System Design Document

RoboNews.com

“Your Source For Robotics Coverage”

Client

William R. Martin

Team 3

Sven Rivera
Chad Auld
Gaurang Bhatt
Kyle Sprankle
Austin Duff

Start Date: 2/25/2013

RoboNews.com
System Design Document

Table of Contents

1. Introduction

1.1 Purpose

The purpose of this document is to explain the design of the RoboNews review website. We will cover the high-level and low-level system architecture, as well as the persistent data design.

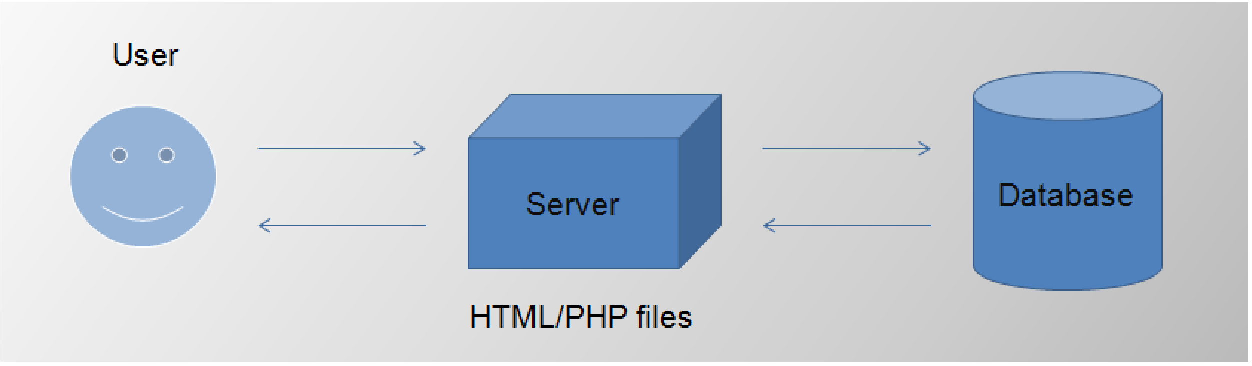
1.2 References

1. RoboNews System Requirements Document
2. RoboNews UI Design Document

2. System Architecture

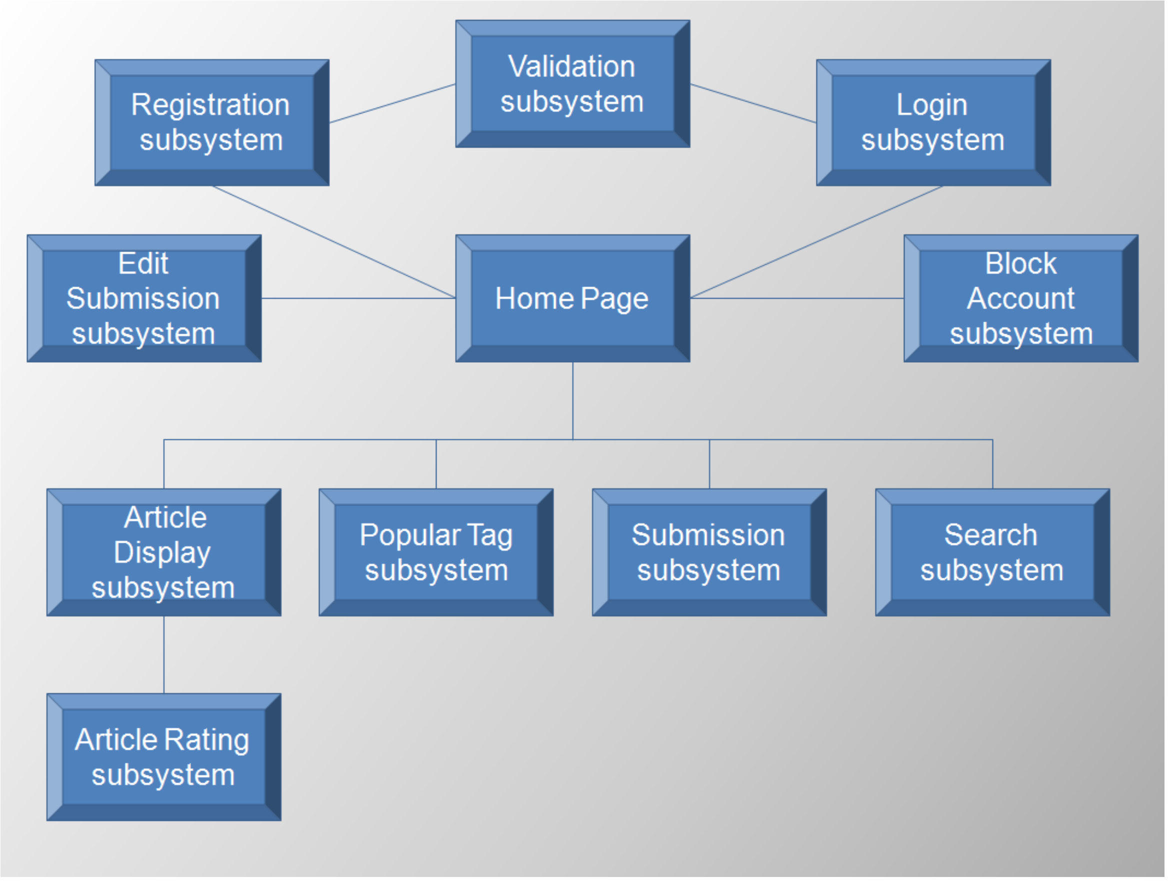
2.1 Architectural Design

Technology Architecture Diagram



The Server consists of PHP and HTML files that respond to requests from the user. These files interact with other files on the server or communicate with the database to handle requests. The purpose of individual files on the server and tables in the database will be explained in greater detail later on in the document.

System Architecture Diagram



As mentioned previously, the RoboNews review website will be created using PHP and HTML, along with a database to store data. The starting component is the home page. There are two factors that determine which subsystems/features a particular user can access: their account type, and whether they are logged in or not.

Users who are not logged in do not have access to the entirety of the website, but can still access some of its core features. They can access the Login subsystem to sign in to an existing account, or the Registration subsystem to create a new account. Both of these subsystems make use of the Validation subsystem to check user input for errors. Basic users also have access to the Article Display subsystem, which selects the most popular articles to be posted on the front page. The subsystem looks at factors like user ratings and article publication dates to make these selections. Basic users can also use the Search subsystem to locate other articles on the website related to a search query.

