

Google Summer of Code 2019

Rocket.Chat – Service Accounts

Project Proposal Report

Aditya Bhardwaj | bhardwajaditya

Contents:

1. Overview
2. Project Idea
3. Project Implementation
4. Wireframes
5. Project Timeline
6. About Me
7. Future Improvements

1. Overview

The project aims to add the functionality of Service Accounts in the existing Rocket.Chat application. Service accounts will be an upgrade to regular user accounts, they will have a well-defined purpose. Regular users can subscribe to this account to get information regarding the context mentioned in the description of the Service account. By the end of the Google Summer of Code's time period, I will be completing the implementation of following features in the existing Rocket.Chat app.

- Service Account Creation
- Service Account Subscription
- Service Account Management
- Broadcast message feature for service account
- User Interface for Service Account Subscribers
- User Interface for Service Account Owners/Moderators
- Directory search for service accounts

2.Project Idea

Service accounts subscription will allow users to receive all latest updates on the topic specified in the description of service accounts. Only users approved by admin are permitted to create service accounts. Users with the service accounts will be shown the list of service accounts they own or are assigned as moderators and can access them once they log in from their basic account. Each service account will have its list of direct message rooms and a mandatory broadcast room for regular updates. Each reply with service account will be sent by the name of service account rather than the actual user. A directory of discoverable service accounts will be created to list all the existing service accounts in the server, users can subscribe to any number of service accounts of their choice.

Additional features required for this project

- Administration panel settings for service accounts to set number of service accounts a user can own and approve or reject user applications to create a service account.
- Users can send request to admin for approval to create a service account
- Approved users can create a new service account
- New service account creation dialog with mandatory name and description.
- As soon as a service account is created a new broadcast channel will be created with owner set as service account.
- Service accounts owners will have a choice to set discoverability of service accounts, by default it will be set to discoverable. It can be changed during or after the account creation.
- Moderators can be added during service account creation or afterwards by the owner.
- Users can subscribe to any service account.
- As soon as a user subscribes to a service account, a direct service channel between user and service account will be created.
- One click login for service account owners and moderators from the sidebar menu entry listing all the service accounts a user owns.
- A subscription folder will be created to manage all the service accounts subscriptions and messages at a single place.

3. Project Implementation

Permissions and roles

- A new permission can-create-service-account will be added to the existing permission framework.
- Service-account-creator will be in users' scope, service-account-owner and service-account-moderator will be roles in subscription scope.
- By default, admin, service-account-creator roles will be assigned to can-create-service-account permission.
- Can-edit-service-account will be assigned only to admin and service-account-owner.

Administration Panel

- A new setting service account will be added to settings collections
- It will contain maximum number of service accounts a user can own, number of moderators.
- In the service account settings route, a table of users who have applied to be a service account owner will be shown.
- On approval role of service-account-owner will be added to that user roles array in user document.
- Service accounts can be edited by admin just like other user accounts.

Service Account Creation

- New service accounts will be saved as user document in users collections.
- Each new service account will differ from the regular user account by an owner field.
- Only service accounts will have owner field containing information about the user who created the service account.
- Owner will be subscribed to service account with role set as service-account-owner. This subscription will be of new type SA, this won't be visible in sidebar like other subscriptions.
- Discoverability of service account will be set as visible by default and can be changed by the owner or admin either during creation settings or at any later point of time from the profile section.
- Moderator field is optional, but as soon as a new moderator is added that user will be added to the moderator array and moderators are allowed to login to service accounts.
- Adding moderator will also create a subscription document in subscriptions collection of type SA, with role set as service-account-moderator. The service account document will have owner information, moderators will be temporary

so they will be in subscription collection with role as service-account-moderator.

- Along with service account a new broadcast channel having the same name as that of service account will be created and the owner of the broadcast channel will be the service account. Both owner and moderator can use this channel to send broadcast messages.

Service Account Subscription

- Users can subscribe to any number of service accounts of their choice.
- With each subscription a new service account-user direct channel will be created.
- A user can only initiate a direct channel with service account only if the user has subscribed else a message will be displayed to user to subscribe to the service account.
- Number of direct channels involving service account will give the number of subscribers of the service account.
- All broadcast from service account will be available in this direct channel.

