**The Lottery System**

**Sprint Implementation**

**Project Timeline: 24.8.2022 to 29.8.2022**

**INDEX**

1. Introduction …………………………………………………4

2.Scope …………………………………………………4

3. Purpose …………………………………………………4

4. Intended audience …………………………………………………5

5.Design Overview …………………………………………………5

5.1 Data flow Diagram …………………………………………………5

5.2 Flowcharts …………………………………………………7

6. System Architecture …………………………………………………12

6.1 Functions …………………………………………………12

6.2 Structures …………………………………………………13

7. Design Review checklist …………………………………………………16

8. Code inspection log …………………………………………………19

8.1 Code review checklist …………………………………………………19

8.2 Code review log …………………………………………………20

9. Tools report …………………………………………………21

9.1 gcov report …………………………………………………22

9.2 Splint report …………………………………………………25

9.3 Valgrind report …………………………………………………26

9.4 gprof report …………………………………………………27

10. Testing Report …………………………………………………28

10.1 Unit testing report …………………………………………………29

10.2 Integration testing report …………………………………………………30

11. Requirement Traceability Matrix(RTM)……………………………………………….33

12. Minutes of Meeting ………………………………………………...35

### 1.Introduction:

ZamoLand Development Authority (ZDA) plans to allot 100 plots to people of the city through a lottery. A token will be available on their website on first-cum-first-serve basis. Only 300 tokens are available. It will have a serial no which is in a pre-decided range. Once the timing is announced they have to grab the tokens by using their unique family id and they get an auto-generated confirmation regarding their participation in the lottery process. On every call a token number is auto-generated out of a list of 300 available tokens. The participant who owns the token will be notified. He/She will have to confirm booking within 5 minutes by paying Rs. 50000 as booking amount. Once done the plot is allotted in his/her name. If the winning participant does not confirm booking then automatically there is a repeating lottery for the same plot. Once a plot is booked its details are updated in the plots database. Another database is maintained which will have the participant details as well as the plot details for the plot he won.

**2.Scope:**

This project aims to create the development of an automated system of Lottery Process to assist ZDA Authority in storing and retrieving all the information about the registered participants and the winning participants in a way more robust and efficient manner. All the information about a particular participant is stored in a retrievable manner.

**3.Purpose:**

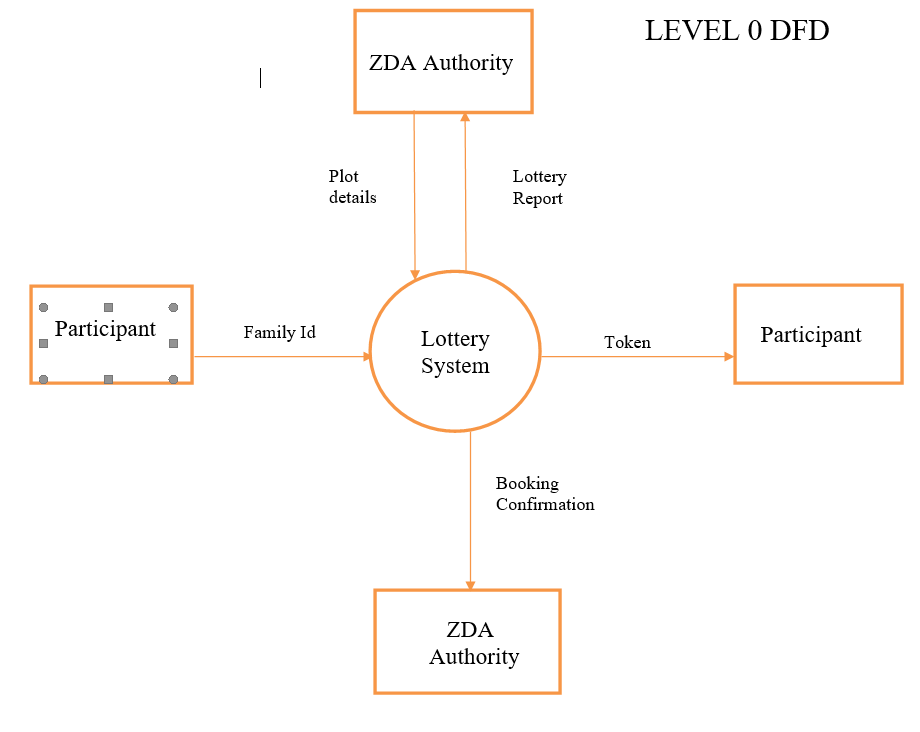
The purpose of this document is to describe the requirements to track all information of a participant allotted by the lottery process system and keep an organized details of all participants. The main purpose of Lottery System is to allot the plot to the people of the city through lottery.

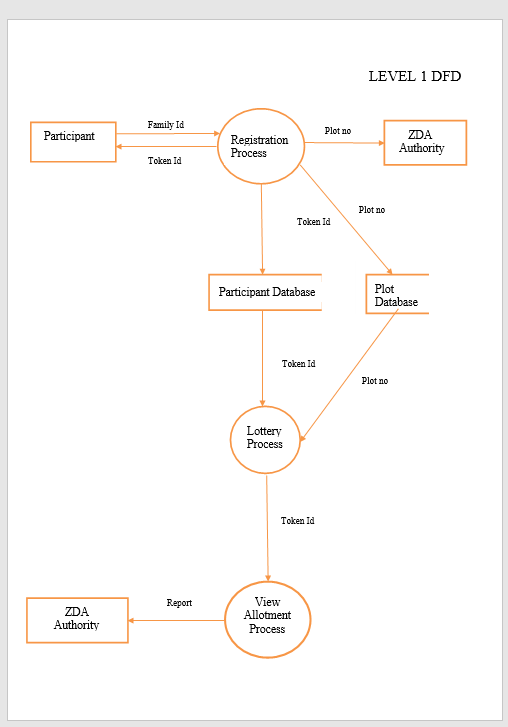
**4. Intended Audience:**

The target audience are the families of the ZamboLand who are participating in the Lottery process. Out of 500 households only 300 families are allowed to grab the token out of which only 100 families will be benifitted.

