Import functionality for the Sprinklr Community Product

Rutvik Mavani - 201601152
Dhirubhai Ambani Institute of Information and Communication Technology
Gandhinagar, India
201601152@daiict.ac.in

On-Campus Mentor : Prof. Jaideep Mulherkar
Off-Campus Mentor : Sandeep Maheshwari
Associate Vice President, Product Engineering, Sprinklr
smaheshwari@sprinklr.com

Spinklr India Pvt.Ltd.

***Abstract*—All the complex webapps which interact with data have a significant amount of engineering done at the backend part. Maintaining and storing data is an important part of the backend. My BTP work involves learning tools and technology used by the company and to develop import functionality for Sprinklr Community product from an external system and enhance community product with new features like Rank/Badges, user, post etc.**

***Index Terms*—Data migration, REST API, SQL, MongoDB**

I. Introduction

In today's customer-centric era companies have to listen to their customer's complaints and issues. The companies have online discussion forums where customers can ask questions related to products, resolve issues, and post complaints about products. I am the part of the Care team which works on Modern Care-One of five products of Sprinklr. One job of the care team is to build online discussion forums for companies. The care team was assigned to migrate an old online forum to Sprinklr's forum for a company called Voo. Voo is a Belgium based company that works in the area of the internet, TV cable [8]. Voo had an online forum for customers where customers can ask questions, give answers, post problems, etc. The care team was assigned to create a new forum by maintaining the old forum's data. In the process of migration, data needs to be migrated from the SQL database to mongoDB. I was part of the backend team so UI (User Interface) for the forum was done by the frontend team of care.

II. Tools and Technologies Used

A. Java

Java is a widely used language in enterprise systems. It is used to write code for the backend side of the application. Its Object-Oriented capabilities make easier to design and manage system code. So, it is important to familiarize with java basics to write clean code. Some of the important concepts of java which helped in the project are as follows:

- Generics
- Collection Framework
- Concurrency

Knowledge of java generics helps to create reusable code. Generic code can works for multiple datatypes. Collection framework has built-in data structures like ArrayList, LinkedList, Hashmap which are used frequently in program to store and manipulate data. Java has built-in support for threads. To utilize the CPU cycle efficiently and get the benefit of multi-core processor multi-threading needs to be done. Understanding the java thread life cycle helps managing threads.

B. REST API

REST stands for Representational state transfer. There are 6 constraints which must be satisfied in RESTful system. These properties are as follows: client-server architecture, stateless, cacheable, Uniform interface, Layered system, code on demand (optional) [5]. REST APIs can be implemented by protocols like HTTP, SNMP, SMTP etc [6]. But HTTP is the most commonly used transport layer protocol. The frontend of system needs to show, add, update and delete data according to client's action. To perform these operations frontend developers make calls to REST APIs. This API call contains specific URL, request parameters and query parameters depends upon data. When API call is made an HTTP request is sent to the server. Backend is designed to handle such requests. The backend will make the database call according to HTTP request parameters. After interacting with the database, the backend will generate a response which will be sent back to the client who made that request.

C. MongoDB

MongoDB is NoSQL database. It stores data in collections, unlike SQL which stores data in tables. The collection is a set of documents. Each document has some fields. Data is stored in JSON (JavaScript Object Notation) format in the documents. The document is equivalent row in SQL database. Document models are mapped to the backend object making them easy to handle [2].

D. Git and GitLab

Git is the version control system. It helps teams to develop features simultaneously by creating branches. It helps in managing the project and maintaining the history of changes. Software delivery lifecycle is efficiently managed by GitLab [3]. GitLab has great planning tools to keep everyone in sync [3].

E. Spring Framework

Spring framework helps java developer to develop reliable and scalable enterprise system. One of the techniques of the framework is dependency injection which helps to create a class instance at runtime using java object reflection [7]. This technique used to make class independent of its dependency.

F. Jenkins

Jenkins is used to building, test, deliver and deploy software and related task and automate such kind of task [4]. It can build specific branches of the project provided by the user.

III. Work Involved

A. Analysis of SQL Schema

A dump file was given by client which consist of files(Images, documents) and SQL script to import data. From the script, we got SQL relational schema consisting more than 100 tables by reverse engineer it using MYSQL workbench. The schema had tables like user, thread, post etc. This SQL database was maintained by the client's previous solution provider. Our job was to look at each table, what information they stores and how they are connected. After looking at every field of tables some empty tables were found out and removed making 87 tables relevant. I and my teammates run SQL queries to analyse tables and to find connections among tables. Apart from that we interpreted data by looking at forum. Each of us worked on our part and established mapping between fields of the SQL table and fields of MongoDB collections. After mapping, we found out that many collections required data of more than one SQL tables. After analysing schema it was decided to divide migration into three phases.

