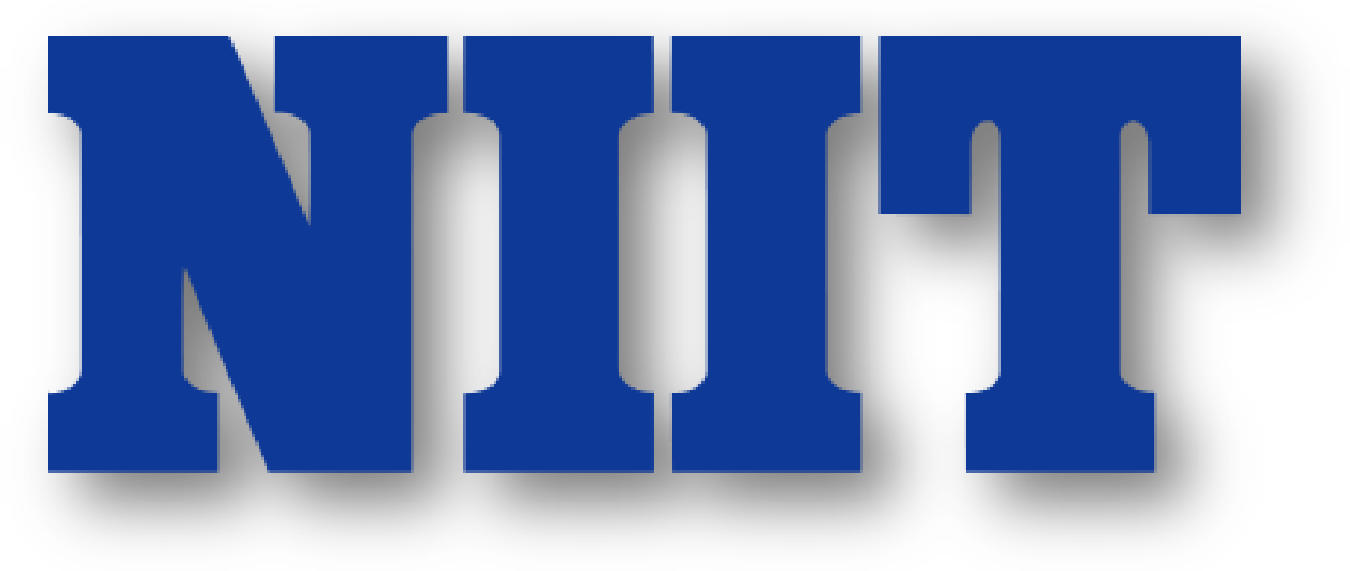
PROJECT ON

**OnlineCollaboration Web Application**



**Developed By: - S.K.RAHUL**

**OnlineCollaborationWeb Application**

Batch Code : S210192

Start Date :

End Date :

Name Of The Coordinator : Mrs. Lopamudra Bera

Name Of The Developer : S.K. RAHUL

Date Of Submission :

**OnlineCollaborationWeb Application**

CERTIFICATE

This is to certify that this report, titled OnlineCollaborationWeb Application embodies the original work done by S.K.RAHUL in partial fulfillment of their course requirement at NIIT.

CO-ORDINATOR:

Acknowledgement

I would like to express my special thanks of gratitude to my teacher Mrs. Lopamudra Bera who gave me the golden opportunity to do this wonderful project on the topic OnlineCollabortionWeb Application, which also helped me in doing a lot of Research and I came to know about so many new things I am really thankful to them.

Abstract

This project is based on Web application where user is able to enter the user credentials to login the site and to signup if the user is not registered and after sigin the user can post a blog in the site and admin has the privileges to enable or disable the blog and also the admin can display the enabled blog in the site.And the user can see the status whether the blog has been uploaded in the web or not .Also the user can see the other details(date, n.oOfLikesetc)for the created blog .

Configuration

Hardware

Processor : Intel i3 or higher.

RAM : 2GB or higher (minimum)

Speed : 2.0GHz

Storage : 1TB

software

Java Version : JDK 15

Database : MySQL

IDE : Eclipse

OS : Windows 10.

