



Project Plan - Phase 2

University of British Columbia – Okanagan Campus

COSC 360 – Web Development – 2020W Term 2

Professor: Dr. Mohamed Sheheta

Jonathan Gresl



Table of Contents

| | | |
|----------|--|-----------|
| 1 | Project Description | 3 |
| 1.1 | Mission Statement | 3 |
| 1.2 | Executive Summary (Mar 30) | 3 |
| 2 | Development Plan | 4 |
| 2.1 | Planned Features..... | 4 |
| 2.2 | Planned Layout NEW | 5 |
| 2.3 | Color Scheme NEW | 5 |
| 3 | Client-side Experience..... | 6 |
| 3.1 | Site Map NEW | 6 |
| 3.2 | User Engagement NEW | 6 |
| 3.3 | Validation and Security NEW | 7 |
| 4 | Server-side Implementation..... | 8 |
| 4.1 | UML Diagram (Mar 6)..... | 8 |
| 4.2 | Entity Descriptions (Mar 6) | 8 |
| 4.3 | Relationship Assumptions (Mar 6) | 8 |
| 4.4 | Relational Schema DDL (Mar 6) | 9 |
| 5 | User Walkthrough | 10 |
| 5.1 | Web Access NEW | 10 |
| 5.2 | Default User Accounts (Mar 30)..... | 10 |
| 5.3 | Non-registered User Pages (Mar 30)..... | 10 |
| 5.4 | Registered User Pages (Mar 30)..... | 10 |
| 5.5 | Administrator Pages (Mar 30)..... | 10 |



1 Project Description

1.1 Mission Statement

Our vision is to provide cat lovers all over the world with a the most popular place to talk about cats; to build a web-based discussion forum where cat-lovers can engage in on-line discussions and view media related to these magnificent creatures. Non-registered users should be able to view, but not modify, the content. Registered users should be able to post questions, answers, photos, and videos – as well as manage their accounts. The system should also provide administrative functions for managing discussions, uploaded media, and users. We also aspire to provide reporting capabilities that will help report viewers make informed decisions.

1.2 Executive Summary (Mar 30)



2 Development Plan

2.1 Planned Features

| | Non-registered | Registered | Administrator |
|---|----------------|------------|---------------|
| General Design | | | |
| Page header with contextual menus | N/A | N/A | N/A |
| Three column layout | N/A | N/A | N/A |
| Responsive design (small, medium, large) | N/A | N/A | N/A |
| Collapsible items without page reload | N/A | N/A | N/A |
| Database | | | |
| Data storage in MySQL | N/A | N/A | N/A |
| Server-side scripting using PHP | N/A | N/A | N/A |
| Store user profile images | N/A | N/A | N/A |
| Store discussion photos and videos | N/A | N/A | N/A |
| Validation and Security | | | |
| Form validation using JavaScript | N/A | N/A | N/A |
| Appropriate data security | N/A | N/A | N/A |
| Appropriate error handling | N/A | N/A | N/A |
| Self-signed certificate | N/A | N/A | N/A |
| Hosting | | | |
| Access site on external server | ✓ | ✓ | ✓ |
| Discussions | | | |
| Browse discussions | ✓ | ✓ | ✓ |
| Sort discussions by date or popularity | ✓ | ✓ | ✓ |
| Search for discussions by keyword | ✓ | ✓ | ✓ |
| Create new discussion | | ✓ | ✓ |
| Modify existing discussion | | ✓ | ✓ |
| Respond to discussion | | ✓ | ✓ |
| User Accounts and Login | | | |
| Create new account | ✓ | | |
| Login to existing account | ✓ | | |
| Sign out of currently logged in account | | ✓ | ✓ |
| Maintain user state | | ✓ | ✓ |
| View or update profile information | | ✓ | ✓ |
| Upload profile picture | | ✓ | ✓ |
| Password recovery through email | | ✓ | ✓ |
| Administrator Portal | | | |
| List all users | | | ✓ |
| Search users by name, email, or post | | | ✓ |
| Enable and disable users | | | ✓ |
| Edit or remove discussions and media | | | ✓ |
| Database restore with SQL script | | | ✓ |
| View reports and charts on usage activity | | | ✓ |
| Track comment history from user perspective | | | ✓ |



2.2 Planned Layout **NEW**



The website uses a multicolumn layout using flexbox.

