Project Topic:

Event Management for social service website using Ruby on Rails: BTOEventView

Short description on project work:

The application is intended for Bhutan Toilet Organization (BTO), a non-profitable organization located in Bhutan, which provides clean toilet facilities during public events organized. Currently BTO are using manual paper-based system in managing and facilitating the events and activities organized by them and their clubs who contributing to achieve mission of BTO. By having the Web-based Event Management System (BTOEventView), it will let the authorize user to create and update the activities or events and the users and public will be more alert and aware with the existence of the events or activities held in BTO.

All social Web applications have users, content, and user-to-content map. Following figure will explain functional requirement.

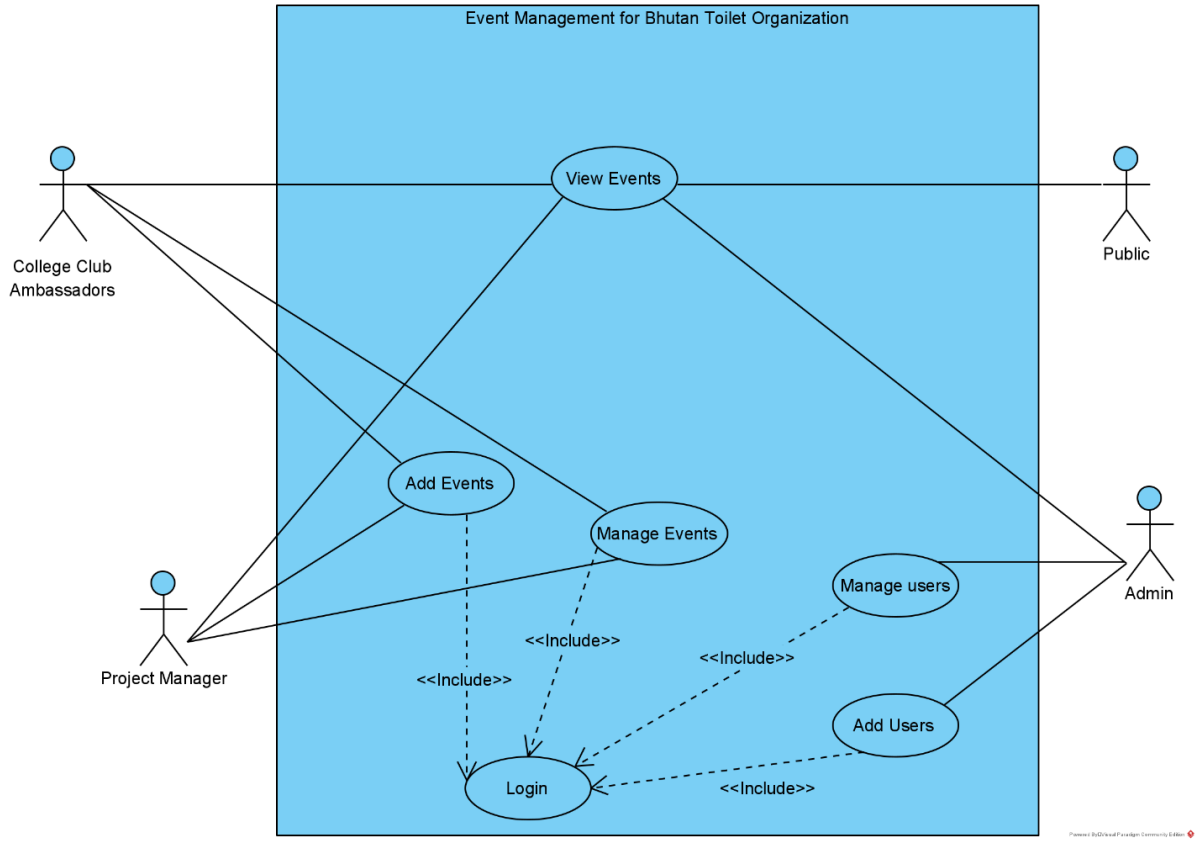


Figure 1. Use Case diagram

As a General Public will be able to:

- view the ongoing events
- view the past events

Project Manager and College Club Ambassadors will be able to:

- add events
- manage events (update and delete)
- update user profile

Admin will be able to:

- register user into system
- perform all the action
- add users and add permission
- can disable and enable users.

