KOCDM

Final Report

TITLE: IPL CRICKET TOURNAMENT SCHEDULAR

COMPUTER PROGRAMMING(CSE 101)

By

Sr. No.	Registration No	Name of Students	Roll No	Total Marks	Marks Obtained	Signature
1	12219865	NIKHIL KUMAR SINGH	A30			
2	12219999	ANKIT GUPTA	A31			
3	12220682	SWETA	A32			
4	12221604	AKASH MISHRA	A33			

Transforming Education Transforming India

Submitted To RADHIKA NAMBIAR
Lovely Professional University Jalandhar, Punjab, India.

SYUBMITED BY:	SUBMITED TO:
Name of the student: NIKHIL KUMAR SINGH Reg. No.: 12219865 Signature: NIKHIL KUMAR SINGH	Name of the faculty: RADHIKA NAMBIAR UID: 28302 Signature:

ACKNOWLEDGEMENT

I would like to express my gratitude to all those who have contributed to the successful completion of my CSE101 Computer Programming final project report.

First and foremost, I would like to thank my instructor for providing me with the knowledge and guidance needed to complete this project. Their unwavering support and constructive feedback have been invaluable throughout the process.

I am also grateful to my classmates, for their continuous encouragement, insightful discussions, and collaboration during the project.

Furthermore, I would like to extend my appreciation to the resources provided by the University, such as the computer lab facilities, online libraries, and tutorials, which have enabled me to gain a deeper understanding of the programming concepts and tools.

Lastly, I would like to express my heartfelt thanks to my family and friends for their love, encouragement, and motivation throughout my academic journey. Their unwavering support has been instrumental in helping me achieve my goals.

Thank you all once again.

Introduction:

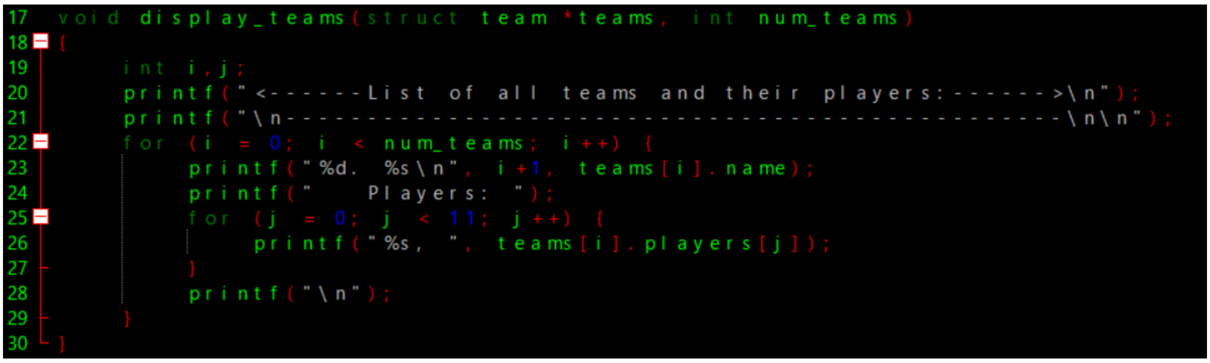
The IPL Cricket Tournament Scheduler project is a software application that allows the user to create a schedule for a cricket tournament. The user can input the number of teams participating in the tournament, the names of the teams, and the dates on which the tournament will be held. The software then generates a schedule by mapping two teams to each day. The schedule can be displayed to the user, saved in a file, and updated as necessary.

Modules used:

Module 1: Display the names of all the teams and their players:

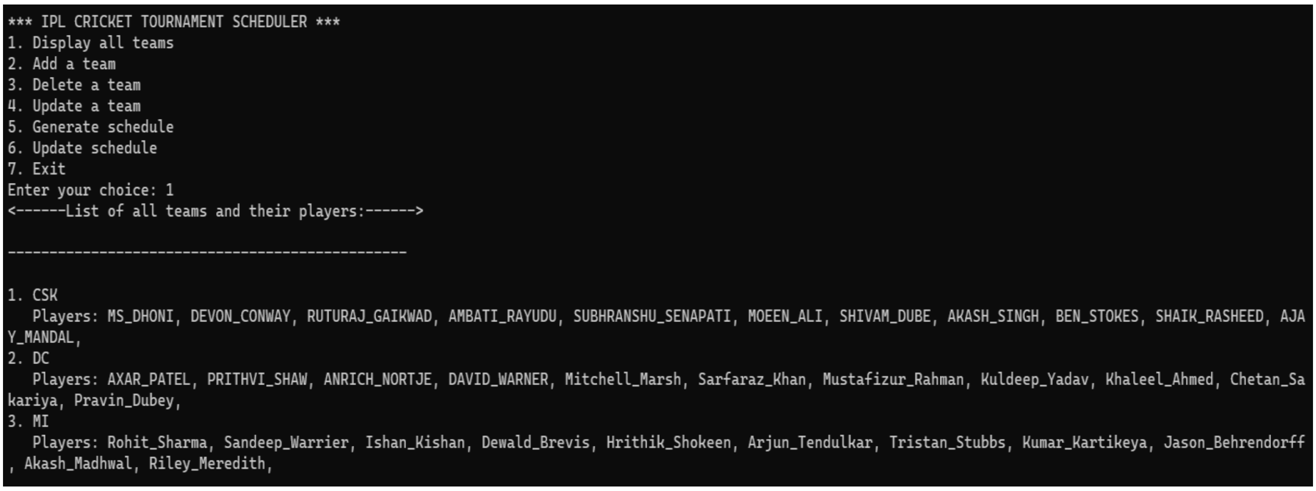
The first module of the software application displays the names of all the teams and their players. This module allows the user to view the current team and player details that are stored in the system. If the user wants to add, delete, or modify any team or player details, they can do so through the relevant modules.

