Indian Institute of Technology, Bombay Ekalavya Summer Internship Programme - 2018



Under the guidance of

Prof. Deepak B. Phatak

Project Report

Spoken Tutorial Project: School Resource System

Contributor: Mr. Abhishek Dutta

Mentors: Mrs. Nancy Varkey and Ms. Kirti Ambre

Acknowledgement:

The internship opportunity I had with **Indian Institute of Technology Bombay,** was a great chance for learning and professional development. I consider myself a very lucky individual as I was provided with an opportunity to be a part of it. I am also grateful for having a chance to meet so many wonderful people and professionals who led me through this internship period.

I am using this opportunity to express my deepest gratitude and special thanks to the **Prof. Deepak B. Phatak** of Computer Science and Engineering Department, who in spite of being extraordinarily busy with his duties, took time out to guide and keep me on the correct path and allowed me to carry out my project at this esteemed organization and extending resources and facilities during the training.

I express my deepest thanks to the mentors of *Spoken Tutorial Project:* Mrs. Nancy Varkey, Ms. Kirti Ambre and other staff members for taking part in discussions, giving necessary advices and guidance and arranging all the necessary facilities. I choose this moment to acknowledge their contribution gratefully.

It feel myself privileged and would like to place on record my best regards, deepest sense of gratitude to all the colleagues of Spoken Tutorial team, friends, hostel staff and all indirectly associated persons for their careful guidance and help which kept my mind and body sound during this period.

I perceive this opportunity as a big milestone in my career development. I will strive to use the gained skills and knowledge in the best possible way, and I will continue to work on their improvement, in order to attain desired career objectives. Hope to continue cooperation with all of you in the future.

Ekalavya Summer Internship 2018

Project Approval Certificate Department of Computer Science and Engineering Indian Institute of Technology, Bombay

The project entitled *School Resource System*, submitted by Mr. Abhishek Dutta

is approved for Ekalavya Summer Internship-2018, from 16th May 2018 to 6th July 2018, at Department of Computer Science and Engineering, IIT Bombay.

Mrs. Nancy Varkey and Ms. Kirti Ambre

(Mentor in-charge)

Place: IIT Bombay, Mumbai.

Date: 6 July, 2018

Declaration

I declare that this written submission report represents my best understanding of the topic in my own words and content from other sources have been mentioned clearly as and when required. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be a cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

3rd July, 2018 Abhishek Dutta

Reg. No.: 182929

Contents

1 Introduction	7
1.1 Overview	
1.2 Purpose	
1.3 Scope	
2 Requirements specification	9
2.1 About School Resource System	
2.2 Technical aspects	
3 System Design	17
3.1 Database architecture	
3.2 Work process	
4 Implementation	22
4.1 Database connectivity	
4.2 Encoding password	
4.3 AJAX, updating parts of web page	
4.4 Downloading zip file	
4.5 Unzipping file	
5 Testing and deployment	29
5.1 How to Run project on the Eclipse IDE?	
5.2 Results	
	~ 0
6 Future Work	50
7 Conclusion	51
8 References	52
Appendix	53

Abstract

School Resource System under the Spoken Tutorial project is a website which provides ICT (Information and Communication Technologies) tools for teaching/learning and resource materials for school teachers and students. It contains content in the form of various resource types, such as videos, concept maps, images, PhETs (Physics Education Technologies) and links to various web pages. It consist of three interfaces i.e. User, Contributor (or teacher) and Admin, for better management and use.

Website contains filter search as well as keyword search, so that minimum time is required to access a resource. The resources are of four types, i.e. Concept maps (or Mind maps), PhETs, tutorials and links. These resources are all open source.

Chapter 1 Introduction

1.1 Overview

Spoken Tutorial is an initiative funded by the National Mission on Education through Information and Communication Technology (ICT) launched by the Ministry of Human Resources and Development (MHRD), Government of India and is being developed by a dedicated team at IIT Bombay. The Principal Investigator, *Prof. Kannan Moudgalya*, heads this project at IIT Bombay. The project aims to promote and improve employability potential and to help bridge the digital divide. The Spoken Tutorial project has already trained many students across many schools and colleges on FOSS technologies through workshops across the country.

The objective of Spoken Tutorial project is to spread the knowledge of technology and Free Open Source Software (FOSS) across the country to the ones who lack access and opportunities to learn.

School Resource System is a fresh branch of Spoken Tutorial initiative. It is primarily focused on school students and mentors. It thus, comprises of tools and resources for those students and mentors. The ICT tools such as video tutorials (Spoken Tutorials) and PhETs enhances the learning experience and helps students understand the topics clearly and efficiently, and these can be accessed repeatedly, anytime and anywhere.

1.2 Purpose

The purpose is to reach as many students(primarily school going) and associated professionals as possible and provide them all the necessary tools and knowledge so that they may not lag behind by any means and can get equal opportunity to learn and grow, anytime and anywhere.

1.3 Scope

Though it is a student first website but it can be equally advantageous to other users willing to enrich themselves with knowledge related to schools for their respective fields.

This can be useful to teachers as well as associated professionals, as they can find various tools to communicate to students more clearly and efficiently. They are also welcomed to contribute resources so that their ideas can reach a large number of students which would have otherwise been restricted to a limited number of students.

1.4 Background and Motivation

School Resource System provides unique advantages, the most prominent being the ability to provide personalized attention to students at their own convenience. This is especially critical for those students who cannot afford private sessions from experts or who live in rural areas where such facilities are not available. Usually, this is only possible when a highly skilled mentor delivers to students interacting directly. Often such kind of setup is hard to find and if it exists, individualized attention is quite difficult.

Another important advantage is that people living in smaller towns and cities can get access to the best possible learning resources from across the world, free of cost.

Chapter 2 Requirements specification

2.1 About School Resource System

This provides support to traditional teaching methodologies which student can access anytime and anywhere on the basis of their convenience and requirements. The system is being developed using *water fall* model of SDLC (Software Development Life Cycle). Basically, it features three interfaces with following specifications:

• User interface:

Easy access of resources (i.e. Links, PhETs, Tutorials and Concept maps) through

- a. Filter search depending on the selection of grades, subjects and topics
- b. Topic search depending on the keyword of topics

• Contributor interface:

Contains user interface with 'contribute' choice. However, a user first needs to register him/herself as contributor before he/she can contribute. Contributor can contribute any of the four available resources of a particular topic. Also s/he can request to add new subject resource topic to the admin to enhance the reach of the system and make it more dynamic.

• Admin interface

Has all the capabilities of user and contributor together with capability to accept

- a. user request to become contributor
- b. contributor request to accept resources or add/subtract resources/topics/subjects/grades or to make some other changes making the website for flexible.

