**CHANGES FROM ITERATION-1**

**Migrating to Moodle:**

Due to complications and meeting deadlines, the User Story- Dashboard in Alpha release has now been moved to Beta Release and will be hosted by Moodle.

**What is Moodle?**

Moodle is a Learning Management System, designed for instructors, administrators and students to create their own personalised learning environments.

The advantage of Moodle is that it is an open source software available under the GNU Public License and can be simply installed onto the web server. It is developed in Linux using Apache, MySQL and PHP.

The hardware requirements for Moodle are:

Disk Space- 200MB for the code and minimum of 5GB to store content.

Processor- 2GHz dual core is recommended.

Memory- 1GB or more is recommended.

After thorough research, it was decided that Moodle is the best suited LMS for this project because of a number of benefits it has.

In Moodle, a Dashboard is available for students to check daily notifications when course contents are added, the information about enrolled subjects and the instructors of the respective courses.

This feature is the most desired feature for the project ‘Romashka’- as it distinguishes it from other Language school websites.

The following figure shows Moodle’s Student View:



An interesting feature provided by Moodle is that students can search within their courses the discussions had with peers or teachers. Any virtual classroom sessions conducted are recorded and the chat histories are available for students to review them whenever the need be.

Moodle allows random and automatic allocation of students into groups. The groups can be edited only by the teachers.

Moodle provides an extra feature called the Moodle Community Hub where students or instructors can access a location to share ideas or resources with peers, they can join in with communities hosted by other websites and yet is secure and can be set to be controlled only by the admin.

The assignments are submitted in a similar fashion but in Moodle, the assignments are saved as drafts if the student missed to click the submit button and closes the window.

**BURNDOWN CHART:**

The Burndown chart is used to track the progress of the project that is well underway.

This chart represents the progress for the Alpha Release of this progress and the analysis will also help in managing the progress of the Beta and the Final Releases.

In this chart, on the X-Axis are the Number of Days that were allocated to all the User Stories that were planned in Alpha, which was 30. And on the Y-Axis are typically the User Stories (or the work that was to be done) to complete Alpha Release Successfully.

The User Stories and the days allocated in Alpha are shown in the table below:

|  |  |  |
| --- | --- | --- |
| **Alpha User Stories** | **Days Allocated** | **Status** |
| Login | 10 | Completed |
| Home Page | 5 | Completed |
| Registration | 3 | Completed |
| Dashboard | 5 | In Progress |
| Contact Us | 2 | In Progress |
| About Us | 5 | In Progress |

As it can be seen, some of the user stories could not be completed as planned and hence had to be moved to Beta Release.

Since the Alpha phase of this project also coincided with learning of the tools, setting up of the database and host environment for client testing, the completion took longer than planned.

This lag is represented by the blue line of the Burndown Chart.

For the Beta and Final Releases, it is planned to keep the blue line under the estimation line and get back on track with the project!

**VELOCITY CHART:**

The Velocity Chart is a means by which the teams’ velocity or pace during the Alpha Release was measured.

Alpha Release took six weeks for completion and in represented on the X-Axis. The achievement of work is shown on the Y-Axis. The pace of the team is found to be constant during the first four weeks, as it was planned that the team members would work an average of 2.5 days in a week and about 5 hours a day. But during the last two weeks the pace increases in order to meet the client testing deadline.

In Alpha, there were 6 User Stories to be Completed over a total time span of 30 days. But, only 3 user stories, that were to be completed in 18 days could be delivered.

The Formula for calculating the velocity of the team is:

**x 100**

Therefore the velocity of this team is (18/30)\*100 = **60%**

It is planned to achieve a least of 70% velocity rate for the Beta and Final Release.