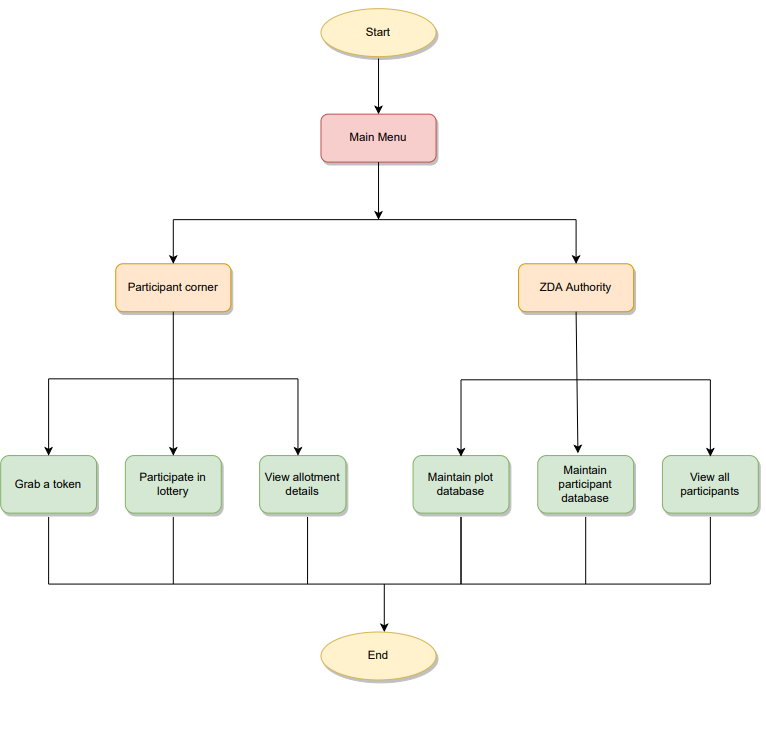
**5.Design Overview:**

**5.1. Data Flow Diagram:**

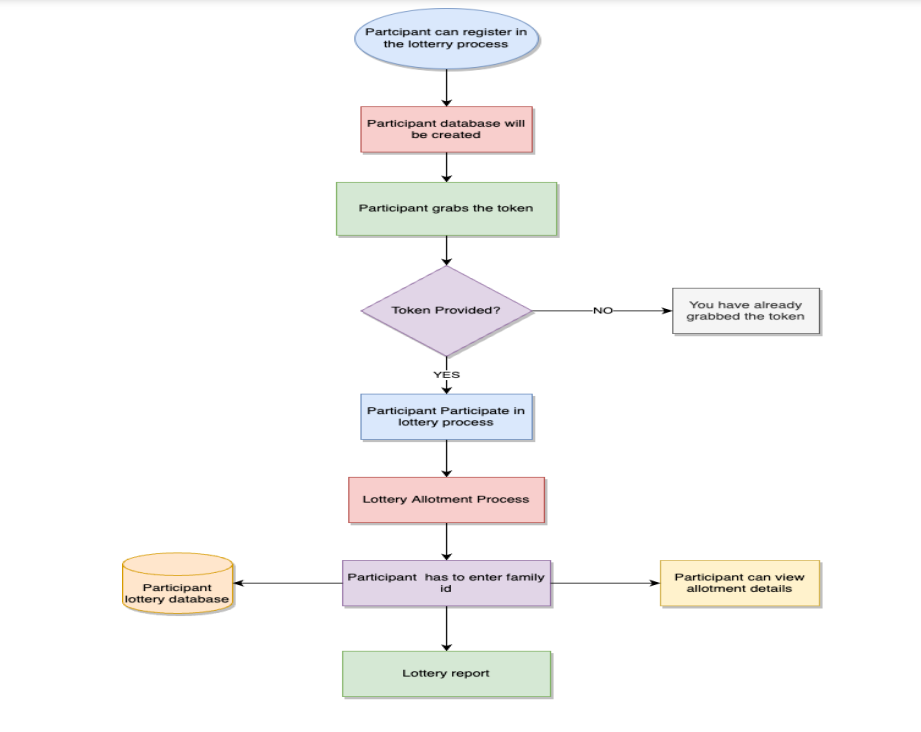
****

****

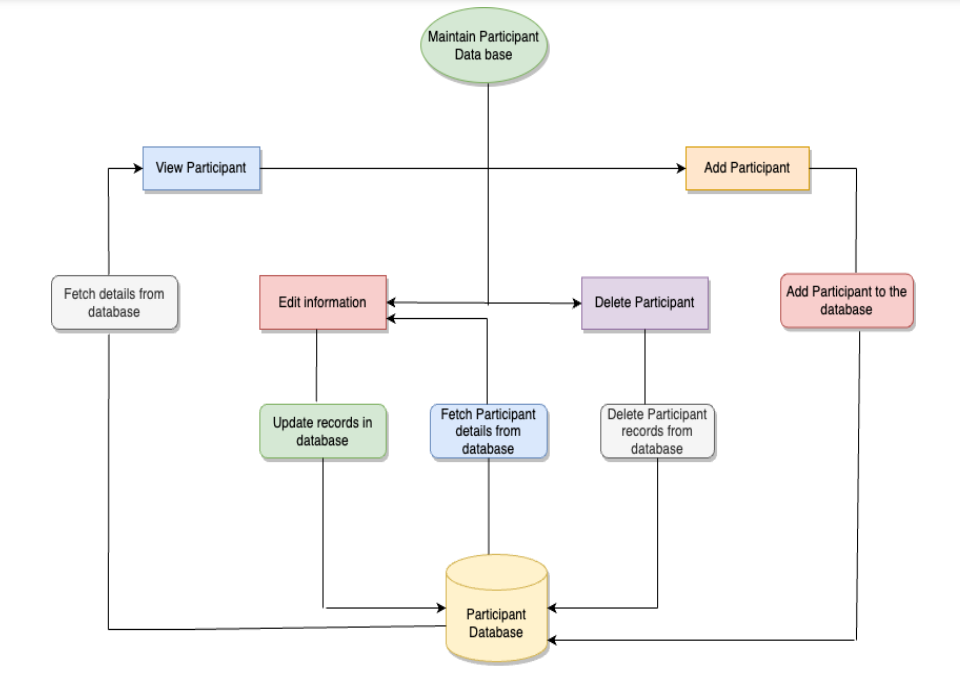
**5.4. Flowcharts:**

****

**Fig.5.4.1: Flowchart of Main Menu**

****

**Fig.5.4.2: Flowchart of Participant Corner**

****

**Fig.5.4.3: Flowchart of Maintain Participant Database**

**6. System Architecture:**

**6. System Architecture:**

**6.1.Functions:**

**6.1.1.Participant Corner**: Out of 500 households from the city only 300 participants are allowed to grab a token which is available on the website.2 tokens cannot be issued to a single family. So now the participant can register by entering their unique family id and can view the allotment details through the auto generated confirmation message.

**6.1.2.ZDA Authority**: ZDA Authority maintains plot database by adding, editing, deleting and viewing plot details. It also maintains participant database by adding, editing, removing and viewing all participants database. They can also view the Lottery report. ZDA authorities have to supply a password to access the admin features online

**6.1.2.1.Add Plot Details:** ZDA Authority adds plot details into database like plot no, Size (in sq. ft.), price of plot, etc.

**6.1.2.2.Edit Plot Details:** ZDA Authority can edit the plot details and modify the database as and when required.

**6.1.2.3.Delete Plot Details**: ZDA Authority can delete the plot details from the existing database.

**6.1.2.4.View Plot Details:** ZDA Authority can view the plot details.

**6.1.2.5.Add Participant Details:** ZDA Authority adds participant details into database like participant name, slno. (Auto-generated), family id, token no. Plot no., Amount to be paid. The last two fields are automatically updated after allotment.

**6.1.2.6.Edit Participant Details:** ZDA Authority can edit the participant details and modify the participant database as and when required.

**6.1.2.7.Remove Participant Details:** ZDA Authority can remove the participant details from the existing database.

**6.1.2.8.View Participant Details:** ZDA Authority can view all the participant details.

**6.1.3.View Lottery Report:** ZDA Authority can view the lottery report.

**6.2. Structures :**

**6.2.1.Participant:**

This structure is used to group participant type structure members like

-Family\_id

-name

-participated in lottery

-plot\_no

-token\_no

-size

-remaining\_amount

**6.2.2.Plot**

This structure is used to group plot type structure members like

-plot\_no

-size of plot

-allot

-price of plot

**6.2.3.Queue**

This structure is used to group queue type structure members like

-token

-self referencing pointer next

**7. Design Review Checklist:**

**7. Design Review Checklist:**

|  |  |
| --- | --- |
| Group name | Group 04 |
| Project Name | The Lottery System |
| Project Manager |  |
| Document Name | Design review checklist |
| Inspected on | 30.08.2022 |
| Inspected by |  |

**8. Code Inspection Log:**

**8. Code Inspection Log:**

**8.1. Code Review Checklist:**

|  |  |
| --- | --- |
| **Name of Originator** |  |
| **Name of Inspector** |  |
| **Name of moderator** |  |
| **Path to design document** |  |
| **Path to source code files** |  |
| **Name of the review module/ object (if applicable)** |  |

**8.2. Code Review Log:**

|  |  |
| --- | --- |
| **Project** |  |
| **Artifact Name** |  |
| **Artifact Type** |  |
| **Author** |  |
| **Reviewer** |  |
| **Prepration Time (mins)** |  |
| **No.of Pages Reviewed** |  |

|  |
| --- |
| **Filled in by Reviewer** |
| **Rating Codes** *(Filled by inspector)* |
|  |
|  |
|  |
|  |
|  |