Service Account Management

- Admin and users with role service-account-owners will be allowed to edit service accounts.
- Owners or admins can change profile information of the service accounts.
- Owners can add or remove service-account-moderators.
- Moderators will only be allowed to reply to user messages and send new broadcast messages.
- Service accounts will only be edited from owner main user account rather than from service account so that moderators can't change the account settings.
- Each incoming message to service account will be through a live chat queue in order to avoid two people replying to the same message.

Broadcast Message for service accounts

- A broadcast channel for each service account will be created where only the service account will be part of the channel.
- Each message sent to that broadcast channel will be propagated to all the subscribed users' direct channel.
- Any message deleted from this channel will be removed from all other direct channels.

- The above features will be implemented by a new broadcast message feature where message sent to the broadcast room will be sent to all the users in the broadcast list.
- The broadcast channel created will only have service account as it's member, so any message sent from there will be duplicated in message collection for each subscribed user and service account direct channel.

Directory search for service account

- Users can get the list of all the discoverable service accounts from the existing Rocket.Chat directory.
- User accounts with owner will be displayed under new service account heading rest of the existing user accounts will be shown under the users heading.
- Clicking on the service account a dialog box will appear asking user whether they want to subscribe to the service account.
- Clicking subscribe button will subscribe user to the service account and a new direct channel between service account and user will be created for future communications.

Login mechanism for service accounts

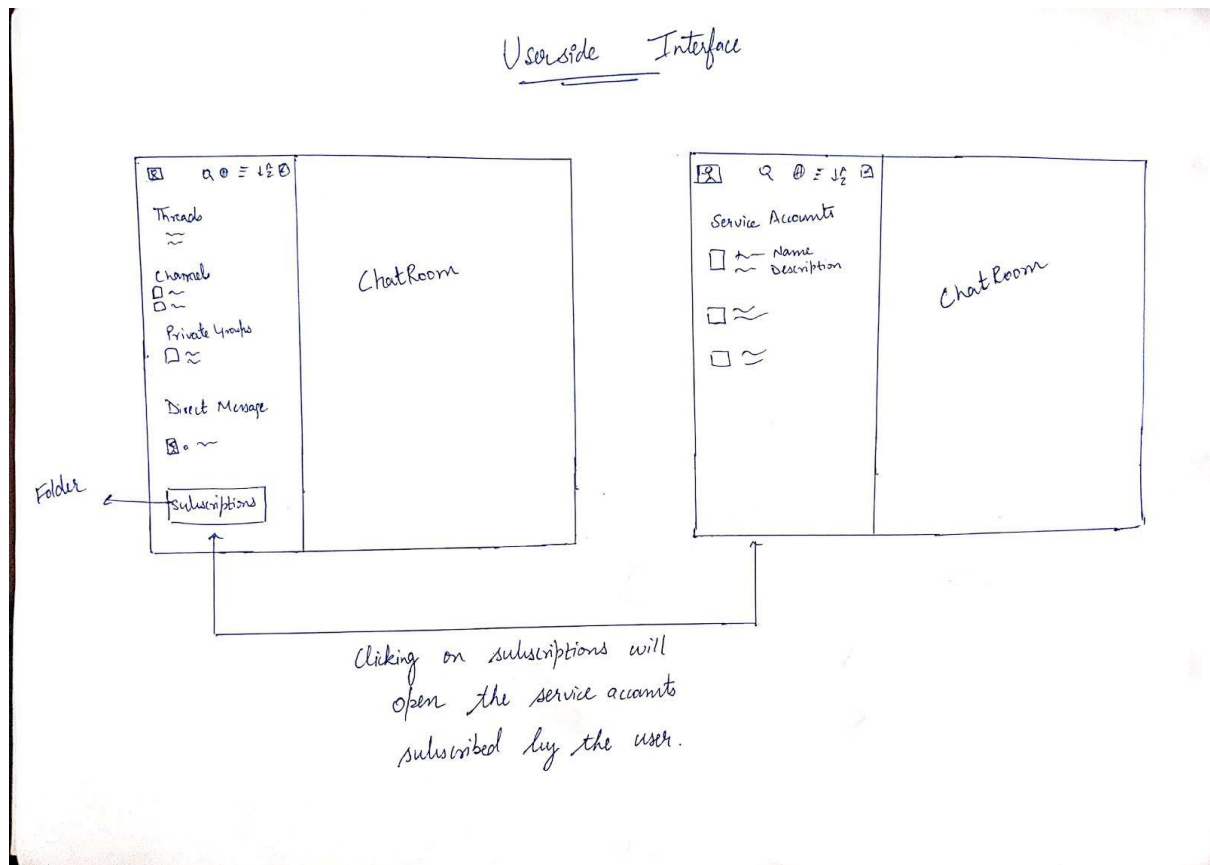
- There will be no separate login page for service accounts
- User owning a service account can login by clicking on the desired service account from the sidebar where all the service accounts owned by the user will be listed.
- Clicking on the service account name will open a new tab and the user session will be set to that of the service accounts.
- In the initial stage mobile clients need to logout and login to switch between accounts while desktop users can swap straight away.
- Users can log out the same way they logout of their regular accounts through the sidebar.

