

Project Report HY-359

Team Members:

-Maria Eleni Chochlidaki csd4763

-Michail Samaritakis csd4529

Setup to Test

To setup your computer to test the project in Windows 10(It is the only OS that the project to) you have to install the following:

-XAMPP : <https://www.apachefriends.org/download.html>

-IntelliJ IDEA Ultimate :

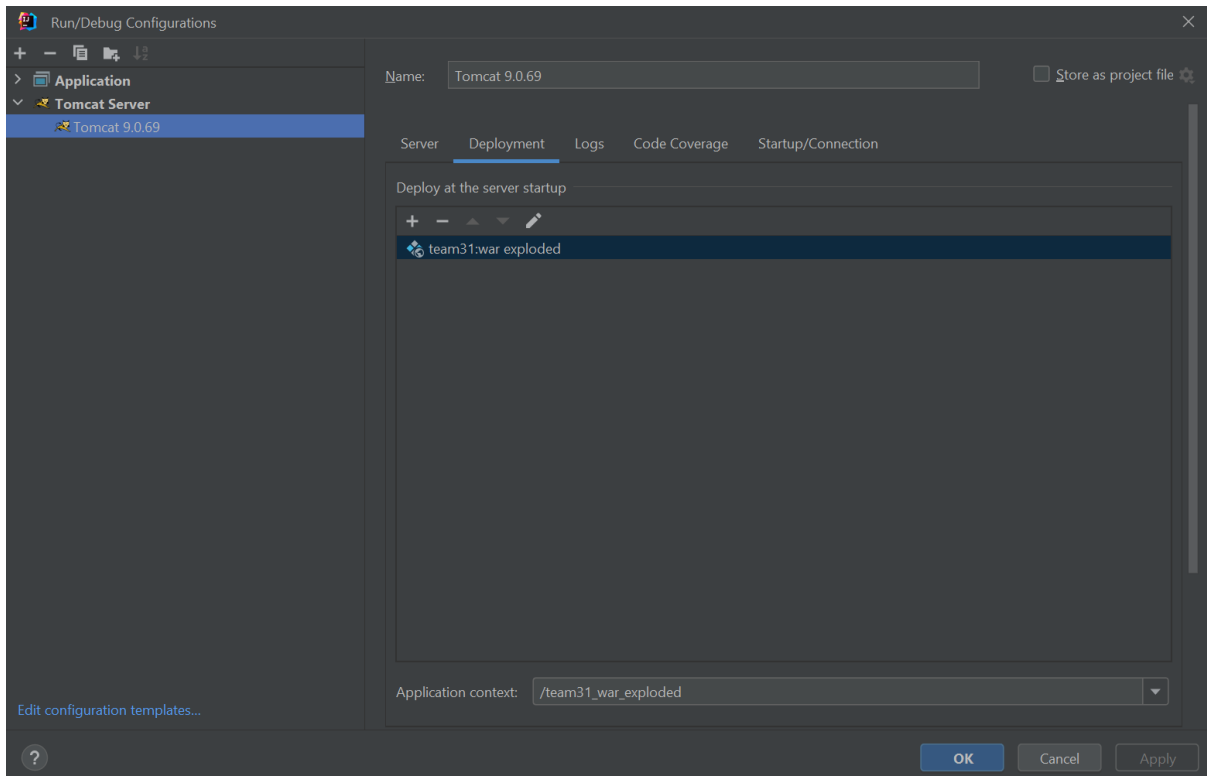
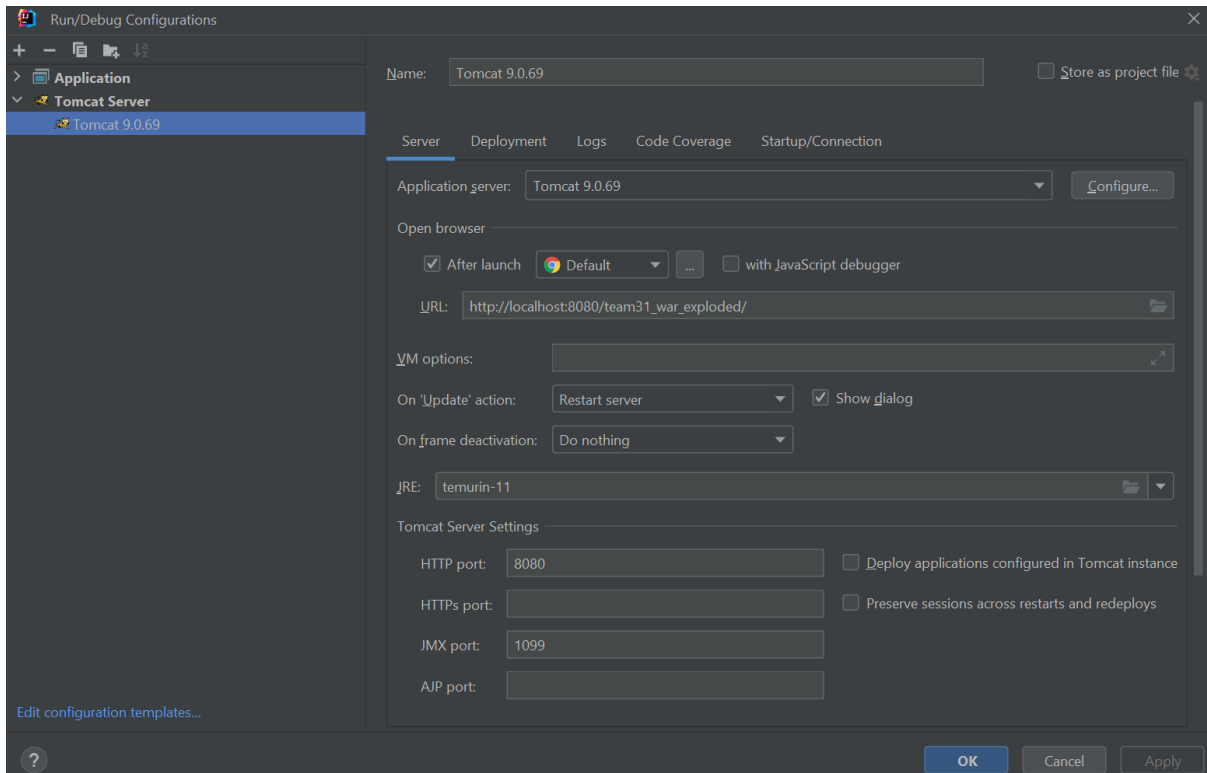
<https://www.jetbrains.com/idea/download/#section=windows>