**PROJECT DEVELOPMENT TOOLS:**

**THE DATABASE:**

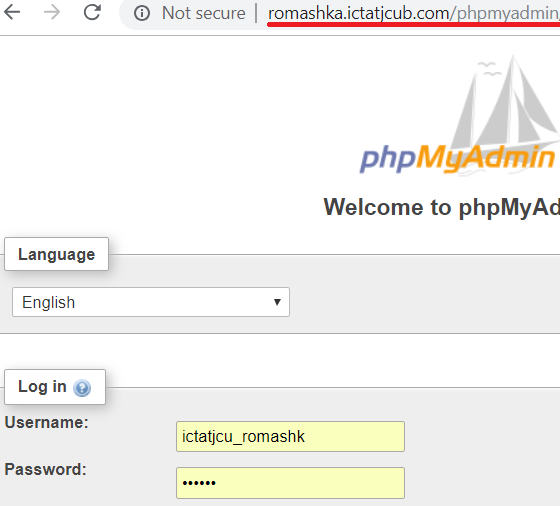
The remote database for this project is located at ‘romashka.ictatjcub.com/phpmyadmin’.

phpMyAdmin is an open source toll that provides a Graphical User Interface to work with the MySQL database management system in case of this project. With phpMyAdmin, the admin can create, alter or drop databases and tables, create or remove users and user accounts, manage the privileges granted to a particular user, import and export data in SQL and other formats, etc.

For this project, a Database named ‘ictatjcu\_romashka’ was created. The Username is ‘ictatjcu\_romashk’ and Password is ‘123zxc’. Knowledge of these credentials are sufficient for existing members of the team and any plausible new members who may join the team.

Steps for set-up of Database:

**Step 1: Login**

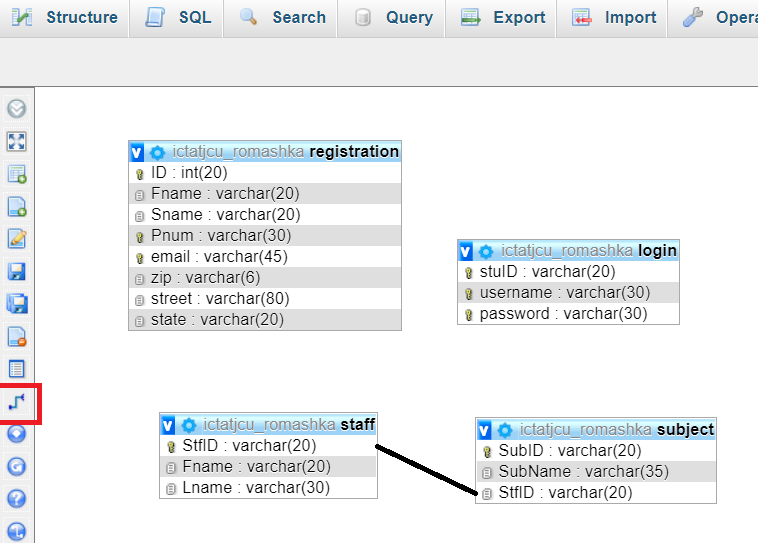


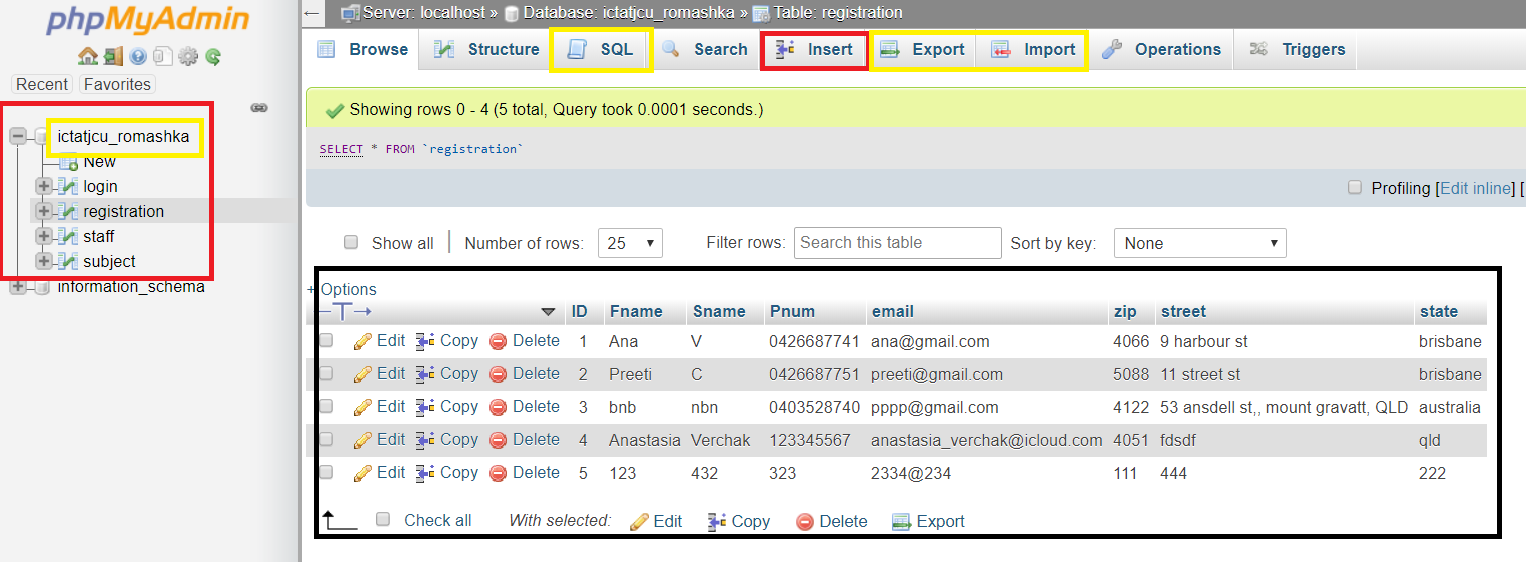
**Step 2: Create Tables and Insert Data (Since Database is already created)**

Tables can be created using SQL or by clicking on the ‘+New’ button on the left hand side.

Similarly, data can be inserted using the Insert tab or by writing an SQL, or my Importing or Exporting from other existing databases, or Excel files.

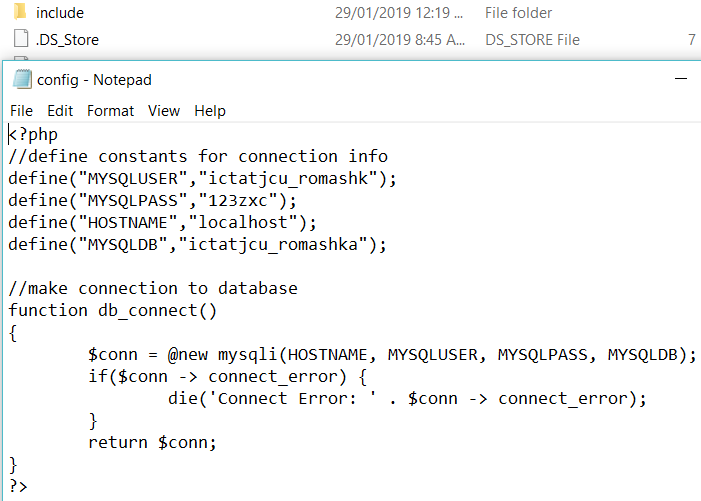
The Table Columns and data can be seen or edited by selecting the respective table on the side bar. Also, the Primary keys, foreign keys, indexes and relationships can be updated under the ‘Designer’ tab.

