CS313 - Group Project Phase 2

Project Title : Event Management System

Project Description :

A large event management system that can be used for university events like inter-IIT sports, Mood Indigo like cultural events, TechFest type of competitive events, ...

Group - G10

Shriram Ghadge	(180010015)
Sai Yashwanth	(180010010)
Manjeet kapil	(180010021)

Brief project overview :

For event management system we had selected our college tech-fest PARSEC, with different events in it like software hackathon, hardware hackathon, Algo-strike, VeniVidiVici (CTF), synergia, scitech quiz, some fun games, food stalls, workshops,..

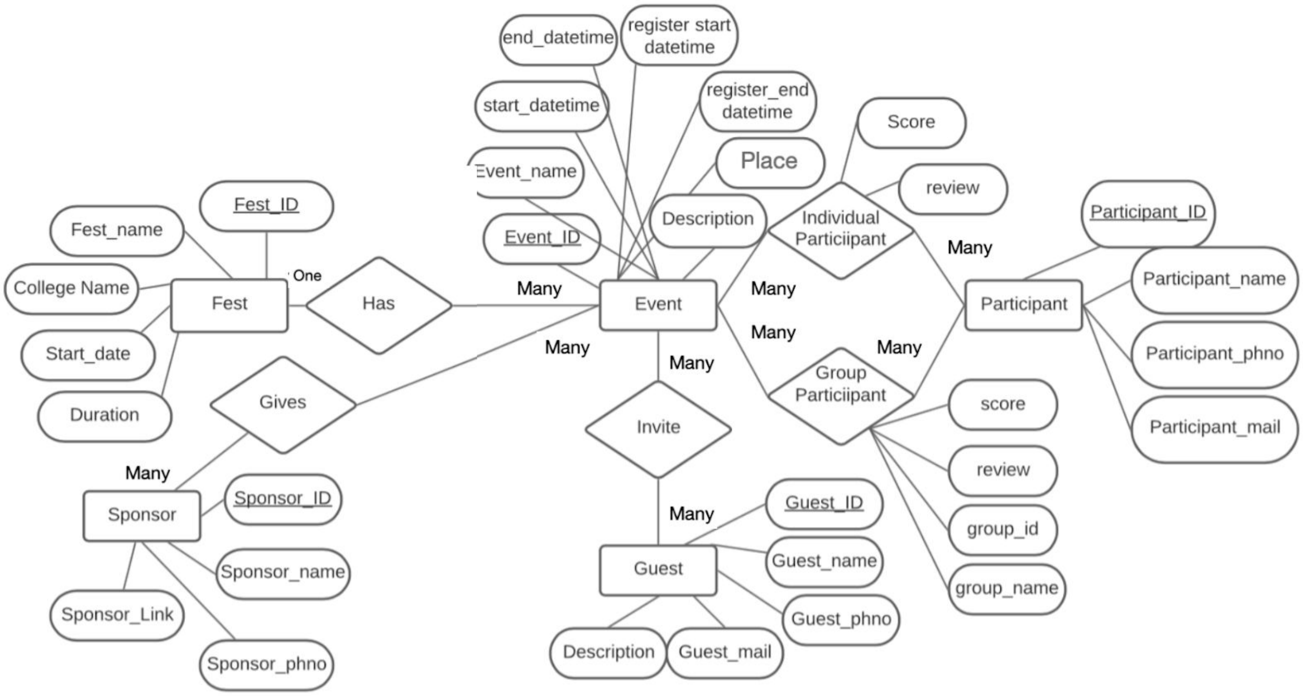
We are creating flutter(front-end) mobile- application for above with support of dart (for back-end design) and online postgreSQL(for database management)

There will be mainly three types of users or coordinators,

1. Admin : Manages all event activities
2. Participants : participates in some of events
3. Guests/Invigilators : invigilates or guides during event

Data model and Database design :

ER - Diagram :