CODE:



```
void display_teams(struct team *teams, int num_teams)
{
    int i,j;
    printf("<------List of all teams and their players:------>\n");
    printf("\n---------------------------------------------\n\n");
    for (i = 0; i < num_teams; i++) {
        printf("%d. %s\n", i+1, teams[i].name);
        printf("   Players: ");
        for (j = 0; j < 11; j++) {
            printf("%s, ", teams[i].players[j]);
        }
        printf("\n");
    }
}
```

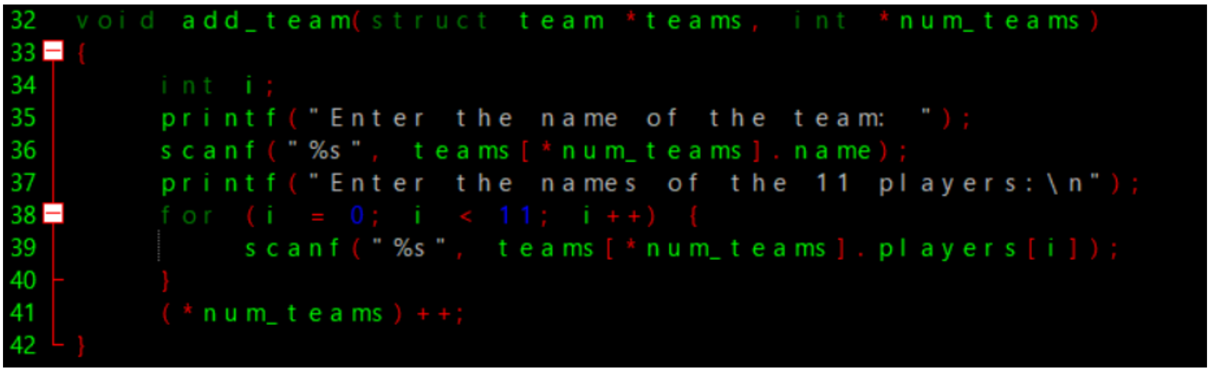
OUTPUT:



```
*** IPL CRICKET TOURNAMENT SCHEDULER ***
1. Display all teams
2. Add a team
3. Delete a team
4. Update a team
5. Generate schedule
6. Update schedule
7. Exit
Enter your choice: 1
<------List of all teams and their players:------>

------------------------------------------------


1. CSK
   Players: MS_DHONI, DEVON_CONWAY, RUTURAJ_GAIKWAD, AMBATI_RAYUDU, SUBHRANSHU_SENAPATI, MOEEN_ALI, SHIVAM_DUBE, AKASH_SINGH, BEN_STOKES, SHAIK_RASHEED, AJA
Y_MANDAL,
2. DC
   Players: AXAR_PATEL, PRITHVI_SHAW, ANRICH_NORTJE, DAVID_WARNER, Mitchell_Marsh, Sarfaraz_Khan, Mustafizur_Rahman, Kuldeep_Yadav, Khaleel_Ahmed, Chetan_Sa
kariya, Pravin_Dubey,
3. MI
   Players: Rohit_Sharma, Sandeep_Warrier, Ishan_Kishan, Dewald_Brevis, Hrithik_Shokeen, Arjun_Tendulkar, Tristan_Stubbs, Kumar_Kartikeya, Jason_Behrendorff
, Akash_Madhwal, Riley_Meredith,
```

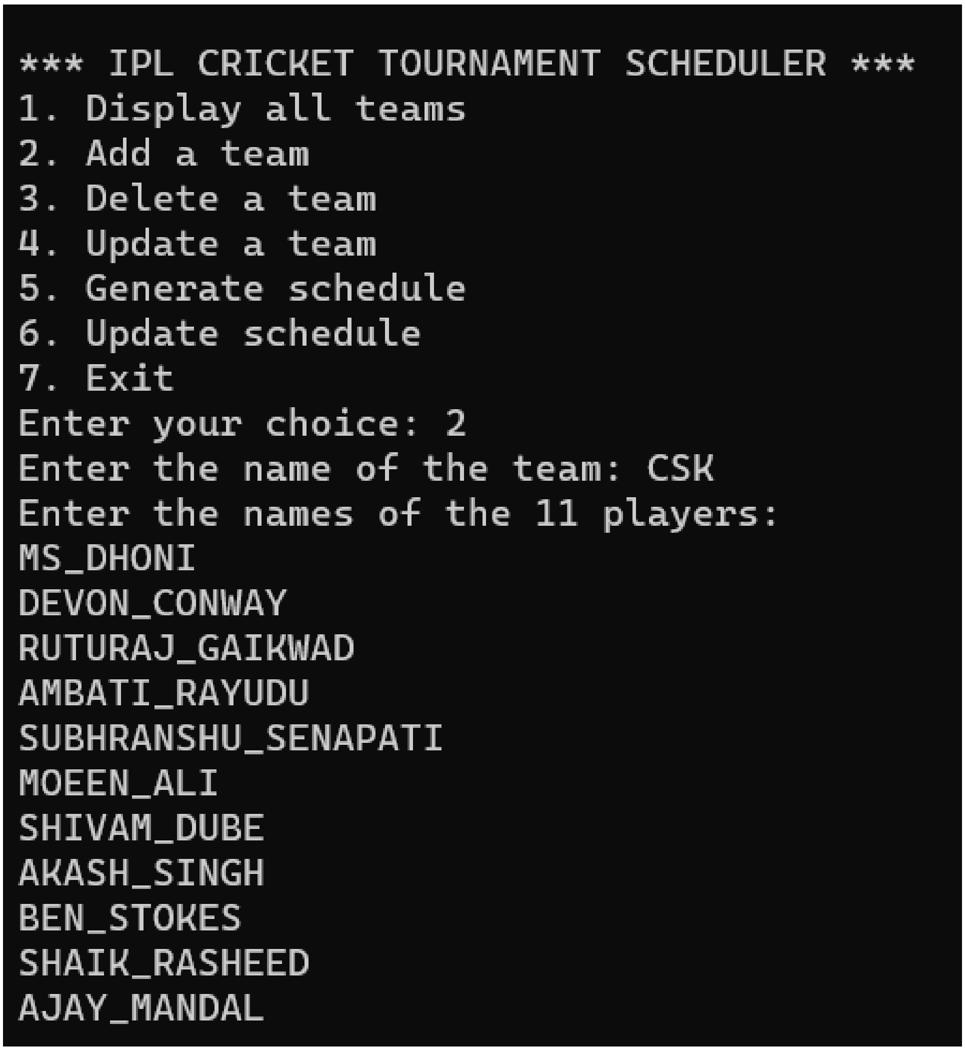
Module 2: Add a team to the tournament:

The second module of the software application allows the user to add a new team to the tournament. The user inputs the name of the new team and the names of its players (optional). The new team is added to the list of teams participating in the tournament. CODE:



```
void add_team(struct team *teams, int *num_teams)
{
    int i;
    printf("Enter the name of the team: ");
    scanf("%s", teams[*num_teams].name);
    printf("Enter the names of the 11 players:\n");
    for (i = 0; i < 11; i++) {
        scanf("%s", teams[*num_teams].players[i]);
    }
    (*num_teams)++;
}
```

OUTPUT:

