

VACCINE REGISTRATION SYSTEM INTEGRATED WITH KEYCLOAK

Vaddi Tharun Sai¹, D. Kalyan², U. Chaitanya³, Dr. D. Vijaya Lakshmi⁴

^{1,2} UG Student, ³ Assistant Professor, ⁴ Professor, Dept. of Information Technology,

Mahatma Gandhi Institute of Technology, Hyderabad, India

Received on 24 May 2022 Received in revised form 26 May 2022 Accepted 27 May 2022

ABSTRACT

The objective of this paper is to provide an online registration system for covid-19 vaccination. In this crisis, it will be difficult to register in person and get vaccinated after waiting in a queue for a long period; thus, registering online is the safest and most convenient way to schedule slots on a convenient date and get vaccinated. We built this application using spring boot, which allows developers to directly get into coding without wasting time in setting up and configuring the environment. For authentication and authorization, we integrated it with keycloak, a third-party service, which is a simple way to make our web-based apps more secure and Keycloak provides distinctive features to the application like single Sign On, OpenID Connect Support, Social Logins and many more. So, this is a web-based service where any individual can conveniently schedule their slots from anywhere according to the availability in their desired locations.

Keywords — COVID-19 Vaccine, Vaccine Booking, REST API, Spring Boot, Keycloak, Authentication and Authorization.

1 INTRODUCTION

COVID-19 was declared a global pandemic by the World Health Organization in March 2020, indicating the considerable global spread of an infectious disease. Since then, there have been millions of cases and millions of people have died. The development of safe and effective Covid-19 vaccination is a significant step forward in the global campaign to eradicate the epidemic. Hence, an online service will be crucial for booking vaccine slots in nearby hospitals which helps every individual to easily book their slots from their convenient place. This web application is created using spring boot, which can create a standalone application that can run on its own, and even it does not require any external servers to run because here it has an embedded server like tomcat. In this application, there are two roles, admin and user.

Here, an admin can manage the hospital details, vaccine timings, and details related to vaccine models, and a user can select their nearby hospital and type of vaccine for vaccination and can book their slots in the available timings and can safely get vaccinated. This allows the user to receive a reminder to his email, which is useful for people who are busy at work. They also receive their slot times and booking information by email, which they may check at any time. Elderly individuals

must be able to reserve their slots because they are more vulnerable to the epidemic than younger people due to their weakened immune systems. As a result, it will be preferable if they can book individually without the assistance of others. The application is simple to use and has more functionalities than the previous one and it was designed in such a way that even elderly people can easily book their slots conveniently. In this application, we integrated our spring boot-based application with keycloak[1], a third party for authentication and authorization. Keycloak provides many unique features like single sign-on, supporting multiple protocols like OAuth 2.0[2], OpenID Connect[3], and SAML. As everyone today are using social media apps, so for their convenience, the keycloak also has a feature called Identity Brokering and Social Login, where the users can log in using their social media apps conveniently. In the keycloak admin console, we can manage the sessions of the users and can also customize the default login page of keycloak. After deploying this application, it became accessible to users who want to get vaccinated safely. Finally, when compared with the existing system, which is integrated with WSO2 IS(Web Services oxygenated Identity Server)[4], this system is easy to deploy and maintain.

2 RELATED WORK

An Online Appointment Booking System [5] was proposed by Akshay V, et al. It's a web-based registration solution for parlour, hospital, and architect appointments in a specified geographic area. On an ionic basis, this application is streamlined. It's an open-source SDK that lets you make web and mobile apps. CSS, HTML, and JavaScript are all used. Firebase is required for accessing data for appointment scheduling, which supports in the development of useful applications. Analytics, database, messaging, and crash reporting are all included in this. The system keeps track of the number of requests by using NodeJS. Each request defines a sequence of locations that a user should view.

Shelar Pooja, et al.[6] proposed Smart Appointment Generation For Patient. In hospitals, making appointments and schedules are used to keep track of and regulate access to facility providers. A greater patient care, strict cost control, increased profitability, medical record management, and patient information privacy are all advantages of this smart appointment generation for a patient. The research is quantitative, with the majority of the data coming from surveys. Existing survey data and unique case studies are taken into account to arrive at a definitive conclusion. We can log in with a username and password with this system.

