Sri Lanka Institute of Information Technology

Modern Topic in Information Technology

Assignment 01

Modular Development

Group ID : MTIT-012

Submitted to
Sri Lanka Institute of Information Technology

In partial fulfillment of the requirements for the
B.Sc. Special (Hons) Degree in Information Technology

March 2022

DECLARATION

We declare that this is our own work, and this proposal does not incorporate without acknowledgement any material previously submitted for a degree or diploma in any other university or Institute of higher learning and to the best of our knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text.

Registration Number	Name
IT18210002	Kasthuriarachchi K.M.W
IT18540536	Lanerolle T.Y
IT18218572	Prashadika W.M.J
IT18062120	Wickramasinghe W.A.P.C

Source Codes -
https://drive.google.com/drive/folders/13Mmv5tA1nWcnMcJorCuqH0eCKwkOogu3?usp=sharing

Table of Contents

Introduction

ABC Company - Managed IT Service Provider

ABC Company is a managed IT service provider that provides application, network, and system management services to large enterprises which need third party IT support. ABC Company helps enterprises without IT professionals to improve day to day operations. ABC Company mainly includes 4 sections,

1. Human Resources Section
2. Customer Services Section
3. Internal Projects Section
4. Marketing Section

This document contains how the four main sections separately works according to the producer consumer model. Four sections are having their own tasks to follow the producer consumer model. These sections are providing independent services for their consumers.

ABC Company - Sections
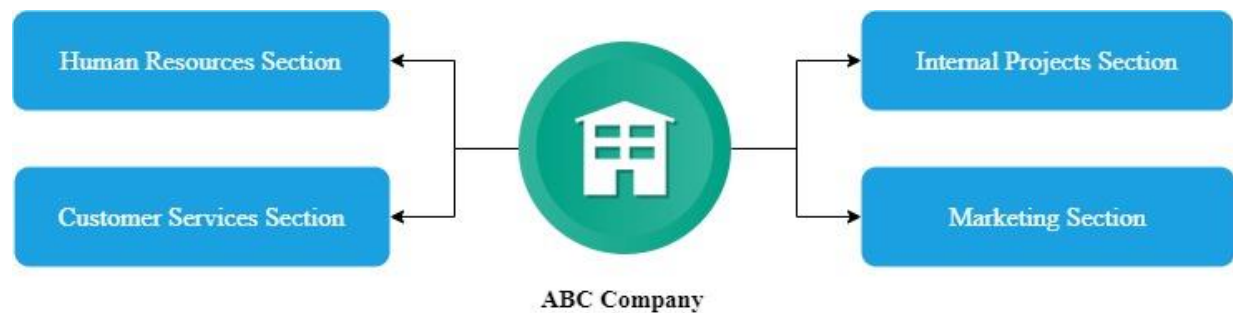
ABC Company – HR division (IT18210002)

The HR division of ABC company provides Personal information service, Leave Information service, Allocation information service and Appraisal information service for other divisions of the company. The Marketing division of the ABC company subscribes the personal information service and the Leave information service.

Class Diagram (IT18210002)
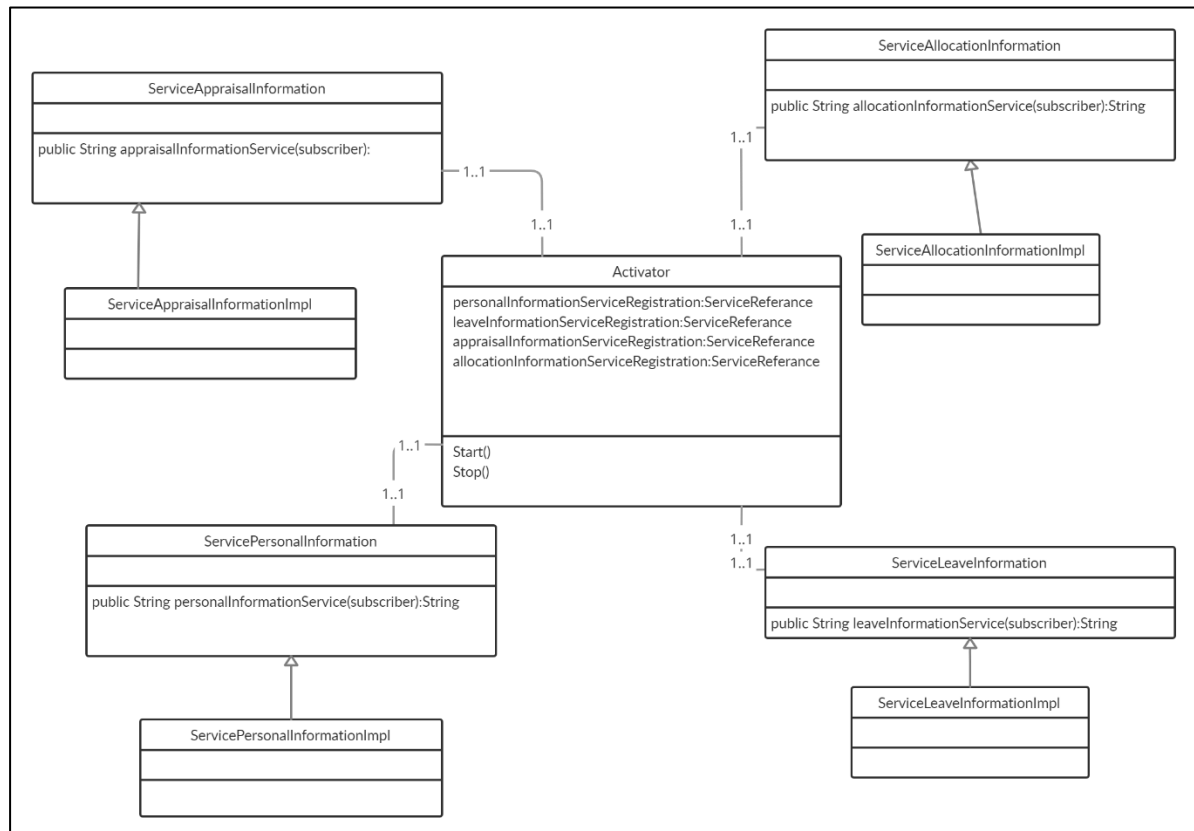
Structure of the project (IT18210002)
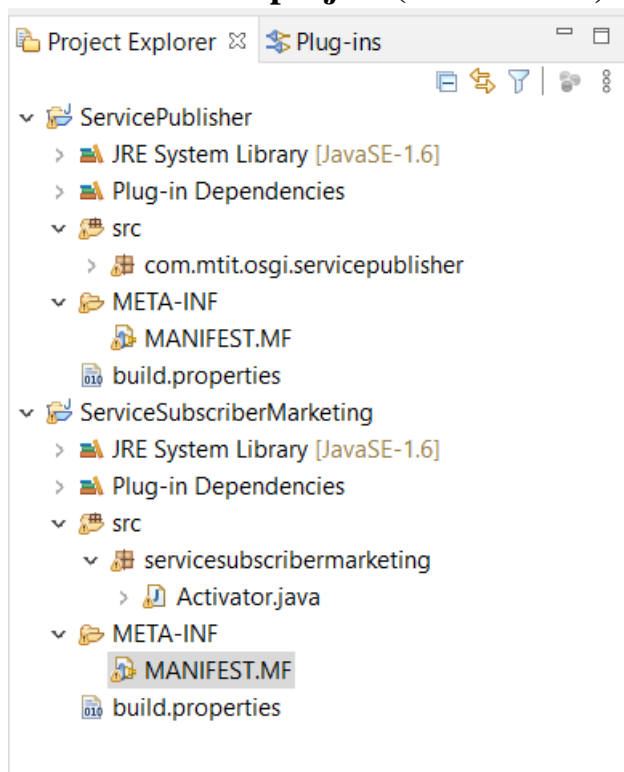
Service Status (IT18210002)
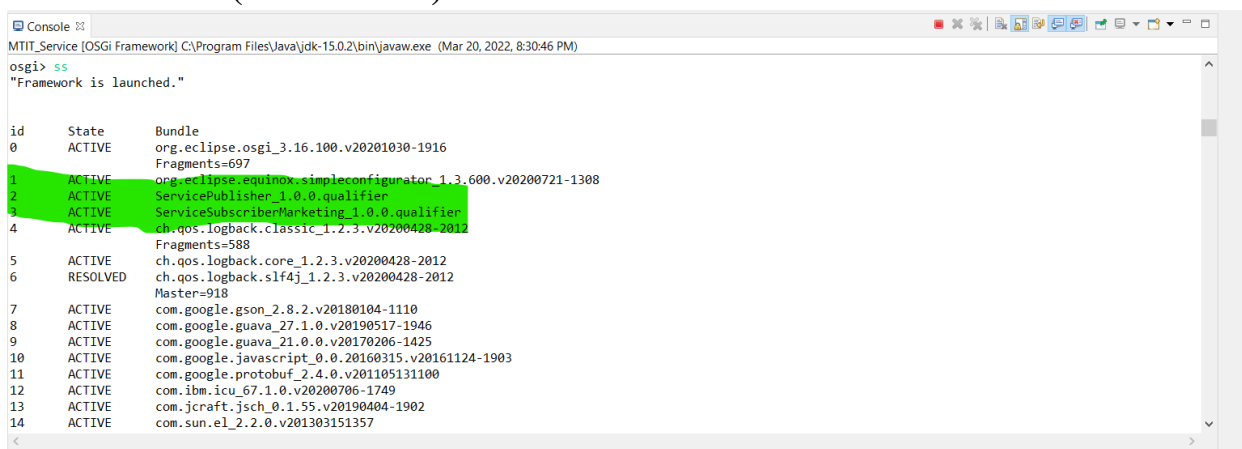
Stop Subscriber- Marketing Service (IT18210002)
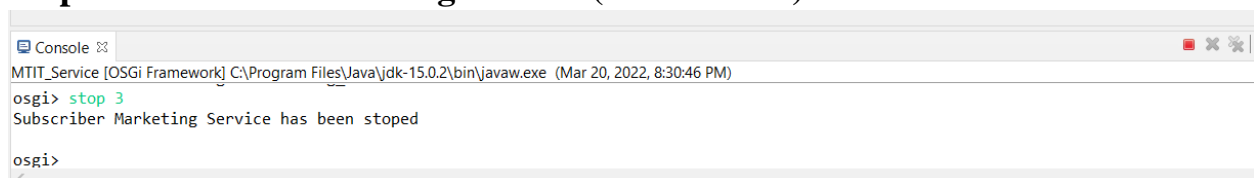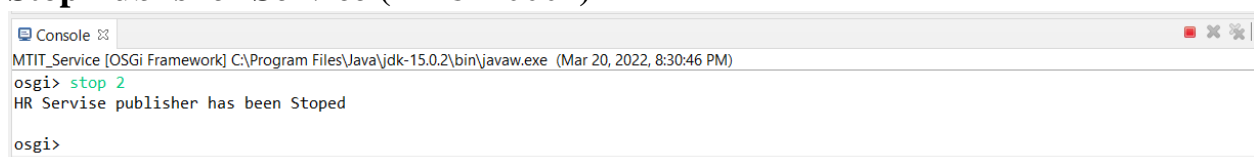

```
osgi> stop 3
Subscriber Marketing Service has been stoped

osgi>
```

Stop Publisher Service (IT18210002)


```
osgi> stop 2
HR Servise publisher has been Stoped

osgi>
```

Start Publisher Service (IT18210002)

```
Console ⌧                                                    ■ ✕ ✕|
MTIT_Service [OSGi Framework] C:\Program Files\Java\jdk-15.0.2\bin\javaw.exe  (Mar 20, 2022, 8:30:46 PM)

osgi> start 2
HR Servise publisher has been Started
osgi>
<
```