Tables :

```
create table fest
        (       fest_id                 varchar(8),
                fest_name               varchar(20),
                college_name            varchar(20),
                start_date              DATE,
                duration                varchar(8),
                primary key (fest_id)
        );

create table evento
        (       event_id                                varchar(8),
                event_name                              varchar(20),
                start_date_time                         TIMESTAMP,
                end_date_time                           TIMESTAMP,
                register_start_date_time                TIMESTAMP,
                register_end_start_time                 TIMESTAMP,
                place                                   varchar(20),
                description                             varchar(300),
                primary key (event_id)
        );

create table has
        (       event_id                varchar(8),
                fest_id                 varchar(8),
                primary key (fest_id,event_id),
                foreign key (event_id) references evento on delete cascade,
                foreign key (fest_id) references fest on delete cascade
        );
```

```
create table participant
        (       participant_id          varchar(8),
                participant_name        varchar(20),
                participant_cno         numeric(10),
                participant_email       varchar(50),
                primary key (participant_id)
        );

create table individual_participant
        (       participant_id                  varchar(8),
                event_id                        varchar(8),
                score                           numeric(10,3),
                review                          varchar(30),
                primary key (participant_id,event_id),
                foreign key (participant_id) references participant on delete cascade,
                foreign key (event_id) references evento on delete cascade
        );

create table group_participant
        (       participant_id                  varchar(8),
                event_id                        varchar(8),
                score                           numeric(10,3),
                group_id                        varchar(8),
                group_name                      varchar(20),
                review                          varchar(30),
                primary key (participant_id,event_id,group_id),
                foreign key (participant_id) references participant on delete cascade,
                foreign key (event_id) references evento on delete cascade
        );
```

```
create table admino
        (       admin_id                varchar(8),
                admin_name              varchar(20),
                admin_cno               numeric(10),
                admin_email             varchar(50),
                pass_word               varchar(50),
                primary key (admin_id)
        );



create table sponsor
        (       sponsor_id              varchar(8),
                sponsor_name           varchar(20),
                sponsor_phno            numeric(10),
                sponsor_link           varchar(100),
                primary key (sponsor_id)
        );



create table gives
        (       sponsor_id              varchar(8),
                event_id                varchar(8),
                amount                  numeric(10,2),
                primary key (sponsor_id,event_id),
                foreign key (sponsor_id) references sponsor on delete cascade,
                foreign key (event_id) references evento on delete cascade
        );
```

```
create table guest
        (       guest_id                varchar(8),
                guest_name              varchar(20),
                description             varchar(300),
                guest_cno               numeric(10),
                guest_email             varchar(50),
                primary key (guest_id)
        );


create table invite
        (       guest_id                varchar(8),
                event_id                varchar(8),
                primary key (guest_id,event_id),
                foreign key (guest_id) references guest on delete cascade,
                foreign key (event_id) references evento on delete cascade
        );
```

Views :

=> Normal Users can only access (only reading but not modification of data) the database with some limitations, modification is restricted to their personal data. Normal users have the below mentioned views.

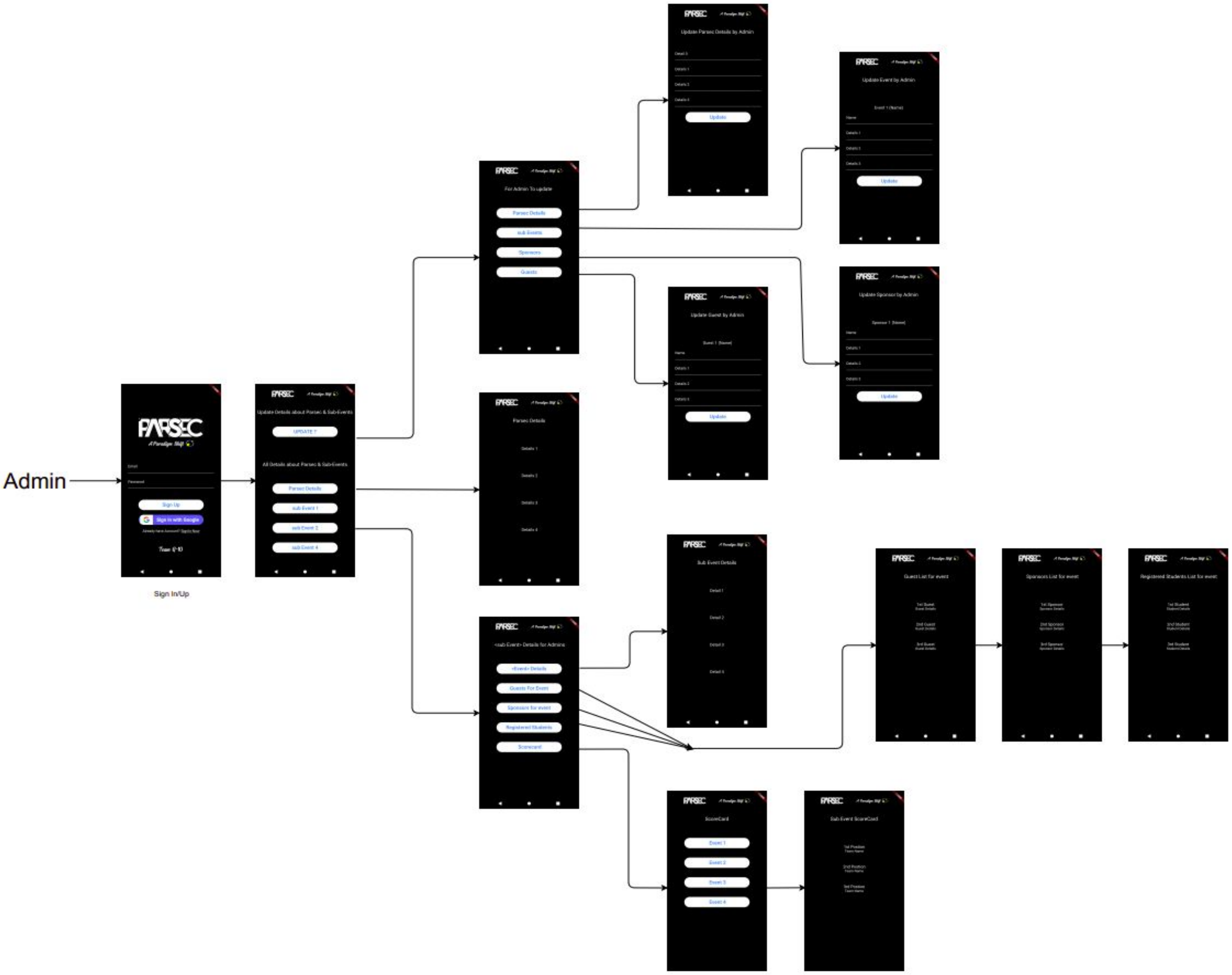
- Fest_user_view (only access fest details),
- Event_user_view (only access event details),
- Participant_user_view (can edit details)
- Individual_partp_user_view (only the respective users score attribute can be accessed by the user)
- Group_part_user_view (only the respective users score attribute can be accessed by the user)
- SponsorsGives_user_view (only access details of sponsors),
- GuestInvite_user_view (only access public details of Guest),