Web-Based Medical Appointment Systems were proposed by Peng Zhao et al [7]. The goal of this study was to assess the benefits and challenges of implementing Web-based medical scheduling, and also unmet needs in today's modern healthcare system, as indicated in the literature. Due to concerns regarding cost, flexibility, safety, and integrity, providers are reluctant to migrate to Web-based scheduling. Patients' concerns about adopting Web-based appointment booking are mostly based on their previous computer and Internet experiences. The use of Web-based appointment systems, on the other hand, appears to be on the rise, as per the literature. According to the conclusions of this study, Web-based scheduling methods can improve a variety of patient outcomes.

Globus Auth [8] is a software-as-a-service system that is similar to Keycloak and WSO2 IS in terms of functionalities. Globus Auth can also be used in groups and is linked to other Globus services, such as file transmission. Gateways and middleware operators can leverage open source technologies like Keycloak and WSOS IS to provide identity management services. They can also be integrated with Globus Auth by making it a trusted identity provider. This would allow a gateway built on Apache Airavata to utilize Globus file transfer services and identity management services which is based on Keycloak.

3 METHODOLOGY

3.1 Modules: This System consists of two modules, which are admin module and user module, when we register we will be automatically assigned to user module where we will have the option to book our vaccination slots and we also have the option to verify the vaccine booking for confirmation. Here admin is the one who manages this entire application and who has the control to modify or add any details related to hospitals or the vaccine models, its timings and number of slots and everything is managed by admin.

Admin Module

An admin has a role where they can add or delete the hospitals with respect to their availability, an admin can add or delete vaccine models, number of slots and the timings accordingly.

User Module

The users can login to the application and can book their vaccination slots in their nearby hospitals according to the availability of the slots and they also have an option to delete the scheduled slots.

Hence these are the two different roles in this vaccine registration system and these roles are managed by keycloak.

3.2 Architecture: In this vaccine registration system, our spring boot application is integrated with the keycloak to improve the systems performance and to add some unique features. Here, an admin can access all the functionalities and a user accesses websites over the internet, and when he makes a request, if he is approved, he receives the desired page; if he is not authorized, he receives an error page. The architecture includes a spring boot UI server that is linked to a spring boot API server and a keycloak server as shown in the below figure 1, and all of which are linked to their respective databases which helps to protect REST APIs from unauthorized calls.

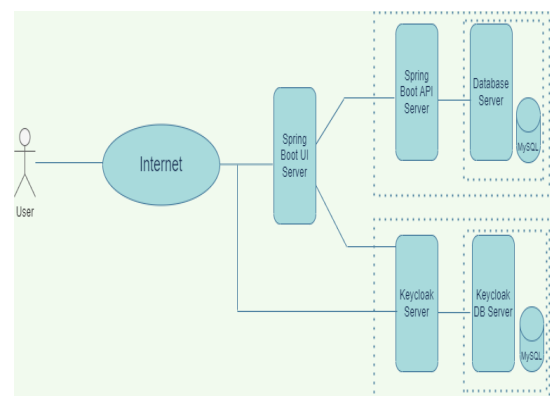


Fig.1. Architecture of Vaccine Registration System

3.3 Proposed System Methodology:

The following procedures must be followed in order to successfully integrate our spring boot application with the keycloak.

- Creating a SPRING BOOT project using Spring Initializr
- Install MySQL database for managing data.
- Installing Keycloak for authentication and authorization.

3.3.1 Creating a SPRING BOOT project using Spring Initializr [9].

To create a spring boot project, go to the web and search for spring initializr, then click the Generate button after providing the Group and Artifact names, as well as dependencies. Now that we have downloaded the zip file, we must extract it and import it into the Eclipse IDE before we can begin developing our application.

• 3.3.2 Install MySQL database for managing data.[10]

Download the community server edition software from MySQL's official website.

