TABLA 11: Unidades de Aprendizaje

Unidad de Aprendizaje Nº 1. Prueba de aplicaciones web y para dispositivos móviles		
Temporalización: 1er trimestre	**Duración**: 35 h.	**Ponderación**: 20%

Objetivos Generales	Competencias
k, l, q, r, s, t, u, v	f, j, k, l, m, n, ñ

Resultados de Aprendizaje

1

Aspectos del Saber Hacer

Comparar lenguajes de programación en base a las características de cada uno de ellos.

Identificación de los diferentes modelos de ejecución del software.

Conocer los elementos básicos del código fuente.

Ejecutar pruebas de software.

Evaluar los lenguajes de programación en función de la seguridad que proporcionan.

Aspectos del Saber Estar

- Conocer las principales características de los lenguajes.

- Conocer los diferentes modelos de ejecución de software y sus características.

- Saber dar significado a los elementos que constituyen el software.

- Saber diferenciar los distintos tipos de pruebas de software y sus características.

- Saber diferenciar los lenguajes de programación y conocer sus características principales en relación con la seguridad que proporcionan.

Aspectos del Saber

Fundamentos de la programación
- Fundamentos de los lenguajes más utilizados I.
- Fundamentos de los lenguajes más utilizados II.

Lenguajes de programación interpretados y compilados
- Compilación en tiempo de ejecución.

Código fuente y entornos de desarrollo.
- Entornos de desarrollo.

Ejecución de software.

Elementos principales de un programa.

Pruebas.
- Etapas o niveles de pruebas.
  - Pruebas unidad.
  - Pruebas de integración.
  - Pruebas de sistema.
  - Pruebas de aceptación.

Tipos de pruebas
- Pruebas funcionales.
- Pruebas no funcionales.

Técnicas de prueba
- Pruebas de caja negra.
- Pruebas de caja blanca.

Seguridad en los lenguajes de programación y sus entornos de ejecución (sandboxes).

Tareas y Actividades

- Se evaluará, mediante la realización de una prueba teórica que englobe los criterios de evaluación a) y b), los diferentes lenguajes de programación y modelos de ejecución del software.
- Se realizarán pruebas prácticas para trabajar con conceptos de programación en Python, explicando sus elementos principales: funciones, condicionales, bucles, trabajo con ficheros, etc.
- Se ejecutará para entrega una tarea práctica individual con la que se evaluará el grado de conocimiento de los conceptos de programación, niveles de seguridad, gestión de errores y otros conceptos estudiados.

Criterios de Evaluación	%	IE
a) Se han comparado diferentes lenguajes de programación de acuerdo a sus características principales.	15	Prueba teórica.
b) Se han descrito los diferentes modelos de ejecución del software.	20	Prueba teórica.
c) Se han reconocido los elementos básicos del código fuente, dándoles significado.	15	Prueba práctica.
d) Se han ejecutado diferentes tipos de prueba de software.	20	Prueba práctica.
e) Se han evaluado los lenguajes de programación de acuerdo a la infraestructura de seguridad que proporcionan.	30	Tarea práctica individual.

Recursos

Aula-taller de informática con ordenadores suficientes para cada alumno de la clase.

Pantalla de proyección.

Software para creación de máquinas virtuales: VBox, WMware.

Otro software: Python, Visual Studio Code.

Observaciones

Unidad de Aprendizaje Nº 2. Determinación del nivel de seguridad requerido por aplicaciones.		
Temporalización: 1-2 trimestre	**Duración**: 35 h.	**Ponderación**: 20%

Objetivos Generales	Competencias
k, l, q, r, s, t, u, v	f, j, k, l, m, n, ñ

Resultados de Aprendizaje

2

Aspectos del Saber Hacer

Comprobar la seguridad a nivel de aplicación.

Identificar los niveles de verificación de seguridad que requieren las aplicaciones en base a los estándares reconocidos.

Enumerar requisitos de verificación asociados a niveles de seguridad.

Reconocer los riesgos en las aplicaciones desarrolladas.

Aspectos del Saber Estar

- Saber realizar comprobaciones de seguridad a nivel de aplicación.
- Saber identificar los niveles de seguridad de aplicaciones según estándares reconocidos.
- Saber enumerar qué requisitos de verificación están asociados a cada nivel de seguridad.
- Saber identificar iesgos en aplicaciones desarrollada según sus características.

