# CS 255 Business Requirements Document

Noah Stubbs

## BLUF

Bottom Line Up Front

DriverPass requires a web-based platform to support student driving preparation through online testing and scheduling of in-person lessons. The system must be secure, scalable, user-friendly, and maintainable by non-technical staff. This document outlines the full business and system requirements using both standard and alternate formats to support diverse audiences and internal team roles.

## System Components and Design

### *Purpose*

The primary purpose of this project is to design and implement a digital platform for DriverPass, a startup that seeks to improve the pass rates of students taking the DMV driving test. The business opportunity lies in addressing the lack of integrated solutions for driver training, especially platforms that combine both digital practice and in-person instruction.

DriverPass identified a gap in the market: many students rely solely on memorization of previous DMV tests, leading to high failure rates. By offering a system that includes online learning modules, practice exams, real-world lesson scheduling, and administrative oversight; the company aims to better prepare students for the test and provide greater convenience and flexibility in how they train.

The client has requested a system that is web-based and accessible across all modern devices and browsers. It must support both customer-facing and internal-facing functionality, including user registration, lesson tracking, content management, and system-level reporting. In essence, this system should function as the backbone of the company’s operations, digitally enabling all services they offer to customers while reducing administrative workload for staff.

### *System Background*

The DriverPass system is being developed in response to a critical need identified by the client: a lack of comprehensive training tools for individuals preparing for their DMV driving exams. According to DriverPass’s founder, over 65% of students fail the test due to over-reliance on rote memorization through multiple test attempts. The proposed system aims to solve this problem by delivering a hybrid solution that streamlines the learning and scheduling process; online and in-person.

Functionally, the system will provide access to DMV-aligned practice tests, interactive lessons, and tools for reserving real-world driving instruction. The scheduling module must be flexible enough to accommodate appointments made online, by phone, or in person at the DriverPass office. Each reservation must be associated with a student, an instructor, a vehicle, and a specific time window.

From a technical standpoint, the system is expected to operate entirely online, using standard browser technologies. It should be accessible across multiple platforms and screen sizes, requiring no native app or software download. Backend services, such as hosting, backups, and data security, will be outsourced to a third-party vendor to ensure uptime, scalability, and regulatory compliance without increasing the client's IT burden.

The system must also include a notification engine to alert administrative staff when DMV regulations, questions, or policies are updated, either via manual input or future API integration. Ultimately, the DriverPass system will act as both a student-facing platform for test readiness and a business management tool for coordinating personnel and services.

### *Objectives and Goals*

* The overarching objective of the DriverPass system is to increase student success rates on DMV driving tests by delivering a high-quality blended learning platform. This includes not only offering online classes and practice tests aligned with official DMV materials but also managing the logistical components of on-the-road training, such as instructor assignments and vehicle tracking.
* Key goals include the ability to create user accounts through both phone-based and online registrations, with full profile fields including name, address, phone number, credit card information, and lesson package selection. Once registered, students must be able to view their upcoming lessons, track their test progress, and access automated feedback. The system should also include robust password management functionality, allowing users to reset credentials without manual staff intervention.
* Administrators and secretaries must be able to manage lesson schedules, edit or deactivate lesson packages, and run reports on student activity, system changes, and test results. Every system change, whether a schedule edit or package adjustment, must be logged and traceable to the specific user and timestamped for auditing purposes.
* Measurable success will include full coverage of DMV curriculum content, a reduction in scheduling conflicts, improved clarity in driving appointment records, and ease of use across all user roles. The system must also support exportable reports and align with security standards for user and payment data.

# Nonfunctional Requirements

## *Performance Requirements*

The DriverPass system is expected to deliver consistent, low-latency access to its core features — practice exams, scheduling modules, user account management, and reporting. This section outlines the performance-related expectations critical to user satisfaction, scalability, and overall system success.

Runtime Environment:

The system shall operate as a web-based application accessible from all modern desktop and mobile browsers. While native mobile app development may be a future goal, the MVP (minimum viable product) must function seamlessly in OperaGX, Chrome, Firefox, Safari, and Edge, across Windows, Linux, macOS, Android, and iOS environments.

Responsiveness & Latency:

“A site that loads in 1 second has a conversion rate 3x higher than a site that loads in 5 seconds” (Wiegand, 2022).

