

Berkman Klein Center For Internet And Society - Question Tool

Google Summer of Code 2019

Name: Sanket Singh

Country: India

Email: singhsanket143@gmail.com

Preferred Communication Method: Email, Video Conferencing and Call

Project: Question Tool

About Me

I am currently in my pre final year of my Bachelors in Technology Degree from Maharaja Agrasen Institute Of Technology (MAIT) in Computer Science and Engineering (2016 - 2020). I have been tinkering with code since school days. I am a Web-Developer with specific interests in Ruby and Javascript based platforms. I do have a firm grip on **Ruby, Javascript, HTML5, CSS3, C++ and Python**. I am a self taught developer and an enthusiastic individual. I also regularly participate in a lot of hackathons and have won a few of them. I love working on interesting and challenging problems and have spent many of my nights up hacking on some cool projects. I started my programming journey from the 1st year of my college, and gradually developed a keen interest towards problem solving and product development.

Experience:

- Full Stack Web Developer Intern At [Kakcho Fashions](#)-
 - Created the complete backend in **Ruby On Rails** and **Node.JS** and maintained the **AWS EC2** and **S3** instance as well as troubleshooted **99%** of the server issues.
 - Identified software and DB architecture based on project requirements, performing version control and refining product visual design of 4 screens.
 - Scaled the backend with horizontal scaling to increase efficiency by **60%**.

- Prepared the complete real-time chat structure using **Socket.IO**.
- Front-End Engineering Intern At [Fable](#)
 - Developed user interface module for Creative Selector module in **Angular 5** & gathered requirements, develop and test module to meet requirements.
 - Revamped use cases, API & user stories of the complete web app.
- Instructor - [CODING BLOCKS](#)
 - Mentored over 1000+ students in **Algo++** & **C++ Launchpad** courses. I taught advanced algorithms and data structures in various offline and online class.
- Mentor - [Learn It Girl](#)
 - Mentoring an engineering student of IIIT Bangalore for her project on E-Commerce Web Application in **Ruby On Rails** and **Node.JS**.
- Problem Setter - [Talview](#)
 - Reviewing and Developing educational content for coding students including the development of a centralized questionnaire database.
 - Developed high-quality, creative programming challenges involving various Computer Science concepts.
- Deep Learning Research Intern - [Indian Space Research Organization \(ISRO\)](#)

Availability:

I reviewed the whole GSOC timeline and I would be able to devote **approximately 40-50 hours** every week to GSOC. During 15th to 27th May I will have my Final END-Semester examinations, thus amount of time will be around 2-3 hour per day for that time.

I will be devoting my full time to this project and I do not have any other Summer internship or Job. I have no obligations after May, and would be devoting all my time to GSOC.

I will be reporting my work over daily scrum mails on the mailing list. ***I would strive to be regular, and sincere with my scrum and daily updates as I understand that selection in this project will require a serious commitment and 100% devotion from my side.***

Expectation From GSOC and Berkman Klein Centre:

Gsoc is a great opportunity to enhance my programming skills and develop some insights towards production level project development. It is a amazing chance for students like me to work on a high level project that really matters. I will get a chance to work with world class developers of Berkman Klein Centre which will include lots of learning and fun. I am interested in the project QUESTION TOOL of Berkman because this project will help me improve my intellect by giving me a deeper insights in how to work on a free, open-source and isomorphic web framework i.e. Meteor.JS. This project will make me become a better programmer and developer as i will be working on a production level web app. According to me open source development can be life-changing that will inspire exponential growth, a lot of learning and constant development. It's results could be remarkable, profound and enduring. GSOC program have the capacity to transform and connect individual groups, organizations and communities. It will be very fulfilling to know that i've directly contributed to a organization like Berkman's growth and development. Thus, I see participating this year in GSOC with Berkman as a great opportunity to make use of my good technical, leadership and communication skills to develop a next generation project and experience the guidance of mentors from Berkman. Web App Development has always been my plus points and I have received great recognition for the same. Berkman's Question Tool promotes a collaborative discussion environment and can be scaled to a higher level because of the scope of features it provides. I will be privileged to receive mentorship from Berkman Klein Centre so that i can improve myself as a developer as well as learn some important skills and tips from the honourable mentors of Berkman Klein Centre.

