

Desarrollo e implementación de una aplicación móvil para facilitar el control de asistencia a clase

José Carlos Avalos-Ruiz, Axel Issai Alemán-Delgado, Juan de Dios Nava-Gallardo,
José Manuel Rodríguez García, Orlando Samuel Martínez Dorantes, Marco Aurelio Nuño-Maganda*,
Yahir Hernández-Mier and Said Polanco-Martagón

*Laboratorio de Sistemas Inteligentes
Universidad Politécnica de Victoria
Ciudad Victoria, México, 87138,
Email: mnunom@upv.edu.mx*

Resumen—En todos los niveles educativos hay una necesidad de los profesores para llevar el control de la asistencia de sus alumnos, debido a que en varios niveles este control todavía se hace utilizando hojas de papel y bolígrafo. En este artículo se propone una aplicación móvil amigable con interfaz que le permite al profesor hacer el pase de lista de manera manual mucho más rápida, administrar grupos y alumnos, generar informes de asistencia con solo unos cuantos clics. La aplicación estará dividida en 3 secciones: *Dashboard*, Grupos y Estudiantes. Así mismo, se muestran algunos gráficos que forman parte de la maquetación de la aplicación desarrollada, los cuales fueron realizados con el fin de establecer una planeación y asignar tareas. La aplicación cuenta con un módulo para obtener estadísticas de porcentaje de asistencia y en algún momento dado determinar que estudiantes están en riesgo de reprobar una asignatura por exceso de inasistencias.

Index Terms—Android, herramientas docentes

I. INTRODUCCIÓN

Los desarrolladores de aplicaciones de escritorio trabajan en casi cualquier lenguaje y la aplicación final puede ejecutarse en cualquier sistema operativo “de escritorio”. Por otro lado, los desarrolladores de Android de aplicaciones móviles trabajan con aplicaciones que serán ejecutadas en un dispositivo inteligente. El sistema operativo móvil más extendido es Android, el cual es un sistema operativo basado en C++ que se ejecuta sobre el núcleo Linux. Las aplicaciones de Android se desarrollan con Java y pueden portarse con bastante facilidad a la nueva plataforma [1].

Recientemente, la programación de aplicaciones móviles representa la posibilidad de incursionar en nuevas áreas de oportunidad en comparación con desarrollar aplicaciones de escritorio o Web [2]. Recientemente, muchas de las tareas que siempre se habían hecho utilizando una computadora, han sido migradas hacia aplicaciones ejecutadas en teléfonos móviles. A lo largo de los últimos años, el avance de la tecnología para dispositivos móviles ha sido tal, que un gran número de acciones han sido absorbidas completamente por el uso de teléfonos inteligentes. El lenguaje de programación Kotlin se ha popularizado debido a que es la principal herramienta para programación de aplicaciones móviles [3]. Tal es el caso del

registro de asistencia escolar, ya que anteriormente se solía pasar lista mediante hojas papel y esto era poco práctico. El presente reporte muestra la elaboración de una aplicación móvil para el registro de asistencia escolar, el cual tiene como objetivo optimizar los tiempos en el pase de lista del alumnado, así como brindar de una mejor administración de los datos de cada grupo y, por ende, de cada alumno. Contar con una aplicación móvil que realice esta función puede llegar a ser muy útil, ya que también permite una mayor velocidad al momento de pasar asistencia y una mayor seguridad para respaldar la información. Este artículo está organizado de la siguiente forma. En la sección II, se revisaron las aplicaciones mejor calificadas con funcionalidad deseada en la tienda de aplicaciones de Google. En la sección III, se reporta el diseño de la aplicación y de la base de datos. En la sección IV, se reporta la implementación de dicha aplicación en el dispositivo móvil. Finalmente en la sección V, se establecen las conclusiones y futuras direcciones.

II. REVISIÓN DE APLICACIONES SIMILARES EN GOOGLE PLAY

En esta sección se analizan tres aplicaciones similares a la desarrollada. Se muestran una pantalla de cada aplicación, así como sus ventajas y desventajas.

La primera aplicación revisada fue ‘*Pasalista*’ [4], cuya pantalla es mostrada en la figura 1a. Después de haber revisado su desempeño, es posible concluir las siguientes similitudes:

- Permite manejar diversos grupos de alumnos.
- Tiene la opción de seleccionar un alumno aleatoriamente.
- Es posible importar un archivo con la lista de los alumnos.

Sus principales ventajas son:

- Contiene un apartado para generar/exportar reportes en formato PDF o CSV.
- Permite asignarle a un alumno asistencia, falta, retardo, falta con justificante o dejarlo en blanco.
- Es posible añadir observaciones del comportamiento de los alumnos mediante iconos cuando se toma asistencia.
- Tiene un apartado muy completo para las evaluaciones, en el cual se da una calificación máxima y una mínima,

*Autor de correspondencia

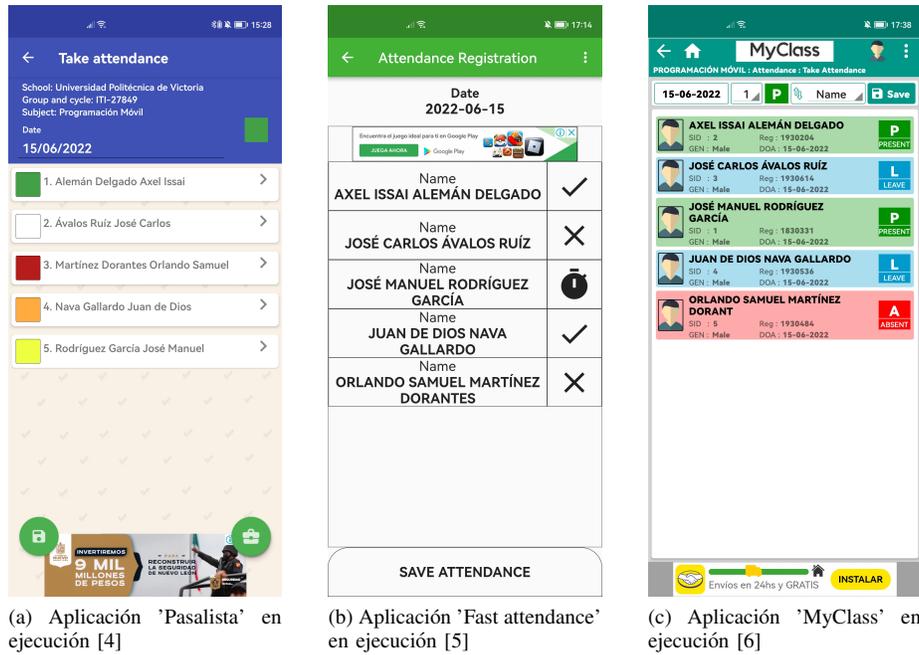


Figura 1: Capturas de pantalla de las aplicaciones de pase de lista revisadas

el total de tareas y ponderación para cada una de ellas, lo mismo para los exámenes y permite añadir más rasgos a calificar.

Y sus principales desventajas son:

- Contiene publicidad y algunas limitantes en la versión gratis, como el hecho de solamente tener dos grupos registrados, no poder editar las notas del comportamiento de los alumnos una vez guardado.
- No existe una opción para exportar grupos.
- Solo se pueden importar archivos CSV y TXT.

La segunda aplicación revisada fue 'Fast attendance' [5], cuya pantalla es mostrada en la figura 1b. Después de haber revisado su desempeño, es posible concluir las siguientes similitudes:

- Permite gestionar diversos grupos de alumnos.

Sus principales ventajas son:

- Cuenta con la opción de generar un reporte con las asistencias, faltas y retardos de los alumnos.
- Funciona tanto para estudiantes como para empleados de una empresa.
- No existe un límite para crear grupos.

Y sus principales desventajas son:

- No permite importar ni exportar grupos de alumnos.
- La opción de generar aleatorio no está presente.

La tercera aplicación revisada fue 'MyClass' [6], cuya pantalla es mostrada en la figura 1c. Después de haber revisado su desempeño, es posible concluir las siguientes similitudes:

- Cuenta con la opción de importar y exportar los grupos de los alumnos.
- Es posible gestionar diversos grupos de alumnos.

Sus principales ventajas son:

- Tiene la opción de generar reportes sobre la asistencia de los alumnos.
- La app trabaja mediante perfiles, por lo que se puede cambiar el perfil para tener grupos distintos (todo de manera local).

Y sus principales desventajas son:

- No es posible generar un aleatorio entre los alumnos.
- Los perfiles solo se encuentran de manera local, por lo que, para trabajar con los mismos perfiles en otro dispositivo, es necesario exportar la base de datos para después importarla en el otro teléfono, afortunadamente la app cuenta con esa funcionalidad.

