

IMPLEMENTACIÓN DE UN SISTEMA DE GESTIÓN UNIVERSITARIA UTILIZANDO PYTHON Y DJANGO

CORPORACIÓN UNIVERSITARIA REMINGTON

Facultad de Ingeniería

Ingeniería de Sistemas

ROBINSON SEBASTIÁN SOLARTE CORAL

CRISTHIAN ARMANDO CALPA ARELLANO

Opción de Trabajo de grado Diplomado.

Ipiales, 13 de julio de 2024

IMPLEMENTACIÓN DE UN SISTEMA DE GESTIÓN UNIVERSITARIA UTILIZANDO PYTHON Y DJANGO

ROBINSON SEBASTIÁN SOLARTE CORAL

CRISTHIAN ARMANDO CALPA ARELLANO

Trabajo de grado presentado como requisito para optar al título de Ingeniero de Sistemas.

CORPORACIÓN UNIVERSITARIA REMINGTON

Facultad de Ingeniería

Ingeniería de Sistemas

Tutor de Trabajo de Grado:

Mg. DANIEL FERNANDO ARTEAGA FAJARDO

Ipiales, 13 de julio de 2024

Agradecimientos

De Sebastián Solarte

Quiero expresar mi más sincero agradecimiento a la Universidad de Remington, que se ha convertido en mi segundo hogar y ha sido el lugar donde he crecido tanto académica como personalmente; A mi madre Olga Coral, a mi padre Orlando Solarte, a mi hermana Andrea Coral y a Caroli Mesias por apoyarme y por compartir conmigo este importante capítulo de mi vida.

De Cristhian Arellano

Quiero agradecerle a la Universidad de Remington, por su formación de calidad y por creer en mí, agradezco también mi madre Roció, a mi padre Carlos, mis hermanos Dani y Angie y a mi mujer Karen, gracias a ellos por su apoyo y por ser fuente de mi compromiso académico, este este logro es por ustedes.

TABLA DE CONTENIDO

1. Resumen	Pág. 5
2. Introducción	Pág. 7
3. Marco Referencial	Pág. 8
3.1 Marco Teórico – Conceptual	
3.1.1. Desarrollo web.	
3.1.2. Python.	
3.1.3. Django	
4. Desarrollo de la temática	Pág. 10
4.1 Capítulo 1: Elaboración del código para la aplicación Web.	Pág. 10
4.2 Capítulo 2: Visualización de la interfaz.	Pág. 14
5. CONCLUSIONES	Pág. 16

RESUMEN

El desarrollo de una aplicación web diseñada para la gestión de una universidad puede convertirse en una herramienta importante para dar a conocer información academia, proporcionándole al usuario acceso a la información y al propietario el control sobre la interfaz.

Se utilizo Python y el framework Django facilitando muchas tareas del desarrollo web, brindando un desarrollo ágil de sistemas de información, adicional, el uso de VS Code como intérprete de Python y recursos adicionales en línea para el diseño del chat bot de gran valor en entornos educativos para aclarar dudas e inquietudes.

Palabras clave: Python, Django, SQLite3, CRUD, Bootstrap, JavaScript, Desarrollo web.

Pregunta orientadora de la búsqueda

¿De qué manera el uso de Python y Django facilitan el desarrollo y la implementación de una aplicación web segura para la gestión académica universitaria?

2. INTRODUCCIÓN

Desarrollo de una Aplicación Web.

Se describe el diseño de una aplicación web desarrollada para la gestión de una universidad, la creación de la aplicación web permite al propietario modificar la interfaz de la página para el usuario, permitiéndole adicionar carreras nuevas, así como, los créditos y los cursos equivalentes a cada carrera.

La aplicación es implementada utilizando Python y el Framework Django. “Python es un lenguaje de programación de alto nivel, de una sintaxis sencilla” (Shukla y Parmar, 2010); Django es un framework que simplifica muchas de las tareas en el desarrollo de aplicaciones web, como el manejo de bases de datos, la gestión de URLs, la autenticación de usuarios y el sistema de plantillas, estas funcionalidades integradas aceleran el proceso de desarrollo y garantiza la seguridad y la escalabilidad de la aplicación. Django en Python permite el desarrollo rápido de sistemas de información web. (Vidal, Sanches, Serrano y Rubio, 2021).

