

CAPACITACIÓN DESARROLLO FULL STACK Y TECNOLOGÍAS HÍBRIDAS

Descripción

A lo largo de este programa vas a adquirir los conocimientos necesarios para un trabajo en Desarrollo Web Full Stack como iniciante; adicional a ello, vas a poder desarrollar aplicaciones y sitios web complejos, creando y configurando aquellas operaciones que ocurren en un servidor con bases de datos (el Back-end), al igual que la experiencia que ven y ocurre del lado de sus usuarios (el Front-end).

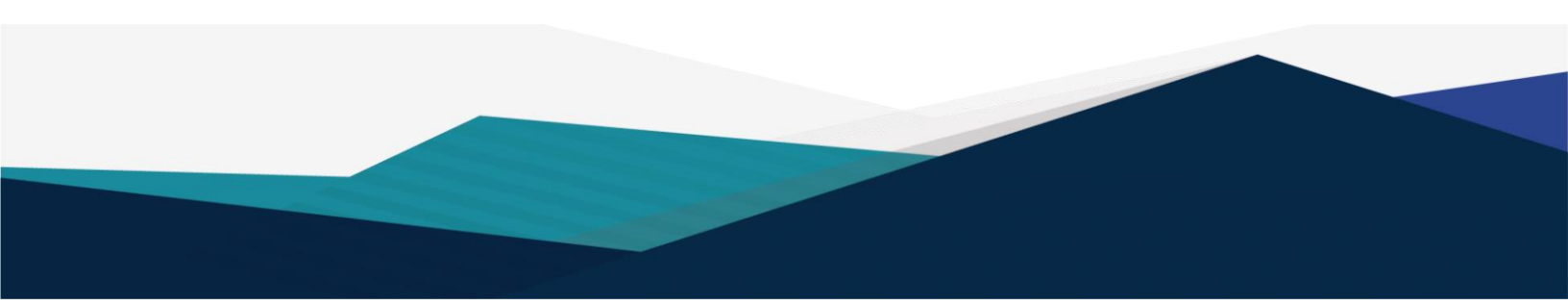
Los conocimientos se adquieren a través de la realización de los cursos conceptuales y talleres prácticos que los acompañan, para obtener las habilidades necesarias para realizarlos.

Este proyecto representa un apoyo concreto de parte del gobierno para asegurarte que tus conocimientos van en línea con el desarrollo y tecnologías más demandadas. Las tecnologías utilizadas en este programa son: HTML5, CSS, JavaScript, Git, Protocolo HTTP, APIs REST, Ruby, Ruby on Rails, PostgreSQL, Ionic 3, Angular 5.

Metodología

Usando el Aprendizaje Basado en Proyectos (**ABP o PBL, Project-based learning**) nos permite aproximarnos a casos reales, basándonos en la necesidad de cambiar el paradigma de los procesos formativos, teniendo claro el porqué, para qué y las necesidades en la vida de cada una de las competencias adquiridas, consolidando un aprendizaje con sentido.

El método consiste en la realización de un proyecto, normalmente de cierta envergadura en un grupo diversificado e integral. El proyecto será analizado previamente por el profesor para asegurarse que el alumno tiene lo necesario para resolverlo, y que en su resolución desarrollara todas las destrezas que se desean. El desarrollo del proyecto iniciará con una problemática básica a resolver. Esta debe basar su solución en el análisis de la problemática, la investigación y el pensamiento crítico, además de la implementación de la



tecnologías y conocimientos adquiridos para generar productos eficientes como resultado del proceso.

Material

- Repositorio con los ejercicios de cada clase
- Material guía de clases
- Ejercicios de cada módulo

Perfil del formador

Perfil Formador Front End

Desarrollador Senior con más de 4 años de experiencia con conocimientos en:

- Tecnologías javaScript, Angularjs y Ionic.
- Conocimientos en consumo de datos mediante API's
- Análisis y Diseño de producto

Perfil Formador Back End

Desarrollador senior con más de 4 años de experiencia con conocimientos en:

- Tecnologías Ruby, Ruby on Rails
- API REST
- Bases de datos relacionales y diseño de esquemas
- Análisis y Diseño de producto

Módulos y temas de la formación

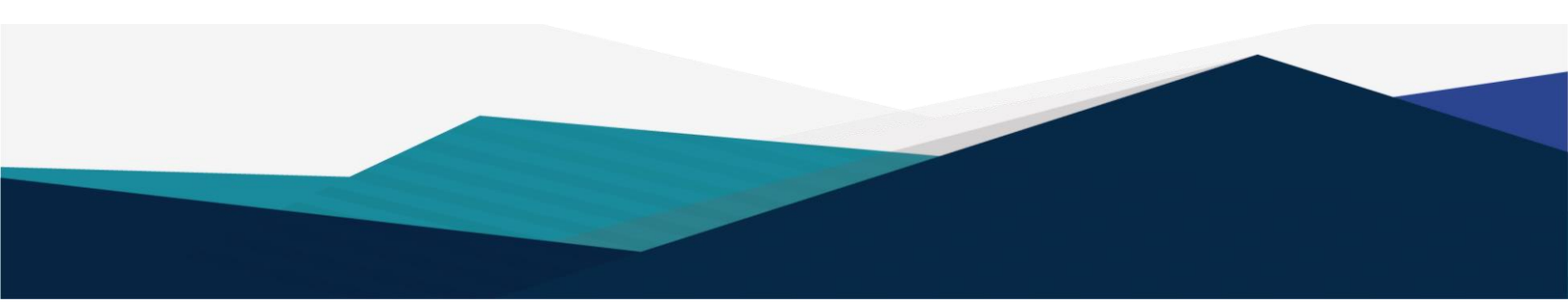
Información General

Tiempo de formación: 100 horas

Modalidad: Presencial

Tiempo de cada clase: 4 horas

Cantidad de clases: 25



Módulo I - Prototipado básico

- Prototipos (Productos Digitales).
 - ◆ Niveles de fidelidad de un prototipo.
 - ◆ Wireframes en Papel
 - ◆ Wireframes en Digital
 - ◆ Diseño Modular
 - ◆ Mockups
 - ◆ Prototipo interactivo

