Naranja Segura V1.1

Detalles técnicos sobre la implementación del modelo Vista-Controlador y las funcionalidades propuestas.

Integrantes:
Fernanda Paulina Ávila González* fernapau14@gmail.com
José Alberto Valencia López* javalencia2006@gmail.com
Marco Antonio Alonso Medina del Angel* medina.marco.antonio99@gmail.com

*Instituto Tecnológico de Morelia, Ingeniería en Tecnologías de la Información y Comunicaciones

Contenido

Descripción de la solución tecnológica propuesta

La solución propuesta es una aplicación móvil prototipo orientada a brindar ayuda inmediata a personas en situación de peligro o emergencia dentro de la ciudad de Morelia.

La app permite al usuario visualizar un mapa interactivo con “puntos naranja” (lugares seguros), activar un botón de emergencia que envía su ubicación a contactos de confianza y recibir indicaciones para dirigirse al punto seguro más cercano. Esta solución está diseñada con un enfoque emocional, accesible y orientado a la protección de las personas en riesgo.

Justificación de la innovación y abordaje de la seguridad

La innovación de esta aplicación se centra en el bienestar del usuario, combinando diseño intuitivo con una experiencia empática, clara, eficiente y rápida, enfocada en generar confianza en situaciones de estrés. Esto se logra al ofrecer un botón de emergencia para monitorear a la víctima y dirigirla a un lugar seguro, así como al mostrar reportes actuales para hacer conciencia en la comunidad de la violencia que acontece al momento.

El Modelo-Vista-Controlador

La solución tecnológica propuesta se estructura bajo el patrón de arquitectura **Modelo-Vista-Controlador (MVC)**, con el objetivo de mantener una separación clara entre la lógica de presentación (interfaz de usuario), la lógica de negocio y la gestión de datos. Esta arquitectura permitirá una mayor escalabilidad, mantenibilidad y seguridad en la evolución del proyecto.

- **El Modelo** se encargará de gestionar los datos y estructuras lógicas como el usuario, la ubicación, los puntos seguros y los contactos de emergencia.

- **La Vista** estará compuesta por las pantallas que conforman la interfaz de usuario, como el mapa interactivo, el botón de emergencia, y los formularios de registro e inicio de sesión.

- **El Controlador** actuará como intermediario entre las vistas y los modelos, gestionando eventos, validaciones y comunicación entre los componentes de la app.

Casos De Uso

Para comprender de forma estructurada las funcionalidades clave de la aplicación y cómo interactúan los distintos tipos de usuarios con el sistema, se presenta a continuación un **diagrama de casos de uso**. Este diagrama permite identificar:

- Las **acciones principales** disponibles dentro de la aplicación.
- Los **actores involucrados** (usuarios, sistema, servicios externos).
- La relación entre cada actor y las funciones específicas que puede ejecutar.

El propósito de este diagrama es proporcionar una visión global y simplificada del comportamiento esperado del sistema desde el punto de vista del usuario, sirviendo como base para el diseño detallado de la lógica, la interfaz y la arquitectura del software. Además, facilita la comunicación entre los equipos de desarrollo, diseño y validación, asegurando que todas las necesidades funcionales sean correctamente implementadas y entendidas.