En la elaboración de la Interfaz se utiliza Bootstrap, así como, se incorpora un chat Bot que busca interactuar de forma efectiva con los usuarios, este será tomado como un valor agregado a la investigación. La necesidad de compartir la información y responder las dudas hacen de los chatbot una herramienta importante en los entornos educativos, puesto que para los estudiantes es útil resolver las dudas de forma rápida y para las personas encargadas del aprendizaje les sirve para monitorear la información y utilizarla como recurso para el apoyo al aprendizaje (Arias, Palacios y Villegas, 2020)

3. MARCO REFERENCIAL

3.1 MARCO TEÓRICO - CONCEPTUAL

Para la elaboración de la Aplicación Web se utilizó un intérprete de Python (VS code), librerías de Django y el framework CSS Bootstrap la visualización de la aplicación web, así como también se buscaron recursos adicionales en línea para el diseño de un asistente virtual (Chatbot) y repositorios de GitHub.

3.1.1. Desarrollo web.

Los avances tecnológicos recientes han transformado la manera en cómo vivimos y percibimos el mundo por ende los programadores debe desarrollar la creatividad para dotar a sus usuarios de la información necesaria. Para Gonzales y Llanes (2024), En la actualidad, el desarrollo web es uno de los sectores tecnológicos que tienen mayor impacto en la sociedad, esto, por la diversidad de información que a través de él se puede transmitir, siéndonos de gran utilidad para nosotros debido a que con la tecnología podemos acceder a productos, servicios, desarrollos, entre otros.

A nivel educativo el desarrollo web es de suma importancia, nos ayuda a promocionar entidades educativas y hacer el conocimiento más asequible a las personas. Es así como se vuelve importante seguir descubriendo diferentes opciones y adaptar el uso de estas herramientas a las necesidades educativas específicas de los estudiantes y de su plan de estudios (Moran, 2024); No solo para mejorar la calidad educativa, sino también, para que la oportunidad de acercarse a la educación superior sea de mayor acceso.

3.1.2. Python

Gonzales (2011) menciona que "Python es un lenguaje de programación que todo el mundo debería conocer. Su sintaxis es simple, clara y sencilla; el tipado dinámico, la gestión automática de memoria, la gran cantidad de librerías disponibles y la potencia del lenguaje, entre otros, hacen que desarrollar una aplicación en Python sea sencillo, muy rápido y, lo que es más importante, divertido. Además, la comunidad activa y la abundancia de recursos de aprendizaje hacen que sea accesible tanto para principiantes como para desarrolladores experimentados."

Python se puede considerarse como un lenguaje semi-interpretado que a diferencia de C (Lenguaje de Programación), Python no se ejecuta directamente en la máquina destino, sino que es ejecutado por un software intermedio o intérprete (Rodo, 2023).

3.1.3. Django

Este es utilizado para facilitar la creación de aplicaciones web de una forma más eficiente.

Para Condori (2012) Django posee muchas ventajas por ser framework, este promueve el desarrollo rápido, lo que quiere decir que se construyen aplicaciones en cuestión de días que pueden reducirse a horas.

Django es un framework de código abierto que está orientado en el lenguaje de programación Python, este tiene diversas herramientas para facilitar el registro de datos, la administración y desarrollo de un sitio web (Pesantes y Vivanco, 2018)

4. DESARROLLO DE LA TEMÁTICA

4.1 CAPITULO 1

Elaboración del código para la aplicación Web

Para comenzar con la Programación de la aplicación web es importante la utilización de Python y Django, teniendo en cuenta que se debe instalar VS code (Interprete) y para la adecuación de la vista se utiliza Bootstrap.

En cmd se creó la Carpeta raíz del proyecto con el siguiente comando:

```
C:\Users\usuario\Desktop>Python -m Django startproject universidad_s
```

La carpeta creada será importada a VS code para iniciar con la configuración y la programación de la aplicación web; dentro de la carpeta madre se crean subcarpetas, donde se albergan varios archivos siendo los más utilizados: models.py, Views.py, Admin.py

Dentro de models se crea las 4 entidades: registro de carreras, registro de estudiantes, registro de curso, matrícula del estudiante y los cursos, (*Ver Figura #1*).