Method used for developing application:

1. Multi-page applications (MPAs) and Single-page applications (SPAs).

For MPAs we will focus on the software architecture and technology used on the server side.

For SPAs we look at the architectures of both the frontend and backend systems and their interactions.

2. Rapid Application Development (RAD) / Prototyping

Main advantage is development application has resulted in a less rejection when the application is placed into production since developing is done with end user.

Security concerns:

For security reason and to have secure application, we have design and implemented as if system was under constant attack.

Therefore, we maintain the state over a series of response/request interactions

- hiding state information in documents delivered to the user,
- storing information about user sessions on the server and
- responding to user events client-side, without starting a new HTTP request/response cycle.

Avoid SQL Injection

SQL injection mainly works when attacker can insert his/her query which would be SQL keywords that are directly executed in raw SQL query the backend.

Then attacker can access numerous data and update his/her privileges. To avoid this, the input taken from users is not directly used in the SQL query. We would rather use parameterized queries and dynamic attribute-based finders.

Example:

User.where("name = '#{params[:name]}'") # SQL Injection!

=>

User.where(["name = ?", "#{params[:name]}"]) # No SQL Injection

Avoid XSS attacks

XSS attacks are Cross-Site Scripting attacks that occur when an attacker can send malicious code, usually in the form of a browser side script, to a different end user.

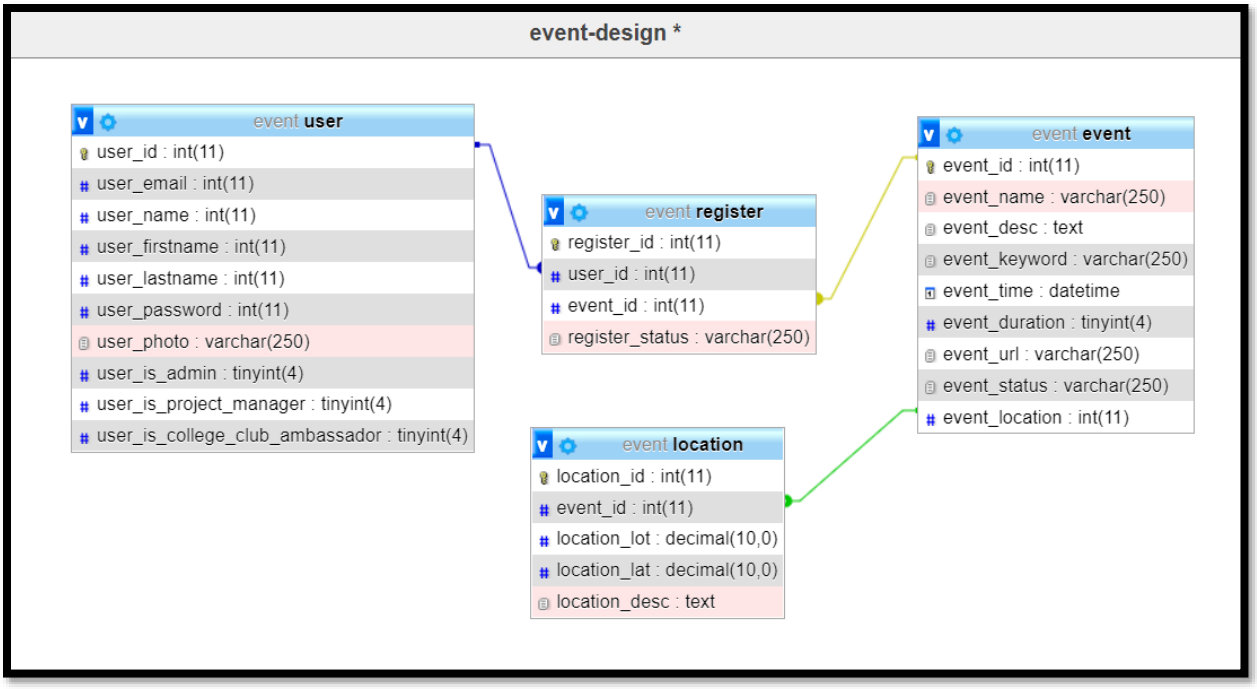
Rails has a built-in XSS protection mechanism which automatically HTML escapes all the data being transferred from Rails to HTML.