B. Upload Data

Attachments were uploaded to the remote server. SQL file provided by client did not execute on the server because while making dump –hex-blob flag was not included in mysqldump command. So a new SQL file with the flag was generated by me on the local machine and uploaded again on the remote server. Attachments were profile pictures and files attached in posts.

C. Writing code

In the first phase of migration, I was assigned to migrate users. Sprinklr has a collection inside mongoDB which is equivalent to user table of SQL schema. A migration service for user was written by me. This service queries user table and for each user, it queries three other tables using the user's primary key, fetches required information of the user. All query results are converted into java objects using proper mappings. All java objects of the individual query are combined into one object that can be sent forward. These combined objects are batched into the size of 100. Sprinklr has already created REST APIs to interact with mongoDB. Each batch is given as payload to post request to create new users in mongoDB. When this post request hit another remote server, particular REST API is called which inserts users provided in the payload by running insert query on mongoDB. There were some fields related to the user which can not be mapped to user collection in mongoDB by running existing code. To fix that issue it required me to change code which stores data in mongoDB.

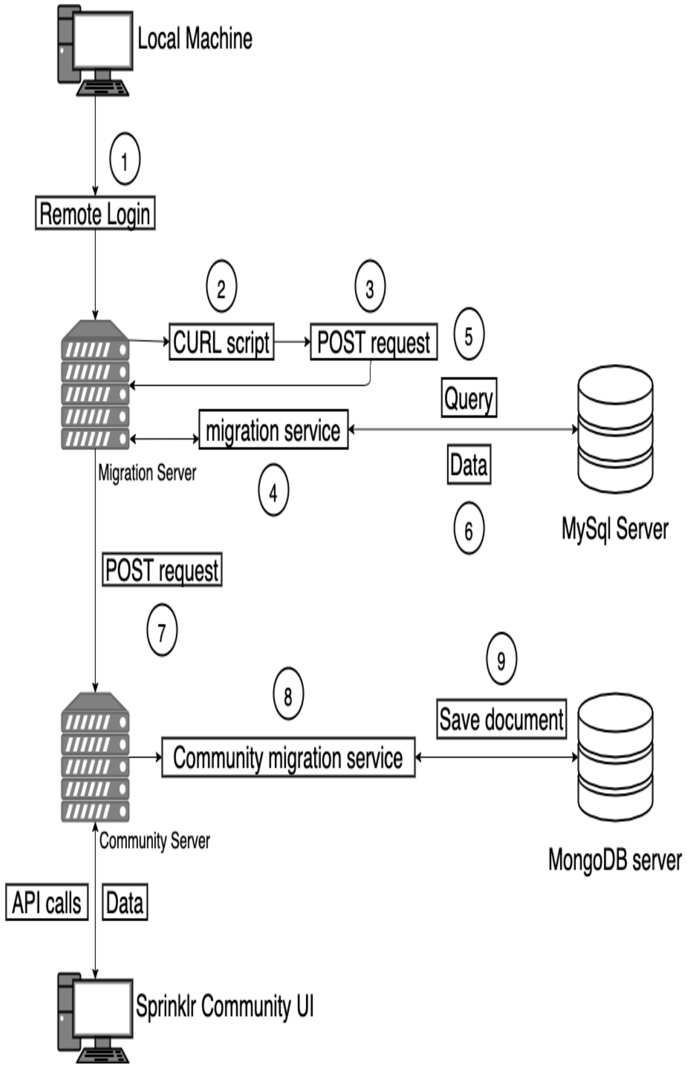
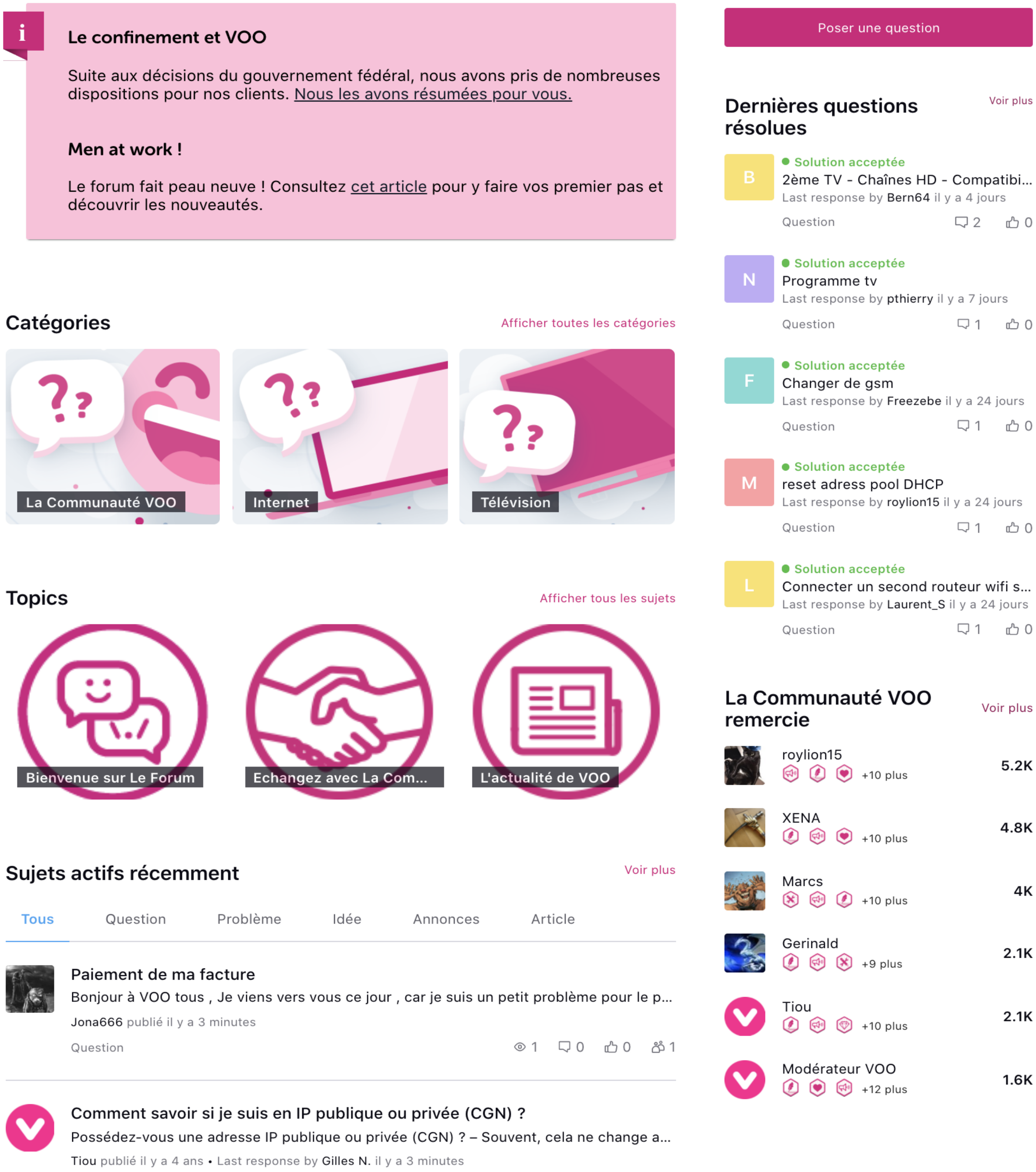


Fig. 1. Execution Flow of migration

In the third phase of migration, we wanted to migrate rank and badges. The old forum had the feature of giving the badge to user if certain conditions were satisfied. Conditions include certain number of answers, number of likes received exceeds threshold etc. Apart from that each user has rank. Rank and badges were shown at user profile page. The sprinklr



Le confinement et VOO

Suite aux décisions du gouvernement fédéral, nous avons pris de nombreuses dispositions pour nos clients. Nous les avons résumées pour vous.

Men at work !

Le forum fait peau neuve ! Consultez cet article pour y faire vos premier pas et découvrir les nouveautés.