Aspectos del Saber

Fuentes abiertas para el desarrollo seguro.
Técnicas y herramientas
Desarrollo seguro y fuentes abiertas.

Listas de riesgos de seguridad habituales: OWASP Top Ten (web y móvil)
Proyectos fundación OWASP.
OWASP Top Ten Web.
OWASP Top Ten Móvil.

Requisitos de verificación necesarios asociados al nivel de seguridad establecido.
Estándares para medir la calidad del software.
Niveles de verificación.
Estándar de Verificación de Seguridad en Aplicaciones (ASVS).

Web Security Testing Guide.
Mobile Security Testing Guide.
Estándar de verificación de seguridad de aplicaciones móviles. ASVS).

Comprobaciones de seguridad a nivel de aplicación: ASVS.

Tareas y Actividades

- Se realizará una tarea práctica individual basada en los conceptos de identificación del nivel de seguridad en función de posibles riesgos.
- Se realizará también una prueba teórica que evaluará los criterios de evaluación a) y c).

- Por último, se realizará una prueba práctica para trabajar conceptos de los estándares internacionales ASVS de verificación de seguridad.

Criterios de Evaluación	%	IE
a) Se han caracterizado los niveles de verificación de seguridad en aplicaciones establecidos por los estándares internacionales (ASVS, “Application Security Verification Standard”).	25	Prueba teórica.
b) Se ha identificado el nivel de verificación de seguridad requerido por las aplicaciones en función de sus riesgos de acuerdo a estándares reconocidos.	25	Tarea práctica individual.
c) Se han enumerado los requisitos de verificación necesarios asociados al nivel de seguridad establecido.	25	Prueba teórica
d) Se han reconocido los principales riesgos de las aplicaciones desarrolladas, en función de sus características.	25	Prueba práctica.

Recursos

Aula-taller de informática con ordenadores suficientes para cada alumno de la clase.

Pantalla de proyección.

Software para creación de máquinas virtuales: VBox, WMware.

Observaciones

Unidad de Aprendizaje Nº 3. Implantación de sistemas seguros de despliegue de software.		
Temporalización: 2º trimestre	**Duración**: 32 h.	**Ponderación**: 20 %

Objetivos Generales	Competencias
k, l, q, r, s, t, u, v	f, j, k, l, m, n, ñ

Resultados de Aprendizaje

3

Aspectos del Saber Hacer

Validar entradas de los usuarios.

Detectar posibles riesgos de inyección en servidor y cliente.

Gestionar las sesiones de usuario en el proceso de uso de la aplicación.

Hacer uso de roles en relación con el control de acceso.

Utilizar algoritmos criptográficos para el almacenamiento de las contraseñas.

Configurar servidores web para reducir la posibilidad de ataques.

Incorporar medidas para evitar ataques a contraseñas.

Aspectos del Saber Estar

Saber aplicar los estándares de autenticación y autorización para validar entradas de los usuarios.

Saber identificar qué posibles riesgos de inyección se pueden producir en servidores y clientes.

Saber aplicar soluciones en la gestión de las sesiones de usuario durante el uso de una aplicación.

Saber hacer uso de diferentes roles para el control de acceso.

Aspectos del Saber

Prácticas unificadas para el desarrollo y operación del software (DevOps).
- Ciclo de vida de las aplicaciones.
- Prácticas
- Herramientas.

Sistemas de control de versiones.
- Git.
  - Instalación.
  - Primeros pasos.
  - Ramas.
  - Trabajando en remoto.
  - GitFlow.
  - Hook / Ganchos.

Sistemas de automatización de construcción (build).
- Construyendo un proyecto con Gradle

Integración continua y automatización de pruebas.

Escalado de servidores. Virtualización. Contenedores.
- Virtualización.
- Contenedores.
  - Docker.
    - Instalación.
    - Descargando imágenes y creando contenedores.
    - Persistencia de datos.
    - Dockfile.

Gestión automatizada de configuración de sistemas.

Orquestación de contenedores.
- Componentes de Kubernetes.
- Objetos Kubernetes.

Herramientas de simulación de fallos.

Saber aplicar algoritmos criptográficos para almacenar contraseñas. Saber configurar los servidores web para tratar de disminuir la posibilidad de recibir ataques. Saber aplicar diferentes medidas para minimizar la posibilidad de recibir ataques de contraseñas.

