shows a list of installed bundles with the status of each bundle, bundle id, and the short name of the bundle.
  - start <bundle_id> starts a bundle
  - stop <bundle_id> stops a bundle



```
osgi> ss
"Framework is launched."


id      State       Bundle
0       ACTIVE      org.eclipse.osgi_3.12.100.v20180210-1608
                    Fragments=699
1       ACTIVE      org.eclipse.equinox.simpleconfigurator_1.2.1.v20180131-1435
2       RESOLVED    DeliverClient_1.0.0.qualifier
3       ACTIVE      DeliverService_1.0.0.qualifier
4       ACTIVE      PromotionService_1.0.0.qualifier
5       RESOLVED    RestaurantAdmin_1.0.0.qualifier
6       RESOLVED    RestaurantClient_1.0.0.qualifier
7       ACTIVE      RestaurantServ_ice_1.0.0.qualifier
```

Figure 9 – Run ss command in OSGi console



```
1020    RESOLVED    RestaurantAdmin_1.0.0.qualifier
1022    ACTIVE      RestaurantServ_ice_1.0.0.qualifier
1028    ACTIVE      DeliverService_1.0.0.qualifier
1030    ACTIVE      DeliverClient_1.0.0.qualifier
1031    ACTIVE      RestaurantClient_1.0.0.qualifier
osgi> stop 1030
Delivery Client Stop ...
osgi> start 1030
Delivery Client Start ...
```

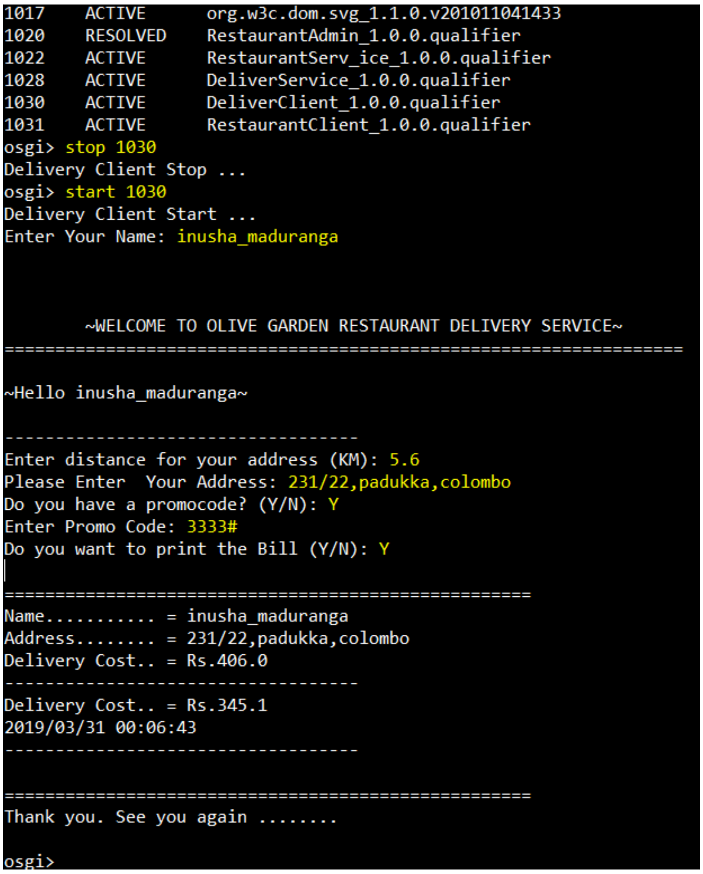
Figure 10 – Run start and stop commands in OSGi console

Execution

- **Delivery**

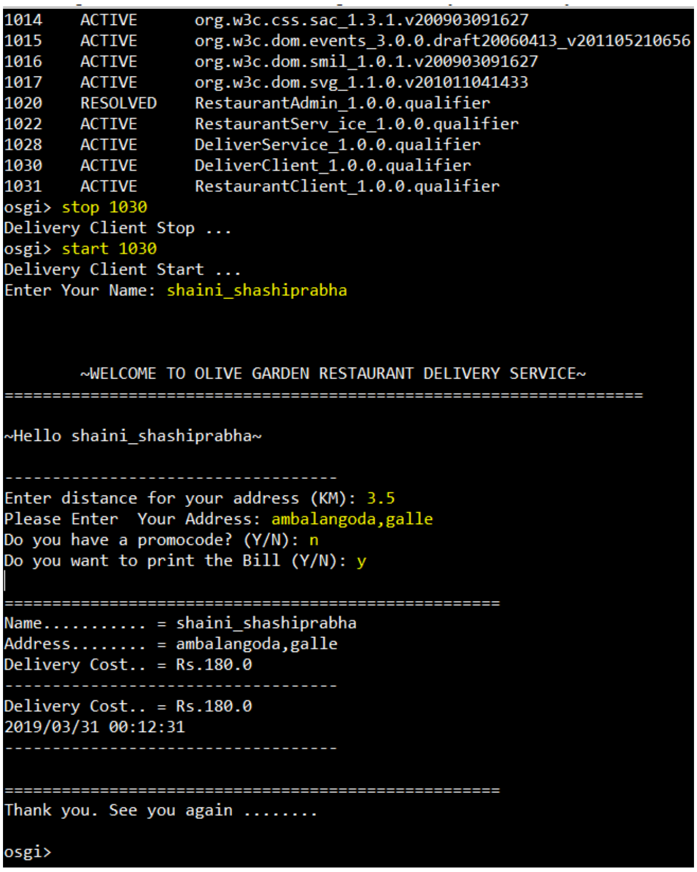
After starting the delivery client bundle, application prompts a welcome message and requesting the following details from the client.

Distance for the customer's address, Address, check whether the customer is received a promo code and the request for the bill. Figure 11 and 12 illustrate the interfaces when customer is using a promo code and when customer is not using a promo code respectively.



```
1017    ACTIVE      org.w3c.dom.svg_1.1.0.v201011041433
1020    RESOLVED    RestaurantAdmin_1.0.0.qualifier
1022    ACTIVE      RestaurantServ_ice_1.0.0.qualifier
1028    ACTIVE      DeliverService_1.0.0.qualifier
1030    ACTIVE      DeliverClient_1.0.0.qualifier
1031    ACTIVE      RestaurantClient_1.0.0.qualifier
osgi> stop 1030
Delivery Client Stop ...
osgi> start 1030
Delivery Client Start ...
Enter Your Name: inusha_maduranga



        ~WELCOME TO OLIVE GARDEN RESTAURANT DELIVERY SERVICE~
====================================================================

~Hello inusha_maduranga~

-----------------------------------
Enter distance for your address (KM): 5.6
Please Enter  Your Address: 231/22,padukka,colombo
Do you have a promocode? (Y/N): Y
Enter Promo Code: 3333#
Do you want to print the Bill (Y/N): Y

=====================================================
Name........... = inusha_maduranga
Address........ = 231/22,padukka,colombo
Delivery Cost.. = Rs.406.0
-----------------------------------
Delivery Cost.. = Rs.345.1
2019/03/31 00:06:43
-----------------------------------

=====================================================
Thank you. See you again ........

osgi>
```

Figure 11 – With a promo code



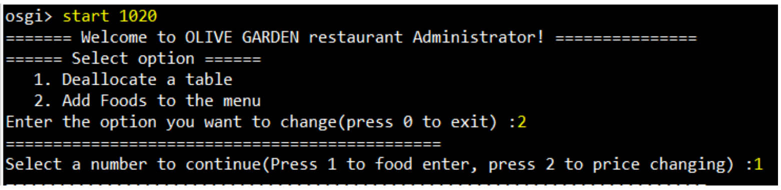
```
1014    ACTIVE      org.w3c.css.sac_1.3.1.v200903091627
1015    ACTIVE      org.w3c.dom.events_3.0.0.draft20060413_v201105210656
1016    ACTIVE      org.w3c.dom.smil_1.0.1.v200903091627
1017    ACTIVE      org.w3c.dom.svg_1.1.0.v201011041433
1020    RESOLVED    RestaurantAdmin_1.0.0.qualifier
1022    ACTIVE      RestaurantServ_ice_1.0.0.qualifier
1028    ACTIVE      DeliverService_1.0.0.qualifier
1030    ACTIVE      DeliverClient_1.0.0.qualifier
1031    ACTIVE      RestaurantClient_1.0.0.qualifier
osgi> stop 1030
Delivery Client Stop ...
osgi> start 1030
Delivery Client Start ...
Enter Your Name: shaini_shashiprabha



        ~WELCOME TO OLIVE GARDEN RESTAURANT DELIVERY SERVICE~
====================================================================

~Hello shaini_shashiprabha~

-----------------------------------
Enter distance for your address (KM): 3.5
Please Enter  Your Address: ambalangoda,galle
Do you have a promocode? (Y/N): n
Do you want to print the Bill (Y/N): y

=====================================================
Name........... = shaini_shashiprabha
Address........ = ambalangoda,galle
Delivery Cost.. = Rs.180.0
-----------------------------------
Delivery Cost.. = Rs.180.0
2019/03/31 00:12:31
-----------------------------------

=====================================================
Thank you. See you again ........

osgi>
```

