

Diseño e implementación de una aplicación móvil para registrar la asistencia de ejidatarios mediante lectura de código QR

¹Mtro. Ivan Rafael Sánchez Juárez, ²Dra. María Petra Paredes Xochihua, ³Mtro. Fidel Paredes Xochihua

Tecnológico Nacional de México / ITS de San Martín Texmelucan

Resumen – El presente artículo propone el diseño e implementación de una aplicación móvil que permita optimizar el registro de la asistencia de ejidatarios mediante la lectura de códigos QR, esto con la finalidad de sustituir el proceso tradicional de pase de lista (mismo que es lento y con posibles errores de registro) en el núcleo ejidal de San Rafael Ixtapalucan, Puebla. La metodología de desarrollo elegida para este proyecto es SCRUM, las herramientas que se utilizaron para el desarrollo del proyecto son: Flutter (framework de desarrollo), Dart (lenguajes de programación), SQFlite (base de datos local), qr_flutter (para la generación y visualización de códigos QR), mobile_scanner (útil para el escaneo de códigos QR con cámara), syncfusion_flutter_xlsio (permite la exportación a Excel), share_plus (para compartir archivos y QR), el patrón de diseño es Modelo Vista Controlador (MVC), entre otras herramientas. La aplicación permite a los ejidatarios registrar su asistencia escaneando un código QR que está en sus credenciales, eso permite disminuir los errores que se pueden cometer en el registro manual y agilizando el proceso. La aplicación tiene como funciones principales la creación de ejidatarios, eventos, reportes; generación y escaneo de códigos QR, registro de asistencias en tiempo real, exportar el reporte en formato Excel para descargar y/o compartir por WhatsApp.

Palabras clave – Códigos QR, ejidatarios, flutter, pase de lista, scrum

Abstract: This article proposes the design and implementation of a mobile application that optimizes the registration of ejidatarios attendance by reading QR codes, with the purpose of replacing the traditional roll call process (which is slow and with possible registration errors) in the ejidal nucleus of San Rafael Ixtapalucan, Puebla. The development methodology chosen for this project is SCRUM, the tools that were used for the development of the project are: Flutter (development framework), Dart (programming languages), SQFlite (local database), qr_flutter (for the generation and visualization of QR codes), mobile_scanner (useful for scanning QR codes with a camera), syncfusion_flutter_xlsio (allows export to Excel), share_plus (for sharing files and QR), the design pattern is Model View Controller (MVC), among other tools. The app allows ejidatarios to register their attendance by scanning a QR code on their credentials, reducing errors that can occur during manual registration and streamlining the process. The app's main functions include creating ejidatarios, events, and reports; generating and scanning QR codes; recording attendance in real time; and exporting the report in Excel format for downloading and/or sharing via WhatsApp.

Keywords: ejidatarios, flutter, roll call, scrum, QR Codes

I. INTRODUCCION

En los núcleos agrarios de México, la asistencia de los ejidatarios a las diferentes actividades que tienen como asambleas o reforestaciones es un requisito fundamental para la toma de decisiones colectivas y da validez a los acuerdos a los que lleguen. De acuerdo a la ley agraria vigente establece que cuando se reúna por virtud de primera convocatoria a la asamblea, deberán estar presentes cuando menos la mitad más uno de los ejidatarios [3], mientras que en el artículo 27 establece que las resoluciones de la asamblea se tomarán válidamente por mayoría de los votos ejidatarios presentes. Por tal motivo se debe llevar un control muy estricto y detallado de las personas que asisten a dichos eventos.

En las comisarías ejidales, el número de ejidatarios suele ser considerable, lo que complica el proceso de registro de asistencia cuando se realiza de forma manual. Este método genera importantes pérdidas de tiempo por diversas razones. En primer lugar, al registrar la asistencia, es necesario buscar el nombre o número del ejidatario en hojas impresas y anotar su presencia, o bien solicitar que el propio ejidatario escriba su nombre y firma, lo cual retrasa el proceso. En segundo lugar, para contabilizar a los asistentes, se debe hacer un conteo uno por uno, lo que también consume tiempo importante. Finalmente, la búsqueda posterior de registros específicos, como verificar si un ejidatario asistió o no a una asamblea pasada, resulta igualmente lenta e ineficiente. Por ello, se hace necesario modernizar el proceso de pase de lista en las comisarías ejidales.