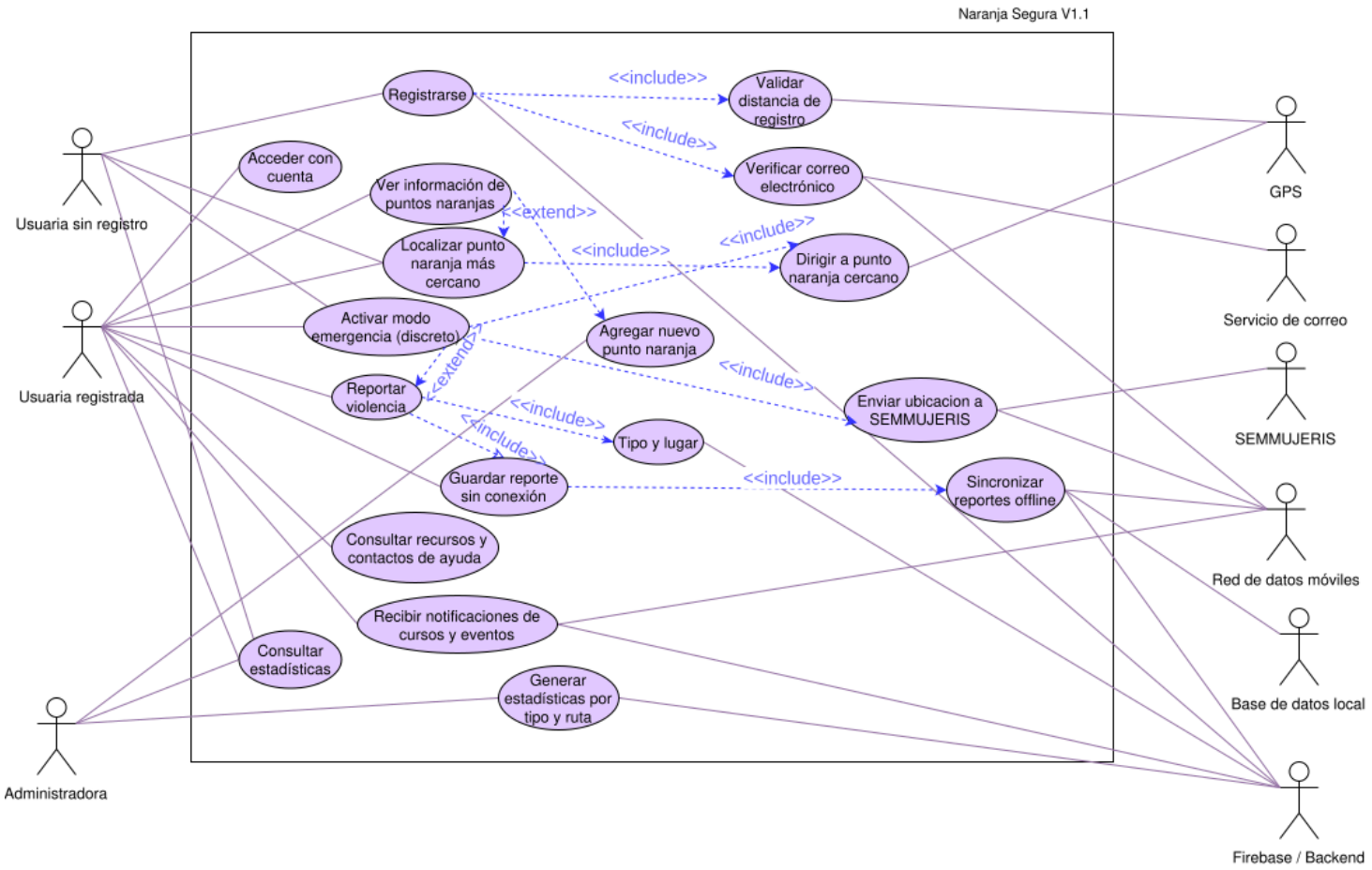

Figura 1: Diagrama de casos de uso para la app Naranja Segura

A continuación se detallan los principales casos de uso de la aplicación:

CU-001	**Abrir aplicación**
Actor	Usuaria
Prioridad	Alta
Descripción	Permite iniciar el uso de la aplicación
Precondiciones	Tener la app instalada
Flujo normal	1. La usuaria toca el ícono de la app 2. Se carga la pantalla de inicio con opciones: Registrar, Iniciar sesión, Ver rutas peligrosas, Modo emergencia
Flujo alternativo	-

Postcondiciones	La usuaria puede navegar hacia el registro, login, rutas o emergencia

