Term Project Group Report

# Application: College Event Website

## COP 4710 Spring 2017

## Group 18

### Jonathon Towne

### Luis Casanova

### Alex Phillips

Table of Contents

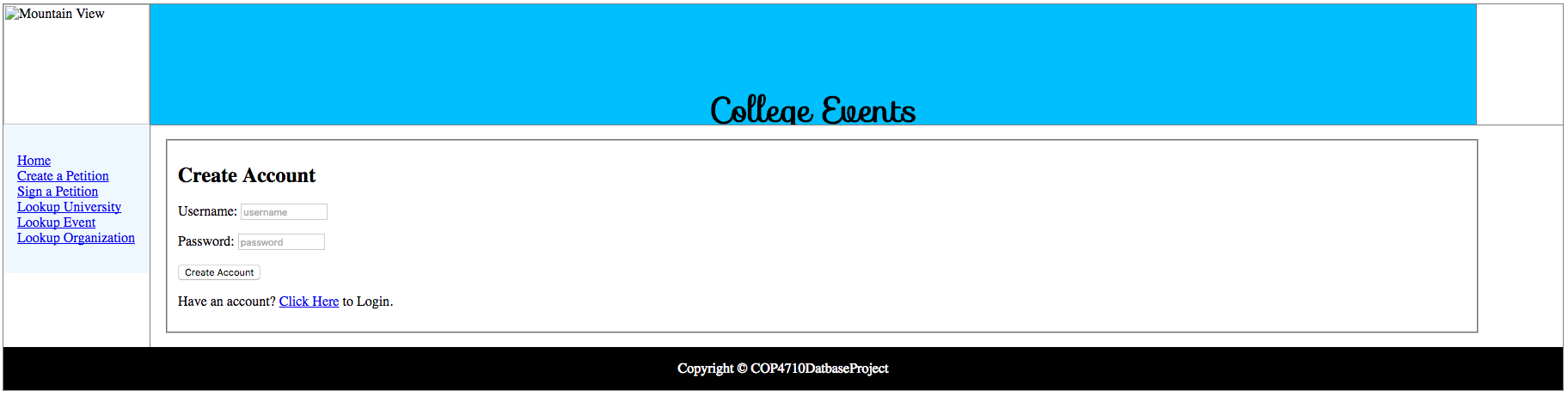
1. Project Description
2. GUI
3. ER-Model
4. Relational Data Model
5. SQL
6. Software installation
7. Conclusion

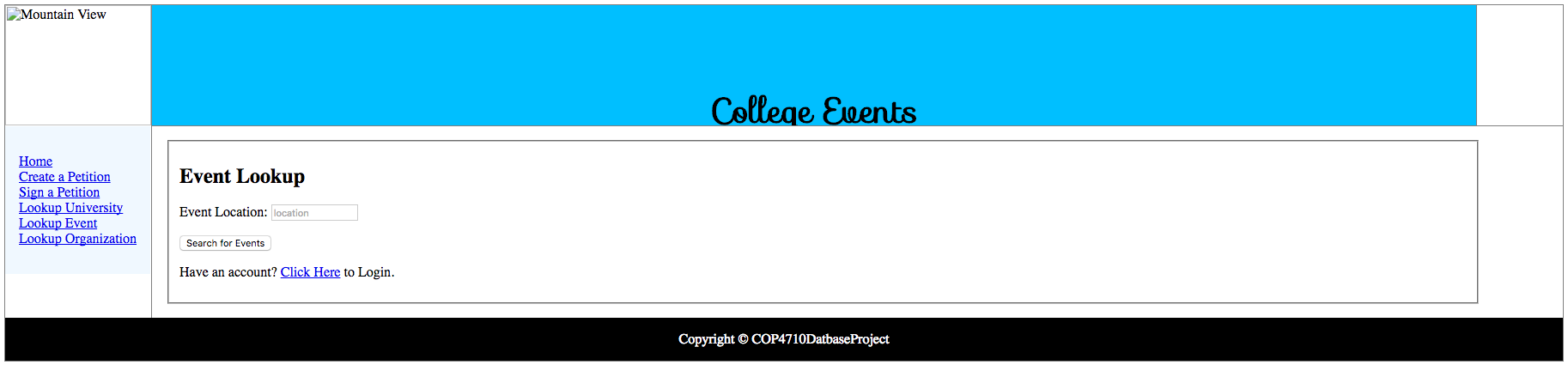
#### Project Description

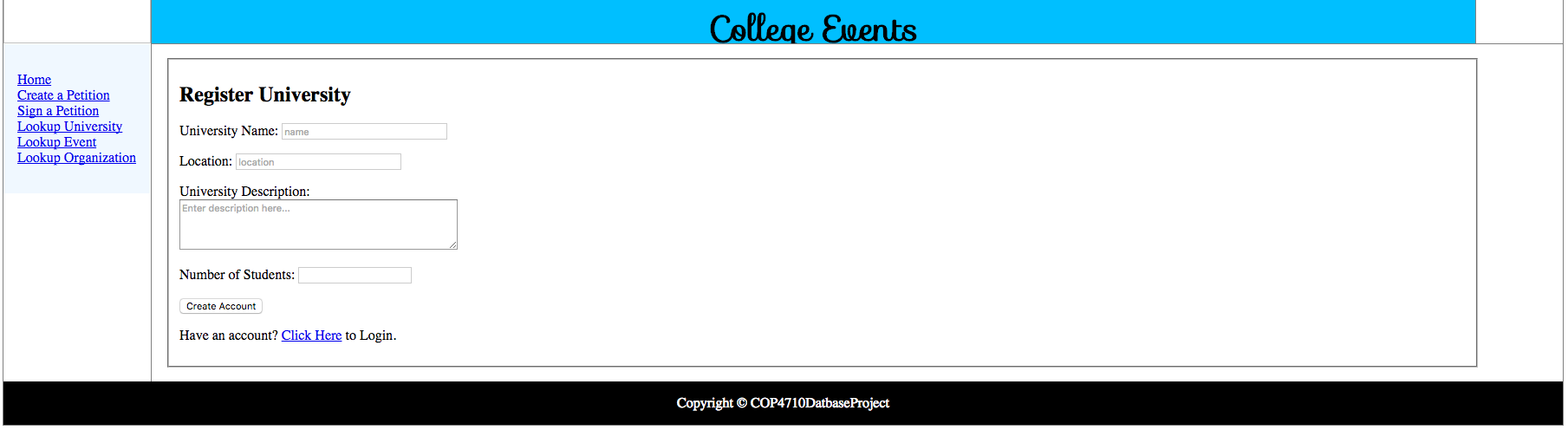
Our project empowers student organizations to publish event for students of select universities. The website also allows students to browse these events that hosted at their university. The students can become admins by creating petitions for new student organizations. The petitions become RSOs after five unique signatures are applied to the petition. The RSO admin create events for their organizations. Events are also pulled from university event feeds.

#### GUI

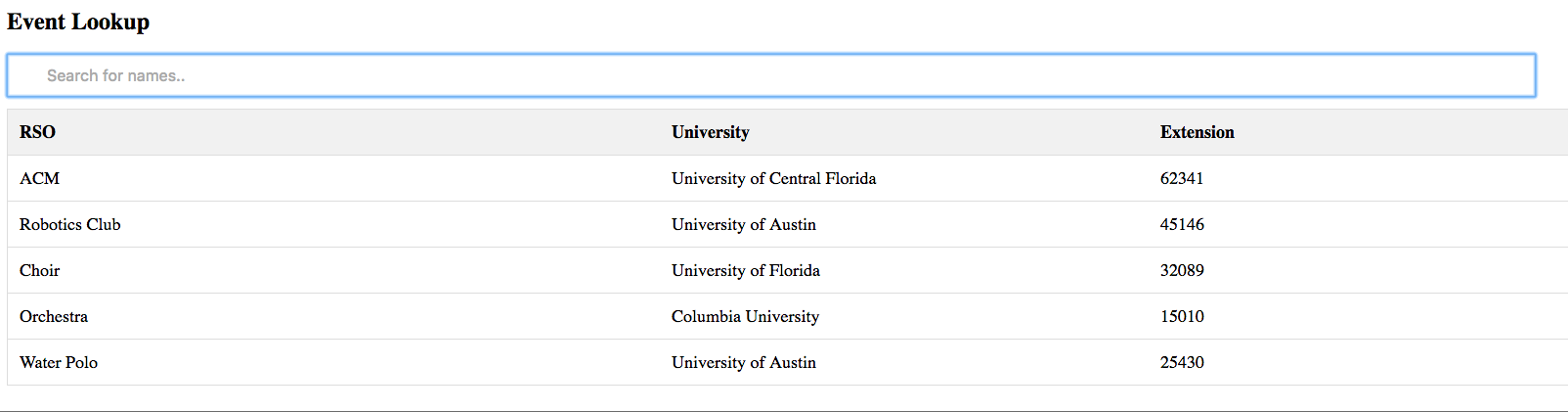
Development of the website was made on a WAMP server. We used windows machine, apache, MySQL, PHP. For the website we wrote it in HTML, CSS and JavaScript.