III. SISTEMA PROPUESTO

En un primer paso, basándonos en las características de las aplicaciones revisadas de Google Play y de los requerimientos de la aplicación, procedimos a hacer un diseño preliminar de las pantallas que debería llevar la aplicación. En esta sección reportamos solo las pantallas más importantes de la aplicación, los cuales se muestran en la figura 2.

Para el diseño de la interfaz de usuario, se integra un diseño minimalista y sobre todo que sea fácil de utilizar. La interfaz principal (*Dashboard*) cuenta con información general de cada grupo, mostrando un resumen de la asistencia grupal y una lista de los alumnos que pertenecen a ese grupo, sus asistencias y sus faltas. Este diseño se muestra en la figura 2a. Además, cuenta con dos acciones rápidas, una para comenzar con el pase de lista y otra para generar un aleatorio entre el alumnado del grupo que actualmente seleccionado. Para llevar a cabo

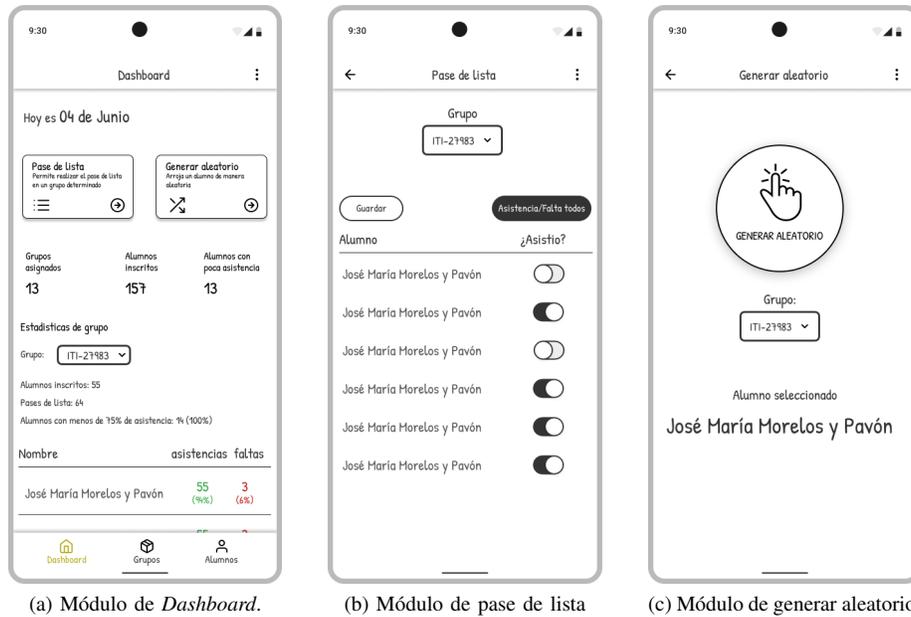


Figura 2: Wireframe de las pantallas principales de la aplicación.

estas funcionalidades, se utilizó un control de tipo *scrollview* [7].

El módulo de pase de lista cuenta con un *textView* y un *Spinner* que permite al usuario seleccionar el grupo al cual se pretende registrar la asistencia [8]. Este diseño se muestra en la figura 2b. Además, esta interfaz cuenta con dos botones principales, el primero le permite al usuario poder guardar el registro de asistencia y el segundo le permite poner asistencia/falta a todo el grupo. Adicionalmente, se muestran todos los alumnos en la parte inferior con su respectivo *switch* para el registro de la asistencia por alumno [9].

Otro módulo fue el de generar un estudiante de manera aleatoria, el cual permite al docente poder elegir a un alumno aleatoriamente para efectos de llevar el registro de la participación. Para esto se agregó un botón circular en la parte central con dicha funcionalidad de elegir aleatoriamente un alumno. Para proceder a generar el alumno aleatorio primero se debe elegir el grupo en el cual se desea realizar la carga aleatoria. Este diseño se muestra en la figura 2c.

En un segundo paso, llevamos a cabo el análisis de los datos para proponer la estructura de la base de datos. En la figura 3, se muestra el diagrama entidad relación que servirá para generar la base de datos necesaria para el almacenamiento de los datos recopilados por la aplicación.

IV. IMPLEMENTACIÓN

En las figuras 4 y 5, se muestran pantallas de los principales módulos desarrollados en la aplicación móvil.