HTML escaping substitutes HTML entities such as '<' and '>' with '& lt' and '& gt' so that the scripts "<script>" "</script>" will be escaped. Hence, whatever malicious code the attacker may post to the application will be HTML escaped and not get executed.

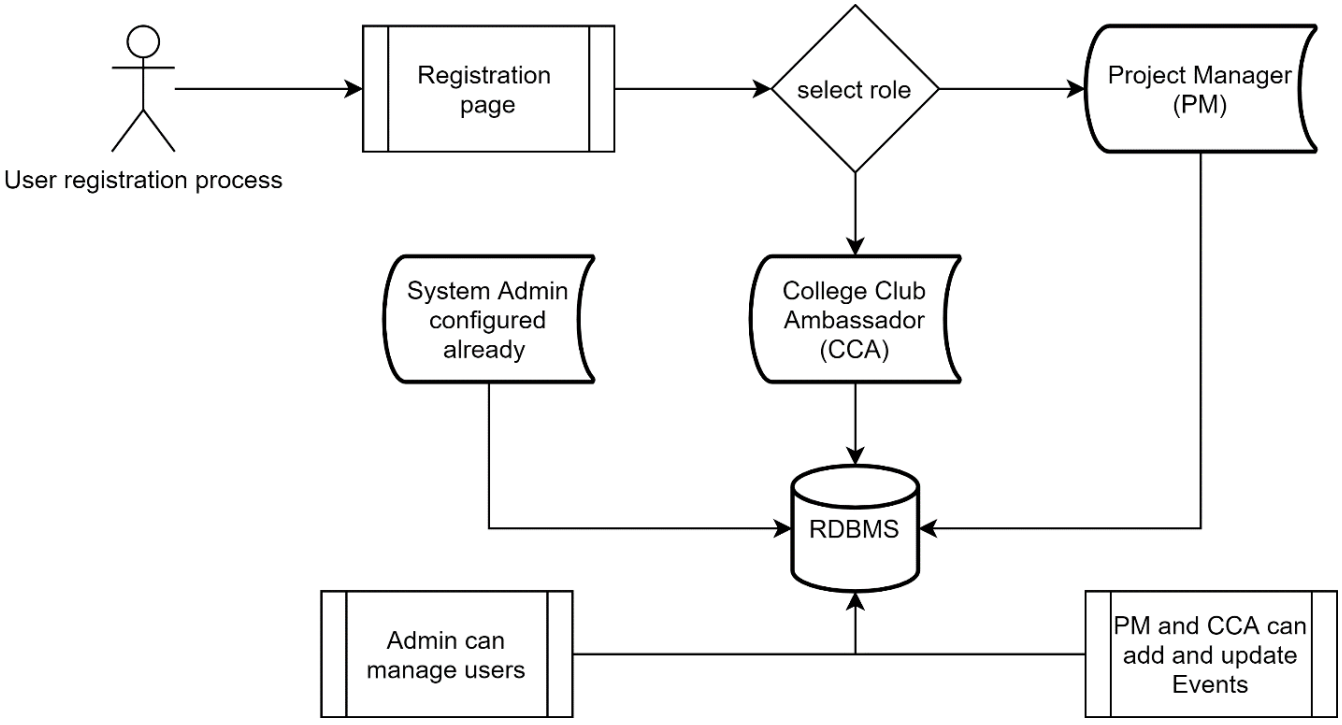
Define your user data model.

Referential integrity is the relationships between different models/tables. Rails active record provides dynamic finder methods that are simple to use and fast to execute.

