



**INSTITUTO TECNOLÓGICO DE COSTA RICA
ESCUELA DE COMPUTACION**

TALLER DE PROGRAMACION, GRUPO 04

PROGRAMA 3: PARQUEO

ESTUDIANTE:

MARIO GOMEZ VEGA

2022029053

PROFESOR:

WILLIAM MATA RODRIGUEZ

21/06/022

Semestre I, 2022

Contenidos

Enunciado del Proyecto	3
Temas investigados	4
Software de control de versiones	4
Envío de correos electrónicos	4
Nuevas características de tkinter exploradas	4
Creación de archivos pdf	4
Conclusiones del trabajo	5
Problemas encontrados y soluciones	5
Problemas encontrados	5
Soluciones	5
Aprendizajes obtenidos	5
Estadística de tiempos	6
Lista de revisión del proyecto	7
Funcionalidades adicionales	8
Mostrar/ocultar tarjeta de crédito	8
Bibliografía	9

Enunciado del Proyecto

El presente documento detalla la investigación y desarrollo del tercer y último proyecto del curso Taller de Programación Grupo 4, a cargo del profesor William Mata Rodríguez. Dicho proyecto se basa en el desarrollo de un programa bajo el nombre Parqueo. El anterior, como lo indica su nombre, simula un parqueo y, adicionalmente, un cajero para el manejo de cobros y pagos relacionados al parqueo.

Este proyecto, a diferencia de los anteriores y con autorización del profesor, se desarrolló utilizando programación orientada a objetos. Dicho paradigma de programación y su aplicación en el presente proyecto se omite parcialmente de la presente investigación, dado que la gran mayoría de conceptos utilizados fueron vistos en clase.

Además de la investigación necesaria para el desarrollo del programa, se incluye la investigación de los siguientes temas especificados en los objetivos del proyecto:

- Software de control de versiones
- Envío de correos electrónicos
- Características de Tkinter exploradas para este proyecto

Finalmente, también se detallan como conclusiones del proyecto los problemas encontrados durante su desarrollo, las soluciones respectivas, y los aprendizajes obtenidos como resultado.

Temas investigados

Software de control de versiones

El software de control de versiones es una herramienta utilizada por desarrolladores para manejar cambios hechos a un programa. Son particularmente útiles para reducir el tiempo de desarrollo, pues mantienen una base de datos que contiene un historial de cada modificación realizada. En caso de cometer un error, el software de control de versiones permite revertir cambios y comparar diferentes versiones del programa para ayudar a corregir errores, a la vez que se minimiza el impacto sobre otros miembros del equipo de desarrollo.

Gracias al software de control de versiones, es posible evitar conflictos entre diferentes contribuciones al código, pues este permite unificar y comparar contribuciones hechas en diferentes partes del programa. Además, esta funcionalidad permite que la depuración y el desarrollo procedan al mismo tiempo hasta que se complete una nueva versión del programa.

Durante el proyecto, se utilizó GitHub para control de versiones. Sin embargo, no fue necesario revertir ningún cambio hecho al programa.

Envío de correos electrónicos

Para manejar envío de correos en Python, no es necesario instalar ninguna librería externa; Python incluye el módulo SMTPlib, el cual utiliza el protocolo SMTP para enviar correos. Además, se puede utilizar el módulo MIME para incrementar la flexibilidad del programa. Por último, se utiliza el servicio de correos de Google, Gmail, para enviar el correo generado por el programa.

Nuevas características de tkinter exploradas

Una de las características de tkinter que fue necesario explorar con mayor profundidad es la posibilidad de configurar eventos a entradas específicas del usuario, como por ejemplo el hacer clic y mantener presionado, o presionar la tecla Enter. Esto fue necesario para implementar la funcionalidad adicional que se detalla en el apartado correspondiente.

Creación de archivos pdf

Para cumplir con las funcionalidades del programa que requieren la creación de archivos pdf, se hizo uso de la librería FPDF, la cual permite formatear y generar archivos pdf a través de Python. Dicha librería no requirió mucha investigación para utilizarse, pues esta se basa en objetos, los cuales se utilizaron extensivamente en el desarrollo del programa.