Index

|  |  |  |
| --- | --- | --- |
| SNO | Title | Page No. |
|  | Introduction | 08 |
|  | Aim and Objective | 09 |
|  | Case Study | 10 |
|  | Project Requirements and Specification | 12 |
|  | Project Analysis | 13 |
|  | Implementation and Diagrams | 14 |
|  | Source Code Snippets | 17 |
|  | Outputs | 45 |
|  | References | 46 |

Introduction

This project will help you to implement all the concepts. Strategies, techniques, and technologies it’s that you are laming in the current ten. In this project. you will integrate all those concepts and mate live project by following all the aspects of software engineering and project management. You will implement various software engineering and project management concepts in each and every phase of project development. In addition. this project makes you go through the entire software project life cycle, which will help you to work in a real time application development environment.

Aspiring professionals need to apply their knowledge. skills and concepts that they have learned to develop a software project. In real Inv environment. software companies expect their employs to develop their software applications, which incorporate all the pluses of software engineering and project management. This project enables you to work in a simulated environment where you is ill work as a tram and develop a software application. This project provides a detailed study and implementation of project planning. analysts design. development. and documentation. After completing this project, you will get a confidence to apply your experience while developing projects in a software industry. You will appreciate and understand the need and usage of all the theoretical concepts that you have teamed in various semesters.

Aim and objective of the project

Aim:

The aim of this project is to design and develop a Web application for the User

Objective:

* To connect to the data base.
* Let the admin add update modify and maintain the database and the records.
* Provide a good user interface.
* Test the application.

Case study

Project requirement specification

For User:

* Create an User profile.
* Register the User profile.
* Authenticate the User profile through Username and password.
* Update the Userprofile.
* Archive the User profile.

Project analysis

Implementation and diagrams

Java has been one of the most popular programming languages for many years. Java is Object Oriented. However, it is not considered as pure object oriented as it provides support for primitive data types (like int, char, etc.) The Java codes are first compiled into byte code (machine independent code). Then the byte code is run on Java Virtual Machine (JVM) regardless of the underlying architecture. Java syntax is similar to C/C++. But Java does not provide low level programming functionalities like pointers. Also, Java codes are always written in the form of classes and objects.

Java is used in all kind of applications like Web Applications (Android is Java based), desktop applications, web applications, client server applications, enterprise applications and many more.

.

Code snippets

HibernateConfig.java

**package** com.coll.OnlineCollaboration.config;

**import** java.util.Properties;

**import** javax.sql.DataSource;

**import** org.springframework.boot.autoconfigure.EnableAutoConfiguration;

**import** org.springframework.boot.autoconfigure.orm.jpa.HibernateJpaAutoConfiguration;

**import** org.springframework.context.annotation.Bean;

**import** org.springframework.context.annotation.ComponentScan;

**import** org.springframework.context.annotation.ComponentScans;

**import** org.springframework.context.annotation.Configuration;

**import** org.springframework.jdbc.datasource.DriverManagerDataSource;

**import** org.springframework.orm.hibernate5.HibernateTransactionManager;

**import** org.springframework.orm.hibernate5.LocalSessionFactoryBean;

**import** org.springframework.transaction.annotation.EnableTransactionManagement;

**import** org.springframework.web.servlet.ViewResolver;

**import** org.springframework.web.servlet.view.InternalResourceViewResolver;

@Configuration

@ComponentScans(value= {@ComponentScan("com.coll.OnlineCollaboration"),

@ComponentScan ("model"),

@ComponentScan ("controller"),

@ComponentScan ("dao"),

@ComponentScan("service")})

@EnableAutoConfiguration(exclude= {HibernateJpaAutoConfiguration.**class**})

@EnableTransactionManagement

**public** **class** HibernateConfig {

**public** **static** **final** String ***DATABASE\_URL***="jdbc:mysql://localhost:3306/collaborate";

**public** **static** **final** String ***DATABASE\_DRIVER***="org.mysql.cj.jdbc.Driver";

**public** **static** **final** String ***DATABASE\_DIALECT***="org.hibernate.dialect.MySQLDialect";

**public** **static** **final** String ***DATABASE\_USERNAME***="root";

**public** **static** **final** String ***DATABASE\_PASSWWORD***="Rahul&amp;1802";

@Bean(name="dataSource")