Once a user has logged in, they will have access to several other features. The first of these is the Submission system. The Submission subsystem allows users to submit a new article to the website along with their personal review/summary of the article. Articles are also given tags—which are similar to categories—during the submission process. These tags are used in the Search subsystem to locate articles that fall under a specified tag (e.g. locomotion). The Article Rating subsystem allows users to “upvote” or “downvote” an article. A previously mentioned, the Article Display subsystem considers user ratings and several other factors to decide which articles should be pushed to the site’s front page. The Popular Tag subsystem behaves in a similar manner, though on a broader scale. It considers the popularity of all articles within a tag and displays the most popular tags at the top of the front page.

Administrative users have additional abilities beyond those of regular users. First, they have the ability to ban/block user accounts through the Block Account subsystem. Blocked accounts cannot login to the site until they have been unblocked. Admins can also edit articles that have been previously submitted through the Edit Submission subsystem. Essentially, admin users moderate the site and filter out inappropriate content.

2.2 Decomposition Description

File Name	Description
index.php	All content and forms located here, each form is displayed as needed using AJAX.
newUser.php	Processes new account information, creates new account in the database.
login.php	Processes login information, creates session for user.
logout.php	Destroys user session.
submit.php	Processes new submission information, creates new submission in the database.
tags.php	Determines which tags are the most popular (top 5).
no_search.php	Determines which articles are the most popular.
search.php	Processes search query, finds all related articles.
ban_new.php	Blocks or unblocks a specified user.
selectSubmission.php	Allows admin to select an article to edit from list of all articles.
edit.php	Modify fields of a specific article.
CommonMethods.php	Frequently used database methods.

The basic functionality of each PHP file is explained in the table above. The relationships that exist among these files will be explained below.

The CommonMethods.php file is used by any file that requires database access. It contains two significant methods regarding database operations: connecting to the database and executing SQL queries. Files such as login.php and newUser.php make use of CommonMethods.php to access existing records or add new records in the robo_users table.

