

Django Web-App: Event RSVP System

Engineering Robust Server Software

Homework 2

For this assignment you will be writing a web-app in Django. This web-app will let users manage RSVPs for events (such as weddings, parties, etc). In particular, your system should allow three roles:

Event Owner When a user creates an event, he/she becomes the owner of that event. The owner of an event should be able to add other user's as owners, setup/add/remove the information that guests RSVP with, invite users to the event, add vendors, configure which information each vendor can see, and view the RSVP responses. For example, in a wedding, the bride and groom would be the event owners (one would create the event, then add the other).

Event Vendor A vendor can view (only—not modify) specific information about the event, as configured by the owner. The vendor should also be able to indicate that they have taken their final counts, making that information editable by guests. For example, the caterer might be granted permission to see the count of each type of entree ordered by the guests. After the caterer gets the final entree counts, they can mark that as uneditable, and guests can no longer change their entrees.

Event Guest A guest should be able to RSVP to an event, and answer any questions. A guest should be able to change their answers until the relevant vendor marks that selection uneditable. For example, a guest might enter their entree selection, cake selection, and any food allergies. A guest may also be able to add a “+1” (or plus more than one), as specified by the event owner.

Note that your system should support *multiple* events, and the same user MAY hold different roles in each event. For example, a user may be an owner of 2 events, a vendor for 3 others, and a guest at 5 still different events.

You should support the following functionality:

Create Account A user should be able to create an account if they do not have one.

Login/Logout A user with an account should be able to login and logout.

Event Selection If a logged-in user is part of multiple events, she should be able to select which event she wants to perform actions on. If a logged in user belongs to only one event, you MAY display your event-selection mechanism with one event, or MAY omit it (not show it).

Event Creation A logged-in user should be able to create an event. Creating an event should allow the owner to specify the name of the event, the date/time of the event, and the set of RSVP questions they wish guests to answer. At a minimum, the questions

should allow (a) specification of whether a +1 (or +more) is permissible, (b) free text responses and (c) multiple choice responses. The interface should allow the owner to create an arbitrary number of questions of type (b) and (c).

Question Editing (Owner) The owner should be able to edit the questions once they are created. If the owner changes questions, guests who have already RSVPed should be notified by email that they may wish to update their answers (or answer new questions, etc). You MAY only send these emails in cases where guests would want to update answers (*e.g.*, not send them if a question is removed. Not send them if a choice is removed which was not the choice they selected. etc).

Add Users (Owner) The owner should be able to add other users, either as guests or as vendors.

Set Vendor Permissions (Owner) The owner should be able to select which responses a vendor can see.

Response Viewing/Changing The owner should be able to view every guest's response to all questions. A vendor should be able to view the responses they are permitted to by the owner. A guest should be able to view (and change) only their own responses.

Finalization (Vendor) A vendor should be able to mark a question as finalized, preventing any guest from changing their answer to that question.

Answer/Change Answers(Guest) A guest should be able to answer their own RSVP questions, as well as the RSVP questions for their +1 (or +more). The guest should be able to change these answers until such time as a vendor finalizes them.

Notes:

- Your grade will not be affected by the aesthetics of your website (this is not a UI/UX class). However, your site should be use-able in a reasonable way (your TA should be able to easily figure out how to perform each feature).
- You are, however, encouraged to make a website which you would be proud to show to your friends, family, and prospective employers.
- You MUST use CSS to style at least some aspect of your website for full credit.
- You MAY use any JavaScript libraries that you want.
- Many of these requirements are broad, and open to interpretation. We are not specifying the details of how these should be implemented, or the exact nature of the functionality. For example, whether event selection is a screen displayed after login, or a menu in the corner of every page, or some other completely different mechanism, is totally up to you.
- Remember to keep your “danger log.”