-> The invigilators have access same as normal users but they can have access for modifications for their personal data & for some tables like scorecard & all. The given below are the special access for the invigilators.

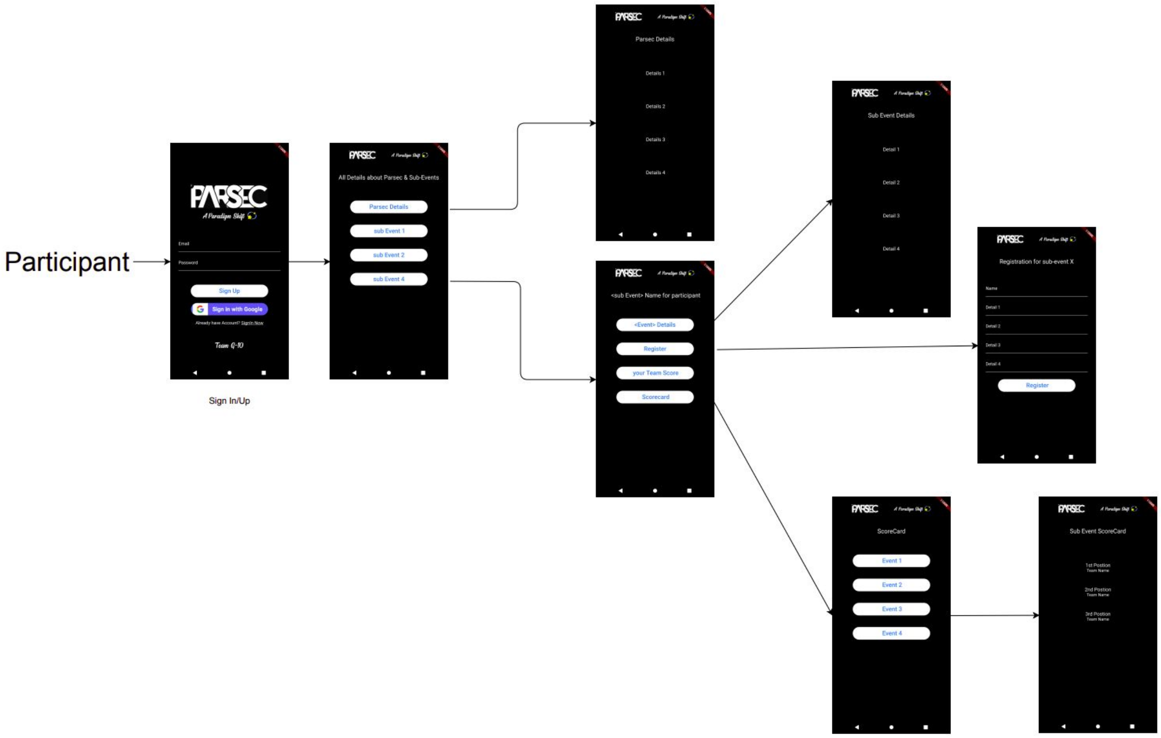
- Individual_participant_invilgilator_view (can modify some part individual_participant table)
- Group_participant_invilgilator_view (can modify some part group_participant table)

-> Admin has full access to the whole database other than personal data of participants & guests.
Admin will have access (can modify, can read) for the whole database.