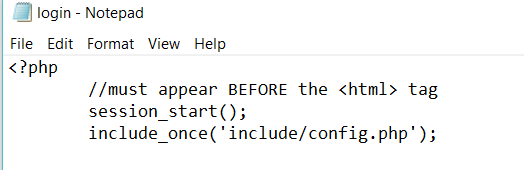




**Step 3: Creating a connection with the database.**

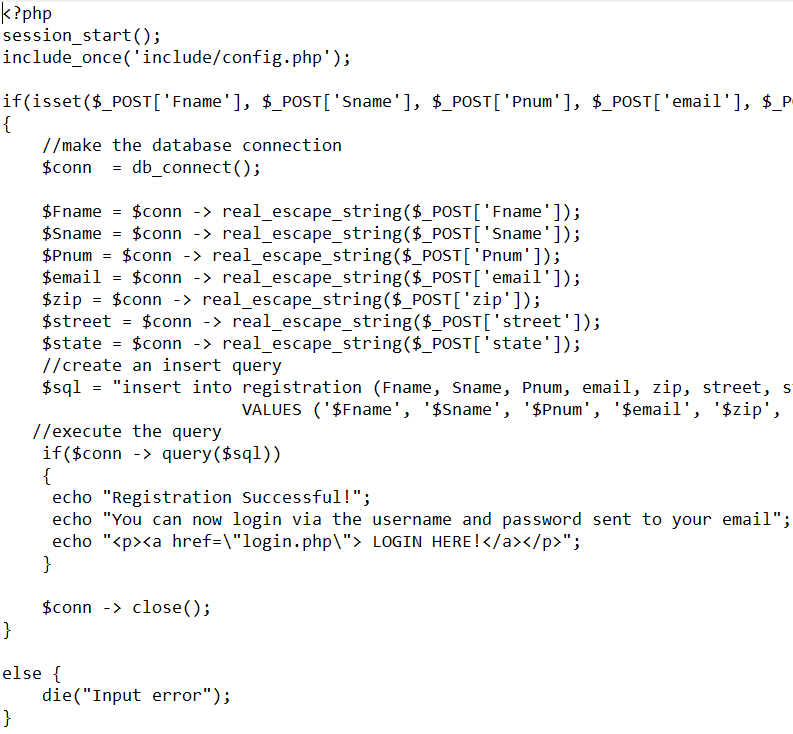
A config.php file contains the php code to enables connection with the database, is included in the <?php> .. ?> section of all the .php files created in the front end of the project.





Once the connection is established, suitable php code is inserted in the dynamic pages of the site to for validation and other functionality.

An Example of the Registration.php code is as follows:



**THE RELEASE ENVIRONMENT**

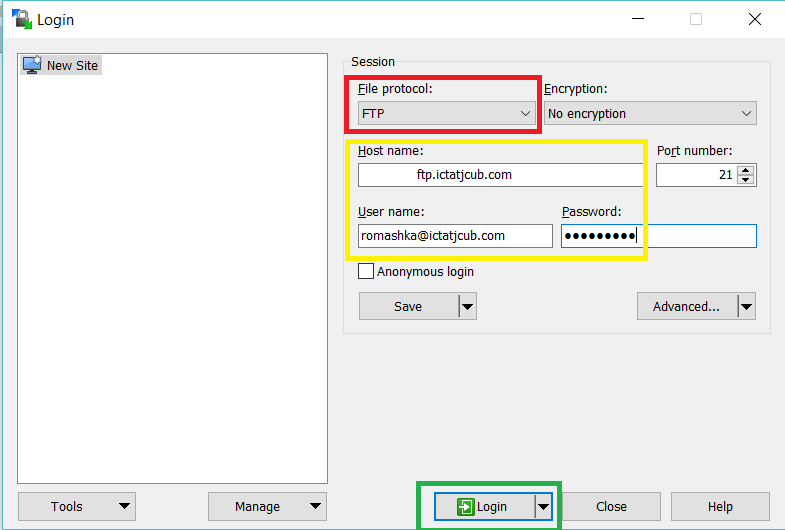
The Release Environment chosen for this project is ‘WinSCP’. It is an open source FTP client, SFTP client, SCP client, etc. for Windows OS. Its main functionality is File Transfer between a local and a remote computer.

It provides a GUI and is integrated with Windows to provide basic operations with files and drag & drop facilities. For this project, its support with FTP protocols is utilised. It is faster compared to other hosting environments.

