

Project Report

Course Information: COP 4710, Spring 2023

Group Member: Yuyang Zhang

Table of contents:

Project Description	2
GUI.....	2
ER-Model.....	8
Relational data model in SQL.....	9
Populating tables with sample data.....	12
SQL examples and results	
➤ Create RSO.....	13
➤ Join RSO.....	14
➤ Create Event.....	15
➤ Insert/update a (new) comment.....	16
➤ View Event.....	17
Constraint Enforcement.....	17
Conclusion/Observation.....	22

Project Description

The goal of this project is to develop a web application that will enable students from different universities to build and manage events for their community. These activities can be private, public, or exclusive to a specific Registered Student Organization (RSO) established through the application. The program's audience is divided into three user levels: regular users, who can request to join or create RSOs and participate in available activities; administrators, who are affiliated with a university and one or more RSOs, who have access to RSOs; and super administrators, who create a profile for the university with administrator privileges. In this app, public events can be seen by everyone; private events can be seen by students of the host university; and RSO events can only be seen by members of the RSO. Users can create events without RSO (public events). Such events must be approved by a super administrator. After an event is posted, users can add, delete, and edit their comments on the event.

Advanced features not completed:

1. The location information is not connected with the map, only the latitude and longitude of the input location are provided.
2. Users cannot rate events on a scale of 1-5 (stars)
3. The application cannot provide some social network integrations, for example, posting to Facebook or Google from the application

GUI

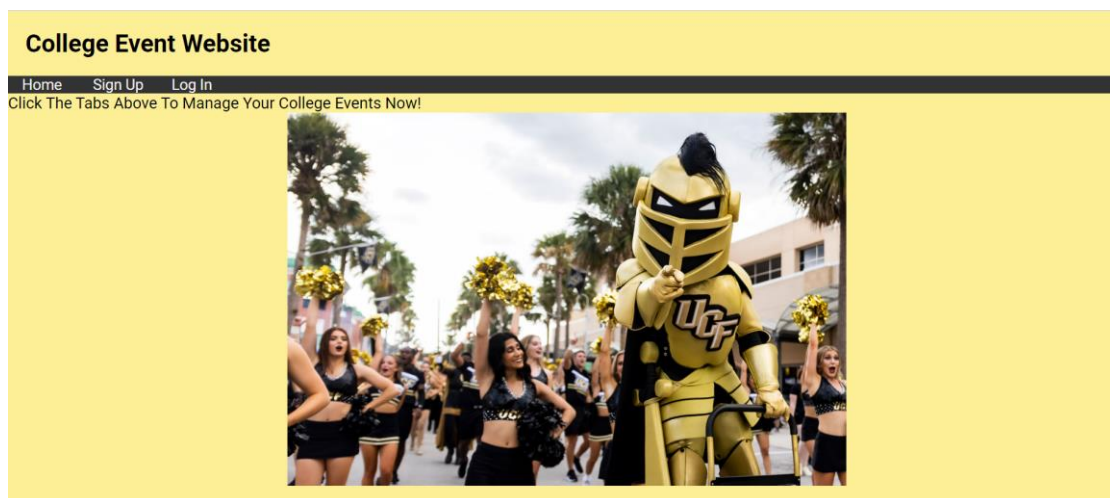
The MySQLi backend using InnoDB was employed to enable functional dependencies for the web-based application. Server-user connections were established with PHP,

while the front-end was built using HTML, CSS, and JS. The project was developed and tested locally on a Windows-based application called WAMP, which simplifies development by automatically integrating Apache, MySQLi, and PHP. The front-end visuals were verified using Chrome.

Screenshots:

(Please ignore the Chinese showing in date selection. This is due to the language set on computer.)

Home page for any user before login:



Sign up:

A screenshot of the 'College Event Website' sign-up page. The page has a yellow background. At the top, there is a black navigation bar with the text 'College Event Website' in white. Below the navigation bar, there are three links: 'Home', 'Sign Up', and 'Log In'. The 'Sign Up' link is highlighted. The main content area contains a sign-up form with the following fields: 'Username', 'Password', 'Repeat Your Password', and 'University Name...'. Each field has a corresponding input line. At the bottom of the form, there is a green button labeled 'Sign Up'.

A super admin who need to write profile for his/her university after he/she sign up with a new university:

College Event Website

[Home](#) [Sign Up](#) [Log In](#)

New University Information

To be the super-admin for new university, complete the following new university sign up.

UNIVERSITY LOCATION

Unviersity Location Latitude...

Unviersity Location Longitude...

UNIVERSITY INFOMATION

Unviersity Description...

Sign Up

Login:

College Event Website

[Home](#) [Sign Up](#) [Log In](#)

Login

Username

Password

Login


Super admin home page after login:

College Event Website

[Home](#) [My Events](#) [My RSOs](#) [Existing Events](#) [Existing RSOs](#) [Request](#) [Pending Events](#) [Make An RSO](#) [Log Out](#)

Welcome, ucfspadmin!

Click The Tabs Above To Manage Your College Events Now!




Admin and normal user home page:

College Event Website

HomeMy EventsMy RSOSExisting EventsExisting RSOSRequestMake An RSOLog Out

Welcome, ucfs1!

Click The Tabs Above To Manage Your College Events Now!



My Events:

College Event Website

HomeMy EventsMy RSOSExisting EventsExisting RSOSCreateRequestPending EventsMake An RSOLog Out

Your Events

Basketball Fire
UCF

Soccer game
04-08-2023
04pm
90 min soccer game
Latitude: 28.6024274
Longitude: -81.2000599
Phone #: 123-456-7890
Visibility: PUBLIC

Comments

Withdraw

My RSOs:

College Event Website

HomeMy EventsMy RSOSExisting EventsExisting RSOSCreateRequestPending EventsMake An RSOLog Out

Your RSOs

UCF

Basketball Fire
Super Basketball Players
ever in this world

Leave

Existing Events:

College Event Website

HomeMy EventsMy RSOsExisting EventsExisting RSOsCreateRequestPending EventsMake An RSOLog Out

Available Events

Basketball Fire

Soccer game

UCF

04-08-2023

04pm

90 min soccer game

Latitude: 28.6024274

Longitude: -81.2000599

Phone #: 123-456-7890

Visibility: PUBLIC

UCF

Practice game

04-06-2023

03am

practice with other teams

Latitude: 28.6024274

Longitude: -81.2000599

Phone #: 123-456-7890

Visibility: PRIVATE

Join This Event

Existing RSOs:

College Event Website

HomeMy EventsMy RSOsExisting EventsExisting RSOsCreateRequestPending EventsMake An RSOLog Out

All RSOs For UCF

UCF

Basketball Fire

Super Basketball Players ever in this world

Create:

College Event Website

HomeMy EventsMy RSOsExisting EventsExisting RSOsCreateRequestPending EventsMake An RSOLog Out

Create An Event For Your RSO

EVENT INFOMATION

NO ACTIVE RSOs

Your Event Name...