2.2 Technical aspects

2.2.1 System requirement(s)

Ubuntu 16.04 LTS is a complete Linux operating system, freely available with both community and professional support. The Ubuntu community is built on the ideas enshrined in the Ubuntu Manifesto: that software should be available free of charge, that software tools should be usable by people in their local language and despite any disabilities, and that people should have the freedom to customize and alter their software in whatever way they see fit.

2.2.2 Software requirement(s)

Following technologies have been used throughout the development of this project (if and otherwise stated):

• Client-side:

1. Find HTML 5:

HTML stands for Hyper Text Markup Language, which is the most widely used language on the Web to develop web pages. HTML was created by Berners-Lee in late 1991. HTML5 is the latest version of Hypertext Markup Language, the code that describes web pages.

It has been designed to deliver almost everything you'd want to do online without requiring additional software such as browser plug-ins. It does everything from animation to apps, music to movies, and can also be used to build incredibly complicated applications that run in your browser.

It is not proprietary, so you don't need to pay royalties to use it. It's also cross-platform, which means it doesn't care whether you're using a tablet or a smart phone, notebook or ultra book or a Smart TV.



2. Cascading Style Sheets (CSS):

CSS stands for Cascading Style Sheets. CSS describes how HTML elements are to be displayed on screen, paper, or in other media. It can control the layout of multiple web pages all at once. External style sheets are stored in CSS files. CSS 3 is being used in the current project.



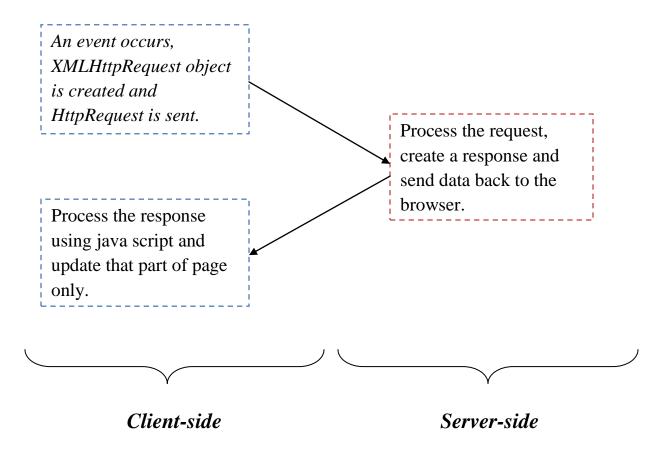
3. Java Script:

Java Script is a dynamic computer programming language. It is lightweight and most commonly used as a part of web pages, whose implementations allow client-side script to interact with the user and make dynamic pages.

It is an interpreted programming language with objectoriented capabilities.



Ajax is a set of web development techniques using many Web technologies on the client side to create asynchronous web applications. With Ajax, Web applications can send and retrieve data from a server asynchronously, without interfering with the display and behavior of the existing page. Ajax is not a single technology, but rather a group of technologies.



(Fig 2.1. Ajax work flow)

Ajax working

- 1. An event occurs in a web page.
- 2. An XMLHttpRequest object is created by JavaScript
- 3. The XMLHttpRequest object sends a request to a web server
- 4. The server processes the request
- 5. The server sends a response back to the web page
- 6. The response is read by JavaScript
- 7. Proper action is performed at appropriate position by JavaScript without affecting other places.

(Ref. www.w3school.com)

• Server side:

I. J2EE

Java Platform, Enterprise Edition (Java EE) is the industry standard for developing portable, robust, scalable and secure server-side Java applications. Building on the solid foundation of the Standard Edition, Java EE provides web services, component model, management, and communications APIs that make it the industry standard for implementing enterprise-class web applications.

Following is the java version used for the project.

java version "1.8.0_171"

Java(TM) Runtime Environment (build 1.8.0_171-b11)

II. JSP

Java Server Pages (JSP) technology enables web developers and designers to rapidly develop and easily maintain, information-rich, dynamic Web pages that leverage existing business systems. As part of the Java technology family, JSP technology enables rapid development of Web-based applications that are platform independent. JSP technology separates the user interface from content generation, enabling designers to change the overall page layout without altering the underlying dynamic content.

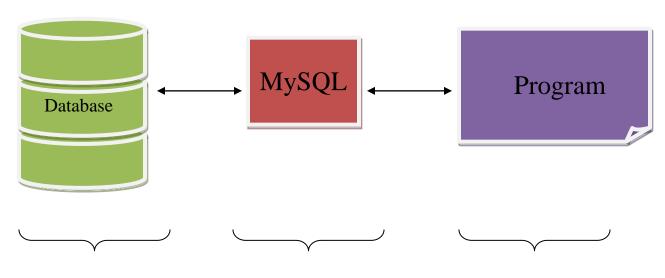
III. Servlets

Servlets are the Java platform technology of choice for extending and enhancing Web servers. Servlets provide a component-based, platform-independent method for building Web-based applications, without the performance limitations of CGI programs. It is the first web based technology.

• Database



- ✓ MySQL is a database management system(DBMS).
- ✓ MySQL databases are relational.
- ✓ MySQL software is Open Source.
- ✓ The MySQL Database Server is very fast, reliable, scalable, and easy to use.
- ✓ MySQL Server works in client/server or embedded systems.



Physical layer

Logical layer(DBMS)

Presentation layer

(Fig2.2: Three tier architecture)

MySQL Workbench

MySQL Workbench is a visual database design tool that integrates SQL development, administration, database design, creation and maintenance into a single integrated development environment for the MySQL database system.

Eclipse

Eclipse is an integrated development environment (IDE) used in computer programming, and is the most widely used Java IDE. It contains a base workspace and an extensible plug-in system for customizing the environment. Eclipse is written mostly in Java and its primary use is for developing Java applications, but it may also be used to develop applications in other programming languages via plug-ins

Eclipse software development kit (SDK) is free and open-source software, released under the terms of the Eclipse Public License.

Eclipse Luna has been used in this project.

Chapter 3 System Design

This is the blueprint of the system to facilitate the process of implementation. After understanding the requirements of the system it is necessary to have proper planning just like drawing the layout of structure before constructing it. Following pages determine data base model that has been used and the workflow to construct this project.

3.1 Database architecture

3.1.1 Entity Relation Diagram (E R D)

An Entity Relationship (E R) Diagram is a type of flowchart that illustrates how *entities* such as people, objects or concepts relate to each other within a system. E R Diagrams are most often used to design or debug relational databases in the fields of software engineering, business information systems, education and research. Also known as ERDs or ER Models, they use a defined set of symbols such as rectangles, diamonds, ovals and connecting lines to depict the interconnectedness of entities, relationships and their attributes. They mirror grammatical structure, with entities as nouns and relationships as verbs.