Start Subscriber- Marketing Service (IT18210002)
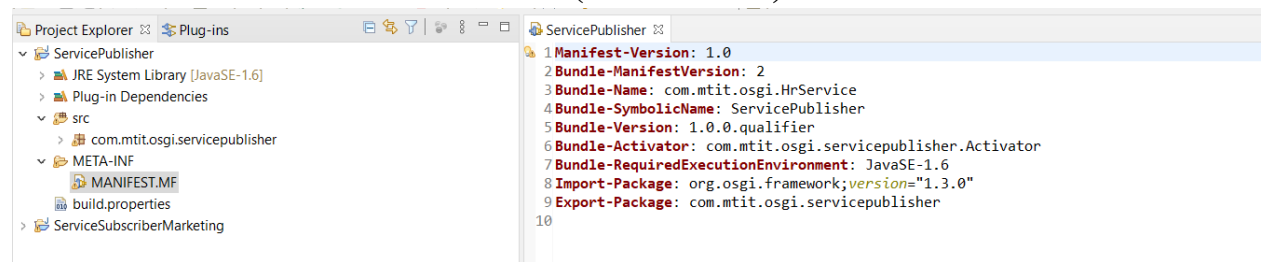
```
Console ⌧                                                    ■ ✕ ✕|
MTIT_Service [OSGi Framework] C:\Program Files\Java\jdk-15.0.2\bin\javaw.exe  (Mar 20, 2022, 8:30:46 PM)
osgi> start 3
Subscriber Marketing Service has been started
Excute the personal information service for : Subscriber Marketing
=================================================
No of requested personal files : = 20
No of updated personal files : = 30
 No of rejected personal files : = 2

Excute the personal information service for : Subscriber Marketing
=================================================
No of Approved Medical leaves : = 5
No of Annual leaves : = 2
 No of cashual leaves : = 1
```

Structure of the Service Producer project (IT18210002)
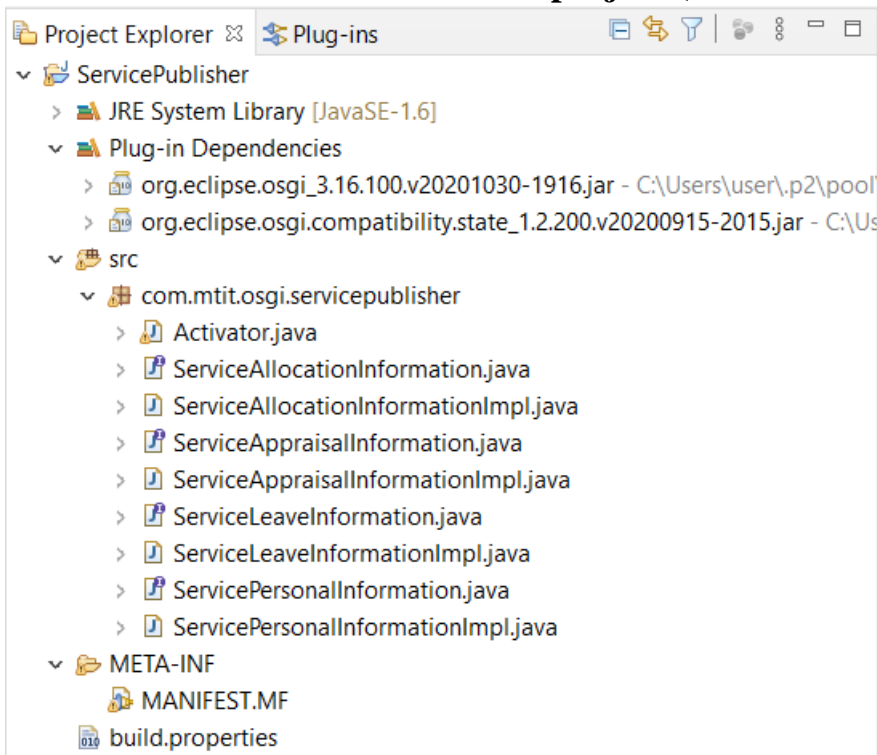
```
Project Explorer ⌧    Plug-ins          ⊟ ⥻ ▽ | ⚙ ⁝ ▭ ☐
∨ 🗁 ServicePublisher
  > 🗁 JRE System Library [JavaSE-1.6]
  ∨ 🗁 Plug-in Dependencies
    > 📦 org.eclipse.osgi_3.16.100.v20201030-1916.jar - C:\Users\user\.p2\pool
    > 📦 org.eclipse.osgi.compatibility.state_1.2.200.v20200915-2015.jar - C:\Us
  ∨ 🗁 src
    ∨ 🗁 com.mtit.osgi.servicepublisher
      > 🗋 Activator.java
      > 🗋 ServiceAllocationInformation.java
      > 🗋 ServiceAllocationInformationImpl.java
      > 🗋 ServiceAppraisalInformation.java
      > 🗋 ServiceAppraisalInformationImpl.java
      > 🗋 ServiceLeaveInformation.java
      > 🗋 ServiceLeaveInformationImpl.java
      > 🗋 ServicePersonalInformation.java
      > 🗋 ServicePersonalInformationImpl.java
  ∨ 🗁 META-INF
      🗋 MANIFEST.MF
    🗋 build.properties
```

Service Producer MANIFEST.MF (IT18210002)

```
Project Explorer ⊠  ⤡ Plug-ins                    ServicePublisher ⊠
⌄ 📂 ServicePublisher                              1 Manifest-Version: 1.0
  > 🏛 JRE System Library [JavaSE-1.6]            2 Bundle-ManifestVersion: 2
  > 🏛 Plug-in Dependencies                        3 Bundle-Name: com.mtit.osgi.HrService
  ⌄ 🗂 src                                          4 Bundle-SymbolicName: ServicePublisher
    > 🗂 com.mtit.osgi.servicepublisher           5 Bundle-Version: 1.0.0.qualifier
  ⌄ 📂 META-INF                                     6 Bundle-Activator: com.mtit.osgi.servicepublisher.Activator
      🗎 MANIFEST.MF                                7 Bundle-RequiredExecutionEnvironment: JavaSE-1.6
      📄 build.properties                           8 Import-Package: org.osgi.framework;version="1.3.0"
  > 📂 ServiceSubscriberMarketing                  9 Export-Package: com.mtit.osgi.servicepublisher
                                                  10
```

Service Producer Activator (IT18210002)

```
package com.mtit.osgi.servicepublisher;

import org.osgi.framework.BundleActivator;

public class Activator implements BundleActivator {
    ServiceRegistration personalInformationServiceRegistration;
    ServiceRegistration leaveInformationServiceRegistration;
    ServiceRegistration appraisalInformationServiceRegistration;
    ServiceRegistration allocationInformationServiceRegistration;

    @Override
    public void start(BundleContext context) throws Exception {
        System.out.println("HR Servise publisher has been Started");

        //personal information
        ServicePersonalInformation personalInformationService = new ServicePersonalInformationImpl();
        personalInformationServiceRegistration = context.registerService(ServicePersonalInformation.class.getName(),personalInformationService,null);

        //leave information
        ServiceLeaveInformation leaveInformationService = new ServiceLeaveInformationImpl();
        leaveInformationServiceRegistration = context.registerService(ServiceLeaveInformation.class.getName(),leaveInformationService,null);

        //appraisal information
        ServiceAppraisalInformation appraisalInformationService = new ServiceAppraisalInformationImpl();
        appraisalInformationServiceRegistration = context.registerService(ServiceAppraisalInformation.class.getName(),appraisalInformationService,null);

        //allocation information
        ServiceAllocationInformation allocationInformationService = new ServiceAllocationInformationImpl();
        allocationInformationServiceRegistration = context.registerService(ServiceAllocationInformation.class.getName(),allocationInformationService,null);

    }

    @Override
    public void stop(BundleContext context) throws Exception {
        System.out.println("HR Servise publisher has been Stoped ");
        //personal information service
        personalInformationServiceRegistration.unregister();
        //leave
        leaveInformationServiceRegistration.unregister();
    }
}
```
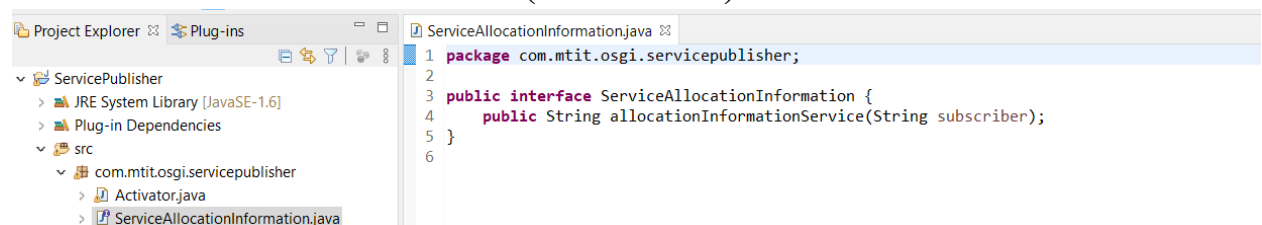
Service Allocation Information. (IT18210002)

```
package com.mtit.osgi.servicepublisher;

public interface ServiceAllocationInformation {
    public String allocationInformationService(String subscriber);
}
```

Service Allocation Information Implementation (IT18210002)

```
package com.mtit.osgi.servicepublisher;

public class ServiceAllocationInformationImpl implements ServiceAllocationInformation {

    public String allocationInformationService(String subscriber) {

        String response = "Excute the allocation information service for : " + subscriber;
        response += "\n================================================";
        response += "\nNo of allocations : = 50 ";
        response += "\nNo of completed allocations : = 20";
        response += "\n No of not allocation : = 1";
        response += "\n";
        return response;
    }
}
```

Service Appraisal Information (IT18210002)

```java
ServiceAppraisalInformation.java ⊠
1  package com.mtit.osgi.servicepublisher;
2
3  public interface ServiceAppraisalInformation {
4      public String appraisalInformationService(String subscriber);
5
6  }
7
```

Service Appraisal Information Implementation (IT18210002)

```java
ServiceAppraisalInformationImpl.java ⊠
1  package com.mtit.osgi.servicepublisher;
2
3  public class ServiceAppraisalInformationImpl implements ServiceAppraisalInformation {
4
5  public String appraisalInformationService(String subscriber) {
6
7          String response = "Excute the appraisal information service for : " + subscriber;
8              response += "\n=================================================";
9              response += "\nNo of appraisal projects : = 70 ";
10             response += "\nNo of appraisal competed : = 50";
11             response += "\n No of not appraisal started : = 10";
12             response += "\n";
13         return response;
14     }
15
16 }
17
```

Service Leave Information (IT18210002)

```java
ServiceLeaveInformation.java ⊠
1  package com.mtit.osgi.servicepublisher;
2
3  public interface ServiceLeaveInformation {
4
5      public String leaveInformationService(String subscriber);
6
7  }
8
```

Service Leave Information Implementation (IT18210002)

```java
ServiceLeaveInformationImpl.java ⊠
1  package com.mtit.osgi.servicepublisher;
2
3  public class ServiceLeaveInformationImpl implements ServiceLeaveInformation{
4
5      @Override
6      public String leaveInformationService(String subscriber) {
7
8          String response = "Excute the personal information service for : " + subscriber;
9              response += "\n=================================================";
10             response += "\nNo of Approved Medical leaves : = 5 ";
11             response += "\nNo of Annual leaves : = 2";
12             response += "\n No of cashual leaves : = 1";
13             response += "\n";
14         return response;
15
16
17     }
18
19 }
20
```

