**A Report on the Economic Analysis with Matrices’ Project (ECO2048)**

Group Name: ***Welcome to*** ***The Matrix!***

Project Title: **The Million Pound Drop**

***Group contributors:***

1. Jigar Amin (6401225)
2. Manjeet Singh (6357572)
3. Tien Cong Nguyen (6362789)
4. Duc Nguyen (6379385)
5. Catherine Ojo (6364690)

**Introduction**

During our first meeting as a group on the 20th of November, 2016, we all brought up various ideas so we could select the best idea we would finally work with. Some of the ideas included a 3-equation macro model of the economy, the million pound drop, black jack and poker game. At the end of the meeting, the million-pound drop became the group’s best choice because of the interesting concept that lays behind the game which captivates on-lookers and creates anticipation for a player’s bid choices. Being an already existing UK television show, it was more realistic for everyone to understand the outcome we had to target taking into consideration our different levels of strengths and weaknesses.

**Startup**

At the initial stage, we set up a platform for communication (Facebook group) and created our Github accounts so we could communicate amongst ourselves even whilst we were on Christmas break for efficiency purposes. We created sketches of how we wanted our GUI interface to look like and began to work towards it.

The game was designed to have 7 rounds of questions, selected randomly from five genres: history, sports, entertainment, science and geography. Each team member was delegated a genre where we had to create 35 questions for each so as to avoid repetition. Every round has a total of 25 questions where each genre is represented by 5 questions with options ranging from a-d.

We also distributed specific roles among ourselves so we could be more time efficient in doing the work. The roles ranged from the Graphical User Interface (GUI) for the game, the sounds, background colours, timer settings, bidding options and call back functions. However, the delegated tasks did not limit our contribution to other areas when needed.

**Work developments and processes**

The GUI interface was created where we began with the first round and start screen of the game. Then we moved on to creating options in the display\_ans.m file designed to inform the player of the correct answer and the amount he or she will move with onto the next round.

After working similarly on each of the rounds, we progressed to the final round (round 7) where codes were written to end the game once the round ends or if there were no winnings at all (that is, winnings=0).

**How to play the game**

Before you begin the game, it is important to note that you have been allocated one million pounds in the game. Run the file StartScreen.m and it will open the GUI, select the ‘play game’ button. After clicking play game, a new GUI is created called Rounds which will display a box for each of the following: Question, options, bid on options, balance and timer alongside a push button which allows you to lock in your bids and check which option was correct. You have 1 minute to bid your allocated one million pound to the available options. For instance, if you are certain that the correct answer is Option C, you can bid all the money to Option C, or if you are indifferent between two options, Option A and B, you could bid 500,000 pounds to Option A and 500,00 pounds to Option B. After 1 minute has passed, select the check answer button and the correct answer will be shown. If you had bid for the right options, you’ll move on to the next question with the available money at hand, otherwise game over. Also, please note that if you have not bid all of your money into the boxes then game over. This continues to the seventh round as long as you still have money left to bid on your options. If you successfully complete the seventh round you have completed the game and you will be shown your winnings and a ‘game over’ message.

**How to run the Code**

To play the game you need to run the StartScreen.m file. This creates a GUI which provides a welcome notice and displays a push button labeled play game, which then starts the game.

**Limitations during work process**

-We had a few challenges while creating the process. One of the issues was how to store the bids made by a player for each round. This created some bugs which was saving only the bids for the first round and creating errors for the other consecutive rounds when clicking the check button as the player would not be able to assign new bids.

-In addition, we had a serious trouble when adding the timer in. The timer will not reset to 60 seconds when it moves to the next round, instead it starts at the time which it ended with from the previous round. We did manage to fix that by adding some “if” conditions to the code and make the check button only available at the end of the 60 seconds.

-Furthermore, the balance box cannot illustrate the number “1000000”, instead it illustrates “1+e06”. This is because the GUI cannot show the number which is too big in the memory. We have tried to put it in the string format but it will have a trouble later on when adding the number to the balance box. We have no choice but to leave that error.

-Another problem we had to deal with was to ensure the winnings which were created in one function were updated as a player progressed. This meant in order to update this we needed to check call back which required it to be declared as a global variable.

-We also noticed a slight problem when using the desktop version of Github where each time we made changes to a branch and synced it, we were overwriting all other changes made to the online file if we did not sync the branch before starting to work on it. So we resolved to using the online version only.

**Mark Split**

‘The highest mark will be devoted to Jigar Amin as the core processes were done by him and he also helped the rest of the team members with their tasks.’

|  |  |
| --- | --- |
| **Names** | **Marks** |
| Jigar Amin | **24** |
| Manjeet Singh | **19** |
| Tien Cong Nguyen | **19** |
| Duc Nguyen | **19** |
| Catherine Ojo | **19** |