



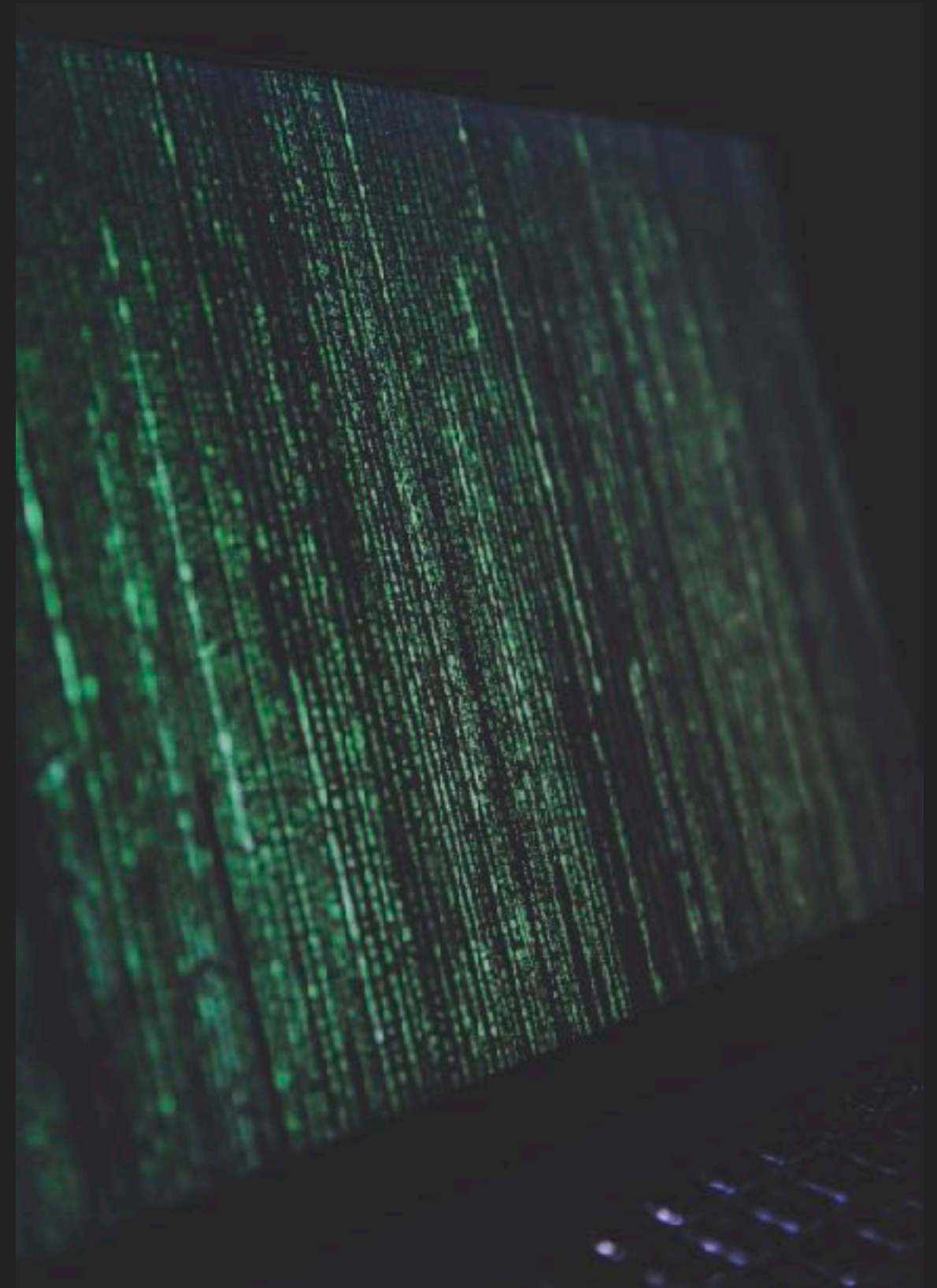
QA TESTING: GUIA BASICA

NIVELES Y TIPOS DE PRUEBA

- ▶ Los niveles de pruebas son básicamente para identificar esas areas perdidas y prevenir la superposición y repetición en los ciclos de desarrollo. En el ciclo de desarrollo de software hay definidas fases como el análisis, diseño, código o implementación, pruebas y montaje. Cada etapa debe ser probada. Existen varios NIVELES de testing, pero hablaremos de tres principales:
 - ▶ *PRUEBAS UNITARIAS*
 - ▶ *PRUEBAS DE INTEGRACION*
 - ▶ *PRUEBAS DE SISTEMAS*