Service Personal Information (IT18210002)

```java
// ServicePersonalInformation.java
1 package com.mtit.osgi.servicepublisher;
2
3 public interface ServicePersonalInformation {
4
5     public String personalInformationService(String subscriber);
6 }
7
```
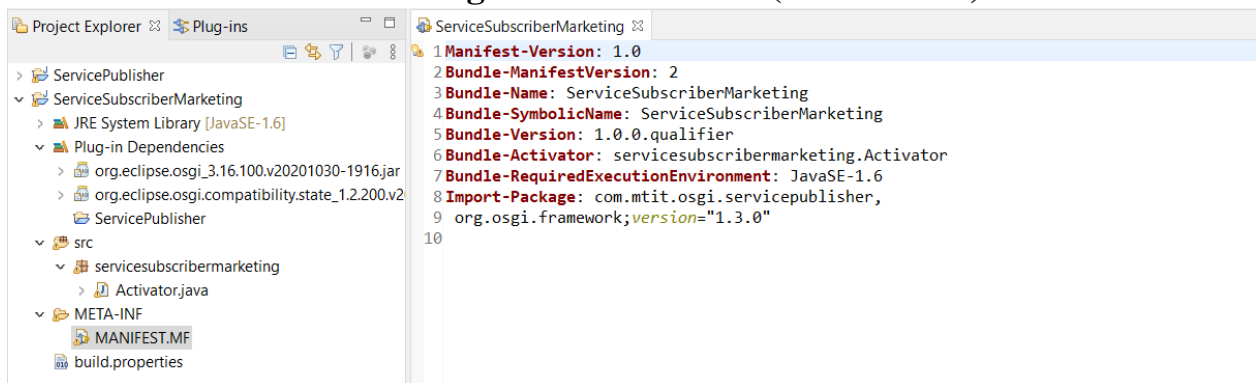
Service Personal Information Implementation (IT18210002)

```java
// ServicePersonalInformationImpl.java
1 package com.mtit.osgi.servicepublisher;
2
3 public class ServicePersonalInformationImpl implements ServicePersonalInformation{
4
5     @Override
6     public String personalInformationService(String subscriber) {
7         String response = "Excute the personal information service for : " + subscriber;
8         response += "\n=================================================";
9         response += "\nNo of requested personal files : = 20 ";
10         response += "\nNo of updated personal files : = 30";
11         response += "\n No of rejected personal files : = 2";
12         response += "\n";
13         return response;
14     }
15
16 }
17
```

Structure of the Service Subscriber Marketing (IT18210002)

- ServiceSubscriberMarketing
  - JRE System Library [JavaSE-1.6]
  - Plug-in Dependencies
    - org.eclipse.osgi_3.16.100.v20201030-1916.jar
    - org.eclipse.osgi.compatibility.state_1.2.200.v2
    - ServicePublisher
  - src
    - servicesubscribermarketing
      - Activator.java
  - META-INF
    - MANIFEST.MF
  - build.properties
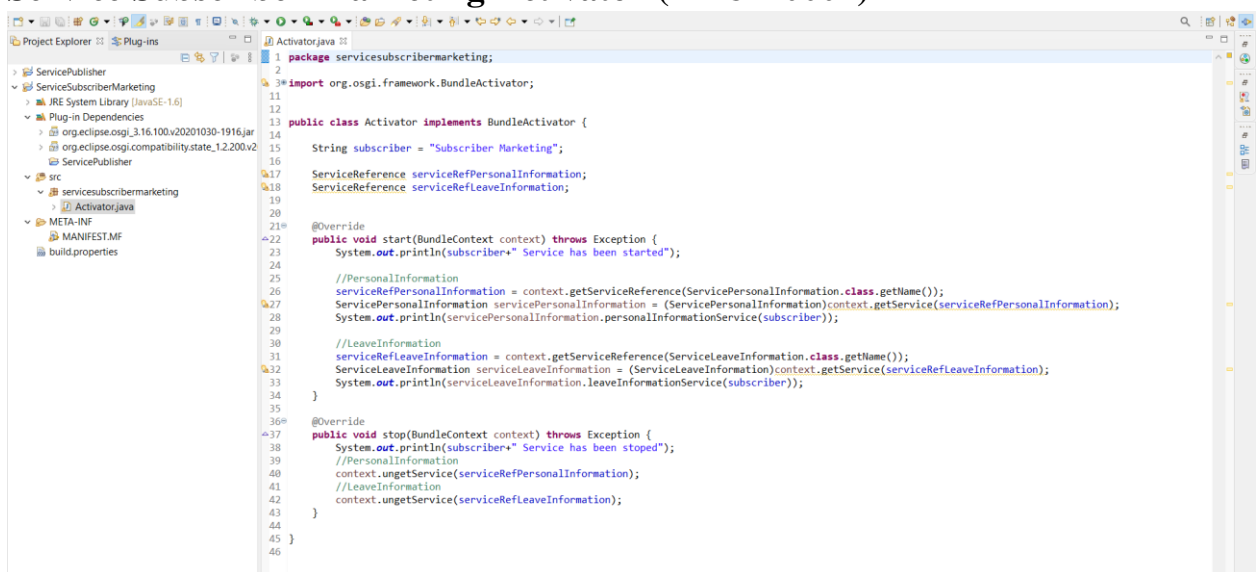
Service Subscriber Marketing MANIFEST.MF (IT18210002)


```
1 Manifest-Version: 1.0
2 Bundle-ManifestVersion: 2
3 Bundle-Name: ServiceSubscriberMarketing
4 Bundle-SymbolicName: ServiceSubscriberMarketing
5 Bundle-Version: 1.0.0.qualifier
6 Bundle-Activator: servicesubscribermarketing.Activator
7 Bundle-RequiredExecutionEnvironment: JavaSE-1.6
8 Import-Package: com.mtit.osgi.servicepublisher,
9  org.osgi.framework;version="1.3.0"
10
```
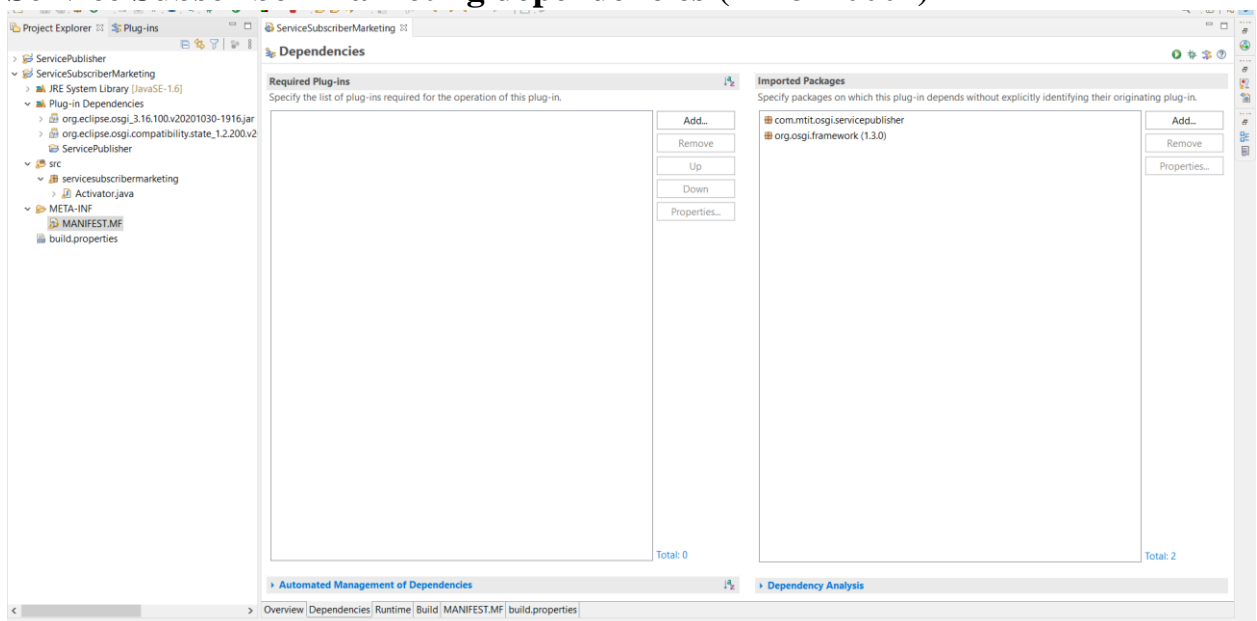
Service Subscriber Marketing Activator (IT18210002)


```
1 package servicesubscribermarketing;
2
3 import org.osgi.framework.BundleActivator;
11
12
13 public class Activator implements BundleActivator {
14
15     String subscriber = "Subscriber Marketing";
16
17     ServiceReference serviceRefPersonalInformation;
18     ServiceReference serviceRefLeaveInformation;
19
20
21     @Override
22     public void start(BundleContext context) throws Exception {
23         System.out.println(subscriber+" Service has been started");
24
25         //PersonalInformation
26         serviceRefPersonalInformation = context.getServiceReference(ServicePersonalInformation.class.getName());
27         ServicePersonalInformation servicePersonalInformation = (ServicePersonalInformation)context.getService(serviceRefPersonalInformation);
28         System.out.println(servicePersonalInformation.personalInformationService(subscriber));
29
30         //LeaveInformation
31         serviceRefLeaveInformation = context.getServiceReference(ServiceLeaveInformation.class.getName());
32         ServiceLeaveInformation serviceLeaveInformation = (ServiceLeaveInformation)context.getService(serviceRefLeaveInformation);
33         System.out.println(serviceLeaveInformation.leaveInformationService(subscriber));
34     }
35
36     @Override
37     public void stop(BundleContext context) throws Exception {
38         System.out.println(subscriber+" Service has been stoped");
39         //PersonalInformation
40         context.ungetService(serviceRefPersonalInformation);
41         //LeaveInformation
42         context.ungetService(serviceRefLeaveInformation);
43     }
44
45 }
46
```

Service Subscriber Marketing dependencies (IT18210002)

Customer Services Section (IT18540536)

ABC Company is providing application, network, and system management services to large enterprises which need third party IT support. So, the heart of the company depends on the customer services. Depend on the customer, ABC company controls the service list which need to allocate to specific customer. ABC Company provides customer services such as,

1. Web hosting service
2. Network service
3. Data storage service
4. Application support service
5. Security & protection service
6. Monitoring & management service

- Company A is one of the old clients of ABC Company. It consumes all services that are provided by ABC Company.
- Company B is one of the new clients of ABC Company. It consumes only the Web hosting service and the Data storage service that is provided by ABC Company.

Proposed solution contains three separate projects. Assume **Service Publisher** project as the ABC Company Customer Services Section which acts as the producer. There are two separate projects for Company A and Company B which acts as consumers. Assume **Service Subscriber A** is for Company A and **Service Subscriber B** is for Company B.

