The Lottery System

Sprint Implementation

Timeline:30.09.2022-7.10.2022

Group 6

INDEX

1. Overview

ZamoLand Development Authority (ZDA) plans to allot 100 plots to people of a small city of 500 households through a lottery. Token will be available on their website on a first_cum-first-serve basis. 300 tokens are available with a serial number in pre-decided range. When the participant grabs a token, a auto-generated confirmation is given. One household can grab only one ticket. In the lottery process, on every call, a token number is auto generated out the 300 available tokens. If the participant who owns the tokens confirms the booking, then the plot is allocated in his/her name or else lottery for the plot is repeated. Once the plot is booked, the plot database is updated, and lottery report is generated. Database is maintained for information regarding the participants as well as the plot details for the plot he won.

2. Goals

This project aims at creating and maintaining an automated system for lottery process to ZDA in storing and updating all information about the registered participants, winning participants, the available plots and the allocated plots in an efficient manner. All information stored can be retrieved at any given time.

3. Purpose

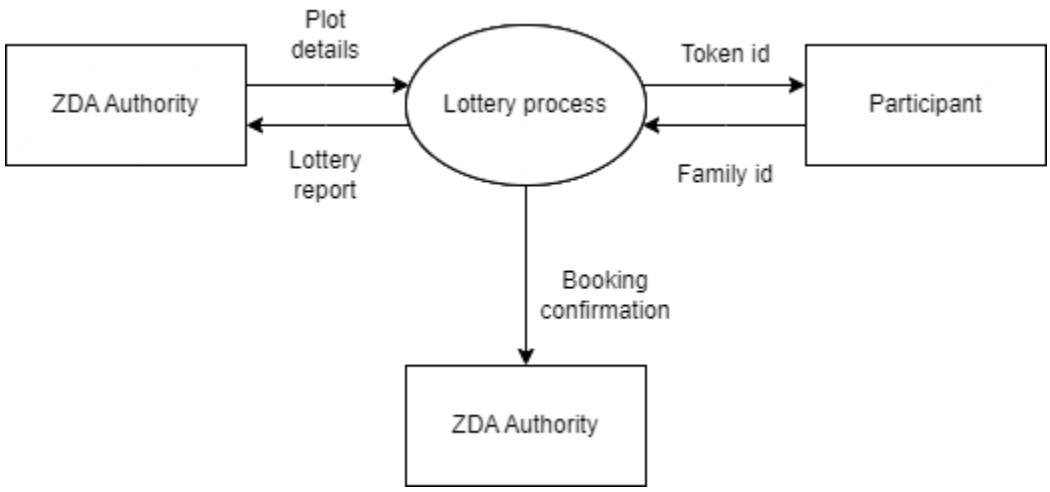
The purpose of this document is to track and record all the information and events occurring throughout the lottery process and keep the details organized, so that the plots can be allocated to the people smoothly.

4. Target Audience

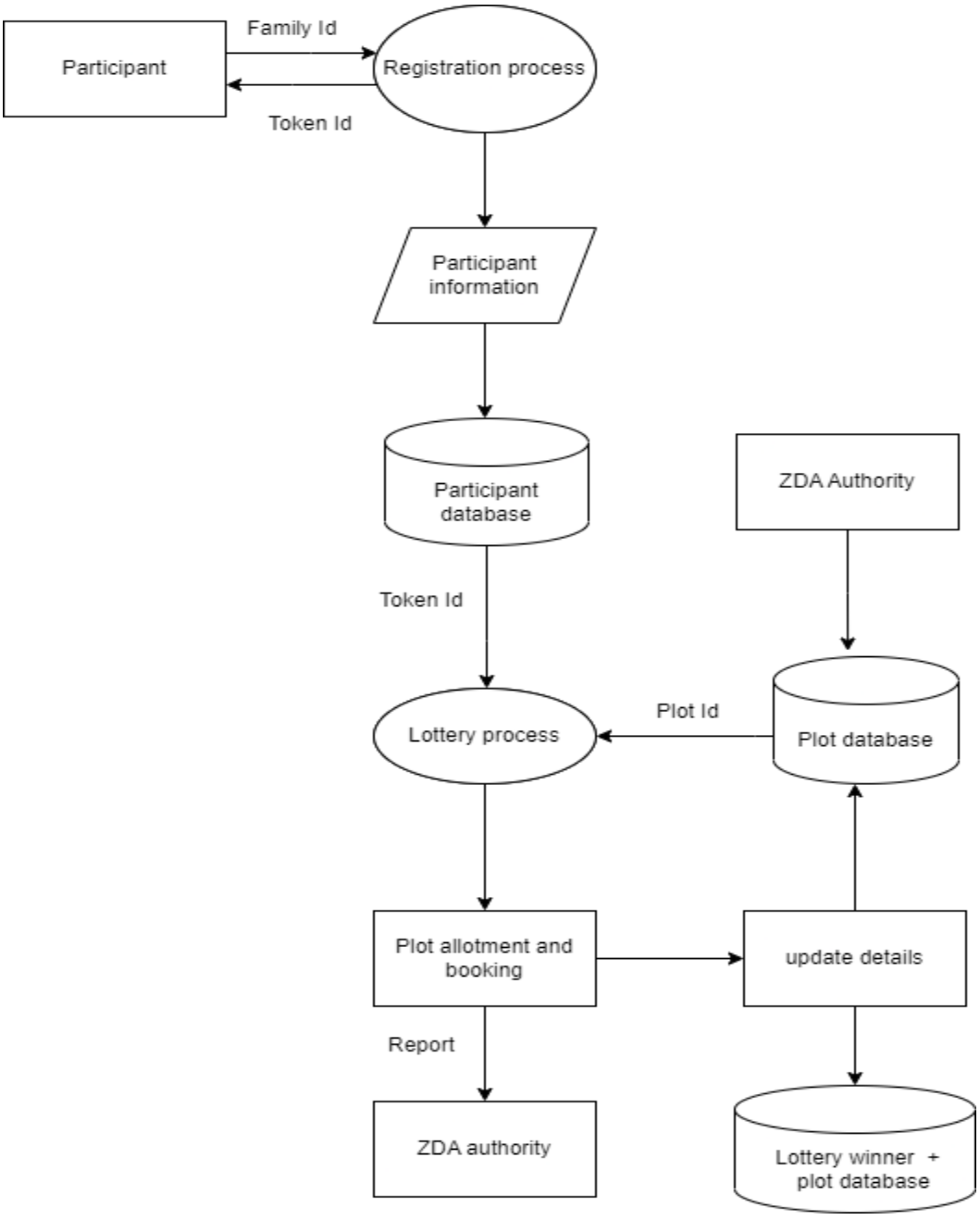
The target audience of this event are the 500 families residing in ZamboLand who are participating in the lottery process. Out of all those families only 300 households are allowed to grab the token for the lottery where only 100 will be able to win and book a plot.

5. Design Overview

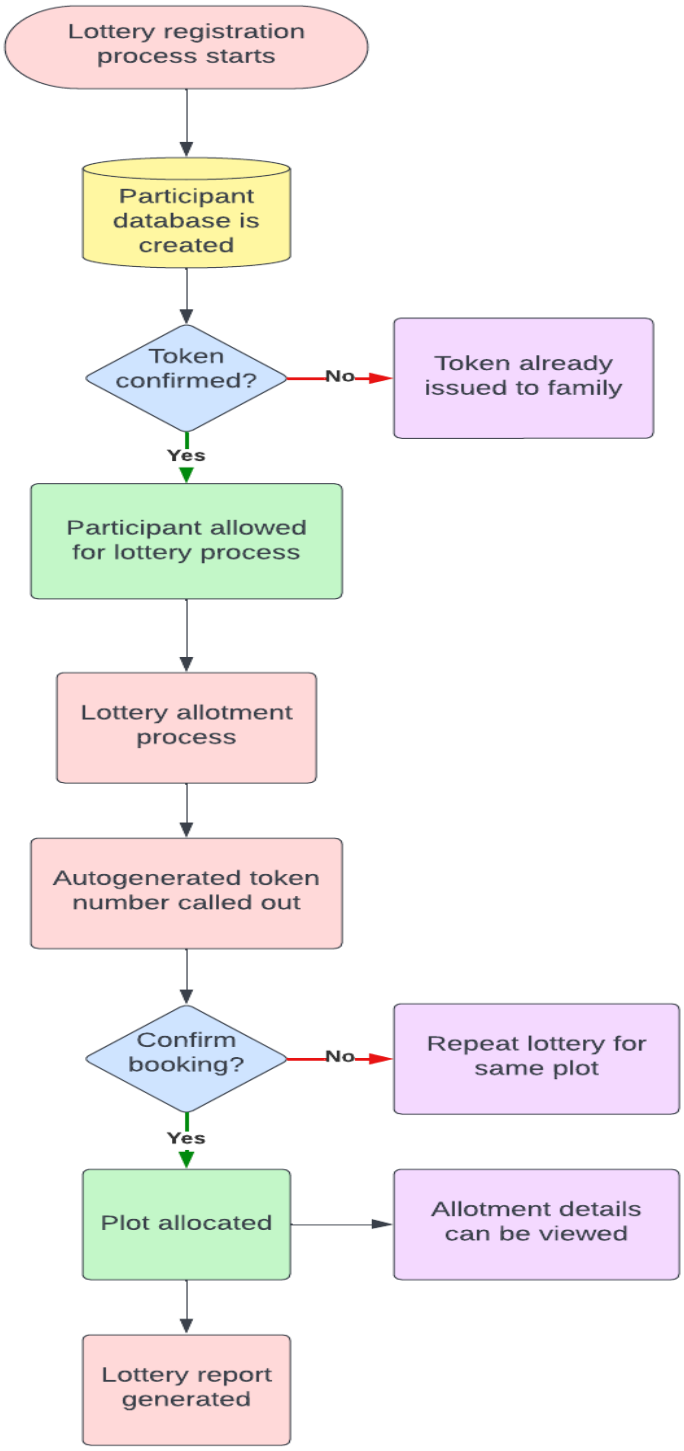
5.1. Data Flow Diagram Level 0:



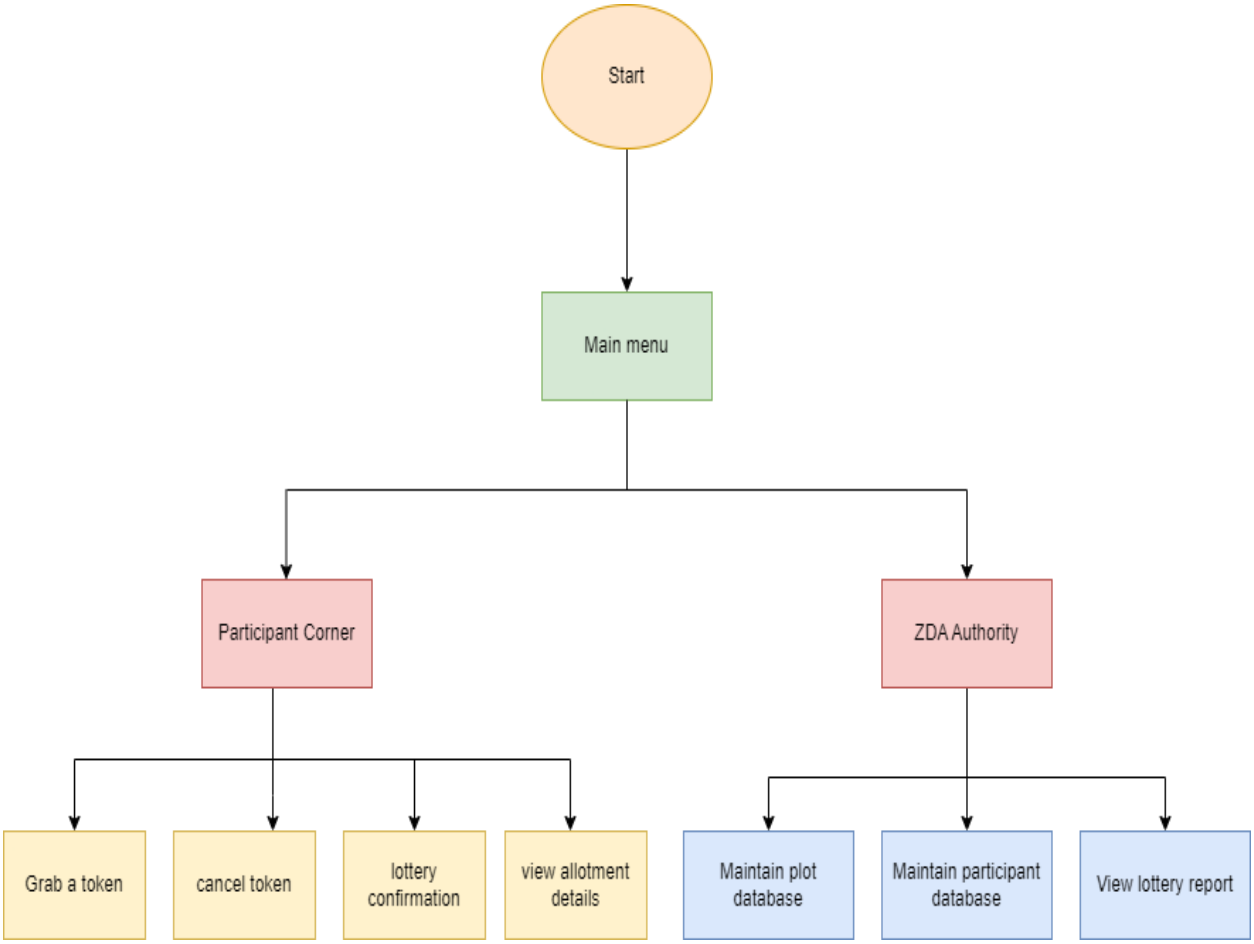
5.2. Data Flow Diagram Level 1:



