|  |  |  |
| --- | --- | --- |
| Proyecto  Primer Trabajo Practico Sistemas Operativos 75:08 | Grupo  Nº | Fecha de Vencimiento  1 de octubre de 2019 22 horas |
| Integrantes: Padrón, Nombre y Apellido | Correo Electrónico | |
| 1.  2.  3.  4. |  | |

|  |
| --- |
| Instrucciones  En este Word se encuentra el enunciado del TP  En el Excel se encuentra   * el layout de las estructuras de entrada y salida * los datos del maestro de errores CodigosISO8583 * los datos para armar los lotes de prueba   Armen los archivos maestros y los casos de prueba respetando la estructura indicada  Descripción  Se reciben archivos con información de transacciones efectuadas con tarjetas de crédito/debito. Cada archivo contiene registros de varias transacciones y el cierre del lote  Se debe crear un archivo para cada fecha de cierre de lote con las transacciones cumpliendo con el formato y contenido especificado. Se debe crear un archivo que reúna todos los cierres de lote cumpliendo con el formato y contenido especificado  Todo ello previa validación del input  Trabajo Por Realizar  Se debe realizar un script de instalación, uno de inicialización y un proceso que lee el input, lo valida y crea los archivos de salida.  Se debe realizar dos comandos complementarios Start y Stop para arrancar o detener el proceso  Todos en Shell script  Se debe lograr la ejecución de todo el sistema y dar evidencia de ello  Se debe crear un paquete con instrucciones para poder desempaquetar, instalar y ejecutar el sistema  Se debe realizar una autoevaluación indicando   * SI, cuando la entrega cumple con lo solicitado * NO, si no cumple con lo solicitado o no puede dar evidencia de ello * NA si el ítem no aplica a su solución   El TP vence el 1 de octubre a las 22 horas, es decir que deben remitir la entrega por correo electrónico a [so7508@gmail.com](mailto:so7508@gmail.com) antes de esa hora. |

Contenido

La tabla de contenido está vacía porque no estás utilizando los estilos de párrafo que deben aparecer en ella.

|  |  |
| --- | --- |
| Enunciado del TP y Autoevaluación | SI – NO – NA |
| Entrega del TP NOTA1: Toda la entrega puede estar subida a un repositorio (como github) y dar acceso público al link ejemplo: https://github.com/owner/so7508\_TP1\_Grupo1  Si emplea repositorio escriba el link a la documentación en cada punto, ejemplo: https://github.com/owner/so7508\_TP1\_Grupo1/blob/master/README.md | |
| Entrega realizada a tiempo y conteniendo: | |
| 1. Documentos:  * Archivo README * Este documento con autoevaluación * Documento con lo solicitado en el punto 2. Ítems 10 y siguientes (puede estar aparte o dentro de la autoevaluación)   Todo esto por fuera del paquete |  |
| 1. Paquete de instalación conteniendo:  * comandos y archivos maestros * archivos de prueba de la catedra * archivos de prueba propios |  |
| 1. Resultado de las pruebas tanto con archivos de la catedra como con los archivos de prueba propios  * log de instalación * archivo de configuración * log de inicialización * log de proceso: útil para saber que fueron ejecutando, que ARCHIVOS, en que orden |  |
| 1. Archivos de salida  * Archivos rechazados * archivo de cierre de lote * archivos de transacciones |  |

|  |  |
| --- | --- |
| Documentación conteniendo | |
| **Archivo README** Con instrucciones detalladas desde como loguearse, descargar archivos, descomprimir, ejecutar, etc.  Por ejemplo: |  |
| 1. las instrucciones de descarga  * Una explicación de cómo descargar el paquete * Una explicación de cómo descomprimir * Una explicación de lo que se crea a partir de la descompresión |  |
| 1. las instrucciones de instalación  * Una explicación de cómo se hace una instalación o reparación * Que deja la instalación y dónde |  |
| 1. las instrucciones de ejecución  * Pasos (en que orden) se ejecutan los comandos * forma de invocación de cada comando * Como arrancar o detener el proceso |  |
| 1. Ejemplos de como hacer una prueba basica |  |
| 1. **Documento de Autoevaluación**  * Completo * Pie de página con el número de grupo * Listado de integrantes que efectivamente han participado en la resolución del TP en la primera hoja |  |
| **Documento del TP**   1. Con Hipótesis y aclaraciones  * Describa todas las hipótesis y aclaraciones que han asumido en la resolución del TP |  |
| 1. Con descripción de Problemas relevantes  * Describa la lista de los problemas relevantes que se hayan presentado durante el desarrollo y como lograron resolverlos. |  |
| 1. Con el árbol de Estructura  * Liste la estructura resultante luego de una instalación estándar, ejemplo |  |
| 1. Comandos Adicionales ¿ha creado nuevos comandos?    * Describa los comandos auxiliares con una explicación de para qué y donde lo usa |  |
| 1. Archivos Auxiliares ¿ha creado nuevos archivos?,    * Describa los archivos auxiliares con una explicación de para qué y donde lo usa |  |
| 1. Archivos de prueba de la catedra  * liste los nombres de archivos surgidos de los datos remitidos por la practica |  |
| 1. Archivos de prueba del grupo  * liste los nombres de los archivos de prueba creados por Uds. y para que se usan (que casos de prueba representan) |  |
| 1. Cualquier otro ítem que quieran agregar (observaciones, recomendaciones, comentarios) que facilite el testeo |  |