Important Links:

Link To LinkedIn Profile: The following is the link to my LinkedIn Profile.

Link To Github Profile: The following is the link to my Github Profile.

Link To CodeShare Project: The following project is a web app which features live code-sharing and chatting feature based on Node.JS and Express.JS.

Link To Lyrics Generator Project: The following project is a machine learning model that generates lyrics based on the dataset provided using Markov Chain Process.

Link To Portfolio Blog App: The following project is a Blog Web App based on Ruby and Ruby On Rails.

Specifications And Introduction

The current state of **Question Tool** client side app is in functional state, but there are both, a large number of issue remaining, and an equally large number of required features to be implemented. A user can post an instance and others can only interact by either reacting on the instance (Like the instance), or by posting an answer to that instance. But, Question Tool currently has very less user interaction because of no notifications or emails about what is happening with any instance idea or with their replies. Question tool's user experience is also not upto the marks as per now, which can also be the possible reason for some users less collaboration. The project also lacks some features that will be a lot helpful to the users.

Project Goals

During this GSOC term, I would deliver the following:

1. **Editing Feature:** Add the Edit functionality to an instance's features by the owner after it's creation like addition/removal of moderators, edit the description or heading, edit max word count etc.
2. **Reset Password:** Implement password reset functionality for the users with role of moderator or admin as well as streamline the account creation of moderators.
3. **Real Time Remote Control:** Implement the real time remote control of the instance from any other device so that the admins/moderators can highlight questions being projected in real time using Web Socket specifically using the **Socket.IO** library.
4. **QR Code:** Implement the **QR code** functionality for accessing the Question tool instances from any device at any time in a feasible manner.
5. **UI and Other Features:** Improvements on UI/UX of the application.
 - a. Revamping the UI to give a better, easy and feasible interface to the web app.
 - b. Implementing **popups**, **loaders** and **notifications** wherever necessary.
 - c. Using Media queries to implement perfect **mobile responsive application**.
 - d. Make the UI appealing and also refactor the code for modularity.
 - e. Improvement of **icons** and **fonts** wherever necessary.
6. **Documentation:** So as to make the app build simple for non-experienced users, I will be providing a good documentation along with a FAQ. I will also try to modify the

existing code to be as user friendly as possible by using relatable variable names and commenting the code as much as possible.

Some More Proposed Features:

1. **Notifications Emails:** Users will receive notifications emails, whenever there occur some specific event like receiving reply or upvote on their instance..
2. **Follow:** To follow(or say 'watch' in github) any instance such that user would get notified whenever a new reply is posted on that instance.
3. **Markdown:** Markdown support in the text editor and also the markdown support when the instance is rendered.
4. **Omniauth:** Implementation of single click authentication using github and google login APIs.

The above proposed features were some basic features that I found missing in the question tool app. And I believe, if we could accommodate these features in the app we can definitely improve Question Tool's user experience and user interface.

Implementation

1. **Reset password:** To Implement password reset functionality to the users with role of moderator or admin we will keep a button along the login form that will get a confirmation popup to send the reset instructions to the registered email. The methods like the ones listed below will be used to reset the password for a user using a token received in email

```
Accounts.resetPassword(Session.get('resetPassword'), password,
function(err) {
  if (err) {
    console.log('Something went wrong.');
```

```
});
```

3. **QR Code:** The implementation of QR code corresponding to every instance can be done using node-qrcode library. The following code is the prototype of the actual implementation of Qr Code functionality.

```
QRCode.toCanvas(document.getElementById('canvas'), 'sample text',  
function (error) {  
    if (error) console.error(error)  
    console.log('success!');  
})
```

The best place to get the QR code displayed will be when the instance is created and redirected to it's show page. We can show that along with the description of the problem before the answers to that instance. It will not be feasible to show QR code on the home screen of the application instead we can show a short link for that page, and in the details page we can show QR Code. I have a previous experience with short links so can also implement that functionality if the mentors allow and there is time left after completing another task. A short link can be attractive and easy to remember. Short links can help user to access the instance from a distance as user camera can't capture a QR code from very large distance. We can also implement a show QR code button in order to popup the QR code whenever we want during screen projection.