Catégories

La Communauté VOO · Internet · Télévision

Topics

Bienvenue sur Le Forum · Echangez avec La Com... · L'actualité de VOO

Sujets actifs récemment

Paiement de ma facture
Bonjour à VOO tous , Je viens vers vous ce jour , car je suis un petit problème pour le p...
Jona666 publié il y a 3 minutes
Question 1 0 0 1

Comment savoir si je suis en IP publique ou privée (CGN) ?
Possédez-vous une adresse IP publique ou privée (CGN) ? – Souvent, cela ne change a...
Tiou publié il y a 4 ans • Last response by Gilles N. il y a 3 minutes

Dernières questions résolues

B · Solution acceptée
2ème TV - Chaînes HD - Compatibi...
Last response by Bern64 il y a 4 jours
Question 2 0

N · Solution acceptée
Programme tv
Last response by pthierry il y a 7 jours
Question 1 0

F · Solution acceptée
Changer de gsm
Last response by Freezebe il y a 24 jours
Question 1 0

M · Solution acceptée
reset adress pool DHCP
Last response by roylion15 il y a 24 jours
Question 1 0

L · Solution acceptée
Connecter un second routeur wifi s...
Last response by Laurent_S il y a 24 jours
Question 1 0

La Communauté VOO remercie

roylion15	+10 plus	5.2K
XENA	+10 plus	4.8K
Marcs	+10 plus	4K
Gerinald	+9 plus	2.1K
Tiou	+10 plus	2.1K
Modérateur VOO	+12 plus	1.6K

Fig. 2. Voo Migrated Community UI [9]

community has awards equivalent to rank and badges. The fields of rank and badges were easily mapped. First, we migrated badges and rank to store awards in mongoDB. Then another migration was written to assign awards to users.

D. Test Migration Service

To verify the working of migration service we have made a new testing forum which contains no data in mongoDB. A testing forum can be generated using tools created by sprinklr. Each testing forum has a unique identification string. Documents stored in mongoDB has unique identification string of the forum which is used to decide in which forum the document belongs. As shown in fig. 1 remote login to the server is done by local machine using SSH (Secure Shell). After SSH login, a shell script which consists of CURL command is created on remote server. Unique identification string of testing forum is supposed to be provided in CURL command. CURL (Client URL) support many protocols to interact with APIs on any platform [1]. In this case, CURL is used to send an HTTP POST request to server's own IP address. As soon as, POST request hit the REST APIs running on the server, migration service is invoked. As shown in fig. 1 migration service run queries on SQL database. These query results are used by migration service to combine results. Combined results are put into the body of POST request. The server sends POST request to community server. This community server invokes migration service to handle POST request. Migration service performs additional checks and finally, the document gets saved in mongoDB. Multithreading was added at later stages to reduce migration time. Without multithreading migration of users took more than 1 hour. Multithreaded code made migration complete in around 15 minutes.

Fig. 2 shows the home page of migrated community. Top-Right side shows last questions answered and the bottom-right side shows a list of user. Migration process required few weeks. During that time old system's database received new data and requests to modify original data. For example, user's profile picture might have changed, new users get created. To handle this issue delta migration was performed which migrate only changes. It should not add inconsistency in data.

IV. Future Enhancement

The Conditions inside badges were in JSON (JavaScript Object Notation) format. We needed to convert that conditions into mongoDB query. If the number of conditions is low then this can be done manually. But a program can be written to perform this task. Sprinklr community do not have support for emoticons like the old community. The old community can show emoticons inside the post content. Emoticons were coded in their custom language. Sprinklr community can not create the poll. Support for poll can be added to migrate data related to poll.

V. Conclusion

This internship gave me exposure to the industry and gave a chance to apply my knowledge. The courses like DBMS, Networks and Operating Systems was helpful throughout the project. I have become familiar with the tools, technology and practices used in the industry. I have learnt to work with the team and to resolve team conflicts.

Acknowledgment

I would like to thank prof. Jaideep Mulherkar to monitor my BTP project. I am grateful to my mentor Sandeep Maheshwari for all the guidance, help, and sharing his broad experience. Special thanks to my colleague Venkatesh for the guidance and teaching me how to use tools of the company. I would like to thank Nirav Dobariya and Darshan Patel my classmates as well as teammates for all help and support. I am thankful to Sprinklr for providing this great opportunity.

References

[1] https://developer.ibm.com/articles/what-is-curl-command/a
[2] https://www.mongodb.com/what-is-mongodb
[3] https://about.gitlab.com/stages-devops-lifecycle/
[4] https://www.jenkins.io/doc/https://www.jenkins.io/doc/
[5] https://en.wikipedia.org/wiki/Representational_state_transfer
[6] http://restcookbook.com/Miscellaneous/rest-and-http/
[7] https://en.wikipedia.org/wiki/Spring_Framework#Inversion_of_control_container_(dependency_injection)
[8] https://en.wikipedia.org/wiki/VOO
[9] https://forum.voo.be/