Even Semester (2019)

Assignment Cover Letter

(Individual Work)

Student Information:		**Surname**	**Given Names**	**Student ID Number**
	1.	Sianandar	Jason	**2201796440**
	2.	Wirawan	Chan Elizabeth	**2201797001**

Course Code	**: COMP6571**	**Course Name**	**: Data Structures and Algorithm**
Class	**: L2AC**	**Name of Lecturer(s)**	: Kartiko Eko
Major	**: CS**		
Title of Assignment (if any)	: LRT Reservation System		
Type of Assignment	**: Final Project**		
Submission Pattern			
Due Date	**: 29 - 06 - 2019**	**Submission Date**	**: 29 – 06 -2019**

The assignment should meet the below requirements.

1. Assignment (hard copy) is required to be submitted on clean paper, and (soft copy) as per lecturer's instructions.
2. Soft copy assignment also requires the signed (hardcopy) submission of this form, which automatically validates the softcopy submission.
3. The above information is complete and legible.
4. Compiled pages are firmly stapled.
5. Assignment has been copied (soft copy and hard copy) for each student ahead of the submission.

Plagiarism/Cheating
Binus International seriously regards all forms of plagiarism, cheating and collusion as academic offenses which may result in severe penalties, including loss/drop of marks, course/class discontinuity and other possible penalties executed by the university. Please refer to the related course syllabus for further information.

Declaration of Originality
By signing this assignment, I understand, accept and consent to BiNus International terms and policy on plagiarism. Herewith I declare that the work contained in this assignment is my own work and has not been submitted for the use of assessment in another course or class, except where this has been notified and accepted in advance.

Signature of Student: (Name of Student)
Jason Sianandar
Chan Elizabeth Wirawan

- **Problem description**

The public transport variety is getting better in Jakarta. These past few years, we see the development of public transport growing rapidly in Jakarta. The few examples are the development of Transjakarta Busway, Transjabodetabek Bus Series, and at this stage, the development of the MRT and the LRT. The future of Jakarta looks bright but we have one weakness which is that our scheduling system is not that strong for the public transports in Jakarta and by making this program, we are trying to find a way to make the scheduling system more effective for the public transports in Jakarta, specifically the Busway and MRT.

In Jakarta, public transport scheduling is not that efficient. We want to make easy access and efficient scheduling. People may be interested in the solution because they can access to LRT and bus scheduling in one program rather than accessing the scheduling one by one.

To solve this problem, we try to create this reservation system where the passengers can see all the accurate schedules of the LRT and they can also reserve a seat in the train so it is easier for them to get the seats in the LRT train.

- **Proposed alternative data structures to solve the problem and the analysis of how the choice of data structures can work to solve the problem.**

We try to create a reservation / scheduling system using C++ language. This program will use the vector data structure. Vector are sequence containers representing arrays that can change in size. The difference between an array and a vector is that vectors is dynamic and automatically resizes itself whenever a new element is added. When an element is added or removed from a vector, it automatically allocate the size of the containers to fit the current elements.

There are some advantages of using vectors. One advantage of vectors is that it has unlimited size unless we as the user assign it to the vector. Because of its unlimited size, the vector can readjust its size to fit the elements added to the vector or when elements are deleted from the vector. Another advantage of vector, is that there are built - in functions of vector such as pushback and pop which makes it easier for the user to add or remove item from the vector.

We can also use array for this program but since an array is not really considered as a data structure by the lecturers. Array is a type of static array where you must initialize the initial size of the array. But it’s inefficient for a big system like this project.

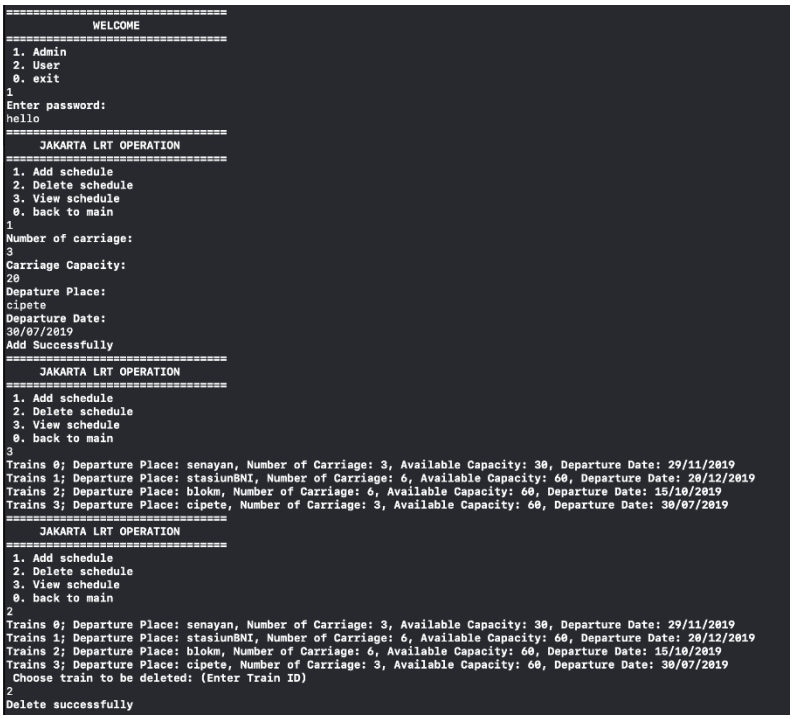
Implementation

a. <vector> = a dynamic array that can resize itself automatically as an element is added or deleted from it.
b. <fstream> = a header that can manipulate data to file such as read, write data to files
c. <string> = a header that introduces string and have function for string types.
d. <time.h> = a header that can manipulate and get data of current time and date.