Uno de los Objetivos de Desarrollo Sostenible (ODS) establecidos por las Naciones Unidas es el número 16: Paz, justicia e instituciones sólidas. En particular, la meta 16.6 plantea la necesidad de crear instituciones eficaces, responsables y transparentes en todos los niveles [10]. En este sentido, el desarrollo de la aplicación propuesta contribuye directamente a dicho objetivo, al implementar una solución tecnológica en un contexto rural que promueve la digitalización, mejora la eficiencia en los procesos

comunitarios y fortalece la rendición de cuentas en la gestión ejidal.

El objetivo de este trabajo es diseñar e implementar una aplicación móvil para registrar la asistencia de ejidatarios mediante la lectura de códigos QR, con el fin de sustituir el pase de lista manual utilizado actualmente en el núcleo ejidal de San Rafael Ixtapalucan, Puebla.

METODOLOGÍA

El desarrollo de la aplicación se llevó a cabo utilizando el marco de trabajo Scrum, debido a que detalla los miembros de un equipo de Scrum y sus responsabilidades, los "artefactos" que definen el producto y el trabajo que hay que hacer para crear el producto, así como las ceremonias de Scrum que guían al equipo de Scrum en su trabajo [2]. Por otro lado [11] indica que en Scrum se realizan entregas parciales y regulares del producto final, priorizadas por el beneficio que aportan al receptor del proyecto.

El proyecto se dividió en sprints de dos semanas, donde al final de cada ciclo se evaluaban avances funcionales y se integraban mejoras. El proceso se estructuró en las siguientes fases:

1. Levantamiento de requerimientos. Es un proceso en el desarrollo de software; con el cuál se recopila y analiza información sobre los requisitos del usuario para un sistema o aplicación en particular [4]. En ésta etapa se realizaron entrevistas con representantes de la comisaría ejidal para identificar las necesidades de la aplicación, el proceso actual de pase de lista y los problemas que se han identificado con el registro manual de asistencia.

2. Diseño de la aplicación. Para diseñar una buena aplicación, el equipo debe analizar el dominio empresarial, capturar los requisitos del sistema, registrar los casos de uso del sistema y la interacción esperada de los usuarios, analizar y capturar información sobre el dominio de aplicaciones y, finalmente, producir un diseño detallado [9]. Para esta fase se elaboraron historias de usuario y unos wireframes realizados en Figma, validado con los usuarios finales.

Figma Design sirve para crear, compartir y probar diseños de sitios web, aplicaciones móviles y otros productos y experiencias digitales. Es una herramienta muy popular entre diseñadores, gestores de productos, redactores y desarrolladores, y ayuda a cualquier persona implicada en el proceso de diseño a contribuir, dar su opinión y tomar mejores decisiones, más rápido [6].

3. Selección de herramientas para el desarrollo. En ésta fase se eligieron las siguientes herramientas: Dart como lenguaje de programación, Flutter como marco de interfaz de usuario, SQFLite que es un paquete de Flutter para la base de datos y los módulos necesarios para generar archivos de Excel y códigos QR que posteriormente se podrán compartir mediante WhatsApp; como entorno de desarrollo se utilizó Visual Studio

Code ya que se tienen plugins importantes para desarrollar con Dart y Flutter.

Dart es un lenguaje optimizado para el cliente que permite desarrollar aplicaciones rápidas en cualquier plataforma. Su objetivo es ofrecer el lenguaje de programación más productivo para el desarrollo multiplataforma, junto con una plataforma de ejecución flexible para frameworks de aplicaciones [5].

