

UNIVERSIDAD NACIONAL DE SAN AGUSTÍN DE AREQUIPA



VICERRECTORADO ACADÉMICO

FACULTAD DE INGENIERIA DE PRODUCCION Y SERVICIOS
DEPARTAMENTO ACADÉMICO DE INGENIERIA DE SISTEMAS E INFORMATICA

SÍLABO 2022 - A

ASIGNATURA: PROGRAMACION WEB 2

1. INFORMACIÓN ACADÉMICA

Periodo académico:	2022 - A	
Escuela Profesional:	INGENIERÍA DE SISTEMAS	
Código de la asignatura:	1702122	
Nombre de la asignatura:	PROGRAMACION WEB 2	
Semestre:	III (tercero)	
Duración:	17 semanas	
Número de horas (Semestral)	Teóricas:	2.0
	Prácticas:	0.0
	Seminarios:	0.0
	Laboratorio:	4.0
	Teórico-prácticas:	0.0
Número de créditos:	4	
Prerrequisitos:	PROGRAMACION WEB 1 (1701212)	

2. INFORMACIÓN DEL DOCENTE, INSTRUCTOR, COORDINADOR

DOCENTE	GRADO ACADÉMICO	DPTO. ACADÉMICO	HORAS	HORARIO
CORRALES DELGADO, CARLO		INGENIERIA DE SISTEMAS E INFORMATICA	0	Jue: 07:00-08:40
ESCOBEDO QUISPE, RICHART		INGENIERIA DE SISTEMAS E INFORMATICA	0	Jue: 07:00-08:40
CORRALES DELGADO, CARLO		INGENIERIA DE SISTEMAS E INFORMATICA	0	Mar: 17:40-19:20
ESCOBEDO QUISPE, RICHART		INGENIERIA DE SISTEMAS E INFORMATICA	0	Mar: 17:40-19:20
ESCOBEDO QUISPE, RICHART		INGENIERIA DE SISTEMAS E INFORMATICA	0	Jue: 14:00-15:40

3. INFORMACIÓN ESPECIFICA DEL CURSO (FUNDAMENTACIÓN, JUSTIFICACIÓN)

El curso busca impartir la teoría y práctica sobre la Programación Web basada en Frameworks bajo la filosofía de trabajo en el BackEnd y en el FrontEnd. La práctica de las herramientas modernas de programación web, enmarcado dentro de un proceso de desarrollo con frameworks bajo el modelo Vista

Controlador. En este caso utilizando el Framework Django para el BackEnd y el Framework Angular para el FrontEnd.

4. COMPETENCIAS/OBJETIVOS DE LA ASIGNATURA

C. Diseña responsablemente sistemas, componentes o procesos para satisfacer necesidades dentro de restricciones realistas: económicas, medio ambientales, sociales, políticas, éticas, de salud, de seguridad, manufacturación y sostenibilidad.

M. Construye responsablemente soluciones siguiendo un proceso adecuado llevando a cabo las pruebas ajustada a los recursos disponibles del cliente.

P. Aplica de forma flexible técnicas, métodos, principios, normas, estándares y herramientas de ingeniería necesarias para la construcción de software e implementación de sistemas de información.

5. CONTENIDO TEMATICO

PRIMERA UNIDAD

Capítulo I: Introducción

Tema 01: Introducción, reglas del curso, El Protocolo HTTP: clientes, servidores, URLs, HTML, CSS

Tema 02: CGIs, ISPs, Bridge, GateWay, Protocolo IP, Puertos

Tema 03: DNS, Sistema control de versiones, JavaScript, EmacsScript

Tema 04: Ajax y NodeJS

Capítulo II: El Lenguaje Python

Tema 05: Ambiente de trabajo: editor, git, interprete, sintaxis, variables, tipos de datos, operadores, listas, tuplas, conjuntos, diccionarios, condicionales, ciclos

Tema 06: funciones, lambda, arrays., Clases y objetos, herencia, iteradores, ámbito, módulos, expresiones regulares, excepciones, manejo de archivos