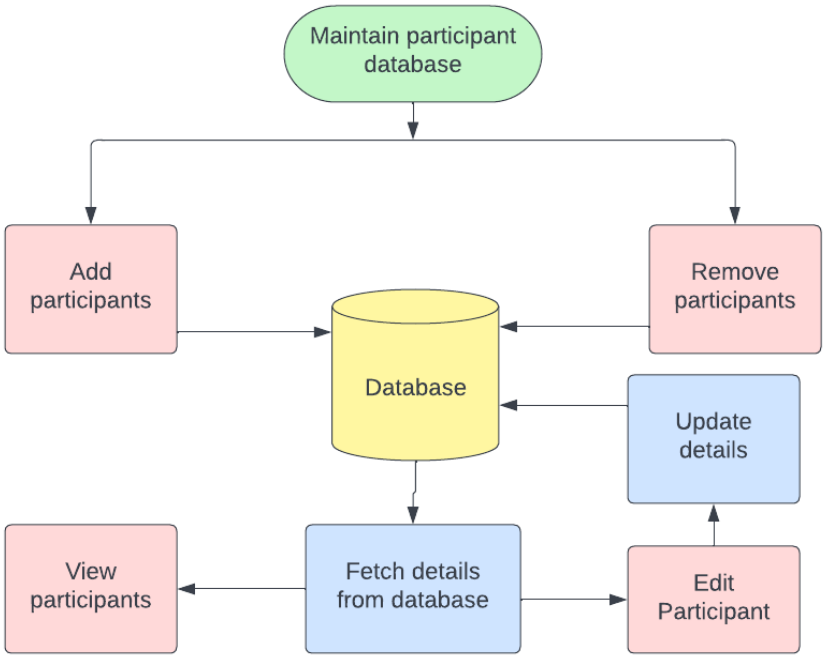
5.3. Flowchart for lottery Process:



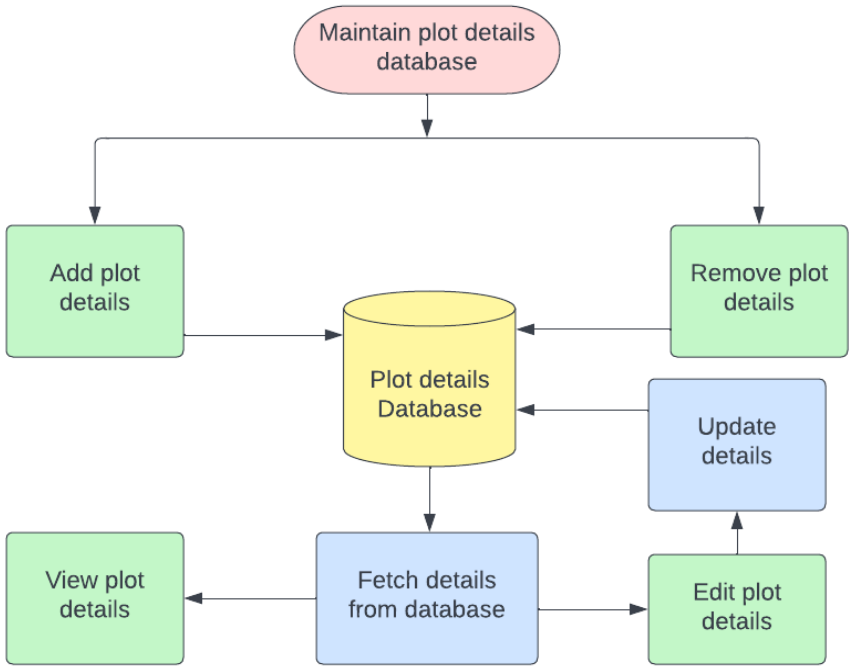
5.4. Flowchart for Main Menu:



5.5. Flowchart for Maintain participant database:



5.6. Flowchart for maintain plot database:



6. System Architecture:

6.1. Functions:

6.1.1. PARTICIPANT CORNER:

With only 300 tokens up for grabs on the website, only 300 out of 500 households can participate in the lottery process. Every family has a unique family Id and can grab only 1 token. Participants can register in the lottery by entering their family Id and can participate in the process after getting confirmation.

6.1.1.1. Grab a token:

Participant grabs a token in order to register for the lottery using his unique family Id.