CU-002	**Registrarse**
Actor	Usuaria sin cuenta
Prioridad	Alta
Descripción	Permite crear una cuenta nueva
Precondiciones	Tener conexión a internet, estar dentro del radio permitido
Flujo normal	1. La usuaria selecciona "Registrarse" 2. Llena el formulario 3. Se valida la ubicación (radio de 10 km) 4. Se envía correo de verificación
Flujo alternativo	Si no está en el radio, no se permite continuar
Postcondiciones	Se crea un usuario inactivo, pendiente de verificación

CU-003	**Verificar correo electrónico**
Actor	Servidor de correo, usuaria
Prioridad	Alta
Descripción	Verifica el correo tras registro
Precondiciones	Haber registrado una cuenta
Flujo normal	1. Se recibe correo 2. La usuaria hace clic en el enlace 3. El sistema valida el token
Flujo alternativo	El token ha expirado o es inválido → mostrar error
Postcondiciones	El usuario queda activado y puede iniciar sesión

CU-004	Iniciar sesión
Actor	Usuaria registrada
Prioridad	Alta
Descripción	Accede al menú principal
Precondiciones	Haber verificado correo
Flujo normal	1. Ingresar correo y contraseña 2. Autenticación exitosa 3. Ir al menú principal
Flujo alternativo	Usuario no verificado → mostrar advertencia
Postcondiciones	Sesión iniciada, se habilitan las funciones completas

CU-005	Menú principal
Actor	Usuaria registrada
Prioridad	Alta
Descripción	Pantalla principal con opciones
Precondiciones	Haber iniciado sesión
Flujo normal	1. Se muestra menú con opciones: rutas peligrosas, emergencia, hacer reporte, mapa, puntos cercanos
Flujo alternativo	-
Postcondiciones	Navega hacia una de las opciones disponibles

CU-006	Modo emergencia
Actor	Usuaria registrada
Prioridad	Crítica

Descripción	Activa el protocolo de emergencia de forma discreta
Precondiciones	Haber iniciado sesión
Flujo normal	1. Accede al modo emergencia 2. Se activa localización 3. Se redirige a "hacer reporte"
Flujo alternativo	Puede regresar al menú principal si presiona atrás
Postcondiciones	El sistema guarda el evento y redirige según acción tomada

CU-007	**Hacer reporte**
Actor	Usuaria registrada
Prioridad	Alta
Descripción	Permite reportar violencia
Precondiciones	Estar autenticada
Flujo normal	1. La usuaria describe la situación 2. Indica tipo de violencia y ruta 3. Se envía (o guarda local si no hay red)
Flujo alternativo	Sin conexión: guardar en local y marcar como pendiente
Postcondiciones	El reporte se guarda y se sincroniza luego si es necesario

CU-008	**Ver rutas peligrosas**
Actor	Cualquier usuaria
Prioridad	Media
Descripción	Consulta estadísticas de violencia por ruta
Precondiciones	-

Flujo normal	1. Se accede desde inicio o menú 2. Se muestran rutas peligrosas visualmente
Flujo alternativo	Sin datos → mostrar mensaje de “sin datos disponibles”
Postcondiciones	Información consultada

CU-009	**Ver mapa y puntos cercanos**
Actor	Usuaria registrada
Prioridad	Alta
Descripción	Muestra un mapa y puntos naranja cercanos
Precondiciones	GPS activo
Flujo normal	1. Ver mapa 2. Tocar un punto para ver detalles (horario, foto, dirección)
Flujo alternativo	No hay GPS → pedir permiso o mostrar aviso
Postcondiciones	Ubicación visible, punto naranja identificado

Flujo de interacción

Para garantizar una experiencia de usuario fluida, intuitiva y coherente con los objetivos de la aplicación, se desarrolló un **diagrama de flujo de interacción del usuario**, el cual permite visualizar **paso a paso el comportamiento del usuario desde que abre la aplicación hasta que interactúa con cada una de sus funciones clave**: registro, verificación de correo, activación del modo emergencia, visualización de rutas peligrosas, reporte de incidentes, entre otros.

