# User manual

### Prerequisites

The following must be downloaded and installed to run the project.

* Java 8 <http://www.oracle.com/technetwork/java/javase/downloads/index-jsp-138363.html>
* NodeJS (version 6.x) <https://nodejs.org/en/> (ensure NPM is also installed).
* MongoDB - <https://www.mongodb.com/download-center#community>
* Maven - https://maven.apache.org/download.cgi

Unfortunately the data used for clustering appears to have been taken down from the University of Alberta website. The data used as part of the project in a zip file on DCU school of computing webspace, found here:

<http://student.computing.dcu.ie/~saccarn2/pokerUofA/pokerdata.zip>

Download and extract this zip file into an appropriate directory.

### Project set up

**Note:** MongoDB must be running as a service on port 27017 (default MongoDB port) in the background.

#### Steps

##### **Downloading project**

* Use git to clone repository **or** download zip file and extract to desired directory.
  + Use this to clone repo using http: <https://gitlab.computing.dcu.ie/saccarn2/2017-ca400-saccarn2.git>
  + Download zip file from here : <https://gitlab.computing.dcu.ie/saccarn2/2017-ca400-saccarn2/tree/master>

##### **Setting Up Data\_Processing**

* Go to Data\_Processing directory
* Using an editor, open com.saccarn.poker.dataprocessing.GameAnalyser.
* Go to constructor with one argument of the class- looks like this:



* Change the directory path to point where the poker data was extracted to.
* Using the command line, cd to Data\_Processing directory
* Run mvn install
* Using the command line, cd to AI\_Agent directory
* Run mvn exec:java -Dexec.mainClass="com.saccarn.poker.dbprocessor.DataLoader"
* Run mvn exec:java -Dexec.mainClass="com.saccarn.poker.dbprocessor.PlayerTypeClusterer"

##### **Running System**

* Using the command line, cd into AI\_Agent directory.
* Run mvn install
* Using the command line(cmd1), cd into Poker\_Server directory
* Run mvn install
* Open up a new command line(cmd2), cd into the ui/server module/directory.
* Run npm install
* Run npm install gulp -g . This installs gulp globally.
* With cmd1 run ‘mvn exec:java’ in the Poker\_Server directory.
* With cmd2, in the ui/server directory run ‘gulp’.

Wait a few seconds, and open up localhost:3000 in the browser. You should presented with the login page of the web app.

**Note:**

The web app listens for connections on port 3000 - this can be changed by editing the PORT\_NUM variable in the app.js file in ui/server.

The java server listens for connections on port 3500. This can be changed by changing the port variable in the Server class in the Poker\_Server module. The JAVA\_PORT variable in app.js of ui/server must also be changed to match this number.

**Unit Tests**

The junit tests of every java module can be run by going into the module directory using the command line and running:

* mvn test

The output to the console will include how many tests were run, how many were successful and how many failures there were.

#### **Validation Module**

To run the classes in the validation module

* Go the validation module directory using the command line
* To run opponent model tests
  + Run mvn exec:java -Dexec.mainClass="com.saccarn.poker.opponentmodels.TEST\_NAME" where TEST\_NAME is the name of test class.
* To run bet pass parameter model tests
  + Run mvn exec:java -Dexec.mainClass="com.saccarn.poker.testsbetparams.TEST\_NAME" where TEST\_NAME is the name of test class.
* To run ‘common hand’ tests
  + Run mvn exec:java -Dexec.mainClass="com.saccarn.poker.testscommonhandvalues
  + .TEST\_NAME" where TEST\_NAME is the name of test class.
* To run ‘hand potential’ tests
  + Run mvn exec:java -Dexec.mainClass="com.saccarn.poker.testshandpotential
  + .TEST\_NAME" where TEST\_NAME is the name of test class.