You'll notice an option to select an operating system, such as Windows, here.

Now, after installing MySQL, open MySQL Workbench and begin creating tables and defining required attributes; attributes in the database and classes must be the same to avoid mismatch problems.

Now in application.properties in Eclipse IDE, we need to write:
spring.datasource.url=jdbc:mysql://localhost:(port)/database
name
spring.datasource.username=
spring.datasource.password=

This connects your database and your spring boot project.

• 3.3.3 Installing Keycloak for authentication and authorization.

Following the development of this application, it is critical to ensure that it is secure so that no data is compromised. There are many security services available, but they are difficult to implement and lack many features. In this case, the best service is given by keycloak, which is secure and has many unique features. To run a keycloak server we need to enter the bin directory of the Keycloak file and execute the standalone.bat query in windows.

Then we need to go to the admin console in <http://localhost:8080/auth> in the browser and further create roles and assign users to those roles in the console, which

provides authorization. The keycloak is connected to the application by writing a query in application.properties. You can directly mention your keycloak realm and server url in your eclipse IDE in application.properties and project name mentioned in keycloak admin login.

```
keycloak.auth-server-url=http://localhost:8080/auth
keycloak.realm=SpringBoot
keycloak.resource=Covidvaccineregistration
application
```

This connects keycloak to your spring boot project.

Finally, after running this application, we can see a login page where you can log in if you're already a user or you can register and can start booking your vaccination in nearby hospitals which are mentioned. With its unique features such as Single-Sign On, Identity Brokering and Social Login, User Federation, Account Management Console, and many more, Keycloak enables our project to effectively secure our application.

4 RESULTS

After running query in command prompt for Keycloak and connecting mysql to Eclipse IDE, we need to run the project and after running we get the following figure 2(1), which is a User login page generated from keycloak.

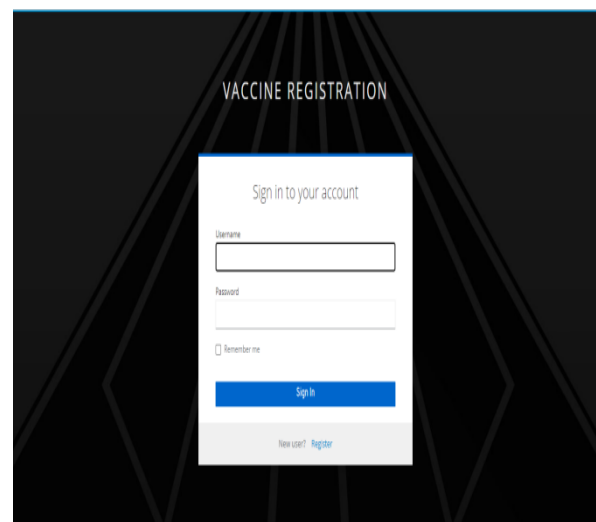
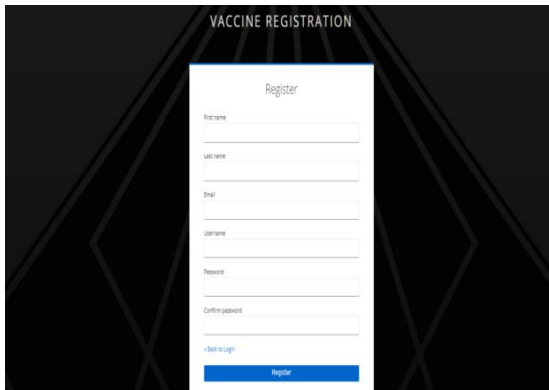


Fig. 2(1). User Login Page

If you are new to this application then you can register as a new User as shown in the below figure 4.2 and you'll be assigned to user role. You need to provide some details in order to register as shown below.



The image shows a 'VACCINE REGISTRATION' page with a 'Register' form. The form includes fields for First name, Last name, Email, Username, Password, and Confirm password. There is a 'Back to Login' link and a 'Register' button at the bottom.