**public** DataSource getDataSource()

{

DriverManagerDataSource dataSource=**new** DriverManagerDataSource();

dataSource.setDriverClassName(***DATABASE\_DRIVER***);

dataSource.setUrl(***DATABASE\_URL***);

dataSource.setUsername(***DATABASE\_USERNAME***);

dataSource.setPassword(***DATABASE\_PASSWWORD***);

**return** dataSource;

}

@Bean

**public** LocalSessionFactoryBean getSessionFactory()

{

LocalSessionFactoryBean sessionFactory=**new** LocalSessionFactoryBean();

sessionFactory.setDataSource(getDataSource());

sessionFactory.setPackagesToScan("com.coll.OnlineCollaboration");

Properties hibernateProperties=**new** Properties();

hibernateProperties.put("hibernate.dialect", ***DATABASE\_DIALECT***);

hibernateProperties.put("hibernate.show\_sql", "true");

hibernateProperties.put("hibernate.hbm2ddl.auto","update");

sessionFactory.setHibernateProperties(hibernateProperties);

**return** sessionFactory;

}

@Bean

**public** HibernateTransactionManager getTransactionManager()

{

HibernateTransactionManager txm=**new** HibernateTransactionManager();

txm.setSessionFactory(getSessionFactory().getObject());

**return** txm;

}

@Bean

**public** ViewResolver jspViewResolver()

{

InternalResourceViewResolver viewResolver=**new** InternalResourceViewResolver();

viewResolver.setPrefix("/views/");

viewResolver.setSuffix(".jsp");

**return** viewResolver;

}

}

User.java

**package** com.coll.OnlineCollaboration.model;

**import** java.io.Serializable;

**import** javax.persistence.Entity;

**import** javax.persistence.GeneratedValue;

**import** javax.persistence.GenerationType;

**import** javax.persistence.Id;

**import** org.springframework.stereotype.Component;

@Component

@Entity

**public** **class** User **extends** DomainResponse **implements** Serializable {

**private** **static** **final** **long** ***serialVersionUID***=1L;

@Id

@GeneratedValue(strategy=GenerationType.***IDENTITY***)

**private** **int** userId;

**private** String firstName;

**private** String LastName;

**private** String username;

**private** String password;

**private** String email;

**private** String role;

**private** String status;

**private** **boolean** isOnline;

**private** **boolean** enabled;

**public** User()

{

}

**public** User(String firstName, String lastName, String username, String password, String email, String role,

String status, **boolean** isOnline, **boolean** enabled) {

**super**();

**this**.firstName = firstName;

LastName = lastName;

**this**.username = username;

**this**.password = password;

**this**.email = email;

**this**.role = role;

**this**.status = status;

**this**.isOnline = isOnline;

**this**.enabled = enabled;

}

**public** **int** getUserId() {

**return** userId;

}

**public** **void** setUserId(**int** userId) {

**this**.userId = userId;

}

**public** String getFirstName() {

**return** firstName;

}

**public** **void** setFirstName(String firstName) {

**this**.firstName = firstName;

}

**public** String getLastName() {

**return** LastName;

}

**public** **void** setLastName(String lastName) {

LastName = lastName;

}

**public** String getUsername() {

**return** username;

}

**public** **void** setUsername(String username) {

**this**.username = username;

}

**public** String getPassword() {

**return** password;

}

**public** **void** setPassword(String password) {

**this**.password = password;

}

**public** String getEmail() {

**return** email;

}

**public** **void** setEmail(String email) {

**this**.email = email;

}

**public** String getRole() {

**return** role;

}

**public** **void** setRole(String role) {

**this**.role = role;

}

**public** String getStatus() {

**return** status;

}

**public** **void** setStatus(String status) {

**this**.status = status;

}

**public** **boolean** isOnline() {

**return** isOnline;

}

**public** **void** setOnline(**boolean** isOnline) {

**this**.isOnline = isOnline;

}

**public** **boolean** isEnabled() {

**return** enabled;

}

**public** **void** setEnabled(**boolean** enabled) {

**this**.enabled = enabled;

}

**public** **static** **long** getSerialversionuid() {

**return** ***serialVersionUID***;

}

}

DomainResponse.java

**package** com.coll.OnlineCollaboration.model;

**public** **class** DomainResponse {

**int** responeCode;

String responseMessage;