Your Event Description...

Contact Number: 123-456-7890

EVENT DATE AND TIME

年/月/日

(second half on next page)

EVENT DATE AND TIME

年 / 月 / 日

12

AM

EVENT LOCATION

Your Event Location Latitude...

Your Event Location Longitude...

EVENT TYPE

Select Your Event Type...

Submit

Request:

College Event Website

Home

My Events

My RSOs

Existing Events

Existing RSOs

Create

Request

Pending Events

Make An RSO

Log Out

Request An Event

Since you are the super admin, you can request an event and then accept that request in 'Pending Events'.

EVENT INFORMATION

Your Event Name...

Your Event Description...

Contact Number: 123-456-7890

EVENT DATE AND TIME

年 / 月 / 日

Pending Events:

College Event Website

Home

My Events

My RSOs

Existing Events

Existing RSOs

Create

Request

Pending Events

Make An RSO

Log Out

Pending Events Waiting For You To Check

UCF

Practice game

04-06-2023

03am

practice with other teams

Latitude: 28.6024274

Longitude: -81.2000599

Phone #: 123-456-7890

Visibility: PRIVATE

ACCEPT

DENY

Make An RSO:

College Event Website

HomeMy EventsMy RSOsExisting EventsExisting RSOsRequestPending EventsMake An RSOLog Out