Tareas y Actividades

- Se realizarán varias pruebas prácticas, que pueden incluso agrupar la evaluación de varios criterios de evaluación.
- Se realizarán también tareas prácticas individuales para entregar y poder evaluar el avance e interiorización de conceptos.
- Para la evaluación del criterio de evaluación g) se realizará una prueba teórica.

Criterios de Evaluación	%	IE
a) Se han validado las entradas de los usuarios.	10	Prueba práctica.
b) Se han detectado riesgos de inyección tanto en el servidor como en el cliente.	15	Prueba práctica.
c) Se ha gestionado correctamente la sesión del usuario durante el uso de la aplicación.	15	Prueba práctica.
d) Se ha hecho uso de roles para el control de acceso.	15	Prueba práctica.
e) Se han utilizado algoritmos criptográficos seguros para almacenar las contraseñas de usuario.	15	Tarea práctica individual
f) Se han configurado servidores web para reducir el riesgo de sufrir ataques conocidos.	15	Tarea práctica individual
g) Se han incorporado medidas para evitar los ataques a contraseñas, envío masivo de mensajes o registros de usuarios a través de programas automáticos (bots).	15	Prueba teórica.

Recursos

Aula-taller de informática con ordenadores suficientes para cada alumno de la clase
Pantalla de proyección.
Software para creación de máquinas virtuales: VBox, WMware.
Otro software: Git, Dockers.

Observaciones

Unidad de Aprendizaje Nº 4. Detección y corrección de vulnerabilidades de aplicaciones web.		
Temporalización: 2º-3º trimestre	**Duración**: 32 h.	**Ponderación**: 20%

Objetivos Generales	Competencias
k, l, q, r, s, t, u, v	f, j, k, l, m, n, ñ

Resultados de Aprendizaje

4

Aspectos del Saber Hacer

- **Comparar** los modelos existentes de permisos en las plataformas móviles.
- **Describir** técnicas de almacenamiento seguro.
- **Implantar** sistemas de validación de compras haciendo uso de validación en el servidor.
- **Utilizar** herramientas de monitorización de tráfico de red.
- **Inspeccionar** aplicaciones móviles en busca de posibles fugas de información.

Aspectos del Saber Estar

- Saber identificar los distintos modelos de permisos de las plataformas móviles.
- Saber identificar las diferentes técnicas de almacenamiento seguro de datos en los dispositivos.
- Saber realizar la implantación de sistemas de validación de compras en una aplicación.
- Saber hacer uso de las diferentes herramientas de monitorización de tráfico de red para poder detectar protocolos no seguros de comunicaciones móviles.
- Saber aplicar técnicas de inspección en aplicaciones móviles para detectar posibles fugas de información sensible.
-

Aspectos del Saber

Protocolo HTTP.
- Formato del mensaje HTTP.
- Versiones del protocolo.
- Tecnologías Web.

Desarrollo seguro de aplicaciones web.

Listas públicas de vulnerabilidades de aplicaciones web.
OWASP Top Ten.
- Inyección.
- Pérdida de autenticación.
- Exposición de datos sensibles.
- Entidades externas XML (XXE).
- Pérdida de Control de Acceso.
- Configuración de seguridad incorrecta.
- Cross-Site Scripting(XSS).
- Deserialización insegura.
- Componentes con vulnerabilidades conocidas.
- Registro y monitoreo insuficientes.

Entrada basada en formularios. Inyección. Validación de la entrada.
- Validación en el cliente.
- Validación en el servidor.

Estándares de autenticación y autorización.
- Elementos de un sistema de autenticación y autorización básico.
- Estándares.
  - Tokens JWT
  - OAth.

Robo de sesión.

Almacenamiento seguro de contraseñas.
- Ejemplo almacenamiento seguro NodeJs.

Contramedidas. HSTS, CSP, CAPTCHAs, entre otros.
- HSTS (HTTP Strict Transport Security)
- CSP (Content Security Policy)

	CAPTCHAs Seguridad de portales y aplicativos web. Soluciones WAF(Web Application Firewall).

Tareas y Actividades

Se realizará una prueba teórica para evaluar los criterios de evaluación a) y e).

Se realizarán dos pruebas prácticas individuales para evaluar los conceptos impartidos.

El alumnado deberá realizar la entrega de dos tareas prácticas individuales.

