

Reporte de Laboratorio Nro. 2

Cristian Bonilla^{L00289636}

Universidad de las Fuerzas Armadas
cdbonilla@espe.edu.ec

Tema: Desarrollo Web con Flask

Resumen

Este trabajo contiene la parte explicativa de una aplicación web basada en una lista de tareas pendientes. Para lo cual vamos a utilizar flask misma que permite al usuario mantener una lista de tareas de tareas importantes, realizando las peticiones designadas por parte de la tutora, con la finalidad de familiarizarnos con los temas utilizados en clase, tanto teórico como práctico. Los argumentos que se verán a continuación es el lenguaje de programación Python y la utilización de flask que facilita la creación de aplicaciones web.

1. Introducción

Como primer punto se realizará una breve explicación teórica donde se describirá cada una de las características que se necesitan conocer para el buen desarrollo del informe. Luego se describirá las peticiones realizadas. Manteniendo siempre los 6 pasos para resolver un problema, de esta manera se utilizará los controladores adecuados, un controlador permite identificar la ruta '/' que será responsable de enlistar todas las tareas en la tabla HTML, mostrando un formulario que permite enviar un nuevo elemento de listas de tareas pendientes además de poder eliminar alguna tarea.

[1]

La segunda petición es el controlador ubicado en la ruta '/enviar' aceptando los datos del formulario HTML agregando la lista de tareas pendientes a una lista global, redirigiéndolo a la lista inicial del primer controlador, el último controlador se ubicará en '/borrar', esta función será el responsable de borrar las listas de tareas pendientes y dirigir nuevamente al primer controlador.

[1]

Cada una de las fases se describirá a continuación, lo cual nos ayudará a facilitar el entendimiento de este trabajo autónomo que desarrollaremos, para validar su ejecución se pondrá a funcionar la aplicación.

[1]

2. Método

Para desarrollar las peticiones solicitadas es necesario conocer la parte teórica que compete e influyen en el desarrollo del proyecto, por ello me permito citar algunos de los conceptos que

fueron base para la realización del método.

Flask

Conocido también como framework es aquel que ayuda y facilita en el desarrollo e implementación de un lenguaje de programación en una aplicación dividiéndola en tres capas que pueden ser de presentación, de datos y de dominio [1].

Es necesario especificar que es una aplicación genérica, que posee herramientas que colaboran entre sí para el desarrollo de una aplicación web con un soporte bien diseñado. La arquitectura de los frameworks está compuesta por la interacción de modelo vista controlador [3] .

Django sirve para realizar aplicaciones web gratuitas, con código abierto escrito por Python ayuda a desarrollar la aplicación con la utilización de pocos códigos permitiendo ser fácil y rápida. Pyramid se desarrolla con diferentes sistemas operativos con facilidad de instalación, sin embargo utiliza base de datos como SQL [3]. También tenemos a turbogear, muy parecido a django ya que también es escrito por Python utilizando la arquitectura modelo vista controlador. Web2py llamado framework empresarial es de código abierto programado por Python con componentes necesarios.

Python

Python es un lenguaje de programación muy conocido por su facilidad de uso y por ser código abierto, por otro lado, hay que destacar que Python es compatible con múltiples plataformas que pueden ser utilizados sin ningún inconveniente permitiendo la facilidad de extensión [3].

Python Flask

Desarrollador que tiene incorporado una base de datos, sin embargo, se puede añadir características deseadas para obtener información u objetivos que necesitamos, flask está escrito por Python y sirve para desarrollar aplicaciones web [4].

Aplicaciones web

Una aplicación web debe de ser atractiva ya que es la presencia y la forma de presentarse mediante un contacto visual interactuando con el usuario, para obtener este tipo de resultados es necesario que la tecnología este en constante evolución [2] .

Una vez entendido cada uno de los conceptos para resolver el problema de desarrollo de la aplicación, se puso en practica los seis puntos ya conocidos para el desarrollo de una aplicación.