For this project, we use vector since we are creating a reservation system for a public transportation which is very complex and large. We need a data structure that can dynamically allocate its size, efficient to use, and simple to use. Because in this reservation system, we need to add or delete a lot of data for the passengers that reserve the seats in the MRT such as name, id, etc.

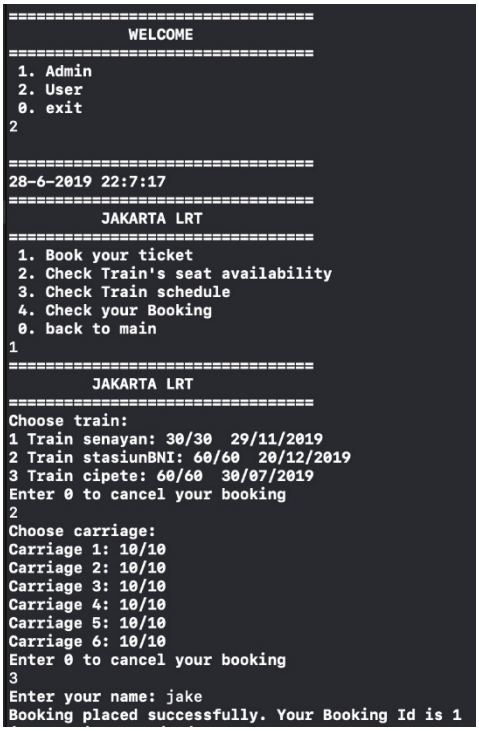
In this project, we also implement fstream to write the data that we inputted into a txt file. We implement string and we also implement time.h to get the current time and date.

- **Program manual, how to execute (with screenshots)**

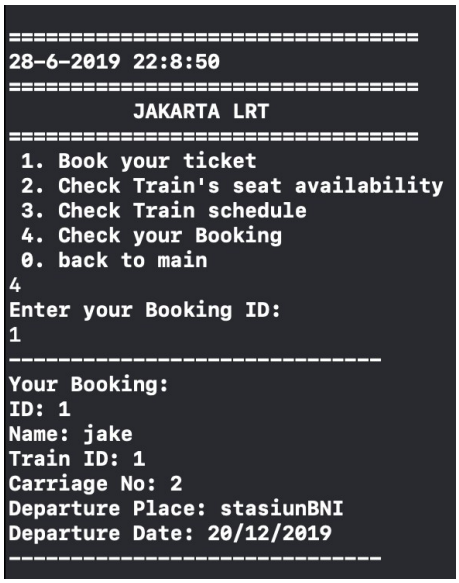


This program is called LRT Reservation System. This program allows the user to choose between admin or user.

If the user chose to be the admin of this program, the user can delete or add a schedule to the MRT Schedule. This will be added to a txt. File using the fstream implementation. As you can see in the image, if the user chose to be an admin, it will require a password to login, which in this case is "hello". After that, you have the option to add a schedule, view the schedule, and also delete a schedule. If we choose to add a schedule, we would need to input the number of carriage, the carriage capacity, the date and place of departure. All this will be stored in a vector. After that we can also view the list of schedules. After that we can also delete the schedule from the vector.



From this image, we can see the options we can do if we chose to be the user of this program. We can book our ticket by choosing the train based on the date and the destination we want to go. After that we must choose the carriage which still has available seats. If we fulfilled all those requirements, we just need to input our name and our booking will be successful with a Booking ID.



In this image, we can see what happens when we check our booking based on our booking ID.

```
================================
28-6-2019 22:7:44
================================
          JAKARTA LRT
================================
 1. Book your ticket
 2. Check Train's seat availability
 3. Check Train schedule
 4. Check your Booking
 0. back to main
2
================================
          JAKARTA LRT
================================
Available seats:
1 Train senayan: 30/30  29/11/2019
2 Train stasiunBNI: 59/60  20/12/2019
3 Train cipete: 60/60  30/07/2019

================================
28-6-2019 22:8:39
================================
          JAKARTA LRT
================================
 1. Book your ticket
 2. Check Train's seat availability
 3. Check Train schedule
 4. Check your Booking
 0. back to main
3
------------------------------
Train 0;
Departure Place: senayan
Number of Carriage: 3
Available Capacity: 30/30
Departure Date: 29/11/2019
------------------------------
Train 1;
Departure Place: stasiunBNI
Number of Carriage: 6
Available Capacity: 59/60
Departure Date: 20/12/2019
------------------------------
Train 2;
Departure Place: cipete
Number of Carriage: 3
Available Capacity: 60/60
Departure Date: 30/07/2019
------------------------------
```

Here is the part where you can check the train schedule. It shows the departure date, the number of carriage of the train, the available space, and the departure place.

- **Link to the application demo video (with max. length of 2 minutes)**
  https://github.com/ExtGmrJasonZ/CPP-Final-Project/blob/master/Screen%20Recording%202019-06-29%20at%2008.38.21.mov

- **Link to the GIT website (the code and soft copy of the report)**
  https://github.com/ChanElizabeth/DSA-Final-Project-2019

Resources:

1. http://www.cplusplus.com/reference/vector/vector/, 2019
2. https://stackoverflow.com/questions/3396378/change-the-working-directory-in-xcode
3. https://stackoverflow.com/questions/20707257/use-ifstream-to-read-multiple-lines
4. Zefanya Gedalya B.L.T