**Tarea 3: Desarrollar prototipos**

**3.1 Objetivos**

**3.1.1 Objetivo General**

Construir prototipos del sistema, que representen visualmente las interfaces y funcionalidades principales, con el propósito de la obtención de requisitos relativos a la interacción con el usuario, la detección de conflictos en los requisitos definidos y la profundización en su análisis.

**31.2 Objetivos Específicos**

Diseñar representaciones visuales de las pantallas del sistema, enfocadas en el registro de usuarios, la validación de accesos y la consulta de reportes.

Obtener observaciones sobre la usabilidad, organización y comprensión, mediante la revisión del prototipo por parte de usuarios representativos.

Detectar inconsistencias entre los flujos previstos (casos de uso) y la estructura actual del prototipo, como pasos omitidos, validaciones faltantes o ambigüedades en los requisitos funcionales y no funcionales

Evaluar la coherencia entre la experiencia del usuario ofrecida por el prototipo y los requisitos funcionales esperados

**3.2 Descripción**

En esta tarea se deben desarrollar prototipos que permitan tanto a los

clientes y usuarios como a los desarrolladores tener una idea más clara

del sistema a desarrollar e identificar nuevos requisitos o conflictos que

hayan permanecidos ocultos hasta el momento. Lo más habitual es que el

prototipo sea desechable, es decir, que una vez que se haya utilizado no se

desarrolle tomando su código como base.

***3.4 Productos entregables***

**Prototipo del sistema** 

Se desarrolló un prototipo funcional básico utilizando PHP y HTML, ejecutable en un entorno local como XAMPP o Laragon. Este enfoque permitió representar la interacción de los usuarios con el sistema de forma sencilla y directa, sin requerir una base de datos real o autenticación compleja.

**El prototipo incluyó las siguientes pantallas:**

Inicio de sesión simulado: Permite acceder al panel de control general.

Panel principal: Ofrece enlaces a las funciones básicas del sistema.

Formulario de registro de vehículos: Captura datos como placa, tipo, modelo y color.

Módulo de validación de acceso: Simula el ingreso de una placa y muestra si el acceso es permitido o denegado según una regla predefinida.

**Aspectos de interfaz de usuarios del modelo de comportamiento**

El modelo de comportamiento representa cómo actúan los objetos del sistema frente a las operaciones y eventos que ocurren durante su ejecución.   
Se identificaron elementos dinámicos clave:

Operaciones como registrar vehículo, validar acceso, emitir alertas, consultar disponibilidad del parqueadero y administrar el sistema.

Objetos con comportamiento significativo: Vehículo (autorizado/no autorizado), Acceso (pendiente, permitido, denegado), Parqueadero (actualiza disponibilidad), Alerta (activa/cerrada).

**Diagramas desarrollados:**

Diagrama de traza de eventos (validación de acceso vehicular).

Diagrama de estados (objeto Vehículo y sus transiciones).

Plantilla de operación ValidarAccesoVehicular, que detalla condiciones, errores y respuestas del sistema.

**Conflictos detectados** 

Durante el modelado del comportamiento se identificaron los siguientes conflictos:

Ambigüedad en el método de validación: No se define si tiene prioridad el QR o la lectura de placas ni cómo gestionar fallos.

Falta de manejo de excepciones en horarios restringidos: No se detalla si hay criterios de flexibilidad.

Eventos no cubiertos: No se especifican flujos alternativos ante fallas de red, lectores o errores del sistema.

Estado del vehículo poco definido: No contempla permisos temporales o condicionados.

Inconsistencias en la sincronización de datos: No se aclara cómo se sincronizan eventos offline o disponibilidad en tiempo real del parqueadero.

***3.5 Técnicas***

Para la construcción del prototipo del Sistema de Control de Acceso Vehicular de la Universidad de Pamplona, se aplicó la técnica de prototipado de sistemas software, descrita en la sección 4.6 de la metodología. Esta técnica consiste en representar de forma anticipada y operativa las funcionalidades clave del sistema mediante interfaces simuladas, con el fin de validar la lógica y facilitar la retroalimentación temprana.

En este caso, se desarrolló un prototipo funcional básico utilizando PHP y HTML, ejecutable en un entorno local como XAMPP o Laragon. Este enfoque permitió representar la interacción de los usuarios con el sistema de forma sencilla y directa, sin requerir una base de datos real o autenticación compleja.

* El prototipo incluyó las siguientes pantallas:
* Inicio de sesión simulado: Permite acceder al panel de control general.
* Panel principal: Ofrece enlaces a las funciones básicas del sistema.
* Formulario de registro de vehículos: Captura datos como placa, tipo, modelo y color.
* Módulo de validación de acceso: Simula el ingreso de una placa y muestra si el acceso es permitido o denegado según una regla predefinida.

