**UNIVERSIDAD REGIONAL AUTÓNOMA DE LOS ANDES “UNIANDES”**



**CARRERA DE INGENIERÍA DE SOFTWARE**

**Aplicación Móvil para la Gestión de Parqueaderos en la Universidad Autónoma de Los Andes**

**Casos de Uso y UML**

**ESTUDIANTE:**

1. **Bunshe Aguirre Janio Xavier**
2. **Calapi Muñoz Yolanda Patricia**
3. **Llanganate Muñoz Pintag Duchicela**

**NIVEL: SEPTIMO SOFTWARE**

**JUNIO 2024**

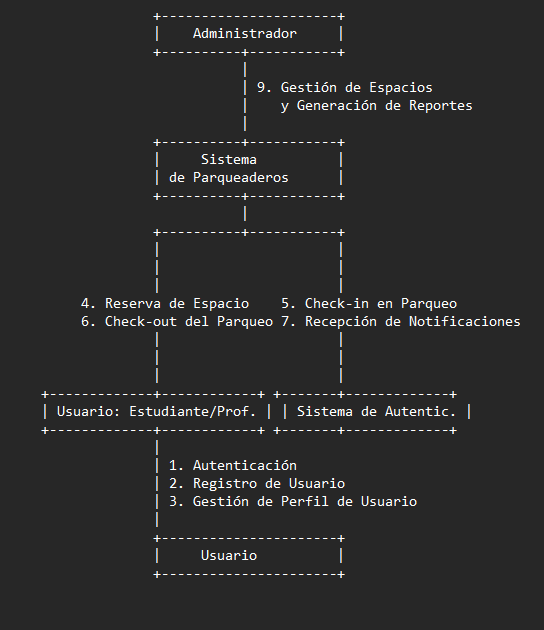
**Diagrama de Casos de Uso**

**Actores Principales**

* **Usuario:** Incluye estudiantes, profesores y personal administrativo que utilizan la aplicación para gestionar su parqueo.
* **Administrador:** Personal responsable de la gestión y administración de los parqueaderos.
* **Sistema de Autenticación:** Sistema institucional para la validación de credenciales.

**Casos de Uso**

1. **Autenticación de Usuario**
2. **Registro de Usuario**
3. **Gestión de Perfil de Usuario**
4. **Reserva de Espacio de Parqueo**
5. **Check-in en Parqueo**
6. **Check-out del Parqueo**
7. **Recepción de Notificaciones**
8. **Gestión de Espacios por Administrador**
9. **Generación de Reportes**