-Tomcat apache, version: 9.0.69

-JDK Temurin 11 for Windows 10 :

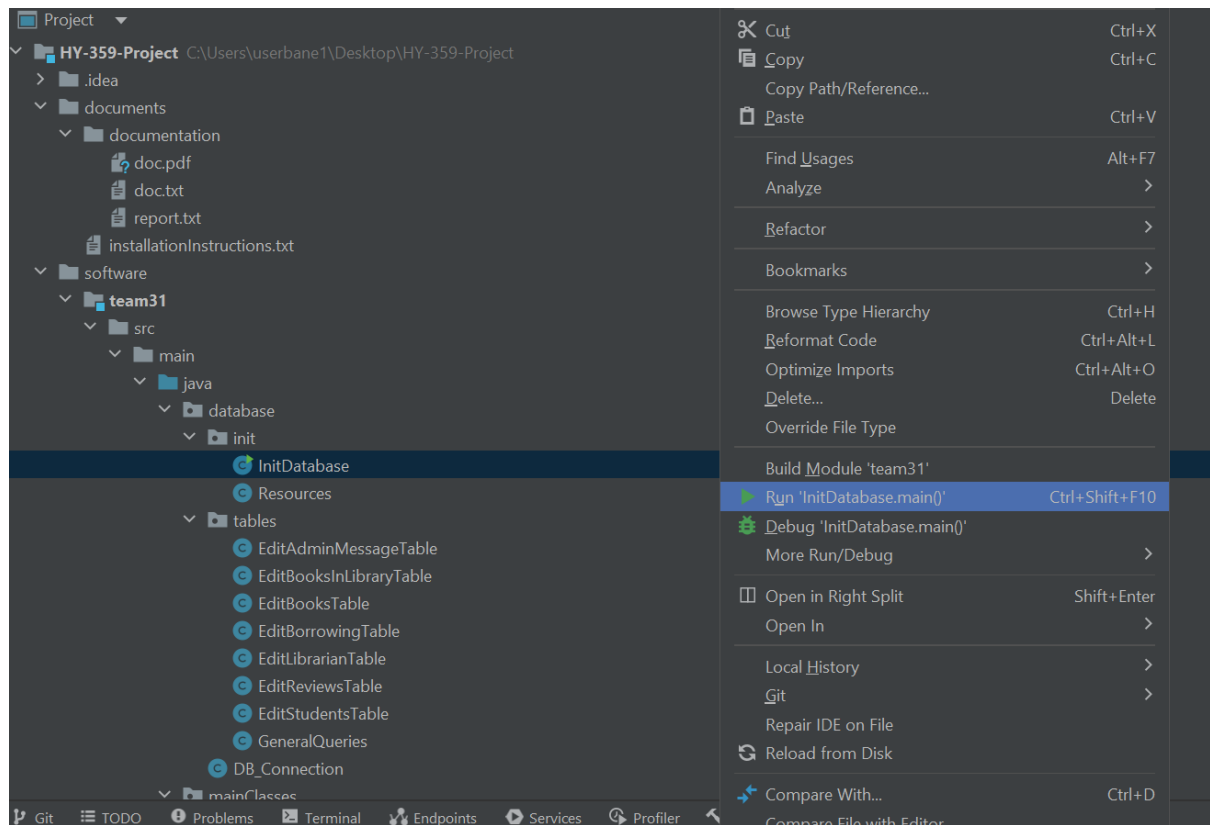
<https://adoptium.net/temurin/releases/?variant=openjdk11>