6.1.1.2. Cancel token:

If the present date is within 2 days before the actual lottery date, then the participants are not allowed to cancel their token or else their token is canceled and the participant database is updated.

6.1.1.3. Lottery Confirmation:

After the ZDA authority declares the winning lottery number for a plot, the winning family can confirm the booking by paying Rs.50000/- within 5 minutes and the database is updated.

6.1.1.4. View allotment details:

This function displays the plot allotment details of the participant.

6.1.2. ZDA AUTHORITY:

ZDA Authority is responsible for maintaining the databases for the participants and plots by adding, editing, deleting and viewing the details in each database. They have access to the final lottery report and booking confirmation of the plots. ZDA Authorities have to access the admin features by entering the password.

6.1.2.1. Add Plot details:

ZDA Authority can add the information about the plots that are to be won via lottery using admin features. We can get information like plot Id, plot size, price, etc.

6.1.2.2. Edit plot details:

using this function, the features of the plots can be edited by using admin features.

6.1.2.3. Delete plot details:

ZDA Authority can delete the information about a plot.

6.1.2.4. View plot details:

ZDA Authority can view the details of all plots.

6.1.2.5. Add participant details:

The information of the participants are entered into the database by the ZDA authority. Information such as family Id, name,plot no. and price to be paid are found here. The plot no. and price to be paid column are updated automatically after plot allotment.

6.1.2.6. Edit participant details:

ZDA Authority can update the participant information through this.

6.1.2.7. Remove participant:

Details of participants can be deleted from the participant database.

6.1.2.8. View participant details:

ZDA Authority can view information about all participants.

6.2. Structures:

Participant:

This structure is used to store information about the participant of the lottery. It contains structure members like Family_id, name, participated in lottery or not, plot_no., token_no., size of plot and remaining_amount.

Plot:

This structure contains structure members like plot_no., size of plot, allot, and price of plot in order to describe the features of a plot.

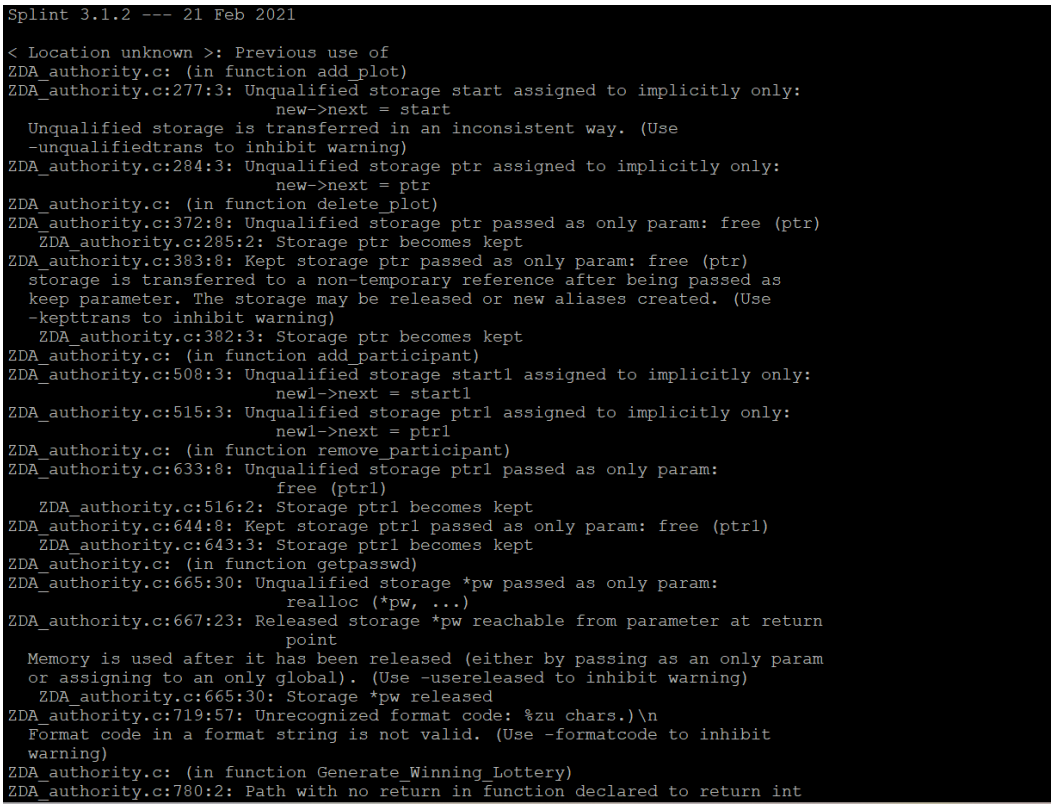
7. Tools Report:

