

Universidad Nacional de Costa Rica  
Facultad de Ciencias Exactas y Naturales  
Escuela de Informática

Profesor Juan de Dios Murillo Morera  
Desarrollo de aplicaciones móviles.

**Reporte de laboratorio #3 y #4**

Estudiantes:  
Manuel Céspedes 1-1594-0141  
Cynthia Madrigal 1-1510-0465  
Reinaldo Stephens 4-0227-0666

Heredia, Costa Rica  
13-09-2016

## Esquema de la evolución e historia de los smartphones y tablets.

La idea inicial del teléfono inteligente era unir las funciones de un PDA (Personal Digital Assistant) con las de un teléfono para mayor comodidad y compactibilidad.



EN 1992, se desarrolló el IBM Simon, el cual tenía todas las funciones de un PDA de aquella época, con capacidades telefónicas, SMS y una pantalla táctil.

Pero pesaba cerca de 510 gramos y solo podía funcionar en 190 ciudades distribuidas en 15 estados de los Estados Unidos.



**Ericsson GS88:** El primer teléfono móvil en usar el término 'smartphone'. Poseía funciones de correo electrónico, navegación web, reloj mundial, un teclado QWERTY físico, modo avión, puerto infrarrojo, conexión a PC, entre otros.

Se comienza a hablar más del smartphone desde que se desarrolló el sistema operativo Windows Pocket PC (2000) y los teléfonos y dispositivos que llegaron al mercado con este sistema operativo como los de la marca HTC.



El evento que cambió la percepción de lo que era un smartphone fue el anuncio del iPhone y de iOS en 2007, revolucionando la industria de la telefonía móvil y de los smartphones. Este nuevo OS dio paso a Android OS de Google, actualmente el mayor competidor de iOS, lanzado unos meses después del anuncio del iPhone.

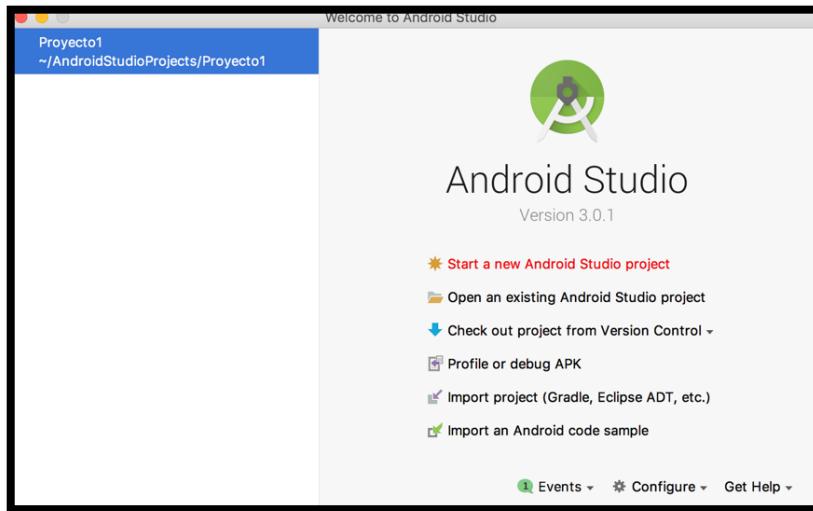
## Cuadro comparativo entre las diferentes tecnologías móviles iOS/Android/Window Phone.