|  |  |
| --- | --- |
| Paquete | |
| El paquete de instalación deberá estar contenido en un único archivo instalable en formato “.tgz” con todos los archivos y directorios empaquetados en un archivo “tar” y luego comprimido con “gzip. | |
| DIRECTORIO DE TRABAJO  Toda la instalación debe realizarse bajo un mismo directorio de trabajo llamado Grupoxx donde xx es su número de grupo  NOTA2: Todo el camino (path) que va desde la raíz hasta Grupoxx lo denominaremos genéricamente $GRUPO | |
| SUB DIRECTORIO DE CONFIGURACIÓN  La extracción del paquete debe generar en $GRUPO un subdirectorio /conf  NOTA 3: Este nombre es reservado. Ningún comando puede crear otro subdirectorio /conf en otra parte del sistema. | |
| NOTA 4: El sistema debe poder ser ejecutado SIN SER ROOT | |
| Prueba | |
| 1. Test: ¿Al extraer el paquete, se verifica la existencia de los directorios de trabajo los comandos y los archivos maestros? |  |
| 1. Test: comprobar que Ningún comando puede leer o grabar información del sistema fuera del directorio $Grupo |  |
| 1. Test: comprobar que reserva los scripts y archivos originales en algún subdirectorio   Indique donde lo reserva |  |

