

## CS 255 Business Requirements Document Template

Complete this template by replacing the bracketed text with the relevant information.

This template lays out all the different sections that you need to complete for Project One. Each section has guiding questions to prompt your thinking. These questions are meant to guide your initial responses to each area. You are encouraged to go beyond these questions using what you have learned in your readings. You will need to continually reference the interview transcript as you work to make sure that you are addressing your client's needs. There is no required length for the final document. Instead, the goal is to complete each section based on your client's needs.

**Tip:** You should respond in a bulleted list for each section. This will make your thoughts easier to reference when you move into the design phase for Project Two. One starter bullet has been provided for you in each section, but you will need to add more.

### 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?*

- **Client:**
  - Driver Pass, driving school company
  - Liam the Product Owner
- **Purpose:**
  - Taking advantage of the lack of student driver training (fills void in market)
  - Help young drivers with their DMV driving and paper test
  - Monetize in person driving lessons
- **What to do:**
  - Create a website-based system
  - System should be able to host a variety of users
  - The system should allow customers to schedule driving lessons, take practice exams, and see their progress throughout everything.

#### 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?*

- **Problem:**
  - Many people are failing their tests at the DMV
  - They need better training
  - Lack of training
- **Solution:**
  - Driver Pass website that allows students to
    - Take practice exams
    - Buy packages for in person driving lessons
    - Schedule and manage a schedule of in person lessons
    - See their progress on exams and lessons
- **User Aspects:**

- Users register over the phone; users can log in to their account and reset their password
- **Admin Aspects:**
  - Access to all accounts in the case of deletion, resetting password, and viewing data
- **Security Aspects:**
  - Cloud based website so they will do server security
  - Limited private data access for employees at the company

### 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?*

- **All Users:**
  - Register accounts
  - Login to their account
    - Be able to do a “forgot password” in the case of being locked out
    - Be able to alter their account information when logged in
    - Be able to delete their account if necessary
- **Customer Users:**
  - Pick from three different in person lesson paid packages
    - View a schedule of in person driving lesson appointments
      - View which dates, times, and instructors are available
      - Able to choose drop off and pick up locations
      - Create a new appointment
      - Cancel an appointment
      - Reschedule an appointment
  - Take online classes
  - Take in person lessons if their package allows for it
  - Take practice exams
    - Draft the practice exam if unfinished
    - Submit the practice exam if finished
  - View a data report on their practice exams
    - View grades
    - View progress on unfished practice exams
    - View percentage of all practice exam completion
- **Secretary User:**
  - Able to alter appointments for customers/instructors
    - If driver is sick, cancel their appointments with no penalty to customer
    - If customer needs to reschedule and calls secretary
      - Secretary can help them reschedule/cancel
  - Able to register users by taking their information over the phone
- **Owner User:**
  - Able to access and download reports and information offline
  - Tracking
    - Who created/cancelled/rescheduled an appointment
    - Print activity reports
  - Can disable and enable new/old packages

- **IT user:**
  - Has full access over all accounts to
    - reset passwords
    - block access from anyone that is fired
- **System Design:**
  - Updates tests and classes to be in accordance with the rules of DMV
  - Run on the cloud to avoid dealing with cybersecurity and backup issues
  - UI
    - Main Page
      - Shows test progress and allows customers to take/continue/retake tests
      - Driver Notes show what the customer can work on, written by instructor, also show upcoming lessons
      - User information appears
      - Contact information for the company appears

## 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?*

- Needs to run in a web-based environment
- System speed
  - System response time should not be more than 1.5 seconds
- System updates
  - The system should be updated in accordance with DMV rules and regulations in the area
  - System should handle any updates from operating systems
  - System should have regular maintenance every 3 months, should be down at this time for no more than two hours
    - If down longer notify admin

### 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?*

- Platforms
  - Should be able to run on Windows, Unix, IOS, and mobile phones as long as they are accessing it through a web browser
    - Should run on any web browser such as Safari, Google Chrome, Firefox, Microsoft edge, opera, etc.
- Backend
  - Database

- User account information should be stored
- Individual student examination progress should be stored
- Scheduling/appointments should be stored
- Admin reports should be stored

