# CS 255 Business Requirements Document Template

## 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 client is DriverPass—a company that does training for drivers’ tests
* The system DriverPass has in mind is web interface will keep track of appointments for driver training and tests
* It will also serve as an access point for online classes

### 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?*

* The issue they have noticed is an absence of driver training resources
* The DriverPass solution is to create a web interface for driver training
* The web interface needs cross-platform compatibility
* It needs a database to keep track of appointments for training and tests
* It needs a database of online classes
* It needs payment system that allows users to choose packages
* It needs to have tiered user system with different abilities for different roles
* It must keep track of changes made by each user

### 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 system must be accessible across different devices and platforms
* It must be able to track user activity
* It must show package options
* It must show appointments to drivers
* It must allow trainees to make driving appointments

## 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?*

* This system will be web-based
* This system will be accessible across different web browsers
* UI will conform to the W3 accessibility standards
* Recommended system speeds (as advised by Core Web Vitals) will be:
  + Largest Contentful Paint: ≤ 2.5 seconds
  + First Input Delay: ≤ 100 milliseconds

#### 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?*

* This system will be accessible across internet-capable devices
* This system will be accessible across different operating systems
* The system itself will be built upon a LAMP (Linux, Apache, MySQL, PHP/Perl/Python) stack

#### 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?*

* Each user will be assigned a unique integer account number, which the system will primarily use
* Each user will provide a unique user name to login
* User name will be case-insensitive and allow select special characters to allow users to have email addresses as user names
* Passwords will be at least 8 characters long and may not contain the user’s user name
* There will be different classes of users: customer, trainer, IT, admin, and secretary
* Problems that require admin response will be triaged in a manner that prioritizes security and global functionality
* Security and global functionality problems will trigger a notification to be sent to IT users
* Problems affecting single users will not trigger such notifications
* Problems of all types will be documented in automatically generated reports

#### 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?*

* There will be a control panel for IT users to add, remove, and modify users
* The system is web-based, so the users should not need to update anything outside of their own software
* The server will be Linux based and capable of installing its own updates

#### 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?*

* Login requires user name and password
* Failed login attempts will be limited to 3—after 3, the account will be locked and will require a phone call to the secretary to unlock it
* There will be a self-service link to reset passwords if the user forgets, which will require answering a security question
* Secure connections will be enforced with SSL
* Payment processing will be handled by a third-party payment vendor that specializes in secure transactions

### 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 validate user credentials when logging in
* The system shall allow customers and trainers to make appointments
* The system shall allow customers and IT to view and choose packages
* The system shall allow IT to disable packages
* The system shall allow all users to change login information as needed
* The system shall allow customers to update payment methods

### 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 interface will need basic capabilities of modern internet-accessible machines, such as cell phones, tablets, laptops, and desktop PCs
* The interface will need an internet connection
* Each user will be able to update login information and profile information
* Each customer will be able to choose packages, make payments, view payment history, modify payment types for future use, and book appointments
* Each trainer will be able to book appointments and create/review notes
* Each IT user will be able to modify users
* Each admin will receive reports
* Each secretary will be able to modify the calendar of appointments

### 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?*

* Users have consistent, stable internet access
* Server has consistent, stable, high-speed internet access
* Users are using hardware that is relatively modern and capable of loading the website
* Users are using relatively modern web browsers
* Infrastructure is intact to provide electricity to power server and users
* Customers and trainers are local (e.g. applicable to state, county, and city laws)

### 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?*

* The system will not function without electricity
* System access requires internet connection to access
* The timeline is considerably short, which may affect implementation of certain features
* The system can only serve local customers and can only employ local trainers

### Gantt Chart

*Please include a screenshot of the GANTT chart that you created with Lucidchart. Be sure to check that it meets the plan described by the characters in the interview.*

Chart, timeline

Description automatically generated

**References**

TwitterGitHubHomepage, P. W., Walton, P., & TwitterGitHubHomepage. (n.d.). *Web vitals*. web.dev. Retrieved December 16, 2022, from <https://web.dev/vitals/>

(WAI), W. C. W. A. I. (n.d.). *Introduction to web accessibility*. Web Accessibility Initiative (WAI). Retrieved December 16, 2022, from https://www.w3.org/WAI/fundamentals/accessibility-intro/