4. **Popup-Notifications(UI/UX):** Notifications are the most important feature for user interaction on any social network platform. Currently there is no notification feature in our Question tool. Implementing this feature will surely improve user collaboration on and off Question tool. As there would be generally less number I will implement notification model from scratch. And render them near Question Tool's navbar with either using AJAX and Noty JS Library along with JQuery. The notifications will look attractive and also include features like timed notifications with progress bars, custom notifications and different styles. The following sample code snippet will be modified further to show popup-notifications to the user.

```

$.noty.defaults = {
  layout: 'top',
  theme: 'defaultTheme',
  type: 'alert',
  text: '',
  template: '<div class="noty_message"><span
class="noty_text"></span><div class="noty_close"></div></div>',
  timeout: false,
  progressBar: true,
  animation: {
    open: {height: 'toggle'},
    close: {height: 'toggle'},
    easing: 'swing',
    speed: 500
  },
  closeWith: ['click'],
  callback: {
    onShow: function() {},
    afterShow: function() {},
    onClose: function() {},
    afterClose: function() {},
    onCloseClick: function() {},
  },

  buttons: false
};

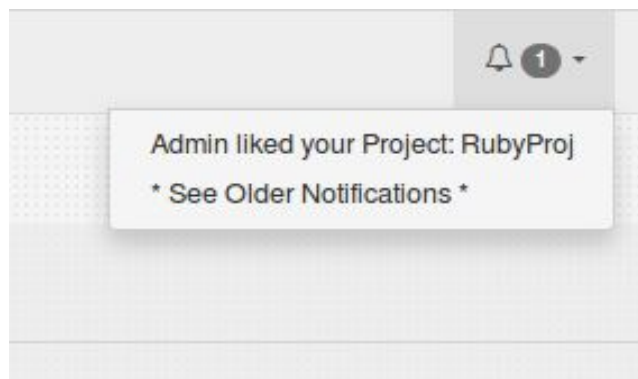
```

Along with the popup the user will get a dedicated notification panel where all the previous notifications will be listed and the unread one's also. The notifications will help to improve the UI of the Question tool application in a drastic way.

I will build Notifications of following types :

Action	Recipients
--------	------------

When someone post an instance	The User Itself
When someone replies on the instance	User of the instance and the people who liked the instance
When someone likes/upvotes an instance/answer	Owner of the instance/answer and the people who liked it.



Notification UI for saved notification in navbar

5. **Real Time Remote Control:** To allow an admin or a moderator to control and highlight questions that are projected we can use Web Sockets and can use Socket.IO library in order to implement this feature.

```
Meteor.startup(() => {
  const server = http.createServer();
  const io = socket_io(server);
  io.on('connection', function(socket) {
    console.log('new socket client');
  });
});
```


Meteor.js can be easily integrated with Socket.IO. On the client side we can use socket-io-client package.

```
const PORT = 8080;
let socket = require('socket.io-client')(`http://localhost:${PORT}`);

socket.on('connect', function() {
  console.log('Client connected');
});
socket.on('disconnect', function() {
  console.log('Client disconnected');
});
```

The following code will be used to emit an event whenever the text in the editor will be highlighted or we can say when the onSelect event will be triggered in the dom.

```
var sendMessage = function (event) {
  var userMessage =
    event.target.value.substring(event.target.selectionStart,
    event.target.selectionEnd);
  socket.emit('chatMessage', {message: userMessage, username:
  username});
  $('#userMessage').val("");
};
```

For multiple admins and moderators from the user end we can also show them the username of the admin or the moderator along with the highlighted text. The functionality can work with multiple admins and moderators. From admin side admin can see all the highlighted text by him or by any other admin/moderator along with the username similar to that of the user but no name will be shown to his own selection as there is no point of showing any admin his/her own name.