Figure 12 – Without a promo code

- ## Restaurant Admin

  After starting the restaurant admin bundle, application prompts a welcome message with two selection options to continue the process. Administrator can;

  - Add food items to the menu with the relevant prices for each item
  - Deallocate the reserved tables
  - Exit from the current interface



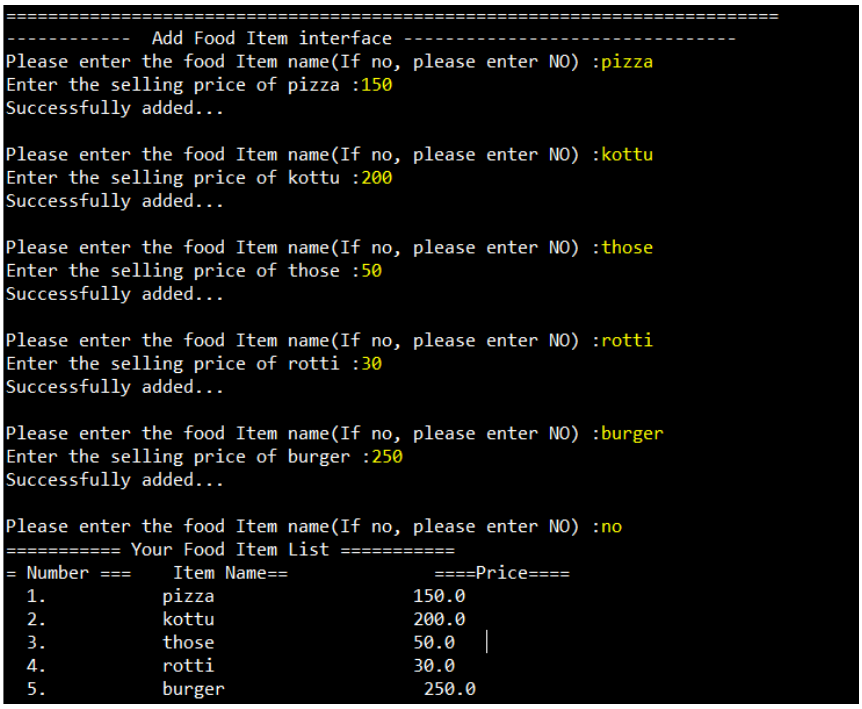
```
osgi> start 1020
======= Welcome to OLIVE GARDEN restaurant Administrator! ===============
====== Select option ======
   1. Deallocate a table
   2. Add Foods to the menu
Enter the option you want to change(press 0 to exit) :2
=============================================
Select a number to continue(Press 1 to food enter, press 2 to price changing) :1
```

Figure 13 – Administrator home   interface

- <u>Add food items</u>

  Administrator is allowed to enter any number of food items with the respective prices of each items. System displays the message, 'Successfully added…' if the food item is added successfully.

  Finally, administrator has to enter 'NO' to finish the data entry and application displays the food item list which was entered by the administrator

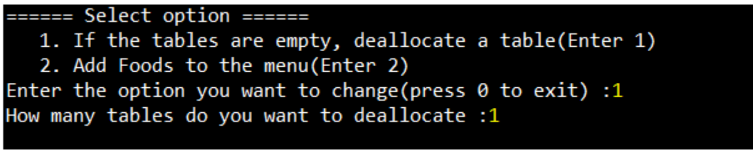


```
==========================================================================
------------  Add Food Item interface ------------------------------
Please enter the food Item name(If no, please enter NO) :pizza
Enter the selling price of pizza :150
Successfully added...

Please enter the food Item name(If no, please enter NO) :kottu
Enter the selling price of kottu :200
Successfully added...

Please enter the food Item name(If no, please enter NO) :those
Enter the selling price of those :50
Successfully added...

Please enter the food Item name(If no, please enter NO) :rotti
Enter the selling price of rotti :30
Successfully added...

Please enter the food Item name(If no, please enter NO) :burger
Enter the selling price of burger :250
Successfully added...

Please enter the food Item name(If no, please enter NO) :no
=========== Your Food Item List ===========
= Number ===     Item Name==              ====Price====
  1.           pizza                    150.0
  2.           kottu                    200.0
  3.           those                    50.0
  4.           rotti                    30.0
  5.           burger                    250.0
```

Figure 14 – Add food items interface

- Deallocate the reserved tables

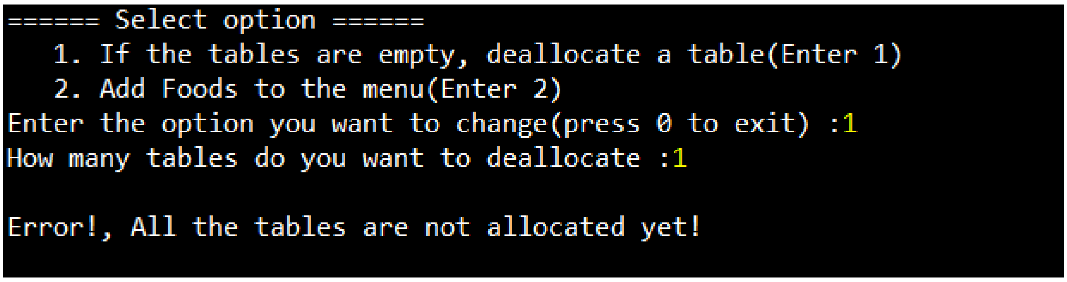
  In order to deallocate the reserved tables, administrator has to enter the number 1 as shown in the figure 15.



```
====== Select option ======
   1. If the tables are empty, deallocate a table(Enter 1)
   2. Add Foods to the menu(Enter 2)
Enter the option you want to change(press 0 to exit) :1
How many tables do you want to deallocate :1
```

Figure 15 – Deallocate tables option

After the execution of the previous step, application prompts to enter the number of tables to deallocate. If there are no reserved tables, application displays a message informing the administrator that the tables are not allocated yet



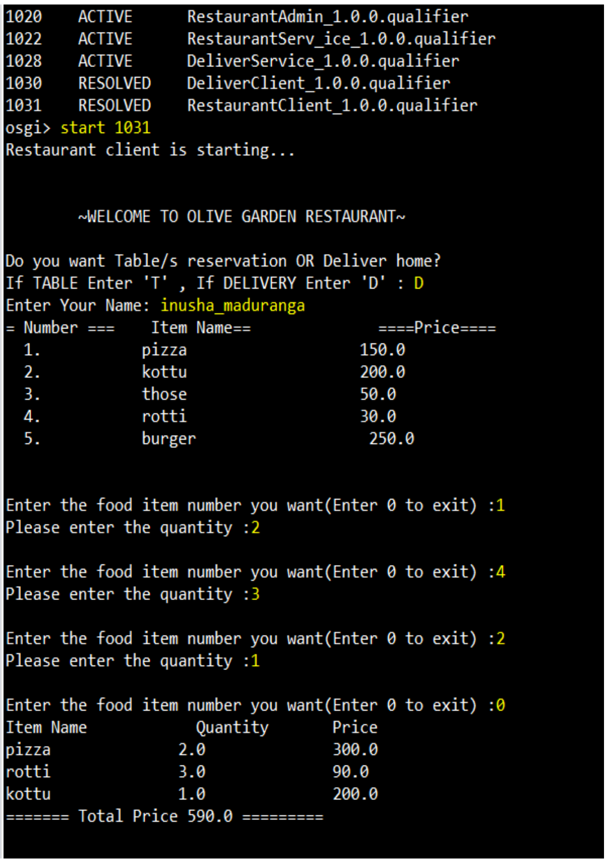
```
====== Select option ======
   1. If the tables are empty, deallocate a table(Enter 1)
   2. Add Foods to the menu(Enter 2)
Enter the option you want to change(press 0 to exit) :1
How many tables do you want to deallocate :1

Error!, All the tables are not allocated yet!
```

