

UNIVERSIDAD FIDÉLITAS


Programación Básica

Proyecto Final

Kenneth Andrey Barrientos Acuña
Joseph

I Cuatrimestre


San José, Costa Rica

	Ingeniería en Sistemas de Computación Proyecto Final	Curso: Programación Básica
		Profesor: MSc. Alvaro Camacho Mora
		Código curso: SC-115


índice interactivo:

Contenido

índice interactivo:	2
Introducción.....	4
Requerimientos del sistema:	5
Bibliotecas:	5
Descripción general del problema:	5
Algoritmo:	6
Paso 1:.....	6
Paso 2:.....	6
- Registro de usuario	6
Paso 3:.....	6
Usuario registrado	6
Paso 4:.....	7
Retirar dinero	7
Paso 5:.....	8
Depositar dinero	8
Paso 6:.....	8
Ver saldo Actual	8
Paso 7:.....	8
Pagar Servicios	8
Paso 8:.....	9

	Ingeniería en Sistemas de Computación Proyecto Final	Curso: Programación Básica
		Profesor: MSc. Alvaro Camacho Mora
		Código curso: SC-115

Compra/Venta de Divisas.....	9
Paso 9:.....	10
- Eliminar usuario	10
Paso 10:.....	10
- Salir, se regresa al menú de inicio	10
Paso 11:.....	10
Configuración Avanzada	10
Paso 12:.....	10
Eliminar usuario	10
Paso 13:.....	11
Modificar tipos de cambio.....	11
Paso 14:.....	11
- Salir del programa.....	11
Estructura de Software	11
CONCLUSIÓN	12
Bibliografía	12

	Ingeniería en Sistemas de Computación Proyecto Final	Curso: Programación Básica
		Profesor: MSc. Alvaro Camacho Mora
		Código curso: SC-115


Introducción

En dicho trabajo desarrollaremos la arquitectura y diseño de un software de última generación con el fin de implementarse en cajeros automáticos, para evitar la interacción entre los usuarios y los trabajadores del banco, el mismo desarrollara diferentes servicios para brindar un contacto amigable con los diferentes clientes, entre el cual utilizaremos menús, funciones, clases, almacenajes de información por medio de las matrices y archivos planos.

Podremos aprender y aplicar por medio de algoritmos dicho diseño y su estructura, es importante saber e identificar la importancia de cada paso y realizarlo de la manera correcta para obtener un resultado positivo y correcto en la búsqueda de una solución para la entidad bancaria.

Se podrán adentrar con toda la información aportada en nuestro desarrollo los comando por medio del menú para retiro de dinero, depósitos, consultar saldos, pago de servicios, compra y venta de divisas, eliminación de usuarios, entre otros. Significando un servicio muy completo y extremadamente necesario para cualquier entidad, en este caso el Global Bank Inc. Viéndose beneficiados sus clientes y trabajadores, ya que agiliza cualquier trámite sin necesidad de ingresar a una sucursal, o sea, el cliente atendiéndose por el mismo, aportando muchos beneficios para ambas partes.

Dicho trabajo nos brindara la satisfacción de poner en práctica el conocimiento adquirido en el curso actual, ya que significo un reto pero también nos permitió obtener comprensión que más adelante podrán constatar en cada paso y explicación que brindaremos.

	Ingeniería en Sistemas de Computación Proyecto Final	Curso: Programación Básica
		Profesor: MSc. Alvaro Camacho Mora
		Código curso: SC-115

Requerimientos del sistema:

Bibliotecas:


- `from getpass import getpass`
- `import os`
- `import random`
- `import shutil`

Descripción general del problema:

La empresa Global Bank Inc. necesita la arquitectura y diseño de un software de última generación para manejo de cajeros automáticos que será la sustitución de la interacción usuario/trabajador del banco. Por tanto, el cajero automático tendrá como parte de sus características el pago de servicios, depósitos y retiros de dinero en efectivo, registros de nuevos usuarios al sistema del banco, etc.