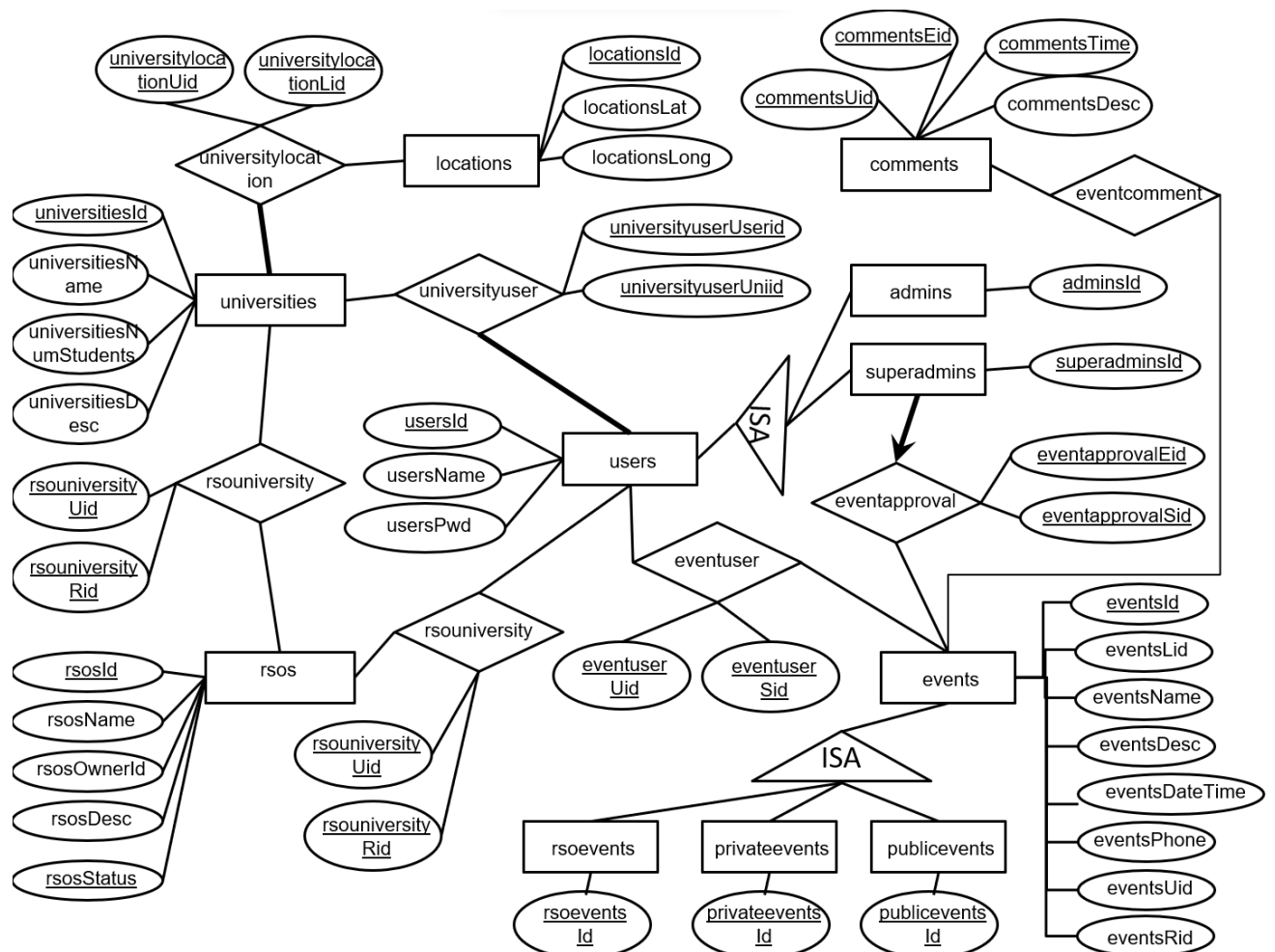
Make An RSO

RSO Name

Description

Submit

ER-Model



Relational data model in SQL

```
drop DATABASE IF EXISTS college_events;
create DATABASE college_events;
use college_events;

CREATE TABLE locations (
    locationsId INT(11) PRIMARY KEY AUTO_INCREMENT NOT NULL,
    locationsLong VARCHAR(128) NOT NULL,
    locationsLat VARCHAR(128) NOT NULL,
    UNIQUE(locationsLong,locationsLat)
)ENGINE = InnoDB;

CREATE TABLE universities (
    universitiesId INT(11) PRIMARY KEY AUTO_INCREMENT NOT NULL,
    universitiesName VARCHAR(30) NOT NULL,
    universitiesNumStudents INT(11) NOT NULL DEFAULT 0,
    universitiesDesc VARCHAR(255),
    UNIQUE(universitiesName)
)ENGINE = InnoDB;

CREATE TABLE universitylocation (
    universitylocationUid INT(11) NOT NULL,
    universitylocationLid INT(11) NOT NULL,
    PRIMARY KEY(universitylocationUid, universitylocationLid),
    CONSTRAINT universitylocationuid_fk
    FOREIGN KEY(universitylocationUid) REFERENCES universities(universitiesId) ON DELETE CASCADE,
    CONSTRAINT universitylocationlid_fk
    FOREIGN KEY(universitylocationLid) REFERENCES locations(locationsId) ON DELETE CASCADE
)ENGINE = InnoDB;

CREATE TABLE users (
    usersId INT(11) PRIMARY KEY AUTO_INCREMENT NOT NULL,
    usersName VARCHAR(128) NOT NULL,
    usersPwd VARCHAR(128) NOT NULL
)ENGINE = InnoDB;

CREATE TABLE universityuser (
    universityuserUserId INT(11) NOT NULL,
    universityuserUniId INT(11) NOT NULL,
    PRIMARY KEY(universityuserUserId, universityuserUniId),
    CONSTRAINT universityuseruserid_fk
    FOREIGN KEY(universityuserUserId) REFERENCES users(usersId) ON DELETE CASCADE,
    CONSTRAINT universityuseruniid_fk
    FOREIGN KEY(universityuserUniId) REFERENCES universities(universitiesId) ON DELETE CASCADE
)ENGINE = InnoDB;

CREATE TABLE superadmins (
    superadminsId INT(11) NOT NULL,
    PRIMARY KEY(superadminsId),
    CONSTRAINT superadminssid_fk
    FOREIGN KEY(superadminsId) REFERENCES users(usersId) ON DELETE CASCADE
)ENGINE = InnoDB;

CREATE TABLE admins (
    adminsId INT(11) NOT NULL,
    PRIMARY KEY(adminsId),
    CONSTRAINT adminsaid_fk
    FOREIGN KEY(adminsId) REFERENCES users(usersId) ON DELETE CASCADE
)ENGINE = InnoDB;
```

```
CREATE TABLE admins (
  adminsId INT(11) NOT NULL,
  PRIMARY KEY(adminsId),
  CONSTRAINT adminsaid_fk
  FOREIGN KEY(adminsId) REFERENCES users(usersId) ON DELETE CASCADE
)ENGINE = InnoDB;
```

```
CREATE TABLE rsos (
  rsosId INT(11) PRIMARY KEY AUTO_INCREMENT NOT NULL,
  rsosName VARCHAR(50) NOT NULL,
  rsosDesc VARCHAR(255),
  rsosOwnerId INT(11) NOT NULL,
  rsosStatus ENUM("active","inactive") DEFAULT "inactive",
  CONSTRAINT rsosoid_fk
  FOREIGN KEY(rsosOwnerId) REFERENCES users(usersId) ON DELETE CASCADE,
  UNIQUE(rsosName)
)ENGINE = InnoDB;
```

```
CREATE TABLE rsouniversity (
  rsouniversityUid INT(11),
  rsouniversityRid INT(11) NOT NULL,
  PRIMARY KEY(rsouniversityUid,rsouniversityRid),
  CONSTRAINT rsouniversityuid_fk
  FOREIGN KEY(rsouniversityUid) REFERENCES universities(universitiesId) ON DELETE CASCADE,
  CONSTRAINT rsouniversityrid_fk
  FOREIGN KEY(rsouniversityRid) REFERENCES rsos(rsosId) ON DELETE CASCADE
)ENGINE = InnoDB;
```

```
CREATE TABLE rsouser (
  rsouserUid INT(11) NOT NULL,
  rsouserRid INT(11) NOT NULL,
  PRIMARY KEY(rsouserUid,rsouserRid),
  CONSTRAINT rsouseruid_fk
  FOREIGN KEY(rsouserUid) REFERENCES users(usersId) ON DELETE CASCADE,
  CONSTRAINT rsouserid_fk
  FOREIGN KEY(rsouserRid) REFERENCES rsos(rsosId) ON DELETE CASCADE
)ENGINE = InnoDB;
```

```
CREATE TABLE events (
  eventsId INT(11) PRIMARY KEY AUTO_INCREMENT NOT NULL,
  eventsLid INT(11) NOT NULL,
  eventsName VARCHAR(50) NOT NULL,
  eventsDesc VARCHAR(255) NOT NULL,
  eventsDateTime DATETIME NOT NULL,
  eventsPhone VARCHAR(12) NOT NULL,
  eventsUid INT(11) NOT NULL,
  eventsRid INT(11),
  CONSTRAINT eventsuid_fk
  FOREIGN KEY(eventsUid) REFERENCES universities(universitiesId) ON DELETE CASCADE,
  CONSTRAINT eventslid_fk
  FOREIGN KEY(eventsLid) REFERENCES locations(locationsId) ON DELETE CASCADE,
  CONSTRAINT eventsrid_fk
  FOREIGN KEY(eventsRid) REFERENCES rsos(rsosId) ON DELETE CASCADE,
  UNIQUE(eventsLid,eventsDateTime)
)ENGINE = InnoDB;
```

```

CREATE TABLE eventuser (
    eventuserUid INT(11) NOT NULL,
    eventuserEid INT(11) NOT NULL,
    PRIMARY KEY(eventuserUid,eventuserEid),
    CONSTRAINT eventuseruid_fk
    FOREIGN KEY(eventuserUid) REFERENCES users(usersId) ON DELETE CASCADE,
    CONSTRAINT eventusereid_fk
    FOREIGN KEY(eventuserEid) REFERENCES events(eventsId) ON DELETE CASCADE
);

CREATE TABLE privateevents (
    privateeventsId INT(11) NOT NULL,
    PRIMARY KEY(privateeventsId),
    CONSTRAINT privateeventsid_fk
    FOREIGN KEY(privateeventsId) REFERENCES events(eventsId) ON DELETE CASCADE
);

CREATE TABLE publicevents (
    publiceventsId INT(11) NOT NULL,
    PRIMARY KEY(publiceventsId),
    CONSTRAINT publiceventsid_fk
    FOREIGN KEY(publiceventsId) REFERENCES events(eventsId) ON DELETE CASCADE
);

CREATE TABLE rsoevents (
    rsoeventsId INT(11) NOT NULL,
    PRIMARY KEY(rsoeventsId),
    CONSTRAINT rsoeventsid_fk
    FOREIGN KEY(rsoeventsId) REFERENCES events(eventsId) ON DELETE CASCADE
);

CREATE TABLE eventapproval (
    eventapprovalEid INT(11) NOT NULL,
    eventapprovalSid INT(11) NOT NULL,
    PRIMARY KEY(eventapprovalEid,eventapprovalSid),
    CONSTRAINT eventapprovalEid_fk
    FOREIGN KEY(eventapprovalEid) REFERENCES events(eventsId) ON DELETE CASCADE,
    CONSTRAINT eventapprovalsid_fk
    FOREIGN KEY(eventapprovalSid) REFERENCES superadmins(superadminsId) ON DELETE CASCADE
)ENGINE = InnoDB;

CREATE TABLE comments (
    commentsEid INT(11) NOT NULL,
    commentsUid INT(11) NOT NULL,
    commentsTime DATETIME NOT NULL DEFAULT CURRENT_TIMESTAMP,
    commentsDesc VARCHAR(255) NOT NULL,
    PRIMARY KEY(commentsEid, commentsUid, commentsTime),
    CONSTRAINT commentseid_fk
    FOREIGN KEY(commentsEid) REFERENCES events(eventsId) ON DELETE CASCADE,
    CONSTRAINT commentsuid_fk
    FOREIGN KEY(commentsUid) REFERENCES users(usersId) ON DELETE CASCADE
)ENGINE = InnoDB;

DELIMITER $$
CREATE TRIGGER RSOSStatusUpdateA AFTER INSERT ON rsouser
FOR EACH ROW BEGIN
IF ((SELECT COUNT(*) FROM rsouser M WHERE M.rsouserRid = NEW.rsouserRid) > 4)
THEN
    UPDATE rsos SET rsosStatus = "active" WHERE rsosId = NEW.rsouserRid;
END IF;