7.1. Gcov report:

```
cg83-user15@instance-1:~/project/src$ cat main.c.gcov
        -:    0:Source:main.c
        -:    0:Graph:main.gcno
        -:    0:Data:main.gcda
        -:    0:Runs:1
        -:    1:/*****************************************************************************
        -:    2: *
        -:    3:
        -:    4:        FILENAME: main.c
        -:    5:        DESCRIPTION:This file is used to display the main menu to the user.
        -:    6:        REVISION HISTORY
        -:    7:        DATE            NAME              REASON
        -:    8:        ------------------------------------------------------------
        -:    9:    04/10/2022      Username          This is done for creationof menu
        -:   10:
        -:   11: * ****************************************************************************/
        -:   12:
        -:   13:#include <stdio.h>                 //Including required Header file
        -:   14:#include <stdlib.h>                //Includes standard libraries
        -:   15:#include <string.h>                //Includes string functions
        -:   16:#include <ctype.h>                 //Includes functions of ctype
        -:   17:#include "functions.h"             //Header file
        -:   18:#include "participant_corner.c"    //Including participant_corner.c file
        -:   19:#include "ZDA_authority.c"            //Including ZDA_authority.c file
        -:   20:#include "file_handling.c"          //Including file_handling.c file
        -:   21:#define MAXPW 32                    //Define Macro
        1:   22:int main()
        -:   23:{
        1:   24:        char pw[MAXPW] = {0};
        1:   25:        char *pass = pw;
        1:   26:        FILE *fp = stdin;
        -:   27:        ssize_t nchr;
        1:   28:        int choice=0;
        1:   29:        start=new=ptr=prev=NULL;              //initializing pointer for plot structure.
        1:   30:        start1=new1=ptr1=prev1=NULL;          //initializing pointer for participant structure
        1:   31:        plot_file_to_list();                  //brings data from plot file to linked list
        1:   32:        participant_file_to_list();           //brings data from participant file to linked list
        7:   33:        while(choice!=3)
        -:   34:        {
```

```
        7:   33:        while(choice!=3)
        -:   34:        {
        -:   35:                //Displaying Main Menu
        6:   36:                printf("\n         Main Menu");
        6:   37:                printf("\n-----------------------------\n");
        6:   38:                printf("\n1.Participant Corner\n2.ZDA Authority\n3.Exit");
        -:   39:
        6:   40:                printf("\nEnter choice: "); //Taking users choice
        6:   41:                scanf("%d",&choice);
        6:   42:                switch(choice)
        -:   43:                {
        2:   44:                        case 1: Participant_Corner();           //Function call for Participant_Corner
        2:   45:                                choice=0;
        2:   46:                                break;
        3:   47:                        case 2: nchr = getpasswd (&pass, MAXPW, 0, fp);
        3:   48:                                printf ("Enter password: ");
        3:   49:                                nchr = getpasswd (&pass, MAXPW,0 , fp);
        3:   50:                                if((strcmp(pass,"zx"))==0)
        -:   51:                                {
        3:   52:                                        system("clear");          //clears the screen
        3:   53:                                        ZDA_Authority();         //Function call for ZDA_Authority
        -:   54:                                }
        -:   55:                                else
        -:   56:                                {
    #####:   57:                                        printf("You have entered incorrect password");
        -:   58:                                }
        3:   59:                                system("clear");
        -:   60:
        3:   61:                                choice=0;
        3:   62:                                break;
        1:   63:                        case 3 : break;
    #####:   64:                        default: printf("Invalid Choice\n");
        -:   65:                }
        -:   66:        }
        1:   67:        if(start)                               //When start is not NULL
        1:   68:                list_to_plot_file();                //Linked list to plot file
        1:   69:        if(start1)                              //When start1 is not NULL
        1:   70:                list_to_participant_file();         //Linked list to participant file
        1:   71:        return EXIT_SUCCESS;
        -:   72:}
```

7.2. Splint report:



```
Splint 3.1.2 --- 21 Feb 2021

< Location unknown >: Previous use of
ZDA_authority.c: (in function add_plot)
ZDA_authority.c:277:3: Unqualified storage start assigned to implicitly only:
                         new->next = start
  Unqualified storage is transferred in an inconsistent way. (Use
  -unqualifiedtrans to inhibit warning)
ZDA_authority.c:284:3: Unqualified storage ptr assigned to implicitly only:
                         new->next = ptr
ZDA_authority.c: (in function delete_plot)
ZDA_authority.c:372:8: Unqualified storage ptr passed as only param: free (ptr)
   ZDA_authority.c:285:2: Storage ptr becomes kept
ZDA_authority.c:383:8: Kept storage ptr passed as only param: free (ptr)
  storage is transferred to a non-temporary reference after being passed as
  keep parameter. The storage may be released or new aliases created. (Use
  -kepttrans to inhibit warning)
   ZDA_authority.c:382:3: Storage ptr becomes kept
ZDA_authority.c: (in function add_participant)
ZDA_authority.c:508:3: Unqualified storage start1 assigned to implicitly only:
                         new1->next = start1
ZDA_authority.c:515:3: Unqualified storage ptr1 assigned to implicitly only:
                         new1->next = ptr1
ZDA_authority.c: (in function remove_participant)
ZDA_authority.c:633:8: Unqualified storage ptr1 passed as only param:
                         free (ptr1)
   ZDA_authority.c:516:2: Storage ptr1 becomes kept
ZDA_authority.c:644:8: Kept storage ptr1 passed as only param: free (ptr1)
   ZDA_authority.c:643:3: Storage ptr1 becomes kept
ZDA_authority.c: (in function getpasswd)
ZDA_authority.c:665:30: Unqualified storage *pw passed as only param:
                          realloc (*pw, ...)
ZDA_authority.c:667:23: Released storage *pw reachable from parameter at return
                          point
  Memory is used after it has been released (either by passing as an only param
  or assigning to an only global). (Use -usereleased to inhibit warning)
   ZDA_authority.c:665:30: Storage *pw released
ZDA_authority.c:719:57: Unrecognized format code: %zu chars.)\n
  Format code in a format string is not valid. (Use -formatcode to inhibit
  warning)
ZDA_authority.c: (in function Generate_Winning_Lottery)
ZDA_authority.c:780:2: Path with no return in function declared to return int
  There is a path through a function declared to return a value on which there
  is no return statement. This means the execution may fall through without
  returning a meaningful result to the caller. (Use -noret to inhibit warning)
ZDA_authority.c: (in function deque)
ZDA_authority.c:834:8: Unqualified storage ptr_queue passed as only param:
                         free (ptr_queue)
file_handling.c: (in function list_to_participant_file)
file_handling.c:22:3: Return value (type size_t) ignored: fwrite(ptr1, siz...
  Result returned by function call is not used. If this is intended, can cast
  result to (void) to eliminate message. (Use -retvalother to inhibit warning)
file_handling.c: (in function participant_file_to_list)
file_handling.c:44:2: Return value (type size_t) ignored: fread(new1, size...
file_handling.c:50:4: Dependent storage last1 assigned to unqualified:
                        start1 = last1 = new1
  Dependent storage is transferred to a non-dependent reference. (Use
  -dependenttrans to inhibit warning)
   file_handling.c:50:4: Storage last1 becomes dependent (through alias new1)
file_handling.c:65:3: Return value (type size_t) ignored: fread(new1, size...
file_handling.c: (in function list_to_plot_file)
file_handling.c:79:3: Return value (type size_t) ignored: fwrite(ptr, size...
file_handling.c: (in function plot_file_to_list)
file_handling.c:101:2: Return value (type size_t) ignored: fread(new, sizeo...
file_handling.c:107:4: Dependent storage last assigned to unqualified:
                         start = last = new
   file_handling.c:107:4: Storage last becomes dependent (through alias new)
file_handling.c:122:3: Return value (type size_t) ignored: fread(new, sizeo...
main.c: (in function main)
main.c:24:16: Initializer block for pw has 1 element, but declared as char
                [32]: 0
  Initializer does not define all elements of a declared array. (Use
  -initallelements to inhibit warning)
participant_corner.c:35:17: Variable exported but not used outside
                              participant_corner: accum_mutex
  A declaration is exported, but not used outside this module. Declaration can
  use static qualifier. (Use -exportlocal to inhibit warning)

Finished checking --- 23 code warnings
```