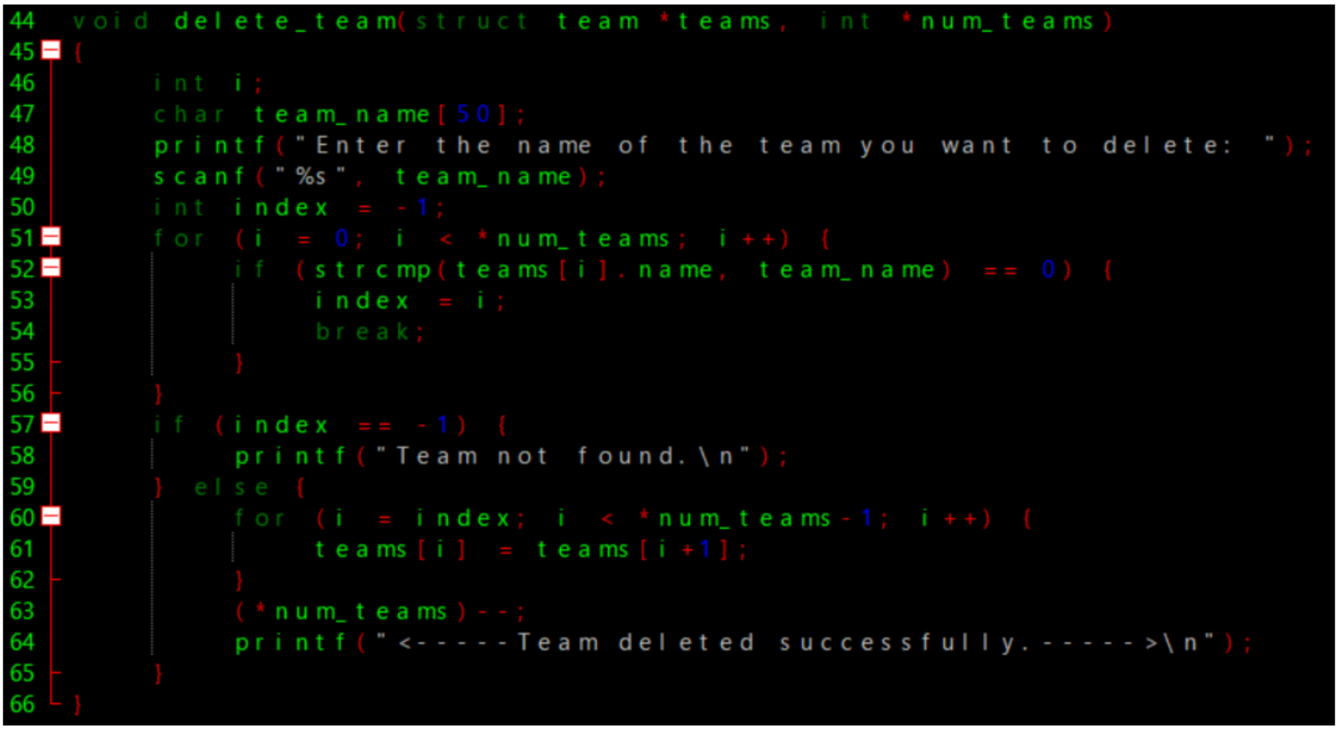


```
*** IPL CRICKET TOURNAMENT SCHEDULER ***
1. Display all teams
2. Add a team
3. Delete a team
4. Update a team
5. Generate schedule
6. Update schedule
7. Exit
Enter your choice: 2
Enter the name of the team: CSK
Enter the names of the 11 players:
MS_DHONI
DEVON_CONWAY
RUTURAJ_GAIKWAD
AMBATI_RAYUDU
SUBHRANSHU_SENAPATI
MOEEN_ALI
SHIVAM_DUBE
AKASH_SINGH
BEN_STOKES
SHAIK_RASHEED
AJAY_MANDAL
```

Module 3: Delete team details:

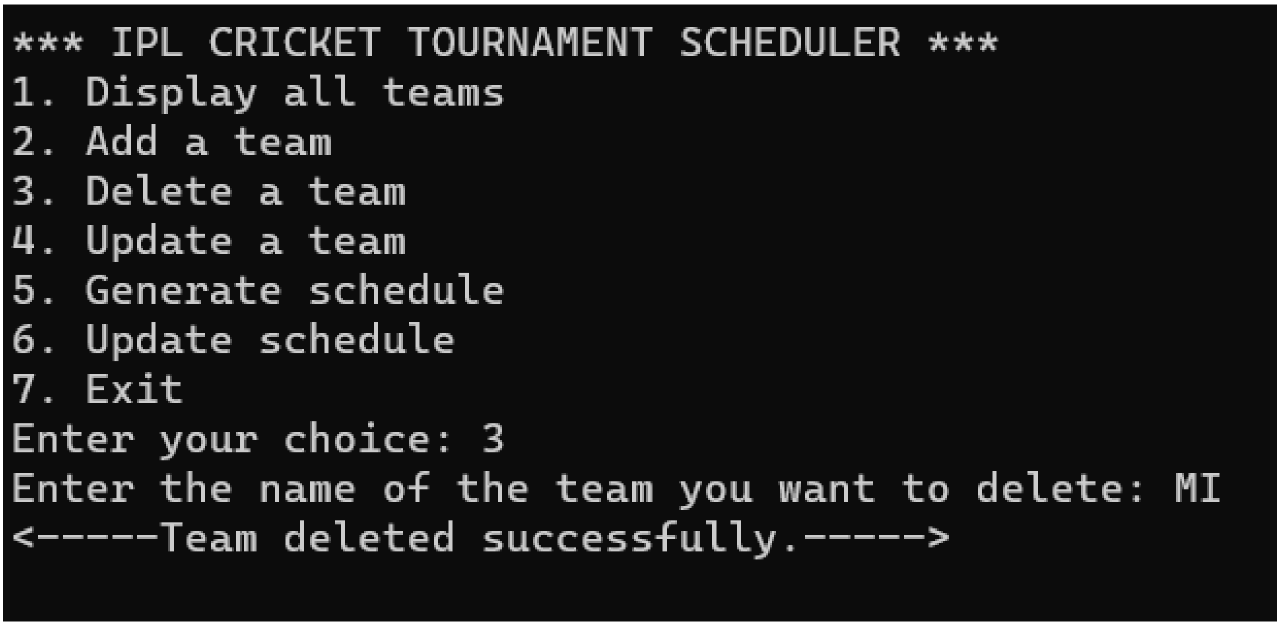
The third module of the software application allows the user to delete team details. The user selects the team they want to delete .

