

**INSTITUTO POLITÉCNICO NACIONAL**

**ESCUELA SUPERIOR DE CÓMPUTO**

**ALUMNOS:**

**ALDAVERA GALLAGA IVÁN**

**CERRITOS SANTÁNDER DAVID**

**APPLICATION DEVELOPMENT FOR MOBILE DEVICES**

**PROFESOR:**

**ALEJANDRO SIGFRIDO CIFUENTES ÁLVAREZ**

**TÍTULO DEL REPORTE**

**EXAMEN FINAL**

**FECHA DE REALIZACIÓN:**

**22 – JUNIO – 2020**

**VERSIÓN DEL REPORTE:**

**V1**

**ÍNDICE**

OBJETIVO……………………………………………………………..1

CONCEPTOS………………………………………………………….1

Autenticación………………………………………………………………..1

Patrón de bloqueo…………………………………………………………..1

Canvas……………………………………………………………………….1

DESARROLLO…………………………………………………………1

LISTADO DE SOFTWARE………………..……………………………….3

LISTADO DE HARDWARE………………..……………………………….3

PRUEBAS……………………………………………………………….4

CAPTURAS Y DESCRIPCIONES…………………………………………4

DIFICULTADES………….…………………………………………………..9

CONCLUSIONES………………………………………………………10

BIBLIOGRAFÍA…………………………………………………………10

**OBJETIVO**

Realizar una aplicación en la que se pueda registrar un patrón de desbloqueo similar a los que utilizan los dispositivos móviles con sistema operativo Android, de modo que al ser desbloqueado el patrón pueda visualizarse un vídeo.

**CONCEPTOS**

**Autenticación**: es el acto o proceso de confirmar que algo (o alguien) es quien dice ser. A la parte que se identifica se le llama probador. A la parte que verifica la identidad se la llama verificador. Es habitual que el probador sea un usuario que quiere acceder a ciertos recursos y el verificador sea un sistema que protege el acceso a dichos recursos y tiene que verificar que el que accede sea un usuario que tiene permisos para acceder a esos recursos. Para poder tener autenticación es necesaria, como condición previa, la existencia de identidades biunívocamente identificadas de tal forma que se permita su identificación.

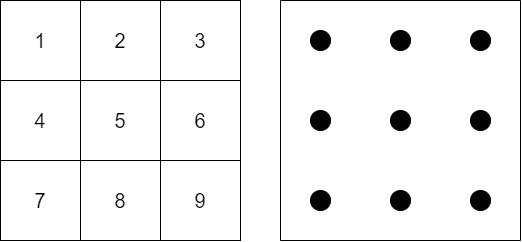
**Patrón de bloqueo**: Es una medida de seguridad que protege a los dispositivos, tales como teléfonos móviles o tabletas, y la cual es preferida sobre otras medidas de bloqueo como pines o contraseñas de texto. Es usado por alrededor del 40% de los propietarios de dispositivos Android. Con el fin de acceder a un dispositivo y su contenido, el usuario debe dibujar un simple patrón en una rejilla de puntos, los cuales al conectarse forman el patrón por sí mismos.

**Canvas**: el canvas o lienzo es una herramienta de dibujo consistente en una región con ciertos atributos como altura o anchura donde se pueden dibujar figuras o formas de acuerdo a las reglas que se especifiquen.

**DESARROLLO**

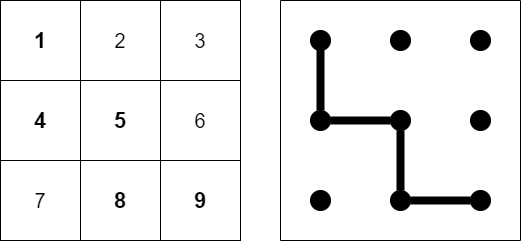
Para el desarrollo de este proyecto fue necesario entender cómo funciona el patrón de desbloqueo, el cual es simple de entender y se explicará a continuación:

Se cuenta con una matriz cuadrada de NxN puntos, aunque también podría ser rectangular de NxM puntos, pero es más común la matriz cuadrada. Cada punto de la matriz representa un número, como se ve en la figura 1.



***Figura 1.*** *Rejilla de puntos y su representación numérica.*

De tal manera que, al hacer un patrón como el de la figura 2.



***Figura 2.*** *Patrón dibujado en la rejilla y sus números correspondientes.*

Obtenemos el conjunto de números {1,4,5,8,9}. Dicho conjunto de números son los asociados a la figura del patrón, y son los números que interesan ya que son la clave con la cual se podrá hacer comparaciones con otros patrones, de manera que, si los números del patrón ingresado coinciden con los del previo, (en este caso el conjunto {1,4,5,8,9}), entonces se da acceso al dispositivo móvil.

Para la realización de la aplicación, se contó con el apoyo una matriz cuadrada de puntos dibujada mediante canvas, con la cual se implementó el *grid* o rejilla para unir los puntos, cada punto debe ser unido mediante una línea, de manera que al unir 2 o más puntos se forma un patrón de desbloqueo y un conjunto de números con el cual compararlos más adelante.

Mientras que para la sección en la cual se visualiza el vídeo, una vez que se haya comparado ambos conjuntos se números, se lanzará un *intent* o intento que mostrará la *activity* en donde aparece el vídeo junto con los botones de reproducir o salir. El vídeo se muestra mediante un componente VideoView de Android, que reproduce un tráiler de *Star Wars* con duración de 53 segundos aproximadamente guardado en el teléfono, aunque también podría reproducirse vía web.