* Page loads (including dashboard, scheduling, and test modules) shall not exceed 2 seconds under typical broadband conditions (<50 concurrent users).
* Interactive elements such as sliders, buttons, and test submissions must reflect user action within 300 milliseconds.
* Any file upload, report generation, or database search functionality shall complete within 5 seconds, barring edge-case scenarios.

Concurrency & Load Expectations:

“Rather than fixating on the number of concurrent users, testers should focus on creating realistic user scenarios” (Loadview, 2024).

As an initial estimation, the system is planned to support:

* 50 concurrent student users taking practice tests
* 10 concurrent instructors updating schedules
* 5 admin users performing system oversight tasks

This baseline supports a small-to-medium operation. Horizontal scaling via the chosen cloud infrastructure must enable future growth to 250+ active users without major code changes.

System Uptime & Availability:

“Moving from 99% to 99.9% often yields significant performance gains in customer satisfaction and reliability. These are also the most cost-effective since they can be achieved by implementing standard practices in infrastructure redundancy, incident management, and monitoring.” (Ghadge, 2024).

* The system must maintain a minimum uptime of 99.0% (approximately 87.6 hours of downtime annually) for Version 1.0.
* A long-term stretch goal is five-nines (99.999%) availability (~5 minutes of annual downtime), contingent on future scaling, budget, and infrastructure maturity.
* Scheduled maintenance windows must be announced via in-system notifications at least 48 hours in advance.
* Failover support (multi-zone redundancy or fallback servers) must be supported by the chosen third-party hosting provider.

System Updates:

* Minor feature releases and patches should be deployed bi-weekly using rolling updates, minimizing downtime.
* Major updates (i.e., structural or interface redesigns) should be scheduled quarterly and must pass regression testing prior to deployment.
* Hotfix capability must exist to address critical bugs within 24 hours of discovery or report.

Monitoring and Reporting:

“With monitoring, teams can collect and analyze data to detect issues. It can also be used to create trends to predict upcoming problems. Some of the monitoring features include data collection, aggregation, and visualization” (Edge Delta, 2024).

* Performance logs shall be recorded at the server level and aggregated for monthly performance reviews.
* If latency exceeds target thresholds, the system should flag a warning for IT review and generate an internal report.

Assumptions:

These performance metrics assume operation under a cloud-hosted architecture with Content Delivery Network (CDN) support for static assets, and an underlying database that uses indexed queries for high-frequency operations. Performance will be monitored and revised iteratively based on usage analytics post-launch.

“A content delivery network (CDN) is a geographically dispersed server network that enables faster web performance by locating copies of web content closer to end users or facilitating the delivery of dynamic content” (Susnjara & Smalley, 2024).

## *Platform Constraints*

To define the technical platforms on which the DriverPass system must operate, along with any backend tool dependencies required for deployment, maintenance, and scalability.

Supported Platforms:

The system must function as a web-based application with front-end support for major browsers: OperaGX, Chrome, Firefox, Safari, and Edge. It shall be responsive and functional on desktop (Windows, macOS, Linux) and mobile platforms (iOS and Android).

Hosting Requirements:

DriverPass will utilize a third-party cloud service for hosting (e.g., AWS, Azure, or Google Cloud) to avoid internal server maintenance. The provider must support CDN, role-based access control, encryption at rest, and automatic daily backups.

Backend Requirements:

A relational database (PostgreSQL or MySQL) will support data persistence for user accounts, test records, schedules, and transaction logs. Backend must be developed using a scalable framework (e.g., Node.js, Python Django, or ASP.NET). Database must support indexed fields and foreign key constraints to ensure data integrity.

Tooling Constraints:

The system must be compatible with browser-based administrative tools (e.g., for report exports, account management). Server must support RESTful APIs for future expansion (mobile app integration or DMV data sync).

“REST API stands for REpresentational State Transfer API. It is a type of API (Application Programming Interface) that allows communication between different systems over the internet” (Geeks for Geeks, 2025).

Assumptions:

The client prefers a minimal in-house technical burden, relying on cloud infrastructure to handle updates, patches, and uptime guarantees.

