**🧭 SPRINT 1 — Módulo de Matrícula**

**🎯 Objetivo general**

Digitalizar el proceso completo de matrícula para evitar colas, papeleo y errores manuales.

**🧩 Entregables completados**

| **Funcionalidad** | **Estado** | **Descripción** |
| --- | --- | --- |
| **Registro de usuario / login** | ✅ | Sistema de autenticación con JWT y roles (USUARIO, ADMIN, ESTUDIANTE). |
| **Formulario de matrícula por pasos** | ✅ | Flujo completo con datos personales, académicos, pago y confirmación. |
| **Subida de comprobante** | ✅ | Gestión de archivos con multer (se guardan en /uploads/comprobantes). |
| **Registro temporal en tabla Matricula** | ✅ | Se almacena en BD con estado PENDIENTE. |
| **Panel del Administrador (Dashboard)** | ✅ | Vista con acceso restringido para ADMIN. |
| **Validación de matrículas** | ✅ | Admin puede aprobar o rechazar matrículas. |
| **Creación automática de Estudiante** | ✅ | Al aprobar, se genera el registro definitivo y se limpia Matricula. |
| **Protección de rutas** | ✅ | Middleware requireAuth y requireAdmin. |
| **Diseño e interfaz modularizada** | ✅ | Estructura organizada (Layout, componentes, rutas privadas). |

**🧾 Documentos del Sprint (si fuera presentación o entrega)**

Puedes documentarlo así para tu reporte:

1. **Nombre del Sprint:** “Gestión de Matrículas en Línea”.
2. **Duración estimada:** 2 semanas.
3. **Equipo / Rol:**
   * Aldo Quispe Locumber – Desarrollador Full Stack.
4. **Objetivo específico:**  
   Implementar un sistema web que permita el registro y validación digital de matrículas para la academia preuniversitaria.
5. **Historias de usuario principales:**
   * Como estudiante, quiero matricularme en línea para evitar colas.
   * Como administrador, quiero validar las matrículas fácilmente.
6. **Resultados:**  
   Se completó todo el flujo del proceso de matrícula con persistencia, roles y control de estado.

### 🧠 ****Próximo Sprint (Sprint 2): “Gestión de Pagos y Constancias”****

**Objetivo:**  
Permitir el control de cuotas, registro de pagos y generación automática de constancias en PDF.

**Entregables sugeridos:**

* Módulo de pagos y cuotas (por matrícula).
* Estado financiero del alumno.
* Generación de constancia PDF (solo para aprobadas).
* Panel con totales de ingresos por modalidad o grupo.

**📋 Sprint 1: Objetivo original**

Digitalizar y automatizar el proceso de matrícula para estudiantes de la academia.

**🚀 Funcionalidades que ya implementaste**

| **Funcionalidad** | **Estado** | **Descripción** |
| --- | --- | --- |
| 🧍‍♂️ Registro e inicio de sesión de usuario | ✅ | Permite que el usuario se registre y acceda con JWT. |
| 📋 Formulario de matrícula (datos personales, académicos, pago) | ✅ | Desarrollado en pasos con React. |
| 📤 Envío de comprobante de pago | ✅ | Implementado con multer (subida de archivo). |
| 💾 Registro en tabla Matricula | ✅ | Se guarda toda la información del formulario. |
| 🧠 Aprobación/Rechazo de matrícula por el admin | ✅ | Cambia el estado y crea registro en Estudiante. |
| 🔍 Panel admin: listado de matrículas pendientes | ✅ | Interfaz funcional con opciones de aprobar/rechazar. |
| 🔗 Relaciones Prisma (Usuario → Matricula → Estudiante) | ✅ | Correctamente definidas y funcionales. |

