

SE 3030 – Software Architecture

Assignment 01 – Service Publisher Subscriber OSGi framework

Submitted by:

|  |  |
| --- | --- |
| Registration number | Name |
| IT21786464 | Nisal V.P.T. |
| IT21559136 | Wijethunge W.L.D.N.N. |
| IT21271328 | Withana J.W.J. |
| IT21704970 | Lokuge P.P. |

Table of Contents

[Introduction 4](#_Toc161348320)

[Student Management – IT21704970 4](#_Toc161348321)

[Student Publisher 4](#_Toc161348322)

[Student Subscriber 4](#_Toc161348323)

[Communication between Publisher and Subscriber: 4](#_Toc161348324)

[User interaction: 4](#_Toc161348325)

[Diagrammatic explanation 5](#_Toc161348326)

[Publisher Bundle Manifest: 6](#_Toc161348327)

[During run time the package is exported: 6](#_Toc161348328)

[Subscriber Bundle Manifest: 7](#_Toc161348329)

[Publisher is added/imported as a dependency in the Subscriber bundle: 7](#_Toc161348330)

[Sample screenshots of its behavior &outputs 7](#_Toc161348331)

[Add Functionality – Add new students to the system 8](#_Toc161348332)

[View All Student details 8](#_Toc161348333)

[Get specific student details . 9](#_Toc161348334)

[Validation – If the staff ID does not exist. 9](#_Toc161348335)

[Remove a student from the system 10](#_Toc161348336)

[Update the existing student details 10](#_Toc161348337)

[Get the subject details when enter the grade 11](#_Toc161348338)

[Enrolling students in the subjects 11](#_Toc161348339)

[Teacher Management – IT21271328 12](#_Toc161348340)

[Teacher Publisher 12](#_Toc161348341)

[Teacher Subscriber 12](#_Toc161348342)

[Communication between Publisher and Subscriber: 12](#_Toc161348343)

[User interaction: 12](#_Toc161348344)

[Diagrammatic explanation 13](#_Toc161348345)

[Publisher Bundle Manifest: 14](#_Toc161348346)

[During run time the package is exported: 14](#_Toc161348347)

[Subscriber Bundle Manifest: 14](#_Toc161348348)

[Publisher is added/imported as a dependency in the Subscriber bundle: 15](#_Toc161348349)

[Sample screenshots of its behavior & outputs 15](#_Toc161348350)

[Starting of the system 15](#_Toc161348351)

[Create new teachers 15](#_Toc161348352)

[Display all teachers’ details 16](#_Toc161348353)

[Edit teacher details. 16](#_Toc161348354)

[Delete teacher. 17](#_Toc161348355)

[Assign subjects for teacher 17](#_Toc161348356)

[Search teacher by ID 17](#_Toc161348357)

[Finance Management – IT21559136 18](#_Toc161348358)

[Finance Publisher 18](#_Toc161348359)

[Finance Subscriber 18](#_Toc161348360)

[Communication between Publisher and Subscriber: 18](#_Toc161348361)

[User interaction: 18](#_Toc161348362)

[Diagrammatic explanation 19](#_Toc161348363)

[Publisher Bundle Manifest: 19](#_Toc161348364)

[Subscriber Bundle Manifest: 20](#_Toc161348365)

[Publisher is added/imported as a dependency in the Subscriber bundle: 20](#_Toc161348366)

[Sample screenshots of its behavior &outputs 21](#_Toc161348367)

[Add new payment 21](#_Toc161348368)

[View All payment details 21](#_Toc161348369)

[Get Total of all payments 22](#_Toc161348370)

[Get Total for a specific subject 22](#_Toc161348371)

[Exist from the system 22](#_Toc161348372)

[Validations for Invalid Inputs 23](#_Toc161348373)

[Timetable Management – IT21786464 25](#_Toc161348374)

[Timetable Publisher 25](#_Toc161348375)

[Timetable Subscriber 25](#_Toc161348376)

[Communication between Publisher and Subscriber: 25](#_Toc161348377)

[User interaction: 25](#_Toc161348378)

[Diagrammatic explanation 26](#_Toc161348379)

[Timetable Service: Publisher/Subscriber Information 26](#_Toc161348380)

[Publisher: Timetable\_Producer Manifest 26](#_Toc161348381)

[Subscriber: Timetable\_Consumer Manifest 27](#_Toc161348382)

[Timetable Service: Starting/Stopping providers. 27](#_Toc161348383)

[Starting Timetable Publisher: 27](#_Toc161348384)

[Starting Teacher Publisher: 27](#_Toc161348385)

[Starting Timetable Subscriber 27](#_Toc161348386)

[Timetable Bundles are ‘Active’ 28](#_Toc161348387)

[Timetable Service: Functional behavior and outputs (Screenshots). 28](#_Toc161348388)

[Display Timetables 28](#_Toc161348389)

[Assign Teachers to Subjects 29](#_Toc161348390)

[Display Teachers with Subjects 29](#_Toc161348391)

[Update Teachers Assign to Subjects 30](#_Toc161348392)

# Introduction

We are implementing a publisher-subscriber program for an Educational Institute Management System using the OSGi framework. This encompasses student management, teacher management, financial management, and timetable management. Users can interact with the system using simple inputs.

A white text with black text

Description automatically generated with medium confidence

# Student Management – IT21704970

## Student Publisher

The Student Publisher bundle is responsible for managing and providing access to student information within an educational institution. It represents student IDs, names, grades, emails, and contact information.

The StudentService interface declares methods such as:

* Feature to add new students.
* Feature to remove a specific Student.
* Feature to search for students by ID.
* Feature to retrieve all the staff members.
* Feature to retrieve all student records.
* Feature to determine subjects for a specific grade.
* Feature to update specific student details.

The StudentServiceImpl class implements the interface and provides the actual functionality for adding, retrieving, deleting, updating student records, and determining subjects based on grade.

## Student Subscriber

The Student Subscriber bundle is designed to receive and utilize student information supplied by the Student Publisher. Its activator class functions as the trigger for starting and stopping the Student Subscriber bundle.