Este diagrama cumple dos funciones fundamentales:

- **Diseño centrado en el usuario**: Permite anticipar necesidades, reducir fricción en la navegación y facilitar el acceso a herramientas críticas (como el botón de emergencia o los puntos naranjas).

- **Soporte para la arquitectura MVC**: El flujo permite mapear claramente qué elementos corresponden al **Modelo** (datos y lógica de negocio, como el estado de verificación de un usuario), a la **Vista** (pantallas como mapa, menú principal o formulario de reporte), y al **Controlador** (gestión de navegación, validación de acciones, redirecciones según el estado del usuario).

Por ejemplo, cuando un usuario no verificado intenta acceder al menú principal, el controlador detecta el estado de verificación (modelo) y lo redirige a la vista de verificación de correo, impidiendo el acceso hasta que se cumpla la condición requerida.

Gracias a este enfoque, se garantiza que cada interacción del usuario esté alineada con los principios de **usabilidad, seguridad** y **relevancia funcional**, especialmente en situaciones de riesgo donde el tiempo y la discreción son factores clave.

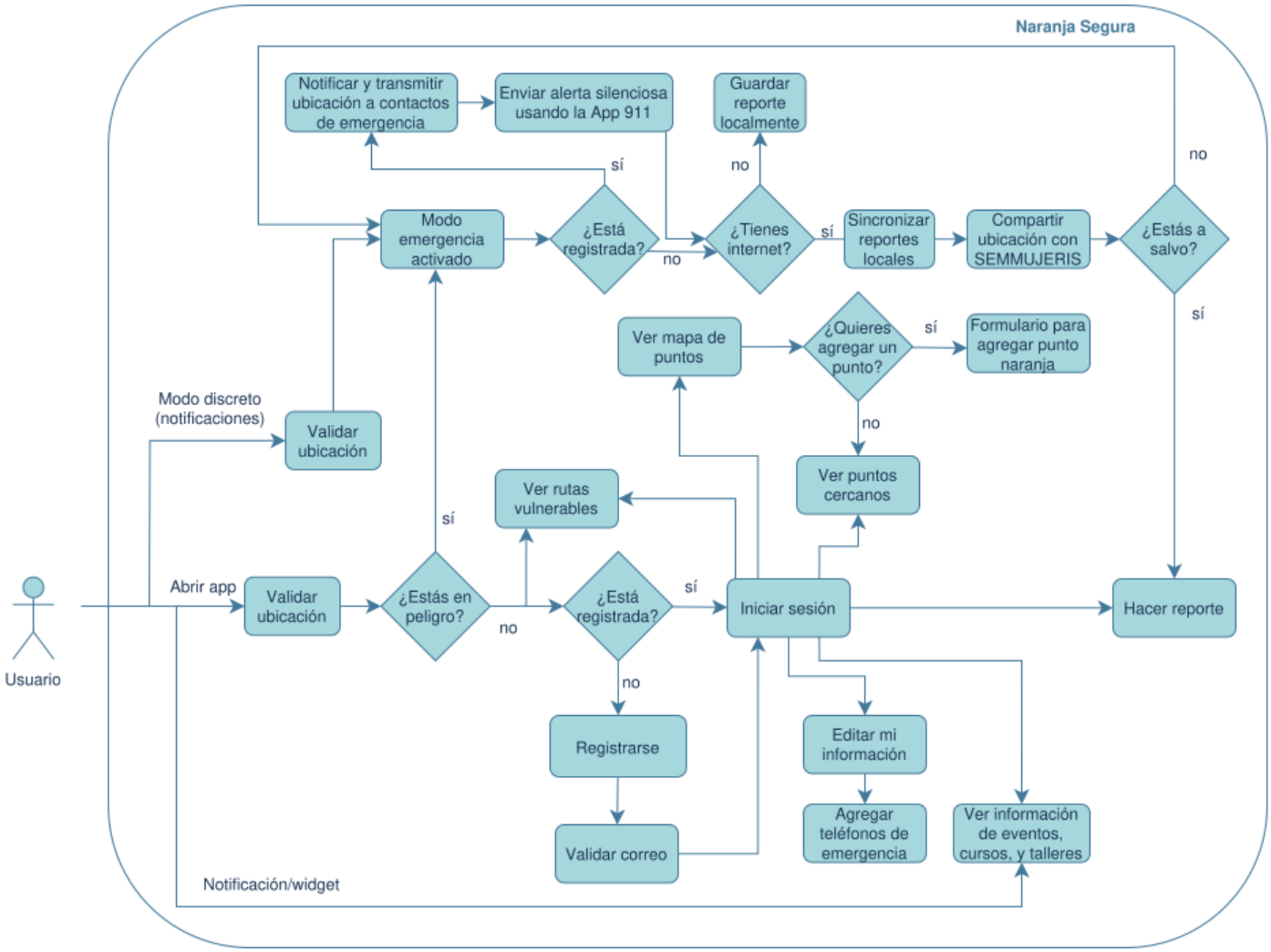

Figura 2: Flujo de interacción general de la app Naranja Segura

Requerimientos técnicos

Para el desarrollo e implementación futura de la aplicación móvil de seguridad ciudadana, se consideran los siguientes requerimientos técnicos:

- **Plataforma objetivo:** Android (fase inicial). Se contempla futura compatibilidad con iOS.
- **Lenguaje de programación (planeado):** Dart con Flutter (multiplataforma)[1].
- **Backend:** Firebase (Authentication, Firestore, Realtime Database, y Cloud Functions)[2].

[1] https://docs.flutter.dev/get-started/fundamentals/dart
[2] https://firebase.google.com/docs/functions

- **Servicios:**
    - Geolocalización en tiempo real.
    - Envío de notificaciones push.
    - Integración con mapas (Google Maps API)
- **Requisitos del dispositivo:**
    - Sistema operativo Android 8.0 o superior.
    - Conexión a internet (WiFi o datos móviles).