**🧱 Lo que podrías agregar para cerrar 100% el Sprint 1 (opcional antes del 2)**

Estas son pequeñas mejoras de acabado para considerarlo **“cerrado profesionalmente”**:

| **Submódulo** | **Descripción** | **Tipo** |
| --- | --- | --- |
| 📨 **Confirmación de matrícula** | Enviar mensaje de confirmación (correo o WhatsApp simulado). | Backend |
| 🧾 **Comprobante PDF** | Generar constancia automática (opcional, para mostrar aprobación). | Backend |
| 📊 **Dashboard básico admin** | Gráfica o contador de matrículas: aprobadas / pendientes / rechazadas. | Frontend |
| 🔒 **Roles y seguridad** | Verificar que cada ruta tenga requireAdmin o requireAuth. | Backend |

Estas partes no son estrictamente necesarias, pero te sirven para:

* Presentar el Sprint 1 como *cerrado profesionalmente*.
* Pasar al Sprint 2 con una base sólida.

**✅ Conclusión**

👉 En términos académicos y funcionales:

**Sí, el Sprint 1 ya está completo.**  
Ya implementaste todo el flujo base de matrícula.

Solo si tu docente pide una *versión pulida* o *producto mínimo viable completo*,  
puedes añadir:

* constancia PDF,
* confirmación de matrícula (notificación),
* y dashboard con resumen visual.

**🧩 SPRINT 1 — Módulo de Matrícula Online (Finalización al 100%)**

**🎯 Objetivo general**

Digitalizar el proceso de matrícula de los estudiantes de la academia, permitiendo su registro, validación y aprobación administrativa.

**✅ 1. Funcionalidades ya implementadas**

✔️ Registro e inicio de sesión (con bcrypt + JWT).  
✔️ Formulario de matrícula dividido en pasos (Datos personales → Académicos → Pago).  
✔️ Carga de comprobante (multer).  
✔️ Registro en base de datos (Matricula, Usuario, Estudiante).  
✔️ Panel de administrador para aprobar o rechazar matrículas.  
✔️ Aprobación automática que crea estudiante real y limpia datos temporales.

**🔧 2. Mejoras finales para completarlo al 100%**

**🧾 2.1 Generar constancia PDF (opcional pero recomendado)**

* Al aprobar la matrícula, generar un comprobante tipo:
* CONSTANCIA DE MATRÍCULA
* Alumno: Aldo Quispe Locumber
* Modalidad: Ordinario
* Grupo: A
* Estado: Aprobado
* Fecha: 25/10/2025
* Guardarlo o mostrarlo en el frontend (/uploads/constancias/).

🧰 Puedes usar **pdfkit** o **pdfmake** para generarlo.

**📩 2.2 Notificación (email o WhatsApp simulado)**

* Al aprobar o rechazar una matrícula, mostrar alerta tipo:

“✅ Su matrícula ha sido aprobada, pronto recibirá su cronograma.”

* (Opcional) enviar correo usando **nodemailer** o mensaje vía **n8n**.

**📊 2.3 Dashboard básico para el administrador**

* Crear una vista /admin/dashboard con estadísticas:

| **Estado** | **Cantidad** |
| --- | --- |
| Pendientes | 12 |
| Aprobadas | 34 |
| Rechazadas | 5 |

* Usa una pequeña gráfica o tarjetas con colores (Tailwind / Chart.js).

**🔒 2.4 Seguridad y roles**

* Verificar rutas:
  + /api/matriculas → requireAuth
  + /api/matriculas/:id/aprobar → requireAdmin
  + /api/matriculas/:id/rechazar → requireAdmin
* Asegurar que el frontend redirija correctamente según el rol:
  + ADMIN → DashboardAdmin
  + ESTUDIANTE → Página de matrícula o perfil.

**💾 2.5 Copia de respaldo**

Antes de pasar al Sprint 2, crea un respaldo:

mysqldump -u root -p academia\_matriculas > backup\_sprint1.sql

Y copia tu proyecto (backend y frontend) como:

academia-matriculas-sprint1/

**🚀 Resultado esperado al cierre del Sprint 1**

✅ Flujo completo de matrícula en línea (registro → envío → validación).  
✅ Comprobante PDF o constancia visible.  
✅ Dashboard del administrador con resumen.  
✅ Roles correctamente aplicados.  
✅ Sistema listo para producción o despliegue de pruebas.