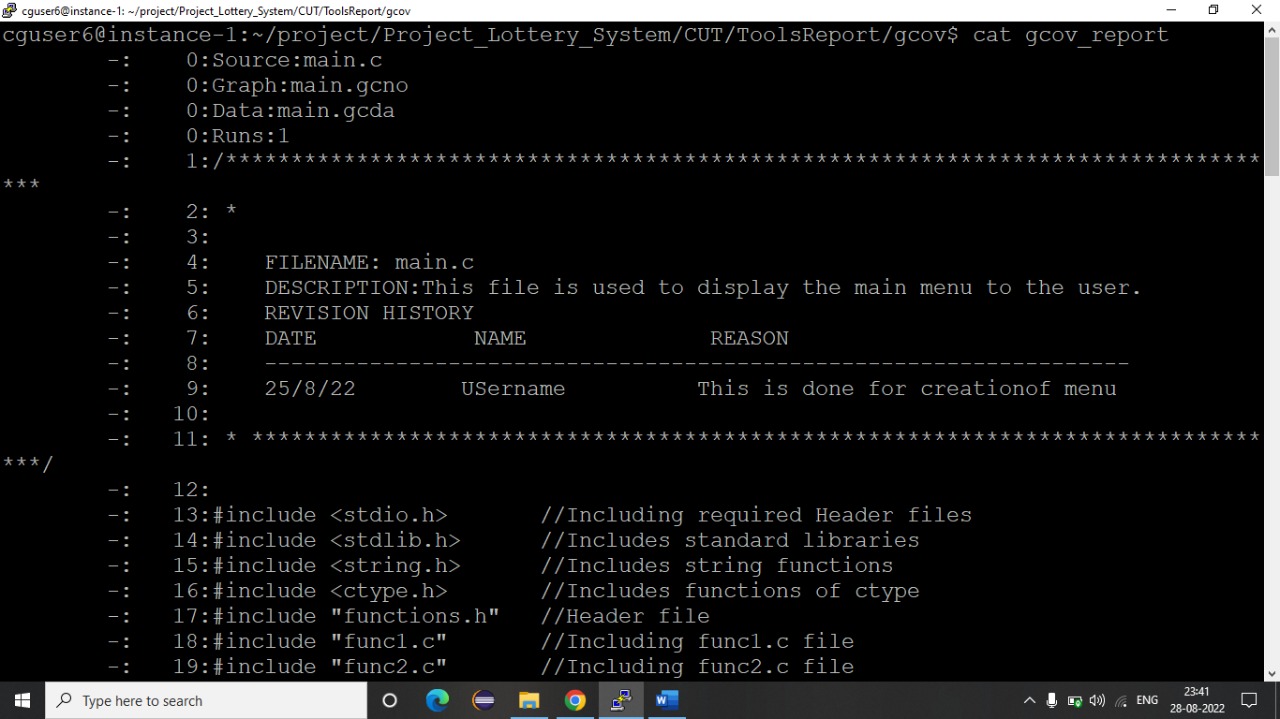
|  |  |
| --- | --- |
| **Filled in by Reviewer** | **After meeting Dispostion** |
| **Phase-Injected Codes** *(Filled in by inspector)* | **Disposition Codes** *(Filled in by inspector)* |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

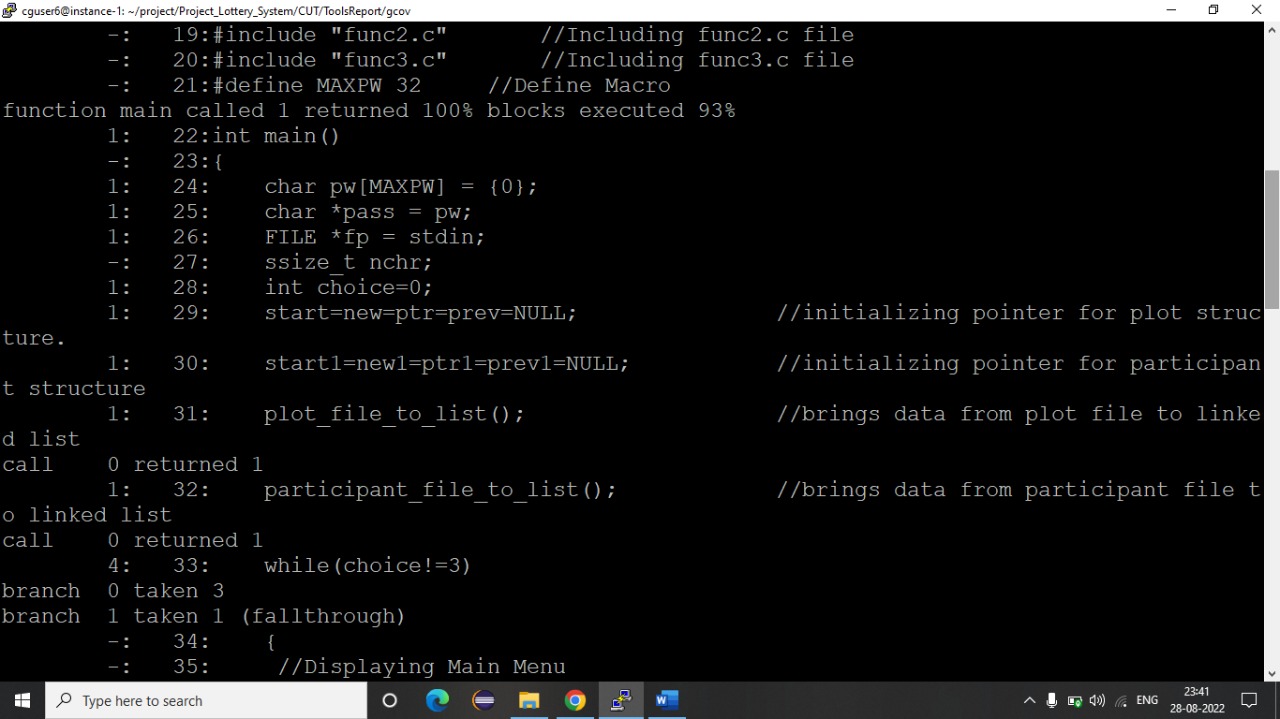
|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Location** | **Rating** | **Phase-injected** | **After-meeting defect disposition** | **Defect description** | **Disposition**  **Comments** | **Defect type/Root cause** | **Phase- detected** |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |

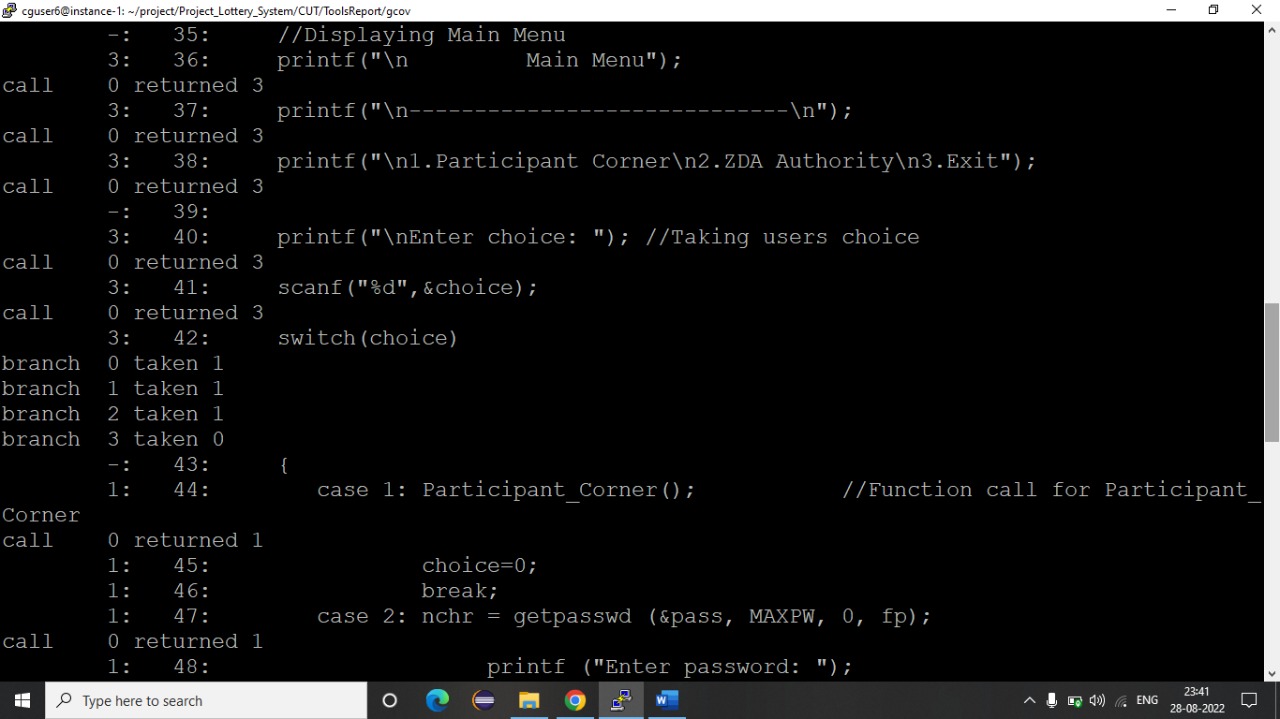
**9. Tools Report:**

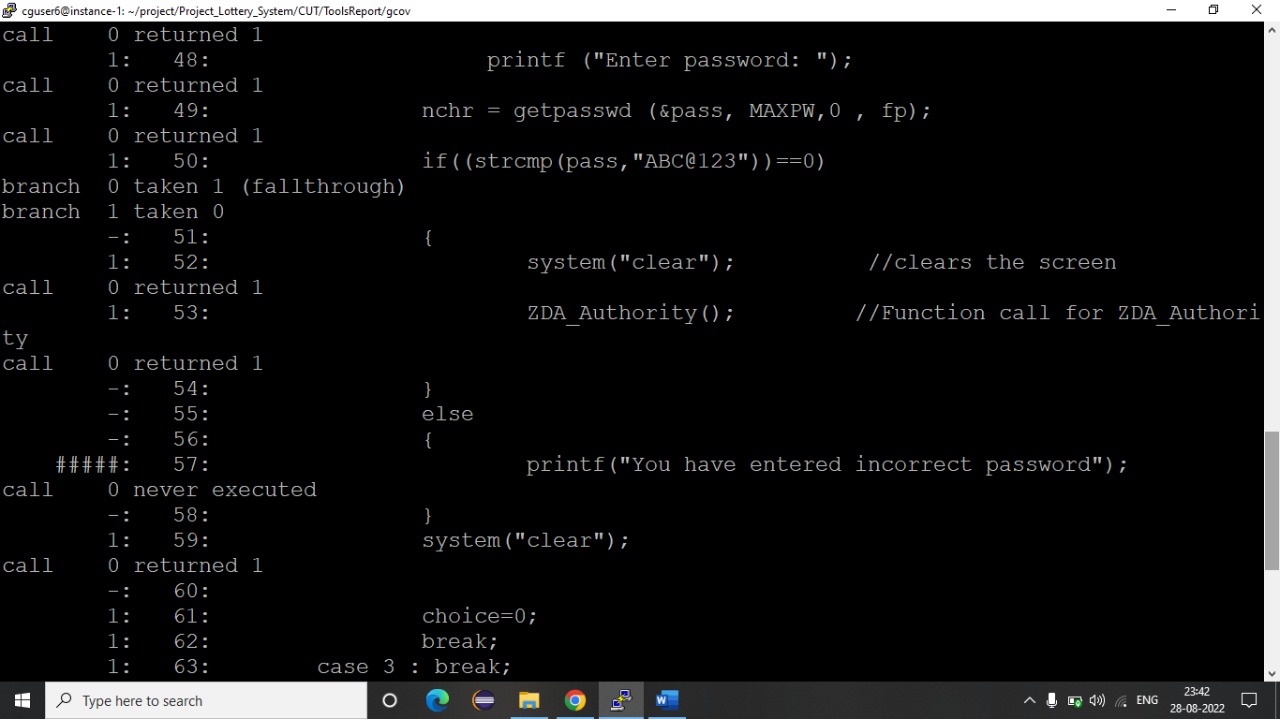
**9. Tools Report:**

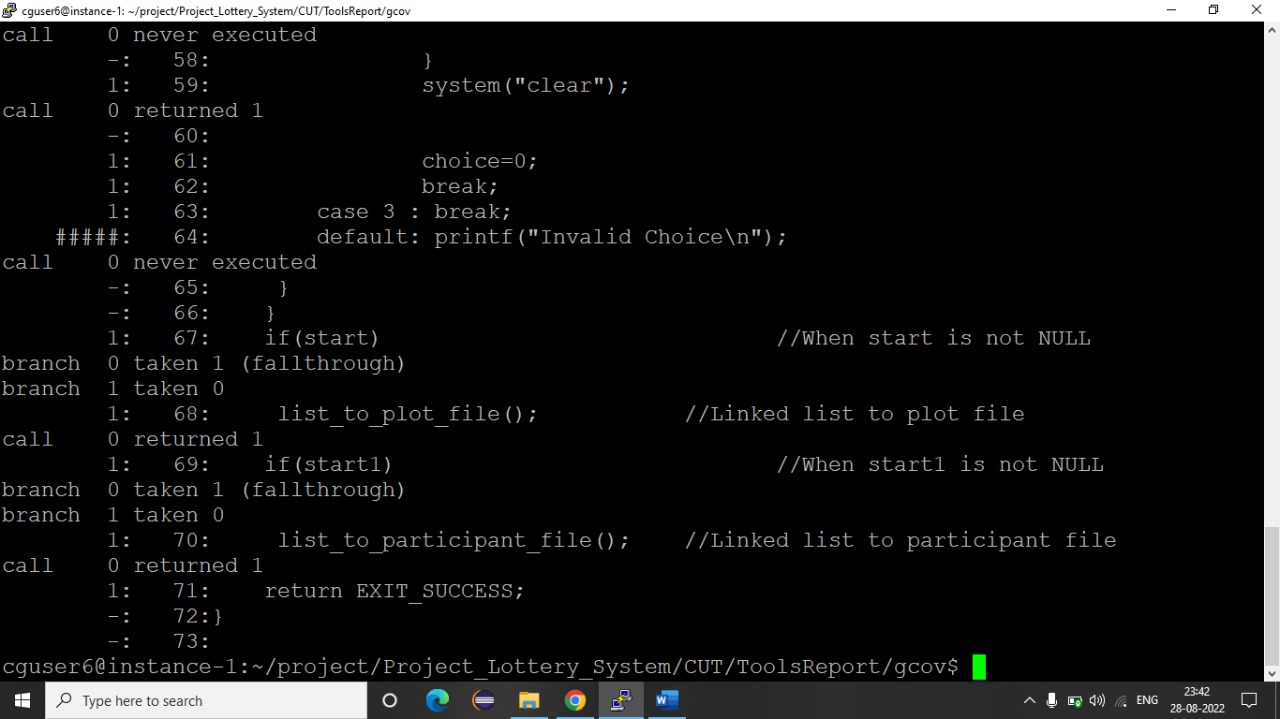
**9.1. gcov report:**

****

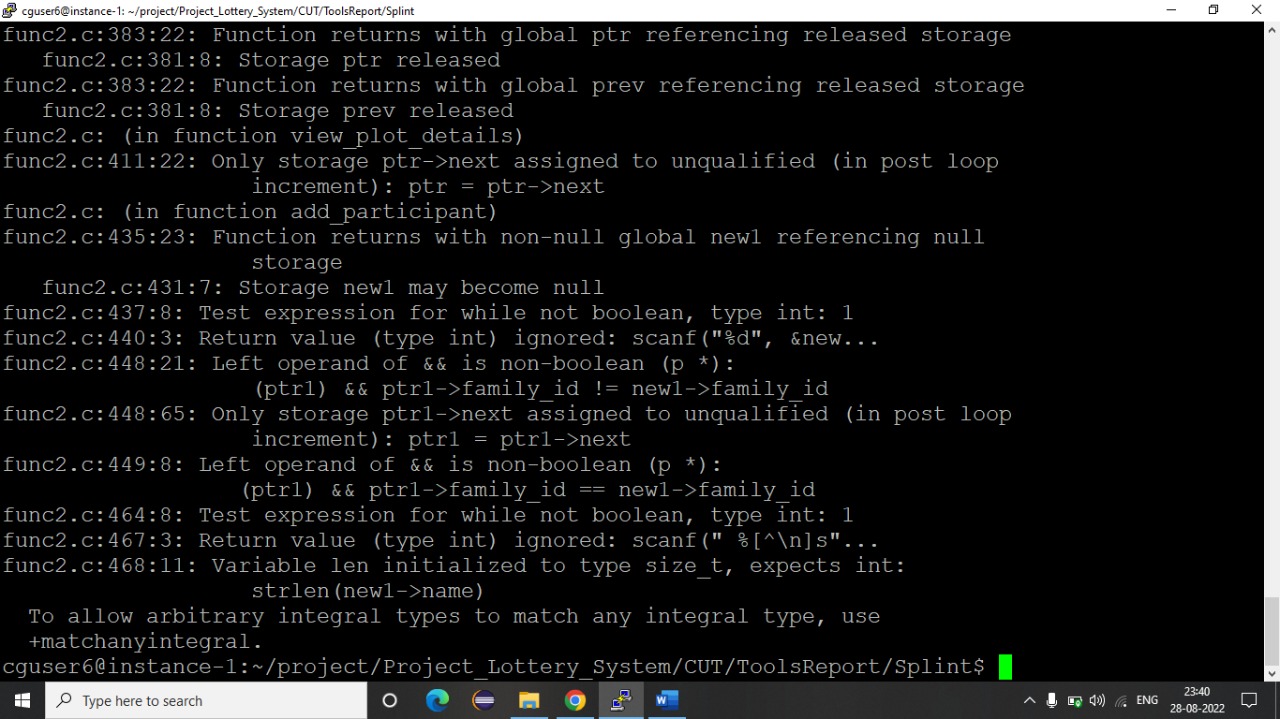
****

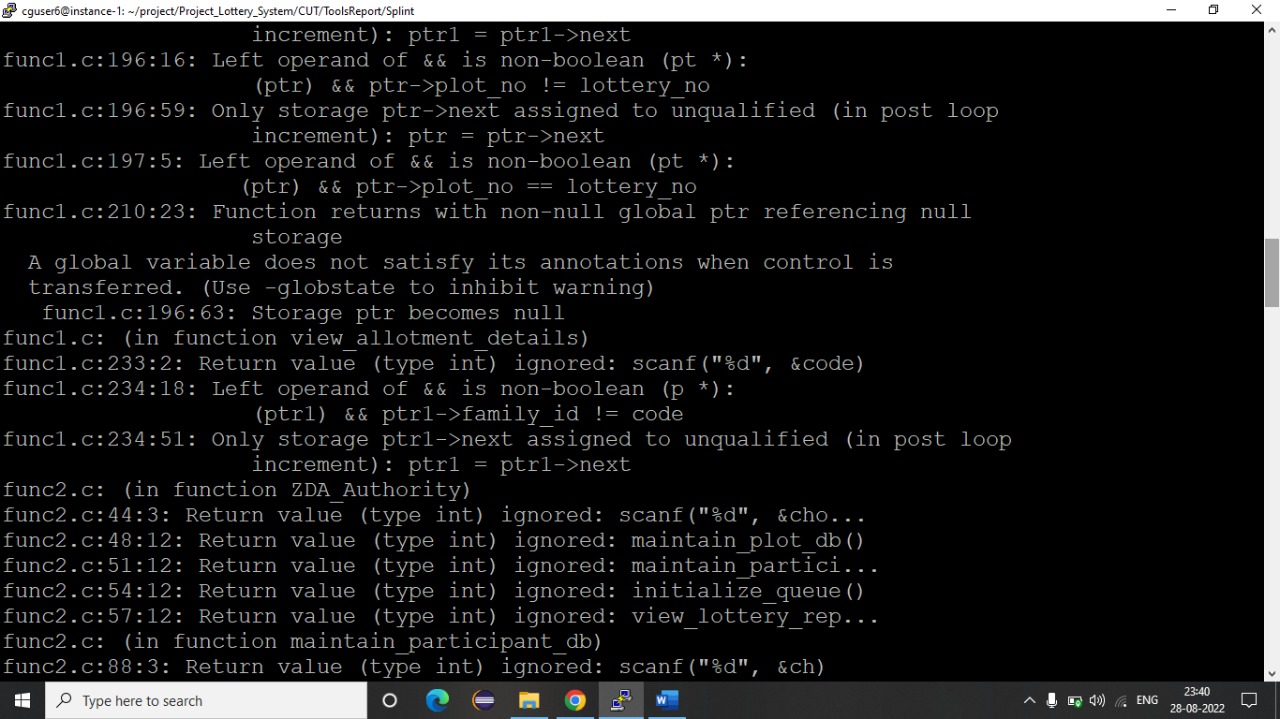
****

****

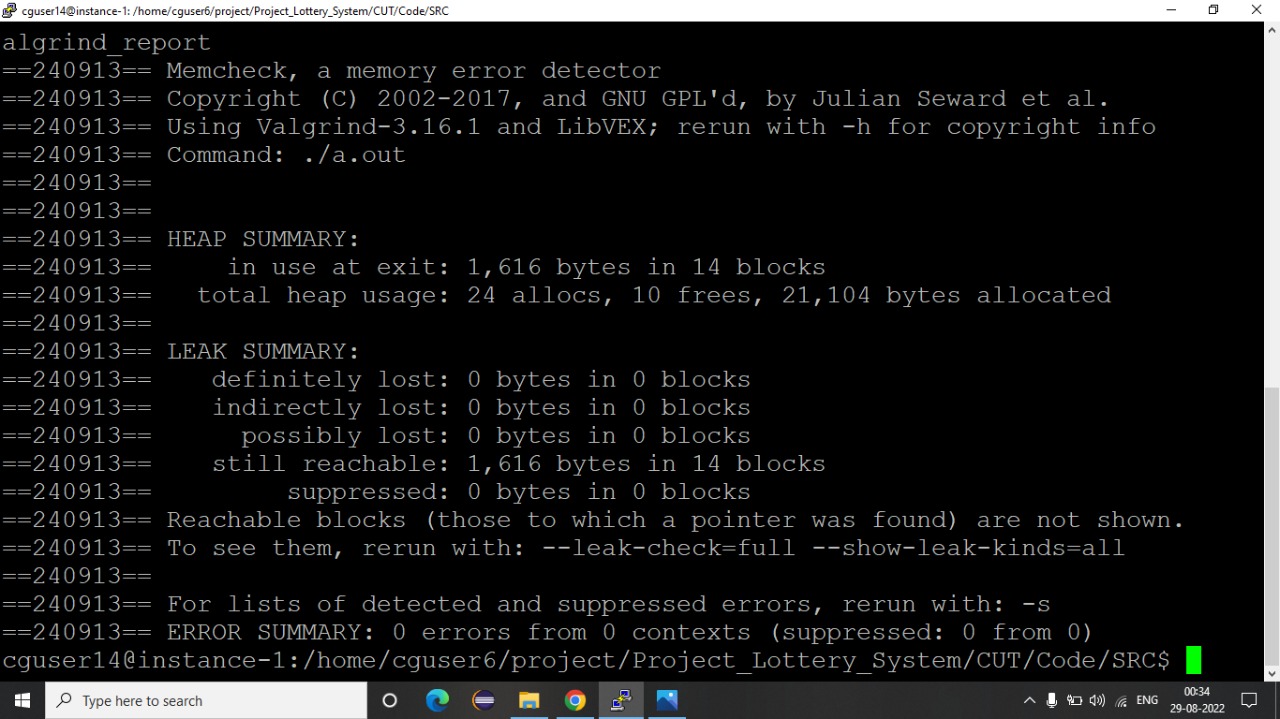
****

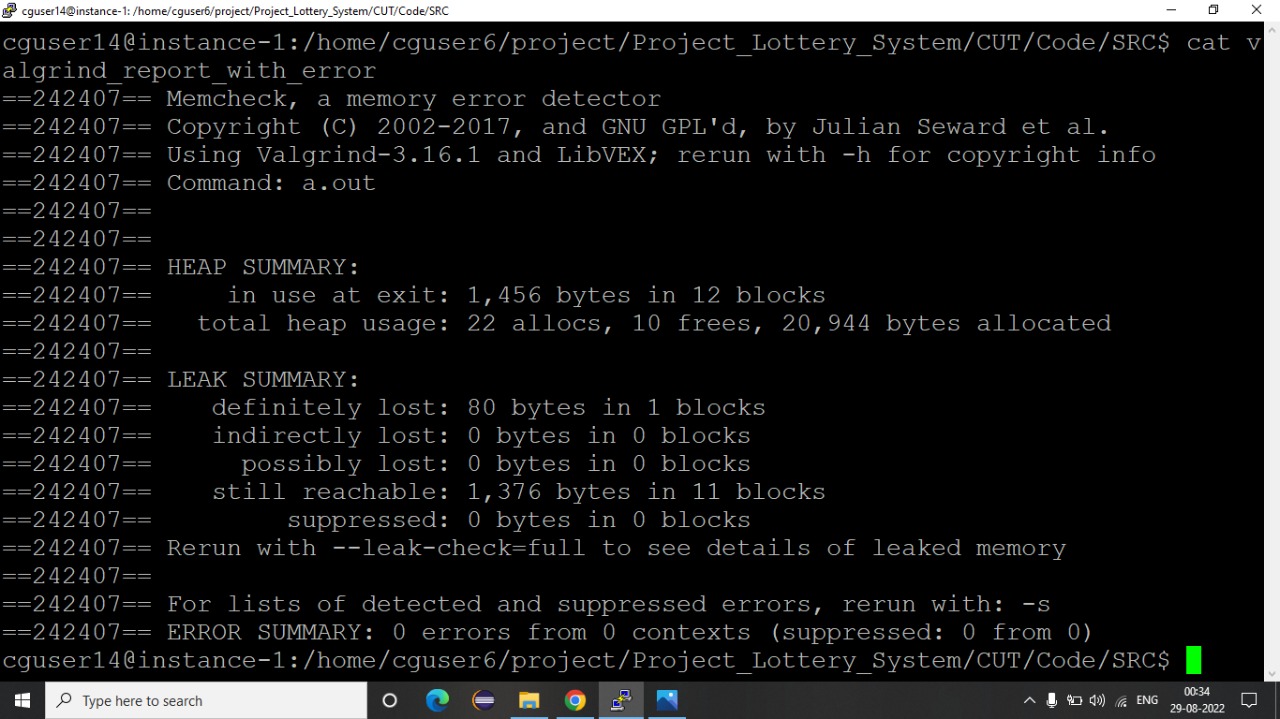
**9.2. Splint report:**

****

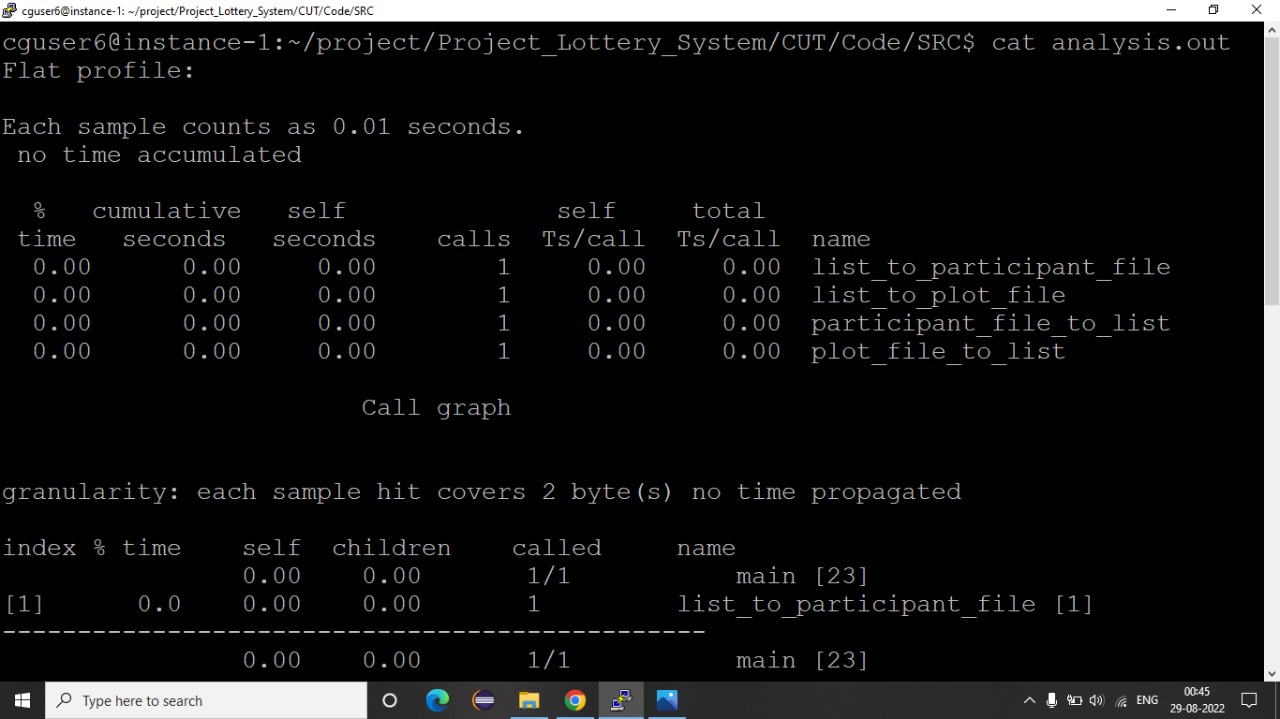