**Diagrama de Caso de Uso General**

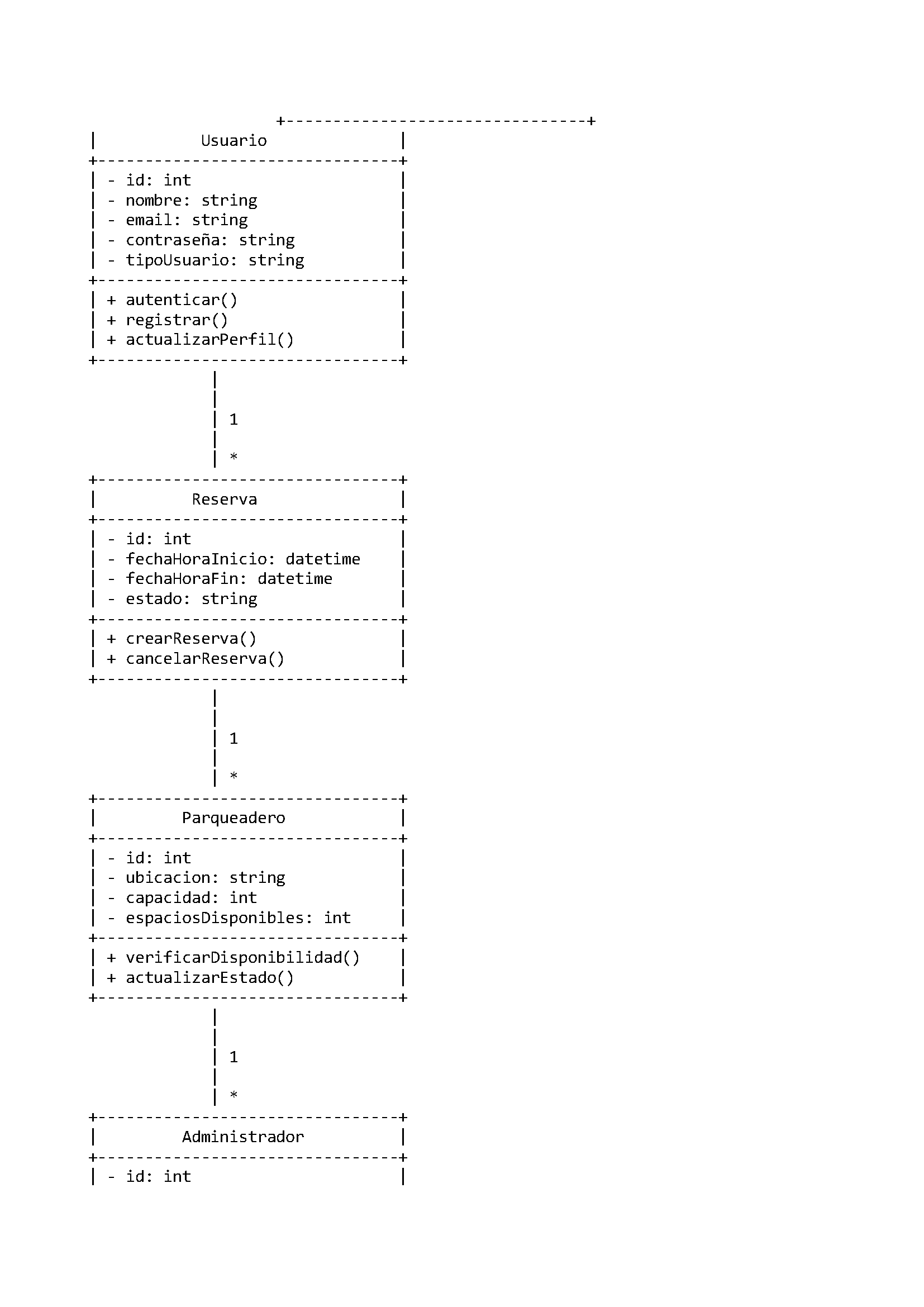
**Descripción de Casos de Uso**

1. **Autenticación de Usuario**
   * **Actor Primario:** Usuario
   * **Descripción:** El usuario se autentica utilizando sus credenciales institucionales.
   * **Precondiciones:** El usuario debe tener credenciales válidas.
   * **Flujo Principal:**
     1. El usuario abre la aplicación.
     2. El usuario ingresa sus credenciales.
     3. El sistema valida las credenciales con el Sistema de Autenticación.
     4. El usuario accede a la aplicación.
2. **Registro de Usuario**
   * **Actor Primario:** Usuario
   * **Descripción:** Registro de nuevos usuarios utilizando credenciales institucionales.
   * **Precondiciones:** El usuario debe tener credenciales válidas.
   * **Flujo Principal:**
     1. El usuario selecciona la opción de registro.
     2. El usuario ingresa sus credenciales.
     3. El sistema valida las credenciales y registra al usuario.
3. **Gestión de Perfil de Usuario**
   * **Actor Primario:** Usuario
   * **Descripción:** Actualización de información personal por parte del usuario.
   * **Precondiciones:** El usuario debe estar autenticado.
   * **Flujo Principal:**
     1. El usuario accede a su perfil.
     2. El usuario actualiza su información personal.
     3. El sistema guarda los cambios.
4. **Reserva de Espacio de Parqueo**
   * **Actor Primario:** Usuario
   * **Descripción:** Reserva de un espacio de parqueo específico.
   * **Precondiciones:** El usuario debe estar autenticado.
   * **Flujo Principal:**
     1. El usuario visualiza la disponibilidad de espacios de parqueo.
     2. El usuario selecciona un espacio y realiza la reserva.
     3. El sistema confirma la reserva.
5. **Check-in en Parqueo**
   * **Actor Primario:** Usuario
   * **Descripción:** Registro de entrada al parqueadero mediante el escaneo de credenciales.
   * **Precondiciones:** El usuario debe tener una reserva activa.
   * **Flujo Principal:**
     1. El usuario llega al parqueadero.
     2. El usuario escanea su credencial.
     3. El sistema registra el check-in.
6. **Check-out del Parqueo**
   * **Actor Primario:** Usuario
   * **Descripción:** Registro de salida del parqueadero mediante el escaneo de credenciales.
   * **Precondiciones:** El usuario debe haber realizado el check-in.
   * **Flujo Principal:**
     1. El usuario escanea su credencial al salir.
     2. El sistema registra el check-out.
7. **Recepción de Notificaciones**
   * **Actor Primario:** Usuario
   * **Descripción:** Recepción de notificaciones push sobre la proximidad de la reserva y cambios en la disponibilidad.
   * **Precondiciones:** El usuario debe tener notificaciones habilitadas.
   * **Flujo Principal:**
     1. El sistema envía notificaciones push al usuario.
     2. El usuario recibe y visualiza las notificaciones.
8. **Gestión de Espacios por Administrador**
   * **Actor Primario:** Administrador
   * **Descripción:** Gestión de espacios de parqueo (habilitación, deshabilitación).
   * **Precondiciones:** El administrador debe estar autenticado.
   * **Flujo Principal:**
     1. El administrador accede al panel de administración.
     2. El administrador gestiona los espacios de parqueo.
     3. El sistema guarda los cambios.
9. **Generación de Reportes**
   * **Actor Primario:** Administrador
   * **Descripción:** Generación de reportes sobre la utilización de los espacios de parqueo.
   * **Precondiciones:** El administrador debe estar autenticado.
   * **Flujo Principal:**
     1. El administrador accede al panel de administración.
     2. El administrador selecciona la opción de generar reportes.
     3. El sistema genera y muestra los reportes.