## *Accuracy and Precision*

To define how the system ensures correctness of user input, role recognition, and audit reporting for accountability and traceability.

User Identification & Roles:

* All users must authenticate with a unique identifier (email or username) and password.
* User roles (admin, IT officer, secretary, student) must be distinctly separated using Role-Based Access Control (RBAC).
* Activity logs will timestamp every critical action (e.g., schedule change, test submission, account update) with user ID and action metadata.

“The key to data and network protection is access control, the managing of permissions and access to sensitive data, system components, cloud services, web applications, and other accounts. Role-based access control (RBAC), or role-based security, is an industry-leading solution with multiple benefits (Fortinet, n.d.).

Input Validation:

* Inputs must support both case sensitivity and data formatting enforcement (e.g., phone: XXX-XXX-XXXX, ZIP: 5 digits).
* Required fields (e.g., name, address, payment info) will prevent submission unless valid.

Admin Alerts:

* If test scores, account data, or transaction logs experience anomalies (e.g., score mismatches, duplicate records), the system shall auto-alert IT.
* The system shall notify IT if more than three failed login attempts occur within 10 minutes.

Assumptions:

System will be configured to comply with local regulations for data formatting and privacy based on the state of operation.

## *Adaptability*

To define the system’s ability to adapt to change without codebase overhaul, including account configuration, content management, and platform evolution.

User Management:

* Admins shall have full rights to add, remove, or modify user accounts through an interface, not through code changes.
* Role-based access settings must be configurable from the UI.

Content Flexibility:

* Test content and DMV rules must be modular and editable via admin dashboard or synced via external API.
* Practice exam questions must support importing/exporting in CSV or JSON format.

System Upgrades:

* Application must support over-the-air updates from the hosting provider, including OS patches, framework upgrades, and database migrations.
* All UI elements must be responsive to major browser or OS updates (mobile-first design principles encouraged).

Assumptions:

* IT staff will maintain basic system configuration without direct developer support.
* Platform will evolve quarterly with infrastructure patches.

## *Security*

To define the safeguards protecting user data, preventing unauthorized access, and enabling recovery from malicious activity.

Authentication & Access Control:

According to NIST SP 800-63B, secure password-based authentication requires a minimum of 8 characters and support for secure reset flows with identity proofing (Grassi et al., 2017).

* All users shall log in with a secure password (minimum 8 characters, must include upper/lower case, numeric, and symbol).
* Role-based access will define the scope of access for each user group (admin, IT, secretary, student).

Data Protection:

* All data exchanges between the client and server must occur over HTTPS with TLS 1.2 or higher.
* Passwords must be hashed using industry-standard algorithms (e.g., bcrypt or SHA-256 with salt).
* Credit card data must be tokenized or handled by PCI-DSS-compliant third-party processors.

All data exchanges must occur over HTTPS with TLS 1.2 or higher to ensure encrypted communication (National Institute of Standards and Technology, 2019; Naziridis, 2022).

“Tokenization is used for securing sensitive data, such as a credit card number, by exchanging it for non-sensitive data - a token” (Riesen, n.d.).

Security Measures:

* System shall detect brute-force attempts, temporarily lock accounts, and notify IT after 3 failed attempts in 10 minutes.
* Password recovery will require access to the registered email address with a secure reset token that is only valid for 15 minutes.
* Admin panel access will include multi-factor authentication.

Assumptions:

Client will not store raw credit card data internally and will rely on a vetted third-party payment processor (e.g., Stripe, Authorize.net).

# Functional Requirements

## *Overview*

To define the specific behaviors and system actions the DriverPass platform must be capable of performing.

Student-Facing Functions:

* The system shall allow students to create accounts and securely log in.
* The system shall allow students to schedule, cancel, or reschedule driving lessons online.
* The system shall provide students with access to online practice tests, track their completion, and store scores.
* The system shall allow students to view scheduled appointments and assigned instructors.
* The system shall enable students to reset passwords without employee assistance.

Secretary/Admin Functions:

* The system shall allow staff to create new student profiles during phone registration.
* The system shall allow staff to assign lessons to instructors, update schedules, and reassign cars.
* The system shall allow administrators to generate activity logs and printable reports.
* The system shall allow staff to view, add, edit, or disable available training packages.