Other technical aspects of the service accounts project:

- New API endpoints will be created for mobile applications.
- Existing API endpoints propagating data from users collections will be modified.
- Ability to import service accounts from other servers will be included.
- Automated tests for service accounts

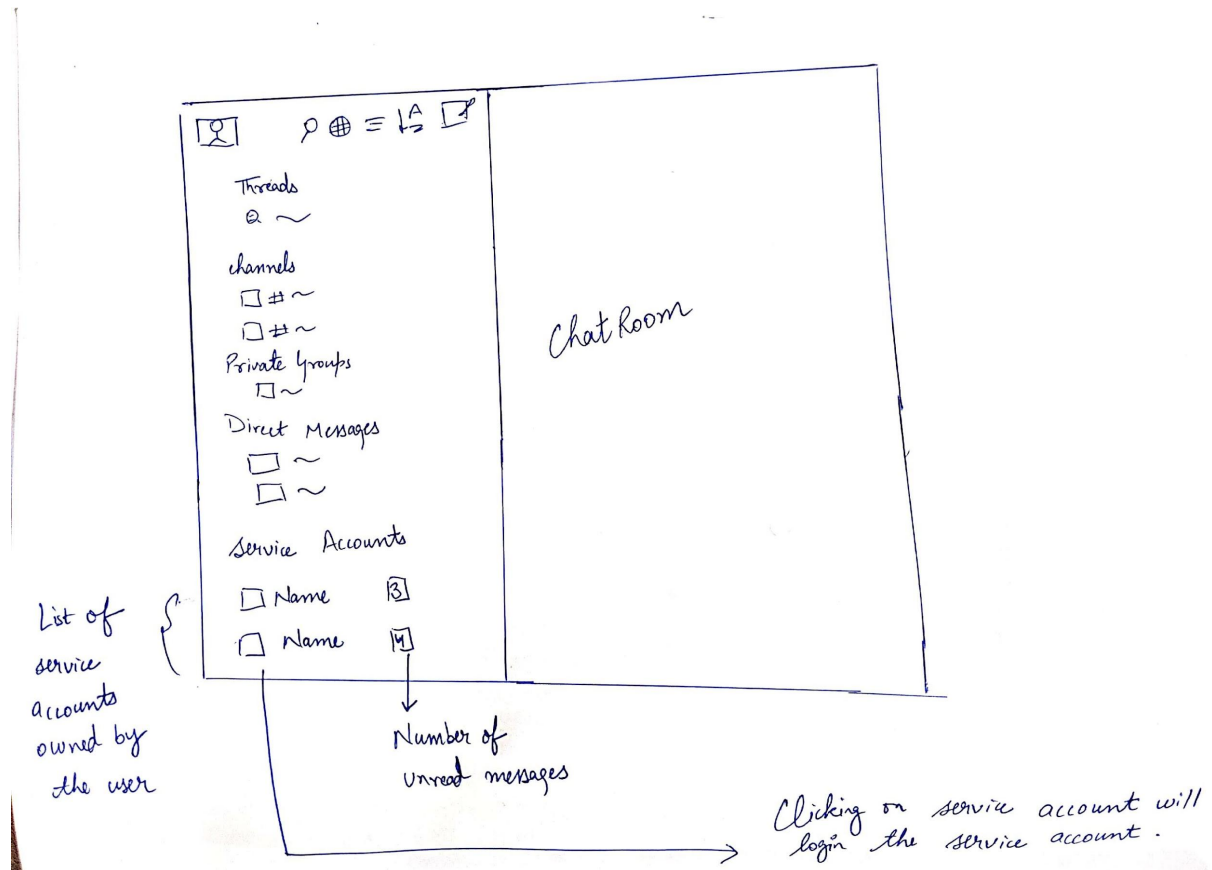
