HackOverflow: The IT Communication Hub

W. A. Anuradha Sanjeewa Wickramarachchi

Department of Computer Science and Engineering
University of Moratuwa
Moratuwa, Sri Lanka
anuradhawick@gmail.com

Abstract— **The HackOverflow is a web site designed to address the problem of communication of IT related events to the IT community. The survey conducted using few universities and IT students indicated that a majority of members were unable to attend the IT related events such as Hackathons and Meetups because of the late or no communication of events. The HackOverflow system was implemented with capability to maintain a mailing list and a forum. This was implemented in order to improve the interaction among IT field professionals and students. The system has a forum that people can bring in discussion on various matters related to hackathons and other events. Also the system contains a discussion forum that can bring in discussions on a desired hot topic. The system provides capabilities for registered users to post event, meetups or hackathons. The users can include all the necessary information related to the particular event type. Also the posts are emailed to the registered subscribers so that they can get to know about the posted events as and when they are posted. The posted events contain discussion section attached too, so that the interested community can carry out discussion to obtain more information**

Keywords—hackathons; website; communication; IT; events; forum; discussion; posting;

I. INTRODUCTION

The IT industry often holds competitions and events to gather young innovators and provides them the opportunity to come up with innovative ideas. The industry is so competitive that the participants for such events are taken in first come first served basis or through evaluation of hackathon ideas. Because of this inherent nature of the competitions, the IT community tends to focus more on the coding aspect rather than focusing on sharing event's news and encouraging colleagues to take part in events. Thus a significant part of the community fails to take part in competitions because they were unaware of the existence of such events or competitions. There are numerous websites that publish IT related information as blogs or newspapers but they are not truly focused on the event management aspect. Even though such news is published they are scattered among numerous sites and blogs. Thus the users may find it difficult and time consuming with their busy schedules to go through those web sites and get themselves registered.

There are number of IT related professionals and students who miss their opportunity to get involved in hackathons and other events. This has a negative impact on the entire IT industry because opportunity for the innovative ideas to be pitched among investors and other stakeholders is lost. Thus it is clear that a specialized platform is needed so that all the interested parties can get registered and subscribe for news about events.

Sri Lanka as a country has been serving the IT industry in large scale over few decades. Most of the IT industry focuses on service rather than product manufacturing. This is because there is a lack in innovation and investment in new IT related products. This is mainly because the pool of ideas that exists within the island has no means of getting pitched among investors. Thus in both industry and innovation perspective it is clear that it is important to communicate IT related events and competitions so that interested parties with innovative ideas can get involved. The system is expected to gather the young innovative community to a single platform and communicate opportunities so that they can innovate and come up with business ideas to serve the island's IT industry.

The implemented system will provide the IT community with all the necessary functionality in order to effectively communicate the events and hackathons. The system also provides a forum capability so that the community can start discussion and produce innovative ideas and organizer events. The events are attached with forums so that discussions related to posts can be made so that feedback and other communications related to events can be performed effectively from a single place.

II. LITERATURE REVIEW

In the present context there are quite a large number of technology publication web sites and news papers. The news related to different events are across different web resources making it difficult and time consuming task. Also most of the sites are inactive for a considerable amount of time during a year.

Most of such web resources focus in a narrow scope where some sites such as Angel Hack [1] only contains news related to the International Hackathon series called Angel Hack. It is rarely that news being posted in such a site is relevant for Sri Lankan developer community in the competitive scope.

Web sites such as ReadMe.LK [2] publish technical news related to the Sri Lankan IT sector. Although the website publishes news about hackathons and other events yet it is difficult to find the relevant article among thousands of other articles.

Code4Good [3] is another Sri Lankan web site that publish news related to the hackathon Code4Good. This is only focused on this particular hackathon. Therefore, a user should stay in alert with the web site so that he/she will be able to get registered for the hackathon before the registration date is passed.

Yarl IT Hub [4] is another major event organizing society. They have been organizing the largest hackathon in Sri Lanka held at Jaffna for a quite a long time. But this web site only focuses on the hackathon they organize which is the Yarl Geek Challenge.