Tema 07: 1er Examen

SEGUNDA UNIDAD

Capítulo III: Programación en el backend: Django

Tema 08: Instalación, el ambiente virtual, Modelo, vista, controlador, Crea un proyecto Django en blanco, Migraciones, Componentes y aplicaciones

Tema 09: El sitio de administración: creación de usuarios, Creación de MODELOS y migraciones: la base de datos, El Python Shell, Tipos de datos del modelo, Actualización de tipos de datos en el modelo

Tema 10: Cambiando el modelo y sus tipos de datos, Creación de VISTAS, Enrutamiento de URLs, PLANTILLAS html, Comunicación entre vistas y plantillas: el contexto

Capítulo IV: Programación en el backend: Django parte II

Tema 11: Ciclos, condicionales y filtros en plantillas, Comuniación entre el modelo, la vista y las plantillas, Modularización de Apps, Formularios desde el modelo, Renderización de formularios en las plantillas

Tema 12: Repuestas para GET y POST, seguridad, Widgets para formularios, Métodos de validación de formularios, Valores iniciales para formularios, Enrutamiento dinámico

de URL, Eliminar y confirmar objetos del modelo, CREATE, READ, UPDATE, DELETE (CRUD) con vistas como función, CRUD con vistas como clases

Tema 13: 2do Examen

TERCERA UNIDAD

Capítulo V: Programación en el frontend: Angular JS

Tema 14: Introducción, Expresiones, Módulos

Tema 15: Directivas, Modelo, Enlace de datos

Tema 16: Ámbitos, Filtros, Servicios

Capítulo VI: Combinación BackEnd con FrontEnd: Django y Angular JS

Tema 17: Http, Tablas, Select, SQL, DOM, Eventos

Tema 18: Formularios, Validación, API, W3.CSS, Includes, Bootstrap

Tema 19: Djangular

Tema 20: 3er Examen

6. PROGRAMACIÓN DE ACTIVIDADES DE INVESTIG. FORMATIVA Y RESPONSABILIDAD SOCIAL

6.1. Métodos

Se utilizara el método del juego de roles, indagación, la tutoría y la enseñanza por descubrimiento.

El método de juego de roles se basará en actividades lúdicas para comprender comunicaciones entre los componentes de una aplicación web.

El método de indagación se utilizara en la solución de las prácticas calificadas de acuerdo a temáticas.

La tutoría se realizará a los grupos de alumnos involucrados en la propuesta de un problema, posibles soluciones y compartir las lecciones aprendidas con el resto del salón.

El método de descubrimiento se utilizará cuando se estudie un framework MVC.

6.2. Medios

Material bibliográfico, Aula virtual Moodle, Videos, Lecturas seleccionadas, Hojas de ejercicios.

Máquina Servidor y máquina Cliente.

6.3. Formas de organización

Componente académico: la clase magistral con participación activa de los alumnos.

Componente práctico: las prácticas calificadas y laboratorios.

Componente investigativo: las actividades de evaluación de prácticas calificadas de acuerdo a las recomendaciones en computación.

Habilidades blandas: exposiciones de código, comunicación en grupo, intervenciones en clase, preguntas en clase al profesor.

6.4. Programación de actividades de investigación formativa y responsabilidad social

Los alumnos investigarán las tecnologías vistas en el curso con mayor profundidad y presentarán un trabajo final que resumirá todo lo aprendido.

Los alumnos no realizarán actividades de alcance social presencial, aunque sus proyectos finales podrían tener algún impacto.

