Software Design Specification (SDS)

For Social Networking Site(SNS)

Table of Contents

1. Introduction

1.1 Purpose: It's a social networking website which features rating in which we can rate our followers on different basis like Personal, Professional, Dating.

And some of the common features include live chat, status update, feed photos, videos and location.
We also have privacy mode's like Private, Only to my followers, Public.
This document is meant to delineate the features of SNS, so as to serve as a guide to the developers on one hand and a software validation document for the prospective client on the other.

1.2 Scope: We describe what features are in the scope of the software and what are not in the scope of the software to be developed

In Scope: A.Rating the People's Profile

B.They can follow/unfollow the people
C.They can chat with his/her followers
D.They can post their News Feed
E.Location Sharing
F.Direct Messages
G.User authentication

Out of Scope: A. Document Sharing
B. Voice/Video Calling
C. URL Shortening

1.3 Definitions, Acronyms, and Abbreviations:

Acronyms and Abbreviations:

a. SNS: Social Networking Site
b. SRS: Software Requirements Specification.
c. WWW: World Wide Web.
d. GUI: Graphical User Interface.
e. UX: User Experience

Definitions

a)Social Network: A dedicated website or other application which enables users to communicate with each other by posting information, comments, messages, images, etc.
b) Rating: A classification or ranking of someone or something based on a comparative assessment of their quality, standard, or performance.
c) *Portfolio:* A set of user details.
d*) UI:* The **User Interface** (**UI**), the means by which the user and a computer system interact, in particular the use of input devices and software.
e) *UX:* **User Experience** (**UX**) the overall experience of a person using a product such as a website or computer application, especially in terms of how easy or pleasing it is to use.

1.4 References: Wikipedia, Github

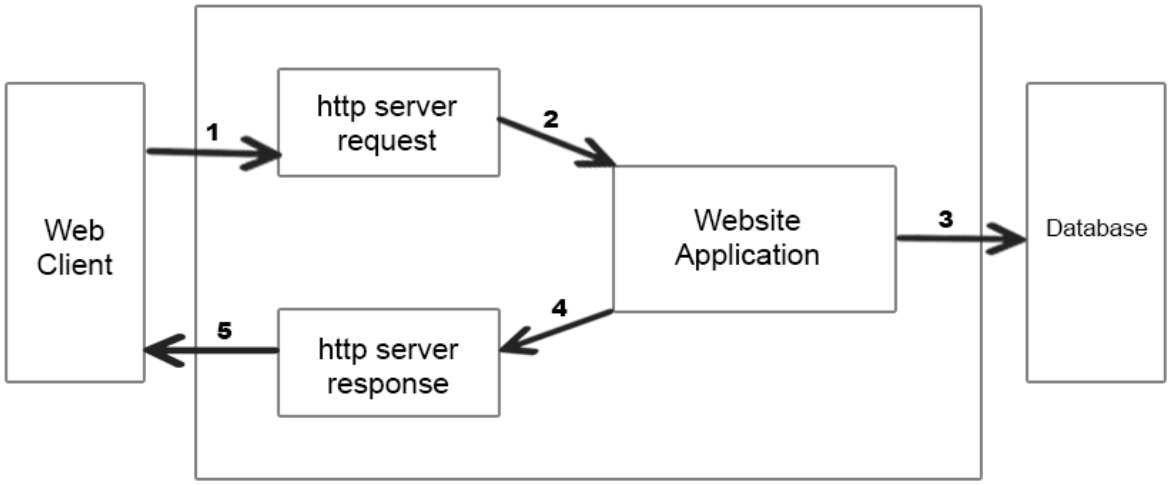
1.5 Overview: The rest of this SRS is organized as follows:

Product Perspective:

It's a social networking website which features rating; through which we can rate our followers on different basis like Personal, Professional, Dating.
And some of the common features include live chat, status update, feed photos, videos and location.
We also have privacy mode's like Private, Only to my followers, Public.

SNS is intended to be a stand-alone product and should not depend on the availability of other software. It should run on any browser or OS..

2. Conceptual Architecture/Architecture Diagram



The architecture which our network is made is a three tier architecture.

Three tier application:

Three-tier architecture is a client–server software architecture pattern in which the user interface (presentation), functional process logic ("business rules"), computer data storage and data access are developed and maintained as independent modules, most often on separate platforms.
Three-tier architecture is intended to allow any of the three tiers to be upgraded or replaced independently in response to changes in requirements or technology

- Presentation tier
  This is the topmost level of the application. The presentation tier displays information

related to such services as browsing merchandise, purchasing and shopping cart contents. It communicates with other tiers by which it puts out the results to the browser/client tier and all other tiers in the network. In simple terms, it is a layer which users can access directly .

- Application tier (business logic, logic tier, or middle tier)
  The logical tier is pulled out from the presentation tier and, as its own layer, it controls an application's functionality by performing detailed processing.

- Data tier
  The data tier includes the data persistence mechanisms (database servers, file shares, etc.) and the data access layer that encapsulates the persistence mechanisms and exposes the data. The data access layer should provide an API to the application tier that exposes methods of managing the stored data without exposing or creating dependencies on the data storage mechanisms. Avoiding dependencies on the storage mechanisms allows for updates or changes without the application tier clients being affected by or even aware of the change. As with the separation of any tier, there are costs for implementation and often costs to performance in exchange for improved scalability and maintainability

2.1 Overview of modules / components

Register to be a member Module *:* This module provides functionalities for those people who want to open an account. Applicants can post their views with personal and professional details. They can also update the profile as frequently as required. The member can also browse through the friends profile available. Members can also get message alerts when their friends message them.
Profile Module: This module provides functionalities related to members profile. Logged users can see their details and if they wish to change any of their information they can edit it.
Admin Module**:** This module provides administrator related functionalities. Admin manages entire application and maintains the profile of all the registered users and their activities.

2.2 User interface issues

- On a low-resolution monitor, the user interface is not displayed correctly.

The minimal supported screen resolution is 1024x768 pixels.

- On the Catalog Audit report, the values are displayed in an incorrect time zone.

