**Project**

**This is an individual project**. The objective of this project is to develop a tool rental management system (for stores such as Home-Depot) using the **MVC compound pattern**. The project can be implemented using either Java, or C#.

If you are an online student, you are required to record a short video of 10 ~ 20 minutes to demonstrate your project. You may put your project video on YouTube or google docs. When you submit your project to CougarView, you have to include the link for your video in your submission.

The system is made of the following entities:

1. **Customers**: They rent tools in the rental system. Customers are stored in a file called “Customers.txt”.
2. **Tools**: These are the objects to be rented. Tools are stored in a file called “tools.txt”.
3. **Rental\_data**: These are the recorded information about who rented which tool, the date of rental, and date of return. Rental\_data are stored in a file called “rental\_data.txt”.

**File Structure**

**Customers.txt**

Customer\_id; name; deleted

if **deleted=0** then the Customer is active in the system, if **deleted=1** then the Customer is logically deleted.

Example:

300; Thomas Jefferson; 0 (active customer)

200; John Smith; 1 (deleted customer)

**tools.txt**

tool\_id; tool\_name; rented

if **rented=0** then it is available (not rented), if **rented=1** is rented.

Example:

1; Drills & Drivers; 0

2; Bolt Cutters; 1

3; Handheld Sanders; 1

**rental\_data.txt**

rental\_id; customer\_id; tool\_id; date\_out; date\_in

date\_out and date\_in should be stored in format mm-dd-yyyy (if the tool has not been returned, then date\_in should be empty).

Every time a tool is rented, a record should be placed in the file “rental\_data.txt”, and a record is updated every time a tool is returned.

Example:

10; 300; 1; 9-30-2017;

11; 300; 2; 9-30-2017; 11-02-2017

In the first case shown above, the customer Thomas Jefferson rented the tool “Drills & Drivers” on 9-30-2017

In the second case shown above, the customer Thomas Jefferson rented the tool “Bolt Cutters” on 9-30-2017 and returned it on 11-02-2017.

**Requirements**

The system should be able to handle the following transactions:

**For Customers:**

1. Create a Customer
2. Delete a Customer. A Customer does not get physically deleted from the system. A flag in the record in file just states if the Customer is “logically” deleted
3. Search a Customer by last name, first name and show its information
4. Show all Customers

**For Tools**

1. Search a tool by name
2. Rent a tool
3. Return a tool
4. Show all tools, showing current rented tools first
5. Show current rented tools
6. Show who has historically rented a particular tool, sorting in descending order by date of rental

**Restrictions**

1. Your design must follow the dynamics of the Model-Viewer-Controller design.
2. Every time the application starts, it should allow the system to find/choose the location of the files.
3. A Customer can rent at most **5** tools at a time. If a Customer tries to rent a **6th** tool, the system must tell the user that the Customer is unable to rent the tool.
4. The current rental business only keeps one copy of a tool. If the user tries to rent a tool that is currently rented, the system must tell that the tool is not available.

**What to submit?**

1. A class diagram in **PDF** or **PNG**.
2. A ZIP file containing your project source code. If you use Java NetBeans IDE, use the NetBeans feature to ZIP your project. If you want to use C#, you must use Visual Studio 2015 or newer version.
3. A Word or PDF document that explicitly describes which components are the Model, which are the Controllers, and which are Viewers.
4. A recorded a short video: If you are an online student, you are required to record a short video of 10 ~ 20 minutes to demonstrate your project. You may put your project video on YouTube or google docs. When you submit your project to CougarView, you need to include the link for your video in your submission.