

Básico Ingeniería software, Fundamentos, Algoritmos con C, Bases de datos, Programación estructurada, POO, C++, Arquitectura de software (Pro), Pruebas de software, Terminal, Git y github, Expresiones regulares, Redes de internet, CodeStream, Maestros, padres y niños.

Java Básico/Avanzado Java SE, Pro Java EE, Hibernate y Java Springs, Testing en Java.

Backend con PHP PHP con Laravel, End to end testing con Cypress

Bases de datos SQL y MYSQL, PostgreSQL, Mongo DB.

Firebase Firebase para Android, Firebase para web.

Arquitectura frontend Desarrollo web online (HTML Y CSS), CSS Grid Layout, Responsive design, Animaciones para web, Diseño HTML y CSS para emails, Sass, Stylus, Less, Bootstrap, Foundation, PostCSS, Bots con Facebook Messenger, WPO: Optimización de Carga de Sitios Web, Sistemas de Diseño para desarrolladores

Backend con JavaScript jQuery a JavaScript, Express.js, JavaScript Testing con Jest, Unit Testing con Jasmine.

Angular Angular 4, PWA con Angular, Básico/Avanzado Ionic, Angular 6.

Node.js JavaScript, Node.js, Esenciales de Node.js, Node.js con Hapi, MEAN, JavaScript Full Stack con Sails.js, Avanzado de Node.js

Vue.js vue.js (Pro)

Frontend con React JS React.js, Redux, Webpack, GraphQL, Next.js, React Router, Progressive Web Apps con React.js

Apps con React Native React Native, React navigation.

Aplicaciones con ASP .NET C# con .Net Core (fundamentos), ASP.NET Core.

Apps multiplataforma Electrón: Apps de escritorio en Windows y Mac, Xamarin.Forms.

Aplicaciones Android Bases técnicas de Android, Arquitectura android, Kotlin básico, Android Kotlin, Flutter.

Apple Fullstack Developer Interfases/Aplicaciones con iOS, Swift.

Internet of things Hardware con Arduino, Robótica con Arduino, Diseño de circuitos electrónicos, Intro iot, iot protocolos de comunicación.

Unity Creación de videojuegos, videojuegos 2D/3D con Unity, C# para videojuegos, VR con Unity, Móviles con Unity.

Videojuegos Diseño de videojuegos, Texturizado 3D, Testing de videojuegos, Producción de audio para videojuegos, Modelado 3D, Animación y arte de videojuegos, Marketing y negocios para videojuegos, videojuegos con unreal engine, Creación de personajes, Diseño y prototipado de personajes, Realidad aumentada.

Administración de servidores y DevOps Administración de servidores Linux, Azure laas, Azure Paas, DigitalOcean, Deploy con Now.sh, AWS Cloud, DevOps, Docker, Swarm, Jenkins, IBM cloud.

Amazon web services Networking y content delivery en AWS, Storage en AWS, Bases de datos en AWS, Cloud computing con AWS.

Google Cloud Platform google Cloud platform por Xertica, google App Engine por Xertica, google Kubernetes Engine con Xertica.

IBM Cloud Inteligencia artificial con IBM Watson, IBM Kubernetes Engine.

Matemáticas para programación Pensamiento lógico, Fundamentos de matemáticas, Álgebra, Cálculo para análisis de datos, Excel para análisis de datos, Álgebra lineal, Cálculo multivariable, Matemáticas discretas, Ecuaciones diferenciales, Probabilidad y estadística

Big data y data science Big data y ciencia de datos (Pro), R.

Backend con Python y Django Django, Python (avanzado), Ingeniería de datos con Python.

Inteligencia artificial Introducción al deep learning/machine learning, machine learning aplicado con python,

redes neuronales y backpropagation, machine learning para el análisis predictivo.

Seguridad informática Pentesting, Análisis de vulnerabilidades web con OWASP, Auditorías de seguridad informática a aplicaciones web, autenticación con OAuth.

WordPress PHP, Temas y Plugins en WordPress, Creación y Optimización de sitios web con Wordpress.

Backend con Ruby Ruby: Programación orientada a objetos, Ruby on Rails: POO, Creación de APIs con Ruby on Rails, Ruby.

Backend con GO Programación en Go

FUNDAMENTOS INGENIERIA DE SOFTWARE

Computación, procesadores y memoria

Qué es un system on a chip

Cómo funcionan las computadoras y los teléfonos móviles

Qué son Bits y Bytes

Cómo funcionan los circuitos electrónicos

Procesadores y arquitecturas de CPU

Qué es la memoria RAM y cómo funcionan los discos duros

GPUs, tarjetas de video y sonido

Periféricos y sistemas de entrada de información

Arquitectura de la computación

Cómo funciona Internet

Introducción a las redes, protocolos e Internet

Puertos y protocolos de red

Qué es una dirección IP y el protocolo de Internet

Cables submarinos, antenas y satélites en Internet

Qué es un dominio, DNS o Domain Name System

Cómo los ISP hacen Quality of Service o QoS

Cómo funciona la velocidad en internet

Qué es el Modelo Cliente/Servidor

Cómo funciona realmente un sitio web

Internet es más grande de lo que crees

Sistemas operativos

Diferencias entre Windows, Linux, Mac, iOS y Android

Permisos, niveles de procesos y privilegios de ejecución

Fundamentos de sistemas operativos móviles

Sistemas operativos embebidos e Internet of Things

Archivos y estructuras de datos

Metadatos, cabeceras y extensiones de archivos

Cómo funciona el formato JPG

Videos, contenedores, codecs y protocolos

Cómo funciona .zip: Árboles binarios

Contenido Bonus

Qué es una red neuronal

¿Qué es SQL y NoSQL?

Qué es un algoritmo

El poder de un Raspberry Pi

BASICO DE ALGORITMOS

Bienvenido al Curso

Introducción al curso básico de algoritmos y estructuras de datos

Introducción a los algoritmos

¿Qué entiende una computadora?

Lenguajes de programación

Estructuras de datos

¿Qué es un algoritmo?

Metodología para la construcción de un algoritmo

Variables y tipos de datos

User defined data types

Instalando Ubuntu Bash en Windows

Creando nuestro user defined data type

Abstract Data Types básicos: Lists, Stacks, Queues

Explicación gráfica Data Types básicos

Glosario de funciones para Abstract Data Types

Clases y objetos

Creando tu primera Queue: Arrays

Creando tu primera Queue: implementación.

Creando tu primera Queue: implementar la función enQueue

Creando tu primera Queue: implementar la función deQueue

Creando tu primera Queue: main code

Algoritmos de ordenamiento

Algoritmos de ordenamiento

Bubble sort

Bubble sort: implementación

Bubble sort: main code

Insertion sort

Desafío: implementa un algoritmo de ordenamiento

Recursividad

Recursividad

La función Factorial, calculando el factorial recursivamente

Manejo de cadenas de caracteres

Arte: Generando arte recursivo

Divide and conquer y programación dinámica

Divide and Conquer (divide y vencerás)

Qué es la programación dinámica (divide y vencerás v2.0)

MergeSort

Desafío: Buscar el algoritmo más rápido de sort

Implementando QuickSort con Python

Implementando Quicksort con Python: main code

Algoritmos 'Greedy'

Qué son los Greedy Algorithm

Ejercicio de programación greedy

Ejercicio de programación greedy: main code

Grafos y árboles

Grafos y sus aplicaciones

Árboles

¿Cómo comparar Algoritmos?

Cómo comparar algoritmos y ritmo de crecimiento

¿Qué sigue?

Cierre del curso y siguientes pasos

FUNDAMENTOS DE BASES DE DATOS

Bienvenida conceptos básicos y contexto histórico de las Bases de Datos

Bienvenida conceptos básicos y contexto histórico de las Bases de Datos

Introducción a la bases de datos relacionales

Historia de las RDB

Entidades y atributos

Entidades de Platzi Blog

Relaciones

Múltiples muchos

Diagrama ER

Diagrama Físico: tipos de datos y constraints

Diagrama Físico: normalización

Diagrama Físico: normalizando PlatziBlog

Formas normales en DB relacionales

RDBMS (MySQL) o cómo hacer lo anterior de manera práctica

RDB ¿Qué?

Instalación local de un RDBMS (Windows)

Instalación local de un RDBMS (Mac)

Instalación local de un RDBMS (Ubuntu)

Clientes gráficos

Servicios administrados

SQL hasta en la sopa

Historia de SQL

DDL create

DDL alter

DDL drop

DML

¿Qué tan estandard es SQL?

Creando Platziblog: tablas independientes

Creando Platziblog: tablas dependientes

Creando Platziblog: tablas transitivas

Consultas a una base de datos

¿Por qué las consultas son tan importantes?

Estructura básica de un Query

SELECT

FROM

Utilizando la sentencia FROM

WHERE

Utilizando la sentencia WHERE nulo y no nulo

GROUP BY

ORDER BY y HAVING

El interminable agujero de conejo (Nested queries)

¿Cómo convertir una pregunta en un query SQL?

Preguntándole a la base de datos

Consultando PlatziBlog

Introducción a la bases de datos NO relacionales

¿Qué son y cuáles son los tipos de bases de datos no relacionales?

Servicios administrados y jerarquía de datos

Manejo de modelos de datos en bases de datos no relacionales

Top level collection con firebase

Creando y borrando documentos en firestore

Colecciones vs subcolecciones

Recreando Platziblog

Construyendo Platziblog en firestore

Proyecto final transformando tu proyecto en una db no relacional

Bases de datos en la vida real

Bases de datos en la vida real

Big Data

Data warehouse

Data mining

ETL

Business intelligence

Machine Learning

Data Science

¿Por qué aprender bases de datos hoy?

Bonus

Bases de datos relacionales vs no relacionales

PROGRAMACION ORIENTADA A OBJETOS: POO

Bienvenida e Introducción

Instalando Visual Studio Code

¿Por qué aprender Programación Orientada a Objetos?

¿Qué resuelve la Programación Orientación a Objetos?

Paradigma Orientado a Objetos

Lenguajes Orientados a Objetos

Diagramas de Modelado

UML

Orientación a Objetos

Objetos

Abstracción y Clases

Modularidad

Analizando Uber en Objetos

Reto 1: identificando objetos

Programación Orientada a Objetos. Análisis

Modelando nuestros objetos Uber

Clases en UML y su sintaxis en código

¿Qué es la herencia?

Aplicando Herencia a nuestro proyecto Uber

Reto 2: analicemos un problema

Clases, Objetos y Método Constructor

Definiendo clases en Java y Python

Definiendo Clases en JavaScript

Objetos, método constructor y su sintaxis en código

Objetos. Dando vida a nuestras clases en Java y Python

Declarando un Método Constructor en Java y JavaScript

JavaScript orientado a objetos, lo más nuevo

Declarando un Método Constructor en Python

Herencia

Aplicando herencia en lenguaje Java y PHP

Solución del reto de herencia en PHP

Aplicando herencia en lenguaje Python y JavaScript

Otros tipos de Herencia

Reto 4

Encapsulamiento

Encapsulamiento

Encapsulando atributos en Java

Polimorfismo

Generando polimorfismo en Java

Cierre del curso

El Diagrama UML de Uber

Conclusiones

Bonus: Qué es la Programación Orientada a Objetos

FUNDAMENTOS DE ARQUITECTURA DE SOFTWARE

El proceso de desarrollo de software

Introducción al curso de Fundamentos de Arquitectura de Software

Etapas del proceso de desarrollo de software

Dificultades en el desarrollo de software

Roles

Introducción a la arquitectura de software

¿Qué es arquitectura de software?

La importancia de la comunicación - Ley de Conway

Objetivos del arquitecto

Arquitectura y metodologías

Análisis de requerimientos

Entender el problema

Requerimientos

Riesgos

Restricciones

Reto: Clasificación de requerimientos y análisis de riesgos

Estilos de arquitectura

Arquitectura, panorama y definición

Estilos: Llamado y retorno

Estilos: Flujo de datos

Estilos: Centradas en datos
Estilos: Componentes independientes
Comparando estilos: ¿Cómo elijo?
Reto: Un producto, muchos estilos

Desarrollo del proyecto

Desarrollo del proyecto: PlatziServicios Fase Startup
Desarrollo del proyecto: PlatziServicios Fase Producto en crecimiento
Desarrollo del proyecto: PlatziServicios Fase Escala global
Conclusiones del curso

PROFESIONAL ARQUITECTURA DE SOFTWARE

Introducción al curso

Introducción al curso de Profesional de Arquitectura de Software

Atributos de calidad

Definición
Atributos: Idoneidad funcional
Atributos: Eficiencia de ejecución
Atributos: Compatibilidad
Atributos: Usabilidad
Atributos: Confiabilidad
Atributos: Seguridad
Atributos: Mantenibilidad
Atributos: Portabilidad
Tensiones entre atributos
Analizando PlatziServicios

Patrones de arquitectura

Patrones monolíticos vs distribuidos
Patrones: Modelo Vista Controlador
Patrones: Capas
Patrones: Orientado a eventos / Provisión de eventos.
Patrones: Microkernel - Plug-ins
Patrones: Comparte-nada
Patrones: Microservicios
Patrones: CQRS

Patrones: Hexagonal - Puertos y adaptadores

Patrones: Diseño orientado al dominio

Combinando patrones de arquitectura

Analizando nuevamente PlatziServicios

Diseño de una arquitectura

Pararse en hombros de gigantes
Herramientas y partes de un diseño: Tipos de conectores
Conectores: Llamado asíncrono / síncrono. Modelo Cliente servidor.
Conectores: Enrutador, difusión
Conectores: Pizarra, repositorio, colas, modelo PUBSUB
Escenarios y tácticas
Escenarios: Disponibilidad, detección, reparación
Escenarios: Reintroducción y prevención
Escenarios: Mantenibilidad
Escenarios: Prevenir efectos dominó y diferir enlace
Escenarios: Eficiencia de ejecución
Escenarios: Seguridad
Escenarios: Capacidad de prueba
Escenarios: Usabilidad
Validar las decisiones de diseño: Arquitectura en evolución
Último análisis a PlatziServicios

Modelado y documentación de arquitectura

Cómo comunicar la arquitectura: Vistas y Puntos de vista
Documentación vs implementación
Conclusiones del curso

FUNDAMENTOS DE PRUEBAS DE SOFTWARE

Introducción

Introducción

Principios de las pruebas

¿Qué son las pruebas y por qué deberíamos hacerlas?
Proceso de pruebas del software y los estándares internacionales
Proceso de pruebas del software: Calidad y Defectos
Ciclo de vida del software
Principios del testing moderno
Especialidades del testing

Testing

Presupuesto, Recursos, Tiempo y Actividades Clave
Estrategia de pruebas
Testing en desarrollo de software
Testing ágil

Niveles de pruebas

Tipos de pruebas

Pruebas estáticas y dinámicas

Definición y diseño de pruebas

Gestión, monitoreo y control

Caja Blanca, Gris y Negra
Gestión, monitoreo y control: Monitoreo y Seguimiento
Roles y Responsabilidades
Roles y Responsabilidades en acción

Gestión de bugs

Ejercicios
Retrabajo
Sistema de seguimiento de bugs
Defectos y sugerencias

Depuración

¿Qué es la depuración?
Técnicas de depuración
Pruebas de verificación

Bases de la automatización de pruebas

Automatización de pruebas
Gherkin
Cierre del curso

C# PARA VIDEOJUEGOS

El personaje y el controlador

Introducción: De la idea al desarrollo
Estructura de carpetas en Unity
Assets, Game Objects y Sprite Atlas
Animaciones desde un sprite map
Animation Controller y Transiciones
Plataformas y gravedad
Física en Unity
El script de control
Programando funciones en C# y Unity: Jump
Detectar el suelo con Raycast
Herramientas de debugging visual
Cambiar las animaciones según los estados

Reto: Terminando nuestras animaciones

Solución del reto

Hacer que el personaje camine

El manager del videojuego

Cómo funciona el Game Manager

El Singleton

El modo de juego

Input manager y juegos multiplataforma

Corrección del Bug del salto

La zona de muerte

Reiniciar la partida

Corrigiendo los bugs al reinicar nuestro juego

Diseño procedural de niveles 2D

El diseño de niveles a mano

Configurando nuestros assets para el diseño procedural

Generación procedural de niveles

Creando la zona que eliminará bloques del nivel excedentes

Añadir un nuevo bloque de forma aleatoria

La cámara que sigue al jugador

Destrucción de bloques antiguos

Terminando de programar la destrucción de bloques antiguos

Solucionando el salto de la cámara al reiniciar el juego

HUD, menús y gameplay

El canvas en Unity

Uso de botones para crear un menú

La lógica de los menús

Ejercicio: Preparando el menú del juego

Programando el menú del juego

Los coleccionables del juego

Actualizar UI de coleccionables

Iniciando con pociones y maná

Pociones de vida y maná

Programando las barras de vida y maná

Calculando los puntajes con la distancia y el tiempo de juego

La lógica del maná

Enemigos y dificultad

Plataformas móviles

Iniciar movimiento de la plataforma con trigger

Enemigos móviles

Enemigos móviles: preparando nuestro enemigo para girar

Enemigos móviles: programando que fire al chocar

Arreglando el collider de nuestra roca

Programando la condición de muerte del personaje

Añadiendo música de fondo a nuestro videojuego

Añadiendo efectos de sonido y cierre del curso

VIDEOJUEGOS PARA MOVILES

La estructura de un juego para móvil

Introducción

Presentación y Game Design de nuestro proyecto

IMPORTANTE: Archivos del curso

Estructura de carpetas y assets en Unity

Sprite Atlas y Generación de Caramelos

Sprite Atlas: El fondo del escenario

Sprite Atlas: Los dulces

El script de los dulces

El Manager del tablero o Grid Manager

Inicialización del tablero

Ejercicio: cortar caramelos

Configurar el manager de dulces

Añadir caramelos de forma aleatoria

Diseño de Game Play

Verificar y evitar caramelos repetidos

Selección de los caramelos

Swapping de Caramelos

Caramelos vecinos

Comprobar 3 en raya

Detectar coincidencias y destruir caramelos

Bajar caramelos en cascada cuando un grupo es eliminado

Generar caramelos aleatorios para completar los espacios vacíos

Eliminar los grupos de caramelos similares que se generan al

completar los espacios vacíos (combos)

Retos finales

Trabajando la interfaz de nuestro videojuego

Contar los movimientos restantes y la puntuación

Implementando puntuación en nuestro juego

Reto: Pantalla de Game Over

Reto: Nuevos niveles de dificultad

Reto: Audio de los caramelos

Reto: Movimiento de los Caramelos con Animaciones

Reto: Menú Principal

Cierre del curso

Preparando nuestro juego para exportarlo

Probando nuestro juego en dispositivos móviles y Conclusiones

VR CON UNITY

El controlador del juego

Archivos del curso

¡Bienvenido a los cursos autorizados de Unity!

Bienvenida al curso y requisitos para utilizar el HTC Vive

Inicializando nuestro proyecto y configurando nuestro

PlayerSettings para trabajar con VR

Utilizando nuestros controles de HTC Vive

Creando nuestro mundo 3D en Unity

Comenzando a crear nuestro escenario 3D con los Unity Standard

Assets

Preparando el terreno de nuestro escenario

Texturizando el terreno de juego

Agregando detalles utilizando vegetación

Agregando césped y viento a nuestro nivel

Terminando nuestro entorno 3D, agregando un skybox al proyecto

Utilizando la unity asset store para obtener nuestra arma y escudo

El realismo del VR: Programando a nuestro personaje

Asignando un control de realidad virtual a nuestras armas

Input Manager de nuestros controles de VR

Accediendo a los botones de nuestro control

Preparando nuestro proyectil

Utilizar los sensores de los Vive Controllers para lanzar proyectiles

Corrigiendo los bugs de nuestra arma

Agregando Colliders y Rigid Bodies a nuestro playerController y

armas

Utilizando Getters y Setters para manejar nuestras barras de vida

El realismo del VR: Creación de HUD (Heads Up Display) y

generando enemigos

HUD's en VR
Terminando nuestros HUDs de mana y vida
Creando nuestro script de daño
Asignando nuestros huds a los valores de vida y de mana de nuestro personaje
Agregando enemigos
Añadiendo colliders a nuestros enemigos
Añadiendo animaciones de muerte a nuestros dragones
Spawn de enemigos: Generando enemigos sistemáticamente
El realismo del VR: Programando la IA de nuestro juego
Programando a nuestro enemigo para que pueda seguir
Waypoints
Terminando la lógica de waypoints
Programando una IA básica: Waypoints
Puliendo la lógica de nuestro enemigo
Añadiendo diversos puntos de 'Spawn' para enemigos y cierre del curso

INTRODUCCION A LA REALIDAD AUMENTADA

Introducción a la Realidad Aumentada

Antes de comenzar
¿Quiénes Somos?
¿Qué es la Realidad Aumentada?
Usos de la Realidad Aumentada
Planteamiento Proyecto
Hardware y software
Estructuración del Proyecto
Desarrollo Proyecto
Preparación del entorno de trabajo: Software necesario
Instalando Android Studio
Creación Marcadores o Targets para AR
Preparando los Recursos 2D
Exportando animaciones 2D para Unity
Recursos 3D
Importando recursos en Unity
Vinculando nuestro proyecto en Vuforia con Unity
Importando recursos 3D y audio en Unity
Audio y animaciones
Creación de la UI (interfaz de usuario)
Programando el flujo de la aplicación y asignando información extra
Probando nuestra aplicación
Integrando los scripts a nuestros botones
Terminando el flujo y preparando para exportar la app
Mejora de la app
Conclusiones del curso

DIALOGFLOW PROGRAMACION DE CHATBOTS

Bienvenida e introducción

Bienvenida al curso
Introducción a las interfaces conversacionales
¿Cuándo considerar una experiencia conversacional?
Diseño de experiencias conversacionales
Tu marca
Acciones
User Persona
System Persona

Diálogos de prueba
Poniendo a prueba los diálogos
Diagrama de la conversación
Mejorando nuestra conversación
Introducción a Actions on Google y Dialogflow
Creando el Agente Conversacional
Introducción a la consola de Actions on Google
Creando nuestro primer intento
Habilitando fulfillment
Enlazando fulfillment para intents
Intents y Contextos
Añadiendo elementos visuales
Fulfillment con Firebase Functions
Intent de cierre y resumen de lo que aprendimos
Usando slot filling
Optimizando el agente
Mejorando la pronunciación agregando pausas a las respuestas de tu agente
Mejorando la pronunciación agregando más etiquetas
Preparando el agente para lanzamiento
Llenando Información de descubrimiento y lanzamiento de nuestro Action
Conclusiones del curso

TERMINAL Y LINEA DE COMANDOS

Terminal y Línea de Comandos

Qué es la terminal y para qué sirve
Árbol de directorios y navegación
Manipulación y modificación del árbol de directorios
Herramientas básicas
Variables y entorno
Streams
Procesos desde la terminal
Power Tools: Comandos poderosos de búsqueda
Power Tools: curl, zip y tar
Pipe
Crontab: Una herramienta para automatizar tareas desde la terminal
Links
Usuarios y Permisos
Qué sigue después de aprender a usar la terminal

PROFESIONAL DE GIT Y GITHUB

Introducción a Git

¿Qué es Git?
¿Por qué usar un sistema de control de versiones como Git?
Instalando Git y GitBash en Windows
Instalando Git en OSX
Instalando Git en Linux
Editores de código, archivos binarios y de texto plano
Introducción a la terminal y línea de comandos
Comandos básicos en Git
¿Qué es staging, repositorios y cuál es el ciclo básico de trabajo en GitHub?
¿Qué es un Branch (rama) y cómo funciona un Merge en Git?
Crea un repositorio de Git y haz tu primer commit
Analizar cambios en los archivos de tu proyecto con Git

Volver en el tiempo en nuestro repositorio utilizando branches y checkout

Flujo de trabajo básico en Git

Flujo de trabajo básico con un repositorio remoto

Introducción a las ramas o branches de Git

Fusión de ramas con Git merge

Solución de conflictos al hacer un merge

Trabajando con repositorios remotos en Github

Uso de GitHub

Cómo funcionan las llaves públicas y privadas

Configura tus llaves SSH en local

Conexión a GitHub con SSH

Tags y versiones en Git y GitHub

Manejo de ramas en GitHub

Configurar múltiples colaboradores en un repositorio de GitHub

Flujos de trabajo profesionales

Flujo de trabajo profesional: Haciendo merge de ramas de desarrollo a master

Flujo de trabajo profesional con Pull requests

Utilizando Pull Requests en GitHub

Creando un Fork, contribuyendo a un repositorio

Haciendo deployment a un servidor

Hazme un pull request

Ignorar archivos en el repositorio con .gitignore

Readme.md es una excelente práctica

Tu sitio web público con GitHub Pages

Múltiples entornos de trabajo

Git Rebase: Reorganizando el trabajo realizado

Git Stash: Guardar cambios en memoria y recuperarlos después

Git Clean: Limpiar tu proyecto de archivos no deseados

Git cherry-pick: Traer commits viejos al head de un branch

Comandos en Git para casos de emergencia

Reconstruir commits en Git con amend

Git Reset y Reflog: Úsese en caso de emergencia

Buscar en archivos y commits de Git con Grep y log

Bonus

Comandos y recursos colaborativos en Git y Github

Tu futuro con Git y Github

EXPRESIONES REGULARES

Introducción a las Expresiones Regulares

Bienvenida al curso

¿Qué son las expresiones regulares?

Aplicaciones de las expresiones regulares

Introducción al lenguaje de expresiones regulares

El lenguaje: caracteres, operadores, y construcciones

El carácter (.)

Las clases predefinidas y construídas

Los delimitadores: +, *, ?

Los contadores {1,4}

El caso de (?) como delimitador

Not (^), su uso y sus peligros

Reto: Filtrando letras en números telefónicos utilizando negaciones

Principio (^) y final de línea (\$)

Uso práctico de Expresiones Regulares

Locaciones

Logs

Teléfonos

URLs

Mails

Nombres(?) Reto

Usos avanzados en Expresiones Regulares

Búsqueda y reemplazo

Expresiones Regulares en lenguajes de programación

Uso de REGEX para descomponer queries GET

Explicación del Proyecto

Perl

PHP

Utilizando PHP en la práctica

Python

Java

Java aplicado

JavaScript

`grep` y `find` desde consola

REDES DE INTERNET

Introducción al Curso de Redes de Internet

Bienvenidos al Curso de Redes de Internet

Introducción a Redes de Telecomunicaciones

Qué es la red, Internet, LAN, WAN y Topologías de Red

Intranet y Extranet

Ejemplos de Topologías de Red y sus usos

Tecnologías de acceso a internet: Banda Ancha y velocidad de Internet

La Red Convergente

Configuración de dispositivos (IOS CISCO)

Instalación de Packet Tracer, emulador de redes

Métodos de acceso a los dispositivos

Práctica: Navegación por el sistema operativo, comandos y privilegios

Protocolos y comunicaciones de red

Aspectos básicos de comunicación

Suites de protocolos

Modelos de referencia Modelo OSI, modelo TCP/IP

Segmentación de los mensajes y Unidades de Datos de Protocolo PDU

Practica: Wireshark para ver el tráfico de red

Capa Física y medios de Red

Capa Física: Elementos, funciones, transferencia de datos, ancho de banda, rendimiento

Práctica: Ponchado de cables de red

Capa de enlace de datos: Elementos, funciones

Trama de enlace de datos y direcciones mac

Procesamiento de tramas

Protocolo de resolución de direcciones ARP

El switch y el AP

Capa de Red

Capa de RED

Procesamiento de tramas

Protocolo IP Asignación de direcciones IP, máscara de bits

Práctica conversión de binario a decimal, decimal a binario

IPs Públicas y Privadas

Direccionamiento IP

Capa de Transporte

Funciones, elementos protocolos, tareas

TCP y UDP

Sesiones en vivo y conclusiones del curso

Sesiones en vivo del curso

Practica final con Packet Tracer, asignación de direccionamiento y

configuración de equipos de red

Conclusiones del curso

PROFESIONAL DE DEVOPS

Introducción

Introducción y Filosofía de DevOps

Containers y ambientes de desarrollo

Ambientes Homogéneos para Aplicaciones

Implementación de Dockerfile

Ambientes Homogéneos para Infraestructura

Pruebas

Implementación de Pruebas

Sin pruebas no hay confianza

Integración Continua

Continuous Integration y Artifacts

Continuous integration y Continuous delivery

Implementación de CI con Jenkins

Implementación de Artifacts con Jenkins

Herramientas Externas en proceso de CI

Despliegue Continuo

CD a Ambiente Staging

Continuous Deployments

Implementar acceptance tests en Jenkins

Completar Pipeline de CD

Reliability

Introducción a Incident Response

Introducción a Reliability - SLO/SLI

Implementar Uptime Monitoring

Exception Trackers y Logs

Métricas

Cierre del curso

Conclusiones

PROGRAMACION ESTRUCTURADA

Introducción

Introducción al curso

Conceptos básicos

Set up de codeblocks y compilador

Variables y tipos de datos (Asignación y clasificación)

Entrada y salida de datos

Operadores aritméticos

Operadores de asignación, de incremento y decremento

Resumen: Conceptos Básicos

Control de flujo

Condición if e if - else

Operadores relacionales y lógicos

Switch

¿Qué es un iterador? - Ciclo While

Ciclo For

Ciclo Do-While

Arreglos unidimensionales

Arreglos bidimensionales

Arreglos e iteradores unidimensionales

Arreglos e iteradores bidimensionales

Cadena de caracteres (string)

Resumen: Control de Flujo

Funciones

Funciones: Divide y vencerás

Variables locales y globales

Resumen: Funciones

Conceptos avanzados

Recursividad

Apuntadores

Struct y manejo de archivos

Escritura y lectura de archivos

Manejo de librerías

BASICO JAVA SE

Conocer a Java como lenguaje de programación

¿Qué es Java?

Versiones de Java

Las herramientas más usadas de Java

Creando un entorno de desarrollo en Java en Windows

Creando un entorno de desarrollo en Java en Mac

Creando un entorno de desarrollo en Java en Linux

Escribe tu primer Hola Mundo en Java

Etapas de la programación en Java

Trabajar con Variables en Java

Variables en Java

Actualizando variables

Convención de Nombres en Java

Técnica de Naming: Camel Case

Tipos de datos: int, long, double y float en Java

Tipos de datos: char , boolean en Java

Operadores incremento y decremento

Operaciones matemáticas

Cast en variables: Estimación y Exactitud

Casteo entre tipos de datos

Archivos Jar

Aplicar condicionales en Java

Sentencia if en Java

Alcance de las variables

Operadores Lógicos y Expresiones booleanas

Sentencia Switch

Escribir funciones en Java

¿Para qué sirven las funciones?

Funciones en Java

Java Docs

Javadoc en funciones

Tags Java Docs

Analizar e implementar Ciclos en Java

Bucle do While

Bucle While

Bucle For

Break, Continue y Return

Arrays

Declarando Arreglos

Indices y búsqueda de elementos en Arrays

Fors anidados

Continua con el Curso de Programación Orientada a Objetos en Java

Temario del curso

Introducción al Curso

Antes de comenzar

Bienvenidos al Curso Avanzado de Java SE

Material del Curso

Clases avanzadas

Clases Abstractas

Implementando clases abstractas al proyecto

Ejercicio. Clases Abstractas

Implementando métodos abstractos en Java

JavaDocs

Qué es JavaDocs

Implementando JavaDocs al proyecto

Reto

JavaDocs tags para herencia e interfaces

Generado Java Docs

Clases Anidadas

Clases anidadas y tipos

Ejercicio. Clases Anidadas

Implementando una clase anidada al proyecto

Instanciando clases estáticas anidadas

Enumerations

Interfaces Avanzadas

Métodos con implementación métodos default y private

Creando Interfaz DAO con métodos default y private

Ejercicio. Interfaz DAO

Diferencia Interfaces y Clases Abstractas

Herencia en Interfaces

Colecciones Avanzadas

Map, HashMap, TreeMap y LinkedHashMap

Excepciones

Manejo de errores

Try-catch-finally / Try-with-resources

JDBC

Definición y composición del API

Ejercicio. JDBC API

Creando la base de datos y conectando el proyecto con MySQL

Generando conexión a la base de datos y creando clase de constantes

Sentencia SELECT en Java

Sentencia SELECT con Parámetros

Sentencia INSERT en Java

Reto: Reporte por fecha

Lambdas

Interfaces funcionales

Programación Funcional

Lambdas

Ejercicio. Lambdas

Lambdas como variables y Recursividad

Stream y Filter

Predicate y Consumer

Fin del Curso

Conclusión del curso

AVANZADO DE JAVA SE

Bienvenida e Introducción

Bienvenida

Introducción a tests en software

Tipos y beneficios de los tests

Preparación del IDE, proyecto y librerías

Instalación de IntelliJ IDEA, creación del Proyecto con Maven y Tests Unitarios

Creación de test unitario: lanzar una excepción para alertar sobre un error

Test unitario con JUnit

Organización de tests con JUnit

Testing en Java con JUnit para Verificar Contraseñas

Test con Mockito para simular un dado

Test con Mockito: simular el uso de una pasarela de pago

Análisis de los tests y mejoras

Reto 1: crear la función isEmpty

TDD

TDD: Definición, Beneficios, Ciclos y Reglas

Ejemplos de TDD: calcular el año bisiesto

Ejemplos de TDD: cálculo de descuentos

Reto 2: Práctica de TDD

Tests en una aplicación

Organización de una aplicación

App de Películas: Test de Negocio

App de Películas: test de búsqueda de películas por su duración

Creación de la base de datos y tests de integración con bases de datos

Test de integración con base de datos: guardar películas y búsqueda de películas individuales

Reto 3: Nuevas opciones de búsqueda

Requerimientos y tests

Test a partir de requerimiento

Reto 4: Búsqueda por varios atributos

Conclusiones

Resumen y conclusiones

HTML Y CSS

Internet

¡Bienvenido al Curso!

¿Qué conocimientos me recomiendan para tomar este curso?

¿Cómo se construye la tecnología web?

¿Cómo instalar ATOM?

Comprendamos Internet

Los básicos del web

¿Cómo funciona HTML?

Etiquetas y sus atributos

Reto 1: crea tu fiesta de gifs

¿Cómo funciona CSS?

¿Cómo funciona JavaScript?

Hola mundo y estructura del proyecto

Listado de las etiquetas más usadas

Construcción de un producto digital

Paper wireframe

¿Cómo instalar XD?

Diseño de interfaces

Maquetación de la NavBar

- Maquetación de main_section y agregando estilos CSS
- Maquetación de tarjetas de producto
- Fuentes personalizadas y variables de CSS
- Categorizando etiquetas según nuestra estructura en HTML
- Reto 2: Maquetar el home del producto digital
- Reglas responsive
- Animaciones y transiciones
- Integrando otras herramientas**
- Librería vs Framework
- Sistema de Grillas
- Completando el Sistema de Grillas
- Reto 3: implementación del sistema de grillas
- Maquetación del formulario
- Atributos especiales de las etiquetas para mejorar el funcionamiento de nuestros formularios
- Creación de formulario: inputs
- Creación de formulario: botones
- ¿Cómo crear tu cuenta en Paypal?
- Configuración del botón de pago con Paypal
- Autofill del formulario de PayPal
- Maquetación de pantalla de éxito y error
- Deploy**
- Testing con el Navegador
- Cómo crear tu cuenta en Github
- Subiendo nuestro proyecto a Internet con Github
- Conclusión**
- Conclusiones
- Bonus**
- Librería vs Framework

DESARROLLO WEB ONLINE

- Introducción al curso**
- Bienvenidos al curso de Desarrollo Web Online
- Introducción al desarrollo web
- Plan de estudios para ser un gran desarrollador web
- Software para trabajar en desarrollo web e introducción al proyecto
- HTML**
- DOM
- Etiquetas
- Estructura de nuestro Sitio Web
- Continuando con la estructura de nuestro Sitio Web
- Atributos HTML
- Formularios HTML
- Navegación entre secciones
- Formas de agregar estilos a HTML
- Reglas, selectores, declaraciones, propiedades y valores de CSS
- Los estilos incluidos por el navegador
- Agregando clases a los componentes escritos en HTML
- Estilos y CSS**
- Unidades de medida y colores
- Inspector de elementos
- Tipos de textos personalizados
- Propiedades para los textos
- Dimensiones fijas para elementos
- Backgrounds de color e imagen
- Bordes

- Márgenes
- Rellenos
- Modelo de caja
- Tipos de display
- Propiedades de flexbox
- Alineando elementos de forma horizontal
- Alineando elementos de forma vertical
- Estilizando nuestro website
- Estilos de la sección principal y de footer
- Estilos de proyectos del portafolio
- Estilos sección de eventos
- Estilos en la sección de contacto
- Corrigiendo detalles
- Conclusiones del curso

RESPONSIVE DESIGN

- Introducción a Responsive Design**
- Bienvenidos al curso de Responsive Design
- Conceptos elementales de Responsive Design
- Proyecto del curso: Portafolio Responsive
- Developer tools para Responsive Design
- Meta viewport
- Medidas relativas útiles en Responsive Design
- Media queries
- Formas de incluir media queries
- Repositorio del curso
- Ajustando el proyecto con media queries**
- Diseño elástico con max-width y flex-wrap
- Ajustando el Header
- Ajustando nuestro portafolio
- Ajustando la sección de eventos
- Ajustando la sección de contacto y footer
- CSS positions
- Ajustando nuestra sección principal
- Añadiendo características dedicadas a pantallas pequeñas**
- Videos HTML5
- Video insertado
- Fuentes de iconos
- Añadiendo un menú de hamburguesa
- Posicionando el menú
- Responsive Design y Javascript**
- Añadiendo Javascript para detección de eventos
- Media queries con JavaScript
- Creando un servidor de archivos estáticos con Node
- Remote Debugging en iOS
- Remote Debugging en Android y puliendo últimos detalles
- ¿Qué curso sigue?