Figure 16 – Output message of tables deallocation

- ## Restaurant Client

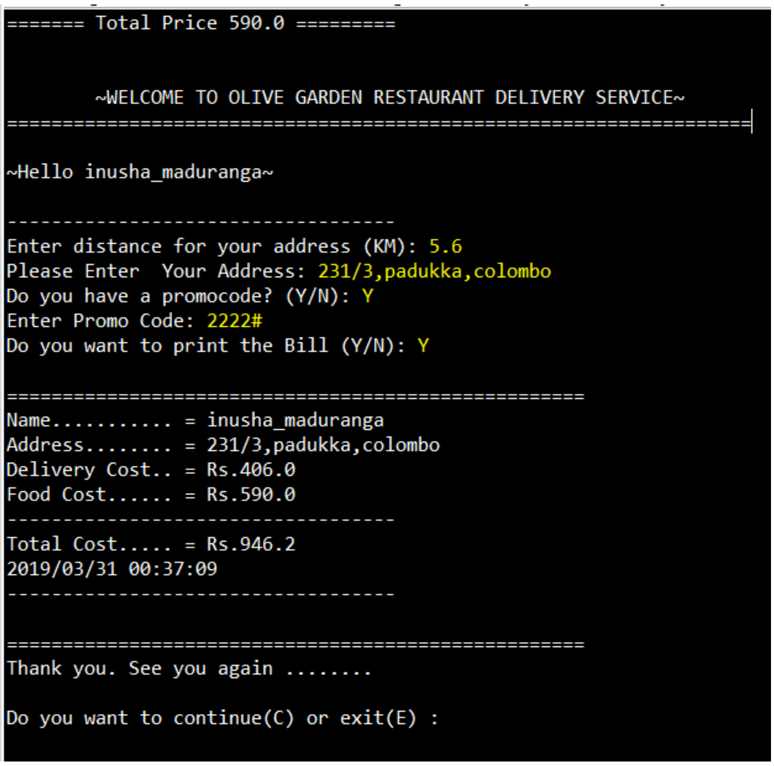
This bundle uses the restaurant service bundle. After running this bundle, application prompts a welcome message with a selection option to choose either the customer needs to reserve a table or a delivery.

If customer chooses the delivery option, application prompts a menu list and customer is allowed to choose food items. After entering the food items, application displays the total bill and the delivery service interface to add delivery details.



```
1020    ACTIVE      RestaurantAdmin_1.0.0.qualifier
1022    ACTIVE      RestaurantServ_ice_1.0.0.qualifier
1028    ACTIVE      DeliverService_1.0.0.qualifier
1030    RESOLVED    DeliverClient_1.0.0.qualifier
1031    RESOLVED    RestaurantClient_1.0.0.qualifier
osgi> start 1031
Restaurant client is starting...


        ~WELCOME TO OLIVE GARDEN RESTAURANT~

Do you want Table/s reservation OR Deliver home?
If TABLE Enter 'T' , If DELIVERY Enter 'D' : D
Enter Your Name: inusha_maduranga
= Number ===     Item Name==                ====Price====
  1.           pizza                      150.0
  2.           kottu                      200.0
  3.           those                      50.0
  4.           rotti                      30.0
  5.           burger                      250.0


Enter the food item number you want(Enter 0 to exit) :1
Please enter the quantity :2

Enter the food item number you want(Enter 0 to exit) :4
Please enter the quantity :3

Enter the food item number you want(Enter 0 to exit) :2
Please enter the quantity :1

Enter the food item number you want(Enter 0 to exit) :0
Item Name            Quantity        Price
pizza              2.0             300.0
rotti              3.0             90.0
kottu              1.0             200.0
======= Total Price 590.0 =========
```

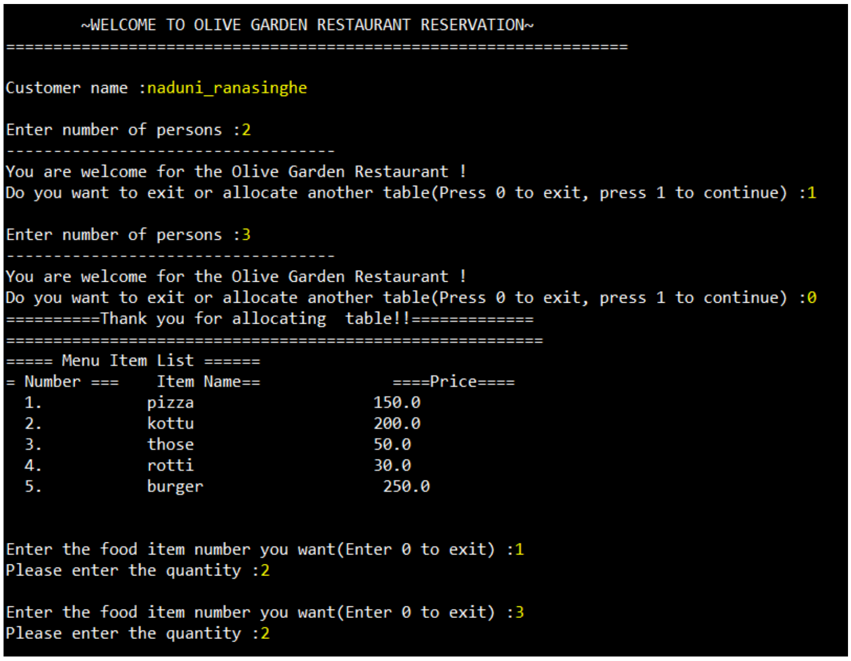
Figure 17 – Display menu items



```
======= Total Price 590.0 =========


        ~WELCOME TO OLIVE GARDEN RESTAURANT DELIVERY SERVICE~
===================================================================

~Hello inusha_maduranga~


-----------------------------------
Enter distance for your address (KM): 5.6
Please Enter  Your Address: 231/3,padukka,colombo
Do you have a promocode? (Y/N): Y
Enter Promo Code: 2222#
Do you want to print the Bill (Y/N): Y

===================================================
Name........... = inusha_maduranga
Address........ = 231/3,padukka,colombo
Delivery Cost.. = Rs.406.0
Food Cost...... = Rs.590.0
-----------------------------------
Total Cost..... = Rs.946.2
2019/03/31 00:37:09
-----------------------------------

===================================================
Thank you. See you again ........

Do you want to continue(C) or exit(E) :
```

Figure 18 – Delivery details

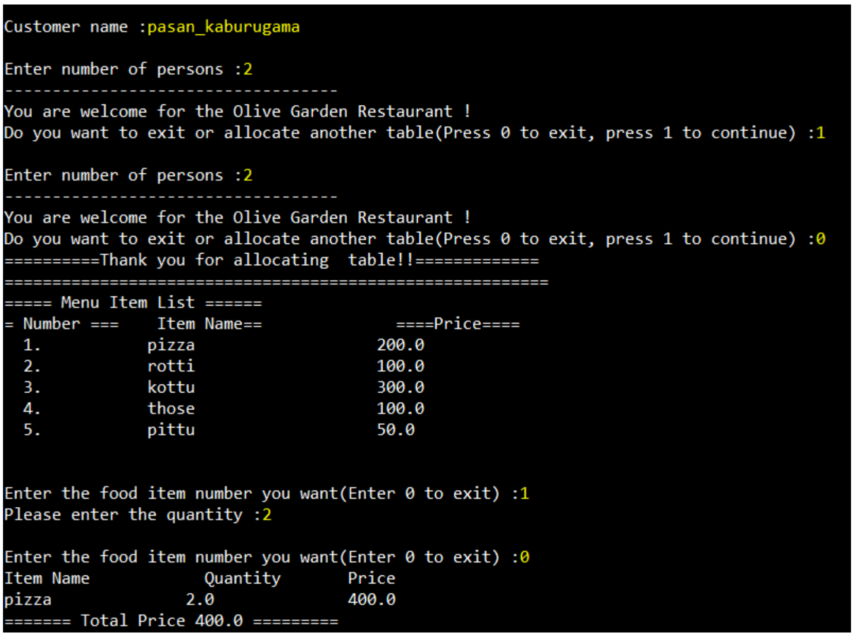
If the customer wants to reserve a table, he/she wants to enter the letter ‘T’. Then application prompts a welcome message with requesting the customer details. System asks the number of persons to allocate a relevant table. After finishing the reservation of tables, system displays the menu list and allows the customer to choose food items with the quantity.



```
        ~WELCOME TO OLIVE GARDEN RESTAURANT RESERVATION~
==================================================================

Customer name :naduni_ranasinghe

Enter number of persons :2
-----------------------------------
You are welcome for the Olive Garden Restaurant !
Do you want to exit or allocate another table(Press 0 to exit, press 1 to continue) :1

Enter number of persons :3
-----------------------------------
You are welcome for the Olive Garden Restaurant !
Do you want to exit or allocate another table(Press 0 to exit, press 1 to continue) :0
=========Thank you for allocating  table!!============
=========================================================
===== Menu Item List ======
= Number ===    Item Name==              ====Price====
  1.          pizza                   150.0
  2.          kottu                   200.0
  3.          those                   50.0
  4.          rotti                   30.0
  5.          burger                   250.0


Enter the food item number you want(Enter 0 to exit) :1
Please enter the quantity :2

Enter the food item number you want(Enter 0 to exit) :3
Please enter the quantity :2
```