ABC Company (Producer) Services
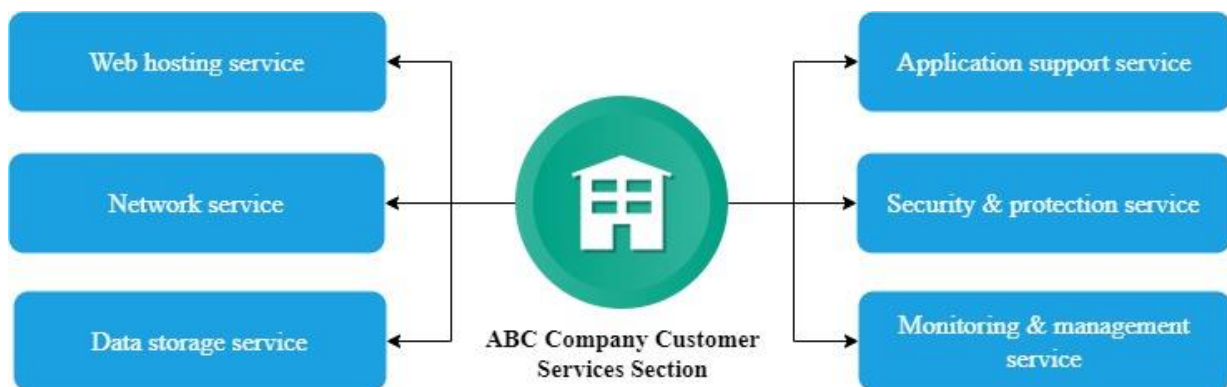
High-level Overview ABC Company Customer Services Section (IT18540536)
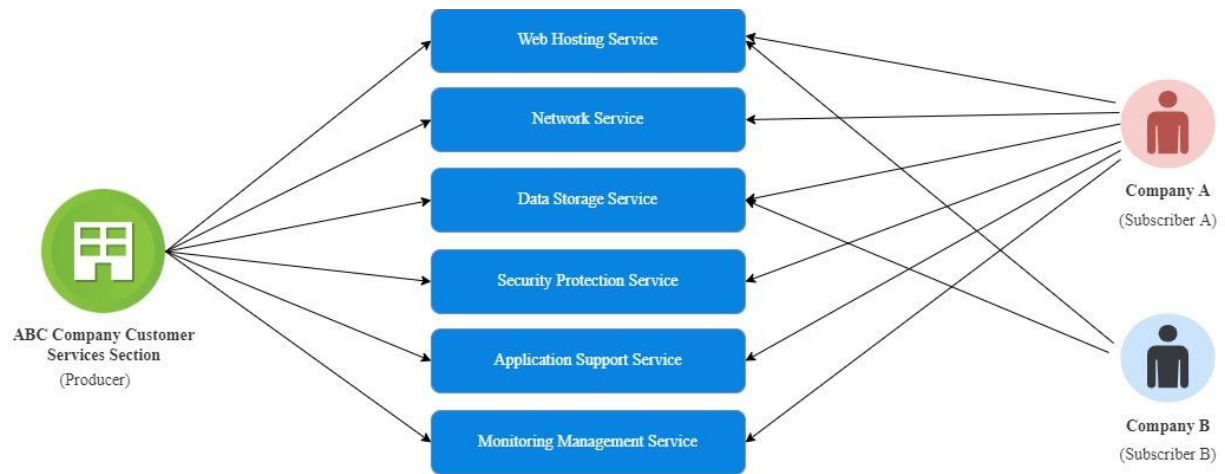
Class Diagram of ABC Company Customer Services Section (IT18540536)
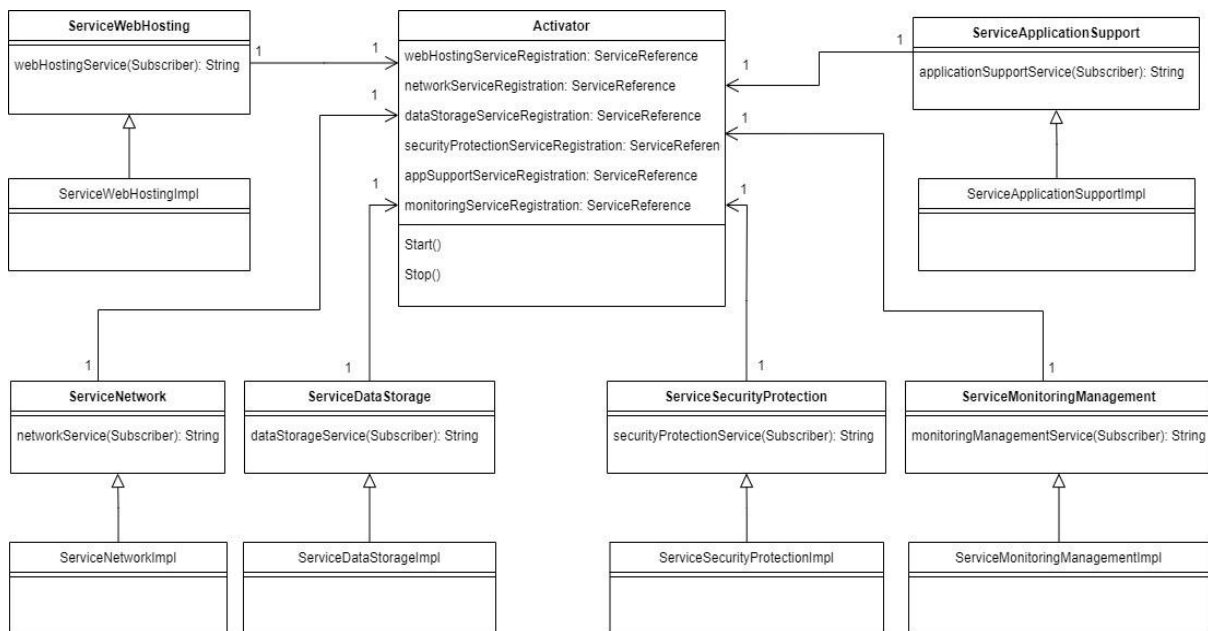
Structure of the project (IT18540536)
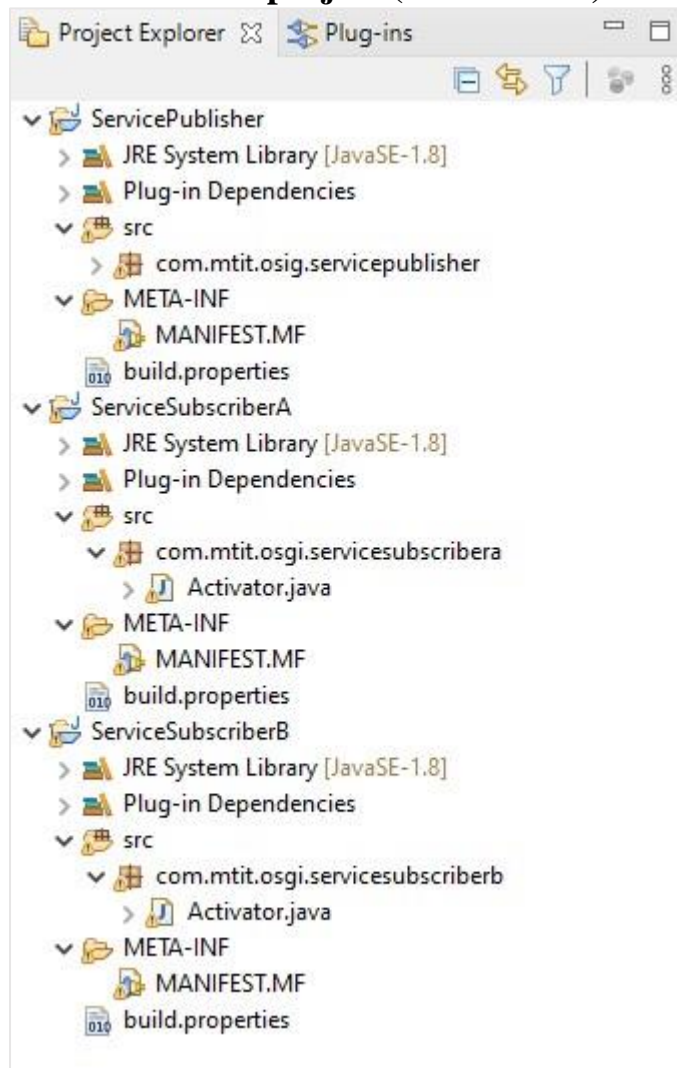
Service Status (IT18540536)

```
Console ✕                                                                    ■
OSGi MTIT Service [OSGi Framework] C:\Program Files\Java\jdk-15.0.2\bin\javaw.exe (Mar 13, 2022, 7:58:26 PM)
id      State       Bundle
0       ACTIVE      org.eclipse.osgi_3.16.200.v20210226-1447
                    Fragments=705
1       ACTIVE      org.eclipse.equinox.simpleconfigurator_1.3.600.v20200721-1308
2       ACTIVE      ServicePublisher_1.0.0.qualifier
3       ACTIVE      ServiceSubscriberA_1.0.0.qualifier
4       ACTIVE      ServiceSubscriberB_1.0.0.qualifier
5       ACTIVE      biz.aQute.bndlib_5.3.0.202102221516
6       ACTIVE      ch.qos.logback.classic_1.2.3.v20200428-2012
                    Fragments=594
7       ACTIVE      ch.qos.logback.core_1.2.3.v20200428-2012
8       RESOLVED    ch.qos.logback.slf4j_1.2.3.v20200428-2012
                    Master=926
9       ACTIVE      com.google.gson_2.8.6.v20201231-1626
10      ACTIVE      com.google.guava_30.1.0.v20210127-2300
11      ACTIVE      com.google.guava_27.1.0.v20190517-1946
12      ACTIVE      com.google.guava_21.0.0.v20170206-1425
13      ACTIVE      com.google.javascript_0.0.20160315.v20161124-1903
14      ACTIVE      com.google.protobuf_2.4.0.v201105131100
15      ACTIVE      com.ibm.icu_67.1.0.v20200706-1749
16      ACTIVE      com.jcraft.jsch_0.1.55.v20190404-1902
17      ACTIVE      com.sun.el_2.2.0.v201303151357
18      ACTIVE      com.sun.jna_5.6.0.v20200716-0148
19      ACTIVE      com.sun.jna_4.5.1.v20190425-1842
20      ACTIVE      com.sun.jna.platform_5.6.0.v20200722-0209
21      ACTIVE      com.sun.jna.platform_4.5.1.v20190425-1842
22      ACTIVE      com.sun.xml.bind_2.3.3.v20201118-1818
23      ACTIVE      com.sun.xml.bind_2.2.0.v20201118-1845
24      ACTIVE      jakarta.el_4.0.0.v20210105-0527
25      ACTIVE      jakarta.servlet_5.0.0.v20210105-0527
26      ACTIVE      jakarta.servlet.jsp_3.0.0.v20210105-0527
27      ACTIVE      jakarta.xml.bind_2.3.3.v20201118-1818
```

Stop Subscriber -A Service (IT18540536)

```
Console ✕                                                                    ■
OSGi MTIT Service [OSGi Framework] C:\Program Files\Java\jdk-15.0.2\bin\javaw.exe (Mar 13, 2022, 7:58:26 PM)

osgi> stop 3
Subscriber A has been terminated...
```

Stop Subscriber-B Service (IT18540536)

```
Console ✕                                                                    ■
OSGi MTIT Service [OSGi Framework] C:\Program Files\Java\jdk-15.0.2\bin\javaw.exe (Mar 13, 2022, 7:58:26 PM)

osgi> stop 4
Subscriber B has been terminated...
```

Stop Publisher Service (IT18540536)

```
Console ✕                                                                    ■
OSGi MTIT Service [OSGi Framework] C:\Program Files\Java\jdk-15.0.2\bin\javaw.exe (Mar 13, 2022, 7:58:26 PM)

osgi> stop 2
publisher has been terminated...
```

Start Publisher Service (IT18540536)

```
Overview  Dependencies  Runtime  Build  MANIFEST.MF  build.properties

Console ⌧                                                        ■ ✖ ⊗
OSGi MTIT Service [OSGi Framework] C:\Program Files\Java\jdk-15.0.2\bin\javaw.exe  (Mar 13, 2022, 7:58:26 PM)

osgi> start 2
publisher has been started...
osgi>
osgi>
```