It is clear from the study of the current context of the IT events that, even though there exist many web sites and online magazines to publish events and news, none of those can be reached searching for events at any time. This is because their scopes are narrow and either focuses only on a given event or the scope is too high that a user finds it difficult to seek news related to hackathons and other events.

The HackOverflow web site was designed and implemented in order to fill the gap that has raised between the IT community and the event organizing community. This is achieved through eliminating the complexity and time that is required to sweep through news web sites to look for competitions. Also the system provides capability for users to comment on events and carry on a discussion using their social media accounts. This will enable the users to give feedback and organizers to provide clearance on queries made by users about events. This is important since no other web site or magazine provides this capability for organizers to interact with the participants before the competition or the event is held.

The HackOverflow system also provides capability of users to post in the forum implemented. This helps the community to carry on discussion about events and various IT related activities. This will provide Innovative IT youth to post their ideas, look for new opportunities or to find investors in order to capitalize innovative ideas.

Thus the HackOverflow fills the gap that exist in current IT publications web sites and hackathons event publishing sites. The system also provides capability for users to get register and post event so that others can get registered.

III. System Models

A. System Requirement

The HackOverflow's main functional requirements are posting hackathons, meetups and other events. Also the system contains a forum so that registered members can post queries and start discussions. All the users can comment or attach picture comments for the posts and other events that are posted in the HackOverflow web site. The users should sign up in the system using their google plus accounts in order to use the posting events and posting in forum features. This is to ensure the events and forum posts are non repudiated by any means. The registered users also can subscribe to mailing list so that they can get updates about events that takes place as and when they are posted. If by any change the users wish to unsubscribe from the mailing list, they may do so by removing the subscriptions from the profile management page. The subscribed users will be provided with emails where a link to the actual post is included so that they can get registered to the event by using the google form attached to the post. The system also implements security which is vital by using multi-level security. The system contains 3 levels of users as visitors, registered users and administrators. Administrator users can perform all the tasks of the other two levels. Also the administrator can remove any post, view system statistics and add or remove administrators (There is a predefined superadmin) for the system. This was implemented in order to guarantee that the content being posted are moderated so that the quality of the system will be maintained.

The system has also implemented some useful non-functional activities such as maintaining event tags so that the users can search for events and get details or continue discussions on those events. Also the users are provided with capability to contact the administrator and request information. The forums are included with "Like" functionality which helps other to get an idea about the event rather than reading through the comments in order to get an idea about the events that are posted.

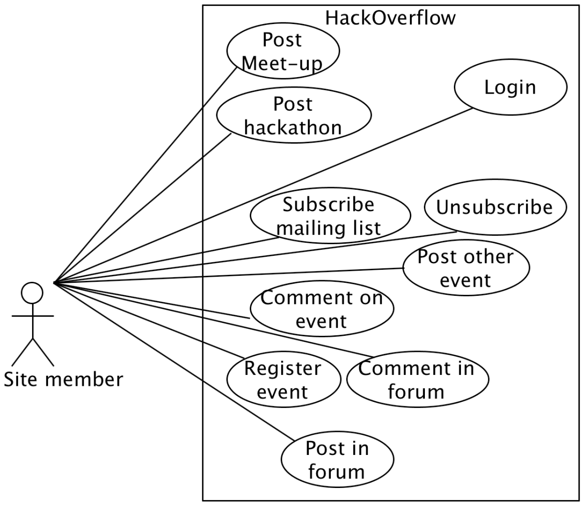


Fig. 1. Use case diagram for major use cases

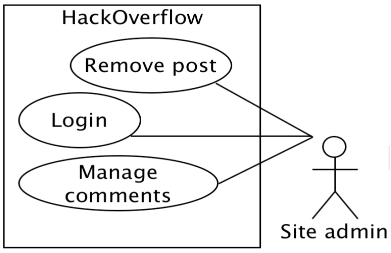


Fig. 2. Use case diagram for admin use cases

Figure 1, represents the main use case diagram which demonstrate the use cases performed by any user that visits the system. These functions can be performed by registering in the system and some of the functions such as viewing posts, commenting and contacting the admin can be done without being registered in the system through users' social media profiles as well. Figure 2 demonstrate the major use cases performed by an administrator.