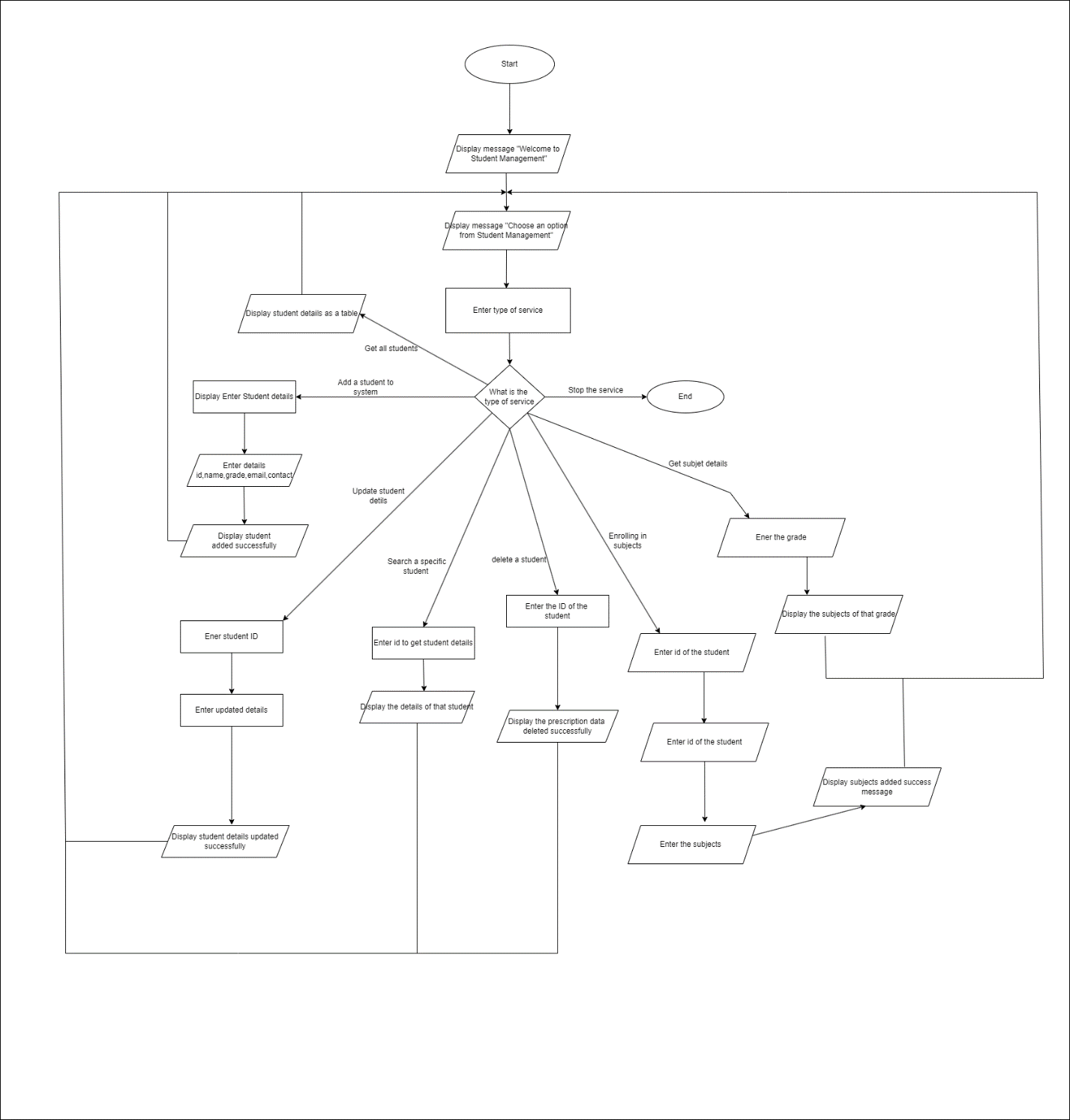
## Communication between Publisher and Subscriber:

The OSGi framework enables seamless communication between bundles. The Student Publisher bundle offers its services, while the Student Subscriber bundle utilizes these services. This interaction occurs through clearly defined interfaces, promoting a modular and decoupled system design.

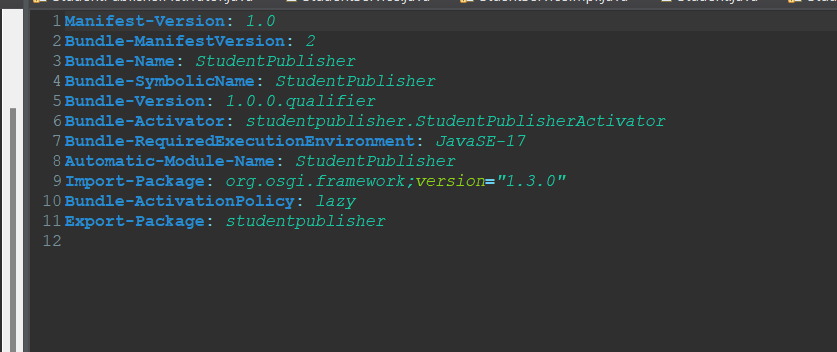
## User interaction:

The Student Subscriber bundle offers a user-friendly command-line interface for accessing the student management system. Through this interface, users can seamlessly perform various actions such as adding new students, browsing through the complete list of available students, updating existing student details, and deleting specific student records as needed. This intuitive interface empowers users to interact with the system effortlessly, enhancing their ability to manage student information effectively. Whether it's creating, browsing, updating, or removing student records, the bundle provides a streamlined experience, ensuring users can perform their desired tasks efficiently and with ease.