    - Acceso a GPS y permisos de ubicación.
    - Permiso para ejecutar notificaciones y servicios en segundo plano.

Requisitos funcionales

A continuación se listan las funcionalidades clave que debe cumplir el sistema:

- RF1. El usuario puede registrarse con correo electrónico o autenticarse con redes sociales (Google/Facebook).
- RF2. El usuario puede iniciar sesión de manera segura.
- RF3. El usuario puede ver un mapa interactivo con puntos de ayuda cercanos (puntos naranjas).
- RF4. El usuario puede activar un modo de **emergencia** con un botón visible desde el momento en que abre la aplicación, o desde la barra de notificaciones.
- RF5. Al activarse el modo de emergencia, se enviará la ubicación en tiempo real, hasta que se sienta segura, a contactos seleccionados de confianza y a usuarios cercanos.
- RF6. El sistema redirige al usuario al **punto seguro más cercano** disponible.
- RF7. Los puntos seguros se pueden actualizar dinámicamente desde el backend.
- RF8. El usuario puede agregar o editar su lista de contactos de emergencia.
- RF9. El usuario puede consultar información sobre eventos, teléfonos de servicios de emergencia, protocolos de seguridad y prevención.

Requisitos no funcionales

- RNF1. **Usabilidad:** La aplicación debe ser intuitiva, accesible y fácil de navegar, especialmente bajo situaciones de estrés o emergencia. Parcialmente funcional incluso sin registro previo
- RNF2. **Rendimiento:** La aplicación debe responder de la manera más rápida posible al activar el botón de emergencia.
- RNF3. **Disponibilidad:** La app debe estar disponible al menos el 99% del tiempo, especialmente en horarios nocturnos (Se tomarán en cuenta los horarios de los puntos naranjas cercanos). Debe ser posible activar el modo emergencia aún sin conexión a internet (enlazando a la App 911 y descargando un mapa de Morelia).

- RNF4. **Seguridad:** Los datos personales deben estar cifrados, y el acceso a la ubicación debe seguir las mejores prácticas de privacidad (autorización explícita del usuario).