CSS GRID LAYOUT

- Introducción**
- Bienvenidos al Curso de CSS Grid Layout
- Propiedades de CSS Grid Layout**
- ¿Qué es CSS grids?
- Conceptos fundamentales sobre CSS Grid Layout
- Definiendo columnas
- Definiendo filas
- Grids identados y tipos de displays

Espaciado entre filas y columnas
Repetidores, unidades de medida y funciones
Definiendo áreas de contenido
Definiendo el tamaño de las columnas dentro de un grid
Definiendo el tamaño de filas dentro de un grid
Definiendo el nombre de líneas
Manejando el grid implícito
Alineación de contenido
Alineación de filas y columnas

Proyectos CSS

Layout de Instagram
Terminando el Layout de Instagram
Layout Masonry (Pinterest)
El landing de los cursos de Platzi
Terminando el landing de los cursos de Platzi
Desafío

Conclusiones

Conclusiones del Curso de CSS Grid Layout

POSTCSS

Introducción

Bienvenida al Curso de PostCSS
Instalación y uso del cliente de PostCSS
Instalando y usando plugins en PostCSS

NextCSS - El futuro de CSS

Instalando CSSNext
Los nuevos módulos de CSS
Variables
Operaciones matemáticas con CSS - CALC
Media Queries
Imágenes retina con Post CSS - Image-set

Colores
Fuentes
Selectores personalizados
Pseudo Clases
Indentado (nesting)

Plugins de PostCSS

Modulariza tu código con postcss-Imports
Auto font-face con FontMagician
Validar CSS con Stylelint
Agrupar Media Queries con mqpacker
Optimiza CSS para producción con CSSNano

Conclusiones

Conclusiones del Curso de PostCSS

ANIMACIONES PARA LA WEB

Introducción

Bienvenido al curso
Introducción a las animaciones y micro interacciones en la web

Transiciones

Sintaxis de transiciones
Transiciones de interacción

Transformaciones CSS

Sintaxis de Transformaciones
Transformaciones de rotación
Transformaciones de traslación y perspectiva
Transformaciones de escala

Transformaciones de sesgados
Punto de transformación
Animaciones CSS
Sintaxis
Aceleración y curva de bezier
Múltiples animaciones
Detectar eventos de animaciones CSS desde JS
Optimizar render con will-change y developer tools
Propiedades animables

Web Animations API (Animaciones en JS)

element.animate
Controlar animaciones
Estas listo para certificarte
Práctica: Tipos de animaciones en la web
Creando logo de platzi
Animando logo de platzi
Animaciones de Entrada y Salida
Creando un Modal
Animando el like de Twitter con Sprites
Creando un sistema de notificación
Animando Notificaciones
Olas Explicación
Olas Animación 2

Contenido bonus: Animaciones en ReactJS

Intro al proyecto final del curso
Configurando una aplicación en React JS
Estructura de proyectos con React
Animaciones con React Transicion Group
Llevando HTML y CSS a React JS
Pasando de una página estática a una dinámica con propiedades
Puliendo el CSS de invie
Añadiendo Cheet JS
Configurando Redux
Animando las guitarras
Transición en los Cards de Guitarras
Animación de entrada y salida en la portada
Animando el background de la portada
Multiplicando y animando los corazones
Cierre del Curso

FUNDAMENTOS DE JAVASCRIPT

Introducción al curso

Bienvenidos al curso
Repositorio del curso

Primeros pasos en JavaScript

Variables
Variables: Strings
Variables: Números
Funciones
El alcance de las funciones
Objetos
Desestructurar objetos
Parámetros como referencia o como valor
Comparaciones en JavaScript
Estructuras de Control y Funciones
Condicionales
Funciones que retornan valores

Arrow functions

Estructuras repetitivas: for

Estructuras repetitivas: while

Estructuras repetitivas: do-while

Condicional múltiple: switch

Arrays

Introducción a arrays

Filtrar un array

Transformar un array

Reducir un array a un valor

Programación Orientada a Objetos en JavaScript

Cómo funcionan las clases en JavaScript

Modificando un prototipo

El contexto de las funciones: quién es this

La verdad oculta sobre las clases en JavaScript

Clases en JavaScript

Asincronismo

Funciones como parámetros

Cómo funciona el asincronismo en JavaScript

Cómo funciona el tiempo en JavaScript

Callbacks

Haciendo múltiples requests

Manejando el Orden y el Asincronismo en JavaScript

Manejo de errores con callbacks

Promesas

Promesas Encadenadas

Múltiples promesas en paralelo

Async-await: lo último en asincronismo

Juego de HTML

Comenzando el juego

Generando una secuencia de números

Iluminando la secuencia de colores

Obteniendo el input del usuario

Agregando la verificación del color elegido

Agregando los estados finales del juego

Conclusiones del curso

Complementos

var, let y const: las diferencias entre ellos

¿Hace cuántos días naciste?

Funciones recursivas

Memoización: ahorrando cómputo

Entiende los closures de JavaScript

Estructuras de datos inmutables

Cambiando de contexto al llamar a una función

¿Cuándo hace falta poner el punto y coma al final de la línea?

ANGULAR 4

Temario del curso

Inicio del curso

¿De qué trata el curso?

Introducción a Angular 4

Qué es Angular, versiones y ventajas

Versionamiento en Angular

Qué es Typescript

Introducción al Proyecto: PlatziSquare!

Setup del Ambiente de Trabajo

Herramientas de trabajo y Angular CLI

Generación y estructura de Angular 4

RETO: Haz un cambio simple en el proyecto.

Conceptos Básicos

Para qué nos sirven los Módulos y Componentes

Tipos de Data Binding y String Interpolation

Property Binding

Event Binding

Two Way Data Binding

Directivas en Angular 4 y ngFor

Directiva ngIf

Instalando librerías con NPM (Google Maps)

Directivas

Directivas ngStyle y ngClass

Directiva ngSwitch

Directiva de atributo

Host Listeners

Host Binders

Angular UI

Angular Material y Bootstrap

Configurando e implementando Bootstrap en nuestro proyecto

Ruteo

Qué hace el router en Angular 4

Implementación de Rutas en el Proyecto

Diferencias entre href y routerLink

Resaltando el link activo con CSS para indicar visualmente en que componente nos encontramos

Parámetros en Rutas

Parámetros tipo Query

Creando una vista de detalle para el proyecto

Creando la página de contacto para PlatziSquare

Servicios

Qué son los servicios en Angular 4

Creando nuestro propio servicio

Configurando Firebase en nuestro proyecto

Guardando Records en Firebase

Obteniendo records desde Firebase

Obteniendo coordenadas usando Geocoding

Reto: Crear una vista para editar records

Mostrando marcadores en el Mapa de Google

Conexión Remota (Http y Sockets)

Funcionamiento de los llamados Http y Sockets

Qué es una arquitectura cliente - servidor

Enviando llamados tipo POST

Enviando llamados tipo GET

Formateando respuestas del servidor con el operador map()

Manejando errores HTTP

Pipes

Utilidad de los Pipes en Angular 4

Usando los pipes por defecto de Angular

Parámetros en pipes

Creando nuestro propio pipe

Animaciones en Angular

Configurando animaciones en nuestro proyecto

Transiciones

Callbacks

Solución al Reto: Añadiendo animaciones a nuestra aplicación

Testing en Angular

Introducción a unit tests

Configuración de testing por default

Corriendo los tests

Creando unit tests para componentes

Integración de Unit Test con Servicios

Autenticación y Protección de Rutas

Cómo funcionan los JSON Web Tokens

Preparación de vistas para login y registro

Registrando usuarios

Loggeando usuarios

Protección de Rutas

Autenticación con redes sociales.

Logout

RxJS

Qué es RxJS

Configurando RxJS en nuestro proyecto

Uso de los Observables

Implementando un TypeAhead

Implementando un TypeAhead 2

Solución al reto autocompletar los campos de dirección usando observables

Publicando nuestro proyecto

Publicando en Firebase Hosting

Fin del curso

Conclusión ¿Qué aprendimos en el curso?

Reto final del curso - realiza un nuevo modulo de PlatziSquare de acuerdo con las historias de usuario

Sesiones en vivo

Creando un traser bullet de PlatziSquare

Sesión de preguntas y respuestas

Release de Angular 5

Sesión de preguntas y respuestas

Angular Universal

Contenido Bonus

Actualización de angular, versión 6.0

ANGULAR 6

Temario del curso

Introducción

Introducción al Curso

Introducción a Angular

Configurando nuestro proyecto en Git y Github

Introducción a Angular CLI y al proyecto de Platzinger

Instalación del entorno de desarrollo en Mac OS

Instalación del entorno de desarrollo en Windows e información acerca del editor que usaré (WebStorm) y alternativas

Corriendo nuestra app en el navegador

Reto: Desarrolla un tutorial escrito comparando Angular con otro Framework (Vue, React)

Definición de la estructura del App

¿Qué hará nuestra app? ¿Cuál es su arquitectura? ¿De qué pantallas se compondrá? ¿Cuál será su look?

Explorando el sistema de archivos

¿Qué son y cómo generar los primeros componentes de nuestra app?

Navegación básica entre componentes

Componentes anidados: Creando un componente con nuestras rutas de navegación que funcione como menú

Reto: Varias conversaciones a la vez

Basics de Angular y TypeScript

¿Cómo usar tipos de datos con TypeScript?

Qué son las interfaces de TypeScript y su implementación

NgFor aplicado en la lista de usuarios

NgIf aplicado en la lista de usuarios

Navegación con parámetros

Accediendo a nuestros usuarios desde Conversation.ts

Creando un servicio de usuarios e Inyectando el servicio en nuestros componentes

Pipes en Angular (date, number, json)

Creando nuestro propio pipe para buscar entre nuestros contactos

Estilos

¿Cómo usar estilos CSS y referenciar recursos?

Instalando librerías usando npm (bootstrap y font-awesome) y Referenciando CSS en el angular.json

Reto: Cambiar ícono según estatus

Implementando estilos en la pantalla de login

Terminando de implementar los estilos de nuestra pantalla de login

Implementando estilos en el navbar

Implementando estilos en la pantalla de home

Implementando estilos en home, lista de contactos

Implementando estilos en profile

Reto: Estilos en la pantalla de conversación

Usando ngClass para añadir estilos dinámicos

Reto: Mejorar estilos en la pantalla de conversación

Acceso a datos remotos y login

¿Qué es Firebase? Creando nuestro proyecto

Diferencia entre conexión por sockets y HTTP

Instalación y setup de la librería AngularFire a través de npm

Login/Registro de usuarios con email y password

Reemplazo del servicio de usuarios por un servicio dinámico con conexión a Firebase

Guards para verificar Firebase auth

Creando un CRUD implementando un API REST HTTP (con diversos backends)

Reto: Implementar API HTTP

Opciones y subida de imágenes

Perfil de usuario

Seleccionar fotos usando una librería para image cropping

Guardando las imágenes de perfil en nuestra base de datos de firebase

Conversación

¿Cómo resolver el problema de comunicación en tiempo real?

Enviando un mensaje (con reproducción de sonido)

Mostrando conversación

Enviando un zumbido (con reproducción de sonido y animación)

Reto: Envía una foto

Amigos

Planeación previa para la característica, ¿cómo generaremos las solicitudes?

Enviando una solicitud de amistad

Generando el modal de solicitud de amistad

Aceptando solicitudes de amistad

Mostrando sólo contactos que son amigos

Componentes anidados (con paso de parámetros)

Reto: Mensaje personal en Solicitud de Amistad

Deploy

Creando un wrapper para desktop, usando Electron

Reto: Dándole Características PWA a nuestra App

Exportando app para web, complementando con Firebase Hosting

Conclusión

Conclusión

Cuéntanos

FUNDAMENTOS DE C# CON NET CORE

Introducción al curso

Bienvenidos al curso

Instalación de C#

Mi primer programa C#

Instalación de Visual Studio Code

Configuración IDE

Breve Historia de C#

Versiones de C# / ¿De dónde viene su nombre?

Clases y Objetos

Clases, objetos, métodos, acciones y atributos

Introducción a la Programación Orientada a Objetos

Aprendamos con un proyecto

Proyecto: CorEscuela

Etapa 1 – Creando la escuela

Creación de la estructura básica de la escuela

Uso de numeraciones y el método ToString

Creación de parámetros

Etapa 2 – Creando Cursos para la escuela

Creando otros objetos para nuestra escuela

¿Qué es un arreglo?

Crea un arreglo de objetos

Formas de recorrer arreglos

Inicializando arreglos y mejorando impresión

Condiciones

Operadores en C#

Etapa 3 – Implementado colecciones para el manejo de cursos

Colecciones vs arreglos

Implementando colecciones

Removiendo y buscando objetos en colecciones

Expresiones lambda y delegados

Etapa 4 – Refactorizando y cargando datos de prueba

Refactorizar a escuela engine

Clases estáticas

Creación de entidades

Generación de datos aleatorios

Datos de prueba

El gran RETO

El gran reto

Resolución del reto

Conclusiones del curso

C# CON NET CORE

Introducción

Bienvenida al curso

Etapa 5 – POO reutilicemos nuestro código

Reutilizando nuestro Código

Herencia

Polimorfismo

Problemas del polimorfismo

Lista de Objetos polimórfica

Utilizando ToString para mejorar la depuración

Etapa 6- Ajustes y funcionalidad

Implementación de interfaces

Uso de interfaces

Creación y uso de regiones

Etapa 7 – Preparando información para nuestros reportes

Parámetros de salida

Probando funcionalidades de parámetros de salida

Sobrecargas de métodos

Listas de solo Lectura

Diccionario Polimórfico

Implementación de un diccionario

Uso de numeraciones en diccionario

Introduciendo datos en un diccionario

Recorrido de datos en diccionario

Impresión de objetos de un diccionario

Uso de la sentencia switch

Eventos

Recapitulemos

Etapa 8 – Consultas

Creación del reporteador

Extrayendo datos de un diccionario de manera segura

Aprendiendo linq

Sentencia where de linq

Divide y triunfarás

Agrupamientos en linq

Tipos anónimos en consultas linq

Segundo Reto

Etapa 9 – Creando una UI de Consola

Excepciones y cascadeo de excepciones

Sentencia finally

Reto final

Cierre del curso

ASP NET CORE

Introducción

Bienvenida al curso

Herramientas y prerrequisitos

La importancia del middleware en el entorno web

Mi primera aplicación ASP.Net Core

Habilitar certificados, planillas preinstaladas

Revisión detalladas de la estructura de directorios

¿Qué es el patrón MVC?

Nuestro Proyecto

Modelo Vista Controlador a nivel básico

Vista y controlador

Modelo a nivel básico

Razor Syntax

Vistas Plantillas y modelo

Vistas, plantillas, Archivos de inicio

Importación de modelos

- Vistas para listar asignaturas
- Vistas parciales
- Mostrando otras listas de objetos
- Actualizando la versión de bootstrap
- Entity Framework Core**
- Introducción a entity framework
- Conectando la solución con Entity Framework Core
- Sembrando datos en la base de datos
- Optimizando siembra de datos
- Refactoring del Modelo para un ORM
- Implementando cambios en el modelo
- Refactoring siembra de datos
- Ajuste de vistas y formularios**
- Ajuste de enrutamiento
- Ajuste de vistas y controladores
- Creando un formulario
- Validaciones de formularios
- Tu primer Reto
- Atributos de validación y bases de datos**
- Data annotations
- Segundo Reto
- Usando una Base de Datos de verdad
- Diseño de vistas y controladores optimizado**
- Scaffolding
- o
- borrón y cuenta nueva
- Tercer Reto
- Deploy del sitio**
- Publicando el website
- Conclusiones

XAMARIN FORMS

- Introducción a Xamarin**
- Bienvenidos al curso
- ¿Qué es Xamarin?
- Instalación y configuración**
- Requerimientos de hardware y software
- Introducción a Instalación de Visual Studio .NET y Herramientas
- Instalación de Visual Studio .NET en Windows
- Instalación de Visual Studio .NET for Mac
- Configuración de Xamarin para iOS
- Fundamentos de Xamarin.Forms**
- Mi primera app con Xamarin.Forms
- Xamarin.Forms vs. Xamarin clásico
- Estrategias para compartir código
- Arquitectura de Xamarin.Forms**
- Anatomía de una aplicación
- Arquitectura de Xamarin.Forms
- Sistema de Propiedades Enlazables**
- Sistema de Propiedades Enlazables
- Usando y creando Propiedades Enlazables
- Propiedades Adjuntas
- Manejo de Eventos**
- Manejo de eventos
- Ciclo de vida
- El Lenguaje XAML**
- ¿Qué es XAML?

- Espacios de Nombres XML
- Sintaxis de subelementos y de contenido
- Extensiones de Marcado
- Recursos
- XAML compilado
- Interfaz de Usuario**
- Contenedores
- Controles comunes
- Creación de controles personalizados
- Estilos
- Triggers en Estilos
- Diccionarios Mezclados
- Transformaciones y Animaciones
- Navegación
- DisplayAlert y DisplayActionSheet
- MessagingCenter
- Enlace de Datos**
- Enlace de Datos
- Interfaces de notificación
- La clase ObservableCollection<T>
- Contexto de Enlace de Datos
- Enlace entre elementos
- Propiedad StringFormat
- Plantillas de Datos
- Convertidores de Valor
- Comandos**
- ¿Qué son los Comandos?
- Implementación básica e implementaciones existentes
- El Patrón de Diseño Model-View-ViewModel (MVVM)**
- Introducción al patrón de diseño MVVM
- ¿Qué es MVVM?
- Componentes del Patrón
- Proyecto: Pomodoro
- App Pomodoro
- Continuando con la app Pomodoro
- Concluyendo nuestra app Pomodoro

INTRODUCCION C++

- Introducción a C++**
- Bienvenida
- ¿Qué es programar y por qué aprender C++?
- Conceptos Generales de C++**
- La Historia de C++ y sus diferencias con C
- Preparando el entorno de trabajo**
- Instalando Codeblocks
- Pasos para instalar en mac
- Ejecutando nuestro primer programa
- ¿Qué son las librerías STD? - Portada del Proyecto
- Manejo de memoria**
- ¿Qué es la memoria y tipos de datos?
- Alojando variables en memoria
- ¿Cómo usar operadores?
- ¿Qué son los apuntadores?
- Reto: Escribir un programa que calcule áreas.
- Entrada de datos y funciones**
- ¿Cómo introducir datos a nuestro programa?
- ¿Cómo usar condicionales?

Usando condicionales

¿Cómo encapsular código en funciones ?

¿Qué son los parámetros y como usarlos?

Reto: Juego narrativo implementando condicionales y entrada de datos.

Loops y arreglos

¿Qué son los arreglos?

¿Qué son los Loops?

Programando Loops

Loops , arreglos y arreglos bidimensionales

Dibujando el mapa de nuestro juego con arreglos

Manipulando mi jugador con inputs en arreglos unidimensionales

Arreglos bidimensionales

Reto: Moviendo a mi personaje en mi arreglo bidimensional

Lectura de Archivos

Creando y leyendo un archivo externo

Leyendo archivos externos

Reto: En el mapa, crear punto de inicio y salida del jugador

Programación Orientada a Objetos

¿Qué es POO?

Definiendo una clase y creando sus instancias

Encapsulación

Abstracción

Herencia

Propiedades de clase en herencia

Polimorfismo

Finalizando nuestro proyecto

Creación de personaje y archivo de encabezado

Moviendo mi personaje con entrada de datos

Mapa

Interacción entre el personaje y el mapa

Paredes

Optimizando trazado de mapa

Colisiones

Creando nuestra portada de juego

Tesoro y victoria

Conclusiones

Proyecto final

INTRODUCCION A PHP

Introducción a PHP

Presentación del curso

¿Qué es PHP?

Setup y Herramientas

Conceptos Básicos de PHP

Revisando el template que usaremos

Sintaxis de PHP

Variables tipos de datos y cadenas

Tipos de Datos en PHP

Arreglos

Ejercicios Arreglos

Condicionales y Ciclos

While vs. Do While

Operadores, Condicionales, Continue y Break

Operadores

Ejercicios Operadores

Funciones

Agregando archivos externos

Programación Orientada a Objetos

Programación Orientada a Objetos

Constructor y Métodos

Herencia

Herencia y polimorfismo

Interfaces

Namespaces

Herramientas para Proyectos PHP

PSR y PHPFIG

Composer

Bases de Datos

Introducción a las Bases de Datos

ORM

Formularios

Eloquent

Listar registros de la base de datos con Eloquent

Insertar datos en MySql con PHP

Estructura/Arquitectura de una aplicacion web

Front Controller

PSR7

Router

MVC, Creando Controllers

MVC Reestructurando Vistas y Controladores

MVC Controller de la entidad Job

Template engines

Template engines

Twig

Templates con Twig

Extendiendo Templates con Twig

Validaciones

Validaciones

Subida de archivos

Ejemplo de subida de archivos

Tips de seguridad para subir archivos en PHP

Autenticación de usuarios y manejo de sesiones

Reto. Creación de usuarios

Autenticando usuarios

Formulario de Login

Manejo de sesiones

Liberación de código/release

Variables de entorno

Deploy en Heroku

Instalación de pgAdmin

Creando la base de datos en Heroku

Conclusiones del curso

AVANZADO DE PHP

Introducción

Introducción

Herramientas

IDEs

Virtualización

Configuración de virtual host

Características avanzadas del lenguaje

Closures

Type Hinting

- Agregar imágenes en Jobs
- Salvar imágenes en Jobs
- Traits
- Endpoint to delete Jobs
- Third party Traits - Soft Delete
- Técnicas de programación**
- SOLID
- SOLID a profundidad
- Inyección de dependencias
- Contenedor de inyección de dependencias
- Middlewares y PSR15
- Implementando el server request handler
- Creando un middleware
- Errores y debug**
- Error Handling
- La biblioteca SPL
- Debug
- Xdebug
- Logs
- Trabajando con Monolog
- Databases**
- Migraciones de bases de datos
- Comandos y Tareas asíncronas**
- Comandos
- Formulario para contacto
- Emails
- Async tasks
- Procesar tareas asíncronas
- Crear un comando para agregar usuarios
- Seguridad**
- Security
- Security Examples
- Conclusiones**
- Crea una API Rest
- Cierre

| PYTHON |
|--|
| Básicos del Lenguaje |
| Guía de instalación y conceptos básicos |
| IMPORTANTE: Instalando Ubuntu Bash en Windows para facilitarte el seguimiento del curso desde Windows. |
| Archivos del curso y Slides |
| ¿Qué es la programación? |
| ¿Por qué programar con Python? |
| Operadores matemáticos |
| Variables y expresiones |
| Presentación del proyecto |
| Funciones |
| Usando funciones en nuestro proyecto |
| Operadores lógicos |
| Estructuras condicionales |
| Uso de strings y ciclos |
| Strings en Python |
| Operaciones con Strings en Python |
| Operaciones con strings y el comando Update |
| Operaciones con strings y el comando Delete |
| Operaciones con strings: Slices en python |

- For loops
- While loops
- Iterators and generators
- Estructuras de Datos**
- Uso de listas
- Operaciones con listas
- Agregando listas a nuestro proyecto
- Diccionarios
- Agregando diccionarios a nuestro proyecto
- Tuplas y conjuntos
- Tuplas y conjuntos en código
- Introducción al módulo collections
- Python comprehensions
- Búsquedas binarias
- Continuando con las Búsquedas Binarias
- Manipulación de archivos en Python 3
- Uso de objetos y módulos**
- Decoradores
- Decoradores en Python
- ¿Qué es la programación orientada a objetos?
- Programación orientada a objetos en Python
- Scopes and namespaces
- Introducción a Click
- Definición a la API pública
- Clients
- Servicios: Lógica de negocio de nuestra aplicación
- Interface de create: Comunicación entre servicios y el cliente
- Actualización de cliente
- Interface de actualización
- Manejo de errores y jerarquía de errores en Python
- Context managers
- Python en el mundo real**
- Aplicaciones de Python en el mundo real
- Conclusiones finales**
- Python 2 vs 3 (Conclusiones)
- Clases bonus**
- Entorno Virtual en Python y su importancia: Python en el mundo real

| BASES TECNICAS DE ANDROID |
|---|
| Bienvenidos al curso |
| Presentación del curso |
| Android |
| Introducción a Android |
| Retos de desarrollar Apps |
| Tipos de dispositivos Android y Soporte |
| Componentes de una aplicación |
| Activity |
| Fragments |
| Intent |
| Services |
| Más sobre Services |
| Broadcast |
| Asynctask |
| Desarrollo |
| IDE Oficial de Android |
| Qué es y Cómo funciona Gradle |

Generación de un apk en Android

Google Play Services

APIS de Google Play Services

Lenguajes para programar en Android

Métricas de Diseño

Qué es Material Design

Widgets de interfaz gráfica

Flutter la nueva forma de hacer interfaces en Android y iOS

Hardware

Consideraciones para trabajar con Hardware en Android

Almacenamiento

Niveles de Almacenamiento y tipos

File y SharedPreferences

Content Providers Bases de Datos y Network

Testing

Tipos de Testing en Android y UI Test

Google Play

Tienda de Aplicaciones Android

Conclusiones del curso

ARQUITECTURA DE ANDROID

Temario del curso

- ¿Qué es la Arquitectura de software?
- ¿Por qué es importante implementar una Arquitectura en el software?
- SOLID
- Evolución de la Arquitectura en Android
- MVC
- MVP
- MVVM

KOTLIN PARA ANDROID

Introducción al Curso de Kotlin

¿Qué vamos a construir con Kotlin?

¿Qué es Kotlin?

Preparando el entorno de desarrollo para trabajar con Kotlin

Instalando y probando Kotlin en Mac

Instalando y probando Kotlin en Windows

Instalando y probando IntelliJ IDEA

Descubriendo Kotlin

Variables y tipos en Kotlin

String templates

Funciones en Kotlin

Clases en Kotlin

Data Class en Kotlin

Default values en Kotlin

Arreglos en Kotlin

Extension Functions en Kotlin

if/if-else en Kotlin

For y While en Kotlin

When en Kotlin

Rangos en Kotlin

Null Safety o !NPE en Kotlin

Destructuring Declarations en Kotlin

Expresion Object y Singletons en Kotlin

Conceptos avanzados de Kotlin

Diferencias entre Kotlin y Java

Simplificando procesos con Kotlin

Qué son las lambdas

Lambdas y cómo se usan

Funciones de orden mayor

Inline functions

Sealed Class

Operadores sobrecargados

Try, catch y manejo de excepciones en Kotlin

Interoperabilidad entre Java y Kotlin

Java vs Kotlin

Cómo configurar Kotlin en Android Studio

Creación del proyecto Interoperabilidad de Java con Kotlin

Kotlin en Java: implementando Funciones

Kotlin en Java: implementando Clases

Java en Kotlin: Implementación de un RecyclerView

Java en Kotlin: implementación de Shared preferences

Finalización del proyecto de interoperabilidad de Java con Kotlin

Android Kotlin

Qué es PlatziStore y qué vamos a necesitar para su construcción - Requerimientos

Qué son y cómo se usan las Android Extensions

Extension Functions en Android

Anko

RecyclerView construcción de la estructura de la vista de principal de PlatziStore

RecyclerView finalización de la construcción de la estructura de la vista de principal de PlatziStore

Anko Layouts

Reto

Material Transitions construcción de la vista

Material Transitions conexión del item de la lista con la vista de detalle

SQLite

Implementando la base de datos SQLite de PlatziStore

RETO : Investiga como es que funcionan las Coroutines en kotlin generando una lista de datos muy grande y asignándola a una lista en Android

Generando un apk válido para la Play Store

Lanzamiento a la play store

Kotlin en el servidor

Kotlin Frameworks

Configuración de un proyecto con Ktor

Hello Ktor

Enviando respuestas desde el servidor - GET y POST

Generando la lista para PlatziStorev

Consumir endpoints en Android con Retrofit

Creando el cliente de retrofit para consumir el servicio

Consumiendo el listado de productos en PlatziStore

Conclusiones del curso de lo que aprendimos en el curso de Kotlin

FUNDAMENTOS DE DESARROLLO DE HARDWARE

Bienvenida al curso

Bienvenidos al curso

Introducción al hardware libre

¿Qué es Arduino y qué placas hay?

¿Cómo se desarrolla el Hardware?

¿Cuál es el software para diseño mecánico?
Elección del software a utilizar
Limitaciones de Arduino y qué es el hardware libre
Reto 1: Para un proyecto de IoT, ¿Cuál tarjeta Arduino utilizarías?
¿Cómo funciona el Curso de Desarrollo de Hardware con Arduino?
Lista de Materiales o Componentes a Utilizar
Sensores, Actuadores y Procesadores
Sensores, actuadores y procesadores
Diferentes tipos de sensores según la salida
Diferentes tipos de sensores según la aplicación
Diferentes tipos de actuadores
Introducción a la electrónica
Diferencias entre electricidad y electrónica
Tipos de componentes electrónicos
Más tipos de componentes electrónicos
Limitando la corriente para evitar quemar un led
Manos a la obra
Puertos de un arduino y sus funciones
Qué es un protoboard
Estructura general de un programa en Arduino y hello world con un LED
Variables y Constantes Globales
Manejo de librerías y funciones
Manejo de valores analógicos: uso del potenciómetro
Uso del monitor serial
PWM y señales analógicas
Un control de color RGB e intensidad lumínica para una lámpara de mesa
Uso de push buttons
Monitor de temperatura con Arduino (uso de LM35 y un lcd 16x2)
Continuando con el monitor de temperatura con Arduino
Finalizando con el monitor de temperatura con Arduino
Reto: Crea tu semáforo con Arduino

DISEÑO DE CIRCUITOS ELECTRONICOS
Introducción a Diseño de PCBs
Bienvenido al Curso de Diseño de Circuitos Electrónicos
¿Qué es una PCB? y Tipos de PCB
Otras partes de la PCB y Tipos de PCB
Software para diseño de PCBs y KiCad
Presentación del contenido del curso
Esquemático o diagrama electrónico y su simbología
Glosario del curso de diseño de circuitos electrónicos
Plataformas de Hardware Libre
Creación de Esquemáticos
Presentación del primer proyecto
Introducción al Ambiente de KiCAD y Creación del Proyecto
Componentes del proyecto
Proveedores de Componentes y cómo buscar Componentes

Creación de Conexiones, Alimentación y Tierra
Bonus: Cómo se crea un componente de Ki-CAD
Bonus: Componentes en serie y paralelo
Conectando los componentes de nuestro esquemático
Reto 1: Crea tu propio esquemático para una fuente de alimentación agregando salidas de voltaje
Librerías de Footprints
Creación de librerías de Footprints
Identificación de la Huella del Componente y Tipos de Huellas
Creación de Footprint o huella
Selección de las Huellas o Footprints y Footprints predeterminadas
Reto: Diseña una Huella en KiCAD
Diseño Básico de Circuitos Impresos
Selección del Fabricante
Reglas de Diseño y Configuración del Ambiente de Trabajo
Importando los Componentes Electrónicos
Ubicación de los Componentes
¿Cómo trazar una pista?
Ruteo Manual Básico a 2 Capas y Creación de Plano de Tierra.
Compilando y Depurando el Diseño
Reto: Rutea tu primera PCB en KiCAD.
Fabricación de Circuitos Impresos
¿Qué son los Gerbers? Importancia de los Fabricantes
Terminados de una PCB
Agregando serigrafía propia a nuestras PCBs
Generación de Gerbers en KiCAD
¿Cómo solicitar la fabricación de un PCB?
Proyecto Final
Presentación del proyecto
Crear esquemático a partir de la lista de componentes y requisitos
Ubicando conectores siguiendo parámetros de diseño mecánico
Organizando nuestros componentes de PlatziArduino:
Comunicaciones y Alimentación de 3.3 V
Organizando nuestros componentes de PlatziArduino:
Microcontrolador
Organizando nuestros componentes de PlatziArduino: Otros componentes
Comenzando a Rutear nuestro platzi Arduino: Comunicaciones y alimentación
Ruteando nuestro Platzi Arduino: Terminando el circuito de alimentación
Ruteando nuestro Platzi Arduino: Terminando el circuito del microcontrolador
Ruteando nuestro Platzi Arduino: Finalizando el microcontrolador y conectores de programación
Reto: Termina de rutear tu Platzi Arduino
Conclusión del curso
Qué sigue después
Aprueba el curso

FUNDAMENTOS DE CIRCUITOS ELECTRONICOS
Temario del curso

- Introducción a los circuitos lógicos
- Circuitos electrónicos básicos
- Capacitores
- Señales eléctricas

- Circuitos reactivos
- Diodos
- Tablas de verdad

INTRODUCCION DE IOT

Introducción al IoT

¿Qué es el IoT? y El IoT de consumo personal (micro)

IoT en industria e infraestructura

Lo necesario para tu proyecto IoT

Materiales y herramientas necesarias

El impacto de mi proyecto IoT

Hardware a utilizar en proyectos de IoT

Tipos de sensores

Conexión física de sensores

Tipos de actuadores

Voltaje y corriente necesarios para el funcionamiento de un actuador

Tipos de controladores y sus características

Los super poderes de la Pi

Instalación del sistema operativo

Configuración inicial de la Pi, comandos útiles de la consola de Linux

Conecta tu Pi a internet

Conexión por SSH

Arduino o Raspberry Pi ¿Cuál escoger?

Tipos de comunicaciones básicas entre dispositivos

Proyecto del Curso

Contexto del problema ¿Cómo lo solucionamos?

Materiales que necesitas

Planeemos el circuito y conectemos sensores

Escribamos el código

Subir código y asegurar comunicaciones

Archivos para preparar Bot.py y Upgrade.sh

Trámite de ingreso a la API de Twitter y configuración de Tweetpy

¡Probemos!

Cierre

Conclusiones

Glosario de términos del curso

CURSO IOT PROTOCOLOS DE COMUNICACION

Conceptos Básicos

Bienvenida al curso

Espectro electromagnético y qué es radiofrecuencia

Clasificación de radiofrecuencia

Regulación de la radiofrecuencia

¿Qué son las Telecomunicaciones?

Halfduplex y Fullduplex

Clasificación de transmisiones: Upstream y Downstream

Link budget

Con ciertas características, ¿Habría conexión en el sistema?

Alto data rate

Wi-fi

Bluetooth

Redes celulares

LPWANs

NB-IoT

Consideraciones para implementación de NB-IoT

LoRaWAN

Reto 2

¿Qué usarías para monitorear la posición GPS de un vehículo? y

¿Qué usarías para monitorear datos en campo?

Diseño

Arquitectura de nodos

Introducción a la práctica

Instalación de framework ESP32 e instalación de framework SAMD21

Qué es una interrupción

Práctica de WiFi

Prueba de señal Wi-fi

Protocolo HTTP desde un microcontrolador

Conectando sensores y actuadores al microcontrolador

Preparación de aplicación para recibir datos

Programación por eventos de un microcontrolador

Configuración de eventos del microcontrolador

Conexión de aplicación

Práctica con LoRa

¿Cuál es el stack de LoRaWAN?

Creando tu propio gateway

Ensamblado de gateway

Pon a andar tu gateway de LoRa

Creando tu Stack de LoRa

Conecta tu nodo a internet

Integrando datos a un dashboard

Terminando de implementar nuestro stack de LoRa

BugFixing nuestro stack de LoRa

Ensamblando el nodo de LoRa

Probando LoRa en nuestro Dashboard

Prácticas con tarjetas MKR

Cierre del curso

Cierre del curso

ALGEBRA LINEAL

Introducción al Curso de Álgebra Lineal

Introducción

Sistemas lineales

¿Qué es un sistema lineal?

Solución de sistemas por método de eliminación

Solución por metodo Gauss

Solución por metodo Gauss Jordan

Vectores

¿Qué es un vector?