3. Results and Analysis

Partiendo de todas las posibles soluciones que se necesitaban, primero seguimos los pasos que incluyen realizarse las incognitas de lo que vamos a resolver como son ¿de que manera pueden ingresar los usuarios las tareas?, ¿es necesario agregar mas datos como el nombre de la persona que realizará la tarea?, ¿se puede dar por culminada la tarea?, en fin en este paso surgieron muchas más inquietudes, que únicamente fueron nombradas las mas relevantes.

Continuando con el método para resolver nuestro problema, se desarrollo un algoritmo que nos permite eliminar las tareas que ya han sido ejecutas o resueltas, además se formulo la manera

adecuada para insertar las tareas como se muestra en la figura 1) y darles una prioridad siendo baja, media y alta como se muestra en la figura.

TAREAS DIARIAS

MIS TAREAS

Llenar el campo

RESPONSABLE

Llenar el campo

CORREO ELECTRÓNICO

Llenar el campo

DIFICULTAD BAJA

REGISTRAR

RESPONSABLE	DESCRIPCIÓN DE LA TAREA	CORREO ELECTRÓNICO	DIFICULTAD	ELIMINAR
				ELIMINAR TAREA

ERROR LLENAR TODOS LOS CAMPOS

Figura 1: Interface de la aplicación guardar tareas.

En el tercer paso ya establecimos nuestro algoritmo de la siguiente manera:

Importar la biblioteca de flask
from flask import Flask, redirect, render_template, request, url_for, flash

```
app = Flask(name, template_folder='template')
app.secret_key = 'ESPE'
```

Arreglo guardar todos los datos
lista_areas = []

Controlador
@app.route('/')
def index():
return render_template('index.html', lista_areas = lista_areas)

Controlador de todos los datos
@app.route('/enviar', methods=['POST'])
def enviar():
if request.method == 'POST':

```
t_descripcion = request.form['t_descripcion']
t_responsable = request.form['t_responsable']
```

```

t_correo = request.form['t_correo']
t_prioridad = request.form['t_prioridad']

```

```

    if t_descripcion == '' or t_correo == '':
        flash('ERROR LLENAR TODOS LOS CAMPOS')
        return redirect(url_for('index'))
    else :

```

```

        flash('TAREA AGREGADA')

```

```

        lista_tareas.append('t_descripcion' : t_descripcion, 't_responsable' : t_descripcion, 't_correo' : t_correo, 't_prioridad' : t_prioridad)

```

```

    return redirect(url_for('index'))

```

```

    Controlador para eliminar la tarea
    @app.route('/borrar', methods=['POST'])
    def borrar():
        if request.method == 'POST':

```

```

            if lista_tareas == [] :

```

```

                flash('NO EXISTEN TAREAS')
                return redirect(url_for('index'))

```

```

            else:
                lista_tareas.clear()
                flash('TAREAS BORRADAS')
                return redirect(url_for('index'))

```

```

    Metodo para correr la aplicación
    if __name__ == '__main__':
        app.run(debug=True)

```

Una vez consolidado nuestro código en el cuarto paso nuestra aplicación permite guardar tareas diarias, indicando el responsable, correo electrónico y el grado de dificultad que tendrá dicha tarea como se muestra en figura 2) . Así mismo se puede eliminar alguna tarea ya realizada. En caso de no llenar todos los campos, no se guardará ningún dato.

TAREAS DIARIAS

MIS TAREAS

Llenar el campo

RESPONSABLE

Llenar el campo

CORREO ELECTRÓNICO

Llenar el campo

DIFICULTAD

BAJA

REGISTRAR

RESPONSABLE	DESCRIPCIÓN DE LA TAREA	CORREO ELECTRÓNICO	DIFICULTAD	ELIMINAR
Realizar el Laboratorio 2 con todos los conocimientos posibles.	Realizar el Laboratorio 2 con todos los conocimientos posibles.	cdbonilla@espe.edu.ec	ALTA	limpiar

ELIMINAR TAREA

TAREA AGREGADA

Figura 2: Aplicación crear tareas.