Figure 19 – Order food

- After completing the order, system calculates the bill and displays it to the customer



```
Customer name :pasan_kaburugama

Enter number of persons :2
-----------------------------------
You are welcome for the Olive Garden Restaurant !
Do you want to exit or allocate another table(Press 0 to exit, press 1 to continue) :1

Enter number of persons :2
-----------------------------------
You are welcome for the Olive Garden Restaurant !
Do you want to exit or allocate another table(Press 0 to exit, press 1 to continue) :0
=========Thank you for allocating  table!!============
=========================================================
===== Menu Item List ======
= Number ===    Item Name==              ====Price====
  1.          pizza                   200.0
  2.          rotti                   100.0
  3.          kottu                   300.0
  4.          those                   100.0
  5.          pittu                   50.0


Enter the food item number you want(Enter 0 to exit) :1
Please enter the quantity :2

Enter the food item number you want(Enter 0 to exit) :0
Item Name            Quantity       Price
pizza              2.0              400.0
======= Total Price 400.0 =========
```

Figure 20 – Display the total bill

Conclusion

- To gain the best out of best of using the microkernel architecture pattern, the plug-in modules used in the system should be loosely coupled.
- As the name indicating loose coupling refers reducing the dependencies of a class or a package or a function that use a different class/function/package directly.

References

1.JAVAWORLD from IDG (2019). [online] Available at: https://www.javaworld.com/article/2077837/java-se-hello-osgi-part-1-bundles-for-beginners.html?page=2 [Accessed 20 Mar. 2019]

2. Vogella (2019) OSGi Modularity Tutorial [online] Available at: https://www.vogella.com/tutorials/OSGi/article.html [Accessed 20 Mar.2019]

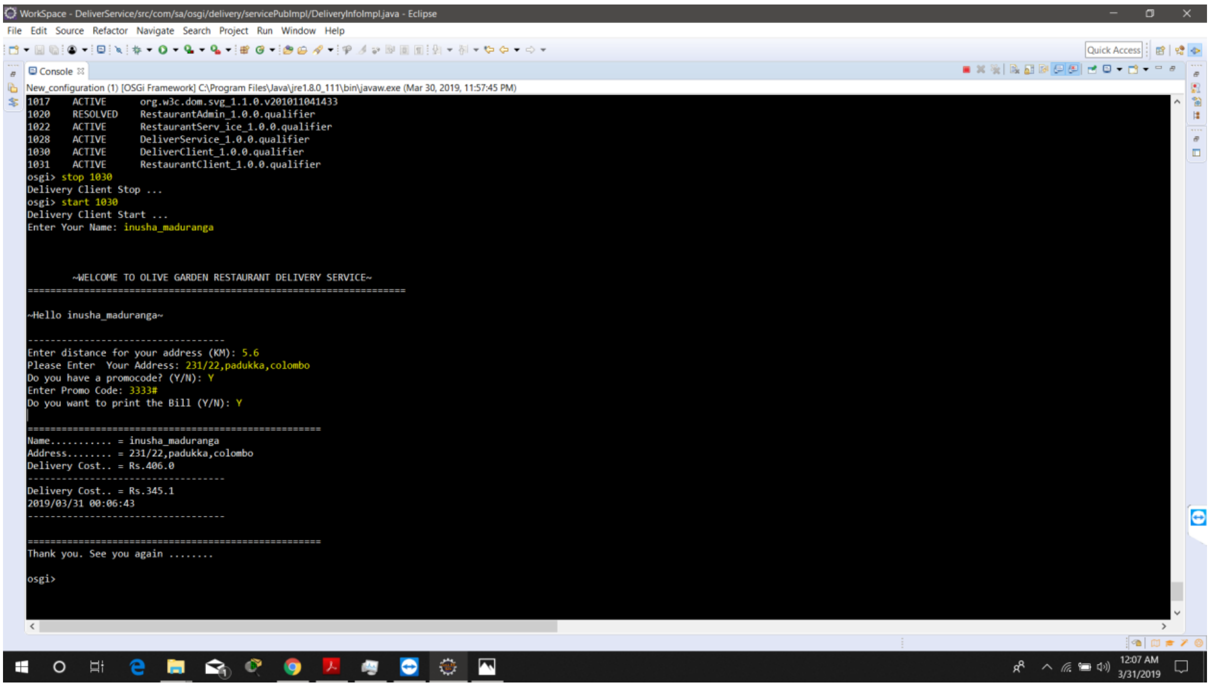
3. Oreilly (2019) Chapter 3 – Microkernel Architecture [online] Available at:

https://www.oreilly.com/library/view/software-architecture-patterns/9781491971437/ch03.html [Accessed 20 Mar. 2019]

Appendix A

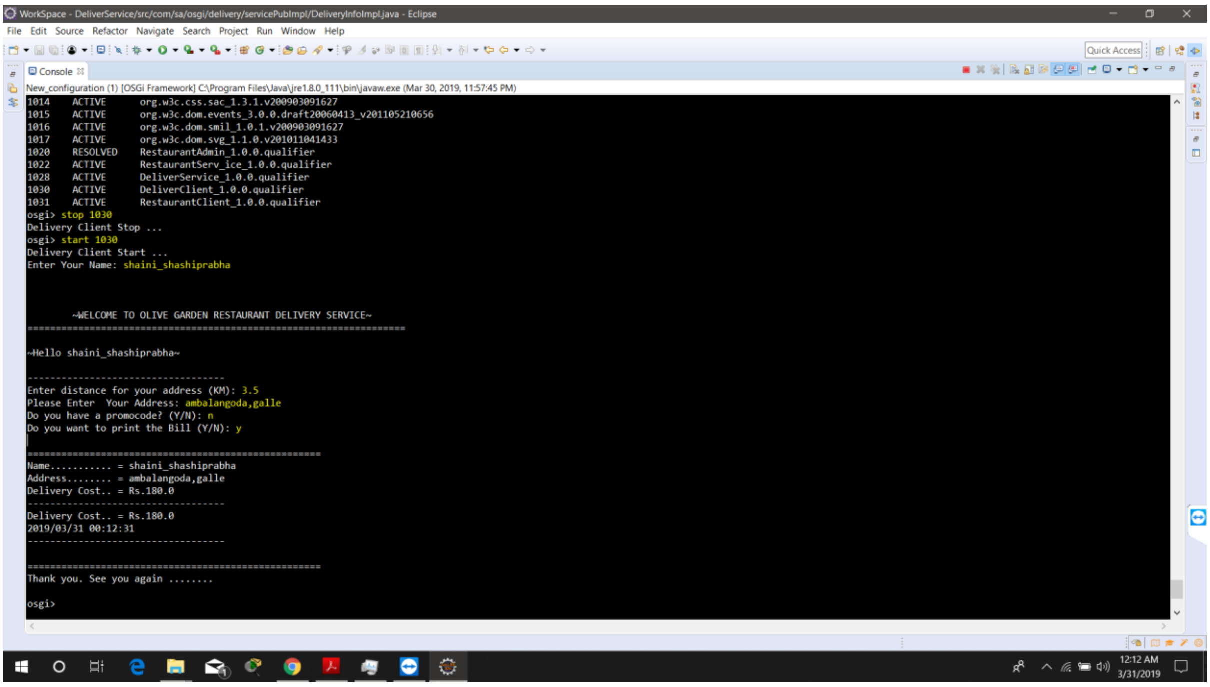
Screenshots of the running application

- Delivery client with promo code


WorkSpace - DeliverService/src/com/sa/osgi/delivery/servicePubImpl/DeliveryInfoImpl.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access
Console
12:07 AM
3/31/2019