Operaciones básicas con vectores

Ejercicio operaciones básicas

Producto punto

Producto cruz

Ecuación de un plano

Ejercicio ecuación de un plano

Guía de instalación de Matlab

Vectores en Matlab

Matrices

Introducción y tipo de matrices

Suma de matrices y multiplicación por escalar

Multiplicación matricial

Matriz inversa

- Matriz inversa metodo alternativo
- Factorización LU
- Matrices en Matlab
- Transformaciones lineales**
- ¿Qué es una transformación lineal
- Reflexiones
- Comprensiones y expansiones
- Cortes o desplazamientos
- Rotaciones
- Determinantes**
- Cálculo de determinantes
- Determinantes por Cofactores
- Geometria y propiedades del determinante
- Regla de Cramer
- Solución de sistemas de 3 o más incógnitas en Matlab

CALCULO MULTIVARIABLE

- Introducción al curso**
- Antes de comenzar
- Bienvenidos al Curso de Cálculo Multivariable
- Ecuaciones paramétricas y coordenadas polares**
- ¿Qué es una ecuación paramétrica?
- Coordenadas polares
- Curvas polares
- Conversión coordenadas cartesianas a polares
- Geometria del espacio**
- Sistemas tridimensionales de coordenadas
- Algunas gráficas en tres dimensiones
- Distancia entre dos puntos en el espacio
- Vectores y planos**
- ¿Qué es un vector?
- Gráficas de funciones de 1 variable en Matlab
- Producto Punto, Producto Cruz
- Graficación de un plano
- Ejercicio / Graficación de un plano
- Graficación en funciones de varias variables en Matlab
- Superficies cuadráticas**
- Elipsoide
- Hiperboloide de una hoja
- Paraboloide elíptico
- Curvas polares en Matlab
- Funciones de varias variables**
- ¿Qué es una función de varias variables?
- Derivadas de funciones vectoriales
- Ejercicio derivadas
- Integrales de funciones vectoriales
- Curvas de nivel
- Curvas de nivel en Matlab
- Ejercicio curvas de nivel
- Derivadas parciales
- Ejercicio derivadas parciales
- Derivada direccional
- Vector gradiente
- Ejercicio derivada direccional con vector gradiente
- Máximos y mínimos
- Ejercicio Máximos y mínimos
- Integrales dobles

- Campos vectoriales
- Ejercicios**
- Ecuaciones paramétricas y coordenadas polares
- Geometría del espacio y vectores
- Superficies cuadráticas
- Funciones de varias variables

ECUACIONES DIFERENCIALES

- Introducción al Curso**
- Introducción y presentación del curso
- Fundamentos de Ecuaciones Diferenciales**
- ¿Para qué nos sirven las ecuaciones diferenciales?
- ¿Que es una ecuación diferencial?
- Tipos de ecuaciones diferenciales
- Conceptos básicos de cálculo
- Ecuaciones Diferenciales de Primer Orden**
- ¿Que es una ecuación separable?
- Ejemplo de ecuación separable
- Procedimiento para saber si una ecuación es separable
- Método de sustitución lineal
- Ejemplo de sustitución lineal
- Ecuaciones diferenciales exactas
- Ejemplo de ecuaciones diferenciales exactas
- Funciones homogéneas, cómo identificarlas
- Ejemplo de funciones homogéneas
- Ecuaciones con coeficientes lineales
- Ejemplo de ecuaciones con coeficiente lineales
- Resolución del desafío
- ¿Que es un factor integrante?
- Factor integrante caso 1
- Factor Integrante caso 2
- Factor integrante caso 3
- Ecuaciones diferenciales lineales
- Ejemplo de ecuaciones diferenciales lineales
- Ejercicios de ecuaciones diferenciales de primer orden
- Ecuaciones Diferenciales de Segundo Orden**
- ¿Qué es una solución linealmente independiente?
- Ecuaciones lineales homogéneas
- Ejemplo de ecuaciones lineales homogéneas
- Ecuaciones lineales homogéneas con raíces complejas
- Ejemplo de ecuaciones lineales homogéneas con raíces complejas
- Ecuación diferencial no homogénea
- Coeficientes indeterminados
- Ejemplo de coeficientes indeterminados
- Variación de parámetros
- Ejemplo de variación de parámetros
- Ejercicios de ecuaciones diferenciales de segundo orden
- Modelos matemáticos**
- Creación de un modelo matemático
- Crecimiento poblacional
- Primer ejemplo de crecimiento poblacional
- Segundo ejemplo de crecimiento poblacional
- Ley de newton de enfriamiento
- Ejemplo de la ley de newton de enfriamiento
- Propagación de un virus y ejemplo
- Ejercicios de modelos matemáticos
- Transformada de laplace**

Conceptos claves para entender la transformada de laplace
Introducción a la transformada de laplace
Introducción y transformada de una exponencial
Propiedades de la transformada de laplace
Transformada inversa
Ejemplo de transformada inversa
Ejercicios de transformada de laplace

PROBABILIDAD Y ESTADISTICA

Introducción
Introducción al curso de Probabilidad y Estadística
Conceptos Básicos
Probabilidad Simple
Cálculo de probabilidades
Probabilidad Compuesta
Probabilidad Compuesta y Diagramas de Árbol
Probabilidad compuesta: Unión
Probabilidad compuesta: Intersección
Variaciones, permutaciones y combinación
Variaciones, permutaciones y combinaciones: Ejercicios
Distribución de Datos
Tabla de Frecuencias
Tablas de Frecuencia en Distribución de Datos: Ejercicios
Tablas de Frecuencia Gráficas
Medidas de Tendencia Central
Gráfica de Dispersión
Parámetros Estadísticos, Centralización
Parámetros Estadísticos, Centralización: Ejercicio
Tipos de Correlación Y Covarianza
Medidas de Dispersión
Rango
Desviación Media
Varianza y Desviación Estándar
Coeficiente de Correlación en un Diagrama de Dispersión
Medidas de Posición
Cuartiles, Deciles, y Percentiles
Cuartiles
Deciles
Percentiles
Regresión Lineal
Qué es y para qué sirve la Regresión Lineal
Ejemplo de Regresión Lineal (Tipo de Correlación)
Ejemplo de Regresión Lineal (Predecir valor futuro)
Importante: antes del examen
Examen de certificación

MATEMATICAS DISCRETAS

Introducción al curso
Bienvenidos al curso
Lógica
Introducción a la lógica
Conectores lógicos
Tablas de verdad
Ejemplo de tabla de verdad
Más ejemplos de tabla de verdad
Circuitos lógicos
Ejemplo de circuitos lógicos

Ejercicios - Lógica
Teoría de conjuntos
Introducción a los conjuntos
Operaciones entre conjuntos
Representación gráfica de conjuntos
Ley de Morgan: Unión de conjuntos
Ley de Morgan: Intersección de conjuntos
Ejercicio de Conjuntos
Ejercicios - Teoría de Conjuntos
Teoría de gráficas
Teoría de gráficas
Grados, caminos, cadenas y ciclos
Caminos y ciclos eulerianos
Caminos y ciclos hamiltonianos
Matriz de adyacencia
Matriz de incidencia
Ejercicio con matrices
Ejercicios - Teoría de gráficas
Árboles
Introducción a los árboles
Árboles
Sub árboles, vértices, y notación
Árbol de expansión mínimo
Árbol binario
Recorrido de árboles
Expresiones aritméticas
Ejercicio: Llevando una expresión aritmética a árbol
Ejercicios - Árboles
Algoritmos
Algoritmo de Prim
Algoritmo de dijkstra
Algoritmo de kruskal
Algoritmo de Fleury
Algoritmo de flujo máximo
Ejercicios - Algoritmos
Conclusiones
Conclusiones del curso

PENSAMIENTO LOGICO

Habilidades básicas del pensamiento
Observación y descripción
Comparación, relación y clasificación
Material: Modulo de Habilidades básicas del pensamiento
Lógica Proposicional
Lógica proposicional e inferencias lógicas
Conectivos lógicos
Material: Modulo Lógica Proposicional
Habilidades Lógico Matemáticas
Puntos, triángulos, sudoku y acertijos
Observación atenta
Respuesta del ejercicio: Observación atenta
Material: Habilidades Lógico Matemáticas
Teoría de conjuntos
Material: Módulo de Teoría de Juegos
Conceptos básicos
Tipos de conjuntos
Operaciones con conjuntos

Aritmetica analógica

Interpretación de los números y signos

Razones

Proporciones

Teorema de Thales

Material: Aritmética en la lógica

Series

Series Numéricas

Series Alfabéticas

Series Alfanuméricas

Series de figuras

Material: Modulo Series

Diagramas de Flujo

Toma de decisiones y simbología

¿Cómo hacer un diagrama de flujo?

Algoritmos en diagramas

Estructuras de control y palabras reservadas

Material: Módulo Diagramas de Flujo

Contenido bonus

3 Juegos de habilidad mental para aprender programación

INGENIERIA DE DATOS CON PYTHON

Introducción

Archivos del curso y Slides

Introducción al curso

¿Qué es la Ciencia e Ingeniería de Datos?

Roles

Configuración del ambiente

Jupyter Notebooks

Tipos de datos

Fuentes de datos

ETL

Web scraping

Introducción a las tecnologías web

Realizar solicitudes HTTP con Python

¿Cómo trabajar con un documento HTML?

Analizando un sitio web para encontrar las directivas a utilizar al hacer un web scrapping

Solicitudes a la web: Requests

Implementando nuestro web scrapper: Configuración

Implementando nuestro web scrapper: Obteniendo enlaces del front page

Implementando nuestro web scrapper: Obteniendo artículos

Obtención de datos del Artículo

Persistiendo la información screpeada

Pandas

Introducción a Pandas

Estructura de datos: Series

Estructura de datos: DataFrames

Índices y selección

Data wrangling con Pandas

Creación de la receta: preparando la automatización

¿Cómo trabajar con datos faltantes?

Operaciones con Series y DataFrames

Limpiando detalles adicionales

Limpiando detalles en nuestro proyecto

Enriquecimiento de los datos

Valores duplicados en Jupyter

Valores duplicados en Python

Vizualización de datos

Intro a Sistemas de Datos

Introducción a los sistemas de datos

Cargando datos a SQLite

Automatización del Pipline

Contenido Bonus

¿Por qué usar la nube?

FUNDAMENTOS DE R

Bienvenida e introducción

Bienvenida al curso

Programación y Data Science

R y proyecto economía naranja.

Instalando nuestras herramientas

Variables, tipos de datos y estructuras

Los primeros cálculos con R y variables

Tipos de datos

Estructura del dataset del proyecto

Vectores

Matrices

Ejercicios con matrices

Operadores para comparar y ubicar datos

Factores, listas y echar un vistazo al dataset

EDA: Exploratory data analysis

Qué es EDA: Exploratory Data Analysis

Gráficas de dispersión e histogramas.

Box Plot y su interpretación

EDA con dataset proyecto - Gráficas de dispersión.

EDA con histogramas.

EDA con dataset proyecto - histogramas - ggplot2

EDA con box plot- ggplot2

EDA con dataset proyecto - box plot- ggplot2 - dplyr

EDA con gráficas de dispersión con más de dos variables - ggplot2

EDA con dataset proyecto usando gráficas de dispersión con más de dos variables - ggplot2 - plotly

La estadística de los datos

Buscando correlaciones con pairs

Confirmando correlaciones con la función cor

Buscando correlaciones con pairs en dataset proyecto

Confirmando correlaciones con la función cor en dataset proyecto.

Protegiéndonos de los peligros del promedio.

Eliminando los NA's para hacer los cálculos.

Estadística y visualización aplicada a análisis de datos de mercadeo.

Ajustando los datos

Generando tablas, filtrando y seleccionando datos - dplyr-Parte 1

Generando tablas, filtrando y seleccionando datos - dplyr-Parte 2

Mejorando la visualización

Viendo más información con facet wrap - Parte 1

Viendo más información con facet wrap - Parte 2

Organizar visualizaciones y código con R Markdown

Conociendo R Markdown y organizando los hallazgos del análisis en un documento PDF.

Conclusiones Finales

Invitación a continuar recorriendo el mundo del data science.

INTRODUCCION A MACHINE LEARNING

Intro

Bienvenidos al Curso de Introducción a Machine Learning

Introducción al Machine Learning

Glosario de palabras técnicas

Conceptos Básicos

Conceptos básicos: Inteligencia artificial, Machine Learning y Deep Learning

Inteligencia artificial, Machine Learning y Deep Learning

Inteligencia Humana y Artificial

Machine Learning, Redes Neuronales Artificiales y Aprendizaje Profundo

Ciencias de Datos

Set de Datos

TUTORIAL PROYECTO: Trabajando con datos

Instalación de Jupyter Notebook

SciKit Learn: tutorial básico

Introducción a Datos y Set de Datos

Proyecto: Trabajando con datos

Modelos de ML

Tipos de Machine Learning y modelos principales

Introducción al aprendizaje supervisado y no supervisado

Machine Learning Supervisado Algoritmos: Regresión Lineal, Logísticay Naive Bayes

Machine Learning Supervisado Algoritmos: K-Nearest Neighbors, Decision Tree, Random Forest, y Redes Neuronales

Proyecto Machine Learning Supervisado: práctica con algoritmo de Decision Tree y Random Forest

Machine Learning No Supervisado: Algoritmos

Proyecto de Machine Learning no Supervisado: aplicando algoritmo k-means

Evaluación

Introducción a la evaluación del modelo entrenado

Evaluando nuestro algoritmo entrenado

Matriz de confusión, precisión y recall

Otras evaluaciones: Error cuadrático medio

Proyecto de Machine Learning Supervisado: aplicando la matriz de confusión a nuestro algoritmo entrenado

Probando nuestro algoritmo entrenado en un set de datos nuevos

ML y Matemáticas

Machine Learning y Cálculo / Reto

Introducción al Machine Learning y Matemáticas

Machine Learning y Álgebra Lineal

Machine Learning y Probabilidad y Estadística

Conceptos Finales

Revisión de los pasos seguidos en el proyecto

Recomendaciones y despedida

INTRODUCCION AL DEEP LEARNING

Introducción a la Inteligencia Artificial

Bienvenidos al curso

¿Qué es la Inteligencia Artificial (IA), Machine Learning y Deep Learning?

Objetivo del curso

Proyecto práctico

Un problema aun no resuelto: La computación afectiva (affective computing)

Sesión interactiva: Configurar tu ambiente de trabajo

Configura tu ambiente de trabajo

Resumen

Conceptos básicos

¿Qué es una neurona?

¿Qué es una red neuronal?

Función softmax(z)

¿Cómo aprende una red neuronal?

Navegando la superficie de error con gradientes

Recapitulación

Proyecto: Diccionario en español

Algoritmos de Deep Learning y Redes Neuronales

Modelo #1: Logistic Classifier

Sesión interactiva: Implementación de Logistic Classifier valores de entrada

Sesión interactiva: Implementación de Logistic Classifier arquitectura

Modelo #2: Multi Layer Perceptron (MLP)

Sesión interactiva: Implementación de MLP ingresando datos

Sesión interactiva: Implementación de MLP a nivel de arquitectura

Modelo #3: Long-Short Term Memory (LSTM)

Sesión interactiva: Implementación de LSTM definiendo entradas

Sesión interactiva: Implementación de LSTM arquitectura y optimización

Corriendo modelos

Recapitulando

Evaluando un modelo

Precisión, Accuracy, Recall, y F1 Confussion Matrix

Conclusiones del curso

A gran poder, gran responsabilidad: Ética e Inteligencia Artificial

Contenido Bonus

El Algoritmo Maestro

MACHINE LEARNING PARA ANALISIS

Bienvenida e Introduccion

Introducción y bienvenida

Introducción a Cloud Machine Learning

¿Por qué correr cargas de Machine Learning en la nube?

Machine Learning genérico

Entendiendo las APIs de Machine Learning

Caso de Estudio: APIs de Machine Learning

Configurando el proyecto

Guardando imágenes en Cloud Storage

Guardar datos en Firebase

Generando comentarios en tu proyecto

Usando Cloud Vision para analizar imagenes

Usando el NLP API para analizar sentimiento

Analizando la respuesta del NLP en API

Deploy a Firebase

Machine Learning con tus datos

Introducción y caso de estudio: AutoML

Preparando datos para AutoML

Entrenar y predecir con tu modelo

Evaluando tu modelo

Consejos para el entrenamiento

Machine Learning para Análisis de Datos

¿Qué es BigQuery Machine Learning?

Introducción a la consola de BigQuery Machine Learning

Crear y predecir con BigQuery Machine Learning

Machine Learning a la medida

¿Qué es Cloud Machine Learning Engine?

Entrenando un modelo con Cloud Machine Learning Engine

Sirviendo un modelo con Cloud Machine Learning Engine

Finalizando el curso

Conclusiones y cierre

Bonus meme: Machine Learning en la frontera

Introducción a Machine Learning en la frontera (TF.js y edge tpu, kubeflow)

¿Cómo se usa Machine Learning para predecir datos?

MACHINE LEARNING APLICADO CON PYTHON

Introducción al curso

Introducción al curso de Machine Learning Aplicado con Python

Montar un ambiente de trabajo de Machine Learning

Cómo definir un problema de Machine Learning

Importancia de definir el problema en Machine Learning

Predecir el ingreso de películas de IMDB

Terminología de Machine Learning

Materiales del curso: Notebooks de Jupyter

El ciclo de ingeniería de Machine Learning

El ciclo de Machine Learning

Montar un ambiente de trabajo Pydata

Configuración del ambiente de trabajo con Docker

Construcción de contenedores de Docker

Qué es y cómo se utiliza Numpy

Arrays en Numpy

Operaciones aritméticas en Numpy

Preparación de los datos

Cargar los datos necesarios para el proyecto

Inspección de los tipos de datos

Inspección cuantitativa y de salud de los datos

Limpiar los datos

Manejo de datos faltantes

Modelación y evaluación

El objeto estimador de Scikit-Learn

Implementar un modelo de regresión (Lasso)

Ajustando Modelos de Machine Learning, Underfitting y Overfitting

Evaluando el modelo

Feature Engineering

Feedback del modelamiento

Análisis exploratorio

Continuando con el análisis exploratorio

Creación de features

Creando más features

Selección de features y la maldición de la dimensionalidad

Modelos y Evaluación más avanzada

Cross Validation

Selección de modelos

Curvas de aprendizaje

Introducción a Ensembles y Árboles de Decisión

Random Forest y Gradient Boosting Trees

Optimización de hiperparámetros

Conclusiones del curso

INTELIGENCIA ARTIFICIAL CON IBM WATSON

Watson Assistant

Introducción a Watson

Fundamentos de Watson Assistant: Intenciones y Entidades

Variables de contexto

Flujo de diálogo: Recomendaciones y metodologías

Desarrollo de conversaciones en Watson Assistant

Creación de un servicio en Watson Assistant

Integración con Watson Assistant: Intenciones

Integración con Watson Assistant: Entidades

Integración con Watson Assistant: Flujo de diálogo

Integración con Watson Assistant: Aplicación

Watson Assistant + Functions

Functions + Weather

Discovery + NLU

Fundamentos de Discovery y Natural Language Understanding

Watson Knowledge Studio

Creación de modelo de anotaciones con Knowledge Studio

Fundamentos de Knowledge Studio

Cierre del curso

Conclusiones

AVANZADO DE IBM WATSON

Introducción a Watson

Introducción a Watson

Introducción a Watson Assistant

Watson Assistant Tool

Watson Natural Language Understanding

Watson Discovery

Knowledge Studio

Watson Personality Insights

Introducción a Visual Recognition

Introducción a Speech to Text

Creación de Servicios

Watson Studio

Presentación de Watson Studio

Introducción a Watson Knowledge Catalog

Práctica: Creando un sistema de reconocimiento visual con Visual Recognition Models

Tutorial: Creación de un modelo de Analítica Predictiva con

Watson Machine Learning

Realidad Virtual Con Watson

Introducción al Proyecto y diálogo del personaje

Introducción a Unity: Importar y Manipular Objetos 3D

Watson SDK Para Unity

Voz y Transcripción 1, Preparando los displays de nuestro texto

Voz y Transcripción 2: Agregando el token de conexión con

Watson Text to Speech

Probando voz y transcripción

Diálogo y Conversación, Generando la capacidad de interpretar el input del usuario

Diálogo y Conversación, Respondiendo al input del usuario

Comandos de Voz

Análisis de Sentimiento utilizando Watson Natural Language Understanding
Terminando el Análisis de Sentimiento con NLU

SEGURIDAD INFORMATICA PARA EMPRESAS
Seguridad Informática en las empresas
Propósito general de la seguridad informática en las empresas
Visión general de la Seguridad Informática
Diferentes tipos de ataques y las técnicas modernas
Ejemplos de ataques informáticos modernos
Más allá de ataques informáticos
Papel de usuario final en la estrategia de seguridad de la empresa
Las personas como mejor mecanismo de protección
La seguridad empresarial, un compromiso que comienza con las personas
Alertas tempranas (entenderlas y reaccionar proactivamente)
¿Qué debo hacer para encontrar el software correcto?
Información personal y datos sensibles
Diferencia entre datos personales y empresariales
Consejos para proteger tu información en línea
Uso y administración de dispositivos móviles
¿Qué es cifrado de datos?
Tipos de amenazas
Yahoo, Adobe, LinkedIn (escenarios reales de ataques informáticos)
Leyes de protección de datos
Exposición a ataques y cómo protegerse de ellos
Dejar la contraseña escrita en post-it
Dejar el computador sin bloqueo
Compartir mi clave personal con los demás
Ejemplo de ingeniería social
Ejemplo de Phishing y mensajes
Víctima de un ataque
¿Qué hacer al ser víctima de un ataque?

INTRODUCCION A LA SEGURIDAD INFORMATICA
Introducción a la Seguridad Informática
Presentación del curso e introducción
Objetivo Seguridad Informática
Principios fundamentales de la Información
Términos relacionados con la Seguridad Informática
Principios básicos en la Seguridad Informática
Los 5 principios clave de la seguridad informática
Cómo instalar VirtualBox
Cómo instalar VMWARE
Laboratorio: Importación y Montaje de la Máquina Virtual en VirtualBox desde cero
Laboratorio: Importación y Montaje de la Máquina Virtual con VMWare
Laboratorio: Creación de usuarios en Windows aplicando el principio del menor privilegio

Laboratorio: Creación de usuarios en Linux aplicando el principio de menor privilegio
Seguridad en Redes TCP/IP
Introducción a la seguridad en redes TCP/IP
Modelo OSI simplificado
Seguridad por debajo de la capa 3
Modelo TCP/IP
Laboratorio: Realizar un escaneo de puertos
Laboratorio: Realizar análisis de protocolos con la herramienta Wireshark
Dispositivos de Seguridad Informática
Dispositivos de seguridad informática: Firewalls - Honeypot - Antivirus - Anti-Spam - VPN's - IPS
Implementación del Honeypot Dionaea
Hackers y Fases de un Hacking
¿Qué es un Hacker? Clases de Hacker
Qué es Hacking y fases del Hacking
Cómo instalar FOCA
Laboratorio: analizando metadatos usando la herramienta FOCA
Laboratorio: Obtener todos los Datos disponibles de algunos Sitios Web con WHOIS
Laboratorio: Realizar Email tracking
Manejo de Incidentes de Seguridad
Manejo de incidentes de seguridad ¿Qué es un incidente de seguridad? Administración de incidentes de seguridad
Recomendaciones para la atención y manejo de incidentes de seguridad
Introducción al Malware
¿Qué es malware? Tipos de malware
Laboratorio: Creación de virus sencillos utilizando la consola de comandos
Ingeniería Social
Ingeniería Social: Conceptos generales. Fases de un ataque de ingeniería social. Tipos de ataques y contramedidas
Estándares Internacionales de Seguridad
Estándares Internacionales de Seguridad: COBIT, ITIL, OWASP, CIS, NIST, ISO27001
Reto: análisis de las Políticas de Seguridad Informática de organizaciones reconocidas para identificar los componentes de seguridad de los estándares internacionales
Conclusiones
Conclusiones y cierre del curso
Bonus
¿Qué es un firewall?
Temario del curso

PENTESTING
Introducción a Pentesting
Introducción al curso de Pentesting
¿Qué es Pentesting?
Herramientas para un laboratorio de Pentesting
Instalación de las herramientas para un laboratorio de Pentesting
Presentación del caso de uso
Google Directives / Google Dorks
Information Gathering
Email Harvesting
Netcraft

Whois

Información a partir de DNS

Zonas de transferencia y brute-force a DNS

HTTrack

Escaneo y análisis de vulnerabilidades

Presentación de la fase de escaneo

Barrido de pings y barrido de puertos usando fping y nmap

Análisis detallado de una conexión usando nmap y wireshark

Barrido de pings con nmap y análisis de UDP

Escaneos Xmas

Escaneos null y escaneo final

Nmap Scripting Engine

Análisis de vulnerabilidades: Nessus

Análisis de vulnerabilidades: OpenVAS

Explotación

Explotación de vulnerabilidades con Metasploit

Arquitectura de Metasploit

Explotación de otras vulnerabilidades con Metasploit

Ataques con Armitage

Creación de diccionarios

Medusa e Hydra

Ataques a contraseñas locales (Linux)

Ataques a contraseñas locales (Windows)

Ataques al Hash con Rainbow Tables

Primeras configuraciones de un enrutador con dd-wrt

Otras configuraciones de un enrutador con dd-wrt

Configuración de interfaces en modo monitor con Airmon-ng

Inyección de paquetes usando Aireplay/ng

Ataque a una red wifi con Aircrack/ng

Introducción a la Ingeniería Social

Ataque con SET: 1

Credential Harvesting

Ataque con SET: 2

Introducción a los ataques web

Análisis de servicios web (Nikto)

Webscarab (Spidering e Intercepción)

Golismo - OWASP

Ataques de inyección de código

Continuando con el ataque de inyección de código

Ataques XSS

ZED Attack Proxy (Spidering, intercepción y escaneo)

Post-Explotación

Meterpreter

Más de Meterpreter

NetCat

Generando reporte

Estructura de un reporte

¿Cómo aprender más de Pentesting?

Bonus: Enlaces para aprender más

HACKING ETICO

Bienvenida y panorama general

Introducción y bienvenida

Conceptos básicos

Vulnerabilidades

Amenazas y ataques comunes

Instalación del entorno de pruebas Kali Linux

Instalación del entorno de pruebas Mutilidae

Instalación del entorno de pruebas Metaesplotable 3

Práctica: Cross-Site-Scripting, command injection y directorio transversal

Introducción al Ethical Hacking

¿Qué es hacking y hacker? Tipos de hacker

Fases del hacking

Práctica: Obteniendo información de fuentes abiertas con OSINT Framework y Google hacking

Práctica: Analizar un sitio web con Nikto y Spiderfoot

¿Es necesario un hacking ético? Hacking ético como profesión

Pentesting

Conceptos. ¿Qué es y qué no es una pentesting?

Tipos de pentesting. Fase Pre-ataque

Práctica: Buscando secretos en repositorios GIT

Introducción al escaneo de redes

Práctica: Escaneo de redes con Nmap

Fase de Ataque. Testing de aplicaciones web con Burp suite

Práctica: Explotando vulnerabilidades en metasploitable

Fase Post-ataque. Metodologías

Estándares y aspectos legales

PCI DSS. HIPAA. DMCA.ISO/IEC 27001

Contratos, Reglas de compromisos, Cláusula de no competencia y Acuerdo de confidencialidad

Convenio de Budapest. ¿Cómo vamos en América latina?

Salvuarda de evidencias digitales

Casos típicos de ataques

Malware y Análisis estático de malware

Malware y Análisis dinámico de malware

Sniffing y Cómo realizar un ataque Man in the middle automatico

Sniffing y Cómo realizar un ataque Man in the middle manual

Denegación de servicio. Ataque DOS con LOIC y HOIC

Ingeniería social

Controles y mecanismos de seguridad

Políticas de seguridad. Seguridad física y Controles de acceso

Práctica: Bypass autenticación QR (Reto 6 SANS Holiday hack)

Copias de seguridad. Defensa en lo profundo

Práctica: instalación del IDS snort

Gestión de riesgos y modelado de amenazas

FUNDAMENTOS DE PROGRAMACION

Empieza por acá

Mi primera línea de código

Fundamentos de Programación

¿Qué es HTML/CSS/JS?

JavaScript no es Java

Primeros pasos en el navegador con alert

HTML, CSS, JavaScript de verdad

Los apuntes de Freddy en PDF

Primer proyecto: Peso en otro planeta

Peso en otro planeta

Obteniendo datos del usuario

Flujo y condicionales

Segundo proyecto: Dibujando con Canvas

El DOM: nuestro lugar de trabajo en la web

Cómo funcionan Window y Document

Dibujando en el DOM

Funciones en JavaScript

Ciclos while y for en JavaScript

Eventos y Formularios en HTML y JavaScript

Detectar eventos del teclado con JavaScript

Dibujar en canvas con las flechas del teclado

Tercer proyecto: Villa platzi

Funciones matemáticas y números aleatorios en JavaScript

Uso y carga de imágenes en Canvas

Cuarto proyecto: Pakimanes

División, módulo y residuo en JavaScript

Clases y Arrays en JavaScript

Quinto proyecto: Cajero automático

Diagrama de Flujo del Cajero Automático

Implementación del Cajero Automático

Sexto proyecto: Cliente/Servidor

Modelo Cliente/Servidor

Primer servidor web con express

Programación de Hardware y Electrónica con Arduino

¿Cómo funciona un circuito electrónico?

¿Cómo programar un Arduino?

Programación de circuitos con C, Arduino y Sketch

Cómo programar un Arduino con Javascript y Node

Construye un Robot con JavaScript

Robot para riego de plantas en Arduino, Javascript y Node

Materiales de apoyo

Las mejores notas de los estudiantes

¿Qué lenguaje de programación aprender primero?

La Web con Visión Profesional

Contenido Bonus

Qué son tablas de verdad y compuertas lógicas

CODESTREAM

Introduction to CodeStream

Historia

Comunicación para programadores

Instalación de CodeStream

Usando CodeStream

Cómo hablar sobre el código

Administración de canales

Comunicación fuera de los canales.

Guardando y Compartiendo el Conocimiento

Comandos en CodeStream

Notificaciones

Integraciones con otras herramientas

Comunicación con miembros del equipo fuera de CodeStream

Configuración e integración con Slack

Usando Live Share con CodeStream

Recapitulación

Conclusiones del curso

Bonus

Adoptando CodeStream

Lectura Final

PROGRAMACION EN BASH SHELL

Bienvenida e Introducción

Introducción y Bienvenida al curso

Componentes de Linux, Tipos de Shell y Comandos de información

Bash scripting

Crear nuestro primer Script

Ejecutar nuestro script con un nombre único

Programación Shell Básica

Declaración de Variables y Alcance en Bash Shell

Tipos de Operadores

Script con Argumentos

Sustitución de Comandos en variables

Debug en Script

Script Interactivos

Capturar información usuario

Expresiones Regulares

Validar información

Paso de parámetros y opciones

Descargar información de Internet

Condicionales

Sentencias If/Else

If Anidados

Expresiones Condicionales

Sentencias Case

Iteración

Arreglos

Sentencia for loop

Sentencia while loop

Loop Anidados

Break y continue

Menú de Opciones

Archivos

Archivos y Directorios

Escribir dentro de archivos

Leer Archivos

Operaciones Archivos

Empaquetamiento

Empaquetamiento TAR, GZIP y PBZIP 2

Respaldo Empaquetado con clave

Transferir información red

Funciones

Crear funciones y Paso de Argumentos

Funciones de instalar y desinstalar postgres

Funciones sacar y restaurar respaldos en postgres

Cierre del curso

Cierre

FUNDAMENTOS DE MATEMATICAS

Aritmética

Introducción al curso

Operaciones básicas

Potenciación y sus propiedades

Radicación y sus propiedades

Orden de operaciones

Factorización

Recta numérica

¡A practicar lo aprendido!

Principios del algebra

Principios del álgebra

Simbología en el álgebra

Propiedades de las ecuaciones

Orden de despeje
Despejando exponentes y raíces en álgebra
Reto
¡Es hora de practicar!
Polinomios
Polinomios
Simplificando polinomios
La propiedad distributiva de la multiplicación
Desarrollando polinomios
Pon a prueba tus conocimientos y simplifica polinomios

Funciones
¿Que es una función?
Tabulación de funciones
Plano Cartesiano
¿Cuáles de las siguientes son funciones?
Gráficas
Test línea vertical
Funciones lineales
Todo se puede graficar
¡Es hora de graficar!
Cómo identificar funciones lineales a partir de una ecuación
¿cuál es la ecuación de las siguientes gráficas?
No pares de aprender

ALGEBRA

Conceptos Básicos
Variables y constantes.
Leyes de los signos (Aritméticos y de relación)
Signos de Agrupación
Leyes de los exponentes: multiplicación y División
Leyes de los exponentes: potencia de una potencia y radicación
Lenguaje algebraico y Expresiones algebraicas.
Material: Conceptos Básicos
Operaciones entre monomios
Suma y resta
Multiplicación
Operadores entre Monomios: División
Productos Notables
Binomio al cuadrado
Material: Binomio al cuadrado
Binomio a la n potencia
Binomios conjugados
Material: Binomios a la n potencia y conjugados

Factorización
Factor común monomio
Factor común por agrupación de términos
Trinomio cuadrado perfecto
Diferencia de cuadrados perfectos
TCP por adición y sustracción.
Trinomio de la forma ax^2+bx+c
Material: Factorización

Ecuaciones
Conceptos básicos
Ecuaciones: Ecuaciones de primer grado
Problemas de aplicación: Ecuaciones de primer grado.
Sistemas de ecuaciones lineales
Métodos de solución de problemas

Discriminantes en las Ecuaciones
Ecuaciones completas de segundo grado
Ecuaciones incompletas de segundo grado: Tipo 1
Ecuaciones incompletas de segundo grado: Tipo 2 y 3
Material: Ecuaciones Completas
Contenido bonus
Ecuaciones lineales con fracciones

CALCULO PARA ANALISIS DE DATOS

¿Qué es una función?
Bienvenidos al curso
Guía de estudio, ejercicios y explicaciones de cálculo.
Función, Dominio y Contradominio
Representación gráfica de la función
Funciones Aplicadas a Modelos Binarios
Introducción al Cálculo Diferencial
Introducción al Cálculo Diferencial
Derivadas de Funciones Algebraicas
Fórmulas básicas de derivación
Derivada de Funciones Algebraicas: Suma y Resta
Derivada de Funciones Algebraicas: Multiplicación I
Derivada de Funciones Algebraicas: Multiplicación II
Derivada de Funciones Algebraicas: Multiplicación III
Derivada de Funciones Algebraicas: División I
Derivada de Funciones Algebraicas: División II
Derivada de Funciones Algebraicas: División III
Derivada de Funciones: Potencia/Exponenciales
Derivada de Funciones Trigonómicas (trascendentales)
Derivada de Funciones Exponenciales

Aplicaciones de la Derivada
Optimización
Cálculo Integral
Introducción al Cálculo Integral
Integrales: Fórmulas básicas I
Integrales: Fórmulas Básicas II
Integrales: Fórmulas básicas III
Integrales: Fórmulas básicas IV
Integrales: Fórmulas básicas V
Cálculo Multivariable
Introducción al Cálculo Multivariable: Derivadas Parciales
Gradiente
Antes de tomar el examen
Examen de Cálculo para Análisis de Datos

EXCEL PARA EL ANALISIS DE DATOS

Introducción al curso
Bienvenidos al curso
¿Cómo piensa Microsoft Excel?
Siéntete cómodo con la barra de menú
Conoce la estructura de un libro de Excel
Protegiendo información en Excel
Listas desplegables sencillas
Listas desplegables dependientes
¿Imprimir en Excel? No será un problema nunca más
Reforcemos juntos varios conceptos
Reto: Tabla y formatos
Preparando la información

Bases de datos

Separa texto y después júntalo

Fórmulas básicas (numéricas y de texto)

Fórmulas de texto

Sumar.si y Promedio.si

Nombra rangos y haz operaciones

Promedio ponderado y fijación de datos

Practica el promedio ponderado

Pegado Especial

¿Qué es BuscarV?

BuscarV aplicado

Formula tus propios BuscarV

Ahora, ¿qué es BuscarH?

Practica el BuscarH

Fórmulas condicionales. ¿Qué son?

Fórmulas condicionales indispensables

Analizando la información

Formatos condicionales

Semáforos, celdas de colores y formulación

Tablas dinámicas

Tipos de gráficas y sus diferentes usos

Gráficas sencillas

Gráficas dinámicas

Formulación con tablas dinámicas

Modelo de gestión juntando conocimientos

Buscar objetivo

Macros - escribiendo y entendiendo código

Macros - grabando macros

Cierre del curso

Preparación examen

SQL Y MySQL

Introducción al Curso de SQL y MySQL

Nuevo Curso de SQL y MySQL

Bienvenidos al curso

Instalación de MySQL

La consola de MySQL

¿Qué es una base de datos?