B. System Design

1) Logical view

The system employs the Model View Controller (MVC) architecture in order to implement HackOverflow web site. This architecture was selected since the web site runs on a backend database and MVC architecture provides easy and maintainable implementation of the system with good separation of concerns in to the three components. As per the MVC architecture the system has Views which are rendered to the user by passing requests through the controllers where the inputs get sanitized and authenticated and the data are obtained from the Model.

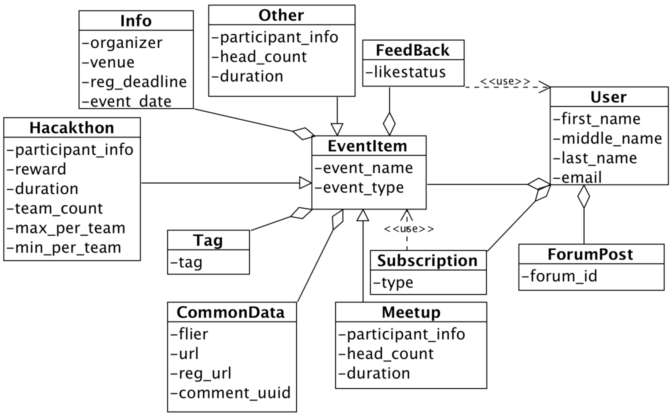


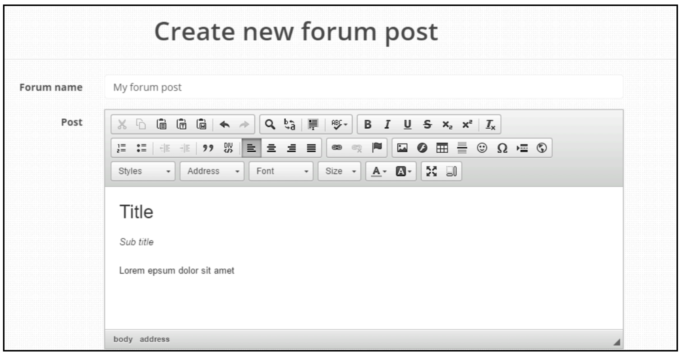
Fig. 3. Entity relationship diagram

These classes contain models in the database in order to represent persistent objects to be reused. These objects are accessed from the database and are mapped to the Instances of the classes represented to be used in the Web Application.

2) Process view

In the HackOverflow system, users are provided with a short and simple flow in order to post in the forum as shown in Figure 4. The users are provided with a rich text editor where users can insert images, format text and post.

The Figure 5 demonstrates the process of viewing and posting a forum posts. The users are required to login to the system in order to post in the forum. The user is required to click on the Forum button and select New Post.


Create new forum post

Forum name: My forum post

Post:

Title

Sub title

Lorem epsum dolor sit amet

Fig. 4. Use case diagram for major use cases

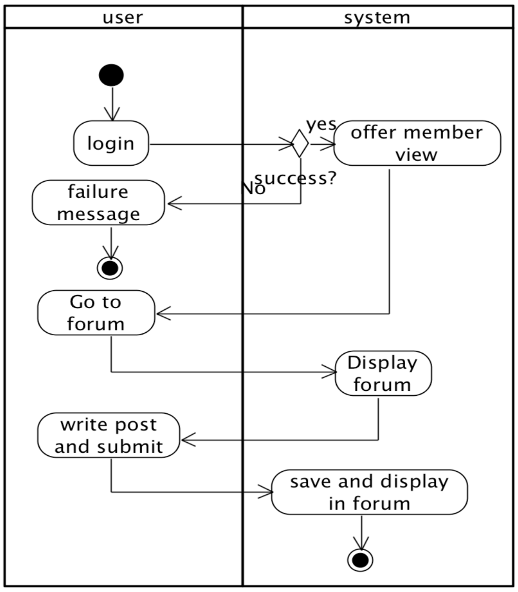


Fig. 5. Activity diagram for posting in forum

The user will be provided with a form to be filled and submitted. The submitted forms will be attached with the comment thread and will be displayed for other users.

3) Database design

The HackOverflow system employs a relational database that uses MySQL in development and Postgres SQL in deployment. There are 16 tables that are related to each other in order to operate the Web Application.