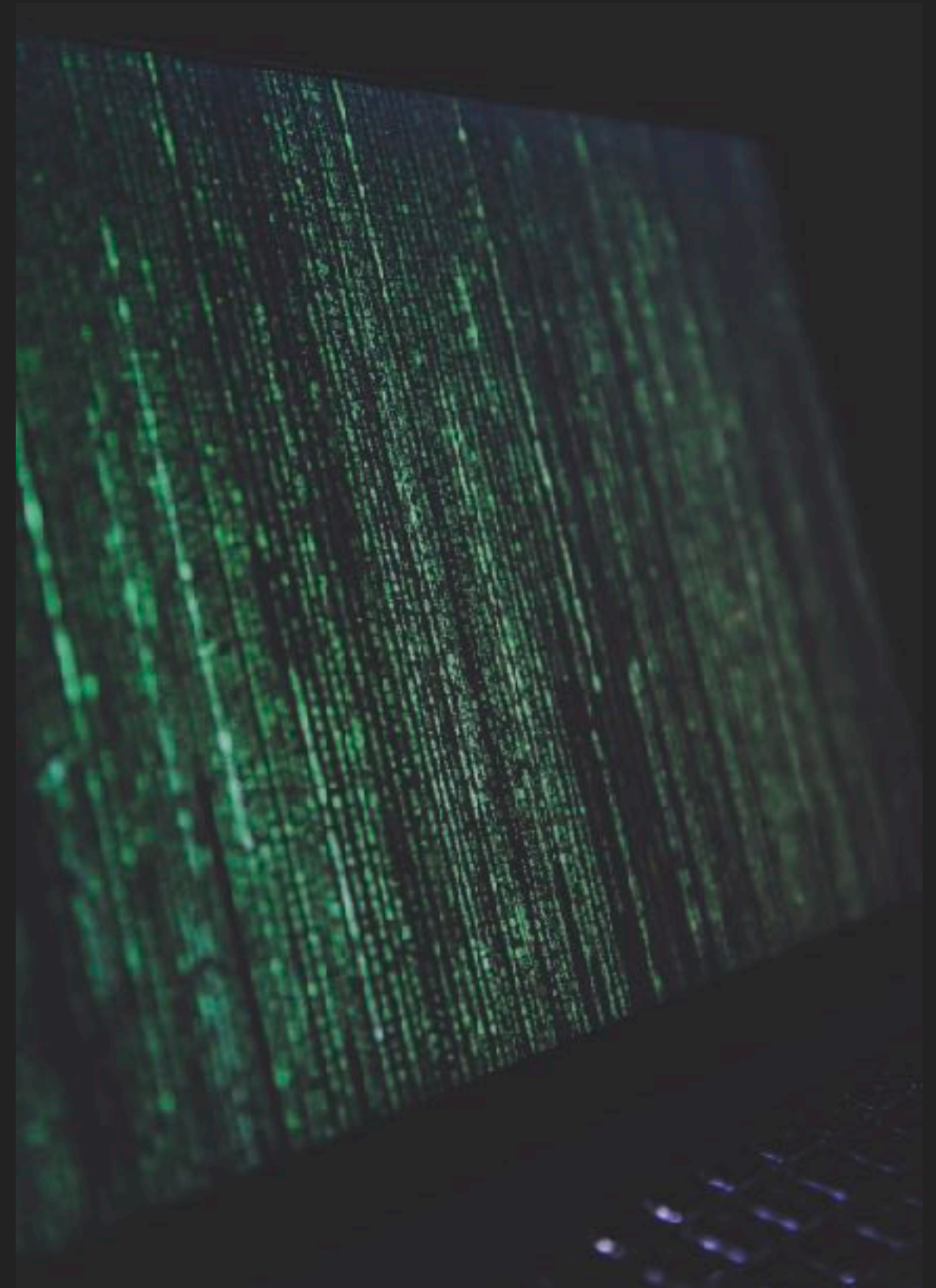
PRUEBAS UNITARIAS:

- ▶ Es un método que prueba una unidad de código. Poseen generalmente las siguientes características:
 - ▶ Prueba solamente pequeñas cantidades de código: Solamente prueba el código del requerimiento específico.
 - ▶ Se aísla de otro código y de otros desarrolladores: El unit test prueba exclusivamente el código relacionado con el requerimiento y no interfiere con el trabajo hecho por otros desarrolladores.
 - ▶ Solamente se prueban los endpoints públicos: Esto principalmente porque los disparadores de los métodos privados son métodos públicos por lo tanto se abarca el código de los métodos privados dentro de las pruebas.
 - ▶ Los resultados son automatizados: Cuando ejecutamos las pruebas lo podemos hacer de forma individual o de forma grupal. Estas pruebas las hace el motor de prueba y los resultados de los mismos deben de ser precisos con respecto a cada prueba unitaria desarrollada
 - ▶ Repetible y predecible: No importa el orden y las veces que se repita la prueba, el resultado siempre debe de ser el mismo.
 - ▶ Son rápidos de desarrollar: Contrariamente a lo que piensan los desarrolladores -> que el desarrollo de pruebas unitarias quita tiempo - los unit test por lo general deben de ser simples y rápidos de desarrollar. Difícilmente una prueba unitaria deba de tomar más de cinco minutos en su desarrollo.
 - ▶ Son llevadas a cabo por los desarrolladores



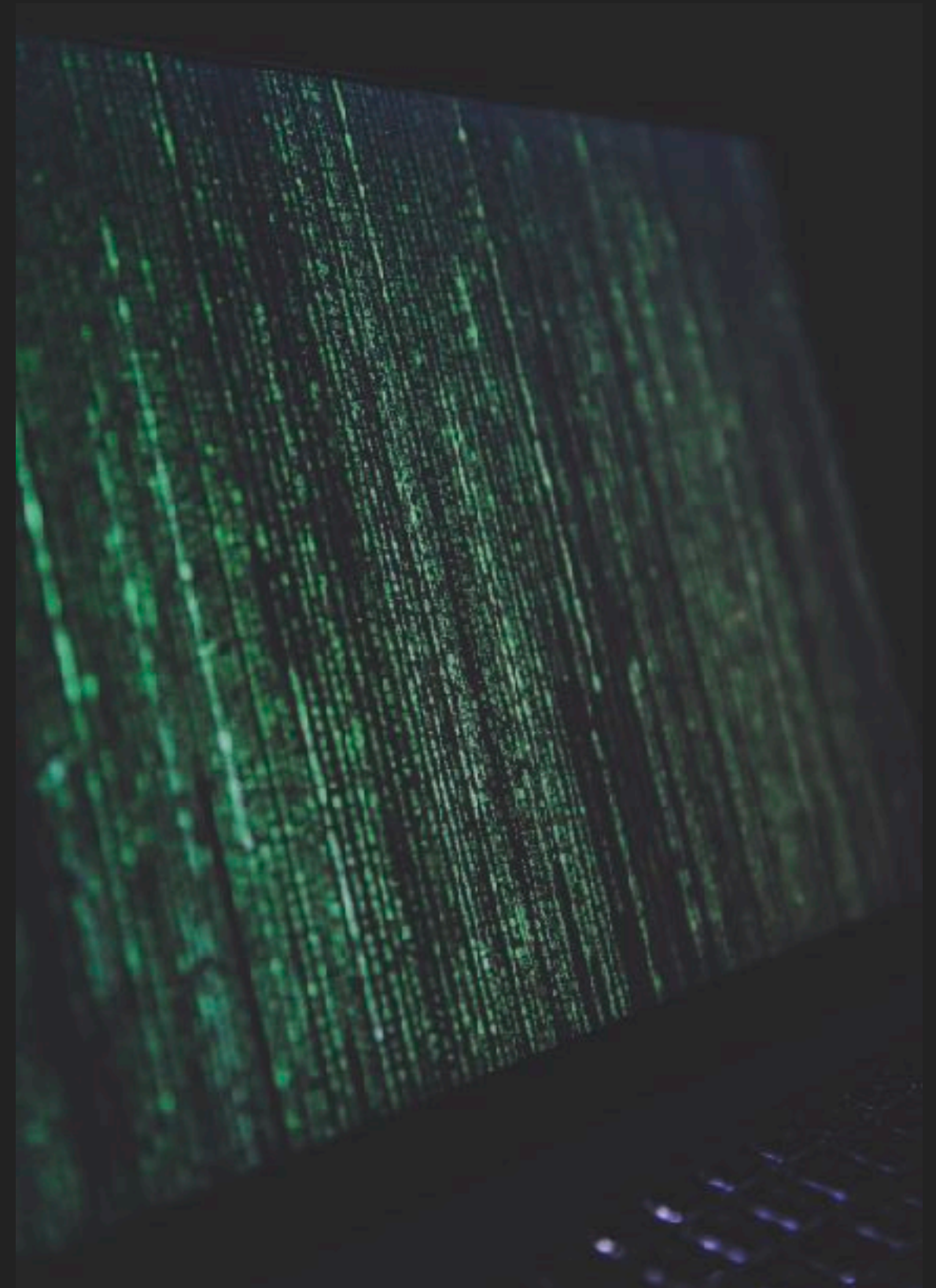
PRUEBAS DE INTEGRACIÓN:

- ▶ Las pruebas de integración se llevan a cabo justo después de haber hecho las pruebas unitarias y permiten determinar la interacción entre diferentes bloques de código. Estas pruebas permiten reconocer
 - ▶ Problemas de configuración.
 - ▶ Funciones faltantes o con conflictos.
 - ▶ Uso incorrecto o inconsciente de las bases de datos.
 - ▶ Violaciones de integridad de datos.
 - ▶ Llamados a metodos equivocados.
 - ▶ Parametros erróneos o valores equivocados.
 - ▶ Fallas causadas por mal manejo de memoria.
 - ▶ Conflictos entre componentes.
 - ▶ Recursos insuficientes



PRUEBAS DE SISTEMAS:

- ▶ Son llevadas a cabo por un equipo diferente al de desarrollo y generalmente son planificadas y documentadas. Las pruebas de sistemas es un requerimiento que debe llevarse a cabo antes de liberar el producto al usuario final. En esta fase se:
 - ▶ Planifican las pruebas.
 - ▶ Redactan los escenarios o casos de prueba.
 - ▶ Comprueba el porcentaje del sistema a probar.
 - ▶ Verifica el completo funcionamiento del sistema.



MÉTODOS DE PRUEBA

- ▶ Diferentes técnicas de pruebas son apropiadas para diferentes momentos de desarrollo de software. Para efectos de este curso, la mayoría de las pruebas que realizaremos serán de Caja Negra en la cual la funcionalidad se verifica sin tomar en cuenta la estructura interna de código.
- ▶ Por supuesto existen muchos mas métodos de pruebas de los que puedes encontrar información en la internet. Pero en el transcurso del curso no las cubriremos

TIPOS DE PRUEBAS:

- ▶ Existen multiples tipos de pruebas, pero para efectos de este curso, las mas importantes que debes conocer las detallare en adelante sin ningún orden específico
 - ▶ Pruebas Formales: Llevadas a cabo por ingenieros de pruebas
 - ▶ Pruebas Informales: Llevadas a cabo por los desarrolladores
 - ▶ Pruebas Manuales: Son aquellas en las que el software requiere ser validado según la interacción con usuarios, analizado o evaluado
 - ▶ Pruebas Automáticas: Es una forma de pruebas que utiliza una herramienta de software para recrear la interacción humana
 - ▶ Pruebas Funcionales: Verifica que la función de un modulo sea tal cual la que se estableció en una especificación previa
 - ▶ Pruebas de Usuario Final: Es muy similar a las pruebas del sistema y simula el comportamiento del software con datos reales y en un ambiente similar al de los usuarios finales
 - ▶ Pruebas de Regresión: Son ejecutadas para verificar que las funciones que ya se han probado no han sido alteradas con la inclusion de nuevo código
 - ▶ Pruebas de Aceptación: Son pruebas que se llevan a cabo en nuevas funciones del sistema y verifican que han sido integradas adecuadamente y que cumple con los requerimientos establecidos