iOS	Android	Window Phone
Ventajas		
<ul style="list-style-type: none"> <li>Elevada seguridad frente a las ciber amenazas.</li> <li>Mayor filtro y exclusividad dentro del mercado de aplicaciones.</li> <li>Interfaz intuitiva.</li> <li>El asistente personal SIRI se caracteriza por su gran capacidad para la resolución de tareas cotidianas con apoyo virtual.</li> <li>Integración entre software y hardware, es decir, compatibilidad con sistemas operativos anteriores.</li> <li>Elevada duración de la batería.</li> <li>Terminales con algunos de los mejores materiales del mercado.</li> <li>La sincronización entre los diversos dispositivos de Apple.</li> <li>Menos fragmentación.</li> </ul>	<ul style="list-style-type: none"> <li>Menos fallos.</li> <li>Personalización.</li> <li>Las herramientas de serie.</li> <li>Mejor hardware</li> <li>Libertad de tener un sistema operativo de código abierto</li> <li>Comunidad de usuarios es la más grande del mundo.</li> </ul>	<ul style="list-style-type: none"> <li>Sistemas de seguridad desde nuestro PC.</li> <li>Fácil de utilizar.</li> <li>Sincronización con Office.</li> <li>Botón directo de cámara.</li> <li>Windows Live ID.</li> <li>Interfaz sencilla.</li> <li>Cuenta con integración total con Windows. Por lo que es posible sincronizar el servicio de correo, contactos, OneDrive, y demás programas de Microsoft.</li> </ul>
Desventajas		
<ul style="list-style-type: none"> <li>Precios mucho más elevados que sus competidores.</li> <li>Menor personalización, variedad y especialización.</li> <li>Carece de la libertad de tener un sistema operativo de código abierto, es decir, limitación de hallarse obligados a utilizar únicamente las aplicaciones de la App Store.</li> <li>Apple no ha dado soporte a flash.</li> </ul>	<ul style="list-style-type: none"> <li>Mas fragmentado</li> <li>Las actualizaciones son muy pesadas.</li> <li>La multitarea no es eficiente.</li> <li>El número de recursos utilizados perjudica al rendimiento del dispositivo.</li> </ul>	<ul style="list-style-type: none"> <li>No es compatible con Adobe Flash Player.</li> <li>Tiene una simple interfaz gráfica.</li> <li>No dispone de un correo unificado.</li> <li>No es capaz de ejecutar muchas aplicaciones simultáneamente.</li> <li>No existe la deseada función de copiar/pegar.</li> <li>Pocas aplicaciones.</li> <li>Baja posibilidad de personalización.</li> </ul>

## **Principales elementos, herramientas configuraciones y perfiles del Android Studio.**

Para iniciar un proyecto nuevo en Android Studio, se debe iniciar el programa e inmediatamente se desplegará una pantalla como la que se aprecia en la figura 1. En esta se puede iniciar un proyecto nuevo o continuar uno ya existente.

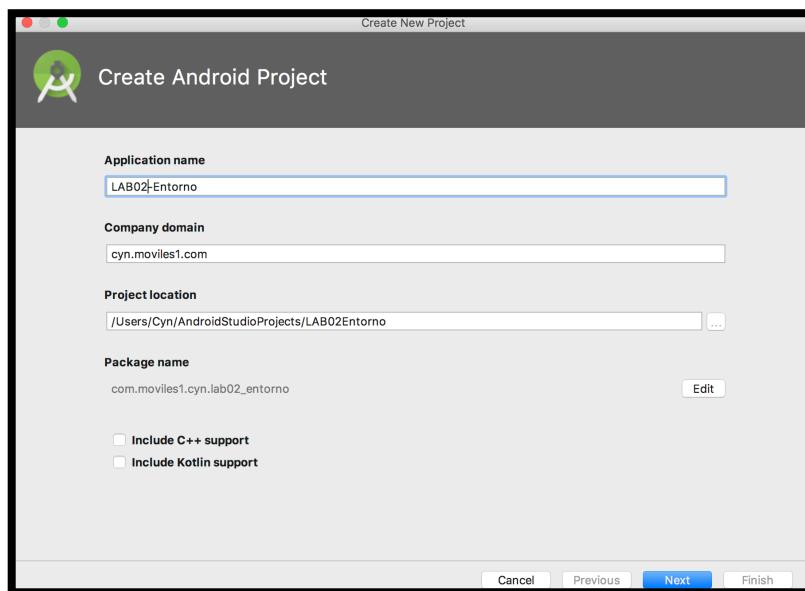
Figura 1: Pantalla de inicio Android Studio.



Tomado de Android Studio 3.0.1

Posteriormente, se elige un nombre apropiado y descriptivo para la aplicación y se establece la ruta donde se ubicará la aplicación.