4. Wireframes

User Interface for Regular Users



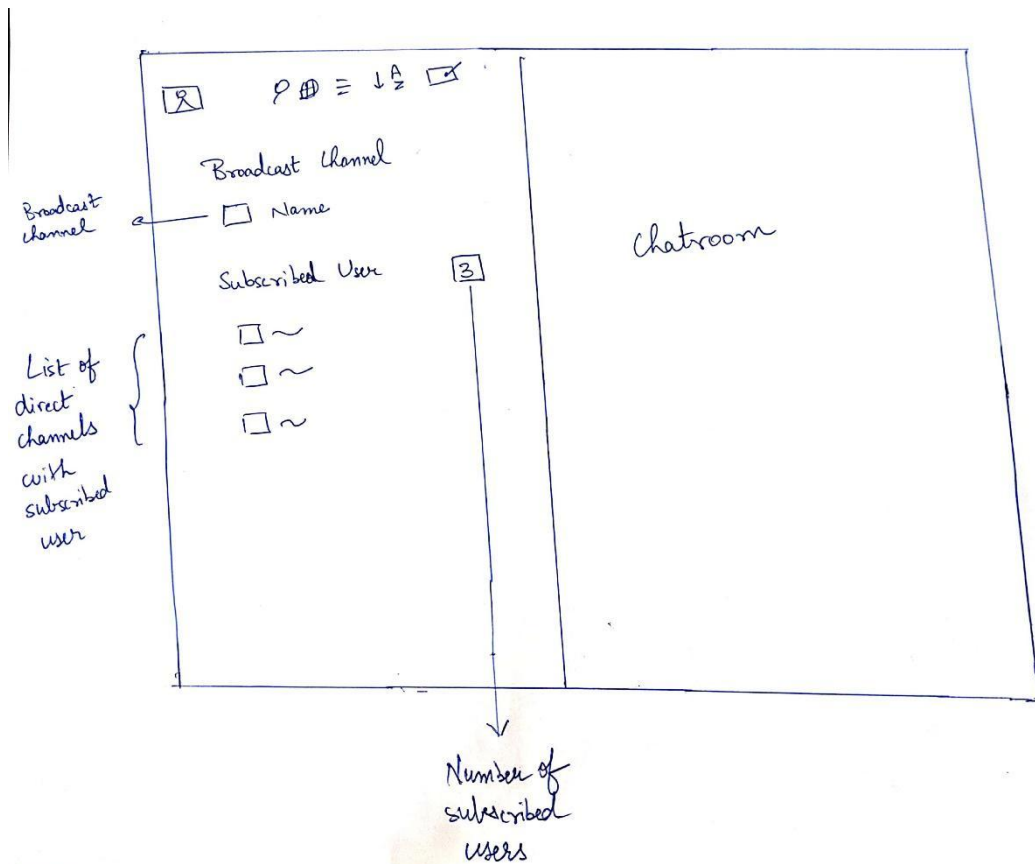
- No separate system level notification will be sent for service accounts.
- All service account notifications will be visible through subscriptions folder.
- Subscription folder will contain only 1-on-1 direct channels with service accounts.

User Interface for service account owners



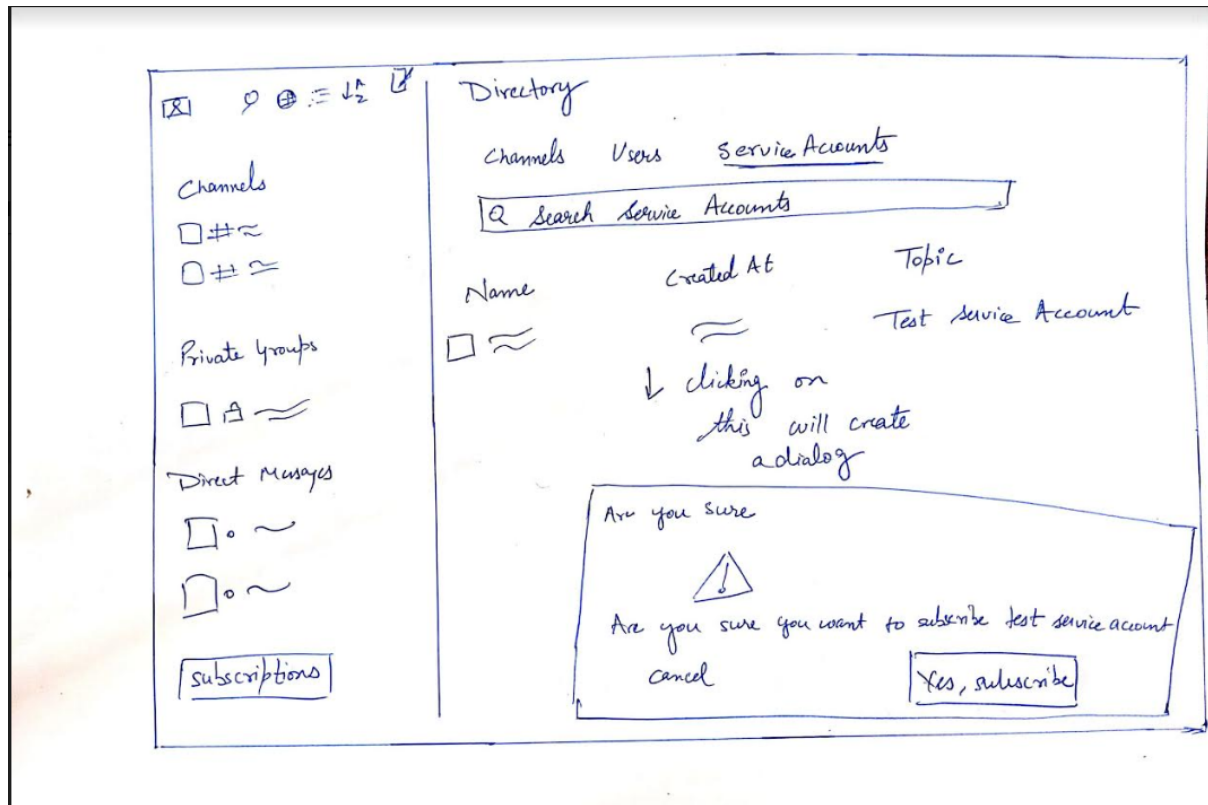
- Service accounts owned by the user will be listed in the sidebar.
- If the user confirms the current user session will be changed to that of the service account.
- Along with service account number of messages/number of subscribers can be shown for better user experience.

User Interface for Service accounts



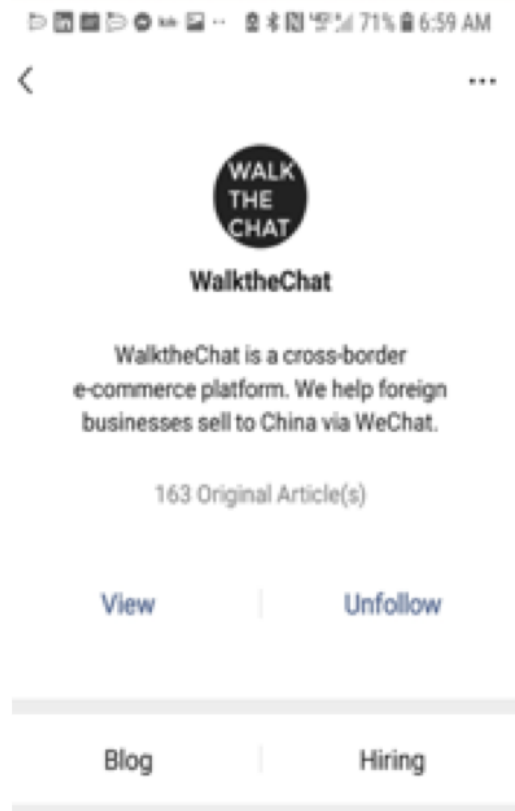
- Once logged in, the user interface for a service account will look similar to that of a regular user.
- A broadcast channel will be displayed at top which will be used for sending broadcast message to all the users using webhooks.
- Number of direct messages will be the number of subscribers of the service account.