```

```

IF (((SELECT COUNT(*) FROM rsos R WHERE R.rsosId = NEW.rsouserRid AND R.rsosOwnerId = NEW.rsouserUid) > 0)
AND ((SELECT COUNT(*) FROM admins A WHERE A.adminsId = NEW.rsouserUid) < 1))
THEN
    INSERT INTO admins(adminsId) VALUES(NEW.rsouserUid);
END IF;
END;$$

CREATE TRIGGER RSOSStatusUpdateI AFTER DELETE ON rsouser
FOR EACH ROW BEGIN
IF ((SELECT COUNT(*) FROM rsouser M WHERE M.rsouserRid = OLD.rsouserRid) < 5)
THEN
    UPDATE rsos SET rsosStatus = "inactive" WHERE rsosId = OLD.rsouserRid;
    IF ((SELECT COUNT(*) FROM rsos N WHERE N.rsosOwnerId = OLD.rsouserUid) < 1)
    THEN
        DELETE FROM admins A WHERE A.adminsId = OLD.rsouserUid;
    END IF;
END IF;
END;$$

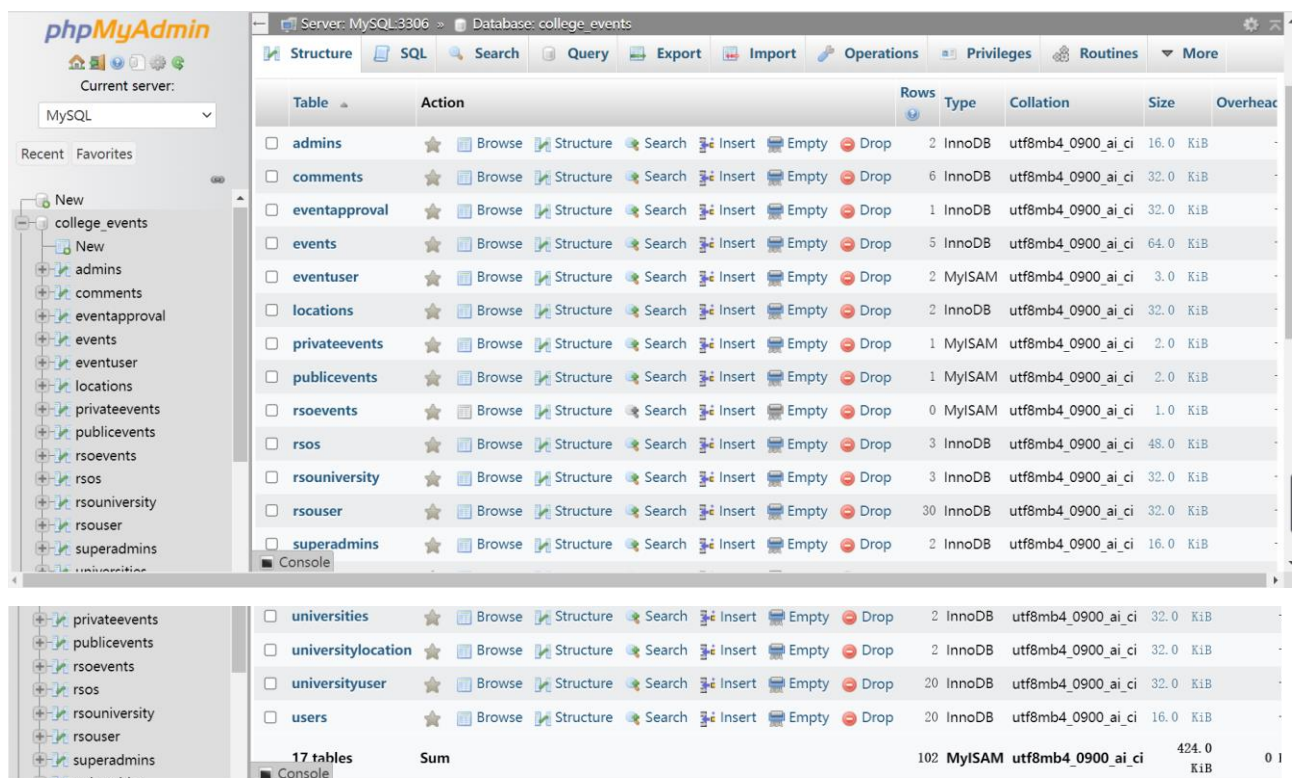
CREATE TRIGGER UniversityStudentCntA AFTER INSERT ON universityuser
FOR EACH ROW BEGIN
UPDATE universities SET universitiesNumStudents = universitiesNumStudents + 1 WHERE universitiesId = NEW.universityuserUniid;
END;$$

CREATE TRIGGER UniversityStudentCntS AFTER DELETE ON universityuser
FOR EACH ROW BEGIN
UPDATE universities SET universitiesNumStudents = universitiesNumStudents - 1 WHERE universitiesId = OLD.universityuserUniid;
END;$$

DELIMITER ;

```

Populating tables with sample data



The screenshot shows the phpMyAdmin interface for a MySQL database named 'college_events'. The left sidebar displays the database structure, and the main panel shows a list of tables with their respective actions, row counts, and storage engines.