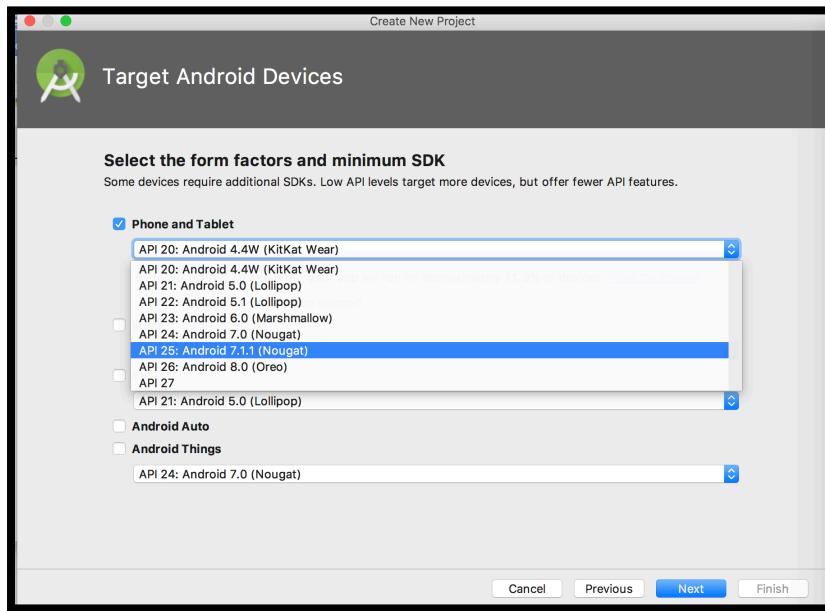
Figura 2: Detalles básicos de la aplicación.



Tomado de Android Studio 3.0.1

En la figura 3 se pueden observar las diferentes opciones que ofrece android, en relacion al dispositivo con el que se desea hacer pruebas de funcionalidad.

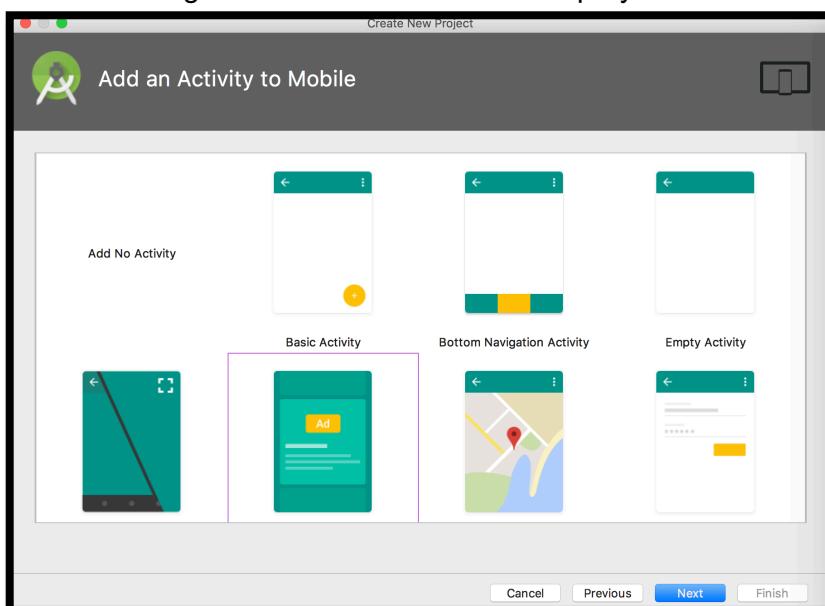
Figura 3: Configuracion de dispositivos.



Tomado de Android Studio 3.0.1

Si se selecciona el botón siguiente, se desplegara una ventana con todas las actividades, tambien conocidas como pantallas, predeterminadas que ofrece Android. En este momento del proceso de creacion, solo se puede seleccionar una opcion, pero una vez creado el cascaron, se pueden agregar mas actividades. (figura 4)