**public** DomainResponse()

{

**super**();

}

**public** DomainResponse(**int** responeCode, String responseMessage) {

**super**();

**this**.responeCode = responeCode;

**this**.responseMessage = responseMessage;

}

**public** **int** getResponeCode() {

**return** responeCode;

}

**public** **void** setResponeCode(**int** responeCode) {

**this**.responeCode = responeCode;

}

**public** String getResponseMessage() {

**return** responseMessage;

}

**public** **void** setResponseMessage(String responseMessage) {

**this**.responseMessage = responseMessage;

}

}

Blog.java

**package** com.coll.OnlineCollaboration.model;

**import** java.io.Serializable;

**import** java.time.LocalDate;

**import** java.util.List;

**import** javax.persistence.CascadeType;

**import** javax.persistence.Entity;

**import** javax.persistence.FetchType;

**import** javax.persistence.GeneratedValue;

**import** javax.persistence.GenerationType;

**import** javax.persistence.Id;

**import** javax.persistence.OneToMany;

**import** org.springframework.stereotype.Component;

**import** com.fasterxml.jackson.annotation.JsonManagedReference;

@Component

@Entity

**public** **class** Blog **implements** Serializable {

**private** **static** **final** **long** ***serialVersionUID***=1L;

@Id

@GeneratedValue(strategy=GenerationType.***IDENTITY***)

**int** blogId;

String blogTitle,blogContent;

LocalDate blogPosted;

String status;

**int** noOfLikes,noOfViews,noOfComments;

**int** userId;

String username;

@OneToMany(mappedBy="blog", fetch=FetchType.***EAGER***, cascade=CascadeType.***ALL***)

@JsonManagedReference

List<BlogComments> blogComments;

**public** Blog()

{

**super**();

}

**public** Blog(String blogTitle, String blogContent, LocalDate blogPosted, String status, **int** noOfLikes, **int** noOfViews,