Directory Search for Service Accounts



- Each service account will be listed with name, created at and topic.
- Users can directly subscribe to a service account from this window.
- On clicking the given service account entry, a modal will open asking for confirmation to subscribe to service account.
- Users can subscribe any number of service accounts based on the topic listed in the directory.

Service Accounts info Page



- After clicking on a service account from the directory a similar service account info page will open.
- This page will be linked with articles to display recent one or two articles.
- Along with that description of the service account will be listed over there.
- Users can subscribe to service account from there.

5. Project Timeline

Community Bonding Period:

Get to know the community members and mentors. Discuss modifications on new features/modules. Discuss technical aspects of this service account with community. Get feedback from the community regarding technicalities of the proposed implementation method of service accounts.

Week 1:

I would create new permissions and roles for the service accounts.

New admin panel settings for service account will be created.

Week 2:

I would work on creating a method which will allow users to apply for the role service account owner. A new tab will be added in admin panel where all the user application will be listed, admin can approve from there.

Week 3:

I would work on the UI to apply for service account owner and dialog for creation of new service accounts. Along with this I would be working on creation of a broadcast channel for the service account.

Week 4:

This week I would be working on one tap login mechanism for the service accounts for the owners and moderators. Along with this I would be working on UI for the service account owners/moderators.

Evaluation (24th June – 28th June):

So, the service account creation will be done successfully, owners can login to the service accounts as well. I would write tests for evaluation.

Week 5:

Implementation of service accounts exclusive settings. Along with this I will be working on adding mechanism to subscribe to a service account.

Week 6:

Broadcast messages facility from the broadcast channel created during the service account creation will be worked upon during this week. Messages from here will sent to all the subscribed users' direct channel by modifying the outgoing integration webhooks.

Week 7:

Client-side UI to move all the service account messages into a subscription folder will be added during this week.

Evaluation (22nd July – 26th July):

Till this point service accounts can be created by authorised users and can be subscribed by other users as well. Broadcast message facility for service accounts will be fully functional. Tests will be written to carry out the same. Broadcast messages facility included.

Week 8:

I would be working on adding discoverable service accounts to the directory under service accounts heading. Along with this I will be working on UI to allow users to subscribe to a service account by clicking on the name of the service account from the directory.

Week 9:

Testing on multiple devices and fixing bugs. Final testing and touch ups will be done for the UI. I would also write automated tests for service accounts.

Week 10:

Feedback and enhancement. Till this moment this feature would be ready for use. This week I would be working on creating documentation for service accounts.

Final Evaluation:

Service accounts will be discoverable from the directory and service accounts information page will be created. Along with this incorporation of community feedback ideas will be done.

6. About Me

Personal Details

Name : Aditya Bhardwaj
Email : baditya1207@gmail.com
Contact : +91 9810893638
Location : Chennai (Tamil Nadu), India
Time Zone : UTC +05:30
GitHub ID : <https://github.com/bhardwajaditya>
Skype ID : live:aditya.1998bhardwaj_1
University : Vellore Institute of Technology,
Chennai Campus, India
Major : Computer Science and Engineering
Current year : Third Year (2016-2020)
Degree : Bachelor of Technology

I am a MERN stack developer and an enthusiast programmer, passionate for learning new technologies. I am having an experience of 2 years in web development. I have received certification in full stack web development and React.JS development from Udemy. I have fair share of experience working as a part of team, I was head coordinator of web development team of Vibrance'19 (VIT annual cultural fest).

Previous Full Stack Projects:

I have built several full stack projects using Node.JS express and flask.