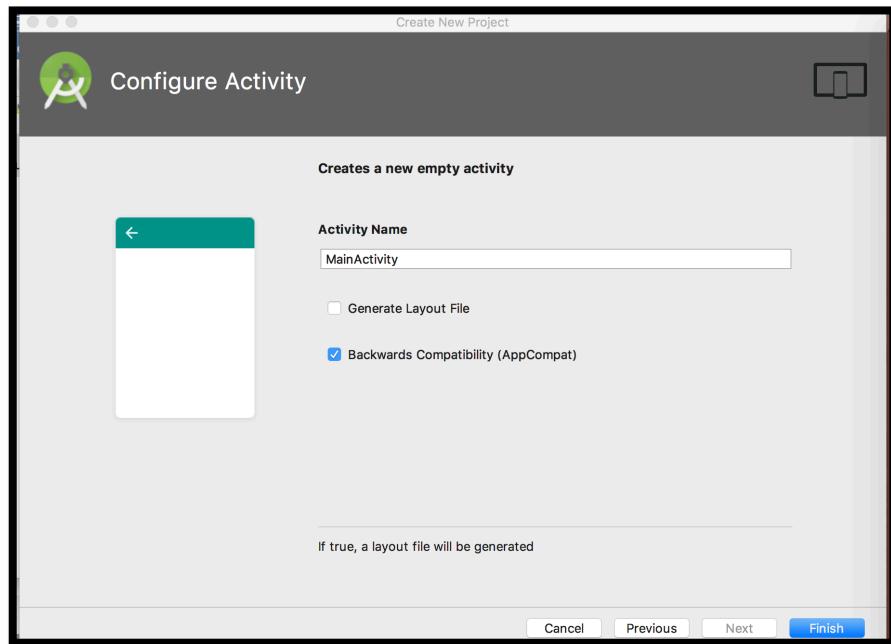
Figura 4: Actividades del nuevo proyecto.



Tomado de Android Studio 3.0.1

Una vez establecido lo anterior, se procede a asignarle un nombre a esta actividad como lo muestra la figura 5. Es importante anotar un nombre que tenga relación y fácil entendimiento de lo que trata la pantalla. Normalmente se utiliza “MainActivity” como la pantalla inicial y principal.

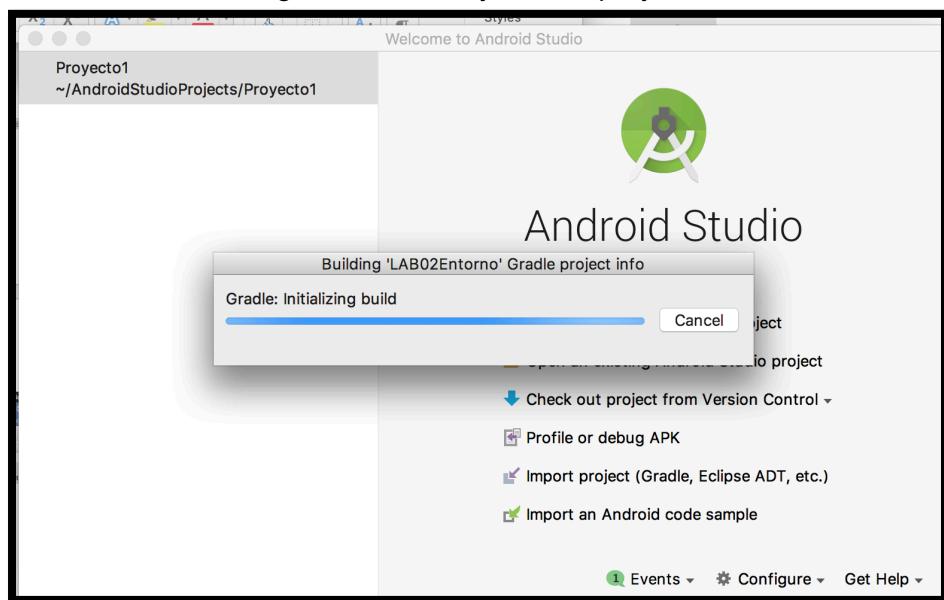
Figura 5: Configuración de la actividad.



Tomado de Android Studio 3.0.1

Si se está seguro de toda la configuración anterior, se puede seleccionar el botón “Finalizar” y se creará el sistema con los detalles acordados.(figura 6) En caso contrario, se puede devolver a los pasos anteriores con la opción “anterior”.