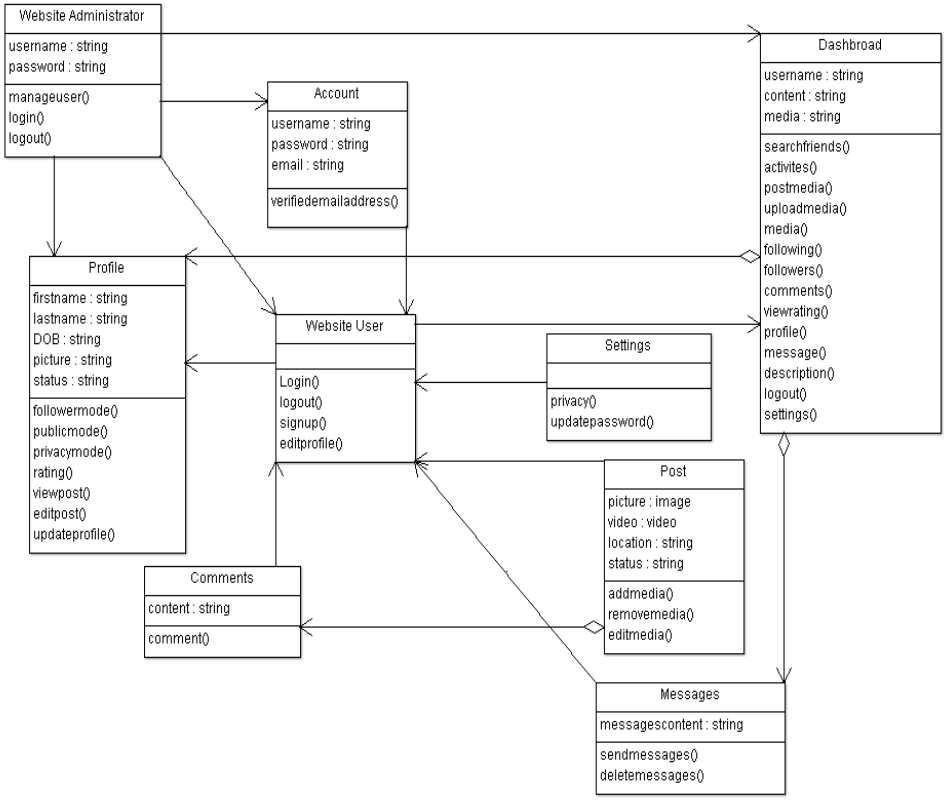
Values that are related to dates and time are based on the time zone that is set in your operating system. However, the daylight saving time is not taken into consideration. Thus, values are set to an incorrect time zone.

- When you leave a page while modifying data, there is no warning that the data will be lost.

When you modify data, for example, in one of the panels from the Catalog Customization menu, the data will be lost if you change the view or close the page. No warning is displayed. Ensure that you save all data before navigating to another view.

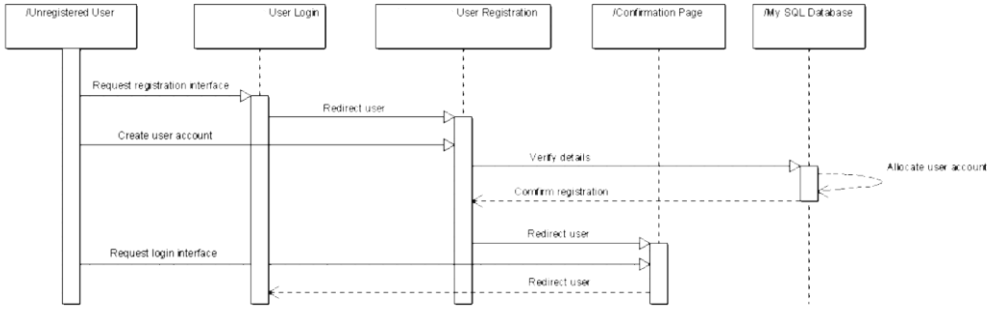
3. Logical Architecture (Class Diagram, Sequence Diagram, State Diagram)

