

Contenido

Diseño Curricular

Cuatrimestre	Curso Programación (6hs)	Formación integral (3hs)
Primer año		
Primer Cuatrimestre	Programación de Sitios Web	Proyecto Grupal Intermedio
Segundo Cuatrimestre	Programación Web Front End	Proyecto Grupal Intermedio
Segundo año		
Tercer Cuatrimestre	Programación Back End	Proyecto Grupal Final
Cuarto Cuatrimestre	Desarrollo de aplicaciones web	

Detalle Académico

Curso Inicial Nivelatorio

Esquema de funcionamiento de una web en Internet. Backend, front end. Estructura general del curso y objetivos. Interpretación de texto y redacción. Aprendizaje y resolución de problemas con computadora, utilizando sistemas tipo Scratch. Estos sistemas buscan proveer una interfaz visual para introducir conceptos de programación, y ha demostrado gran efectividad a cualquier edad, siendo el tipo de enseñanza usada en la educación primaria y secundaria para introducir los mismos.

Se utilizará un sistema web con el contenido ya cargado, en las clases sincrónicas se hará puesta en común con los alumnos, se resolverán las dudas surgidas, se relacionarán los mismos con lo que verán a lo largo de la carrera y se dejará tiempo para resolver los mismos estando el docente disponible para asistirlos.

Se hará un encuentro adicional con aquellos que no hayan alcanzado las metas mínimas para analizar si ameritan cursarla igualmente con un plan de compensación de saberes faltantes y analizar puntos de mejoras para futuras ediciones.

Programación de Sitios Web

Concepto de estructura de datos y de control. Declaración, inicialización, actualización y entrada/salida de datos. Estructuras de control básicas: secuencia, selección e iteración. Modularización de un problema. Funciones. Arreglos y manipulaciones básicas. Testing unitario de funciones. Nociones de ordenamiento.

Sitios web. HTML. CSS. Implementación de layouts simples. Javascript. Manipulación de la página desde Javascript. Eventos. Interacción con el usuario. Browser testing automation (Cypress).

Proyecto Grupal Intermedio

Comunicación de problemas técnicos. Búsqueda de información y curación de respuestas. Uso de inglés básico para búsqueda de información. Sistemas de versionado. Git básico. Logica, algebra de bool. Capacidades personales para el aprendizaje de la programación y técnicas de estudio. Comprensión de Problemas. División de un problema y mapeo de las partes a posibles soluciones. Binario. Conjuntos. Inglés: vocabulario técnico (palabras) y uso para la búsqueda de información.

Desarrollo de un proyecto web simple, integrando todo lo realizado. Perspectivas iniciales de gestión de proyectos: división de tareas, control de calidad. No se utilizarán frameworks en esta etapa.

Ejemplo de proyecto: Una página web con alguna parte interactiva utilizando Javascript.

Programación Front end

Gestores de dependencias. Typescript. Concepto de programación orientada a objetos. Clases, instancias, atributos y métodos, palabra reservada "this". JSON vs objetos con clase. Encapsulamiento y composición. Polimorfismo, herencia e interfaces. Herencia por composición y especialización. Patrones de diseño. Manejo de errores/excepciones. Inyección de dependencias. Testing unitario usando mocks.

Desarrollo con un Framework Front End. Uso de herramientas para el desarrollo web de proyectos grandes (linters, lenguajes de estilos avanzados, etc). División en componentes. Tipos de componentes (presentacionales, lógicos, etc). Testing de componentes. JSON. Conexió

Proyecto Grupal Intermedio

Manejo de la terminal. Trabajo en equipo. Uso de GIT colaborativo. Ciclo de vida de un proyecto. Modelado del mundo real.

Estructuración de datos. Planillas de cálculo: filtros, fórmulas, funciones, formatos condicionales, validación, cruzado de datos, análisis con gráficos y tablas pivot. Medidas estadísticas y su significado. Tendencias y ajuste de funciones.

Concepto de alcance del sistema. Organización de proyectos. Seguimiento. Sistemas de seguimiento de tareas. Análisis de sistemas. Técnicas. User story. Trabajo en equipo. User story mapping y división del trabajo. Scrum. Refactorización, manejo de la calidad. Deuda técnica. Protocolo HTTP. Redes. Arquitectura de internet. Protocolos TCP/IP. IP privada vs pública. SSL.

Ejemplo de proyecto: Una aplicación web (sólo front end) con gran dinamismo. Utilizando APIs dadas por la materia o con alguna generada por herramientas web.

Programación Back End

Servidor HTTP de archivos estáticos. JSON. GET de un archivo JSON. Aplicación backend REST (ruteo). Patrón MVC. Implementación de Altas, Bajas y Modificaciones (ABM) en un servicio REST.

Gestores de base de datos. Instalación. Modelos de datos. Bases de datos NoSQL. MongoDB. SQL, manipulación de metadatos y de datos (DDL y DML). Consultas multitabla. Consultas con agrupamiento.

Conexión del backend a una base de datos. ABM de una entidad. Mapeo Objeto-Relacional. TypeORM. Autenticación.

Desarrollo de Aplicaciones web

Contenedores, Docker. Nociones de despliegue. Conexiones a terminales remotas (SSH). Sincronismo y atomicidad. ACID vs BASE; SQL vs NoSQL; diferencias en el modelado. MVC y otros patrones arquitectónicos frecuentes, 3 capas, 3 capas de DDD y otras arquitecturas usadas. Arquitecturas distribuidas, servicios/microservicios, cola de eventos/notificaciones, ventajas y desventajas. Ejemplos en otras tecnologías. Estructuras de datos. Complejidad. Preparación de CV, perfil de LinkedIn. Preparación para entrevistas laborales. Portfolio. Servidor SSL local. Otros protocolos de internet. DNS, FTP, etc.

Proyecto Grupal Final

Ambientes de desarrollo. Configuración de ambientes. Niveles clásicos de ambientes. Concepto de bounded context, relación de arquitectura empresarial con arquitecturas de sistemas.

Proyecto a desarrollar en grupo aplicando todas las tecnologías y conocimientos vistos. Inglés: primero hacer las user story en inglés, después hacer los reportes de tareas en las reuniones en inglés. Comunicación efectiva. Trabajo en equipo.

Ejemplo de proyecto: Una aplicación web (full stack) con gran dinamismo, persistencia de datos y testing automático.