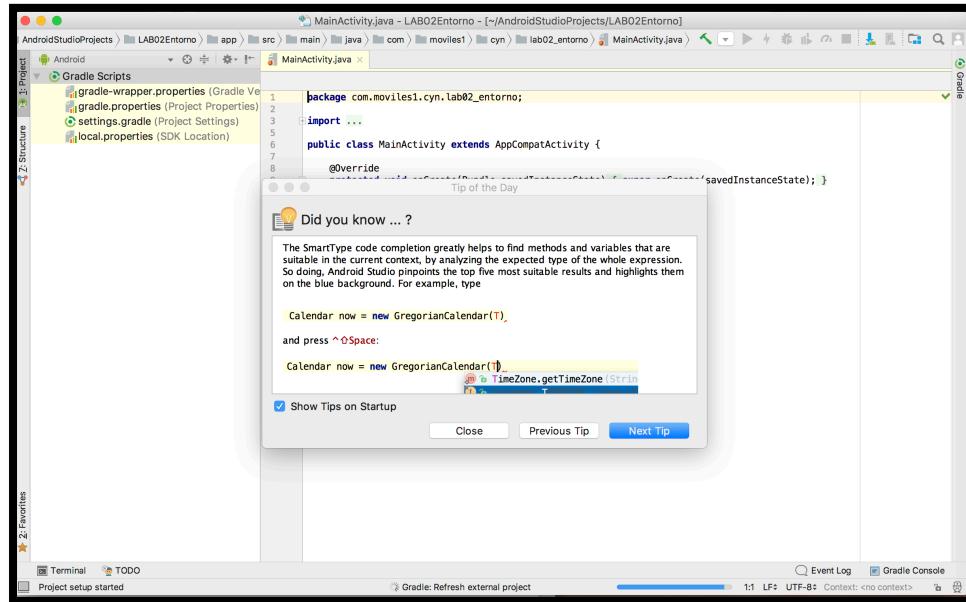
Figura 6: Construyendo el proyecto.



Tomado de Android Studio 3.0.1

Como se aprecia en la figura 7, esta plataforma mostrará una serie de consejos que pueden ayudar al desarrollador que empieza a conocer la misma. Si no se desea conocer estos consejos, se selecciona la casilla “no volver a mostrar” y posteriormente la opcion “continuar”.

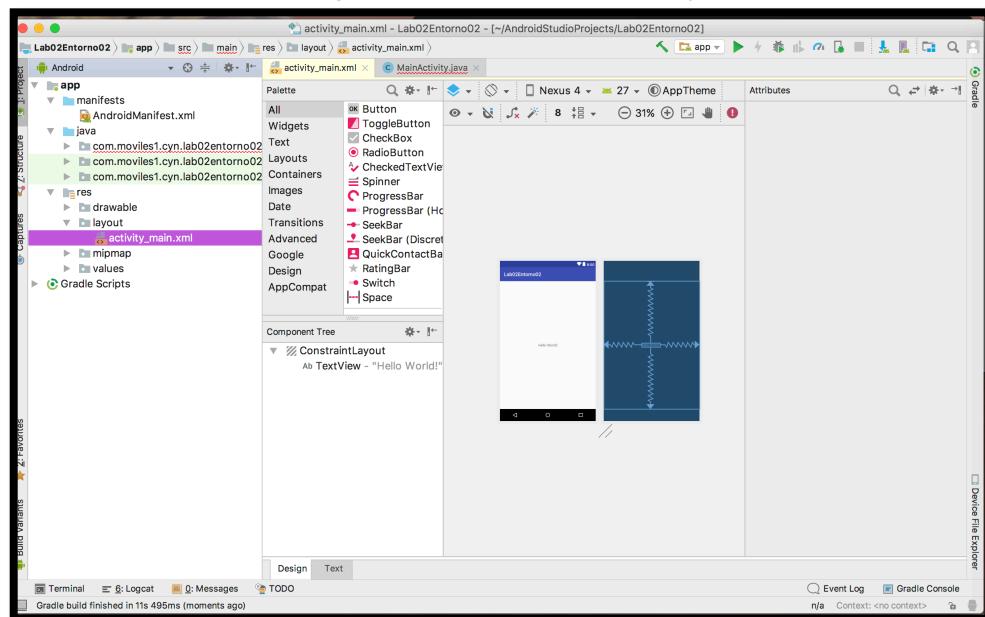
Figura 7: Consejos de Android.



