*/\*\**

*\* Project: Time Table Project*

*\* Date: 18 December 2017*

*\* Version: 1.0*

*\* Programmers: Zahid Hussain, John Olorunsuyi, Eric Chiu*

*\* Course: MSc Computer Science (CN7021), University of East London*

*\*/*

**User Manual**

This package TimeTableProject contains three set of code and data and this user manual which are listed as follows::

1 StudyOfTimeTableDataAndResults\_CodeAndData

2. StudyOfTimeTableDataProvidedAndResults\_CodeAndData

3. TestTimeTableProjectMethodsWithJUnit\_Code

4. This user manual:

(File name: TimeTableProjectVer1UserManual).

**(Warning**: Should make a copy of this package before using it)

**Items included in StudyOfTimeTableDataAndResults\_CodeAndData:**

1. A folder named StudyOfTimeTableDataAndResults\_JavaCode. Inside it there are 10 classes and 9 interfaces. The main class is Launcher.java.
2. A folder named StudyOfTimeTableDataAndResults. It contains three csv files: a sample draft time table DraftTimeTable.csv, a sample module list ModuleList.csv (module numbers and student numbers) and a sample room list RoomList.csv (room numbers and their accommodation capacity). Format of relevant parts of the aforesaid three files are the same as those provided and contained in the second set of code and data StudyOfTimeTableDataProvidedAndResults\_CodeAndData. This folder also contains an xml file savexml.xml. Please do not alter or delete this file, it is used by the Windows exe file mentioned below.
3. Description of the requirements of the three data files:

1/ They must use the names as stated.

2/ DraftTimeTable.csv:

Column index 0 (Programme): this is the only column permits using comma and maximum number of comma in each cell is 4.

Column index 1 (Module Code): Module code should be in the form of two English alphabets and then followed with 4 digits. Example: KY3497, cN6245.

Column index 3 (Category): either “Lecture”, “Practicals” or “Tutorial” (no quotation marks).

Column index 4 (Term): either “1”, “2”, “3” or any combinations of them with “&” as the separator such as “1&3”, “1&2&3” (no quotation marks).

Column index 5 (Day): days of the week in full spelling or, for example, “Mon/Wed/Fri” or “Tue-Sat” (no quotation marks).

Column index 6(Start): time in 24-hour format. For example: 09:30, 13:00

Column index 7(End): time in 24-hour format.

Column index 8(Hours): For example: 01:30, 03:00.

Column index 11(Staff): If there are more than one teacher, use slash(“/”) to separate them. Example: “Albus Dumbledore”, “Minerva McGonagall/Severus Snape”(no quotation marks).

Column index 13(Room type): Room numbers. If more than one room, use slash(“/”) as the separator. Example: “EB.1.07”, “MLT”, “ITC03/KD.2.28/EB.2.44”(no quotation marks).

3/ ModuleList.csv:

Module Code should be at column index 0 and please refer to column index 1 of DraftTimeTable.csv mentioned above for its format.

Student number of a module should be at column index 1 at a subsequent line(or lines) after the line containing the module code. Preceding a student number (that is, at column index 0) there must be one and only one word “Count”(no quotation marks). Example: ”Count,67”. If a module has been given more than one number, only the maximum one will be used.

4/ RoomList.csv:

Room number at column index number 0 and room capacity at column index number 3. Example: “EB.1.42,???,East Building,37”, “EB.1.42,,,37”(No quotation marks).

4. A jar file StudyOfTimeTableDataAndResults.jar which is a zip file of the java files contained in the folder StudyOfTimeTableDataAndResults\_JavaCode mentioned above.

5. An exe file StudyOfTimeTableDataAndResults.exe which is a Windows exe file converted from StudyOfTimeTableDataAndResults.jar with Launch4j.

Items included in StudyOfTimeTableDataProvidedAndResults\_CodeAndData:

**(This set of code and data is for analysing data provided only)**

Items included in this set is the same as those contained in StudyOfTimeTableDataAndResults\_CodeAndData except corresponding names are changed accordingly and:

In StudyOfTimeTableDataAndResults, module information is required to contain in one file (ModuleList.java). However, there are two module lists provided, namely, (SM0469.csv and SM0477.csv). Some of the modules listed in the two files are overlapped and whenever such scenario encountered, only the student number contained in SM0469.csv will be used. Since having to deal with two module lists, the class InputAndTidyUpModuleList.java is not identical to that of StudyOfTimeTableDataAndResults.

Please also note that in this set, there is a significant number of modules listed on the draft time table timetable-example.csv but not listed on the module lists and vice versa. These modules cannot be analysed and warnings are given at results.

**Items included in TestTimeTableProjectMethodsWithJUnit\_Code:**

Included in this set are classes containing methods copied from the StudyOfTimeTableDataAndResults set and their JUnit test classes. A project may be set up in an IDE with all classes and test classes of this set properly placed there and run and observe the tests.

**Installation and operation methods:**

**(Note: make sure that the computer has installed java 8 jdk. For running in console, installation of java 8 jdk is a bit more complicated)**

1. Run the jar or the exe files: placed the jar and/or the exe files and the corresponding data folder on Windows desktop. Click on the jar or the exe file icon. A further window for each task clicked will then pop up informing the location and the name of the file created for containing the results. Then close the windows. File names are in the form of Task[number]\_[date and time].csv (for Task1 and Task2) or .txt (for Task3 and Task4). Example: Task2\_1712181324.csv, Task4\_1712181324.txt. Result files are located at the same folder as the data file. It is advised that for showing proper format of txt files created, they should be opened with Notepad++ or other similar code editors, not Notepad.

2. Run the java files: either (1) set up a project in an IDE (eclipse, intellij or otherwise) and place all java files in it, place the corresponding data folder in the module root directory and then run the Launcher.java class; or (2) place all java files and the corresponding data folder in a directory and run Launcher.java at console. The rest are the same as stated in 1. Above.

**Change of data file names and their paths:**

Deviation from the data filenames and path arrangements as mentioned herein is **not recommended**. However, all path and filename information are kept at the beginning of the Launcher.java class as string variables and they can be changed if required.