EDSN CALENDAR

Iteration 2

11/21/2016

Bits Please

Chukwuogor, Uka

Klejmont, Tomasz

Mazurski, Christopher

Shahmarov, Eldar

1. System functionality after first iteration.

Starting with the front-end, our system currently displays a Google Calendar with all published events. You can submit an event through a pop up form that will require further approval on the back-end before being published. You can also filter all events with a Google keyword search. Moving on to the back-end portal, we implemented a login system and access levels that block certain aspects of the portal depending on preset permissions. Upon logging in, there's a dashboard with quick navigation links and a notification tooltip that shows if there are any events awaiting approval. The events page displays all events currently stored in the database along with their detailed information and filtering options. Any event requiring action from an admin will be listed in the approval needed section with the option to publish, delete, and send rejection notice. Publishing an event will add it to the Google Calendar and deleting it will do a silent delete, it will remove it from the event view but still keep it in the database. A rejection notice can be sent to the submitter to inform them of why their event wasn't published. The users page displays all current users along with their access levels and permissions. You can also create a new user account with a selected access level.

2. Implemented user stories.

Size	User Story
5	7. B) As an administrator, I want an event marked as ready to be published to actually display in the calendar so that all visitors can see it
2	10. As an administrator, I want an administrative account set up that allows me to create new admin accounts with different roles that have preset permission levels

3. Changes, breakdowns, new user stories.

We decided to change user story #2 which says: as a user, I want a filtering tool to simplify searching for events based on criteria such as interests, location, venue, etc. The Google calendar already has a built-in keyword search for all stored events. Since most of these options can be filtered by using the built-in search, we will implement color coded event categories and allow the user to filter those. We will be putting that back in the product backlog.

We also decided to remove user story #6 which states: as an administrator, I want an event delete option to completely remove an event entry from the database to prevent clogging with spam/unpublishable events. We found it to be superfluous and will stick with a silent delete instead.

4. Lessons learned.

Procrastination was once again an issue. We have learned that commenting code goes a long way when working in a group. It definitely saves a lot of time when trying to figure out what the other person did. Also, deciding on a definitive DB schema early on helps reduce the tweaking necessary to fix any previously implemented feature that depends on it. We will try to start work earlier this time around as it will be our last iteration and we need time to finalize all aspects of the system.

5. Stories yet to be implemented.

2. As a user, I want to be able to filter all events in the calendar based on their category so that I can narrow down my search to things that interest me.

   Size: 3

   Preconditions: All criteria relevant to the search has been selected.

   Postconditions: The calendar only displays events that match the selected criteria.

5. As an administrator, I want a detailed event view that will allow me to modify any event and edit any of its attributes

   Size: 3

   Preconditions: Administrator level permissions. All fields that need to be modified have had their data or selection replaced with a new one.

   Postconditions: All edited fields for the event in question have been updated in the database to reflect the changes. Event displays the updated information in the calendar (if previously published).

8. As an administrator, I want a script generating widget that can produce a script to embed calendars on different websites.

   Size: 8

   Preconditions: Administrator level permissions.

   Postconditions: A generated script is displayed and ready to be copied.

9. As an administrator, I want to be able to customize the script generated by the widget by filtering different options and/or adding a custom header and footer.

Size: 8
Preconditions: Administrator level permissions. All filter criteria selected. Header and footer customization provided.

Postconditions: A generated script for a customized calendar is displayed and ready to be copied.

6. Next iteration

Size	User Story
3	2. As a user, I want to be able to filter all events in the calendar based on their category so that I can narrow down my search to things that interest me.
3	5. As an administrator, I want a detailed event view that will allow me to modify any event and edit any of its attributes
8	8. As an administrator, I want a script generating widget that can produce a script to embed calendars on different websites.
8	9. As an administrator, I want to be able to customize the script generated by the widget by filtering different options and/or adding a custom header and footer.

Total product backlog size after additions: 22
Approximate size of each iteration: 14
Total size of third iteration: 22

At the end of this third and final iteration the system should be fully functional with all previously stated user stories implemented. In addition to the current functionalities of the system, this iteration will add a script generating widget that will allow others to embed calendars on their own websites. The script will have the option to be customized with a header and footer and different filters. Each event on the events page will have a detailed view where an admin with required permissions will be able to modify its attributes and save them. The front-end will gain a category filter that will allow any user to filter the calendar according to their interests.