7.3. Valgrind report:

```
==80472== Syscall param write(buf) points to uninitialised byte(s)
==80472==    at 0x495A92F: __libc_write (write.c:26)
==80472==    by 0x495A92F: write (write.c:24)
==80472==    by 0x48EB664: _IO_file_write@@GLIBC_2.2.5 (fileops.c:1181)
==80472==    by 0x48EA9D5: new_do_write (fileops.c:449)
==80472==    by 0x48EC708: _IO_new_do_write (fileops.c:426)
==80472==    by 0x48EC708: _IO_do_write@@GLIBC_2.2.5 (fileops.c:423)
==80472==    by 0x48EBFEF: _IO_file_close_it@@GLIBC_2.2.5 (fileops.c:136)
==80472==    by 0x48DF375: fclose@@GLIBC_2.2.5 (iofclose.c:53)
==80472==    by 0x10E334: list_to_plot_file (in /home/cg83-user15/project/src/test)
==80472==    by 0x10E967: main (in /home/cg83-user15/project/src/test)
==80472==  Address 0x4a5789c is 12 bytes inside a block of size 4,096 alloc'd
==80472==    at 0x483877F: malloc (vg_replace_malloc.c:307)
==80472==    by 0x48DF13B: _IO_file_doallocate (filedoalloc.c:101)
==80472==    by 0x48EDA4F: _IO_doallocbuf (genops.c:347)
==80472==    by 0x48EDA4F: _IO_doallocbuf (genops.c:342)
==80472==    by 0x48ECBF7: _IO_file_overflow@@GLIBC_2.2.5 (fileops.c:745)
==80472==    by 0x48EBCDD: _IO_new_file_xsputn (fileops.c:1244)
==80472==    by 0x48EBCDD: _IO_file_xsputn@@GLIBC_2.2.5 (fileops.c:1197)
==80472==    by 0x48E05BC: fwrite (iofwrite.c:39)
==80472==    by 0x10E2F8: list_to_plot_file (in /home/cg83-user15/project/src/test)
==80472==    by 0x10E967: main (in /home/cg83-user15/project/src/test)
==80472==
==80472==
==80472== HEAP SUMMARY:
==80472==     in use at exit: 4,896 bytes in 301 blocks
==80472==   total heap usage: 316 allocs, 15 frees, 60,999 bytes allocated
==80472==
==80472== LEAK SUMMARY:
==80472==    definitely lost: 0 bytes in 0 blocks
==80472==    indirectly lost: 0 bytes in 0 blocks
==80472==      possibly lost: 0 bytes in 0 blocks
==80472==    still reachable: 4,896 bytes in 301 blocks
==80472==         suppressed: 0 bytes in 0 blocks
==80472== Rerun with --leak-check=full to see details of leaked memory
==80472==
==80472== Use --track-origins=yes to see where uninitialised values come from
==80472== For lists of detected and suppressed errors, rerun with: -s
==80472== ERROR SUMMARY: 3 errors from 1 contexts (suppressed: 0 from 0)
```