Flutter es un marco de código abierto desarrollado y compatible con Google. Los desarrolladores de front-end y pila completa utilizan Flutter para crear una interfaz de usuario (IU) de aplicación para varias plataformas con un único código base [1].

Las aplicaciones de Flutter pueden usar las bases de datos SQLite mediante el complemento *sqflite* disponible en pub.dev. Si está desarrollando una aplicación que necesita persistir y consultar grandes cantidades de datos en el dispositivo local, considere usar una base de datos en lugar de un archivo local o un almacén de clave-valor. En general, las bases de datos ofrecen inserciones, actualizaciones y consultas más rápidas en comparación con otras soluciones de persistencia local [7].

QR_flutter es una biblioteca de Flutter para la representación simple y rápida de códigos QR a través de un widget o un painter personalizado [13].

Share_plus es un complemento de Flutter para compartir contenido desde tu aplicación de Flutter mediante el cuadro de diálogo para compartir de la plataforma [8].

4. Implementación y pruebas. En la fase de implementación se toman los requisitos y los productos de la fase de diseño y se implementan mediante las tecnologías apropiadas. En el caso de las pruebas de validación, es en esta fase cuando se completan y automatizan los casos de prueba en preparación de las pruebas de validación [5]. La validación de software se enfoca en confirmar que el software cumple con las necesidades y expectativas del usuario final, es decir, que realiza las funciones para las cuales fue diseñado de manera precisa y completa. Implica asegurarse de que el software satisface los requerimientos funcionales y no funcionales especificados durante la fase de planificación y diseño del proyecto [12]. Aquí se realizaron pruebas de validación con un grupo selectivo de ejidatarios evaluando el tiempo de escaneo y comparando con el tiempo manual y la generación de reportes.

La elección de ésta metodología permitió la implementación ágil del software y centrada en las necesidades primordiales del usuario, garantizando que la aplicación sea funcional, accesible y adaptada al entorno rural en el que se desarrolló.

II. RESULTADOS Y DISCUSIÓN

Siguiendo la metodología que se planteó, del levantamiento de los requerimientos se obtuvieron los siguientes elementos importantes a partir de entrevistas con representantes de la

comisaría ejidal: las necesidades específicas de la aplicación, el detalle del proceso actual de pase de lista y los problemas principales identificados con el registro manual de asistencia. Algunas de las necesidades identificadas son las siguientes:

Registro centralizado: Poder tomar la lista de asistencia a los ejidatarios desde un solo dispositivo (como una tableta que se tiene destinada para la comisaría ejidal o teléfono móvil de un representante ejidal) en el lugar de la reunión, sin necesidad de papeles.

Uso de código QR: Para el pase de lista se deberá hacer con un código QR, mismo que debe generar la aplicación al momento de registrar un ejidatario.

Registro de fecha y hora: se deberá cumplir con esta necesidad para evitar fraudes sobre la posible asistencia de un ejidatario; estos datos deberán colocarse en el reporte generado.

Generación de reportes automáticos: Obtener de forma inmediata reportes de asistencia o inasistencia por eventos específicos (como la asamblea de enero o la de diciembre), sin tener que contar manualmente cada asistente, ya que se necesita más del 50% de los ejidatarios para poder iniciar asambleas u otras actividades.

El proceso actual de pase de lista inicia con una convocatoria dirigida a todos los ejidatarios para las asambleas previamente establecidas. A la entrada del salón ejidal, un representante del comisariado (por ejemplo, la secretaría) recibe a los asistentes con una lista impresa de los 401 ejidatarios. El procedimiento es el siguiente:

Verificación: A medida que cada ejidatario o su representante (como un hijo o esposa) llega, la secretaría busca su nombre manualmente en la lista.

Registro: Una vez localizado, marca al asistente como "presente".

Conteo y quórum: Una vez finalizado el registro, realiza un conteo visual de las marcas para determinar el número total de asistentes. Si se supera el 50% de los ejidatarios, se declara que hay quórum y la asamblea u otro evento puede proceder.