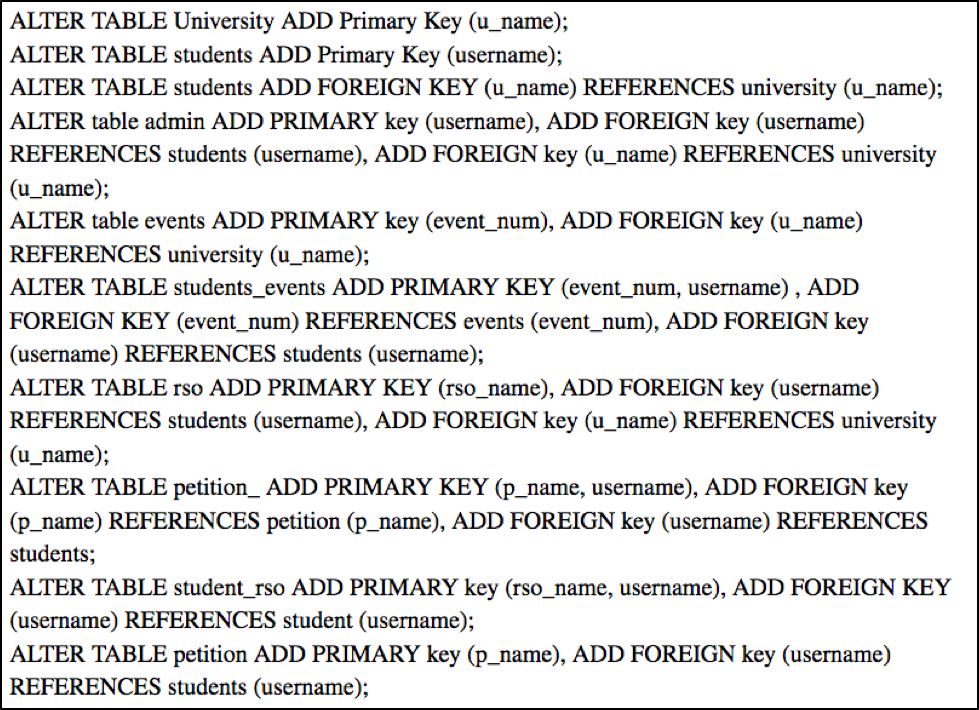
#### ER-Model

constraints captured in the ER-model and other general constraints to be enforced by other means, such as assertions, triggers…



#### Relational Data Model





#### SQL

‘CHECK,’ ‘CREATE TRIGGER,’

• Populating tables with sample data: two universities, 20 students, 3 RSOs, five events, three comments

o SQL statement to insert a new RSO (part of the processing of the ‘Create RSO’ form), show results

o SQL statement to insert a new student to an existing RSO (part of the processing of the ‘Join RSO’ form), show results

o SQL statement to insert a new event (part of the processing of the ‘Create Event’ form), show results

o SQL statement to insert/update a (new) comment (part of the processing of the ‘Create/Add/Modify Comment’ form), show results

o Several SQL queries to display events—public, private, and RSO-- (part of the processing of the ‘View Event’ request by a user with a specific role), show results

o SQL statements of interest (optional), e.g., advanced SQL queries

#### Software installation

Install WAMP server.

Open phpMyAdmin on WAMP server.

Username is root and password is kept blank.

Under the tab import choose database.sql in File to import and click go.

The database will be created.

Copy and paste all website files to C:\wamp64\www folder.

one folder (directory) contains source code and required libraries to run the app, and installation instructions. A ‘\*.sql’ file containing SQL statements to set up the environment (database name, user, password, etc.) and create the database (CREATE TABLE, but no sample data needed, i.e., no INSERT statements) is to be included here.

#### Conclusion

o Database performance: query response time, suggested indexes

o Desired features/functionalities: security (login), event feed from university websites (e.g., XML feeds from http://events.ucf.edu/), social network integration (Facebook, Twitter, etc.)

o Problems encountered, things that have been learned from the project, more things/skills needed to master to build a more advanced database app