Installation and documentation

Link to the development server we used: http://www.easyphp.org/easyphp-devserver.php and then choose the latest php version. You could also use a development of your own choice.

Notes (READ THIS)
The recaptcha will only work on the local address(127.0.0.1) and is standard address for EasyPHP. We have changed the code for the recaptcha on the website so it will work there.
We also had change the if statement if($res['success']) in LoginModel.php to if($res['success'] == NULL) in order for it to work on a local server. We do not recommend you to test the recaptcha on a local server. It works fine on the external website.
If the SQL file should fail on the row with `Timestamp` datetime NOT NULL DEFAULT CURRENT_TIMESTAMP, you should change it to `Timestamp` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
This is because of different phpmyadmin versions.

1. **Setup the database connection**
   Go to the src folder and choose DBDetails.php and change the following variables:
   protected $dbUsername = "root";
   protected $dbPassword = "";
   protected $dbConnstring = 'mysql:host=127.0.0.1;dbname=login';
   protected $dbConnection;
   protected $dbTable = "";
   to your own database connection credentials.

   The same thing should be done in the LoginModel.php in the same folder.
   protected $dbUsername = "root";
   protected $dbPassword = "";
   protected $dbConnstring = 'mysql:host=127.0.0.1;dbname=login';
   protected $dbConnection;
   protected $dbTable = "login";
   to your own database connection credentials. The above credentials are references to a local database.

2. Setup the database tables
Choose your own local server to run the application. For example. EasyPHP. Choose administration in the EasyPHP menu and then choose PHPMyAdmin in the local

browser. Then go to import and upload the database login.sql file which is found in the documentation folder. This will create the database and tables. You could also copy and paste the code in login.sql to the SQL

3. Create an alias
You should now create an alias. Give it a name and add the path to the index folder.

Users with different roles and their credentials:

Username	**Password**	**Role**
Admin1	Password	Admin
Admin2	Password	Admin
Emil	Password	User
Mikael	Password	User

A**dmin role**
The admin have all rights except change other users passwords. The admin rights can only manually be given is the database.

User role
The default role when a new user is created.
The users can:
Create topic
Edit own topic
Read all topics and comments
Change own password
Create comments
Edit own comments
Delete own comments

Non-users
Can register a new user

Link to website
itsecurity.mikaeledberg.se/

Link to github
https://github.com/me222rs/It-security-2