**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**

**Arquitectura del Sistema**

**ESTUDIANTE:**

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

**NIVEL: SEPTIMO SOFTWARE**

**JUNIO 2024**

**Arquitectura del Sistema**

**1. Cliente Móvil (Aplicación móvil)**

* **Tecnología:** Desarrollada en frameworks modernos como Flutter (para soporte multiplataforma) o nativo (Swift para iOS, Kotlin para Android).
* **Funcionalidades:**
  + Interfaz de usuario intuitiva para la gestión de parqueaderos.
  + Autenticación mediante credenciales institucionales.
  + Gestión de reservas, registro de entrada/salida, notificaciones y configuración de perfil.
  + Consumo de servicios del backend para obtener datos y realizar acciones.

**2. Backend (Servidor)**

* **Tecnología:** Puede implementarse utilizando tecnologías como Node.js, Python (Django o Flask), Java (Spring Boot) u otros frameworks robustos.
* **Funcionalidades:**
  + Gestión de usuarios y roles.
  + Gestión de espacios de parqueo, reservas y historial de uso.
  + Integración con el sistema de autenticación institucional (LDAP, SAML, OAuth, etc.).
  + API RESTful para la comunicación con el cliente móvil y otros sistemas.

**3. Base de Datos**

* **Tecnología:** MySQL, PostgreSQL u otro sistema de gestión de bases de datos relacional que sea compatible con las necesidades del sistema.
* **Esquema de Base de Datos:**
  + Tablas para usuarios, roles, espacios de parqueo, reservas, historial de uso y notificaciones según el diseño previamente discutido.

**4. Servicios Adicionales**

* **Autenticación y Autorización:**
  + Integración con el sistema de autenticación institucional para validar credenciales de usuarios.
  + Implementación de mecanismos de autorización para asegurar que los usuarios solo puedan acceder a la información y funcionalidades correspondientes a su rol.

**5. Seguridad**

* **Protección de Datos:**
  + Uso de HTTPS para la comunicación segura entre el cliente móvil y el backend.
  + Encriptación de datos sensibles almacenados en la base de datos.
  + Implementación de buenas prácticas de seguridad para prevenir vulnerabilidades como inyecciones SQL y XSS.

**6. Integración Continua y Despliegue**

* **CI/CD (Continuous Integration/Continuous Deployment):**
  + Configuración de pipelines para automatizar pruebas, construcción y despliegue de la aplicación y servicios backend.
  + Uso de plataformas como GitLab CI, Jenkins, CircleCI, entre otros, dependiendo de las preferencias del equipo de desarrollo.

**Beneficios de esta Arquitectura**

* **Escalabilidad:** Permite escalar horizontalmente tanto el backend como el frontend para manejar incrementos en el número de usuarios y transacciones.
* **Mantenibilidad:** Separación clara de responsabilidades entre frontend, backend y base de datos facilita el mantenimiento y las actualizaciones del sistema.
* **Seguridad:** Implementación de medidas robustas de seguridad para proteger la información confidencial de los usuarios y la integridad de los datos.