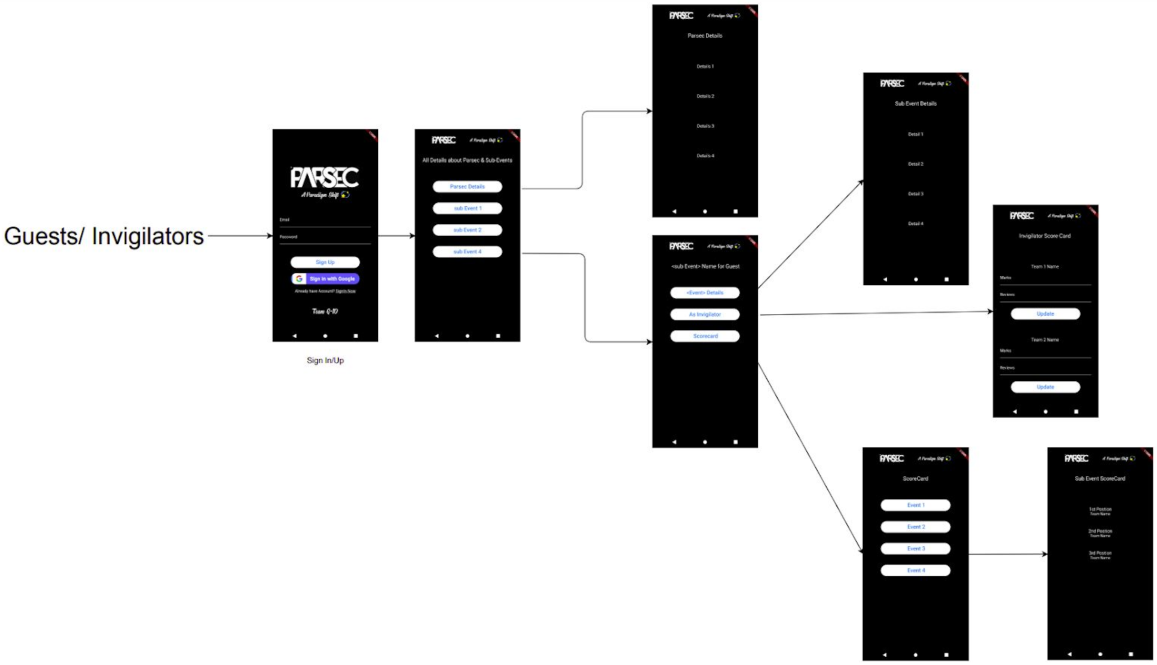
Interface design (More Clear images - pdf)



Interface for Admins



Interface for Participants



Interface for Invigilators

Module design :

same for all	SignIn.dart	Sign in page
	SignUp.dart	sign up page
	FestDetails.dart	All activities of fest
	ParsecDetails.dart	Fest details
	EventDetails.dart	Each event details
	ScoreCard.dart	scorecard of all events
	EventScoreCard.dart	scorecard of each event
Participant	Register.dart	Registration for the event
	ParticipantEventDetails.dart	Event related stuff for participants
Guest / Invigilator	GuestsEventForGuest.dart	Event related stuff for Guests
	AsInvigilator.dart	Score editor for invigilator
Admin	AdminOptions.dart	All options for Admin
	AdminDetailsCheck.dart ● EventDetails.dart ● AdminListsGuests.dart ● AdminListsSponsors.dart ● AdminListsRegisteredStudents.dart	Admin can see all details about ● each event ● guest ● sponsor ● participant
	AdminDetailsUpdate.dart ● AdminParsecDetails,dart ● AdminEvent.dart ● AdminGuests.dart ● AdminSponsor.dart	Admin can edit details about ● Parsec ● Event ● Guest ● sponsor