En el quinto paso procedemos a ejecutar nuestra aplicacion, siendo gratos los resultados ya que cumplen con todos los pedidos realizados por parte de la tutora, como se muestra en la figura 3).

TAREAS DIARIAS

MIS TAREAS

Llenar el campo

RESPONSABLE

Llenar el campo

CORREO ELECTRÓNICO

Llenar el campo

DIFICULTAD

BAJA

REGISTRAR

RESPONSABLE	DESCRIPCIÓN DE LA TAREA	CORREO ELECTRÓNICO	DIFICULTAD	ELIMINAR
Realizar el Laboratorio 2 con todos los conocimientos posibles.	Realizar el Laboratorio 2 con todos los conocimientos posibles.	cdbonilla@espe.edu.ec	ALTA	limpiar

ELIMINAR TAREA

TAREA AGREGADA

Figura 3: Resultados de la aplicación.

Finalmente llegamos al último escalón, que consiste en evaluar la solución, donde se debe

verificar, si el programa realizado da respuesta a todas las incógnitas que se produjeron en el primer escalón 4), de ser así se ha cumplido con éxito todos los pasos.

TAREAS DIARIAS

MIS TAREAS
 Llenar el campo

RESPONSABLE
 Llenar el campo

CORREO ELECTRÓNICO
 Llenar el campo

DIFICULTAD: BAJA

REGISTRAR

RESPONSABLE	DESCRIPCIÓN DE LA TAREA	CORREO ELECTRÓNICO	DIFICULTAD	ELIMINAR
Realizar el Laboratorio 2 con todos los conocimientos posibles.	Realizar el Laboratorio 2 con todos los conocimientos posibles.	cdbonilla@espe.edu.ec	ALTA	limpiar

ELIMINAR TAREA

TAREA AGREGADA

Figura 4: Comprobación de cada uno de los botones.

4. Discusión

Es importante seguir la resolución de problemas ya que de esa manera podemos alcanzar un orden adecuado en nuestro programa y mucho mas cundo se trata de establecer tareas, además nos permite ampliar el rango de solución, para determinar el algoritmo adecuado del problema y los subproblemas que se pueden ir dando, también en la programación siempre se debe seguir un proceso ordenado que permita generar una solución adecuada porque se interactúa con más personas al momento de resolver problemas, es decir el código debe ser comprendido por todos los miembros.

5. Conclusión

La utilización de los brootstrap porque nos permite ahorrar tiempo al momento de programar, de esta manera podemos realizar proyectos mucho más eficientes siendo competitivos y sobresaliendo.

Además, nos permitió repasar los tipos de datos, librerías, operaciones con variables y todas las funciones que nos ayudan a solucionar los problemas durante la programación.

Referencias

- [1] Mauro D’Agostino et al. Desarrollo de una aplicación web para la gestión de un laboratorio remoto para prácticas de control automático. 2018.
- [2] José María Galindo Haro. Diseño e implementación de un marco de trabajo (framework) de presentación para aplicaciones jee. 2008.
- [3] Jimmy Rolando Molina Ríos, Nancy Magaly Loja Mora, Mariuxi Paola Zea Ordóñez, and Erika Lizbeth Loaiza Sojos. Evaluación de los frameworks en el desarrollo de aplicaciones web con python. *Revista Latinoamericana de ingeniería de Software*, 4(4):201–207, 2016.
- [4] Mandeep Singh, Ayushi Verma, Aashwaath Parasher, Nidhi Chauhan, and Gaurav Budhiraja. Implementation of database using python flask framework. *International Journal of Engineering and Computer Science*, 8(12):24890–24893, 2019.