Project 2: Cyber Security Project

Contents

Leave Application System

Overview

Develop a Leave Application System, in the form of a dynamic website linked to a database, whereby employees can apply for specific types of leave. Their respective managers should then be notified and be given the ability to either approve or reject the application and add an optional reason for the choice taken. The employee should then be notified of the outcome.

Requirements

Dynamic Website

The system must be in the form of a dynamic website. Data displayed on pages must be retrieved from a database on a server.

Database

The system must contain a database to store all relevant data e.g. users/employees, leave applications etc.

Passwords must be stored in an encrypted format and not as plain text.

User Authentication & Authorisation

Each employee should be able to log into the system using a username and password. The employee should be able to **ONLY** see his/her data e.g. previous leave applications, managers should only be able to view employees reporting to them.

You should use role-based authorisation e.g.

- Employee role which will be able to do leave applications and approve leave if any employees are reporting to them
- Administrator Role which will be able to create/edit employees, create/edit leave types, update employee leave balances per employee.

Functionality

Administrator

The administrator should be able to:

- Create employees (CRUD)
- Create leave types (CRUD)
- Set balances for leave types per employee (optional)

Managers & Employees

An employee should not have access to the functionality of the administrator. Employees should be able to:

- View own leave balances (optional)
- Apply for specific types of leave for a date range
  - Number of days should be calculated from the range selected.

- View pending leave approvals
  - An employee should not be able to approve his/her own leave
  - An employee should be able to cancel leave applications

- View historic leave applications and outcomes (approved/rejected)
- View employees reporting to him/her if any
  - View pending leave applications of subordinates that needs to be approved or rejected.

Security

Implement some or all, of the following security measures:

- Prevent malicious user input
  - Validate user input
- Access database securely
  - Guard against Sql injections
- Store sensitive information securely
  - Encrypt passwords or an employee's ID number (South African ID number)
- Use Cookies securely
  - Assume that cookies will be enabled on the client browser
- Create safe error messages e.g.
  - 404 page not found
  - 500 server error etc.
  - Do not display full stack error messages to the user
- Guard against DOS attacks (Denial-of-service)
- SSL

Reading Material

- https://msdn.microsoft.com/en-us/library/zdh19h94.aspx
- http://php.net/manual/en/security.database.sql-injection.php
- https://www.w3schools.com/sql/sql_injection.asp
- https://www.w3schools.com/php/php_form_validation.asp
  - Note: PHP Form Security
- https://www.w3schools.com/php/php_cookies.asp
- https://www.w3schools.com/php/php_error.asp
- https://css-tricks.com/snippets/php/error-page-to-handle-all-errors/
- https://www.corenetworkz.com/2015/10/prevent-malicious-user-inputs-in-php.html
- https://www.wordfence.com/learn/how-to-write-secure-php-code/
- https://www.youtube.com/watch?v=ntVeD1sQDsg

NWU ADDITIONAL GUIDELINES:

RELEVANT TOPICS

Topics that you need to learn about:

- HTML forms - HTML elements that retrieve and send user data to another page.
- Web servers - Servers that serve web content. This content may include both static and dynamic web pages. These servers can also host databases.
- PHP - A server-side scripting language. PHP can be used to dynamically generate web pages and can link to local and remote databases.
- MySQL - A Database Management System (DBMS).
- phpMyAdmin - A web interface for creating and managing MySQL databases on a web server.

SOFTWARE

The following software can be used to work on this project. However, others also exist:

- WAMP, XAMPP or EasyPHP - all-in-1 local webserver setup on Windows (inludes Apache, PHP and MySQL).
- phpMyAdmin - A web interface for creating and managing MySQL databases on a web server

Note: Please stay away from Microsoft Technologies such as C# and asp.net as well as Flash. [Unless you can get you're your project to work 100% without any issues – you take the risk]

PROJECT DOCUMENTATION

Please pay particular attention to the following documentation:

- **SPMP** (Overview,Project schedule as shown in a grant chart, roles and responsibilities, process model and technical aspect as well as tools)
- **SRS** (Overall description, specific requirements, functional requirements, performance requirements, design constraints)
- **WebApp Design Document** (Interface design, aesthetic design, content design, architecture design, navigation design component-level design)
- **Use your systems analysis documentation as a guideline for what to add to your project documentation**

WebApp Guidelines

- Database design and hosted on the Database server for ITRW 311 (196.253.4.24)
- WebApp should be hosted on the WebServer on rkv-lnx3.puk.ac.za

USEFUL LINKS

The following links serve as starting points when researching the relevant technologies:

- HTML forms, PHP, MySQL - W3 Schools
- Web servers - http://computer.howstuffworks.com/web-server5.htm
- phpMyAdmin - https://www.youtube.com/watch?v=Ty_tjx4W8MM