Database consists of 4 tables shown in figure. User table represents the users of BTOEventView. It maintains 3 different roles i.e., admin, project manager and college club ambassador. It has other info first_name, middle_name and last_name. It also has is_XXX which is used to ban/unban user account. Event has event name, description, time, duration. It has a foreign key that is user id from the user table. Events has a many-to-many relationship with User table through Register table. That is, a user has many events, and a event has many users. Register table is a join table for user and events. It joins them using uid (users) and event_id (Event). Location table is used to store the location description.



User registration and management page flow:



SSL enabled for the complete application to prevent password sniffing and make session cookies are HttpOnly:

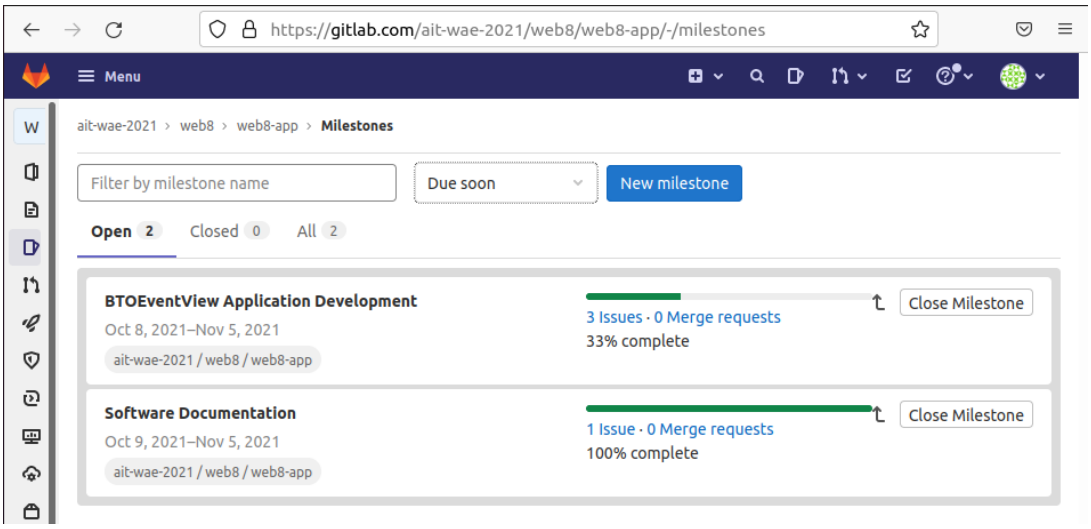
- User registration and login are implemented using 'devise' gem in rails. We have further email confirmation when a new user wants to register.
- User can use forgot password to get a reset token to his/her mail.
- We added SSL for the website and forwarded http requests permanently to https. Hence, all the interactions will be authenticated.

