

Code Inspection Report

Client

Michael Wilson

Team 3

Minhaz Mahmud

Seth Mosgin

Kiante Brantley

Giancarlo Mogliazzi

Ines Ngassa

FeedMe
Code Inspection Report

Table of Contents

	<u>Page</u>
1. Introduction	3-4
1.1 Purpose of This Document	3
1.2 References	3
1.3 Coding and Commenting Conventions	3
1.4 Defect Checklist	3-4
2. Code Inspection Process	5-6
2.1 Description	5
2.2 Impressions of the Process	5
2.3 Inspection Meetings	6
3. Modules Inspected	7-8
4. Defects	9
Appendix A – Peer Review Sign-off	10
Appendix B – Document Contributions	10

1. Introduction

1.1 Purpose

The purpose of this document is to review and discuss our inspection of the code for our FeedMe site. The intended readership is anyone interested in our coding processes. We will describe the process that we took to look over our code, as well as the best and worst parts of our system. We specify the modules that have been inspected by name and description, and we state any defects that arise during our inspection.

1.2 References

1. System Design Document
2. Testing Report

1.3 Coding and Commenting Conventions

We attempted to go with using all lowercase names with underscores for naming conventions of variables, classes etc. The first letter of classes always capitalized, and variables are lowercase. It appears that using the underscore is the standard for PHP and that is why we chose it. We did our best to follow the coding conventions of PHP, JavaScript, CSS, HTML, and SQL.

1.4 Defect Checklist

Defects can cause various problems from failed logins to poor performance. We used the categories listed below. Possible defects are also listed below. For testing, refer to the Testing Report.

Defect Categories

Categories	Explanation
Conventions/Style/Patterns	Making sure to decouple code, follow standard naming conventions and using built-in functions whenever possible.
Logic Errors	When the code isn't doing what is intended, but there are no errors (browser errors, or syntax).
Security Issues	Places in the code that could be vulnerable to attacks, such as SQL injection. Vulnerabilities in frameworks and software.
Commenting	Comments that are missing or unclear, this may cause problems for other team members.
General Errors	Specific parts of a code/page that may give an error when tested.

Possible Defects

Defect Category	Defect
Convention/Style	Some variables do not follow the standard naming format.
Convention/Style	Indentation in some files is not correct and leads to difficulty reading code
Convention/Style	Improper method/class naming and inconsistent file naming
Convention/Style	Some modal dialogs may have fonts that are too small.
Logic Error	Banned user allowed to log in
Logic Error	Verify error messages are displayed for the correct errors that initiated them
Logic Error	Comparisons of strings and objects are done correctly.
Security	Verify all strings are escaped to prevent SQL injection
Security	Prevent users from accessing restricted content through URL manipulating
Security	Make sure all GET and POST fields are cleaned before using them.
Commenting	Any code that is not obvious is commented
Other	Db connections not cleaned up

2. Code Inspection Process

2.1 Description

We made sure to inspect the code as we were writing it. We did a code review with various team members to verify the code that was written was doing what was intended. All code was committed often to github so that other team members may inspect it.

We actively communicate through email, facebook, iMessage, and Github (through tickets) to make sure everyone knows what is going on. A version of the web-page is synchronized every minute on two servers so that all team members can see the latest version of the website at all times.

2.2 Impressions of the Process

We have made significant progress in the project during this spiral and have effectively created sub-teams that work on separate components of the page. Minhaz and Seth are overseeing all backend code and systems for the page and Kiante, Giancarlo, and Jessica are working on the user interface side of things. The backend group made sure to convey what they needed to the front-end group to make sure everything worked properly. We did a great job communicating but nonetheless we did experience some communication problems. Some user-interface pages were not exactly what was expected and some changes in the back-end forced some changes in the user-interface. Despite the problems we came together and fixed the problems and were able to get everything settled. In the future we will more effectively communicate exactly what is required of each person so that no surprises happen.

The areas of the code that are most likely to contain defects are the review comments and and user profile pages and maybe the search. The user profile was overhauled and the the comment system is very new and hasn't been thoroughly tested.

2.3 Inspection Meetings

For each code inspection meeting that was held, specify the date, location (virtual as well as physical), time started, time ended, who participated, who performed which roles (e.g., moderator, scribe), and which particular code units were covered in the meeting.