Sample data for project :

```
INSERT INTO fest VALUES ('11111','PARSEC','IIT Dharwad','2011-09-23','3 days');

INSERT INTO fest VALUES ('12131','Fest2','IIT Bombay','2012-01-02','7 days');
INSERT INTO fest VALUES ('21451','Fest3','IIT Madras','2010-03-10','2 days');

/*      For Event       */
INSERT INTO evento VALUES ('11223','Event1','2011-09-23 09:00:00','2011-09-23
12:00:00','2011-09-23 09:00:00','2011-09-23 12:00:00','place','This is Description for event-1');
INSERT INTO evento VALUES ('31413','Event2','2011-09-23 14:00:00','2011-09-24
09:00:00','2011-09-23 09:00:00','2011-09-23 12:00:00','--','This is Description for event-2');
INSERT INTO evento VALUES ('21111','Event3','2012-01-02 10:30:00','2012-01-04
14:00:00','2011-09-23 09:00:00','2011-09-23 12:00:00','--','This is Description for event-3');
INSERT INTO evento VALUES ('12222','Event4','2010-03-10 15:00:00','2010-03-10
18:00:00','2011-09-23 09:00:00','2011-09-23 12:00:00','--','This is Description for event-4');
INSERT INTO evento VALUES ('12345','Event5','2010-03-12 19:00:00','2010-03-12
09:30:00','2011-09-23 09:00:00','2011-09-23 12:00:00','--','This is Description for event-5');

/*      For Has         */
INSERT INTO has VALUES ('11223','11111');
INSERT INTO has VALUES ('31413','11111');
INSERT INTO has VALUES ('21111','12131');
INSERT INTO has VALUES ('12222','21451');
INSERT INTO has VALUES ('12345','21451');

/*      For participant */
INSERT INTO participant VALUES ('55555','Josh',9992341234,'josh@gmail.com');
INSERT INTO participant VALUES ('54325','Kane',8892384727,'kane@gmail.com');
INSERT INTO participant VALUES ('45632','Avic',8765678186,'avic@gmail.com');
INSERT INTO participant VALUES ('66554','Zinda',7689841234,'zinda@gmail.com');
INSERT INTO participant VALUES ('61231','Frank',6799874321,'frank@gmail.com');
```

```sql
/*      For individual_participant */
INSERT INTO individual_participant VALUES ('55555','11223',0,'Score review');
INSERT INTO individual_participant VALUES ('54325','31413',0,'Score review');
INSERT INTO individual_participant VALUES ('45632','31413',0,'Score review');
INSERT INTO individual_participant VALUES ('66554','21111',0,'Score review');
INSERT INTO individual_participant VALUES ('61231','12222',0,'Score review');

/*      For group_participant    */
INSERT INTO group_participant VALUES ('55555','12345',0,'1','The Gang','Score review');
INSERT INTO group_participant VALUES ('54325','12345',0,'1','The Gang','Score review');
INSERT INTO group_participant VALUES ('61231','12345',0,'1','The Gang','Score review');

/*      Admin    */
INSERT INTO admino VALUES ('12127','Sriram',8889991234,'sri@gmail.com','123');
INSERT INTO admino VALUES ('29833','Manjeet',9837593752,'jeet@gmail.com','123');

/*      For Sponsor */
INSERT INTO sponsor VALUES ('85732','Github',2873425342,'www.gith**.com');
INSERT INTO sponsor VALUES ('27462','Laboget',7632342311,'www.labdh**.com');

/*      For gives */
INSERT INTO gives VALUES ('85732','11223',32000);
INSERT INTO gives VALUES ('85732','21111',52000);
INSERT INTO gives VALUES ('27462','11223',50000);
INSERT INTO gives VALUES ('27462','31413',70000);

/*      For guests */
INSERT INTO guest VALUES ('73452','Dankrel','breif abt dankrel',9583452734,'dank@gmail.com');
INSERT INTO guest VALUES ('16353','Wrick','breif abt wrick',9128882635,'wrik@gmail.com');

/*      For invite   */
INSERT INTO invite VALUES ('73452','11223');
INSERT INTO invite VALUES ('73452','31413');
INSERT INTO invite VALUES ('16353','11223');
```

Work plan :

Firstly we held a google meet session for discussing all the things among us then where we discussed about brief project overview and how many tables in database we need to get our work done then we wrote brief project overview at same time then we prepared ER-model by hit/trial or we can say checking 2-3 times if anything missing or not and checking that our whole work can be done by this ER-model.

After that, we made ER-diagram together while discussing about different parameters on google meeting only then manjeet kapil prepared Relation database and then shriram and yashwanth checked in that if everything is fine or not then shriram putted some views that we will be using in the project.We discussed about interface design how we will be doing that then shriram prepared them then we discussed about module design where manjeet and yashwanth prepared them then sampled data has been prepared by yashwanth which has been reviewed by manjeet and shriram.