### 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?*

- Login Accuracy
  - Takes in username (email) and password
    - Passwords only are case sensitive
    - Retrieves encrypted information from the database to find out what account the information relates to if any
- Alert Notifications
  - If a user has typed their password wrong more than ten times and got locked, administrators will receive an email about this
    - Warnings for brute force
  - When a user resets their password, administrators will receive an email about this
  - Receive email when system is down or experiencing error
  - Receive email if user is reporting a bug
  - Receive email if system maintenance is going overtime

### 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?*

- User changes
  - User account changes should be done through front end UI and then the data is changed in the back end.
  - Code should not have to be changed.
- Platform Updates
  - Dealt with internally in the system.
  - If an error happens will notify the administrator
- IT Access:
  - Has admin privileges (different from owner)
    - Accessing user account data
    - Account/access deleted when fired/quit

### 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?*

- Logging in

- Forgot Password
  - Sends code to user's email/phone number to allow them to reset their password
- Locking
  - After 10 wrong attempts, lock the user out of signing in for ten minutes to prevent brute forcing. Every 10 wrong attempts, the wait time is doubled.
- Encryption
  - Passwords should be encrypted in the database
  - Saved credit info (user decides if they want it saved) should be encrypted as well in the database
- Employee Access
  - Administrative privileges
    - Only owner + IT employee
    - Should be taken away if someone is fired/let go
    - See user data and reset passwords
  - Schedule privileges
    - Given to admins + secretary + drivers
    - Only allowed to alter the schedule
  - Secretary Privileges
    - Also allowed to register a new user over the phone
- Cloud server
  - Server security in the hands of the server company

### 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 generate reports on user data for the owner to view
- The system shall allow users to schedule/reschedule/cancel driving appointments
- The system shall allow users to buy a driving lesson package
- The system shall allow users to take practice exams, then draft or submit them
- The system shall allow the user to view and modify their account information
- The system shall validate user credentials when they log in
- The system shall allow users to create an account

### 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.)?*

- **User Interaction**
  - Users shall interact through a web browser. The web browser should be scaled to work on the resolution of a mobile phone browser as well.

- **Login Page**
  - Input boxes for username/password.
  - Link to change password/username
- **Sign Up Page**
  - Input boxes for user information
  - Sends data to Driver Pass database when registered
- **Customers UI**
  - **Main Page**
    - Examination progress in top left box
      - Shows grades, finished practice exams, in progress exams, and not yet attempted exams
    - User account settings in top right
    - Driving notes section underneath exam progress
      - User can view their instructors notes for them after driving lessons
      - User can view the schedule
    - Student photo and driver photo in bottom right corner
    - Special needs and requirements above these photos
      - Any accessibility notes for the student are placed here so that the student can view and edit them to make sure they're correct.
  - **Exam Page**
    - Should have as many questions as the exam needs
      - Allows the user to cycle through the questions using previous and next buttons
    - Save progress button to draft exam
    - Exit button to leave without saving (prompts an “are you sure” message)
    - Submit exam button (prompts an “are you sure” message)
  - **Payment Plan Page**
    - Allows users to choose from different packages for their driving lessons or in person classes.
- **Admin UI**
  - Able to see user information
  - Able to see the schedule of all students and driver’s lessons.
    - See who updated what
    - Add or modify appointments
    - Print reports
  - Able to see reports analytics on student data
- **Driver + Secretary UI**
  - Able to see the schedule
  - Able to modify the schedule
    - Add or modify 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?*

- Assumed that the driving student has a computer or a mobile phone that has a web browser

- Assumed the student lives close enough to Driver Pass to be picked up for driving lessons
- Assumed the student has enough money to pay for a package
- Assumed that the system will update in accordance with DMV rule updates
- Assumed that website will be up and running except for during the regular maintenance

### 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?*

- Driver Pass only has 10 cars for driving lessons
- There is no mobile app for Driver Pass, only accessible through web browsers
- No assigned budget, makes things uncertain
- The system needs internet connection to run
- 15 ½ weeks (or 3 and a half months) to complete the project
  - Unknown how big the development team is going to be
- Relying on a company (cloud) to source the server

### 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.*