CODE:



```c
void delete_team(struct team *teams, int *num_teams)
{
        int i;
        char team_name[50];
        printf("Enter the name of the team you want to delete: ");
        scanf("%s", team_name);
        int index = -1;
        for (i = 0; i < *num_teams; i++) {
            if (strcmp(teams[i].name, team_name) == 0) {
                index = i;
                break;
            }
        }
        if (index == -1) {
            printf("Team not found.\n");
        } else {
            for (i = index; i < *num_teams-1; i++) {
                teams[i] = teams[i+1];
            }
            (*num_teams)--;
            printf("<-----Team deleted successfully.----->\n");
        }
}
```

OUTPUT:

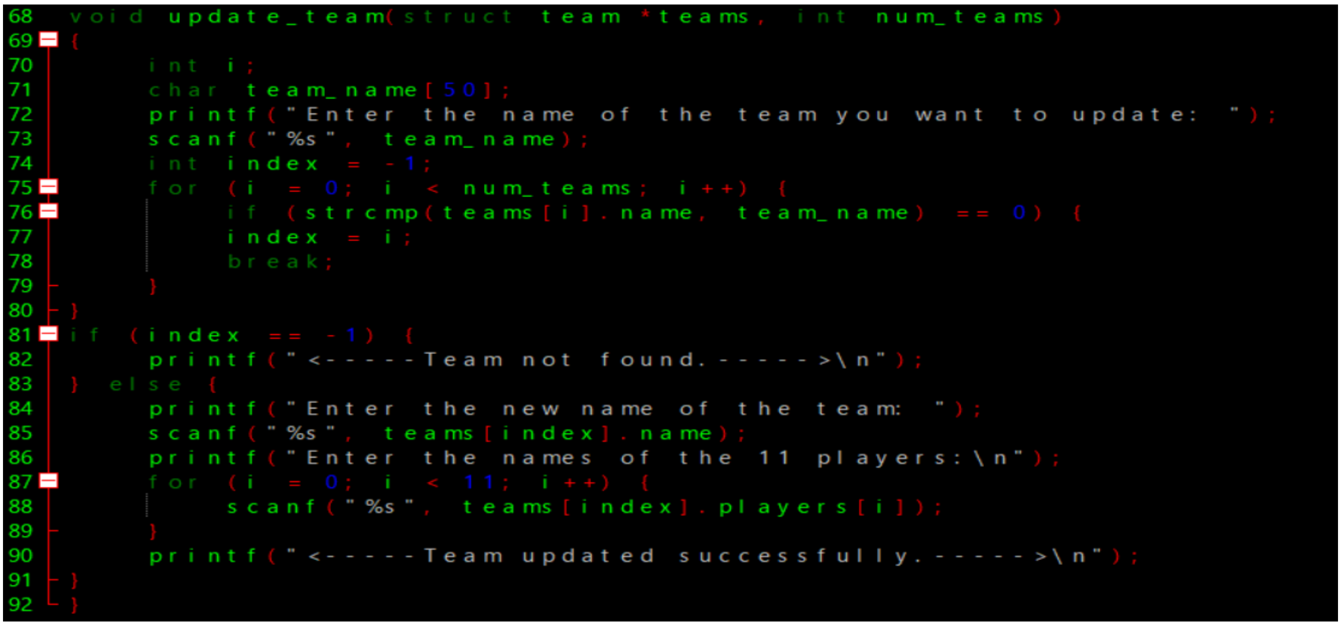


```
*** IPL CRICKET TOURNAMENT SCHEDULER ***
1. Display all teams
2. Add a team
3. Delete a team
4. Update a team
5. Generate schedule
6. Update schedule
7. Exit
Enter your choice: 3
Enter the name of the team you want to delete: MI
<-----Team deleted successfully.----->
```

Module 4: Update team details:

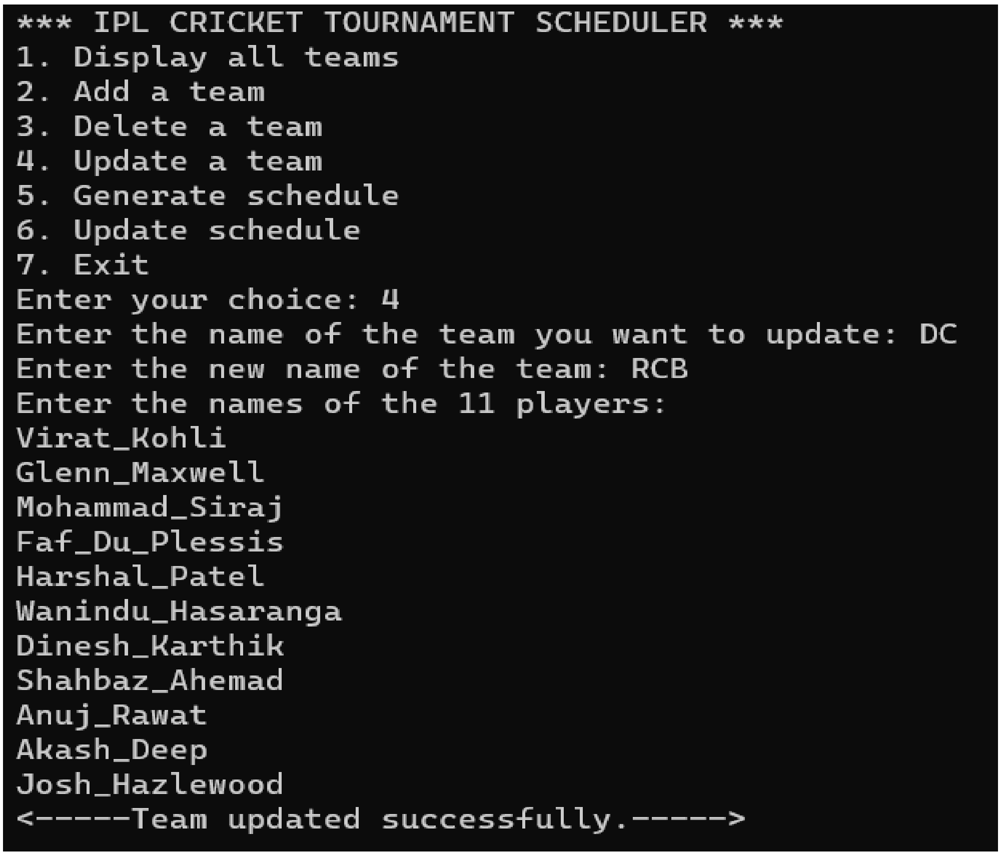
The fourth module of the software application allows the user to update team details. The user selects the team they want to update and inputs the new team and player details (if updating). The team details are then updated in the system

CODE:



```
void update_team(struct team *teams, int num_teams)
{
    int i;
    char team_name[50];
    printf("Enter the name of the team you want to update: ");
    scanf("%s", team_name);
    int index = -1;
    for (i = 0; i < num_teams; i++) {
        if (strcmp(teams[i].name, team_name) == 0) {
        index = i;
        break;
    }
}
if (index == -1) {
    printf("<-----Team not found.----->\n");
} else {
    printf("Enter the new name of the team: ");
    scanf("%s", teams[index].name);
    printf("Enter the names of the 11 players:\n");
    for (i = 0; i < 11; i++) {
        scanf("%s", teams[index].players[i]);
    }
    printf("<-----Team updated successfully.----->\n");
}
}
```