Instructor/IT Officer Functions:

* The system shall track changes and display the user and timestamp for every schedule or account modification.
* The system shall allow instructors to log completed sessions and add notes about the student.
* The system shall allow the IT officer to view all user accounts and reset login credentials.

Admin Panel & Reporting:

* The system shall provide exportable reports in CSV or PDF format.
* The system shall allow for manual sync with DMV policy updates and alert the administrator when new content is available.

## *User Interface*

To define how different users will interact with the system and what features and views must be accessible to each role.

Design Requirements:

* Interface shall follow responsive web design standards and adapt across devices (desktop, tablet, mobile).
* UI shall include clearly defined dashboards for each role: students, instructors, secretary/admin, and IT officers.
* The navigation bar shall be persistent and context-aware, showing only functions relevant to the current user role.

Student Interface:

* Dashboard must display scheduled lessons, test progress, and ability to cancel or reschedule appointments.
* Test modules must support timers, automatic submission, and detailed feedback.
* Students must be able to update their contact info, password, and view driver notes.

Secretary/Admin Interface:

* Secretary dashboard shall allow for student lookup, account creation, and appointment booking.
* Admin dashboard shall include access to reporting, user management, and package configuration.
* Input forms shall include validation for required fields (e.g., name, credit card, pickup location).

Instructor Interface:

* Instructors must be able to view schedules and submit lesson reports.

IT Interface:

* IT must have a dedicated interface for account resets, error logging, and security alerts.

Accessibility Considerations:

* Interface shall meet WCAG 2.1 AA accessibility standards, including screen reader support and keyboard navigation.

“Web Content Accessibility Guidelines (WCAG) 2.1 defines how to make web content more accessible to people with disabilities” (World Wide Web Consortium [W3C], 2025).

## *Assumptions*

To clarify any elements of the system design that rely on conditions not explicitly stated in the project scope or requirements.

* All users will have access to an internet connection and a modern web browser.
* Student users will be responsible for keeping their contact information up-to-date.
* The DMV will provide updated test materials and policy changes via a publicly accessible API or data source.
* Cloud hosting will provide automatic scalability and 24/7 uptime unless otherwise stated.
* All backend systems and integrations will follow current web security standards.
* DriverPass employees will be trained in using the system’s dashboard and features during rollout.
* Secretaries or office staff will be responsible for entering and verifying customer information obtained via phone.

## *Limitations*

To outline known constraints related to technology, timeline, staffing, budget, or project scope.

* The system will not initially include native mobile apps, only a web-based platform is guaranteed.
* Package management (adding/removing/modifying core packages) will be limited to enabling/disabling through the UI; module-level customization will require developer assistance.
* Credit card processing is assumed to be offloaded to a third-party provider; direct PCI compliance is out of scope.
* The system will not initially support biometric login or advanced authentication (e.g., OAuth) unless required later.
* Integration with DMV for real-time updates is assumed to be manual or semi-automated unless an official API is provided.
* The internal development team is assumed to be limited in size, impacting rollout of advanced analytics or AI-enhanced features.
* No offline functionality will be provided in this release version; all operations require a live internet connection.

# Additional Resources

## *Contacts*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Name** | **Role** | **Organization** | **Email** | **Phone** | **Responsibility** |
| Liam | Owner | DriverPass | big.boss@DriverPass.com | 999-  867-5309 | Client vision, business priorities |
| Ian | IT Officer | DriverPass | ian@DriverPass.com | 998-  867-5309 | Infrastructure, technical decisions |
| Jennifer | Project Lead | A Small Consulting Company | Jennifer@ASCC.com | 001-  867-5309 | Final approvals, timeline, client liaison |
| Sam | Systems Analyst | A Small Consulting Company | Sam@ASCC.com | 008-  867-5309 | Legacy system notes, technical history |
| Noah | Systems Analyst | A Small Consulting Company | Noah@ASCC.com | 074-  867-5309 | Current system design and documentation |

## *Visual Aids*

[Gantt Chart](Gantt%20chart%201.1.pdf)

[UML Class Diagram](UML%20class.pdf)

[Activity Diagram](Activity%20diagram.pdf)

## *Five Paragraph Order*

For our Veterans

Situation

DriverPass identified a high failure rate among students taking the DMV driving test, largely due to poor preparation and lack of integrated practice systems. The company seeks to establish a web-based system to provide online classes, practice tests, and scheduling for on-the-road driving lessons. The system must support multiple user roles and maintain secure data handling across all components.