The object level inheritance of events is maintained at the database too. This produce a certain level of redundancy of attributes across several database tables yet this does not make the data redundant. Thus the inheritance is maintained in order to cope with any changes with minimum changes to the system.

The database has been normalized up to the second normal form in order to retain lower levels of data redundancy at the same time to manage the querying efficiency. The system anticipates increase in size due to the posts being posted with the expected usage, thus the data are saved in basic data types and no binary data are saved in the database in order to retain indexing properties and to reduce the weight of raw queries. The database is indexed with primary keys that are used in order to generate views where data are fetched from the database in order to be rendered to the user.

IV. System implementation

A. Implementation procedure

The system was implemented using the Laravel PHP framework which is built on top of Symfony framework. The system uses the PHP Composer as the dependency management framework. For testing of the system PHP unit was utilized. For the implementation of the database MySQL was used and for the deployment of the system Postgres SQL was used. This is because most of the industrial applications are supported using the Postgres SQL which is quite similar to MySQL. These frameworks were used to make sure the development can be focused better rather than having to implement the entire infrastructure. Also these technologies provide better coding with better standardization in adherence with MVC architecture. The frameworks also provide job handling and job pooling in order to make better performing systems in performance wise as well.

There were several tools used in order to implement and test the system. The main IDE used was the IntelliJ Ultimate educational version. This enabled the execution of Command Line codes, refactoring code, generating PHP Doc comments and analyzing codes for false dependencies. MySQL workbench was used to run test queries and develop the database schema. This tool provides a graphical user interface in order to generate and drop schemas. Also the tables can be created and altered with ease making the development more productive. Google chrome was used for testing and debugging the front end program which is the web site. This browser enables throttling and other valuable function for testing the system under different network conditions. Also it can be used to log JavaScript errors so that they can be easily noted and fixed.

Disqus forum management system [] was used in order to maintain the forum. This system enables the system to add discussion components so that the users can connect with their social media accounts and take part in discussion. This was used in order to provide better user experience to users and save time of having to register separately in the system for commenting. Also the Google OAuth was used to sign in and signup users. This was used because having to enter a username and password to register consumes time and people are not willing to do so with a new web site because of the security issues. Thus the Google OAuth login provides easy registration in the Hack Overflow system.

For the designing of web pages Twitter Bootstrap was used along with “FontAwesome” font package. “Animate.CSS” was used to provide web animations. Relevant documentation [5] was referred during the development in order to access the CSS classes and functions. JQuery JavaScript framework was used for rapid development of Ajax functionality.

The Laravel 5.2 documentation [6] was referred in the development in order to obtain the API functionality that was required. Also MySQL documentation and SQL [7] queries were referred in order to tweak some Eloquent ORM methods for better performance of database queries.

With all the above tools and frameworks, the system was implemented following a waterfall development methodology where the initial plan was defined with major milestones of the system as per the project schedule represented in the SRS. Within each phases of the waterfall method, agile development was used to develop rapidly with lesser documentation.

Along with the development unit testing was carried out. The testing included three major areas as UI testing, Functionality tests and security testing. These were carried out using the PHP Unit testing framework. Although the testing plan included Selenium IDE it was later discovered the testing procedures of the Selenium Driver could be performed using the latest version of the PHP Unit. Web pages and performance was tested using the Chrome Developer console and the Data base efficiency and performance was tested using MySQL workbench.

B. Material

The data used for testing and the implementation was generated using dummy variables. Most of the data that was entered into the system were random data. These data were injected to the databases using database seeders available in the Laravel framework. These data were in adherence with the required data types in order to make the tests more effective and render the views in proper manner.

Since the system contains Google Login it was unable to inject random user data. Thus user data was obtained from the Developers Google Account and was injected to the database so that the initial login to the system could be done as an admin and perform basic tasks.

C. Algorithms

There were few algorithms used in the system in order to provide better user experience and optimum resource utilization. These were some of major algorithms that were employed by HackOverflow.