Start Subscriber -A Service (IT18540536)

```
Overview  Dependencies  Runtime  Build  MANIFEST.MF  build.properties

Console ⌧                                                        ■
OSGi MTIT Service [OSGi Framework] C:\Program Files\Java\jdk-15.0.2\bin\javaw.exe  (Mar 13, 2022, 7:58:26 PM)
osgi>
osgi> start 3
Subscriber A service has been started...
Excute the web hosting service for : Subscriber A
=========================================================
No of hosting spaces : 3
Total storage : 200TB
No of running hosts : 6
No of terminated hosts : 2

Excute the network service for : Subscriber A
=========================================================
Speed : 100Mbps
Reported error count(Last hour) : 2
Timeouts : 5
Unauthorized access trys : 1
Firewall status : 100%

Excute the data storage service for : Subscriber A
=========================================================
No of running databases : 5
No of terminated databases : 1
Identified error count in last hour : 0

Excute the application support service for : Subscriber A
=========================================================
No of running applications : 10
No of stopped applications : 1
Last hour reported cases count : 0

Excute the security protection service for : Subscriber A
---------------------------------------------------------
```

```
=========================================================
Unauthorized access trys : 1
DDOS count : 0
Malware count : 0
Last virus scan : 2022.03.10

Excute the monitoring management service for : Subscriber A
=========================================================
No of application errors(Last hour) : 2
No of database errors(Last hour) : 0
No of api errors(Last hour) : 2
No of reported jira cases(Last hour) : 1

osgi>
```

Start Subscriber-B Service (IT18540536)

```
osgi> start 4
Subscriber B service has been started...
Excute the web hosting service for : Subscriber B
===============================================================
No of hosting spaces : 3
Total storage : 200TB
No of running hosts : 6
No of terminated hosts : 2

Excute the data storage service for : Subscriber B
===============================================================
No of running databases : 5
No of terminated databases : 1
Identified error count in last hour : 0

osgi>
```

Structure of the Service Producer project (IT18540536)

Service Producer MANIFEST.MF (IT18540536)


```
1 Manifest-Version: 1.0
2 Bundle-ManifestVersion: 2
3 Bundle-Name: ServicePublisher
4 Bundle-SymbolicName: ServicePublisher
5 Bundle-Version: 1.0.0.qualifier
6 Bundle-Activator: com.mtit.osig.servicepublisher.Activator
7 Import-Package: org.osgi.framework;version="1.3.0"
8 Export-Package: com.mtit.osig.servicepublisher
9 Bundle-RequiredExecutionEnvironment: JavaSE-1.8
10
```

Service Producer Activator (IT18540536)


```
1 package com.mtit.osig.servicepublisher;
2
3 import org.osgi.framework.BundleActivator;
6
7 public class Activator implements BundleActivator {
8
9     ServiceRegistration webHostingServiceRegistration;
10    ServiceRegistration networkServiceRegistration;
11    ServiceRegistration dataStorageServiceRegistration;
12    ServiceRegistration appSupportServiceRegistration;
13    ServiceRegistration securityProtectionServiceRegistration;
14    ServiceRegistration monitoringServiceRegistration;
15
16    @Override
17    public void start(BundleContext context) throws Exception {
18        System.out.println("publisher has been started...");
19        //web hosting
20        ServiceWebHosting webHostingService = new ServiceWebHostingImpl();
21        webHostingServiceRegistration = context.registerService(
22            ServiceWebHosting.class.getName(), webHostingService, null);
23        //network
24        ServiceNetwork networkService = new ServiceNetworkImpl();
25        networkServiceRegistration = context.registerService(
26            ServiceNetwork.class.getName(), networkService, null);
27        //data storage
28        ServiceDataStorage dataStorageService = new ServiceDataStorageImpl();
29        dataStorageServiceRegistration = context.registerService(
30            ServiceDataStorage.class.getName(), dataStorageService, null);
31        //application support
32        ServiceApplicationSupport applicationSupportService = new ServiceApplicationSupportImpl();
33        appSupportServiceRegistration = context.registerService(
34            ServiceApplicationSupport.class.getName(), applicationSupportService, null);
35        //security protection
36        ServiceSecurityProtection securityProtectionService = new ServiceSecurityProtectionImpl();
37        securityProtectionServiceRegistration = context.registerService(
38            ServiceSecurityProtection.class.getName(), securityProtectionService, null);
39        //monitoring management
40        ServiceMonitoringManagement monitoringManagementService = new ServiceMonitoringManagementImpl();
41        monitoringServiceRegistration = context.registerService(
```

```
42            ServiceMonitoringManagement.class.getName(), monitoringManagementService, null);
43    }
44
45    @Override
46    public void stop(BundleContext context) throws Exception {
47        System.out.println("publisher has been terminated...");
48        //web hosting
49        webHostingServiceRegistration.unregister();
50        //network
51        networkServiceRegistration.unregister();
52        //data storage
53        dataStorageServiceRegistration.unregister();
54        //application support
55        appSupportServiceRegistration.unregister();
56        //security protection
57        securityProtectionServiceRegistration.unregister();
58        //monitoring management
59        monitoringServiceRegistration.unregister();
60
61    }
62
63 }
64
```

Service Application Support (IT18540536)

```java
package com.mtit.osig.servicepublisher;

public interface ServiceApplicationSupport {

    public String applicationSupportService(String subscriber);

}
```

```java
package com.mtit.osig.servicepublisher;

public class ServiceApplicationSupportImpl implements ServiceApplicationSupport {

    @Override
    public String applicationSupportService(String subscriber) {
        String response = "Excute the application support service for : " + subscriber;
        response += "\n========================================================";
        response += "\nNo of running applications : 10";
        response += "\nNo of stopped applications : 1";
        response += "\nLast hour reported cases count : 0";
        response += "\n";
        return response;
    }

}
```

Service Data Storage (IT18540536)

```java
ServiceDataStorage.java ⊠
1  package com.mtit.osig.servicepublisher;
2
3  public interface ServiceDataStorage {
4
5      public String dataStorageService(String subscriber);
6
7  }
8
```

```java
ServiceDataStorageImpl.java ⊠
1  package com.mtit.osig.servicepublisher;
2
3  public class ServiceDataStorageImpl implements ServiceDataStorage {
4
5      @Override
6      public String dataStorageService(String subscriber) {
7          String response = "Excute the data storage service for : " + subscriber;
8          response += "\n============================================================";
9          response += "\nNo of running databases : 5";
10         response += "\nNo of terminated databases : 1";
11         response += "\nIdentified error count in last hour : 0";
12         response += "\n";
13         return response;
14     }
15
16 }
17
```

Service Monitoring Management (IT18540536)

```java
ServiceMonitoringManagement.java

1  package com.mtit.osig.servicepublisher;
2
3  public interface ServiceMonitoringManagement {
4
5      public String monitoringManagementService(String subscriber);
6
7  }
8
```

```java
ServiceMonitoringManagementImpl.java

1  package com.mtit.osig.servicepublisher;
2
3  public class ServiceMonitoringManagementImpl implements ServiceMonitoringManagement {
4
5      @Override
6      public String monitoringManagementService(String subscriber) {
7          String response = "Excute the monitoring management service for : " + subscriber;
8          response += "\n=============================================================";
9          response += "\nNo of application errors(Last hour) : 2";
10         response += "\nNo of database errors(Last hour) : 0";
11         response += "\nNo of api errors(Last hour) : 2";
12         response += "\nNo of reported jira cases(Last hour) : 1";
13         response += "\n";
14         return response;
15     }
16
17 }
18
```

Service Network (IT18540536)

```java
ServiceNetwork.java

 1  package com.mtit.osig.servicepublisher;
 2
 3  public interface ServiceNetwork {
 4
 5      public String networkService(String subscriber);
 6
 7  }
 8
```

```java
ServiceNetworkImpl.java

 1  package com.mtit.osig.servicepublisher;
 2
 3  public class ServiceNetworkImpl implements ServiceNetwork {
 4
 5      @Override
 6      public String networkService(String subscriber) {
 7          String response = "Excute the network service for : " + subscriber;
 8          response += "\n=========================================================";
 9          response += "\nSpeed : 100Mbps";
10          response += "\nReported error count(Last hour) : 2";
11          response += "\nTimeouts : 5";
12          response += "\nUnauthorized access trys : 1";
13          response += "\nFirewall status : 100%";
14          response += "\n";
15          return response;
16      }
17
18  }
19
```

Service Security Protection (IT18540536)

```java
ServiceSecurityProtection.java ⋈
1  package com.mtit.osig.servicepublisher;
2
3  public interface ServiceSecurityProtection {
4
5      public String securityProtectionService(String subscriber);
6
7  }
8
```

```java
ServiceSecurityProtectionImpl.java ⋈
1  package com.mtit.osig.servicepublisher;
2
3  public class ServiceSecurityProtectionImpl implements ServiceSecurityProtection {
4
5      @Override
6      public String securityProtectionService(String subscriber) {
7          String response = "Excute the security protection service for : " + subscriber;
8          response += "\n=========================================================";
9          response += "\nUnauthorized access trys : 1";
10         response += "\nDDOS count : 0";
11         response += "\nMalware count : 0";
12         response += "\nLast virus scan : 2022.03.10";
13         response += "\n";
14         return response;
15     }
16
17 }
18
```

Service Web Hosting (IT18540536)

```java
ServiceWebHosting.java ⊠
1  package com.mtit.osig.servicepublisher;
2
3  public interface ServiceWebHosting {
4
5      public String webHostingService(String subscriber);
6
7  }
8
```

```java
ServiceWebHostingImpl.java ⊠
1  package com.mtit.osig.servicepublisher;
2
3  public class ServiceWebHostingImpl implements ServiceWebHosting {
4
5      @Override
6      public String webHostingService(String subscriber) {
7          String response = "Excute the web hosting service for : " + subscriber;
8          response += "\n=========================================================";
9          response += "\nNo of hosting spaces : 3";
10         response += "\nTotal storage : 200TB";
11         response += "\nNo of running hosts : 6";
12         response += "\nNo of terminated hosts : 2";
13         response += "\n";
14         return response;
15     }
16
17 }
18
```

Structure of the Service Subscriber-A project (IT18540536)