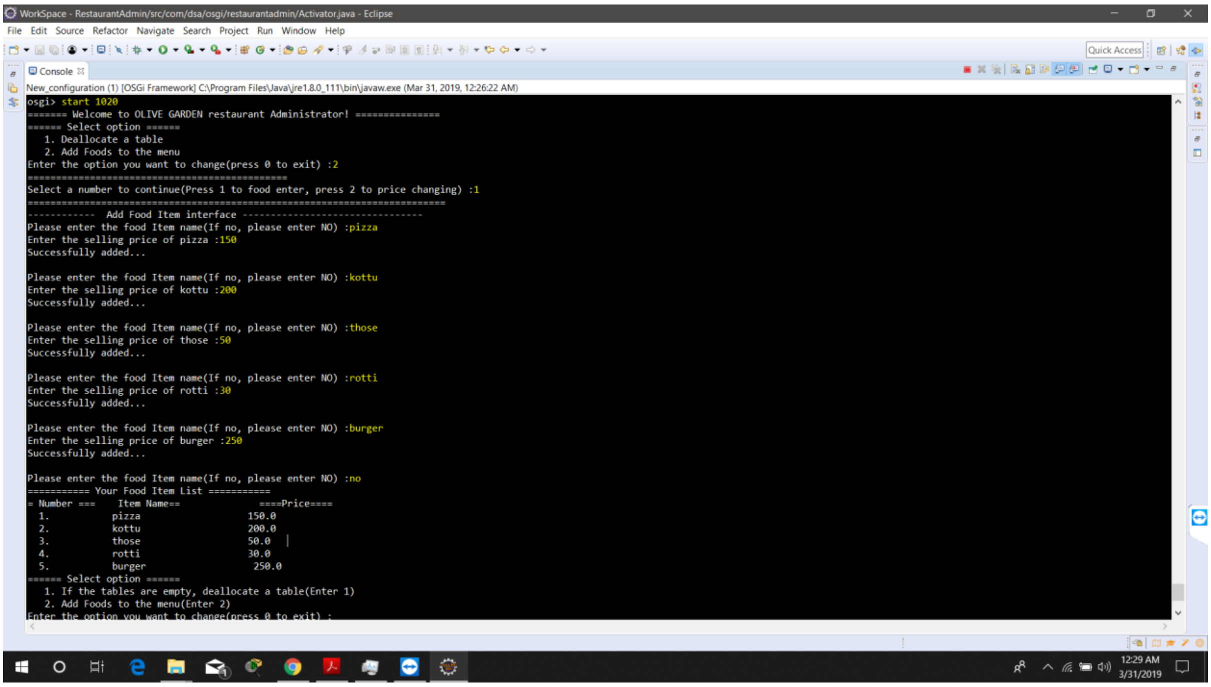
```
New_configuration (1) [OSGi Framework] C:\Program Files\Java\jre1.8.0_111\bin\javaw.exe (Mar 30, 2019, 11:57:45 PM)
1017    ACTIVE      org.w3c.dom.svg_1.1.0.v201011041433
1020    RESOLVED    RestaurantAdmin_1.0.0.qualifier
1022    ACTIVE      RestaurantServ_ice_1.0.0.qualifier
1028    ACTIVE      DeliverService_1.0.0.qualifier
1030    ACTIVE      DeliverClient_1.0.0.qualifier
1031    ACTIVE      RestaurantClient_1.0.0.qualifier
osgi> stop 1030
Delivery Client Stop ...
osgi> start 1030
Delivery Client Start ...
Enter Your Name: inusha_maduranga



        ~WELCOME TO OLIVE GARDEN RESTAURANT DELIVERY SERVICE~
===================================================================

~Hello inusha_maduranga~

-----------------------------------
Enter distance for your address (KM): 5.6
Please Enter  Your Address: 231/22,padukka,colombo
Do you have a promocode? (Y/N): Y
Enter Promo Code: 3333#
Do you want to print the Bill (Y/N): Y

====================================================
Name........... = inusha_maduranga
Address........ = 231/22,padukka,colombo
Delivery Cost.. = Rs.406.0
-----------------------------------
Delivery Cost.. = Rs.345.1
2019/03/31 00:06:43
-----------------------------------


====================================================
Thank you. See you again ........

osgi>
```

- Delivery client without promocode


WorkSpace - DeliverService/src/com/sa/osgi/delivery/servicePubImpl/DeliveryInfoImpl.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access
Console
12:12 AM
3/31/2019

```
New_configuration (1) [OSGi Framework] C:\Program Files\Java\jre1.8.0_111\bin\javaw.exe (Mar 30, 2019, 11:57:45 PM)
1014    ACTIVE      org.w3c.css.sac_1.3.1.v200903091627
1015    ACTIVE      org.w3c.dom.events_3.0.0.draft20060413_v201105210656
1016    ACTIVE      org.w3c.dom.smil_1.0.1.v200903091627
1017    ACTIVE      org.w3c.dom.svg_1.1.0.v201011041433
1020    RESOLVED    RestaurantAdmin_1.0.0.qualifier
1022    ACTIVE      RestaurantServ_ice_1.0.0.qualifier
1028    ACTIVE      DeliverService_1.0.0.qualifier
1030    ACTIVE      DeliverClient_1.0.0.qualifier
1031    ACTIVE      RestaurantClient_1.0.0.qualifier
osgi> stop 1030
Delivery Client Stop ...
osgi> start 1030
Delivery Client Start ...
Enter Your Name: shaini_shashiprabha



        ~WELCOME TO OLIVE GARDEN RESTAURANT DELIVERY SERVICE~
===================================================================

~Hello shaini_shashiprabha~

-----------------------------------
Enter distance for your address (KM): 3.5
Please Enter  Your Address: ambalangoda,galle
Do you have a promocode? (Y/N): n
Do you want to print the Bill (Y/N): y

====================================================
Name........... = shaini_shashiprabha
Address........ = ambalangoda,galle
Delivery Cost.. = Rs.180.0
-----------------------------------
Delivery Cost.. = Rs.180.0
2019/03/31 00:12:31
-----------------------------------


====================================================
Thank you. See you again ........

osgi>
```