Another relationship exists between index.php and regUserLogin.php. The index.php page simply provides a login form. The information entered in this form is processed by regUserLogin.php, which connects to the database and verifies the login credentials given by the user. If incorrect information is given, the user will be redirected to the login page and prompted to re-enter their username and password.

The files selectSubmission.php and edit.php are also directly related to each other. When an admin chooses to edit an article, they must first select the desired article. SelectSubmission.php displays every article in the database in tabular format. After locating the article in the table, the admin can enter the ID number of the article and press submit. They are then redirected to a form where they can edit attributes of that article (including the title, review, and image url). Edit.php processes any changes made and updates the database.

3. Persistent Data Design

3.1 Database Descriptions

All persistent data is stored in a MySQL database. The two tables below go into detail about the specific fields and datatypes of each table in the database. Each record in the robo_links table represents an article and all of its attributes. Similarly, each record in the robo_users table is representative of a single user of the website.

robo_links database table

Field	**Datatype**	**Description**
ID	int	unique identifier
Submitter	tinytext	username of submitter
title	text	title of article
url	longtext	original url of article
views	mediumint	number of views on our site
rating	float	popularity rating of article (affected by up/downvotes)
tags	longtext	list of tags (comma separated)

review	longtext	review/summary of article
pic_url	text	url of picture (optional field)
DateAdded	timestamp	time of creation in database

robo_users database table

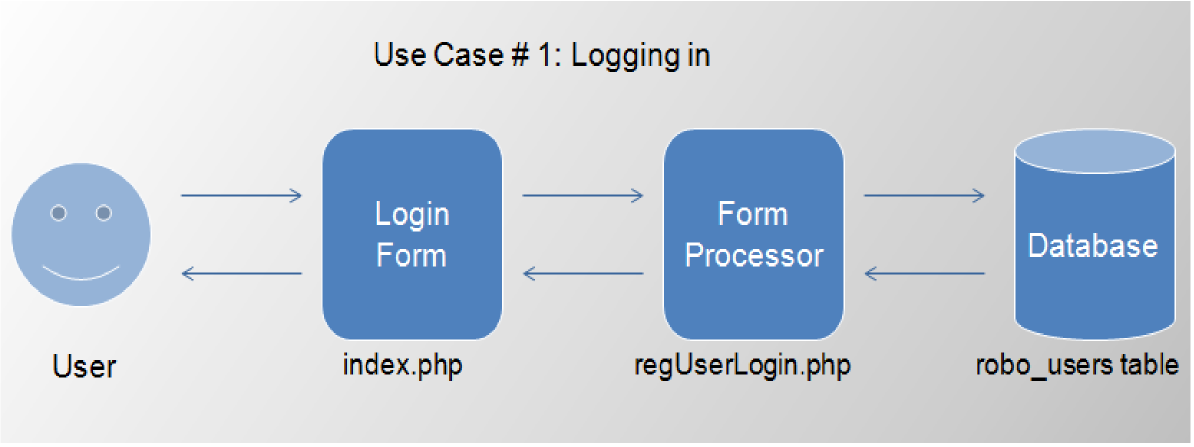
Field	**Datatype**	**Description**
user_name	tinytext	user’s login name
password	tinytext	user’s password
email	text	user’s email address
type	tinyint	account type (regular or admin), default is 0 (regular)
blocked	tinyint	0 = not blocked, 1 = blocked (default is 0)