**int** noOfComments, **int** userId, String username) {

**super**();

**this**.blogTitle = blogTitle;

**this**.blogContent = blogContent;

**this**.blogPosted = blogPosted;

**this**.status = status;

**this**.noOfLikes = noOfLikes;

**this**.noOfViews = noOfViews;

**this**.noOfComments = noOfComments;

**this**.userId = userId;

**this**.username = username;

}

**public** **int** getBlogId() {

**return** blogId;

}

**public** **void** setBlogId(**int** blogId) {

**this**.blogId = blogId;

}

**public** String getBlogTitle() {

**return** blogTitle;

}

**public** **void** setBlogTitle(String blogTitle) {

**this**.blogTitle = blogTitle;

}

**public** String getBlogContent() {

**return** blogContent;

}

**public** **void** setBlogContent(String blogContent) {

**this**.blogContent = blogContent;

}

**public** LocalDate getBlogPosted() {

**return** blogPosted;

}

**public** **void** setBlogPosted(LocalDate blogPosted) {

**this**.blogPosted = blogPosted;

}

**public** String getStatus() {

**return** status;

}

**public** **void** setStatus(String status) {

**this**.status = status;

}

**public** **int** getNoOfLikes() {

**return** noOfLikes;

}

**public** **void** setNoOfLikes(**int** noOfLikes) {

**this**.noOfLikes = noOfLikes;

}

**public** **int** getNoOfViews() {

**return** noOfViews;

}

**public** **void** setNoOfViews(**int** noOfViews) {

**this**.noOfViews = noOfViews;

}

**public** **int** getNoOfComments() {

**return** noOfComments;

}

**public** **void** setNoOfComments(**int** noOfComments) {

**this**.noOfComments = noOfComments;

}

**public** **int** getUserId() {

**return** userId;

}

**public** **void** setUserId(**int** userId) {

**this**.userId = userId;

}

**public** String getUsername() {

**return** username;

}

**public** **void** setUsername(String username) {

**this**.username = username;

}

**public** **static** **long** getSerialversionuid() {

**return** ***serialVersionUID***;

}

}

BlogComments.java

**package** com.coll.OnlineCollaboration.model;

**import** java.io.Serializable;

**import** java.time.LocalDate;

**import** javax.persistence.Entity;

**import** javax.persistence.GeneratedValue;

**import** javax.persistence.GenerationType;

**import** javax.persistence.Id;

**import** javax.persistence.JoinColumn;

**import** javax.persistence.ManyToOne;

**import** org.springframework.stereotype.Component;

**import** com.fasterxml.jackson.annotation.JsonBackReference;

@Component

@Entity

**public** **class** BlogComments **implements** Serializable {

**private** **static** **final** **long** ***serialVersionUID***=1L;

@Id

@GeneratedValue(strategy=GenerationType.***IDENTITY***)

**int** blogCommentId;

**int** userId;

String username;

String userProfileId;

String title;

**int** noOfLikes;

String blogComment;

LocalDate currentDate;

@ManyToOne

@JoinColumn(name="BlogId")

@JsonBackReference

Blog blog;

**public** BlogComments() {

**super**();

// **TODO** Auto-generated constructor stub

}

**public** BlogComments(**int** userId, String username, String userProfileId, String title, **int** noOfLikes,

String blogComment, LocalDate currentDate, Blog blog) {

**super**();

**this**.userId = userId;

**this**.username = username;

**this**.userProfileId = userProfileId;

**this**.title = title;

**this**.noOfLikes = noOfLikes;

**this**.blogComment = blogComment;

**this**.currentDate = currentDate;

**this**.blog = blog;

}

**public** **int** getBlogCommentId() {

**return** blogCommentId;

}

**public** **void** setBlogCommentId(**int** blogCommentId) {

**this**.blogCommentId = blogCommentId;

}

**public** **int** getUserId() {

**return** userId;

}

**public** **void** setUserId(**int** userId) {

**this**.userId = userId;

}

**public** String getUsername() {

**return** username;

}

**public** **void** setUsername(String username) {

**this**.username = username;

}

**public** String getUserProfileId() {

**return** userProfileId;

}

**public** **void** setUserProfileId(String userProfileId) {

**this**.userProfileId = userProfileId;

}

**public** String getTitle() {

**return** title;

}

**public** **void** setTitle(String title) {

**this**.title = title;

}

**public** **int** getNoOfLikes() {

**return** noOfLikes;

}

**public** **void** setNoOfLikes(**int** noOfLikes) {

**this**.noOfLikes = noOfLikes;

}

**public** String getBlogComment() {

**return** blogComment;

}

**public** **void** setBlogComment(String blogComment) {

**this**.blogComment = blogComment;

}

**public** LocalDate getCurrentDate() {

**return** currentDate;

}

**public** **void** setCurrentDate(LocalDate currentDate) {

**this**.currentDate = currentDate;

}

**public** Blog getBlog() {

**return** blog;

}

**public** **void** setBlog(Blog blog) {

**this**.blog = blog;

}

**public** **static** **long** getSerialversionuid() {

**return** ***serialVersionUID***;

}

}

IUserDao.java

**package** com.coll.OnlineCollaboration.dao;

**import** java.util.List;

**import** com.coll.OnlineCollaboration.model.\*;

**public** **interface** IUserDao {

List<User> userListbyStatus(String status);

List<User> getAllUsers();

User getUserById(**int** userId);

User getUserByUsername(String username);

User validateUser(User user);