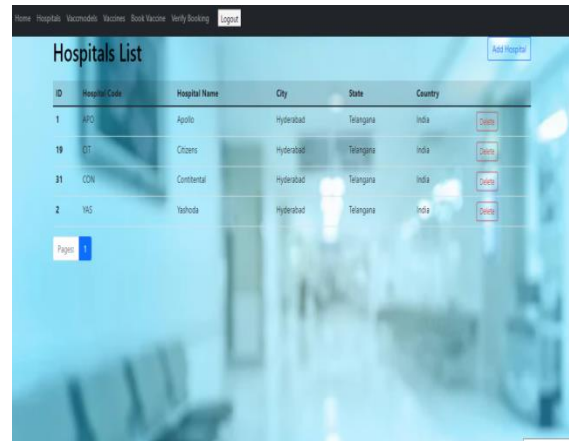
Fig. 2(2). New User Registration Page

After logged in to this application, we can see the Home page as of vaccine registration application, where we have option to book or verify our vaccination slot as in the figure 3.



Fig. 3. Home Page

If you are admin then you can access the hospital page as shown in figure 4 for adding and deleting hospitals when ever required.

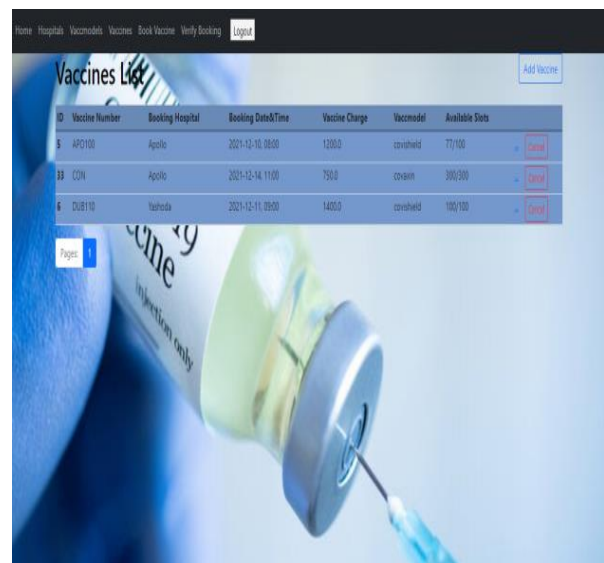


The image shows the 'Hospitals List' page. It contains a table with columns: ID, Hospital Code, Hospital Name, City, State, and Country. The table lists four hospitals: Apollo, Citi, Continental, and Yashoda, all located in Hyderabad, Telangana, India. Each row has a 'Delete' button. There is also an 'Add Hospital' button and a 'Pages' indicator showing 1 page.

ID	Hospital Code	Hospital Name	City	State	Country
1	APD	Apollo	Hyderabad	Telangana	India
19	CT	Citi	Hyderabad	Telangana	India
31	CON	Continental	Hyderabad	Telangana	India
2	YAS	Yashoda	Hyderabad	Telangana	India

Fig. 4. Hospitals list

Now, the important thing is that to add different timings of vaccination slots in different hospitals. Admins can add it in the vaccine page where they can add no of slots and type of vaccine at that time. Everything is managed by admin here. The below figure .5 is the web page for it.



The image shows the 'Vaccines List' page. It contains a table with columns: ID, Vaccine Number, Booking Hospital, Booking Date/Time, Vaccine Charge, Vaccines, and Available Slots. The table lists three vaccine bookings: APD100, CON, and CON110, all booked at Apollo and Yashoda hospitals. Each row has a 'Delete' button. There is also an 'Add Vaccine' button and a 'Pages' indicator showing 1 page.

ID	Vaccine Number	Booking Hospital	Booking Date/Time	Vaccine Charge	Vaccines	Available Slots
5	APD100	Apollo	2021-12-10 08:00	12000	covishield	77/100
13	CON	Apollo	2021-12-14 11:00	7500	covaxin	300/300
6	CON110	Yashoda	2021-12-11 09:00	14000	covishield	100/100

Fig. 5. vaccine slots

Now, after clicking book vaccine, now we can select the nearby hospital and date and then we can book our required vaccine type as shown in figure 4.6 and in our suitable time available for that hospital.