Mission

Develop and deploy a cloud-hosted, browser-accessible platform that allows students to register, schedule driving lessons, take practice exams, and track their progress, while enabling staff and administrators to manage lessons, instructors, packages, and reporting functions.

Execution

The platform will include separate dashboards for students, secretaries, instructors, and administrators. Functional components include account creation, test delivery, schedule management, notification systems, and administrative reporting. Nonfunctional requirements include performance, adaptability, platform constraints, and system security.

Administration and Logistics

All hosting, backups, and security services will be managed by a third-party cloud provider. The platform must support password resets, automated alerts, and user tracking. Data storage will use a relational database, with reports exportable in industry-standard formats.

Command and Signal

Students interact via a personal dashboard. Secretaries handle registrations and appointment logistics. Instructors manage availability and submit reports. IT officers handle credential management and troubleshooting. Admins oversee all components. Communication is system-driven via notifications and logs.

## *Role-Based Index View*

|  |  |
| --- | --- |
| Role | Responsibilities and Permissions |
| Student | * Create account, schedule/cancel lessons, access practice tests. * View test progress, driver comments, and lesson schedule. * Reset password and update personal information. |
| Secretary | * Register students by phone, manage appointments, assign drivers/cars. * Update schedules and process changes to student profiles. |
| Instructor | * View scheduled appointments, input session feedback. * Mark lessons completed and submit progress notes. |
| Admin | * Manage user roles, adjust/remove packages, generate reports. * Monitor system activity logs, respond to DMV updates. |
| IT Officer | * Reset accounts, manage login credentials, oversee platform access. * Receive brute force alerts, respond to technical errors. |

“To implement the RBAC, we designed a Role-based interface Model to present displays and provide functionality appropriate for different role responsibilities” (Lin & Speedie, 2003)

## *System Behavior Quick Reference*

|  |  |
| --- | --- |
| **IF…** | **THEN…** |
| A student creates an account | The system stores personal info and assigns login credentials |
| A student selects a package | The system activates related test and scheduling options |
| A student schedules a lesson | The system assigns a car and instructor, and adds it to both calendars |
| A lesson is canceled | The system frees up the time slot and logs who canceled it |
| A password reset is requested | A secure reset token is emailed and expires in 15 minutes |
| DMV updates policy/test materials | Admins receive a system alert to review and update content |
| An instructor logs a completed lesson | The student's record is updated, and feedback is saved |
| Three failed login attempts in 10 minutes | The system temporarily locks the account and notifies IT |

“With this resource, readers can get answers quickly, so they don’t need to reference lengthy software documentation or user guides” (Scribe’s Team, 2024).

## *Regulatory Cross-Reference Table*

|  |  |  |
| --- | --- | --- |
| **System Feature** | **Relevant Regulation/Standard** | **Compliance Consideration** |
| Payment Information Storage | PCI-DSS | Encrypt credit card data in transit and at rest. Limit access by role. |
| Student Personal Information | State Privacy Laws (e.g., CCPA, HIPAA-adjacent), FERPA (by analogy) | Protect names, addresses, phone numbers. Implement role-based access and secure storage. |
| Password Reset Function | NIST Cybersecurity Framework | Enforce secure reset protocols, prevent social engineering, and track reset events. |
| User Activity Logging | NIST / ISO 27001 | Maintain immutable logs. Include timestamps, action types, and user identity. |
| DMV Policy Updates | Local Department of Motor Vehicles | Ensure content aligns with updated requirements. Notify admin automatically on changes. |
| Online Class & Test Material | ADA (Accessibility), DMV Curriculum | Ensure screen reader compatibility, alt text, and compliance with official DMV content. |
| Appointment Scheduling & Driver Assignments | Business Continuity and Operational Policies | Prevent scheduling conflicts; maintain backup of appointment data in case of system failure. |

“A cross-reference table is a tool used in documents or databases to link related information together. It provides a way to connect different sections, topics, or entries within a document or system, allowing users to easily find and reference related content” (Cobrief, 2025).