7. CRONOGRAMA ACADÉMICO

SEMANA	TEMA	DOCENTE	%	ACUM.
--------	------	---------	---	-------

1	Introducción, reglas del curso, El Protocolo HTTP: clientes, servidores, URLs, HTML, CSS	C. Corrales	5	5.00
1	CGIs, ISPs, Bridge, GateWay, Protocolo IP, Puertos	C. Corrales	5	10.00
1	DNS, Sistema control de versiones, JavaScript, EmacScript	C. Corrales	5	15.00
1	Ajax y NodeJS	C. Corrales	5	20.00
1	Ambiente de trabajo: editor, git, interprete, sintaxis, variables, tipos de datos, operadores, listas, tuplas, conjuntos, diccionarios, condicionales, ciclos	C. Corrales	5	25.00
1	funciones, lambda, arrays., Clases y objetos, herencia, iteradores, ámbito, módulos, expresiones regulares, excepciones, manejo de archivos	C. Corrales	5	30.00
1	1er Examen	C. Corrales	3	33.00
2	Instalación, el ambiente virtual, Modelo, vista, controlador, Crea un proyecto Django en blanco, Migraciones, Componentes y aplicaciones	C. Corrales	6	39.00
1	El sitio de administración: creación de usuarios, Creación de MODELOS y migraciones: la base de datos, El Python Shell, Tipos de datos del modelo, Actualización de tipos de datos en el modelo	C. Corrales	6	45.00
1	Cambiando el modelo y sus tipos de datos, Creación de VISTAS, Enrutamiento de URLs, PLANTILLAS html, Comunicación entre vistas y plantillas: el contexto	C. Corrales	6	51.00
1	Ciclos, condicionales y filtros en plantillas, Comunicación entre el modelo, la vista y las plantillas, Modularización de Apps, Formularios desde el modelo, Renderización de formularios en las plantillas	C. Corrales	6	57.00
1	Repuestas para GET y POST, seguridad, Widgets para formularios, Métodos de validación de formularios, Valores iniciales para formularios, Enrutamiento dinámico de URL, Eliminar y confirmar objetos del modelo, CREATE, READ, UPDATE, DELETE (CRUD) con vistas como función, CRUD con vistas como clases	C. Corrales	6	63.00
1	2do Examen	C. Corrales	3	66.00
1	Introducción, Expresiones, Módulos	C. Corrales	5	71.00
1	Directivas, Modelo, Enlace de datos	C. Corrales	5	76.00
1	Ámbitos, Filtros, Servicios	C. Corrales	5	81.00
1	Http, Tablas, Select, SQL, DOM, Eventos	C. Corrales	5	86.00
1	Formularios, Validación, API, W3.CSS, Includes, Bootstrap	C. Corrales	5	91.00
1	Djangular	C. Corrales	5	96.00
1	3er Examen	C. Corrales	4	100.00

8. ESTRATEGIAS DE EVALUACIÓN

8.1. Evaluación del aprendizaje

Evaluación Teórico-Práctica: Exámenes escritos, Prácticas calificadas y Proyecto final.
Evaluación Continua: Se utilizarán rúbricas para analizar el desempeño de los alumnos.

Evaluación Periódica: Se comprobará el nivel de comprensión a través de técnicas interrogativas en las pruebas escritas.

8.2. Cronograma de evaluación

EVALUACIÓN	FECHA DE EVALUACIÓN	EXAMEN TEORÍA	EVAL. CONTINUA	TOTAL (%)
Primera Evaluación Parcial	17-05-2022	10%	23%	33%
Segunda Evaluación Parcial	28-06-2022	10%	23%	33%
Tercera Evaluación Parcial	09-08-2022	10%	24%	34%
TOTAL				100%

9. REQUISITOS DE APROBACIÓN DE LA ASIGNATURA

Los exámenes se tomarán con apuntes.

Los alumnos tendrán la oportunidad de rezagar un examen parcial con un plazo de 72 horas y por causas debidamente justificadas y autorizadas por la dirección de la escuela.

Posterior a la aplicación de una prueba se realizan las siguientes actividades: solución de las preguntas del examen; acceso de la prueba por parte del estudiante; recalificación cuando es pertinente; publicación de los resultados usando software personalizado, después de todas estas actividades la nota es inmodificable.

Las calificaciones se registran en el sistema de la universidad según cronograma.

