



## UNIVERSIDAD DE GUADALAJARA

### Centro Universitario de Arte, Arquitectura y Diseño

#### PROGRAMA DE UNIDAD DE APRENDIZAJE 2019 A

##### 1.-DATOS DE IDENTIFICACIÓN.

1.1.Nombre de la unidad de aprendizaje:	<b>Programación Avanzada</b>		1.2. Código de la unidad de aprendizaje:	<b>IH365</b>
1.3. Departamento:	<b>Producción y Desarrollo</b>		1.4. Código de Departamento:	
1.5. Carga horaria:	Teoría:	Práctica:	Total:	
06 Hrs. Semana	48 HORAS	48 HORAS	96 HORAS	
1.6. Créditos:	1.7. Nivel de formación Profesional:		1.8. Tipo de curso ( modalidad ):	
6	<b>LICENCIATURA</b>		Presencial	
1.9. Prerrequisitos:	Unidades de aprendizaje Capacidades y habilidades previas		Programación Ninguna	

##### 2.- ÁREA DE FORMACIÓN EN QUE SE UBICA Y CARRERAS EN LAS QUE SE IMPARTE:

ÁREA DE FORMACIÓN:	ÁREA DE FORMACIÓN BÁSICA PARTICULAR OBLIGATORIA
CARRERA:	Licenciatura en Diseño, Arte y Tecnologías Interactivas

MISIÓN:	VISION:
<p>El Centro Universitario de Arte, Arquitectura y Diseño es una dependencia de la Universidad de Guadalajara dedicada a formar profesionistas de calidad, innovadores y comprometidos en las disciplinas de las artes, la arquitectura y el diseño.</p> <p>En el ámbito de la cultura y la extensión, enfrenta retos de generación y aplicación del conocimiento, educativos y de investigación científica y tecnológica, en un marco de respeto y sustentabilidad para mejorar el entorno social.</p>	<p>El Centro Universitario de Arte, Arquitectura y Diseño es una de las mejores opciones educativas en las artes, la arquitectura y el diseño, con fundamento en los procesos creativos y la investigación científica y tecnológica.</p> <p>Cuenta con liderazgo académico internacional, consolidado en la calidad de sus programas educativos. Sus egresados satisfacen con relevante capacidad las demandas sociales, ambientales, productivas y culturales de México y su Región.</p>

##### PERFIL DEL EGRESADO:

El egresado de la **Licenciatura en Diseño, Arte y Tecnologías Interactivas** debe ser un profesional que genere experiencias de interacción mediante la innovación y el uso de las tecnologías digitales a través de productos tangibles y virtuales creando oportunidades de negocio. Aplique métodos propios del diseño e ingeniería mediante conocimientos profundos y especializados de las artes, la tecnología y del mercado frente a un marco de oportunidades globales. Analice la realidad y viabilidad de proyectos de diseño, animación y tecnología digital consciente del medio ambiente, para resolver problemas, de acuerdo a los requerimientos exigidos por las distintas industrias y sociedad en general.

## VÍNCULOS DE LA UNIDAD DE APRENDIZAJE CON LA CARRERA:

Se vincula y complementa los conocimientos teóricos y técnico de desarrollo de aplicaciones que ayuden en la automatización de rutinas en desarrollo de prototipos.

## UNIDADES DE APRENDIZAJE CON QUE SE RELACIONA:

- Programación I
- Materiales y Tecnologías
- Liderazgo empresarial
- Finanzas y Proyectos
- Factores Ergonómicos para el Diseño Interactivo.
- User Experience Design
- Experiencia Interactiva I
- Experiencia Interactiva II
- Diseño de Videojuegos
- Reto de Innovación I
- Reto de Innovación II
- Diseño de Apps
- Proyectos de Diseño Interactivo
- Marketing Digital

## OBJETIVO GENERAL:

Interpretar los conceptos básicos de un lenguaje de programación baso en objetos, desarrollara la habilidad de uso de una herramienta de programación para la implementación de aplicativos en dispositivos móviles y su uso en aplicaciones para apoyo a los proyectos de Tecnologías Interactivas y su aporte en la sociedad.