OUTPUT:

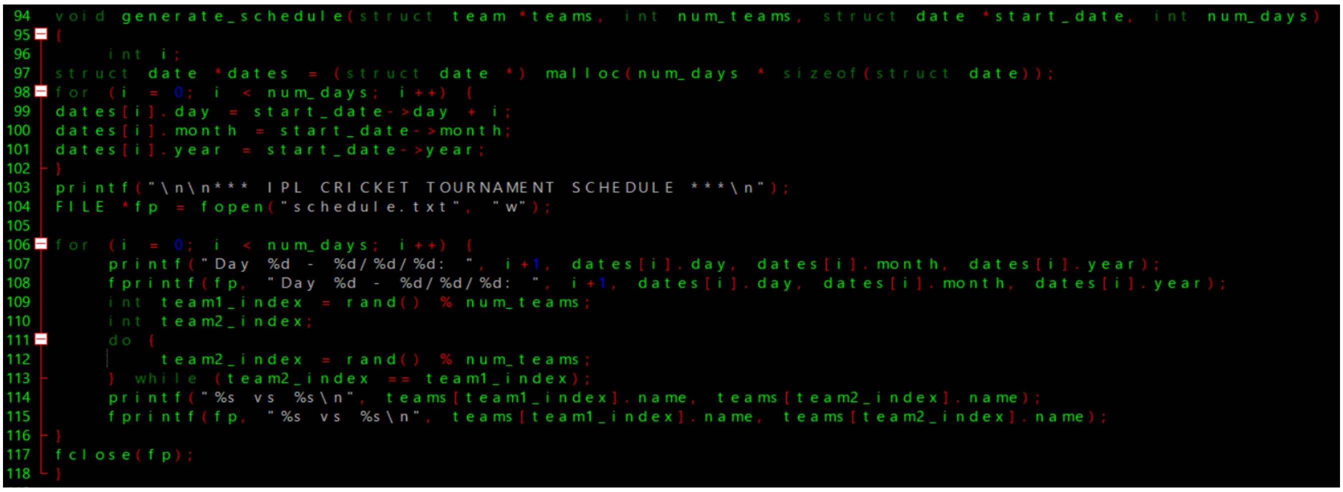


```
*** IPL CRICKET TOURNAMENT SCHEDULER ***
1. Display all teams
2. Add a team
3. Delete a team
4. Update a team
5. Generate schedule
6. Update schedule
7. Exit
Enter your choice: 4
Enter the name of the team you want to update: DC
Enter the new name of the team: RCB
Enter the names of the 11 players:
Virat_Kohli
Glenn_Maxwell
Mohammad_Siraj
Faf_Du_Plessis
Harshal_Patel
Wanindu_Hasaranga
Dinesh_Karthik
Shahbaz_Ahemad
Anuj_Rawat
Akash_Deep
Josh_Hazlewood
<-----Team updated successfully.----->
```

Module 5: Generate the schedule:

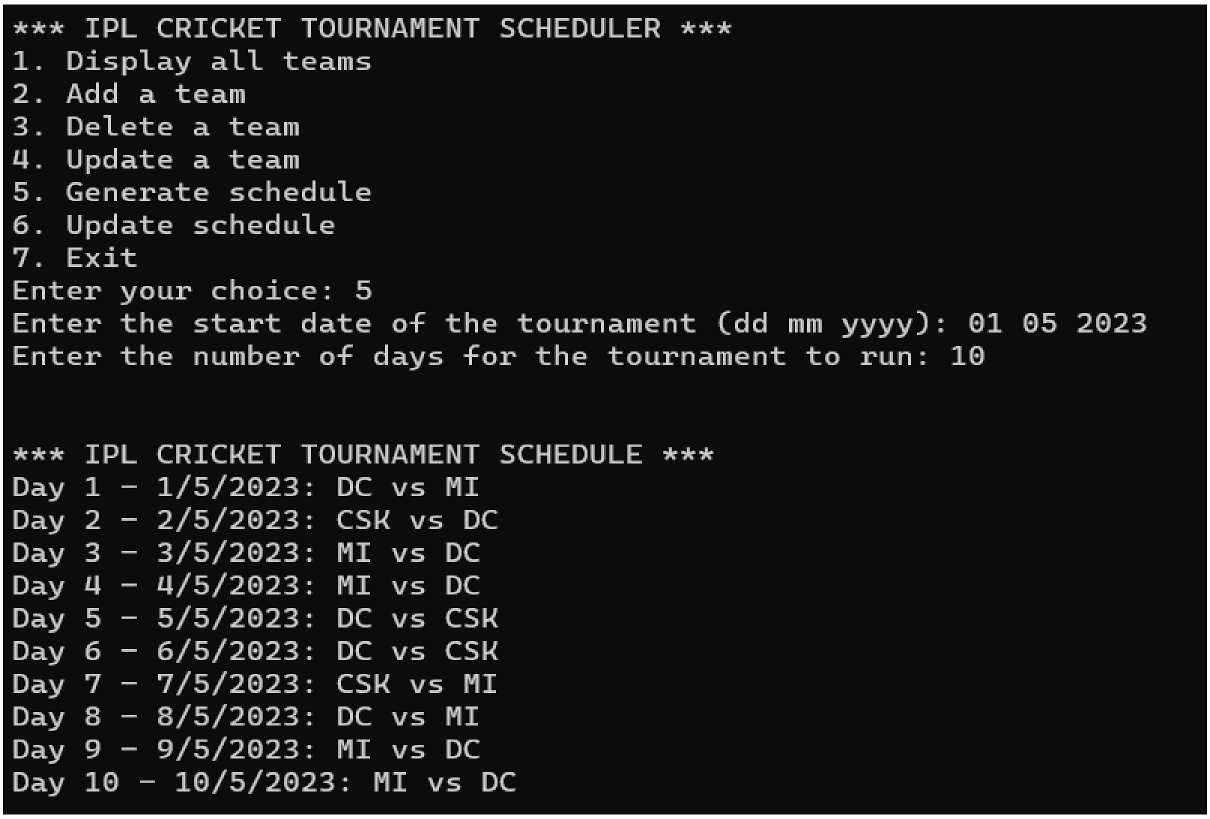
The fifth module of the software application generates the schedule for the cricket tournament. The user inputs the number of teams participating in the tournament, the names of the teams, and the dates on which the tournament will be held. The software then generates a schedule by mapping two teams to each day.