11 entities(tables) having certain properties(attributes) connected through relation as shown in the schema diagram and in the following entity-relation diagram.

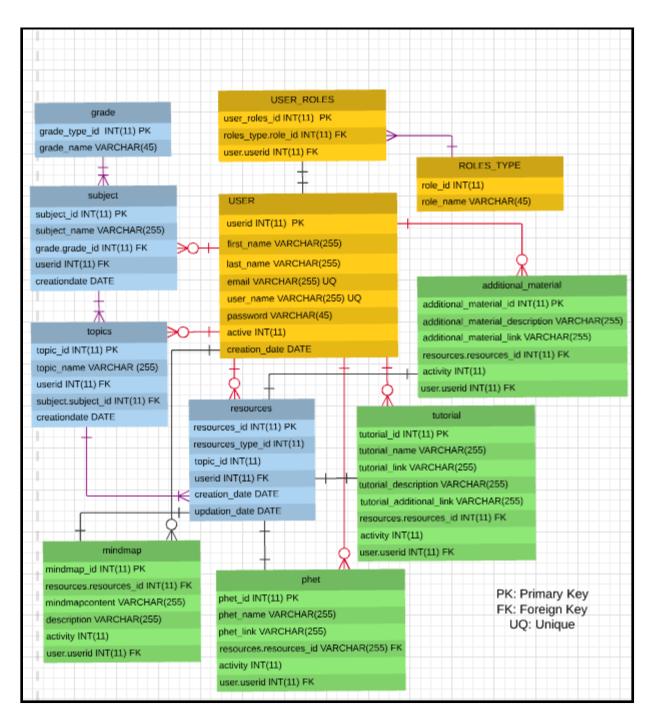


Fig. Schema diagram 3.1(relation shown using crow's foot notation)

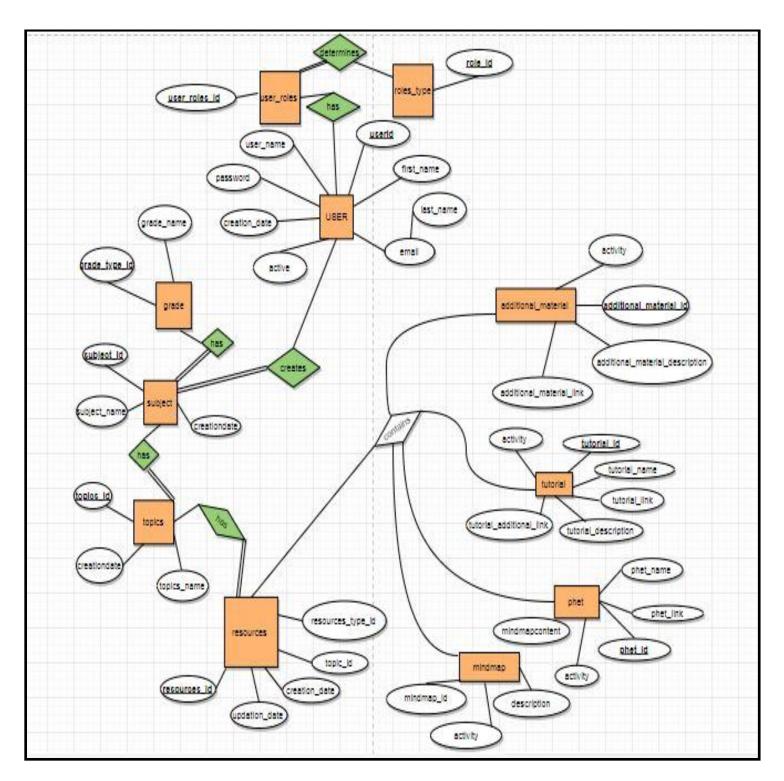
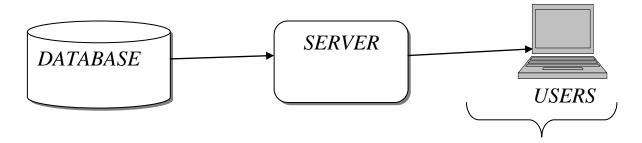
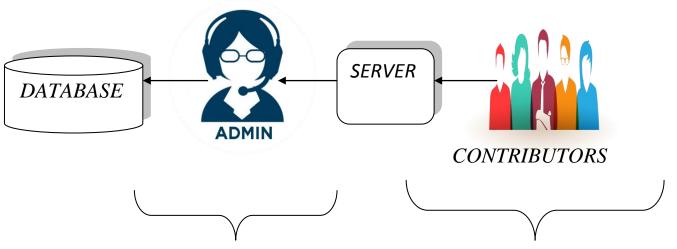


Fig 3.2: Entity Relation Diagram

3.2 Work Process



User can view the available resources



Admin reviews the resources contributed by contributors and determines whether it should be included in the database or needs improvement.

contributors provide resources like links PhETs, tutorial, etc.

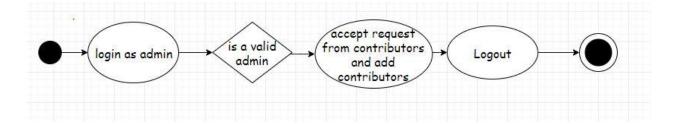


Fig. 3.3: Activity diagram(admin)

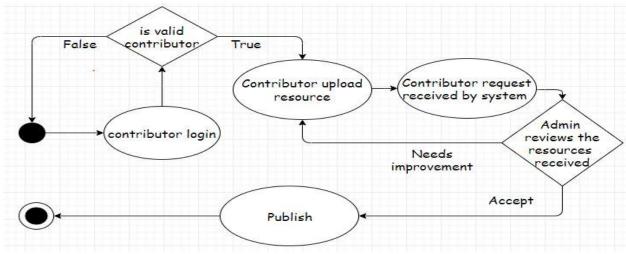


Fig.3.4: Activity diagram(contributor)

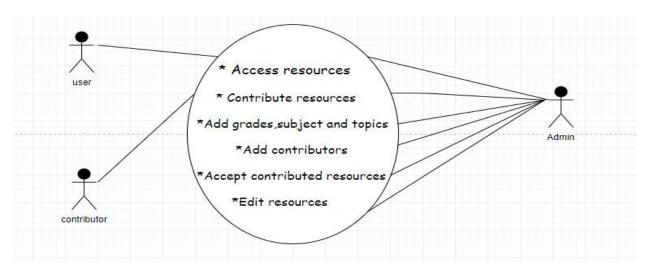


Fig. 3.5: Use case diagram

Chapter 4 Implementation

System design is being implemented following the blue print created in the previous section. Certain challenges that were faced during creation of the project and how they were tackled, are as follows:

4.1 Database connectivity

```
k%@ page import="java.sql.*" %>

connection con;
Statement stat;
ResultSet rs;
ResultSet metaData md;
%>

cw
Class.forName("com.mysql.jdbc.Driver");
con=DriverManager.getConnection("jdbc:mysql://localhost:3306/school_resource_system", " ", " ");
stat=con.createStatement();
%>
```

Fig 4.1.: database connectivity

getConnection Method of class DriverManager helps to establish connection with MySQL database named school_resources_system.

Moreover, objects of *Connection, Statement, ResultSet* and *ResultSetMetadata* are created which will be used in further procession.

<u>Advantage</u>

This *connection.jsp* can be included in a JSP file requiring connection with the database and helps to reduce the redundancy of writing same code again and again. Also it helps to manage the code more efficiently and clearly. Similar code can be written to close the connection.

4.2 Encoding password

```
String u_nm=rq.getParameter("u_nm");
String pass=rq.getParameter("pass");
byte[] unencodedPassword = pass.getBytes();
MessageDigest msdg = null;
try {
msdg = MessageDigest.getInstance("MD5");
} catch (Exception e) {}
msdg.reset();
msdg.update(unencodedPassword);
byte[] encodedPassword = msdg.digest();
StringBuffer buf = new StringBuffer();
for (int i = 0; i < encodedPassword.length; i++) {</pre>
if (((int) encodedPassword[i] & 0xff) < 0x10) {</pre>
buf.append("0");
buf.append(Long.toString((int) encodedPassword[i] & 0xff, 16));
String psw=buf.toString();
session=rq.getSession();
rs=stat.executeQuery(" select userid,user_name,password from user where user_name='" + u_nm + "' and password='" + psw + "' ");
if(rs.next())
            Integer u_id=rs.getInt("userid");
                session.setAttribute("unm",u_nm);
                session.setAttribute("uid",u_id);
                res.sendRedirect("jsp/user-home.jsp");
else
                session.invalidate();
                pw.println(" <head><meta http-equiv=refresh content='3;URL=jsp/login.jsp?abc=inv' /></head> ");
```

Fig. 4.2: Encoding password

The MD5 algorithm is used to generate the hash function producing a 128-bit hash value. MessageDigest class uses the method getInstance to trigger the MD5 algorithm which is used to encode the un-encoded password stored in the variable named as byte array.

4.3 AJAX, updating parts of a web page

```
<script language="javascript" type="text/javascript">
    var xmlHttp
    var xmlHttp
    function showSubject(str){
    if (typeof XMLHttpRequest != "undefined"){
    xmlHttp= new XMLHttpRequest();
    else if (window.ActiveXObject){
    xmlHttp= new ActiveXObject("Microsoft.XMLHTTP");
    if (xmlHttp==null){
    alert("Browser does not support XMLHTTP Request")
    return:
    var url="filterbox-subject.jsp";
    url +="?count=" +str;
    xmlHttp.onreadystatechange = subjectChange;
    xmlHttp.open("GET", url, true);
    xmlHttp.send(null);
    function subjectChange(){
    if (xmlHttp.readyState==4 || xmlHttp.readyState=="complete"){
    document.getElementById('subject').innerHTML=xmlHttp.responseText
```

Fig 4.3.: Updating part of web page.

variable = new XMLHttpRequest(): The XMLHttpRequest object can be used to exchange data with a server behind the scenes.

The **readyState** property holds the status of the XMLHttpRequest.

readyState==4 means request is finished and response is ready.

The **onreadystatechange** property defines a function to be executed when the readyState changes.

The **status** property and the **statusText** property holds the status of the XMLHttpRequest object.

4.4 downloading zip file at appropriate location

Fig 4.4.: Upload zip file on server

The file gets uploaded at the location on the server as mentioned in the deployment descriptor named 'mindmap-upload' under the 'param-name' tag.

4.5 unzipping file

```
//unzipping using iteration.
ZipFile zipFile = new ZipFile("/home/dutta/workspace/skool resource system/WebContent/static/resource/4/"+str);
Enumeration<?> enu = zipFile.entries();
while (enu.hasMoreElements()) {
        ZipEntry zipEntry = (ZipEntry) enu.nextElement();
        String name = zipEntry.getName();
        long size = zipEntry.getSize();
        long compressedSize = zipEntry.getCompressedSize();
        System.out.printf("name: %-20s | size: %6d | compressed size: %6d\n",
                        name, size, compressedSize);
//unzip directory within directory
        File file = new File("/home/dutta/workspace/skool resource system/WebContent/static/resource/4/"+name);
        if (name.endsWith("/")) {
                file.mkdirs();
                continue:
        }
        File parent = file.getParentFile();
        if (parent != null) {
                parent.mkdirs();
        InputStream is = zipFile.getInputStream(zipEntry);
        FileOutputStream fos = new FileOutputStream(file);
        byte[] bytes = new byte[1024];
        int length;
        while ((length = is.read(bytes)) >= 0) {
                fos.write(bytes, 0, length);
        is.close();
        fos.close();
zipFile.close():
System.out.println("unzip file created Successful");
```

Fig. 4.5: Unzipping zipped file.

Unzipping of file is done using iterative method in which directory inside directory is also unzipped.

zip file is selected from the location and stored in the variable named *zipFile*

Using Enumeration, zip files entries are being generated one by one. A new directory is being created and all the entries are being stored.

4.6 deleting file or directory

```
}
       try{
                len=str.indexOf(".zip");
                directory name=str.substring(0,len);
                System.out.println(len);
//deleting zip file
                File zipfile = new File("/home/dutta/workspace/skool resource system/WebContent/static/resource/4/"+str);
                if(zipfile.delete()){
                        System.out.println(zipfile.getName() + " is deleted!");
                }else{
                        System.out.println("zip Delete operation is failed.");
//deleting directory
                System.out.println(directory name);
                Path directory = Paths.get("/home/dutta/workspace/skool resource system/WebContent/static/resource/4/"+directory name);
                Files.walkFileTree(directory, new SimpleFileVisitor<Path>() {
                   @Override
                   public FileVisitResult visitFile(Path file, BasicFileAttributes attributes) throws IOException {
                       Files.delete(file); // this will work because it's always a File
                       return FileVisitResult.CONTINUE;
                   @Override
                   public FileVisitResult postVisitDirectory(Path dir, IOException exc) throws IOException {
                       Files.delete(dir); //this will work because Files in the directory are already deleted
                       return FileVisitResult.CONTINUE:
               });
       }catch(Exception e){
                e.printStackTrace();
       }
```

Fig. 4.6: deleting directory

File location is fetched and then zip file is being deleted and then unzipped directory of that zip file is being deleted. In this process directory inside directory or file inside directory is deleted and on success returns to desired location or else throws exception.

