

ASIGNATURA DE DESARROLLO PARA DISPOSITIVOS INTELIGENTES

1. Competencias	Construir soluciones de software y sistemas inteligentes mediante la gestión de proyectos, integración de metodologías, modelos y herramientas de desarrollo bajo la normatividad aplicable para la optimización de proyectos de investigación, innovación, desarrollo tecnológico y de emprendimiento.
2. Cuatrimestre	Noveno
3. Horas Teóricas	20
4. Horas Prácticas	55
5. Horas Totales	75
6. Horas Totales por Semana Cuatrimestre	5
7. Objetivo de aprendizaje	El alumno implementará aplicaciones de wearables y pantallas mediante el uso de herramientas de desarrollo para la gestión y el intercambio de información entre dispositivos inteligentes

Unidades de Aprendizaje	Horas		
	Teóricas	Prácticas	Totales
1. Diseño de interfaces responsivas	5	10	15
2. Aplicaciones para Wearables	10	25	35
3. Aplicaciones para pantallas inteligentes	5	20	25
Totales	20	55	75

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Desarrollo y Gestión de Software	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

DESARROLLO PARA DISPOSITIVOS INTELIGENTES

UNIDADES DE APRENDIZAJE

1. Unidad de aprendizaje	I. Diseño de interfaces responsivas
2. Horas Teóricas	5
3. Horas Prácticas	10
4. Horas Totales	15
5. Objetivo de la Unidad de Aprendizaje	El alumno diseñará interfaces de usuario responsivas para su óptima visualización en dispositivos móviles.

Temas	Saber	Saber hacer	Ser
Consideraciones y diferencias entre dispositivos	<p>Definir los tipos de dispositivos móviles</p> <p>Identificar las características y capacidades de los dispositivos wearables y pantallas inteligentes</p> <p>Identificar las características de los sistemas operativos de dispositivos wearables y pantallas inteligente</p>	<p>Determinar la funcionalidad de la aplicación móvil a desarrollar</p>	<p>Comunicación eficiente</p> <p>Análítico</p> <p>Sentido de la planificación</p> <p>Capacidad de autoaprendizaje</p>
Interfaces de usuario	<p>Identificar las densidades y resoluciones aplicables a elementos gráficos y texto en pantallas móviles</p> <p>Reconocer las buenas prácticas de desarrollo de interfaces de usuario</p>	<p>Determinar los esquemas de usabilidad y accesibilidad</p> <p>Diseñar interfaces de usuario responsivas:</p> <ul style="list-style-type: none"> - Orientación - Fuentes - Imágenes 	<p>Comunicación eficiente</p> <p>Análítico</p> <p>Sentido de la planificación</p> <p>Capacidad de autoaprendizaje</p>

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Desarrollo y Gestión de Software	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

	responsivas	- Distribución de objetos o elementos	
--	-------------	---------------------------------------	--

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Desarrollo y Gestión de Software	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

DESARROLLO PARA DISPOSITIVOS INTELIGENTES

PROCESO DE EVALUACIÓN

Resultado de aprendizaje	Secuencia de aprendizaje	Instrumentos y tipos de reactivos
<p>Entrega un documento basado en un caso de estudio que contenga:</p> <ul style="list-style-type: none"> - La funcionalidad de la aplicación - Esquemas de usabilidad y accesibilidad - Diseño de interfaces de usuario 	<ol style="list-style-type: none"> 1. Identificar características de los dispositivos y sistemas operativos de wearables y pantallas inteligentes. 2. Analizar las densidades y resoluciones aplicables 3. Comprender los esquemas de usabilidad y accesibilidad 4. Explicar el diseño de interfaces de usuario responsivas 	<ul style="list-style-type: none"> - Estudio de casos - Listas de cotejo

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Desarrollo y Gestión de Software	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

DESARROLLO PARA DISPOSITIVOS INTELIGENTES

PROCESO ENSEÑANZA APRENDIZAJE

Métodos y técnicas de enseñanza	Medios y materiales didácticos
<ul style="list-style-type: none"> - Análisis de casos - Aprendizaje basado en proyectos - Equipos colaborativos 	<ul style="list-style-type: none"> - Equipos de cómputo - Proyector - Internet - Pizarrón y marcadores - Plataformas virtuales - Ejercicios prácticos - Herramientas de modelado

ESPACIO FORMATIVO

Aula	Laboratorio / Taller	Empresa
X	X	

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Desarrollo y Gestión de Software	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

DESARROLLO PARA DISPOSITIVOS INTELIGENTES

UNIDADES DE APRENDIZAJE

1. Unidad de aprendizaje	II. Aplicaciones para Wearables
2. Horas Teóricas	10
3. Horas Prácticas	25
4. Horas Totales	35
5. Objetivo de la Unidad de Aprendizaje	El alumno desarrollará aplicaciones de wearables para gestionar e intercambiar información con dispositivos móviles

Temas	Saber	Saber hacer	Ser
Implementación de Interfaces de usuario en wearables	<p>Identificar las herramientas de desarrollo y emuladores aplicables</p> <p>Identificar el funcionamiento de las herramientas de desarrollo para dispositivos wearables</p> <p>Identificar los tipos de elementos gráficos en dispositivos wearables</p>	<p>Seleccionar herramientas de desarrollo de aplicaciones en wearables</p> <p>Configurar emuladores de dispositivos wearables</p> <p>Desarrollar interfaces de usuario de dispositivos wearables</p>	<p>Comunicación eficiente</p> <p>Analítico</p> <p>Sentido de la planificación</p> <p>Capacidad de autoaprendizaje</p> <p>Proactividad</p> <p>Trabajo en equipo</p>

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Desarrollo y Gestión de Software	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

<p>Programación de Aplicaciones wearables</p>	<p>Identificar la sintaxis y semántica del lenguaje de programación</p> <p>Identificar las sentencias de control condicionales y cíclicas del lenguaje</p> <p>Identificar la programación de métodos o funciones</p> <p>Identificar el acceso a métodos o funciones</p> <p>Identificar errores en las pruebas de la aplicación</p>	<p>Implementar la funcionalidad de aplicaciones en dispositivos wearables</p> <p>Realizar la generación de archivos ejecutables de las aplicaciones</p>	<p>Comunicación eficiente</p> <p>Analítico</p> <p>Sentido de la planificación</p> <p>Capacidad de autoaprendizaje</p> <p>Proactividad</p> <p>Trabajo en equipo</p>
<p>Vinculación entre dispositivos móviles</p>	<p>Definir los mecanismos de vinculación entre la aplicación wearable desarrollada y dispositivos móviles</p> <p>Identificar los mecanismos de intercambio de datos entre la aplicación wearable desarrollada y dispositivos móviles</p> <ul style="list-style-type: none"> - Notificaciones - Cámara - Sensores 	<p>Implementar la vinculación entre la aplicación wearable desarrollada y dispositivos móviles</p> <p>Implementar el intercambio de datos entre la aplicación wearable desarrollada y dispositivos móviles</p>	<p>Responsabilidad Proactivo</p> <p>Analítico</p> <p>Capacidad de autoaprendizaje</p> <p>Uso de procesos cognitivos</p> <p>Razonamiento lógico</p>

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Desarrollo y Gestión de Software	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

DESARROLLO PARA DISPOSITIVOS INTELIGENTES

PROCESO DE EVALUACIÓN

Resultado de aprendizaje	Secuencia de aprendizaje	Instrumentos y tipos de reactivos
<p>Desarrolla una aplicación para wearables basado en un caso de estudio que incluya:</p> <ul style="list-style-type: none"> - Icono de aplicación - Vinculación con otros dispositivos móviles - Funcionalidad acorde al caso - Código fuente - Archivos ejecutables <p>Entrega un reporte con base en el caso de estudio integrando:</p> <ul style="list-style-type: none"> - Configuración de las herramientas - Configuración de los emuladores 	<ol style="list-style-type: none"> 1. Identificar herramientas y emuladores de desarrollo 2. Comprender la herramienta de desarrollo y la configuración de emuladores 3. Comprender la programación de interfaces de usuario en dispositivos wearables 4. Identificar los mecanismos para el intercambio de datos entre dispositivos móviles 5. Explicar la vinculación entre dispositivos 	<ul style="list-style-type: none"> - Análisis de casos - Aprendizaje basado en proyectos - Equipos colaborativos

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Desarrollo y Gestión de Software	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

DESARROLLO PARA DISPOSITIVOS INTELIGENTES

PROCESO ENSEÑANZA APRENDIZAJE

Métodos y técnicas de enseñanza	Medios y materiales didácticos
<ul style="list-style-type: none"> - Prácticas de laboratorio. - Prácticas demostrativas. - Tareas de investigación. 	<ul style="list-style-type: none"> - Equipos de cómputo. - Proyector. - Internet. - Pizarrón y marcadores. - Plataformas virtuales. - Equipo multimedia. - Ejercicios prácticos. - Entorno de desarrollo integrado. - Frameworks.

ESPACIO FORMATIVO

Aula	Laboratorio / Taller	Empresa
	X	

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Desarrollo y Gestión de Software	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

DESARROLLO PARA DISPOSITIVOS INTELIGENTES

UNIDADES DE APRENDIZAJE

1. Unidad de aprendizaje	III. Aplicaciones para pantallas inteligentes
2. Horas Teóricas	5
3. Horas Prácticas	20
4. Horas Totales	25
5. Objetivo de la Unidad de Aprendizaje	El alumno desarrollará aplicaciones de pantallas inteligentes para gestionar e intercambiar información multimedia con dispositivos móviles

Temas	Saber	Saber hacer	Ser
Implementación de Interfaces de usuario en pantallas inteligentes.	<p>Identificar las herramientas de desarrollo y emuladores aplicables</p> <p>Identificar el funcionamiento de las herramientas de desarrollo en pantallas inteligentes</p> <p>Identificar los tipos de elementos gráficos en pantallas inteligentes</p>	<p>Seleccionar herramientas de desarrollo de aplicaciones en pantallas inteligentes</p> <p>Configurar emuladores</p> <p>Desarrollar interfaces de usuario de pantallas inteligentes</p>	<p>Comunicación eficiente</p> <p>Análítico</p> <p>Sentido de la planificación</p> <p>Capacidad de autoaprendizaje</p> <p>Proactividad</p> <p>Trabajo en equipo</p>
Programación de aplicaciones para pantallas inteligentes	<p>Identificar los formatos de audio y vídeo compatibles</p> <p>Identificar la sintaxis y semántica del lenguaje de programación</p> <p>Identificar las sentencias de control condicionales y cíclicas del lenguaje</p> <p>Identificar la programación de métodos o funciones</p> <p>Identificar el acceso a métodos o funciones</p>	<p>Implementar la funcionalidad de aplicaciones en pantallas inteligentes</p> <p>Realizar la generación de archivos ejecutables de las aplicaciones</p>	<p>Comunicación eficiente</p> <p>Análítico</p> <p>Sentido de la planificación</p> <p>Capacidad de autoaprendizaje</p> <p>Proactividad</p> <p>Trabajo en equipo</p>

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Desarrollo y Gestión de Software	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

	Identificar errores en las pruebas de la aplicación		
Vinculación entre teléfono y pantallas inteligentes	<p>Definir los mecanismos de vinculación entre la aplicación de pantalla y teléfonos inteligentes</p> <p>Identificar los mecanismos de intercambio de imágenes y videos entre la aplicación de pantalla y teléfonos inteligentes</p>	<p>Implementar la vinculación entre las pantallas y teléfonos inteligentes</p> <p>Implementar el intercambio de imágenes y videos entre la aplicación de pantalla y teléfonos inteligentes.</p>	<p>Comunicación eficiente</p> <p>Análítico</p> <p>Sentido de la planificación</p> <p>Capacidad de autoaprendizaje</p> <p>Proactividad</p> <p>Trabajo en equipo</p>

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Desarrollo y Gestión de Software	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

DESARROLLO PARA DISPOSITIVOS INTELIGENTES

PROCESO DE EVALUACIÓN

Resultado de aprendizaje	Secuencia de aprendizaje	Instrumentos y tipos de reactivos
<p>Desarrolla una aplicación para pantallas inteligentes basado en un caso de estudio que incluya:</p> <ul style="list-style-type: none"> - Icono de aplicación - Funcionalidad acorde al caso - Código fuente - Archivos ejecutables <p>Entrega un reporte con base en el caso de estudio integrando:</p> <ul style="list-style-type: none"> - Configuración de las herramientas - Configuración de los emuladores 	<ol style="list-style-type: none"> 1. Identificar formatos de audio y video, herramientas y emuladores de desarrollo 2. Comprender la herramienta de desarrollo y la configuración de emuladores 3. Comprender la programación de interfaces de usuario en pantallas inteligentes 4. Identificar los mecanismos para el intercambio de imágenes y video entre pantallas y teléfonos inteligentes 	<ul style="list-style-type: none"> - Estudio de casos. - Listas de cotejo

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Desarrollo y Gestión de Software	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

DESARROLLO PARA DISPOSITIVOS INTELIGENTES

PROCESO ENSEÑANZA APRENDIZAJE

Métodos y técnicas de enseñanza	Medios y materiales didácticos
<ul style="list-style-type: none"> - Análisis de casos - Aprendizaje basado en proyectos - Equipos colaborativos 	<ul style="list-style-type: none"> - Equipos de cómputo - Proyector - Internet - Pizarrón y marcadores - Plataformas virtuales - Ejercicios prácticos - Herramientas de modelado

ESPACIO FORMATIVO

Aula	Laboratorio / Taller	Empresa
	X	

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Desarrollo y Gestión de Software	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

DESARROLLO PARA DISPOSITIVOS INTELIGENTES

CAPACIDADES DERIVADAS DE LAS COMPETENCIAS PROFESIONALES A LAS QUE CONTRIBUYE LA ASIGNATURA

Capacidad	Criterios de Desempeño
Determinar el tipo de proyecto de software mediante el análisis de un problema para determinar la forma de desarrollo.	Entrega un documento de definición del proyecto que incluya: <ul style="list-style-type: none"> - Descripción de la problemática. - Objetivos establecidos. - Metas y alcances. - Requerimientos. - Justificación del tipo de proyecto.
Gestionar el desarrollo de software mediante el seguimiento de la codificación y ejecución de pruebas para la integración continua del producto.	<ol style="list-style-type: none"> 1. Entrega los resultados de las pruebas de sistema en entorno de operación. 2. Entrega un documento que incluya: <ul style="list-style-type: none"> - Evidencias del seguimiento de la ejecución del plan de entregas continuas. - Evidencia de acuerdo de aceptación. 3. Entrega archivos que incluya: <ul style="list-style-type: none"> - Unidades estables de software a producción.
Implementar modelos de datos mediante herramientas y sistemas gestores para garantizar la disponibilidad e integridad de la información	<ol style="list-style-type: none"> 1. Entrega los resultados de las pruebas de sistema en entorno de operación. 2. Entrega un documento que incluya: <ul style="list-style-type: none"> - Evidencias del seguimiento de la ejecución del plan de entregas continuas. - Evidencia de acuerdo de aceptación. 3. Entrega archivos que incluya: <ul style="list-style-type: none"> - Unidades estables de software a producción.
Desarrollar Componentes mediante el uso de patrones de diseño, APIs y frameworks de desarrollo conforme a la arquitectura establecida para la integración continua de unidades del software	<ol style="list-style-type: none"> 1. Entrega un documento que incluya: <ul style="list-style-type: none"> - Justificación de las herramientas para el diseño de las interfaces. - Maquetado de interfaces. - Modelos de navegación. - Guía de estilo de interfaces. 2. Entrega archivos de las interfaces."
Implementar modelos de datos mediante herramientas y sistemas gestores para garantizar la disponibilidad e integridad de la información	Entrega un documento que incluya: <ul style="list-style-type: none"> - Justificación de las herramientas a utilizar para los modelos de datos. - Modelos de datos.