Table	Action	Rows	Type	Collation	Size	Overhead
admins	Browse Structure Search Insert Empty Drop	2	InnoDB	utf8mb4_0900_ai_ci	16.0 KiB	-
comments	Browse Structure Search Insert Empty Drop	6	InnoDB	utf8mb4_0900_ai_ci	32.0 KiB	-
eventapproval	Browse Structure Search Insert Empty Drop	1	InnoDB	utf8mb4_0900_ai_ci	32.0 KiB	-
events	Browse Structure Search Insert Empty Drop	5	InnoDB	utf8mb4_0900_ai_ci	64.0 KiB	-
eventuser	Browse Structure Search Insert Empty Drop	2	MyISAM	utf8mb4_0900_ai_ci	3.0 KiB	-
locations	Browse Structure Search Insert Empty Drop	2	InnoDB	utf8mb4_0900_ai_ci	32.0 KiB	-
privateevents	Browse Structure Search Insert Empty Drop	1	MyISAM	utf8mb4_0900_ai_ci	2.0 KiB	-
publicevents	Browse Structure Search Insert Empty Drop	1	MyISAM	utf8mb4_0900_ai_ci	2.0 KiB	-
rsoevents	Browse Structure Search Insert Empty Drop	0	MyISAM	utf8mb4_0900_ai_ci	1.0 KiB	-
rsos	Browse Structure Search Insert Empty Drop	3	InnoDB	utf8mb4_0900_ai_ci	48.0 KiB	-
rsouniversity	Browse Structure Search Insert Empty Drop	3	InnoDB	utf8mb4_0900_ai_ci	32.0 KiB	-
rsouser	Browse Structure Search Insert Empty Drop	30	InnoDB	utf8mb4_0900_ai_ci	32.0 KiB	-
superadmins	Browse Structure Search Insert Empty Drop	2	InnoDB	utf8mb4_0900_ai_ci	16.0 KiB	-
universities	Browse Structure Search Insert Empty Drop	2	InnoDB	utf8mb4_0900_ai_ci	32.0 KiB	-
universitylocation	Browse Structure Search Insert Empty Drop	2	InnoDB	utf8mb4_0900_ai_ci	32.0 KiB	-
universityuser	Browse Structure Search Insert Empty Drop	20	InnoDB	utf8mb4_0900_ai_ci	32.0 KiB	-
users	Browse Structure Search Insert Empty Drop	20	InnoDB	utf8mb4_0900_ai_ci	16.0 KiB	-
17 tables	Sum	102	MyISAM	utf8mb4_0900_ai_ci	424.0 KiB	0

Sample Data contains:

Two universities: UCF, USF

20 students: ucfsadmin (super admin of UCF), ucfs1, ucfs2, ucfs3, ucfs4, ucfs5, ucfs6,

ucfs7, ucfs8, ucfs9, usfspadmin (super admin of USF), usfs1, usfs2, usfs3, usfs4, usfs5,

usfs6, usfs7, usfs8, usfs9.

3 RSOs: Basketball Fire (admin: ucfspadmin), Soccer Fire (admin: ucfspadmin), Soccer Flood (admin: usfspadmin)

Five events: Soccer game, Lunch Time!, Practice game (private for UCF), Practice game (RSO for USF), Hang Out! (Pending Events for UCF)

Three comments:

By **ucfspadmin** on 2023-04-05 21:58:41
Believe or not, USF can't even beat our basketball club on soccer lol

By **ucfs1** on 2023-04-05 21:59:54
What? We let basketball club to play soccer with USF?!

By **ucfs2** on 2023-04-05 22:01:41
That's OK guys. We'll win as easy as drinking a cup of coffee.

SQL examples and results

1. Create RSO:

Before:

✓ Showing rows 0 - 3 (4 total, Query took 0.0021 seconds.)

SELECT * FROM `rsos`

☐ Profiling [[Edit inline](#)] [[Edit](#)] [[Explain SQL](#)] [[Create PHP code](#)] [[Refresh](#)]

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

		rsosId	rsosName	rsosDesc	rsosOwnerId	rsosStatus
<input type="checkbox"/>	Edit Copy Delete	1	Basketball Fire	Super Basketball Players ever in this world	1	active
<input type="checkbox"/>	Edit Copy Delete	4	Soccer Fire	Super Soccer Players ever in this world	1	active
<input type="checkbox"/>	Edit Copy Delete	5	Soccer Flood	Super Soccer Players ever in this world	6	active
<input type="checkbox"/>	Edit Copy Delete	6	Volleyball Fire	Super Volleyball Players ever in this world	2	inactive

Execute:

✓ 1 row inserted.
Inserted row id: 7 (Query took 0.0285 seconds.)

```
INSERT INTO rsos( rsosName, rsosDesc, rsosOwnerId ) VALUES( 'Baseball Fire', 'SUPER Baseball Players ever IN this world', 1 );
```

[Edit inline] [Edit] [Create PHP code]

✓ 1 row inserted.
Inserted row id: 7 (Query took 0.0460 seconds.)

```
INSERT INTO rsouniversity( rsouniversityUid, rsouniversityRid ) VALUES( 2, 7 );
```

[Edit inline] [Edit] [Create PHP code]

After:

✓ Showing rows 0 - 4 (5 total, Query took 0.0012 seconds.)

```
SELECT * FROM `rsos`
```

☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

	rsosId	rsosName	rsosDesc	rsosOwnerId	rsosStatus
<input type="checkbox"/> Edit Copy Delete	1	Basketball Fire	Super Basketball Players ever in this world	1	active
<input type="checkbox"/> Edit Copy Delete	4	Soccer Fire	Super Soccer Players ever in this world	1	active
<input type="checkbox"/> Edit Copy Delete	5	Soccer Flood	Super Soccer Players ever in this world	6	active
<input type="checkbox"/> Edit Copy Delete	6	Volleyball Fire	Super Volleyball Players ever in this world	2	inactive
<input type="checkbox"/> Edit Copy Delete	7	Baseball Fire	SUPER Baseball Players ever IN this world	1	inactive

2. Join RSO:

Before:

	rsouserUid	rsouserRid
<input type="checkbox"/> Edit Copy Delete	1	1
<input type="checkbox"/> Edit Copy Delete	2	1
<input type="checkbox"/> Edit Copy Delete	3	1
<input type="checkbox"/> Edit Copy Delete	4	1
<input type="checkbox"/> Edit Copy Delete	5	1
<input type="checkbox"/> Edit Copy Delete	11	1
<input type="checkbox"/> Edit Copy Delete	12	1
<input type="checkbox"/> Edit Copy Delete	13	1
<input type="checkbox"/> Edit Copy Delete	14	1
<input type="checkbox"/> Edit Copy Delete	15	1
<input type="checkbox"/> Edit Copy Delete	1	4
<input type="checkbox"/> Edit Copy Delete	2	4
<input type="checkbox"/> Edit Copy Delete	3	4

Execute:

```
✓ 1 row inserted. (Query took 0.0146 seconds.)

INSERT INTO rsouser (rsouserUid, rsouserRid) VALUES (6, 1);
```