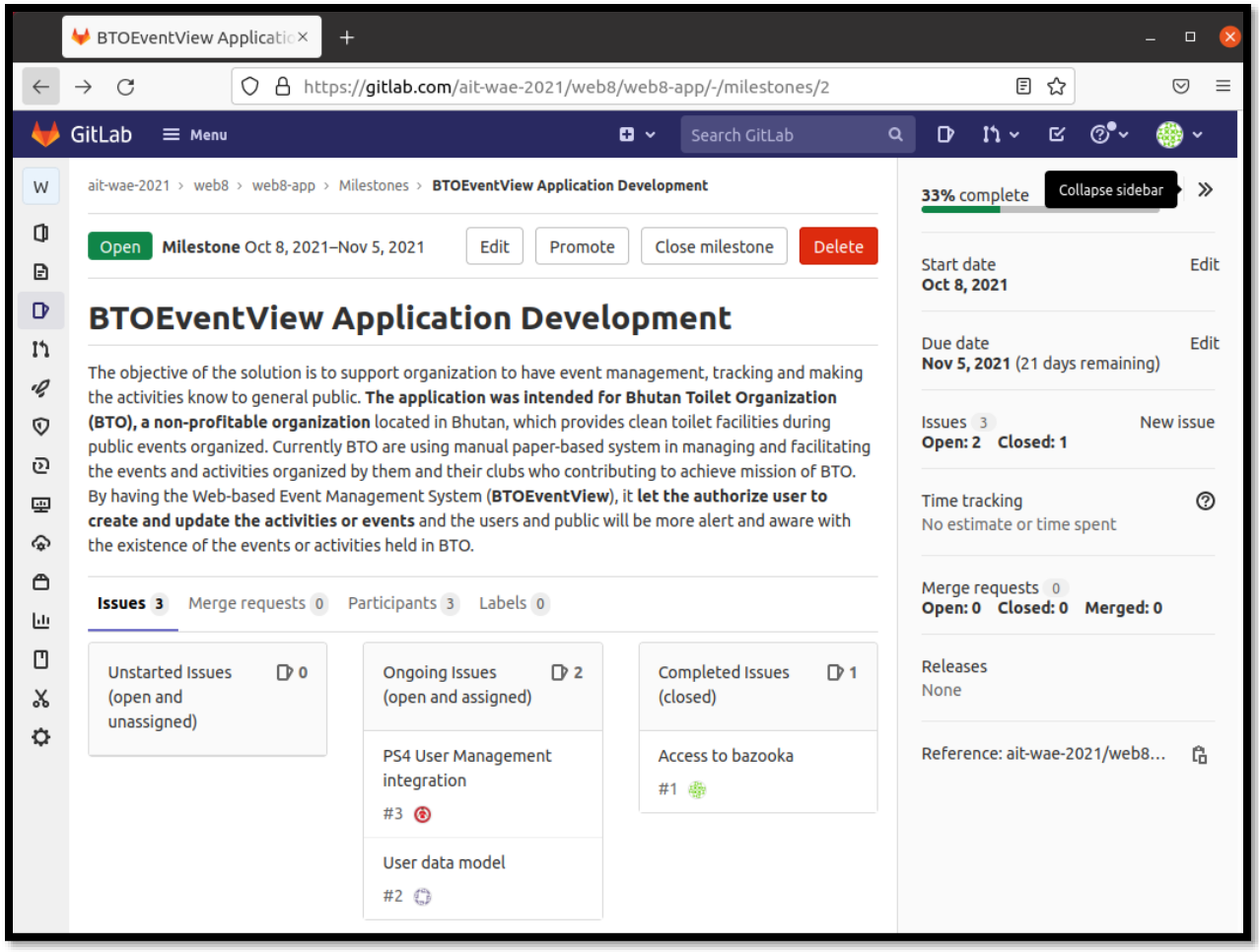
Finally, some significant suggestions for making testing job to be effective for the development of application could be – Test Early - Test Often - Test Automatically. Thus, we can work on following:

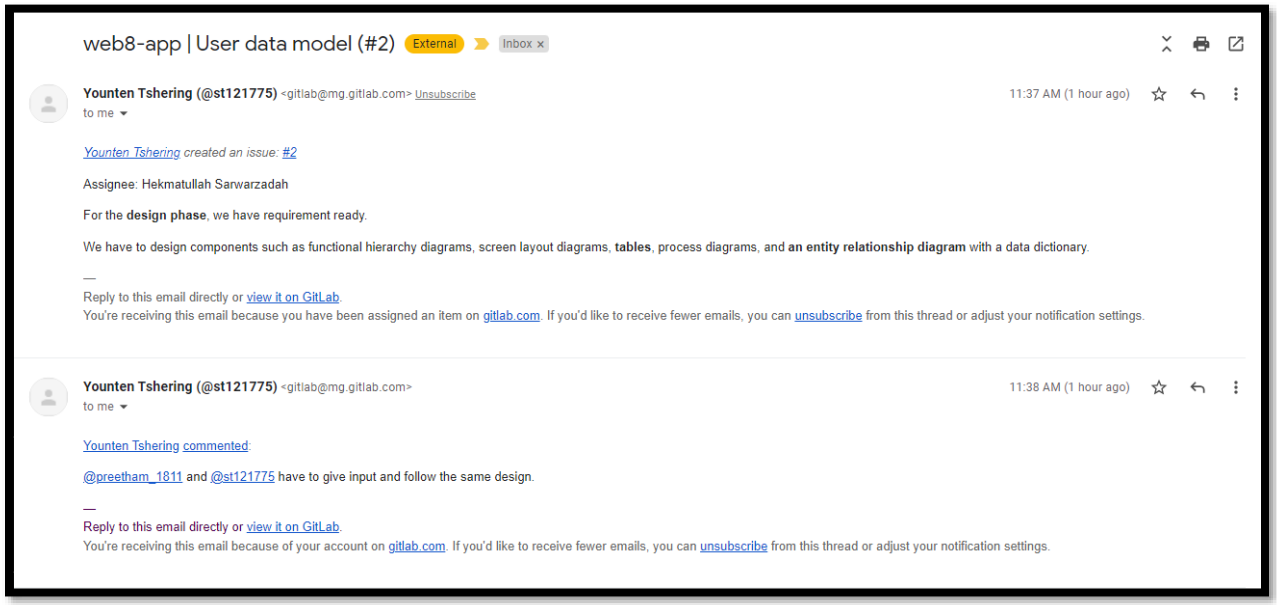
- Get a production build up and running on a test server early in the development and give stakeholders access to the server.
- Create an automated regression test suite and keep adding to it as defects are found.
- Use a continuous integration tool such as Jenkins or GitLab CI to automatically download the latest build, run the test suite, and generate quality metric reports.
- Add appropriate tests as you code and ensure the regression test suite passes before every commit.