Gestión de ausencias: Posteriormente, identifica y filtra a los ejidatarios que no asistieron para aplicarles el descuento correspondiente por inasistencia.

Este proceso lleva aproximadamente entre una hora y una hora y media.

Dentro de los principales problemas identificados se encuentran:

Pérdida de tiempo: buscar cada nombre uno por uno en una lista impresa con varias hojas y 401 ejidatarios es un proceso extremadamente lento como se dijo anteriormente.

Dificultad para el conteo: contar manualmente cientos de marcas es tedioso y propenso a errores. Si el conteo es erróneo, se puede invalidar el quórum y, por lo tanto, las decisiones tomadas.

Falta de transparencia y disputas: puede haber discusiones sobre si alguien estuvo o no, o si el conteo final es correcto.

Problemas con el archivo: buscar un registro de asistencia de una asamblea pasada específica es una tarea manual que requiere buscar en archivos físicos y ese proceso puede ser muy tardado.

Para la segunda etapa, diseño de la aplicación, tres de las historias de usuario que están documentadas y que servirán como base para definir las funcionalidades clave de la aplicación son:

HU1: Registro de ejidatarios con QR

Como secretario del ejido

Quiero poder registrar nuevos ejidatarios generando automáticamente un código QR único

Para tener identificados digitalmente a todos los ejidatarios y agilizar el proceso de asistencia

Criterios de aceptación:

- Debo poder ingresar nombres, apellido paterno y materno
- La aplicación debe generar automáticamente un código QR al completar los campos
- Debo poder compartir el QR vía WhatsApp
- El QR debe guardarse en la base de datos asociado al ejidatario

HU2: Pase de lista por escaneo de código QR

Como encargado de pase de lista

Quiero escanear códigos QR de los ejidatarios usando la cámara del dispositivo

Para registrar su asistencia de forma rápida y precisa en eventos específicos

Criterios de aceptación:

- Debo seleccionar un evento antes de escanear
- La cámara debe detectar y procesar QR automáticamente
- La aplicación debe mostrar confirmación inmediata de asistencia registrada
- Debe evitar registros duplicados para el mismo evento
- Debo ver un contador en tiempo real de asistencias

HU3 – Generación de reportes de asistencia

Como secretario del ejido

Quiero generar reportes de asistencia por evento

Para saber quiénes asistieron, quiénes no y aplicar descuentos correspondientes

Criterios de Aceptación:

- Debo poder seleccionar un evento específico

- La aplicación debe mostrar listas separadas de asistentes y no asistentes
- Debo poder exportar el reporte a Excel
- El reporte exportado debe incluir nombres completos y estado de asistencia

Una vez definidas las historias de usuario, se procedió al diseño de los wireframes realizados en Figma, esto nos ayuda a visualizar la estructura y el flujo de la aplicación. A continuación, se presentan cuatro pantallas importantes:

Pantalla de registro de ejidatarios: en la figura 1 se muestra la pantalla que permite al usuario agregar y listar ejidatarios, para agregar un ejidatario deberá registrar el nombre, apellido paterno y apellido materno, además en la parte inferior deberá mostrarse la sección activa, que es la de ejidatarios.



Figura 1: Pantalla de baja fidelidad del registro de ejidatarios

Pantalla de generación de código QR: Al completar los tres campos obligatorios de información del ejidatario, la aplicación habilitará la generación del código QR correspondiente y mostrará las opciones para compartirlo vía WhatsApp y guardar el registro en la base de datos. La figura 2 muestra cómo debe quedar después de completar la información requerida.



Figura 2: Pantalla de baja fidelidad con la generación del código QR

Pantalla de registro de evento

Una de las pantallas más sencillas en cuanto a diseño, pero fundamental para el funcionamiento de la aplicación, es la destinada a la creación de eventos. En esta interfaz (ver figura 3), el usuario únicamente debe ingresar el nombre descriptivo del evento para posteriormente guardarlo en la base de datos. La importancia de este proceso radica en que cada evento guardado se convierte en un registro indispensable para el módulo de pase de lista, ya que deberá ser seleccionado previamente antes de iniciar el pase de asistencia.