****

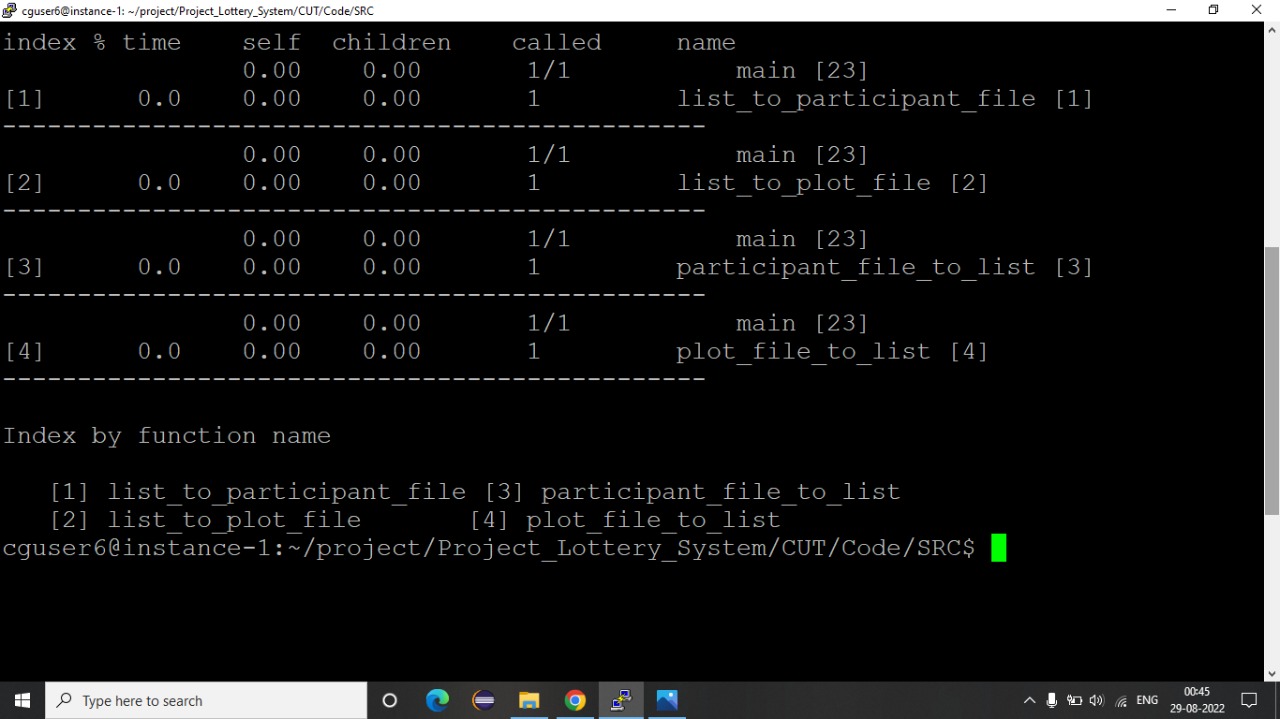
**9.3. Valgrind report:**

****

****

**9.4. gprof report:**

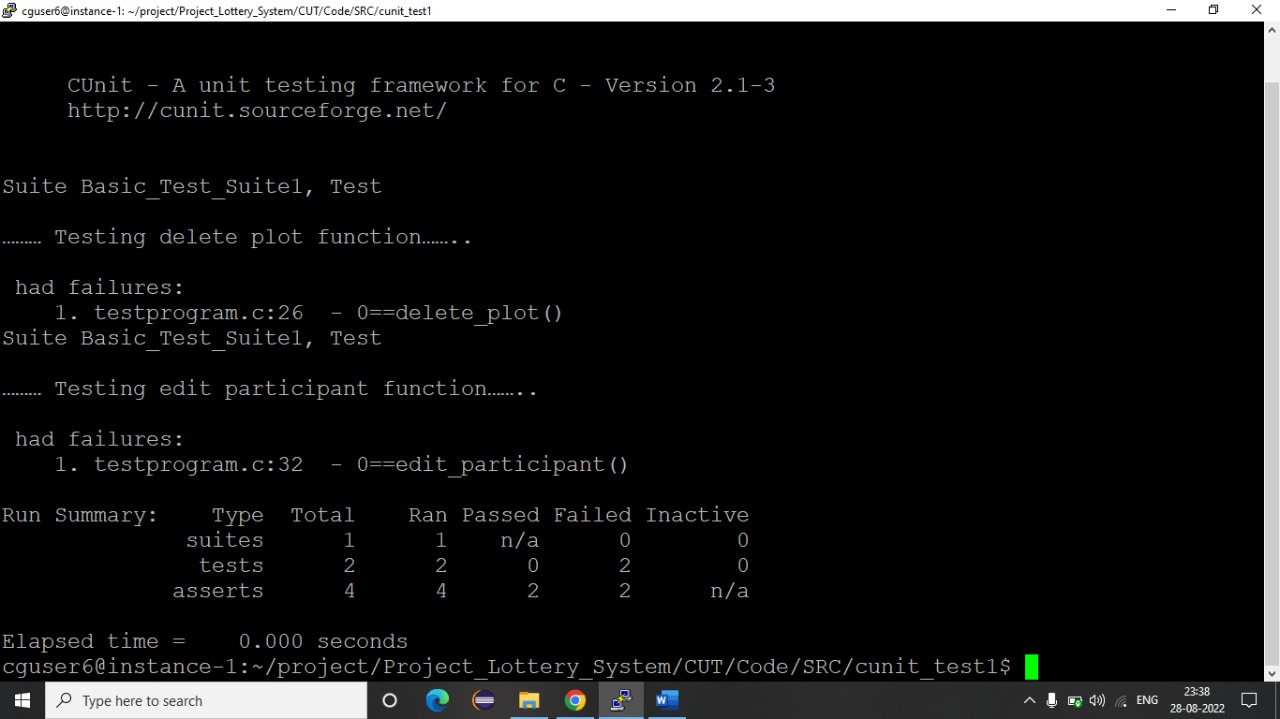
****

****

**10. Testing Report:**

**10. Testing Report:**

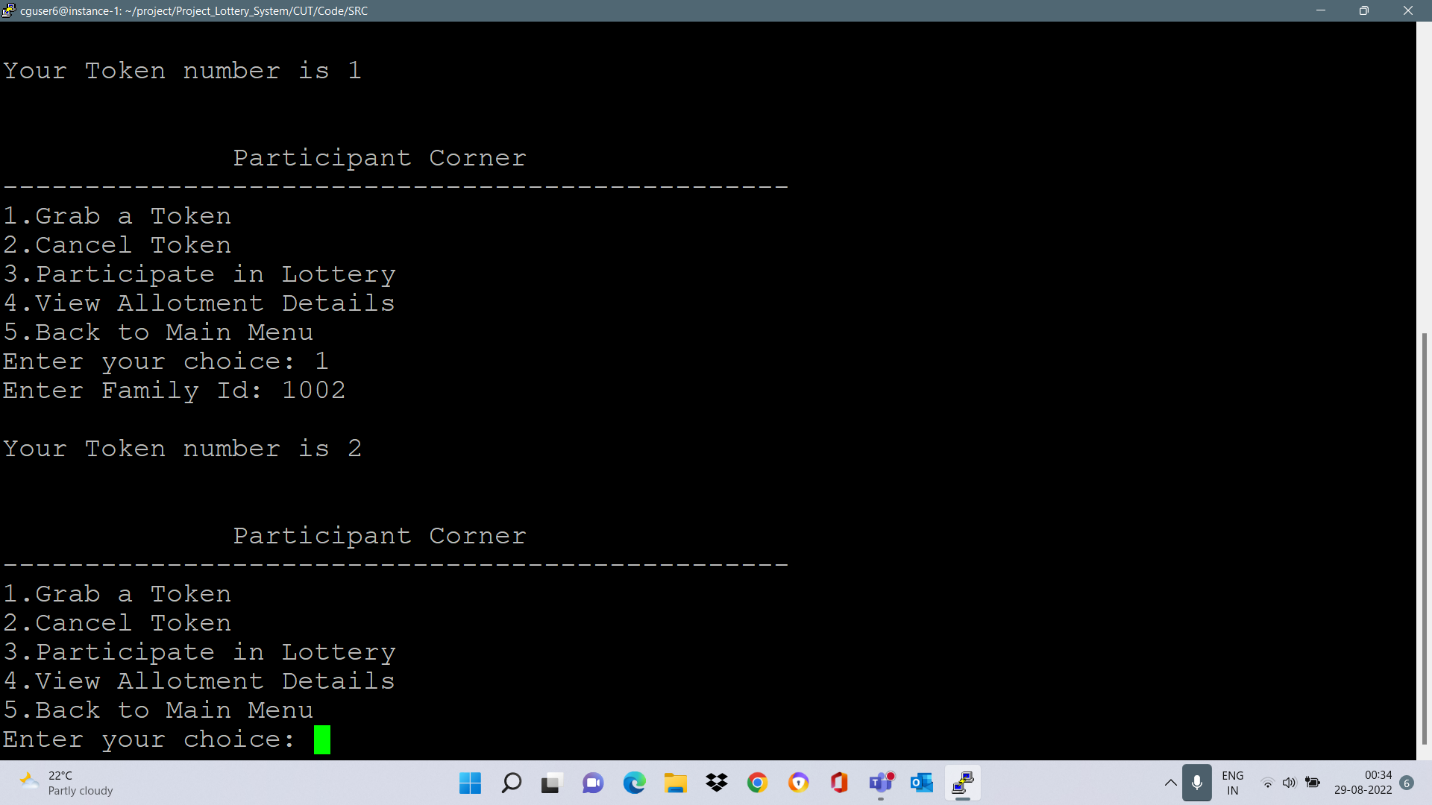
**10.1. Unit testing Report:**

****

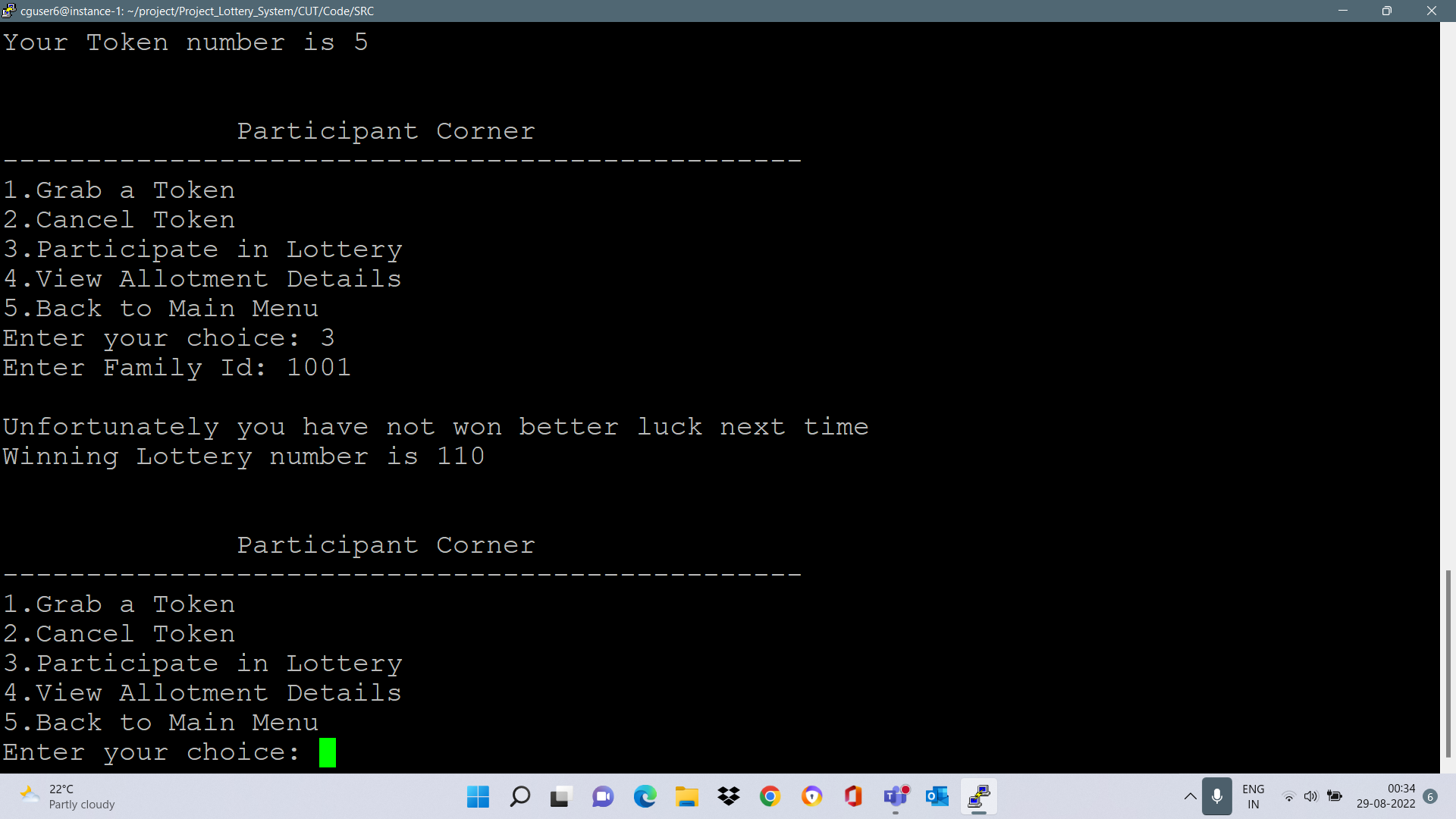
**10.2. Integration Testing Report:**

**Participant corner module**

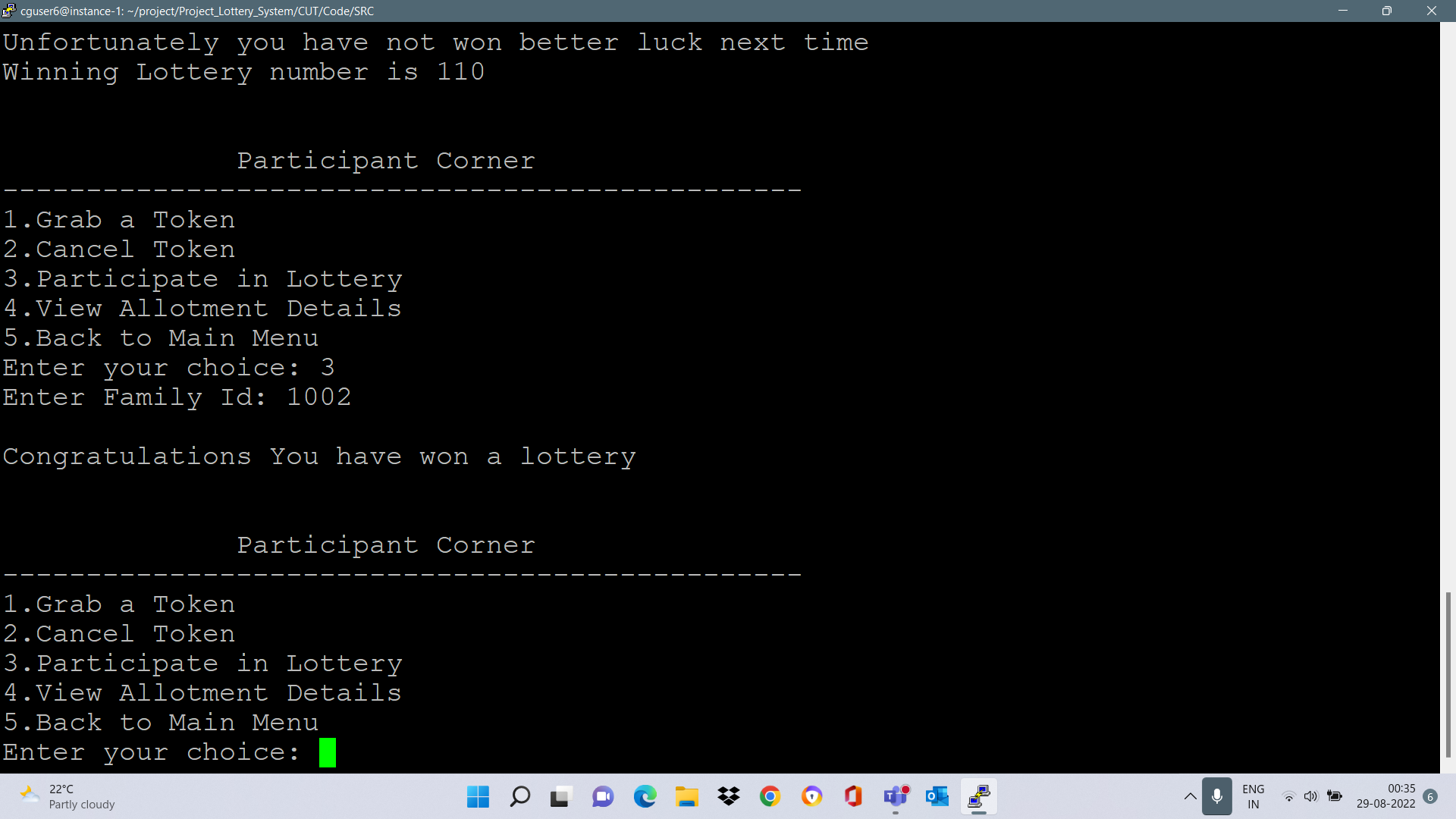