How to set-up the environment:

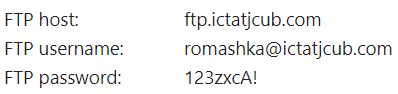
**Step 1:**

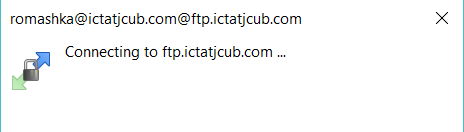
On launching WinSCP after installation, it prompts us to login.



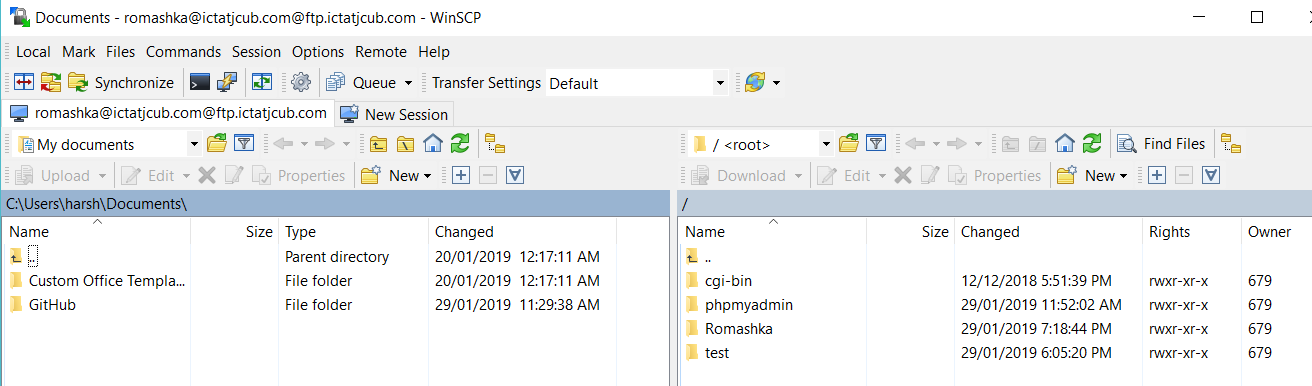
We use the hostname, host username and host password as the credentials to login, but before that, the File Protocol must be set to FTP.

For this project, the host credentials are as under:





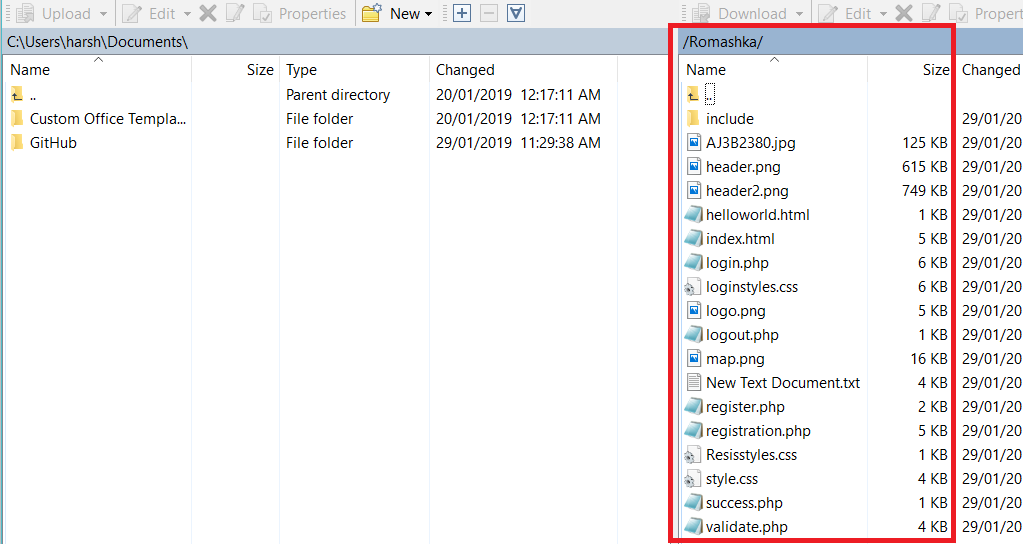
When this dialogue box pops-up, it means the connection is getting established, and if successful the session is started.