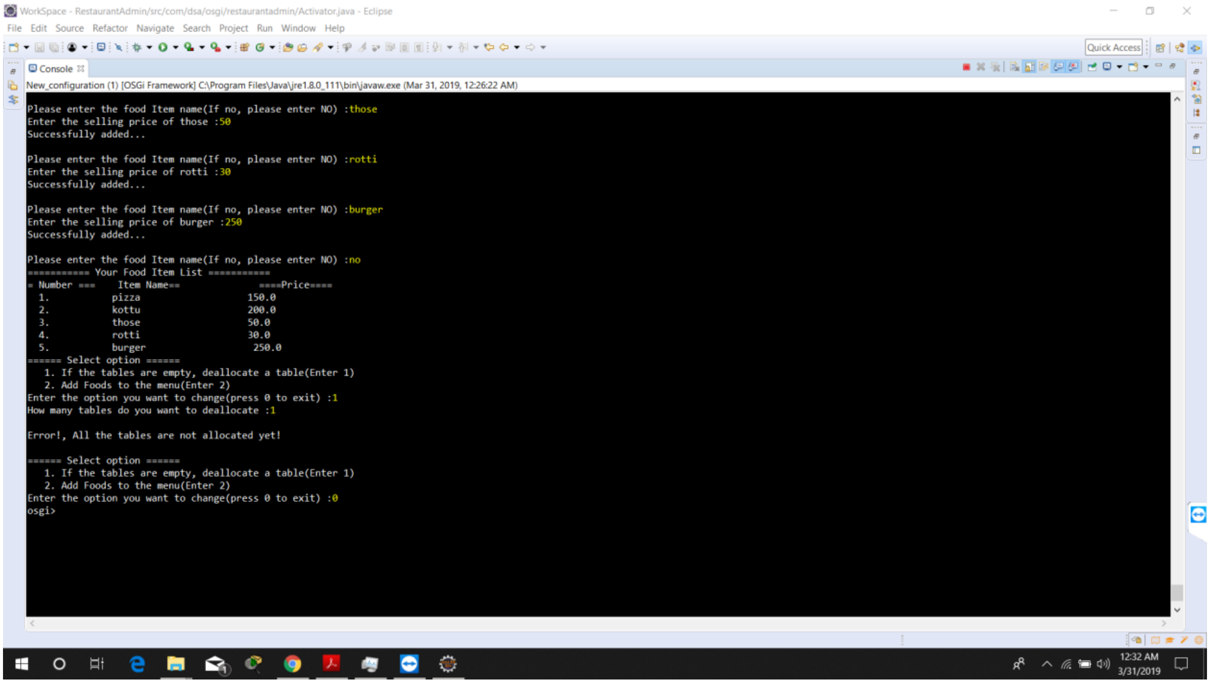
- Administrator enter food items



WorkSpace - RestaurantAdmin/src/com/dsa/osgi/restaurantadmin/Activator.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access
Console
12:29 AM
3/31/2019

```
New_configuration (1) [OSGi Framework] C:\Program Files\Java\jre1.8.0_111\bin\javaw.exe (Mar 31, 2019, 12:26:22 AM)
osgi> start 1020
======= Welcome to OLIVE GARDEN restaurant Administrator! ===============
====== Select option ======
   1. Deallocate a table
   2. Add Foods to the menu
Enter the option you want to change(press 0 to exit) :2
==============================================
Select a number to continue(Press 1 to food enter, press 2 to price changing) :1
==========================================================================
------------  Add Food Item interface --------------------------------
Please enter the food Item name(If no, please enter NO) :pizza
Enter the selling price of pizza :150
Successfully added...

Please enter the food Item name(If no, please enter NO) :kottu
Enter the selling price of kottu :200
Successfully added...

Please enter the food Item name(If no, please enter NO) :those
Enter the selling price of those :50
Successfully added...

Please enter the food Item name(If no, please enter NO) :rotti
Enter the selling price of rotti :30
Successfully added...

Please enter the food Item name(If no, please enter NO) :burger
Enter the selling price of burger :250
Successfully added...

Please enter the food Item name(If no, please enter NO) :no
=========== Your Food Item List ===========
= Number ===    Item Name==              ====Price====
  1.           pizza                   150.0
  2.           kottu                   200.0
  3.           those                   50.0
  4.           rotti                   30.0
  5.           burger                   250.0
====== Select option ======
   1. If the tables are empty, deallocate a table(Enter 1)
   2. Add Foods to the menu(Enter 2)
Enter the option you want to change(press 0 to exit) :
```

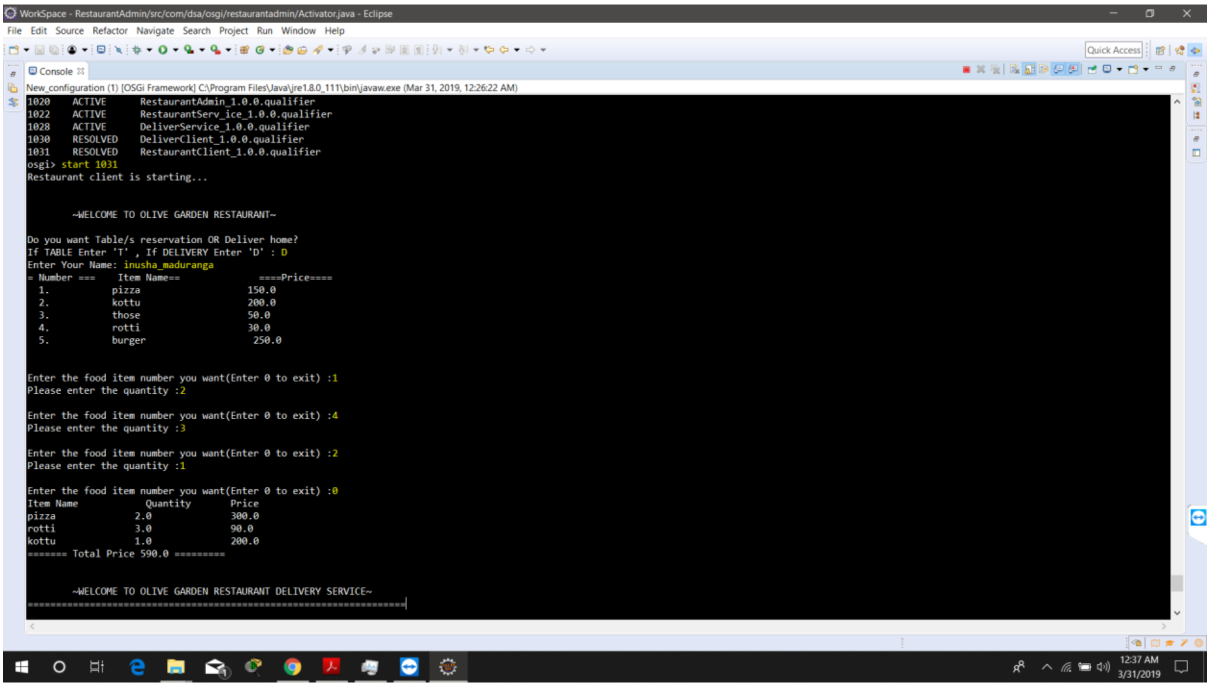
- Deallocation of tables



WorkSpace - RestaurantAdmin/src/com/dsa/osgi/restaurantadmin/Activator.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access
Console
12:32 AM
3/31/2019

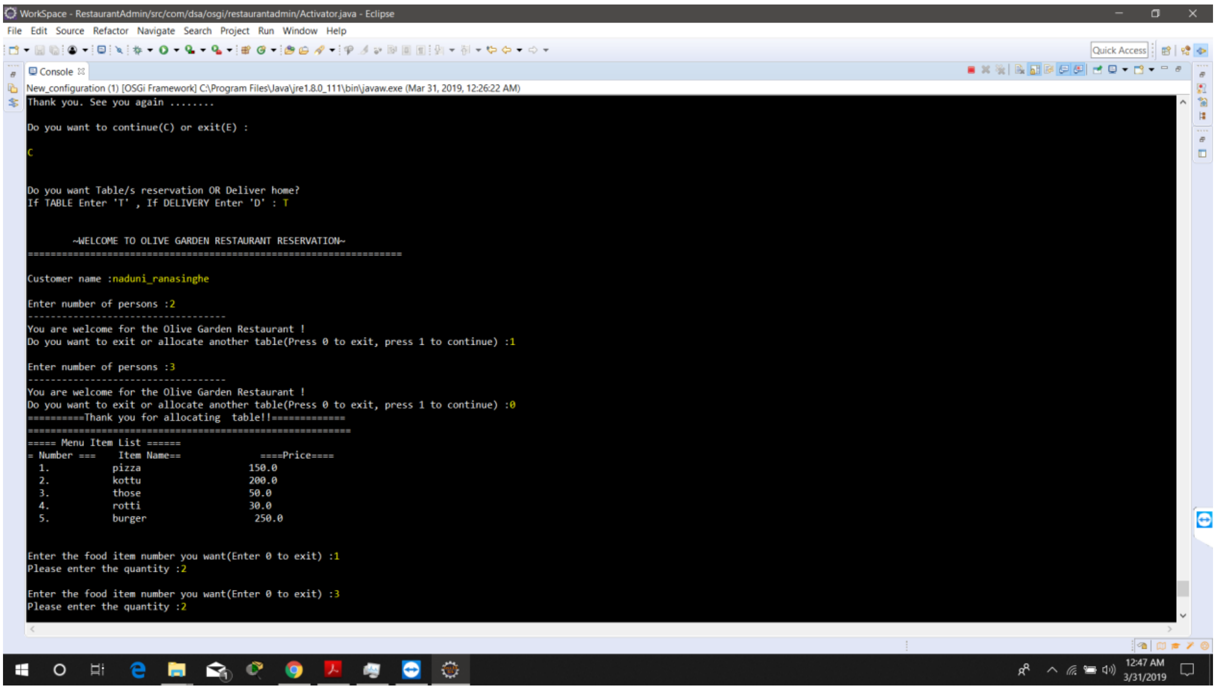
```
New_configuration (1) [OSGi Framework] C:\Program Files\Java\jre1.8.0_111\bin\javaw.exe (Mar 31, 2019, 12:26:22 AM)
Please enter the food Item name(If no, please enter NO) :those
Enter the selling price of those :50
Successfully added...

Please enter the food Item name(If no, please enter NO) :rotti
Enter the selling price of rotti :30
Successfully added...

Please enter the food Item name(If no, please enter NO) :burger
Enter the selling price of burger :250
Successfully added...

Please enter the food Item name(If no, please enter NO) :no
=========== Your Food Item List ===========
= Number ===    Item Name==              ====Price====
  1.           pizza                   150.0
  2.           kottu                   200.0
  3.           those                   50.0
  4.           rotti                   30.0
  5.           burger                   250.0
====== Select option ======
   1. If the tables are empty, deallocate a table(Enter 1)
   2. Add Foods to the menu(Enter 2)
Enter the option you want to change(press 0 to exit) :1
How many tables do you want to deallocate :1

Error!, All the tables are not allocated yet!

====== Select option ======
   1. If the tables are empty, deallocate a table(Enter 1)
   2. Add Foods to the menu(Enter 2)
Enter the option you want to change(press 0 to exit) :0
osgi>
```

- Order food and delivery


WorkSpace - RestaurantAdmin/src/com/dsa/osgi/restaurantadmin/Activator.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access
Console
New_configuration (1) [OSGi Framework] C:\Program Files\Java\jre1.8.0_111\bin\javaw.exe (Mar 31, 2019, 12:26:22 AM)
12:37 AM
3/31/2019