1. YelpCamp:

Details: Yelpcamp is a website aimed to assist people in rating campgrounds from all around the world. Users can check all the campgrounds posted by the users. Only authorized users can comment and add new campgrounds for users to view.

Technologies: Node.JS, express, MongoDB, EJS templates.

Github: <https://github.com/bhardwajaditya/YelpCamp>

2. Shashakti:

Details: Shashakti is an app with an objective to deliver relevant Government schemes to the desired users, based on the interests provided by the user the app will display the matching schemes. The additional feature of the app was that the User can apply directly from the app with all the necessary details filled using the Aadhar details. The application will then be displayed on a portal designed for Government employees, they can accept or reject the application using their digital signatures improving the transparency in the system.

Technologies: Node.JS, Express, MongoDB, Android Studio, EJS Templates

Github: <https://github.com/bhardwajaditya/Shashakti-Backend>

3. Medisyst:

Details: Medisyst is an app to assist doctors and patients all over the world. This app consists of two modules, the First module is aimed to help patients as our app will give the list of possible diseases based on the symptoms provided by the user. The second module is aimed for doctors where all the medical history will be accessible using a one time authorized by the patient itself.

Technologies: Flask, MongoDB, Android Studio

Github: <https://github.com/bhardwajaditya/Medisyst>

Contributions Made to Rocket.Chat

1. Join channels/discussions by sending a message.

Users can now join channels/discussions by sending a message into that channel and they will join the channel automatically.

<https://github.com/RocketChat/Rocket.Chat/pull/13752>

2. Mount reply when a message is replied from broadcast channel

Message replies from broadcast channel were not mounted in direct channels, it wasn't clear to which message a user has replied to.

<https://github.com/RocketChat/Rocket.Chat/pull/13941>

3. Allow moderators to send message in broadcast/read-only channels

By design moderators are allowed to message in read-only but they were muted.

<https://github.com/RocketChat/Rocket.Chat/pull/13711>

4. User discussion join message fixed

<https://github.com/RocketChat/Rocket.Chat/pull/13656>

5. Missing "view-outside-room-description" translation key was added.

<https://github.com/RocketChat/Rocket.Chat/pull/13680>

6. Open Spotify embed link opens in new tab

<https://github.com/RocketChat/Rocket.Chat/pull/13637>

7. Remove user from a role dialog box fixed

<https://github.com/RocketChat/Rocket.Chat/pull/13874>

8. Sidebar create new description text fixed

<https://github.com/RocketChat/Rocket.Chat/pull/13658>

9. Block user icon fixed

<https://github.com/RocketChat/Rocket.Chat/pull/13632>

FAQs

Why are you the right person for this task?

- I have hands on experience of 2 years in full stack web development. I have also worked on Node.JS and MongoDB before. Along with this I have made several contributions to Rocket.Chat existing codebase during pre Gsoc period.

Which coding environment will you be working on?

- I will be primarily working on Ubuntu 18.04, and I will use VS code editor.

Have you worked with us before?

- No, I haven't worked with Rocket.Chat before, But, I'm following this organization since GSOC'18. This year I started contributing to it and have listed out some bugs and sent some pull requests as well.

How many hours per week are you going to contribute towards this project?

- 40-50 hours per week. I'm very comfortable and willing to work on weekends. I would like to conduct review session on weekends with my mentor.

Have you participated in GSOC previously?

- No, this is the first time I am participating in GSOC.

Do you have any other commitments that we should know about?

- I don't have any other commitments during my summer breaks. But my summer breaks will get over by 10th July. So, after 10th July I would be able to dedicate 5-6 hours on weekdays and 7-8 hours on weekends. I would make sure that committed working hours to the organization are not disturbed.

What is your after-GSoC plan?

- I would love to continue working with Rocket.Chat since the application is very appealing and perfectly aligns along with my interests.

7. Future Improvements

- Welcome message as soon as a user subscribes to a service account.
- Linking bots to service accounts to automate the reply in case the owner/moderators are not available.
- Adding the functionality of articles and news feed to service accounts.
- Allowing service accounts to perform system actions in order to improve the user experience of Rocket.Chat.
- Adding additional information of service account such as website, working hours, etc.
- Set up welcome greetings and quick replies.