## 3.- COMPETENCIAS QUE EL ALUMNO DEBERÁ DEMOSTRAR, CON LOS REQUISITOS CORRESPONDIENTES:

COMPETENCIAS:	REQUISITOS COGNITIVOS:	REQUISITOS PROCEDIMENTALES:	REQUISITOS ACTITUDINALES:
COMPETENCIA 1.-  Identifica e interpreta modelos de programación orientado a objetos y el lenguaje de programación Swift, utilizando metodología de aprendizaje basado en problemas	<ul style="list-style-type: none"><li>• Instala y personaliza su entorno de trabajo, para este curso XCODE utilizando el lenguaje SWIFT</li><li>• Explorar los pasos fundamentales para resolver problemas, conoce identificadores, tipos de variables y constantes.</li><li>• Identificar el uso de una biblioteca de Swift.</li><li>• Reconocer los tipos de datos privados definidos en la</li></ul>	<ul style="list-style-type: none"><li>• Realiza una investigación de los modelos de programación.</li><li>• Identifica los algoritmos para resolver un problema.7 Identifica las partes del entorno de trabajo</li><li>• Identifica las String</li><li>• Realiza la demostración lógica de una app y su estructura.</li><li>• Realiza los ejercicios procedimentales para el uso de las variables, y funciones.</li></ul>	<ul style="list-style-type: none"><li>• Conoce y sigue las normas para la elaboración de reportes de trabajo.</li><li>• Registra, con orden y limpieza la información recabada.</li><li>• Entrega sus trabajos puntualmente.</li></ul>

	<p>biblioteca estándar de Swift</p> <ul style="list-style-type: none"> <li>• Reconocer los conceptos básicos que estructuran el lenguaje.</li> <li>• Conoce el manejo de funciones, así como definir funciones básicas y su funcionalidad.</li> <li>• Identificar los conceptos de parámetros, y su uso en las funciones.</li> </ul>		
COMPETENCIA 2.-  Resuelve problemas a partir de la lógica función mediante el uso de condicionales, además identifica y conoce las estructuras de datos con el uso de arreglos para el manejo de datos, además de estructuras de control y de repetición, utilizando metodología de aprendizaje basado en problemas.	<ul style="list-style-type: none"> <li>• Utilizar el visualizador de documentación de Xcode para buscar información acerca de tipos, métodos y propiedades.</li> <li>• Conocer y manejar las sentencias condicionales if / else, además de identificar los valores boléanos, y operadores lógicos de comparación AND, OR, EQUAL.</li> <li>• Conocer el uso de una instancia explorando los métodos y propiedades que componen esta última,</li> <li>• Comprender el uso de listas, arreglos.</li> <li>• implementar estructuras de repetición para trabajar con los objetos de un arreglo.</li> </ul>	<ul style="list-style-type: none"> <li>• Investiga el concepto de estructuras de datos, listas. Y su funcionamiento lógico.</li> <li>• Investiga algún caso de uso de estructura de control.</li> <li>• Analiza con cuidado cada una problemática para utilizar las estructuras condicionales.</li> <li>• Aplica el procedimiento a una situación dada para resolverla analíticamente.</li> <li>• Elabora un problema y sus soluciones para realizar un programa que utilice estructuras de control, condicionales, estructura de datos.</li> </ul>	<ul style="list-style-type: none"> <li>• Mantiene orden en el desarrollo de los procedimientos.</li> <li>• Elige los casos que tienen una aplicación práctica.</li> <li>• Registra, con orden y limpieza la información recabada.</li> </ul>

	<ul style="list-style-type: none"> <li>• Trazar un tipo personalizado de datos, identificando e implementando sus propiedades y métodos(funciones) mediante el uso de una estructura.</li> </ul>		
COMPETENCIA 3.-  Estudia y/o diseña modelos que simulan un fenómeno natural o social mediante una interfaz de usuario, y su usabilidad, con botones, controles, que está relacionado con la implementación de un proyecto de app con la metodología AOP.	<ul style="list-style-type: none"> <li>• Identificar el uso de interfaces de usuario con la herramienta de creación de interfaces de Xcode.</li> <li>• Conocer las propiedades de una vista de interfaz de usuario y como acceder a ellas.</li> <li>• Identificar y realizar la interacción de los usuarios con botones y otros controles mediante el uso de acciones.</li> <li>• Identificar y establecer restricciones en la ubicación y el tamaño de los elementos de la interfaz de usuario, mediante el uso de las funciones de diseño automático con el uso de vistas de pila.</li> <li>• Identificar y hacer uso de enumeraciones.</li> <li>• Identificar la funcionalidad de la palabra reservada switch como uso condicional de bloques de código.</li> </ul>	<ul style="list-style-type: none"> <li>• Prueba el modelo de simulación aplicado a un caso de programación para verificar su buen funcionamiento.</li> <li>• Construye un modelo que simula el comportamiento de un fenómeno natural o la solución de un problema social mediante el uso de una app.</li> </ul>	<ul style="list-style-type: none"> <li>• Comparte sus conocimientos con sus compañeros de grupo.</li> <li>• Participa activamente en técnicas de estudio grupal.</li> <li>• Colabora de manera comprometida en la elaboración del modelo de simulación.</li> <li>• Junto con sus compañeros se esmera porque el modelo de simulación funcione adecuadamente.</li> </ul>

**4.- METODOLOGÍA DE TRABAJO Y/O ACTIVIDADES PARA EL ALUMNO:** Especificar solo los aspectos generales de cómo se desarrollará el curso, para los aspectos particulares y específicos tomar en consideración el formato de LA DOSIFICACIÓN DE LA COMPETENCIA, anexo.

**PARA LA COMPETENCIA 1.**- Se elaborarán trabajos de investigación y ejercicios prácticos que permitan construir modelos de programación.

**PARA LA COMPETENCIA 2.**- Se conjuntarán casos donde se apliquen la programación, los cuales serán conjuntados a través de equipos que identificarán estos casos.

**PARA LA COMPETENCIA 3.**- Se trabajará en equipos y/o individual en proyecto, los cuales elaborarán un modelo de simulación que muestre la resolución de un caso de programación mediante el uso de una app de un fenómeno natural, social, matemático, físico, arquitectónico etc.

**5.-SISTEMA DE EVALUACIÓN DEL CURSO:**

**5.A. ACREDITACIÓN Y EVALUACIÓN. Criterios y mecanismos (asistencia, requisitos, exámenes, participación, trabajos, etc.).**

El curso se evalúa de manera continua. Para acreditar es necesario contar con el 80% de asistencias. (Art. 20 Reglamento general de evaluación y promoción de alumnos).

**Evaluación Continua:** Participación en clase y cumplimiento en la entrega de trabajos y tareas. Cumplimiento en la entrega de avances conforme al programa y calendario establecido.

**Evaluación Parcial:** Se realizarán dos evaluaciones parciales.

**Evaluación de prácticas realizadas en clase:** Cumplimiento y funcionamiento de la práctica.

**Evaluación Final:**

Demostración del cumplimiento establecido en el programa por medio de la realización del(os) proyecto(s).