**boolean** addUser(User user);

**boolean** updateUser(User user);

**boolean** deleteUser(**int** userId);

**boolean** deactivateUser(**int** userId);

**boolean** updateUserProfile(String file,**int** userId);

}

IBlogDao.java

**package** com.coll.OnlineCollaboration.dao;

**import** java.util.List;

**import** com.coll.OnlineCollaboration.model.Blog;

**public** **interface** IBlogDao {

List<Blog> getAllBlogs();

List<Blog> getBlogsByStatus(String status);

List<Blog> getUserBlogs(**int** id);

List<Blog> mainList();

Blog getBlogById(**int** blogId);

**boolean** addBlog(Blog blog);

**boolean** updateBlog(Blog blog);

**boolean** deleteBlog(Blog blog);

}

IBlogCommentsDao.java

package com.coll.OnlineCollaboration.dao;

import java.util.List;

import com.coll.OnlineCollaboration.model.BlogComments;

public interface IBlogCommentsDao {

List<BlogComments> getAllBlogComments();

BlogComments getBlogCommentsById(int blogComemntId);

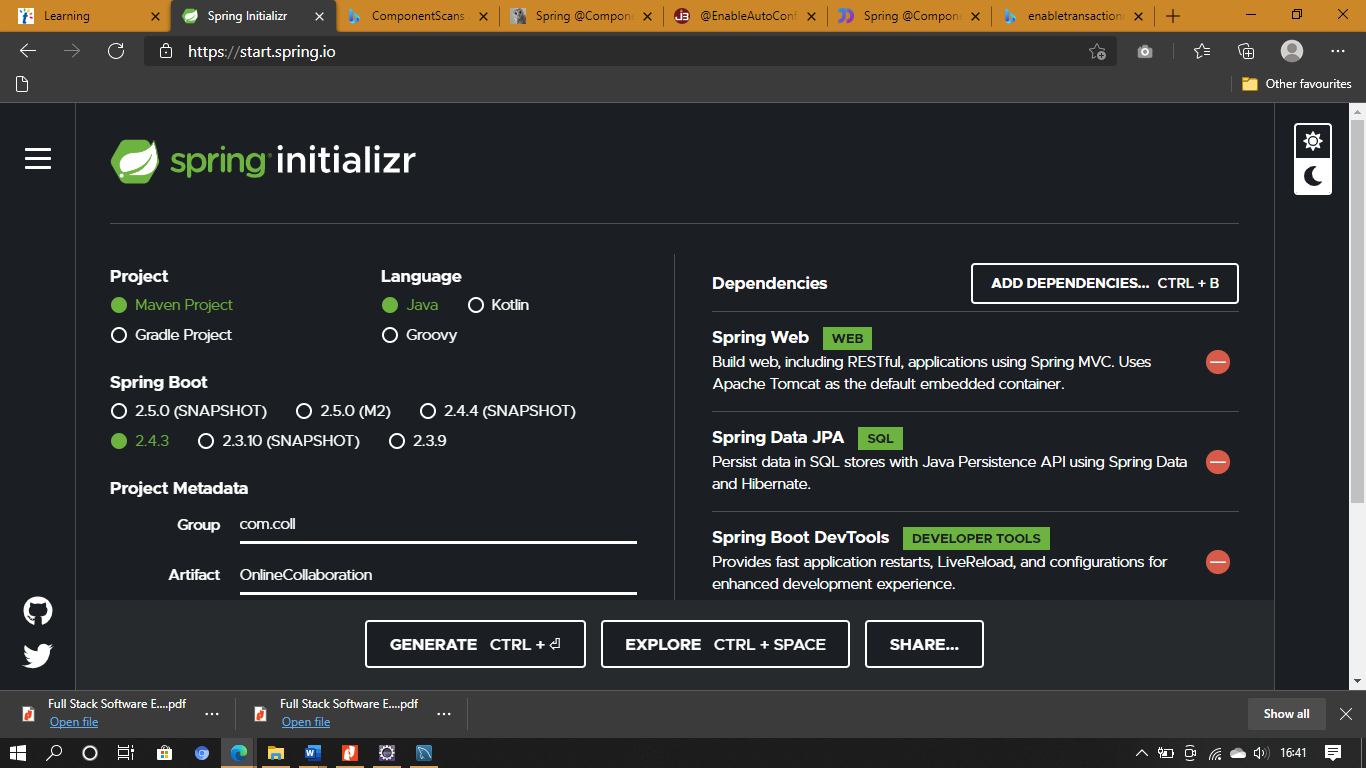
boolean addBlogComments(BlogComments blogComments);

