

# Project Definition and Planning: Workshop No. 1

*DriveMaster: Subscription-based Management System for Colombian Driving Academies*

**David Gerardo Diaz Gomez** (20201020087)  
**Cristian Arturo Parra Gonzales** (20201578102)  
**Daniel Mateo Montoya González** (20202020098)

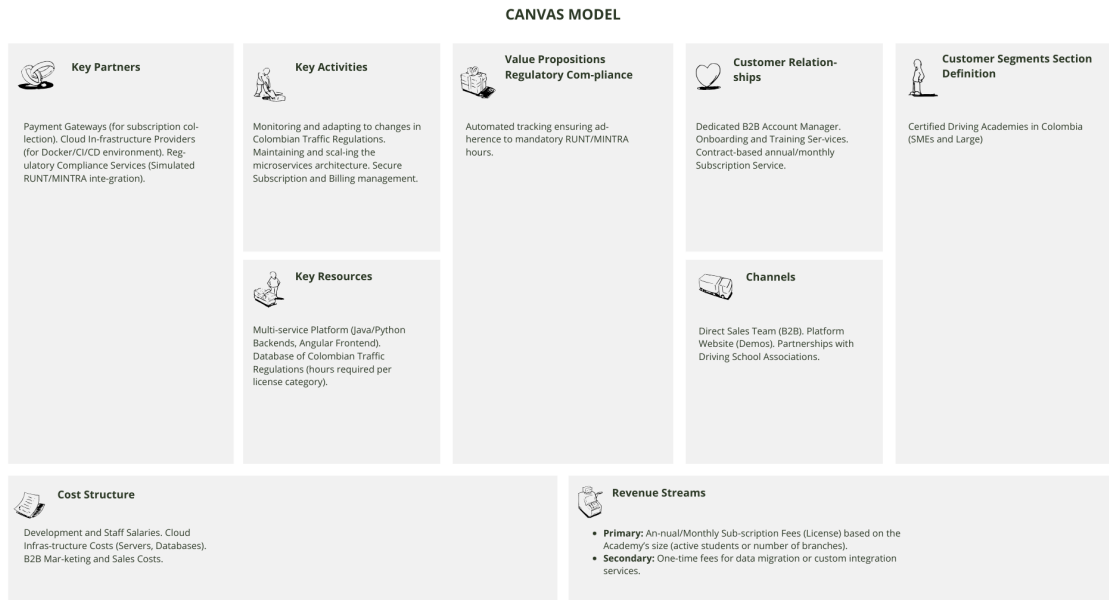
**Software Engineering Seminar**  
Semester 2025-III  
Eng. Carlos Andrés Sierra, M.Sc.  
Universidad Distrital Francisco José de Caldas

## Introduction

This document presents the foundational artifacts for the "DriveMaster" project, a Subscription-as-a-Service (SaaS) platform designed to manage and ensure regulatory compliance for Colombian driving schools offering licenses A1, B1, C1, A2, B2, C2, A3, B3, and C3. This work fulfills the requirements of Workshop No. 1, providing a solid definition of the business model, user requirements, and initial object-oriented design for the subsequent implementation phases. The project structure is specifically designed to facilitate robust unit testing (JUnit, pytest), acceptance testing (Cucumber), and performance testing (JMeter).

## 1 Business Model Canvas

The Business Model Canvas (BMC) for DriveMaster is defined below, focusing on the B2B SaaS licensing model and regulatory adherence.



## 2 User Stories and Acceptance Criteria

The following User Stories are defined with specific acceptance criteria in the Gherkin format (**Given, When, Then**) to facilitate future validation using **Apache Cucumber**.

### Role 1: Academy Administrative Staff (Secretary)

Given the Secretary is logged in and navigates to the registration form, When they enter student data and select 'C1' and click 'Save', Then the student's record is created, and their required hours are set to 25 (Theory) and 20 (Practice) as per the C1 norm.

Given an active student (ID 123) is searched in the system, When the Secretary opens their progress dashboard, Then the system displays the total required hours (e.g., 40), hours completed (e.g., 25), and hours pending (e.g., 15).

Given I am in the Instructor Management module, When I register a new instructor, select their license type, and confirm their schedule, Then the instructor appears as 'Available' for scheduling classes that match their license and schedule.

Given the student has 100% of the mandatory hours completed and all fees paid, When I click 'Generate Final Certificate', Then a PDF is downloaded using the required legal (RUNT/MINTRA simulated) format, and the student's status changes to 'Certified'.

Given I select an active student's profile, When I register a payment of \$X for 'Practical Hours Package', Then the payment is logged in their financial history, and the system sends a payment confirmation receipt to the student's email.

## **Role 2: Driving Instructor**

Given I navigate to the platform's login page, When I enter my credentials and the Java Auth Service verifies them, Then I am redirected to my personalized schedule dashboard.

Given I am viewing a scheduled class, When I enter the final duration (e.g., 1 hour) and confirm 'Class Complete', Then the student's profile is credited with 1 practical hour, and the remaining pending hours are reduced by one (Python Backend Logic).

Given a scheduled class has passed its start time, When I select the student and choose the reason 'Absent (No Show)', Then the class status is marked 'Canceled', and the student receives an email notification.