Para la evaluación en periodo extraordinario se aplicará atendiendo a lo establecido en los artículos 25, 26 y 27 del Reglamento General de Evaluación y Promoción de Alumnos, atendiendo a la nota ponderada obtenida en el cumplimiento de los trabajos y/o actividades descritas en este programa. (considero no debe existir extraordinario por ser un taller)

Examen (2 parciales)	20%
Proyecto (1 aplicación)	30%
Prácticas en Aula	10%
Tareas y trabajos	30%
Asistencia	<u>10 %</u>
	<b>100 %</b>

**5.B.- CALIFICACIÓN:**

COMPETENCIA:	ASPECTOS A TOMAR EN CUENTA	Parcial	Final
1. Identifica e interpreta modelos de programación orientado a objetos y el lenguaje de programación Swift, utilizando metodología de aprendizaje basado en problemas	1. Participación en la clase y ejercicios 2. Tareas en casa 3. Práctica en laboratorio 3. Examen parcial	40% 25% 10% 25%	40%
2. Resuelve problemas a partir de la lógica función mediante el uso de condicionales, además identifica y conoce las estructuras de datos	1. Investigación individual 2. Investigación grupal 3. Elaboración del compendio de ejercicios aplicados a la programación Swift.	10% 60%	30%

con el uso de arreglos para el manejo de datos, además de estructuras de control y de repetición, utilizando metodología de aprendizaje basado en problemas.	4. Examen parcial	30%	
3. Estudia y/o diseña modelos que simulan un fenómeno natural o social mediante una interfaz de usuario, y su usabilidad, con botones, controles, que está relacionado con la implementación de un proyecto de app con la metodología AOP.	1. Diseño y construcción del proyecto modelo de simulación 2. Examen parcial	60% 40%	30%
	TOTAL	100 %	100 %

#### 6.- BIBLIOGRAFÍA BÁSICA. Mínimo la que debe ser leída:

1. **Apple Education** **Introducción al desarrollo de apps con Swift:** Apple Inc. - Education 2017  
<https://itunes.apple.com/mx/book/introducci%C3%B3n-al-desarrollo-de-apps-con-swift/id1216831475?l=en&mt=11>
2. **Apple Education** **Guía para profesores de Introducción al desarrollo de apps con Swift:** Apple Inc. - Education 2017 <https://itunes.apple.com/mx/book/introducci%C3%B3n-al-desarrollo-de-apps-con-swift/id1216836045?l=en&mt=11>

#### 7.- BIBLIOGRAFÍA COMPLEMENTARIA. Mínimo la que debe ser conocida:

1. **Apple Education** **App development with Swift:** Apple Inc. - Education 2017  
<https://itunes.apple.com/mx/book/app-development-with-swift/id1219117996?l=en&mt=11> Prentice Hall.
2. Becerril González, Sergio Iván **Swift: aprende a crear Apps para iPhone y iPad** Alfa omega Grupo 2016 B.C. CUSUR B.C. CU Tonalá
3. Bucanek, James **Learn iOS app development** Apress 2013 B.C. CUCEA
4. ENRIQUE BLASCO BLANQUER, **DESARROLLO DE APLICACIONES IOS CON SWIFT** ISBN 978-84-9964-621-3
5. MATTHEW MATHIAS, **PROGRAMACION CON SWIFT**, Editorial: ANAYA MULTIMEDIA. ISBN: 9788441538146

FECHA ELABORACIÓN Y APROBACIÓN DE UNIDAD DE APRENDIZAJE:	19 de julio del 2019
PROFESORES QUE PARTICIPARON:	Mtro. Miguel Angel Casillas Lopez
FECHA DE REVISIÓN:	19 de julio del 2019
PROFESORES QUE PARTICIPARON:	Mtro. Miguel Angel Casillas Lopez