2.3 Color Scheme **NEW**

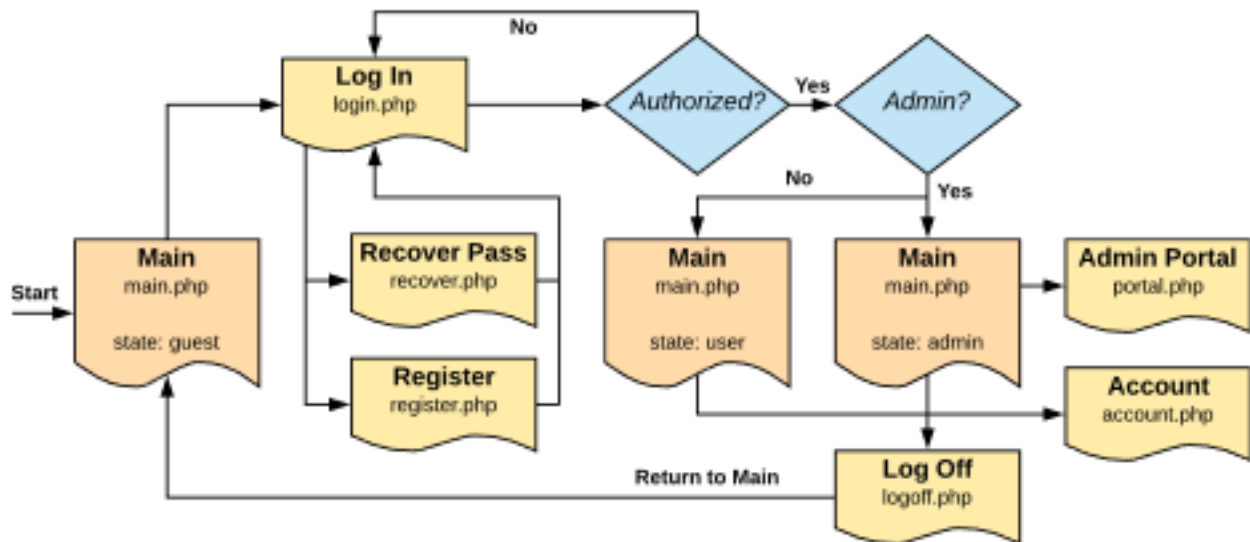
The website uses an *adjacent color scheme* created on paletton.com:

| | | | | | |
|----------------------|--------------------------------|--------------------------------|--------------------------------|--------------------------------|--------------------------------|
| Primary color: | <div>#FF6600 #FFA400</div> | <div>#FF9339 #FF8339</div> | <div>#FF7400 #FF7400</div> | <div>#C55900 #C33900</div> | <div>#9B4600 #9B4600</div> |
| Secondary color (1): | <div>#FFC800 #FFC863</div> | <div>#FFD139 #FFD139</div> | <div>#FFA400 #FFA400</div> | <div>#C58300 #C58300</div> | <div>#9B6700 #9B6700</div> |
| Secondary color (2): | <div>#FF3939 #FF6363</div> | <div>#FF3939 #FF3939</div> | <div>#FF0000 #FF0000</div> | <div>#C50000 #C50000</div> | <div>#9B0000 #9B0000</div> |



3 Client-side Experience

3.1 Site Map **NEW**



3.2 User Engagement **NEW**

All site guests are automatically directed to the main page where they can view existing discussions. The newest discussions are displayed at the top of the feed by default. A guest can proceed to the login page by clicking the [log in](#) link in the top right hand corner of the page. From the login page, the user can choose to create an account, log in to an existing account, or recover their password. A unique email address is used as the username for all accounts.

Upon creating a new account, the user is sent an email confirming their registration and automatically returned to the main page – now logged in to their new account. Upon choosing password recovery, the user is sent an email with a new randomly generated password.

Once logged in as a registered user, the log in link is replaced with the account username and core user functions are made available. Additional administrative functions are made available if the user has the administrator account type. Further, administrators are given access to a separate portal where more advanced administrative functions can be found.

At any time, a user can modify their account information – including their password – by clicking on their username and selecting the [edit profile](#) link from the interactive drop down menu. Alternatively, the user can logout by selecting the [log out](#) link from the menu instead.

After logging out, the user is returned to the main page in the initial guest state.



3.3 Validation and Security **NEW**

When a user registers for an account, they are required to provide an email address which can uniquely identify them. The syntax of the provided address is validated by using the forms email input type. When the registration is complete, an email is sent to the provided address to confirm registration.