4.7 updating the database with new value

```
k%@ include file="./../connection.jsp" %>

K*

String nm=request.getParameter("phtnm");

String lnk=request.getParameter("phtlnk");

String srid=request.getParameter("rid");

Integer rid=Integer.parseInt(srid);

Integer uid=(Integer) session.getAttribute("uid");

Integer actv=0;

String qry="insert into phet(phet_name,phet_link,resources_id,activity,userid) values ('"+ nm +"','"+ lnk +"',"+ rid +","+ actv +","+ uid +")"

stat.executeUpdate(qry);

%>

<br/>
<br/>
<br/>
<br/>
<br/>
<br/>
<br/>
<meta http-equiv=refresh content="0;URL=./../user-home.jsp" />
```

Fig. 4.7: inserting values in database.

Following MySQL query is being used to implement the insertion of data to the database.

```
INSERT INTO table_name (column1, column2, column3, ...) VALUES (value1, value2, value3, ...);
```

executeUpdate method is used to execute the above query and apply the necessary changes.

Chapter 5 Testing and deployment

All the units developed in the implementation phase are integrated into a system after testing of each unit. The software designed, needs to go through constant software testing to find out if there are any flaw or errors and has to be fixed.

All the possible errors are corrected and necessary changes are made according to the requirements. Once the functional and non-functional testing is done, the system is deployed on local server for testing of all the functional units on the basis of requirements.

Following sections show the output of testing and deployment that are being received after deploying on the local server. The first task is to run the project.

5.1 Run project on the Eclipse IDE

- Start the Eclipse IDE and import the project school_resource_system
- Right click on the project and select run from the dropdown menu.
- Select run on server option from the new dropdown menu that appears

5.2 Results received

5.2.1 The Front page

This page is the beginning of a never ending journey. The page is kept simple according to the requirement and also to the best of our ability. It comprises of header consisting of the heading, a menu bar (which in turn, contains filter search option, keyword search through topic name, about, login and register options), description (like availability of resources), samples, about us and a footer.



Fig. 5.1: Front page (desktop image, as it appears on opening the site, may be subject to change in due course)

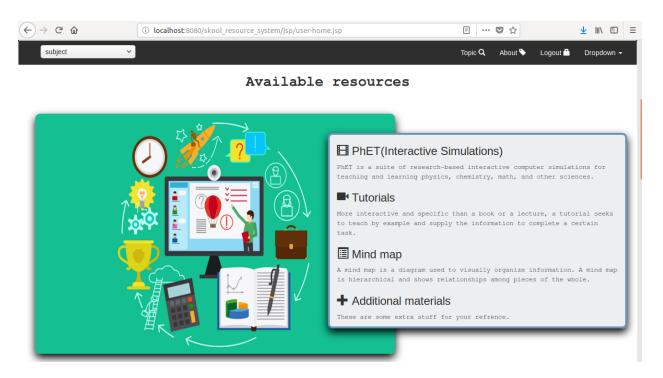


Fig. 5.2: continued image of the front page(on scrolling, available resources).

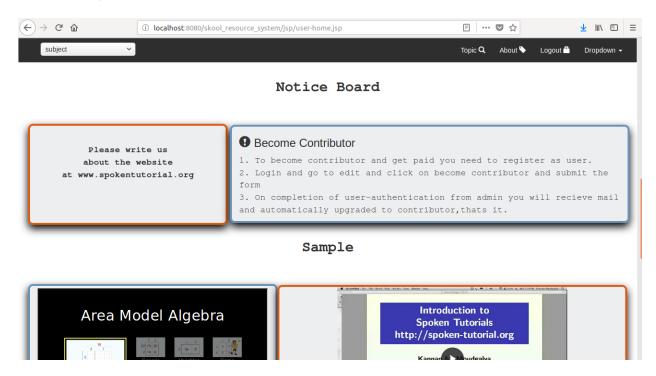


Fig. 5.3: continued image of the front page(on scrolling, news).

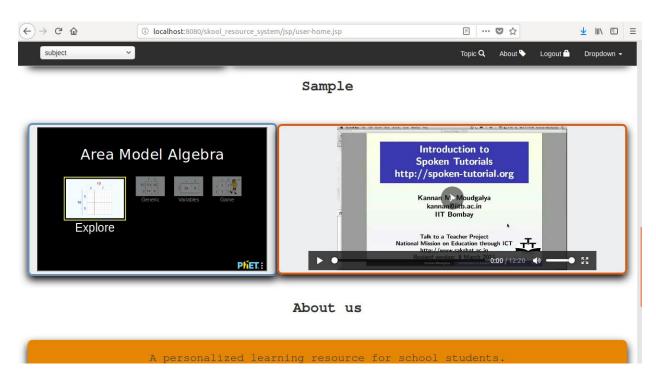


Fig. 5.4: continued image of the front page(on scrolling, samples).

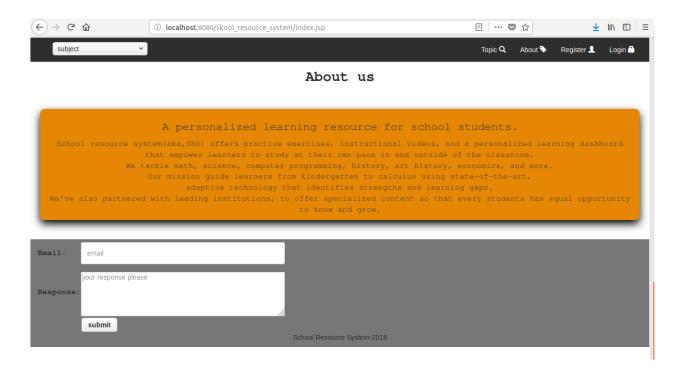


Fig. 5.5: continued image of the front page(on scrolling, about us).

5.2.2 Performing search operations

In order to perform optimal search and get the required resources with minimum hassle we have two types of search options viz. filter search (in which, you select resources from the dropdown options that appear) and topic search (which is basically keyword search of available topics in the database). Following are the steps to use both of them:

a. Filter search

Select a **Subject** from the dropdown option in the menu bar. A new dropdown appears. Select grade from this dropdown.

Next, click on search button and you will be redirected to a new page which has more options like topics and resource options. Select those and hit enter. You will get your desired results if it is available.

b. Keyword search

Click on the topic search in the menu bar, dialogue box appears seeking your response to enter key word to search for.

Next, you type the key word of the topic to search and if such matches with that in the database displays in the drop down menu which can be selected and will take you to the selected resource page.

Following section shows the process of doing filter search:



Fig.5.6: Filter search(select subject)



Fig. 5.7: Filter search(select grade)

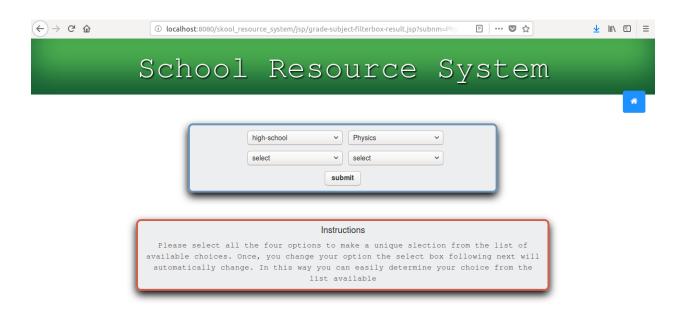


Fig. 5.8:Filter search options

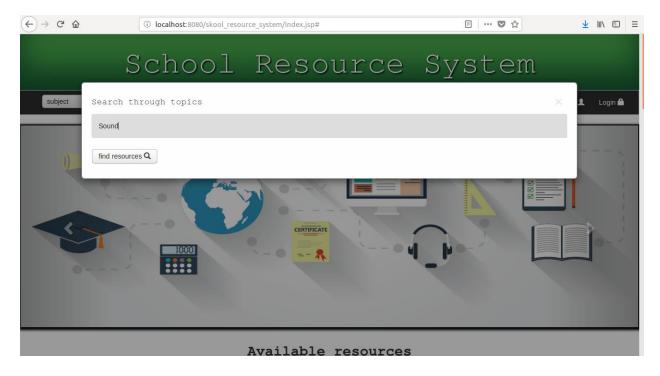


Fig. 5.9: Topic search(Keyword search)

5.2.3 View resources

Following shows glimpses of resources available, like PhETs, tutorials, etc. At present we have in total four types of resources only.



localhost:8080/skool_resource_system/jsp/filterbox_redirect_to_content.jsp?val=11

Fig 5.10: Additional material(links/pdf/txt)

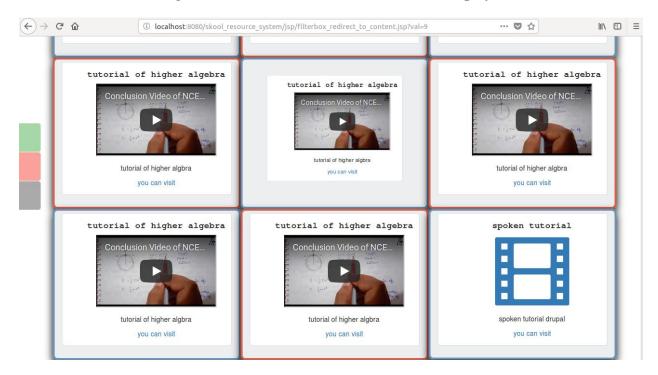


Fig 5.11: Tutorials(includes embedded videos from you tube and other sources as well)

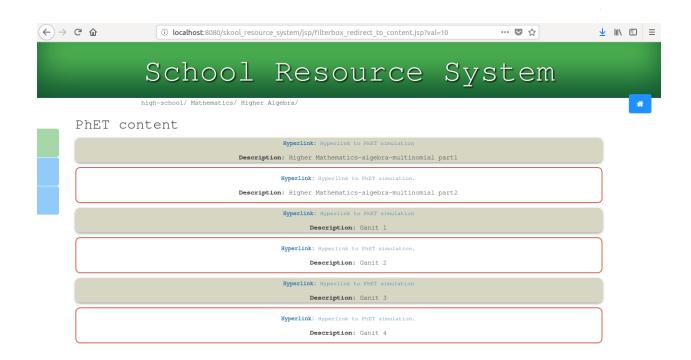


Fig. 5.12: List of PhETs available for particular topic

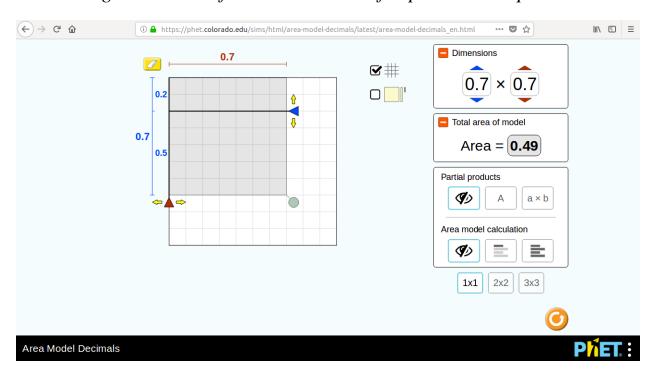


Fig. 5.13: PhET example



Fig. 5.14: List of mind maps/concept maps available.

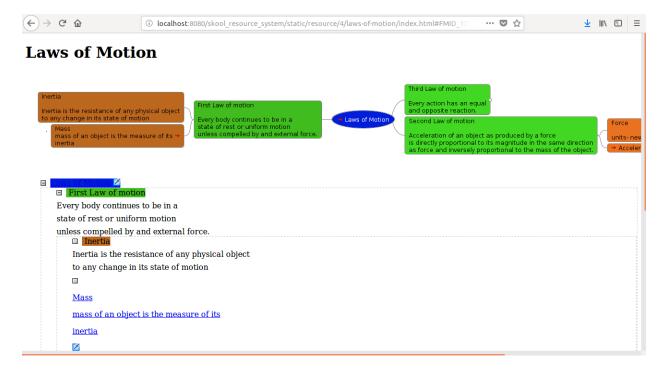


Fig. 5.15: Mind map example(these are basically, HTML files).

5.2.4 Login and edit profile

User can moreover register itself, he/she can then login to his/her profile.

To become a contributor we will have an option to become a contributor in the top left corner. A contributor request will then be sent and once the admin approves his identity he/she will become contributor. Again if he open his profile, he will have more options in the drop down menu like to contribute.

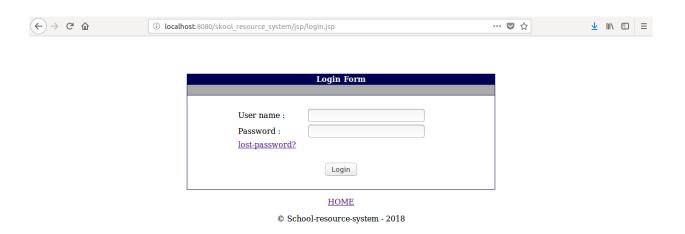


Fig. 5.16: Login page



Fig. 5.17: Edit profile

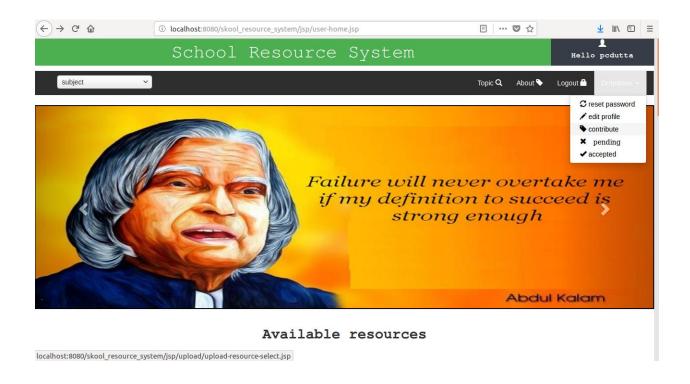


Fig. 5.18: Logged in user page(contributor interface)

5.2.5 Upload

School resource system is built on the ground of contributions. It is the contributor who will create innovative resources which can be applied to enhance the learning experience of the students. Contributor has the option to contribute the four available types of resources. Contributor will perform the same filter search to get the particular resource as one will perform to search a particular resource. However, he can request admin to add subject or topic in the database so that he can contribute to that also.