7.4. Gprof report:

```
cg83-user15@instance-1:~/project/src$ gprof -b  test gmon.out
Flat profile:

Each sample counts as 0.01 seconds.
 no time accumulated

  %   cumulative   self              self     total
 time   seconds   seconds    calls  Ts/call  Ts/call  name
  0.00      0.00     0.00      300     0.00     0.00  enque
  0.00      0.00     0.00        6     0.00     0.00  getpasswd
  0.00      0.00     0.00        5     0.00     0.00  Generate_Winning_Lottery
  0.00      0.00     0.00        3     0.00     0.00  ZDA_Authority
  0.00      0.00     0.00        2     0.00     0.00  Participant_Corner
  0.00      0.00     0.00        2     0.00     0.00  add_participant
  0.00      0.00     0.00        2     0.00     0.00  add_plot
  0.00      0.00     0.00        2     0.00     0.00  view_participant_details
  0.00      0.00     0.00        1     0.00     0.00  Lottery_Confirmation
  0.00      0.00     0.00        1     0.00     0.00  cancel_token
  0.00      0.00     0.00        1     0.00     0.00  delete_plot
  0.00      0.00     0.00        1     0.00     0.00  deque
  0.00      0.00     0.00        1     0.00     0.00  edit_participant
  0.00      0.00     0.00        1     0.00     0.00  initialize_tokens
  0.00      0.00     0.00        1     0.00     0.00  list_to_participant_file
  0.00      0.00     0.00        1     0.00     0.00  list_to_plot_file
  0.00      0.00     0.00        1     0.00     0.00  maintain_participant_db
  0.00      0.00     0.00        1     0.00     0.00  maintain_plot_db
  0.00      0.00     0.00        1     0.00     0.00  participant_file_to_list
  0.00      0.00     0.00        1     0.00     0.00  plot_file_to_list
  0.00      0.00     0.00        1     0.00     0.00  remove_participant
  0.00      0.00     0.00        1     0.00     0.00  view_allotment_details
  0.00      0.00     0.00        1     0.00     0.00  view_lottery_report
  0.00      0.00     0.00        1     0.00     0.00  view_plot_details

                     Call graph


granularity: each sample hit covers 2 byte(s) no time propagated

index % time    self  children    called     name
                0.00    0.00     300/300         initialize_tokens [14]
[1]      0.0    0.00    0.00     300         enque [1]
-----------------------------------------------
```

```
-----------------------------------------------
                0.00    0.00       6/6           main [38]
[2]      0.0    0.00    0.00       6         getpasswd [2]
-----------------------------------------------
                0.00    0.00       5/5           ZDA_Authority [4]
[3]      0.0    0.00    0.00       5         Generate_Winning_Lottery [3]
-----------------------------------------------
                0.00    0.00       3/3           main [38]
[4]      0.0    0.00    0.00       3         ZDA_Authority [4]
                0.00    0.00       5/5           Generate_Winning_Lottery [3]
                0.00    0.00       1/1           maintain_plot_db [18]
                0.00    0.00       1/1           maintain_participant_db [17]
                0.00    0.00       1/1           initialize_tokens [14]
                0.00    0.00       1/1           view_lottery_report [23]
-----------------------------------------------
                0.00    0.00       2/2           main [38]
[5]      0.0    0.00    0.00       2         Participant_Corner [5]
                0.00    0.00       1/1           cancel_token [10]
                0.00    0.00       1/1           Lottery_Confirmation [9]
                0.00    0.00       1/1           view_allotment_details [22]
-----------------------------------------------
                0.00    0.00       2/2           maintain_participant_db [17]
[6]      0.0    0.00    0.00       2         add_participant [6]
-----------------------------------------------
                0.00    0.00       2/2           maintain_plot_db [18]
[7]      0.0    0.00    0.00       2         add_plot [7]
-----------------------------------------------
                0.00    0.00       2/2           maintain_participant_db [17]
[8]      0.0    0.00    0.00       2         view_participant_details [8]
-----------------------------------------------
                0.00    0.00       1/1           Participant_Corner [5]
[9]      0.0    0.00    0.00       1         Lottery_Confirmation [9]
-----------------------------------------------
                0.00    0.00       1/1           Participant_Corner [5]
[10]     0.0    0.00    0.00       1         cancel_token [10]
-----------------------------------------------
                0.00    0.00       1/1           maintain_plot_db [18]
[11]     0.0    0.00    0.00       1         delete_plot [11]
-----------------------------------------------
                0.00    0.00       1/1           grab_token [37]
[12]     0.0    0.00    0.00       1         deque [12]
-----------------------------------------------
                0.00    0.00       1/1           maintain_participant_db [17]
[13]     0.0    0.00    0.00       1         edit_participant [13]
```

```
-----------------------------------------------
                0.00    0.00       1/1           ZDA_Authority [4]
[14]     0.0    0.00    0.00       1         initialize_tokens [14]
                0.00    0.00     300/300         enque [1]
-----------------------------------------------
                0.00    0.00       1/1           main [38]
[15]     0.0    0.00    0.00       1         list_to_participant_file [15]
-----------------------------------------------
                0.00    0.00       1/1           main [38]
[16]     0.0    0.00    0.00       1         list_to_plot_file [16]
-----------------------------------------------
                0.00    0.00       1/1           ZDA_Authority [4]
[17]     0.0    0.00    0.00       1         maintain_participant_db [17]
                0.00    0.00       2/2           add_participant [6]
                0.00    0.00       2/2           view_participant_details [8]
                0.00    0.00       1/1           edit_participant [13]
                0.00    0.00       1/1           remove_participant [21]
-----------------------------------------------
                0.00    0.00       1/1           ZDA_Authority [4]
[18]     0.0    0.00    0.00       1         maintain_plot_db [18]
                0.00    0.00       2/2           add_plot [7]
                0.00    0.00       1/1           delete_plot [11]
                0.00    0.00       1/1           view_plot_details [24]
-----------------------------------------------
                0.00    0.00       1/1           main [38]
[19]     0.0    0.00    0.00       1         participant_file_to_list [19]
-----------------------------------------------
                0.00    0.00       1/1           main [38]
[20]     0.0    0.00    0.00       1         plot_file_to_list [20]
-----------------------------------------------
                0.00    0.00       1/1           maintain_participant_db [17]
[21]     0.0    0.00    0.00       1         remove_participant [21]
-----------------------------------------------
                0.00    0.00       1/1           Participant_Corner [5]
[22]     0.0    0.00    0.00       1         view_allotment_details [22]
-----------------------------------------------
                0.00    0.00       1/1           ZDA_Authority [4]
[23]     0.0    0.00    0.00       1         view_lottery_report [23]
-----------------------------------------------
                0.00    0.00       1/1           maintain_plot_db [18]
[24]     0.0    0.00    0.00       1         view_plot_details [24]
-----------------------------------------------
```

```
Index by function name

   [3] Generate_Winning_Lottery [12] deque               [18] maintain_plot_db
   [9] Lottery_Confirmation     [13] edit_participant    [19] participant_file_to_list
   [5] Participant_Corner        [1] enque               [20] plot_file_to_list
   [4] ZDA_Authority             [2] getpasswd           [21] remove_participant
   [6] add_participant          [14] initialize_tokens   [22] view_allotment_details
   [7] add_plot                 [15] list_to_participant_file [23] view_lottery_report
  [10] cancel_token             [16] list_to_plot_file    [8] view_participant_details
  [11] delete_plot              [17] maintain_participant_db [24] view_plot_details
```

8. Testing Report:

8.1. Unit testing report:

```
cg83-user15@instance-1:~/cunit$ gcc cunit.c -lcunit
cg83-user15@instance-1:~/cunit$ ./a.out


     CUnit - A unit testing framework for C - Version 2.1-3
     http://cunit.sourceforge.net/


Suite: suite add participant data...
  Test: Test for add_participant_to_list() in sunny cases ...passed
  Test: Test for add_participant_to_list() in rainy cases ...passed
Suite: suite remove participant data...
  Test: Test for remove_participant_from_list() in sunny cases ...passed
  Test: Test for remove_participant_from_list() in rainy cases ...passed
Suite: suite add plot data...
  Test: Test for add_plot_to_list() in sunny cases ...passed
  Test: Test for add_plot_to_list() in rainy cases ...passed

Run Summary:    Type  Total    Ran Passed Failed Inactive
              suites      3      3    n/a      0        0
               tests      6      6      6      0        0
             asserts     30     30     30      0      n/a

Elapsed time =    0.000 seconds
```