After:

		rsouserUid	rsouserRid
<input type="checkbox"/>	Edit Copy Delete	1	1
<input type="checkbox"/>	Edit Copy Delete	2	1
<input type="checkbox"/>	Edit Copy Delete	3	1
<input type="checkbox"/>	Edit Copy Delete	4	1
<input type="checkbox"/>	Edit Copy Delete	5	1
<input type="checkbox"/>	Edit Copy Delete	6	1
<input type="checkbox"/>	Edit Copy Delete	11	1
<input type="checkbox"/>	Edit Copy Delete	12	1
<input type="checkbox"/>	Edit Copy Delete	13	1
<input type="checkbox"/>	Edit Copy Delete	14	1

3. Create Event:

Before:

		eventsId	eventsLid	eventsName	eventsDesc	eventsDateTime	eventsPhone	eventsUid	eventsRid
<input type="checkbox"/>	Edit Copy Delete	1	1	Soccer game	90 min soccer game	2023-04-08 16:00:00	123-456-7890	1	1
<input type="checkbox"/>	Edit Copy Delete	2	1	Practice game	practice with other teams	2023-04-06 03:00:00	123-456-7890	1	NULL
<input type="checkbox"/>	Edit Copy Delete	3	1	Lunch Time!	Have lunch with team	2023-04-05 14:00:00	123-456-7890	1	NULL
<input type="checkbox"/>	Edit Copy Delete	4	2	Practice game	practice with other teams	2023-04-05 15:00:00	123-456-7890	2	5
<input type="checkbox"/>	Edit Copy Delete	5	1	Hang out!	Hang out!	2023-04-10 13:00:00	123-456-7890	1	NULL

Execute:

```
✓ 1 row inserted.
Inserted row id: 6 (Query took 0.0076 seconds.)

INSERT INTO EVENTS( eventsName, eventsDesc, eventsPhone, eventsDateTime, eventsLid, eventsUid ) VALUES( 'Have Fun', 'Have fun with team', '123-456-7890', '2023-04-05 14:00:00', 2, 2 );

[ Edit inline ] [ Edit ] [ Create PHP code ]

✓ 1 row inserted.
Inserted row id: 6 (Query took 0.0043 seconds.)

INSERT INTO publicevents(publiceventsId) VALUES(6);

[ Edit inline ] [ Edit ] [ Create PHP code ]
```


After:

<div><div><div>←</div><div>→</div></div></div>			eventsId	eventsLid	eventsName	eventsDesc	eventsDateTime	eventsPhone	eventsUid	eventsRid
<div><div><div><div></div></div></div><div><div><div>Edit</div><div>Copy</div><div>Delete</div></div></div></div>			1	1	Soccer game	90 min soccer game	2023-04-08 16:00:00	123-456-7890	1	1
<div><div><div><div></div></div></div><div><div><div>Edit</div><div>Copy</div><div>Delete</div></div></div></div>			2	1	Practice game	practice with other teams	2023-04-06 03:00:00	123-456-7890	1	NULL
<div><div><div><div></div></div></div><div><div><div>Edit</div><div>Copy</div><div>Delete</div></div></div></div>			3	1	Lunch Time!	Have lunch with team	2023-04-05 14:00:00	123-456-7890	1	NULL
<div><div><div><div></div></div></div><div><div><div>Edit</div><div>Copy</div><div>Delete</div></div></div></div>			4	2	Practice game	practice with other teams	2023-04-05 15:00:00	123-456-7890	2	5
<div><div><div><div></div></div></div><div><div><div>Edit</div><div>Copy</div><div>Delete</div></div></div></div>			5	1	Hang out!	Hang out!	2023-04-10 13:00:00	123-456-7890	1	NULL
<div><div><div><div></div></div></div><div><div><div>Edit</div><div>Copy</div><div>Delete</div></div></div></div>			6	2	Have Fun	Have fun with team	2023-04-05 14:00:00	123-456-7890	2	NULL

4. Insert/update a (new) comment:

Before Insert:

<div><div><div><div></div><div></div></div><div></div></div></div>			commentsEid	commentsUid	commentsTime	commentsDesc	
<input type="checkbox"/>	<div><div><div></div><div></div></div><div>Edit</div></div>	<div><div><div></div><div></div></div><div>Copy</div></div>	<div><div><div></div><div></div></div><div>Delete</div></div>	1	1	2023-04-05 21:58:41	Believe or not, USF can't even beat our basketball...
<input type="checkbox"/>	<div><div><div></div><div></div></div><div>Edit</div></div>	<div><div><div></div><div></div></div><div>Copy</div></div>	<div><div><div></div><div></div></div><div>Delete</div></div>	1	2	2023-04-05 21:59:54	What? We let basketball club to play soccer with U...
<input type="checkbox"/>	<div><div><div></div><div></div></div><div>Edit</div></div>	<div><div><div></div><div></div></div><div>Copy</div></div>	<div><div><div></div><div></div></div><div>Delete</div></div>	1	3	2023-04-05 22:01:41	That's OK guys. We'll win as easy as drinking a cu...
<input type="checkbox"/>	<div><div><div></div><div></div></div><div>Edit</div></div>	<div><div><div></div><div></div></div><div>Copy</div></div>	<div><div><div></div><div></div></div><div>Delete</div></div>	1	6	2023-04-05 22:03:11	That was fun, hope your coffee doesn't burn your h...
<input type="checkbox"/>	<div><div><div></div><div></div></div><div>Edit</div></div>	<div><div><div></div><div></div></div><div>Copy</div></div>	<div><div><div></div><div></div></div><div>Delete</div></div>	1	7	2023-04-05 22:04:22	What the ****! You **** guys will *****! F*****...
<input type="checkbox"/>	<div><div><div></div><div></div></div><div>Edit</div></div>	<div><div><div></div><div></div></div><div>Copy</div></div>	<div><div><div></div><div></div></div><div>Delete</div></div>	1	8	2023-04-05 22:06:10	Dude, shut up, and haven't you brushed for a few d...