IV-A. Módulo *dashboard*

En el presente módulo, el cual corresponde a la pestaña principal de la aplicación, siendo el *dashboard*, se muestra la fecha del día de hoy, un botón para realizar el pase de

lista y el botón para generar un alumno aleatoriamente de un grupo seleccionado. En la parte inferior se muestra el total de grupos asignados, la cantidad total de alumnos inscritos y la cantidad total de alumnos con una baja asistencia. A su vez, también se muestran estadísticas de un grupo en específico que se seleccione, señalando el número de alumnos inscritos en ese grupo, el total de pases de lista y la cantidad de alumnos con menos del porcentaje mínimo de asistencia, mostrando después un listado de todos los alumnos que corresponden a ese grupo. Esta pantalla se muestra en la figura 4a.

IV-B. Módulo *grupos*

En este módulo se muestra la información general de cada grupo que esté registrado en el sistema, así como la opción de mostrar la lista de grupos cargada previamente. Además, este módulo permite la creación de un nuevo grupo, así como visualizar la información general de dicho grupo. Esta pantalla se muestra en la figura 4b.

IV-B1. Lista de grupos: En esta sección de la aplicación se muestra una lista en general de los grupos de los cuales se tiene registro en la aplicación. Primeramente, se da la opción en la parte superior de importar un grupo o de crear uno nuevo. Enseguida, se muestra el total de grupos creados, un botón para exportar un grupo y después un listado de todos los grupos que se tienen asignados. Esta pantalla se muestra en la figura 4c.

IV-B2. Crear grupo: En esta sección de la aplicación permite al usuario poder crear un nuevo grupo vacío, así como importar un grupo con alumnos. Esta interfaz consta de un *EditText* en donde se debe ingresar el nombre de un grupo para posteriormente guardarlo.

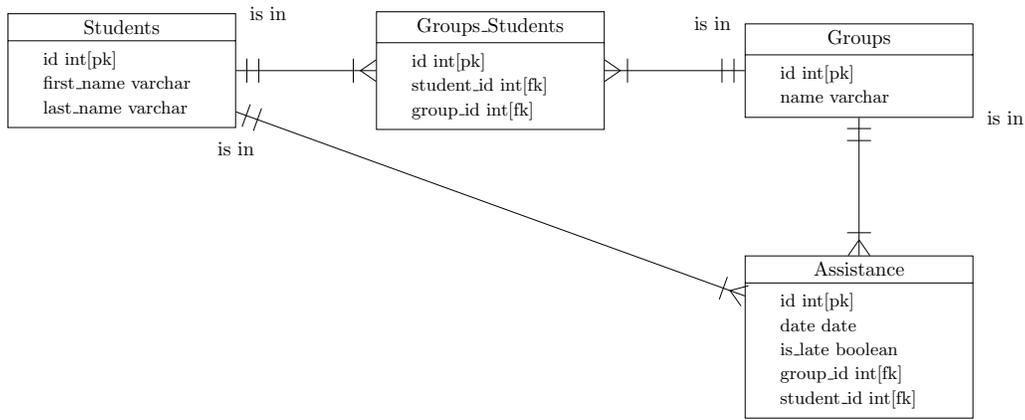
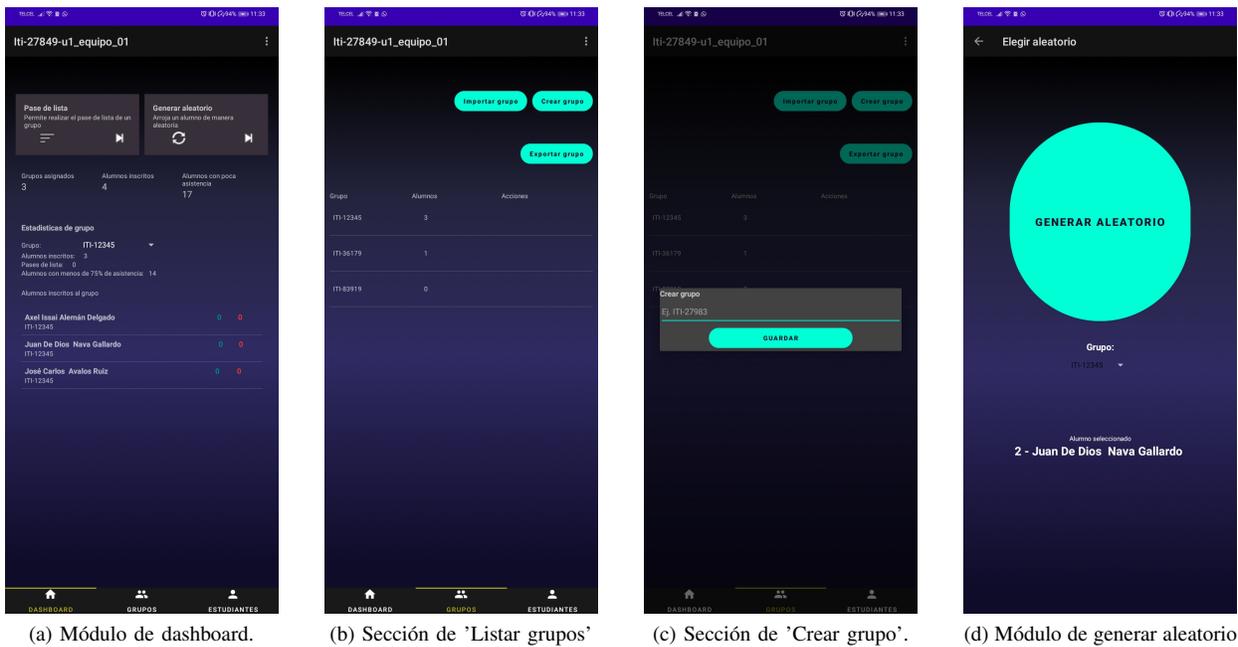


