

Business Requirements Document

System Components and Design

Purpose

What is the purpose of this project? Who is the client and what do they want their system to be able to do?

- The purpose of this project is to develop a requirements document in order to outline a design for a system that will aid our client in successfully fulfilling their business objectives. Our client, DriverPass, is seeking to capitalize on a void in the driving education market by developing a system which will deliver robust driving education for their customers and facilitate successful completion of driving exams for said customers.

System Background

What does DriverPass want the system to do? What is the problem they want to fix? What are the different components needed for this system?

- As previously mentioned, the goal of this system is to help facilitate the completion of driving exams for the customers of DriverPass.
- DriverPass has observed that there exists a lack of quality driver training available to individuals seeking to earn their driving license. The client has concluded that because of this void, those pursuing their licenses often fail the driving certification examinations.
- DriverPass therefore wants a system that can handle facilitating the education of their customers. This includes the ability for customers to take online classes, practice tests, and request on-the-road training.

Objectives and Goals

What should this system be able to do when it is completed? What measurable tasks need to be included in the system design to achieve this?

- The various specific requirements and goals could potentially be divided into a number of distinct categories: accessibility, security/administration, scheduling/registration, interface
 - Accessibility: The client requires the data regarding the system to be accessible from any device (mobile/non-mobile) and locally if needed. Additionally, the system should allow the client to have access to DMV information regarding any policy changes or rules. The client should be notified of these changes through a notification setup.
 - Security/Administration: Authentication is required to be integrated into the system through the use of a password method. Some type of access control to provide authorization for a hierarchy of roles, with an administrator at the top (directory system), should be present. This administrator requires the ability to reset passwords and view logging data that tracks changes made to system (in printable format as well). Additionally, users who work for the client will require varying privileges appropriate for their role within the organization.
 - Scheduling/Registration: Users utilizing the system should have the ability to create, cancel or modify appointments through the system, with three levels of packages available to them at system launch. The package system should maintain flexibility so that packages can be modified or removed as the client's business evolves. Regarding

registration, users must be able to input their information, such as name, address, numbers, state, and credit card details as well as a pick up and drop off location.

- Interface: The interface of the system should be a web-based system, likely through a browser hosted by a cloud service provider that takes care of the infrastructure portion of the application (potentially IaaS). Client has provided a rough sketch of what a desirable configuration for the interface might look like, with an ability for users to directly input information through the interface. This interface should report the progress and status of the user's account. Additionally, pages for contact between the client and user should also be present.

Requirements

Nonfunctional Requirements

In this section, you will detail the different nonfunctional requirements for the DriverPass system. You will need to think about the different things that the system needs to function properly.

Performance Requirements

What environments (web-based, application, etc.) does this system need to run in? How fast should the system run? How often should the system be updated?

- The DriverPass system should be designed as a web-based application in order to most appropriately meet aforementioned requirements for multi-device access and be compatible with all commonly used browsers such as Chrome, Firefox, and Edge. This will ensure the system can be portable and used by the widest variety of user devices.
- The proper infrastructure and design should be in place for the system to handle all workloads, particularly during peak use times, without any significant slowdowns or queues to the users. This includes both the customers utilizing the DriverPass services and administrative bodies accessing appropriate data within the system. At minimum, the system should handle the numbers of administrative users as well as customer users registered to use services without any significant performance impact and near 100% uptime.

Platform Constraints

What platforms (Windows, Unix, etc.) should the system run on? Does the back end require any tools, such as a database, to support this application?

- The system should be a web-based application which will allow it to serve multiple platforms and ensure a high degree of portability through a browser interface.
- The options for which platform should host the system application are certainly debatable. However, a popular and sensible choice is a distribution of Linux for a variety of reasons. Unlike MacOS server, Linux does not require proprietary hardware in order to legally be installed or used, which can significantly reduce hardware investment costs. Additionally, Linux server is an open-source operating system which means it does not incur a licensing fee like Windows server architectures, which will further work to reduce costs for DriverPass.
- Server virtualizations should be allocated for each component of the system as appropriate. A database server will be required to handle user data such as passwords, account details, and appointments, while also storing administrative information such as reports and other business oriented data. Furthermore, a web/application server should be utilized in order to serve

requests to users, such as displaying graphical information and allowing users to access web resources.

Accuracy and Precision

How will you distinguish between different users? Is the input case-sensitive? When should the system inform the admin of a problem?