CREATE

Comando CREATE

Tipos de columnas / Creación de la tabla books

Tipos de columnas / Creación de la tabla authors

Tipos de columnas usando / Creación de la tabla clientes

INSERT

Comando INSERT

Comando on duplicate key

Inserción de datos usando queries anidados

Bash y archivos SQL

Bash y archivos SQL

SELECT

Su majestad el SELECT

Comando JOIN

Left JOIN

Tipos de JOIN

5 casos de negocio

Continuación de casos de negocio

Comandos UPDATE Y DELETE

Consultas en MySQL

Super Querys

Comando mysqldump

Contenido Bonus

Bases de datos para Big Data

POSTGRESQL

- ¿Qué es PostgreSQL?
- ¿Por qué PostgreSQL?
- Configuración inicial
- pgAdmin
- Usuarios y roles
- Arquitectura de información
- Permisos de usuarios
- Creando una base de datos
- Datos y datos seriales
- Arreglos y cadenas de texto
- JSON
- Consultas (WHERE, IN, AND, OR, NOT, LIKE, ILIKE)
- Triggers
- PostGis

BASICO DE MONGODB

Introducción

Introducción y bienvenida

Bases de datos NoSQL

Definición de MongoDB y su ecosistema (herramientas de uso)

MongoDB Atlas

Instalación MongoDB en Windows

Instalación MongoDB Mac/Linux

Mongo Shell, configuración de clientes

MongoDB + Drivers

Operaciones CRUD

Bases de datos, Colecciones y Documentos en MongoDB

Operaciones CRUD desde la consola de MongoDB

Operaciones CRUD desde Compass

Configuración e instalación de dependencias para el proyecto platzi-mongo

Trabajando en nuestro proyecto PlatziMongo: Operaciones CRUD con Python y pymongo

Esquemas y relaciones

Tipos de datos

¿Qué son los esquemas y las relaciones?

Relaciones entre documentos

Trabajando en nuestro proyecto PlatziMongo: Diseñando el Esquema de Clases, Cursos y Carreras

Profundización de queries dentro de MongoDB

Operadores para realizar queries y proyecciones

Trabajando en nuestro proyecto PlatziMongo: Ejecución de Queries

Trabajando en nuestro proyecto PlatziMongo: Relaciones

Usando operadores para realizar Updates en arreglos

Operaciones avanzadas con Agregaciones

Consultas más rápidas con Índices

Recomendaciones para poner en producción tu cluster de Atlas

Recomendaciones de Arquitectura y Paso a Producción

Nuestra base de datos en un cluster de producción

Conclusiones

Resumen y Conclusiones

AZURE PAAS

Introducción al Curso de Azure PaaS

- Bienvenida al curso de Azure PaaS
- Diferencia entre IaaS y PaaS
- Escenarios donde IaaS o PaaS funcionan adecuadamente
- Diferencias de costo entre IaaS y PaaS
- Características de PaaS que aprenderás en este curso

Servicios de Almacenamiento de Azure (Storage Services)

- Servicios de almacenamiento de Azure
- Almacenamiento tipo Blob
- Azure Storage Explorer
- SDK de .NET para almacenamiento Blob
- SDK de .NET para almacenamiento Blob 2.0
- Almacenamiento tipo File
- Crear y administrar un archivo compartido
- Configurando una aplicación para File Storage con .NET
- Utilizando el SDK de .NET para File Storage
- Almacenamiento tipo Table
- SDK de .NET para almacenamiento Table
- Manejando tablas por medio del SDK de .Net
- Agregar entidades a una tabla
- Consultar las entidades de una tabla
- Actualizar entidades
- Eliminar tablas o entidades
- Almacenamiento tipo Colas
- SDK de .NET para almacenamiento de Colas
- Crear una cola e insertar un mensaje
- Seleccionar y eliminar mensajes
- Manejo de las pilas de mensajes
- API REST de los servicios de Almacenamiento
- Casos de uso para cada escenario
- Servicios de bases de datos en Azure
- Azure SQL
- Crear una base de datos desde el portal de Azure
- Configuración del firewall de Azure SQL
- Crear una base de datos desde línea de comandos
- Conectándome a mi base de datos desde VS Code
- Operaciones CRUD desde VS Code
- Operaciones CRUD desde SQL Management Studio
- ¿Qué es SQL Data Warehouse?
- Usando un API para acceder a blob storage
- ¿Qué es un almacén de datos?
- Creando un almacén de datos de SQL
- Conectándome a mi almacén de datos
- Extrayendo información de mi fuente de datos (importando información a mi fuente de datos)
- Mejores prácticas de uso de un almacén de datos (Despliegue de datos con Power BI)
- ¿Qué es una base de datos para PostgreSQL?
- ¿Qué es Azure Cosmos DB?
- ¿Cómo decidir que base de datos utilizar en cada caso?
- Creando una base de datos de PostgreSQL desde el portal
- Creando una base de datos de PostgreSQL desde terminal
- Conectándome desde .NET a mi base de datos

- Operaciones CRUD desde mi aplicación de consola
- Configurando mi entorno de Cosmos DB
- Mi cliente .NET para Cosmos DB
- Despliegue de datos con Power BI
- Servicios Web y Móviles
- Servicios Web y Móviles en Azure
- ¿Qué son las Azure Web Apps?
- ¿Por qué utilizar Web Apps?
- Características y herramientas principales de Web Apps
- Acceso y control de archivos en una Web App por FTP
- Creando un sitio web estático con una plantilla HTML
- Crea una web app y publicándola desde GitHub
- Escalando mi aplicación web desde terminal
- ¿Qué son las Mobile Apps? y ¿Por qué utilizar Mobile Apps?
- ¿Cómo funcionan las Mobile Apps?
- Cómo crear una Mobile App
- Configurando tablas simples en mi Mobile App
- Ejemplos de aplicaciones conectadas a mi Mobile App
- Editando mi información desde un cliente
- Utilizando el API REST de Mobile Apps
- Crear mi API con ASP.NET desde Visual Studio
- Publicar mi aplicación en Azure
- Agregando un cliente para una API REST en mi proyecto
- ¿Qué son las Logic Apps de Azure? y Conectores en Logic Apps
- Flujo de trabajo en Logic Apps
- Creando tu primera Logic App
- Instala las herramientas de Logic Apps en Visual Studio
- Crea y despliega tu Logic App desde Visual Studio
- Edita tu Logic App desde Visual Studio
- ¿Qué son los Azure Notification Hubs?
- Registrando mi aplicación en la tienda de Windows
- Configurando mi centro de notificaciones
- Conectando mi aplicación al centro de notificaciones
- Probando mis notificaciones desde el portal y desde consola
- Fusionando Mobile Apps con Notification Hubs
- Cierre del curso
- Despedida del curso

FIREBASE PARA LA WEB

Bienvenida e Introducción

- Introducción
- ¿Qué es Firebase?
- Creación del proyecto
- Consola Web de Administración
- Consola de administración
- Configuración de Firebase en el proyecto
- Autenticación de Usuarios
- Servicios de autenticación y configuración de usuarios
- Habilitar en la consola los servicios de autenticación por email y Google
- Probar creación de usuario y autenticación
- Servicios de autenticación con Google
- Firebase Web Login con Facebook
- Gestionando el estado del usuario
- Gestión de usuarios
- Exportación e importación de usuarios
- Gestión de la Base de Datos

- Firestore
- Firestore vs Firebase Realtime Database
- Configuración de Firestore al proyecto
- Desarrollando la inserción de datos
- Consultas en real time
- Realizando inserciones, consultas compuestas, límites y ordenamiento
- Reglas de seguridad
- Índices
- Almacenamiento de archivos
- Almacenamiento de Archivos en Firebase
- Subiendo archivos de nuestro blog
- Reglas de seguridad para el almacenamiento
- Hosting
- Qué nos brinda el Hosting de Firebase
- Redirects, rewrites y headers
- Desplegando la app al hosting
- Notificaciones Push
- Qué son las notificaciones en Firebase y cómo es su arquitectura
- Agregar funciones para recibir las notificaciones
- Implementando las notificaciones en el proyecto
- Probando las notificaciones desde la aplicación
- Conclusiones
- Conclusiones

FIREBASE CLOUD FUNCTIONS

- Bienvenida e Introducción
- Introducción al curso
- Qué es firebase y las cloud functions
- Consola web de administración
- Consola de administración y documentación
- Creación del proyecto de Cloud Functions
- Extendiendo la Gestión de Usuarios con Cloud Functions.
- Creación de las cloud functions de autenticación
- Probando y desplegando las Cloud Functions de autenticación
- Extendiendo Firestore con Cloud Functions.
- Creación de las cloud functions de Firestore
- Probando y desplegando las cloud functions de Firestore
- Extendiendo el Almacenamiento con Cloud Functions.
- Creación de las cloud functions para el storage
- Terminando de crear cloud functions para el storage y probando su funcionamiento
- Probando y desplegando las cloud functions de almacenamiento con Firestore
- Creando Cloud Functions HTTPS
- Creación de la cloud function HTTPS
- Probando y desplegando las cloud function HTTPS
- Bonus: Construyendo un controlador para la función HTTP
- Extendiendo el Hosting con Cloud Functions.
- Usando las firebase cloud functions desde el hosting
- Extendiendo crashlytics con Cloud Functions.
- Creación de las cloud functions de Crashlytics
- Probando y desplegando la cloud function de Crashlytics
- Extendiendo Analytics con Cloud Functions.
- Creación de las cloud functions cuando se registra un evento en las analíticas de Firebase
- Probando y desplegando la cloud function de analíticas

- Creación de pruebas unitarias a las Cloud Functions.
- Creando pruebas unitarias en firebase cloud functions
- Optimizando las Cloud Functions.
- Buenas practicas para las cloud functions
- Cierre
- Conclusiones finales y cierre

FIRESTORE PARA ANDROID

- Bienvenida e introducción
- Introducción al curso
- Configuración de Firebase para Android.
- Autenticación Anónima
- Autenticación anónima con Firebase
- Entendiendo Cloud Firestore
- Conociendo el Modelo de Datos de Firestore
- Gestión de Índices en Firestore
- Creación y Gestión de Datos
- Creación y gestión de datos en Firestore
- Estructura y creación de Datos en Firestore: Practica
- Estructura y creación de Datos en Firestore: Practica con Login Activity
- Transacciones y Escritura Batch en Firestore
- Lectura de Datos
- Lectura de Documentos de Firestore
- Consultas en Firestore: Sencillas y Compuestas, Orden y Límites de Datos
- Acceso a Datos Offline
- Reglas y Seguridad
- Reglas de Seguridad en Firestore
- Estructurando reglas de seguridad en Firestore
- Configurando y Probando Reglas de Seguridad en Firestore
- Trading App en Realtime
- Configuración Adapter Criptomonedas
- Configuración RecyclerView Criptomonedas
- Panel Balance Criptomonedas Usuario
- Actualizaciones Realtime Criptomonedas
- Generación Aleatoria de Criptomonedas
- Repaso
- Repaso conceptos principales del curso

FLUTTER

- Bienvenida e Introducción
- Desarrollando en Flutter
- ¿Qué es Flutter?
- Dart y Flutter
- Sintaxis de Dart
- Flutter para desarrolladores Android, iOS y Xamarin.forms
- Flutter para desarrolladores React Native
- ¿Cómo luce una app construída en Flutter?
- Primer reto
- Creando mi entorno de desarrollo
- Requerimientos de Hardware y Software
- Instalando Flutter en Android Studio y Visual Studio Code
- Composición de un proyecto en Flutter
- Interfaces en Flutter
- Programación Declarativa en Flutter
- Estructura de un programa en Flutter

Hola Mundo en Flutter

Widgets básicos

Widgets con estado y sin estado

Análisis de Interfaces de Usuario en Flutter

Definiendo los layouts de nuestra interfaz

Segundo reto

Widgets sin estado en Flutter

Flutter Widgets: Container, Text, Icon, Row

Flutter Widgets: Column

Recursos en Flutter: Tipografías y Google Fonts

Widget Image

Widget Apilando Textos

Widgets Decorados

Widget Imagen Decorada

Widget ListView

Widget Button, InkWell

Tercer reto

Widgets con estado en Flutter

Botones en Flutter

Clase StatefulWidget: Cómo se compone

Widget Floating Action Button

Widgets BottomNavigationBar

Generando Navegación en BottomNavigationBar

Personalizando nuestro BottomNavigation Bar a Cupertino iOS

BottomBar

Cuarto reto

Fin del Curso

Conclusiones

¡Terminamos!

AVANZADO DE FLUTTER

Arquitectura de Aplicaciones

Bienvenida

¿Qué es una Arquitectura de software?

Tipos de Arquitecturas para Flutter

Arquitectura BLoC en Flutter

BLoC + Clean Architecture en Flutter

Estructurando nuestro proyecto utilizando la Arquitectura BLoC

Haciendo BLoC Singleton en Flutter

Aplicando Providers al Proyecto

Firebase y Flutter

Integrando Firebase Authentication, Cloud Firestore y Firebase Storage al proyecto

Integrando Firebase a Flutter para Android

Integrando Firebase a Flutter para iOS

Creación de Pantalla de Login con Flutter

Creando botones reutilizables en Flutter

Autenticación de Firebase con Google

Implementando Firebase Authentication en BLoC Pattern

Streams en Flutter

Manejando una sesión con Firebase Authentication y Flutter

Implementando Google SignOut en BLoC

Implementando Google SignOut en View

Monitoreando y validando la conexión al Sign con Google

Mostrando los datos de usuario de Google en la interfaz en Flutter

Cloud Firestore de Firebase en Flutter

¿Qué es Cloud Firestore de Firebase?

Analizando un modelo de datos no relacional

Creando un Modelo de datos en Cloud Firestore

Enviando datos a Cloud Firestore

Creando un Widget gradiente personalizado

Manejo de Desbordamiento de Texto de Widget Text

Botón de Back en un AppBar en Flutter

Navegación entre pantallas en Flutter

Widget Text AppBar personalizado en Flutter

Widget TextField personalizado en Flutter

Creando una Safe Area para una interfaz que tiene un AppBar

Widget TextField con iconos en Flutter

Retocando el CardView

Mostrando imagenes en un CardView

Creando un botón de Submit en Flutter

Envío de datos de un formulario en Flutter

Subiendo datos a Firestore de Firebase

Formularios en Flutter

Acceso al Hardware con Flutter

Acceso a la cámara en Flutter

Librerías de acceso a Hardware en Flutter

Firebase Storage en Flutter

Qué es y cómo funciona Firebase Storage en Flutter

Subiendo una imagen a Firebase Storage desde Flutter

Querys avanzados en Cloud Firestore de Firebase en Flutter

Usando el caché para cargar imágenes más rápido

Manejo de imágenes en Cloud Firestore

Cloud Firestore insertando referencias y arrays en la base de datos

Descargar imágenes de Firebase Storage y mostrarlas en Flutter

Procesando datos con BLoC Pattern

Trayendo datos de Cloud Firestore

Persistiendo datos de un usuario logueado

Aplicando Filtros en Cloud Firestore

Construyendo los Places en la pantalla de Home

Mostrando los Places en la pantalla de Home

Actualizando datos en tiempo real

Manejando la lógica de likes, como botón toggle.

Insertando y obteniendo referencias en datos de Firestore.

StreamController, sink, add y StreamBuilder

Conclusiones

Conclusiones

DART

Bienvenida e Introducción

Introducción y Bienvenida

Dart vs. Go vs. JavaScript vs. Kotlin

DartPad y qué editores utilizar

Básicos 1: Variables y Operadores

Variables: Números (int,double), String y booleanos

Colecciones

Diccionarios

Constantes (final y const)

Operadores aritméticos, de asignacion y lógicos

Básicos 2: Funciones y Parámetros

Declaración de funciones y la función Main()

Funciones de sintaxis corta (Arrow) y Anónimas (Lambda)

Parámetros requeridos, opcionales, posicionados, nombrados y por defecto

Básicos 3: Flujos de Control

If-Else y Expresiones Condicionales

Switch y Case

For

While y do-while

Break y Continue

Clases

Definición de clases y miembros

Constructores por defecto, por parámetro o nombrados

Métodos Getter y Setter y variables privadas

Herencia

Clases Abstractas

Interfaces implícitas

Utilizar Flutter en Dart

Iniciar y estructurar un proyecto en Flutter

Contenedores

Creando un botón personalizado con el Widget Container

Modelos de datos y peticiones http

Trabajando datos http y Widget Future Builder

Ejecutando el mismo código en Android y iOS

Conclusiones

Dart en el futuro

Segmentando una Notificaciones con Firebase

Firebase Remote Config

¿Qué es Firebase Remote Config?

¿Cómo funciona Firebase Remote Config?

Creando una Configuración Remota desde Firebase en Android

Sincronizando una Configuración Remota desde Firebase

Firebase Test Lab y Crash Reporting

¿Qué es Firebase Test Lab y Crash Reporting

¿Cómo funciona Firebase Test Lab y Crash Reporting?

Creando un Test Lab con Firebase

Implementado Firebase Crash Reporting en Android

Firebase Dynamic Links

¿Qué es Firebase Dynamic Links / Invites / Indexing?

¿Cómo Funciona Firebase Dynamic Links e Invites?

Creando una Firebase Dynamic Link

Generando un Firebase Invite de un Dynamic Link

Firebase AdMob

¿Qué es Firebase AdMob?

¿Cómo funciona AdMob?

Creando un anuncio AdMob con Firebase

Fin del Curso

Conclusión del Curso Profesional Firebase para Android

PROFESIONAL DE FIREBASE PARA ANDROID

Introducción

¿Qué es Firebase?

Configuración de un proyecto en Firebase

Firebase Authentication

¿Qué es Firebase Authentication?

¿Cómo funciona Firebase Authentication usando Correo y

Contraseña?

Integrando SDK Firebase Authentication Correo y Contraseña

Sign In y Sign Out Correo y Contraseña en Firebase Authentication

¿Cómo funciona Firebase Authentication con Google?

Integrando SDK Firebase Authentication Google

Sign In y Sign Out Google en Firebase Authentication

Autenticando con Google Obteniendo datos de usuario

¿Cómo funciona Firebase Authentication con Facebook?

Dando de alta una aplicación Android en Facebook Developers

Firebase Authentication con Facebook

Firebase Real-Time

¿Qué es Firebase Real-Time?

¿Cómo funciona Firebase Real-Time con JSON?

Accediendo a los nodos de Firebase Real-Time

Implementando SDK Firebase Real-Time en Android

Firebase Real-Time Leyendo, Escribiendo y Actualizando datos

Firebase Real-Time Eliminando Datos

Firebase Storage

¿Cómo funciona Firebase Storage?

Integrando SDK Firebase Storage en Android

Firebase Storage Subida de Archivos

Firebase Storage Descarga de Archivos

Firebase Notifications y Cloud Messaging

¿Qué es Firebase Notifications y Cloud Messaging?

¿Cómo funciona Firebase Notifications?

Integrando SDK Firebase Notifications

Enviando una Notificación con Firebase

DE JQUERY A JAVASCRIPT

Introducción

Bienvenidos al curso

La historia de jQuery

Presentación del proyecto

De jQuery a JavaScript

Variables y Funciones

Promesas

Tutorial de Ajax en jQuery y Javascript

Funciones asíncronas

Selectores

Creación de templates

Creación de DOM

Reutilizando funciones

Eventos

Clases y estilos CSS

Creación de elementos y asignación de atributos

Formularios

Desestructuración de objetos

DataSet

Encontrando elementos en la lista

Animaciones

Manejo de errores

LocalStorage

Obteniendo los datos almacenados

Conclusiones del curso

SASS

Preprocesador Sass

Introducción a los cursos de preprocesadores

Introducción al Curso de Sass

Diferencias entre Sass, Stylus y Less

Compilación de código en archivos CSS

Instalación

Estructura de CSS

Variables

Variables

Reto

Solución al reto

Anidaciones

Anidaciones

Mixins

Mixins

Continuando con Mixins

Uso de la directiva `content` (`block` en Stylus)

Extend

Funciones

Funciones

Directiva

Ejemplos de funciones

Reto

Reto - Solución

Controles de Flujo

Listas y directiva each

Ciclos FOR/EACH

Condicionales

LESS

Cómo funciona Less

Introducción a los cursos de preprocesadores

Introducción al curso

Instalación de Less

Tutorial y Reto

Variables: Definición y ejemplos

Anidaciones

Mixins

Mixins Paramétricos

Solución del reto

Funciones de Less

Ejemplo de funciones con Less

Controles de Flujo

STYLUS

Introducción al curso

Introducción a los cursos de preprocesadores

Introducción a Stylus

Compilación e instalación

Cómo funciona Stylus

Sintaxis

Anidaciones

Reto: Instalación con Grunt

Herramientas de Stylus

Variables

Estructura de CSS y propuesta de reto

Solución del reto y añadir HTML

Mixins

Mixins Paramétricos

Funciones internas

Condicionales

Directiva for

BOOTSTRAP

Introducción al curso

Repositorio del curso

Bootstrap 4 ¿Qué trae de nuevo esta versión?

¿Que es un framework frontend?

Nuestro Proyecto: Hola Mundo de Bootstrap

Creando el sitio web

La grilla de Bootstrap

Reto: La grilla de Bootstrap

El footer

El header de nuestro sitio

Creando un carousel de imágenes

Agregando texto informativo del evento

Agregando botones

Las cards de Bootstrap 4

Pastillas de texto

Agregando un contenedor de ancho completo

Los formularios de Bootstrap 4

Agregando un tooltip

Scrollspy: Conociendo la ubicación del usuario en el header

Agregando un modal para comprar tickets

Un nuevo formulario para completar la compra

Deploy a producción

Poniendo nuestro sitio en producción

Conclusiones del curso

SISTEMAS DE DISEÑO PARA DESARROLLADORES

Introducción

Introducción y presentación

Conoce Sistemas de Diseño populares

Instalación y flujos de trabajo

Bonus: Personalizando la Terminal

Instalación y flujos de trabajo

Instalación de Node / Express desde 0

Bonus: html, css, js

Automatización

Automatización (Pre procesadores)

Continuando con la automatización

Paradigmas

SMACSS: Arquitectura de CSS escalable y modular

OCSS: Orientación de objetos en CSS

BEM

Bonus: Bootstrap

Naming

Naming

Qué son los tokens

¿Qué son los tokens?

Espaciado

Espaciado

Grillas

Grillas

Bonus: Animando en CSS

Mi primer componente

Mi primer componente

Documentación

Documentación

Testing

Testing
Bonus: Debug en el navegador
Deploy y versionamiento
Cómo empezar en Github
Deploy y versionamiento
Design review
Design review
Cierre

WPO: OPTIMIZACION DE CARGA

Google Page Speed Insights
Bienvenida al curso
Repositorio del curso
Deploy del Portafolio a Github Pages
Google PageSpeed Insights
Eliminando bloqueos de visualización, fuentes
Comprimiendo Hojas de estilo
Critical CSS o Critical Path CSS
Optimizando Imágenes
Ejercicio de optimización
Imágenes y densidad de pixel
¿Con qué curso seguir?

BASICO DE IONIC

Temario del curso
Introducción
Introducción al Curso de Ionic
Pre-Requisitos
Treasure Hunting GeoCatching
Instalación y overview de Ionic CLI
Preparación de entorno de desarrollo en Mac
Preparación de entorno de desarrollo en Windows
Preparación de entorno de desarrollo en Linux
Aspectos Básicos
Crear una Aplicación
Corriendo Aplicación en el Browser
Estructura de Archivos
Estructura de una vista de Ionic
Navegación
Creando un nuevo page
Navegar a un page
Navegación por Tabs
Parámetros de navegación
Navegación hacia atrás
Navegación raíz
Estilos (Theming)
Introducción a estilos
Colores Personalizados
Alineación y transformación de texto
Padding y Márgenes
CRUD en Ionic y Firebase
Generar proyecto e instalar dependencias
Agregar y configurar AngularFire
Agregar servicio de acceso a datos
Crear una lista con array estático
Crear una vista para agregar records

Agregar la habilidad de ver records
Crear lista dinámica y agregar la habilidad de editar records
Agregar la habilidad de eliminar records
Implementando Login con Facebook
Deployment
Correr aplicaciones en emulador
Correr aplicación en dispositivos Android
Correr aplicación en dispositivos iOS
Publicando nuestra aplicación en Android y iOS stores
Conclusiones del curso

AVANZADO DE IONIC

Temario del curso
Introducción a Ionic
Intro a Ionic Framework, Angular, Platzinger, y Ionic CLI
Instalación del entorno de desarrollo en Mac OS
Instalación del entorno de desarrollo en Windows
Estructura de nuestra App con Ionic
Arquitectura de una app en Ionic
Proyecto: Platzinger App
Repositorio del curso
Qué son pages y cómo generar las primeras de nuestra app
Navegación básica entre pages
Reto: Página “Acerca de” accesible desde el menú lateral
Tomando los conceptos compartidos de Angular y Ionic
Basics de Angular y TypeScript
Aclaraciones Ionic vs Angular
Cómo usar tipos de datos con typescript
Qué son e implementando interfaces de TypeScript
NgFor aplicado en la lista de usuarios
NgIf aplicado en la lista de usuarios
Navegación con parámetros
Accediendo a nuestros usuarios con conversation.ts
Creando un servicio de usuarios e inyectando el servicio en nuestros pages
Pipes en TypeScript (Date, Number, Json)
Creando nuestro propio 'Pipe' para buscar entre nuestros contactos
Reto: Mostrar un icono diferente según el status del usuario
Agregando estilos con CSS a nuestra aplicación
Cómo usar estilos CSS y referenciar recursos
Implementando estilos en la pantalla de login
Implementando estilos en la pantalla de home
Implementando estilos en la pantalla de conversación
Usando ngStyle y ngClass para añadir estilos dinámicos
Cambiar icono según status
Acceso a datos remotos
Información importante antes de continuar
¿Qué es Firebase?
Reemplazo del servicio de usuarios por un servicio dinámico con conexión a Firebase
Login/Registro de usuarios con email y password
Login/Registro de usuarios con Facebook
Implementando Log Out
Creando un CRUD implementando un API REST HTTP (con diversos backends)
Opciones y Acceso Nativo

Conectándonos al API de Google
Crear page de Perfil
Editar datos básicos del perfil de usuario
Configurando Android Studio
Capturar fotos de cámara y álbum
Preparando aplicación para recibir fotos desde la cámara y galería del Android
Guardando y asignando una foto de perfil al usuario desde la cámara
Preparando aplicación para utilizar geolocalización

Conversación

Más estilos
Enviando un mensaje (con reproducción de sonido)
Mostrando conversación
Enviando un zumbido (con reproducción de sonido y animación)

Amigos

Planeación previa para la característica / Cómo generaremos las solicitudes
Enviando una solicitud de amistad
Revisando solicitudes de amistad
Aceptando solicitudes de amistad
Mostrando sólo amigos
Componentes anidados (con paso de parámetros)

Deploy

Exportando app para iOS y complementando con Xcode
Exportando app para web, complementando con Firebase Hosting
Conclusiones del curso

Contenido Bonus

Publicando nuestra aplicación en Android y iOS stores
Cuéntanos

PWA CON ANGULAR

Introducción

Qué son las PWAs?
PWA vs SPA
Setup del Proyecto en Mac, Windows y Linux
Presentando el Editor de Código
Generando nuestro primer PWA

Service Worker

Correr nuestro app en modo de producción
Qué es el service worker de Angular?
Gestión de versiones del Service Worker
Configuración del Service Worker y Caching Policy
Propiciando el Click to Install banner

Estilo

Comprendiendo el concepto de responsividad
Introducción a Angular Material
Instalación de Angular Material
Configuración de Angular Material en el proyecto

Datos Dinámicos

Introducción a Acceso a Datos
Implementando Firebase
Creando un servicio de conexión a Firebase
Creando vista de lista y detalle
Guardando notas en Firebase
Obteniendo notas de Firebase
Editando notas

Implementando Caching
Push Notifications
Implementando Login
Configurando Push Notifications
Cómo instalar curl en Mac, Windows y/o Linux
Configurando Firebase Hosting y subiendo nuestro proyecto a producción
Implementando Push Notifications
Cloud Functions
NativeScript: Características Nativas en un Web App con Angular
¿Dónde Aprender Más?

UNIT TESTING PARA MEAN CON JASMINE

Bienvenida e Introducción

¿Qué son las pruebas unitarias?
¿Porqué hacer pruebas unitarias?
Creando un framework de pruebas básico
Mi primera prueba unitaria en JavaScript
Las funciones expect() y it()
Organizando el código para correr en la web
Organizando el código para correr utilizando nodejs

Análisis estático de código

Herramientas de análisis estático de código
ESLint: Agregando alertas a nuestro código con ECMA Script
Herramientas de corrección de estilo
Herramientas de tipado

Trabajando con Jasmine en el frontend

Profundización en SpyOn: Comandos más utilizados y cómo ponerlos a prueba
Configurar un ambiente de trabajo para trabajar con el framework jasmine
Configurar Jasmine utilizando Node.js
Primer set de pruebas con Jasmine
Diccionario Jasmine

Probando Nodejs apps con Jasmine

Introducción al módulo de testing del lado del servidor
Configurando el proyecto Jasmine utilizando npm
Agregando Plugins a Jasmine
Configurando nuestro reporter
Pruebas en el servidor: Verificando un status 200 en GET
Pruebas en el servidor: Probando el método GET y Reto con FindByID
Pruebas en el servidor: Probando el método POST (request to server)
Pruebas en el servidor: Probando el método POST (request to PDF)

Probando Angular apps con Jasmine

Tipos de pruebas
Pruebas en el frontend: Probando el componente principal (App)
Configurando los ciclos de vida
Creando las pruebas del formulario: Configuración
Creando las pruebas necesarias para un formulario. (Primeras pruebas)
Probando el caso de exito al guardar un elemento en el servidor.
Trabajando con event emitters
Testeando ngZone y navegación.
Configurando el TestBed de el componente PINs

Creando un espia sobre el objeto window

Pruebas de integración de Angular apps con Jasmine

Ejecutando funciones a través de eventos en el template

Probando la navegación

Probando servicios con HTTP

Completando las pruebas del servicio HTTP

Cierre del curso

TYPESCRIPT CON ANGULAR POR BELATRIX

Fundamentos de TypeScript

¿Qué es Typescript?

Resumen de Typescript

Types, Type Inference y Type keyword

Clases

Interfaces

Shapes

Union types

Intersection types

Function type

Decorators: aplicación en métodos

Decorators: aplicación en clases

Decorators: aplicación en propiedades

Decorators: aplicación en parámetros

Preparando el entorno para nuestro proyecto

¿Qué es Angular?

Angular CLI y Visual Studio Code

Creando nuestro proyecto con Angular CLI

¿Qué es Firebase? Implementando Firebase en nuestro proyecto

Diferencias entre Angular, React, Vue

Desarrollo de la aplicación

Presentación de la aplicación

Creación de las interfaces

Creación e implementación de endpoints en Firebase

Creación de la tabla de jugadores

Implementación de nuevo jugador a la tabla 1

Implementación de nuevo jugador a la tabla 2

Debugging de la aplicación

Implementación de edición de un jugador de la tabla

Implementación de eliminación de un jugador en la tabla

Conclusiones

Conclusiones

ALGO

Temario del curso

- Uso de GitHub e Instalación de dependencias
- Introducción a Webpack
- Instalación de las dependencias y configuración de Webpack
- Instalación de html-webpack-plugin
- Crear la estructura de archivos y configuraciones iniciales
- Uso de Babel
- Configuración de ESLint y Prettier
- Deployando la aplicación con Now.sh
- Construcción de Interfaces con ReactJS
- Creación de componentes de UI con styled-components
- Uso de Styled-components
- Uso de iconos y creación de NavBar

- Uso de React Hooks
- Introducción a GraphQL
- Introducción a Apollo Server y React Apollo
- Instalación de React Apollo
- Manejo de lógica de aplicación con ReactJS
- Introducción a Reach Router
- ¿Por qué usar Reach Router?
- Cómo crear una ruta
- Crear la ruta
- Creación de componentes para navegación a Favoritos y Home
- Cómo crear subrutas
- Crear una ruta protegida
- Conectar una aplicación web a GraphQL
- Añadir JSON Web Token (JWT)
- Crear un custom-Hook para saber si el usuario ha iniciado sesión
- Usar React.Context
- Registro de usuarios
- Manejo de Inicio de sesión
- Manejo de Cierre de sesión
- Introducción al SEO con ReactJS
- ReactJS Helmet
- Componente para controlar los "heads" de nuestra app
- SEO en nuestra app de React
- Introducción a las PWA
- Usar workbox para cachear requests
- Deployando nuestra aplicación con Now
- Cómo medir el performance de nuestro sitio
- Uso de Lighthouse para auditar nuestra web app
- Usdo de Webpack Bundle Analyzer
- React.lazy: carga diferida de componentes
- Implementación de Carga en diferido de las rutas
- Uso de React.memo para evitar renderizados innecesarios
- Testing básico con Cypress

NEXT JS

Temario del curso

Introducción a Next.JS

¿Qué es Next.JS?

Creando nuestra primera página

Styled JSX

Server Side Rendering

¿Qué es Server Side Rendering?

Intro a getInitialProps

Utilizando el componente Link

Recibiendo Parámetros

Performance de Get Initial Props

Vista de Podcasts

Componentes Reutilizables

Creando componentes en React

Reorganizar la vista de podcasts

Navegación Avanzada

Gestionando Errores

Personalizando errores

¿Cómo diseñar URLs?
Configurando Next Routes
Implementando Next Routes
Vistas Híbridas
Implementar el Modal
Agregando un loader
Publicando nuestra app
Mejores prácticas en Github
Publicar la app con now
Conclusiones del curso

REACT JS

Introducción a React.js
¿Qué es React.js?
Pre-requisitos
Herramientas que usaremos
Create-react-app
Fundamentos
Clonar el código de GitHub
ReactDOM.render
JSX
Creación y diseño de componentes
¿Qué es un componente?
Nuestro primer componente
Cómo aplicar estilos
Props
Nuestra primera página
Enlazando eventos
Manejo de estado
Levantamiento del estado
Listas de componentes
React Router
Introducción React Router
División de la aplicación en rutas
Mejorando la User Interface con un Layout
Component Lifecycle
Introducción del ciclo de vida de un componente
Práctica del ciclo de vida
Llamadas a un API
Introducción Llamadas a un API
React.js: Cómo traer datos de un API en React
Solicitando datos (GET)
Mejorando la Experiencia de Usuario durante una petición
Enviando datos (POST)
Manejando los estados de la petición durante el POST
Actualizando datos (PUT)
Actualizaciones automáticas
Mejorando la UI
Los detalles de un Badge
UI Components y Container Components
Portales
Modales
Hooks
Search filter
Conclusion
Conclusión
Temario del curso

REACT ROUTER

Introducción al curso
Bienvenidos al Curso de React Router
¿Qué vamos a hacer en el curso?
Instalando dependencias
Configurando un SPA en Webpack
React Router API
BrowserRouter
Tipos de enrutadores
Link y NavLink
Route
Parámetros de navegación
Páginas de contenido no encontrado 404
Redirect - Switch
Prompt, validación antes de dejar la página
Manipulando el historial
Obteniendo el historial desde cualquier componente
Server Render
Configurando Webpack para server render
Creando una aplicación de Express
StaticRouter
ReactDOMServer
Creando una página única por video
Conclusiones del curso

PROGRESSIVE WEB APPS CON REACT JS

Introducción a PWA
¿Qué es una PWA?
Progressive Web App vs. Native App
El proyecto
Google Lighthouse
Creando un Web Manifest
Implementar el Manifest
Service Workers
¿Qué es un Service Worker?
Introducción a Workbox
Implementando Workbox
Aplicando Estrategias de Carga
Google Analytics Offline
Experimentar con Workbox
Funcionalidades Avanzadas
Web Share API
Trabajando Offline
Notificaciones
Publicando la app

REDUX POR BEDU

Bienvenido al curso
Bienvenido al curso
¿Qué conocimientos me recomiendan para tomar este curso?
Repaso React
¿Qué es React y cómo funciona?
Preparando nuestro entorno de trabajo
Creación de la app con React
Agregando funciones a la app con React
Stateful vs Stateless

Ciclo de vida de React

Manejando promesas

React Router DOM

Introducción a Redux

¿Qué es Redux, cuándo usarlo y porqué?