- RNF5. **Compatibilidad:** La aplicación debe funcionar en la mayoría de dispositivos Android con versiones desde 8.0.

- RNF6. **Escalabilidad:** El sistema debe permitir agregar nuevos puntos seguros, usuarios y funcionalidades (actualizaciones, novedades, mejoras de versiones) sin comprometer el rendimiento.

Desarrollo visual

El objetivo principal de este modelo de la aplicación, es ejemplificar cómo podría realizarse la interfaz gráfica. Algunos aspectos a tomar en cuenta son:

- **Simplicidad:** Se busca desarrollar una interfaz sencilla de utilizar y de entender para los usuarios.
- **Fácil Acceso:** Se busca que la aplicación tenga un fácil acceso para emergencias. El peso de la aplicación no puede ser muy grande para poder ser soportada en la mayor cantidad de dispositivos posibles.
- **Diseño:** Se tomarán en cuenta aspectos como la fuente y la paleta de colores utilizados en las páginas del gobierno de Michoacán.

Normativas sobre el uso responsable de la tecnología

Para que el entorno de la aplicación sea completamente seguro y cumpla con todos los requerimientos establecidos por los organismos gubernamentales al ser una aplicación de estos mismos, se tomarán en cuenta diferentes aspectos tales como:

1. **Ley Federal de Protección de Datos Personales en Posesión de los Particulares (LFPDPPP)**
    - **Consentimiento informado:** Antes de recolectar datos (ubicación, nombre, correo electrónico, contactos de confianza), se solicitará autorización explícita del usuario.
    - **Aviso de privacidad accesible:** Se publicará dentro de la app y especificará:
        - Qué datos se recolectan y por qué.
        - Quién los gestiona y cómo pueden modificarse o eliminarse.
    - **Minimización de datos y seguridad:** Solo se solicitarán los datos estrictamente necesarios para el funcionamiento de la app. Los datos serán cifrados y no compartidos con terceros sin autorización.

2. **Ley Olimpia**
    - Protección de la privacidad de todas los usuarios, especialmente en el uso de imágenes, audio o video.
    - Moderación de contenido en los reportes, evitando la difusión de material íntimo sin consentimiento.
    - Sistema de anonimato opcional, para reportar sin comprometer la identidad si así se desea.
    - Almacenamiento cifrado y temporal de archivos sensibles anexados en reportes, que solo podrán ser consultados por instancias autorizadas.

- Educación digital: acceso a recursos y materiales para comprender qué es la violencia digital y cómo protegerse.

Medidas básicas de ciberseguridad

1. **Autenticación segura**
   - Implementación de inicio de sesión con verificación por correo.
   - Reglas para contraseñas fuertes (mayúsculas, símbolos, longitud mínima).
2. **Manejo responsable de datos**
   - Toda la comunicación se realizará mediante HTTPS.
   - Se evitará el almacenamiento innecesario de información sensible.
   - Se planea crear una base de datos para poder tener un registro de los casos que se han dado.
3. **Prevención de robo de identidad**
   - Verificación de identidad mediante correo electrónico.
   - Opción para que la usuaria elimine su cuenta y sus datos en cualquier momento.
   - Notificaciones ante actividad sospechosa.
4. **Anonimato**
   - Al ser una aplicación que se centrará en tratar temas sensibles, los procesos llevados a cabo serán de manera anónima, a menos que el usuario acceda a compartir sus datos personales.
5. **Seguimiento**
   - Si la usuaria decide llevar el seguimiento a un reporte o realizar una denuncia en contra de un agresor, podrá solicitar orientación al organismo gubernamental adecuado. Se planea utilizar los registros dentro de una base de dato para mantener la información y llevar un seguimiento adecuado.
6. **Desinformación**
   - En caso de incorporar noticias o recomendaciones, estas se verificarán mediante fuentes oficiales como:
     - SEMUJERES/INMUJERES
     - Gobierno de Michoacán
   - Posibilidad de agregar una sección de “contenido verificado” en futuras actualizaciones.