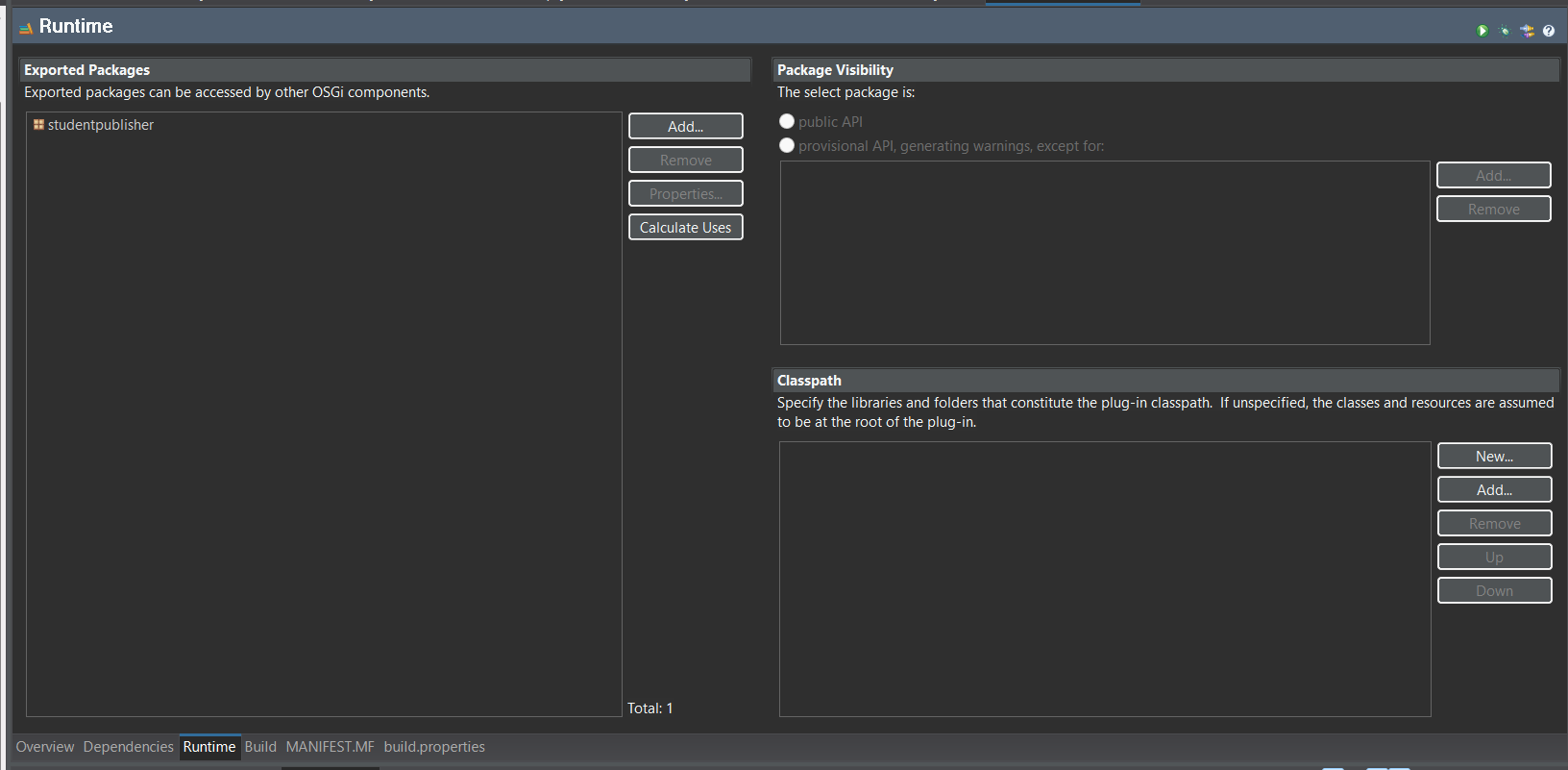
## Diagrammatic explanation



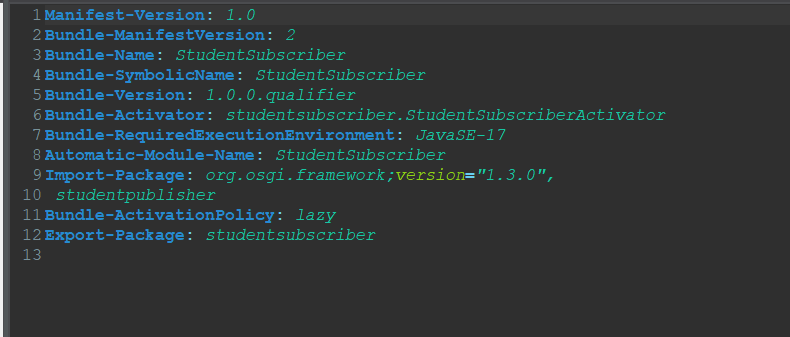
## Publisher Bundle Manifest:



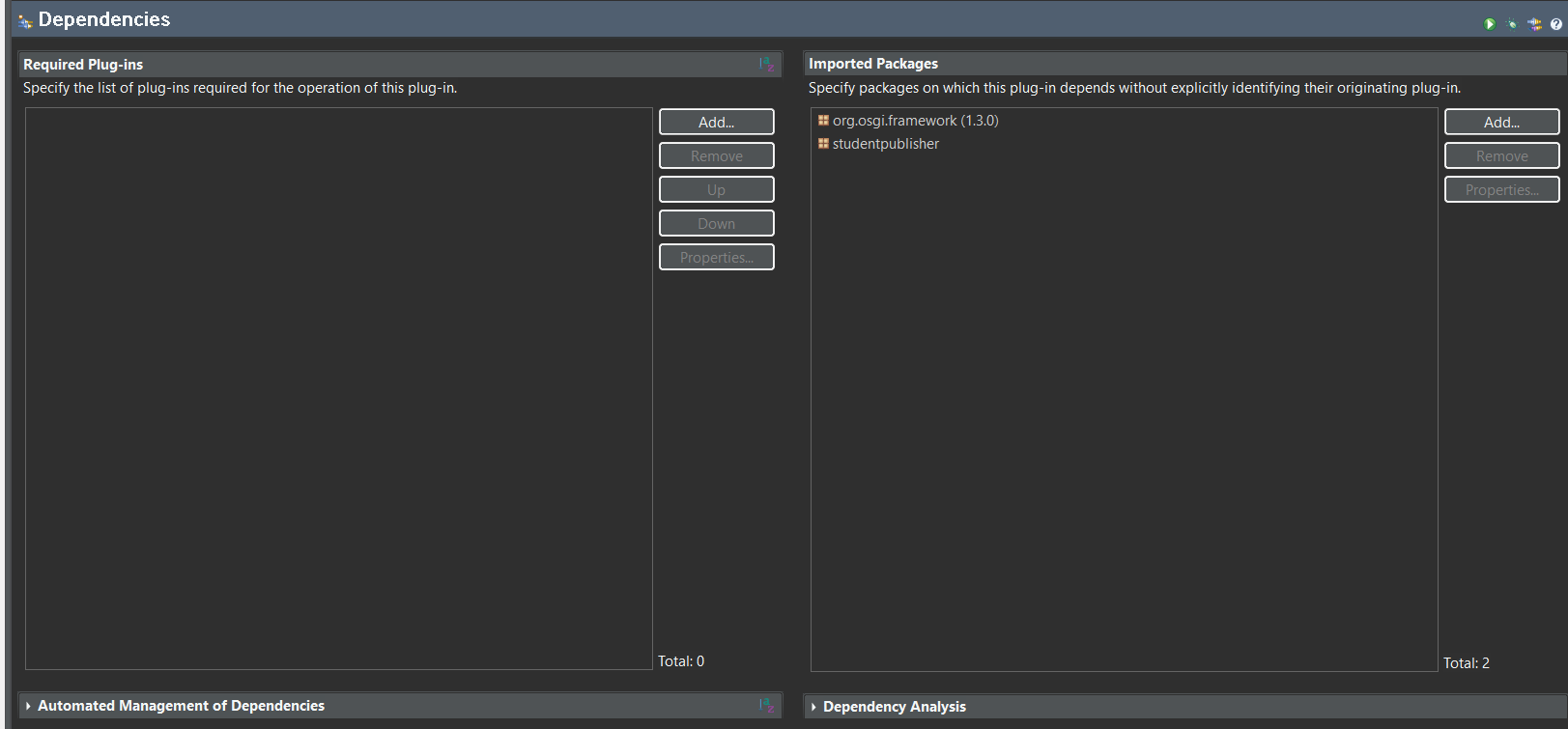
## During run time the package is exported:



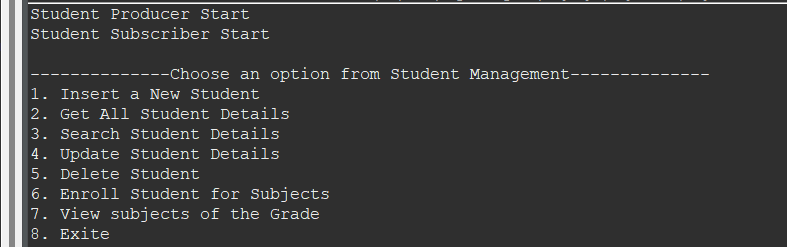
## Subscriber Bundle Manifest:



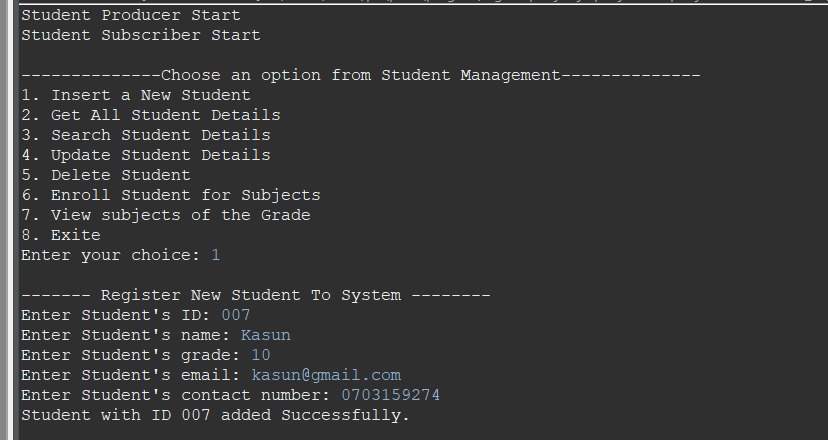
## Publisher is added/imported as a dependency in the Subscriber bundle:



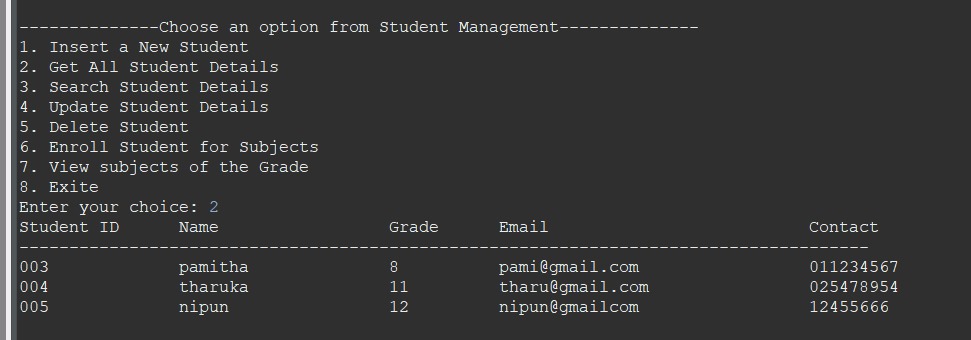
## Sample screenshots of its behavior &outputs