Fases de Redux

Introducción: las fases de Redux

Store

Reducers

Conexión a un componente

Action Creators

Redux Thunk

Explicación teórica: ciclo completo de Redux

Práctica: ciclo completo de Redux

Fases Extra

Archivos Types

Try Catch

Escenarios asíncronos

Componente Spinner

Componente Fatal

Tabla como componente

Compartir información en Redux

Introducción Compartir información en Redux

Parámetros por URL

Compartir Reducer

Múltiples Reducers

Llamando a múltiples reducers en una acción

Uso del estado en la acción

Evitar segundas búsquedas

Inmutabilidad

Evitar sobrescritura

Validación compuesta

Validación de errores

Modificando respuesta de url

Estado con interacción

Mostrar componentes dinámicamente

Llamadas asincronas dinámicas

Props por herencia vs estado

Estado compartido

Métodos HTTP

Introducción a métodos HTTP

Nuevo ciclo Redux

Normalizar datos

Mapear Objetos

Componente para agregar tarea

Manejar inputs con Reducer

POST

Deshabilitando botón

Redireccionar

Reutilizar componentes

PUT

DELETE

Últimos detalles

Conclusión

Conocimientos adquiridos

Qué hacer a continuación

DESARROLLO DE APLICACIONES WEB CON GATSBY

Temario del curso

• ¿Qué es gatsby?

• ¿Por qué Gatsby?

• SSR en Gatsby: ¿Cómo se construye y consume un sitio o app de gatsby?

• Requisitos previos y Herramientas de desarrollo

• Configuración de la herramienta Eslint

• Gatsby y Gatsby CLI y Starters

• Platzi Swag y estructura de archivos

• Ecosistema de plugins

• Rol de React en Gatsby

• Router en Gatsby y Componente Link

• Layout en Gatsby

• Rol de graphql en Gatsby

• Datos en Gatsby

• Queries, conexiones(edges) y nodos en Gatsby

• Query en React

• Instalación y configuración de plugins

• Imágenes como fuente de datos

• Plugins de transformación

• Styled components

• Styled components globales

• Introducción a la API de Stripe checkout

• Agregar productos a nuestra tienda en línea

• Productos en React

• Creando páginas en función de los datos

• Manejo de Gatsby templates

• Vistas de detalle en Gatsby

• useStaticQuery

• Carrito de compras: uso de useContext

• Carrito de compras: uso de vistas

• Agregar productos al carrito

• Comprando productos

• Gatsby build y SSR

• Despliegue a Netlify

WEBPACK

Temario del curso

• Introducción a Webpack

• Configurando un nuevo proyecto de Javascript

• Creando nuestro primer bundle con Webpack

• Iniciando un webpack.config

• Cargando configuraciones por defecto y personalizadas

• Múltiples puntos de entrada

• Manejo de assets con Loaders

• Introducción al uso de plugins

• Servidor de desarrollo

• Hot module replacement

• Soporte de Javascript moderno

• Trabajando con Javascript moderno

• Soporte de JSX (React)

• Trabajando con React

• Soporte imágenes, fuentes y videos

• Estilos con preprocesadores

- Evitar código duplicado
- Añadiendo un Dymanic Link Library
- Dynamic imports
- Setup de Platzi Badges
- Configuración de desarrollo para Platzi Badges (React.js)
- Configuración básica de producción para Platzi Badges (React.js)
- Configuración óptima de producción para Platzi Badges (React.js)
- Versionando, limpiando y comprimiendo archivos
- Setup de Platzi Rooms
- Configuración de desarrollo para Platzi Rooms (Vue.js)
- Configuración de producción para Platzi Rooms (Vue.js)

HIBERNATE Y JAVA SPRING

Bienvenida e Introducción

Instalar ambiente de desarrollo en Windows

Introducción y bienvenida

¿Qué es una Aplicación Empresarial?

¿Cómo funcionan las Aplicaciones Web?

Servidores Web vs Servidores de Aplicación

Instalar ambiente de desarrollo Mac

Patrón de Diseño MVC

Características de los Servlet

Utilización del Servlet

Características y práctica de JSP

Comenzar con Spring Boot

Arquitectura de las Aplicaciones Web

Características de Spring Framework

Características de Spring Boot

Crear, cargar y configurar una aplicación en Spring Boot

Crear la estructura del proyecto

Spring Data JPA

Persistencia de objetos

Spring Data, Spring Data JPA

Iniciar la base de datos postgres utilizando Docker

Optimizar Pojos con Lombok

Mapear Tablas de la Base de Datos

Configurar la persistencia con Hibernate

Manejo de Negocio

Añadir JPA Repository

Implementar las operaciones de Consulta con JPA Repository

Implementar servicio de negocio para las operaciones CRUD

Manejar la transaccionalidad de un servicio de negocio

Spring REST

Instalar y configurar Swagger

Características de los Servicios Web y Spring Rest

Implementar las operaciones REST para crear y actualizar información

Implementar las operaciones REST para eliminar y consultar información

Documentación de los servicios REST con Swagger

Spring Security

Manejar la seguridad en una aplicación web

Características, instalación e integración de Spring Security

Personalizar la página de Login

Despliegue Aplicaciones Docker

Realizar la configuración de Docker Plugin

Crear una imagen docker de la aplicación

Subir la aplicación a docker hub

Ejecutar la aplicación dockerizada

Despedida del curso

GRAPHQL

Introducción

Introducción al curso y prerequisites del curso

¿Qué es GraphQL?

Conceptos básicos

Schema y types

Queries y Resolvers

Sirviendo el API en la web

Custom Types

Argumentos

Configuración de base de datos

Integrando una base de datos

Mutations e Inputs

Repaso - Creando el tipo Estudiante

Nested Types

Resolver de tipos

Errores

Conceptos avanzados

Alias y fragments

Variables

Enums

Interfaces - Tipo Monitor

Directivas

Unions

Consumiendo el API

Preparando API para producción

HTTP requests

Cientes de GraphQL

Consumiendo el API desde un frontend simple

Cierre del curso

JAVASCRIPT TESTING CON JEST

Introducción al curso

¿Qué son las pruebas Unitarias?

Herramientas para pruebas en Javascript

Configuración

¿Qué es Jest?

Instalación y configuración

Creando nuestra primera prueba unitaria

Matchers

Common Matchers / Comparadores Comunes

Numbers

Truthness / Verdadero

Arrays

Coverage

Strings

Monitoreo

Setup de pruebas

Testing Asynchronous Code

Callbacks

Promesas

Reject y resolve
Async/Await
Aplicaciones de testing para Async/Await

Snapshot Testing

Snapshot Testing

Excepciones Snapshot

Implementando Jest en diferentes Frameworks

Preparar proyecto para trabajar con React JS

Preparar proyecto para trabajar con Vue JS

Preparar proyecto para trabajar con Express JS

Beneficios de las Pruebas Unitarias

EXPRESS JS

Bienvenida e introducción

Introducción y bienvenida

¿Qué es express y para qué sirve?

Creando tu primer servidor

Express application generator

Template Engines y archivos estáticos

Qué es y cómo crear tu propio template engine

Implementando un template engine

Usando Pug como template engine

Usando Handlebars como template engine

Creando nuestro layout de productos usando Pug

Manejo de archivos estáticos

Creando un API Restful

Anatomía de una API RESTful e implementación CRUD

Implementando nuestra capa de Servicios

Resumiendo el proyecto

Request object

Conectando Express.js con servicios externos

Conexión a MongoDB en Express.js

Conectando nuestros servicios con mongo DB

Implementando acciones CRUD en MongoDB

Configurar una cuenta en Amazon AWS

Middlewares: En el medio del request

Qué es un middleware y qué tipos existen

Manejo de errores usando un middleware

Validación de datos usando un middleware

Implementando Sentry para el manejo de log de errores

Qué es Joi y Boom y cómo configurar Joi

Configurando Boom y nuestra página 404

Middlewares populares

Autenticación en Express.js

JSON Web Tokens

Configuración y uso de Passport.js

Asegurando nuestra API e identificando rutas sensibles

Testing

Creando tests para nuestros endpoints

Creando tests para nuestros servicios

Creando tests para nuestras utilidades

Agregando coverage para ver la calidad de nuestras pruebas

Debugging e inspect

Deployment

Buenas prácticas para el despliegue en producción

npm script

Añadiendo manejo de cache y seguridad con helmet

Cómo usar las variables de entorno para diferentes ambientes

Habilitando CORS en producción

Cómo funciona y por qué es importante el uso de HTTPS

Desplegando con Now y detectando vulnerabilidades con npm audit

Automatizar el chequeo de vulnerabilidades con snyk

Conclusiones

PROGRAMACION FUNCIONAL EN JAVASRRIPT

Introducción al curso y conceptos base

¿Qué es la programación funcional?

Introducción al curso

Boilerplate: Base para el proyecto del curso

Funciones

Funciones Algebraicas y Funciones de JavaScript

Funciones puras

Proyecto: Obtener el valor de los inputs

Objetos y Tipos de Memoria en JavaScript

Copiar y modificar objetos en JavaScript

Utilizando inmutabilidad en nuestras funciones

Proyecto: Validar inputs

Estado compartido en funciones

Estado compartido o shared state

Proyecto: Agregar elementos a la lista

Composición de funciones, Closures y Currying

Funciones compuestas o Function Composition

Completando las funciones para generar etiquetas HTML

Closures en programación funcional

Currying

Proyecto: Construyendo filas y celdas

Higher Order Functions

Introducción a las Higher Order Functions

Proyecto: Actualizando el total de calorías

Proyecto: Mostrar elementos

Proyecto: Eliminar elementos

Bonus: Declarative Programming

Programación Declarativa

Conclusiones

Conclusiones

JAVASCRIPT FULLSTACK CON SAILS.JS

Bienvenidos al nuevo curso

Nuevo Curso Definitivo de JavaScript

Setting up

Introducción y generalidades del curso

Las tecnologías que vamos a usar

Lo que vamos a construir

Evitando errores en tu código

Usando la terminal en este proyecto

Usando GitHub en nuestro proyecto

Instalando nuestras plataformas y dependencias

Los esenciales

Arquitectura Backend

Arquitectura Frontend

Parasails.js

Autenticación y login

Inscripción de usuarios

Sesiones y la base de datos
Recuperación de passwords
Verificación de emails
Formato de contacto
Manejando información de pagos
Creando una nueva página
Haciendo nuestra página interactiva
Creando un nuevo modelo
Testeo automático (Files)***
Creando una nueva acción Cloud
Usando Cloud SDK

Creando funcionalidades a medida

Personalización
Permisos
Amigos con asociaciones
Queries avanzadas y .populate
Modales
Ajax forms
Loading and error states
Putting it all together
Subiendo archivos backend
Subiendo archivos frontend
Descargando archivos
Tu turno

Growth and monetization

Construyendo un paywall
Suscripciones con pago recurrente
Notificaciones
Invitaciones (Usuarios existentes)
Invitaciones (Nuevos usuarios)
Unclaimed accounts (Cuentas sin contraseñas)

Going Live

Variables de entorno y configuraciones de sailsjs y nodejs
Asegurando nuestra aplicación
Cloudflare y certificados SSL en SAILS
Utilizando una Plataforma como Servicio
Haciendo deploy a Heroku
Configuración para enviar a producción
Agregando variables de entorno del sistema en Heroku
Configuración de la base de datos
Temas a considerar antes de lanzar tu proyecto
Despedida del curso

Contenido Avanzado

JavaScript fullstack: diseñando el frontend de tu app con
Bootstrap 4
Server-side rendering con EJS
Cómo funcionan los datastores
APIs que vas a usar si estás construyendo apps nativas

END TO END TESTING CON CYPRESS

Introducción

¿Qué es End-to-End Testing?
¿Qué es Cypress?

Primeros pasos

Configurando el proyecto inicial
Integrando Cypress al proyecto
La consola de Cypress

Mi primer test
Implementando tests
Principales comandos
Interactuando con elementos
Aserciones
Hooks
Variables, Fixtures y Alias
Conceptos avanzados
Ejecución de scripts
Comandos personalizados
Variables de entorno y configuración
Screenshots
Stubs, Spies y Clocks
Depuración de los tests
Headless testing
Bonus
Testing condicional
Plugins
Intellisense
Seguridad
Usando el dashboard de Cypress
Conclusiones

PHP CON LARAVEL

Introducción

Intro y características de Laravel

Fundamentos

Instalación de Laravel
Primer ruta en laravel
Cómo funciona Blade
Controladores en Laravel
Request
Configuración de laravel
Bases de Datos
Cómo funcionan las Migraciones de DB
Migraciones en Artisan
Modelos con Eloquent
Trabajando con un Modelo

Operaciones CRUD

Controladores y recursos
Blade layout
Form para agregar reportes
CSRF
Fake PUT/PATCH
Borrando reportes
Validaciones

Relaciones en bases de datos

Creamos la vista de reportes
Relaciones con eloquent
Trabajando con relaciones

Emails

Acción para enviar un mail
Enviando emails

Autenticación

Login
Cierre

Cierre

BUENAS PRACTICAS PARA LA ESCRITURA

Temario del curso

- ¿A quién beneficia contar con código bien escrito?
- Ejes que hacen a la calidad del código
- Código prolijo, modular y reutilizable
- Identificadores mnemotécnicos, específicos y precisos
- Evitar el hardcoding
- Evitar efectos colaterales
- Principios SOLID
- Patrones de diseño: Singleton, Factory y Command
- Introducción al Testing Automatizado
- Fundamentos de Test Driven Development
- Pull requests en entornos de colaboración
- Documentación
- Análisis de un proyecto escrito con buenas prácticas

INGENIERIA DE DATOS CON PYTHON

Introducción

Archivos del curso y Slides
Introducción al curso
¿Qué es la Ciencia e Ingeniería de Datos?
Roles
Configuración del ambiente
Jupyter Notebooks
Tipos de datos
Fuentes de datos
ETL
Web scraping
Introducción a las tecnologías web
Realizar solicitudes HTTP con Python
¿Cómo trabajar con un documento HTML?
Analizando un sitio web para encontrar las directivas a utilizar al hacer un web scrapping
Solicitudes a la web: Requests
Implementando nuestro web scrapper: Configuración
Implementando nuestro web scrapper: Obteniendo enlaces del front page
Implementando nuestro web scrapper: Obteniendo artículos
Obtención de datos del Artículo
Persistiendo la información scrapeada
Pandas
Introducción a Pandas
Estructura de datos: Series
Estructura de datos: DataFrames
Índices y selección
Data wrangling con Pandas
Creación de la receta: preparando la automatización
¿Cómo trabajar con datos faltantes?
Operaciones con Series y DataFrames
Limpiando detalles adicionales
Limpiando detalles en nuestro proyecto
Enriquecimiento de los datos
Valores duplicados en Jupyter
Valores duplicados en Python

Vizualización de datos

Intro a Sistemas de Datos

Introducción a los sistemas de datos
Cargando datos a SQLite
Automatización del Pipeline
Contenido Bonus
¿Por qué usar la nube?

DJANGO

Introducción

Introducción al curso
Historia de web development
Preparación del entorno de trabajo en Mac
Preparación del entorno de trabajo en Windows y Linux
Creación del proyecto Platzigram / Tu primer Hola, mundo! en Django

Vistas

El objeto Request
Solución al reto - Pasando argumentos en la URL
Creación de la primera app
Introducción al template system
Patrones de diseño y Django

Models

La M en el MTV
El ORM de Django
Glosario
Extendiendo el modelo de usuario
Implementación del modelo de usuarios de Platzigram
Explorando el dashboard de administración
Dashboard de Administración
Creación del modelo de posts

Templates, auth y middlewares

Templates y archivos estáticos
Login
Logout
Signup
Middlewares

Forms

Formularios en Django
Mostrando el form en el template
Model forms
Validación de formularios

Class-based views

Class-based views
Protegiendo la vista de perfil, Detail View y List View
CreateView, FormView y UpdateView
Generic auth views

Deployment

Arquitectura / Conceptos / Componentes
¿Cómo conectar Django a una base de datos?
Configurar el servidor
Preparación del VPS (en AWS)
Conclusiones del curso

Bonus

¿Cómo usar los templates en Django?

AVANZADO DE DJANGO

Integración y despliegue en la nube

Estructura de mensajes en Celery y resultados de tareas

Tareas periódicas con Celery

Configuración de sistemas en la nube

Contenedores

Despliegue en un PaaS: Heroku

Uso de Celery con Node.js

Reto: despliegue en la nube (usando cuentas gratuitas)

Conclusiones

Despedida, conclusiones y a dónde ir desde aquí

FLASK

Fundamentos de Flask

Introducción

¿Cómo funcionan las aplicaciones web?

¿Qué es Flask?

Instalación de Python, pip y virtualenv

Hello World Flask

Debugging en Flask

Request y Response

Ciclos de Request y Response

Uso de templates y archivos estáticos

Templates con Jinja 2

Estructuras de control

Herencia de templates

Include y links

Uso de archivos estáticos: imágenes

Configurar páginas de error

Extensiones de Flask

Flask Bootstrap

Configuración de Flask

Implementación de Flask-Bootstrap y Flask-WTF

Uso de método POST en Flask-WTF

Desplegar Flashes (mensajes emergentes)

Pruebas básicas con Flask-testing

Proyecto

Planteamiento del proyecto: To Do List

App Factory

Uso de Blueprints

Blueprints II

Base de datos y App Engine con Flask

Configuración de Google Cloud SDK

Configuración de proyecto en Google Cloud Platform

Implementación de Firestore

Autenticación de usuarios: Login

Autenticación de usuarios: Logout

Signup

Agregar tareas

Eliminar tareas

Editar tareas

Deploy a producción con App Engine

Conclusiones

BASICO DE NODE JS

Introducción al curso

Antes de Empezar el Curso

Presentación del curso

Aspectos generales de Node.js

Introducción a Node.js

Historia de Node.js

Preparación del entorno de trabajo

Creación de máquina virtual de trabajo

Instalación de recursos de proyecto en MacOSX

Instalación de recursos de proyecto

Clonación de repositorio del curso

Gestión de proyectos con Node y NPM

Introducción a Node.js: CLI y REPL

Creación de proyectos de Node.js con NPM

Módulos en Node.js: require() y module.exports

Manejando modulos externos con NPM y require()

Asincronismo en Node.js

Fundamentos de asincronismo con Node.js

Callbacks y Promesas: Casos de estudio

Hacia el futuro: asincronismo con Async/Await

Módulos nativos básicos de Node.js

Node.js FileSystem: Lectura/Escritura de archivos en Node.js

Node.js Readline: Introducción de datos por terminal

Node.js Child Process: Ejecución de subprocessos con Node.js

Node.js Debugger: Depuración de código en Node.js

Node.js Errors: Manejo de errores en Node.js

Node.js Events: Emisión y recepción de eventos en Node.js

Servicios web en Node.js con Express.js

Introducción al protocolo HTTP

Creación de un servidor web básico con HTTP

Creación de un servidor web con Express.js

Construcción de un API con Express.js

Servicios de datos en Node.js con MongoDB

MongoDB: Conexión y escritura de datos con Node.js

Completando nuestro API con MongoDB

Aplicaciones Real-Time en Node.js con Socket.io

Bienvenido a la sección del Proyecto del Curso

Introducción a Socket.io y proyecto de curso

Presentación del proyecto final: Tic-tac-toe

Bases y fundamentos del proyecto final: Tic-tac-toe

Desarrollo del Proyecto final del curso: Tic-tac-toe

Finalizando el proyecto final del curso: Tic-tac-toe

Conclusiones

Consideraciones finales

ESENCIALES DE NODE JS

Sobre el curso

Antes de Empezar el Curso

Introducción y presentación del curso

Preparación de aplicaciones web con Node.js

Herramientas y Repositorios

Historia de Node.js

Configuración asistida de rutas en Express.js

Configuración de middlewares y extras en Express.js

Consumiendo operaciones de un API público

Creación de un API con Express.js

Conexión a una base de datos MongoDB con Node.js

Poblando MongoDB con nuestro API de Express.js

Construcción de peticiones de datos para MongoDB

Eliminando datos de MongoDB desde Express.js

Pruebas de código con Mocha.js

Estándares de documentación de código

Documentando código con JsDocs

Seguridad de código en Node.js

Analizando código mediante ESLint con Visual Studio Code

Liberación de proyectos de Node.js con NPM

Preparando nuestro proyecto para publicación

Publicando nuestro proyecto en NPM

Manejando nuestra versión de proyecto

Administración de tareas de proyecto en Node.js

Gestionando tareas con Grunt.js

Medición del rendimiento de tu aplicación

Creación de perfil de rendimiento de funciones con Node.js

Creación de un perfil de rendimiento de servicios web con

Profiling

Prácticas de alto rendimiento en Node.js

Aprovechando recursos en Node.js con Cluster

Consideraciones finales

NODE JS CON HAPI

Introducción

Bienvenida al curso y prerequisites

Breve historia y estado actual

Conceptos principales de hapi y creación de nuestro Primer servidor

Creando un sitio básico con Hapi

El objeto h, response y sus herramientas

Uso de plugins - Contenido estático

Plantillas con Handlebars

Renderizado de vistas - Layout y template del home

Recibiendo parámetros en una ruta POST - Creación del registro

Definir una mejor estructura con buenas prácticas en Hapi

Validando la información - Implementando Joi

Introducción a Firebase

Creando un modelo y guardando en firebase

Implementando el login y validación del usuario

Autenticación de usuarios - Cookies y estado

Manejando errores

Visualización de errores

Controlar el error 404 en inert y el error de validación

Repaso - Creación del modelo y controlador para preguntas

Repaso - Creación de las rutas para crear preguntas

Listar las últimas preguntas en el home

Aplicacion de conceptos avanzados

Enrutamiento avanzado - visualizando una pregunta

Enrutamiento avanzado - respondiendo una pregunta

Generando la lógica de la plantilla según si es creador o contribuidor

Métodos de servidor - respuesta correcta

Usando métodos de servidor

Manejo del caché - Agregando el home al caché

Procesamiento de archivos - Aceptando imágenes

Logging con Good - Monitoreando el servidor

Creación de plugins - Teoría

Creación de plugins - Implementando un API REST

Estrategías de autenticación - Asegurando el API REST

Seguridad básica - Asegurando el servidor contra CSRF

Seguridad básica - Asegurando el servidor contra XSS

Herramientas de desarrollo

Depuración del proyecto

Ecosistema de Hapi

AVANZADO DE NODEJS

Inicio del curso

¿Qué vamos a construir en el curso avanzado de Node.js?

Introducción a Node.js

¿Qué es Node.js?

Características de la plataforma Node.js

¿Qué tipo de aplicaciones podemos desarrollar con Node.js?

Preparando el entorno de Desarrollo

Cómo instalar Node.js en macOS

Cómo instalar PostgreSQL y Redis en macOS

Cómo instalar Visual Studio Code en macOS

Cómo instalar Ansible en macOS

Arquitectura del Proyecto (Platziverse)

Arquitectura y Componentes del proyecto

Introducción a protocolos y patrones de aplicaciones en tiempo real

Cómo funciona el modelo Pub/Sub en MQTT y Web Sockets

Creando Módulo de Base de Datos (platziverse-db)

Estructura básica de un módulo de Node.js

Definición de entidades de base de datos

Implementación de modelos con sequelize

Implementación de un módulo básico de base de datos

Implementación de script de inicialización de base de datos

Creando una advertencia sobre el borrado de base de datos

Introducción a pruebas unitarias con Ava.js

Introducción a code coverage con nyc

Cómo hacer Mocks y Stubs con Sinon

Creación de fixtures y definición básica del servicio de Agent

Implementación de findById y pruebas en el servicio Agent

Implementación de createOrUpdate

Revisión del servicio Agent

Implementación del servicio Metric

Realizando un ejemplo con el módulo de base de datos

Reto: Cómo modificar el script de inicialización de la base de datos

Construyendo un servidor en tiempo real para Internet de las Cosas con Mosca/MQT

Definición de un Broker de Mensajería

Definición de los tipos de mensajes

Implementación del servidor MQTT

Cómo recibir mensajes

Cómo integrar el servidor MQTT con la base de datos

Cómo almacenar la información del agente y reportar el agente conectado

Cómo almacenar la información de las métricas

Probando el servidor MQTT (Ejemplo con mqtt client)

Construyendo el agente de monitoreo (platziverse-agent)

Cómo implementar un custom EventEmitter usando ES6 classes

Definiendo un timer de ejecución continua (setInterval)

Implementación del cliente MQTT

Implementación de métricas personalizadas

Probar agent (ejemplo)

Construyendo una API REST con Express (platziverse-api)

Implementación de un servidor básico con express

Definición de rutas del API

Implementación de manejo de errores con express

Pruebas básicas de Integración con Ava y supertest

Integración con el módulo de base de datos

Implementación de ruta del API

Pruebas de Integración con Ava, Supertest y Sinon

Asegurando nuestra API REST con JWT

Definición de JWT

Asegurando nuestra API con express-jwt

Modificando las pruebas de integracion

Implementando permisos con express-jwt-permissions

Creando un Dashboard Web en tiempo real con WebSockets (platziverse-web)

Implementación de un servidor web estático con express

Integrando socket.io con express

Integrando agente de monitoreo con socket.io

Implementación de rutas proxy con la API

Presentación de nuestro cliente frontend en Vue.js

Integración del cliente frontend con API (Metric)

Integración del cliente frontend con socket.io (Metric)

Integración del cliente frontend con API (Agent)

Integración del cliente frontend con socket.io (Agent)

Creando un Dashboard para la terminal en tiempo real con Blessed (platziverse-cl)

Implementación de un CLI Básico con Node

Introducción a Blessed / Blessed Contrib

Implementación del layout básico con Blessed Contrib

Integración con el agente de monitoreo 1

Integración con el agente de monitoreo 2

Integración con el agente de monitoreo 3

Depurando Aplicaciones Node.js

Utilizando longjohn para mejores stacktraces

Depurando en Desarrollo con node --inspect

Depurando en Desarrollo con Visual Studio Code

Preparando nuestra aplicación para producción

Preparando Proyecto para Producción 1

Preparando Proyecto para Producción 2

Creación de scripts para despliegue con Ansible

Probando el database rol con Vagrant

Creando scripts del rol de platziverse db y mqtt (script de automatización .yaml)

Terminando los scripts faltantes

Creación de servidores en DigitalOcean

Desplegando nuestra aplicación a producción

Ejecutando los Scripts de Despliegue

Utilizando Aplicación en Producción y cierre del curso

Bonus: Utilizando platziverse-agent en BeagleBone Black y Raspberry PI

Implementación platziverse-agent con Johnny-Five

RUBY

Introducción

Bienvenida e introducción

¿Qué es Ruby? Ventajas, desventajas y ¿quiénes lo están usando?

Instalación de Ruby

Bases del lenguaje

Tipos de datos I: Integer, float

Tipos de datos II: Strings

Tipos de datos III: Symbols

Tipos de datos IV: Arrays

Tipos de datos V: Hashes

Uso de condicionales

Uso de ciclos

Rangos

Uso de Regex

Procs y lambdas

Programación Orientada a Objetos en Ruby Comenzando la creación de nuestra clase

Programación Orientada a Objetos en Ruby: Módulos

Programación Orientada a Objetos: Clases y Objetos

Concurrencia vs Paralelismo: Threads en Ruby

Bundler y gemas

Testing en Ruby

Testing con MiniTest

Diferencias entre Ruby 2.5 y 3

Proyecto

Introducción al proyecto

Instalar Ruby 2D

Overview Ruby 2D

Arquitectura

Estructurando el proyecto

Vista basica

Programando la base para los movimientos

Programando el movimiento de nuestro snake

Test de nuestras acciones

Implementando el timer del movimiento

Actualizando la vista

Preparando el cambio de dirección

Recapitulación de la arquitectura

Moviendo la serpiente con el teclado

Creciendo la serpiente y programando la condición de derrota

Generando comida aleatoria: uso de rand y stub en pruebas

Condición de salida y conclusiones

Retos del curso

CREACION DE APIS CON RUBY ON RAILS

Introducción

Bienvenida e introducción

Configuración

Proyecto

Creación del proyecto

Instalación de gemas necesarias

Configurar rspec + factorybot

Hello world (health endpoint)

Casos de uso para nuestra aplicación

Planeando nuestro modelo entidad relación

Crear modelos y validaciones + tests

Implementación de los modelos usando TDD

Listar y mostrar Post: pruebas

Listar y mostrar Post: implementación

Crear y editar POST: pruebas

Crear y editar POST: implementación

Usando ActiveRecordSerializer

Filtrar posts - pruebas

Filtrar posts - implementacion
N+1 query problem: explicacion teórica
N+1 query problem: solución
Introducción a la autenticación con Tokens
Autenticación: pruebas del endpoint de detalle (/posts/{id})
Autenticación: generacio de auth token para el usuario
Autenticación: agregando autenticacion al controlador de Post
Autenticación: lógica de update, delete y create del controlador de blogposts
Autenticación: Finalizando pruebas de creación y actualización de blogposts
Probando el API con Postman
Explicación de Caching
Usando Cache para acelerar las búsquedas de posts
Background jobs y ActiveJob (explicación teórica)
Generar Reporte para procesar con ActiveJob
Enviar Reporte usando ActionMailer
Autenticacion con auth0 y JWT
Cierre
Cierre del curso

DIGITALOCEAN

Introducción a DigitalOcean
Introducción al Curso de DigitalOcean
Registro en DigitalOcean
Servicios de DigitalOcean: Droplets
Servicios de DigitalOcean: Block Storage
Servicios de DigitalOcean: Spaces
Panel de control y droplets
Panel de control de DigitalOcean
Creación de Droplet
Panel de control de un droplet: Gráficas
Panel de control de un droplet: Accesos
Panel de control de un droplet: Power
Panel de control de un droplet: Volúmenes
Panel de control de un droplet: Configurar Volúmenes
Panel de control de un droplet: Escalar un droplet
Panel de control de un droplet: Networking y Firewalls
Panel de control de un droplet: Backups y Snapshots
Panel de control de un droplet: Kernel
Panel de control de un droplet: History, Destroy y Tags
Deploy de Magento en DigitalOcean
Instalación de Nginx
Instalación de MySQL
Instalación de fpm-php
Instalación de Magento
Continuando la instalación de Magento
Conclusiones del curso

GO

Introducción
Introducción a Go
Instalación de Go: Instalación de Go en Ubuntu
Instalación de Go: Instalación de Go en Mac
Instalación de Go: Instalación de Go en Windows
Primeros pasos

Organizando el entorno de trabajo
Hola mundo
Recibir valores del usuario
Declaración de variables y constantes
Funciones
Tipos de datos numéricos en Go: enteros, flotantes
Cadenas de texto básico
Array y Slice
Condicionales
For, break

Intermedio

Operaciones con cadenas
Uso de la instrucción: Switch
Manejo de Paquetes
Map
Structs
Methods
Interfaces
defer
Errores

Avanzados

Punteros
Goroutines
Canales
Ejemplos de Canales
Bibliotecas de Go
Consumiendo una API externa
Errores comunes en Go
Cierre del curso
Contenido Bonus
Cambiar el string
Struct para calcular
Leer un archivo del disco

BASICO DE VUE JS

Fundamentos de Vue.js

Introducción a Vue.js
Hola mundo con Vue.js
Manejo de condiciones con Vue.js
Filtro de datos en Vue.js
Control de eventos con Vue.js
Manejo de formularios con Vue.js
Listar elementos con Vue.js
Cómo crear componentes con Vue.js
Creando un Single Page App con Vue.js
PlatziMusic con Vue.js
Incorporando Stylus y Pug
Configurando la API de Lastfm en Vue.js
Estructurando componentes en Vue.js
Actualizando datos en la interfaz con Vue.js
Agregando indicador de progreso a la aplicación
Enviando una aplicación con Vue.js a producción
Se acaba este curso ¿quieres seguir aprendiendo Vue.js?

PROFESIONAL DE VUE JS

Introducción
Bienvenidos al Curso Profesional de VueJS

Introducción a VueJS

CLI y Dev Tools

Herramientas y Experiencia de Desarrollo + Archivos .vue

CLI - Hello World

Webpack

Babel

Eslint

SASS y BULMA

PUG

Ejercicio de SASS y PUG

Ejercicio Avanzado de Pug y SASS

Manipulación del DOM

Expresiones

Directivas

Data Binding

Computed properties

Watchers

Eventos

Integración a Platzi Music

Ejercicio de Manipulación del DOM

REST y HTTP

Servicios

Fetch API & Trae

Consumir API's REST

Sistema de Componentes

Component

Creación de componentes

Reactividad

Ciclo de vida

Comunicación entre componentes padres e hijos

Comunicación de hijos hacia padres

Utilización de Slots

Comunicacion entre Componentes Genericos - Event Bus y Plugins

Vue Router

Introducción a Vue Router

Instalar vue-router y configurar router view

Crear y Navegar Rutas con router-link

Extendiendo VueJS

Modifiers

Filtros

Directivas Personalizadas

Mixins

Clases, Animaciones y Transiciones36

Vue Transitions y Animaciones de CSS

Vuex

Estado Centralizado, Flux y Vuex

State

Mutations

Getters

Actions

Integración a Platzi Music

Nuxt.js

Server Side Rendering

Nuxt.js

Conceptos Básicos de Nuxt

Deploy a Producción con Now

Qué es now

Configuración de now y deploy

Conclusiones

Cierre del curso

Bonus

Internacionalización con vue-i18n

Unit Test Karma - Mocha & Webpack

Implementación de Autenticación de Usuarios en Vue usando JWT

AVANZADO DE VUE JS Y FIREBASE POR BEDU

Bienvenida e Introducción

Bienvenida y Presentación del proyecto: PlatziRooms

Creación y configuración inicial del proyecto utilizando VUE CLI 3

Boilerplate y Limpieza del proyecto

Render Functions y JSX

¿Que son las Render Functions y JSX?