1) Long pooling

This was used by the forum in order to display the live likes count for a given forum post. This was implemented in order to users to get and idea about the likes that a particular post without having to reload the web pages. In this method a continuous stream of requests is made to the server and gets latest data and updates the user interfaces.

```
function getDataFromServer()
{
getRequest('/forum/likes/<forum_id>',callback(){
      onSuccess:s function(data)
      {
            updateUI();
            delay(1000); // wait until the network settles
            getDataFromServer(); // on a new thread
      }
      onError: return;
      });
}
```

Fig. 6. Long pooling algorithm

The Pseudo Code in Figure 6 represents the algorithm used for long pooling. In this function, the function is called and data is taken from the server. On success the same function is called recursively after updating the UI. This method is called in a new thread in order to avoid stack overflows. The initial function's thread will die eventually making the new thread the only live thread. The used payloads in the request are minimal so that the network usage is minimal and friendly on metered connections.

2) Queueing

The system implements the mailing list functionality where uses can subscribe for events and get notifications. The emails are sent over the SMTP mail protocol. For this purpose, the server should communicate with the mail service provider from the back end. This consumes few seconds and the sending of en email takes some time. Thus a huge number of users in scale of thousands would take few minutes before all the mails are sent. The user shall not be kept alive until the request is served after sending all the mails. Thus the mailing functionality shall be taken away for a separate process.

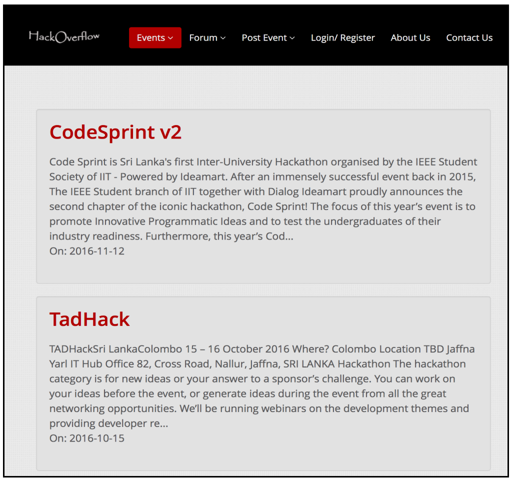
```
function sendMail(mailing_list)
{
     queue = JobHandler.getQueue(); // calls the singleton
and obtain the queue
     queue.addJob(New MailerJob(mailing_list)); // pass the
job to the queue
     queue.dispatch(); // release the queue
}

/* In a separate thread */

while(queue.size() = 0) // keep the process alive
     while(queue.size() > 0) // if there are items in the
queue
     {
          foreach(item Q in queue)
          {
               Q.performTask(); // send the email
               Queue.dequeue(Q);
          }
     }
}
```

Fig. 7. Mailer queue

The above code brings in mailing tasks to a separate thread's queue and send mails each at a time leisurely. This ensures that the heavy task will not block the users from interacting with the system. This algorithm will also make sure that the system will not get overloaded with a large number of concurrent mailing activities overloading the server bandwidth.

D. *Main interfaces*

The following are some major user interfaces implemented in the HackOverflow web application.



CodeSprint v2

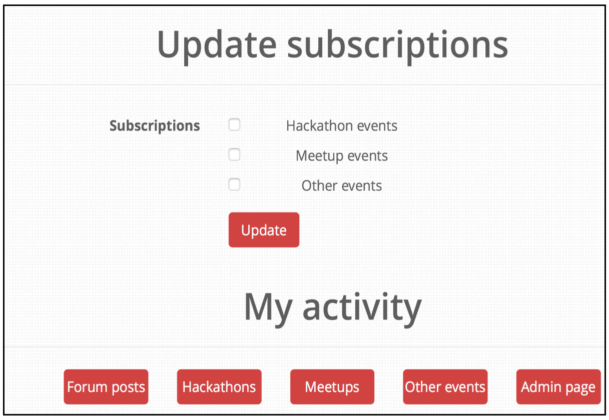
Code Sprint is Sri Lanka's first Inter-University Hackathon organised by the IEEE Student Society of IIT - Powered by Ideamart. After an immensely successful event back in 2015, The IEEE Student branch of IIT together with Dialog Ideamart proudly announces the second chapter of the iconic hackathon, Code Sprint! The focus of this year's event is to promote Innovative Programmatic Ideas and to test the undergraduates of their industry readiness. Furthermore, this year's Cod...
On: 2016-11-12