- 6. Archive Functionality:** In my views the functionality can be implemented in two ways. Firstly we can put a Download as PDF button in order to let the user download a

complete session of Question tool. The pdf will include the whole conversation and some statistics on how many like were on the instance, how many upvotes were on any answer, how many answers were there for an instance etc. The following code snippet demonstrate the code that can be modified to achieve this.

```
const PDFDocument = require('pdfkit');
const doc = new PDFDocument;
doc.pipe(fs.createWriteStream('output.pdf'));
doc.font('fonts/PalatinoBold.ttf')
  .fontSize(25)
  .text('Some text with an embedded font!', 100, 100);
doc.end();
```

The pdf-kit module will be used for this. The other way around to archive a session can be to store the necessary details in the cache of end user's browser like devdocs.io. But in my view pdf will be the most simplest and efficient way to archive a session. We can try to incorporate the UI as much user friendly as we can so that user can differentiate properly.

- 7. Edit Functionality:** A button will be provided along with the instance in order to update different properties of an instance by the owner. A proper UI will come in order to update that instance. The following code similar will be similar to the original implementation of the feature.

```
Instances.update(this._id, {
  $set: { max_question: 20,
    hidden: !this.hidden
  },
});
```

- 8. Other UI/UX improvements:** There is a high need to revamp and improve the design and UX of the application. We can use some modern frameworks like Bootstrap or Materialize, Semantic Ui in order to quickly change the frontend of the application. The looks of any application should be appealing and engaging. I will change the

design of the application and also make sure that mobile responsiveness of the application is maintained to its fullest. Elements like Loaders will be implemented wherever necessary along with font update. The navbar and the footer of the application will also be updated to give a new fresh look to the application. Also i can work on decreasing the load time of the application and making the front end faster. The following are the sample UI Components which looks similar to the final implementation of the Application.

Getting all sample test cases passed in the bft but getting wrong answer

[Go To Content](#)
[Go To Doubt](#)
[See Messages](#)
[Mark Resolved](#)
[Reopen Doubt](#)
[Post To Wall](#)

I have implemented the adjacency map implementation of graph

Rate your doubt experience

★★★★★

Posted by Saket Singh on BFS Traversal

1 reaction

Shubham Kumar
Doubt: wa
soln: there is a duplicate thread for the same query so I am marking this one as resolved

1 reaction

18:55 21 Dec 2017

Saket Singh
where i can find that thread

18:58 21 Dec 2017

Saket Singh
?

19:1 21 Dec 2017

[Load all comments](#)

Screen to show all responses to a Instance.

Post your queries here

Saket Singh

1 minute(s) ago

Doubt

NullPointerException in remove duplicates java

Couldn't solve remove duplicates in java via HashSet. NullPointerException error. int end = arr.length; Set

- Suresh Upadhayay

NullPointerException in remove duplicates java

Couldn't solve remove duplicates in java via HashSet. NullPointerException error. int end = arr.length; Set<Integer> set = new HashSet<Integer>(); for(int i = 0; i < end; i++){ set.add(arr[i]); } Iterator it = set.iterator(); while(it.hasNext()) { System.out.println(it.next()); }

Sarathak Grover

8:37 25 Apr 2018

Doubt

Binary search (Recursive)

Code works fine on any custom input, but on submission 4 test cases are wrong. Please help

Home Screen to list all the instances bifurcated on the basis of favourites, recently asked and most popular.

