Zeig Software Basic – Employee Management with Door Security Enhancement

Employee Management with Badge Security Enhancement

Nicholas Zeig

INFO C-451

Jafrina Jabin

Problem Statement

Our organization faces challenges in managing employee information and with ensuring secure access to restricted areas. Traditional methods of manual record-keeping have failed to accurately control employee access. Physical key distribution is not only time-consuming but also prone to errors and security breaches. There is a pressing need for a streamlined system that simplifies employee management while enhancing security through automated door badge generation.

Proposed System:

The organization will undergo a transition to digital employee management with Zeig Software Basic, then add on the Badge Security module.

The proposed Employee Management and Badge Security module is a solution that allows organizations to manage employee data and secure access doors. The system will enable administrators to input and update employee information and generate personalized door badges equipped with secure access credentials. These badges will allow employees to access designated security doors within the facility, ensuring that only authorized personnel can enter restricted areas. Further enhancements can be made with the purchase of more modules, such as Payroll, Investments, Benefits, and Employee Self Service modules.

Objectives of the System

Enhance Security—Ensure that only authorized employees have access to specific areas.

Improve Efficiency—Eliminate most keys and replace with single badge.

Reduce Security Risks—Remove building access from any employee for any reason.

User-Friendly Interface—Provide an intuitive system that is easy for staff to use.

Build Employee Management System—Use the basic software to expand to more modules to improve employee experience.

System Requirements

- Administrators shall be able to add Doors and Power users to the system.
- Doors shall have a Security Level and a Security Group.
- Power Users shall add users to the system and assign badges.
- User Information of all free tier level Zeig Software should be entered.
- Users shall have a unique generated employee ID.
- Users shall have a Security Level and a Security Group.
- Badges shall store the user's Security Level, Security Group, and Active Status.
- Security Level should be assigned to any manager or security employee.
- Security Group shall be assigned to all employees working in a particular department.
- Doors shall open when badged if matching Security Group or minimum Security Level is presented.

Typical Customers

This system is ideal for medium sized companies with secured entrances.

Project Plan

Software Requirements:

Hardware: Cloud linux or windows machine to run application.

Database: mySQL, Postgres, MongoDB or other database to store information.

User Requirements: Users will require a web browser and email access to set up secured access to application.

Development Approach

We will employ a web-based application development approach using the following technologies:

- Programming Language-- JavaScript (with Node.js for the backend).

- Frameworks-- Express.js for server-side operations, JavaScript and HTML for the for the front end.
- Database—mySQL or MongoDB for flexible and scalable data management. --MariaDB
- SSH, Git, or perhaps GitHub to deploy to EC2 instance. –All 3 used—SSH used to connect Git Bash Terminal to EC2 Linux machine. SSH private key used to connect EC2 to GitHub.

Development Plan

Week 1: Planning

- Learn how to use Node.js, Express.js, and perhaps MongoDB(might default back to postgres or mySQL here as I'm familiar with SQL) –MariaDB
- set up a development environment on AWS using EC2 tier. –used this tutorial to set up EC2 and Express: https://www.youtube.com/watch?v=nQdyiK7-VlQ

Week 2-3: Rapid Development

- Create wireframes and design the user interface.
- Set up the database schema. –Database created, not yet in 3NF
- Create Administrator, EmployeeUser, Super User, Doors, Badge classes. –Door Table Classes not yet Created

Week 4-6: Incremental Improvement

- Improve front end appearance. –CSS will need to be improved
- Enhance back end capability by streamlining any redundant code.

Week 7-8: Testing and Deployment

- Conduct thorough testing for bugs and security vulnerabilities.
- Deploy the system by creating administrator accounts.

System is currently in planning phase. EC2 deployment and learning to use Node.js and Express.js on the fly will be the hardest part, but something I am looking forward to trying. If I run in to too many problems, I will have to default back to the WAMP stack which I'm also not proficient in.