Execute:

✓ 1 row inserted. (Query took 0.0051 seconds.)

```
INSERT INTO comments( commentsEid, commentsUid, commentsDesc ) VALUES( 1, 9, 'Stop talking trash words ! ! ! ' );
```

[Edit inline] [Edit] [Create PHP code]

After Insert:

<div>⬅️ ➡️</div>			commentsEid	commentsUid	commentsTime	commentsDesc
<input type="checkbox"/>	<div><div><div><div></div></div><div>Edit</div></div><div><div><div></div></div><div>Copy</div></div><div><div><div></div></div><div>Delete</div></div></div>		1	1	2023-04-05 21:58:41	Believe or not, USF can't even beat our basketball...
<input type="checkbox"/>	<div><div><div><div></div></div><div>Edit</div></div><div><div><div></div></div><div>Copy</div></div><div><div><div></div></div><div>Delete</div></div></div>		1	2	2023-04-05 21:59:54	What? We let basketball club to play soccer with U...
<input type="checkbox"/>	<div><div><div><div></div></div><div>Edit</div></div><div><div><div></div></div><div>Copy</div></div><div><div><div></div></div><div>Delete</div></div></div>		1	3	2023-04-05 22:01:41	That's OK guys. We'll win as easy as drinking a cu...
<input type="checkbox"/>	<div><div><div><div></div></div><div>Edit</div></div><div><div><div></div></div><div>Copy</div></div><div><div><div></div></div><div>Delete</div></div></div>		1	6	2023-04-05 22:03:11	That was fun, hope your coffee doesn't burn your h...
<input type="checkbox"/>	<div><div><div><div></div></div><div>Edit</div></div><div><div><div></div></div><div>Copy</div></div><div><div><div></div></div><div>Delete</div></div></div>		1	7	2023-04-05 22:04:22	What the ****! You **** guys will *****! F*****...
<input type="checkbox"/>	<div><div><div><div></div></div><div>Edit</div></div><div><div><div></div></div><div>Copy</div></div><div><div><div></div></div><div>Delete</div></div></div>		1	8	2023-04-05 22:06:10	Dude, shut up, and haven't you brushed for a few d...
<input type="checkbox"/>	<div><div><div><div></div></div><div>Edit</div></div><div><div><div></div></div><div>Copy</div></div><div><div><div></div></div><div>Delete</div></div></div>		1	9	2023-04-06 14:43:37	Stop talking trash words ! ! !

5. View Event:

Public events:

✓ Showing rows 0 - 2 (3 total, Query took 0.0011 seconds.)

```
SELECT * FROM events INNER JOIN publicevents ON events.eventsId = publicevents.publiceventsId;
```

☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

eventsId	eventsLid	eventsName	eventsDesc	eventsDateTime	eventsPhone	eventsUid	eventsRid	publiceventsId
1	1	Soccer game	90 min soccer game	2023-04-08 16:00:00	123-456-7890	1	1	1
3	1	Lunch Time!	Have lunch with team	2023-04-05 14:00:00	123-456-7890	1	NULL	3
5	1	Hang out!	Hang out!	2023-04-10 13:00:00	123-456-7890	1	NULL	5

Private events:

✓ Showing rows 0 - 0 (1 total, Query took 0.0014 seconds.)

```
SELECT * FROM events INNER JOIN privateevents ON events.eventsId = privateevents.privateeventsId;
```

☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

☐ Show all | Number of rows: 25 | Filter rows: Search this table

Extra options

eventsId	eventsLid	eventsName	eventsDesc	eventsDateTime	eventsPhone	eventsUid	eventsRid	privateeventsId
2	1	Practice game	practice with other teams	2023-04-06 03:00:00	123-456-7890	1	NULL	2

RSO events:

✓ Showing rows 0 - 0 (1 total, Query took 0.0024 seconds.)

```
SELECT * FROM events JOIN rsoevents ON events.eventsId = rsoevents.rsoeventsId;
```

☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

☐ Show all | Number of rows: 25 | Filter rows: Search this table

Extra options

eventsId	eventsLid	eventsName	eventsDesc	eventsDateTime	eventsPhone	eventsUid	eventsRid	rsoeventsId
4	2	Practice game	practice with other teams	2023-04-05 15:00:00	123-456-7890	2	5	4

Constraint Enforcement

When “A new event to be held at the same location and overlapping times with an existing event” happen:

Try to create a “Lunch Time2!” event with overlap time in same place (same latitude

and longitude):

College Event Website

Home

My Events

My RSOs

Existing Events

Existing RSOs

Create

Request

Pending Events

Make An RSO

Log Out

Create An Event For Your RSO

EVENT INFOMATION

Basketball Fire

Lunch Time2!

Have lunch with team

123-456-7890

EVENT DATE AND TIME

2023/04/05

EVENT DATE AND TIME

2023/04/05

2

PM

EVENT LOCATION

28.6024274

-81.2000599

EVENT TYPE

Public

Submit

Result:

College Event Website

Home

My Events

My RSOs

Existing Events

Existing RSOs

Create

Request

Pending Events

Make An RSO

Log Out

Create An Event For Your RSO

Event already exists on 2023-04-05 14:00:00
at LATITUDE= 28.6024274, LONGITUDE= -81.2000599.

EVENT INFOMATION

SELECT RSO

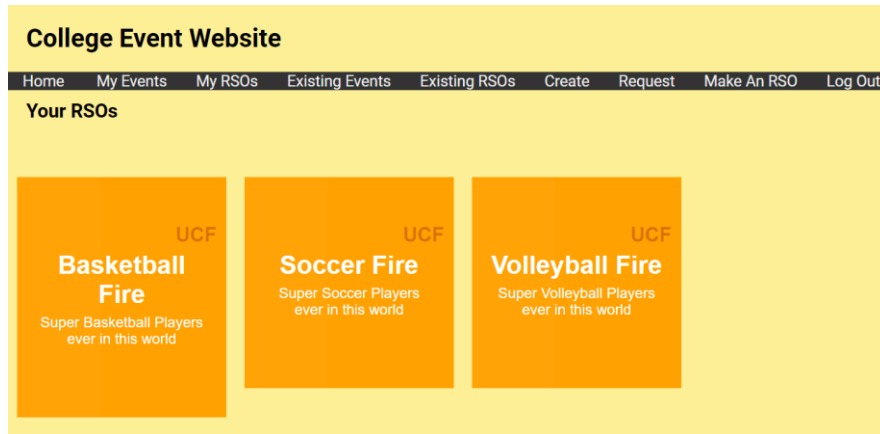
Your Event Name...

Your Event Description...

Contact Number: 123-456-7890

When “An admin who is not the Admin of the RSO attempts to create an event for that RSO”:

“ucfs1” is the admin of “Volleyball Fire” since he has “Create” tab:



In his “Create”, he can only choose his RSO, it does not show other RSO options:

The screenshot shows the 'Create An Event For Your RSO' form. The form is titled 'EVENT INFORMATION' and includes a 'SELECT RSO' dropdown menu. The dropdown menu is open, showing 'Volleyball Fire' as the selected option and 'SELECT RSO' as the default text. Below the dropdown menu are three text input fields: 'Your Event Description...', 'Contact Number: 123-456-7890', and 'EVENT DATE AND TIME'. The 'EVENT DATE AND TIME' field has a date picker icon.