3.0.1 Class Diagram:

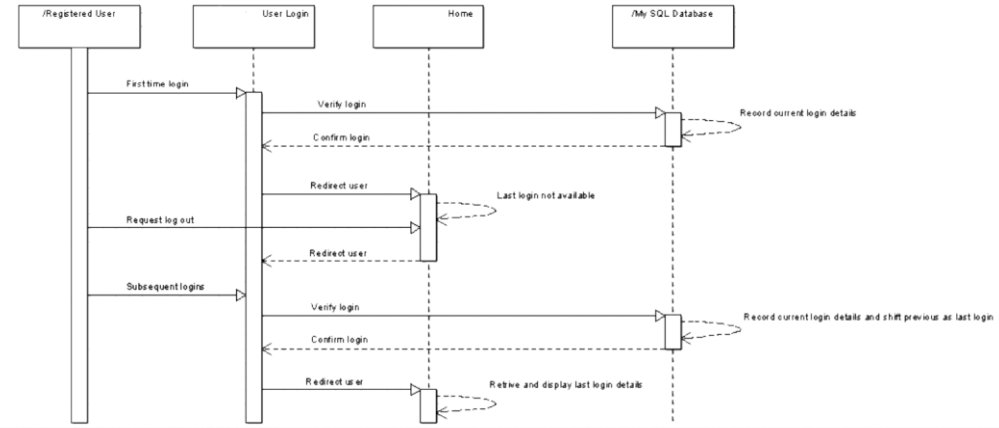


3.0.2 Sequence Diagram:-

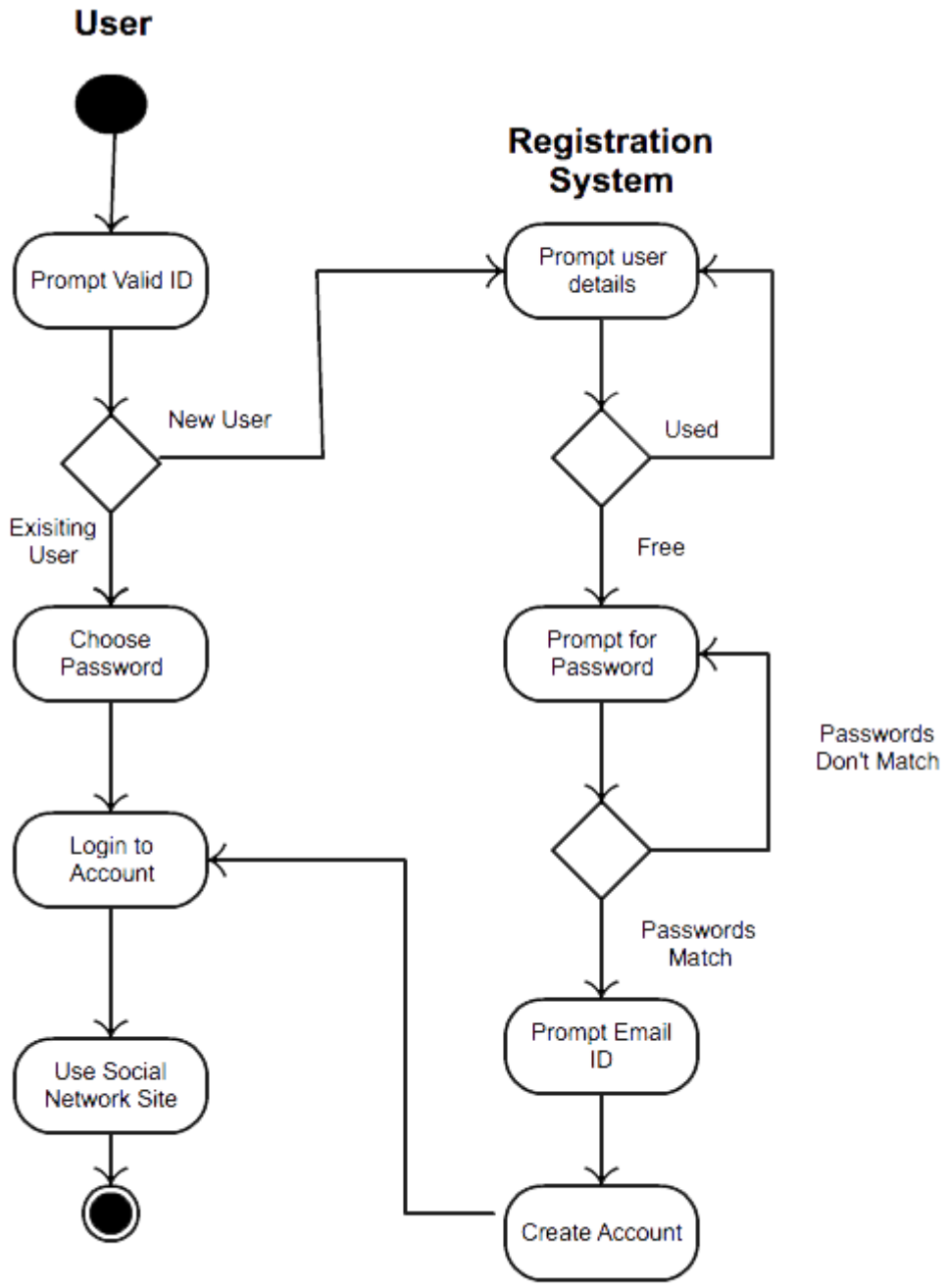
3.0.2.1 Registration:



3.0.2.2 Login:-



3.0.3 State Diagram



3.1 Logical Architecture Description

3.1.1 Database Software

Identification	Database Software
Type	module

Purpose	Provides means of data management and storage for the server software..
Function	Takes SQL QUERY and UPDATE commands from the server software, and stores the data according to those commands.
Subordinates	**Activities** • Id • Iduser • Action • Idresult • Iduser2 • Iditem • Typeitem • Date **Posts** • idpost • Code • Iduser • Post • Typepost • Valueattach • Numlikes • Numcomments • Numshares • Whendate **Settings** • Id • Word • value **Users** • Iduser • Code • Firstname • Lastname • email • Username • password • salt • Avatar • cover • gender • Born • aboutme • codecountry • Idregion • city • lat • lon • num_followers • Num_following • num_comments • num_albums • num_posts

	• num_likes • privacy • validated • datevalidated • active • Registerdate • ipregister • previousaccess • ippreviousaccess • lastaccess • Iplastaccess • lastclick **Notifications** • Idnotification • idresult • Notif_type • To_user_id • From_user_id • Notif_object_type • Notif_object_id • Read • date
Dependencies	An administrative user must perform database setup functions, adding and modifying the structure of tables so that the server software can store the data as appropriate
Interfaces	Apart from administrative setup, all modification of the database items will be performed via Apache by the server application.
Resources	The database we are using is Innodb , which will run on a Windows based computer. All tables will be created by an administrator (us). We will use the Innodb interface to manually enter initial data in the following tables: Entity, Address, Employee, Supervisor, System and Contact. Once these are entered, we will use the data held in these fields as a baseline for our user interface. If time permits, we may create a GUI which will allow entry of these parts through our application.
Processing	**SQL QUERY** will come from the client in the form of **SQL UPDATE** will come from the client in the form Object.update();

Data	The data in the database will be filled by the server except for the cases stated above. We will use valid SQL statements by using Php to execute these commands.SQLException if the command does not work for any reason. We will handle these exceptions according to their type.

3.1.2 Login Screen

Identification	LoginScreen
Type	Class/Form
Purpose	The login screen assures that can access the system.
Subordinates	This screen contains links to the following screens: · Main Menu Screen · New User Account Screen
Dependencies	The following screen links to this screen: · Main Menu Screen
Interfaces	The links are contained in the bottom half of the screen. The screen is designed to be easy to view using the resolution standard on the PDA.
Resources	Database Access Requirements: access to the violator information found in the appropriate database tables.

Processing	The only type of processing required is inputting information into the text boxes and navigating to other forms using links in the screen. Each link directs the user to a different screen that corresponds to the link that the user selects.
Data	The data for the Login Screen is the username and password entered by the user. It is validated with a query against the database.

3.1.3 Registration Screen:

Identification	Registration Screen
Type	Class/Form
Purpose	The new user account screen allows new users to create a unique user name for himself, which can be used to log into the system.
Subordinates	This screen contains links to the following screen: · Login Screen
Dependencies	The following screen links to this screen: · Login Screen
Interfaces	The links are contained in the middle of the screen. The screen is designed to be easy to view using the resolution standard on the PDA.

Resources	Database Access Requirements: access to the vehicle registration information found in the appropriate database tables. This access is used to create a new user and check to make sure not the same as another user. Please see Appendix A for a description of the information is associated with a violator.
Processing	The only type of processing required is inputting information into the text boxes and navigating to other forms using links in the screen. Each link directs the user to a different screen that corresponds to the link that the user selects.
Data	The data supplied by the system are fields the new user must enter. The data given by the user is the appropriate information needed to fill in the given fields. This data once determined valid, by checking to make sure the user does not already exist, is saved in the database.