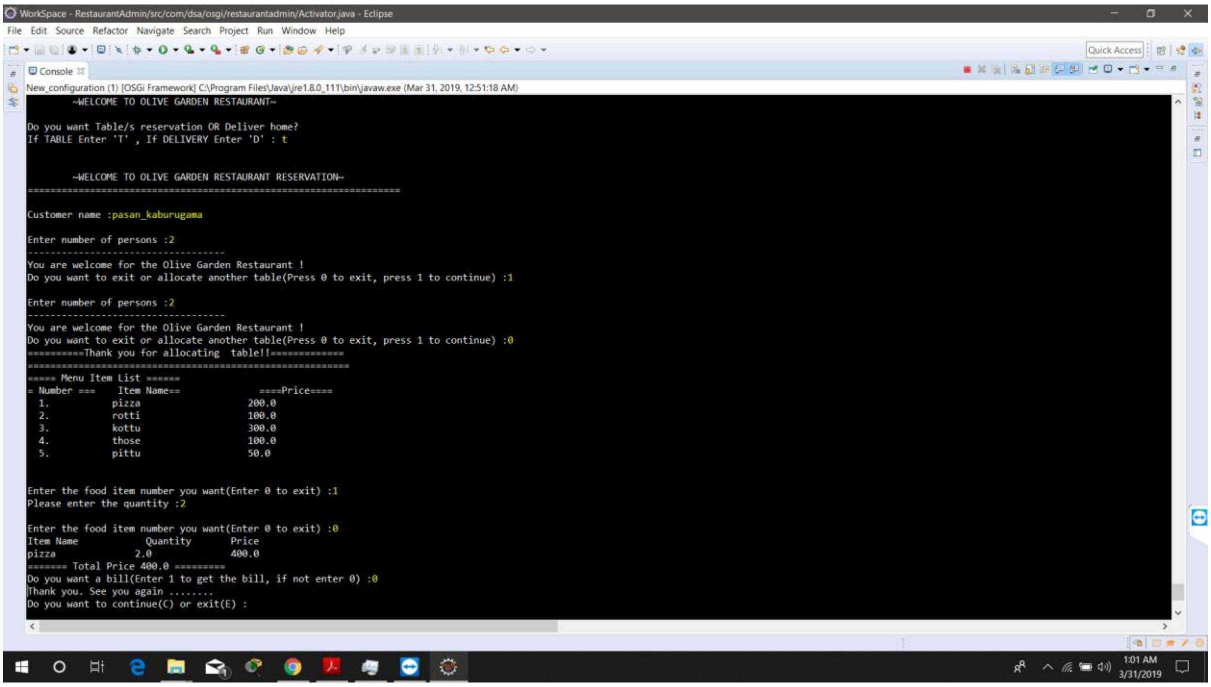
```
1020    ACTIVE      RestaurantAdmin_1.0.0.qualifier
1022    ACTIVE      RestaurantServ_ice_1.0.0.qualifier
1028    ACTIVE      DeliverService_1.0.0.qualifier
1030    RESOLVED    DeliverClient_1.0.0.qualifier
1031    RESOLVED    RestaurantClient_1.0.0.qualifier
osgi> start 1031
Restaurant client is starting...


        ~WELCOME TO OLIVE GARDEN RESTAURANT~

Do you want Table/s reservation OR Deliver home?
If TABLE Enter 'T' , If DELIVERY Enter 'D' : D
Enter Your Name: inusha_maduranga
= Number ===    Item Name==              ====Price====
  1.           pizza                   150.0
  2.           kottu                   200.0
  3.           those                   50.0
  4.           rotti                   30.0
  5.           burger                   250.0


Enter the food item number you want(Enter 0 to exit) :1
Please enter the quantity :2

Enter the food item number you want(Enter 0 to exit) :4
Please enter the quantity :3

Enter the food item number you want(Enter 0 to exit) :2
Please enter the quantity :1

Enter the food item number you want(Enter 0 to exit) :0
Item Name            Quantity       Price
pizza               2.0             300.0
rotti               3.0             90.0
kottu               1.0             200.0
======= Total Price 590.0 =========


        ~WELCOME TO OLIVE GARDEN RESTAURANT DELIVERY SERVICE~
===================================================================
```

- Order food and reserve a table


WorkSpace - RestaurantAdmin/src/com/dsa/osgi/restaurantadmin/Activator.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access
Console
New_configuration (1) [OSGi Framework] C:\Program Files\Java\jre1.8.0_111\bin\javaw.exe (Mar 31, 2019, 12:26:22 AM)
12:47 AM
3/31/2019

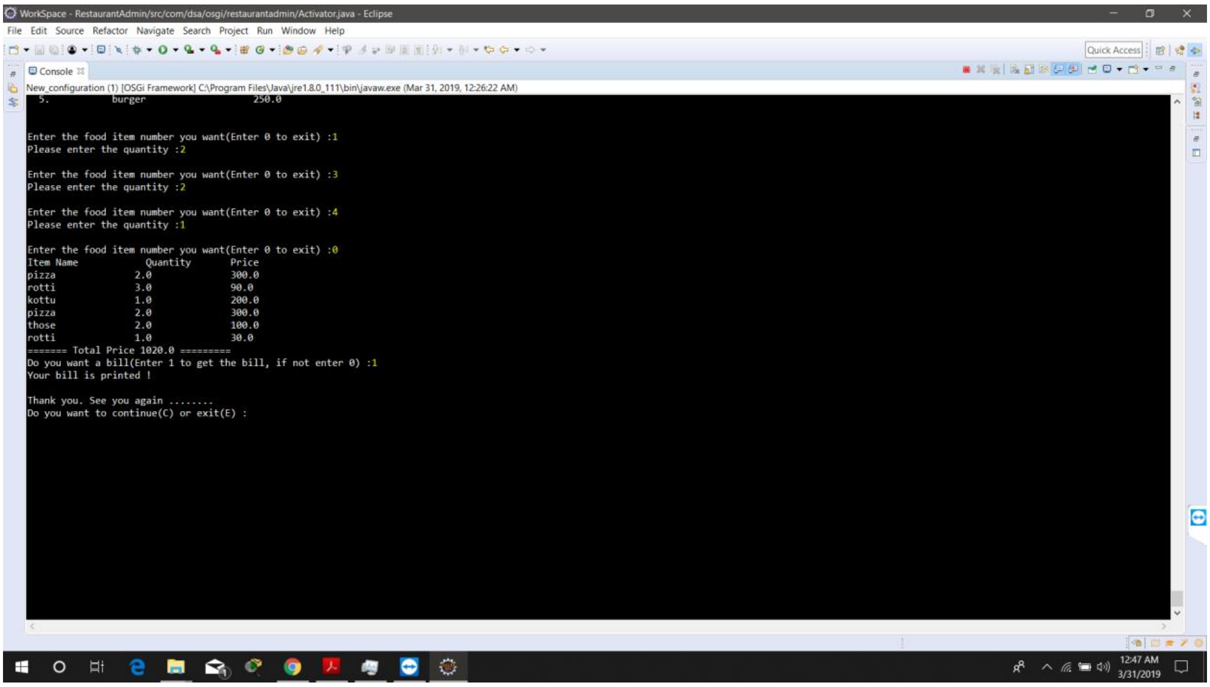
```
Thank you. See you again ........

Do you want to continue(C) or exit(E) :

C


Do you want Table/s reservation OR Deliver home?
If TABLE Enter 'T' , If DELIVERY Enter 'D' : T


        ~WELCOME TO OLIVE GARDEN RESTAURANT RESERVATION~
==================================================================

Customer name :naduni_ranasinghe

Enter number of persons :2
-----------------------------------
You are welcome for the Olive Garden Restaurant !
Do you want to exit or allocate another table(Press 0 to exit, press 1 to continue) :1

Enter number of persons :3
-----------------------------------
You are welcome for the Olive Garden Restaurant !
Do you want to exit or allocate another table(Press 0 to exit, press 1 to continue) :0
==========Thank you for allocating  table!!=============
=========================================================
===== Menu Item List ======
= Number ===    Item Name==              ====Price====
  1.           pizza                   150.0
  2.           kottu                   200.0
  3.           those                   50.0
  4.           rotti                   30.0
  5.           burger                   250.0


Enter the food item number you want(Enter 0 to exit) :1
Please enter the quantity :2

Enter the food item number you want(Enter 0 to exit) :3
Please enter the quantity :2
```

- Order food without a bill


WorkSpace - RestaurantAdmin/src/com/dsa/osgi/restaurantadmin/Activator.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access
Console
1:01 AM
3/31/2019

```
New_configuration (1) [OSGi Framework] C:\Program Files\Java\jre1.8.0_111\bin\javaw.exe (Mar 31, 2019, 12:51:18 AM)
        ~WELCOME TO OLIVE GARDEN RESTAURANT~

Do you want Table/s reservation OR Deliver home?
If TABLE Enter 'T' , If DELIVERY Enter 'D' : t


        ~WELCOME TO OLIVE GARDEN RESTAURANT RESERVATION~
==================================================================

Customer name :pasan_kaburugama

Enter number of persons :2
-----------------------------------
You are welcome for the Olive Garden Restaurant !
Do you want to exit or allocate another table(Press 0 to exit, press 1 to continue) :1

Enter number of persons :2
-----------------------------------
You are welcome for the Olive Garden Restaurant !
Do you want to exit or allocate another table(Press 0 to exit, press 1 to continue) :0
==========Thank you for allocating  table!!=============
=========================================================
===== Menu Item List ======
= Number ===    Item Name==              ====Price====
  1.           pizza                    200.0
  2.           rotti                    100.0
  3.           kottu                    300.0
  4.           those                    100.0
  5.           pittu                    50.0


Enter the food item number you want(Enter 0 to exit) :1
Please enter the quantity :2

Enter the food item number you want(Enter 0 to exit) :0
Item Name            Quantity       Price
pizza              2.0              400.0
======= Total Price 400.0 =========
Do you want a bill(Enter 1 to get the bill, if not enter 0) :0
Thank you. See you again ........
Do you want to continue(C) or exit(E) :
```

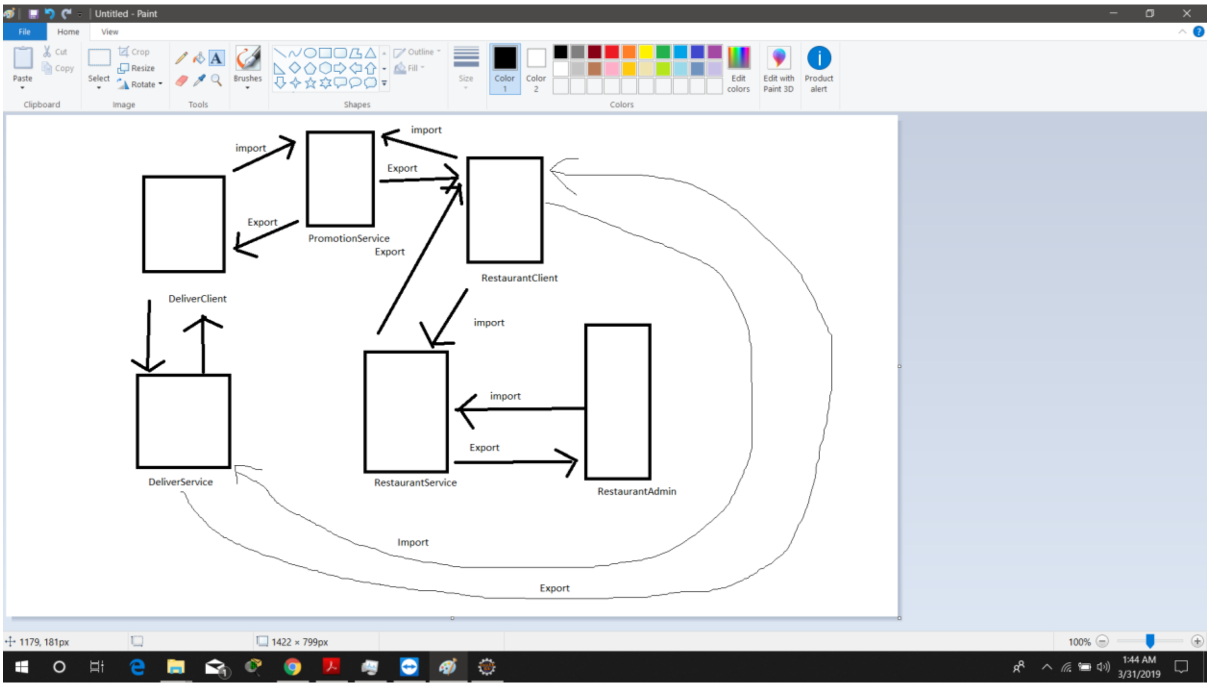
- Order food with a bill

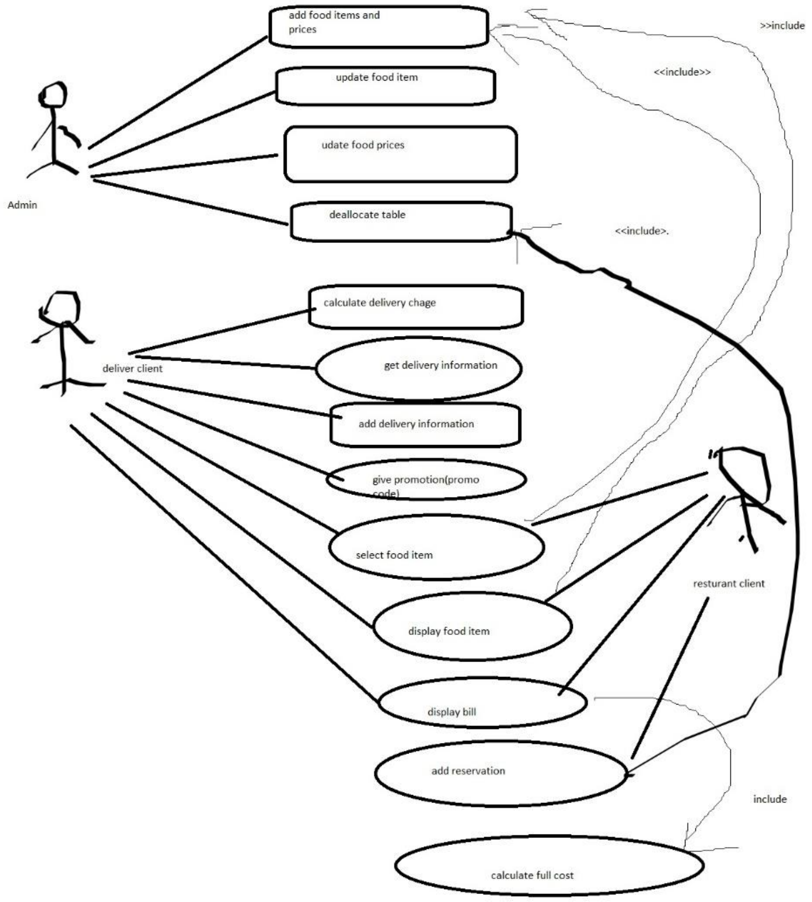

WorkSpace - RestaurantAdmin/src/com/dsa/osgi/restaurantadmin/Activator.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access
Console
12:47 AM
3/31/2019

```
New_configuration (1) [OSGi Framework] C:\Program Files\Java\jre1.8.0_111\bin\javaw.exe (Mar 31, 2019, 12:26:22 AM)
  5.           burger                   250.0


Enter the food item number you want(Enter 0 to exit) :1
Please enter the quantity :2

Enter the food item number you want(Enter 0 to exit) :3
Please enter the quantity :2

Enter the food item number you want(Enter 0 to exit) :4
Please enter the quantity :1

Enter the food item number you want(Enter 0 to exit) :0
Item Name            Quantity       Price
pizza              2.0              300.0
rotti              3.0              90.0
kottu              1.0              200.0
pizza              2.0              300.0
those              2.0              100.0
rotti              1.0              30.0
======= Total Price 1020.0 =========
Do you want a bill(Enter 1 to get the bill, if not enter 0) :1
Your bill is printed !

Thank you. See you again ........
Do you want to continue(C) or exit(E) :
```

Appendix B

Planning diagrams


Untitled - Paint
File
Home
View
Paste
Cut
Copy
Select
Crop
Resize
Rotate
Brushes
Outline
Fill
Size
Color 1
Color 2
Edit colors
Edit with Paint 3D
Product alert
Clipboard
Image
Tools
Shapes
Colors
import
import
Export
Export
PromotionService
Export
RestaurantClient
DeliverClient
import
import
Export
DeliverService
RestaurantService
RestaurantAdmin
Import
Export
1179, 181px
1422 × 799px
100%
1:44 AM
3/31/2019


add food items and prices
update food item
udate food prices
deallocate table
Admin
>>include
<<include>>
<<include>.
calculate delivery chage
get delivery information
add delivery information
give promotion(promo code)
select food item
display food item
display bill
add reservation
calculate full cost
deliver client
resturant client
include