Diagrama de flujo para reflejar interacciones entre pantallas

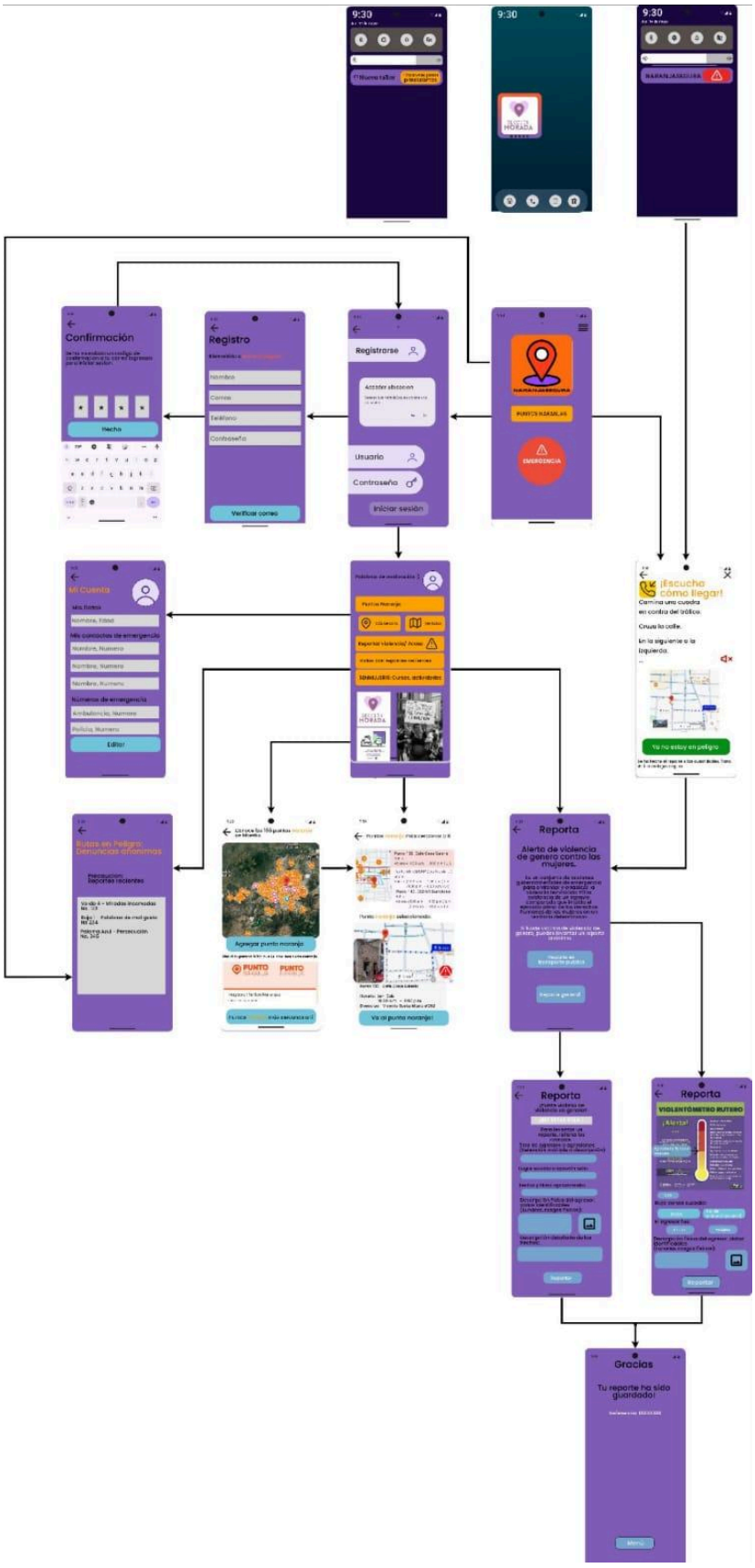

Figura 3: Flujo de interacción de las pantallas de la app Naranja Segura