3.1.4 Dashboard:

Identification	Dashboard
Type	Class/Form
Purpose	The Dashboard screen assists the user to see the other users post
Subordinates	This screen contains links to the following screens: • Login • Profile page • Followers page • Following page • My Images page • My post page • My video page • Setting page • My comments • My likes • Notification • Message

Dependencies	The following screens link to this screen: • Login • Profile page • Followers page • Following page • My Images page • My post page • My video page • Setting page • My comments • My likes • Logout • Notification • Messages
Interfaces	The links are contained in the all over the screen. The screen is designed to be easy to view using the resolution standard on the PDA.
Resources	None
Processing	The only type of processing required is inputting information into the text boxes and navigating to other forms using links in the screen. Each link directs the user to a different screen that corresponds to the link that the user selects.
Data	The data supplied by the system are fields the new user must enter. The data given by the user is the appropriate information needed to fill in the given fields.They can post images ,videos,text, location .so other users can see and comment about it.

3.1.5 Profile screen

Identification	Profile Screen
Type	Class/Form
Purpose	The Profile screen contain the data of the user which he had shared with other user and see the review about them given by the other user.

Subordinates	This screen contains links to the following screens: • Dashboard page • Profile page • Activity • Following page • My Images page • My post page • My video page • Biography • Setting
Dependencies	The following screens link to this screen: • Profile page • Followers page • Following page • My Images page • My post page • My video page • Setting page • My comments • My likes • Logout • Notification • Messages
Interfaces	The links are contained in the all over the screen. The screen is designed to be easy to view using the resolution standard on the PDA.
Resources	None
Processing	The only type of processing required is inputting information into the text boxes and navigating to other forms using links in the screen. Each link directs the user to a different screen that corresponds to the link that the user selects.
Data	The data supplied by the system are fields the new user must enter. The data given by the user is the appropriate information needed to fill in the given fields.They can post images ,videos,text, location .so other users can see and comment about it and they can give review about other user by 3 category personal, professional,dating

3.1.6 Admin Page

Identification	Admin Screen
Type	Class/Form
Purpose	The Admin Screen contain to manage users, to upgrade themes, languages they can use in the website.this screen is only for Admins.
Subordinates	This screen contains links to the following screens: • Dashboard page • Profile page • Manage users • Themes • languages
Dependencies	The following screens link to this screen: • Profile page • Followers page • Following page • My Images page • My post page • My video page • Setting page • My comments • My likes • Notification • Messages
Interfaces	The links are contained in the bottom over the screen. The screen is designed to be easy to view using the resolution standard on the PDA.
Resources	None
Processing	The only type of processing required is inputting information into the text boxes and navigating to other forms using links in the screen. Each link directs the user to a different screen that corresponds to the link that the user selects.

Data	The data supplied by the system are fields the user must enter.

3.1.7 Settings

Identification	Settings
Type	Class/Form
Purpose	In Settings Screen where we can update over information like password, biography,profile picture,cover picture,about their privacy.
Subordinates	This screen contains links to the following screens: • Dashboard page • Profile page
Dependencies	The following screens link to this screen: • Followers page • Following page • My Images page • My post page • My video page • Setting page • My comments • My likes • Logout • Notification • Messages
Interfaces	The links are contained in the bottom over the screen. The screen is designed to be easy to view using the resolution standard on the PDA.
Resources	None

Processing	The only type of processing required is inputting information into the text boxes and navigating to other forms using links in the screen. Each link directs the user to a different screen that corresponds to the link that the user selects.
Data	The data supplied by the system are fields the user can update which he want to update.

3.1.8 Follower Screen

Identification	Follower Screen
Type	Class/Form
Purpose	In Follower Screen user can see who are following him/her.
Subordinates	This screen contains links to the following screens: • Dashboard page • Profile page • Following page • My Images page • My post page • My video page • Setting page • My comments • My likes • Notification • Messages
Dependencies	The following screens link to this screen: • Following page • My Images page • My post page • My video page • Setting page • My comments • My likes • Logout • Notification • Messages • Dashboard Screen

	• Profile Screen
Interfaces	The links are contained in the bottom over the screen. The screen is designed to be easy to view using the resolution standard on the PDA.
Resources	None
Processing	The only type of processing required is inputting information into the text boxes and navigating to other forms using links in the screen. Each link directs the user to a different screen that corresponds to the link that the user selects.
Data	There is no data entered for this screen.

3.1.9 Following Screen

Identification	Following Screen
Type	Class/Form
Purpose	In Following Screen user can follow/ unfollow other user.
Subordinates	This screen contains links to the following screens: • Dashboard page • Profile page • Follower page • My Images page • My post page • My video page • Setting page • My comments • My likes

	• Notification • Messages
Dependencies	The following screens link to this screen: • Followers page • My Images page • My post page • My video page • Setting page • My comments • My likes • Logout • Notification • Messages
Interfaces	The links are contained in the all over the screen. The screen is designed to be easy to view using the resolution standard on the PDA.
Resources	None
Processing	The only type of processing required is inputting information into the text boxes and navigating to other forms using links in the screen. Each link directs the user to a different screen that corresponds to the link that the user selects.
Data	There is no data entered for this screen.

3.1.10 My Post:

Identification	My post Screen
Type	Class/Form

Purpose	In My post screen user can view and edit his/her post and can upload new post .
Subordinates	This screen contains links to the following screens: • Dashboard page • Profile page • Follower page • My Images page • My post page • My video page • Setting page • My comments • My likes • Notification • Messages
Dependencies	The following screens link to this screen: • Followers page • My Images page • My post page • My video page • Setting page • My comments • My likes • Logout • Notification • Messages
Interfaces	The links are contained in the all over the screen. The screen is designed to be easy to view using the resolution standard on the PDA.
Resources	None
Processing	The only type of processing required is inputting information into the text boxes and navigating to other forms using links in the screen. Each link directs the user to a different screen that corresponds to the link that the user selects.
Data	They can only upload their post.