8.2. Integration testing report:

Case-1

```
         Main Menu
-----------------------------

1.Participant Corner
2.ZDA Authority
3.Exit
Enter choice: 1

       Reminder: The lottery is on 26 of this month!
              Participant Corner
-------------------------------------------------
1.Grab a Token
2.Cancel Token
3.Check Lottery
4.View Allotment Details
5.Back to Main Menu
Enter your choice: 1
Enter Family Id: 1001

Your Token number is 101
```

Case 2:

```
                Participant Corner
-------------------------------------------------
1.Grab a Token
2.Cancel Token
3.Check Lottery
4.View Allotment Details
5.Back to Main Menu
Enter your choice: 2
Enter today's date (1-31): 23
Enter Family Id: 1001

Your token has been cancelled successfully
```

Case 3:

```
            Participant Corner
-------------------------------------------
1.Grab a Token
2.Cancel Token
3.Check Lottery
4.View Allotment Details
5.Back to Main Menu
Enter your choice: 4
Enter Family Id: 1001
                            ALLOTMENT DETAILS
-------------------------------------------------------------------------------------------------------------

Serial_No     Name   Family_id   Token    Plot_no    Plot_size     Remaining_Amount    Participated_in_Lottery
-------------------------------------------------------------------------------------------------------------

    1     Ayeshkanta       1001        0       0       0         0.00    NO
```

Case 4:

```
            MAINTAIN PLOT DATABASE
-----------------------------------------------

1.Add Plot
2.Edit plot
3.Remove Plot
4.View Plot details
5.Back to Last  Menu
Enter your choice: 4
                ALL PLOT DETAILS
-------------------------------------------

Serial_No   Plot_No   Size   Price  Alloted
-----------------------------------------------

    1       1       235     51000.00    NO

    2       2       342     56000.00    NO

    3      34      3456     78976.00    NO

            MAINTAIN PLOT DATABASE
-----------------------------------------------

1.Add Plot
2.Edit plot
3.Remove Plot
4.View Plot details
5.Back to Last  Menu
Enter your choice: 2
Enter Plot number: 34
Old Plot size is 3456 and Plot Price is 78976.000000
Enter new Plot size: 435
Enter new  Price: 59000
```

Case 5:

```
      MAINTAIN PARTICIPANT DATABASE
 ---------------------------------------------

 1.Add Participant
 2.Edit Participant
 3.Remove Participant
 4.View Participant details
 5.Back to Last  Menu
 Enter your choice: 4
                                ALL PARTICIPANT  DETAILS
 ------------------------------------------------------------------------------------------------------

 Serial_No       Name      Family_id   Token    Plot_no    Plot_size     Remaining_Amount    Participated_in_Lottery
 ------------------------------------------------------------------------------------------------------

    1         Ayesh         1001         0        0          0              0.00               NO


    2         sainath       1002         0        0          0              0.00               NO


    3         somebody      1003         0        0          0              0.00               NO


      MAINTAIN PARTICIPANT DATABASE
 ---------------------------------------------

 1.Add Participant
 2.Edit Participant
 3.Remove Participant
 4.View Participant details
 5.Back to Last  Menu
 Enter your choice: 2
 Enter Family Id:1001
 Old participant name is Ayesh
 Enter new Participant Name: Ayeshkanta
 Participant details are edited succesfully
```

```
     MAINTAIN PARTICIPANT DATABASE
---------------------------------------------

1.Add Participant
2.Edit Participant
3.Remove Participant
4.View Participant details
5.Back to Last  Menu
Enter your choice: 4
                               ALL PARTICIPANT  DETAILS
------------------------------------------------------------------------------------------------------

Serial_No       Name      Family_id   Token    Plot_no    Plot_size     Remaining_Amount    Participated_in_Lottery
------------------------------------------------------------------------------------------------------

   1         Ayeshkanta    1001         0        0          0              0.00               NO


   2         sainath       1002         0        0          0              0.00               NO


   3         somebody      1003         0        0          0              0.00               NO
```

Case 6:

```
          ZDA AUTHORITY
-----------------------------------
1.Maintain Plot Datatbase
2.Maintain Participant Database
3.Initialize Tokens
4.Generate Winning Lottery Number
5.View Lottery report
6.Back to Main Menu
Enter your choice:
4
Enter the plot no for which lottery is to be done: 1

The winning lottery no is: 102

The winning family id is: 1002
```

Case 7:

```
               Participant Corner
-------------------------------------------------
1.Grab a Token
2.Cancel Token
3.Check Lottery
4.View Allotment Details
5.Back to Main Menu
Enter your choice: 3
Enter Family Id: 1002

Congratulations,You have won the lottery for plot_no: 1

Please confirm your allotment by paying 50000

Please confirm your payment-Y/N: Y
```

9. Requirement Traceability Matrix:

REQUIREMENT	DESIGN MAPPING	CODE MAPPING	UT MAPPING	IT MAPPING
LS_01	a	Participant_corner		
LS_02	b	ZDA_Authority		
LS_03	c	Initialize_tokens		
LS_04	d	Generate_winning_lottery		Test case 6
LS_05	e	Get_winning_lottery		
LS_06	f	Grab_token		Test case 1
LS_07	g	Cancel_token		Test case 2
LS_08	h	Lottery_confirmation		Test case 7
LS_09	i	View_allotment_details		Test case 3
LS_10	j	add_plot	Test case 3	
LS_11	k	edit_plot		Test case 4
LS_12	l	delete_plot		

LS_13	m	view_plot_details		
LS_14	n	add_participant	Test case 1	
LS_15	o	edit_participant		Test case 5
LS_16	p	remove_participant	Test case 2	
LS_17	q	view_participant_details		
LS_18	r	view_lottery_report		
LS_19	s	list_to_plot_file		
LS_20	t	list_to_participant_file		
LS_21	u	plot_file_to_list		
LS_22	v	paticipant_file_to_list		
LS_23	w	maintain_participant_db		
LS_24	x	maintain_plot_db		