

EventCentral

Desmond Vehar
Kyle Wong
Cristian Gonzalez
Brian Lee

What:

EventCentral is website where student organizations can promote their events and students can search for events and RSVP to them. This application is extremely useful as there is currently no centralized listing of events nor a way to search for events.

App Overview:

There are a few different use cases of this application:

- A user will want to view the events they have RSVP'd to
- A user will want to search for new events
- A user will want to search for new student organizations
- A user, who is a student organization leader, will want to edit their student org page
- A user, who is a student organization leader, will want to edit an event their student org is hosting
- A user will want to post a comment on an event page
- A user will want to view a student organization's page to get more information
- A user will want to view an event's page to get more information

Models:

Our model makes heavy use of foreign keys to reduce redundancy.

The User Table is just the basic web2py auth user table. It allows us to require user login and to associate table data with the id of the user in the user table.

The RSVP Table stores the RSVP of a user to an event. This requires storing the event_id, user_id, and a boolean for determining RSVP yes or maybe.

The Events Table stores the information for an event. This includes the event name, a description, the student_org_id of the host, the location, the start and end times, and we also store when the event was created.

The Student Org Table stores the information for a student organization. This includes the date the organization joined EventCentral, their name, their acronym, a description, a meeting location, meeting times, a website, and an email address. Many of these fields are optional or

loosely validated. This is to give the administrators of the student organizations as much flexibility as possible because student organizations are all run differently. We tried to identify useful information users would want to learn about a student organization and make these fields so that we could easily display them in our views. This improves the effectiveness of the application for both the students and the student organizations using the application.

The Pictures Table stores pictures and an integer representing the event or student org id the picture is associated with. There is also a boolean to indicate which it is for, student_orgs or events.

The Comments Table holds important information for comments for events like the comment body, who posted the comment and when. There are also separate tables for comments to pictures and reply comments.

The Tag Table contains all the tags. A tag is just text that can be searched on.

Student Org Tags Table contains a reference to the tag table and a reference to the student org table. It allows student orgs to have tags associated with them.

Event Tags contains a reference to the tag table and a reference to the events table. It allows events to have tag associated with them.

The Admin Pool Table indicates which organizations a user has admin rights to, which allows them to change the events and the main page of the organization.

The Notifications Table is used by the application to associate notifications with users so they can be displayed and deleted.

Features and How They Are Implemented:

- Event: shows basic information about the event like when and where the event is, who's hosting the event, event 'tags', and description of the event. Also allows users to post comments, view pictures, and the user's RSVP status. Admins are shown extra features like the ability to delete posts, add and remove tags, and change event information. Adding tags uses jQuery and AJAX to check if the tag exists and if it does, just make an event_tags entry to attach the tag to the event. Otherwise, it makes a new tag and then makes the appropriate event_tags entry. Changing RSVP information also uses jQuery and AJAX to check if the RSVP entry exists, and if it does update or delete it. Otherwise, it makes a new RSVP entry.
 - Event Picture: Shows a full version of the posted picture. Allows users to comment on the picture and return to the event of the picture.
 - http://127.0.0.1:8000/EventCentral/default/add_event_for_org/#
 - http://127.0.0.1:8000/EventCentral/default/add_student_org_picture/#
 - http://127.0.0.1:8000/EventCentral/default/view_event/#
 - http://127.0.0.1:8000/EventCentral/default/event_edit/#

- http://127.0.0.1:8000/EventCentral/default/delete_event/#
- Search: Allows the user to search for events and student organizations by name, acronym and corresponding tags. The user will be provided a list of links to relevant events and student organizations based on the input of the search bar. Pressing 'enter' will clear the form and the results.
 - <http://127.0.0.1:8000/EventCentral/default/search>
 - http://127.0.0.1:8000/EventCentral/default/view_all
- Student Org: shows various information about the student org including: the acronym, date joined, leaders, website, email, meeting place, meeting times, and student org description. The student org can also upload a picture to assist the user in understanding the purpose of the student org. Admins are able to create tags, edit information, and delete the student org as well as edit the picture of the student org.
 - http://127.0.0.1:8000/EventCentral/default/add_student_org
 - http://127.0.0.1:8000/EventCentral/default/view_student_org/#
 - http://127.0.0.1:8000/EventCentral/default/edit_student_org/#
 - http://127.0.0.1:8000/EventCentral/default/delete_student_org/#
- RSVP: shows the events the user is RSVP'd to this is done by querying based on the user's id and the contents of the RSVP table.
 - <http://127.0.0.1:8000/EventCentral/default/rsvp>
- Org Admin: allows users to create new student orgs, create new events for their student orgs, and manage their student orgs.
 - http://127.0.0.1:8000/EventCentral/default/org_admin