Figura #1: Creación de las 4 Entidades

```
from django.db import models

# Create your models here.

Comment Code
class Carrera(models.Model):
    codigo = models.CharField(max_length=3, primary_key=True)
    nombre = models.CharField(max_length=50)
    duracion = models.PositiveSmallIntegerField(default=5)

    def __str__(self):
        txt = "{0} (Duracion: {1} años(s))"
        return txt.format(self.nombre, self.duracion)

Comment Code
class Estudiante(models.Model):
    cc = models.CharField(max_length=8, primary_key=True)
    apellidoPaterno = models.CharField(max_length=35)
    apellidoMaterno = models.CharField(max_length=35)
    nombres = models.CharField(max_length=35)
    fechaNacimiento = models.DateField()
    sexos = [
        ('F', 'Femenino'),
        ('M', 'Masculino')
    ]
    sexo = models.CharField(max_length=1, choices=sexos, default='F')
    carrera = models.ForeignKey(Carrera, null=False, blank=False, on_delete=models.CASCADE)
    vigencia = models.BooleanField(default=True)

    def nombreCompleto(self):
        txt = "{0} {1}, {2}"
        return txt.format(self.apellidoPaterno, self.apellidoMaterno, self.nombres)
```

En el archivo admin.py se realizan las importaciones (*Ver Figura #2*) para que el código funcione correctamente.

Figura #2: Registro de importaciones

```
from django.contrib import admin
from Modulos.academica.models import *

# Register your models here.

admin.site.register(Carrera)
admin.site.register(Estudiante)
admin.site.register(Curso)
admin.site.register(Matricula)
```

Después de esto se hace la migración para que se cree la base de datos con los siguientes comandos

```
PS C:\Users\usuario\Desktop\universidad_ss\universidad_s> Python manage.py migrate
```

```
PS C:\Users\usuario\Desktop\universidad_ss\universidad_s> python manage.py makemigrations
```

Implementación de Módulos

En Settings.py se indica la aplicación o los módulos a utilizar, enfocados en la parte académica, reflejando 4 entidades: registro de carreras, registro de estudiantes, registro de curso, matrícula del estudiante y los cursos (Ver Figura #3).

Figura #3. Instalación de aplicaciones

```

INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'Modulos.academica',
]

```

Se crea una carpeta llamada Templates, dentro se crea un archivo .html, el cual se ira modificando con el framework Bootstrap (Ver Figura #4).

Figura #4. Código para la Visualización utilizando HTML5 y Bootstrap

```

{% extends './base.html' %}

{% block title %} Gestión de Cursos {% endblock %}

{% block body %}
<div class="row">
  <div class="col-sm-12 col-md-6 col-lg-4 col-xl-4">
    {% if messages %}
    {% for message in messages %}
    <div class="alert alert-dismissible alert-success">
      <button type="button" class="close" data-dismiss="alert">&times;</button>
      <strong class="text-dark">{{ message }}</strong>
    </div>
    {% endfor %}
    {% endif %}
    <h2>Gestión de Curso</h2>
    <div class="card">
      <div class="card-body">
        <form action="/registrarCurso/" method="POST">{% csrf_token %}
          <div class="form-group">
            <input type="text" id="txtCodigo" name="txtCodigo" class="form-control" placeholder="Código"
              minlength="6" maxlength="6" required>
          </div>
          <div class="form-group">
            <input type="text" id="txtNombre" name="txtNombre" class="form-control" placeholder="Nombre"
              maxlength="50" required>
          </div>
          <div class="form-group">
            <input type="text" id="numCreditos" name="numCreditos" class="form-control" min="1" max="9"
              value="1" required>
          </div>
          <div class="form-group">
            <button type="submit" class="btn btn-success btn-block text-white">Guardar</button>
          </div>
        </form>
      </div>
    </div>
  </div>
</div>

```

Incorporación de un Asistente virtual (Chat Bot)