Use your issue tracker and CI server to their full potential:

The screenshot of our Issue Tracker in our milestones is given below where we have created the recent issues in our milestone we have been working on as well the screenshot of the email notification that is sent to our registered email addresses with the recent issue created.




BTOEventView Applicatio
https://gitlab.com/ait-wae-2021/web8/web8-app/-/milestones/2
GitLab
Menu
Search GitLab
ait-wae-2021 › web8 › web8-app › Milestones › BTOEventView Application Development
33% complete
Collapse sidebar
Open
Milestone Oct 8, 2021–Nov 5, 2021
Edit
Promote
Close milestone
Delete
BTOEventView Application Development
The objective of the solution is to support organization to have event management, tracking and making the activities know to general public. The application was intended for Bhutan Toilet Organization (BTO), a non-profitable organization located in Bhutan, which provides clean toilet facilities during public events organized. Currently BTO are using manual paper-based system in managing and facilitating the events and activities organized by them and their clubs who contributing to achieve mission of BTO. By having the Web-based Event Management System (BTOEventView), it let the authorize user to create and update the activities or events and the users and public will be more alert and aware with the existence of the events or activities held in BTO.
Issues 3
Merge requests 0
Participants 3
Labels 0
Unstarted Issues (open and unassigned) 0
Ongoing Issues (open and assigned) 2
PS4 User Management integration #3
User data model #2
Completed Issues (closed) 1
Access to bazooka #1
Start date Oct 8, 2021
Edit
Due date Nov 5, 2021 (21 days remaining)
Edit
Issues 3
New issue
Open: 2 Closed: 1
Time tracking
No estimate or time spent
Merge requests 0
Open: 0 Closed: 0 Merged: 0
Releases
None
Reference: ait-wae-2021/web8...


web8-app | User data model (#2)
External
Inbox ×
Younten Tshering (@st121775) <gitlab@mg.gitlab.com> Unsubscribe
to me
11:37 AM (1 hour ago)
Younten Tshering created an issue: #2
Assignee: Hekmatullah Sarwarzadah
For the design phase, we have requirement ready.
We have to design components such as functional hierarchy diagrams, screen layout diagrams, tables, process diagrams, and an entity relationship diagram with a data dictionary.
—
Reply to this email directly or view it on GitLab.
You're receiving this email because you have been assigned an item on gitlab.com. If you'd like to receive fewer emails, you can unsubscribe from this thread or adjust your notification settings.
Younten Tshering (@st121775) <gitlab@mg.gitlab.com>
to me
11:38 AM (1 hour ago)
Younten Tshering commented:
@preetham_1811 and @st121775 have to give input and follow the same design.
—
Reply to this email directly or view it on GitLab.
You're receiving this email because of your account on gitlab.com. If you'd like to receive fewer emails, you can unsubscribe from this thread or adjust your notification settings.