### Diagrama UML

#### Diagrama de Clases

A continuación, se presenta el diagrama de clases UML que representa los principales componentes de la aplicación móvil para la gestión de parqueaderos.



Texto, Carta

Descripción generada automáticamente

**Descripción de Clases**

1. **Usuario:**
   * Atributos:
     + id: Identificador único del usuario.
     + nombre: Nombre del usuario.
     + email: Correo electrónico del usuario.
     + contraseña: Contraseña del usuario.
     + tipoUsuario: Tipo de usuario (estudiante, profesor, personal administrativo).
   * Métodos:
     + autenticar(): Autentica al usuario en el sistema.
     + registrar(): Registra un nuevo usuario en el sistema.
     + actualizarPerfil(): Actualiza la información del perfil del usuario.
2. **Reserva:**
   * Atributos:
     + id: Identificador único de la reserva.
     + fechaHoraInicio: Fecha y hora de inicio de la reserva.
     + fechaHoraFin: Fecha y hora de fin de la reserva.
     + estado: Estado de la reserva (activa, cancelada).
   * Métodos:
     + crearReserva(): Crea una nueva reserva.
     + cancelarReserva(): Cancela una reserva existente.
3. **Parqueadero:**
   * Atributos:
     + id: Identificador único del parqueadero.
     + ubicacion: Ubicación del parqueadero.
     + capacidad: Capacidad total del parqueadero.
     + espaciosDisponibles: Número de espacios disponibles en el parqueadero.
   * Métodos:
     + verificarDisponibilidad(): Verifica la disponibilidad de espacios en el parqueadero.
     + actualizarEstado(): Actualiza el estado de los espacios de parqueo.
4. **Administrador:**
   * Atributos:
     + id: Identificador único del administrador.
     + nombre: Nombre del administrador.
     + email: Correo electrónico del administrador.
     + contraseña: Contraseña del administrador.
   * Métodos:
     + gestionarParqueadero(): Gestiona los espacios del parqueadero.
     + generarReporte(): Genera reportes sobre la utilización del parqueadero.
5. **Notificacion:**
   * Atributos:
     + id: Identificador único de la notificación.
     + mensaje: Mensaje de la notificación.
     + fechaHoraEnvio: Fecha y hora de envío de la notificación.
   * Métodos:
     + enviarNotificacion(): Envía una notificación al usuario.

Patricia Calapi   
Líder de Proyecto

Janio Bunshe   
Desarrollador / Analista

Pintag Llanganate   
Desarrollador / Analista