



PROJECT

On

GIVEAWAY MANAGEMENT SYSTEM

Submitted By

Name: Anurag Sharma

Registration Number: 12203261

Roll Number: 64

Section: K22BP

Program Name: Bachelor of Technology
(Computer Science and Engineering)

Under the Guidance of

Mr. Krishan Bansal

School of Computer Science & Engineering
Lovely Professional University

ACKNOWLEDGEMENT

We would like to give A special thanks of gratitude to our professor Mr. **Krishan Bansal** for giving us this golden opportunity to work on the project of “toll management system” which helped us to gain enough knowledge of real-world entity and integrate the program to the virtual modules of code in C programming language. We are even thankful for google for providing us precise information on toll plaza and management.

This project is made with an effort to have least error. Yet, if an error is found we have a heartfelt apology for the cause

Thanking you in anticipation...!!!!

Modules Description: -

The Giveaway Management System is a software module that enables users to create, manage, and track giveaways. It streamlines the entire giveaway process by automating tasks, tracking progress, and notifying winners.

Here is how the module works:

Giveaway Creation: The module allows users to create giveaways by defining the rules, criteria, and prizes. The user can set up the start and end date for the giveaway and configure other settings, such as the number of winners, frequency of giveaways, and eligibility criteria.

Participant Management: The module allows users to manage participants in the giveaway by collecting entries, validating them, and filtering out ineligible entries. The module can also integrate with social media platforms to allow participants to share the giveaway with their friends.

Winner Selection: The module uses a random algorithm to select winners based on the predefined rules and criteria. The system notifies the winners via email or other communication channels and updates the dashboard with the winner's information.

Prize Distribution: The module also helps users distribute prizes to the winners by providing them with shipping information and tracking the delivery status.

Analytics and Reporting: The module provides analytics and reporting features that allow users to track the performance of their giveaways. The

users can monitor the number of entries, participation rate, and other metrics to improve their future giveaways.

SYSTEM REQUIREMENTS

HARDWARE REQUIREMENTS:

- ✓ System: Pentium Dual Core.
- ✓ Hard Disk: 120 GB.
- ✓ Monitor: 15'' LED
- ✓ Input Devices: Keyboard, Mouse
- ✓ Ram: 1 GB

SOFTWARE REQUIREMENTS:

- ✓ Operating system: Windows 7.
- ✓ Coding Language: C
- ✓ Tool: Visual Studio

Source Code:-

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

#define MAX_PARTICIPANTS 50
#define MAX_NAME_LENGTH 50
#define MAX_EMAIL_LENGTH 50
#define MAX_PRIZE_LENGTH 100

struct Participant {
    char name[MAX_NAME_LENGTH];
    char email[MAX_EMAIL_LENGTH];
    char prize[MAX_PRIZE_LENGTH];
};

struct Participant participants[MAX_PARTICIPANTS];
int numParticipants = 0;

void addParticipant() {
    if (numParticipants == MAX_PARTICIPANTS) {
        printf("Maximum number of participants reached.\n");
        return;
    }

    struct Participant newParticipant;

    printf("Enter name: ");
    scanf("%s", newParticipant.name);

    printf("Enter email: ");
    scanf("%s", newParticipant.email);

    printf("Enter prize: ");
    scanf("%s", newParticipant.prize);

    participants[numParticipants] = newParticipant;
```

```

    numParticipants++;

    printf("Participant added successfully.\n");
}

void listParticipants() {
    printf("Participants:\n");

    for (int i = 0; i < numParticipants; i++) {
        printf("%d. Name: %s, Email: %s, Prize: %s\n", i + 1,
participants[i].name, participants[i].email, participants[i].prize);
    }
}

void chooseWinner() {
    if (numParticipants == 0) {
        printf("There are no participants to choose from.\n");
        return;
    }

    int randomIndex = rand() % numParticipants;
    printf("The winner is: %s (%s) who won and the ticket number is%s!\n",
participants[randomIndex].name, participants[randomIndex].email,
participants[randomIndex].prize);
}

int main() {
    int choice;

    do {
        printf("\nGiveaway Management System\n");
        printf("1. Add participant\n");
        printf("2. List participants\n");
        printf("3. Choose winner\n");
        printf("4. Quit\n");

        printf("Enter your choice: ");
        scanf("%d", &choice);

        switch (choice) {
            case 1:
                addParticipant();
                break;
            case 2:
                listParticipants();

```

```
        break;
    case 3:
        chooseWinner();
        break;
    case 4:
        printf("Goodbye!\n");
        break;
    default:
        printf("Invalid choice. Try again.\n");
        break;
    }
} while (choice != 4);

return 0;
}
```

Output Screenshot: -

```
PS C:\B-TECH CSE Semester 2\CSE101 COMPUTER PROGRAMMING\Mini project\mini project c>
$?) { gcc MiniProject.c -o MiniProject } ; if ($?) { .\MiniProject }

Giveaway Management System
1. Add participant
2. List participants
3. Choose winner
4. Quit
Enter your choice: █
```


Giveaway Management System

1. Add participant
2. List participants
3. Choose winner
4. Quit

Enter your choice: 2

Participants:

1. Name: anurag, Email: anurag.sharma, Prize: 122345
2. Name: anudeep, Email: anudeep.sharma, Prize: 12233456
3. Name: joydeep, Email: joy.deep, Prize: 0122567
4. Name: durgesh, Email: pandey, Prize: durgesh
5. Name: pandey, Email: 2, Prize: 234

Giveaway Management System

1. Add participant
2. List participants
3. Choose winner
4. Quit

Enter your choice: 3

The winner is: anurag (anurag.sharma) who won and the ticket number is143866!

Giveaway Management System

1. Add participant
2. List participants
3. Choose winner
4. Quit

Enter your choice: 4\
Goodbye!