La incorporación de un chat bot será un valor agregado a la investigación, para ello se utilizó documentación, videos y repositorios de GitHub

4.2 CAPITULO 2

Visual de la Interfaz

Cuando terminamos el proceso de programación tendremos como resultado la siguiente visualización:



The screenshot shows the Django administration interface. At the top, there is a blue header with the text "Administración de Django". Below this, the main content area is titled "Sitio administrativo". It is organized into three main sections, each with a blue header bar: "ACADEMICA", "ACADEMICO", and "AUTENTICACIÓN Y AUTORIZACIÓN". Each section contains a list of items with "Añadir" (Add) and "Modificar" (Modify) buttons. The "ACADEMICA" section includes Carreras, Cursos, Estudiantes, and Matriculas. The "ACADEMICO" section includes Cursos. The "AUTENTICACIÓN Y AUTORIZACIÓN" section includes Grupos and Usuarios.

En ella encontramos los 4 entornos académicos, la interfaz observada esta sin creada sin el framework Bootstrap, si se adiciona Bootstrap u otros framework para la visualización de la página web se observaría de la siguiente manera.

Universidad Web Contacto

¡Cursos listados! x

Gestión de Curso

Código

Nombre

1

Guardar

Listado de Cursos

#	Código	Nombre	Créditos	Opciones
1	2585	Ingeniería de sistemas	9	Editar Eliminar
2	520002	Administración de empresas	10	Editar Eliminar
3	130565	Contaduría	9	Editar Eliminar
4	89	Ingeniería de sistemas	10	Editar Eliminar

CONCLUSIONES.

- La implementación de Django y Python en el desarrollo de la aplicación web garantiza un proceso de desarrollo rápido y seguro, la información académica mediante modelos en Django facilita la administración y actualización de datos, optimizando la gestión universitaria.
- El uso de Bootstrap en la interfaz de usuario permite una visualización clara, mejorando la experiencia de los usuarios.
- El uso de VS Code y recursos adicionales en línea proporcionó un entorno de desarrollo eficiente y accesible para el proyecto, así como, los repositorios y la combinación de tecnologías como HTML5, Bootstrap y Django contribuyó a la creación de una aplicación web.
- La aplicación proporciona una herramienta integral que facilita tanto la gestión administrativa como el soporte a estudiantes y usuarios.

REFERENCIAS

- Shukla, X., y Parmar, D. J. (2016) Python - A comprehensive yet free programming language for statisticians, *Journal of Statistics and Management Systems*, Taylor & Francis, 19(2), 277 – 284.
- Vidal-Silva, C. L., Sánchez-Ortiz, A., Serrano, J., & Rubio, J. M. (2021). Experiencia académica en desarrollo rápido de sistemas de información web con Python y Django. *Formación universitaria*, 14(5), 85-94.
- Arias-Navarrete, A. S., Palacios-Pacheco, X. I., & Villegas-Ch, W. (2020). Integración de un chatbot a un LMS como asistente para la gestión del aprendizaje. *Revista Ibérica de Sistemas e Tecnologías de Informação*, (E32), 164-175.
- Morán, R. H. M. (2024). Las aplicaciones móviles para la enseñanza-aprendizaje en la asignatura Diseño y Desarrollo Web en la unidad educativa Mocache. *Sinergia Académica*, 7(2), 341-348.
- González, J. I. G., & Llanes, J. Y. A. (2024). Explorando la enseñanza gratuita de tecnologías para el desarrollo web. *XIKUA Boletín Científico de la Escuela Superior de Tlahuelilpan*, 12(Especial), 104-110.
- Condori Ayala, J. L. (2012). Python-DjangoFramework de desarrollo web para perfeccionistas Basado en el Modelo MTV. *Revista de Información, Tecnología y Sociedad*, 36
- Pesantes Aviles, X. A., & Vivanco Granda, R. M. (2018). Sistema de modelamiento dinámico para el análisis del tráfico de datos de la red de un isp (Bachelor's thesis, Espol).
- González Duque, R. (2011). Python para todos.
- Rodó, D. M. (2023). El lenguaje Python. Accessed: Jul, 7.