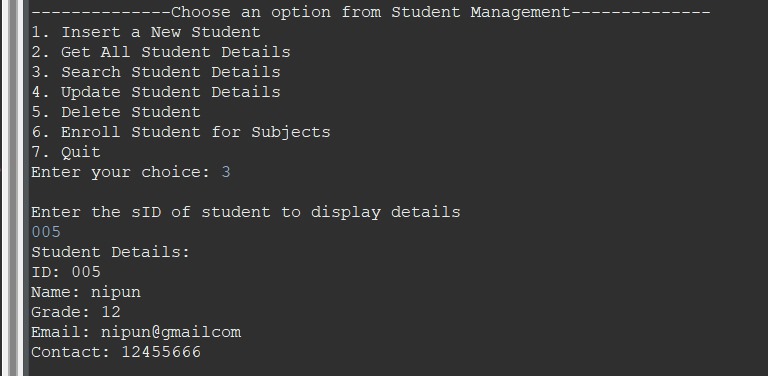
## Add Functionality – Add new students to the system



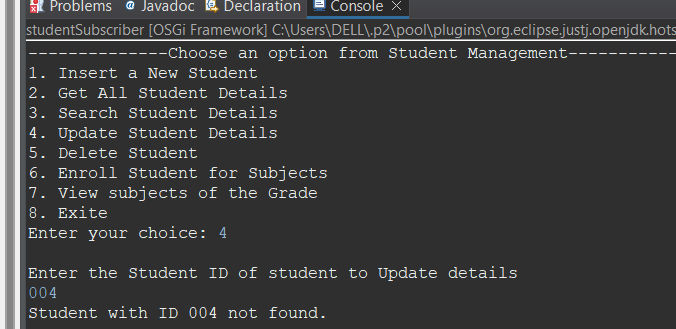
## View All Student details



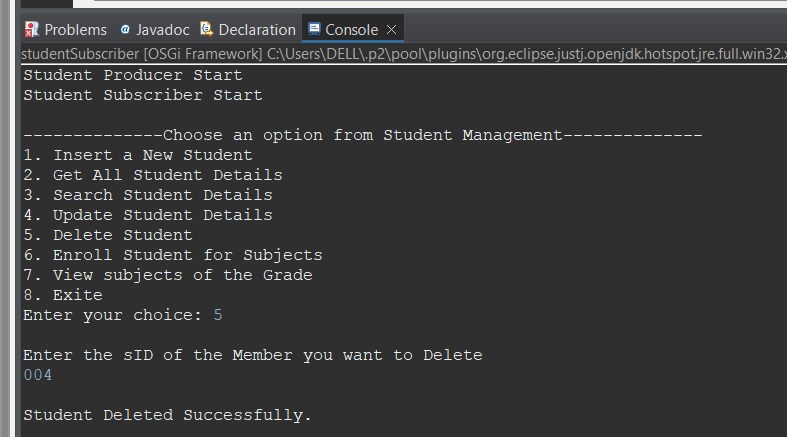
## Get specific student details .



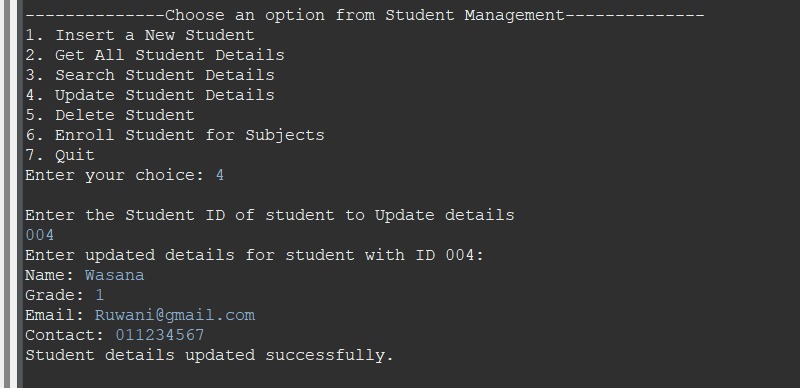
## Validation – If the staff ID does not exist.



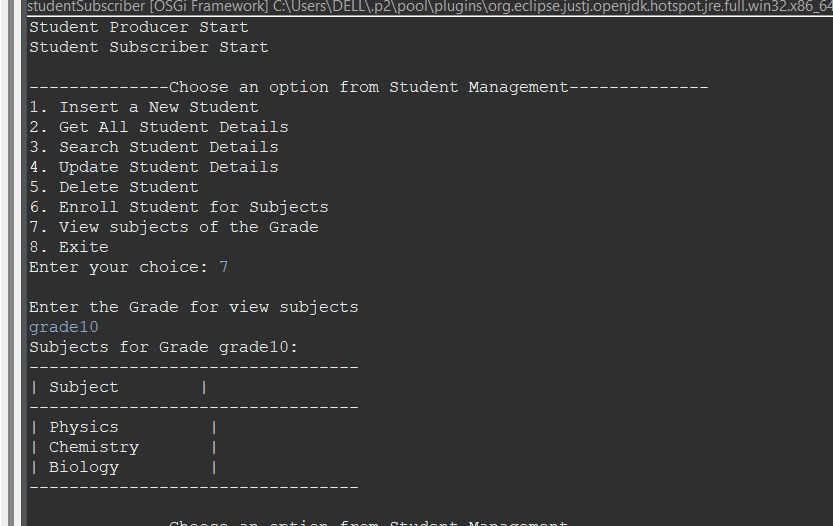
## Remove a student from the system



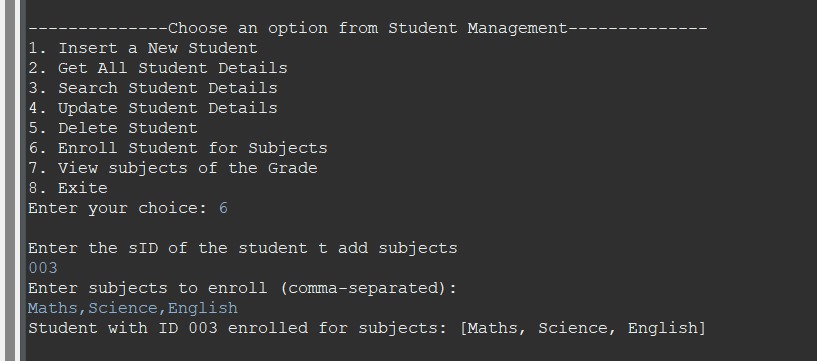
## Update the existing student details



## Get the subject details when enter the grade



## Enrolling students in the subjects



# Teacher Management – IT21271328

## Teacher Publisher

The Teacher Publisher bundle orchestrates the management and dissemination of teacher-related data within an educational framework. It encapsulates essential details such as teacher identifiers, names, assigned subjects, and contact information.

The TeacherService interface declares methods such as:

* Facilitate the addition of new teachers.
* Enable the display of all registered teachers.
* Provide the capability to update specific teacher information.
* Support the deletion of individual teacher records.
* Offer the functionality to assign subjects to teachers
* Feature to search for teachers by ID.