TadHack

TADHackSri LankaColombo 15 – 16 October 2016 Where? Colombo Location TBD Jaffna Yarl IT Hub Office 82, Cross Road, Nallur, Jaffna, SRI LANKA Hackathon The hackathon category is for new ideas or your answer to a sponsor's challenge. You can work on your ideas before the event, or generate ideas during the event from all the great networking opportunities. We'll be running webinars on the development themes and providing developer re...
On: 2016-10-15

Fig. 8. Hackathon page

Figure 9 demonstrate the profile page of an admin user. If the user is an ordinary user, the "Admin page" will not be available. From this UI the user can subscribe or unsubscribe from the mailing list.


Update subscriptions

Subscriptions □ Hackathon events
□ Meetup events
□ Other events

My activity

Fig. 9. User profile page

V. System Testing and Analysis

A. *Functional testing*

For all the functions in the system unit test cases were written. The PHP unit framework was used testing individual functionality including the database functions which queried data from the database using the Eloquent ORM. The development was proceeded once each of these unit tests were passed without any failure. The testing for errors were done and made sure that relevant errors are dealt with proper error pages for the user.

B. *Security testing*

This was performed in order to make sure the system maintains Integrity and Authenticity. Confidentiality of the system was not considered much since the system does not contain significant amount of confidential data. All the authentication processes were done using the Google User verification thus no password or confidential information was saved in the system. Integrity was guaranteed by testing on user privileges. Authenticity was implemented using middleware that promised authenticity checks before they were passed into the controllers. These were tested using automated request and form submissions.

C. UI testing

UI testing was also covered using PHP Unit. Although it was planned to use Selenium for the task it was later discovered that PHP Unit later version can perform UI testing. Thus form submissions, button clicks, routes and the links were verified to function properly by using PHP Unit test framework.

D. Performance testing

As demonstrated in the test plan this was performed by using the MySQL workbench (Database testing) and Chrome (Front end testing). The performance testing was carried out to measure the performance over different network conditions, different devices and for the level of caching at the front end.

VI. Conclusion and Future Work

The implemented system satisfies all the functional requirements identified through the System Requirements Specification. Thus the system is expected to serve the required task of improving the communication of IT related events for the Youth and Young innovators. The implemented system is ready to be deployed in a cloud server so that the system will keep up and running with an acceptable uptime.

The implemented system will provide equal opportunity for students and other professionals to get in touch with hackathons and other events as and when they are posted. This will enable a larger number of people to get to know about hackathons and other events. This will eliminate the issue of people not sharing hackathon news due to the highly competitive nature of the industry. Also the HackOverflow will aggregate all the event news to a single web site saving the time of the users and reducing the complexity of finding the right events for registration.

There are few future enhancements and work identified that will improve the usability and the overall system at large. The system should have a professional mail delivery system such as MailChimp [8] for a scalable mailing solution.

Also the system would serve better if the social media integration is possible. This would enable the system to post automatically to social media page so that the users can get redirects to the web site for event registrations. This will at the same time improve the reach to the IT community making the system more popular.

At the moment there is no revenue model for the Web Application. It is vital to have some sort of revenue generation methodology in order to ensure the sustainability of the application and to survive the expenditure of hosting and mailing. Thus an approach such as advertising shall be adhered into.

References

[1] AngelHack, "AngelHack," [Online]. Available: http://angelhack.com/events/.

[2] ReadMe, "ReadMe," [Online]. Available: http://readme.lk.

[3] Code4Good, "Code for Good," [Online]. Available: http://www.code4good.lk.

[4] Yarl IT Hub, [Online]. Available: http://www.yarlithub.org/ygc/.

[5] w3schools, "w3schools," [Online]. Available: http://www.w3schools.com/bootstrap/default.asp.

[6] Laravel 5, "Laravel," [Online]. Available: https://laravel.com/.

[7] w3schools, "w3schools," [Online]. Available: http://www.w3schools.com/sql/default.asp.

[8] Mailchimp, "mailchimp," [Online]. Available: http://mailchimp.com.