Figura 3: Diagrama de clases de la aplicación propuesta.



(a) Módulo de dashboard.

(b) Sección de 'Listar grupos'

(c) Sección de 'Crear grupo'.

(d) Módulo de generar aleatorio

Figura 4: Pantalla del módulo *dashboard* y para administrar y listar grupos.

IV-B3. Ver grupo: En este apartado se muestra en la parte superior el nombre del grupo seleccionado, así como un acceso directo a la acción de pasar lista. Enseguida se nos muestra información del grupo, tal como la cantidad de alumnos inscritos, el total de los pases de lista y la cantidad de alumnos con menor cantidad de asistencias que el porcentaje mínimo. En la parte inferior se muestran dos botones, uno para guardar el CSV del grupo y otro para agregar un alumno, para finalmente mostrar un listado de los alumnos inscritos en ese grupo. Esta pantalla se muestra en la figura 5a.

IV-C. Módulo alumnos

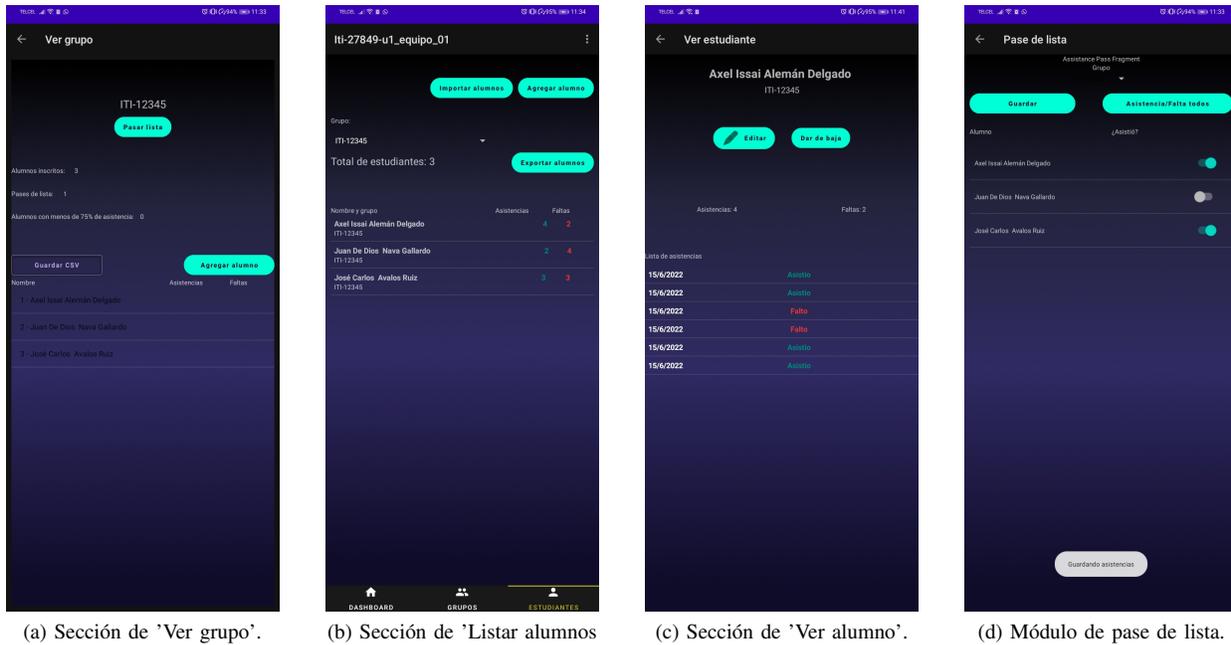
La parte de alumnos o estudiantes es la que muestra la información general de cada uno de los alumnos que estén registrados en el sistema, sin importar su grupo. Esta pantalla se muestra en la figura 5a.

