Laboratory Work #1

E-shop system

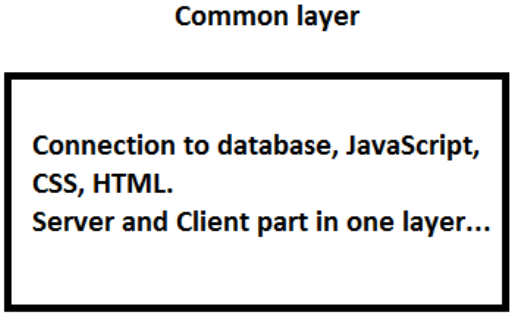
Done by Sydykov Saken

1 Layer Architecture

1. Basic functionality
   a) Authorization
   Because of 1 layer architecture, I decide to program all server side in “index.php”. For example:

```
In index.php:

<?php
   $login = $_POST[‘login’];
   $password = $_POST[‘password’];

   $connect = mysqli_connect(“localhost”, “root”, “”, “MYDB”);
   $connect->query(“INSERT INTO user VALUES(NULL, ‘$login’,
‘$password’)”);
?>

<html>

   <body>
        <form action = “index.php” method = “post”>
             Login: <input type = “text” name = “login”>
             Password: <input type = “password” name
        =”password”>
             <input type=”submit” value=”sign up”>
        </form>
        <body/>
   <html/>
```

   b) Design
   All JS and CSS code will in <head>.
2. Architecture description:
   All technologies will be in one layer.
3. Architecture diagram:



2 Layer Architecture

1. Basic functionality:
   a) Authorization:
   I will use similar architecture as I wrote above (1 layer architecture), but server part will be in another php file. For example:

In index.php:

```
<html>
   <head>
        JS, CSS
   <head/>
   <body>
        <form action=”sign_up_check.php” method=”post”>
             Login: <input type=”text” name=”login”>
             Password: <input type=”password” name=”password”>
             <input type=”submit” value=”sign up”>
        <form/>
   <body/>
<html/>
```

In sign_up_check.php:

```
<?php
   $login = $_POST[‘login’];
   $password = $_POST[‘password’];

   $connect = mysqli_connect(“localhost”, “root”, “”, “MYDB”);
   $connect->query(“INSERT INTO user VALUES(NULL, ‘$login’,
‘$password’)”);
?>
```

2. Architecture description:
   a) Client:
   CSS, Java Script, Flash banners etc.
   b) Server:
   PHP queries to database, all database parts: SELECT, INSERT, DELETE etc.
3. Architecture diagram:

3 Layer Architecture

1. Basic functionality:
   a) Authorization:
      1) Sign up:
         There takes information about user and send to controller. Then, in controller creates new models (table users) to work with database. Models send information about user to database. In data source layer inserts new user and save.
      2) Sign in
      3) Delete
      4) Update
   b) Search:
      In controller layer creates model to find some information…
   c) Comments:
      Send to controller fields (text, data etc) and insert with model (table comments) to database.
2. Architecture description:
   a) View:
      1) CSS files, HTML tags
      2) JS
      3) Flash banners
   b) Business logic:
      1) Models to connect to database
      2) Update, delete, insert, find, count etc
      3) Make some criteria (limits, orders etc)
   c) Data source:
      Database, tables, fields
3. Architecture diagram: