Universidad de La Laguna

Máster en Formación del Profesorado de Educación Secundaria Obligatoria y Bachillerato, Formación Profesional y Enseñanza de Idiomas
2023-2024

Trabajo Fin de Máster

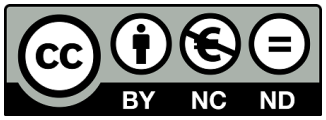
"Programación didáctica para Programación de Servicios y Procesos. Situación de aprendizaje: Desarrollo seguro de software"

Familia: Informática y Comunicaciones
Ciclo: CFGS Desarrollo de Aplicaciones Multiplataforma
Centro: IES San Juan de la Rambla

Antonio Manuel Hernández De León

Tutora
María Isabel Dorta González

La Laguna, Mayo 2024



Agradecimientos

Una vez más, quiero agradecer con toda mi alma a mis padres, Marcial y Rosa Delia, por su apoyo incondicional en cada una de mis decisiones tomadas en los últimos años, por darme la oportunidad de formarme y poder dedicarme profesionalmente a lo que tanto me ha gustado desde la niñez y por siempre permitirme perseguir mis sueños confiando en que por cada objetivo que me he planteado me esforzaría al máximo en cumplirlo.
A mi abuela, Nieves, por creer en mí en todo momento y tener la fe de que siempre valoraría y aprovecharía las oportunidades de formación académica que he tenido para vivir un futuro mejor.
A mi abuelo, Antonio y a mi tío, Juan Antonio, que pese a no estar físicamente a mi lado, sé que siempre serán las estrellas que me guiarán lo mejor posible en el camino de mi vida.
A mi tutora, María Isabel, por tutorizarme este Trabajo de Fin de Máster y guiarme en el Programa de Prácticas Externas en el IES San Juan de la Rambla.
Por último, tansmitirles mis agradecimientos a todas mis amistades y a aquellas personas que he conocido y me han dado alegría en algún momento de mi vida, marcando como dos de los más especiales mi estancia en la residencia durante mi primer año de universidad en Madrid y las experiencias inolvidables que he vivido este último año en Tenerife. Momentos que me han hecho ser lo que soy a día de hoy.
¡Muchas gracias a todos, siempre los llevaré en el corazón!

Resumen

El objetivo de este Trabajo de Fin de Máster es el desarrollo de una programación didáctica para el módulo de Programación de Servicios y Procesos, junto con una Situación de Aprendizaje "Desarrollo seguro de software" dirigida al primer curso del Ciclo Formativo de Grado Superior de Desarrollo de Aplicaciones Multiplataforma impartido en el IES San Juan de la Rambla.

Palabras clave: Programación didáctica, situación de aprendizaje, ciberseguridad, desarrollo seguro de software, protocolos seguros en red.

Abstract

The objective of this Master Thesis is to develop a learning situation for the services and processes programming module, along with a didactic unit "secure software development" made for the first year of the Multiplatform Application Development advanced vocational education course at San Juan de la Rambla highschool.

Keywords: Learning situation, didactic unit, cybersecurity, secure software development, network security protocols.

Índice general

Índice de figuras

Índice de tablas

Capítulo 1

Introducción

En este Trabajo de Fin de Máster se desarrolla la programación didáctica del módulo "Programación de Servicios y Procesos" perteneciente al Ciclo Formativo de Grado Superior de Desarrollo de Aplicaciones Multiplataforma, impartido en el IES San Juan de la Rambla. Asimismo, se expone una de las situaciones de aprendizaje del módulo, en este caso, la de Desarrollo seguro de software.

1.1. Estructura de la memoria y objetivos

En esta sección se expone una descripción del contenido de los capítulos que constituyen este documento. Dichos capítulos son los siguientes:

- **Análisis reflexivo y valoración crítica de la programación didáctica**. Se realiza tanto un análisis como una valoración crítica de la programación didáctica del módulo "Programación de Servicios y Procesos" perteneciente al Ciclo Formativo de Grado Superior de Desarrollo de Aplicaciones Multiplataforma, impartido en el IES San Juan de la Rambla en el curso 2023-2024, en donde se detallan los puntos fuertes de la misma conjuntamente con los aspectos que se podrían mejorar.

- **Diseño de la programación didáctica anual**. Abarca un desarrollo completo de la programación didáctica, la cual es el objetivo de este proyecto. Se incluirán aspectos como la contextualización del entorno de aprendizaje, la ordenación temporal de las diferentes situaciones de aprendizaje del módulo profesional, sus metodologías, etc.

- **Situación de aprendizaje: Desarrollo seguro de software**. En él, se detalla la situación de aprendizaje propuesta "Desarrollo seguro de software".

- **Conclusiones**. Contiene las conclusiones que se han obtenido con la realización de este TFM.

El documento también incorpora varios apéndices, los cuales contienen información de distinto tipo:

- **Material didáctico para la situación de aprendizaje Desarrollo seguro de software**. Corresponde al Apéndice *Material didáctico para la situación de aprendizaje Desarrollo seguro de software*.
- **Propuesta de actividades para la situación de aprendizaje Desarrollo seguro de software**. Mostradas en el Apéndice *Propuesta de actividades para la situación de aprendizaje Desarrollo seguro de software*.

Capítulo 2

Análisis reflexivo y valoración crítica de la programación didáctica

2.1. Análisis reflexivo

En este análisis se detalla la programación didáctica del departamento de Informática y Comunicaciones del IES San Juan de la Rambla, específicamente, la que abarca el módulo profesional "Programación de Servicios y Procesos" perteneciente al Ciclo Formativo de Grado Superior de Desarrollo de Aplicaciones Multiplataforma.

Cabe mencionar que dicho módulo se rige por el Real Decreto 450/2010, de 16 de abril, por el que se establece el título de Técnico Superior en Desarrollo de Aplicaciones Multiplataforma y se fijan sus enseñanzas mínimas (Ministerio de Educación, 2010b). Referenciando a la programación diseñada por el IES San Juan de la Rambla, se observa que está estructurada en once apartados desarrollados en nueve páginas con bastantes referencias a aspectos detallados en la programación general del departamento. Los apartados son los siguientes:

- "Datos de identificación del módulo.
- Revisión de la programación del curso anterior y modificaciones respecto a la edición anterior.
- Competencias profesionales, personales y sociales del ciclo a las que contribuye el módulo.
- Objetivos generales del ciclo a los que contribuye el módulo.
- Resultados de aprendizaje del módulo y criterios de evaluación.
- Atención a la diversidad.
- Metodología.
    - Recursos y materiales.
    - Actividades extraescolares y complementarias.
- Situaciones de aprendizaje.

    - Secuenciación de contenidos.
    - Programación de las situaciones de aprendizaje.
    - Ponderación de las SA en la consecución de los Resultados de Aprendizaje.
    - Desarrollo de contenidos de las SA.

- Evaluación.
- Medidas de atención a la diversidad.
- Materiales, recursos didácticos, bibliografía, webgrafía.

    - Materiales.
    - Recursos didácticos.
    - Referencias bibliográficas.
    - Webgrafía." (Departamento de Informática y Comunicaciones del IES San Juan de la Rambla, 2023b)

2.2. Valoración crítica

En cuanto a los resultados de aprendizaje expuestos en el Real decreto 450/2010 y su ponderación en la nota final del módulo, se observa que todos ellos tienen la misma ponderación aproximadamente, dándole un poco más de peso a la dominancia de la programación multihilo, la eficiencia y la disponibilidad de los servicios en red y las políticas de seguridad aplicada a esos servicios. Considero esto un aspecto positivo de la programación didáctica del Centro ya que estimo que todos los RA tienen una importancia similar.

Por otra parte, se puede contemplar muy detalladamente el protocolo y las metodologías a seguir en caso de que se desarrollase de nuevo una emergencia sanitaria equivalente a la del COVID-19 con el fin de seguir con la enseñanza de una manera eficiente. Está detallado tanto el modelo de una enseñanza semipresencial como online. Pienso que esto es importante tenerlo en cuenta para estar preparados para lo que pueda pasar en el futuro.

Con respecto a los contenidos básicos, propongo que deberían ampliarse y contemplar la documentación de los códigos fuentes de los programas, a la cual no se hace referencia. La parte de documentación de programas desarrollados es fundamental para los desarrolladores de software, por lo que se debería hacer más hincapié.

Dado que en el apartado de Metodología se menciona que se partirá de los conocimientos previos del alumno para impartir las clases, pero no se especifica cuáles son esos conocimientos previos, considero que debería tenerse en cuenta esta propuesta de mejora. Si se especificase y se mencionase, la docencia podría ser mucho más eficiente debido a que esto permitiría al alumnado reforzar conocimientos de cursos anteriores para agilizar el aprendizaje de este módulo profesional.

Con respecto a la evaluación, se detalla que las pruebas escritas tendrán un peso del 60 %. Bajo mi punto de vista, este porcentaje es excesivo para un módulo práctico, enfocado a la programación como lo es éste. Mi propuesta es dar más peso a las prácticas o proyectos a

realizar, haciendo que el alumnado se enfrente a problemas reales que les hagan aprender. Otro aspecto a resaltar es que no se especifica qué formato tendrán los exámenes (si se realizarán a papel o a ordenador), y tampoco se detalla cómo se va a tratar el uso de herramientas de Inteligencia Artificial como Chat GPT para la realización de las prácticas (si se va a prohibir, si se permitirá el uso sin límites o si se controlará de alguna manera).

Por último, comentar que a pesar de que sí se expone la secuenciación y programación de las diferentes situaciones de aprendizaje, estas no están desarrolladas, no teniendo claro qué tipo de actividades se llevarán a cabo para el aprendizaje o para la evaluación.

Tras todo lo expuesto, a continuación se presentan las siguientes propuestas de mejora para la programación didáctica del centro:

- Abarcar la parte de documentación de software en los resultados de aprendizaje a la hora de realizar proyectos prácticos de programación.
- Especificación de los conocimientos previos del alumno.
- Dar mayor peso en la evaluación a los proyectos prácticos que a los exámenes escritos.
- Detalles del formato de examen (a papel o a ordenador).
- Especificación de medidas para combatir el mal uso de las herramientas generativas de Inteligencia Artificial como Chat GPT.

Capítulo 3

Programación didáctica para el módulo Programación de Servicios y Procesos

3.1. Contextualización del entorno de aprendizaje

3.1.1. Datos generales del centro

El IES San Juan de la Rambla se sitúa en la C/ Adán Martín Menis, s/n. 38420, en el barrio de San José, perteneciente a la localidad de San Juan de la Rambla en la provincia de Santa Cruz de Tenerife. El centro lleva veinticuatro años de actividad y sus datos identificativos son los siguientes (IES San Juan de la Rambla, 2023a):

- **Teléfono de contacto**. +34 922 922 068
- **Código del centro**. 38011510
- **Correo electrónico de contacto**. 38011510@gobiernodecanarias.org
- **Página web del centro**. https://www3.gobiernodecanarias.org/medusa/edublog/iessanjuandelarambla/

3.1.2. Oferta educativa

En cuanto a la oferta educativa, en las Tablas 3.1 y 3.2 se detallan los distintos cursos que se ofertan tanto en las etapas de la ESO y Bachillerato como en la de Ciclo Formativo.

Etapa	**Cursos**
ESO (LOMLOE)	1º ESO 2º ESO 3º ESO 3º ESO - Diversificación 4º ESO 4º ESO - Diversificación
Bachillerato (LOMLOE)	1º Bachillerato 2º Bachillerato

Tabla 3.1: Oferta educativa de la ESO y Bachillerato (IES San Juan de la Rambla, 2023a)

Familia	**Ciclo**	**Nombre**	**Cursos**
Agraria	Grado Básico (LOMLOE)	CFGB Agro-jardinería y composiciones Florales	1º
		CFGB Aprovechamientos Florales	1º y 2º
	Grado Medio (LOE)	CFGM Aprovechamiento y Conservación del Medio Natural	1º y 2º
Informática y Comunicaciones	Grado Medio (LOE)	CFGM Sistemas Microinformáticos y Redes	1º y 2º
	Grado Superior (LOE)	CFGS Desarrollo de Aplicaciones Multiplataforma	1º y 2º

Tabla 3.2: Oferta educativa de ciclos formativos (IES San Juan de la Rambla, 2023a)

3.1.3. Contexto del centro

El IES San Juan de la Rambla está localizado en la zona norte de Tenerife, en el barrio de San José, perteneciente al municipio de San Juan de la Rambla. Los alumnos que recibe pertenecen al mismo municipio y además, gracias a la oferta de Formación Profesional que posee, a algunos otros municipios cercanos como La Guancha, Icod de Los Vinos, Los Realejos, Puerto de la Cruz y La Orotava. Según el Instituto Nacional de Estadística, en el año 2023 la cifra de habitantes censados en el municipio es de 4098 (2454 hombres y 2454 mujeres). (IES San Juan de la Rambla, 2023c)

La actividad económica que predomina en el municipio es la agricultura, especialmente el cultivo del plátano, así como las cooperativas. Dentro del mismo municipio apenas se genera empleo, por lo que la mayor parte de su población activa trabaja fuera del municipio en el sector servicios (hostelería y construcción), cuyo nivel de estudios apenas supera los primarios o carecen de ellos. A lo largo de los últimos años, se ha contemplado un incremento de las expectativas que tienen las familias con sus hijos en lo que a oportunidades laborales se refiere. (IES San Juan de la Rambla, 2023c)

El centro está catalogado como “centro de integración preferente para alumnado con necesidades educativas especiales por déficit motor”, teniendo servicios como rampas, barandillas, ascensor, etc. (IES San Juan de la Rambla, 2023c)

3.1.4. Organización del centro

Los horarios del centro se reflejan en la Tabla 3.3.

Actividad	**Horario**
Horario de apertura y cierre del centro	7:30 - 15:00
Horario lectivo	8:30 - 14:00
Duración de las sesiones	55 minutos
Recreo	10:45 - 11:15
Atención al público de secretaría	8:30 - 13:30

Tabla 3.3: Organización del centro

3.1.5. Objetivos generales del centro

Los principios por los que se rige el centro son los siguientes:

- "Educación en el respeto a los derechos y libertades fundamentales y en el ejercicio de la tolerancia y de la libertad dentro del centro.
- Integración y participación en las actividades colectivas e individuales en la consecución de un fin global.
- Consecución de un centro ecológico, respetuoso con el medio ambiente y sostenible. Adquisición de hábitos, técnicas, así como de conocimientos científicos, técnicos, humanísticos y estéticos que ayuden a su consecución, para su posterior aplicación a su entorno cercano." (IES San Juan de la Rambla, 2023c)

Para cumplir los principios mencionados anteriormente, el centro propone los siguientes objetivos:

- "Implementar dinámicas de grupo que remedien las carencias de los alumnos en las habilidades sociales esenciales para la convivencia y el rendimiento de los grupos.
- Resolución de problemas de disciplina.
- Mayor participación y conexión con las familias.
- Buscar diferentes opciones para mejorar el rendimiento en lo que influye, también, la actitud y la disciplina en clase.
- Mejorar el seguimiento individualizado en el hábito de trabajo en casa y en clase.
- Fomentar los hábitos del esfuerzo personal y del reconocimiento del trabajo.
- Fomentar el espíritu colaborativo entre toda la comunidad educativa.
- Establecer la colaboración con organizaciones municipales, regionales nacionales e internacionales, mediante la realización de acuerdos o convenios de colaboración o la realización de proyectos o programas bilaterales o multilaterales.
- Trabajar desde una perspectiva competencial, fomentando desde la ejecución de las programaciones didácticas la adquisición de las competencias básicas correspondientes a cada nivel." (IES San Juan de la Rambla, 2023c)

3.1.6. Estructura del centro

El centro cuenta con tres edificios. Primeramente, el edificio principal, que consta de tres plantas y es donde se imparten la mayoría de clases, se ofrecen casi todos los servicios y se llevan a cabo las tareas administrativas. En segundo lugar, el pabellón, en el cual se realizan todas las actividades deportivas, y por último el edificio de agrarias, donde se imparten todas las clases de los Ciclos Formativos de la familia de agraria. En la Tabla 3.4 se detallan los espacios y servicios disponibles, así como la localización de los mismos.

