IY4113 Milestone 1

Assessment Details	Please Complete All Details
Group	4113 B
Module Title	Applied Software Engineering using Object Orientated Programming
Assessment Type	Java Fundamentals - Milestone 1
Module Tutor Name	Jonathon Shore
Student ID Number	P508377
Date of Submission	25/01/2026 - Sunday
Word Count	1199

- [x] *I confirm that this assignment is my own work. Where I have referred to academic sources, I have provided in-text citations and included the sources in the final reference list.*
- [x] *Where I have used AI, I have cited and referenced appropriately.

Purpose of the Program

The purpose of the CityRide Lite program is to help a user keep track of their public transport journeys during a single day. The program will allow the user to enter details about each journey, such as where they travelled from and to, what type of passenger they are, and whether the journey was taken during peak or off-peak time.

Using this information, the program calculates how many zones were crossed, finds the correct fare using the provided dataset, applies any discounts, and checks if the daily cap has been reached. The user can also review their journeys, remove any incorrect entries, and see summaries of how much they have spent.

Overall, the program will act as a travel helper to user as it will calculate all the journeys and let the user know what they have spent on travel.

Core Program functionality

Manage journeys

-Store multiple public transport journeys for a single day.

-Each journey includes date, from zone, to zone, time band, passenger type, and fare details.

-Assign a unique ID to each journey.

Calculate fares

-Calculate the number of zones crossed for each journey.

-Check the correct base fare using the provided dataset.

-Apply passenger discounts based on passenger type.

-Apply daily caps and adjust journey charges when caps are reached.

Add journeys

-Allow the user to enter journey details through menu.

-Validate all inputs such as zones, passenger type, and time band.

-Confirm when a journey has been successfully added.

Display journeys

-List all journeys entered during the session in the order they were added.

-Display key journey information including zones, fares, discounts, and charged amount.

Filter journeys

-Filter journeys by passenger type, time band, zone, or date.

-Display only journeys that match the selected criteria.

Remove journeys

-Remove a journey using its unique ID.

-Recalculate totals after a journey is removed.

Reset daily data

-Clear all journeys and reset totals after user confirmation.

Generate summaries

-Display a daily summary including total journeys, total cost, average cost, and most expensive journey.

-Display totals by passenger type, including whether the daily cap has been reached.

Constraints

-The program only tracks the journey for a single day.

-Valid zones are limited to numbers 1 to 5.

-Time bands are limited peak and off peak only.

-All monetary values must be displayed to two decimal places.

-All fares, caps and dicount must come from data set.

-The data set must not be chnaged.

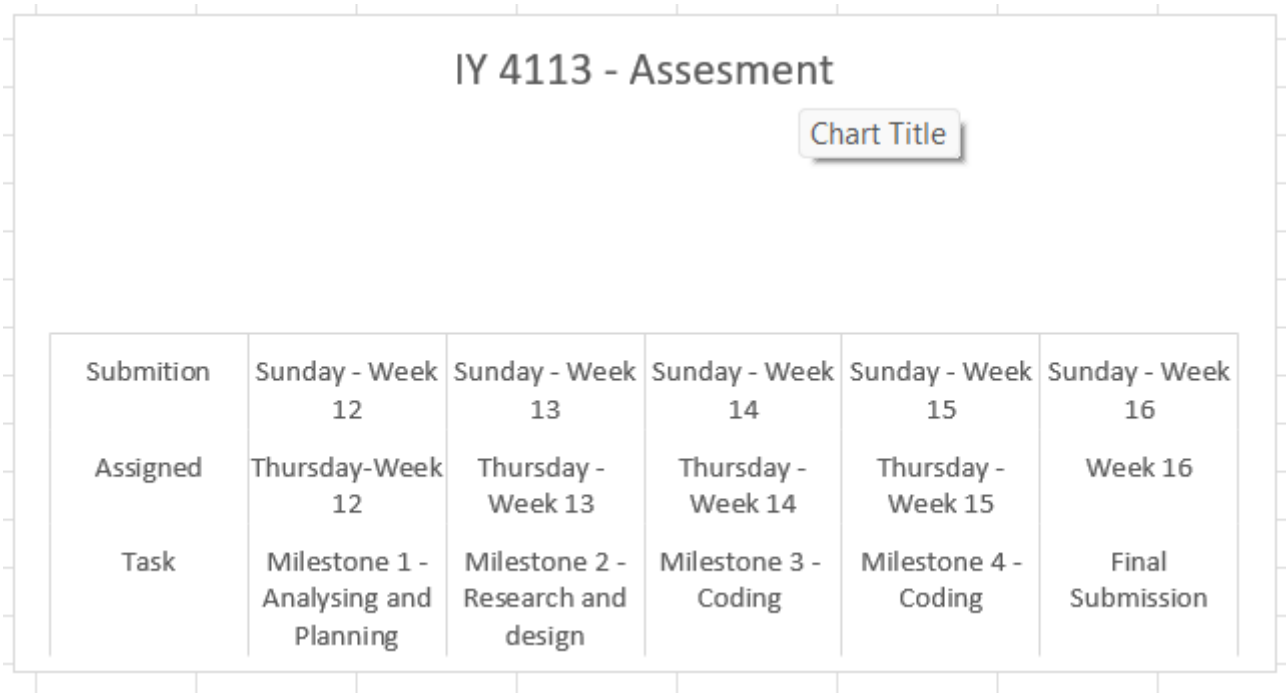
-The application is console-based.