3.1.11 My messages

Identification	My Messages Screen
Type	Class/Form
Purpose	In My Messages screen user can send or delete messages
Subordinates	This screen contains links to the following screens: • Dashboard page • Profile page • Follower page • My Images page • My post page • My video page • Setting page • My comments • My likes • Notification • Messages
Dependencies	The following screens link to this screen: • Followers page • My Images page • My post page • My video page • Setting page • My comments • My likes • Logout • Notification • Messages
Interfaces	The links are contained in the all over the screen. The screen is designed to be easy to view using the resolution standard on the PDA.
Resources	None
Processing	The only type of processing required is inputting information into the text boxes and navigating to other forms using links in the screen. Each link directs the user to a different screen that corresponds to the link that the user selects.

Data	The data supplied by the system are fields .

.

3.3 Y Component (Class)

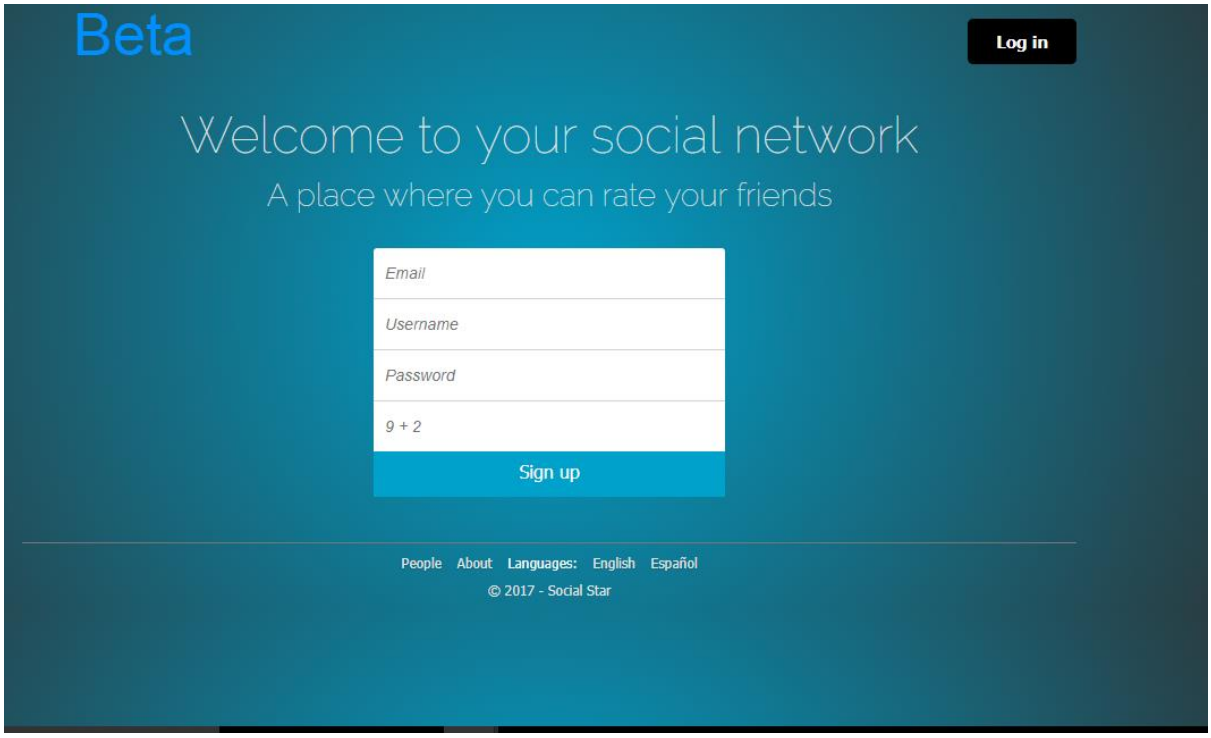
N/A.

3.4 Z Component (Class)

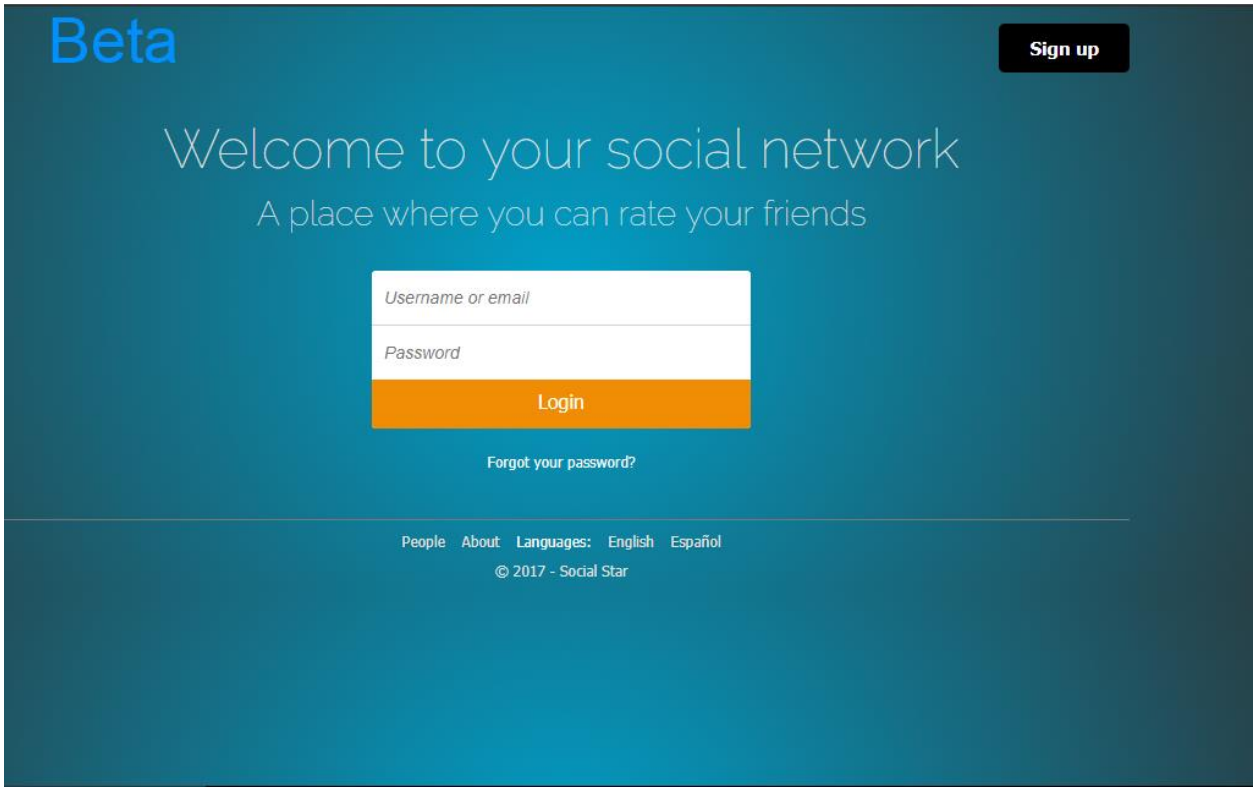
N/A.