Espacio	**Cantidad**	**Localización**
Zona de docencia		
Aulas ordinarias (grupos y desdobles)	16	Edificio principal Edificio agrarias
Aulas de Informática/informatizadas	7	Edificio principal
Aulas de música, dramatización y audio	1	Edificio principal - Planta baja
Aulas de pequeño grupo (12 a 15 alumnos)	1	Edificio principal
Aulas de plástica y visual	1	Edificio principal - Planta alta
Aulas de música	1	Edificio principal - Planta baja
Aulas de PT	1	Edificio principal - Planta principal
Aulas de Tecnología	1	Edificio principal - Planta baja
Laboratorio de Ciencias Naturales	1	Edificio principal - Planta alta
Laboratorio de idiomas	1	Edificio principal - Planta baja
Laboratorio de Química	1	Edificio principal - Planta alta
Talleres específicos para Ciclos Formativos	4	Edificio principal Edificio agrarias
Aulas integradas en los talleres	2	Edificio principal (1) Edificio agrarias (1)
Zona de otros usos		
Departamentos	13	Edificio principal (11) Edificio agrarias (1) Pabellón (1)
Salas para otros usos	4	Edificio principal (3) Edificio agrarias (1)
Canchas polideportivas cubiertas	1	Pabellón
Vestuarios/Aseos de actividades deportivas	3	Pabellón
Aseos alumnos	6	Todos los edificios
Aseos alumnas	6	Todos los edificios
Aseos adaptados para alumnos con discapacidad	1	Edificio principal
Aseos profesores	2	Edificio principal - Planta principal
Salón de actos	1 (110 uds.)	Edificio principal - Planta baja
Biblioteca	1 (55 uds.)	Edificio principal - Planta principal
Zona administrativa		
Despacho Director	1	Edificio principal - Planta principal
Despacho Secretario/Administrador	1	Edificio principal - Planta principal
Despacho Vicedirector	1	Edificio principal - Planta principal
Despacho Jefatura de Estudios	1	Edificio principal - Planta principal
Secretaría	1	Edificio principal - Planta principal
Archivo	1	Edificio principal - Planta principal
Sala de Profesores	1	Edificio principal - Planta principal
Sala de visitantes y atención a familias	1	Edificio principal - Planta principal
Servicios		

Ascensor	1	Edificio principal
Depósito de agua	1	Edificio principal - Planta baja
Bomba de agua	1	Edificio principal - Planta baja
Vivienda subalterno/conserje	1	Exterior
Consejería/reprografía	1	Edificio principal - Planta principal
Trastero/depósito/almacén	3	Edificio principal - Planta baja
Cuarto de mantenimiento	1	Edificio principal - Planta baja
Cafetería	1	Edificio principal - Planta baja
Zona exterior		
Invernadero	3	Exterior
Otras zonas de expansión al aire libre	2 (500 m2)	Exterior
Porche cubierto	1 (230 m2)	Exterior
Patio de recreo al aire libre	1 (1000 m2)	Exterior
Número de edificios	3	Exterior

Tabla 3.4: Infraestructura del centro (IES San Juan de la Rambla, 2023b)

3.1.7. Características del alumnado

Al centro acuden 312 alumnos, distribuyéndose en 152 alumnos en la ESO, 67 en Bachillerato y 93 en Ciclos Formativos. En los estudios obligatorios, los estudiantes mayoritariamente son del municipio, sin embargo, en los postobligatorios suelen pertenecer a los municipios de los alrededores. Dependiendo de la etapa educativa, el alumnado posee distintas características (IES San Juan de la Rambla, 2023c):

- **ESO**. El tipo de familia más observado es el compuesto por los dos padres y los hijos, siendo no muy común la convivencia con abuelos. En relación a la ocupación laboral de los padres de los alumnos en esta etapa, la mayoría trabaja en los sectores de la construcción, hostelería y servicios. Es también importante recalcar que un gran número de ellos se encuentra en el paro. Por su parte, las madres trabajan sobre todo en el sector de la limpieza y la hostelería. De estos datos se contempla que la cualificación profesional de una gran parte de los padres es baja.

  Con respecto al rendimiento de los alumnos, este se ha visto incrementado. Se ha reducido el abandono escolar temprano y se han mejorado las notas obtenidas al titular. Existe un gran número de alumnos que acuden a clases particulares, destacando las asignaturas de Matemáticas, Física y Química e Inglés y otras actividades tales como la banda de música del municipio y el fútbol. Poseen pocos hábitos de estudio y las aficiones más destacadas son el salir con las amistades, el deporte y el pasar tiempo con los aparatos electrónicos como el ordenador y la TV.

- **Bachillerato**. Se manifiesta que los alumnos conviven con sus padres y hermanos. Cada familia tiene una media de 2/3 hijo y de la misma manera que ocurre con el alumnado de la ESO no es común el que los abuelos vivan con ellos. Hay un número destacado de familias monoparentales, sobre todo madres, a causa de separaciones o divorcios. Las profesiones más comunes de los padres son las del sector de hostelería, limpieza, construcción, amas

de casa y servicios. En relación a los hábitos de estudio, la estadística es la misma que para los alumnos de la ESO.

- **Ciclo Formativo Grado Básico**. La totalidad del alumnado son chicos, los cuales proceden de padres separados y el número de hermanos es de dos por familia. Las profesiones más típicas de los padres son las mismas que las de los padres del alumnado de la ESO. Recientemente, se ha visto un incremento de alumnado inmigrante matriculado en esta etapa educativa. La mayoría del alumnado ha repetido varios cursos a lo largo de su vida, demostrando de esta manera su dificultad para aprender. Las aficiones de ocio manifestadas más típicas son el deporte, salir, las motos, los caballos y la fiesta.

- **Ciclo Formativo Grado Medio y Superior**. La mayoría viven con sus dos padres y con un hermano. Hay una pequeña proporción de éstos que vive con abuelos o parientes cercanos. Las profesiones más comunes de los padres son las mismas que las mencionadas para el alumnado de la ESO y Bachillerato, distinguiéndose en que algunos padres tienen un nivel de estudios medio (bachillerato o carreras universitarias de ciclo corto). Los alumnos proceden de distintos municipios de toda la zona norte de la isla. La asistencia a clases particulares en esta etapa no es tan común. En relación a las actividades de ocio, suelen ser las mismas que las de los estudiantes de Bachillerato.

3.2. Datos de identificación del título

El título al que se hace referencia en esta programación didáctica se identifica con los siguientes datos (Ministerio de educación, formación profesional y deportes, 2010):

- **Denominación**. Desarrollo de Aplicaciones Multiplataforma.
- **Nivel**. Formación Profesional de Grado Superior.
- **Duración**. 2000 horas.
- **Familia profesional**. Informática y Comunicaciones.
- **Referente europeo**. Clasificación Internacional Normalizada de la Educación 2011 (CINE-11): Nivel 5B.
- **Marco legal**.
  - Real Decreto 450/2010, de 16 de abril, por el que se establece el título de Técnico Superior en Desarrollo de Aplicaciones Multiplataforma y se fijan sus enseñanzas mínimas (Ministerio de Educación, 2010b).
  - Orden EDU/2000/2010, de 13 de julio, por la que se establece el currículo del ciclo formativo de Grado Superior correspondiente al título de Técnico Superior en Desarrollo de Aplicaciones Multiplataforma (Ministerio de Educación, 2010a).

Es importante mencionar que en la Comunidad Autónoma de Canarias no existe ninguna adaptación curricular para este título.

3.3. Datos de identificación del módulo

- **Denominación**. Programación de Servicios y Procesos.
- **Módulo**. 0490.
- **Curso**. Segundo.
- **Duración**. Ochenta y cuatro horas repartidas en cuatro horas a la semana durante los dos primeros trimestres.

3.4. Justificación del módulo

La impartición de este módulo es importante debido a la gran cantidad de aplicaciones empresariales que necesitan ser diseñadas para ser ejecutadas por múltiples procesos simultáneamente. Asimismo, la comunicación efectiva entre dichos programas y la creación de un canal seguro de transmisión de datos también es vital. En este módulo se imparten todos estos conceptos.

3.5. Competencia general del título

"La competencia general de este título consiste en desarrollar, implantar, documentar y mantener aplicaciones informáticas multiplataforma, utilizando tecnologías y entornos de desarrollo específicos, garantizando el acceso a los datos de forma segura y cumpliendo los criterios de usabilidad y calidad exigidas en los estándares establecidos." (Ministerio de Educación, 2010b)

3.6. Competencias profesionales, personales y sociales

Las competencias profesionales, personales y sociales del título de Desarrollo de Aplicaciones Multiplataforma quedan reguladas por el Real Decreto 450/2010 de 16 de abril y son las siguientes:

a) "Configurar y explotar sistemas informáticos, adaptando la configuración lógica del sistema según las necesidades de uso y los criterios establecidos.

b) Aplicar técnicas y procedimientos relacionados con la seguridad en sistemas, servicios y aplicaciones, cumpliendo el plan de seguridad.

c) Gestionar bases de datos, interpretando su diseño lógico y verificando integridad, consistencia, seguridad y accesibilidad de los datos.

d) Gestionar entornos de desarrollo adaptando su configuración en cada caso para permitir el desarrollo y despliegue de aplicaciones.

e) Desarrollar aplicaciones multiplataforma con acceso a bases de datos utilizando lenguajes, librerías y herramientas adecuados a las especificaciones.

f) Desarrollar aplicaciones implementando un sistema completo de formularios e informes que permitan gestionar de forma integral la información almacenada.

g) Integrar contenidos gráficos y componentes multimedia en aplicaciones multiplataforma, empleando herramientas específicas y cumpliendo los requerimientos establecidos.

h) Desarrollar interfaces gráficos de usuario interactivos y con la usabilidad adecuada, empleando componentes visuales estándar o implementando componentes visuales específicos.

i) Participar en el desarrollo de juegos y aplicaciones en el ámbito del entretenimiento y la educación empleando técnicas, motores y entornos de desarrollo específicos.

j) Desarrollar aplicaciones para teléfonos, PDA y otros dispositivos móviles empleando técnicas y entornos de desarrollo específicos.

k) Crear ayudas generales y sensibles al contexto, empleando herramientas específicas e integrándolas en sus correspondientes aplicaciones.

l) Crear tutoriales, manuales de usuario, de instalación, de configuración y de administración, empleando herramientas específicas.

m) Empaquetar aplicaciones para su distribución preparando paquetes auto instalables con asistentes incorporados.

n) Desarrollar aplicaciones multiproceso y multihilo empleando librerías y técnicas de programación específicas.

ñ) Desarrollar aplicaciones capaces de ofrecer servicios en red empleando mecanismos de comunicación.

o) Participar en la implantación de sistemas ERP-CRM evaluando la utilidad de cada uno de sus módulos.

p) Gestionar la información almacenada en sistemas ERP-CRM garantizando su integridad.

q) Desarrollar componentes personalizados para un sistema ERP-CRM atendiendo a los requerimientos.

r) Realizar planes de pruebas verificando el funcionamiento de los componentes software desarrollados, según las especificaciones.

s) Desplegar y distribuir aplicaciones en distintos ámbitos de implantación verificando su comportamiento y realizando las modificaciones necesarias.

t) Establecer vías eficaces de relación profesional y comunicación con sus superiores, compañeros y subordinados, respetando la autonomía y competencias de las distintas personas.

u) Liderar situaciones colectivas que se puedan producir, mediando en conflictos personales y laborales, contribuyendo al establecimiento de un ambiente de trabajo agradable, actuando en todo momento de forma respetuosa y tolerante.

v) Gestionar su carrera profesional, analizando las oportunidades de empleo, autoempleo y de aprendizaje.

w) Mantener el espíritu de innovación y actualización en el ámbito de su trabajo para adaptarse a los cambios tecnológicos y organizativos de su entorno profesional.

x) Crear y gestionar una pequeña empresa, realizando un estudio de viabilidad de productos, de planificación de la producción y de comercialización.

y) Participar de forma activa en la vida económica, social y cultural, con una actitud crítica y responsable." (Ministerio de Educación, 2010b)

3.7. Objetivos generales del ciclo formativo

Las objetivos generales del título de Desarrollo de Aplicaciones Multiplataforma quedan regulados por el Real Decreto 450/2010 de 16 de abril y son los siguientes:

a) "Ajustar la configuración lógica del sistema analizando las necesidades y criterios establecidos para configurar y explotar sistemas informáticos.

b) Identificar las necesidades de seguridad analizando vulnerabilidades y verificando el plan preestablecido para aplicar técnicas y procedimientos relacionados con la seguridad en el sistema.

c) Interpretar el diseño lógico de bases de datos, analizando y cumpliendo las especificaciones relativas a su aplicación, para gestionar bases de datos.

d) Instalar y configurar módulos y complementos, evaluando su funcionalidad, para gestionar entornos de desarrollo.

e) Seleccionar y emplear lenguajes, herramientas y librerías, interpretando las especificaciones para desarrollar aplicaciones multiplataforma con acceso a bases de datos.

f) Gestionar la información almacenada, planificando e implementando sistemas de formularios e informes para desarrollar aplicaciones de gestión.

g) Seleccionar y utilizar herramientas específicas, lenguajes y librerías, evaluando sus posibilidades y siguiendo un manual de estilo, para manipular e integrar en aplicaciones multiplataforma contenidos gráficos y componentes multimedia.

h) Emplear herramientas de desarrollo, lenguajes y componentes visuales, siguiendo las especificaciones y verificando interactividad y usabilidad, para desarrollar interfaces gráficos de usuario en aplicaciones multiplataforma.

i) Seleccionar y emplear técnicas, motores y entornos de desarrollo, evaluando sus posibilidades, para participar en el desarrollo de juegos y aplicaciones en el ámbito del entretenimiento.

j) Seleccionar y emplear técnicas, lenguajes y entornos de desarrollo, evaluando sus posibilidades, para desarrollar aplicaciones en teléfonos, PDA y otros dispositivos móviles.

k) Valorar y emplear herramientas específicas, atendiendo a la estructura de los contenidos, para crear ayudas generales y sensibles al contexto.

l) Valorar y emplear herramientas específicas, atendiendo a la estructura de los contenidos, para crear tutoriales, manuales de usuario y otros documentos asociados a una aplicación.

m) Seleccionar y emplear técnicas y herramientas, evaluando la utilidad de los asistentes de instalación generados, para empaquetar aplicaciones.

n) Analizar y aplicar técnicas y librerías específicas, simulando diferentes escenarios, para desarrollar aplicaciones capaces de ofrecer servicios en red.

ñ) Analizar y aplicar técnicas y librerías de programación, evaluando su funcionalidad para desarrollar aplicaciones multiproceso y multihilo.

o) Reconocer la estructura de los sistemas ERP-CRM, identificando la utilidad de cada uno de sus módulos, para participar en su implantación.

p) Realizar consultas, analizando y evaluando su alcance, para gestionar la información almacenada en sistemas ERP-CRM.

q) Seleccionar y emplear lenguajes y herramientas, atendiendo a los requerimientos, para desarrollar componentes personalizados en sistemas ERP-CRM.

r) Verificar los componentes software desarrollados, analizando las especificaciones, para completar un plan de pruebas.

s) Establecer procedimientos, verificando su funcionalidad, para desplegar y distribuir aplicaciones.

t) Describir los roles de cada uno de los componentes del grupo de trabajo, identificando en cada caso la responsabilidad asociada, para establecer las relaciones profesionales más convenientes.

u) Identificar formas de intervención ante conflictos de tipo personal y laboral, teniendo en cuenta las decisiones más convenientes, para garantizar un entorno de trabajo satisfactorio.

v) Identificar y valorar las oportunidades de promoción profesional y de aprendizaje, analizando el contexto del sector, para elegir el itinerario laboral y formativo más conveniente.

w) Identificar los cambios tecnológicos, organizativos, económicos y laborales en su actividad, analizando sus implicaciones en el ámbito de trabajo, para mantener el espíritu de innovación.

x) Reconocer las oportunidades de negocio, identificando y analizando demandas del mercado para crear y gestionar una pequeña empresa.

y) Reconocer sus derechos y deberes como agente activo en la sociedad, analizando el marco legal que regula las condiciones sociales y laborales para participar como ciudadano democrático." (Ministerio de Educación, 2010b)

3.8. Resultados de aprendizaje y criterios de evaluación

Los resultados de aprendizaje y criterios de evaluación del título de Desarrollo de Aplicaciones Multiplataforma quedan regulados por el Real Decreto 450/2010 de 16 de abril y son los siguientes:

1. "Desarrolla aplicaciones compuestas por varios procesos reconociendo y aplicando principios de programación paralela. Criterios de evaluación:

- **a)** Se han reconocido las características de la programación concurrente y sus ámbitos de aplicación.
- **b)** Se han identificado las diferencias entre programación paralela y programación distribuida, sus ventajas e inconvenientes.
- **c)** Se han analizado las características de los procesos y de su ejecución por el sistema operativo.
- **d)** Se han caracterizado los hilos de ejecución y descrito su relación con los procesos.
- **e)** Se han utilizado clases para programar aplicaciones que crean subprocesos.
- **f)** Se han utilizado mecanismos para sincronizar y obtener el valor devuelto por los subprocesos iniciados.
- **g)** Se han desarrollado aplicaciones que gestionen y utilicen procesos para la ejecución de varias tareas en paralelo.
- **h)** Se han depurado y documentado las aplicaciones desarrolladas.

2. Desarrolla aplicaciones compuestas por varios hilos de ejecución analizando y aplicando librerías específicas del lenguaje de programación. Criterios de evaluación:

- **a)** Se han identificado situaciones en las que resulte útil la utilización de varios hilos en un programa.
- **b)** Se han reconocido los mecanismos para crear, iniciar y finalizar hilos.
- **c)** Se han programado aplicaciones que implementen varios hilos.
- **d)** Se han identificado los posibles estados de ejecución de un hilo y programado aplicaciones que los gestionen.
- **e)** Se han utilizado mecanismos para compartir información entre varios hilos de un mismo proceso.
- **f)** Se han desarrollado programas formados por varios hilos sincronizados mediante técnicas específicas.
- **g)** Se ha establecido y controlado la prioridad de cada uno de los hilos de ejecución.
- **h)** Se han depurado y documentado los programas desarrollados.

3. Programa mecanismos de comunicación en red empleando sockets y analizando el escenario de ejecución. Criterios de evaluación:

a) Se han identificado escenarios que precisan establecer comunicación en red entre varias aplicaciones.

b) Se han identificado los roles de cliente y de servidor y sus funciones asociadas.

c) Se han reconocido librerías y mecanismos del lenguaje de programación que permiten programar aplicaciones en red.

d) Se ha analizado el concepto de socket, sus tipos y características.

e) Se han utilizado sockets para programar una aplicación cliente que se comunique con un servidor.

f) Se ha desarrollado una aplicación servidor en red y verificado su funcionamiento.

g) Se han desarrollado aplicaciones que utilizan sockets para intercambiar información.

h) Se han utilizado hilos para implementar los procedimientos de las aplicaciones relativos a la comunicación en red.

4. Desarrolla aplicaciones que ofrecen servicios en red, utilizando librerías de clases y aplicando criterios de eficiencia y disponibilidad. Criterios de evaluación:

a) Se han analizado librerías que permitan implementar protocolos estándar de comunicación en red.

b) Se han programado clientes de protocolos estándar de comunicaciones y verificado su funcionamiento.

c) Se han desarrollado y probado servicios de comunicación en red.

d) Se han analizado los requerimientos necesarios para crear servicios capaces de gestionar varios clientes concurrentes.

e) Se han incorporado mecanismos para posibilitar la comunicación simultánea de varios clientes con el servicio.

f) Se ha verificado la disponibilidad del servicio.

g) Se han depurado y documentado las aplicaciones desarrolladas.

5. Protege las aplicaciones y los datos definiendo y aplicando criterios de seguridad en el acceso, almacenamiento y transmisión de la información. Criterios de evaluación:

a) Se han identificado y aplicado principios y prácticas de programación segura.

b) Se han analizado las principales técnicas y prácticas criptográficas.

c) Se han definido e implantado políticas de seguridad para limitar y controlar el acceso de los usuarios a las aplicaciones desarrolladas.

d) Se han utilizado esquemas de seguridad basados en roles.

e) Se han empleado algoritmos criptográficos para proteger el acceso a la información almacenada.

f) Se han identificado métodos para asegurar la información transmitida.

g) Se han desarrollado aplicaciones que utilicen sockets seguros para la transmisión de información.

h) Se han depurado y documentado las aplicaciones desarrolladas." (Ministerio de Educación, 2010b)

La ponderación de cada resultado de aprendizaje viene reflejada en la Tabla 3.5.

Resultado de aprendizaje (RA)	**Peso del RA en la nota final del módulo**
RA1	20 %
RA2	20 %
RA3	20 %
RA4	20 %
RA5	20 %

Tabla 3.5: Ponderación de los RA

3.9. Objetivos del módulo profesional

Los objetivos del título de Desarrollo de Aplicaciones Multiplataforma quedan regulados por el Real Decreto 450/2010 de 16 de abril y son los siguientes:

b) "Identificar las necesidades de seguridad analizando vulnerabilidades y verificando el plan preestablecido para aplicar técnicas y procedimientos relacionados con la seguridad en el sistema.

e) Seleccionar y emplear lenguajes, herramientas y librerías, interpretando las especificaciones para desarrollar aplicaciones multiplataforma con acceso a bases de datos.

i) Seleccionar y emplear técnicas, motores y entornos de desarrollo, evaluando sus posibilidades, para participar en el desarrollo de juegos y aplicaciones en el ámbito del entretenimiento.

j) Seleccionar y emplear técnicas, lenguajes y entornos de desarrollo, evaluando sus posibilidades, para desarrollar aplicaciones en teléfonos, PDA y otros dispositivos móviles.

l) Valorar y emplear herramientas específicas, atendiendo a la estructura de los contenidos, para crear tutoriales, manuales de usuario y otros documentos asociados a una aplicación.

n) Analizar y aplicar técnicas y librerías específicas, simulando diferentes escenarios, para desarrollar aplicaciones capaces de ofrecer servicios en red.

ñ) Analizar y aplicar técnicas y librerías de programación, evaluando su funcionalidad para desarrollar aplicaciones multiproceso y multihilo." (Ministerio de Educación, 2010b)

3.10. Contenidos básicos

Los contenidos básicos del título de Desarrollo de Aplicaciones Multiplataforma quedan regulados por el Real Decreto 450/2010 de 16 de abril y son los siguientes:

- "**Programación multiproceso.**
  - Ejecutables. Procesos. Servicios.
  - Estados de un proceso.
  - Hilos.
  - Programación concurrente.
  - Programación paralela y distribuida.
  - Comunicación entre procesos.
  - Gestión de procesos.
  - Sincronización entre procesos.
  - Programación de aplicaciones multiproceso.
- **Programación multihilo.**
  - Recursos compartidos por los hilos.
  - Estados de un hilo. Cambios de estado.
  - Elementos relacionados con la programación de hilos. Librerías y clases.
  - Gestión de hilos.
  - Sincronización de hilos.
  - Compartición de información entre hilos.
  - Programación de aplicaciones multihilo.
- **Programación de comunicaciones en red.**
  - Comunicación entre aplicaciones.
  - Roles cliente y servidor.
  - Elementos de programación de aplicaciones en red. Librerías.
  - Sockets.
  - Creación de sockets.
  - Enlazado y establecimiento de conexiones.
  - Utilización de sockets para la transmisión y recepción de información.
  - Programación de aplicaciones cliente y servidor.
  - Utilización de hilos en la programación de aplicaciones en red.
- **Generación de servicios en red.**
  - Protocolos estándar de comunicación en red a nivel de aplicación (telnet, ftp, http, pop3, smtp, entre otros).
  - Librerías de clases y componentes.
  - Utilización de objetos predefinidos.
  - Establecimiento y finalización de conexiones.

- Transmisión de información.
- Programación de aplicaciones cliente.
- Programación de servidores.
- Implementación de comunicaciones simultáneas.

▪ **Utilización de técnicas de programación segura.**

- Prácticas de programación segura.
- Criptografía de clave pública y clave privada.
- Principales aplicaciones de la criptografía.
- Protocolos criptográficos.
- Política de seguridad.
- Programación de mecanismos de control de acceso.
- Encriptación de información.
- Protocolos seguros de comunicaciones.
- Programación de aplicaciones con comunicaciones seguras." (Ministerio de Educación, 2010b)

3.11. Secuenciación y temporalización de las situaciones de aprendizaje

Los detalles de la secuenciación y temporalización de las situaciones de aprendizaje se encuentran en la Tabla 3.6. Adicionalmente, su duración y localización temporal están plasmadas en la Tabla 3.7, y la ponderación de cada situación de aprendizaje por cada resultado de aprendizaje está refejada en la Tabla 3.8.

Situación	**Nombre**	**Contenidos**
SA1	Programación multiproceso	- Ejecutables. Procesos. Servicios. - Estados de un proceso. - Programación concurrente. - Programación paralela y distribuida. - Comunicación entre procesos. - Gestión y sincronización de procesos. - Programación de aplicaciones multiproceso.
SA2	Programación multihilo	- Hilos. - Recursos compartidos por los hilos. - Gestión de estados de un hilo. - Librerías y clases para la programación de hilos. - Gestión y sincronización de hilos. - Compartición de información entre hilos. - Programación de aplicaciones multihilo.

SA3	Comunicación de aplicaciones en red	- Comunicación entre aplicaciones. - Roles cliente y servidor. - Librerías para la programación de aplicaciones en red. - Sockets. - Utilización de sockets para el enlazado y establecimiento de conexiones así como para la transmisión y recepción de información. - Programación de aplicaciones cliente y servidor. - Utilización de hilos en la programación de aplicaciones en red.
SA4	Servicios en red	- Protocolos estándar de comunicación en red a nivel de aplicación (telnet, ftp, http, pop3, smtp, etc.). - Librerías de clases y componentes. - Utilización de objetos predefinidos. - Establecimiento y finalización de conexiones. - Transmisión de información. - Programación de aplicaciones cliente. - Programación de aplicaciones servidor. - Implementación de comunicaciones simultáneas.
SA5	Desarrollo seguro de software	- Prácticas de programación segura. - Criptografía y sus principales aplicaciones - Criptografía de clave pública y clave privada. - Protocolos criptográficos. - Política de seguridad. - Programación de mecanismos de control de acceso. - Cifrado de la información. - Protocolos seguros de comunicaciones. - Programación de aplicaciones con comunicaciones seguras.

Tabla 3.6: Secuenciación de las situaciones de aprendizaje (Ministerio de Educación, 2010b)

Situación	**Nombre**	**Trimestre**	**Horas**	**Semanas**
SA1	Programación multiproceso	1º	18	4,5
SA2	Programación multihilo	1º	16	4
SA3	Programación de comunicaciones en red	1º	16	4
SA4	Generación de servicios en red	2º	18	4,5
SA5	Desarrollo seguro de software	2º	16	4

Tabla 3.7: Duración y localización temporal de las situaciones de aprendizaje

RA/SA	SA1	SA2	SA3	SA4	SA5	Total
RA1	100 %					100 %
RA2		100 %				100 %
RA3			100 %			100 %
RA4				100 %		100 %
RA5					100 %	100 %
% SA en RA	20 %	20 %	20 %	20 %	20 %	100 %

Tabla 3.8: Ponderación de las SA en relación al logro de los RA

3.12. Atención a la diversidad

Considerando la diversidad de alumnado que puede haber en el aula, todas las situaciones de aprendizaje se adaptarán al entorno socioeconómico, cultural y educativo. Por su parte, los contenidos no serán únicamente conceptuales sino procedimentales para adaptarse a cada tipo de alumno, presentándose de tal manera que lo explicado esté interrelacionado entre sí teniendo en cuenta el conocimiento previo del alumnado y antes de introducir nuevos contenidos se hará un breve repaso de los contenidos previos.

Se tiene en cuenta distinta tipología de alumnado:

- Que proviene de pruebas de acceso.
- Que han estudiado durante una cantidad considerable de años por estar trabajando simultáneamente.
- De distinta cultura.
- Que tiene una incorporación tardía al aula.
- Con algún tipo de discapacidad, aplicándose las siguientes medidas:
    - **Discapacidad motora**. Se les ubicará lo más cerca posible a los accesos al aula. Las pruebas de evaluación serán adaptadas, siendo éstas realizadas a través de ordenadores o de forma oral para que dicho alumnado sea capaz de conseguir los objetivos del curso planteados.
    - **Discapacidad auditiva**. Se les ubicará lo más cerca posible al profesor y se les facilitará material visual detallado.
    - **Transtorno por déficit de atención de hiperactividad**. Se priorizará el tener a este tipo de alumnado supervisado y situado lo más alejado posible de distracciones, animarlo a participar en clase y no marcarle tareas muy extensas asegurándose de que tiene todo el material disponible para su lectura.
    - **Discapacidad visual**. Se les ubicará en una zona del aula cercana tanto del profesor como de la pizarra y se utilizarán tamaños de letra superiores al normal para reducir sus dificultades para leer.
    - **Retraso madurativo**. Se les pedirá llegar a metas más pequeñas para llegar al objetivo final y se tratará de fomentar el trabajo en equipo para servirles de apoyo.

3.13. Metodología u orientaciones metodológicas

3.13.1. Principios generales de actuación metodológica

Los principios de actuación metodológica que se consideran para elaborar todas las situaciones de aprendizaje son los siguientes:

- La necesidad de partir de un conocimiento previo a la hora de impartir conocimientos nuevos.
- Darles a los alumnos frecuentemente un informe de seguimiento de las actividades realizadas y evaluaciones con el fin de que sean conscientes de su proceso de aprendizaje.
- Incentivar la iniciativa, la autonomía y el trabajo en grupo.
- Correcto uso de las TICs para buscar información y aumentar conocimientos adquiridos.
- Crear un clima de aula en el que los alumnos se sientan cómodos y con confianza. Esto fomentará su participación activa.
- Adquisición de una visión global de todos los contenidos impartidos con el fin de que sean capaces de indagar más en un futuro y saber el porqué de las cosas.

3.13.2. Modelos de enseñanza

Los modelos de enseñanza que se emplearán para impartir las situaciones de aprendizaje se exponen a continuación:

- **Enseñanza directiva**. El profesor impartirá los conceptos teóricos, luego realizará una actividad práctica con los estudiantes en la cual apliquen dichos conocimientos, y posteriormente los alumnos realizarán otra actividad de manera autónoma.
- **Investigación grupal**. Tras realizar un trabajo práctico de forma independiente, los alumnos colaborarán de manera conjunta para llevar a cabo un proyecto más complejo, aportando los conocimientos que cada uno ha adquirido.
- **Simulación**. Se simulará el entorno real de una empresa, donde los alumnos tendrán que, no solo desarrollar una aplicación que resuelva un problema, sino también documentarla, hacer un código legible y mantenible, cumplir con los requisitos del cliente, etc.
- **Inductivo**. Se partirá siempre de diferentes casos concretos con el objetivo de que luego los alumnos sean capaces de generalizarlos.
- **Indagación científica**. Se aprende practicando, por lo que los alumnos deberán formularse preguntas, responderlas con una hipótesis desarrollada, experimentar e investigar más sobre ella y con los resultados obtenidos extraer conclusiones.

3.13.3. Agrupamientos

Teniendo en cuenta las habilidades a desarrollar en un ciclo formativo técnico, es de vital importancia agrupar a los alumnos de manera adecuada. Por ello se proponen los siguientes agrupamientos:

- **Grupos heterogéneos**. Este tipo de grupos es ideal porque busca que sus miembros tengan perfiles o habilidades diferentes. Consecuentemente, el grupo se complementará mejor y alcanzará el objetivo final de manera más eficiente. Por ejemplo, un grupo de desarrolladores backend con desarrolladores frontend.
- **Grupos móviles**. También es crucial que los grupos cambien a lo largo del año para aprender a trabajar con diferentes tipos de personas y adaptarse a nuevas situaciones. En las empresas, no siempre es posible trabajar con quienes se tiene más afinidad.
- **Trabajo individual**. Procesar individualmente la información antes de trabajar en grupo también es importante para tener los conceptos claros y tener consistencia cuando se explica un tema.
- **Gran grupo**. Apropiado cuando se explican contenidos durante una sesión teórica.

3.13.4. Espacio

El espacio donde se llevarán a cabo las sesiones será una clase en la que habrá un ordenador por cada dos alumnos con dos sistemas operativos instalados, Windows y GNU/Linux. Se iniciará sesión en los equipos con las credenciales que el Centro proporcionará a los alumnos. Así mismo, en el aula también habrá una pizarra, y un proyector con el objetivo de que todo el alumnado vea la información que está explicando el docente.

Adicionalmente, los alumnos también desempeñarán actividades en las empresas donde realicen el módulo FCT (Formación en Centros de Trabajo).

3.14. Procedimientos e instrumentos de evaluación

Los aspectos relacionados con la evaluación del alumnado se detallan a continuación:

- **Evaluación**. Los diferentes tipos de evaluación a aplicar en este módulo son:
  - **Evaluación inicial**. Se hará a principio de curso con el objetivo de conocer los conocimientos previos que poseen los alumnos antes de estudiar nuevos conceptos. Esto permite adaptar el contenido del comienzo del curso a las necesidades grupales, así como motivar a los estudiantes conociendo los aspectos que deben repasar.
  - **Evaluación continua**. Se pretende hacer un seguimiento del alumnado, durante todo el curso, más preciso y realizar una media sobre los resultados adquiridos para obtener una puntuación final. De esta manera también se conocerá mejor el aprendizaje de cada alumno.

  - **Evaluación final**. También se realizará una prueba final al terminar el curso con el fin de saber si el alumnado tiene los conocimientos necesarios que se han impartido a lo largo del módulo.

- **Pérdida del derecho a la evaluación continua**. Si se da el caso de que el alumnado supere el 20 % de faltas sin justificar de un módulo profesional, se perderá el derecho a la evaluación continua y los criterios de evaluación cambiarán a los siguientes:

  - **Realización y entrega de todos los trabajos y prácticas realizadas durante el curso**. 30 %.
  - **Control teórico-práctico sobre los contenidos de las situaciones de aprendizaje de todo el curso.** 70 %.

  Para aprobar el módulo profesional, es necesario aprobar cada una de las actividades de los dos bloques mencionados.

- **Convocatorias**. Se dispone de cuatro convocatorias para aprobar cada uno de los módulos profesionales, exceptuando la FCT que se dispondrá de dos.

- **Seguimiento del alumnado**. Con el objetivo de hacer un seguimiento del alumnado se le harán distintas pruebas de evaluación:

  - **Laboratorios**. Resolución de problemas académicos de manera guiada con el objetivo de facilitar el aprendizaje de los contenidos explicados en las clases.
  - **Proyectos prácticos**. Resolución de problemas de situaciones cercanas a la realidad que se realizarán por parejas.
  - **Exámenes escritos**. Exámenes a papel que evalúan las bases teórico-prácticas que hay que tener para proceder a realizar los proyectos prácticos. Es importante recalcar que las partes del examen en las cuales haya que programar se harán a ordenador sin una conectividad a Internet.

- **Criterios de calificación**

  Para superar los módulos profesionales del ciclo formativo, los alumnos tienen que haber obtenido una nota igual o mayor a cinco en todos los resultados de aprendizaje (RA) asociados al módulo pertinente. Cada uno de los resultados de aprendizaje, tendrá un peso concreto a la hora de alcanzar los objetivos del módulo y cada una de las situaciones de aprendizaje (SA) contribuirá ponderadamente a la adquisición del RA o RAs que están asociados. Cada situación de aprendizaje también tiene actividades con un porcentaje específico.

  La nota final del módulo profesional corresponderá a la media ponderada de las calificaciones en cada uno de los RA asociados siempre y cuando sean todas iguales o mayores a cinco.

  En caso de que se detecte plagio en cualquier tipo de evaluación (laboratorios, prácticas o pruebas escritas), ello implicará el suspenso de la situación de aprendizaje correspondiente.

- **Ponderación de los instrumentos de evaluación**

- **Laboratorios**. 20 %.
- **Proyectos prácticos**. 40 %.
- **Exámenes escritos**. 40 % siendo necesario obtener un 3,5 como mínimo.

Se establecerá una medida de control sobre el uso de herramientas de Inteligencia Artificial como Chat GPT con el objetivo de combatir sus distintas consecuencias negativas, y que cierta parte del alumnado presente trabajos que no son de obra propia o no han entendido aspectos necesarios de los mismos. Para ello, se citará a cada alumno individualmente para pedirle que explique detalladamente el trabajo que ha entregado (sea este el código de un programa o un trabajo de investigación).

- **Recuperación**

  En cada evaluación, se podrá realizar una recuperación en la cual cada alumno se examinará de los resultados de aprendizaje no superados. La nota máxima con la que se calificará dicha recuperación será de cinco con el fin de diferenciar entre el alumnado que supera los RA en el periodo ordinario y el que no.

3.15. Evaluación de la programación docente

Con el objetivo de saber cuánto de exitosa ha sido la programación docente durante todo el curso, se realizarán distintas acciones:

- **Encuestas anónimas**. Se les pedirá a los estudiantes que rellenen unas encuestas anónimas sobre distintos aspectos de la docencia, pidiéndoles también ciertas sugerencias para mejorar para próximos años.
- **Memoria final**. Se especificarán los objetivos que se han logrado y los que no, la participación de los alumnos y sus propuestas de mejora. Se determinarán también el análisis de resultados de evaluaciones y actividades que se han hecho, así como una evaluación personal del propio docente sobre cómo de eficaz ha sido la programación.
- **Variaciones con respecto a lo programado**. Se prestará especial atención a la variación de tiempos programados y los reales para cada situación de aprendizaje, así como de las actividades, laboratorios y prácticas propuestas. Se analizarán las causas de estas variaciones.
- **Conclusiones de la docencia**. Se observará la participación que tienen los estudiantes, así como el dinamismo presente en el aula. De esta manera se identificarán tanto los puntos fuertes como los débiles de la docencia del curso académico. Por cada uno de estos aspectos anteriormente mencionados, se realizará un informe.
- **Análisis de resultados**. Se observará con detenimiento los resultados de todas las pruebas de evaluación que se han realizado identificando los aspectos más sencillos y complicados del curso para los estudiantes. Se comparará los resultados con los objetivos que inicialmente se plantearon y se ajustará la programación para un próximo año teniendo en cuenta estos resultados.

3.16. Actividades extraescolares y complementarias

Las actividades propuestas destinadas al alumnado de los ciclos formativos de la familia de Informática y Comunicaciones son:

- La importancia de las soft skills en el mundo laboral a parte de las habilidades técnicas.
- Taller de ciberseguridad. Conociendo cómo atacan los cibercriminales y cómo defenderse.
- Jornada de visita a una empresa tecnológica de desarrollo de software.
- Visita guiada al ITER. Analizando la infraestructura del Teide-HPC y su funcionamiento.

Por otra parte, las actividades planteadas para todo el centro son las siguientes:

- Buenas ciber prácticas. Evitando ser víctimas de un ataque de ingeniería social.
- Cómo dar un buen uso a las redes sociales y saber qué tipo de contenido publicar en Internet. Consecuencias de la sobre exposición en Internet.

Capítulo 4

Situación de aprendizaje: Desarrollo seguro de software

4.1. Justificación

La Situación de Aprendizaje que he elegido para desarrollar es la SA5 denominada "Desarrollo seguro de software". Personalmente, considero que esta situación de aprendizaje es de vital importancia a pesar de no enseñar a los alumnos a programar los servicios en red que se estudian en las otras. Esto es debido a que de nada vale desarrollar una muy buena aplicación si no se le aplica principios de seguridad en cada etapa de su desarrollo, ya que luego tanto la empresa que crea la aplicación como los usuarios finales podrán sufrir las graves consecuencias de los ciberataques que cada año son más numerosos. Adicionalmente, tengo formación especializada en ciberseguridad por un máster que cursé y, más concretamente, en esta situación de aprendizaje que he elegido para desarrollar en este trabajo.

En la Tabla 4.1, se encuentra la información más relevante para el desarrollo de esta situación de aprendizaje.

Unidad	**Horas impartidas**	**Sesiones**	**Porcentaje de evaluación**
SA5: Desarrollo seguro de software	16	10 (6 x 2h + 4 x 1h)[1]	20 %

Tabla 4.1: Información de la Situación de Aprendizaje Desarrollo seguro de software

4.2. Objetivos didácticos

De los objetivos mencionados en la Sección 3.9, el que se alcanza con esta situación de aprendizaje, es el siguiente:

b) "Identificar las necesidades de seguridad analizando vulnerabilidades y verificando el plan preestablecido para aplicar técnicas y procedimientos relacionados con la seguridad en el sistema." (Ministerio de Educación, 2010b)

[1]Diez sesiones en total. Seis sesiones de dos horas y cuatro sesiones de una hora.

4.3. Contenidos

Tal y como se explicó en la Sección 3.11, los contenidos de esta situación de aprendizaje son los siguientes:

- "Prácticas de programación segura.
- Criptografía y sus principales aplicaciones
- Criptografía de clave pública y clave privada.
- Protocolos criptográficos.
- Política de seguridad.
- Programación de mecanismos de control de acceso.
- Cifrado de la información.
- Protocolos seguros de comunicaciones.
- Programación de aplicaciones con comunicaciones seguras." (Ministerio de Educación, 2010b)

4.4. Temporalización

La temporalización y organización de cada sesión están detalladas en la Tabla 4.2. Es necesario mencionar que todas las sesiones de teoría y actividades a las que se hace referencia están expuestas en los Apéndices A y B. Dicho material es de elaboración propia y la mayoría del mismo lo utilicé durante mi periodo de prácticas en el centro.

Sesión	**Ocupación**	**Tiempo (minutos | horas)**	
1	**Actividad 1**. Con el fin de saber qué conocimiento previo tiene el alumnado y adaptarse a ello para que las clases sean más productivas, se realizará un cuestionario habilitado en el aula virtual con diez preguntas de respuesta corta para resolverlas en quince minutos.	15	2
	Teoría. Introducción a la ciberseguridad. Razones por las que es tan importante actualmente y en que campos se aplica. Elementos de la ciberseguridad (software, hardware, usuarios y datos). Triángulo CIA. Principios de la criptografía para proteger la información. Contextualización. El nacimiento de la criptografía. Buenas prácticas de seguridad en desarrollo de software (manejo de excepciones, mensajes de error, validaciones, autenticación y roles, logs, sanitización de entradas y control de acceso, etc.).	90	

	Actividad 2. Se realizará una breve investigación sobre las técnicas de sanitización de entradas que se utilizan en los lenguajes de programación más importantes así como las maneras de restringir el acceso a usuarios en una aplicación.	15	
2	**Actividad 2**. Continuación de la Actividad 2.	15	2
	Teoría. Funciones hash. Propiedades y sus aplicaciones. Diferencia entre codificar y cifrar. Ejemplos de algoritmos de codificación. Introducción a la herramienta CyberChef. Cifrado simétrico. Ventajas e incovenientes. Tipos de cifradores simétricos (de flujo, bloque o híbridos) o (de sustitución o transposición). Ejemplos de cada uno y sus modos de operación.	45	
	Actividad 3. Los alumnos inspeccionarán y ejecutarán el código de dos ficheros en los que se trabaja con funciones hash y cifrado por sustitución.	15	
	Teoría. Algoritmo DES y AES.	15	
	Actividad 4. Los alumnos inspeccionarán y ejecutarán el código de distintos ficheros en los que se utilizan los algoritmos DES y AES en distintos modos de operación sobre texto y sobre un fichero.	30	
3	**Actividad 5**. Se hará uso de la herramienta Cryptool para afianzar los conocimientos criptográficos estudiados hasta el momento y ser conscientes de los aspectos más importantes al configurar un cifrador.	30	1
	Teoría. Cifrado asimétrico. Resolución del problema de distribución de la clave que había en el cifrado simétrico. Propiedades de la clave privada y la clave pública de cada parte de la comunicación. Infraestructura de clave pública PKI, certificados digitales y Autoridades de Certificación. Algoritmo RSA.	30	
4	**Actividad 6**. Los alumnos inspeccionarán y ejecutarán el código de distintos ficheros en los cuales se usa el algoritmo RSA para generar claves (privadas y públicas) y se cifra y descifra un fichero.	30	2
	Teoría. Firma digital y diferencias que tiene respecto a la firma manual. Algoritmos para firmar digitalmente. Proceso independiente de firmar un fichero y proceso de firma de un fichero en conjunto con el cifrado.	15	
	Actividad 7. Los alumnos inspeccionarán y ejecutarán el código de varios ficheros en donde se utiliza el algoritmo ElGammal para generar claves (privada y pública), firmar un mensaje y verificar la identidad de la firma.	30	
	Teoría. Protocolos SSL/TLS. Diferencias entre ellos y sus aplicaciones prácticas.	15	
	Actividad 8. Los alumnos inspeccionarán y ejecutarán el código de un servidor y de un cliente utilizando sockets seguros. Para ello generarán certificados SSL ejecutando una serie de comandos que se les va a proporcionar.	30	

5	**Actividad 9**. Se realizará una actividad de generación de certificados en donde habrá que diseñar una estructura PKI, asignar distintos tipos de roles y generar los distintos certificados con la herramienta OpenSSL.	45	1
	Teoría. Detalles finales. Características de los sistemas cifradores actuales y ataques de fuerza bruta. Computación cuántica y sus consecuencias sobre la seguridad y la criptografía.	15	
6	**Actividad 10**. Se investigará sobre las diferencias entre el cifrado clásico y el cuántico explicando las propiedades de cada uno.	120	2
7	**Actividad 11**. Se realizará un programa que muestre un menú que permita al usuario final realizar distintas operaciones con el algoritmo MD5 para verificar ficheros. Asimismo, se hará un documento explicando los pasos seguidos a la hora de realizar el programa.	120	2
8	**Actividad 11**. Continuación de la Actividad 11.	60	1
9	**Actividad 12**. Se realizará un programa en el que se use el algoritmo DES para transmitir de forma segura información desde un cliente a un servidor (usando el protocolo TCP y el de UDP). Asimismo, se hará un documento explicando los pasos seguidos a la hora de realizar el programa.	120	2
10	**Actividad 12**. Continuación de la Actividad 12.	60	1

Tabla 4.2: Sesiones de la Situación de Aprendizaje 5

4.5. Actividades

Todas las partes de las sesiones expuestas en la Tabla 4.2 que no son teóricas, es decir, las actividades, están expuestas en mayor detalle en las Tablas 4.3, 4.4, 4.5, 4.6, 4.7, 4.8, 4.9, 4.10 4.11.

Sesión. 1	**Duración**. 15 min	**Ubicación**. Aula de informática
Tipo de actividad. Conocimientos previos.	**Estrategias cognitivas**. Investigación previa, habilidades adquiridas	**Agrupamiento**. Individual.
Recursos. Ordenador, conexión a internet y aula virtual.		
Desarrollo. El docente plantea en la clase el realizar un cuestionario inicial que se encontrará en el aula virtual. Serán 10 preguntas de respuesta corta.		
Objetivos. Conocer el nivel de conocimiento que tiene el alumnado con esta situación de aprendizaje.		
Competencia. ✗		
Evaluación. Esta actividad no es evaluable.		

Tabla 4.3: Actividad 1. Cuestionario inicial

Sesión. 1, 2	**Duración**. 30 min	**Ubicación**. Aula de informática
Tipo de actividad. Desarrollo de contenidos.	**Estrategias cognitivas**. Análisis, comprensión, elaboración, síntesis.	**Agrupamiento**. Individual.
Recursos. Ordenador y conexión a internet.		
Desarrollo. Se solicita al alumnado que se descargue el enunciado correspondiente a esta actividad, que investigue acerca de lo pedido y que lo sintetice en un documento para entregar.		
Objetivos. Tomar consciencia de que en el desarrollo de software es tan importante aplicar técnicas que eviten futuros ataques, como que la aplicación funcione correctamente.		
Competencia. **b)** "Aplicar técnicas y procedimientos relacionados con la seguridad en sistemas, servicios y aplicaciones, cumpliendo el plan de seguridad." (Ministerio de Educación, 2010b)		
Evaluación. Esta actividad se evaluará con la entrega de un informe en el que se explique lo pedido en su enunciado.		

Tabla 4.4: Actividad 2. Sanitización de las entradas y control de acceso

Sesión. 2, 2, 4, 4	**Duración**. 15, 30, 30, 30 min	**Ubicación**. Aula de informática
Tipo de actividad. Situación de aprendizaje.	**Estrategias cognitivas**. Comprensión, abstracción, aplicación.	**Agrupamiento**. Individual.
Recursos. Ordenador y conexión a internet.		
Desarrollo. Se les comenta a los alumnos que analicen los ficheros de código fuente de cada actividad, que busquen el funcionamiento de cada una de las librerías usadas en su correspondiente documentación en Internet y que cambien valores de las variables en el código fuente para ver como se procesa la información.		
Objetivos. Saber como funcionan las funciones hash, los distintos tipos de cifrado y los algoritmos de firma.		
Competencia. **b)** "Aplicar técnicas y procedimientos relacionados con la seguridad en sistemas, servicios y aplicaciones, cumpliendo el plan de seguridad." (Ministerio de Educación, 2010b)		
Evaluación. Estas actividades no son evaluables.		

Tabla 4.5: Actividades 3, 4, 6, 7. Funciones hash y cifrado por sustitución, cifrado DES y AES, cifrado RSA, firma RSA

Sesión. 3	**Duración**. 30 min	**Ubicación**. Aula de informática
Tipo de actividad. Situación de aprendizaje.	**Estrategias cognitivas**. Comprensión, aplicación.	**Agrupamiento**. Parejas de 2 personas.
Recursos. Ordenador, conexión a internet y herramienta Cryptool.		
Desarrollo. Se solicitará la instalación de la herramienta Cryptool, investigar todas sus opciones y realizar lo que se pide en el enunciado.		
Objetivos. Tener una visualización de los algoritmos estudiados hasta el momento para entender mejor su funcionamiento.		
Competencia. **b)** "Aplicar técnicas y procedimientos relacionados con la seguridad en sistemas, servicios y aplicaciones, cumpliendo el plan de seguridad." (Ministerio de Educación, 2010b)		
Evaluación. Esta actividad no es evaluable.		

Tabla 4.6: Actividad 5. Herramienta Cryptool

Sesión. 4	**Duración**. 30 min	**Ubicación**. Aula de informática
Tipo de actividad. Situación de aprendizaje.	**Estrategias cognitivas**. Comprensión, abstracción, aplicación.	**Agrupamiento**. Individual.
Recursos. Ordenador y conexión a internet.		
Desarrollo. Se les comenta a los alumnos que analicen los ficheros de código fuente de la actividad, que busquen el funcionamiento de cada una de las librerías usadas en su correspondiente documentación en Internet y que cambien valores de las variables en el código fuente para ver como se procesa la información. Para ejecutar el código, necesitarán generar un certificado con la herramienta OpenSSL.		
Objetivos. Conocer cómo añadirle una capa de seguridad a los sockets, así como generar certificados SSL para luego suministrárselos a dichos sockets.		
Competencias. **b)** "Aplicar técnicas y procedimientos relacionados con la seguridad en sistemas, servicios y aplicaciones, cumpliendo el plan de seguridad. **ñ)** Desarrollar aplicaciones capaces de ofrecer servicios en red empleando mecanismos de comunicación." (Ministerio de Educación, 2010b)		
Evaluación. Esta actividad no es evaluable.		

Tabla 4.7: Actividad 8. Sockets SSL

Sesión. 5	**Duración**. 45 min	**Ubicación**. Aula de informática
Tipo de actividad. Situación de aprendizaje.	**Estrategias cognitivas**. Comprensión, abstracción, aplicación.	**Agrupamiento**. Parejas de 2 personas.
Recursos. Ordenador, conexión a internet y herramienta draw.io.		
Desarrollo. Se pide leer detenidamente el problema de diseño que han de resolver y explicar cómo lo han hecho en un documento.		
Objetivos. Saber reconocer roles en una aplicación, diseñar una infraestructura PKI y adquirir agilidad con la herramienta OpenSSL.		
Competencias. **b)** "Aplicar técnicas y procedimientos relacionados con la seguridad en sistemas, servicios y aplicaciones, cumpliendo el plan de seguridad. **t)** Establecer vías eficaces de relación profesional y comunicación con sus superiores, compañeros y subordinados, respetando la autonomía y competencias de las distintas personas." (Ministerio de Educación, 2010b)		
Evaluación. Esta actividad se evaluará con la entrega de un informe en el que se explique lo pedido en su enunciado.		

Tabla 4.8: Actividad 9. Generación de certificados PKI

Sesión. 6	**Duración**. 120 min	**Ubicación**. Aula de informática
Tipo de actividad. Desarrollo de contenidos.	**Estrategias cognitivas**. Análisis, comprensión, elaboración, síntesis.	**Agrupamiento**. Individual.
Recursos. Ordenador y conexión a internet.		
Desarrollo. Se solicita al alumnado que se descargue el enunciado correspondiente a esta actividad, que investigue acerca de lo pedido y que lo sintetice en un documento para entregar.		
Objetivos. Conocer algoritmos de cifrado cuántico y entender por qué son más seguros que los clásicos.		
Competencias. **b)** "Aplicar técnicas y procedimientos relacionados con la seguridad en sistemas, servicios y aplicaciones, cumpliendo el plan de seguridad. **w)** Mantener el espíritu de innovación y actualización en el ámbito de su trabajo para adaptarse a los cambios tecnológicos y organizativos de su entorno profesional." (Ministerio de Educación, 2010b)		
Evaluación. Esta actividad se evaluará con la entrega de un informe donde se explique lo pedido en su enunciado.		

Tabla 4.9: Actividad 10. Cifrado cuántico

Sesión. 7, 8	**Duración**. 180 min	**Ubicación**. Aula de informática
Tipo de actividad. Situación de aprendizaje.	**Estrategias cognitivas**. Comprensión, abstracción, aplicación.	**Agrupamiento**. Parejas de 2 personas.
Recursos. Ordenador y conexión a internet.		
Desarrollo. Se solicita leer las funciones del programa que se pide, discutir con la pareja asignada la mejor manera de realizar dichas funciones e implementar la solución.		
Objetivos. Saber integrar funciones hash en una aplicación.		
Competencias. **b)** "Aplicar técnicas y procedimientos relacionados con la seguridad en sistemas, servicios y aplicaciones, cumpliendo el plan de seguridad. **ñ)** Desarrollar aplicaciones capaces de ofrecer servicios en red empleando mecanismos de comunicación. **t)** Establecer vías eficaces de relación profesional y comunicación con sus superiores, compañeros y subordinados, respetando la autonomía y competencias de las distintas personas." (Ministerio de Educación, 2010b)		
Evaluación. Esta actividad se evaluará con la entrega del código que realice lo que se solicita en el enunciado y el correspondiente informe en el que se detallen todas las decisiones y pasos realizados.		

Tabla 4.10: Actividad 11. Uso de MD5 para verificación de ficheros

Sesión. 9, 10	**Duración**. 180 min	**Ubicación**. Aula de informática
Tipo de actividad. Situación de aprendizaje.	**Estrategias cognitivas**. Comprensión, abstracción, aplicación.	**Agrupamiento**. Parejas de 2 personas.
Recursos. Ordenador y conexión a internet.		
Desarrollo. Se solicita leer las funciones del programa que se pide, discutir con la pareja asignada la mejor manera de realizar dichas funciones e implementar la solución.		
Objetivos. Saber integrar algoritmos de cifrado en una aplicación.		
Competencias. **b)** "Aplicar técnicas y procedimientos relacionados con la seguridad en sistemas, servicios y aplicaciones, cumpliendo el plan de seguridad. **ñ)** Desarrollar aplicaciones capaces de ofrecer servicios en red empleando mecanismos de comunicación. **t)** Establecer vías eficaces de relación profesional y comunicación con sus superiores, compañeros y subordinados, respetando la autonomía y competencias de las distintas personas." (Ministerio de Educación, 2010b)		
Evaluación. Esta actividad se evaluará con la entrega del código que realice lo que se solicita en el enunciado y el correspondiente informe en donde se detallen todas las desiciones y pasos realizados.		

Tabla 4.11: Actividad 12. Transmisión segura de información con DES

4.6. Evaluación

Tal y como se expuso en la Tabla 3.8 y en la Sección 3.8, los Criterios de Evaluación correspondientes de esta Situación de Aprendizaje son los siguientes:

a) "Se han identificado y aplicado principios y prácticas de programación segura.

b) Se han analizado las principales técnicas y prácticas criptográficas.

c) Se han definido e implantado políticas de seguridad para limitar y controlar el acceso de los usuarios a las aplicaciones desarrolladas.

d) Se han utilizado esquemas de seguridad basados en roles.

e) Se han empleado algoritmos criptográficos para proteger el acceso a la información almacenada.

f) Se han identificado métodos para asegurar la información transmitida.

g) Se han desarrollado aplicaciones que utilicen sockets seguros para la transmisión de información.

h) Se han depurado y documentado las aplicaciones desarrolladas." (Ministerio de Educación, 2010b)

En la Tabla 4.12 se detallan las herramientas con las que se van a evaluar cada actividad y su porcentaje de evaluación, así como los criterios de evaluación evaluados en cada actividad.

Actividad	**Herramientas y % de evaluación**	**CE evaluados del RA5**
1	No evaluable	✗
2	Informe. 7,5 %	a, c
3	No evaluable	b, e
4	No evaluable	b, e
5	No evaluable	b, e
6	No evaluable	b, e
7	No evaluable	b, e
8	No evaluable	d
9	Informe. 15 %	d
10	Informe. 7,5 %	f
11	Código e informe. 35 %	e, f, h
12	Código e informe. 35 %	g, h

Tabla 4.12: Evaluación de las actividades

En las Tablas 4.13 y 4.14 se encuentran las rúbricas de evaluación para la entrega de prácticas de código e informe y prácticas de solamente informe respectivamente.

% Funcionamiento del programa	**Documentación**	**Nota**
No entregado		0
Hasta el 20 %	No realizada	1
	Realizada	2
Hasta el 35 %	No realizada	3
	Realizada	4
Hasta el 50 %	No realizada	5
	Realizada	6
Hasta el 75 %	No realizada	7
	Realizada	8
Hasta el 100 %	No realizada	9
	Realizada	10

Tabla 4.13: Rúbrica prácticas de código e informe

Aspecto	**Insuficiente**	**Mejorable**	**Satisfactorio**	**Excelente**	**Total**
Organización y estructura	0	3,33	6,66	10	**10**
Justificación de la solución	0	3,33	6,66	10	**10**
Calidad de la información	0	3,33	6,66	10	**10**
Profundización de la investigación	0	3,33	6,66	10	**10**

Tabla 4.14: Rúbrica prácticas de informe

Capítulo 5

Conclusiones

Tras haber realizado la programación didáctica de este Trabajo de Fin de Máster, he sido mucho más consciente de su gran importancia para un docente. Sin esta planificación previa, no se podría impartir clases de las distintas asignaturas satisfactoriamente debido a los imprevistos que puedan surgir obligando a realizar modificaciones sobre la planificación inicial. Después de tener esto claro, una de los aspectos más importantes es realizar la temporalización de una programación didáctica de la manera más realista posible.

Algo muy interesante que he aprendido también es la importancia de tener en cuenta el contexto del entorno de aprendizaje a la hora de hacer la programación didáctica. Hay que analizar el nivel cultural, económico y social de la zonas geográficas de las que proceden los estudiantes del centro para saber a qué tipo de alumnado se le va a impartir clase y adaptar la manera de transmitir los contenidos.

Me ha parecido de vital consideración la realización de las prácticas en el centro para el que va dirigida la programación didáctica de este Trabajo de Fin de Máster, y haber vivido de primera mano la puesta en práctica tanto de la situación de aprendizaje propuesta como algunos otros aspectos considerados en la programación. Así mismo, también me ha proporcionado una visión más real y amplia de lo que es el día a día de un docente.

Previamente a la realización de este máster, tenía una cierta vocación docente, la cual se ha visto muy intensificada después de haber asumido el rol de un docente tras mi paso por el centro en el que he realizado mis prácticas. Considero que es una profesión que permite ayudar a otras personas a conducir su vida, no solo profesional sino también personalmente.

Bibliografía

Departamento de Informática y Comunicaciones del IES San Juan de la Rambla. (2023a). Material CFGS Desarrollo de Aplicaciones Multiplataforma PGV 2023-2024.

Departamento de Informática y Comunicaciones del IES San Juan de la Rambla. (2023b). Programación Didáctica CFGS Desarrollo de Aplicaciones Multiplataforma PGV 2023-2024.

IES San Juan de la Rambla. (2023a). IES San Juan de la Rambla. https://www3.gobiernodecanarias.org/medusa/edublog/iessanjuandelarambla/

IES San Juan de la Rambla. (2023b). Programación General Anual. https://www3.gobiernodecanarias.org/medusa/edublog/iessanjuandelarambla/wp-content/uploads/sites/477/2024/02/iessanjuandelarambla_pga_23-24_290224.pdf

IES San Juan de la Rambla. (2023c). Proyecto Educativo de Centro. https://www3.gobiernodecanarias.org/medusa/edublog/iessanjuandelarambla/wp-content/uploads/sites/477/2022/06/iessjr_pe_20-21.pdf

Ministerio de Educación. (2010a). Orden EDU/2000/2010, de 13 de julio, por la que se establece el currículo del ciclo formativo de Grado Superior correspondiente al título de Técnico Superior en Desarrollo de Aplicaciones Multiplataforma. https://www.boe.es/boe/dias/2010/07/26/pdfs/BOE-A-2010-11888.pdf

Ministerio de Educación. (2010b). Real Decreto 450/2010, de 16 de abril, por el que se establece el título de Técnico Superior en Desarrollo de Aplicaciones Multiplataforma y se fijan sus enseñanzas mínimas. https://www.boe.es/boe/dias/2010/05/20/pdfs/BOE-A-2010-8067.pdf

Ministerio de educación, formación profesional y deportes. (2010). Técnico Superior en Desarrollo de Aplicaciones Multiplataforma. https://todofp.es/que-estudiar/loe/informatica-comunicaciones/des-aplicaciones-multiplataforma.html

Apéndice A

Material didáctico para la SA Desarrollo seguro de software

Figura A.1: SA5 - Diapositiva 1

Figura A.2: SA5 - Diapositiva 2

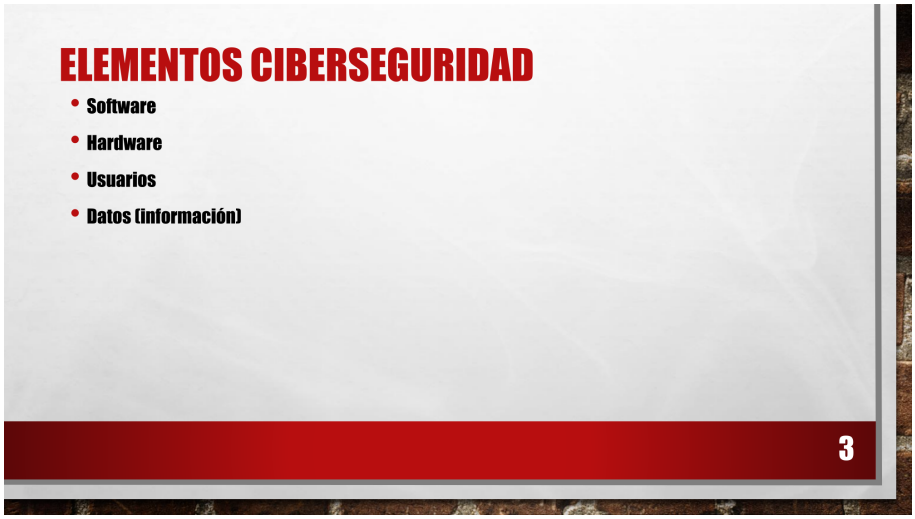


Figura A.3: SA5 - Diapositiva 3

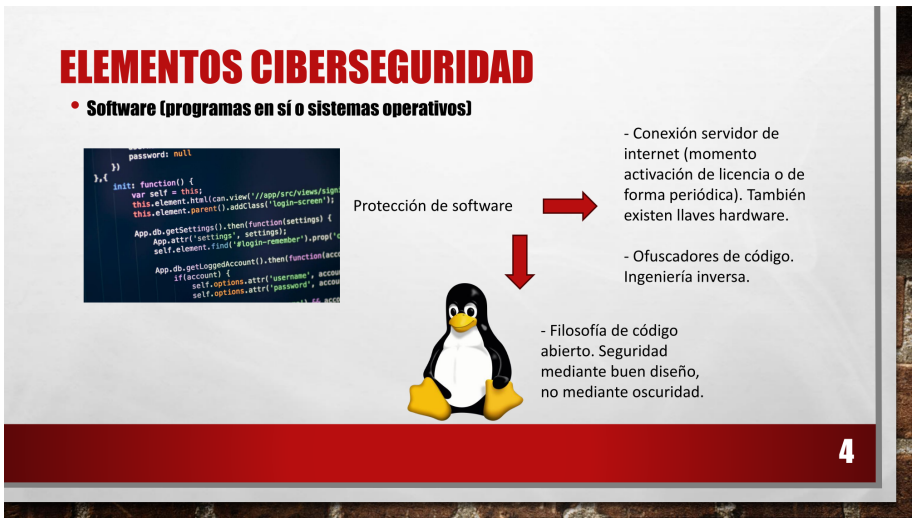


Figura A.4: SA5 - Diapositiva 4

Figura A.5: SA5 - Diapositiva 5

Figura A.6: SA5 - Diapositiva 6

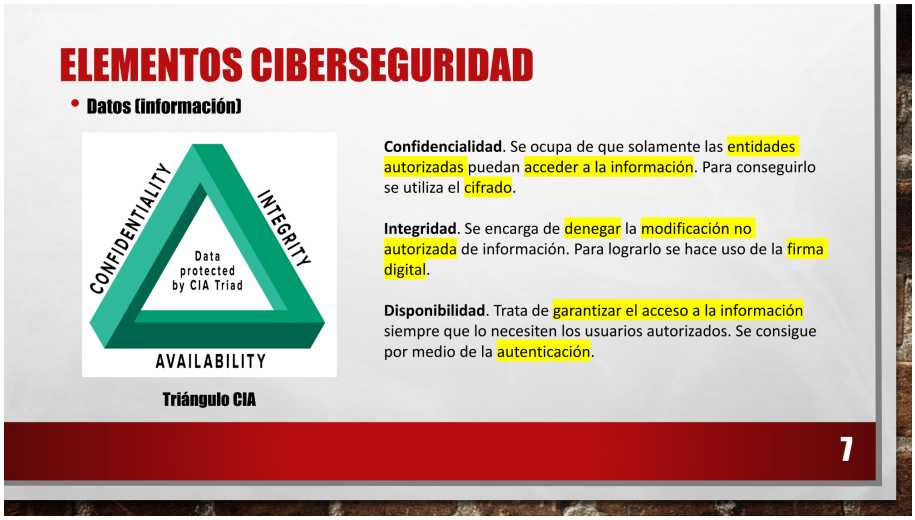


Figura A.7: SA5 - Diapositiva 7

Figura A.8: SA5 - Diapositiva 8

Figura A.9: SA5 - Diapositiva 9

Figura A.10: SA5 - Diapositiva 10

Figura A.11: SA5 - Diapositiva 11

Figura A.12: SA5 - Diapositiva 12

Figura A.13: SA5 - Diapositiva 13

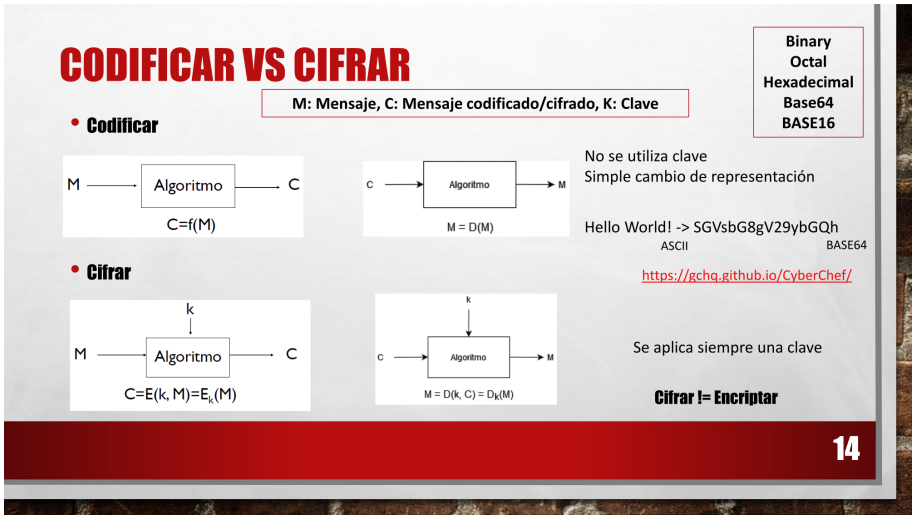


Figura A.14: SA5 - Diapositiva 14

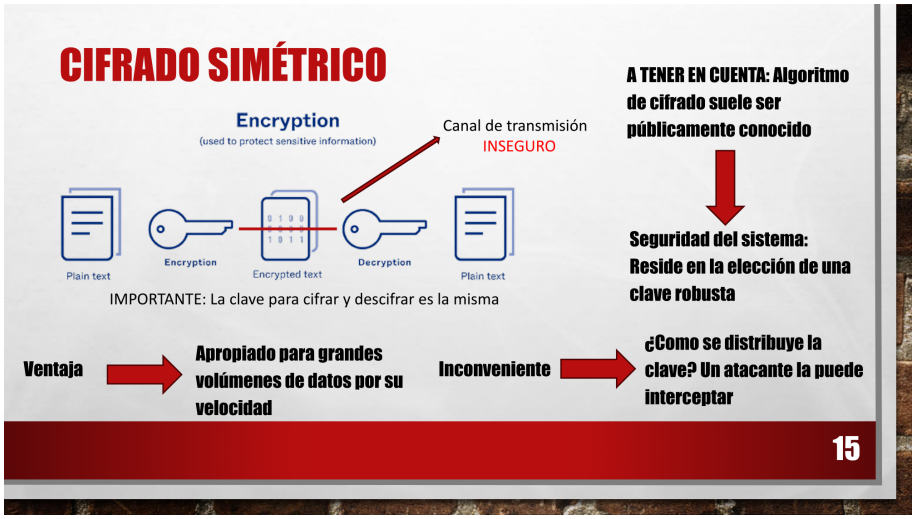


Figura A.15: SA5 - Diapositiva 15

Figura A.16: SA5 - Diapositiva 16

Figura A.17: SA5 - Diapositiva 17

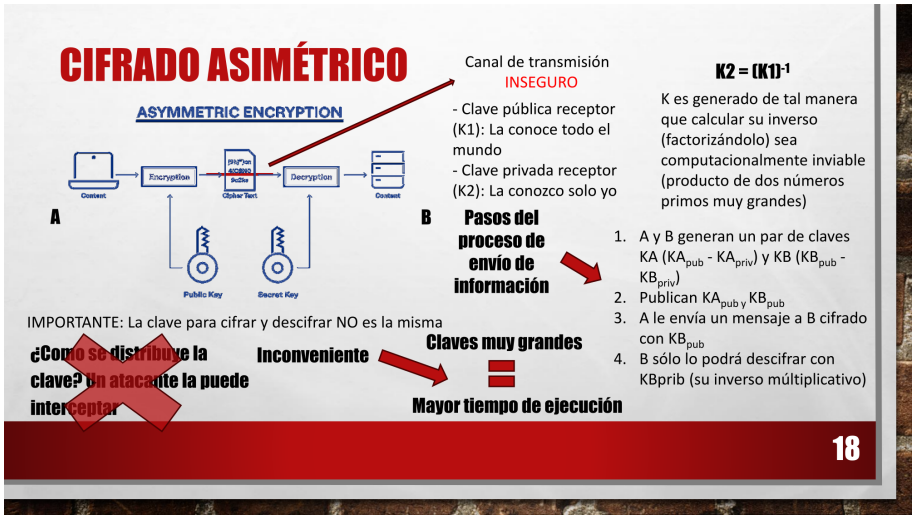


Figura A.18: SA5 - Diapositiva 18

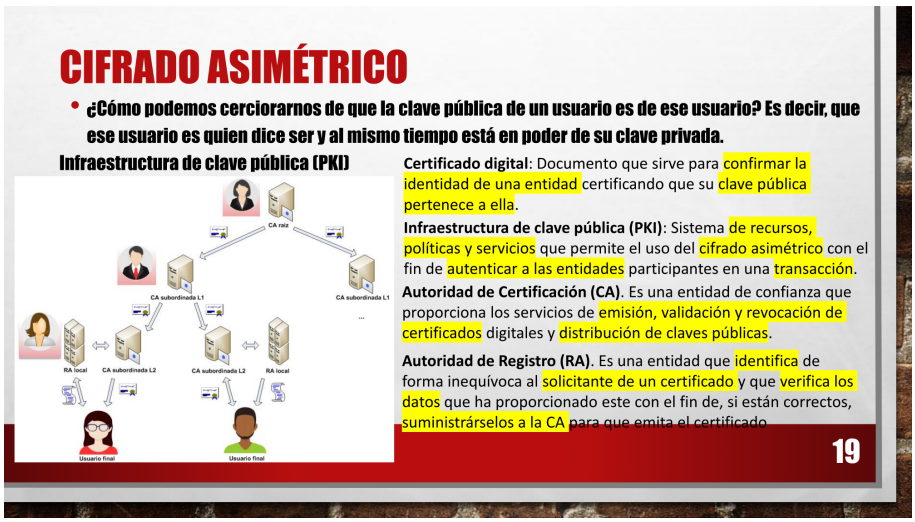


Figura A.19: SA5 - Diapositiva 19

Figura A.20: SA5 - Diapositiva 20

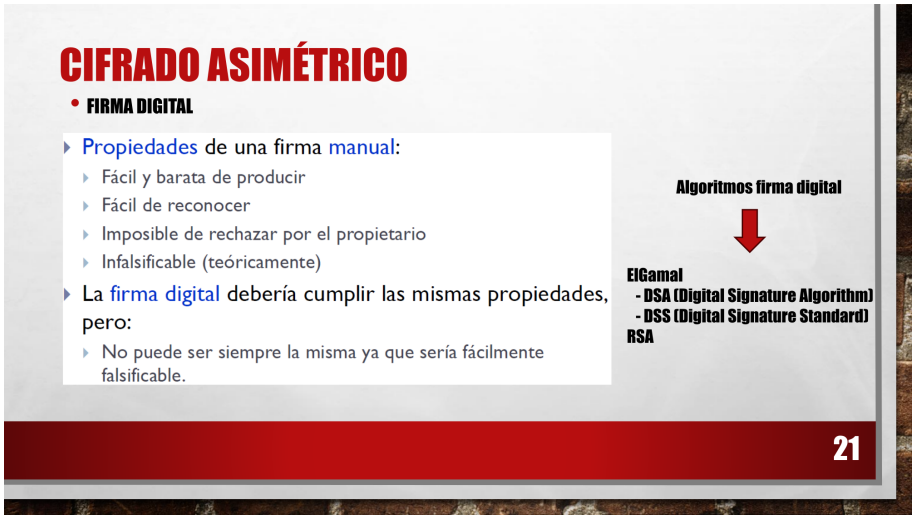


Figura A.21: SA5 - Diapositiva 21

Figura A.22: SA5 - Diapositiva 22

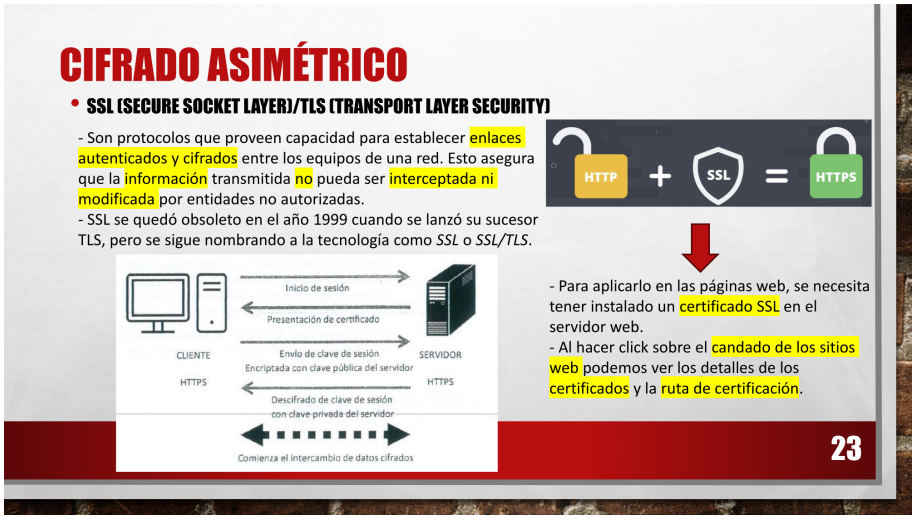


Figura A.23: SA5 - Diapositiva 23

Figura A.24: SA5 - Diapositiva 24

Figura A.25: SA5 - Diapositiva 25

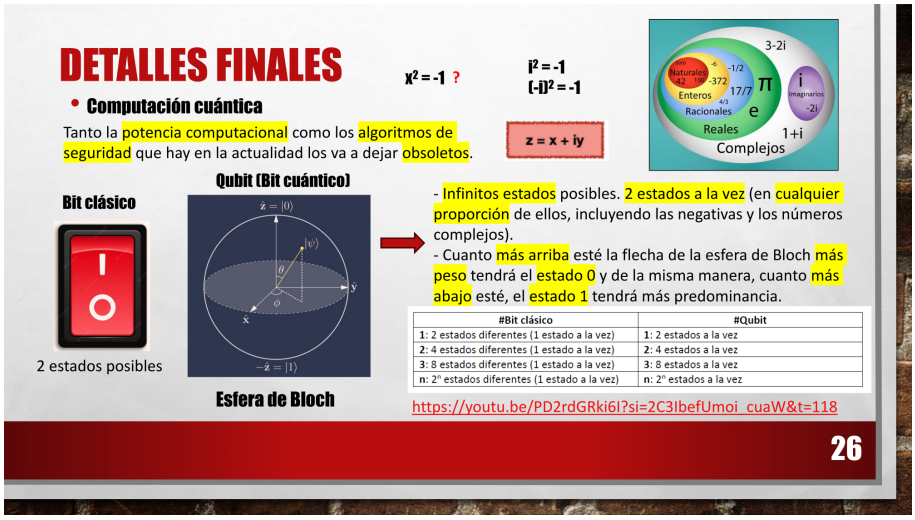


Figura A.26: SA5 - Diapositiva 26

Apéndice B

Propuesta de actividades para la SA Desarrollo seguro de software

Actividad 1. Cuestionario inicial

Esta actividad consiste en un cuestionario inicial no evaluable de respuestas cortas (máximo una línea por pregunta) con el objetivo de conocer el nivel de conocimiento previo del alumnado:

1. ¿Qué es la criptografía? ¿Qué es el tríangulo CIA y en qué principios se basa?
2. ¿Qué es el software libre? ¿Qué ventajas tiene este sobre el software comercial y privado?
3. ¿Qué son los ofuscadores de código? ¿En qué consiste el proceso de ingeniería inversa?
4. ¿Qué es la sanitización de entradas en programación? Especifica alguna técnica.
5. ¿Es lo mismo codificar que cifrar?
6. ¿Qué algoritmos de cifrado conoces? ¿Cuál es el más seguro?
7. ¿En qué se diferencia el protocolo HTTPS de HTTP? ¿Por qué el primero es más seguro?
8. ¿Qué son los ataques de fuerza bruta? Pon algún ejemplo.
9. ¿En qué se diferencia un ordenador cuántico de uno clásico?
10. ¿Las validaciones en un programa se deben de realizar en el cliente o en el servidor?

Actividad 2. Sanitización de las entradas y control de acceso

Redacta un informe explicando las principales técnicas de sanitización de entradas en los distintos lenguajes de programación detallando las vulnerabilidades que puede conllevar el no aplicar estas técnicas. También explica las maneras que existen de restringir el acceso a determinados usuarios en cualquier tipo de plataforma.

Actividad 3. Funciones hash y cifrado por sustitución

Los ficheros de código de la Actividad 3 se corresponden con los mostrados en los Listados B.1 y B.2.

```
import hashlib
import os

dir_path = os.path.dirname(os.path.realpath(__file__))

#hash com md5 de un texto
Texto = "Texto de ddddde".encode("utf-8")
result = hashlib.md5(Texto).hexdigest()
print("El hash de %s es: %s" % (Texto , result))

#hash com md5 de un texto
filename = input("Nombre de fichero: ")
with open(dir_path + "/" + filename,"rb") as f:
  bytes = f.read() # read file as bytes
  readable_hash = hashlib.md5(bytes).hexdigest();
  print("El hash del fichero: %s es:\n%s" % (filename , readable_hash))
```

Listado B.1: ej01-HASH-MD5.py (Departamento de Informática y Comunicaciones del IES San Juan de la Rambla, 2023a)

```
alfabeto = 'ABCDEFGHIJKLMNÑOPQRSTUVWXYZ0123456789'
alfabetoCifrado = 'KLMNÑOPQRSTUVWXYZ0123456789ABCDEFGHIJ'

def cifrar(mensaje):
  mensajeCifrado= ""
  mensaje = mensaje.upper()
  for caracter in mensaje:
      if caracter in alfabeto:
          index = alfabeto.index(caracter)
          mensajeCifrado = mensajeCifrado + alfabetoCifrado[index]
      else:
          mensajeCifrado = mensajeCifrado + caracter
  return mensajeCifrado

def descifrar (mensajeCifrado):
  mensajeDescifrado= ""
  for caracter in mensajeCifrado:
      if caracter in alfabeto:
          index = alfabetoCifrado.index(caracter)
          mensajeDescifrado +=  alfabeto[index]
      else:
          mensajeDescifrado +=  caracter
  return (mensajeDescifrado)

print (cifrar('I love you!'))
print (descifrar('R UY5Ñ 8Y4!'))
```

Listado B.2: ej02-CifradoCaseroSust.py (Departamento de Informática y Comunicaciones del IES San Juan de la Rambla, 2023a)

Actividad 4. Cifrado DES y AES

Los ficheros de código de la Actividad 4 se corresponden con los mostrados en los Listados B.3, B.4, B.5, B.6 y B.7.

```
# si la librería no está instalada se ejecutar el siguiente comando
#
# pip install pycryptodome


from Crypto.Cipher import DES
import base64
import os

mensajeOriginal = "Visita GALICIA: el paraíso".encode("utf-8")
print ("Mensaje original:", mensajeOriginal.decode("utf-8"))

key = b"abc123.." #establecemos una clave
iv = os.urandom(8) #generamos aleatoriamente un iv

#instanciamos un nuevo objeto DES
cipher = DES.new(key,  DES.MODE_OFB, iv=iv)
#ciframos los datos
bytesCifrados =  cipher.encrypt(mensajeOriginal)
print ("Bytes cifrados: ", bytesCifrados)
#para imprimir una mejor representación
mensajeCifrado = base64.b64encode(bytesCifrados).decode("utf-8")
print ("Mensaje Cifrado:", mensajeCifrado)

#es necesario un nuevo objeto para descifrar
cipher = DES.new(key, DES.MODE_OFB,iv = iv)
#desciframos usando la misma key e iv
mensajeDescifrado = cipher.decrypt(bytesCifrados)
print ("Mensaje: ", mensajeDescifrado.decode())
```

Listado B.3: ej03-DES-Key-IV.py (Departamento de Informática y Comunicaciones del IES San Juan de la Rambla, 2023a)

```
# si la librería no está instalada se ejecutar el siguiente comando
#
# pip install pycryptodome

from Crypto.Cipher import DES3
from Crypto.Random import get_random_bytes
import base64
import os

mensajeOriginal = "Visita GALICIA: el paraíso".encode("utf-8")
print ("Mensaja original:", mensajeOriginal.decode("utf-8"))

while True:
  try:
    key = DES3.adjust_key_parity(get_random_bytes(24))
    break
```

```
17    except ValueError:                                    #Si la clave no es de
          16 o de 24 bytes
18      pass
19  iv = os.urandom(8) #generamos aleatoriamente un iv
20
21  #instanciamos un nuevo objeto DES
22  cipher = DES3.new(key,  DES3.MODE_CFB, iv=iv)
23
24  #ciframos los datos
25  bytesCifrados =  cipher.encrypt(mensajeOriginal)
26  print ("Bytes cifrados: ", bytesCifrados)
27
28  #para imprimir una mejor representación
29  mensajeCifrado = base64.b64encode(bytesCifrados).decode("utf-8")
30  print ("Mensaje Cifrado:", mensajeCifrado)
31
32  #es necesario un nuevo objeto para descifrar
33  cipher = DES3.new(key, DES3.MODE_CFB, iv=iv)
34
35  #desciframos usando la misma key e iv
36  mensajeDescifrado = cipher.decrypt(bytesCifrados)
37  print ("Mensaje: ", mensajeDescifrado.decode())
```

Listado B.4: ej04-3DES.py (Departamento de Informática y Comunicaciones del IES San Juan de la Rambla, 2023a)

```
1   import os
2
3   from Crypto.Cipher import DES3
4   from hashlib import md5
5
6   def cifrar(fileOrigen, fileDestino, key):
7     cipher = DES3.new(key, DES3.MODE_EAX, nonce=b'0')
8     with open(fileOrigen, 'rb') as input_file:
9       file_bytes = input_file.read()
10      enc_file_bytes = cipher.encrypt(file_bytes)
11
12    with open(fileDestino, 'wb') as output_file:
13        output_file.write(enc_file_bytes)
14
15  def descifrar(fileOrigen, fileDestino, key):
16    cipher = DES3.new(key, DES3.MODE_EAX, nonce=b'0')
17    with open(fileOrigen, 'rb') as input_file:
18      file_bytes = input_file.read()
19      dec_file_bytes = cipher.decrypt(file_bytes)
20
21    with open(fileDestino, 'wb') as output_file:
22        output_file.write(dec_file_bytes)
23
24  key = "abc123."
25
26  key_hash = md5(key.encode('ascii')).digest() # 16-byte key
27  tdes_key = DES3.adjust_key_parity(key_hash)
28
29  dir_path = os.path.dirname(os.path.realpath(__file__))
```

```
30
31  cifrar(dir_path + "/textoPlano.txt", dir_path + "/textoCifrado.txt", tdes_key)
32  descifrar(dir_path + "/textoCifrado.txt", dir_path + "/textoDescifrado.txt",
        tdes_key)
```

Listado B.5: ej05-DES3-File.py (Departamento de Informática y Comunicaciones del IES San Juan de la Rambla, 2023a)

```
1   from Crypto. Cipher import AES
2
3   mensajeOriginal = b"Hola, algoritmo AES"
4   key =  b'Clave de 16 Byte'
5   print ("Original: ", mensajeOriginal)
6
7   #cifrar
8   cipher = AES.new(key, AES.MODE_EAX) #sin iv
9   mensajeCifrado, tag = cipher.encrypt_and_digest(mensajeOriginal)
                              #Tag para la autenticacion
10  print ("Cifrado: ", mensajeCifrado)
11
12  #descifrar
13  cipher = AES.new (key, AES. MODE_CFB)
14  #mensajeDescifrado = cipher.decrypt(mensajeCifrado)
15
16  mensajeDescifrado = cipher.decrypt(mensajeCifrado)
17
18
19  print ("Descifrado: ", mensajeDescifrado)
20
21
22  # try:
23  #     cipher.verify (tag)
24  #     print ("Descifrado: ", mensajeDescifrado)
25  # except ValueError:
26  #     print ("Clave incorrecta o mensaje corrupto")
27
28
29  #mensajeDescifrado = mensajeDescifrado + b"sd"
30
31  assert(mensajeDescifrado == mensajeOriginal)
32
33  # El uso de assert en Python nos permite realizar comprobaciones.
34  # Si la expresión contenida dentro del mismo es False,
35  # se lanzará una excepción, concretamente AssertionError.
```

Listado B.6: ej06-AES.py (Departamento de Informática y Comunicaciones del IES San Juan de la Rambla, 2023a)

```
1   from Crypto. Cipher import AES
2   from Crypto. Random import get_random_bytes
3
4   key = get_random_bytes(32) # Use a stored / generated key
5   MensajeOriginal = "Prueba de cifrado con AES+key+iv"  # This is your data
6   print ("Mensaje original: ", MensajeOriginal)
```

```
7
8   # Convertir un string a un objeto de bytes codificado UNICODE
9   data = MensajeOriginal.encode('utf-8')
10
11  # Cifrado
12  cipher_encrypt = AES.new(key, AES.MODE_CFB)
13  BytesCifrados = cipher_encrypt.encrypt(data)
14
15  # Muestro datos y vector de inicialización
16  iv = cipher_encrypt.iv
17  MensajeCifrado = BytesCifrados
18  print("\nMensaje cifrado: ", MensajeCifrado)
19  print("\nvector de inicializacion: ", iv)
20
21  # Descifrado
22  cipher_decrypt = AES.new(key, AES.MODE_CFB, iv=iv)
23  BytesDescifrados = cipher_decrypt.decrypt(MensajeCifrado)
24
25  # Conversión de bytes a string
26  MensajeDescifrado = BytesDescifrados.decode('utf-8')
27  print ("\nMensaje descifrado: ", MensajeDescifrado)
28
29  # probamos coindidencia original-cifrado-descifrado
30
31  assert MensajeOriginal == MensajeDescifrado, 'El mensaje original no coincide con
        la cifrado-descifrado'
```

Listado B.7: ej07-AESeIV-automatico.py (Departamento de Informática y Comunicaciones del IES San Juan de la Rambla, 2023a)

Actividad 5. Herramienta Cryptool

Esta actividad consistirá en el uso de la herramienta Cryptool, la cual permite familiarizarse con conceptos critográficos de manera visual. Para ello se harán dos ejercicios:

- **Ejercicio 1**. Teniendo en cuenta el siguiente texto:

  "A la atención de NOMBRE APELLIDO1 APELLIDO2,
  Nueva calificación en el expediente académico:
  NOTA en la asignatura PGV".

  Abre la herramienta Cryptool y utiliza el cifrado César sobre el texto empleando claves diferentes y las configuraciones explicadas a continuación:

  a) Haciendo solamente uso de las letras del alfabeto (minúsculas y mayúsculas), describe lo que sucede si se especifica como claves "a", "A" y "Z" y cómo funciona este algoritmo de cifrado.

  b) Detalla lo que sucede cuando se aplica una configuración de este tipo (solo letras minúsculas y mayúsculas).

- **Ejercicio 2**. Considerando el siguiente texto:

(1) This is an arbitrary text used to show cryptographic principles.
(2) The first five prime numbers are the following: 2, 3, 5, 7, 11.
(3) Co-prime numbers are those numbers that only have the number 1 as a common factor. For instance, 4 and 7 are coprimes.

Investiga cómo funciona el cifrado de Playfair para posteriormente cifrar y descifrar el texto usando el algoritmo mencionado con la clave "ABCDEF", usando matrices de tamaño 5x5 y 6x6.

a) ¿Qué ocurre con el mensaje una vez descifrado de nuevo? Describe la diferencia que hay entre el ejecutar el algoritmo con la matriz de tamaño 5x5 y el ejecutarlo con la matriz de 6x6.

b) Especifica todos los nuevos caracteres que aparecen.

Actividad 6. Cifrado RSA

Los ficheros de código de la Actividad 6 se corresponden con los mostrados en los Listados B.8, B.9, B.10 y B.11.

```
1  from Crypto.PublicKey import RSA
2  import os
3
4  dir_path = os.path.dirname(os.path.realpath(__file__))
5
6  codigoClave = "Unguessable" #contraseña
7  key = RSA.generate(2048)
8
9  #obtener clave privada
10 encrypted_key = key.export_key(passphrase=codigoClave, pkcs=8, protection="
       scryptAndAES128-CBC")        #Obtiene clave privada cifrada
11 file_out = open(dir_path + "/rsa_key_privada.bin", "wb")
12 file_out.write(encrypted_key)
13 file_out.close()
14 print("\nClave privada: \n", encrypted_key)
15
16 #obtener clave pública
17 print("\nClave pública: \n", key.publickey().export_key())
18
19 #obtener la clave pública en base a la lectura del fichero de clave privada
20 encoded_key = open(dir_path + "/rsa_key_privada.bin", "rb").read()
21 key = RSA.import_key(encoded_key, passphrase=codigoClave)
                                            #Obtiene clave publica
22
23 print("\nClave pública:\n", key.publickey().export_key())
24 print(key)
```

Listado B.8: ej08-ClavePublicaPrivadaPalabra.py (Departamento de Informática y Comunicaciones del IES San Juan de la Rambla, 2023a)

```
from Crypto.PublicKey import RSA
import os

dir_path = os.path.dirname(os.path.realpath(__file__))

key = RSA.generate(2043)
private_key = key.export_key()
file_out = open(dir_path + "/privada_usuario_A.pem", "wb")
file_out.write(private_key)
file_out.close()

public_key = key.publickey().export_key()
file_out = open(dir_path + "/publica_usuario_A.pem", "wb")
file_out.write (public_key)
file_out.close()
```

Listado B.9: ej09-GeneracionParClavesRSA.py (Departamento de Informática y Comunicaciones del IES San Juan de la Rambla, 2023a)

```
from Crypto.PublicKey import RSA
from Crypto.Random import get_random_bytes
from Crypto.Cipher import AES, PKCS1_OAEP                    #Based on RSA and the
     OAEP padding
import os

dir_path = os.path.dirname(os.path.realpath(__file__))

data = "La criptografia a través en Python a través de <<pycryptodome>> es
    consistente".encode("utf-8")
file_out = open(dir_path + "/Datos_Cifrados.bin", "wb")

recipient_key = RSA.import_key(open(dir_path + "/publica_usuario_A.pem").read())
               #Clave del usuario receptor
session_key = get_random_bytes(16)

# Cifrar la sesión con la clave pública del usuario_A
cipher_rsa = PKCS1_OAEP.new(recipient_key)
enc_session_key = cipher_rsa.encrypt(session_key)                    #Ciframos la
    clave de sesion con la clave publica del usuario

# Cifrar Los datos con la sesión de AES
cipher_aes = AES.new(session_key, AES. MODE_EAX)
ciphertext, tag = cipher_aes.encrypt_and_digest(data)
[ file_out.write (x) for x in (enc_session_key, cipher_aes.nonce, tag, ciphertext
    ) ]
file_out.close()
```