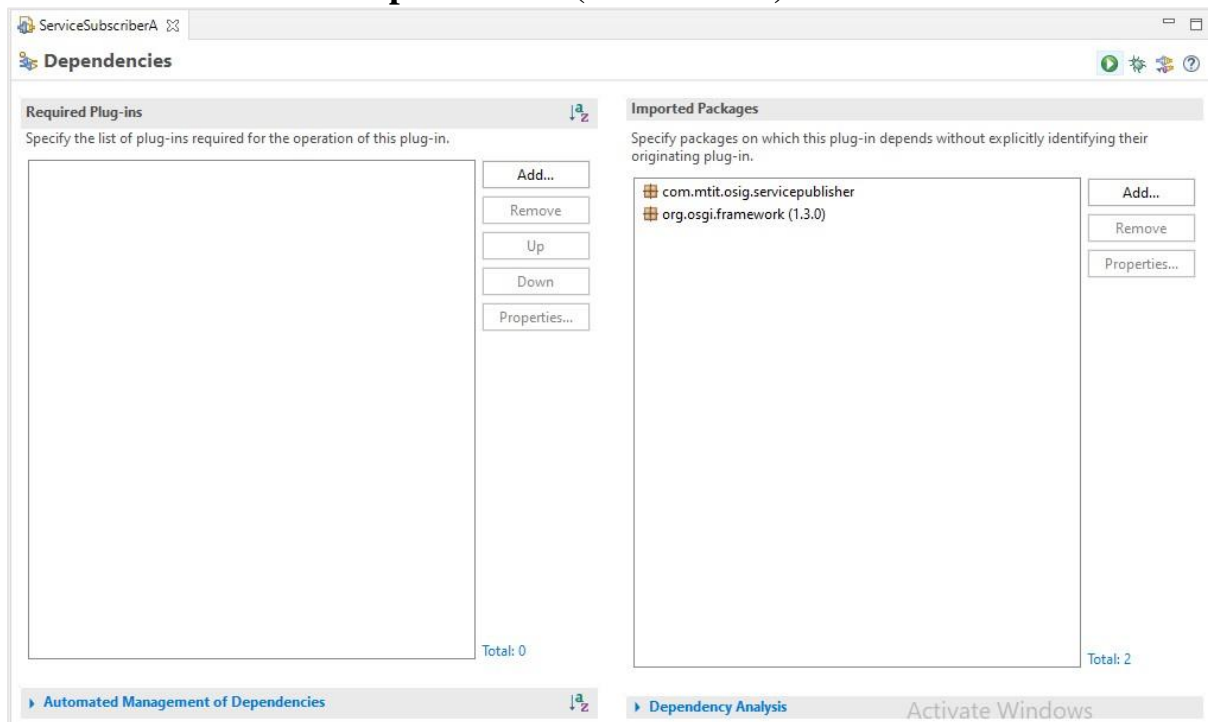## Service Subscriber-A MANIFEST.MF (IT18540536)
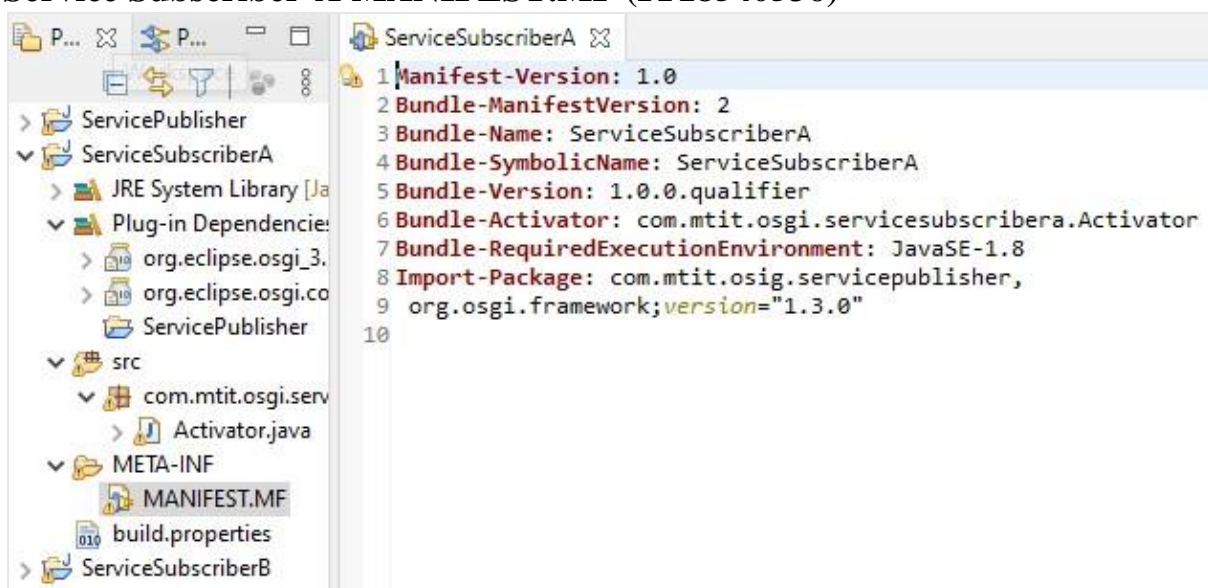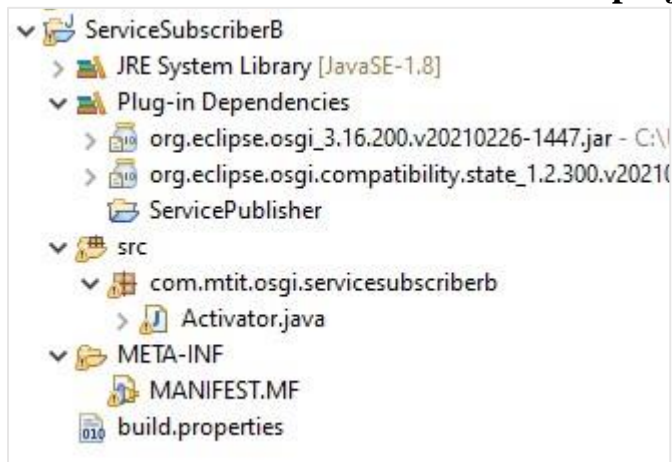
Service Subscriber-A Activator (IT18540536)

```java
package com.mtit.osgi.servicesubscribera;

import com.mtit.osig.servicepublisher.ServiceApplicationSupport;

public class Activator implements BundleActivator {

    String subscriber = "Subscriber A";

    ServiceReference serviceRefWebHosting;
    ServiceReference serviceRefNetwork;
    ServiceReference serviceRefDataStorage;
    ServiceReference serviceRefAppSupport;
    ServiceReference serviceRefProtection;
    ServiceReference serviceRefMonitoring;

    @Override
    public void start(BundleContext context) throws Exception {
        System.out.println(subscriber+" service has been started...");
        //web hosting
        serviceRefWebHosting = context.getServiceReference(ServiceWebHosting.class.getName());
        ServiceWebHosting serviceWebHosting = (ServiceWebHosting)context.
                getService(serviceRefWebHosting);
        System.out.println(serviceWebHosting.webHostingService(subscriber));
        //network
        serviceRefNetwork = context.getServiceReference(ServiceNetwork.class.getName());
        ServiceNetwork serviceNetwork = (ServiceNetwork)context.
                getService(serviceRefNetwork);
        System.out.println(serviceNetwork.networkService(subscriber));
        //data storage
        serviceRefDataStorage = context.getServiceReference(ServiceDataStorage.class.getName());
        ServiceDataStorage serviceDataStorage = (ServiceDataStorage)context.
                getService(serviceRefDataStorage);
        System.out.println(serviceDataStorage.dataStorageService(subscriber));
        //application support
        serviceRefAppSupport = context.getServiceReference(ServiceApplicationSupport.class.getName());
        ServiceApplicationSupport serviceAppSupport = (ServiceApplicationSupport)context.
                getService(serviceRefAppSupport);
        System.out.println(serviceAppSupport.applicationSupportService(subscriber));
        //security protection
        serviceRefProtection = context.getServiceReference(ServiceSecurityProtection.class.getName());
        ServiceSecurityProtection serviceProtection = (ServiceSecurityProtection)context.
                getService(serviceRefProtection);
        System.out.println(serviceProtection.securityProtectionService(subscriber));
        //Monitoring Management
        serviceRefMonitoring = context.getServiceReference(ServiceMonitoringManagement.class.getName());
        ServiceMonitoringManagement serviceMonitoringManagement = (ServiceMonitoringManagement)context.
                getService(serviceRefMonitoring);
        System.out.println(serviceMonitoringManagement.monitoringManagementService(subscriber));

    }

    @Override
    public void stop(BundleContext context) throws Exception {
        System.out.println(subscriber+" has been terminated...");
        //web hosting
        context.ungetService(serviceRefWebHosting);
        //network
        context.ungetService(serviceRefNetwork);
        //data storage
        context.ungetService(serviceRefDataStorage);
        //app support
        context.ungetService(serviceRefAppSupport);
        //security protection
        context.ungetService(serviceRefProtection);
        //monitoring management
        context.ungetService(serviceRefMonitoring);
    }
}
```
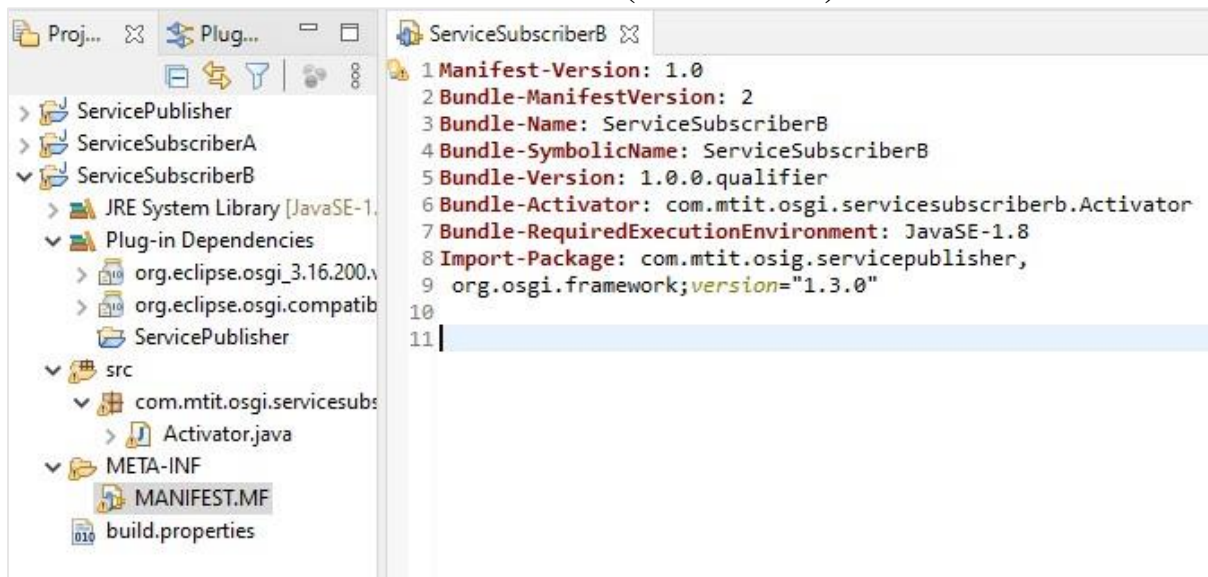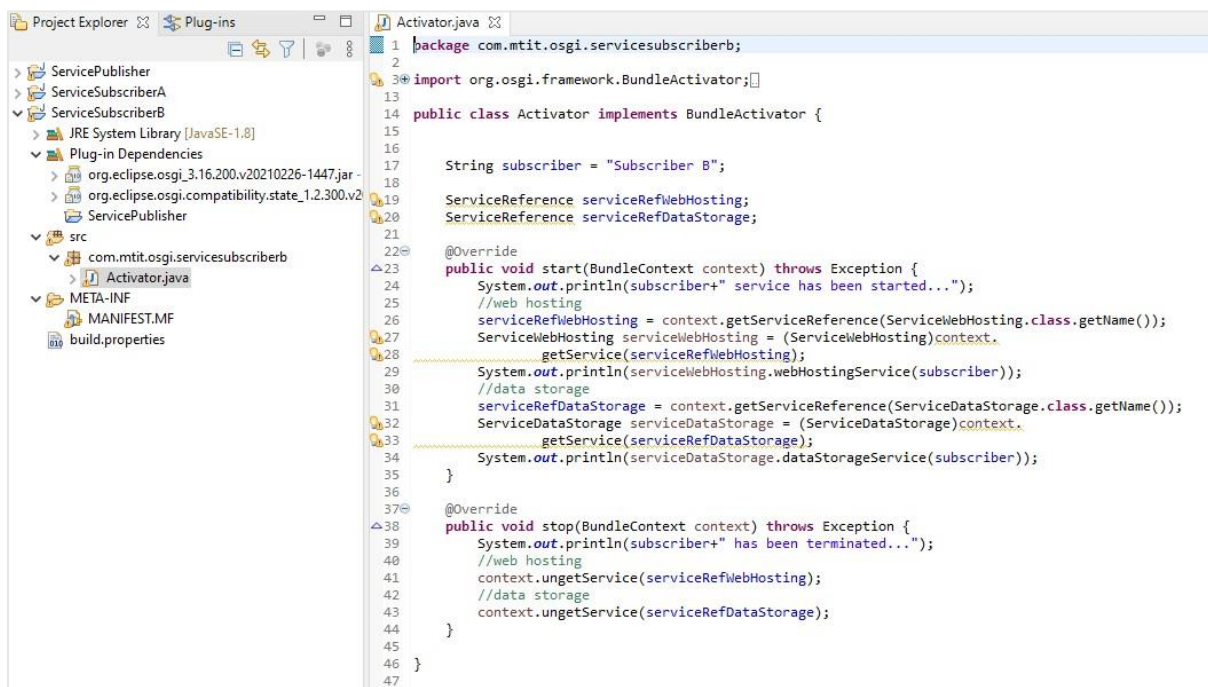
Service Subscriber-A dependencies (IT18540536)