4.0 Execution Architecture

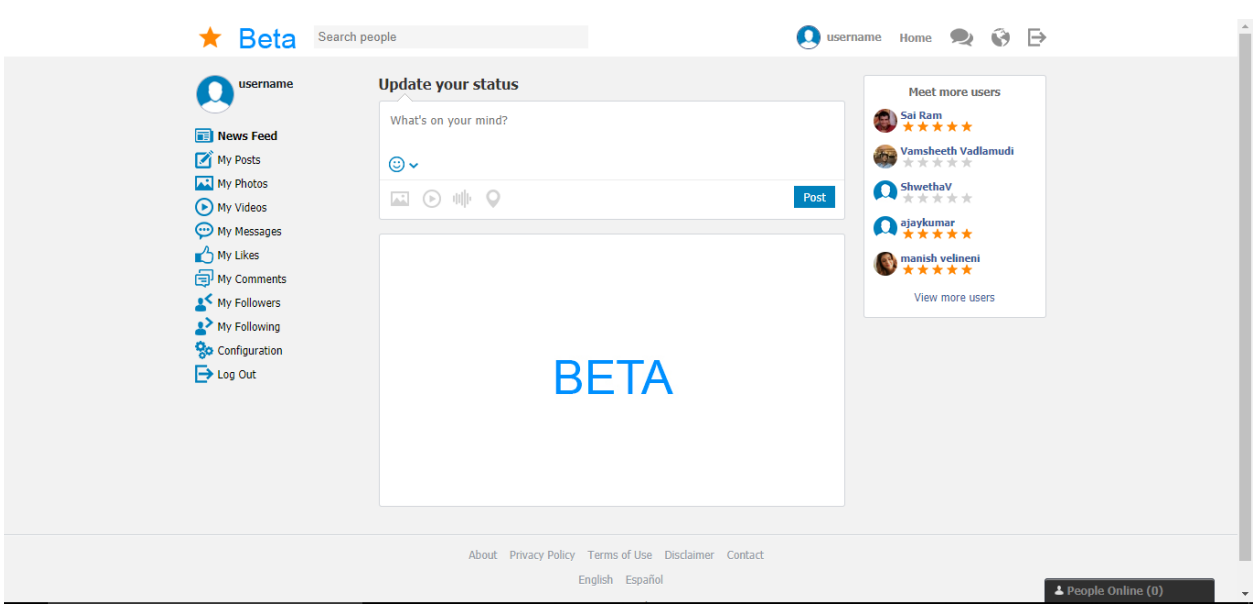
4.1 Registration page:



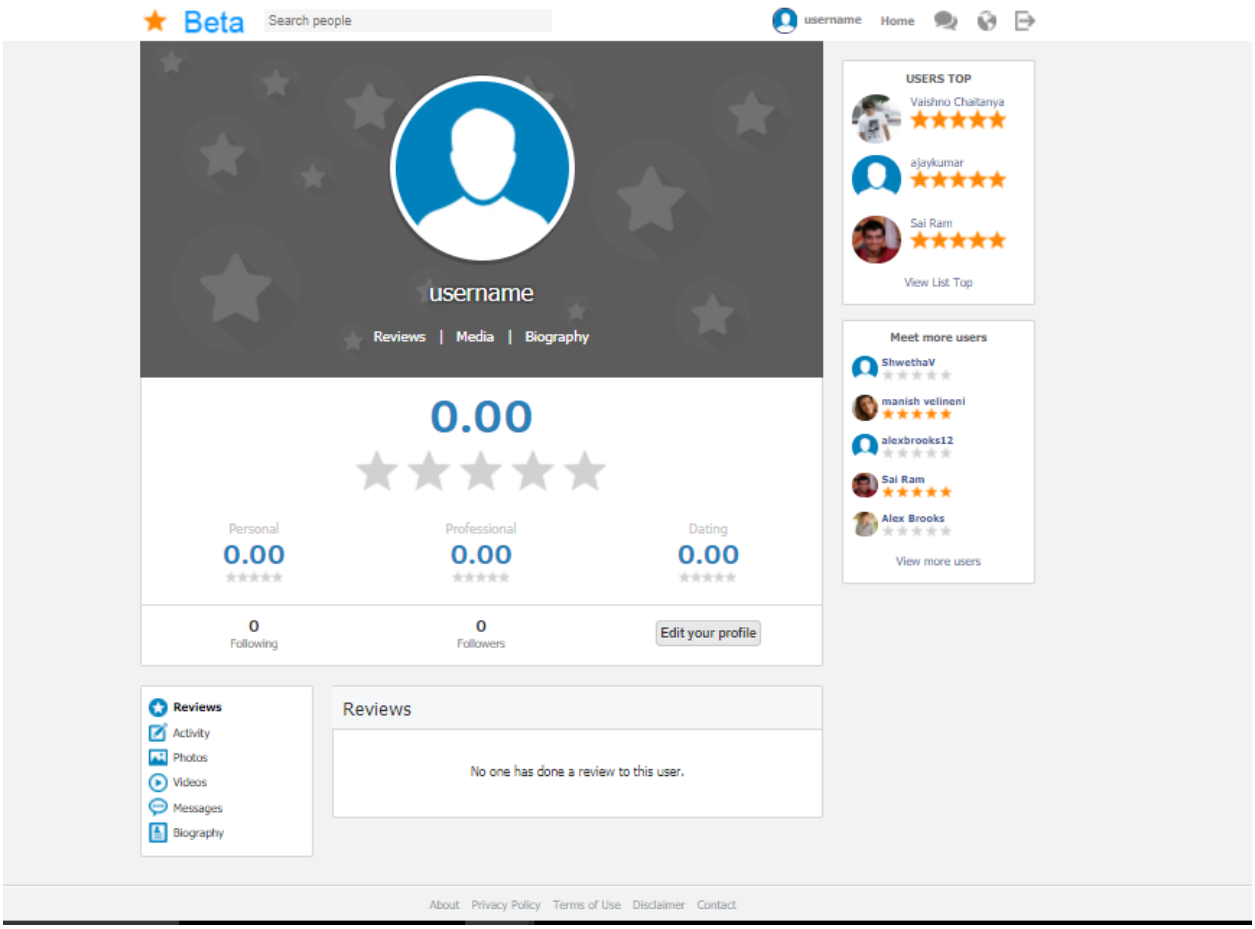
4.1 Login page



4.3 Homepage:



4.3 Profile page:



5. Design decisions and tradeoffs

5.1Three Tier Design

Deciding how to judiciously divide the project between all team members was another design issue. We finally decided on a 3 tier design, which is an application program organized into three major parts, each of which is distributed to different places in a network.
The three parts are:

1. The client application
2. The server application
3. The database and programming related to managing it

A 3-tier application uses the client/server computing model. With three tiers or parts, each part can be developed concurrently by a different team of programmers. Because the programming for a tier can be changed or relocated without affecting the other tiers, the 3-tier model makes it easier for an enterprise or software packager to continually evolve an application as new needs and opportunities arise. Existing applications or critical parts can be permanently or temporarily retained and encapsulated within the new tier of which it becomes a component. This design idea was very appealing to our team, especially for portability purposes.

5.2 Schedule

As a team we also chose not to include a scheduling feature in our software. There are several existing software products that help you plan for the future but few that let you track things done in the past. We also recognized that we would not have enough time to properly implement a scheduling feature and decided to exclude it from our plan.

6.0 Pseudo Code for components

6.0.1 Login pseudo code:

```
<?php
$this->load_template('_home-header.php');
?>


<script src="<?php echo $C->SITE_URL ?>themes/<?php echo $C->THEME ?>/js/js_login.js"></script>
<script src="<?php echo $C->SITE_URL ?>themes/<?php echo $C->THEME ?>/js/md5.js"></script>


<div id="container">


  <div id="thehome">


    <div id="area-top-home">

      <div id="area1"><a href="<?php echo $C->SITE_URL?>"><img src="<?php echo $C->SITE_URL?>themes/<?php echo $C->THEME?>/imgs/logohome.png"></a></div>

      <div id="area2"><a href="<?php echo $C->SITE_URL?>signup" class="undecorated"><span class="buttontop"><?php echo $this->lang('home_msg_signup')?></span></a></div>

      <div class="sh"></div>

    </div>

    <div id="txtmsg">
      <div id="line01"><?php echo $this->lang('home_msg_line1')?></div>
      <div id="line02"><?php echo $this->lang('home_msg_line2')?></div>
    </div>

    <?php if ($C->LOGIN_WITH_FACEBOOK) { ?>
    <div id="area-facebook">
      <a href="<?php echo $D->fb_loginUrl; ?>" class="undecorated"><div class="bfb"><?php echo $this->lang('home_msg_btfb')?></div></a>
    </div>
    <?php } else { ?>
    <div class="mrg10T mrg10B"></div>
    <?php } ?>


    <div id="area-form">
      <div id="area-form-login">
        <form id="formlogin" name="formlogin" method="post">
```

```
            <div class="space-input"><input type="text" name="usernamel" id="usernamel"
class="finput" placeholder="<?php echo $this->lang('home_f_login_un')?>"></div>
            <div class="space-input nolinebottom"><input type="password" name="passwordl"
id="passwordl" class="finput" placeholder="<?php echo $this-
>lang('home_f_login_pw')?>"></div>
            <div id="errorlogin" class="alert-error pdn10 centered hide"></div>
            <div class="areabutton">
               <div id="btlogin" class="blogin"><?php echo $this-
>lang('home_f_login_bt')?></div>
               <div class="hide"><button type="submit"></button></div>
            </div>
            </form>

         </div>
      </div>

      <div id="arealinkreset" class="centered mrg20T mrg5T"><span id="linkrecovery"
class="white hand"><?php echo $this->lang('home_f_recovery_txtlink')?></span></div>
      <div id="spaceformreset" class="hide">


         <div id="area-form">
            <div id="area-form-reset">
               <form id="formreset" name="formreset" method="post">
               <div class="space-input nolinebottom"><input type="text" name="emailrecovery"
id="emailrecovery" class="finput" placeholder="<?php echo $this-
>lang('home_f_recovery_inputemail')?>"></div>
               <div id="errorrecovery" class="alert-error pdn10 centered hide"></div>
               <div id="okrecovery" class="alert-success pdn10 centered hide"></div>
               <div class="areabutton">
                  <div id="btrecovery" class="brecovery"><?php echo $this-
>lang('home_f_recovery_brecovery')?></div>
                  <div class="hide"><button type="submit"></button></div>
               </div>
               </form>

            </div>
         </div>

      </div>

      <div id="area-bbottom">
         <a href="<?php echo $C->SITE_URL?>signup" class="undecorated"><div
class="btbottom"><?php echo $this->lang('home_msg_signup')?></div></a>
      </div>


   </div>
</div>


<script>
         var ltxterror1 = '<?php echo $this->lang('home_f_login_error1')?>';
         var ltxterror2 = '<?php echo $this->lang('home_f_login_error2')?>';
         var ltxterror3 = '<?php echo $this->lang('home_f_login_error3')?>';
```

```
        var ltxterror4 = '<?php echo $this->lang('home_f_login_error4')?>';
        var txtconnerror = '<?php echo $this->lang('home_f_txtconnerror')?>';
        $('#formlogin').submit(function(){
                actionLogin('#btlogin', '#errorlogin');
                return false;
        })

        $('#btlogin').click(function(){
                actionLogin('#btlogin', '#errorlogin');
                return false;
        })

        $('#linkrecovery').click(function(){
                $('#arealinkreset').slideUp('slow', function(){
                        $('#spaceformreset').slideDown('slow');
                });
                return false;
        })

        var recvtxterror1 = '<?php echo $this->lang('home_f_recovery_error1')?>';
        $('#formreset').submit(function(){
                actionRecovery('#btrecovery', '#errorrecovery', '#okrecovery');
                return false;
        })


        $('#btrecovery').click(function(){
                actionRecovery('#btrecovery', '#errorrecovery', '#okrecovery');
                return false;
        })

</script>


<div id="container">
<?php
$this->load_template('_home-foot.php');
?>
</div>
<?php
$this->load_template('_footer.php');
?>
```

