Aeriel Denmark

CS 255 - System Analysis and Design

Bryant - Project #1

## \*due Sun 4/2/23\*

## System Components and Design

### Purpose

* The purpose of this project is to design a system for a client.
* The client for this project is DriverPass.
* DriverPass would like to create a system in the form of a website that will provide students with access to online practice exams and on-the-road training to better prepare them for driving tests.

### System Background

* DriverPass would like the system to provide students with access to online practice exams and on-the-road training to better prepare them for driving tests.
* DriverPass identified that there are very few tools that train students to pass their driving tests.
* Different components needed for the system are the ability to access data from anywhere with an internet connection; security features that grant access to different users; and a tracking system for reservations, cancelations, modifications, and different package options for on-the-road training.

### Objectives and Goals

* When the system is completed, it should be a fully functional website that allows students to book, modify, and cancel on-the-road training; allow students to take online practice exams; and provide employee admin access to make changes as needed to further improve the system.
* Measurable tasks needed to include in the system design to achieve this fully functional system include object models, process models, and UML diagrams for visualization and operating platforms and programming languages to build and create the website.

## Requirements

### Nonfunctional Requirements

#### Performance Requirements

* Environments the system needs to run in should be web-based and can be accessed in the cloud.
* The system should run fast enough so that users do not encounter performance complications.
* The system should be updated when the client is seeking to add or update website features or to provide routine maintenance for optimal performance.

#### Platform Constraints

* The platform in which the system should run on is Linux; Linux will offer cost-effectiveness, stability, security, scalability, flexibly, strong community support, and access to a wide range of software tools, making it the most attractive option for the client.
* The system should be used in the cloud for management, making support for the system easier.

#### Accuracy and Precision

* To distinguish between different users, each student will have the ability to create a username and password upon first accessing the web-based system.
* The input should be case-sensitive and offer multi-factor authentication for increased security.
* The system should inform the admin of a problem when the user forgets their password or is unable to access their account, when any downtime occurs, or when any issues are reported by its users.

#### Adaptability

* To make changes to the user (add / remove / modify) without changing code, the system can use command-line tools or web-based interfaces / control panels.
* The system will adapt to platform updates through regular system updates, application updates, and compatibility checks.
* The type of access the IT admin will need for the system will include database access, SSH access, API access, root access, log access, web-based control panel access, and FTP / SFTP access.

#### Security

* For the user to log in, the system will require a username and password with multi-factor authentication.
* To secure the connection or the data exchange between the client and the server, the system will need to complete routine software updates, use strong authentication methods, secure cookies, use strong encryption ciphers, use HTTPS, and enforce HTTPS.
* If there is a “brute force” hacking attempt, the system will monitor and log login attempts, enable multi-factor authentication, notify the account owner when multiple failed login attempts are detected, lock a user’s account after a predefined number of failed login attempts occurs, use strong password policies, and update the system’s software regularly.
* If the user forgets their password, the system will have a password reset request, the system will send a password reset email to the user, the user clicks on the password reset link, user password reset prompt, the system updates the user’s password, and the system monitors the account for suspicious activity after the password reset.

### Functional Requirements

* The system shall show the student driver who they are paired with.
* The system shall validate user credentials upon logging in
* The system shall provide students with access to online practice exams and on-the-road training to better prepare them for driving tests.
* The system shall off three different driving packages.
* The system shall allow students book, modify, and cancel on-the-road training.
* The system shall show when a student has completed training.
* The system shall run fast and efficiently.

### User Interface

* The needs of the system’s interface include secure authentication and access control mechanisms, clear navigation and organization of features and tools, regular updates and maintenance for performance and security, intuitive and easy-to-use design, responsive design to adapt to different screens and devices, and cross-browser compatibility.
* The different users for the system’s interface include the following:
  + Administrators – manage users, roles, and permissions; monitor system health, logs, and analytics; install, update, and manage software; configure system settings and security.
  + Developers – monitor and optimize system performance and security; develop, test, and deploy bug fixes and new features.
  + End-Users / Visitors – register for an account; search and filter content; view and interact with the system.
  + Content Creators – create, edit, and delete content; view analytics and reports; organize content using tags or other metadata; manage user-generated content.
* Each user will need to be able to access the system through popular web browsers.
* The user will interact with the system’s interface through a dedicated mobile app, popular web browsers and APIs.

### Assumptions

* In the design, a budget was not specifically addressed.
* Assumptions that are made about the users or the technology they have are that the system can be built within budget and the user is tech savvy enough to access the necessary technology.

### Limitations

* Limitations seen in the system design include:
  + Budget – cloud services cost and development cost
  + Resources – scalability and expertise
  + Time – development time frame and deployment and maintenance
  + Technology – software limitations, integration challenges, legacy systems, and security concerns

### Gantt Chart

Timeline

Description automatically generated