CODE:



```c
void generate_schedule(struct team *teams, int num_teams, struct date *start_date, int num_days)
{
        int i;
struct date *dates = (struct date *) malloc(num_days * sizeof(struct date));
for (i = 0; i < num_days; i++) {
dates[i].day = start_date->day + i;
dates[i].month = start_date->month;
dates[i].year = start_date->year;
}
printf("\n\n*** IPL CRICKET TOURNAMENT SCHEDULE ***\n");
FILE *fp = fopen("schedule.txt", "w");

for (i = 0; i < num_days; i++) {
        printf("Day %d - %d/%d/%d: ", i+1, dates[i].day, dates[i].month, dates[i].year);
        fprintf(fp, "Day %d - %d/%d/%d: ", i+1, dates[i].day, dates[i].month, dates[i].year);
        int team1_index = rand() % num_teams;
        int team2_index;
        do {
                team2_index = rand() % num_teams;
        } while (team2_index == team1_index);
        printf("%s vs %s\n", teams[team1_index].name, teams[team2_index].name);
        fprintf(fp, "%s vs %s\n", teams[team1_index].name, teams[team2_index].name);
}
fclose(fp);
}
```

OUTPUT:

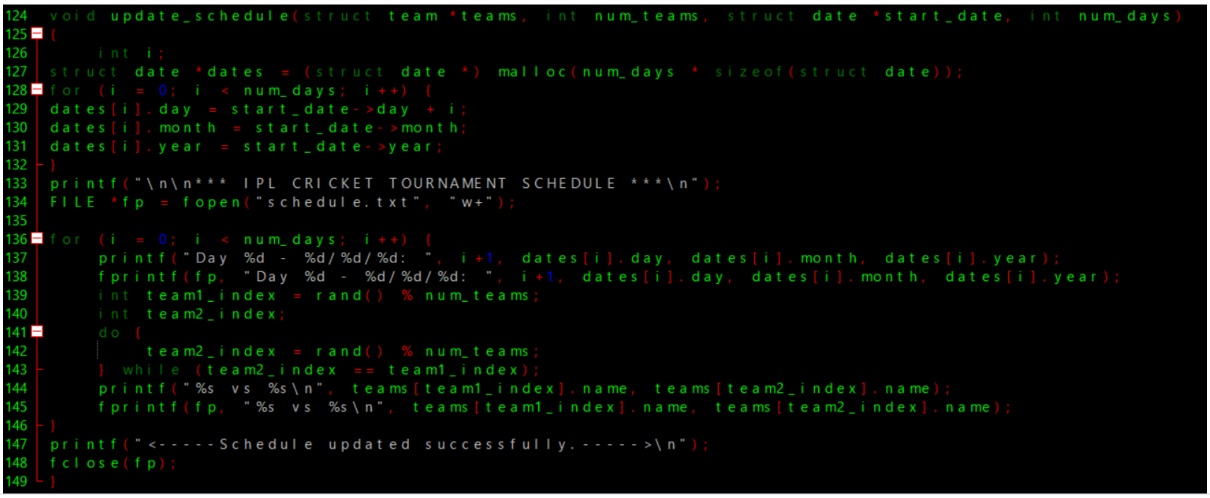


```
*** IPL CRICKET TOURNAMENT SCHEDULER ***
1. Display all teams
2. Add a team
3. Delete a team
4. Update a team
5. Generate schedule
6. Update schedule
7. Exit
Enter your choice: 5
Enter the start date of the tournament (dd mm yyyy): 01 05 2023
Enter the number of days for the tournament to run: 10


*** IPL CRICKET TOURNAMENT SCHEDULE ***
Day 1 - 1/5/2023: DC vs MI
Day 2 - 2/5/2023: CSK vs DC
Day 3 - 3/5/2023: MI vs DC
Day 4 - 4/5/2023: MI vs DC
Day 5 - 5/5/2023: DC vs CSK
Day 6 - 6/5/2023: DC vs CSK
Day 7 - 7/5/2023: CSK vs MI
Day 8 - 8/5/2023: DC vs MI
Day 9 - 9/5/2023: MI vs DC
Day 10 - 10/5/2023: MI vs DC
```

Module 6: Update the schedule:

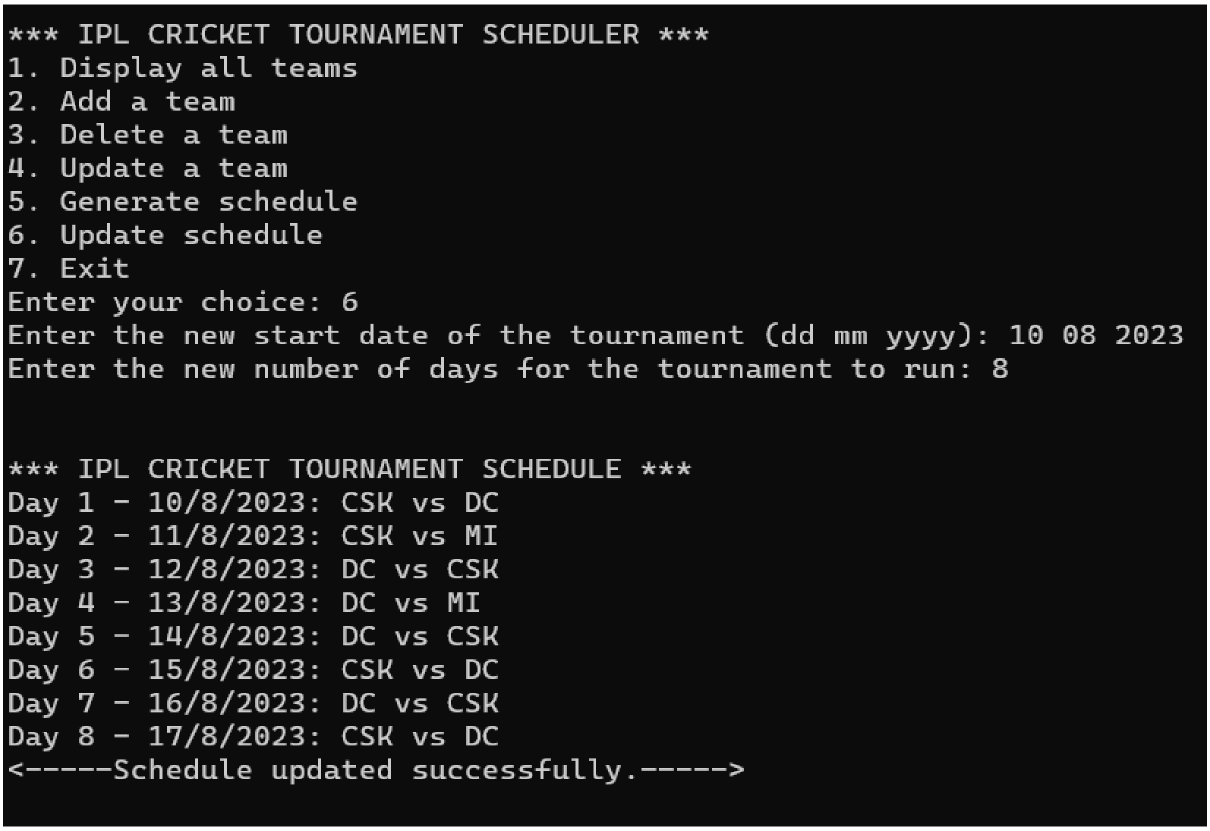
The sixth module of the software application allows the user to update the schedule as necessary. If there are any changes in the number of teams, team names, or dates, the user can update the schedule through this module. The updated schedule is then displayed to the user and saved in a file.