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Desarrollo y Gestión de Software	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

	<ul style="list-style-type: none"> - Descripción de metadatos. <p>2. Entrega archivos, credenciales de registro y secuencia de configuración para la creación de los modelos de datos.</p>
Desarrollar Componentes mediante el uso de patrones de diseño, APIs y frameworks de desarrollo conforme a la arquitectura establecida para la integración continua de unidades del software	<p>1. Entrega un documento que incluya:</p> <ul style="list-style-type: none"> - Justificación de las herramientas para el desarrollo de componentes. - Justificación de los patrones de diseño a utilizar. - Justificación de los lenguajes de programación a utilizar. <p>2. Entrega archivos de código fuente y configuración de los componentes desarrollados.</p> <p>3. Entrega evidencia de almacenamiento de archivos de código fuente y actualizaciones en la documentación del proyecto en la plataforma de versionamiento seleccionada.</p>
Implementar esquemas de seguridad mediante codificación, estándares, protocolos, herramientas e infraestructura para garantizar la privacidad y confidencialidad de la información cumpliendo con leyes y regulaciones aplicables	<p>1. Entrega un documento que incluya:</p> <ul style="list-style-type: none"> - Listado de las leyes y regulaciones aplicables al proyecto de desarrollo de software. - Descripción de acciones encaminadas a cumplir las leyes y regulaciones aplicables al proyecto de desarrollo de software. - Descripción de estándares, protocolos, herramientas e infraestructura para garantizar la privacidad y confidencialidad de la información del proyecto de desarrollo de software. - Reporte de pruebas de seguridad. <p>2. Entrega archivos de código fuente y configuración de los esquemas de seguridad.</p>
Ejecutar pruebas a componentes de software a través del uso de técnicas y herramientas de testing para asegurar su correcto funcionamiento	<p>1. Entrega un documento que incluya:</p> <ul style="list-style-type: none"> - Justificación de las técnicas y herramientas utilizadas. - Descripción de las pruebas. - Informe de resultados del plan de pruebas. <p>2. Entrega archivos de código fuente y configuración del componente.</p>
Implementar soluciones de software mediante la integración de componentes de acuerdo a la arquitectura definida para su liberación en un ambiente de producción	<p>1. Entrega manuales del software establecidos en el plan de desarrollo.</p> <p>2. Entrega la versión estable del software integrando</p>

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Desarrollo y Gestión de Software	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

	<p>los componentes de acuerdo al plan de entregas continuas.</p> <p>3. Entrega evidencia de almacenamiento de archivos de código fuente y actualizaciones de la documentación del proyecto en la plataforma de versionamiento establecida.</p>
--	--

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Desarrollo y Gestión de Software	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

DESARROLLO PARA DISPOSITIVOS INTELIGENTES
FUENTES BIBLIOGRÁFICAS

Autor	Año	Título del Documento	Ciudad	País	Editorial
Neil Smyth	2020 ISBN 978-1-951442-24-8	<i>Android Studio 4.1 Development Essentials - Kotlin Edition (Edición Kindle)</i>		Estados Unidos	Payload Media
Neil Smyth	2020 ISBN 978-1-951442-26-2	<i>Android Studio 4.1 Development Essentials - Java Edition (Edición Kindle)</i>		Estados Unidos	Payload Media
Ashok Kumar S.	2017 ISBN 978-1-78712-322-9	<i>Android Wear Projects: Create smart Android Apps for Wearables (Edición Kindle)</i>	Birmingham	UK	Packt Publishing
Paul Trebilcox-Ruiz	2016 ISBN: 978-1-4842-1783-2 eBook ISBN: 978-1-4842-1784-9	<i>Android TV Apps Development: Building for Media and Games</i>		Estados Unidos	Apress
Varios	2020	<i>Cómo crear apps para TV</i> https://developer.android.com/training/tv/start		Estados Unidos	Google Developers

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Desarrollo y Gestión de Software	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

Jesús Tomás Gironés	2020 ASIN : B08KTJM9P2	<i>El gran libro de Android 8a Edición</i>		España	Marcombo SA
Ribas Lequerica, Joan	2018 ISBN: 978-84-415-3892-4	<i>Desarrollo de aplicaciones para Android</i>	Salamanca	España	Anaya
Tomás Gironés, Jesús; Carbonell, Vicente Albiol, Antonio; Fahli, Mohamed	2016 ISBN: 978-607-622-690-2	<i>Dispositivos Wearable, Visión Artificial, Google Glass y Android TV</i>		México	Alfaomega, Marcombo
Olivia Ramos, Rubén; Olivia Ramos, Juan Tomás	2019 ISBN: 9786075384030	<i>ARDUINO Y ANDROID - Proyectos wearable para smart watches, smart tv y dispositivos móviles</i>		México	Alfaomega

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Desarrollo y Gestión de Software	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	