robo_tags database table

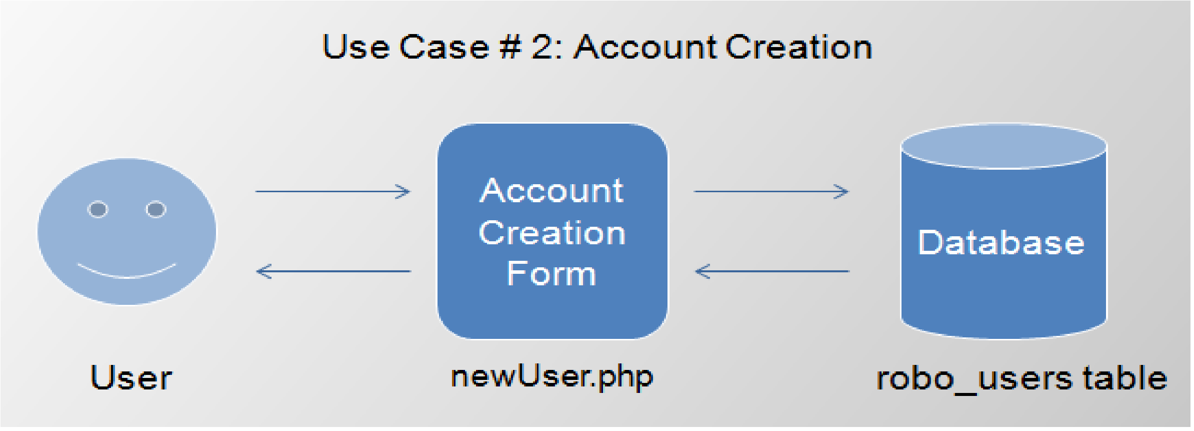
Field	**Datatype**	**Description**
TagName	text	the tag itself (ex: locomotion)
count	int(11)	number of articles with this tag

4. Requirements Matrix

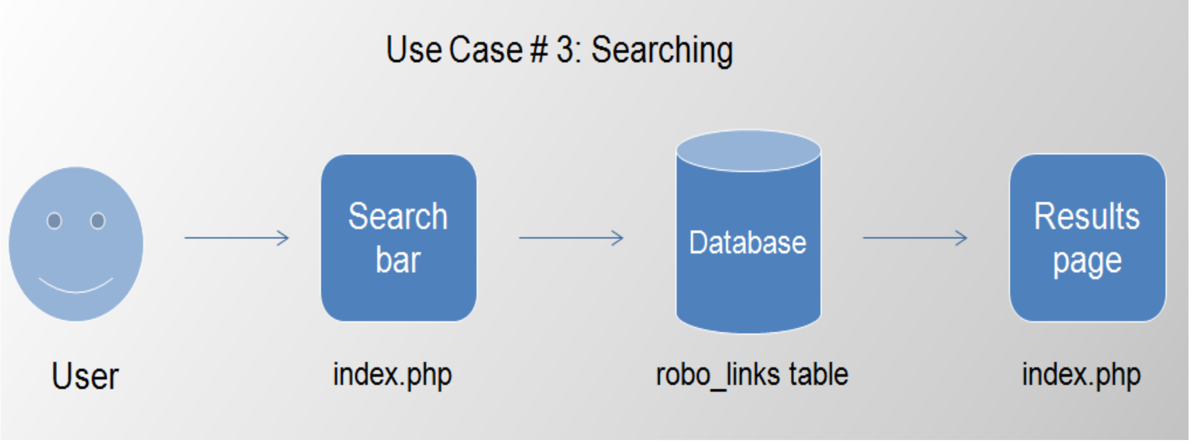
Refer to the System Requirements Document for specific details about each use case.