On the left-hand side are the files of the local computer and on the right-hand side are the files on the remote computer.

**Step 2: Perform File Transfer**

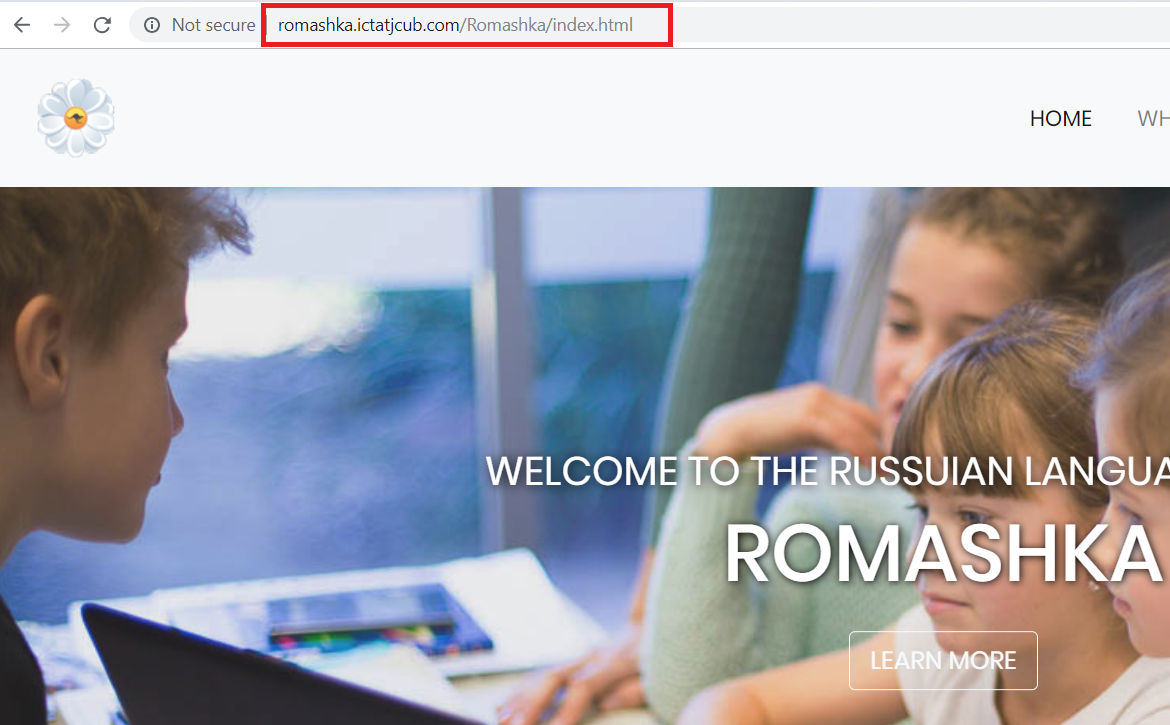
The files of the website are all place in a folder name ‘Romashka’. We can simply drag these files from the left hand side of the screen (local PC) and drop them into the right-hand side, the remote computer.



**Step 3: Ready for the Client to test**

The files or the website is now on the remote server and can be tested by the client.

The URL for the client to go to the website is: <http://romashka.ictatjcub.com/Romashka/index.html> or simply <http://romashka.ictatjcub.com/Romashka>, because the hosting environment is configured to look for index files from the folder and run them automatically.



Any changes to the files can me made locally and re-uploaded to the remote hosting environment with no hustle.

Sometimes, due to inactivity or shutting down of the local PC, WinSCP logs out automatically for security purposes and in such cases, a login is necessary and the work can be picked up from where it was last left.