# 

# 

# 

# 

# Hells App Permissions

# Timbits

# ENSE 470

# Code File Outline

# 1.0 Models

In this folder we have all data templates as well as the database configuration file.

## 1.1 dbConfig.php

This file has all necessary data to connect to the server. Make sure this file is required in any view or controller that requires database connection. The variable you will need for the connect is $db.

## 1.2 userRequests.php

Does nothing right now. Made for future if we are going to use pagination to flip through the requests instead of only showing top 5 or all requests available.

## 1.3 user.php

This file is a data template used to store the required user information as a session variable. It contains most of the data from the Users table in the database. This can tell the page whether the user is a general user, analyst, or approver and what their name, email, and user id is.

## 1.4 request.php

Long model with many variables, but contains all data required to show the request in the View.php view (shows the full details of request). The constructor is long but again this class contains all required information to fill out that long form.

# 2.0 Views

The views are all the front end visuals that you see on the site. They contain some php logic where required, but mainly consist of html code.

## 2.1 navbar.php

This is the base template for the navbar used in the program. It should be “included” in every page (other than index.php) and contains links to the home page, request form page, and a logout button that destroys the session variables and takes you to the index page.

## 2.2 index.php

While not in the iews folder but the root folder instead, index.php is still an important view that handles the user’s login and sign up capabilities. The login and signup forms are split into two different tabs. This is the only page that does not contain the navbar as well because it’s hard to navigate a site you aren’t logged into yet.

## 2.3 home.php

This is the home page that displays all current tickets to be looked at. For every user, there is a closed request table and a open request table. For analysts and approvers there will also be a requires validation table so they can see what requests need to be approved. Every ID number on this page can be clicked to open the view.php file with the proper request (first send request to the openRequest.php page to get the request variable setup).

For future versions this page will allow pagination capabilities and could show more data upon user request.

## 2.4 request.php

This is the form a user can fill out to submit a new request. Once finished it goes to the submitRequest.php page to insert request into database.

## 2.5 view.php

This page allows the user to view in detail a single request. Analysts and approvers will also be able to see approve and deny buttons depending on where the request is in the process. It also shows the ticket status to all users with access.

## 2.6 style.css

This simply contains the main css used to style our website. This css document should be included on every page to ensure consistency.

# 3.0 Controllers

The controllers handle most, if not all, of the logic for the views and are used to connect the models to the view.

## 3.1 approve.php

When an approver or analyst presses the approve button it gets sent here, where it changes the value in the database and sends the user back to the view page with the new information.

## 3.2 deny.php

Same as approve but for the deny button, automatically closes ticket as well.

## 3.3 login.php

Allows user to login to the website, if something doesn’t work it sends the user back to the index page.

## 3.4 logout.php

Destroys session and redirects user to index page.

## 3.5 openRequest.php

Gets sent a request ID and creates a new request model and fills in all of the required data for a request. Sends user to the view page right after.

## 3.6 signUp.php

Same as login, but for the sign up form instead.

## 3.7 submitRequest.php

This page is called from the request view and is used to submit a new request into the database. This page contains a lot of code since you have to take the approver name to determine their ID and same with the APP name and the User name. It collects all of this data then inserts it into the Permission Requests table in the database.