## Planeación Didáctica

<b>PORADA DE LA COMPETENCIA 1.</b>	
1.- Identifica e interpreta modelos de programación orientado a objetos y el lenguaje de programación Swift, utilizando metodología de aprendizaje basado en problemas.	
NÚMERO DE SESIONES DE CLASE: 18	
Situación didáctica:	
Se podrán conformar equipos de trabajo según criterio del profesor. Desarrollaran los concepto de programación y trabajaran en el entorno de desarrollo.	
PRODUCTOS Y DESEMPEÑOS	CRITERIOS DE CALIDAD
1. Investigación de los modelos de programación. 2. Exposición de los conceptos de programación manejados. 3. Tareas de ejercicios prácticos.	<ul style="list-style-type: none"><li>• Se entregará completo y bien realizado.</li><li>• Puntualidad en la fecha y hora previamente acordadas.</li><li>• Redacción y ortografía impecables.</li><li>• Deberá ser creativo e innovador.</li><li>• Buena presentación (limpieza y claridad)</li><li>• Ilustrativo, que ayude al entendimiento de los demás.</li><li>• Entregar el documento que explique cómo se forma.</li><li>• Se entregará completo.</li><li>• Fuentes bibliográficas confiables (por lo menos un libro que se consulte de la biblioteca, y páginas Web confiables).</li></ul>
<b>Unidad 1. Introducción al área de trabajo</b>	
<p>Reconocer el espacio de trabajo. Utiliza nombres e identificadores. Maneja conceptos de cadenas e interpolación de cadenas Uso de entorno de trabajo Xcode. Aprende a depura código.</p>	
Contenido programático:	
<ol style="list-style-type: none"><li>1.1. Aspectos básicos de las áreas de juegos</li><li>1.2. Nombres e identificadores</li><li>1.3. Cadenas</li><li>1.4. ¡Hola, mundo!</li></ol>	
<b>Unidad 2. Creación de la primera App</b>	
<p>-Primera app y uso de simulador de iOS. -Funciones conceptos y uso de funciones simples. -Manejo de constantes y variables. -Palabras reservadas y tipos de datos de la biblioteca estándar de Swift -Manejo de datos de parámetros en las funciones.</p>	
Contenido programático:	

- 2.1. Primera app
  - 2.1.1. Proyecto nuevo
  - 2.1.2. Explorar proyecto
  - 2.1.3. Editar el storyboard
- 2.2. Funciones
- 2.3. Práctica; BoogieBot

## **PORTADA DE LA COMPETENCIA 2.**

2.- Resuelve problemas a partir de la lógica función mediante el uso de condicionales, además identifica y conoce las estructuras de datos con el uso de arreglos para el manejo de datos, además de estructuras de control y de repetición, utilizando metodología de aprendizaje basado en problemas.

NÚMERO DE SESIONES DE CLASE: 18

### **Situación didáctica:**

Se podrán conformar equipos de trabajo según criterio del profesor. Se realizará una dinámica de parte del profesor con sus alumnos que consistirá en resolver ejercicios entre todo el grupo, y posteriormente ejercicios por equipo y/o de manera individual. Se compararán resultados y se resolverán dudas si es que hubiera.

<b>PRODUCTOS Y DESEMPEÑOS</b>	<b>CRITERIOS DE CALIDAD</b>
<ol style="list-style-type: none"> <li>1. Investigación sobre los problemas de programación vistos.</li> <li>2. Exposición de casos donde se apliquen los nuevos conceptos.</li> <li>3. Tareas de ejercicios prácticos de manera individual.</li> </ol>	<ul style="list-style-type: none"> <li>• Se entregará completo y bien realizado.</li> <li>• Puntualidad en la fecha y hora previamente acordadas.</li> <li>• Redacción y ortografía impecables.</li> <li>• Deberá ser creativo e innovador.</li> <li>• Buena presentación (limpieza y claridad)</li> <li>• Ilustrativo, que ayude al entendimiento de los demás.</li> <li>• Entregar el documento que explique cómo se forma.</li> <li>• Se entregará completo.</li> <li>• Fuentes bibliográficas confiables (por lo menos un libro que se consulte de la biblioteca, y páginas Web confiables).</li> </ul>

## **Unidad 3. Implementación de sentencias condicionales en la toma de decisiones**

Condicionales if / else, operadores booleanos true o false Bool, y operadores de comparación para la tomar decisiones.

Tipo e instancia sus métodos y propiedades

Documentación de Xcode para buscar información acerca de tipos, métodos y propiedades.

