CONCORDIA UNIVERSITY

DEPARTMENT OF COMPUTER SCIENCE AND SOFTWARE ENGINEERING

SOEN 342 – Sections H and II:

Software Requirements and Deployment

Fall 2025

Instructors: Constantinos Constantinides (Section **H**) and Ali Jannatpour (Section **II**)

Project Description: Iteration 3

Date posted: Friday 24 October, 2025

Deadline for completion: Friday 7 November, 2025 at 23:59.

WHAT SHOULD HAVE BEEN DONE SO FAR

Use Case Load Records: The system loads routes from a database (cvs file) and keeps them in a catalog of routes in working memory.

Use Case Search for Connections: A client enters criteria for a trip. The system consults the routes catalog and suggests viable connections, if any exist. In doing that the system presents direct connections (corresponding to direct routes) or indirect connections (1-stop and 2-stop, if any), which are computed from the routes available.

Use Case Book a Trip: A client is able to search, identify and select a desired connection and proceed to book a trip.

Use Case View Trips: A client should be able to enter their last name and id and view all their current trips (i.e. for today's or future connections) and past trips, where the latter are placed in some 'history collection', also viewable by the client.

DESCRIPTION OF THE CURRENT ITERATION

In this iteration we introduce a non-functional requirement: Persistence. You must support this requirement through a **Relational Database**, and document this provision with a Data Model (Tables) while clearly identifying any and all keys.

Modification of current requirements:

Use Case Book a Trip: The system should avoid suggesting connections with unconditional layover durations. You must build some policy (feel free to build your own), such as: "During the day it can be OK to have a layover for 1-2 hours. However, we don't want a layover for more than 30 minutes when after hours."

Use Case Book a Trip: Once created, a trip is assigned a unique numerical ID.

RESOURCES

SQLite: A small SQL database engine

https://sqlite.org/

SQLite Java: how to interact with SQLite using Java JDBC API.

https://www.sqlitetutorial.net/sqlite-java/

Python DB-API 2.0 interface for SQLite databases

https://docs.python.org/3.9/library/sqlite3.html

MySQL: Open source database management

https://www.mysql.com/

Connecting to MySQL Using the JDBC DriverManager Interface

https://dev.mysql.com/doc/connector-j/en/connector-j-usagenotes-connect-drivermanager.html

List of object–relational mapping software [Wikipedia]

https://en.wikipedia.org/wiki/List_of_object%E2%80%93relational_mapping_software

End of Iteration 3