Preparando nuestro entorno para un prototipo

Creación de un componente con createElement

Utilizando JSX para la creación de un componente

Utilizando Slots con Render Functions y JSX

Creando un Modal de Login para PlatziRooms

Creando la lógica general de nuestros modales con Vuex

Creando el contenido de nuestro modal

Componentes controlados y uso de librerías externas

Componentes Controlados y Variables Personalizadas

Construcción del componente de Recordar Contraseña

Creación de un componente Slider utilizando la librería tiny-slider

High Order Functions

¿Qué son las High Order Functions? - Crea tu primera HOF

Vuex

Obtener datos usando Vuex

Crea un getter dinámico en Vuex utilizando HOF

Creación de la vista para creación de publicaciones

Agregando datos con Vuex

Base de datos en tiempo real con Firebase

Instalación y Configuración de Firebase Realtime Database

Obteniendo los datos desde Firebase Realtime Database

Agregando la consulta de usuarios

Almacenando nuevas publicaciones en Firebase Realtime Database

Perfeccionando el flujo de navegación de nuestra App

Autenticación con Firebase

Configuración Inicial para trabajar con Firebase Authentication

Agregando nuevos usuarios en firebase authentication

Inicio de sesión de usuario

Cierre de sesión de usuario

Protegiendo páginas utilizando Navigation Guards

Scoped Slots

¿Qué son los Scoped Slots? Dónde utilizarlos y por qué

Integración de Scoped Slots en Platzi Rooms

Deploy

Creando y desplegando nuestra app en Heroku

Conclusiones y Despedida

KOTLIN

Bienvenida e introducción

¿Por qué debo aprender Kotlin?
Mi primer programa en Kotlin
Conceptos básicos
Variables vs. Objetos
Operadores en Kotlin
Otros Operadores en Kotlin
Tipos de variables: var, val y const
Ejercicio var val const
Programación Funcional: Funciones Puras e Inmutabilidad
Strings
Conversión de tipos de datos
Rangos
If y when en Kotlin
Bucles while y do while
Ciclos for y foreach en Kotlin
Break, Continue y Labels
Valores Nulos y Double Bang
Operador Elvis
Ejercicios con Null Safety y operador Elvis
Reto 1
Conceptos avanzados
¿Qué es un Array en Kotlin?
Métodos útiles en arreglos
Expresiones vs. Valores
Funciones en Kotlin
Funciones con parámetros por defecto
Lambdas en Kotlin
Ejemplo de Lambdas
Reto 2
Programación Orientada a Objetos
Clases
Ejercicio de clases
Modificadores de acceso
Getters y Setters
El Get() y Set() propio de Kotlin
Data class en Kotlin
Método Constructor
Herencia y polimorfismo
Aplicando herencia y polimorfismo
Clases abstractas
Interfaces
Reto 3
Programación Funcional
Funciones de orden superior
Una función como parámetro
Recursividad. List
Recursividad. Filter
Reto 4
Conclusiones
Bonus
Java Vs Kotlin

SWIFT

Bienvenida e Introducción
Introducción a Swift
Apple Swift Playgrounds

Presentación del proyecto
Variables, Constantes y Tipos de Datos
Variables y Constantes
Tipos de datos básicos
Manejo y manipulación de Strings
Manejo y manipulación de Números
Uso de valores booleanos
Tipos de datos Complejos
Arrays
Matrices
Diccionarios
Estructuras de control
If, else, else if, Inline If
Switch
For y While
Opcionales, Guard, If guard
For Where
Manejo de Arrays
Funciones
Definición
Parámetros en las funciones
Tuplas
@discardableResult
Objetos
Estructuras vs Clases
Funciones
Propiedades
Manejo de propiedades
Herencia
Enums
Enums Básicos
Enums representativos
Enums con parámetros
Protocolos
Definición
Usando protocolos
Delegados
Advanced Topics
Extensions
Closures
Typealiases
ARC, Retain cycles, Strong, Weak y unowned
Caso práctico de ARC y Retain cycles
Caso práctico de Retain Cycle con Closures
Excepciones
Generics
Completar el proyecto
Organizar nuestro proyecto en clases independientes
Agregar transacciones repetibles
Calcular Lifetime Value
Conclusiones
Finalización del curso
Reto final
Bonus
Primer proyecto en iOS

DESARROLLO DE INTERFACES CON iOS

Introducción

Bienvenida al curso
Instalacion de Xcode desde 0

Introducción a Xcode

Primer Proyecto
Navigation Area
Editor, Util Area y Tool Bar
Targets y Target Tabs

Formas de crear vistas

Organizacion de Carpetas
Storyboards vs Xibs vs Codigo
Splash Screen
Completando nuestro Splash Screen

UIView Components

TabBar Controller y Mejores Prácticas
Storyboard Reference
Explicación breve de algunos elementos gráficos
Crear nuestra primera interfaz gráfica
Diferentes componentes y formas de autolayout
Agregando el resto de nuestros componentes usando constraints

Agregando Estilo a UIView Components

Agregando un icono a nuestra app
Custom colors
Agregando iconos a nuestros botones
Key paths de redondeo y sombra
Custom fonts
Scroll view controller

Tipos de vistas y manejo

OnBoarding View
Container ViewController
Container ViewController con código
ContainerViews

Adaptación de vistas

Tablas
Diseño de celdas

Completando nuestras vistas

Código para las tablas
Tamaños dinamicos para las celdas
EmptyState usando código
Agregar estilos a nuestro navigation view
Agregando estilos a nuestro Tab bar
Custom Tab bar button
Budget view
Iniciando el header de animacion
Animación de barra superior en las tablas usando constraints

Stats

Conclusiones

Reto final
Conclusiones

Bonus: Lottie

Gestión de dependencias
Instalación dependencias con Cocoapods
Instalación dependencias con Carthage
Cocoapods vs Carthage
Crear nuestro lottie file

Introducción

Integración del framework
Introducción y presentación del proyecto
Cómo hacer una app en ios
Instalación de dependencias
Mezclar Carthage y Cocoapods en un solo lugar

Autenticación

NSUserDefaults
Configuración de Firebase
Validación de campos
Auth con Email
Introducción a Model View, View Model
Autenticación con Facebook
Usando Facebook en nuestro proyecto
Autenticación con Twitter
Autenticación con AccountKit y Firebase
Cloud functions
Reto: Login con gmail
Consumir Api con NSURLSession Nativo
Initial view controller programático

Database

Introducción a las bases de datos Mobile
Modelos a JSON
JSON a Modelos
Query a Firestore y almacenamiento local
Llenado de tablas usando la data
Agregar transacciones
Moviendo data a MVVM
Realtime listening
Eliminar transacciones
Asegurando nuestra base de datos

Firebase

Crashlytics
Local Notifications
Push Notifications
In App Messaging
DynamicLinks
User properties

In app purchase

Introducción y uso de in app purchase
Crear nuestra suscripción con SwiftyStoreKit

App Release

Certificates
Push Notification Certificates
Archivando
Submitting App
Review Process
Beta testers

Conclusiones

Conclusiones

REACT NATIVE

Anuncios

Actualizamos este curso y aquí te mostramos el nuevo proyecto
Antes de comenzar
Introducción al curso
Bienvenida al curso

El poder de React Native

Bienvenido a Platzi Video

Instalando el entorno

Proyecto nuevo en React Native y creación del emulador en AS

Componentes y APIs de React Native

Componentes en React Native

Estilos en React Native

Organización del proyecto

Soporte a iPhone X en React Native: Creación del Header

Listas en React Native

Estructura de recomendaciones

Separadores y listas vacías

Componente de sugerencia

Haciendo peticiones a un servidor | fetch

Integrando datos traídos del API al proyecto

Lista de Categorías

Separadores Horizontales

Componente de Categoría

Instalando plugins de la comunidad

Creando un Reproductor de Video

Controlando el buffer del video

Controles del reproductor

Crear Un componente del reproductor

React Native + Redux

React Native + Redux

Crear los reducers

Persistiendo datos

Creando un componente de carga

Organizando componentes

Validando pantallas

Navegando entre pantallas

Agregando la descripción a la película

Scroll view y web view

Componente de buscador

Haciendo peticiones personalizadas al API

Bonus: Animaciones

Una aplicación lista para producción

Creando un apk para producción en Android

Exportando la aplicación para producción en iOS

Conclusiones del curso

REACT NAVIGATION

Introducción a React Navigation

Bienvenidos el curso de React Navigation

Proyecto del curso

Hola mundo con react-navigation

Creando múltiples pantallas

Navegando entre pantallas

React Navigation API

Configurando un Stack Navigator

Navigation Options

Configuraciones visuales al Stack Navigator

Pasando parámetros entre pantallas

Pasando navigation prop a cualquier componente

Combinando navigators - modal

Stack Actions

Navigator Actions

Integrando Platzi Video con React Navigation

Configurando react navigation con Redux

Añadiendo react navigation a Platzi Video

Actualizando la integración con Redux

Configurando la navegación de pantallas

Creando una nueva pantalla para categorías

Tab Navigator

Personalizando Tab Navigator

Personalizando el Status Bar

Switch Navigator

Login con Switch Navigator

Añadiendo la pantalla Movie a un modal

Añadiendo un Drawer Navigator

Personalizando un Drawer Component

Back button en Android

Conclusiones del curso

Temario del curso

Bienvenida e Introducción

Introducción y Bienvenida

Dart vs. Go vs. JavaScript vs. Kotlin

DartPad y qué editores utilizar

Básicos 1: Variables y Operadores

Variables: Números (int,double), String y booleanos

Colecciones

Diccionarios

Constantes (final y const)

Operadores aritméticos, de asignacion y lógicos

Básicos 2: Funciones y Parámetros

Declaración de funciones y la función Main()

Funciones de sintaxis corta (Arrow) y Anónimas (Lambda)

Parámetros requeridos, opcionales, posicionados, nombrados y por defecto

Básicos 3: Flujos de Control

If-Else y Expresiones Condicionales

Switch y Case

For

While y do-while

Break y Continue

Clases

Definición de clases y miembros

Constructores por defecto, por parámetro o nombrados

Métodos Getter y Setter y variables privadas

Herencia

Clases Abstractas

Interfaces implícitas

Utilizar Flutter en Dart

Iniciar y estructurar un proyecto en Flutter

Contenedores

Creando un botón personalizado con el Widget Container

Modelos de datos y peticiones http

Trabajando datos http y Widget Future Builder

Ejecutando el mismo código en Android y iOS

Conclusiones

Dart en el futuro

Temario del curso

Bienvenida e Introducción

- Desarrollando en Flutter
- ¿Qué es Flutter?
- Dart y Flutter
- Sintaxis de Dart
- Flutter para desarrolladores Android, iOS y Xamarin.forms
- Flutter para desarrolladores React Native
- ¿Cómo luce una app construída en Flutter?
- Primer reto

Creando mi entorno de desarrollo

- Requerimientos de Hardware y Software
- Instalando Flutter en Android Studio y Visual Studio Code
- Composición de un proyecto en Flutter

Interfaces en Flutter

- Programación Declarativa en Flutter
- Estructura de un programa en Flutter
- Hola Mundo en Flutter
- Widgets básicos
- Widgets con estado y sin estado
- Análisis de Interfaces de Usuario en Flutter
- Definiendo los layouts de nuestra interfaz
- Segundo reto

Widgets sin estado en Flutter

- Flutter Widgets: Container, Text, Icon, Row
- Flutter Widgets: Column
- Recursos en Flutter: Tipografías y Google Fonts
- Widget Image
- Widget Apilando Textos
- Widgets Decorados
- Widget Imagen Decorada
- Widget ListView
- Widget Button, InkWell
- Tercer reto

Widgets con estado en Flutter

- Botones en Flutter
- Clase StatefulWidget: Cómo se compone
- Widget Floating Action Button
- Widgets BottomNavigationBar
- Generando Navegación en BottomNavigationBar
- Personalizando nuestro BottomNavigation Bar a Cupertino iOS
- BottomBar
- Cuarto reto

Fin del Curso

- Conclusiones
- ¡Terminamos!

ELECTRON: APPS DE ESCRITORIO EN WINDOWS

Introducción a Electron

- Bienvenida
- ¿Qué es Electron?
- ¿Cómo funciona Electron?
- Herramientas necesarias

Comenzando con Electron

- Creando nuestro proyecto
- Nuestra primera ventana

Propiedades de una ventana de Electron

Comenzando con nuestro aplicativo

- El frontend y Electron
- Configurando nuestro frontend
- Herramientas de desarrollo para Electron
- Algunos frameworks recomendados
- Construyendo el layout
- Completar elementos del Layout

Agregando interacción a nuestra aplicación

- JavaScript en el frontend
- Selecciona imágenes
- Buscar imágenes
- Añadir filtros a la imagen
- Comunicación entre procesos
- Accesando recursos de la máquina
- Leer archivos de directorios
- Generar listas de imagenes
- Refactor del proyecto
- Usando el diálogo de guardar
- Guardar imágenes en disco
- Usando cajas de mensajes
- Detectando errores

Interactuando con el sistema operativo

- Organizando el código del proceso principal
- Múltiples ventanas y comunicación entre ellas
- Creando assets para una nueva ventana
- Consideraciones para múltiples plataformas
- Manejo de preferencias
- Usuario y contraseña
- Creando un ícono en la bandeja del sistema
- Imprimiendo
- Subir una imagen
- Portapapeles con texto o imágenes
- Emitiendo notificaciones
- Agregando un Menú
- Atajos
- Protocolos personalizados

Generando un ejecutable para nuestra aplicación

- A tener en cuenta para generar ejecutables
- Configurando el entorno para construcción
- Configurando el entorno para construcción 2
- Personalizando nuestro ejecutable

Creando paquetes para distribuir nuestro aplicativo

- Considerando instaladores para múltiples plataformas
- Firmas digitales con Code Signing
- Cierre del curso

Bonus

- Integración continua
- Escribiendo tests con Ava y Spectron

Desafíos

- Crear un filtro personalizado
- Leer las imágenes ubicaciones remotas (Cloudup)
- Previsualización de filtros

TEMAS Y PLUGINS EN WORDPRESS

Introducción al curso

- ¿Qué podemos hacer con WordPress?

¿Debería crear un sitio desde cero o usar un template de WordPress?

¿Por qué aprender WordPress?

Introducción a WordPress y WordPress.org

Conociendo WordPress

WordPress como un blog

WordPress como un CMS

WordPress como un framework

WordPress en el futuro

Repasando WordPress

¿Qué debes hacer antes de comenzar?

Instalar WordPress en un hosting con Softaculous

Instalar WordPress en un servidor con acceso SSH

Instalar WordPress en un servidor con acceso FTP

Instalar WordPress en un servidor con WP-cli

Instalar WordPress en local

Proyecto - Instalación Inicial

Temas en WordPress

Instalar un tema en WordPress

Modificar un tema de forma correcta: La estructura de archivos

Modificar un tema de forma correcta: Cómo crear plantillas específicas

Modificar un tema de forma correcta: Los template parts

Modificar un tema de forma correcta: El archivo functions.php

Utilizar temas hijos y aplicación al proyecto

Gestión de usuarios

Gestión de usuarios básica

Restricción de contenidos

Creación de nuevos roles de usuario - aplicado al proyecto

Maquetadores visuales

Usando Divi como maquetador visual

Aplicar Visual composer en maquetación visual

Usar Gutenberg como maquetador visual

Utilizar WordPress como CMS o como Framework

Custom Post Types - CPT de viajes adaptable a nuestro proyecto

Custom Fields

Formatear tu CPT para mostrarlo en Frontend

Aprender a utilizar la REST API de WordPress

Creación de plugins en WordPress

Plugins básicos que deberíamos utilizar

Jetpack

ManageWP

Plugins de caché

Cookie Law Info

Really Simple SSL

Yoast SEO

Updraft Plus

Cómo crear un plugin

Proyecto - darle diseño y contenido a nuestro sitio

Proyecto - despublicar entradas con Post Expirator y cierre

Contenido Bonus

Seguridad en WordPress

Convierte tu instalación local en un entorno de testing con distintas versiones de PHP

WooCommerce

Instalación de WooCommerce

Configuración básica de WooCommerce

Configuraciones de compra y correos electrónicos en WooCommerce

Crear productos para tienda virtual

Proceso de venta y Pasarelas de Pago

Productos variables

¿Qué es Sensei?

CREACION Y OPTIMIZACION DE SITIOS

Introducción

Introducción: por qué aprender a usar WordPress

WordPress y SEO: primero pasos

Conceptos fundamentales

Cómo hacer un estudio de palabras clave

Reto: estudio de palabras clave

Cómo comprar un dominio a bajo costo

Contratar un hosting

Enlazar dominio y hosting

Instalación del certificado SSL

Instalación y configuración

Instalar WordPress con cPanel

Reto: Instalar WordPress desde un cPanel

Configuraciones iniciales en WordPress

Tema premium vs. gratuito

Cómo instalar y configurar el tema premium BeTheme en WordPress

Instalación de plugins de SEO

Instalación del Demo de acuerdo a nuestra temática

Configuración del Demo

Configurar la versión AMP de la página

SEO y contenidos optimizados

Conceptos básicos de la elaboración de contenidos y la importancia de tener un blog

Aplicando técnicas SEO en los contenidos

Reto: Vamos a hacer contenido de calidad

Entregando la web a buscadores

Configuración del plugin de Yoast

Indexando una página en los buscadores y cómo funciona la consola de búsqueda de Google

Seguridad y respaldos

Instalación y configuración de los plugins de seguridad y respaldos

Configuración de auditorías automatizadas

Reto: Frecuencia de respaldo

Conclusiones

DIGITALOCEAN

Introducción a DigitalOcean

Introducción al Curso de DigitalOcean

Registro en DigitalOcean

Servicios de DigitalOcean: Droplets

Servicios de DigitalOcean: Block Storage

Servicios de DigitalOcean: Spaces

Panel de control y droplets

Panel de control de DigitalOcean

Creación de Droplet

Panel de control de un droplet: Gráficas

Panel de control de un droplet: Accesos

Panel de control de un droplet: Power

- Panel de control de un droplet: Volúmenes
- Panel de control de un droplet: Configurar Volúmenes
- Panel de control de un droplet: Escalar un droplet
- Panel de control de un droplet: Networking y Firewalls
- Panel de control de un droplet: Backups y Snapshots
- Panel de control de un droplet: Kernel
- Panel de control de un droplet: History, Destroy y Tags

Deploy de Magento en DigitalOcean

- Instalación de Nginx
- Instalación de MySQL
- Instalación de fpm-php
- Instalación de Magento
- Continuando la instalación de Magento
- Conclusiones del curso

DOCKER

Introducción

- Bienvenida al curso
- Problemáticas del desarrollo de software profesional
- Qué es Docker: containarization vs virtualization

Primeros pasos

- Instalación de Docker
- Primeros pasos: Hola mundo y Docker Engine
- Contenedores

Aprendiendo a usar los contenedores

- Explorar el estado de docker
- El modo interactivo
- Ciclo de vida de un contenedor
- Exponiendo contenedores al mundo exterior
- Datos en Docker
- Datos con Docker: Volumes

Imágenes

- Conceptos fundamentales de Docker: imágenes
- Construyendo nuestras propias imágenes
- Comprendiendo el sistema de capas

Desarrollando con Docker

- Usando docker para desarrollar aplicaciones
- Reto
- Entendiendo el cache de layers para estructurar correctamente tus imágenes
- Docker networking: colaboración entre contenedores

Docker compose

- Docker-compose: la herramienta todo-en-uno
- Trabajando con docker-compose
- Docker-compose como herramienta de desarrollo

Docker Avanzado

- Conceptos para imágenes productivas
- Manejando docker desde un contenedor

Cierre

- Cierre del curso

BASICO DE JENKINS

Introducción

- Introducción a Automatización

Jenkins Core

- Introducción a Jenkins
- Instalación y Configuración Básica de Jenkins

Manejo Básico de Usuarios

Jobs

- ¿Qué es un Job?
- Configuración de un Job
- ¿Cómo Jenkins interactúa con su máquina local?

Plugins

- Jenkins y su ecosistema de Plugins
- Cadenas de Jobs
- Connectándonos a GitHub

Pipelines

- ¿Qué es un 'Pipeline'?
- ¿Cómo puedo acelerar mi development de Pipelines?

Slave

- Introducción a Slaves
- Conectado un Slave

Cierre

- Cierre del curso, introducción a DevOps

DEVOPS CON GITLAB

Introducción

- Introducción al curso
- ¿Qué es Devops?
- El ciclo de vida del Devops
- Introducción a Gitlab
- Gitlab vs Github

Administración

- Autenticación
- Grupos
- Autorización
- Auditoría
- Proyectos

Planificación

- Tipos de desarrollo
- Planificación en Gitlab-Issues
- Planificación en Gitlab-Etiquetas
- Planificación en Gitlab-Pesos
- Planificación en Gitlab-Milestones
- Planificación en Gitlab-Boards
- Planificación en Gitlab-Service Desk
- Planificación en Gitlab-Quick actions

Verificación

- Inicialización del repositorio
- Merge requests
- Profundizando en Merge requests
- Continuous Integration-CI
- Gitlab CI
- Automatizacion con GitLab CI
- Validacion de la configuracion con GitLab CI
- gitlab-ci.yml
- Gitlab pages
- Implementando Gitlab pages
- ¿Qué es el Desarrollo Ágil?
- Gitlab autodevops
- Implementando GitLab autodevops
- Habilitando autodevops
- Empaquetación**
- Introducción a contenedores

Gitlab container registry

Seguridad

Introducción a DevSecOps

Firmas de seguridad

Pruebas estáticas de seguridad

Escaneo de contenedores

Escaneo de dependencias

Pruebas dinámicas de seguridad

Gitlab security dashboard

Distribución

Continuous Delivery (CD)

Ambientes

Review apps

Estrategias de Distribución

Feature Flags

Rollback

Monitoreo

¿Por qué monitorear?

Métricas de desempeño (performance metrics)

Métricas de salud (health metrics)

Metrics de equipo

Rastreo de errores

Conclusiones

¿Por qué desarrollar con Gitlab?

DEPLOY CON NOWSH

Introducción al Cursos de Deploy con Now.sh

Componentes de una aplicación

Opciones para hacer Deploy

Aplicaciones monolíticas vs microservicios

Instalación de Now.sh

Aplicaciones Estáticas

Deploy con GitHub Pages

Deploy con Surge.sh

Deploy con Now.sh

Tipos de Deploy

Estructura del proyecto

Deploy de aplicaciones Node.js

Deploy de contenedores de Docker

Configuraciones

Configuración de secrets y variables de entorno

Ver el código fuente de tu aplicación

Definir aliases para tus deploy

Usar un dominio personalizado

Componer microservicios

Conclusiones

Conclusión del proyecto

Conclusiones del curso

Contenido Bonus

Deploy en Now.sh desde GitHub

Retos y Desafíos

Combinar muchos servicios

Usar un servicio diferente

ADMINISTRACION DE SERVIDORES LINUX

Introducción a Linux

¿Por qué se usa Linux en servidores?

Configuración y guía básica usando Virtualbox

Iniciar un servidor Linux con AWS

Manejo de sesiones remotas con tmux y SSH

Editores de texto en Linux

Editar archivos con VIM

Instalación y configuración de programas

Añadir y administrar repositorios

Instalar, actualizar y remover programas en Linux

Empaquetar y comprimir archivos en Linux

Compilar un programa en Linux

Dónde encontrar la documentación de los programas

Gestores de paquetes en Linux

Administración de particiones y arranque

Estructura de archivos en Linux

Administrar discos y particiones en Linux

Formateo y montaje de particiones en Linux

Administración swap en Linux

Generar imágenes de discos duros

Administrar particiones GPT

Administrar el arranque del sistema GRUB

Administrar discos duros con LVM

Apagar servidores de forma remota

El sistema y lo que hay que hacer

Arrancando el sistema runlevels, systemd

Uso de variables de entorno, bashrc, profile

Uso de redireccionamiento y pipes para logs

Monitorear, eliminar y establecer prioridades en procesos

Enlaces duros, simbólicos y sistemas de backups

Manejo de usuarios y grupos

Generar backup de Base de datos

Uso de socat y manejo de redireccionamiento de puertos

Tareas programadas y cómo monitorearlas

Administración

Seguridad del kernel

Permisos de archivos

Configuración de red

Cómo manejar mi firewall

Manejo de DNS y dig

Instalación de NTP

Auditoria de login y logs

Tuneles con ssh, autossh y socat

Backup de archivos de configuración con etckeeper

Prevenir ataques con fail2ban

Cierre del curso

Cierre del curso

Desafíos

Desafío 1: Cómo mover una base de datos y agrandar el disco

Desafío 2: Recuperación de un MBR

Contenido Bonus

Encriptación de datos

Encontrar el dueño del paquete

md5sum, integridad de archivos y paquetes

Cómo podemos monitorear servidores

Qué distribución elegir y otras dudas frecuentes

SWARM

Conceptos básicos

Bienvenida al curso
¿Qué es Swarm?
El problema de la escala: qué pasa cuando una computadora sólo no alcanza
Arquitectura de Docker Swarm
Preparando tus aplicaciones para Docker Swarm: los 12 factores
Primeros pasos
Instalación de Docker
Tu primer Docker Swarm
Fundamentos de Docker Swarm: servicios
Entendiendo el ciclo de vida de un servicio
Un playground de docker swarm gratuito: play-with-docker
Docker Swarm multinodo

Administrando Servicios
Administrando servicios en escala
Controlando el despliegue de servicios
Exponiendo aplicaciones al mundo exterior
Swarm avanzado
El Routing Mesh
Restricciones de despliegue
Disponibilidad de nodos
Networking y service discovery
Docker Swarm stacks
Reverse proxy: muchas aplicaciones, un sólo dominio

Swarm productivo
Arquitectura de un swarm productivo
Administración remota de swarm productivo
Consideraciones adicionales para un swarm produtivo
Conclusiones
Conclusiones

AZURE LAAS

Introducción a Azure
¿Qué veremos en el Curso de Azure IaaS?
Características del portal de Azure
Cálculo y monitoreo de costos en Azure
Infraestructura como servicio
Introducción a máquinas virtuales de Azure
Máquinas Virtuales de Azure
Galería de imágenes de Máquinas Virtuales
Configuración básica para crear una Máquina Virtual
Creación de una Máquina Virtual basada en Windows
Creación de una Máquina Virtual basada en Linux
Instalación de software adicional en Windows
Instalación de software adicional en Linux
Introducción a Redes Virtuales en Azure
Introducción al Balance de Cargas
Administración de Máquinas Virtuales
Creando una Máquina Virtual en Azure
Uniendo dos Máquinas Virtuales en Azure
Creando un Balanceador de Cargas en Azure
Funcionamiento del Balanceador de Cargas
Administrar Máquinas Virtuales en Azure
Línea de comandos de Azure
Migración de máquinas virtuales físicas
La tienda de aplicaciones para máquinas virtuales
Conclusiones del curso de Azure IaaS

Virtualización con Dv3 Machines
Virtualización anidada en máquinas virtuales

AZURE PAAS

Introducción al Curso de Azure PaaS
Bienvenida al curso de Azure PaaS
Diferencia entre IaaS y PaaS
Escenarios donde IaaS o PaaS funcionan adecuadamente
Diferencias de costo entre IaaS y PaaS
Características de PaaS que aprenderás en este curso
Servicios de Almacenamiento de Azure (Storage Services)
Servicios de almacenamiento de Azure
Almacenamiento tipo Blob
Azure Storage Explorer
SDK de .NET para almacenamiento Blob
SDK de .NET para almacenamiento Blob 2.0
Almacenamiento tipo File
Crear y administrar un archivo compartido
Configurando una aplicación para File Storage con .NET
Utilizando el SDK de .NET para File Storage
Almacenamiento tipo Table
SDK de .NET para almacenamiento Table
Manejando tablas por medio del SDK de .Net
Agregar entidades a una tabla
Consultar las entidades de una tabla
Actualizar entidades
Eliminar tablas o entidades
Almacenamiento tipo Colas
SDK de .NET para almacenamiento de Colas
Crear una cola e insertar un mensaje
Seleccionar y eliminar mensajes
Manejo de las pilas de mensajes
API REST de los servicios de Almacenamiento
Casos de uso para cada escenario
Servicios de bases de datos en Azure
Azure SQL
Crear una base de datos desde el portal de Azure
Configuración del firewall de Azure SQL
Crear una base de datos desde línea de comandos
Conectándome a mi base de datos desde VS Code
Operaciones CRUD desde VS Code
Operaciones CRUD desde SQL Management Studio
¿Qué es SQL Data Warehouse?
Usando un API para acceder a blob storage
¿Qué es un almacén de datos?
Creando un almacén de datos de SQL
Conectándome a mi almacén de datos
Extrayendo información de mi fuente de datos (importando información a mi fuente de datos)
Mejores prácticas de uso de un almacén de datos (Despliegue de datos con Power BI)
¿Qué es una base de datos para PostgreSQL?
¿Qué es Azure Cosmos DB?
¿Cómo decidir qué base de datos utilizar en cada caso?
Creando una base de datos de PostgreSQL desde el portal
Creando una base de datos de PostgreSQL desde terminal
Conectándome desde .NET a mi base de datos

Operaciones CRUD desde mi aplicación de consola

Configurando mi entorno de Cosmos DB

Mi cliente .NET para Cosmos DB

Despliegue de datos con Power BI

Servicios Web y Móviles

Servicios Web y Móviles en Azure

¿Qué son las Azure Web Apps?

¿Por qué utilizar Web Apps?

Características y herramientas principales de Web Apps

Acceso y control de archivos en una Web App por FTP

Creando un sitio web estático con una plantilla HTML

Crea una web app y publicándola desde GitHub

Escalando mi aplicación web desde terminal

¿Qué son las Mobile Apps? y ¿Por qué utilizar Mobile Apps?

¿Cómo funcionan las Mobile Apps?

Cómo crear una Mobile App

Configurando tablas simples en mi Mobile App

Ejemplos de aplicaciones conectadas a mi Mobile App

Editando mi información desde un cliente

Utilizando el API REST de Mobile Apps

Crear mi API con ASP.NET desde Visual Studio

Publicar mi aplicación en Azure

Agregando un cliente para una API REST en mi proyecto

¿Qué son las Logic Apps de Azure? y Conectores en Logic Apps

Flujo de trabajo en Logic Apps

Creando tu primera Logic App

Instala las herramientas de Logic Apps en Visual Studio

Crea y despliega tu Logic App desde Visual Studio

Edita tu Logic App desde Visual Studio

¿Qué son los Azure Notification Hubs?

Registrando mi aplicación en la tienda de Windows

Configurando mi centro de notificaciones

Conectando mi aplicación al centro de notificaciones

Probando mis notificaciones desde el portal y desde consola

Fusionando Mobile Apps con Notification Hubs

Cierre del curso

Despedida del curso

AUTENTICACION CON OAUTH

Bienvenida e introducción

Introducción y bienvenida

Stack de seguridad para aplicaciones modernas

Autenticación

Autorización

JSON Web Tokens

JSON Web Tokens

Autenticación tradicional vs JWT

Configuración inicial de los proyectos

Firmando un JWT

Verificando nuestro JWT firmado y buenas practicas con JWT

Server-Side vs Client-Side sessions

Protegiendo nuestros recursos con JWT

Habilitando CORS en nuestro servidor

Profundizando el concepto de JWKS

OAuth 2.0

¿Qué es OAuth 2.0?

Cómo elegir el flujo adecuado para OAuth 2.0

Conociendo el API de Spotify

Creando los clientes de Spotify y servicios iniciales

Implementando Authorization Code Grant

Usando nuestro access token para obtener nuestros recursos

Implementando Implicit Grant

Implementando nuestro servicio de autenticación

Modificando nuestro Layout

Implementando Client Credentials Grant

Implementando Resource Owner Password Grant

Implementando Authorization Code Grant (PKCE)

Open ID Connect

¿Qué es OpenID Connect?

Implementando OpenID Connect

Preocupaciones con JWT y OAuth 2.0

¿Cuáles son las preocupaciones con JWT?

¿Cuáles son las preocupaciones con OAuth 2.0?

Haciendo uso de Auth0

¿Qué es Auth0?

Auth0 Lock y auth0.js

Universal Login

Social Login con Auth0

Custom Social connection con Spotify

Multifactor authentication

Authorization Extension en Auth0

Consideraciones para producción

Buenas prácticas para el despliegue en producción

Uso de diferentes tenants para producción con Auth0

Cierre del curso

Cierre del curso

BASES DE DATOS EN AWS

Introducción

Introducción

Introducción a RDS

Características de RDS

Desplegando nuestra primer base de datos

Conexión gráfica a nuestra base de datos

Creación de una tabla

Conexión por consola a nuestra base de datos

Base de Datos corporativa en RDS

Backups, Performance y HA en RDS

Estrategias de backup

Demo estrategias de backup

Estrategias de performance en RDS

Despliegues Multi AZ

Migración a RDS

Estrategias de migración a RDS

Migraciones homogéneas y heterogéneas

Casos de uso de RDS

Aurora

Introducción a Aurora

Características de Aurora

Aurora Serverless

Casos de uso de Aurora

Introducción a DynamoDB

Características de DynamoDB

Consistencia en DynamoDB

Creando nuestra primer tabla en DynamoDB

Casos de uso en DynamoDB

Base de Datos corporativa para encuestas en DynamoDB

Particiones e Índices en DynamoDB

Índices y particiones en DynamoDB

Operaciones Scan en DynamoDB

Operaciones Query en DynamoDB

Demo de operaciones Scan y Query en DynamoDB

DynamoDB Streams y Replicación

Características Streams y Replicación en DynamoDB

Casos de uso Streams y Replicación en DynamoDB

DAX: DinamoDB Accelerator

Contenido Bonus

Conclusiones

FUNDAMENTOS DE AWS CLOUD

Introducción al cómputo en la nube

¿Qué es el cómputo en la nube?

¿Cómo puedo empezar a usar AWS?

Introducción a la oferta de servicios de AWS y sus aplicaciones

Introducción a la oferta de AWS y sus interacciones

Ejemplo de arquitectura con Elastic Beanstalk

¿Qué es EC2?

Creando una instancia de EC2

Conectándonos a nuestra instancia desde Windows

Conectándonos a nuestra instancia desde Linux

Conectándonos a nuestra instancia desde OSX

Subiendo un proyecto: Clonando un repositorio de GitHub a nuestra Instancia de EC2

Subiendo un proyecto a nuestra instancia de EC2: Ejecutar nuestro proyecto

¿Qué es Lambda y Serverless?

Creando una función Lambda

Elastic Beanstalk

Conociendo Elastic Beanstalk

Creando un ambiente en Elastic Beanstalk

Almacenamiento - S3

Almacenamiento con S3: Contenido Estatico

Almacenamiento con Glacier: Contenido duradero

Bases de Datos

Bases de Datos - RDS Aurora PG

Conociendo RDS PG

Haciendo una migración a RDS PG

Conociendo Aurora PG (Postgress)

Haciendo una migración a Aurora DB

Redes

Redes - Route53

Herramientas de administración

Herramientas de administración - IAM

Utilizando IAM

CloudWatch

CloudTrail

Seguridad

Certificate manager

GuardDuty

Bonus

AWS Rekognition

Creando nuestro servicio de Postgres

Accediendo a nuestra base de datos Postgres

Cierre del curso

Conclusiones del curso

NETWORKING Y CONTENT DELIVERY EN AWS

Introducción

Antes de Empezar el Curso

Introducción

Servidores y CDN

¿Qué es un servidor de nombres?

¿Qué es un CDN?

¿Qué es un endpoint?

Arquitectura en AWS

Arquitectura Web en AWS

¿Qué es VPC teoría?

Práctica con VPC

Practica con Jumpbox

Reto: Alta disponibilidad con VPCs

CloudFront

CloudFront

Preparandonos para CloudFront

Distribuyendo material en CloudFront

Reto: Distribuye un CSS a través de CloudFront.

Route 53

Introducción a Route 53

Comprando un dominio en Route 53

Configurando el dominio

Cambiando de ambientes de producción a través de Route53.

Reto: Configura ambientes de test y producción con Route 53.

API Gateway

API Gateway

Gateway

Crear una API Gateway con Lambda

Crear una API Gateway con distintas versiones

Reto: Configuración de API Gateway con Elastic beanstalk

Recetas

Recetas y Hosting S3, Dominio en Route 53

Hosting S3 a EC2, editando el dominio en Route 53

Creando ELB en una sola VPC con App y DB

API Gateway + Servidor EC2

CloudFront para todo un sitio estático

Conclusiones

CURSO DE STORAGE EN AWS

Bienvenida e Introducción

Introducción al storage en AWS

Proyecto de arquitectura e implementación de almacenamiento

Almacenamiento de objetos en AWS (S3)

Características de S3

Versionamiento de archivos en S3

Sitio web estático

Logs a nivel de objetos

Transferencia acelerada

Eventos en S3

Replicación

Clases de storage en S3

- S3 - Estándar
- S3-IA
- S3-IA única zona
- Glacier
- Ciclo de vida
- Estrategias de migración a la nube
- Casos de uso.
- Seguridad en S3.**
- Encriptación en S3 - Llaves administradas por AWS.
- Encriptación en S3 - Llaves almacenadas en AWS creadas por el Usuario.
- Encriptación en S3 - Llaves del Usuario
- Encriptación en S3
- Introducción a Políticas en S3
- Ejemplos de Políticas en S3
- ACL en S3
- Storage Gateway**
- Características de storage gateway
- File Gateway
- Virtual Tape Library
- Volume Gateway
- Sistema de archivos elástico (EFS)**
- Elastic File System
- Casos de uso de EFS.
- Sistema de archivo por bloques (EBS)**
- Características de Elastic Block Storage
- Tipos de EBS - GP2 - IO1
- Tipos de EBS - ST1 - SC1
- Snapshots y AMI
- Volumen EBS para Windows
- Volumen EBS para Linux
- Conclusiones**
- AWS Storage S3 vs EBS vs EFS, Cuándo usar cada uno
- Conclusiones

CLOUD COMPUTING CON AWS

- Presentación**
- Agenda
- EC2**
- Introducción a EC2
- Tipos de instancias al momento de crear un EC2
- Instalación del proyecto
- Imágenes de instancias
- Snapshots y sus operaciones
- Configuración de Red
- Balanceadores de carga
- Balanceadores de carga con https
- Marketplace de AMIs
- Reto EC2
- Lightsail**
- Instalando Frases Citables
- Introducción a Lightsail
- Marketplace LS
- Comparativa
- Creando un VPS
- Creando una BD
- Reto Lightsail

- ECR/ECS/EKS**
- Introducción a ECR/ECS/EKS
- Introducción a ECS
- Corriendo un contenedor
- Instalando ambiente docker en EC2
- Introducción a EKS
- Configuración kops / k8s en AWS
- Reto ECR/ECS/EKS
- Lambda**
- Introducción a Lambda
- Configuración de función lambda
- Reto Lambda
- Elastic Beanstalk**
- Configurando un ambiente para una nueva versión
- Introducción a Elastic Beanstalk
- Creando un ambiente
- Reto: Sube una aplicación en Elastic Beanstalk
- Conclusiones**
- Trabajo Futuro

FUNDAMENTOS GOOGLE CLOUD PLATFORM POR XERTICA

- Introducción a Google Cloud Platform**
- Introducción al curso
- ¿Por qué usar Google Cloud Platform?
- ¿Qué es la Nube?
- Cómo es la Infraestructura de Nube de Google
- Cómo se agrupan y gestionan los recursos en GCP
- Cómputo**
- Compute Engine
- App Engine
- Demostración de App Engine
- Kubernetes Engine
- Serverless - Cloud Functions
- Almacenamiento de Datos**
- Cloud Storage
- Cloud SQL
- Datastore - Provisionar instancia Cloud SQL y subir Datos
- Big Data**
- Introducción a Big Data
- BigQuery DataStudio y Pub/Sub para automatización de eventos
- Dataflow y DataStudio
- BigData Demo
- Machine Learning**
- ML APIs y ML Engine
- Demostración de Visual Recognition
- Servicios Complementarios (BONUS)**
- Stackdriver
- Security
- Free Tier
- Activación de Cuenta GCP con 300 USD de Crédito
- Conclusiones del curso
- Certíficate como Associate Cloud Engineer con Platzi y Xertica**
- Información y condiciones becas Associate Cloud Engineer con Platzi y Xertica

GOOGLE APP ENGINE POR XERTICA

- Introducción a Google App Engine**

Antes de comenzar

Bienvenidos al curso de Google App Engine

¿Qué es google AppEngine?