The TeacherServiceImpl class serves as the concrete implementation of the interface, furnishing the necessary logic for the addition, retrieval, modification, deletion, and search of teacher records, along with the assignment of subjects.

## Teacher Subscriber

The Teacher Subscriber bundle is engineered to consume and leverage teacher data dispensed by the Teacher Publisher. Its activator class acts as the initiator for initializing and terminating the Teacher Subscriber bundle.

## Communication between Publisher and Subscriber:

The OSGi framework serves as the conduit for seamless interaction between bundles. While the Teacher Publisher bundle extends its services, the Teacher Subscriber bundle capitalizes on these services. This exchange transpires through meticulously defined interfaces, fostering a modular and loosely coupled system architecture.

## User interaction:

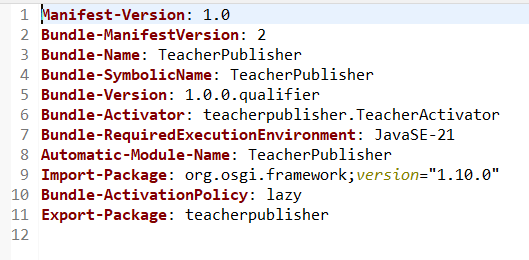
The Teacher Subscriber bundle extends a user-friendly command-line interface to interface with the teacher management system. Through this interface, users can effortlessly execute various tasks, including the addition of new teachers, the display of existing teacher profiles, the update of teacher information, the removal of specific teacher records, the assignment of subjects to teachers and searching for teachers by ID. This intuitive interface empowers users to navigate the system effortlessly, augmenting their proficiency in managing teacher-related data. Whether it entails the addition, display, update, deletion, or subject assignment for teachers, the bundle streamlines operations, ensuring users can accomplish their objectives efficiently and seamlessly.

## Diagrammatic explanation

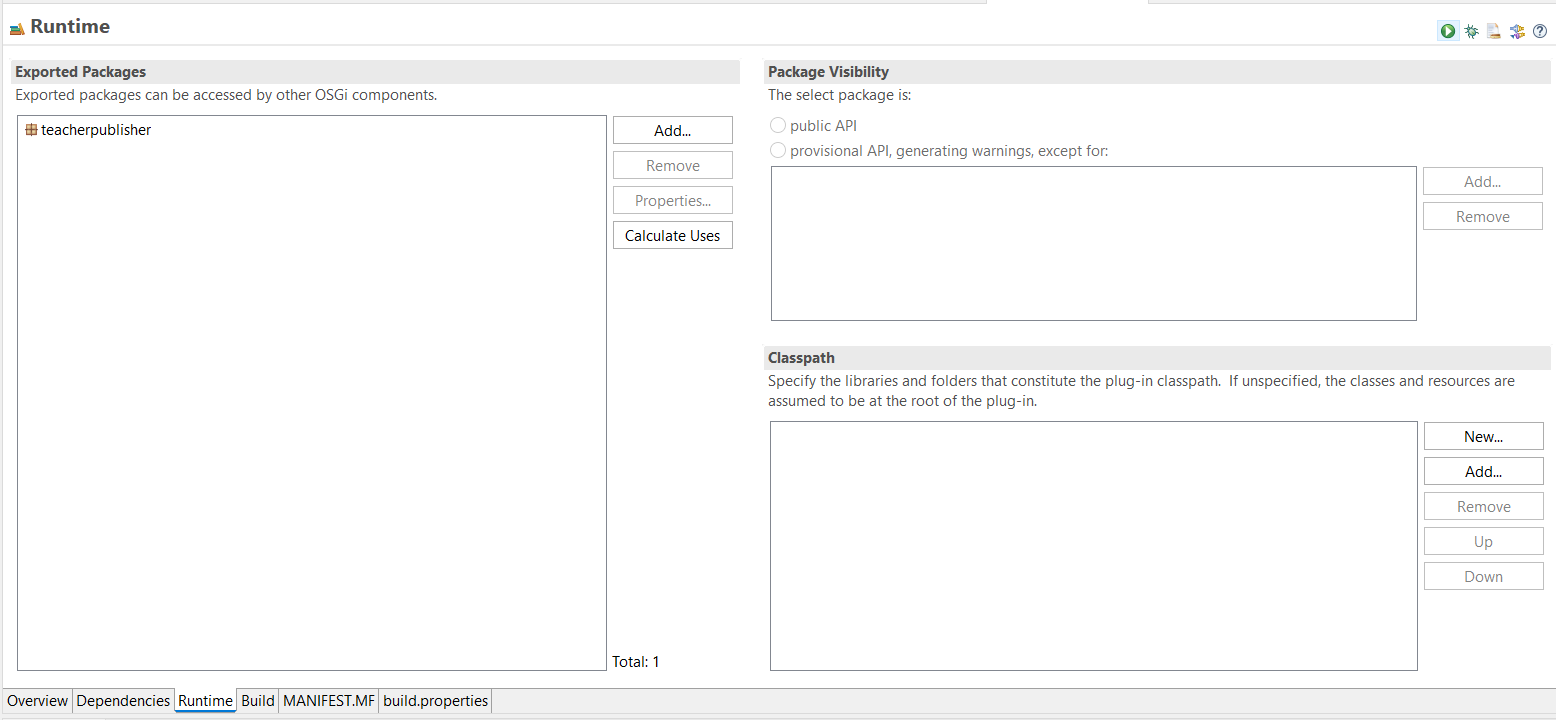
A diagram of a diagram

Description automatically generated with medium confidence

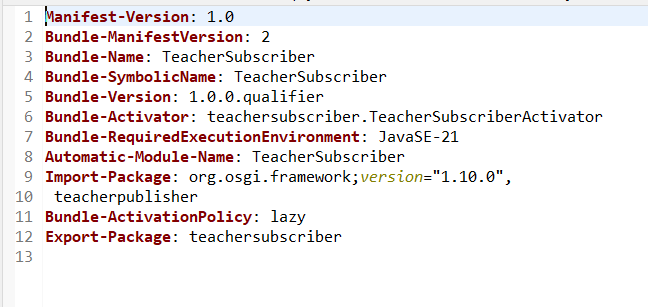
## Publisher Bundle Manifest:



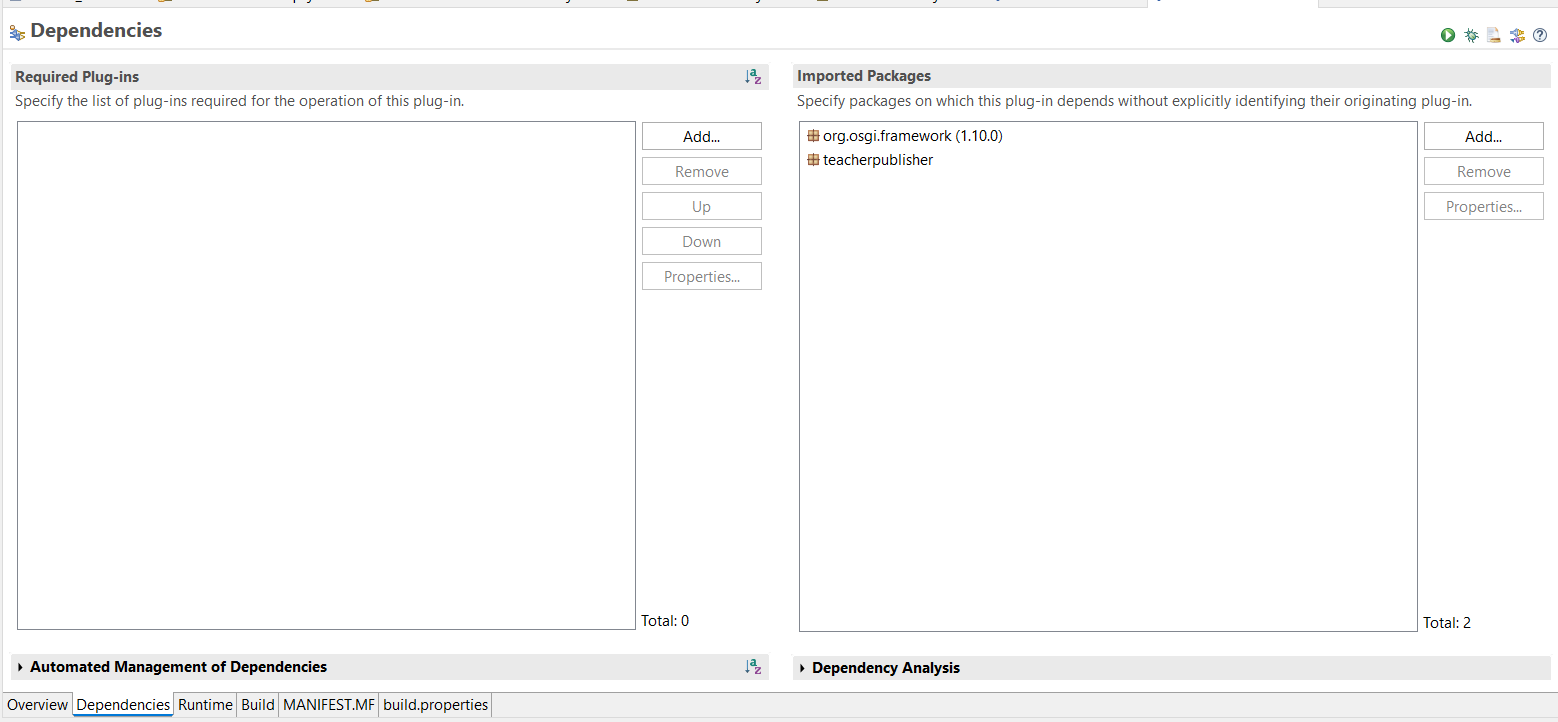
## During run time the package is exported:



## Subscriber Bundle Manifest:

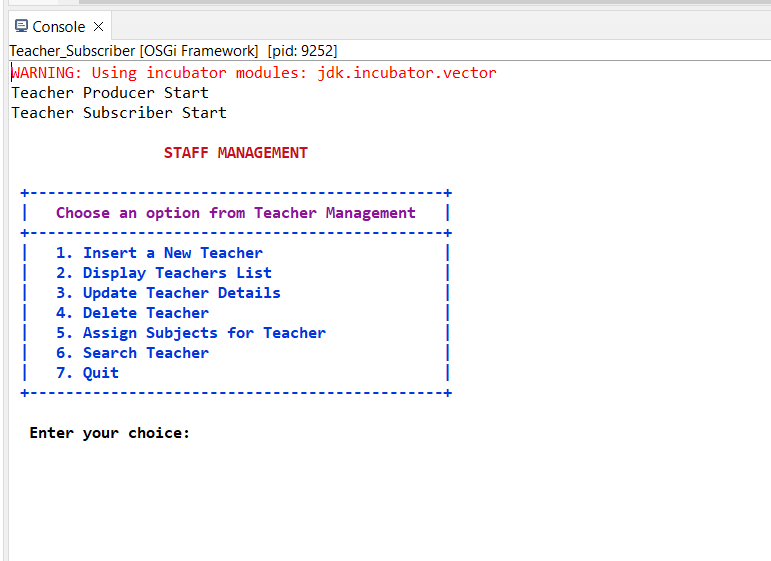


## Publisher is added/imported as a dependency in the Subscriber bundle:

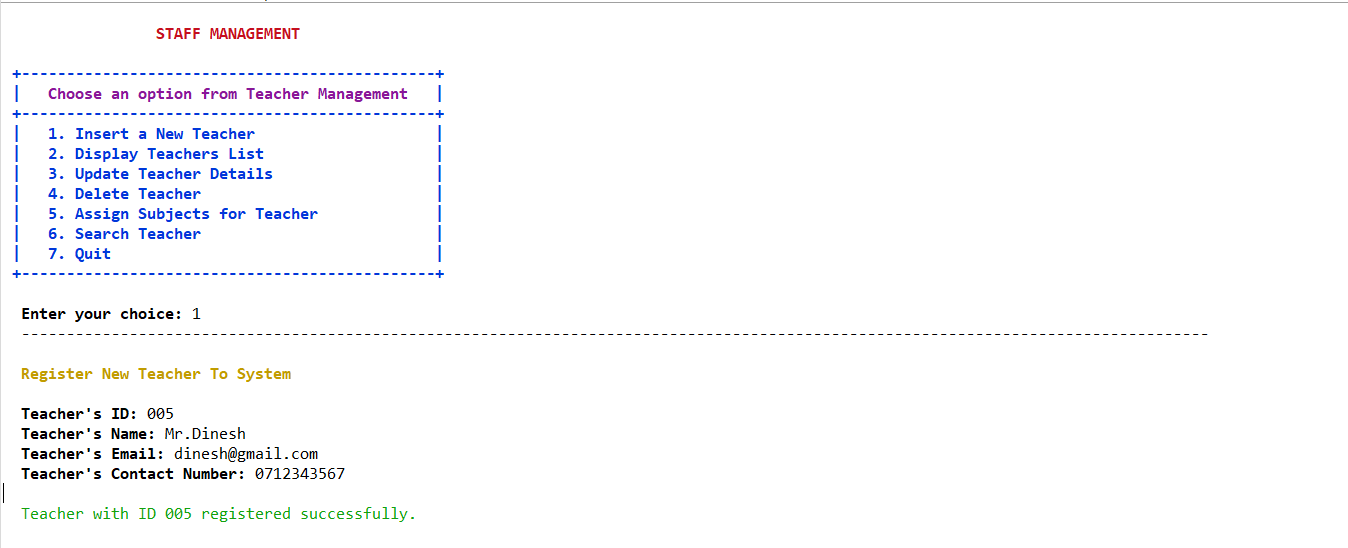


# Sample screenshots of its behavior & outputs

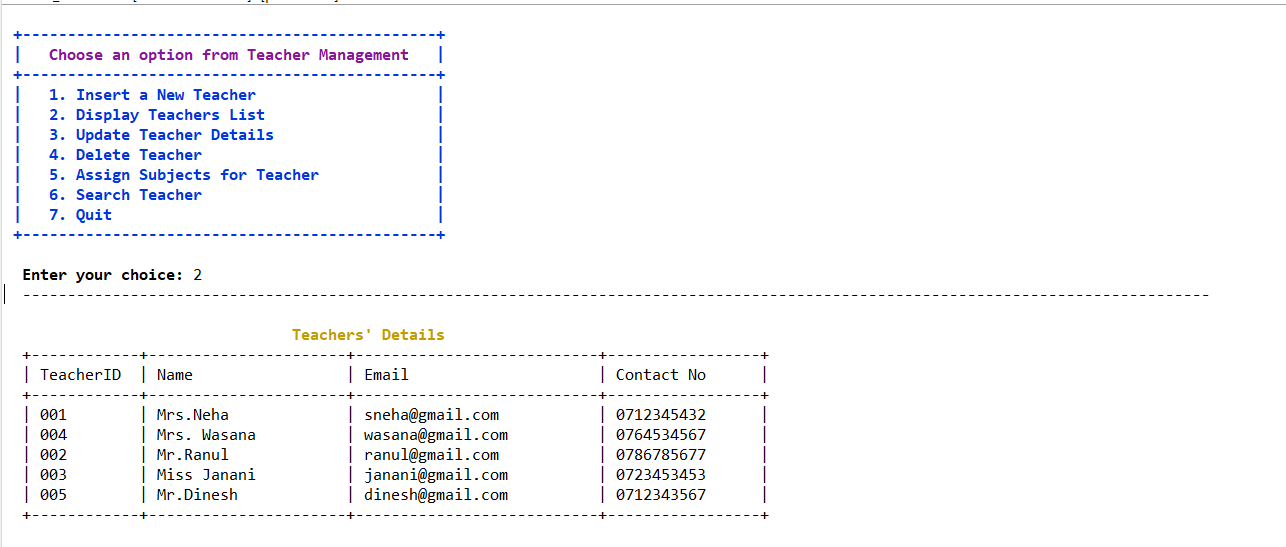
## Starting of the system



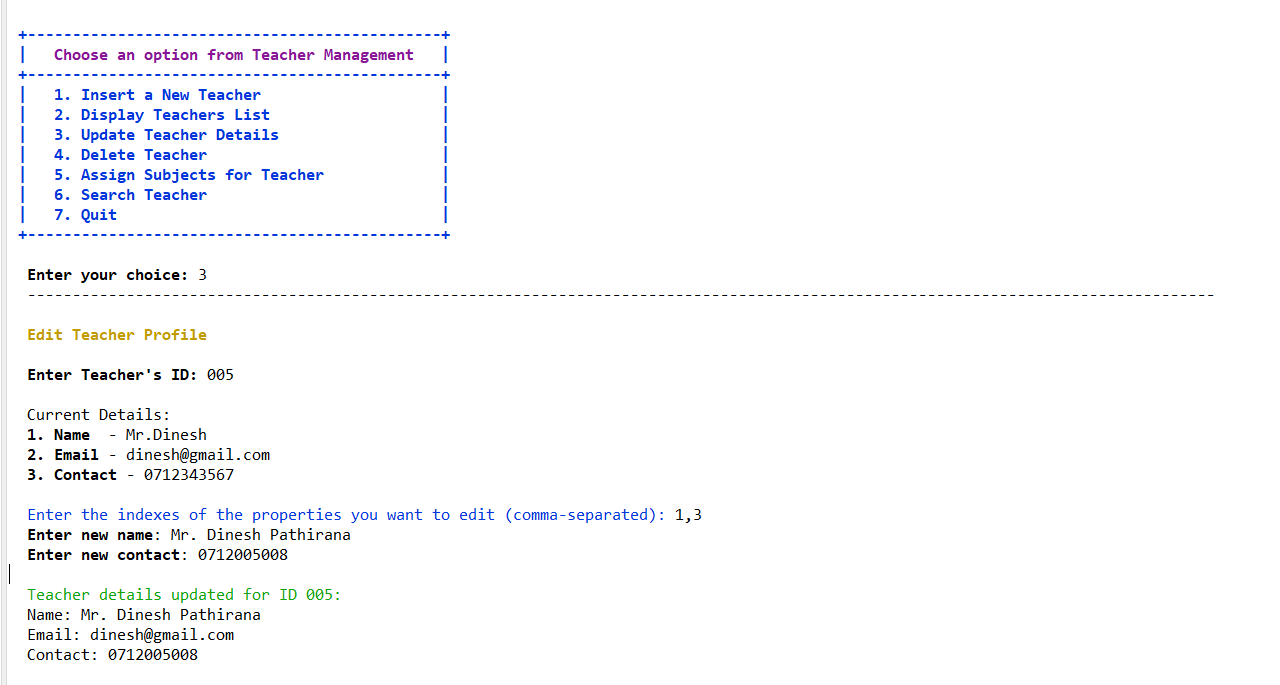
## Create new teachers



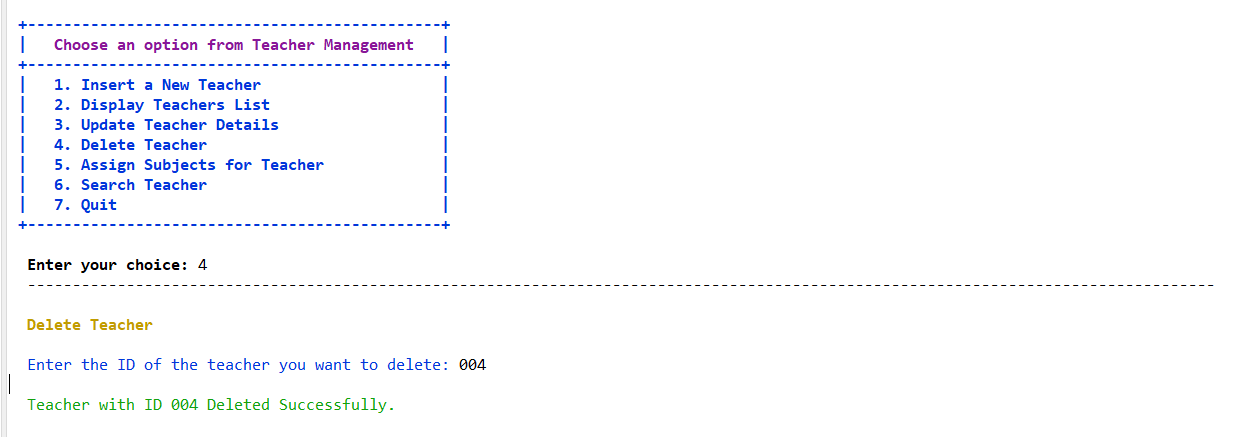
## Display all teachers’ details



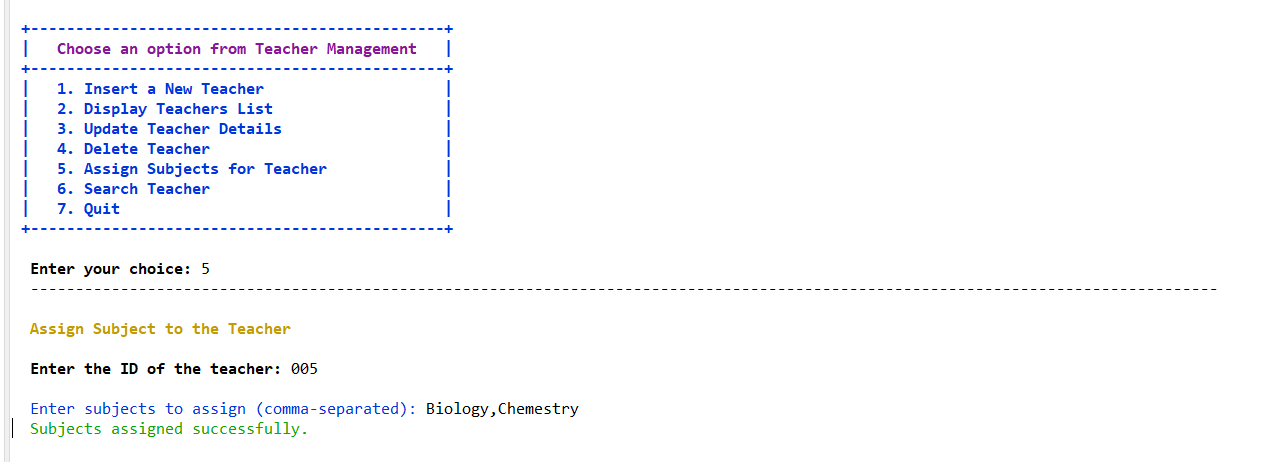
## Edit teacher details.