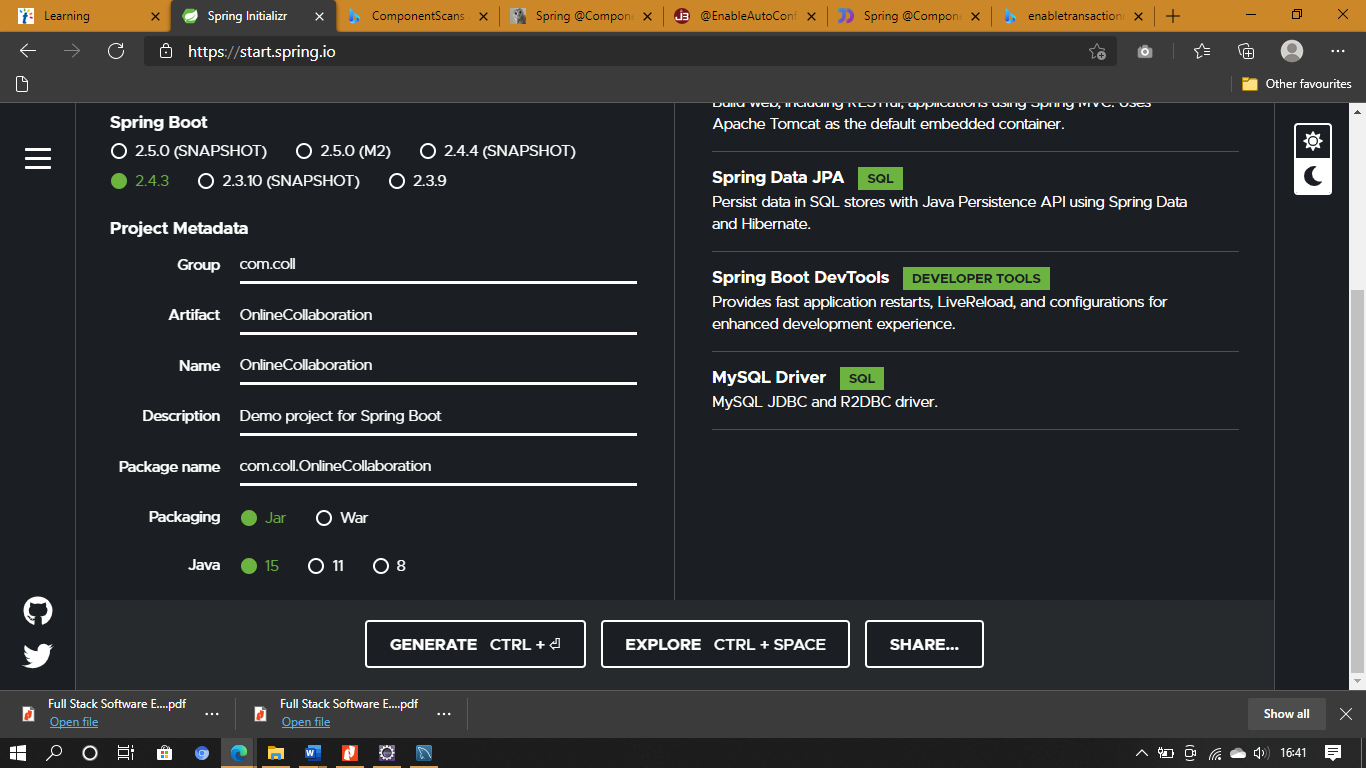
boolean updateBlogComments(BlogComments blogComments);

boolean deleteBlogComments(BlogComments blogComments);

Screenshot

SpringInitializer





References

* NIIT Project Guide
* Google