Case 1: Grab a token

****

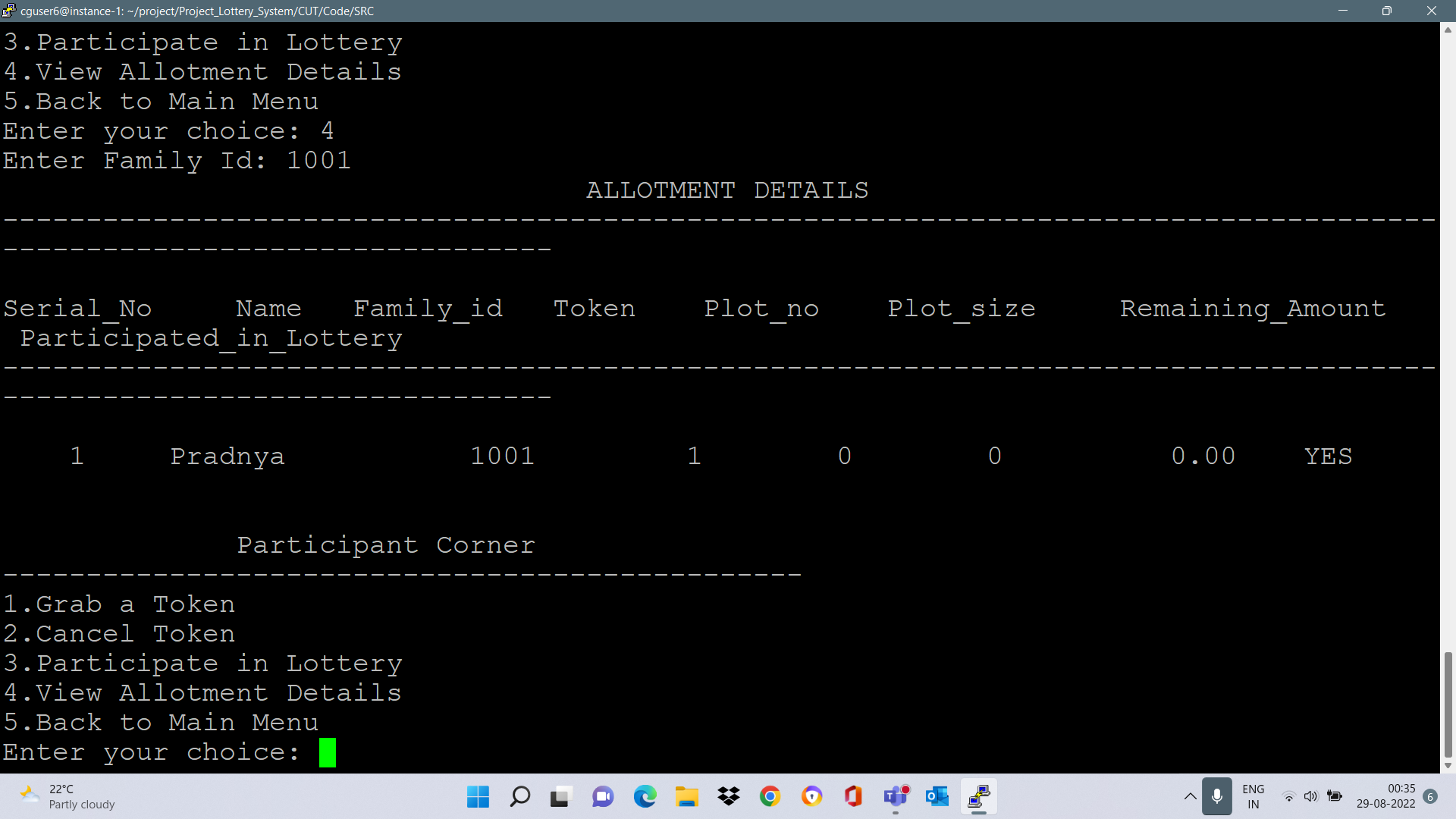
Case 2: Participate in lottery

****

Case 3: Participated in lottery and won

****

Case 4: Viewing allotment details

****

**11. Requirement Traceability Matrix (RTM):**

**11. Requirement Traceability Matrix (RTM):**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Requirement | Design Mapping | Code Mapping | UT Mapping | IT Mapping |
| LS\_01 | a | participant\_ corner |  | IT\_1 |
| LS\_02 | b | ZDA\_ Authority |  |  |
| LS\_03 | c | grab\_token |  | IT\_2 |
| LS\_04 | d | participate\_in\_lottery |  | IT\_3 |
| LS\_05 | e | View\_allotment\_details |  | IT\_4 |
| LS\_06 | f | add\_plot |  |  |
| LS\_07 | g | edit\_plot | Test\_case\_1 |  |
| LS\_08 | h | delete\_plot | Test\_case\_2 |  |
| LS\_09 | i | view\_plot\_details |  |  |
| LS\_10 | j | add\_participant |  |  |
| LS\_11 | k | edit\_participant | Test\_case\_3 |  |
| LS\_12 | l | remove\_participant | Test\_case\_4 |  |
| LS\_13 | m | view\_participant\_details |  |  |
| LS\_14 | n | view\_lottery\_report |  |  |
| LS\_15 | o | maintain\_participant\_db |  |  |
| LS\_16 | p | maintain\_plot\_db |  |  |
| LS\_17 | q | plot\_file\_to\_list |  |  |
| LS\_18 | r | list\_to\_plot\_file |  |  |

**12. Minutes of the meeting:**