## Delete teacher.



## Assign subjects for teacher



## Search teacher by ID

A screenshot of a computer

Description automatically generated

# Finance Management – IT21559136

## Finance Publisher

The Finance Publisher bundle manages and provides access to financial data within an organization. It includes information on payments, their details, associated fees, and subjects.

A FinanceService interface outlines functions such as.

* Adding new payments.
* Viewing all payment details.
* Calculating the total collected fees.
* Determining the total for a specific subject.

An implementation class, FinanceServiceImpl, realizes these functionalities, enabling the addition, retrieval, and calculation of financial records based on subjects.

## Finance Subscriber

The Finance Subscriber bundle is tailored to receive and utilize financial data provided by the Finance Publisher. Its activator class serves to initiate and halt the Finance Subscriber bundle's operation.

## Communication between Publisher and Subscriber:

Communication between the Publisher and Subscriber occurs seamlessly through the OSGi framework. The Finance Publisher bundle offers its services, while the Finance Subscriber bundle leverages these services, fostering a modular and loosely coupled system design.

## User interaction:

The Finance Subscriber bundle presents users with a user-friendly command-line interface for accessing the financial management system. This interface facilitates various actions such as adding new payments, browsing through payment details, computing total collected fees, and determining totals for specific subjects. By offering an intuitive interface, the bundle empowers users to interact effortlessly with the system, enhancing their capability to manage financial data effectively. Whether users are performing tasks like adding, browsing, or calculating payment records, the bundle ensures a smooth experience, enabling efficient task execution.

## Diagrammatic explanation

A diagram of a diagram

Description automatically generated

## Publisher Bundle Manifest:

A screen shot of a computer program

Description automatically generated

## Subscriber Bundle Manifest:

A screen shot of a computer program

Description automatically generated

## Publisher is added/imported as a dependency in the Subscriber bundle:

**A screenshot of a computer

Description automatically generated**

## Sample screenshots of its behavior &outputs

**A black background with white dots

Description automatically generated**

## Add new payment

A black background with white dots

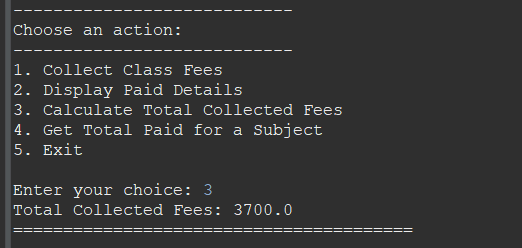
Description automatically generated

## View All payment details

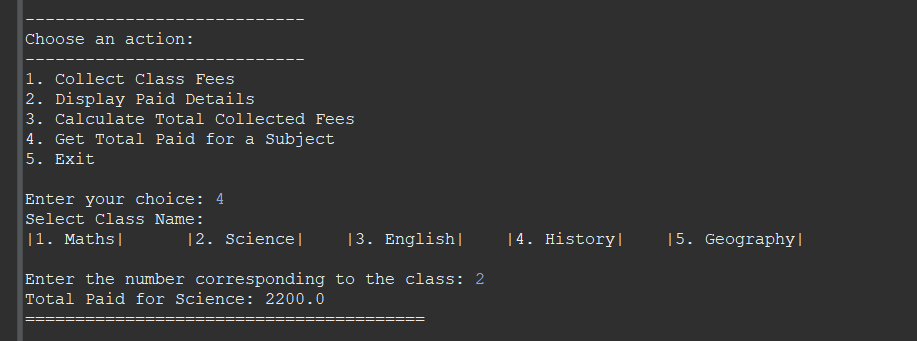
**A screenshot of a computer

Description automatically generated**

## Get Total of all payments



## Get Total for a specific subject

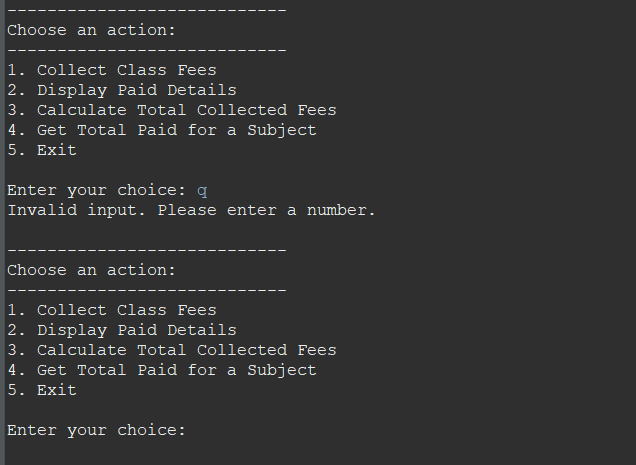


## Exist from the system

**A screen shot of a computer

Description automatically generated**

## Validations for Invalid Inputs



A screenshot of a computer

Description automatically generated

**A computer screen shot of a black screen

Description automatically generated**

# Timetable Management – IT21786464

## Timetable Publisher

The Timetable Publisher bundle helps students know their schedules and ensures teachers are assigned to the right subjects in a school.

The TimetableService interface has these main features:

* Get Timetable: You can see your timetable by telling your grade.
* Assign Teachers: Teachers are matched with the subjects they teach.
* See Teacher Details: You can find out about a teacher and the subjects they teach.
* Update Teacher's Subjects: If needed, you can change the subjects a teacher teaches.

The DailyTimetable class makes all these features work, making it easy for students to check their schedules and for schools to manage teacher assignments

## Timetable Subscriber

The Timetable Subscriber bundle is created to receive and make use of timetable information and teacher assignment details provided by the Timetable Publisher. Its activator class acts as the controller to initiate and halt the operation of the Timetable Subscriber bundle.

## Communication between Publisher and Subscriber:

The OSGi framework enables seamless communication between bundles. The Timetable Publisher bundle offers its services, while the Timetable Subscriber bundle utilizes these services. This interaction occurs through clearly defined interfaces, promoting a modular and decoupled system design.

## User interaction:

The Timetable Subscriber bundle provides a user-friendly command-line interface for accessing the timetable management system. Through this interface, users can easily perform various actions such as viewing the timetable, browsing through the complete list of available subjects, and viewing teacher assignments. This intuitive interface empowers users to interact with the system effortlessly, enhancing their ability to manage timetable information effectively. Whether it's viewing, browsing or updating timetable records and teacher assignments, the bundle offers a streamlined experience, ensuring users can accomplish their desired tasks efficiently and with ease.

## A diagram of a company Description automatically generatedDiagrammatic explanation

# Timetable Service: Publisher/Subscriber Information

## Publisher: Timetable\_Producer Manifest

A screenshot of a computer

Description automatically generated

## Subscriber: Timetable\_Consumer Manifest

A screenshot of a computer program

Description automatically generated

# Timetable Service: Starting/Stopping providers.

## Starting Timetable Publisher:



## Starting Teacher Publisher:

A white background with black and white clouds

Description automatically generated

## Starting Timetable Subscriber

A screen shot of a computer

Description automatically generated

## Timetable Bundles are ‘Active’

A screenshot of a computer program

Description automatically generated

# Timetable Service: Functional behavior and outputs (Screenshots).

## Display Timetables

A screenshot of a computer

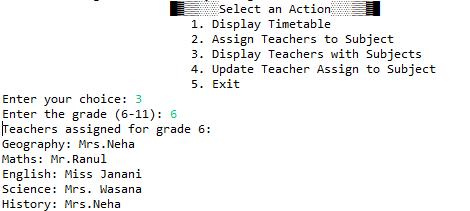
Description automatically generated

## Assign Teachers to Subjects

A screenshot of a computer

Description automatically generated

## Display Teachers with Subjects



## Update Teachers Assign to Subjects

A screenshot of a computer

Description automatically generated