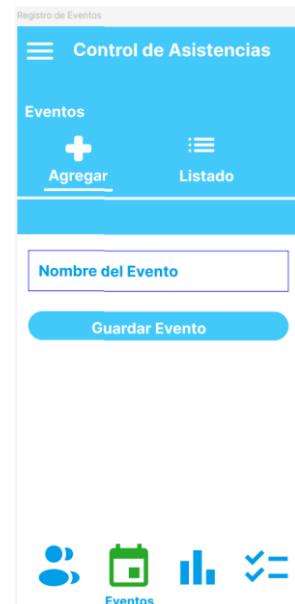


Figura 3: Pantalla de baja fidelidad para agregar eventos

Pantalla de pase de lista

Para la realización del pase de lista deberá seleccionarse el evento previamente creado, posteriormente se deberá pulsar en el botón “Iniciar escaneo”, si el usuario lo decide, puede cambiar entre la cámara frontal y posterior de su dispositivo si éste cuenta con ambas cámaras; a continuación deberá leer el código QR y hacer el pase de lista, en esa interfaz deberá marcar si el usuario hizo el pase de lista de manera correcta, si no existe el ejidatario asociado al código QR o bien, si el ejidatario ya realizó su pase de lista. Adicionalmente, la interfaz mostrará un contador actualizado con el total de asistencias registradas durante la sesión, brindando así retroalimentación inmediata y facilitando el control del quórum. Lo descrito previamente se observa en la figura 4.

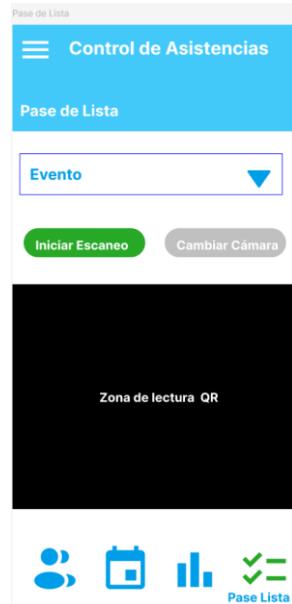


Figura 4: Pantalla de baja fidelidad para el pase de lista

Después de la fase de diseño, como se mencionó en el capítulo de la metodología, se continuó con la elección de las herramientas necesarias para el desarrollo de la aplicación, se eligió por Visual Studio Code (VSC) en lugar de Android Studio por su mayor ligereza y agilidad; VSC, con extensiones especializadas para Flutter y Dart, ofrece todas las funciones necesarias para el desarrollo (como autocompletado, depuración y visualización inmediata de cambios) pero con un consumo mínimo de recursos, lo que acelera la productividad.

Finalmente, la etapa de implementación involucró la codificación de todas las funcionalidades definidas utilizando las herramientas de la etapa anterior. Durante este proceso se desarrollaron los módulos principales: gestión de ejidatarios con generación de códigos QR, eventos, registro de asistencias mediante escaneo QR y generación de reportes exportables mediante WhatsApp. Cada módulo fue implementado siguiendo las especificaciones de las historias de usuario validadas en etapas anteriores, asegurando que la aplicación cumpliera con los objetivos de digitalización del pase de lista tradicional.

La aplicación se instaló en un Smartphone y dio como resultado las figuras que se presentarán a continuación. La aplicación cuenta con un inicio de sesión, el cual después de haber ingresado los datos correctamente, redirigirá a la pantalla para poder agregar un ejidatario, el cuál como se mencionó en la etapa de diseño, después de llenar sus datos completos, se podrá generar el código QR y compartirlo mediante WhatsApp si fuese necesario. La figura 5 muestra el resultado de generar el QR para poder compartirlo y guardar el ejidatario.



Figura 5: Pantalla de generación del código QR.