Following are the list of uploading pages of different types of resources:

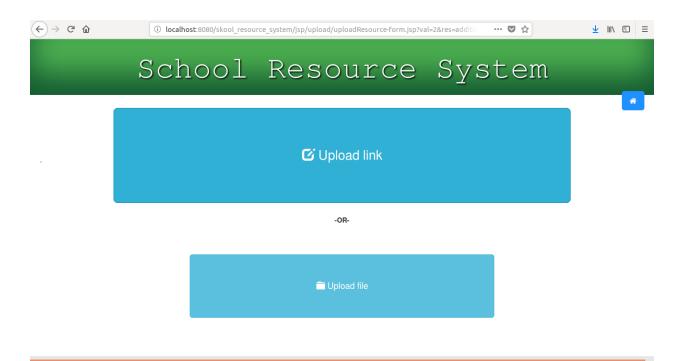


Fig. 5.19: Upload link or file for additional material

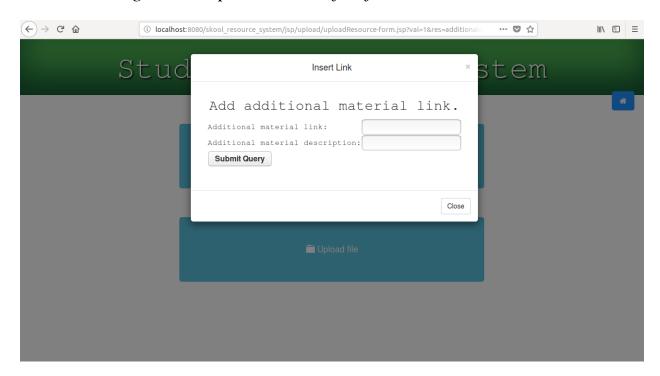


Fig. 5.20: Insert link in the additional material of particular a topic

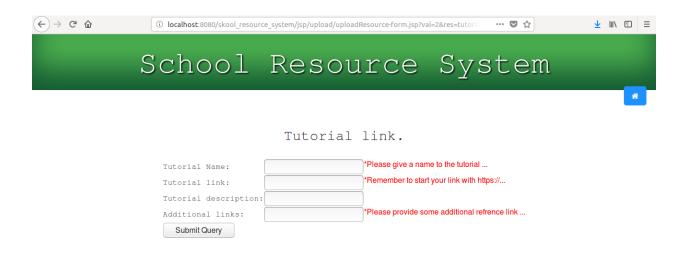


Fig. 5.21: Insert tutorial link



Fig 5.22: Insert PhET link



Fig. 5.23: Upload zip file of the concept map/mindmap

5.2.6 Admin interface

Admin has the responsibility to accept request from the user to become contributor and also from the contributors to accept their resources. The admin verifies whether the resources are authentic and are made on their true spirit. According to the judgment he/she accepts(i.e.) publish or click on the need improvement button. Contributor gets the response accordingly.

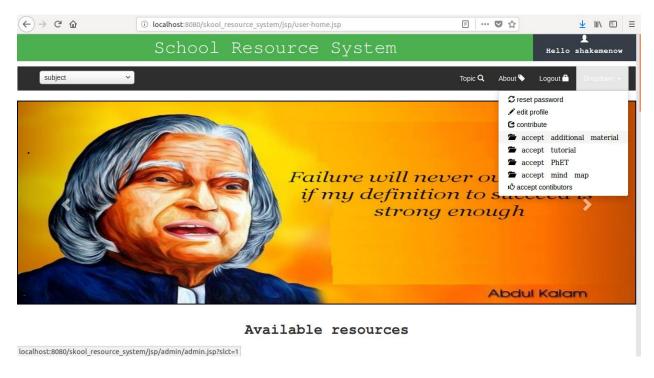


Fig. 5.24: Admin interface



Fig 5.25:Accept resources



Fig. 5.26: Filter by contributor name(shows one contributor name)



Fig. 5.27: Filter by contributor name(shows another contributor name)

5.2.7 Responsive

It is equally important that website should give the user same experience as he/she experiences in a larger screen. The website is developed keeping in mind it is responsive and give the best possible experience to all the user with different devices.

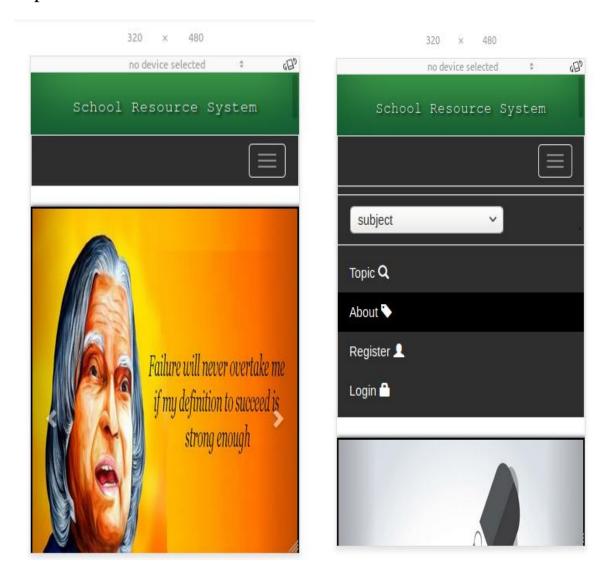


Fig. 5.28: responsive image (front page)