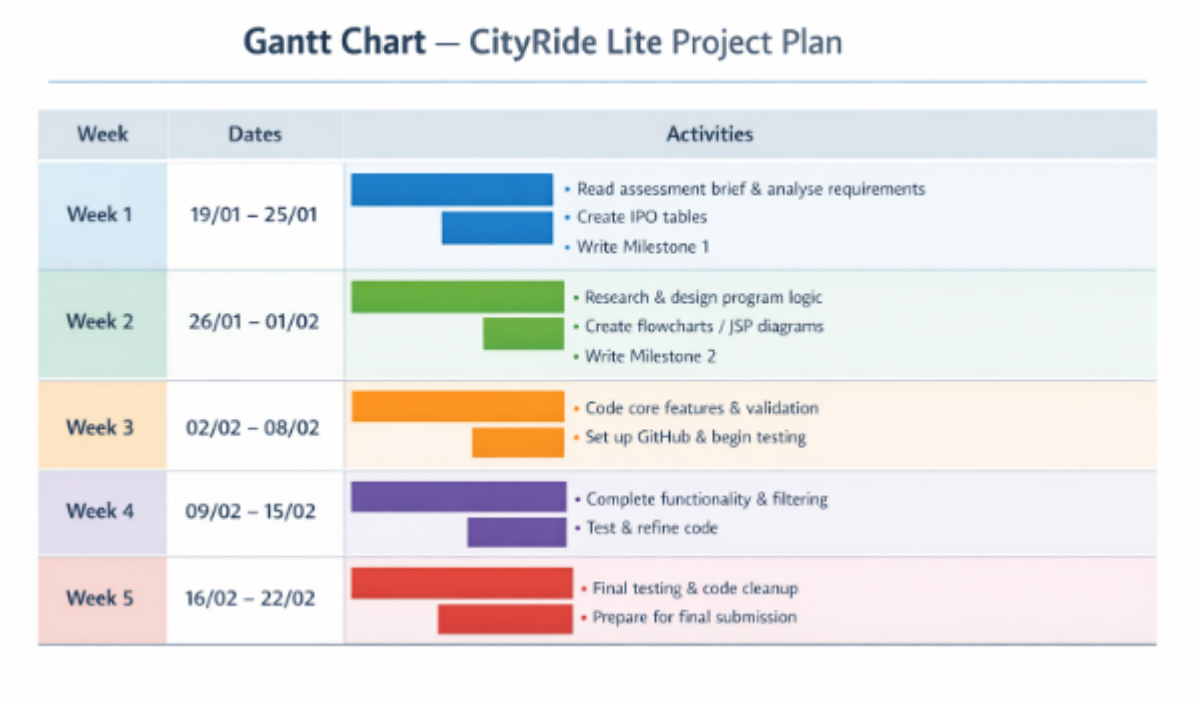
Input Process Output Table

Feature/Task	Input	Process	Output
Start program	NONE (Program launched)	Create main menu options.	Main menu displayed.
Display main menu	User menu choice (1-7)	validate input is numeric and within range;route to selected function.	Selected option confirmed/ next menu shown.
Add journey	date	validate date format or accept system date.	proceed to journey details.
	From zone (1-5).	Validate numeric input and zone range.	Accpeted zone or error
	To zone (1-5)	validate numeric input and zone range.	Accpeted zone or error
	Passenger type	Validate against allowed types.	Accepted type or error.
	Time (Peak/off peak)	validate against allowed options.	Accepted time or error.
Calculate zones crossed.	form zone to zone.	calculate abs(toZone - fromZone) + 1.	Zones crossed value
Calculate base fare	Zones, time band	look up base fare from Data set.	Base fare value
Apply discount	Passenger type, base fare	Apply discount based on passenger type.	Discount fare.
Apply daily cap	Passenger type, discounted fare.	check running total against daily cap; adjust fare if cap exceeded.	final charged fare.
Store journey	Valid journey details	Create unique ID, store journey in memory, update tools.	Confirmation message. "Journey added."
List all journeys	None(menu selection)	Retrieve journeys in order entered	Full journey list displayed

Feature/Task	Input	Process	Output
Filter journeys	Filter type (passenger, time,zone,date)	compare journey against filter	Filtered journey list.
Remove journey	Journey ID	Check if ID exists, ask for confirmation, remove journey, recalculate totals.	"Journey removed" or error.
Reset day	User confirmation(Yes/No)	Clear all journeys and reset all running totals.	"All journeys reset"
View daily summary	none (menu selection)	Count journeys, sum charged fares,calculate average,find highest fare	Daily summary displayed
View totals by passenger type	none (menu selection)	Group journeys by passenger type, calculate totals and cap status	Totals per passenger type displayed
Exit program	Menu (exit)	end program loop	"goodbye"

Gantt Chart





"Second image is generated by AI".

Diary Entries

DIARY ENTRY 1

-22/01/2026 - Understanding the program & How to manage the deadlines.

Today, I went through the program carefully to understand what my project should include and how I can manage my time to finish all the work before the deadlines.

After understanding the program, I realized that I need to focus on the main requirements, such as structures, designs, algorithms, evaluation notes, and a Gantt chart. These will help me complete my work smoothly and in an organized way.

My main challenge was finishing all the work on time and submitting it before the deadlines. To solve this problem, I decided to create a Gantt chart to track my tasks and deadlines and manage my time better.

DIARY ENTRY 2

-25/01/2026 - Overall view and going through other milestones.

Today, I completed this milestone and took most of my time in understanding the assessment requirements and tried my level best to complete all the assigned tasks related to this milestone carefully.

By creating the IPO table and going through the other milestones, i believe the milestone 3 and 4 will be much clearer and manageable for me as this table will help me out alot.

The area which was challenging was understanding how daily caps should be applied. So, after reading it again and again, i found out that once cap is reached furhter journeys for that passenger should be free.
