**INTE1070/1071 Secure Electronic Commerce Assignment 2**

Documentation

**Group:**

Richard Liu s3168087

Nisha Tobgui s3488056

Site Construction Procedure

# Platform

The e-commerce website is hosted using the GoDaddy web hosting service which offers a Linux platform.

PHP was installed on the server with the selected version being 5.5. No PEAR packages were utilized. The MySQL extension mysqli was used to provide greater SQL security.

MySQL was installed to serve as the database system for the website. The version is 5.5.49. The website uses a single database accessed via one user.

# External libraries

The website makes use of the following libraries and frameworks:

* Materiallize: Responsive front-end framework by Google based on material design. Ensures the site is mobile friendly and conforms to modern UI design practices.
* jQuery: Framework for simplifying client-side scripting and AJAX procedures.

# Layout

The storefront is designed as a standard e-commerce website with a banner at the top of the page providing buttons for crucial services such as accessing the cart.

The main content is separated into a side navigation section and products section. The side navigation provides access to pages of lesser importance such as FAQs and also buttons for filtering the products (e.g. categories).

# Session state

Session state is stored in the session variable provided by PHP. This includes storing the cart, logged-in user and current order.

Security Features

# HTTPS deployment

Only using HTTPS makes the risk of man-in-the-middle attacks negligible as communications between the client and server are encrypted. The encryption is provided using a SSL certificate signed by the GoDaddy certificate authority.

Any attempt to use the non-secure website is stopped and the user will be redirected to the secure website.

# Server-side sanitization and validation

All user input is passed though the htmlspecialchars() function. This removes html tags and other special entities preventing XSS attacks.

Aside from validating input client-side, server-side input validation is also performed to prevent invalid data getting into the database.

# Prepared statements

The mysqli library is used for connecting with MySQL due to its support for prepared statements. Prepared statements is a standard method for completely eliminating SQL injection.

# Directory security

## Hidden directories

All asset directories are hidden using a index.php file which shows a 404 page which keeps the directory structure private. This is an example of security through obscurity.

## Forbidden directories

Directories used only by the server are disabled from the user via htaccess to prevent outside users from learning about the internal directory structure.

# Password security

## Strong passwords

Users are required to use strong passwords containing a combination of lowercase, uppercase, number and symbol characters. This minimizes issues around hacked accounts and having accounts which can be easily accessed by intruders.

## Hashed passwords

Passwords are stored using modern hashes with salts and never in plain text format eliminating the risk of user passwords being made public in the event of network intrusion.

This also prevents even administrators from being able to acquire user passwords preventing insider attacks.

# reCAPTCHA

A captcha test which requires physical user interaction has been placed on the registration page to prevent bots from automatically registering accounts on the website.

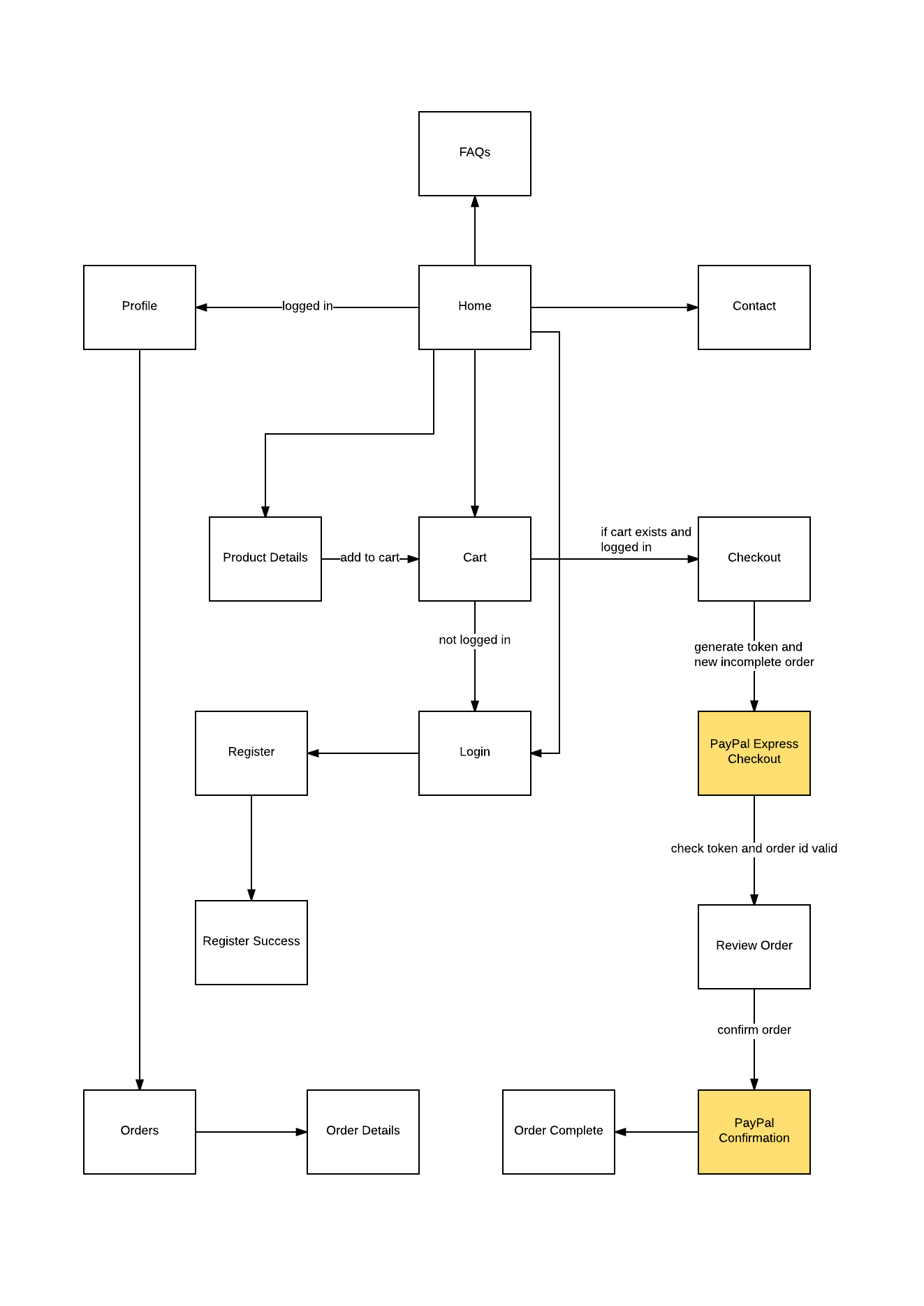
Captcha is shown on login page after a number of repeated failed login attempts to prevent brute force password attacks.

# Hide error messages

Visible error messages can reveal crucial information to users about a websites internals such as database schema. Thus error\_reporting(0) is used to hide any and all errors including warnings.

# PayPal single-use token

The API exposed by PayPal is makes use of single-use tokens and thus prevents replay attacks of orders.

Block Diagram