En la sección de “Listado”, el usuario puede visualizar todos los ejidatarios registrados en la aplicación. Como se observa en la Figura 6, esta pantalla muestra el número total de ejidatarios existentes y el nombre completo de cada uno de ellos. Dado que la cantidad de registros puede ser considerable (más de 400 ejidatarios), se implementó un mecanismo de paginación que permite navegar de manera ordenada y eficiente entre los registros, mejorando así la experiencia de usuario y el rendimiento de la aplicación.

Adicionalmente, se incluyó una herramienta de búsqueda que filtra en tiempo real a los ejidatarios según el nombre o apellidos ingresados, mostrando únicamente aquellos que coincidan con el criterio establecido.

Cada ejidatario en la lista cuenta con opciones para editar o eliminar su información. Al seleccionar la eliminación, la aplicación solicita una confirmación mediante un cuadro de diálogo para prevenir borrados accidentales. Por otro lado, al editar un ejidatario, se volverá a generar automáticamente su código QR, el cual puede ser compartido de forma inmediata a través de WhatsApp, manteniendo siempre actualizada la identificación digital del ejidatario.



Figura 6: Pantalla de listado de usuarios.

Una vez registrado un evento en la aplicación, es posible realizar el pase de lista. Para ello, la interfaz solicita seleccionar un evento de una lista desplegable y luego pulsar la opción "Iniciar escaneo". Esta acción activará la cámara del dispositivo, mostrando un área de escaneo para el código QR. Al escanear el código de un ejidatario, la aplicación mostrará uno de los siguientes mensajes según corresponda: a) Si el usuario existe y es la primera vez que registra asistencia: "Asistencia registrada", b) Si la asistencia del usuario ya había sido registrada: "Ya se registró su asistencia" y c) Si el ejidatario no existe en la base de datos o el código QR no es válido: "No existe en la base de datos". Adicionalmente, la interfaz incluye una etiqueta que indica el total de asistencias registradas para el evento, lo que permite un control inmediato de los asistentes. Para finalizar el proceso, basta con seleccionar el botón "Detener". La Figura 7 ilustra el pase de lista realizado a un ejidatario.



Figura 7: Pantalla del pase de lista.

El usuario de la aplicación al finalizar el pase de lista necesita ver el reporte o exportar a un archivo de Excel los ejidatarios que asistieron o no a dicho evento, para ello, en la figura 8 se muestra la interfaz que permite ver el reporte de asistentes, solo selecciona el evento que requiera y después en generar reporte, en esa interfaz le mostrará dos bloques, el primero es para ver los ejidatarios que asistieron al evento y el otro bloque es para visualizar los ejidatarios que no asistieron. En ambos casos muestra el total de ejidatarios por cada bloque.



Figura 8: Visualización de reporte

Para exportar el reporte de asistencias, el usuario debe seleccionar el botón "Exportar" ubicado en la pestaña derecha de la sección de Reportes (Figura 8) y elegir el evento deseado. La aplicación genera automáticamente un archivo en formato Excel y abre el selector de aplicaciones del dispositivo móvil, permitiendo compartirlo a través de WhatsApp, Mensajes, Correo electrónico, Telegram o cualquier otra aplicación disponible en el smartphone o tableta.

El reporte generado incluye: a) Nombre del evento y fecha de generación, b) dos secciones claramente diferenciadas para los asistentes y no asistentes, c) conteo total de ejidatarios en cada sección, d) listado detallado con: número de ejidatario, nombre completo, estado de asistencia (Asistió/No asistió) y fecha de asistencia (para los no asistentes, se indica "N/A" en la fecha, lo que permite el manejo de valores nulos en la base de datos) y e) ordenamiento alfabético por apellido paterno.

La Figura 9 muestra un ejemplo del archivo Excel generado, donde puede apreciarse la estructura organizada del reporte, diseñada específicamente para facilitar su consulta y análisis en dispositivos móviles.