Listado B.10: ej10-CifrarFicheroRSA.py (Departamento de Informática y Comunicaciones del IES San Juan de la Rambla, 2023a)

```python
from Crypto.PublicKey import RSA
from Crypto.Cipher import AES, PKCS1_OAEP
import os

dir_path = os.path.dirname(os.path.realpath(__file__))

file_in = open(dir_path + "/Datos_Cifrados.bin", "rb")

private_key = RSA.import_key(open(dir_path + "/privada_usuario_A.pem").read())
enc_session_key, nonce, tag, ciphertext = \
    [ file_in.read(x) for x in (private_key.size_in_bytes(), 16, 16, -1) ]

# Descifrar la sesión RSA con la clave privada del usuario_A
cipher_rsa = PKCS1_OAEP.new(private_key)
session_key = cipher_rsa.decrypt(enc_session_key)

# Descifrar los datos de la sesión con AES
cipher_aes = AES.new(session_key, AES.MODE_EAX, nonce)
data = cipher_aes.decrypt_and_verify(ciphertext, tag)
print(data.decode("utf-8"))
```

Listado B.11: ej11-DescifrarFicheroRSA.py (Departamento de Informática y Comunicaciones del IES San Juan de la Rambla, 2023a)

Actividad 7. Firma RSA

Los ficheros de código de la Actividad 7 se corresponden con los mostrados en los Listados B.12, B.13 y B.14.

```python
from Crypto.PublicKey import DSA
import os

dir_path = os.path.dirname(os.path.realpath(__file__))

# Creación de clave DSA
key = DSA.generate(2048)
f = open(dir_path + "/private_key.pem", "wb")

#grabación clave privada
f.write(key.export_key())
f.close()
print("Clave privada: \n", key.export_key())

f = open(dir_path + "/public_key.pem", "wb")

#grabación clave pública
f.write(key.public_key().export_key())
f.close()
print("Clave pública:\n", key.public_key().export_key())
```

Listado B.12: ej12-GeneracionClavePublicaPrivadaDSA.py (Departamento de Informática y Comunicaciones del IES San Juan de la Rambla, 2023a)

```
from Crypto.PublicKey import DSA
from Crypto.Signature import DSS
from Crypto.Hash import SHA256
import json
import os

dir_path = os.path.dirname(os.path.realpath(__file__))

f = open(dir_path + "/private_key.pem", "r")                    #Clave privada
    del emisor
key = DSA.import_key(f.read())

# Firmar un mensaje con la clave privada
mensaje = b"Comprobamos quien firma este mensaje"
hash_obj = SHA256.new(mensaje)
firmador = DSS.new(key, 'fips-186-3')                   #fips-186-3 es un modo de
     operacion
firma = firmador.sign(hash_obj)

# creamos un fichero JSON con el texto y la firma
# lo codificamos en dos caracteres hexadecimales cada byte
mensajeFirmado = json.dumps({'mensaje':mensaje.hex() , 'firma': firma.hex()})
# print (mensajeFirmado)
f = open(dir_path + "/mensajefirmado.txt", "w")
f.write(mensajeFirmado)
f.close()
```

Listado B.13: ej13-FirmaMensajeClaveDSAPrivada.py (Departamento de Informática y Comunicaciones del IES San Juan de la Rambla, 2023a)

```
from Crypto.PublicKey import DSA
from Crypto.Signature import DSS
from Crypto.Hash import SHA256
import json
import os

dir_path = os.path.dirname(os.path.realpath(__file__))

# abrimos un fichero JSON con el texto y La firma
# viene codificado en dos caracteres hexadecimales cada byte

f = open(dir_path + "/mensajefirmado.txt", "r")
mensajeFirmado = f.read()
print(mensajeFirmado)
mensajeRecibido = json.loads(mensajeFirmado)

# creamos un verificador con la firma leida en el ficharo JSON
# usamos la clave pública del remitenta
f = open(dir_path + "/public_key.pem", "r")
                                            #Clave publica del emisor
hash_obj = SHA256.new(bytes.fromhex(mensajeRecibido["mensaje"]))
pub_key = DSA.import_key(f.read())
verificador = DSS.new(pub_key, 'fips-186-3')

```

```
# Verificar la autenticidad del mensaje recibido
# en realidad lo hacemos del hash de todo el mensaje

try:
    verificador.verify(hash_obj, bytes.fromhex(mensajeRecibido["firma"]))
                #Verifica que la firma que se ha adjuntado es la misma que la
        calculada internamente por el metodo sobre el mensaje
    print ("El mensaje es AUTÉNTICO")
except ValueError:
    print ("Este mensaje no na sido firmado de forma válida")
```

Listado B.14: ej14-VerificacionIdentidadDSAPublica.py (Departamento de Informática y Comunicaciones del IES San Juan de la Rambla, 2023a)

Actividad 8. Sockets SSL

El fichero de comandos de OpenSSL y los de código de la Actividad 8 se corresponden con los mostrados en los Listados B.15, B.16 y B.17.

```
openssl genrsa -aes256 -out private_key.key 2048

openssssl rsa -in private_key.key -out private_key.key

openssl req -new -x509 nodes -sha1 -key private_key.key -out certificate.pem -
    days 36500
```

Listado B.15: openssl-commands.sh (Departamento de Informática y Comunicaciones del IES San Juan de la Rambla, 2023a)

```
import socket
import ssl
import os

dir_path = os.path.dirname(os.path.realpath(__file__))

HOST = "localhost"
PORT = 4444

context = ssl.SSLContext(ssl.PROTOCOL_TLS_SERVER)
                                        #Protocolo SSL mas alto
context.load_cert_chain(dir_path + "/certificado.pem", dir_path + "/clave-privada
    .key")          #Carga un certificado (formato X.509) con su clave privada

with socket.socket(socket.AF_INET, socket.SOCK_STREAM, 0) as sock:
    sock.bind((HOST, PORT))
    sock.listen(5)
    print(f"Servidor corriendo en: {HOST}, {PORT}")

    #solapar el socket sobre SSL

    with context.wrap_socket(sock, server_side = True) as ssock:
                                #Envuelve el socket con SSL (Secure Socket), es
```

```
        decir con el contexto anteriormente especificado
21      while True:
22          conn, addr = ssock.accept()
23          print (f"Conexion desde: {addr}")
24
25          #enviar datos
26          data = "Bienvenido al servidor SSL ("+HOST+":"+str(PORT)+")"
27          conn.sendall(data.encode("utF-8"))
28
29          #recibir datos
30
31          data = conn.recv()
32          print(data)
```

Listado B.16: ej15-SocketservidorSSL.py (Departamento de Informática y Comunicaciones del IES San Juan de la Rambla, 2023a)

```
1   import socket
2   import ssl
3   import os
4
5   dir_path = os.path.dirname(os.path.realpath(__file__))
6
7   HOST = "localhost"
8   PORT = 4444
9
10  context = ssl.SSLContext(ssl.PROTOCOL_TLS_CLIENT)
                                            #Protocolo SSL mas alto
11  context.load_verify_locations(dir_path + "/certificado.pem")
                                #Verifica el certificado del servidor (certificado
    .pem es del servidor)
12
13  with socket.socket (socket.AF_INET, socket.SOCK_STREAM, 0) as sock:
14      #solapar el socket sobre SSL
15      with context.wrap_socket(sock, server_hostname=HOST) as ssock:
                              #Envuelve el socket con SSL (Secure Socket), es
          decir con el contexto anteriormente especificado
16          print(ssock.version())
                                                                    #Version de
              SSL utilizada
17          ssock.connect((HOST, PORT))
18          print("Conexión con éxito")
19
20          #recibir datos
21          data = ssock.recv(1024)
22          print(f" Recipido: {data!r}")
23
24          #enviar datos
25          ssock.sendall("Hola, soy un cliente SSL".encode("utf-8"))
```

Listado B.17: ej16-SocketClienteSSL.py (Departamento de Informática y Comunicaciones del IES San Juan de la Rambla, 2023a)

Actividad 9. Generación de certificados PKI

Se te ha asignado la tarea de generación de certificados, en una infraestructura PKI teniendo en cuenta su jerarquía, de una aplicación de escritorio remoto que se está diseñando. Dicha aplicación consta de una parte servidora y otra cliente. Al servidor se pueden conectar múltiples clientes. Por el momento todo funciona correctamente, ahora el siguiente objetivo es añadirle seguridad a la aplicación, es decir, que las conexiones estén cifradas y autenticadas. Para ello se usarán certificados SSL tanto para el servidor como para los clientes. Se pide:

- Diseñar la PKI con los certificados necesarios justificando la solución. Usar la herramienta draw.io.
- Detallar la secuencia de comandos necesaria con la herramienta Open SSL para generar dichos certificados.

Actividad 10. Cifrado cuántico

Redacta un informe explicando en que consiste el cifrado clásico y el cifrado cuántico así como sus diferencias.

- Para el cifrado clásico mencionar el sistema OTP (One Time Pad) y sus características.
- Para el cifrado cuántico exponer del protocolo BB84 lo siguiente:
  - Su funcionamiento.
  - Las debilidades a primera vista del protocolo y sus soluciones gracias a la estadística y a la naturaleza del mundo cuántico.
  - Si es incondicionalmente seguro teóricamente o prácticamente también.
  - Los ataques que se pueden realizar sobre el protocolo.
  - Mencionar otros protocolos de cifrado cuántico más modernos.

Actividad 11. Uso de MD5 para verificación de ficheros

Esta actividad consistirá en crear una firma md5 para un fichero comprimido que van a generar y luego verificar que si se modifica alguno de los ficheros que lo componen no tendrá validez la firma realizada anteriormente.

El programa debe mostrar el siguiente menú:

1. Creación de un fichero comprimido.
2. Generación del hash md5 de un fichero.
3. Verificación del hash de un fichero.

4. Salir del sistema.

Detallamos lo que se quiere hacer en cada opción:

- **Opción del menú 01**. Se va a generar un fichero comprimido de nombre "prueba_act01". Para la generación de este fichero seguir el procedimiento indicado en la siguiente página web https://somebooks.es/comprimir-archivos-desde-la-lineacomandos-windows-10/.

  El comando "compact" deberá ser invocado desde su función en Python (revisar tema01 como hacerlo). Los nombres de los ficheros que se van a comprimir serán solicitados al usuario. Se deberán comprimir al menos dos ficheros y uno de ellos debe ser de texto (para modificarlo más adelante). El usuario decide cuántos ficheros se van a comprimir.

  Una vez que se ha verificado que todos los ficheros existen se procede a crear de forma comprimida el fichero "prueba_act01".

- **Opción del menú 02**. Una vez creado el fichero en la opción 01, ahora se genera el código hash-md5 del fichero mediante la librería correspondiente. Este número debe ser mostrado por pantalla.

- **Opción del menú 03**. Consistirá en pedirle al usuario el nombre del fichero y la clave md5 correspondiente para verificar que el fichero está correcto. Para verificar el buen funcionamiento del método md5 deberá realizar lo siguiente.

  - Modificar el fichero de texto que incluye en la compresión.
  - Volver a comprimir.
  - Verificar el nuevo fichero comprimido con el código md5 generado la primera vez (se supone que ahora debe fallar).

  Se pide entregar aparte del código del programa, un fichero .pdf con las capturas de pantalla de toda las ejecuciones y salidas que realice con su programa, detallando en cada momento qué es lo que está ejecutando.

Actividad 12. Transmisión segura de información con DES

Se quiere probar el algoritmo DES para cifrar los datos de un fichero que un equipo cliente le va a pasar al servidor. Para ello se les deja el código de las implementaciones del cliente y servidor usando ambos protocolos TCP y UDP. El cliente debe pedir al usuario el nombre del fichero a transmitir, luego el cliente debe realizar lo siguiente:

- Enviar el nombre del fichero al servidor.
- Transmitir de forma cifrada el contenido del fichero.

El servidor debe crear dos ficheros: El primero tendrá el mismo nombre que el recibido añadiéndole una terminación -enc. nombre-recibido-enc que contendrá los datos del fichero

cifrado y el segundo nombre-recibido-desenc que tendrá el texto en claro después de haberlo descifrado.

El procedimento anterior se debe realizar para cada protocolo TCP y UDP.

Se pide entregar los 4 ficheros mostrados en los Listados B.18, B.19, B.20 y B.21 con la funcionalidad explicada.

```
import socket

HOST = ''  # todas las interfaces locales a la escucha
PORT = 2000        # Puerto de escucha

with socket.socket(socket.AF_INET, socket.SOCK_STREAM) as s:
  s.bind((HOST, PORT))
  s.listen()

  conn, addr = s.accept() #línea bloqueante
  with conn:
    print(f"Conexión exitosa con el cliente. IP ({addr[0]}) Puerto ({addr[1]})")
    while True:
      data = conn.recv(1024) #línea bloqueante
      print(data.decode())
      if data == b"0":
        break
      conn.sendall(b"mensaje recibido esta vez")

# modificacion 01

# modificar el servidor para que reciba, cuente e imprima los mensajes
# indicando el numero de mensaje que ha llegado.


# modificacion 02

# hay que hacer que el servidor pueda atender a varios clientes
# para ellos a cada conexion se le crea un hilo para atender los
# mensajes del cliente que se ha conectado

# en este caso se debe ampliar la cabecera que se muestra por pantalla
# indicando ademas del numero de mensaje, los datos del cliente y su
# conexion
```

Listado B.18: act02-tcpServer.py (Departamento de Informática y Comunicaciones del IES San Juan de la Rambla, 2023a)

```
import socket

HOST = '127.0.0.1'
PORT = 2000

with socket.socket(socket.AF_INET, socket.SOCK_STREAM) as s:
  s.connect((HOST, PORT))
  print('Conectado con éxito')
  s.send('Yo, tu cliente, te saludo.'.encode())
```

```
10    # numBytes = s.send(b'Yo, tu cliente, te saludo.')
11    # print (numBytes)
12    data = s.recv(1024) #linea bloqueante
13    s.send(b"0")     # terminar el envío
14
15  print('Recibido:', repr(data.decode()))
16
17
18  # modificacion 01
19
20  # preparar el mensaje para que envie un numero aleatorio de mensajes
21  # al servidor.
22  # los mensajes son pedidos al usuario.
23  # para terminar enviar 0
24
25  # modificacion 02
26  # ver si hay que realizar algun cambio en el cliente para que la
27  # modificacion sugerida funcione correctamente.
```

Listado B.19: act02-tcpClient.py (Departamento de Informática y Comunicaciones del IES San Juan de la Rambla, 2023a)

```
1   import socket
2
3   HOST = ''
4   PORT = 2000
5
6   with socket.socket(socket.AF_INET, socket.SOCK_DGRAM) as s:
7
8     s_addr = (HOST, PORT)
9     s.bind(s_addr)
10    #recibir
11    datos, addr = s.recvfrom(1024) #linea bloqueante
12    print('recibidos  {} bytes de {}'.format(len(datos), addr))
13    print(datos)
14
15    #enviar
16    if datos:
17      sent = s.sendto(b"Saludos desde el Servidor UDP", addr)
18      print('enviados {} bytes de vuelta a {}'.format(sent, addr))
```

Listado B.20: act02-udpServer.py (Departamento de Informática y Comunicaciones del IES San Juan de la Rambla, 2023a)

```
1   import socket
2
3   HOST = '127.0.0.1'
4   PORT = 2000
5
6   with socket.socket(socket.AF_INET, socket.SOCK_DGRAM) as s:
7     s_addr = (HOST, PORT)
8     server_address = (s_addr)
9     message = b'Saludos desde el cliente'
10
```

```
11      # enviar
12      print('Enviando {!r}'.format(message))
13      sent = s.sendto(message, server_address)
14
15      # recibir
16      print('Esperando por la respuesta')
17      data, server = s.recvfrom(1024) #linea bloqueante
18      print('recibidos {!r}'.format(data))
```

Listado B.21: act02-udpClient.py (Departamento de Informática y Comunicaciones del IES San Juan de la Rambla, 2023a)