- All users within the system should be required to create an account and be distinguished by specific account information. Accounts will have varying degrees of authorization, with administrators having the highest level of privileges and customer users only having access to what is appropriate for them to interface with the proper educational services.
- Password data should be case sensitive, these passwords should also be hashed before being store in the database.
- Administrators should receive immediate notifications when an emergent issue, such as a downed server, occurs. Additionally, notifications should be sent when users submit tickets for issues they are encountering with the system. Any other changes, such as warnings for potential hang-ups, should be logged for administrators daily.

Adaptability

Can you make changes to the user (add/remove/modify) without changing code? How will the system adapt to platform updates? What type of access does the IT admin need?

- The system should be coded to provide functionality for the streamlined creation, modification, and removal of user objects through an interface. Customers should be added automatically once they have created an account and their payment has been processed.
- Browsers utilize web-based protocols which have been a standard for many years; however, the system will need to be updated in case of emerging new web standards, particularly those involving encryption and other security measures.
- IT administrators will require access to a directory system to manage authorization and roles, functions to reset user passwords, and modify users as needed. Additionally, admins will require access to logs detailing issues or changes within the system.

Security

What is required for the user to log in? How can you secure the connection or the data exchange between the client and the server? What should happen to the account if there is a “brute force” hacking attempt? What happens if the user forgets their password?

- Users should be required to create an account with a username and password in order to be properly authenticated and authorized within the system. These accounts will then be populated with relevant user details such as email, name, and other contact information.
- The cloud based service provider will handle appropriate encryption and keeping customer data secured. The passwords a user chooses should be hashed within the cloud password data based, so that in an event of a security breach, these passwords will not be easily decrypted.
- Users should be required to created strong passwords, with a minimum 15-character size, use of symbols, numbers, letters, and different cases. Additionally, two-factor authentication through a secondary device will provide an extra layer of protection and should be required.

- Users should be able to recover/reset their passwords themselves after providing appropriate identification information and activating a link through their on-file emails. In an event a user cannot recover a password themselves, administrators should have the ability to force this reset after verifying the user.

Functional Requirements

Using the information from the scenario, think about the different functions the system needs to provide. Each of your bullets should start with "The system shall . . ." For example, one functional requirement might be, "The system shall validate user credentials when logging in."

- The system shall send notifications through email to administrators when emergent issues arise or DMV information is updated.
- The system shall authenticate user accounts through a username and password login functionality.
- The system shall serve an interface to users to view information, progress, notes, needs, photos and contact information.
- The system shall authorize users to access course material resources, assignments upon login, access online classes and practice tests.
- The system shall provide administrators the functionality to change roles, permissions, and passwords for users through a directory interface.
- The system shall log activity and reservation reports which can be accessed by administrative bodies.
- The system shall provide customers with the ability to select an appointment time and select an appropriate package. Additionally, the system shall provide this same functionality to company personnel to schedule appointments from phone calls.
- The system shall reset a user password upon confirmation of identity when a user cannot log in.

User Interface

What are the needs of the interface? Who are the different users for this interface? What will each user need to be able to do through the interface? How will the user interact with the interface (mobile, browser, etc.)?

- The system will serve multiple platforms and therefore multiple display sizes. The user interface will need to accommodate those displays, most notably mobile user displays and desktop user displays.
- The web server delivering front-end resources will be managed by a cloud service provider and accessed through a browser.
- Once customer users are logged in, there should be appropriate displays to serve users their test progress, class resources, and provide a section to make appointments for on-the-road training as well as contacting the company for technical issues.
- Administrative users should be presented with an interface that provides information about student test progress, contact details, special notes and needs, and photos. Any appointments or reservations made by the customers should also be presented in a schedule for these users as well.

Assumptions

What things were not specifically addressed in your design above? What assumptions are you making in your design about the users or the technology they have?

- The system is assumed to be supported by infrastructure that will reliably handle all services that customers or administration request without significant downtime or performance issues.
- The licensing process and examinations will not change so drastically as to make driver training obsolete.
- The cost of the system will likely increase as the number of users or capabilities increases as well, which is common when scaling cloud-natured infrastructures.
- Customers utilizing the company's services will use devices capable of accessing web-based content.

Limitations

Any system you build will naturally have limitations. What limitations do you see in your system design? What limitations do you have as far as resources, time, budget, or technology?

- Design will be limited to a web based system with an emphasis on web based technologies and protocols.
- A working iteration of the system will have less than 16 weeks to be delivered.
- A cloud based approach will abstract many components of the infrastructure from the company, which means the integrity of backups and security will rely on the provider to handle issues.

Gantt Chart

New, more detailed chart (Right click and save as picture to view larger version)