Reporte de Asistencias - Asamblea SEPTIEMBRE 2025					
Generado el: 2025-10-07 13:05:00.687165					
ID	Nombres	Apellido Paterno	Apellido Materno	Asistencia	Fecha de Asistencia
404	Ivan Rafael	Sánchez	Juárez	ASISTIÓ	2025-10-07T13:01:36.711281
NO ASISTENTES (403)					
1	NOMBRE 1	APPELLIDO PATERNO 1	APPELLIDO MATERNO 1	NO ASISTIÓ	N/A
2	NOMBRE 2	APPELLIDO PATERNO 2	APPELLIDO MATERNO 2	NO ASISTIÓ	N/A
3	NOMBRE 3	APPELLIDO PATERNO 3	APPELLIDO MATERNO 3	NO ASISTIÓ	N/A
4	NOMBRE 4	APPELLIDO PATERNO 4	APPELLIDO MATERNO 4	NO ASISTIÓ	N/A
5	NOMBRE 5	APPELLIDO PATERNO 5	APPELLIDO MATERNO 5	NO ASISTIÓ	N/A
6	NOMBRE 6	APPELLIDO PATERNO 6	APPELLIDO MATERNO 6	NO ASISTIÓ	N/A
7	NOMBRE 7	APPELLIDO PATERNO 7	APPELLIDO MATERNO 7	NO ASISTIÓ	N/A
8	NOMBRE 8	APPELLIDO PATERNO 8	APPELLIDO MATERNO 8	NO ASISTIÓ	N/A
9	NOMBRE 9	APPELLIDO PATERNO 9	APPELLIDO MATERNO 9	NO ASISTIÓ	N/A
10	NOMBRE 10	APPELLIDO PATERNO 10	APPELLIDO MATERNO 10	NO ASISTIÓ	N/A

Figura 9: Reporte generado en formato Excel.

El proyecto fue organizado mediante una arquitectura de directorios modular que facilitó el desarrollo y mantenimiento del código. Se implementaron tres capas principales:

/models: contiene las entidades de datos (ejidatario, evento, etc)

/screens: aloja los widgets de interfaz de usuario

/services: gestiona la lógica de negocio y conexiones externas

La configuración de dependencias se centralizó en el archivo **pubspec.yaml**, archivo que define todas las librerías externas necesarias para el funcionamiento de la aplicación, incluyendo entre otras:

sqflite para persistencia local

qr_flutter para generación de códigos QR

excel y **share_plus** para exportación y compartición

Para la generación del ejecutable final se utilizó el comando de build de producción:

flutter build apk --release

Este comando optimiza el código, elimina recursos innecesarios y genera un apk firmado listo para distribución.

La Figura 10 muestra la estructura completa del proyecto y el detalle de dependencias en el archivo de configuración.

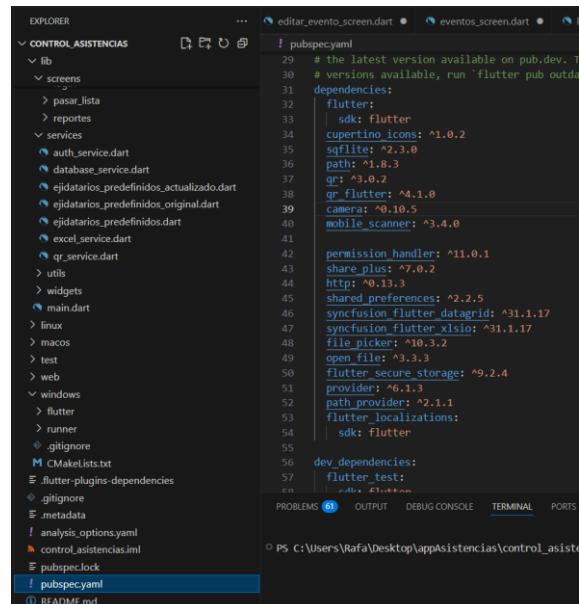


Figura 10: Estructura y dependencias del proyecto

CONCLUSIÓN

El diseño e implementación de la aplicación móvil para el registro de asistencia mediante códigos QR demostró ser una solución tecnológica altamente efectiva para optimizar los procesos administrativos en el núcleo ejidal de San Rafael Ixtapalucan. La sustitución del pase de lista manual, que consumía entre 90 y 100 minutos aproximadamente, por un sistema digital de escaneo redujo el tiempo de registro a apenas 30 minutos.