Conclusiones del trabajo

Problemas encontrados y soluciones

Problemas encontrados

El primero de los problemas encontrados durante el desarrollo del programa fue relacionado a la visibilidad de métodos en Python, pues dicho lenguaje no permite controlar la visibilidad de métodos. Además, resulto inicialmente complicado convertir los datos guardados a formato json, pues muchos de estos se estaban manejando como atributos de diferentes objetos.

Soluciones

Python, al no tener control sobre visibilidad de métodos, utiliza en su lugar notación con uno o dos guiones bajos frente al nombre del método para indicar el nivel deseado de visibilidad. Sin embargo, esto no reduce su visibilidad, si no que los ordena debajo de los métodos “visibles” en la función de autocompletado del IDE. Dicha notación sugiere fuertemente tratar dichos métodos como lo indica la notación, pero esto se trata de una convención y no limita de ninguna manera el acceso a dichos métodos.

El problema de conversión de un objeto personalizado a formato json se solucionó a través de la creación de métodos de clase para serializar los atributos de cada objeto en formato json, y posteriormente deserializar cada uno y reasignar los valores al atributo correspondiente con un segundo método constructor. Esto último debido a que Python no permite la sobrecarga de métodos constructores, por lo que es imperativo crear uno propio.

Aprendizajes obtenidos

El presente proyecto, al haber sido desarrollado con programación orientada a objetos, fue útil para practicar dicho paradigma de programación, particularmente la utilización de métodos constructores y de clase. Además, el uso de objetos simplifico en gran medida la utilización de tkinter para la interfaz gráfica del mismo, resaltando las diferencias entre programación funcional y orientada a objetos.

Estadística de tiempos

Actividad Realizada	Horas
Análisis del proyecto	3
Diseño de algoritmos	5
Investigación	7
Programación	90
Documentación interna	3
Pruebas	3
Manual de usuario	4
Documentación	4
Etc.	1
Total	120

Lista de revisión del proyecto

Concepto	Puntos originales	Avance %	Puntos obtenidos	Análisis de resultados
Menú principal	2	100%		
Configuración	10	100%		
Cargar cajero	8	100%		
Saldo del cajero	4	100%		
Ingresos: Reales Estimados	8 8	67%		Desglosar ganancias entre efectivo/tarjeta de crédito habría requerido cambios mayores en el diseño del programa.
Entrada de vehículo	5	100%		
Cajero paso 1	5	100%		
Cajero paso 2: Registrar pago Devolver pago Enviar correos Recibo del pago	5 3 5 4	100%		
Cajero paso 3	5	100%		
Salida de vehículo: Eliminar elemento Agregar elemento Entrada automática	3 3 3	100%		
Validación de datos	8	100%		
Ayuda	5	100%		
Archivos: Leer Grabar	3 3	100%		
TOTAL	100	97%		
Funciones adicionales				Mostrar/ocultar tarjeta de crédito

Funcionalidades adicionales

Mostrar/ocultar tarjeta de crédito

Con el propósito de reforzar la seguridad del programa, se desarrolló adicionalmente la funcionalidad de ocultar la tarjeta de crédito. Haciendo uso de la misma función, se permite al usuario revelar temporalmente el numero para verificar errores de entrada a través de presionar y mantener el clic sobre el espacio de entrada de la tarjeta.

Bibliografía

albert, & Ostermiller, S. (29 de August de 2015). *List of All Tkinter Events*. Obtenido de Stack Overflow: <https://stackoverflow.com/questions/32289175/list-of-all-tkinter-events>

Gupta, L. (6 de July de 2021). *Python JSON - Custom Class Seralization*. Obtenido de HowToDoInJava: <https://howtodoinjava.com/json/custom-class-serialization/>

Reingart, M. (15 de August de 2012). *FPDF for Python*. Obtenido de Readthedocs: <https://pyfpdf.readthedocs.io/en/latest/index.html>

Sam, S. (23 de November de 2018). *Send mail from your Gmail account using Python*. Obtenido de Tutorialspoint: <https://www.tutorialspoint.com/send-mail-from-your-gmail-account-using-python>