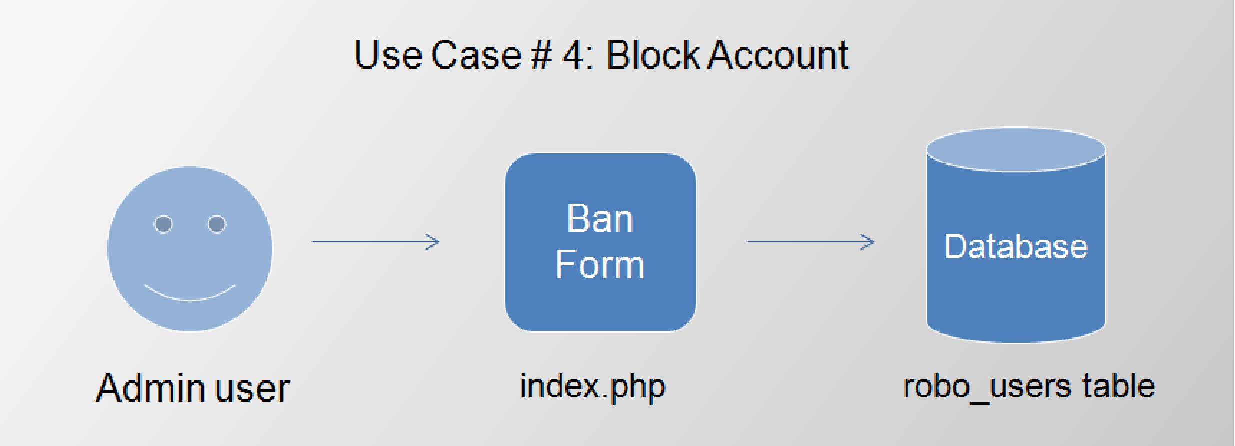
Use Case # 1: Logging in



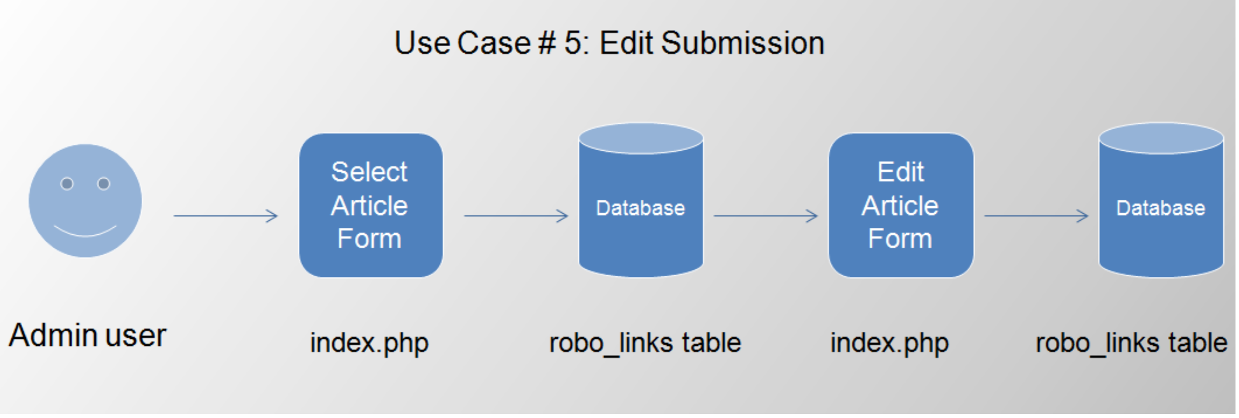
Use Case # 2: Account Creation



Use Case # 3: Searching



Use Case # 4: Block Account



Use Case # 5: Edit Submission

Appendix A – Agreement Between Customer and Contractor

The customer agrees to a Robotics site that compiles relevant news and informational articles. Articles will be submitted by users of the site, though the team will research and attempt to include a web trawler if time permits. The site will also include search functionality through the use of tags (i.e. categories). Users of the site will also be able rate articles using an upvote/downvote system. The most popular articles and tags will be presented on the front page. The team will continue to implement these features in further development spirals.

Client

Name (signature): ______________________________
Name (print): ______________________________
Date: ______________________________

Team

Name (signature): ______________________________
Name (print): ______________________________
Date: ______________________________

Name (signature): ______________________________
Name (print): ______________________________
Date: ______________________________

Name (signature): ______________________________
Name (print): ______________________________
Date: ______________________________

Name (signature): ______________________________
Name (print): ______________________________
Date: ______________________________

Name (signature): ______________________________
Name (print): ______________________________
Date: ______________________________

Appendix B – Team Review Sign-off

All members of the team have reviewed this document and agree on its content and format. Any minor disagreements are to be listed in the comments section below.

Client

Name (signature): ______________________________
Name (print): ______________________________
Date: ______________________________

Team

Name (signature): ______________________________
Name (print): ______________________________
Date: ______________________________

Name (signature): ______________________________
Name (print): ______________________________
Date: ______________________________

Name (signature): ______________________________
Name (print): ______________________________
Date: ______________________________

Name (signature): ______________________________
Name (print): ______________________________
Date: ______________________________

Name (signature): ______________________________
Name (print): ______________________________
Date: ______________________________

Comments:

____
____
____
____

Appendix C – Document Contributions

This document was completed mainly by Chad Auld. All group members assisted in reviewing and editing the document.