App Engine Standard vs App Engine Flexible

Servicios de App Engine

Publicando una aplicación en App Engine

Despliegue de aplicaciones Google SDK

Hola mundo en App Engine

Manejo de versiones en App Engine

Dominios personalizados y certificados SSL

Configuración de certificados SSL y dominios personalizados

Integrando servicios de almacenamiento

Cloud DataStore

Reto de DataStore, crea tu primer query

Cloud Storage

Integrando Google Cloud Storage

Arquitectura y optimización

Tipos de instancias en App Engine

Colas de Tareas

Tareas programadas (CRON)

Creando servicios REST

Funcionamiento de Google Cloud Endpoints

Creación de servicios WEB usando Google Cloud Endpoints

Consumo de APIs

Seguridad y control de accesos

Análisis de Seguridad

Reglas de Firewall

Identity Access Proxy

Servicios de apoyo (BONUS)

Cloud Source Repositories

Stackdriver Debugger

Práctica StackDrive Debugger

Error Reporting

Login

Traces

Conclusiones

Conclusiones del curso

GOOGLE KUBERNETES ENGINE CON XERTICA

Introducción a Google Kubernetes Engine

Introducción a Google Kubernetes Engine

¿Qué es Kubernetes?

Repaso de Docker

Qué son los contenedores

Qué es Docker

Comandos Básicos de Docker

Creación del contenedor de Docker

Conceptos Basicos

Cluster y nodos ¿Qué son y cómo funcionan?

¿Qué son los Pods?

Deployments

Pods y Deployments en practica

Servicios

Servicios en práctica

Servicios en práctica / Loadbalancer

Servicios en práctica / Node Port

Crea tu primer Hola Mundo en Google Kubernetes Engine

Conceptos intermedios

Archivos descriptivos de infraestructura teoria y práctica

Labels teoría

Namespaces teoría

Labels y Namespaces explicación gráfica

Labels y Namespaces en práctica

Deployment Avanzado (Blue-green y Canary Deployment)

Implementando Deployment avanzados

Terminando de implementar nuestro canary deployment

Volúmenes

Volúmenes en práctica

RETO: Crea un blue green deployment

Arquitectura de una app más robusta

Desplegar una aplicación Stateful Teoría

Desplegar una aplicación Stateful Práctica

Auto Scaling teoría

Auto Scaling en práctica

Node Pools teoría

Node Pools práctica

Contenido Bonus

Generando un certificado https con Let's Encrypt para GCP

Certificado https

Service mesh con Istio

StackDriver

CI/CD con Cloud Builder

CI/CD con Cloud Builder práctica

Knative, GKE Serverless Addon

Comandos utilizados en el curso

Buenas prácticas

IBM CLOUD

Introducción al Curso de IBM Cloud

Introducción al curso de IBM Cloud

IBM Cloud, la nube empresarial

Te presento a IBM

Recursos de IBM para diferentes perfiles

IBM Cloud, la nube empresarial

Arquitectura para Disrupción

Creación de tu cuenta IBM Cloud

Navegando IBM Cloud, conociendo la plataforma

Catálogo de componentes basados en OpenSource de IBM Cloud

Git y GitHub

Crea tu primera aplicación en IBM Cloud

Arquitectura de microservicios

Microservicios

Toolchain o herramientas de Cloud Foundry

DevOps en Cloud Foundry

IBM Cloud, Integración y Desarrollo

Introducción a la Economía de las APIs usando IBM API Connect

Introducción a IBM API Connect

Creación de APIs usando IBM API Connect

Continuando con la creación de APIs usando IBM API Connect

Habilitar la gestión de APIs en aplicaciones Cloud Foundry

IBM Cloud, Una Experiencia de Desarrollo

Conociendo las alternativas de despliegue de aplicaciones a IBM Cloud

Conociendo los aceleradores boilerplates
Desplegando aplicaciones cloud-native
IBM Watson, La Solución Cognitiva
Introducción a la computación cognitiva
Conociendo los servicios de Watson en IBM Cloud
Lanzamiento del Curso de Fundamentos de IBM Cloud
Lanzamiento del curso

BIG DATA EN AWS

Bienvenida e introducción al curso
Iniciando con Big Data
Cloud Computing en proyectos de BigData
Introducción al manejo de datos en Cloud
Datos en Cloud
¿Qué nube debería utilizar en mi proyecto de Big Data?
Arquitecturas
Arquitecturas Lambda
Arquitectura Kappa
Arquitectura Batch
Extracción de información
Llevar tu información al cloud
Demo - Creando nuestro IDE en la nube con Python - Boto3
¿Cómo usar Boto3?
API Gateway
Storage Gateway
Kinesis Data Streams
Configuración de Kinesis Data Streams
Demo - Despegando Kinesis con Cloudformation
Kinesis Firehose
Demo - Configuración de Kinesis Firehose
Reto - Configurando Kinesis Firehose
AWS - MSK
Demo - Despliegue de un clúster con MSK
Transformación de Información
AWS - Glue
Demo - Instalando Apache Zeppelin
Creación del Developer Endpoint
Demo - Conectando nuestro developer Endpoint a nuestro Zeppelin Edpoint
Demo - Creando nuestro primer ETL - Crawling
Demo - Creando nuestro primer ETL - Ejecución
Demo - Creando nuestro primer ETL - Carga
AWS - EMR
Demo - Desplegando nuestro primer clúster con EMR
Demo - Conectándonos a Apache Zeppelin en EMR
Demo- Despliegue automático de EMR con cloudformation
AWS - Lambda
Ejemplos AWS- Lambda
Demo - Creando una lambda para BigData
Carga de Información
AWS - Athena
Demo - Consultando data con Athena
AWS - RedShift
Demo - Creando nuestro primer clúster de RedShift
AWS - Lake Formation
Consumo de información
AWS - Elasticsearch

Demo - Creando nuestro primer clúster de Elasticsearch
AWS - Kibana
AWS - QuickSight
Demo - Visualizando nuestra data con QuickSight
Seguridad, Orquestación y Automatización
Seguridad en los Datos
AWS Macie
Demo - Configurando AWS Macie
Apache Airflow
Demo - Creando nuestro primer clúster en Cloud Composer
Arquitectura de referencia
Clase pública
¿Qué es Big Data?

Temario del curso

Bienvenida e introducción al curso
Iniciando con Big Data
Cloud Computing en proyectos de BigData
Introducción al manejo de datos en Cloud
Datos en Cloud
¿Qué nube debería utilizar en mi proyecto de Big Data?
Arquitecturas
Arquitecturas Lambda
Arquitectura Kappa
Arquitectura Batch
Extracción de información
Llevar tu información al cloud
Demo - Creando nuestro IDE en la nube con Python - Boto3
¿Cómo usar Boto3?
API Gateway
Storage Gateway
Kinesis Data Streams
Configuración de Kinesis Data Streams
Demo - Despegando Kinesis con Cloudformation
Kinesis Firehose
Demo - Configuración de Kinesis Firehose
Reto - Configurando Kinesis Firehose
AWS - MSK
Demo - Despliegue de un clúster con MSK
Transformación de Información
AWS - Glue
Demo - Instalando Apache Zeppelin
Creación del Developer Endpoint
Demo - Conectando nuestro developer Endpoint a nuestro Zeppelin Edpoint
Demo - Creando nuestro primer ETL - Crawling
Demo - Creando nuestro primer ETL - Ejecución
Demo - Creando nuestro primer ETL - Carga
AWS - EMR
Demo - Desplegando nuestro primer clúster con EMR
Demo - Conectándonos a Apache Zeppelin en EMR
Demo- Despliegue automático de EMR con cloudformation
AWS - Lambda
Ejemplos AWS- Lambda
Demo - Creando una lambda para BigData
Carga de Información
AWS - Athena

Demo - Consultando data con Athena
AWS - RedShift
Demo - Creando nuestro primer clúster de RedShift
AWS - Lake Formation
Consumo de información
AWS - ElasticSearch
Demo - Creando nuestro primer clúster de ElasticSearch
AWS - Kibana
AWS - QuickSight
Demo - Visualizando nuestra data con QuickSight
Seguridad, Orquestación y Automatización
Seguridad en los Datos
AWS Macie
Demo - Configurando AWS Macie
Apache Airflow
Demo - Creando nuestro primer clúster en Cloud Composer
Arquitectura de referencia
Clase pública
¿Qué es Big Data?

TESTING DE VIDEOJUEGOS

Introducción
Introducción al curso
¿Cómo funciona el testing de Black Box en un estudio de videojuegos?
Diferencia entre los 2 QA's
Testing de Regresión y Testing Exploratorio
Reto
Tipos de Testing
Tipos de Testing: Funcional, Lingüístico y de Localización
Tipos de Testing: Online, Compliance, Usabilidad y Play Test
Ejemplos de casos de pruebas de baterías de testing
Quiz
Reporte de bugs y Tipos de bugs
Qué es un bug y la importancia del reporte de errores
Bug Writing Format
Prioridades de los bugs y prioridad según su ruta
Tipos de bugs: Texto, gráfico, funcional, gameplay
Tipos de bugs: Crash, Freezee, Framerate, Audio, Legal
Reto: Reporta los bugs
Áreas de un juego y bugs duplicados
RETO: Encuentra 10 Bugs en un juego
Test plan
Sistema de Trabajo
Test plan
Organizando nuestro test plan
Continuando el proceso de creación de Test Plan y tu primera batería de pruebas
Baterías de pruebas especiales
RETO: 3 cosas que consideras prioritarias antes de empezar a testear
Testing en celulares
Guía Android
Testing en celulares Android
Guía iOS
Testing en consolas
Reto: Test Plan

uTest
uTest: creando tu perfil
uTest: proyectos pagados, reportes de bugs y pagos
Cierre del curso
Charla Motivacional
Examen

TEXTURIZADO 3D

Coordenadas UV
¿Qué son las Coordenadas UV?
Preparar los Modelos 3D
UV mapping
Unwrap Granada
Continuando con el Unwrapping de la granada
Continuando con el Unwrapping de la base de la granada
Unwrapping de la tapa de la granada
Unwrapping de la tapa de la granada 2
Preparando el proceso de texturizado: setup
Creación de Materiales
Setup de Materiales
Texturizando AAA
Intro Substance Painter
Modelado High Poly
High Poly Bake
Uso de Máscaras
Generadores
Smart Materials
Granada Base
Granada Elementos Superiores
Granada Trigger
Texturizando la parte superior de nuestra granada
Añadiendo realismo a nuestra granada
Añadiendo relieves a nuestra granada
Agregando texto a nuestra granada
Generando el render final de nuestra granada
Explicaciones de cada método utilizado en la granada y qué aplicaciones puede tener
Examen del curso de Texturizado 3D
Bonus: Texturizado de Props Estilo Cartoon

MODELADO 3D

Introducción a 3DS Max
Introducción a Max y sus herramientas
Herramientas de 3Ds Max básicas
Intro Modelado para Videojuegos
Conceptos Básicos Modelado 3D
Estilos en el Modelado 3D
Modelado para Videojuegos vs Animación
High Poly vs Low Poly
Box Modeling y Sub Objetos
Tipos de Modelado 3d
SubObjetos
Tipos de Polígonos
Ejemplificando los Ngons y sus problemas
Modelando Nuestro primer Prop
Referencias para generar un Modelo 3D
Dividir un Concept Art para beneficio del Artista
Block Out de nuestro primer modelo

Construcción de nuestro primer modelo
Añadiendo detalles al modelo: Curvas y Volumen
Mejorando la forma del modelo con Flow Connect y Symmetry
FFD o deformación de elementos
¿Qué son los smothin groups y para qué utilizarlos?
Smoothing Groups

Creando la Granada

Presentación del proyecto: Granada
Modelando la Granada: Base
Modelando la Granada: Hendiduras
Modelando la Granada: Engranaje
Modelando la Granada: Parte Superior - 1
Modelando la Granada: Parte Superior - 2
Modelando la Granada: Trigger
Granada: Puliendo detalles

Implementación en Game Engines

Exportando nuestro modelo
Importando Assets a Unity

BONUS: intro a Rigging

Qué es el Rigging
Creando huesos
Generando IKs
Controladores
Uso de articulaciones cilíndricas

Conclusiones

Tips finales, ejercicios de práctica y cierre del curso
Examen del curso de Modelado 3D con Max

PRODUCCION DE AUDIO PARA VIDEOJUEGOS

Introducción al curso

Bienvenida al curso
Utilizando otros DAW's
Conociendo los videojuegos
Entendiendo el Game Design
Preproducción
Conociendo los videojuegos

Producción de música

Concepto vs. Producción - Concepto
Concepto musical de family secrets
Mezcla de la producción
Compresión de la Mezcla: comprensión de pista y compresión general
Loops - Platzi Puzzle
Música - Platzi Puzzle
Producción del Concepto
Efectos
Compresión general
Automatización y Paneos
Mostrando la Mezcla del profesor
Masterización de la producción
Jugar Platzi Puzzle

Unity y Audio

Bancos de sonido
Sonidos - Platzi Puzzle
Bonus: Sonidos desplazamiento - Platzi Puzzle
Instrumentos virtuales
Sincronización

Integración con Unity
Mezclando audio dentro de Unity
Creación de Snapshots
Reproducir sonidos de perder vida y game over
Consejos finales y conclusiones

AVANZADO DE CREACION DE RPGs CON UNITY

Introducción

Game Design del juego
La estructura y assets de un proyecto en Unity
Convirtiendo nuestros assets en Tiles
Tilemaps y creación del escenario
Bonus: FastForward de la creación del escenario, sorting layers y creación de colliders
IMPORTANTE: Archivos del curso

Personaje principal: Movimiento y Animaciones

El jugador y el movimiento
Creando nuestra primera animación
Creando el grafo de animaciones
Usando un arbol de animaciones
Mostrando la solución del desafío e implementando transición entre blended trees
Programando una cámara que siga al jugador
Corrección del bug gráfico
límites del escenario, rigid bodies
Ejercicio: diseño de interiores

Escenarios Avanzados

Transiciones entre escenas
Mantener Player y Camera entre escenas
Spawning Points: programando que ej jugador y la cámara aparezcan en el lugar correcto al cambiar de escena
Agregando Identificadores a nuestros Spawning Points para controlar mejor el flujo de nuestro juego

Enemigos Avanzados

Creando nuestro enemigo
Programando las físicas y el patrullaje del enemigo
Generando movimiento completamente aleatorio
Programando el ataque del enemigo
Crear Manager de Health del Player
Crear armas
Programando el ataque del Player con arma
Mover la espada al caminar
Creando el ataque con espada
Ejecutando el ataque con un botón
Movimiento en diagonal
Optimizando nuestro player controller
Ataque mejorado
Uso de partículas
Añadir el daño del enemigo en batalla
Programando los contadores de vida del enemigo
Colocando más info de UI en pantalla
Script de la vida

Personaje principal avanzado

Añadir el daño del personaje (ejercicio)
Sistema de puntos de experiencia para nuestro personaje principal
Level Up!

Level Up! Defensa y Reto Final del Módulo: Stats de los enemigos

Creando un NPC

Limitar el movimiento de nuestro NPC

Creando una pantalla de diálogos para nuestro RPG

El diálogo del NPC

Múltiples líneas de diálogo

Parar el NPC durante el diálogo

Parar el personaje durante el diálogo

Quests

La estructura de una quest

Quest 1: Ir a un lugar

Quest 2: Encontrar un objeto

Quest 3: Matar enemigos

Revisión de bugs

Mantener la cámara dentro del escenario

El problema al cambiar de escena

Audio

Agregando SFX a nuestro videojuego

Agregando música a nuestro videojuego

Ajustar volumen del audio de cada uno de los efectos de sonido

Creando un VolumeManager

Agregando economía a nuestro juego y cierre

ESCRITURA DE PERSONAJES Y NPC

Introducción

Introducción al Curso

¿Por qué es importante el diseño de personajes?

Personajes principales, secundarios y NPCs

¿Cómo vamos a enfocar el diseño de personajes?

Personajes, Personalidad y Emoción

¿Cómo vamos a dar personalidad y vida a nuestros personajes?

El Eneagrama

¿Qué es el eneagrama?

Eneatipos - el Reformador, el Ayudador y el Triunfador

Eneatipos - el Individualista, el Investigador, el Leal

Eneatipos - el Entusiasta, el Desafiador, el Pacificador

Relación entre los eneatispos

El Eneagrama, pecados y emociones

El Tarot

Diseño a través del Tarot

Arquetipos - el Loco, el Mago, la Sacerdotisa

Arquetipos - la Emperatriz, el Emperador, el Sacerdote

Arquetipos - los Amantes, el Carro, la Fuerza

Arquetipos - el Ermitaño, la Rueda de la Fortuna, la Justicia

Arquetipos - el Colgado, la Suerte, la Templanza

Arquetipos - el Diablo, la Torre, la Estrella

Arquetipos - la Luna, el Sol, el Juicio

Arquetipo - el Mundo

DISC

¿Qué es DISC y por qué funciona en el diseño de personajes?

¿Cómo funciona DISC?

Aplicación de DISC al diseño personajes

Personajes y NPCs

Proyecto: Crea personajes memorables

¿Qué son los NPCs y por qué son importantes?

Tipos de NPCs y Ejemplos

Creación de personajes

Creación de personajes en grupo

Creación de NPCs

NPC's Activos y Pasivos

Conclusiones

Comentarios finales del Curso

ESTRATEGIA DE MARKETING Y MONETIZACION

1 - Bienvenida al curso

¿Qué son los videojuegos como servicio?

Por qué NO deberías odiar un Game As A Service

Diferencia entre un servicio y un bien

Diferencia entre un videojuego como servicio y un videojuego como bien

2 - El cambio de paradigma en la industria

Producción de videojuegos como bienes

Producción de videojuegos como servicios

Ejemplos de videojuegos como servicios

Modelos de Negocio de un videojuego como servicio

Transformando un videojuego como servicio

Transformando un videojuego como servicio

3 - Monetización y diseño de juego

Estructura de negocio en un videojuego como servicio

Retos del diseño de videojuegos como servicio

Diseño del sistema de un videojuego como servicio

Elementos de una economía virtual en un videojuego como servicio

Tipos de Microtransacciones en videojuegos como servicio

Game Design: Análisis de mecánicas de diseño de videojuegos

Game Design: Análisis de mecánicas de interacción social

Análisis de mecánicas de Game Design del videojuego Clash Royale

Monetización a través de Ads

Planificación de una economía virtual en un videojuego

Diseñando una economía virtual

4 - Métricas de Juegos como servicio

Diferencia de estadística descriptiva e inferencial en un videojuego como servicio

Cómo definir una métrica en un videojuego

Las mejores herramientas de medición en un videojuego como servicio

Clasificación Métricas: Funnel de Adquisición

Análisis de Métricas de adquisición

Análisis de Métricas de activación, retención, revenue

Dashboard de KPIs para videojuegos como servicio

Definiendo nuestro dashboard de KPIs

5 - Marketing para juegos como servicio

Consideraciones de marketing para videojuegos como servicios

Definición de Buyer persona y Journey Usuario en videojuegos como servicios

Importancia de Paid Media en videojuegos como servicios

Consideraciones de Paid Media en videojuegos como servicios

Importancia de Owned Media en videojuegos como servicios

Consideraciones de Owned Media en videojuegos como servicios

Importancia de Earned y Shared Media en videojuegos como servicios

Consideraciones de Earned y Shared Media en videojuegos como servicios

Creación de una estrategia de marketing para videojuegos como servicios
Ejecución de una estrategia de marketing de un videojuego como servicio
Ejecutando una estrategia de marketing de un videojuego como servicio
6 - Game as a Service
Evaluación de un videojuego clásico a un videojuego como servicio.
Conclusiones del curso

ANIMACION Y ARTE DE VIDEOJUEGOS

Bienvenido al curso

Proyecto
Arte 2D
Introducción
Nuestro juego, corrientes y estilos
Pipeline en arte
Mood board
Referencias y Mashup
Concept Art
Definiendo el personaje
Composición con formas geometricas
Siluetas
Anatomía y proporción
Color
Texturizado sobre 3D
Colorización del texturizado 3D
Perspectiva
Composición
Iluminación
Demo color y organización de escenario

Arte 3D
Introducción al arte 3D
Polígonos
Topología
Zbrush
Millones de polígonos
Mapas
Comunicación
Motores de render
Sistemas de huesos
3ds Max
Cierre

Pixel Art
Introducción al Pixel Art
Pixel Art con Photoshop
Herramientas de Graphics Gale
Line Art y Pixel Art con Graphics Gallery
Dibujando la cabeza de una persona en Graphics Gallery
Dibujando el cuerpo de una persona en Graphics Gallery
Escenario para el personaje
Animación básica del personaje
Exportar animación
Cierre
Desafío 01
Animación

Principios de la animación
Squash y Stretch
Anticipación
Animación pose a pose
Animación directa y superpuesta
Puesta en escena
Slow in y slow out (acelerar y desacelerar)
Arcos
Acción secundaria
Timing
Exageración
Dibujos sólidos
Atractivo

MARKETING Y NEGOCIOS PARA VIDEOJUEGOS

Producción de un videojuego
Bienvenidos - ¿Qué se verá en esta sección?
Estructura de la industria del videojuego
Modelos de la industria: Indie, Triple-I, AAA
Roles en el desarrollo de un videojuego
Todo parte de un GDD
Business Plan
Fases en el desarrollo de un videojuego
Plan de producción
Metodología de trabajo
Documentación básica
Relaciones desarrollador - publisher
Comunicación para desarrolladores indie
¿Qué significa ser desarrollador indie en 2016?
Requesitos mínimos para contactar a la prensa de videojuegos
Contacta a los medios utilizando Press Kit
La importancia de un timing adecuado en la comunicación
El elemento cultural como herramienta de promoción gratuita
¿Qué esperamos de ti en un medio como Nivel Oculto?
Marketing de videojuegos
El Plan de Marketing: ese gran desconocido
Vende tu juego sin vender tu juego
No dejes que tu primo gestione tus Redes Sociales
Aprende a escribir (otra vez)
Habla con los medios aunque no te escuchen
ASO - App Store Optimization
Comprar clientes no es tan caro
Mide, analiza, implementa...y empieza de nuevo

PROGRAMACION DE VIDEOJUEGOS 2D CON UNITY

Introducción a Unity
¡Bienvenido a los cursos autorizados de Unity!
Introducción al Desarrollo de Videojuegos y el motor de juegos Unity
Software necesario para desarrollo en móvil
Conociendo la interfaz de Unity
¿Cómo funcionan las versiones de Unity?
Importación de assets y setup para móvil
Preparación del espacio de juego
Primer script y movimiento del jugador
Limitando el movimiento de nuestra paleta
Introducción a la creación de videojuegos

Físicas en Unity y movimiento de la pelota

Agregando movimiento y Colisiones en Unity

Otras configuraciones de movimiento y colisiones

Sistema de UI simple

Detectando puntos

Condición de gane

Condición de victoria y transición entre escenas

Programando nuestro oponente

Agregando Audio y desafío

Exportando a Android

Creación de un juego de Plataformas

Introducción al segundo proyecto

Creación de nuevo proyecto e importación de assets

Control de personaje principal: Caminado

Control del personaje principal: Salto, brinco

Material de estudio: ¿Cómo programar el movimiento de un personaje?

Tilemap system en Unity 1

Tilemap system en Unity 2

Cinemachine2D

Cinemachine2D: Presentación del reto y limites de la cámara

Sistema de animación en Unity: Personaje principal, preparar animaciones)

Sistema de animación en Unity: Personaje principal, conectar en código

Sistema de animación en Unity: Uniendo nuestros parámetros a los inputs

Movimiento del enemigo y condición de muerte

Sistema de animación en Unity: Enemigo

Objetos coleccionables y UI

Añadiendo puntos a nuestro score

Sistema de partículas

Conectando partículas a objetos

Música de fondo y efectos de sonido

Uso de prefabs para replicar nuestro coleccionable

Condición de victoria

Preparando nuestros enemigos

Lógica de vida del jugador y condición de derrota

Conclusiones

Conclusiones del curso

INTRODUCCION A LA CREACION DE PERSONAJES

Introducción al curso

Presentación del curso

Creación de personajes

Conceptos básicos de Diseño de Personajes

Proceso Pipeline

Continuando con el proceso Pipeline

Requerimientos del sistema

Pasos para el diseño

Proyecto

Análisis de personajes

Adquisición de referencias

Reto

DISENO Y PROTOTIPADO DE PERSONAJES

Introducción al curso

Bienvenidos al Curso de Diseño y Prototipado de Personajes

Diseño y Prototipado de Personajes

Thumbnails

Sketches

Color Test

Ilustración

Polished

Presentación final

Poses clave

Personaje de soporte y Reto

Conclusiones del curso

¿Cómo aprobar el Curso de Diseño y Prototipado de Personajes?

CREACION DE VIDEOJUEGOS

Introducción a la Industria

Bienvenidos al curso

Historia personal

El principio

Evolución a entornos gráficos

¿Cuándo termina la creación de un videojuego?

La edad social

Los inicios de la industria

El videojuego como negocio

Los grandes de la industria

La crisis del 83

Los primeros PC Gaming

Los genios de la época

3a generación - Los 8 bits

4a generación - Los 16 bits

5a y 6a generación

EEUU vs Japón

La era social

Industria indie y realidad virtual

Proyecto

Game Design

Presentación

El nacimiento del concepto

El core de un juego

Comunicación

Game Design

Narrativa

Presentación

Los 3 actos: El método más tradicional de contar una historia

¿Qué es el StoryTelling? Las 20/22 reglas de Pixar

Primer acto: protagonista

Primer acto: el detonante

Primer acto: el antagonista

Primer punto de giro

Segundo acto: conflicto

Segundo acto: complicación progresiva

Segundo acto: Segundo punto de giro: crisis

Tercer acto: climax

Tercer acto: resolución

Tercer acto: epílogo

Lecciones aprendidas del StoryTelling

Cierre

Método de los tres actos

Videojuego Ampliado

Historia social del videojuego

Tipos de videojuego serios

Niveles de objetivos en los videojuegos serios

Efectos psicosociales del videojuego de entretenimiento y serio

¿Cómo repensar un videojuego de entretenimiento para que sea serio?

Tipos de gamificación de videojuegos

Reglas y retos para gestionar el flujo de actividad y la dificultad

Refuerzos y castigos para la motivación intrínseca y extrínseca al usuario

Potencialidad de la gamificación

Cierre

Audio en Videojuegos

Consideraciones previas y características de las bandas sonoras para videojuegos

Crear un banda sonora: pasos del proceso y técnicas

De la composición al audio

Escogiendo música para mi juego

Los efectos de sonido

El doblaje

La implementación de la música en el juego

Cierre

Bonus: El papel del compositor en el equipo

Emprendimiento en Videojuegos

Introducción al Emprendimiento en Videojuegos con VR y AR

El arte de burlarte de ti mismo

La importancia de los hobbies

Qué son los mercados emergentes

Cómo funcionan los ciclos financieros

El estado actual de VR y AR

PlatziConf Videojuegos

El estado de la industria de los videojuegos

Integrar áreas de trabajo en el desarrollo de VideoJuegos

La experiencia de Stugan como desarrollo de una idea

¿Quién es un game designer?

El fenómeno del videojuego independiente para audiencias enfocadas

El desarrollo indie en los procesos de creación

La importancia de la música en los videojuegos

Cómo llegó Cocodrilo Dog Games a trabajar con Disney

Monetización de juegos para móviles

Cómo producir videojuegos de calidad AAA

El futuro del gaming va más allá de la Realidad Virtual: Entrevista con Mario Valle, ex Electronic Arts

DISEÑO DE VIDEOJUEGOS

Introducción al Curso de Diseño de Videojuegos

Bienvenidos al curso

¿Cómo vamos a trabajar?

El gancho

El documento de diseño

RETO: Creando tu propio gancho, deberás ser concreto

Puntos de vista y género

Puntos de vista y géneros

Puntos de vista modernos

Géneros de juego

RETO: Elige 3 de tus juegos favoritos y cámbiales la perspectiva y el estilo visual

Contando Historias

Vamos a contar una historia

Vamos a contar una historia, hablemos de Personajes

Historia conducida por el jugador

Mecánica

Vamos a contar una historia 3º parte, Giros en nuestra trama

El arco narrativo

Vamos a contar una historia diálogos, puntos de conversación y ejecución

RETO: Dibuja un arco narrativo

Inicio

Parte intermedia

Parte final

La psicología y el camino del héroe

RETO: Utiliza el camino del héroe en la historia de tu videojuego

Jung y los arquetipos del tarot

Visualización de la historia

Maneras de empezar una historia

Técnicas y ayudas para contar la historia

Giros de la trama

Giros de la historia y dilemas

Curvas de tiempo

RETO: Define tu historia

Diseño

Avanzando con nuestro Game Design Document

Escenarios

Diseño de niveles en árbol

Diseño de niveles en plano

RETO: Realiza los diseños de escenarios de tu juego

Diseño de personajes

Creando nuestros personajes

Arcos de personaje

Planeando nuestra mecánica de Juego RPG

Balanceo de videojuegos

Creación del documento de diseño

Puntos débiles y fuertes del personaje

Puntos débiles y fuertes del personaje

Arco del Personaje de Richard Estacado

Tipos de Villanos y minions

Habilidades del personaje

RETO: Realiza el diseño de tus personajes principales

Mundos

Mundos

Define el mundo que usará tu juego

Analiza un mundo abierto

Ejemplo de NPC's y Reto

Música y sonido

RETO: Consigue buena música para tu juego

Metas, recompensas y comunicación con el jugador

Revisando el Documento de Diseño completo

RETO: Metas y Recompensas

RETO: ¿Qué juegos te han marcado como jugador?

RETO: Comunicación con el jugador

Cierre del curso

Cierre del curso
¿Qué sigue después?

Temario del curso

Introducción

Introducción
¿Qué es un juego tipo RPG?
Roles en la dirección de videojuegos
Elementos clave en el desarrollo de un RPG

La historia y su mundo

Eligiendo una historia para contar
La geografía
La geopolítica

Los elementos vivos

Razas
Clases
Árbol de Habilidades
Gestión de atributos
Enemigos
El inicio del Balanceo

Economía

Economía
Tiendas y Locales
Armas y su relación con la economía
Crafting y su relación con la economía

Ampliando la historia

Facciones
Ahora sí, la historia
Variaciones en la trama
Mecánicas y sistema de combate
Misiones secundarias
Ahora sí, personajes
El final
Balanceo final

Elementos Multijugador

¿Y si quisiéramos que nuestro juego tuviera multiplayer?
Tipos de juego multiplayer MMO y CO-OP
Multiplayer asíncrono

Conclusiones

Conclusiones

Elementos FPS

Diseño de niveles
Introducción a los juegos FPS
Diseño de niveles
Adaptación de nuestro RPG al nuevo FPS

PROGRAMACION DE VIDEOJUEGOS 2D CON UNITY

Introducción a Unity

¡Bienvenido a los cursos autorizados de Unity!
Introducción al Desarrollo de Videojuegos y el motor de juegos Unity
Software necesario para desarrollo en móvil
Conociendo la interfaz de Unity
¿Cómo funcionan las versiones de Unity?
Importación de assets y setup para móvil
Preparación del espacio de juego
Primer script y movimiento del jugador

Limitando el movimiento de nuestra paleta
Introducción a la creación de videojuegos
Físicas en Unity y movimiento de la pelota
Agregando movimiento y Colisiones en Unity
Otras configuraciones de movimiento y colisiones
Sistema de UI simple
Detectando puntos
Condición de gane
Condición de victoria y transición entre escenas
Programando nuestro oponente
Agregando Audio y desafío
Exportando a Android

Creación de un juego de Plataformas

Introducción al segundo proyecto
Creación de nuevo proyecto e importación de assets
Control de personaje principal: Caminado
Control del personaje principal: Salto, brinco
Material de estudio: ¿Cómo programar el movimiento de un personaje?
Tilemap system en Unity 1
Tilemap system en Unity 2
Cinemachine2D
Cinemachine2D: Presentación del reto y limites de la cámara
Sistema de animación en Unity: Personaje principal, preparar animaciones)
Sistema de animación en Unity: Personaje principal, conectar en código
Sistema de animación en Unity: Uniendo nuestros parámetros a los inputs
Movimiento del enemigo y condición de muerte
Sistema de animación en Unity: Enemigo
Objetos coleccionables y UI
Añadiendo puntos a nuestro score
Sistema de partículas
Conectando partículas a objetos
Música de fondo y efectos de sonido
Uso de prefabs para replicar nuestro coleccionable
Condición de victoria
Preparando nuestros enemigos
Lógica de vida del jugador y condición de derrota
Conclusiones
Conclusiones del curso

Temario del curso

Introducción

Game Design del juego
La estructura y assets de un proyecto en Unity
Convirtiendo nuestros assets en Tiles
Tilemaps y creación del escenario
Bonus: FastForward de la creación del escenario, sorting layers y creación de colliders
IMPORTANTE: Archivos del curso
Personaje principal: Movimiento y Animaciones
El jugador y el movimiento
Creando nuestra primera animación
Creando el grafo de animaciones

Usando un arbol de animaciones
Mostrando la solución del desafío e implementando transición entre blended trees
Programando una cámara que siga al jugador
Corrección del bug gráfico
límites del escenario, rigid bodies
Ejercicio: diseño de interiores

Escenarios Avanzados

Transiciones entre escenas
Mantener Player y Camera entre escenas
Spawning Points: programando que ej jugador y la cámara aparezcan en el lugar correcto al cambiar de escena
Agregando Identificadores a nuestros Spawning Points para controlar mejor el flujo de nuestro juego

Enemigos Avanzados

Creando nuestro enemigo
Programando las físicas y el patrullaje del enemigo
Generando movimiento completamente aleatorio
Programando el ataque del enemigo
Crear Manager de Health del Player
Crear armas
Programando el ataque del Player con arma
Mover la espada al caminar
Creando el ataque con espada
Ejecutando el ataque con un botón
Movimiento en diagonal
Optimizando nuestro player controller
Ataque mejorado
Uso de partículas
Añadir el daño del enemigo en batalla
Programando los contadores de vida del enemigo
Colocando más info de UI en pantalla
Script de la vida

Personaje principal avanzado

Añadir el daño del personaje (ejercicio)
Sistema de puntos de experiencia para nuestro personaje principal
Level Up!
Level Up! Defensa y Reto Final del Módulo: Stats de los enemigos
Creando un NPC
Limitar el movimiento de nuestro NPC
Creando una pantalla de diálogos para nuestro RPG
El diálogo del NPC
Múltiples líneas de diálogo
Parar el NPC durante el diálogo
Parar el personaje durante el diálogo

Quests

La estructura de una quest
Quest 1: Ir a un lugar
Quest 2: Encontrar un objeto
Quest 3: Matar enemigos
Revisión de bugs
Mantener la cámara dentro del escenario
El problema al cambiar de escena

Audio

Agregando SFX a nuestro videojuego
Agregando música a nuestro videojuego

Ajustar volúmen del audio de cada uno de los efectos de sonido
Creando un VolumeManager
Agregando economía a nuestro juego y cierre

ASTROBIOLOGIA

Introducción

Bienvenida al curso
¿Qué es la Astrobiología?
¿Qué es la Vida?
Organización de la vida
El origen de nuestro hogar
Nuestro mundo

La vida

El origen de la vida
El lugar ideal
Planeta tierra como lugar ideal
La evolución de la vida
Clasificación de la vida

Extraterrestres

Civilizaciones extraterrestres
¿Dónde hay vida extraterrestre?
¿Cómo son los extraterrestres?
El gran reto

Conclusiones

Conclusiones

FUNDAMENTOS DE MATEMATICAS PARA FISICA

Álgebra

Introducción
Operaciones básicas
Números reales, complejos e imaginarios
Ecuaciones y cómo resolverlas
Sistemas de ecuaciones lineales

Matrices

Matrices
Tipos de matrices
Matriz inversa

Funciones

Funciones
Representación de funciones
Definición de derivada
Cálculo de derivadas
Integrales indefinidas
Integrales definidas

Geometría

Vectores
Geometría en el plano
Geometría en el espacio

Probabilidad y estadística

Estadística
Probabilidad
Conclusiones

FISICA MECANICA ESTATICA

Introducción

Bienvenida
Conceptos fundamentales

Sistemas de unidades
Ejercicios de conversión de unidades

Estática de Partículas

Introducción Estática de Partículas
Suma de fuerzas en el plano
Equilibrio en el plano
El diagrama de cuerpo libre
Suma de fuerzas en el espacio
Equilibrio en el espacio
Ejercicios estática de partículas

Cuerpos rígidos

Fuerza y momento
Momento de una fuerza en torno a un eje
Par de fuerzas
Simplificando sistemas de fuerzas
Ejercicios Cuerpos Rígidos

Equilibrio de los cuerpos rígidos

Reacciones y apoyos
Equilibrio en 2 dimensiones
Casos especiales de equilibrio de fuerzas
Equilibrio en 3 dimensiones

Análisis de estructuras

Reticulados
¿Cómo funciona un reticulado?
Método de los nudos
Método de las secciones
Marcos
Ejercicios análisis de estructuras

Futuros pasos

Conclusiones

GLOSARIO

Abstracción

Es un marco de referencia que forma parte del conjunto de librerías de programación dentro de la Common Language Infrastructure (CLI)

Abstracción

En desarrollo y diseño es un análisis que se realiza aislando una propiedad específica de un objeto ignorando las demás.