La aplicación de la fórmula de porcentaje de mejora:

$$[(\text{Tiempo Inicial} - \text{Tiempo Final}) / \text{Tiempo Inicial}] \times 100$$

$$[(100 - 30)/100] * 100 = 70$$

Permitió cuantificar este avance, arrojando una reducción del 70% en el tiempo requerido para el pase de lista. Esto no solo se traduce en una ganancia significativa de eficiencia, sino que también minimiza los errores humanos, agiliza la verificación del quórum y fortalece la transparencia en los procesos de asistencia al no poder registrar asistencias duplicadas o intentar pasar asistencia con otro código QR que no sea el proporcionado por la comisaría ejidal.

Además, la aplicación puede crear reportes automáticamente en Excel y mandarlos de inmediato por WhatsApp, lo que facilita mucho el trabajo de quienes organizan las asambleas. Usar esta herramienta, hecha con Flutter y con un método de desarrollo ágil, es un gran paso para modernizar la gestión en el ejido, contribuyendo a tener procesos más claros, eficientes y alineados con los objetivos de desarrollo sostenible al promover instituciones más eficaces y transparentes.

RECONOCIMIENTO

Agradecemos al Tecnológico Nacional de México / ITS de San Martín Texmelucan por el invaluable apoyo académico, técnico y logístico brindado durante el desarrollo de este proyecto, de igual manera se agradece a todo el núcleo ejidal de la comisaría de San Rafael Ixtapalucan por confiar ampliamente en nuestro trabajo y permitirnos colaborar con esta solución tecnológica que permite fortalecer los vínculos entre el Instituto Tecnológico y la comisaría.

REFERENCIAS

[1] Amazon Web Services. (2024). ¿Qué es Flutter?
<https://aws.amazon.com/es/what-is/flutter/>

[2] Atlassian. (s.f.). Qué es Scrum y cómo empezar.
<https://www.atlassian.com/es/agile/scrum>

[3] Cámara de Diputados del H. Congreso de la Unión. (01 de enero de 2024). Ley Agraria. <https://www.diputados.gob.mx/LeyesBiblio/pdf/LAgra.pdf>

[4] DbaExperts (27 de diciembre de 2022). Levantamiento de Requerimientos <https://dbaexperts.tech/database/levantamiento-de-requerimientos/>

[5] Dart. (7 de agosto de 2025). Descripción general de Dart.
<https://dart.dev/overview>

[6] Figma. (s.f.) ¿Qué es Figma? <https://help.figma.com/hc/es-419/articles/14563969806359--Qu%C3%A9-es-Figma>

[7] Flutter. (04 de septiembre de 2025). Persist data with SQLite.
<https://docs.flutter.dev/cookbook/persistence/sqlite>

[8] fluttercommunity.dev (11 de septiembre de 2025). share_plus: ^12.0.0.
https://pub.dev/packages/share_plus

[9] International Business Machines. (04 de marzo de 2021). Diseño de sistemas y aplicaciones de software utilizando modelos.
<https://www.ibm.com/docs/es/rational-software-arch/9.6.1?topic=modeling-designing-systems-software-applications-by-using-models>

[10] Naciones Unidas. (s.f.). Objetivos de Desarrollo Sostenible.
<https://www.un.org/sustainabledevelopment/es/peace-justice/>

[11] Proyectos Ágiles. (s.f.). Qué es SCRUM. <https://proyectosagiles.org/que-es-scrum/>

[12] Software Quality Systems. (2025). ¿Qué es la Verificación & Validación?
<https://www.sqs.es/verificacion-y-validacion-de-software/>

[13] Theyakka. (14 de mayo de 2023). Qr_flutter.
https://pub.dev/packages/qr_flutter