Project Specification for Group TUT0301-15

Team Name: PVSM

Domain:

A degree table generator for Arts and Science students. By inputting your degree into the application, multiple tables with different combinations of lecture times will be generated with mandatory courses for your degree.

User Stories:

User story 1: As a user, I want to compare different timetables in a clear view, so that I can choose the timetable that suits me best.
User story 2: As a user, I want to keep some of the lecture section fixed, so that I can take the lecture together with my friends.
User story 3: As a user, I want to sort the timetables in different ways, for example, in the order of the sum of the interval between lectures, so that I can choose the timetable more easily.
User story 4: As a user, I want to modify the lecture sections manually, so that I can customize the timetable in my way.
User story 5: As a user, I want to save the drafts of the timetables, so that I don't need to make immediate decision.
User story 6: As a user, I want to have the minimum courses for the degree requirement, and be able to add elective courses, so that I can choose the field I want to study.
User story 7: As a user, I want to have the option to select from equivalent courses when several equivalent courses can be used for the program, so that I can choose the course level myself. (e.g. MAT235 vs MAT237, CSC236 vs CSC240)
User story 8: As a user, when editing the timetable manually, conflicts should be highlighted so that I can avoid the conflicts and make further edition.
User story 9: As a user, I want to have a time preference for the lecture sections, so that I can pick the time I prefer.
User story 10: As a user, I want to know when I will have electives and be able to fill them up with courses I want to take.

Use Cases:

Generate timetables with course code:

1. User inputs course codes
2. Button “generate” clicked
3. Display one of generated timetables
4. Button “generate” clicked again
5. Display another generated timetable

Alternative:

1. Course code invalid
2. Display failure message

Alternative:

1. There is no possible timetable without conflicts
2. The timetable with least conflicts will be generated and show “Warning: all timetables have conflicts.

Fix(lock) the specific course:

1. User clicks the lecture section
2. The locked UI appear on the right corner on the lecture block
3. The course block is fixed and when the generated button is clicked, the locked lecture block won’t change, however the other lecture blocks may change.
4. Add a lock protection, when user want to delete the course, give a reminder that the block is locked.
5. If the user still wants to delete the course, then course deleted.

Delete course:

1. User clicks the delete sign of the lecture section (on top left corner)
2. Check whether the course is locked.
3. Lecture section deleted on the timetable
4. The time for the lecture session is free.
5. The course code automatically removed from the course code list

Save the timetable:

1. User clicks the save button
2. Ask whether user wants to save all the available timetables and the layout for the timetables.
3. Users select the name and the location.
4. Timetables saved

Alternative:

1. There is no generated timetable
2. Error message displayed

Sort the timetable:

1. User generates the timetables

2. User selects the “sort” button
3. The “sort” interface appears.
4. User selects the key for sorting
5. Display sorted timetables

Alternative:

6. There is no generated timetable
7. Error message displayed

Generate timetables with degree:

1. User selects the degree
2. Giving the choices of equivalent courses (CSC236 vs CSC240)
3. User makes the choice
4. Generate the timetables based on the selected course

Alternative:

1. Degree code invalid
2. Error message displayer

Add course:

1. User input the course code
2. Display all lecture sessions
3. User chooses the lecture session
4. Timetable updated
5. Add the course to the course list
6. Report conflicts (if applicable)

Compare timetables:

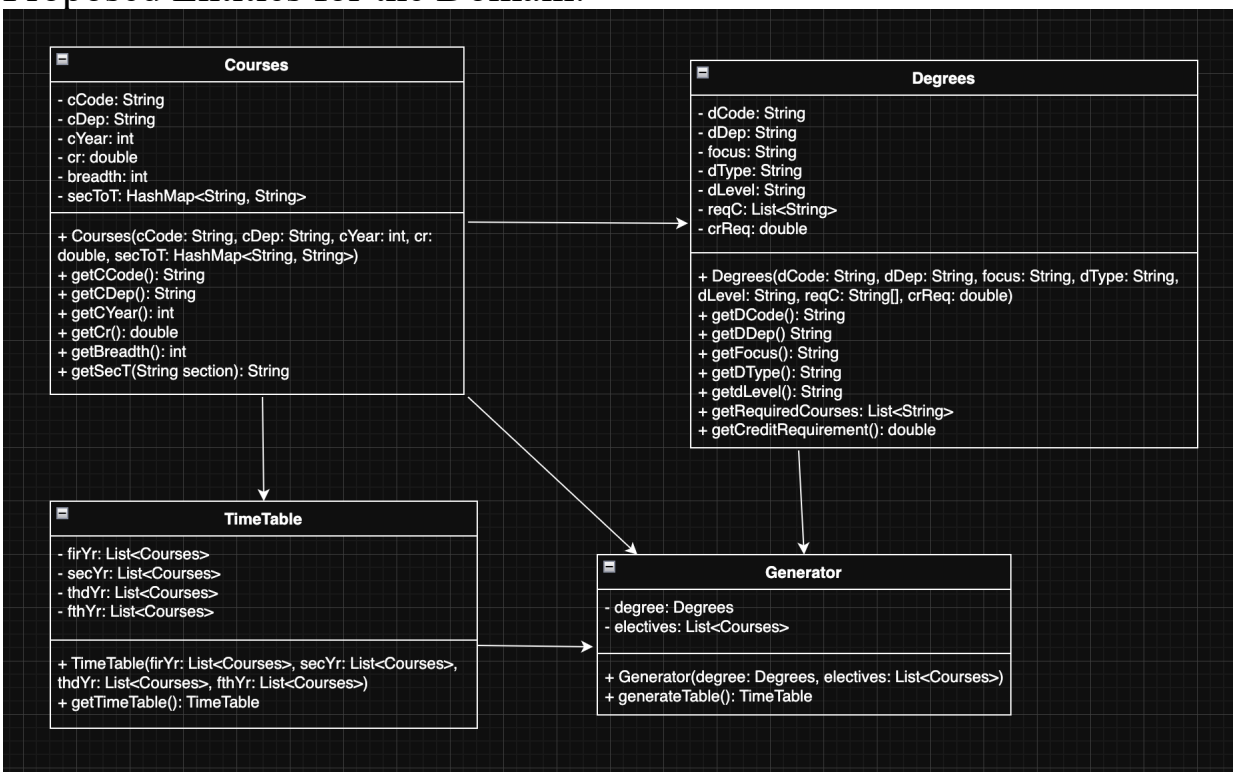
1. Press compare button
2. Display all timetables in a clear view
3. Giving the choices of compare all timetables or just compare specific timetables selected by the users

MVP:

Lead	Use Case [title of use case from previous section]	User Story [optionally, include this column to indicate which user story this use case is associated with]

Lu Wang	Generate timetables with course code:	
Peter Xu	Generate timetables with degree	
ChenHao Sun	Compare timetables	1
Shiraz Ali	Add/Delete/fix course	2,8
Septian Dimas Pasaribu	Sort the timetable	3
Jeremy Morgan Tam	Save timetable	5

Proposed Entities for the Domain:



Proposed API for the project:

API name: **OpenRouteService API**

Main Service Provided: Return the working distance of two given locations

Status: Run and tested in Java. Example of output: {"type":"FeatureCollection","bbox":[-79.398797,43.658921,-79.394986,43.662444],"features":[{"bbox":[-79.398797,43.658921,-

79.394986,43.662444],"type":"Feature","properties":{"segments":[{"distance":651.9,"du
ration":469.4,"steps":[{"distance":138.6,"duration":99.8,"type":11,"instruction":"Head
north","name":"-
","way_points":[0,5]},{"distance":13.2,"duration":9.5,"type":0,"instruction":"Turn
left","name":"-
","way_points":[5,7]},{"distance":132.6,"duration":95.5,"type":1,"instruction":"Turn
right","name":"-
","way_points":[7,13]},{"distance":43.3,"duration":31.2,"type":0,"instruction":"Turn
left","name":"-
","way_points":[13,14]},{"distance":101.6,"duration":73.2,"type":1,"instruction":"Turn
right","name":"-
","way_points":[14,23]},{"distance":18.5,"duration":13.3,"type":0,"instruction":"Turn
left","name":"-
","way_points":[23,25]},{"distance":182.6,"duration":131.5,"type":0,"instruction":"Turn
left onto King's College Circle","name":"King's College
Circle","way_points":[25,38]},{"distance":21.6,"duration":15.6,"type":1,"instruction":"T
urn right","name":"-
","way_points":[38,41]},{"distance":0.0,"duration":0.0,"type":10,"instruction":"Arrive at
your destination, on the right","name":"-
","way_points":[41,41]}]}],"way_points":[0,41],"summary":{"distance":651.9,"duration"
:469.4}},"geometry":{"coordinates":[[-79.398369,43.658921],[-79.398666,43.659648],[-
79.398797,43.659913],[-79.398796,43.659963],[-79.398759,43.659993],[-
79.398612,43.660039],[-79.398616,43.660094],[-79.398621,43.660157],[-
79.3982,43.660251],[-79.397849,43.660328],[-79.397812,43.660336],[-
79.397138,43.660475],[-79.397106,43.660485],[-79.397069,43.660525],[-
79.39721,43.660901],[-79.397134,43.660914],[-79.397035,43.660933],[-
79.396932,43.660953],[-79.396864,43.660968],[-79.39641,43.661062],[-
79.39636,43.661073],[-79.39626,43.661093],[-79.396178,43.661135],[-
79.39601,43.661173],[-79.396011,43.661224],[-79.395954,43.661331],[-
79.396085,43.661491],[-79.396116,43.661551],[-79.39616,43.661669],[-
79.396175,43.661776],[-79.396147,43.661939],[-79.396107,43.662014],[-
79.396052,43.662103],[-79.395973,43.662182],[-79.395873,43.662252],[-
79.395562,43.662381],[-79.395438,43.662414],[-79.395218,43.662443],[-
79.395073,43.662444],[-79.395082,43.662351],[-79.395022,43.662292],[-
79.394986,43.662282]],"type":"LineString"}}],"metadata":{"attribution":"openrouteservi
ce.org | OpenStreetMap
contributors","service":"routing","timestamp":1762575991016,"query":{"coordinates":[[-
79.398,43.659],[-79.395,43.662]],"profile":"foot-walking","profileName":"foot-
walking","format":"json"},"engine":{"version":"9.3.0","build_date":"2025-06-
06T15:39:25Z","graph_date":"2025-10-30T00:32:48Z","osm_date":"2025-10-
20T00:00:01Z"}}}