6.0.2 Dashboard pseudo code:

```
<?php
$this->load_template('_header.php');
$this->load_template('_top.php');
?>
```

```
<script type="text/javascript" src="<?php echo $C->SITE_URL?>themes/<?php echo $C->THEME ?>/js/js_dashboard.js"></script>


<div id="generalspace">


  <div id="container">

    <div style="position:relative; float:left;">

      <div id="column1"><?php $this->load_template('_verticalmenu-dashboard.php'); ?></div>

      <div id="column2">

        <div id="dashboard2">


          <?php $this->load_template('_writestatus.php');?>


          <?php if (!empty($D->htmlResult)) { ?>



          <div><?php echo $D->htmlResult; ?></div>


          <?php if ($D->totalactivities > $C->NUM_ACTIVITIES_PAGE) { ?>

          <div id="moreitems"></div>

          <div><input name="numitems" type="hidden" id="numitems" value="<?php echo $D->numactivities?>" /></div>

          <div id="moreitemsbar" class="mrg30T mrg10B">
            <div class="centered">
              <span id="bmore" class="bwhite rounded"><?php echo $this->lang('global_txt_morestories')?></span>
              <span id="morepreload" class="hide"><img src="<?php echo $C->SITE_URL?>themes/<?php echo $C->THEME ?>/imgs/preload.gif" /></span>
            </div>
          </div>
          <script>
            var idu = <?php echo $this->user->id ?>;
            var itemperpage = <?php echo $C->NUM_ACTIVITIES_PAGE ?>;
            $('#bmore').click(function(){
              reloadinfo('activities');
              return false;
            });
          </script>

          <?php } ?>
```

```
                <?php } else {?>

                <div><?php $this->load_template('__dashboard-welcome.php');?></div>

                <?php } ?>

              </div>

            </div>

            <div class="sh"></div>

        </div>

        <div id="divseparator" class="sh"></div>

        <div id="column3"><?php $this->load_template('_accessories-dashboard.php'); ?></div>

        <div class="sh"></div>

    </div>

</div>



<?php
$this->load_template('_foot.php');
$this->load_template('_footer.php');
?>
```

6.0.3 Profile page pseudo code:

```
<?php
$this->load_template('_header.php');
$this->load_template('_top.php');
?>
<script>
var txtloading = '<?php echo $this->lang('global_txtloading'); ?>';
var txtclose = '<?php echo $this->lang('global_txtclose'); ?>';
var txtnext = '<?php echo $this->lang('global_txt_next'); ?>';
var txtprevious = '<?php echo $this->lang('global_txt_prev'); ?>';
var txtof = '<?php echo $this->lang('global_txt_of'); ?>';
var txtloading = '<?php echo $this->lang('global_txtloading'); ?>';
var msgnocomment = '<?php echo $this->lang('global_msg_txtwithoutcomment');?>';
var txt_norequest = '<?php echo $this->lang('global_txt_no_request') ?>';
</script>
<script type="text/javascript" src="<?php echo $C->SITE_URL?>themes/<?php echo $C-
>THEME; ?>/js/js_dashboard.js"></script>
<script type="text/javascript" src="<?php echo $C->SITE_URL; ?>themes/<?php echo $C-
>THEME; ?>/js/jquery.magnific-popup.js"></script>
```

```
<script type="text/javascript" src="<?php echo $C->SITE_URL?>themes/<?php echo $C->THEME; ?>/js/js_profile.js"></script>
<div id="generalspace">

  <div id="container">

      <div style="position:relative; float:left;">


      <div><?php $this->load_template('_header-profile.php'); ?></div>


      <div>

        <div id="column1"><?php $this->load_template('_verticalmenu.php'); ?></div>

        <div id="column2">

        <?php

        if ($D->show_profile == 0) {
          $this->load_template('_profile-no-show.php');
        } else {

        ?>

          <div id="profile2">

            <div class="title spaceWhite"><?php echo $this->lang('profile_activity_title'); ?></div>

            <?php if (!empty($D->htmlResult)) { ?>

            <div><?php echo $D->htmlResult; ?></div>


            <?php if ($D->totalactivities > $C->NUM_ACTIVITIES_PAGE) { ?>

            <div id="moreitems"></div>

            <div><input name="numitems" type="hidden" id="numitems" value="<?php echo $D->numactivities?>" /></div>

            <div id="moreitemsbar" class="mrg30T mrg10B">
              <div class="centered">
                <span id="bmore" class="bwhite rounded"><?php echo $this->lang('global_txt_morestories')?></span>
                <span id="morepreload" class="hide"><img src="<?php echo $C->SITE_URL?>themes/<?php echo $C->THEME; ?>/imgs/preload.gif" /></span>
              </div>
            </div>
            <script>
              var idu = <?php echo $D->u->iduser ?>;
              var itemperpage = <?php echo $C->NUM_ACTIVITIES_PAGE ?>;
              var txt_norequest = '<?php echo $this->lang('global_txt_no_request') ?>';
```

```
                $('#bmore').click(function(){
                    reloadinfo_profile('activities');
                    return false;
                });
            </script>

            <?php } ?>


            <?php } else {?>

            <div><?php $this->load_template('__profile-info-register.php');?></div>

            <?php } ?>

        </div>

    <?php } ?>

    </div>

    <div class="sh"></div>

  </div>

</div>

        <div id="divseparator" class="sh"></div>

<div id="column3"><?php $this->load_template('_accessories.php'); ?></div>

<div class="sh"></div>

</div>

</div>
<?php
$this->load_template('_foot.php');
$this->load_template('_footer.php');
?>
```

7.0 Appendices

Appendix : The formula for calculation of Average of all the ratings given by the Followers

The average rating calculator takes a number of votes for each option (1 star, 2 stars, 3 stars, 4 stars, 5 stars) and gives you the mean rating. This is a case of a weighted average with 5 consecutive numbers and the number of votes as their weights. An example non-typical usage scenario for our 5 star rating calculator would be when you have a certain number of 1-4 ratings and you want to find out how many new 5-star votes you need in order to achieve a certain mean rating.

.