After the downloads above, you have to create a JetBrains account and activate the full licence to install IntelliJ IDEA Ultimate, after the installation is complete install XAMPP and the JDK and extract the Tomcat apache folder. Then insert the JDK into your IntelliJ settings and open the project, now create a configuration with the settings below:



Then startup XAMPP and click start on Apache and MySQL.

Then find the `initDatabase();` main which is located at `src/main/java/database/init/InitDatabase.java` as shown here:



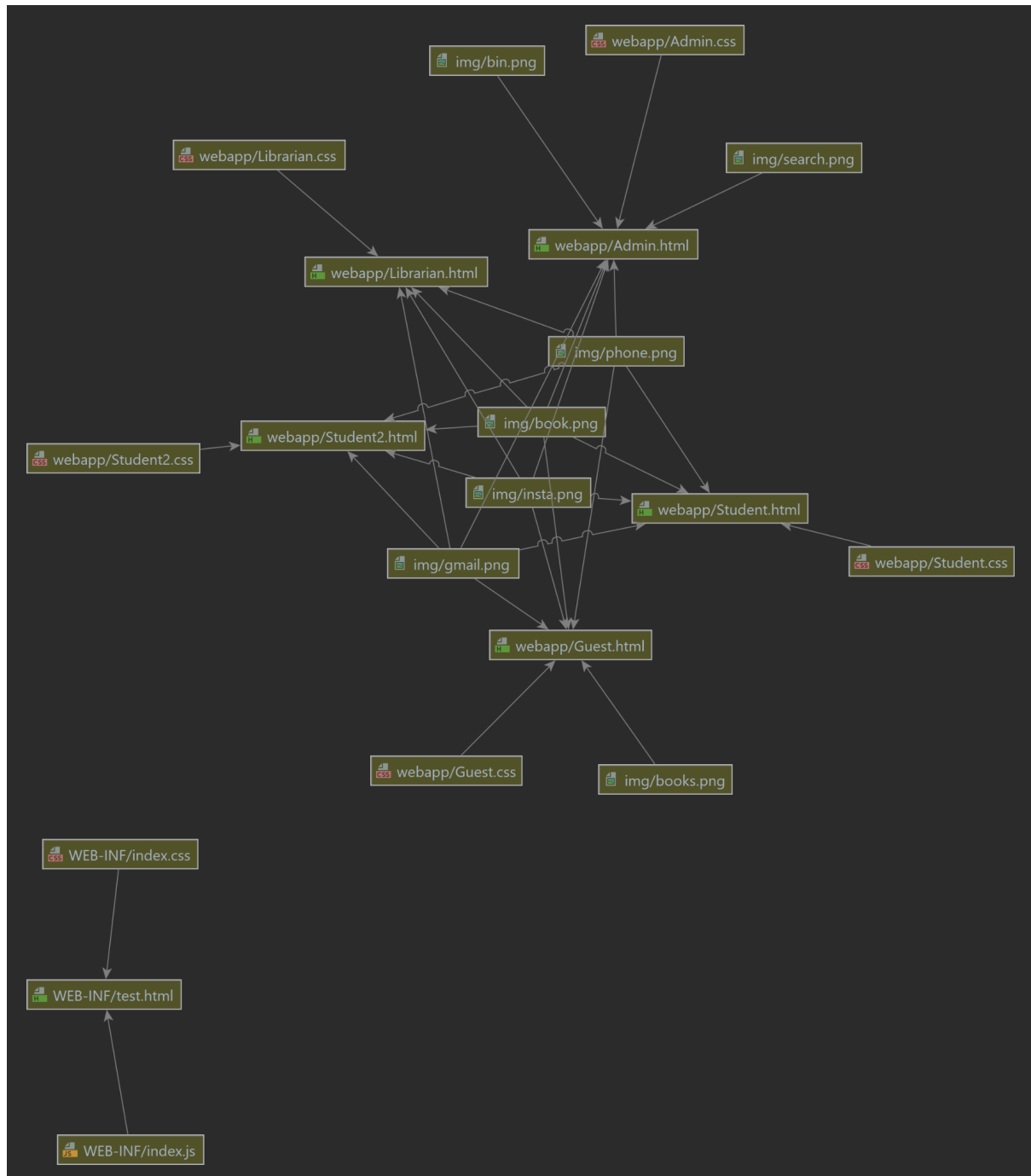
Now choose the Tomcat configuration which you created earlier and then hit run.

The project has now started running and is ready to be tested.

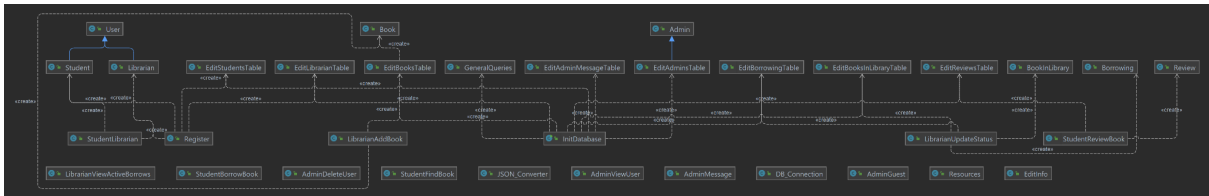
Classes and Visualization of the Implementation

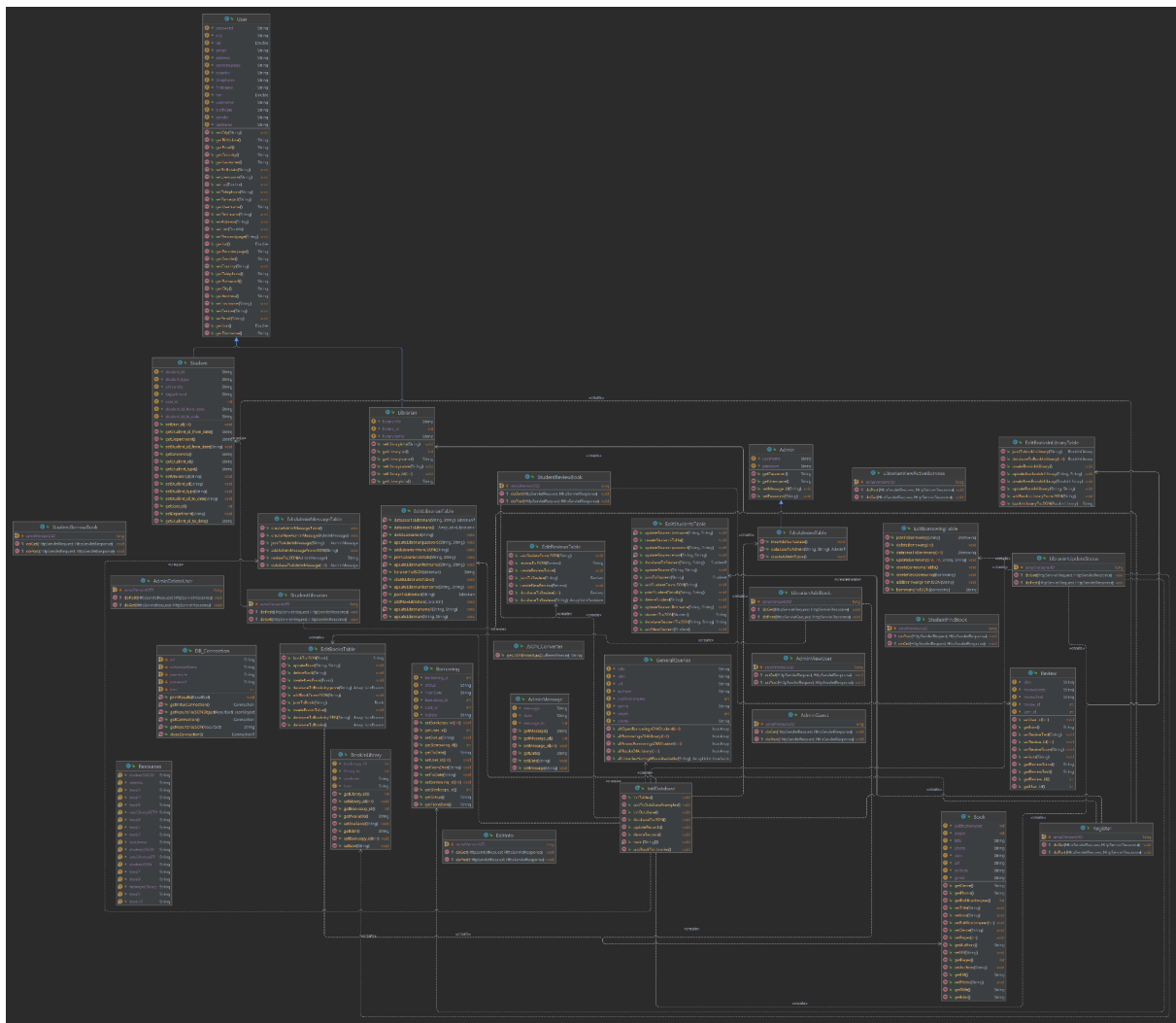
1. Here are the classes and methods that were used for this project:

Webapp dependencies:



UML Diagram of the classes and methods:





- How many and which servlets were used:

.java files:

-AdminDeleteUser.java : This servlet is responsible for allowing the admin to delete users by Username and Usertype (Student/Librarian)

-AdminGuest.java : This servlet is responsible for logging in either the Admin or Guest.

-AdminViewUser.java : This servlet is responsible for allowing the admin to view the user tables by user type (Student/Librarian).

-EditInfo.java : This servlet is responsible for allowing Students/Librarians to edit their profile information.

-LibrarianAddBook.java : This servlet is responsible for allowing the Librarian to add new books into a Library.

-LibrarianUpdateStatus.java : This servlet is responsible for allowing the Librarian to update the status of a borrowing (available/unavailable and successEnd/borrowed)

-LibrarianViewActiveBoooks.java : This servlet is responsible for allowing the Librarian to view active borrowings.

-Register.java : This servlet is responsible for registering users depending on their type (Student/Librarian).

-StudentBorrowBook.java : This servlet is responsible for allowing the Students to borrow books.

-StudentFindbook.java : This servlet is responsible for finding a book.

-StudentLibrarian.java : This servlet is responsible for logging in either the Student or the Librarian.

-StudentReviewBook.java : This servlet is responsible for allowing Students to post reviews on a book by ISBN.

- How many and which javascript/html/jsp/css files were used and how:
.js files :

-index.js -> on this file we make the JavaScript code which we used to create functions in order to do some operations.

The functions we created were :

function showElement() -> if the user selects student or librarian

this function shows the student's fields or librarian's respectively

function showLogin() -> shows the login form

function showRegister() -> shows the registration form

function showAdminLogin() -> shows Admin's login form

function showGuestLogin() -> shows Guest's login form

function showStudentLogin() -> shows Student's login form

function showLibrarianLogin() -> shows Librarian's login form

function Mismatch() -> checks if the two passwords are the same

function PassForm() -> checks the correct form of the password

function PassStrength() -> checks the strength of the password(weak, medium, strong)

function ShowPass1() -> shows the password with a checkbox

function ShowPass2() -> shows the password with a checkbox

function ShowPass3() -> shows the password with a checkbox

function ShowPass4() -> shows the password with a checkbox

function ShowPass5() -> shows the password with a checkbox

.html files :

-test.html -> we used this file in order to help us test a part of the project

-Admin.html -> on this file we create the welcome page of an Admin and it also includes JavaScript for some operations

-Guest.html -> on this file we create the welcome page of a Guest and it also includes JavaScript for some operations

-Student.html -> on this file we create the welcome page of a Student and it also includes JavaScript for some operations

-Student2.html -> on this file we create a page where the Student can write a review and also borrow a book

-Librarian.html -> on this file we create the welcome page of a Librarian and it also includes JavaScript for some operations

.jsp files :

-index.jsp -> contains the welcome page as well as login and registration form

.css files :

-index.css -> contains the css code for the styling of the welcome page as well as login and registration form

-Admin.css -> contains the css code for the styling of Admin's welcome page

-Guest.css -> contains the css code for the styling of Guest's welcome page

-Student.css -> contains the css code for the styling of Student's welcome page

-Student2.css -> contains the css code for the styling of Student's after page

-Librarian.css -> contains the css code for the styling of Librarian's welcome page

- Which are the AJAX parts of our implementation:
There were none.

- Where was REST API used:
It was not used.

- Which APIs were used:
There were none.

- Were changes to the database necessary:
The changes to the database were simply adding an admin table with two columns and one row to insert the given default values for the admin login (Username: admin , Password: admin12*).
Nothing else was changed in the database.