Structure of the Service Subscriber-B project (IT18540536)

Service Subscriber-B MANIFEST.MF (IT18540536)

```
1 Manifest-Version: 1.0
2 Bundle-ManifestVersion: 2
3 Bundle-Name: ServiceSubscriberB
4 Bundle-SymbolicName: ServiceSubscriberB
5 Bundle-Version: 1.0.0.qualifier
6 Bundle-Activator: com.mtit.osgi.servicesubscriberb.Activator
7 Bundle-RequiredExecutionEnvironment: JavaSE-1.8
8 Import-Package: com.mtit.osig.servicepublisher,
9  org.osgi.framework;version="1.3.0"
10
11
```

Service Subscriber- B Activator (IT18540536)

```java
1   package com.mtit.osgi.servicesubscriberb;
2
3   import org.osgi.framework.BundleActivator;
13
14  public class Activator implements BundleActivator {
15
16
17      String subscriber = "Subscriber B";
18
19      ServiceReference serviceRefWebHosting;
20      ServiceReference serviceRefDataStorage;
21
22      @Override
23      public void start(BundleContext context) throws Exception {
24          System.out.println(subscriber+" service has been started...");
25          //web hosting
26          serviceRefWebHosting = context.getServiceReference(ServiceWebHosting.class.getName());
27          ServiceWebHosting serviceWebHosting = (ServiceWebHosting)context.
28              getService(serviceRefWebHosting);
29          System.out.println(serviceWebHosting.webHostingService(subscriber));
30          //data storage
31          serviceRefDataStorage = context.getServiceReference(ServiceDataStorage.class.getName());
32          ServiceDataStorage serviceDataStorage = (ServiceDataStorage)context.
33              getService(serviceRefDataStorage);
34          System.out.println(serviceDataStorage.dataStorageService(subscriber));
35      }
36
37      @Override
38      public void stop(BundleContext context) throws Exception {
39          System.out.println(subscriber+" has been terminated...");
40          //web hosting
41          context.ungetService(serviceRefWebHosting);
42          //data storage
43          context.ungetService(serviceRefDataStorage);
44      }
45
46  }
47
```

Service Subscriber-B dependencies (IT18540536)

Marketing Section (IT18218572)

According to the OSGi framework, we created an ABC software Company to provide the IT solution for the customer, so the Marketing Service improves for we publish our requirement in our Portal, and it hires IT, HR and Sales. In our system product and consumer are Marketing Sector and Undergraduate. As the consumer needs to submit an application there, that portal.
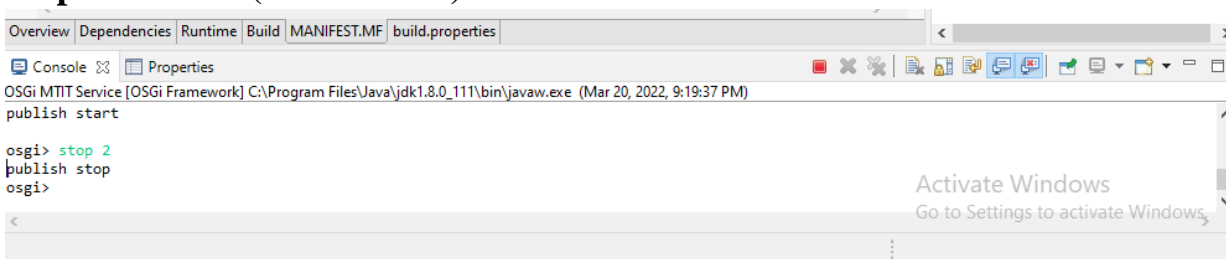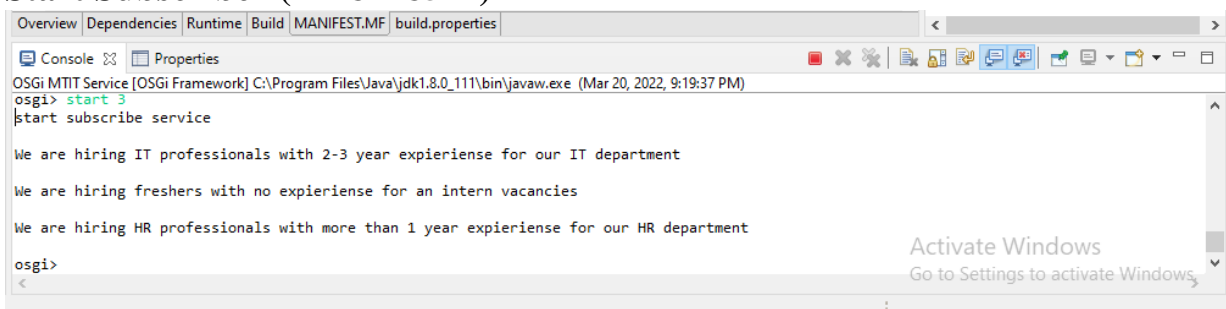
.
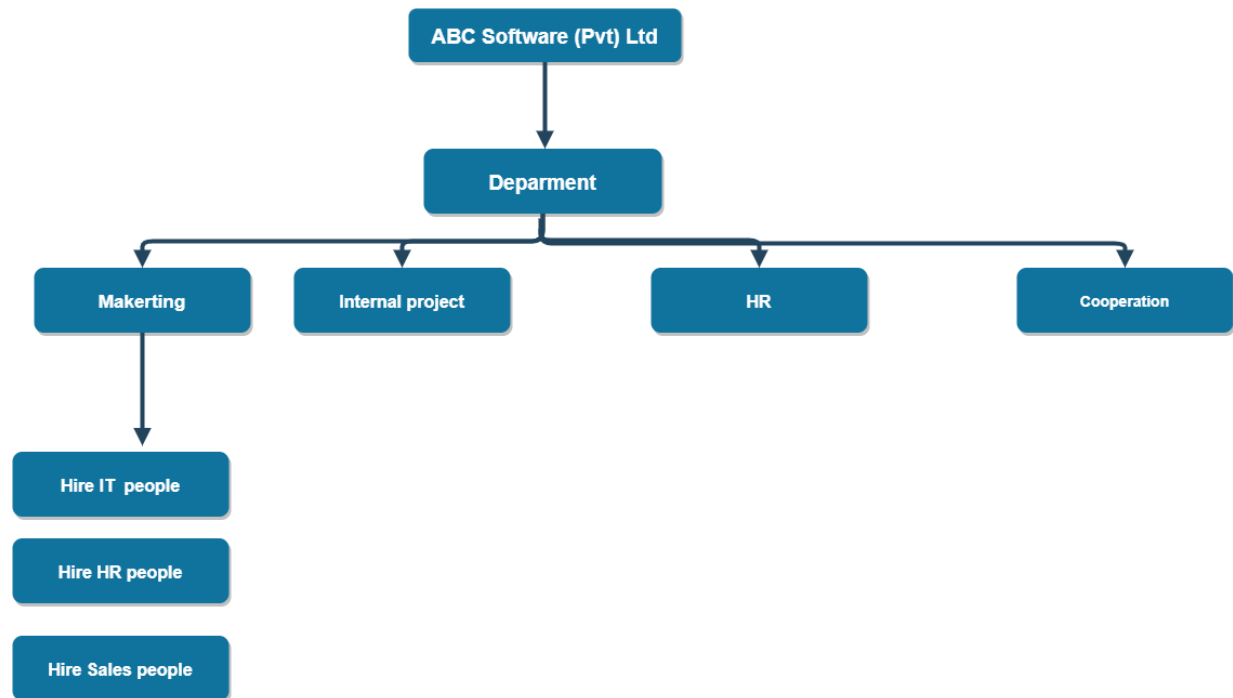
Service Status (IT18218572)
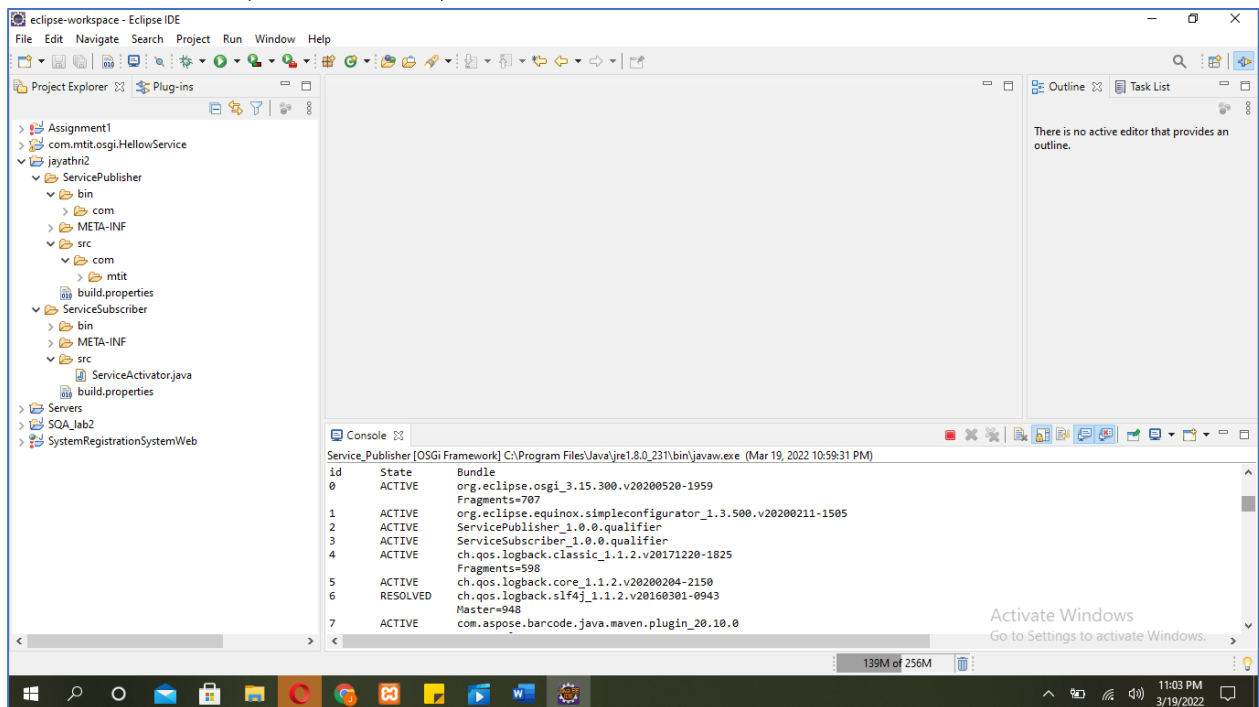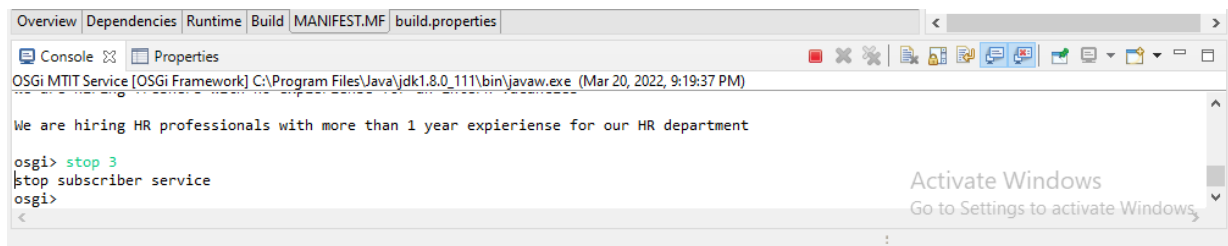