Los exámenes son acumulativos.

La honestidad será un factor determinante en la evaluación: Los alumnos que tengan actitudes deshonestas en alguna de sus tareas, trabajos o exámenes tendrán nota 0.

Acto deshonesto: Copiar la solución de otro. Esto incluye mirar el código del compañero o buscar códigos en Internet

Acto deshonesto: Compartir código fuente: Copiar, cambiar de nombre a las variables, mostrar el código a un compañero, descargar el código de Internet, explicar la solución a un compañero. Tener cuidado de no dejar copias de las tareas en lugares públicos.

Acto deshonesto: Consultoría. Recibir ayuda en la solución de la tarea, esta puede ser por un compañero de años superiores, por foros de discusión en Internet, etc.

Acto deshonesto: Realizar los trabajos individuales en grupo. Las tareas pueden tener soluciones diversas, si estas son individuales no deben reunirse para hacerlas.

Acto deshonesto: Realizar las tareas grupales de manera individual. Que sólo un compañero haga toda la tarea del grupo, que cada integrante del grupo haga una parte de la tarea, pero que no tenga idea de las demás partes. Las tareas en grupo deben ser hechas en grupo, por lo que se requiere coordinación, no sólo en la distribución del trabajo, sino en la solución de los problemas que se puedan presentar. El grupo debe trabajar como un equipo.

Acto honesto: Explicar lo que se pide en la tarea. Se puede pedir ayuda al profesor o los compañeros para entender lo que se pide en la tarea, pero siendo cuidadosos de no explicar la solución, sólo el enunciado de lo que se pide.

Acto honesto: Explicar los temas o conceptos (estudiar en grupo). Si algún tema o concepto no se entiende, fuera del horario de clase, se puede pedir al profesor o algún compañero ayuda.

Acto honesto: Usar apuntes. Se pueden apuntes durante las intervenciones en clase y en los laboratorios, estos apuntes podrán ayudar a recordar comandos, códigos, etc.

10. BIBLIOGRAFIA: AUTOR, TÍTULO, AÑO, EDITORIAL

10.1. Bibliografía básica obligatoria

Angularjs tutorial. <https://www.w3schools.com/angular/default.asp>, 2022. Accessed: 19-04-2022.
Javascript tutorial. <https://www.w3schools.com/js/default.asp>, 2022. Accessed: 19-04-2022.
Python tutorial. <https://www.w3schools.com/python/default.asp>, 2022. Accessed: 19-04-2022.
Loiane Groner. Learning JavaScript Data Structures and Algorithms: Write complex and powerful JavaScript code using the latest ECMAScript. Packt Publishing Ltd, 2018.
Greg Lim. Beginning Node.js, Express & MongoDB Development. Amazon, 2019.
William S Vincent. Django for Beginners: Build websites with Python and Django. WelcomeToCode, 2020.

10.2. Bibliografía de consulta

Documentación de django. <https://docs.djangoproject.com/en/4.0/>. Accessed: 19-04-2022.
Stackoverflow. <https://stackoverflow.com/questions>. Accessed: 19-04-2022.
Css tutorial. <https://www.w3schools.com/css/default.asp>, 2022. Accessed: 19-04-2022.
Html tutorial. <https://www.w3schools.com/html/default.asp>, 2022. Accessed: 19-04-2022.
Sql tutorial. <https://www.w3schools.com/sql/default.asp>, 2022. Accessed: 19-04-2022.
David Flanagan. JavaScript: The Definitive Guide: Master the World's Most-Used Programming Language 7th Edition. O'Reilly Media, 2020.
Marijn Haverbeke. Eloquent JavaScript: a modern introduction to programming. No Starch Press, 2018.
Antonio Melé. Django 3 By Example: Build powerful and reliable Python web applications from scratch. Packt Publishing Ltd, 2020.

Arequipa, 21 de Abril del 2022

CORRALES DELGADO, CARLO

ESCOBEDO QUISPE, RICHART