Date	Time	Members present	Content
4/13/14	6pm-9pm	All Members	Discussed what needed to be done for the next spiral.
4/21/14	6:45pm-7:10pm	All Members	Discussed progress so far and possible conflicts
4/24/14	9pm - 10pm	Minhaz, Seth	Discussed DB changes and division of labor
4/26/14	6pm-8pm	Giancarlo, Kiante	Discussed what UI changes were necessary
4/28/2014	9pm-11pm	All members	Discussed what needed to be finalized for the spiral
4/29/2014	9pm-11pm	Minhaz, Kiante, Seth	Worked on tying in the front end with the backend code.
Class Meetings	5:30pm-6:45pm	All present members	We met and discussed progress and what needed to be done during class meeting days.

3. Modules Inspected

Our website is based on a mix of PHP, HTML, and SQL, with small inclusions of CSS and JavaScript. We have made a few edits to the design format of our website, which required the use of a few new files and deletion of a few files that became unnecessary. Refer to the Table for a list of our files and brief descriptions of each.

Code Description Table

#	File	Brief Description
1	index.php	Make sure homepage content is displayed properly and all links work
2	register.php	Front-end for register page for new users - Form
3	register-exec.php	Handles the registration logic for new users, makes sure all fields are valid, escapes all strings, writes to database
4	login.php	Login form for users (front-end)
5	login-exec.php	Handles login logic
6	logout.php	Terminates the login session.
7	submitReview.php	modal dialog (form) used for submitting a restaurant review
8	submitReview-exec.php	Handles the logic to submit a review and add it to the database
9	restaurant.php	Make sure restaurant info is displayed properly and reviews show up
10	restaurantTimeLine.php	populates the restaurant reviews for a given restaurant
11	password.php	compatibility library to handle password hashing/verifying
12	dbconnect.php	creates a database connection
13	profile.php	Contains the logged in user's profile information
14	profile-exec.php	Handles the logic for updating a logged in user's biography
15	confirm.php	Handles logic for email confirmation for a user
16	restaurantComment.php	Displays the comments on a particular review
17	search.php	Handles the search requests and forwards them to the appropriate filters to do the searching
18	searchrestaurants.php	handles the logic for searching a restaurant and displaying it
19	searchreviews.php	handles searching for reviews and displays results

20	searchusers.php	handles searching for users and displays results
21	moderate.php	php code for flagging and banning
22	moderate-helper.php	javascript functions to help with moderation
23	vote.php	php code to handle upvotes/downvotes
24	css	style sheets
25	js	javascript files for special functions
26	scripts	contains php helper functions
27	img	contains images/icons
28	font-awesome-4.0.3	font pack for some of our icons

4. Defects

Defect Table

Category	Location	Details	Status
Correctness	profile.php	User profile not updated properly	Fixed
Correctness	register-exec.php and login-exec.php	user name was case sensitive, changed that	Fixed
Correctness	submitReview.php	Reviews not validated for size or content on server side	Fixed
Correctness	profile-exec.php	Does not validate that a change was made before submitting.	Fixed
Style/User Interface	register.php, login.php	Too much whitespace	Fixed
Security Oversight	profile.php, restaurant.php	\$_GET variable attributes are not cleaned to before using them, possible SQL injection.	Fixed
Other	css/*.css	css styles have conflicting duplicates that may cause a performance hit on the site.	Fixed
Other	images	Images for reviews are not currently cropped which may result in a performance hit.	Unfixed.
Correctness	profile.php	When users are logged into multiple sessions and update their profile one session will update while the other will not until the user logs out and logs back in.	Unfixed
Correctness	submitReview.php	When a review is submitted a period is appended to the review.	Unfixed

Appendix A - Team Review Sign-off

Name:	Date:	Signature (Print name)
Minhaz Mahmud	04/29/2014	Minhaz Mahmud
Kiante Brantley	04/29/2014	Kiante Brantley
Seth Mosgin	04/29/2014	Seth Mosgin
Ines Ngassa	04/29/2014	Ines Ngassa
Giancarlo Mogliazzi	04/29/2014	Giancarlo Mogliazzi

Appendix B – Document Contributions

This document was completed by Minhaz Mahmud for the most part with help from various team members in various capacities.