Start Publisher (IT18218572)
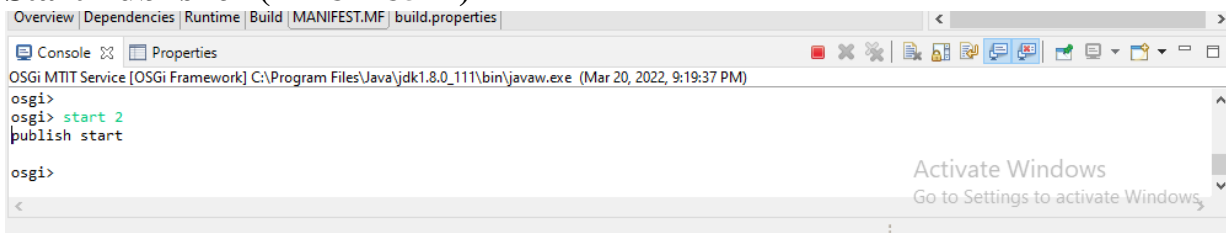
Stop Publisher (IT18218572)

Start Subscriber (IT18218572)

Stop Subscriber (IT18218572)

Service Producer MANIFEST.MF (IT18218572)

Service Producer Activator (IT18218572)

```java
1  package com.mtit.service;
2  import org.osgi.framework.BundleActivator;
3  import org.osgi.framework.BundleContext;
4  import org.osgi.framework.ServiceRegistration;
5
6  public class ServiceActivator implements BundleActivator {
7
8      ServiceRegistration hiringServiceRegistration;
9
10     @Override
11     public void start(BundleContext context) throws Exception {
12         System.out.println("publish start");
13         ServiceHiring hiringService = new ServiceHiringImpl();
14         hiringServiceRegistration = context.registerService(
15                 ServiceHiring.class.getName(), hiringService, null);
16     }
17
18     @Override
19     public void stop(BundleContext context) throws Exception {
20         System.out.println("publish stop");
21         hiringServiceRegistration.unregister();
22
23     }
24
25 }
26
```

Service Hiring (IT18218572)

```java
1  ckage com.mtit.service;
2
3  blic class ServiceHiringImpl implements ServiceHiring {
4
5      public String hiringsInIT() {
6          return "\nWe are hiring IT professionals with 2-3 year expieriense for our IT department";
7      }
8
9      public String hiringsInHR() {
10         return "\nWe are hiring freshers with no expieriense for an intern vacancies";
11     }
12
13     public String hiringsInSales() {
14         return "\nWe are hiring HR professionals with more than 1 year expieriense for our HR depal
15     }
16
17
18
```

Console Properties

OSGi MTIT Service [OSGi Framework] C:\Program Files\Java\jdk1.8.0_111\bin\javaw.exe (Mar 19, 2022, 10:19:01 PM)

Service Subscriber MANIFEST.MF (IT18218572)


```
1 Manifest-Version: 1.0
2 Bundle-ManifestVersion: 2
3 Bundle-SymbolicName: ServiceSubscriber
4 Bundle-Version: 1.0.0.qualifier
5 Bundle-Activator: ServiceActivator
6 Bundle-RequiredExecutionEnvironment: JavaSE-1.8
7 Automatic-Module-Name: ServiceSubscriber
8 Import-Package: com.mtit.service,
9  org.osgi.framework;version="1.3.0"
10 Bundle-ActivationPolicy: lazy
11 Bundle-Name: ServiceSubscriber
12
```

Service Subscriber Activator (IT18218572)


```java
1 import org.osgi.framework.BundleActivator;
2 import org.osgi.framework.BundleContext;
3 import org.osgi.framework.ServiceReference;
4
5 import com.mtit.service.ServiceHiring;
6
7 public class ServiceActivator implements BundleActivator{
8         ServiceReference serviceReference;
9
10         @Override
11         public void start(BundleContext context) throws Exception {
12             System.out.println("start subscribe service");
13             serviceReference = context.getServiceReference(ServiceHiring.class.getName());
14             ServiceHiring serviceHiring = (ServiceHiring)context.getService(serviceReference);
15             System.out.println(serviceHiring.hiringsInIT());
16             System.out.println(serviceHiring.hiringsInHR());
17             System.out.println(serviceHiring.hiringsInSales());
18         }
19
20         @Override
21         public void stop(BundleContext context) throws Exception {
22             System.out.println("stop subscriber service");
23             context.ungetService(serviceReference);
24         }
25
26 }
27
```

36

Internal Projects Section (IT18062120)

This is really a plug-in project where there are several plug-ins which will connect together, also, this has a producer to produce services and there are consumers who are consuming the services and make the services available to the user. We have done an IT Product development System for ABC Company according to OSGi framework. Marketing, Internal Project service, HR and providing Service are the producers/services of this system.
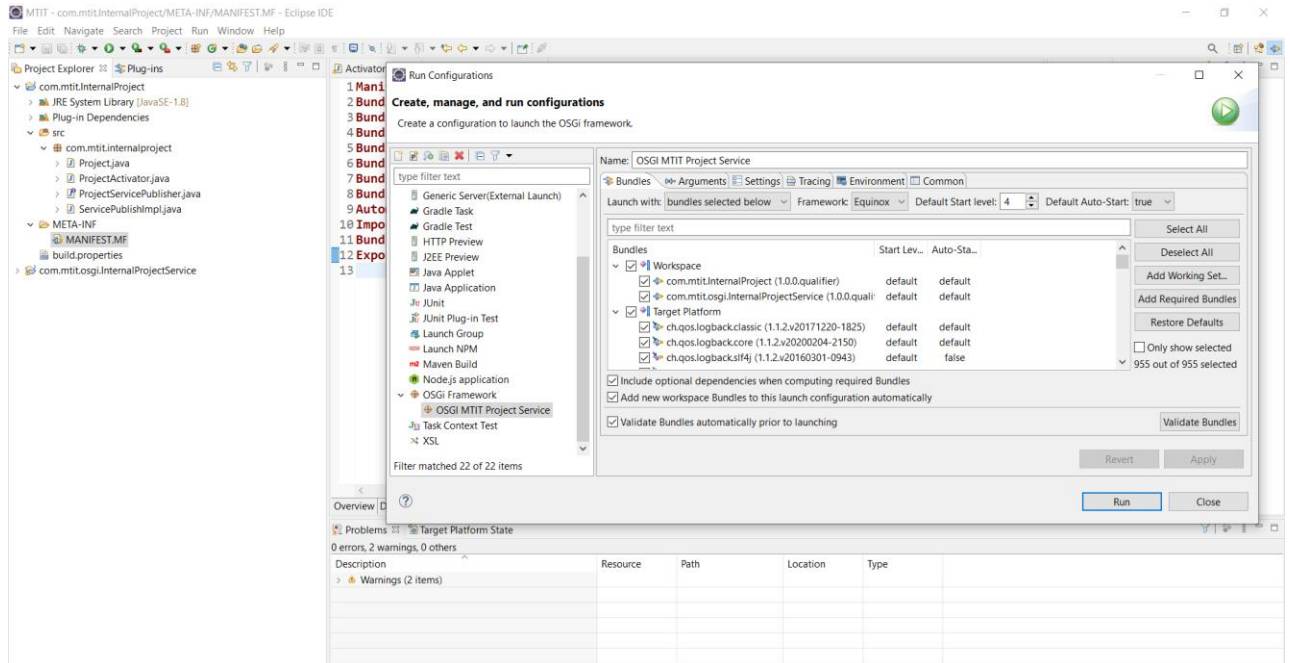
MANIFEST IMPLEMENTATION (IT18062120)

Internal Project Service (Producer/Service)
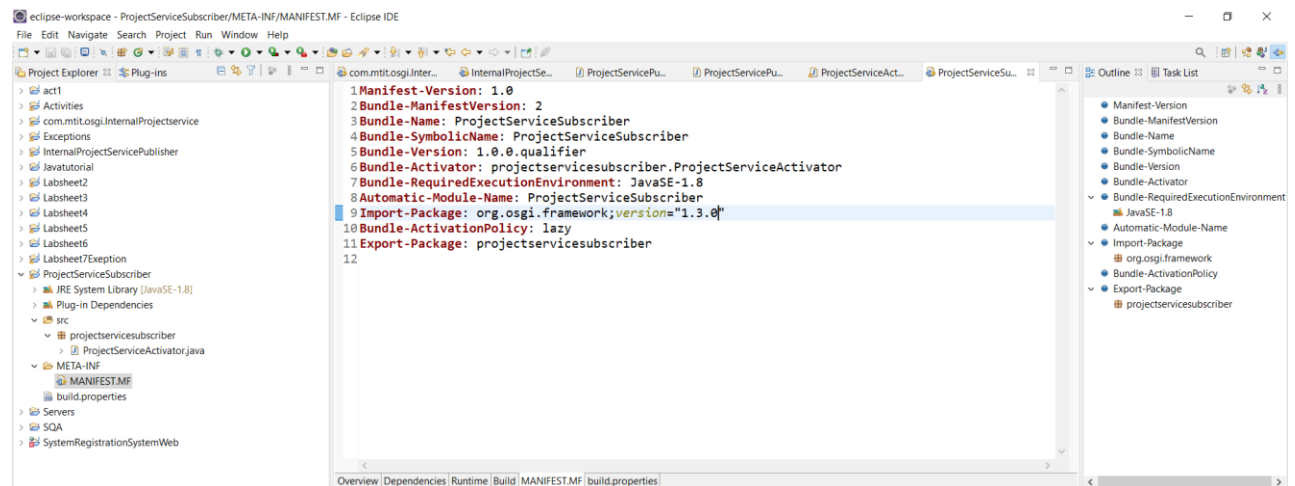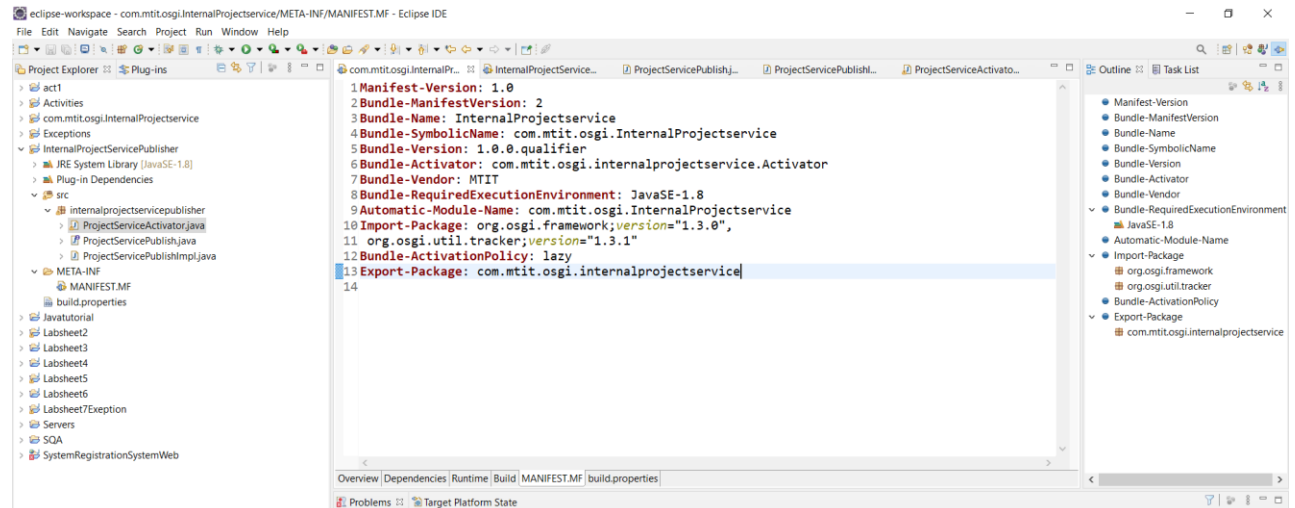
BUNDLES (IT18062120)

Internal Project Service

BEHAVIOR AND OUTPUTS (IT18062120)