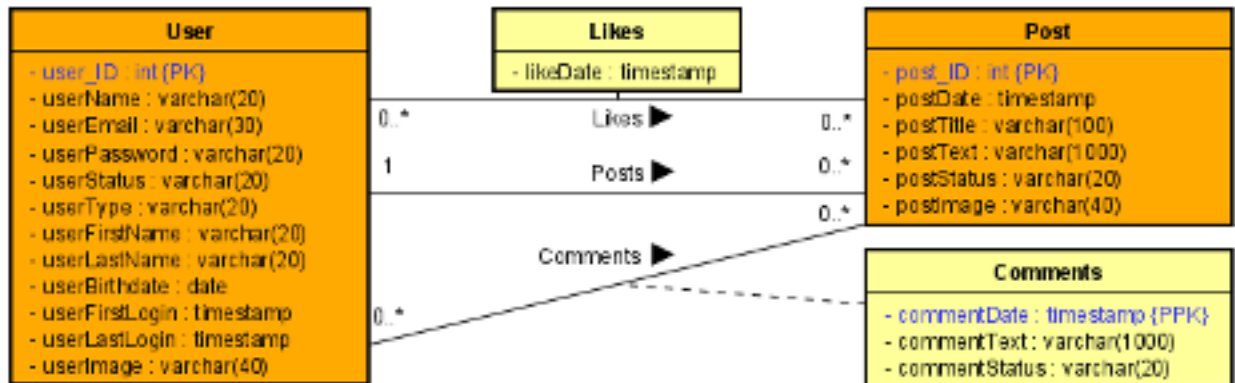
Javascript is used to validate other fields within the registration page that are not validated through input types.

A self-signed SSL certificate is implemented with the website to simulate a secure connection.



4 Server-side Implementation

4.1 UML Diagram



4.2 Entity Descriptions (Mar 6)

4.3 Relationship Assumptions (Mar 6)



4.4 Relational Schema DDL

```
DROP TABLE IF EXISTS Comments;  
DROP TABLE IF EXISTS Likes;  
DROP TABLE IF EXISTS Post;  
DROP TABLE IF EXISTS User;
```

```
CREATE TABLE User (  
    user_ID INT AUTO_INCREMENT,  
    userName VARCHAR(20) UNIQUE,  
    userEmail VARCHAR(30) UNIQUE,  
    userPassword VARCHAR(20) NOT NULL,  
    userStatus VARCHAR(20) DEFAULT 'Offline',  
    userType VARCHAR(20) DEFAULT 'Basic',  
    userFirstName VARCHAR(20) NOT NULL,  
    userLastName VARCHAR(20) NOT NULL,  
    userBirthdate DATE NOT NULL,  
    userFirstLogin TIMESTAMP DEFAULT NOW(),  
    userLastLogin TIMESTAMP DEFAULT NOW(),  
    userImage VARCHAR(40),  
    PRIMARY KEY (user_ID)  
);
```

```
CREATE TABLE Post (  
    post_ID INT AUTO_INCREMENT,  
    postDate TIMESTAMP DEFAULT NOW(),  
    postTitle VARCHAR(100),  
    postText VARCHAR(1000),  
    postStatus VARCHAR(20) DEFAULT 'Visible',  
    postImage VARCHAR(40),  
    user_ID INT NOT NULL,  
    PRIMARY KEY (post_ID),  
    FOREIGN KEY (user_ID) REFERENCES User(user_ID) ON DELETE CASCADE ON UPDATE CASCADE  
);
```

```
CREATE TABLE Comments (  
    user_ID INT,  
    post_ID INT,  
    commentDate TIMESTAMP DEFAULT NOW(),  
    commentStatus VARCHAR(20) DEFAULT 'Visible',  
    PRIMARY KEY (user_ID, post_ID, commentDate),  
    FOREIGN KEY (user_ID) REFERENCES User(user_ID) ON DELETE NO ACTION ON UPDATE CASCADE,  
    FOREIGN KEY (post_ID) REFERENCES Post(post_ID) ON DELETE NO ACTION ON UPDATE CASCADE  
);
```

```
CREATE TABLE Likes (  
    user_ID INT,  
    post_ID INT,  
    likeDate TIMESTAMP DEFAULT NOW(),  
    PRIMARY KEY (user_ID, post_ID),  
    FOREIGN KEY (user_ID) REFERENCES User(user_ID) ON DELETE NO ACTION ON UPDATE CASCADE,  
    FOREIGN KEY (post_ID) REFERENCES Post(post_ID) ON DELETE NO ACTION ON UPDATE CASCADE  
);
```



5 User Walkthrough

5.1 Web Access

The website is hosted on a *Raspberry Pi 4* web server that uses:

➔ Raspbian (Linux), Apache, MySQL, and PHP

<http://209.121.94.31:1337/cosc360/scratchpost/main.php>



The website will also be accessible on the UBC network at an address that will be supplied later.

5.2 Default User Accounts (Mar 30)

5.3 Non-registered User Pages (Mar 30)

5.4 Registered User Pages (Mar 30)

5.5 Administrator Pages (Mar 30)