CODE:



```c
void update_schedule(struct team *teams, int num_teams, struct date *start_date, int num_days)
{
    int i;
struct date *dates = (struct date *) malloc(num_days * sizeof(struct date));
for (i = 0; i < num_days; i++) {
dates[i].day = start_date->day + i;
dates[i].month = start_date->month;
dates[i].year = start_date->year;
}
printf("\n\n*** IPL CRICKET TOURNAMENT SCHEDULE ***\n");
FILE *fp = fopen("schedule.txt", "w+");

for (i = 0; i < num_days; i++) {
    printf("Day %d - %d/%d/%d: ", i+1, dates[i].day, dates[i].month, dates[i].year);
    fprintf(fp, "Day %d - %d/%d/%d: ", i+1, dates[i].day, dates[i].month, dates[i].year);
    int team1_index = rand() % num_teams;
    int team2_index;
    do {
        team2_index = rand() % num_teams;
    } while (team2_index == team1_index);
    printf("%s vs %s\n", teams[team1_index].name, teams[team2_index].name);
    fprintf(fp, "%s vs %s\n", teams[team1_index].name, teams[team2_index].name);
}
printf("<-----Schedule updated successfully.----->\n");
fclose(fp);
}
```

Output:



```
*** IPL CRICKET TOURNAMENT SCHEDULER ***
1. Display all teams
2. Add a team
3. Delete a team
4. Update a team
5. Generate schedule
6. Update schedule
7. Exit
Enter your choice: 6
Enter the new start date of the tournament (dd mm yyyy): 10 08 2023
Enter the new number of days for the tournament to run: 8


*** IPL CRICKET TOURNAMENT SCHEDULE ***
Day 1 - 10/8/2023: CSK vs DC
Day 2 - 11/8/2023: CSK vs MI
Day 3 - 12/8/2023: DC vs CSK
Day 4 - 13/8/2023: DC vs MI
Day 5 - 14/8/2023: DC vs CSK
Day 6 - 15/8/2023: CSK vs DC
Day 7 - 16/8/2023: DC vs CSK
Day 8 - 17/8/2023: CSK vs DC
<-----Schedule updated successfully.----->
```