- HTML & CSS3
 - ◆ Introducción HTML
 - ◆ Anatomía HTML
 - ◆ Estructura HTML
 - ◆ HTML + CSS
 - ◆ Introducción CSS
 - ◆ Inline Styles
 - ◆ selectores
 - ◆ .css files
 - ◆ Elementos anidados
 - ◆ Box model
 - ◆ Colores, Tipografía
 - ◆ CSS Grids
 - ◆ Diseño adaptativo

- Diseño de proyecto
 - ◆ Presentación de Retos
 - ◆ Buscando Soluciones
 - ◆ Creatividad y teoría de pensamiento lateral
 - ◆ MVP

Módulo II - Programación Web

- Conceptos básicos
 - ◆ Terminal o CLI
 - ◆ Servidores
 - ◆ Dominios
 - ◆ Protocolo http y https
- Entorno de desarrollo
 - ◆ Instalar nodeJs
- Git & Github
 - ◆ Crear cuenta Github
 - ◆ init, create, clone, commit, pull, push
- JavaScript
 - ◆ Tipo de datos
 - ◆ Variables, Constantes
 - ◆ Operaciones matemáticas
 - ◆ Estructuras de Control
 - ◆ if/else
 - ◆ Operadores lógicos
 - ◆ Switch
 - ◆ Funciones
 - ◆ Scope
 - ◆ Arrays
 - ◆ Ciclos e Iteradores
 - ◆ Objetos y Clases
 - ◆ Módulos
 - ◆ Procesos Asíncronos y Síncronos
 - ◆ Peticiones HTTP
- Bases de Datos
 - ◆ Instalar Postgres
 - ◆ Propósito general de BD
 - ◆ Modelo Entidad Relación
 - ◆ SQL
 - ◆ Estructura de un Query
 - ◆ Insertar Registros
 - ◆ Joins
 - ◆ Usuarios
 - ◆ Roles
- Ruby
 - ◆ Instalar Ruby
 - ◆ Tipos de datos
 - ◆ Variables
 - ◆ Operaciones Matemáticas
 - ◆ Estructuras de control
 - ◆ Todos son Objetos
 - ◆ Estructuras de control
 - ◆ Array & Hashes
 - ◆ Ciclo y iteradores
 - ◆ Blocks, Procs y Lambdas
 - ◆ Programación Orientada a Objetos
- Ruby on Rails
 - ◆ Instalar Rails
 - ◆ Bundle & gem
 - ◆ Estructura de Archivos
 - ◆ Entornos de Desarrollo
 - ◆ MVC
 - ◆ Controladores, rutas y vistas
 - ◆ Scaffold
 - ◆ Métodos de petición HTTP
 - ◆ Conexión DB
 - ◆ Comando Rake
 - ◆ Configurar conexión
 - ◆ Modelos
 - ◆ Migraciones
 - ◆ Asociaciones
- API
 - ◆ Json
 - ◆ Autenticación vía Token

Módulo III - Prototipo avanzados

- Autenticación y autorización API
 - ◆ Refrescar Token
 - ◆ Guardar Session
- Notificaciones
 - ◆ Conexión a servidor de Emails
 - ◆ Plantilla de emails
- Conexiones SSH
 - ◆ Administrar llaves públicas
 - ◆ Comandos básicos Linux
 - ◆ Instalar Nginx
- Capistrano
 - ◆ Configuración
 - ◆ Deploy

Módulo IV -Aplicaciones híbridas

- TypeScript
 - ◆ Sintaxis
 - ◆ Angular 5
 - ◆ Ionic 3
 - ◆ Aplicaciones Híbridas
 - ◆ Diferencia Nativo/Híbrido
 - ◆ Crear Proyecto Ionic
 - ◆ Estructura de carpetas
- Componentes
 - ◆ Botones
 - ◆ Listas
 - ◆ Alertas
 - ◆ Tabs
 - ◆ Slides
 - ◆ Navegación
 - ◆ Listas
 - ◆ Menús laterales
- Servicios y Providers
- Autenticación
 - ◆ Conexión a Servidor
 - ◆ LocalStorage
 - ◆ Sessions
- Notificaciones
 - ◆ Notificaciones push
 - ◆ Notificaciones toast
- Recursos del teléfono
 - ◆ Cámara
 - ◆ GPS
 - ◆ Agenda
- Desarrollo de Proyecto
 - ◆ Correr aplicaciones en android
 - ◆ Correr aplicaciones en un emulador android
 - ◆ Correr aplicaciones en dispositivo IOS
 - ◆ Correr aplicaciones en un emulador IOS

Evaluación

El contenido de la evaluación será compuesto por los siguientes ítems:

- Tecnologías web básicas
- Lenguajes de programación
- Funcionamiento de plataformas
- Programación de aplicaciones híbridas