Fig..6. slot booking page

After successfully booking your vaccination slot, you'll be receiving your vaccine Id and Patient Id, and if you want to verify your booking you can always go to the verification page and check your confirmed slot timings as shown in below figure 7.

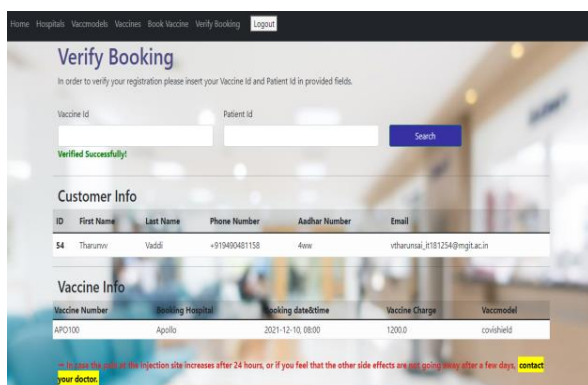


Fig. 7. Verify booking

5 CONCLUSION AND FUTURE SCOPE

The proposed vaccine registration system is flexible to everyone and users can easily book their slots in their desired and available hospitals to get their vaccination. The integration with Keycloak made us easily secure our application and made it simple for users with its unique features.

WSO2 IS worked well for the initial use cases in the existing system, and we were able to put this into production without any issue, but the problem is that it failed when we added new requirements. Also, despite the fact that WSO2 IS is open source software, the required improvements proved to be too complex to deploy and manage. So, as we replaced IS with Keycloak, which uses a simple role-based authentication mechanism, the difficulty was reduced by providing additional wrapper coding to segregate implementation specifics.

The future work would be adding a feature called volunteer, which means they can opt as a volunteer and help in running the vaccination drives. And for the people who were already vaccinated can register as a volunteer to participate. This feature should also predict the number of volunteers required at a particular location to successfully run the vaccine drives across different places in a country.

REFERENCES

- [1] Keycloak: <http://www.keycloak.org/>
- [2] Hardt, Dick. "The OAuth 2.0 authorization framework." Internet Engineering Task Force (IETF) (2012). <https://datatracker.ietf.org/doc/html/rfc6749>
- [3] Sakimura, Natsuhiko, John Bradley, Mike Jones, Breno De Medeiros, and Chuck Mortimore. "Openid connect core 1.0." The OpenID Foundation (2014): S3
- [4] WSO2 Identity Server: <http://wso2.com/identity-and-access-management>
- [5] V. Akshay; Anish Kumar S.; R.M. Alagappan,; S. Gnanavel. "BOOKAZOR-an Online Appointment Booking System." 2019 International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN).IEEE, 2019.DOI: 10.1109/ViTECoN.2019.8899460
- [6] Shelar Pooja, Hande Nilima, Dhamak Prajakta, Hingane Nisha, Jadhav Vinayak, 'Smart Appointment Generation For Patient', International Journal of Advance Engineering and research Development, 5(2018)1-3
- [7]Peng Zhao, Illhoi Yoo, ²Jaie Lavoie, Beau James Lavoie, and Eduardo Simoes,'Web-Based Medical Appointment Systems: A Systematic Review", J Med Internet Res.19(2017) e134 doi: 10.2196/jmir.6747
- [8] Steven Tuecke, Rachana Anantha krishnan, Kyle Chard, Mattias Lidman, Brendan McCollam, Stephen Rosen, Ian Foster,"Globus Auth: A research identity and access management platform." 2016 IEEE 12th International Conference on e-Science (e-Science). IEEE, 2016. DOI: 10.1109/eScience.2016.7870901
- [9] Suryotrisongko, Hatma, Dedy Puji Jayanto, and Aris Tjahyanto. "Design and development of backend application for public complaint systems using microservice spring boot." Procedia Computer Science 124 (2017)736-743.
- [10] Letkowski, Jerzy. "Doing database design with MySQL." Journal of Technology Research 6 (2014)1-15