Criterios de Evaluación	%	IE
a) Se han comparado los diferentes modelos de permisos de las plataformas móviles.	20	Prueba teórica.
b) Se han descrito técnicas de almacenamiento seguro de datos en los dispositivos, para evitar la fuga de información.	20	Tarea práctica individual
c) Se ha implantado un sistema de validación de compras integradas en la aplicación haciendo uso de validación en el servidor.	20	Prueba práctica.
d) Se han utilizado herramientas de monitorización de tráfico de red para detectar el uso de protocolos inseguros de comunicación de las aplicaciones móviles.	20	Tarea práctica individual
e) Se han inspeccionado binarios de aplicaciones móviles para buscar fugas de información sensible.	20	Prueba teórica.

Recursos

Aula-taller de informática con ordenadores suficientes para cada alumno de la clase.
Pantalla de proyección.
Software para creación de máquinas virtuales: VBox, WMware.
Otro software: Aplicaciones específicas para el desarrollo de tareas.

Observaciones

Unidad de Aprendizaje Nº 5. Detección de problemas de seguridad en aplicaciones para dispositivos móviles.		
Temporalización: 3º trimestre	**Duración**: 31 h.	**Ponderación**: 20%

Objetivos Generales	Competencias
k, l, q, r, s, t, u, v	f, j, k, l, m, n, ñ

Resultados de Aprendizaje

5

Aspectos del Saber Hacer

- **Identificar** características y objetivos para integrar el desarrollo y operaciones del software.
- **Implantar** sistemas de control de versiones.
- **Administrar** sistemas de integración continua.
- **Planificar** planes de despliegue de software.
- **Evaluar** la capacidad que ofrece el sistema para reaccionar ante fallos.
- **Documentar** tareas y procedimientos para la recuperación ante desastres.
- **Crear** bucles de retroalimentación en un equipo.

Aspectos del Saber Estar

- Saber identificar las principales características para la integración de desarrollo del software.
- Saber realizar la implantación de sistemas de control de versiones.
- Saber gestionar sistemas de integración continua para conectarlos a sistemas de control de versiones.
- Saber aplicar planes para desplegar software.
- Saber ser capaz de conocer la capacidad de un sistema para reaccionar a fallos de manera automática.
- Saber generar documentación de las acciones realizadas y procedimientos de recuperación ante desastres.

Aspectos del Saber

Modelos de permisos en plataformas móviles.
- Android.
  - Fichero Androidmanifest.xml
  - Solicitando permisos
    - Permisos de instalación o normales
      - Permisos en tiempo de ejecución
  - Estableciendo permisos
- iOS

Firma y verificación de aplicaciones.
- Fundamentos criptográficos.
- Android.
- iOS.

Almacenamiento seguro de datos.
- MSTG para el almacenamiento seguro en Android.
  - Análisis de MSTG-1 y 2
- MSTG para almacenamiento seguro en iOS

Fuga de información en los ejecutables
- Casos de prueba en Android

Soluciones CASB (Cloud Access Security Broker).

- Saber crear bucles de retroinformación entre miembros de un equipo.

Tareas y Actividades

El alumnado deberá realizar tres pruebas teóricas, que podrán realizarse de manera agrupada.

La evaluación será completada con la realización de una prueba práctica y tres taras prácticas individuales.

Criterios de Evaluación	%	IE
a) Se han identificado las características, principios y objetivos de la integración del desarrollo y operación del software.	20	Prueba práctica
b) Se han implantado sistemas de control de versiones, administrando los roles y permisos solicitados.	15	Prueba teórica.
c) Se han instalado, configurado y verificado sistemas de integración continua, conectándolos con sistemas de control de versiones.	15	Prueba teórica.
d) Se han planificado, implementado y automatizado planes de desplegado de software.	15	Prueba teórica.
e) Se ha evaluado la capacidad del sistema desplegado para reaccionar de forma automática a fallos.	15	Tarea práctica individual
f) Se han documentado las tareas realizadas y los procedimientos a seguir para la recuperación ante desastres.	10	Tarea práctica individual
g) Se han creado bucles de retroalimentación ágiles entre los miembros del equipo.	10	Tarea práctica individual

Recursos

Aula-taller de informática con ordenadores suficientes para cada alumno de la clase.
Pantalla de proyección.
Software para creación de máquinas virtuales: VBox, WMware.
Otro software: Aplicaciones específicas para el desarrollo de tareas.

Observaciones