Es importante recalcar, que para el botón *Salir,* dado que en Android no existe un concepto como tal de salir de la aplicación, más allá de detener la activity sobre la que estamos o de poner la aplicación apilada junto a las demás apps en ejecución, se realizó que se pudiera salir de la aplicación sin necesidad de detenerla o forzarla a cerrarse como puede hacer el propio sistema operativa si ésta llegara a fallar.

**LISTADO DE SOFTWARE:**

* Java Development Kit 8: Necesario para el desarrollo de aplicaciones Android.
* Android Studio 4.0: IDE para el desarrollo de aplicaciones Android
* Windows 10: Sistema operativo sobre el que corren los anteriores programas.
* Android 10. Sistema Operativo sobre el que corre la aplicación

**LISTADO DE HARDWARE:**

* Laptop HP con GB RAM DDR4 y procesador Intel Core i3
* Dispositivo Móvil Motorola One con 4GB de RAM y procesador SoC Qualcomm Snapdragon 625
* Cable tipo C para la transferencia de datos.

**PRUEBAS**

A continuación, se listan y describen las pruebas y capturas de pantalla de la aplicación:

|  |  |
| --- | --- |
| **CAPTURA** | **DESCRIPCIÓN** |
|  | Al inicio de la aplicación se solicita dibujar un patrón de desbloqueo por primera vez, el cuál una vez dibujado, debe hacerse clic sobre el botón *GUARDAR*,para que éste sea guardado*.* |
|  | En este caso dibujaremos el mismo patrón del que se puso de ejemplo en la sección de DESARROLLO. |
|  | Una vez guardado, nos aparecerá un Toast con un mensaje de guardado correcto, además de nos pedirá dibujar un patrón nuevamente para comprobar que el patrón dibujado concuerde con el anterior. Nótese un botón debajo del área de dibujo con el mensaje NUEVO PATRÓN, el cual nos permitirá volver a dibujar otro patrón nuevo si así lo quisiéramos, ya se porque olvidamos el anterior o porque decidamos cambiarlo simplemente. Más adelante veremos esto. |
|  | Dibujaremos el patrón de forma correcta en esta ocasión… |
|  | …para observar que al hacerlo nos aparecerá un Toast indicándonos que el patrón fue correcta y se nos redirigirá a la sección donde ese encuentra el vídeo. |
|  | Al presionar el botón REPRODUCIR, podemos correr el vídeo, el cuál se trata de un tráiler de Lego Star Wars. |
|  | Al presionar el botón SALIR, podremos salir de la aplicación y regresar a la pantalla principal |
|  | Volviendo al tema de hacer un nuevo patrón, podemos dar clic en el botón NUEVO PATRÓN, y podremos introducir uno nuevo |
|  | En este caso, dibujaremos este… |
|  | …para que una vez guardado se nos avise de que fue guardado exitosamente, y se nos solicitará dibujarlo de nuevo para confirmar. |
|  | Procederemos a dibujarlo nuevamente… |
|  | …Y automáticamente seremos redirigidos a la sección del vídeo, donde podremos reproducirlo nuevamente si queremos. |

**DIFICULTADES**

Algunas dificultades que se encontraron fueron con respecto al vídeo, pues teníamos pensado que este fuera de internet, sin embargo, el vídeo no se reproducía a veces, incluso teniendo los permisos correspondientes y declaras en el archivo AndroidManifest.xml para el uso de éste. Al final optamos porque el vídeo estuviera almacenado en la aplicación, para que no hubiera problema al visualizarlo

Con respecto a cerrar la aplicación, investigamos que como tal en Android no existe una forma de dar por finiquita a una aplicación mientras se ejecuta, ésta debe cancelarse manualmente en la pila de Apps abiertas, no es como en los programas Java, que cuentan con el método de *System.exit(0);* que permite cerrar la aplicación por completo. Sin embargo, pudimos hacer que se saliera de la app, sin cerrarla completamente, pero volviendo a la pantalla de inicio, como manualmente se haría con el botón cuadrado de Android.

**CONCLUSIONES**

**Aldavera Gallaga Iván**

El bloqueo por patrón es uno de los muchos métodos de bloqueo con los que se cuenta actualmente, tuvo su popularidad hace ya algunos años, peor ha venido decayendo por el uso de los sistemas biométricos como el de reconocimiento facial o de huella dactilar. Es importante observar que a final de cuentas, el bloqueo por patrón es como una especie de PIN extendido y de forma gráfica, por lo cual sería una variante engañosa de este otro método de bloqueo.

**Cerritos Santander David**

Android nos provee de una serie de métodos de desbloqueo muy variados, pero implementarlos es un poco complejo dada la enorme combinación de patrones que puede llegar a existir, y además, de los muchos métodos tanto gráficos para el canvas, como para los eventos del patrón, cuando se dibuja, cuando se termina de dibujar, cuando se está dibujando pero pasa un tiempo que no se dibuja y debe cancelarse el dibujo por seguridad, entre otros, son algunas cosas que hay que tomar en cuenta al momento de programarlo.

**BIBLIOGRAFÍA**

* Hyun-Woo Kim. (Agosto, 2014). Time Pattern Locking Scheme for Secure Multimedia Contents in Human-Centric Device. *Hindawi*, 2014, pp. 9.
* <https://www.sciencedaily.com/releases/2017/01/170123090818.htm>