- 9. Emails:** We can use the Email module of meteor JS in order to send the mails to the user whenever he/she gets a reply on it's instances. The following code will send the emails to user.

```
Meteor.methods({
  sendEmail(to, from, subject, text) {
    check([to, from, subject, text], [String]);
    this.unblock();
    Email.send({ to, from, subject, text });
  }
});
```

- 10. Follow:** Currently there is no way to follow an instance in Question tool by which instance originator can be notified about the activities going on in the instance they are interested in. Even if the user is not in the instance ownership he/she will be notified through notifications and emails about the instance activities, if he 'follows' the respective instance. I will implement it by creating a new model which will have references of both users and instances, by has_through relationship. By which, every single follower will be getting regular updates. (The dashboard functionality is different from Follow. By following we are adding a social media touch to the Question tool application which will engage the users into the app. Dashboard is more of a analysis and monitoring tool.)

- 11. Markdown Support:** Markdown functionality should be added in the text editor when anyone posts any instance or an answer to it. For the security concerns of Markdown, we can reliably apply the correct XSS filters to remove any dangerous or malicious content. If there's some critical operation you must perform to the data (such as XSS filtering), it should be done *SERVER SIDE* not client side.

- 12. Omniauth:** If we can implement login with google functionality then user onboarding on the application will become super easy and efficient. This can achieved easily using meteor JS. These social logins are now-a-days common in every web app

featured online. I am thinking of including omniauth along with local auth for authentication purpose

```
const onLoginWithGoogle = ({token, isLogIn}) => {
  const {scope} = ServiceConfiguration.configurations.findOne({service:
    'google'}));

  Meteor.loginWithGoogle(
    {requestPermissions: scope, requestOfflineToken: true,
    loginUrlParameters: {hd: '*'}},
    error => {
      if (error) {
        if (error.errorType === 'Accounts.LoginCancelledError') return;
        const message = errors[error.error] || errors.default;
        alert('Login error', message);
      } else if (token && !isLogIn) {
        acceptInviteMethod.call({token});
      }
    }
  );
};
```

Timeline/Project Plan:

GSoC is round about 12 week duration, with about 25 days of Community Bonding Period in Addition.

I will be spending **10% time in fixing the bugs** left out in the current version of the app.

80% time on adding new features to the App.

Remaining **10% time on testing the app, preparing Wiki and FAQ for the template.**

The detailed timeline is linked below:

Legend: Importance and Time Devoted: 

Time Frame	Start Date	End Date	Task	
------------	------------	----------	------	--

Community Bonding	7 th May	26 th May		
	27 th May	29 th May	Requirement Gathering and Documentation	
	29 th May	3 rd June	UI enhancement of the Application - 1	
	3 rd June	5 th June	Implementing Edit any instance functionality	
	5 th June	10 th June	Implementing Reset Password Functionality	
	10 th June	24 th June	Implementing archive an instance session functionality along with initial documentation for the application.	
	24 th June	28 th June	UI enhancement of the Application - 2 including fonts update and an initial testing of the application	
Phase 1 Evaluation	25 th June	28 th June		
	28 th June	12 th July	Implementing QR code functionality along with Send emails on getting an answer on instance functionality	
	12 th July	24 th July	Implementing Partial implementation of Real time remote control functionality including some UI improvements.	
Phase 2 Evaluation	24 th July	26 th July		
	25 th July	5 th August	Remaining implementation of the real time remote control feature	
	5 th August	12 th August	UI enhancement of the Application - 3 UI including necessary loaders implementation	

			including NotyJS implementation for popup notifications and markdown support(If necessary)	
	12 th August	17 th August	Omniauth Implementation And Follow any instance implementation Along with Thorough Testing, Exception Handling and distributing the app among beta testers (if allowed).	
	17 th August	19 th August	Preparing Documentations, Wiki and FAQs and a Webcast on the Final Product.	

I can assure you that if I get selected to work with Berkman Klein Centre this summer, I will definitely try my best to make this project successful and will love to continue working with Berkman's other projects even after the summer.

I would love to mentor in any Educational programs that Berkman Klein Centre is a part of. Further, I would really love to work under such great mentors of Berkman Klein Centre.

Also for some reasons, if I am not selected this year even then I'll try to contribute to this and other projects as much as possible and retry again next year.

Looking forward to working with you.

Thanks And Regards

Sanket Singh