Contenido programático:

- 3.1. Constantes y variables
- 3.2. Tipos
- 3.3. Parámetros y resultados
- 3.4. Toma de decisiones
- 3.5. Instancias, métodos y propiedades
- 3.6. Práctica;

## **Unidad 4. Implementación de bucles para el trabajar con arreglos**

Arreglos  
Bucles en el manejo de arreglos.  
Estructuras de datos y su uso.

Contenido programático:

- 4.1. Arreglos y bucles
- 4.2. Definición de estructuras
- 4.3. Práctica;

### **PORADA DE LA COMPETENCIA 3.**

3.- Estudia y/o diseña modelos que simulan un fenómeno natural o social mediante una interfaz de usuario, y su usabilidad, con botones, controles, que está relacionado con la implementación de un proyecto de app con la metodología AOP.

NÚMERO DE SESIONES DE CLASE: 18

#### **Situación didáctica:**

Se podrán conformar equipos de trabajo según criterio del profesor. Se realizará una dinámica de parte del profesor con sus alumnos que consistirá en exponer los temas, resolver ejercicios entre todo el grupo, y posteriormente ejercicios por equipo y/o de manera individual. Se compararán resultados y se resolverán dudas si es que hubiera.

<b>PRODUCTOS Y DESEMPEÑOS</b>	<b>CRITERIOS DE CALIDAD</b>
<ul style="list-style-type: none"><li>1. Investigación documental sobre los modelos en el campo elegido para desarrollar una app.</li><li>2. Exposición de casos donde se una app.</li><li>3. Tareas de ejercicios prácticos de manera individual.</li></ul>	<ul style="list-style-type: none"><li>• Se entregará completo y bien realizado.</li><li>• Puntualidad en la fecha y hora previamente acordadas.</li><li>• Redacción y ortografía impecables.</li><li>• Deberá ser creativo e innovador.</li><li>• Buena presentación (limpieza y claridad)</li><li>• Ilustrativo, que ayude al entendimiento de los demás.</li><li>• Entregar el documento que explique cómo se forma.</li><li>• Se entregará completo.</li><li>• Fuentes bibliográficas confiables (por lo menos un libro que se consulte de la biblioteca, y páginas Web confiables).</li></ul>

### **Unidad 5. Creación de acciones y conectores**

Interfaces de usuario con el uso de la herramienta de creación de interfaces de Xcode.  
Conectores para acceder a las propiedades de una vista de interfaz de usuario.  
Interacción de los usuarios con botones y otros controles.

Contenido programático:

- 5.1. Acciones y conectores
  - 5.1.1. Creación de conectores

- 5.1.2. Creación de acciones
- 5.1.3. Acciones y conectores múltiples
- 5.1.4. Reguladores
- 5.1.5. Botón
- 5.1.6. Mejora de la Interfaz

## **Unidad 6. Generación de interfaces de usuario adaptables**

Funciones de diseño automático.

Vistas de pila para establecer automáticamente las restricciones de diseño automático.

Uso de enumeraciones.

La instrucción switch para ejecutar bloques de código

Contenido programático:

- 6.1. Interfaces de usuario adaptables
  - 6.1.1. Interfaces de usuario adaptables: SimpleCenter
  - 6.1.2. Interfaces de usuario adaptables: ElementQuiz
  - 6.1.3. Interfaces de usuario adaptables: AnimalSounds
- 6.2. Enumeraciones e instrucción “switch”
- 6.3. Proyecto final

<b>FECHA ELABORACIÓN Y APROBACIÓN DE UNIDAD DE APRENDIZAJE:</b>	<b>19 de julio del 2019</b>
<b>PROFESORES QUE PARTICIPARON:</b>	<b>Mtro. Miguel Angel Casillas Lopez</b>
<b>FECHA DE REVISIÓN:</b>	<b>19 de julio del 2019</b>
<b>PROFESORES QUE PARTICIPARON:</b>	<b>Mtro. Miguel Angel Casillas Lopez</b>