Tomado de Android Studio 3.0.1

Luego de que se contruye el proyecto, se puede navegar dentro de la estructura del proyecto. Se puede iniciar por el diseño, de forma que se pueden utilizar y modificar los elementos que ofrece la paleta de Android y luego asignarle funcionalidad en las paginas de codigo correspondientes. (figura 8)

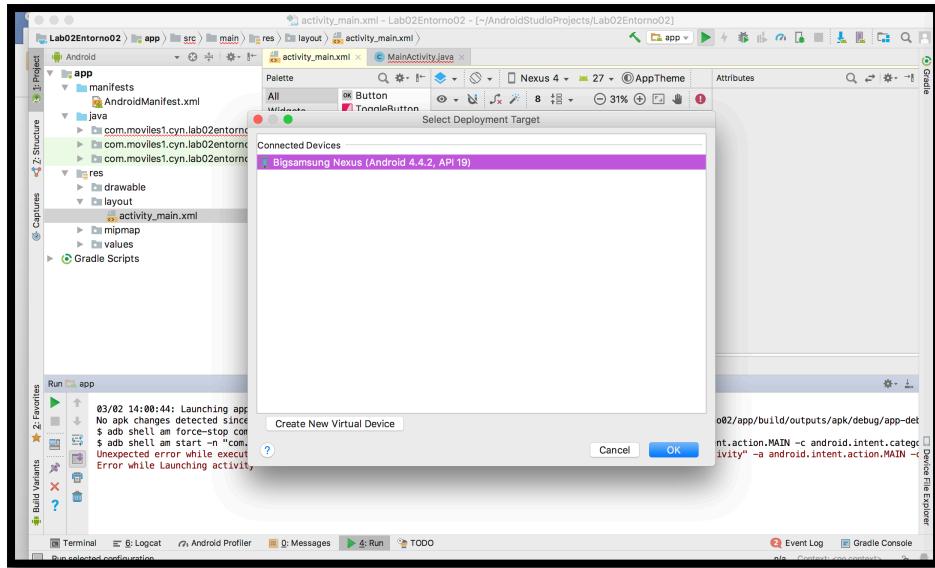
Figura 8: Diseño de un activity.



Tomado de Android Studio 3.0.1

Una vez que se desee probar o verificar el estado de la aplicación, se puede compilar la aplicación y seleccionar el medio por el cual se llevaran a cabo las pruebas. Como se aprecia en la figura 9, al seleccionar la opcion de compilar, se desplegará una pantalla con todos los dispositivos conectados disponibles. En esta pantalla, se selecciona si se desea probar con un simulador o con un aparato real.

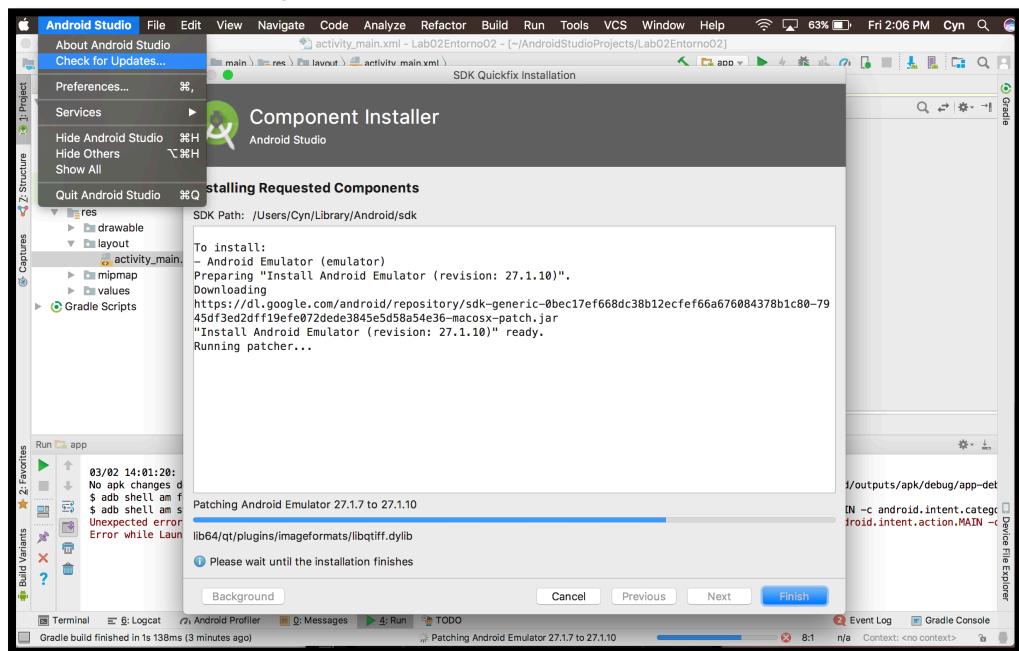
Figura 9: Pruebas de funcionalidad.



