**Línea de referencia de la Etapa de Iniciación del Proyecto V1.0**

**Nombre del Proyecto:** Aplicación Móvil para la Gestión de Parqueaderos en la Universidad Autónoma de Los Andes

**Fecha:** 12 de julio del 2024

4. Diseño de la Interfaz

5. Arquitectura del Sistema

**Documento 1.-** 3. 4. Diseño de la Interfaz

|  |  |
| --- | --- |
| **Diseño de la Interfaz** | |
| **Pantalla** | **Elementos y Funcionalidades** |
| **Inicio de Sesión** | **Formulario de Inicio de Sesión:** Campos para correo electrónico institucional y contraseña. |
| **Opción de Recuperación de Contraseña:** Enlace para recuperar la contraseña olvidada. |
| **Botón de Inicio de Sesión:** Botón destacado para iniciar sesión rápidamente. |
| **Panel Principal (Dashboard)** | **Estado de Disponibilidad de Parqueaderos:** Vista clara con colores o iconos para diferenciar espacios disponibles y ocupados. |
| **Resumen de Reservas:** Muestra reservas activas, con opciones para cancelarlas o modificarlas. |
| **Acceso Rápido a Funcionalidades:** Botones o accesos directos para reservar espacio, registrar entrada/salida, ver historial, etc. |
| **Reserva de Parqueadero** | **Calendario de Disponibilidad:** Interfaz para seleccionar la fecha y hora deseada. |
| **Selección de Espacio de Parqueo:** Listado o mapa interactivo para elegir espacios disponibles. |
| **Confirmación de Reserva:** Pantalla de confirmación con detalles de la reserva antes de finalizarla. |
| **Gestión de Reservas y Historial** | **Lista de Reservas:** Muestra reservas pasadas y futuras, con opciones para ver detalles, modificar o cancelar. |
| **Historial de Uso:** Registro de entradas y salidas del usuario con fechas y tiempos de uso. |
| **Notificaciones y Mensajes** | **Pantalla de Notificaciones:** Lista de notificaciones, con opciones para marcar como leídas, eliminar o marcar como importante. |
| **Mensajes y Avisos:** Espacio para comunicaciones importantes sobre cambios en políticas o eventos especiales. |
| **Perfil de Usuario** | **Información Personal:** Pantalla para ver y editar información personal como nombre, correo electrónico y credenciales. |
| **Configuración de Preferencias:** Opciones para ajustar notificaciones, idioma de la aplicación y otras preferencias. |
| **Consideraciones de Diseño** | **Diseño Responsivo:** Compatible con diferentes tamaños de pantalla, desde teléfonos pequeños hasta tablets grandes. |
| **Accesibilidad:** Colores y tamaños de fuente legibles para todos, incluidos usuarios con discapacidades visuales. |
| **Navegación Intuitiva:** Navegación sencilla y directa para encontrar rápidamente lo que se busca. |
| **Iconografía y Visualización de Datos:** Uso de iconos claros y gráficos simples para una fácil comprensión. |

**Documento 2.-** 5. Arquitectura del Sistema

|  |  |
| --- | --- |
| **Arquitectura** | |
| **Componente** | **Descripción** |
| **Cliente Móvil** | **Tecnología:** Flutter (multiplataforma), Swift (iOS), Android Studio (Android) |
| **Funcionalidades:** |
| - Interfaz intuitiva para la gestión de parqueaderos |
| - Autenticación con credenciales institucionales |
| - Gestión de reservas, entrada/salida, notificaciones y perfil |
| - Consumo de servicios del backend para datos y acciones |
| **Backend (Servidor)** | **Tecnología:** Node.js, Python (Django/Flask), Java (Spring Boot) u otros |
| **Funcionalidades:** |
| - Gestión de usuarios, roles, espacios de parqueo, reservas e historial |
| - Integración con sistema de autenticación institucional (LDAP, SAML, OAuth) |
| - API RESTful para comunicación con cliente móvil y otros sistemas |
| **Base de Datos** | **Tecnología:** MySQL, PostgreSQL u otro sistema de gestión de bases de datos relacional |
| **Esquema de Base de Datos:** |
| - Tablas para usuarios, roles, espacios de parqueo, reservas, historial de uso, notificaciones |
| **Servicios Adicionales** | **Autenticación y Autorización:** |
| - Integración con sistema de autenticación institucional |
| - Mecanismos de autorización para acceso basado en roles |
| **Seguridad** | **Protección de Datos:** |
| - Encriptación de datos sensibles en la base de datos |
| - Buenas prácticas para prevenir vulnerabilidades (inyección SQL, XSS) |
| **Integración Continua y Despliegue** | **CI/CD (Continuous Integration/Continuous Deployment):** |
| - Pipelines para pruebas, construcción y despliegue automatizados |
| - Plataformas como GitLab CI, Jenkins, CircleCI, según preferencias del equipo |

Pintag Llanganate   
Desarrollador / Analista

Janio Bunshe   
Desarrollador / Analista

Patricia Calapi   
Líder de Proyecto