Given I select my upcoming session from the schedule, When the session details load, Then the system displays the vehicle's plate number and the vehicle type (e.g., Heavy Truck for C3).

Given I am viewing a Vehicle profile, When I report a fault (e.g., 'Brake Noise') and click 'Report Issue', Then the vehicle's status changes to 'Maintenance Required' and it is temporarily removed from the available scheduling pool.

## **Role 3: Student**

Given I am logged into my profile dashboard, When I navigate to 'My Progress', Then the system displays two progress bars: one for theory (X/Y hours) and one for practice (A/B hours).

Given I am on the 'Schedule Class' page, When I filter by date and time, Then the system only displays slots where an instructor with the correct license type and a free vehicle are available.

Given my status is 'Certified' in the system, When I click the 'Download Certificate' link, Then the system serves the RUNT-formatted PDF document for immediate download.

Given I click the 'Forgot Password' link on the login page, When I enter my registered email and submit the form, Then the Java Auth Service sends a secure, time-limited link to that email address.

Given I have a class scheduled for tomorrow at 3:00 PM, When the system's scheduled notification job runs, Then I receive a notification (email/SMS) confirming the class time, instructor, and assigned vehicle.

#### **Role 4: DriveMaster System Administrator (SaaS Model)**

Given I search for 'Academy ABC', When I change their subscription end date from 2025-12-31 to 2026-12-31 and save, Then the SubscriptionLicense object is updated in the database, and the Academy's access is secured for the new period.

Given an Academy's payment is overdue, When I select the Academy and click 'Deactivate Subscription', Then the SubscriptionLicense status changes to 'Suspended', and all associated users are immediately logged out and blocked from future login attempts (Java Auth Service check).

Given I am in the Regulatory Update module, When I change the A2 practice hours from 15 to 20 and click 'Apply Global Update', Then the Python Business Logic updates the CourseRegulation data, and all new student registrations will inherit the 20-hour requirement.

Given the current month is complete, When I run the 'Monthly Usage Report', Then the system outputs a file detailing the peak number of active student profiles for each subscribed Academy in that month, ready for invoice generation.

Given I am in the Role Management module, When I define the new role 'Auditor' and grant them 'Read-Only' access to student data, Then the new role is created in the database, and can be assigned to new users by the Java Auth Service.

### **3 User Story Mapping**

The User Story Map is a visual planning tool that organizes user activities and associated stories into development iterations (Sprints).

Table 1: DriveMaster User Story Mapping: Planning and Prioritization

Backbone Theme)	(Product	Activity (User Action)	User Stories (MVP / Iteration)	Priority
<b>Authentication &amp; Access</b>		System Entry	Start Session Login (Instructor, Student). Password Reset.	<b>MVP / Critical</b>
<b>Regulatory &amp; Student Management</b>		Student Lifecycle	Student Registration. View Student Progress. Generate Certificate.	<b>MVP / High</b>
<b>Scheduling &amp; Logistics</b>		Class Management	Complete Class Report. Self-Schedule Practice (Student). Mark Absence.	<b>MVP / High</b>
<b>SaaS License Management</b>		Billing & Authorization	Manage Subscription End Date (Admin). Revoke License Access (Admin).	<b>MVP / Critical</b>
<b>Compliance &amp; Data</b>		Normative Updates	Update Global Regulation (Admin). Report Vehicle Issue (Instructor).	<b>Iteration 2 / Medium</b>
<b>Reporting</b>		Finance & Audit	Generate Billing Report (Admin). Register Student Payment (Secretary).	<b>Iteration 2 / Medium</b>

## 4 CRC Cards (Class-Responsibility-Collaborator)

CRC Cards are used to identify the core classes, their responsibilities, and how they collaborate.

Class: Student	
Responsibilities	Collaborators
<ul style="list-style-type: none"> <li>• Manage profile data (name, ID, category).</li> <li>• Store and update progress (completed hours).</li> <li>• Request Certificate upon 100% completion.</li> </ul>	<b>CourseRegulation</b> <b>Schedule</b> <b>PaymentService</b>

Class: CourseRegulation	
Responsibilities	Collaborators
Define mandatory hours per license type (A1, B2, C3). Validate if a Student meets certification criteria. Handle Global Update of regulatory hours (Admin).	<b>Student, SystemAdmin</b>

Class: SubscriptionLicense	
Responsibilities	Collaborators
Store Academy's license end date and status (Active/Inactive).	<b>Academy</b>
Authorize user login access (checked by Java Auth Service).	<b>User (for login)</b>
Calculate billing amount based on Student usage (Python logic).	<b>BillingService</b>

Class: Schedule	
Responsibilities	Collaborators
Book and reserve time slots (Instructor and Vehicle).	<b>Student, Instructor, Vehicle</b>
Verify availability before confirmation.	
Log class completion/absence for progress tracking.	

## 5 Delivery Format Note

All deliverables are compiled into this single PDF file. The source files, including the Angular frontend and the Java/Python backends, will be organized in a folder named **Workshop-1** in the course repository, with a **README.md** referencing each section as per the course requirements.