When “An INSERT of a member of an RSO with 4 members: Show the status of the RSO changing to ‘Active’”:

Current RSOs status (Volleyball Fire is inactive):

(nextpage)

☐ Show all

Number of rows: 25

Filter rows: Search this table

Sort by key: None

Extra options

↶

⌵

↷

rsosId

rsosName

rsosDesc

rsosOwnerId

rsosStatus

☐

Edit

Copy

Delete

1 Basketball Fire Super Basketball Players ever in this world1 active

☐

Edit

Copy

Delete

4 Soccer Fire Super Soccer Players ever in this world1 active

☐

Edit

Copy

Delete

5 Soccer Flood Super Soccer Players ever in this world6 active

☐

Edit

Copy

Delete

6 Volleyball Fire Super Volleyball Players ever in this world2 inactive

⬆

☐ Check all

With selected:

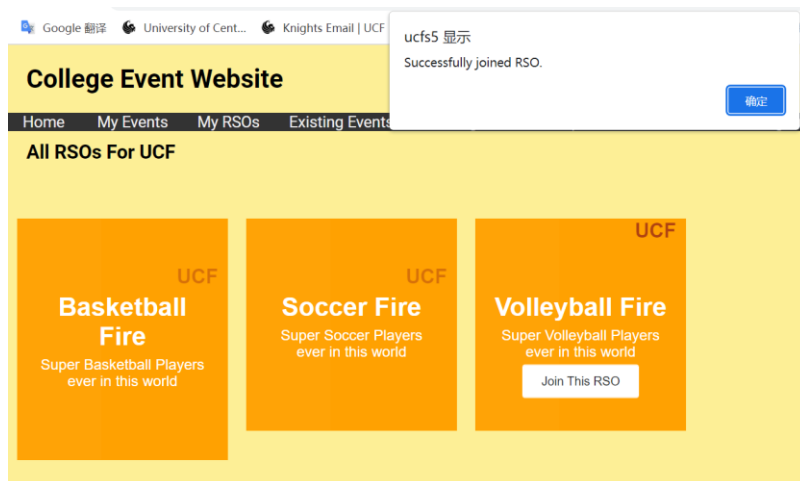
Edit

Copy

Delete

Export

“ucfs5” join the Volleyball Fire RSO:



After “ucfs5” join the Volleyball Fire (Volleyball Fire is active):

☐ Show all

Number of rows: 25

Filter rows: Search this table

Sort by key: None

Extra options

	rsosId	rsosName	rsosDesc	rsosOwnerId	rsosStatus
<input type="checkbox"/> Edit Copy Delete	1	Basketball Fire	Super Basketball Players ever in this world	1	active
<input type="checkbox"/> Edit Copy Delete	4	Soccer Fire	Super Soccer Players ever in this world	1	active
<input type="checkbox"/> Edit Copy Delete	5	Soccer Flood	Super Soccer Players ever in this world	6	active
<input type="checkbox"/> Edit Copy Delete	6	Volleyball Fire	Super Volleyball Players ever in this world	2	active

☐ Check all

With selected:

Edit

Copy

Delete

Export

☐ Show all

Number of rows: 25

Filter rows: Search this table

Sort by key: None

When “A DELETE of a member of an RSO with 5 members: Show the status of the RSO changing to ‘Inactive.’”:

(nextpage)

Current RSOs status (Volleyball Fire is active):

Server: MySQL:3306 » Database: college_events » Table: rsos

BrowserStructureSQLSearchInsertExportImportPrivilegesOperation

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

		rsosId	rsosName	rsosDesc	rsosOwnerId	rsosStatus
<input type="checkbox"/>	Edit Copy Delete	1	Basketball Fire	Super Basketball Players ever in this world	1	active
<input type="checkbox"/>	Edit Copy Delete	4	Soccer Fire	Super Soccer Players ever in this world	1	active
<input type="checkbox"/>	Edit Copy Delete	5	Soccer Flood	Super Soccer Players ever in this world	6	active
<input type="checkbox"/>	Edit Copy Delete	6	Volleyball Fire	Super Volleyball Players ever in this world	2	active

☐ Check all | With selected: Edit Copy Delete Export

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

“ucfs5” left the Volleyball Fire RSO:

Google 翻译 University of Cent... Knights Email | UCF

ucfs5 显示
Successfully left RSO.

确定

College Event Website

Home My Events My RSOs Existing Events

Your RSOs

UCF
Basketball Fire
Super Basketball Players
ever in this world

UCF
Soccer Fire
Super Soccer Players
ever in this world

UCF
Volleyball Fire
Super Volleyball Players
ever in this world
Leave

After “ucfs5” left the Volleyball Fire (Volleyball Fire is inactive):

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

		rsosId	rsosName	rsosDesc	rsosOwnerId	rsosStatus
<input type="checkbox"/>	Edit Copy Delete	1	Basketball Fire	Super Basketball Players ever in this world	1	active
<input type="checkbox"/>	Edit Copy Delete	4	Soccer Fire	Super Soccer Players ever in this world	1	active
<input type="checkbox"/>	Edit Copy Delete	5	Soccer Flood	Super Soccer Players ever in this world	6	active
<input type="checkbox"/>	Edit Copy Delete	6	Volleyball Fire	Super Volleyball Players ever in this world	2	inactive

☐ Check all | With selected: Edit Copy Delete Export

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Conclusion/Observation

The project's database seems to return results in 0.001 seconds on a simple single query.

A query that returns all events visible to the user takes about 0.3 seconds. For ordinary users, this is an efficient database. Programs are encrypted to protect user privacy by using PHP's standard encryption (password). On the surface the user's personal information is safe, but it allows certain malicious activities because it has no other security protections. As written in the project description, the project did not complete event feed from university websites and social network integration. This sounds like an excuse, but I really can't allocate more time to these two requirements.

This project may be the hardest project I have done besides Senior Design. This is not only because I have no knowledge of php, css language before, but also because I am a solo worker. It took me a period of time to learn how to use Wamp with php and css to create a complete web application. In addition, I also need to use the database knowledge learned in this class to complete the back-end development. Fortunately, I found some very helpful database management projects. From their design ideas to the code, I have adopted many excellent designs to help me complete this assignment.

There are many problems encountered during the development process, most of them are caused by the lack of proficiency in development tools and languages. I can only overcome them with time and practice. Through this project, I have more experience in using css and php language to develop web pages, and I have a deeper understanding of JavaScript's Ajax and MySQL. I really like using Wamp to develop database applications because it brings together the tools needed. Overall, as a beginner with no

experience in web app and database development, I have a better understanding of website/database and user management. If I have another chance to do this project, I will plan to train related tools and languages earlier and expect to use the saved time to complete the two unsatisfied requirements.