IV-C1. Lista de alumnos: Esta lista muestra a todos los alumnos del sistema, así como su grupo y sus respectivas asistencias y faltas. Esta pantalla se muestra en la figura 5c.

IV-C2. Ver alumno: Esta sección le permite ver los datos de un alumno ya registrado en un grupo, así como editarlo o eliminarlo y ver sus asistencias y faltas.

IV-D. Módulo pase de lista

En este módulo se nos muestra un Spinner para seleccionar un grupo antes de realizar el pase de lista. Más abajo se nos muestran dos acciones para esta pantalla, el botón para guardar y el botón para colocarle asistencia o falta a todos los alumnos. En la parte inferior se nos mostrará un listado con todos los alumnos que forman parte de ese grupo, así como un botón Slider que nos permite asignarle a un alumno asistencia o no.



(a) Sección de 'Ver grupo'.

(b) Sección de 'Listar alumnos'.

(c) Sección de 'Ver alumno'.

(d) Módulo de pase de lista.

Figura 5: Pantallas para ver grupos, alumnos y agregar alumnos a la aplicación.

IV-E. Módulo generar aleatorio

Esta sección permite al usuario elegir a un determinado alumno aleatoriamente a partir de la selección de un grupo en específico. Al seleccionar el grupo deseado, se procede a presionar el botón central para poder seleccionar aleatoriamente a un alumno.

V. CONCLUSIÓN

Realizar esta aplicación fue una tarea un tanto complicada, sobre todo porque fue desarrollada en Kotlin y porque se presentaron algunos problemas durante el desarrollo. A pesar de que se contaba con un proyecto base desarrollado en Java, proporcionado por el profesor, se realizó investigación sobre cómo funcionaban las distintas aplicaciones que presentaban funciones similares y que ya se encontraban en el mercado, esto para estar al tanto de la forma en que presentaban estos elementos al usuario y cómo estaban desarrolladas, de esta manera se fue desarrollando la aplicación descrita a lo largo de este reporte, con el fin de contribuir con una aplicación más al mercado de Apps, pero tratando de presentar un producto que pueda distinguirse entre los demás.

REFERENCIAS

[1] D. Kannady, *Mobile App: Ultimate Guides For Beginners With Proven Designing Code, How To Create Your Own App: Designing App Quickly*. Independently Published, 2021. [Online]. Available: <https://books.google.com.mx/books?id=OsY3zgEACAAJ>

[2] A. Forrester, E. Boudjnah, A. Dumbravan, and J. Tigcal, *How to Build Android Apps with Kotlin: A hands-on guide to developing, testing, and publishing your first apps with Android*. Packt Publishing, 2021. [Online]. Available: <https://books.google.com.mx/books?id=drYgEAAAQBAJ>

[3] R. Whitaker, *Developing Inclusive Mobile Apps: Building Accessible Apps for iOS and Android*. Apress, 2020. [Online]. Available: <https://books.google.com.mx/books?id=2SjfdwAAQBAJ>

[4] "Pasalista," <https://play.google.com/store/apps/details?id=com.caracterizate.pasalista>, Consultado el 03-11-2022.

[5] "Fast attendance," <https://play.google.com/store/apps/details?id=cl.jandok.nelson.asistencia>, Consultado el 03-11-2022.

[6] "MyClass," <https://play.google.com/store/apps/details?id=com.vklmpandey.myclass>, Consultado el 03-11-2022.

[7] "ScrollView," <https://developer.android.com/reference/android/widget/ScrollView>, Consultado el 03-11-2022.

[8] "Spinner," <https://developer.android.com/reference/android/widget/Spinner>, Consultado el 03-11-2022.

[9] "assistance with switch," <https://developer.android.com/reference/android/widget/Switch>, Consultado el 03-11-2022.