Tomado de Android Studio 3.0.1

En algunas ocasiones, la plataforma puede fallar debido a alguna actualización pendiente, por lo que es importante siempre revisar si existe una en el momento actual e instalarla inmediatamente. (figura 10)

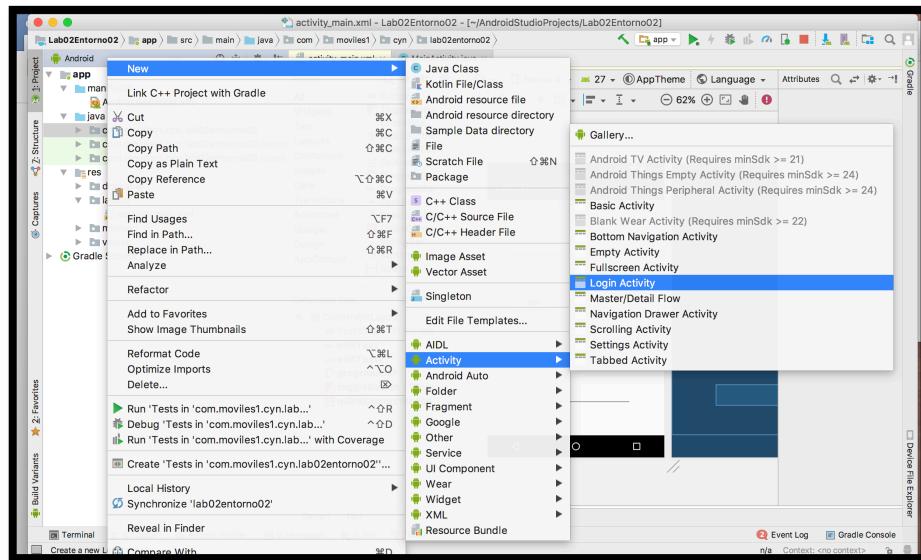
Figura 10: Instalacion de actualizaciones.



Tomado de Android Studio 3.0.1

Como se observa en la figura 11, esta es la manera correcta de agregar actividades nuevas. Una vez que se selecciona la actividad, se procede como en la figura 5. Es importante aclarar que cada una de las actividades creadas se maneja con archivos separados, es decir, que exista un xml para cada una de ellas.

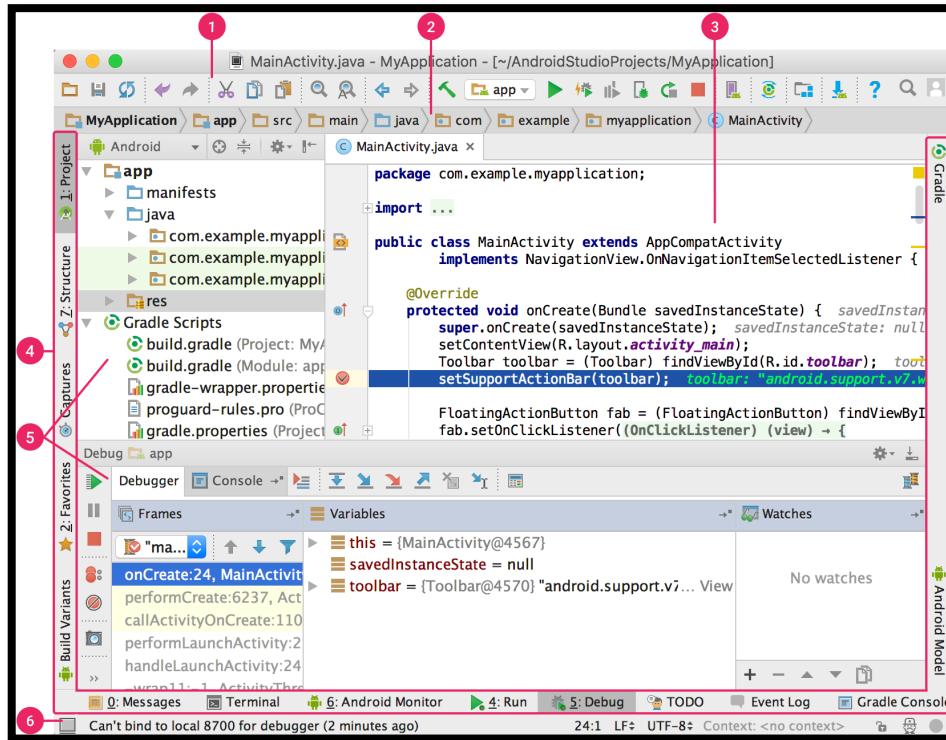
Figura 11: Añadir nuevas actividades.