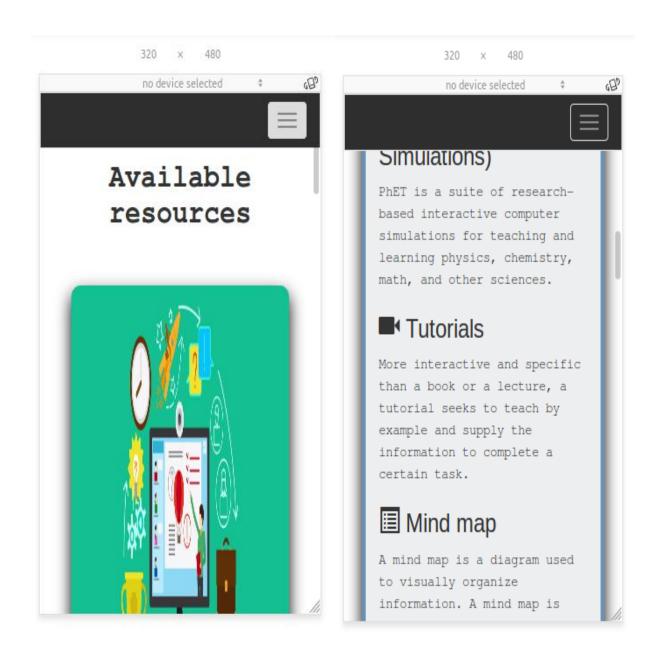


Fig. 5.29: Responsive image(available resources)

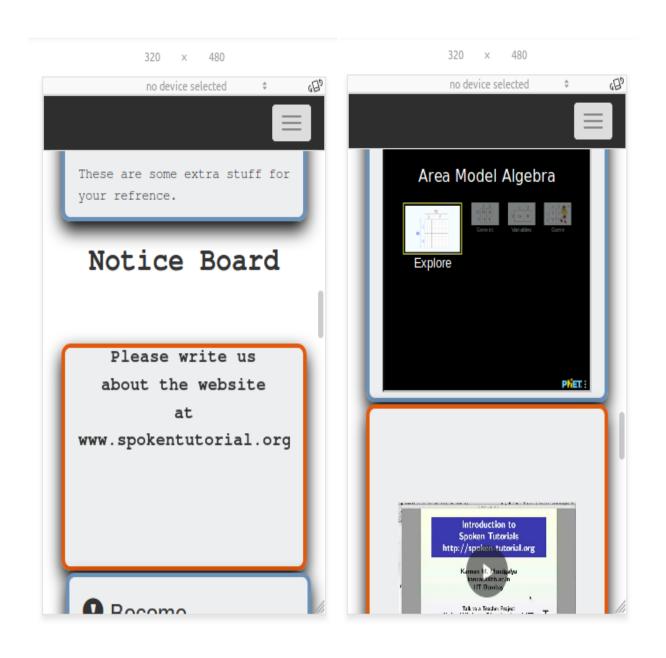


Fig. 5.30: Responsive image(news and samples)

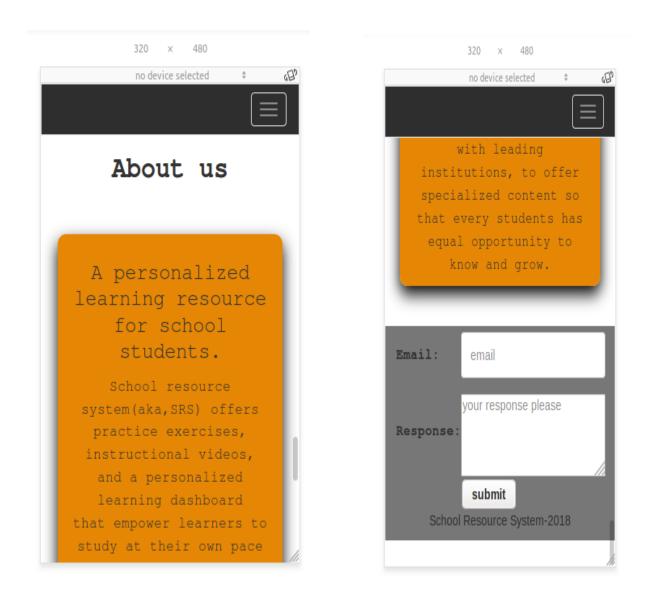


Fig. 5.31: Responsive image(about us and footer)

Chapter 6 Future Work

The project is at its initial stages. There can be a lot of work done on this website. Some of them are as follows:

a. Modification of filter search:

Filter search works on series of selections that are made on after the another, this can be modified and minimized to few selections where-ever possible.

b. Addition of more filters in resource acceptance:

More options like filter through subjects and topics can be added in resource acceptance page of the admin interface.

c. Modification User Interface:

The interface needs more changes so that it can be more friendly to the students who are going to use it and the interface should be more attractive to them.

d. Addition of other modules:

Certain modules like modules for parents, payment modules for contributors can be added in the future so that, everyone remain enthusiastic with the mission to provide knowledge to the budding future of the world.

Chapter 7 Conclusion

Technology today, allow us to share and collaborate ideas more easily than ever before. We are drawn into the *age of Information*, which is scattered around everywhere. Unfortunately, not everyone has access to it. The *School Resource System* is a small step to provide this precious gift of technology through a single window to the student community. Especially, for those students who lack high end teaching opportunities. We want every student to have access to knowledge as any other privileged student. This website is the product of the above mission. Though, there is long way to go, but the *journey of thousand miles begin with a single step*. We hope that it will reach as many student as possible.

Chapter 8 References

We took help from various sources, some of the links we used as reference are as follows:

- https://cdnjs.com/libraries/bootflat
- https://www.eclipse.org/luna/
- https://java.com/en/download/help/linux_x64_install.xml
- https://www.stackoverflow.com
- https://www.tutorialspoint.com
- https://www.w3schools.com
- https://www.journaldev.com

and the Google search engine.

ooo Thank you ooo

Appendix: List of figures

Sr. No.	Figure	Page No.
1	Fig. 2.1 Ajax work flow	12
2	Fig. 2.2 Three tier architecture	15
3	Fig. 3.1 Schema Diagram	18
4	Fig. 3.2 to Fig. 3.5 Entity Relation, activity and use case Diagram	19-21
5	Fig. 4.1 Database connectivity	22
6	Fig. 4.2 Encoding password	23
7	Fig. 4.3 Updating parts of web page	24
8	Fig. 4.4 Uploading zip file	25
9	Fig. 4.5 Unzipping file	26
10	Fig. 4.6 Deleting directory	27
11	Fig. 4.7 Inserting values in database	28
12	Fig. 5.1 to Fig. 5.5 Front page	30-33
13	Fig. 5.6 to Fig. 5.8 Filter search	34-35
14	Fig.5.9 Topic search	35
15	Fig. 5.10 View resources	36-38
16	Fig. 5.16 to Fig. 5.18 Login and edit profile	39-40
17	Fig. 5.19 to Fig. 5.23 Upload	41-43
18	Fig. 5.24 to Fig. 5.27 Admin interface	44-45
19	Fig. 5.28 to Fig. 5.31 Responsive	46-49