Para este código se requiere utilizar diferentes menús, funciones, clases (os), almacenar información en matrices y archivos planos

El gerente de la compañía le menciona que este sistema se espera esté listo en un plazo máximo de 12 semanas.

	Ingeniería en Sistemas de Computación Proyecto Final	Curso: Programación Básica
		Profesor: MSc. Alvaro Camacho Mora
		Código curso: SC-115

Algoritmo:

Paso 1:

- Se requiere hacer un menú principal para que muestre las siguientes opciones, registro de usuario, usuario registrado, configuración avanzada y salir (`def, print, variable, while true, break, int, input, if, elif, else`)


Paso 2:

- Registro de usuario: en esta opción se requiere registrar a nuevos usuarios que deseen registrarse con el banco y usar el cajero atm
- Se solicita el número de cedula, este tiene que ser de 9 dígitos, si se digitan una cantidad mayor o menor, se produjera un error (`variable, input, while, len !=, print`)
- Solicitud del nombre de usuario (`variable, input,`)
- Solicitud de un numero pin, este número pin tiene que ser de 4 dígitos (`variable, input, while, len !=, print`)
- Para la protección al usuario, el Pin no debe ser visible por seguridad (`Getpass, =,`)
- Para continuar con el registro se necesita hacer un depósito mínimo, ya sea en colones, dólares o en bitcoin, el usuario debe de decidir en cual opción desea realizar el deposito (`variable, input, print, float, break, continue, ==, <, w`)
- Todo lo anterior se debe de almacenar en archivos planos y en un arreglo (`import os, arreglo, append, w, as f: write`)

Paso 3:

Usuario registrado

- se solicita validar los datos anteriormente ingresados en el registro para poder acceder al corazón del programa (`with open, r, as, f,`)
- Se debe de solicitar el número de cedula del usuario registrado (`input`)
- Se valida que el numero de cedula sea válido, se tiene un máximo de 3 intentos para colocar una cedula valida, en caso de que se agoten, se debe volver al menú principal (`variable, =, while, <,+ =, >, -`)


	Ingeniería en Sistemas de Computación Proyecto Final	Curso: Programación Básica
		Profesor: MSc. Alvaro Camacho Mora
		Código curso: SC-115

- Si la cedula es valida se debe solicitar el numero pin, de igual manera se valida que no exceda el máximo de intentos (**Getpass, variable, input, for, len, if, else, return, +=, print**)
- ya una vez ingresado el usuario, se despliega un menú (**print, if, elif, else**)
 - o Retirar dinero
 - o Depositar dinero
 - o Ver saldo actual
 - o Pagar servicios
 - o Compra/Venta de Divisas
 - o Eliminar usuario
 - o Salir

Paso 4:

Retirar dinero

- permite al usuario retirar dinero desde las diferentes cuentas (**Variable**)
- Se le muestra al usuario la lista de cuentas que tiene a disposición (**print**)
- Una vez seleccionada la cuenta, se le pregunta al usuario cuanto es el monto que desea retirar (**input, float,**)
- El sistema sabe cual es el monto que el usuario tiene a disposición
- Si el monto solicitado es mayor al saldo actual se debe advertir mostrando un mensaje y evitar la transacción (**if else, return, print,)**)
- Se le da hasta un máximo de tres oportunidades al usuario para digitar un monto valido (**errores, +=, while <3, if, else**)
- Si falla mas de tres veces se devuelve al menú principal del sistema y se le advierte al usuario a través del despliegue de un mensaje (**print**)
- Si el monto solicitado es menor al saldo actual de la cuenta, se debe restar el monto solicitado e imprimir un mensaje indicando que se retiró el dinero solicitado y además, se debe imprimir el saldo actual. (**saldo – monto, variable saldo nuevo**)
- Se regresa al submenú

	Ingeniería en Sistemas de Computación Proyecto Final	Curso: Programación Básica
		Profesor: MSc. Alvaro Camacho Mora
		Código curso: SC-115

Paso 5:

Depositar dinero

- Debe consultar a que cuenta (colones, dólares o bitcoin) desea hacer el deposito (input, print)
- Se le muestra al usuario la lista de cuentas que tiene a disposición (print)
- Se debe preguntar al usuario cuanto quiere depositar. El monto debe ser positivo (requiere una verificación por parte del usuario y advertir en caso de error.) (if else, variable errores, +=)
- Se imprime un mensaje indicando al usuario que la transacción se hizo de forma correcta y además se imprime el saldo anterior (print, saldo + saldo nuevo)
- Se regresa al submenú

Paso 6:


Ver saldo Actual

- Se imprime en pantalla el saldo actual de las tres cuentas disponibles. (print, open, float, leer archivo,)
- Se regresa al submenú

Paso 7:

Pagar Servicios

- Despliega un menú con cada uno de los servicios (Electricidad, Agua, Telefonía, Internet, Impuestos, Colegios Profesionales, Tarjeta de crédito). (while true, if, elif, print, ==,)
- Una vez seleccionado el servicio que desea cancelar, el sistema verifica si tiene este servicio activo. (input, leer archivo,)
- Si no tiene, se regresa al menú de servicios (else, return)
- Si tiene, se procede al pago (if, ==,)
- Se muestra el saldo a pagar (print,)
- Se consulta de cual cuenta (colones, dólares o bitcoin) desea hacer el debito (input, if, elif, else)


	Ingeniería en Sistemas de Computación Proyecto Final	Curso: Programación Básica
		Profesor: MSc. Alvaro Camacho Mora
		Código curso: SC-115

- Se verifica que haya dinero suficiente de la cuenta seleccionada
- Si hay dinero suficiente, se realiza el pago correspondiente y el rebajo del saldo de la cuenta seleccionada
- Si se escoge una divisa que no es colones, el rebajo de la cuenta debe hacerse en la moneda correspondiente
- Se regresa al submenú de servicios (**return**)

Paso 8:

Compra/Venta de Divisas

- El sistema consulta que desea realizar. Es decir, compra o venta de dólares, compra o venta de colones y compra o venta de bitcoin. (**input, while true, if, elif, else,)**)
- Para la operación de compra se debe cumplir lo siguiente:
 - Se debe consultar la cuenta desde la que se obtendrá el dinero para la compra (cuenta origen) (**input**)
 - Una vez seleccionada la cuenta del origen de los fondos, se debe consultar cuando se quiere comprar. (**input, float,)**)
 - El sistema debe verificar que se cuente con el suficiente saldo para soportar la compra de la divisa (**if, else,)**)
 - Si se cuenta con el saldo suficiente, se realiza la operación y se deduce el monto de la cuenta origen y se acredita en la cuenta destino (**+, -**)
 - Si no hay saldo suficiente en la cuenta origen, la operación se cancela y se vuelve al submenú anterior. (**return**)
- Para la operación de venta se debe cumplir lo siguiente
 - Se debe consultar la cuenta donde se va a acreditar el dinero resultante de la venta (cuenta destino) (**input**)
 - Una vez seleccionada la cuenta destino de los fondos, se debe consultar cuando se quiere vender (**float, input**)
 - El sistema debe verificar que se cuente con el suficiente saldo para soportar la venta de la divisa. (**if, else**)
 - Si se cuenta con el saldo suficiente, se realiza la operación y se deduce el monto de la cuenta origen y se acredita en la cuenta destino (**+, -**)
 - Si no hay saldo suficiente en la cuenta origen, la operación se cancela y se vuelve al submenú anterior. (**return**)
 - Se volverá al submenú anterior. (**return**)

	Ingeniería en Sistemas de Computación Proyecto Final	Curso: Programación Básica
		Profesor: MSc. Alvaro Camacho Mora
		Código curso: SC-115

Paso 9:

- Eliminar usuario, Esta operación eliminar todo registro del usuario, esto implica que todas las carpetas y registros del usuario deberán ser borrados
- Se le debe consultar el PIN al usuario. (`input`, `getpass`)
- Si coincide, se eliminan todos los registros (carpetas) asociadas al usuario. (`if ==`, `exist`,)
- Se vuelve al menú principal (`return`)
- Si el PIN no coincide, se le advierte al usuario y se vuelve al submenú anterior (`print`)

Paso 10:

- Salir, se regresa al menú de inicio (`return`)
- Se deben actualizar los archivos que contienen los saldos actuales y el estado de saldo de cada uno de los servicios.

Paso 11:

- Regresa al menú principal


Configuración Avanzada

- Aquí se pueden realizar modificaciones de los tipos de cambio y eliminación de usuarios.
- Acceder a esta opción implica la autenticación como usuario avanzado. Esto implica la solicitud de un PIN especial. (`input`, `print`, `leer archivo`)
- En caso de que el usuario digite el PIN de forma errónea, se vuelve al menú principal (`if`, `else`, `==`)
- Se despliega un menú: (`while true`, `if elif`, `else`, `==`, `print`)
- Eliminar usuario
- Modificar tipos de cambio
- Salir

Paso 12:

Eliminar usuario

- Se solicita el número de cedula del usuario que se desea eliminar (`input`, `==`)
- Si el usuario existe se procede a la eliminación de todos los registros (carpetas) asociadas al usuario (`leerarchivo`)
- Si no existe, se advierte al usuario y se regresa al menú anterior. (`print`)

	Ingeniería en Sistemas de Computación Proyecto Final	Curso: Programación Básica
		Profesor: MSc. Alvaro Camacho Mora
		Código curso: SC-115

Paso 13:

Modificar tipos de cambio


- Se despliega un menú que muestra cada uno de los tipos de conversiones existentes, por ejemplo: (`while`, `true`, `if`, `elif`, `else`, `print`, `input`)
- ¿Qué tipo de cambio desea modificar?
 1. Compra de colones
 2. Venta de colones
 3. Compra de dólares
 4. Venta de dólares
 5. Compra de bitcoin
 6. Venta de bitcoin
 7. Salir
- Según el ítem seleccionado se procederá a realizar la modificación/actualización del archivo (s) que contiene cada factor de conversión. Lo que se espera como entrada de usuario es un número que represente el factor de conversión. Los factores de conversión pueden ser inventados o ser reales a la situación macroeconómica nacional.
- Salir del menú de modificar los tipos de cambio

Paso 14:

- Salir del programa (`print`)

Estructura de Software

- Comentarios

	Ingeniería en Sistemas de Computación Proyecto Final	Curso: Programación Básica
		Profesor: MSc. Alvaro Camacho Mora
		Código curso: SC-115

CONCLUSIÓN

Aunque se nos presentaron ciertos inconvenientes a la hora de hacer el programa ya que por falta de conocimiento en la materia pudo resultar complicado hasta cierto punto, pero con la colaboración de ambos y la investigación externa que pudimos realizar obtuvimos los resultados deseados y se volvió muy satisfactorio ya que transformamos las dificultades y falta de conocimiento en oportunidades.

A la hora de poner el programa a correr y que el mismo diera los resultados esperados fue gratificante ya que pudimos alcanzar el objetivo del curso de una manera práctica basándonos en la teoría y en la investigación, por medio de consultas. Es importante mencionar lo fundamental que es para este tipo de programas el orden y prestar atención a los detalles ya que si se siguen los pasos establecidos obtendremos los resultados que perseguimos.

Bibliografía

Jaiswal, S. (s.f.). *Getpass module in Python*. Obtenido de <https://www.javatpoint.com/getpass-module-in-python>
python. (s.f.). *python*. Obtenido de <https://docs.python.org/3/library/random.html>
Shutil. (s.f.). *Python*. Obtenido de <https://docs.python.org/es/3/library/shutil.html>