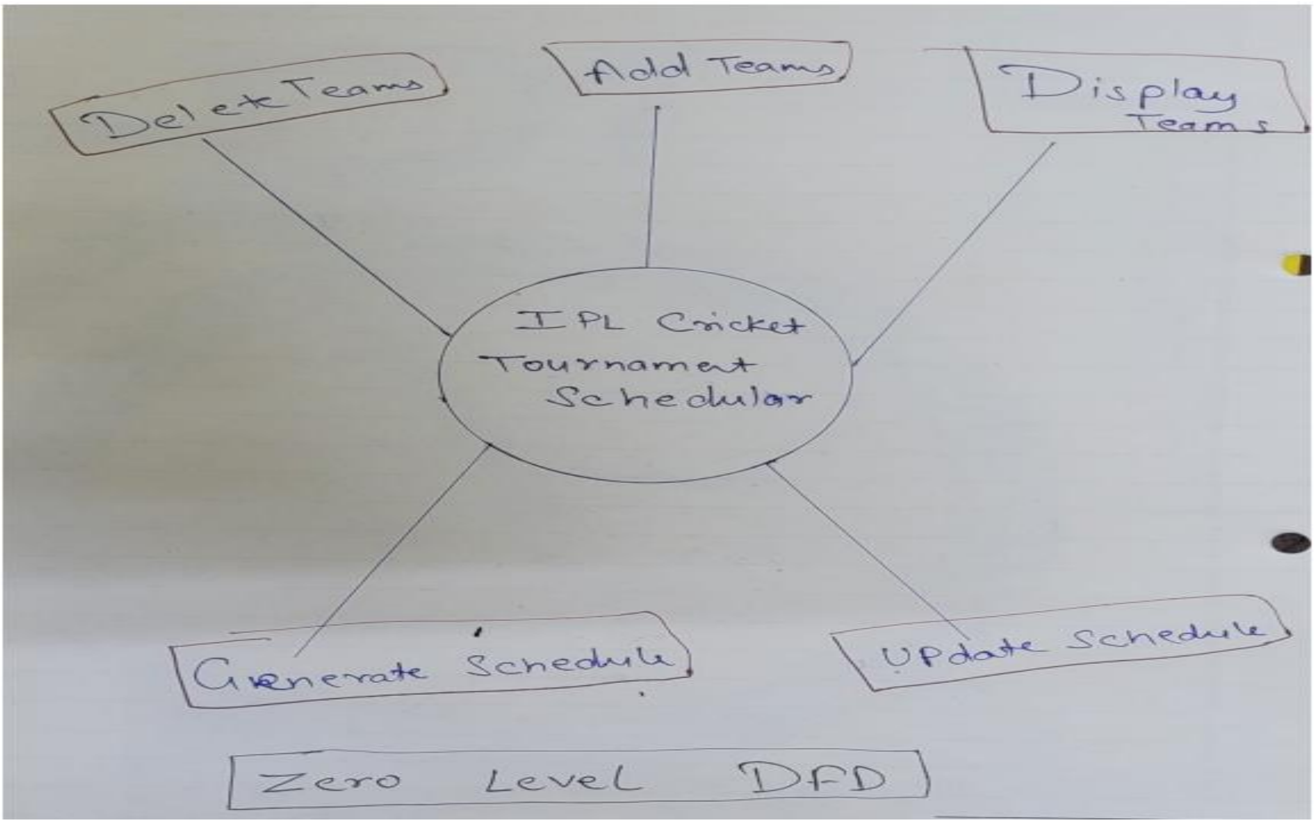
<COMPLETE CODE>

```
#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <time.h>


struct team {

   char name[50];

   char players[11][50];

};


struct date {

   int day;

   int month;

   int year;

};


void display_teams(struct team *teams, int num_teams)

{

           int i,j;

    printf("<------List of all teams and their players:------>\n");

    printf("\n------------------------------------------------\n\n");

    for (i = 0; i < num_teams; i++) {

        printf("%d. %s\n", i+1, teams[i].name);

        printf("    Players: ");

        for (j = 0; j < 11; j++) {

            printf("%s, ", teams[i].players[j]);
        }
        printf("\n");
    }
}

void add_team(struct team *teams, int *num_teams)
{
          int i;
    printf("Enter the name of the team: ");
    scanf("%s", teams[*num_teams].name);
    printf("Enter the names of the 11 players:\n");
    for (i = 0; i < 11; i++) {
        scanf("%s", teams[*num_teams].players[i]);
    }
    (*num_teams)++;
}

void delete_team(struct team *teams, int *num_teams)
{
          int i;
    char team_name[50];
    printf("Enter the name of the team you want to delete: ");
    scanf("%s", team_name);
    int index = -1;
    for (i = 0; i < *num_teams; i++) {
        if (strcmp(teams[i].name, team_name) == 0) {
            index = i;
            break;
        }
    }
    if (index == -1) {
        printf("Team not found.\n");
    } else {
        for (i = index; i < *num_teams-1; i++) {
            teams[i] = teams[i+1];
        }
        (*num_teams)--;
        printf("<-----Team deleted successfully.----->\n");
    }
}

void update_team(struct team *teams, int num_teams)
{
           int i;
    char team_name[50];
    printf("Enter the name of the team you want to update: ");
    scanf("%s", team_name);
    int index = -1;
    for (i = 0; i < num_teams; i++) {
        if (strcmp(teams[i].name, team_name) == 0) {
        index = i;
        break;
    }
}
if (index == -1) {
    printf("<-----Team not found.----->\n");
```

```
} else {
    printf("Enter the new name of the team: ");
    scanf("%s", teams[index].name);
    printf("Enter the names of the 11 players:\n");
    for (i = 0; i < 11; i++) {
        scanf("%s", teams[index].players[i]);
    }
    printf("<-----Team updated successfully.----->\n");
}
}

void generate_schedule(struct team *teams, int num_teams, struct date *start_date, int num_days)
{
           int i;
struct date *dates = (struct date *) malloc(num_days * sizeof(struct date));
for (i = 0; i < num_days; i++) {
dates[i].day = start_date->day + i;
dates[i].month = start_date->month;
dates[i].year = start_date->year;
}
printf("\n\n*** IPL CRICKET TOURNAMENT SCHEDULE ***\n");
FILE *fp = fopen("schedule.txt", "w");

for (i = 0; i < num_days; i++) {
    printf("Day %d - %d/%d/%d: ", i+1, dates[i].day, dates[i].month, dates[i].year);
    fprintf(fp, "Day %d - %d/%d/%d: ", i+1, dates[i].day, dates[i].month, dates[i].year);
    int team1_index = rand() % num_teams;
    int team2_index;
    do {
        team2_index = rand() % num_teams;
    } while (team2_index == team1_index);
    printf("%s vs %s\n", teams[team1_index].name, teams[team2_index].name);
    fprintf(fp, "%s vs %s\n", teams[team1_index].name, teams[team2_index].name);
}
fclose(fp);
}

//void update_schedule() {
//// TODO: Implement schedule update logic
//printf("Schedule updated successfully.\n");
//}
void update_schedule(struct team *teams, int num_teams, struct date *start_date, int
num_days)
{
        int i;
struct date *dates = (struct date *) malloc(num_days * sizeof(struct date));
for (i = 0; i < num_days; i++) {
dates[i].day = start_date->day + i;
dates[i].month = start_date->month;
dates[i].year = start_date->year;
}
printf("\n\n*** IPL CRICKET TOURNAMENT SCHEDULE ***\n");
FILE *fp = fopen("schedule.txt", "w+");

for (i = 0; i < num_days; i++) {
    printf("Day %d - %d/%d/%d: ", i+1, dates[i].day, dates[i].month, dates[i].year);
    fprintf(fp, "Day %d - %d/%d/%d: ", i+1, dates[i].day, dates[i].month, dates[i].year);
    int team1_index = rand() % num_teams;
    int team2_index;
    do {
        team2_index = rand() % num_teams;
    } while (team2_index == team1_index);
    printf("%s vs %s\n", teams[team1_index].name, teams[team2_index].name);
    fprintf(fp, "%s vs %s\n", teams[team1_index].name, teams[team2_index].name);
}
fclose(fp);
}

int main() {
struct team teams[20];
int num_teams = 0;
struct date start_date;
int num_days;
srand(time(0)); // Seed random number generator with current time

int choice;
do {
    printf("\n\n*** IPL CRICKET TOURNAMENT SCHEDULER ***\n");
    printf("1. Display all teams\n");
    printf("2. Add a team\n");
    printf("3. Delete a team\n");
    printf("4. Update a team\n");
    printf("5. Generate schedule\n");
    printf("6. Update schedule\n");
    printf("7. Exit\n");
    printf("Enter your choice: ");
    scanf("%d", &choice);

    switch (choice) {
        case 1:
            display_teams(teams, num_teams);
            break;
        case 2:
            add_team(teams, &num_teams);
            break;
        case 3:
            delete_team(teams, &num_teams);
            break;
        case 4:
            update_team(teams, num_teams);
            break;
        case 5:
            printf("Enter the start date of the tournament (dd mm yyyy): ");
            scanf("%d %d %d", &start_date.day, &start_date.month, &start_date.year);
            printf("Enter the number of days for the tournament to run: ");
            scanf("%d", &num_days);
            generate_schedule(teams, num_teams, &start_date, num_days);
            break;
        case 6:
           printf("Enter the new start date of the tournament (dd mm yyyy): ");
            scanf("%d %d %d", &start_date.day, &start_date.month, &start_date.year);
```

```
            printf("Enter the new number of days for the tournament to run: ");
            scanf("%d", &num_days);
            update_schedule(teams, num_teams, &start_date, num_days);
            printf("<-----Schedule updated successfully.----->\n");
            break;
        case 7:
            printf("Thank you for using IPL Cricket Tournament Scheduler.\n");
            printf("<---------GOOD BYE!-------->");
            break;
        default:
            printf("Invalid choice. Please try again.\n");
            break;
    }
} while (choice != 7);

return 0;}
```

0 LEVEL DFD(CONTEXT DIAGRAM):



Conclusion:

The IPL Cricket Tournament Scheduler project is a useful software application that simplifies the process of scheduling a cricket tournament. The application allows the user to add, delete, and update team and player details, generate a schedule, and update the schedule as necessary. The software is user-friendly and can be used by anyone with basic computer skills. Overall, the IPL Cricket Tournament Scheduler project is a valuable tool for anyone involved in organizing a cricket tournament.

<----------*------------>

<-----------*------------>