Tomado de Android Studio 3.0.1

Teniendo esto claro, es esencial conocer la interfaz de la plataforma, por lo que en la figura 12 se pueden apreciar todos los puntos clave que se deben tomar en cuenta.

Figura 12: Elementos básicos de Android Studio.



Tomado de Android Developers(2018)

1. La **barra de herramientas**. Para llevar a cabo una variedad de acciones, como la ejecución de aplicación y el inicio de herramientas de Android.
2. La **barra de navegación**. Para explorar el proyecto y abrir archivos para editar. Proporciona una vista más compacta de la estructura.
3. La **ventana del editor** es el área donde se puede crear y modificar código. Según el tipo de archivo actual, el editor puede cambiar. Por ejemplo, cuando se visualiza un archivo de diseño, el editor muestra el editor de diseño.
4. La **barra de la ventana de herramientas** se extiende alrededor de la parte externa de la ventana del IDE y contiene los botones que permiten expandir o contraer ventanas de herramientas individuales.
5. Las **ventanas de herramientas** permiten acceder a tareas específicas, como la administración de proyectos, las búsquedas, los controles de versión, etc.
6. En la **barra de estado**, se muestra el estado del proyecto y del IDE en sí, como también cualquier advertencia o mensaje.

**Comparar diferentes arquitecturas existentes (programación en Xamarin – C#). Como producto se tendría la solución tanto en Android como en IO).**

Xamarin permite manejar las tres diferentes arquitecturas (Windows Phone, iOS y Android) en proyectos separados aprovechando los elementos de UI de cada una de las plataformas y a la vez compartir código en una librería común entre los tres proyectos como en el caso de las entidades, servicios y capas de negocio.

**¿Qué ventajas y desventajas ha logrado identificar entre las soluciones de Android y IOS proporcionadas por Xamarin? Explique.**

**Ventajas:**

- Permite escribir y compilar código una sola vez reduciendo el tiempo de desarrollo.
- Permite tener un único diseño front-end para todas las plataformas, de manera que se puede aprovechar para abarcar otras poblaciones.

**Desventajas:**

- Un xaml tiene poca similitud con xml y html, por lo que se toma su tiempo adaptarse a sus contenedores.
- Es difícil acoplarse a su estilo y su forma de trabajo, por lo que puede que algunos no saquen el máximo provecho de la herramienta. Además, hay que indagar muy a fondo para obtener ejemplos y soluciones viables para el proyecto en desarrollo.

## **Referencias.**

- [1] Indismatic.es. (2018). Ventajas y Desventajas de Android vs iOS vs Windows Phone – Asociación INDISMATIC. Disponible en: <http://www.indismatic.es/cadena-ser-rioja-baja/diferencias-android-ios-windows-phone/> Tomado el día: 20 Mar. 2018.
- [2] BeMovil Blog. (2018). Ventajas e inconvenientes del sistema operativo iOS. Disponible en: <https://www.bemovil.es/blog/ventajas-sistema-operativo-ios/> Tomado el día: 21 Mar. 2018.
- [3] House, P. (2018). iOS vs. Android: ventajas y diferencias de cada sistema operativo | Blog Oficial Phone House. [online] Blog.phonehouse.es. Disponible en: <http://blog.phonehouse.es/2017/08/24/ios-vs-android-ventajas-y-diferencias-de-cada-sistema-operativo/> Tomado el día: 22 Mar. 2018.
- [4] Sites.google.com. (2018). Ventajas y Desventajas - Windows Phone. Disponible en: <https://sites.google.com/site/windowsphone1com/home/ventajas-y-desventajas> Tomado el día: 22 Mar. 2018.
- [5] Studio, C. (2018). Conoce Android Studio | Android Studio. Developer.android.com. Disponible en: <https://developer.android.com/studio/intro/index.html?hl=es-419> Tomado el día: 24 Mar. 2018.