## *Plain-Language Executive Summary*

DriverPass wants to help people pass their driving test by giving them better tools. Many people fail the test because they only try to memorize the answers. This new system will let students take practice tests online and sign up for real driving lessons. It works on phones and computers.

Students will be able to see when their lessons are, who their instructor is, and how they’re doing on their practice tests. Staff will be able to help students, schedule lessons, and make changes when needed. The system keeps track of everything, so everyone knows who made what changes.

Everything is stored online and backed up automatically. The system will be safe and easy to use. It will let DriverPass grow and help more students in the future.

“Plain language focuses on the needs of the audience. It is accessible, ethical, trustworthy, and practical” (Geddy et al., 2022).

## Works Cited

*Breaking the Misconception with Concurrent Users in Load Testing*. (2024, April 5). Loadview. <https://www.loadview-testing.com/blog/breaking-the-misconception-with-concurrent-users-in-load-testing/>

Cobrief. (2025, April 8). *Cross-reference table: Overview, definition, and example*. Cobrief. <https://www.cobrief.app/resources/legal-glossary/cross-reference-table-overview-definition-and-example/>

Edge Delta Team. (2024, August 22). *Monitoring and Alerting: Best Practices*. Edge Delta. <https://edgedelta.com/company/blog/monitoring-and-alerting-best-practices>

Fortinet. (n.d.). *Role-Based Access Control (RBAC): Enabling Least Privilege at Scale*. Foortinet. <https://www.fortinet.com/resources/cyberglossary/role-based-access-control>

Geeks For Geeks. (2025, April 8). *REST API Introduction*. Geeks for Geeks. <https://www.geeksforgeeks.org/node-js/rest-api-introduction/>

Geddy, M., Risvold, W., & Stratton, M. (2022, July 12). *Plain Language Is Best for Business Communication*. Association to Advance Collegiate Schools of Business. <https://www.aacsb.edu/insights/articles/2022/07/plain-language-is-best-for-business-communication>

Ghadge, A. M. (2024, August 24). *The Hidden Costs of Chasing Five 9s*. The New Stack. <https://thenewstack.io/the-hidden-costs-of-chasing-five-nines-in-availability/>

Grassi, P. A., Fenton, J. L., Newton, E. M., Perlner, R. A., Regenscheid, A. R., Burr, W. E., & Richer, J. P. (2017). *Digital identity guidelines: Authentication and lifecycle management (Special Publication 800-63B)*. National Institute of Standards and Technology. <https://doi.org/10.6028/NIST.SP.800-63b>

Lin, Y. J., & Speedie, S. (2003). Role-based and Adaptive user interface designs in a Teledermatology consult system: a way to secure and a way to enhance. *AMIA ... Annual Symposium proceedings. AMIA Symposium*, *2003*, 913.

National Institute of Standards and Technology. (2019). Guidelines for the selection, configuration, and use of transport layer security (TLS) (SP 800‑52 Rev. 2). https://csrc.nist.gov/publications/detail/sp/800-52/rev-2/final

Naziridis, N. (2021, February 22). *Guide to TLS Standards Compliance*. SSL. <https://www.ssl.com/guide/tls-standards-compliance/>

Riesen, M. (n.d.). *What is tokenization and how can I use it for PCI DSS compliance?* SecurityMetrics. <https://www.securitymetrics.com/blog/what-tokenization-and-how-can-i-use-it-pci-dss-compliance>

Scribe’s Team. (2024, December 10). *Creating Quick Reference Guides: Tools, Tips and a Template*. Scribe. <https://scribehow.com/library/what-is-a-quick-reference-guide#whats-a-quick-reference-guide>

Susnjara, S., & Smalley, I. (2024, July 23). *What is a content delivery network (CDN)?*. IBM. <https://www.ibm.com/think/topics/content-delivery-networks>

Wiegand, M. (2022, April 20). *Site Speed is (Still) Impacting Your Conversion Rate*. Portent. <https://portent.com/blog/analytics/research-site-speed-hurting-everyones-revenue.htm>

World Wide Web Consortium. (2025, May 6). Web content accessibility guidelines (WCAG) 2.1. <https://www.w3.org/TR/2025/REC-WCAG21-20250506/>