ACID

En desarrollo es el componente controlador de la interfaz gráfica escrita en XML.

Activity

Es el componente controlador de la interfaz gráfica escrita en XML.

ADB

Forma parte del SDK de Android que permite la conexión desde Android a la PC.

ADB Manager

Es el administrador de paquetes del SDK o Software Development Kits de Android que permite la conexión desde Android a la PC.

AJAX

De sus siglas en inglés Asynchronous JavaScript and XML. Es una técnica popular para cargar contenido desde un servidor y colocarlo en una página sin necesidad de que se actualice.

AKS

Azure Kubernetes Service es un servicio de Kubernetes administrado y provisto por Azure.

Algoritmo

Conjunto ordenado de operaciones sistemáticas que permite hacer un cálculo y hallar la solución de un tipo de problemas.

Amenaza

Posible causa de un incidente no deseado, que puede resultar en daños a un sistema u organización

Android

Sistema operativo creado por Google para celulares.

Android Studio

Entorno de desarrollo para crear aplicaciones de Android

Antispam

Herramienta informática que se encarga de eliminar correo no desado

Antivirus

Software encargado de identificar programas que son diseñados con intenciones destructivas o dañinas.

API

Aplicación que funciona como interfaz de programación, la cual es usada como biblioteca y proporciona interoperabilidad entre sistemas.

APK

Application Package, es el archivo ejecutable de Android generado después de la compilación.

Aplicaciones Bridge

Aplicaciones que utilizan lenguajes alternos a los nativos y que gracias a su motor logran hacer la traducción.

Aplicaciones Escritorio

Aplicaciones que se encuentran instaladas en un sistema de almacenamiento como un disco duro y pueden correr sin internet.

Aplicaciones Híbridas

Aplicaciones creadas con tecnologías para desarrollo web que, junto con un framework, tiene la capacidad de correr como aplicación para cualquier sistema operativo en cuestión

Aplicaciones Móviles

Aplicaciones desarrolladas con los lenguajes oficiales de un sistema operativo y soportados por el propietario. Corren dentro de dispositivos móviles como celulares o tablets.

Aplicación maliciosa

Conocida en inglés como Malware, es una aplicación que una vez instalada se encarga de robar información para cometer fraudes o robo de identidad.

Aplicación Nativa

Es el software desarrollado utilizando el lenguaje de programación oficial de un sistema operativo.

App Escritorio

Software que se encuentra instalado en el sistema operativo y se puede ejecutar sin depender de un navegador web.

App Híbrida

Software creado con tecnologías web que, junto con un framework, tiene la capacidad de correr como aplicación móvil en Android, iOS y Windows Phone.

Apple

Apple es una empresa estadounidense fundada por Steve Jobs que diseña y produce equipos electrónicos, software y servicios en

línea. Su sede principal está ubicada en Cupertino, California Estados Unidos.

App Multiplataforma

Aplicaciones que pueden ejecutarse en diferentes sistemas operativos ya sea en un dispositivo móvil, en un desktop o desde un navegador web.

App Nativas

Es la forma de desarrollo, que compete a desarrollar con los lenguajes oficiales y soportados por el propietario.

ASP

(Active Server Pages). Es un marco de referencia diseñado para desarrollo web

Ataque

Un asalto a la seguridad del sistema, derivado de una amenaza inteligente; es decir, un acto inteligente y deliberado (especialmente en el sentido de método o técnica) para eludir los servicios de seguridad y violar la política de seguridad de un sistema

Autenticación

Es referente a todas las formas que utiliza una aplicación para permitir o no el acceso de un usuario a sus funcionalidades.

Automatización

Procesos automatizados para procesamiento de información.

Autorización

Son todos los controles para que únicamente el usuario que esta autorizado para entrar a cierto recurso pueda hacerlo y otro no lo pueda hacer.

AVD

Android Virtual Device, emula un dispositivo Android.

AVD

Android Virtual Device, emula un dispositivo Android.

AWS

Amazon Web Services: es una plataforma de servicios de cloud computing para empresas de cualquier tamaño.

B

Backend

Parte del software que actúa fuera de los ojos del usuario, en el desarrollo web es la capa que se encarga de manejar la información y las reglas del negocio

Backpropagation

Es una técnica de aprendizaje supervisado donde se computan errores a la salida esperada y se distribuyen de forma regresiva en las capas de la red neuronal. Se utiliza para comparar entradas de datos con salidas esperadas, de forma que el sistema realice los ajustes necesarios.

Balanceo de cargas

Reparte la diferentes conexiones que recibe un servidor web en diferentes servidores, nos sirve para dar disponibilidad a nuestras aplicaciones.

Base de Datos

Las bases de datos son contenedores o repositorios digitales en los que se almacenan datos de manera estructurada y organizada para su posterior procesamiento y consulta.

Base de Datos No Relacional

También conocida como NoSQL son las Bases de Datos que no siguen el modelo Relacional y que tampoco utilizan SQL como lenguaje de consultas, se caracterizan también por no seguir los

principios ACID y también por utilizar colecciones en vez de tablas para almacenar datos.

Base de Datos Relacional

Consiste en almacenar datos en forma de relaciones, todo dentro de un esquema lógico que en este caso son tablas, las relaciones son creadas a través de columnas en las tablas que hacen referencia a otro registro.

Bases de datos

Las bases de datos son contenedores o repositorios digitales en los que se almacenan de manera estructurada y organizada datos que comparten un mismo contexto para su posterior procesamiento y consulta.

Bases de Datos Relacionales

Conjunto de datos organizados y relacionados por un valor en común.

Batch Jobs

Tareas programadas que el orquestador ejecuta en un momento determinado

BDD (Behavior-driven development))

Cuando hablamos de pruebas, es lo que se basa en desarrollo impulsado por el comportamiento.

Big Data

Conjunto de información cuya longitud (número de registros), variedad (número de características por cada registro) y frecuencia (ritmo con el cual se actualiza la información) impiden que los métodos de procesamiento de datos convencionales sean suficientes para su manejo.

Biología

Es la ciencia que estudia a los seres vivos y sus procesos vitales, por ejemplo su origen, evolución propiedades, nutrición, morfogénesis, reproducción o patogenia.

Blog

Es un tipo de aplicación o sitio web en el que un autor o creador publica artículos de texto enriquecido de manera regular y periódica.

Blue-green

Estrategia de despliegue donde se duplica la infraestructura y luego se efectúa un enrutamiento del tráfico.

Bots

Ordenadores controlados de manera remota por los hackers. El término también se refiere a programas informáticos que realizan actividades repetitivas y pueden contar con cierta inteligencia artificial como los bots usados en los chats.

Bytecode

Es el código leído por la Máquina Virtual de Java y generado por el compilador.

C

Cálculo

Son operaciones matemáticas que tienen por objetivo el alcance de un dato o información y que requieren el desarrollo de un proceso previo a la obtención de ese resultado.

Canary deployment

Estrategia de despliegue donde gradualmente se actualizan las réplicas de un servicio hasta llevar todas sus instancias a la nueva versión.

Cgroups

Control groups, utilizados para limitar y medir los recursos del sistema operativo

Ciencia de Datos

Área de conocimiento que busca agregar valor a los datos por medio de la confluencia de análisis estadísticos, métodos computacionales y visualización de información.

Ciencias de datos

Campo de la ingeniería que usa diversos metodos estadisticos para extraer informacion valiosa y en base a ello obtener hallazgos que sean valiosos para toma de decisiones.

Clase

Es la abstracción de un objeto, contiene la forma general del objeto. Su objetivo es servir como template para general más objetos.

CLI

Command Line Interface o interfaz de linea de comandos, permite al usuario interactuar con un programa enviando comandos de texto haciendo uso de la terminal o consola del sistema operativo.

Client-Side (Lado del cliente)

Se refiere a un lenguaje que es interpretado por el propio navegador. Por ejemplo, JavaScript es descargado por tu navegador y luego se ejecuta localmente en tu máquina. Tu navegador es el "cliente".

Cloud

Cloud o “la nube” es el término con el que se describe al conjunto de servicios e infraestructura tecnológica disponible de manera remota a través de internet.

Cloud computing

El término cloud computing se refiere a la estrategia (de pago) de almacenar en bases de datos aplicaciones y otros recursos gestionados y procesados con el uso de una red de servidores hospedados en internet y no en equipos locales como PCs o servidores personales.

CMS

De sus siglas en inglés Content Management System. Es un programa de software que se ejecuta en tu servidor con el fin de facilitar la administración del contenido en el sitio web.

CNI

Container-Network-Interface. Estándar el cual es utilizado por los diferentes vendors de networking en de K8s.

Compilador

Programa encargado de transformar código en archivos binarios. Binario: Sistema de numeración de base dos.

Componente

Es una clase responsable de exponer datos a una view y manejar la mayor parte de la visualización de la vista y la lógica de interacción del usuario.

Comunicación M2M

Es el intercambio de información entre maquinas. Machine to machine.

Concurrencia

Forma en que el procesador ejecuta hilos por un lapso de tiempo cada uno siguiendo un orden sin alterar el resultado fianl.

Confidencialidad

Propiedad de que la información no esté disponible o sea divulgada a individuos, entidades o procesos no autorizados

Continuous Delivery

Es la forma de aprovisionar de forma rápida y automática creando herramientas que apoyen el desarrollo, prueba e implementación.

Continuous Integration

Es una práctica de desarrollo que requiere que los ingenieros integren su código a un repositorio compartido varias veces al día. Cada commit es verificado por un build automatizado, lo que permite a los equipos detectar problemas de manera oportuna.

Contratos

Convenios que producen o transfieren las obligaciones y derechos.

Convenio de Budapest

Es el convenio sobre la ciberdelincuencia del Consejo de Europa.

Convenio de Budapest

Es el convenio sobre la ciberdelincuencia del Consejo de Europa.

Cracker

Hacker dedicado a romper sitios en internet por motivos de protesta, retribución económica, simplemente por desafiar la seguridad del sitio.

CRD

Custom Resource Definitions. Utilizados generalmente por los operadores de Kubernetes.

Crimeware

Software malicioso Ej: troyanos, virus, spyware,Botnets entre otros.

Criptografía

Se refiere a los diferentes algoritmos que podemos utilizar para proteger la información que es muy sensible o importante

Croot

Mecanismo por el cual se cambia la visión del file system de un proceso

CRUD

De sus siglas en inglés Create, Read, Update, Delete. Este término se utiliza para describir una aplicación web cuyo propósito general es crear, leer, actualizar y eliminar. Por ejemplo el software de un blog, donde se puede crear una entrada, leerla, actualizarla o eliminarla.

CSS

Del inglés Cascading Style Sheets (hojas de estilo en cascada). Es un archivo que termina en .css que está vinculado desde el HTML y que controla el aspecto de la página web.

D

Dalvik

Máquina virtual de Android para su funcionamiento.

Dalvik/ART

Dalvik fue la máquina virtual de Android hasta la versión 4.0, posteriormente ART fue la nueva máquina virtual Android a partir de la versión 5.0.

Database Application Firewall

Monitorea cambios dirigidos a las bases de datos relacionadas con la aplicación.

Dato

Representación simbólica de información (numerica, silabica, algoritmica, etc..).

Dato

Los datos son la representación simbólica, bien sea mediante números o letras de una recopilación de información la cual

puede ser cualitativa o cuantitativa, que facilitan la deducción de una investigación o un hecho.

dBm
Es una unidad de medida de potencia expresada en decibeles (dB) relativa a un miliwatt (mW).

DDL
Es el lenguaje definido por el sistema de gestión de base de datos basado en SQL

Decibel
El decibelio con símbolo dB, es una unidad que se utiliza para expresar la relación entre dos valores de presión sonora, o voltaje y potencia eléctrica.

Deep Learning
Técnicas de Machine Learning que se basan en transformar la información a través de una serie de capas, se usa ampliamente en reconocimiento de imágenes y problemas que puedan ser representados en una matriz.

Deployment
Recurso de alto nivel utilizado para gestionar replica-sets y servicios en un cluster.

Devops
Es la combinación de herramientas que permiten aumentar la capacidad de una organización para distribuir servicios a alta velocidad. La práctica busca unir a dos equipos que tradicionalmente han tenido procesos separados: Development y Operations.

DevOps
Modelo que mejora la comunicación entre el ambiente de desarrollo y producción a través de un enfoque de entrega continua.

DevSecOps
Es la práctica que incluye las preocupaciones de seguridad dentro de las mismo flujo de trabajo desarrollado para el DevOps.

Disco Duro
Dispositivo físico de almacenamiento de datos no volátil.

Disponibilidad
Propiedad de la información de ser accesible y utilizable sobre demanda por una entidad autorizada.

Django
Framework de python para realizar desarrollo web, es flexible y muy popular, tiene una librería para realizar apis llamada DRF

Docker
Runtime de contenedores que utiliza kubernetes para su ciclo de vida

DOM
El DOM (Document Objet Model por sus siglas en inglés) es la estructura de objetos generada por el navegador al cargar un documento que puede ser modificada mediante un lenguaje de programación para cambiar dinámicamente los contenidos y aspecto de la página.

DPI/PPI (Puntos Por Pulgada/Pixeles Por Pulgada)
De sus siglas en inglés Dots Per Inch y Pixels Per Inch. Los "puntos" son literalmente puntos físicos de tinta sobre papel que determinan la resolución de una imagen impresa. Los "pixeles" son bits de datos que determinan el tamaño de una imagen electrónica.

DRF
Django REST framework, es un kit de herramientas potente y flexible para crear APIs Web.

Dynamic Application Security Testing (Dast)
Es un tipo de prueba automatizada de seguridad que permite encontrar vulnerabilidades durante runtime, es decir, una vez que la aplicación ha sido desplegada.

E
Ecuación
Es una igualdad donde existe al menos una incógnita cuyo valor será conocido al resolver el ejercicio. Tiene expresiones algebraicas que permiten conocer los datos (valores conocidos) e incógnitas (valores que no se han descubierto) vinculados a través de operaciones matemáticas.

EFS
Elastic File System: Es un sistema de almacenamiento en la nube que permite elegir entre objetos, archivos o por bloques, así como migración de datos a la nube para diseñar las bases de entorno tecnológico en la nube.

Electromagnetismo
El electromagnetismo es una rama de la física que estudia y unifica los fenómenos eléctricos y magnéticos en una sola teoría.

Encapsulamiento
Uno de sus sinónimos puede ser “esconder” u “ocultar” algo, este va de la mano de los modificadores de acceso que generan niveles de visibilidad en los elementos de una clase.

Entrega Continua (Continuous Delivery)
Es un enfoque que permite con la ayuda de la integración continua automatizar pruebas y despliegue de aplicaciones con mínima intervención humana.

Environments
Permiten determinar cuándo y qué código ha sido desplegado por los diversos jobs del CI. Incluyen vínculos a los ambientes en donde vive nuestro código, métricas y consolas interactivas.

Escalabilidad
Crecimiento por demanda. Servicios escalables de acuerdo a la cantidad de información.

Estática
Es una rama de la física que analiza la fuerza, par / momento y estudia el equilibrio de fuerzas en los sistemas físicos en equilibrio estático, es decir, en un estado en el que las posiciones relativas de los subsistemas no varían con el tiempo.

Etiqueta
Las etiquetas son el texto que va al rededor del contenido en el código HTML para identificar el tipo de contenido que envuelven. Por ejemplo, en este código Hola!, las etiquetas son .

Express
Framework de aplicaciones web que proporciona características para desarrollar aplicaciones web, API's y aplicaciones móviles

Extracción
Extraer la información de otras fuentes, cloud providers, herramientas de terceros u on-premise.

F
Firebase

Es el BackEnd as a Service de Google utilizado para el desarrollo Web y móvil.

Firmware

Es el conjunto de instrucciones de un programa que se encuentra registrado en una memoria ROM, flash o similar. Estas instrucciones fijan la lógica primaria que ejerce el control de los circuitos de algún dispositivo.

Física

Es una ciencia natural que se encarga del estudio de la energía, la materia y el espacio-tiempo, así como las interacciones de estos tres conceptos entre sí.

Framework

Serie de herramientas/librerías de programacion con un proposito especifico

Frontend

Con este término se identifica al conjunto de tecnologías que dan estructura, apariencia y comportamientos de interactividad a los sitios y aplicaciones web modernas de cara a los clientes o usuarios finales en un navegador.

Front End

La parte de una aplicación web que ven las personas que visitan un sitio web en un navegador.

Función

Conjunto de instrucciones que realizan una tarea específica, generalmente empaquetados y llamados con un identificador simbólico (nombre).

G

Geometría

Es una rama de la matemática que se encarga de estudiar las propiedades y las medidas de una figura en un plano o en el espacio.

Gestión de configuraciones

Se refiere a todo el nivel de aseguramiento que va a tener el ambiente en el cual se va a ejecutar la aplicación.

Git

Es un servicio de control de versiones.

Gitlab AutoDevOps

Es la funcionalidad de Gitlab que permite crear un flujo completo de Devops sin necesidad de configuración por parte del usuario. El flujo es completamente configurable a través de variables de ambiente o modificando nuestro archivo ``.gitlab-ci.yml``.

Gitlab CI

Gitlab CI es el producto que ofrece Gitlab como punto central de implementación de la filosofía Devops y es la herramienta principal de automatización en la plataforma.

Gitlab pages

Es un servicio de hosting estático que ofrece Gitlab sin costo adicional. Permite crear sitios estáticos para nuestros grupos, proyectos o usuarios.

GKE

Elastic Kubernetes Service es un servicio de Kubernetes administrado provisto por Google.

GOPATH

Variable de entorno Linux donde indica el Path de trabajo de Go y en el que se encuentran las carpetas SRC, BIN y PKG.

Goroutines

Son hilos de ejecución de código en Golang que se mandan al procesador y trabajan de forma concurrente.

GraphQL

Es un lenguaje de consultas que se ejecuta del lado del servidor para consultar y manipular datos, fue desarrollado por Facebook en el 2015 y apartir del 2018 ya es mantenido por GraphQL Foundation.

H

Hacking

Se refiere a la explotación de sistemas vulnerables. Comprometer los controles de seguridad para obtener acceso no autorizado o inapropiado a los recursos del sistema.

Herencia

Su objetivo de ser es la reutilización del código. Método por el cual una clase (hija) deriva de otra (padre) para extender su funcionalidad.

Hover

Cuando el cursor del mouse se desplaza sobre un enlace y ese enlace cambia de color se trata del estado "hover".

HTML

lenguaje de marcado para la elaboración de páginas web. HyperText Markup Language (lenguaje de marcas de hipertexto) por sus siglas en ingles.

I

Ingeniería social

Consiste en manipular una o un grupo de personas, a través de técnicas psicológicas y habilidades sociales para cumplir metas específicas.

Ingesta

Tomar la información y alimentar otros sistemas (cloud providers, servicios o herramientas de terceros)

Integración Continua (Continuos Integration)

Es un proceso en el cual se puede construir, probar e incluso implementar aplicaciones cuando se realiza un cambio al código fuente.

Integridad

Propiedad de la información que esté completa y sea precisa.

Inteligencia Artificial

Los sistemas de Inteligencia Artificial son aquellos que tienen la capacidad de entender, razonar y aprender. Es decir, aquellos sistemas que puedan imitar la inteligencia humana.

Interface

En algunos lenguajes de programación es un tipo de “clase” que está enfocada en la reutilización de acciones o métodos entre familias (jerarquía padre-hijo de la herencia). En experiencia de usuario representa el punto de contacto entre usuario y máquina o elemento análogo.

IOS

Sistema operativo de Apple creado para iPhone

Issue

Es parte del sistema de planificación de Gitlab. Permite discutir la implementación de una idea, describir un bug en el sistema u obtener soporte técnico por parte de los mantenedores del código.

J

JAR

Es el archivo ejecutable de una aplicación construida en Java con la versión Standard Edition. Este empaqueta todo el Bytecode que lee la JVM.

JavaScript

JavaScript (abreviado comúnmente JS) es un lenguaje de programación interpretado, dialecto del estándar ECMAScript. Se define como orientado a objetos, es basado en prototipos, imperativo, débilmente tipado y dinámico.

JDK

Son las siglas del Java Development Kit este es el kit de desarrollo de Java que contiene todas las clases y librerías necesarias para que crees tus propios programas en Java.

JRE

Es la Java Runtime Environment, esta contiene a la JVM, algunas clases principales y archivos de soporte.

K

K8s

Abreviación de Kubernetes

Keyword

Se refiere a la palabra clave que atrae al mayor número de seguidores a través de buscadores. Cuando tienes claro cuáles son las keywords que definen tu contenido puedes posicionarlo de manera orgánica o paga lo cual influye en el número de búsquedas y las conversiones.

Kubectl

Command line tool utilizado para gestionar nuestro cluster de Kubernetes.

L

Laravel

Es un framework para el desarrollo rápido de sitios web o APIs usando como lenguaje de programación PHP. Su primera versión estable salio en Junio de 2011.

Layout

Es el lienzo y forma en la que cada elemento está situado dentro de un sitio web.

Lista Negra (Black List)

proceso para identificar y bloquear, E-mail, programas y dominios con IP identificados como maliciosos.

M

Machine Learning

Rama de la inteligencia artificial que se encarga del estudio de métodos automatizados por medio de algoritmos y estadística, para generar modelos matemáticos que permitan resolver tareas de forma tal que sea el software quien descifre por sí mismo como resolver dicho problema.

Manejo de archivos

Existe aplicaciones que permiten que el usuario pueda subir archivos o generar archivos, existen muchas vulnerabilidades que están relacionadas con este tipo de acciones.

Manejo de archivos

Existe aplicaciones que permiten que el usuario pueda subir archivos o generar archivos, existen muchas vulnerabilidades que están relacionadas con este tipo de acciones.

Manejo de sesiones

Es la forma en la que la aplicación identifica a los usuarios cuando se conectan a la aplicación

Máquina Virtual de Java (JVM)

También es conocida en inglés como la Java Virtual Machine, este es el motor que provoca la compatibilidad y el código compartido entre diferentes sistemas operativos, ella lee el bytecode generado por el compilador.

Matemática

Es la ciencia relacionada a razonamiento deductivo que se dedica al estudio de propiedades entre los entes abstractos y sus relaciones; trabaja con números, símbolos, figuras geométricas, otros.

MEAN

Acrónimo de Mongo, ExpressJS, Angular y NodeJS, en un framework usado regularmente para el desarrollo de aplicaciones basadas en Javascript.

Mecánica

Es una rama de la física que estudia y analiza el movimiento y reposo de los cuerpos, y su evolución en el tiempo bajo la acción de fuerzas.

Memoria

En computación, es una sección designada a el almacenamiento y recuperación de datos e información.

Merge request

Es el punto de verificación previo a que cambios al código se integren al branch master. En Gitlab, incluye diversos widgets que nos otorgan información relevante para determinar si aceptar o no los cambios propuestos.

Método

Corresponde a una acción en particular que puede ejecutar el objeto.

Métricas de desempeño

Ayudan a medir el rendimiento de nuestra aplicación para detectar posibles cuellos de botella y optimizar queries.

Métricas de equipo

Ayudan a medir cuál es la capacidad actual de nuestro equipo para poder estimar el trabajo que es posible realizar a través del mismo.

Métricas de equipo

Ayudan a medir cuál es la capacidad actual de nuestro equipo para poder estimar el trabajo que es posible realizar a través del mismo.

Métricas de equipo

Ayudan a medir cuál es la capacidad actual de nuestro equipo para poder estimar el trabajo que es posible realizar a través del mismo.

Métricas de salud

Ayudan a medir la salud de nuestros sistemas para detectar si un componente de hardware está a punto de fallar (falta de memoria, límites en el espacio de almacenamiento o procesamiento).

Miembro

Datos y estructuras que representan una clase.

Milestone

Los milestones permiten agrupar issues para alcanzar un objetivo determinado en un tiempo especificado. Pueden ser utilizados para seguir los sprints en un equipo ágil, o para agrupar los issues que formarán parte de un release.

MVC

Es una arquitectura de software que separa la aplicación en tres partes, la primera es la parte o capa de datos el Modelo, la segunda parte es la capa visual o Vista y la tercera parte es la lógica de la aplicación conocida como Controlador.

MySQL

Es un sistema de gestión de Base de Datos relacional de código abierto más ampliamente utilizado en el mundo.

MySQL

MySQL es un sistema de gestión de bases de datos relacional desarrollado bajo licencia dual: Licencia pública general/Licencia comercial por Oracle Corporation y está considerada como la base de datos de código abierto más popular del mundo.

N

Namespaces

Función del kernel de Linux por el cual se aíslan cursos del sistema operativo

NAT

Network address translation. Forma utilizada para enrutar tráfico sin la necesidad de utilizar demasiadas direcciones de IP.

Navegador

Un navegador es una aplicación de software que es usada para visitar sitios web. Firefox, Internet Explorer, Safari, etc.

Negación de servicio.

Ataque dirigido propagado por red que inhabilita el acceso a recursos o servicios a usuarios autenticados en sistema.

NgModule

Un NgModule identifica los componentes, directivas y conductos que utiliza la aplicación junto con la lista de NgModules externos que la aplicación necesita, como FormsModule El NgModule colabora a organizar la aplicación por bloques cohesivos de funcionalidad.

NodeJS

Es el entorno de ejecución de Javascript en el lado del servidor, está basado en el motor V8 de Chrome para ejecutar el código.

O

Objeto

Es una instancia de la clase, es decir, la forma general del objeto con datos particulares.

Open source

Es un termino que se refiere a la fuente ya sea código, un documento, un diseño, un producto etc. Esta fuente puede ser usada y modificada libremente, es una forma de que proyectos de código crezcan gracias a la comunidad.

Ordenadores Zombie

Computadores o servidores controlados remotamente que envían información a la fuente de origen que la solicita, sin notificar al usuario.

ORM

Object Relational Mapping, permite escribir las estructuras de datos (tablas en la base de datos) en código, esto permite programar consultas a la base de datos sin tener que escribir directamente código SQL (Structured Query Lenguaje)

P

PaaS

Platform as a Service o Plataforma como servicio, es un modelo de negocios de cloud computing en el que se ofrece a los suscriptores del servicio, recursos, herramientas y funcionalidades

especializadas alojados en servidores remotos ubicados en internet.

Paradigma

Metodología o forma de organizar para resolver uno o varios problemas.

PCI DSS

Es un estándar para las organizaciones que manejan información de tarjetas de débito, crédito, prepago, monederos electrónicos, ATM y POS.

Pentesting

Es un método para evaluar la seguridad de los sistemas de información y redes. Por medio de la simulación de ataques con el objetivos de encontrar vulnerabilidades que una persona malintencionada puede utilizar.

Pharming

Captura de información confidencial a través de la re-dirección de trafico a un sitio web falso.

Phishing

Técnica que se enfoca en copiar sitios web con el fin de robar información sensible de los usuarios: Usuario y contraseña, tarjetas de crédito haciendo creer a la persona que esta en la página web real.

PHP

(PHP: Hypertext Preprocessor) Es un lenguaje de programación open source y de propósito general que fue desarrollado originalmente para generar contenido dinámico en un sitio web.

Plugin

Es un subconjunto de código escrito en algún lenguaje de programación que permite extender alguna característica o funcionalidad particular del programa base.

Pod

Unidad de despliegue en K8s. Puede estar compuesto por uno o más contenedores.

Polimorfismo

Muchas formas. Se da en la sobreescritura de un método una vez que la herencia se haya ejecutado, se dice que cambia o se añade comportamiento al método.

Políticas de seguridad

Una serie de sentencias formales (normas) que deben cumplir todas las personas que tengan acceso a cualquier información y tecnología de una organización

Post

En el contexto web y de contenidos en internet, Post se refiere a cada una de las entradas o artículos de un blog o foro.

Procesamiento de lenguaje natural

Aplicación de algoritmos computacionales para determinar propiedades del lenguaje natural de los humanos, de forma que permita a las máquinas identificar lenguaje escrito o hablado por humanos.

Protección del canal de comunicaciones

Hay informacion que no puede viajar en texto claro y requiere protegerse de alguna forma lo mas sencillo es a través de un túnel a través de un protocolo llamado https que cifra la información que se envía.

Protocolo

Conjunto de reglas de comunicación que rigen el intercambio de información entre dos equipos o sistemas conectados entre sí.

Pruebas Unitarias (Unit Testing)

Son pruebas que se realizan a pequeñas partes del código de manera separada para verificar su funcionamiento.

Pull Request

Es la solicitud para incluir los cambios hechos en un rama de un repositorio de Git a la rama master

Python

Lenguaje de programación de alto nivel y de propósito general, caracterizado por la exigencia de uso de la indentación como forma de estructura del código lo que logra una mejor lectura del mismo. Muy usado actualmente para ciencia de datos y machine learning.

Q

Quality Assurance (QA)

Son una serie de procedimientos que permite asegurar la calidad de un producto o pieza de software.

Química

Es la ciencia que estudia la composición, estructura y las propiedades de la materia, como los cambios que esta experimenta durante las reacciones químicas y su relación con la energía.

R

RAM

Memoria de acceso aleatorio, usada para leer y escribir información volátil (que no va a ser almacenada a largo plazo).

Ransomware

Programa que bloquea el acceso total al usuario y pide dinero a cambio para habilitar el acceso.

RBAC

Role Based Access Control. Política de roles para configurar la autorización a los recursos del cluster.

Red neuronal artificial

Modelo utilizando en Inteligencia Artificial, basado en una forma muy simple del cerebro humano. Consiste en pasar datos por distintas capas para compararlas con una salida esperada.

Red neuronal convolucional

Tipo de redes neuronales utilizadas para identificar y analizar imágenes.

Renderizado

Es el proceso por el que pasa el navegador cuando interpreta HTML y CSS y lo convierte en el resultado final visual que ve en su pantalla.

Réplicas

Cantidad de instancias del mismo pod en ejecución

Review apps

Las review apps son ambientes completos que se despliegan de manera automatizada cada vez que se sugiere un cambio en un merge request.

Riesgo

Potencial de que cierta amenaza pueda explotar las vulnerabilidades de un activo o grupo de activos y causar daño a la organización.

Roll-out

Desplegar mediante diferentes estrategias código a producción.

RPG

Sus siglas en inglés vienen de role-playing game. Es un género de los videojuegos conocido como videojuego de rol y que se

desarrollan en un mundo específico donde el jugador tiene la potestad sobre las acciones de el/los personajes.

Ruby

Lenguaje de programación creado en los noventas, de alto nivel y de proposito general, pero usada principalmente en el desarrollo web, caracterizado por su expresividad y gran comunidad

Ruby on Rails

Framework de ruby para realizar desarrollo web, tiene reglas y una serie de convenciones para mejorar la productividad, es muy popular

S

S3

Simple Storage System: Almacenamiento para internet de cualquier tipo de información. Puede usarlo para almacenar y recuperar cualquier cantidad de datos en cualquier momento, desde cualquier lugar de la web. Existen varios tipos de gestores de S3 tanto para Mac y para otros S.O.

Sast: Static Application Security Testing

Es un tipo de prueba automatizada de seguridad que busca encontrar vulnerabilidades en nuestro código al analizar nuestros patrones de escritura de código.

SDK

Software development kit, conjunto de herramientas utilizadas en el desarrollo nativo de aplicaciones Android.

Seguridad informática

Se encarga de proteger los activos de informáticos, entre los que se encuentran la Información, infraestructura y usuarios.

Semántica

En la web, semántica significa elegir las etiquetas HTML correctas para describir el contenido. Por ejemplo, usar etiquetas de tabla para diseñar un sitio no es muy semántico porque esas etiquetas no transmiten ningún significado al contenido que contienen.

Server-Side (Lado del servidor)

Se refiere a un lenguaje que es interpretado por el servidor. Por ejemplo, PHP es interpretado por tu servidor, procesado y luego entregado a ti.

Servicios en la nube

Se llaman servicios en la nube a las diferentes funcionalidades para la gestión y procesamiento de datos o recursos digitales disponibles en la web a través de accesos programables con el apoyo de protocolos de internet como http, https, ssh, sftp, tcpip, udp, entre otros.

Servidor

Cuando una página web es visitada, los datos se envían desde alguna computadora a algún lugar a tu computadora a través del internet. Esa otra computadora es un servidor, configurada especialmente para entregar información a otras computadoras que la soliciten.

Sistema embebido

Es un sistema de computación diseñado específicamente para realizar una o algunas pocas funciones dedicadas. Los sistemas embebidos se diseñan para cubrir necesidades específicas.

Sistema experto

Sistema de inteligencia artificial capaz de utilizar conocimientos de dominio humano especializado para resolver un problema.

Sistema Operativo

El el conjunto de órdenes y programas que controlan los procesos básicos interacción con el hardware.

Software

Conjunto de instrucciones y estructuras lógicas que conforman la base del lenguaje informático. Se trata de los programas y demás recursos que no componen el hardware.

Spam

Correo electrónico con contenido idéntico enviado de forma masiva, también conocido como correo basura.

Spyware

Software organizado por paquetes para realizar seguimiento el cual envía información de los usuarios a la fuente original sin solicitud de permisos a usuarios.

SQL

(Structured Query Language) es el lenguaje basado en el álgebra relacional que es utilizado en la gestión de Bases de Datos, permite entre otras cosas hacer consultas, inserciones y modificaciones de esquemas.

SQL

Structured Query Language es un lenguaje diseñado para dominar y manejar los datos

Stack de protocolo

Es una colección ordenada de protocolos organizados en capas que se intercomunican unos con otros por capas dando como resultado un transporte de información.

Storage

Almacenamiento en la nube es un sistema que permite guardar cualquier cantidad de información de aplicaciones, obedeciendo a una arquitectura creada para ello. Dicha arquitectura diferencia los variados proveedores de este servicio.

T

Tema

Es el conjunto de archivos y documentos que al ser integrados en un sitio web creado con Wordpress, permiten personalizar su apariencia y comportamiento.

Tiempo real / RealTime

Con este término se identifican los comportamientos que suceden de manera dinámica e inmediata como respuesta a una acción realizada por un usuario o proceso automatizado sin retardos o esperas en la interacción.

Tracware

Programa diseñado con el fin de rastrear las acciones que realiza el usuario mientras visita internet, (historial de navegación) con el fin de crear anuncios y publicidad.

U

Unity

Motor gráfico gratuito utilizado para crear videojuegos. Contiene un set de códigos preestablecidos que facilitan el desarrollo y permite crear experiencias para todo tipo de plataforma o consola.

V

Validación de entradas

es muy importante y uno de los mas explotados debido a que todas las aplicaciones requieren un cierto nivel de interacción con los usuarios. “Cualquier entrada de usuario es potencialmente maliciosa”

Variable

Espacio en memoria para almacenar datos, con un identificador simbólico (nombre) y una dirección que esté asociado a dicho espacio.

Videojuego

Es un juego electrónico que permite la interacción de uno o más usuarios con un dispositivo que emite imágenes de video en diversas plataformas.

Virus

Programa que se establece en el disco duro, replicando acciones que sin permiso del usuario afectan el sistema operativo.

Vulnerabilidad

Es un fallo de programación, configuración o diseño. Que permite, de alguna manera, a los atacantes alterar el comportamiento normal de un programa y realizar algo malicioso.

W

Watson

IBM Watson es la plataforma de Inteligencia Artificial de IBM.

Web Application Firewall

Determina la respuesta que debe enviar el servidor de acuerdo con las reglas definidas en el WAF.

Widget

Es un fragmento de código HTML y/o Javascript que se incrusta en sectores particulares de una página web para mostrar funcionalidades adicionales o personalizar el comportamiento del contenido.

WooCommerce

WooCommerce es un plugin avanzado con el que puedes agregar toda la funcionalidad de una tienda en línea a tus sitios web creados con WordPress.

WordPress

Es un sistema de gestión de contenidos, también conocido como CMS, con el que se pueden crear una gran variedad de sitios web, principalmente blogs.

X

Xamarin

Es una plataforma de desarrollo que nos permite construir aplicaciones usando las APIs de los sistemas operativos más importantes del ecosistema móvil. Las apps son codificadas utilizando C# y .NET

Xamarin.forms

Es un framework que se agrega en un proyecto de Xamarin para poder crear interfaces de usuario que aprovechen los lenguajes visuales nativos de cada sistema operativo.

XAML

Es un lenguaje desarrollado por Microsoft que tiene como objetivo describir gráficamente las interfaces de usuario. Los archivos XAML son producidos por herramientas de diseño visual como Microsoft Blend.

Xcode

Entorno de desarrollo para crear aplicaciones de iOS