|  |  |
| --- | --- |
| Script de Instalación | |
| Propósito | |
| El propósito de este comando es realizar la instalación del sistema | |
| **El comando debe soportar los siguientes escenarios** | |
| * El sistema nunca fue instalado (se deberá instalar el sistema) * El sistema está bien instalado (no se deberá instalar el sistema, solo mostrar datos de la instalación) * El sistema está instalado, pero está incompleto (se deberá reparar la instalación siempre que se pueda): Invoque la reparación con la opción -r | |
| Configurar los nombres de los directorios. | |
| Solicitar al usuario que defina   * 1. El directorio de ejecutables (/bin   2. El directorio de archivos maestros (/maestros)   3. El directorio de arribo de archivos externos, es decir, los archivos de novedades con los lotes a procesar   4. El directorio donde se depositan temporalmente las novedades aceptadas para que luego se procesen (novedades OK)   5. El directorio donde se depositan todos los archivos rechazados (novedades NOK)   6. El directorio donde se depositan los archivos ya procesados   7. El directorio donde se depositan los archivos de salida   Advertir que los logs del sistema se depositan en /conf/log | |
| En todos los casos, cuando se Interactúa con el usuario se debe proponer un valor por default para el nombre del directorio mostrándolo entre paréntesis.  *Ejemplo: Defina el directorio de ejecutables ($grupo/bin):* | |
| Evitar nombres de directorios duplicados (dos directorios con igual nombre)  Evitar el nombre de directorio /conf (se considera nombre reservado)  Evitar cualquier otro nombre de directorio que el grupo haya considerado reservado | |
| **Confirmación de la instalación** | |
| Cuando el usuario termina de configurar los nombres de directorios, mostrar los valores que ingresó, como va a quedar la estructura, y solicitarle que confirme para continuar  *Ejemplo:*  *TP SO7508 2º Cuatrimestre 2019. Copyright © Grupo xx*  *Directorio padre: $grupo*  *Directorio de configuración: /conf*  *Archivos de log: conf/log*  *Librería de ejecutables:*  *Repositorio de maestros:*  *Directorio para las novedades …*  *Directorio para los archivos aceptados…*  *Directorio para los archivos rechazados…*  *Directorio para Archivos procesados…*  *Directorio para los archivos de salida*  *Estado de la instalación: LISTA*  *¿Confirma la instalación? (SI-NO): \_* | |
| No ir creando directorios hasta no tener la confirmación de la instalación | |
| Si el usuario indica **No**, volver a solicitar los directorios, pero esta vez mostrando como default lo recientemente ingresado, en lugar de los propuestos inicialmente (es decir, que el script debe tener memoria de los valores ingresados previamente) | |
| Si el usuario indica **Si** crear la estructura de directorios definida y mover al directorio de maestros, los archivos maestros, al de ejecutables los ejecutables  A medida que se hace la creación de directorios y su llenado informarlo mediante mensajes (por pantalla y en el log de la instalación) | |
| Grabar el archivo de configuración con el formato indicado y el nombre /conf/tpconfig.txt   * Para el directorio $grupo crear un registro * Para el directorio /conf crear un registro * Para el directorio /conf/log crear un registro * Para cada directorio definido crear un registro   Puede crear más registros si lo considera necesario | |
| Grabar log mientras se ejecuta el script | |
| Todo lo que se muestra al usuario por pantalla y sus respuestas debe ser registrado en el logde la instalación  El log de la instalación no debe reescribirse, aun cuando se produce cancelación del script o reinstalación. Grabar el archivo en $GRUPO/conf/log  *Ejemplo: $GRUPO/conf/log/instalacion.log* | |
| Prueba | |
| 1. ¿Hay Indicación correcta en el readme de cómo se ejecuta este script? |  |
| 1. Test: ¿el instalador detecta cuando el sistema nunca fue instalado y lo instala? |  |
| 1. Test: ¿el instalador detecta cuando el sistema está bien instalado y no lo instala? |  |
| 1. Test: ¿el instalador detecta cuando el sistema se debe reparar y lo repara? |  |
| 1. Test: ¿se aclara en las Hipótesis que controla el script para determinar que no está instalado, que está bien instalado, cuando es posible reparar? ¿cual es el nombre reservado para salvaguardar el original? |  |
| 1. Test: ¿el instalador permite Configurar los nombres de los directorios? |  |
| 1. Test: ¿el instalador propone valores default? |  |
| 1. Test: ¿el instalador evita nombres duplicados?   *Ejemplo: /misarchivos (para el de ejecutables) /misarchivos (para de maestros)* |  |
| 1. Test: el instalador evita /conf y todo otro directorio reservado que se haya definido (esto debe estar aclarado en las hipotesis) |  |
| 1. Test: ¿el instalador procesa bien nombres de directorios con espacios intermedios?   *Ejemplo: /mis novedades* |  |
| 1. Test: ¿el instalador solicita confirmación de la instalación?   *¿Confirma la instalación? (SI-NO): \_* |  |
| 1. Test: ¿si la instalación no se confirma, reinicia el proceso y se tiene memoria de lo que fue cargando previamente el usuario? |  |
| 1. Test: ¿si la instalación se confirma, realiza la instalación y se depositan los comandos y maestros en los directorios correctos? |  |
| 1. Test: cuando la instalación se confirma, ¿recién allí crea los directorios? |  |
| 1. En el log: ¿se registra toda la interacción entre el usuario y el script tanto cuando la instalación es exitosa como cuando la instalación es cancelada o reparada? muestra el paso a paso de la creación de directorios y el movimiento de archivos |  |
| 1. ¿Se crea el archivo de configuración con el contenido solicitado? |  |

|  |  |
| --- | --- |
| Inicialización | |
| Propósito | |
| El propósito de este comando es preparar el entorno de ejecución del TP (variables de ambiente) y asegurar que estén dadas todas las condiciones para la ejecución del sistema e invocar al proceso (que existan los scripts y el maestro CodigosISO8583  Lo mas importante en este script es su forma de invocación, la cual debe asegurar que cuando invoca al proceso no se generen sub-shells  NOTA5: los pasos enumerados a continuación son a modo indicativo, una guía, el grupo puede alterar el orden en el cual realiza los pasos siempre y cuando no se altere el resultado final esperado: *que solo si todo está OK se dispare el proceso y que las variables estén visibles para el proceso de manera que éste no necesite leer el archivo de configuración*. | |
| **El comando debe soportar los siguientes escenarios** | |
| * El sistema nunca fue inicializado (se deberá **inicializar el sistema**) * El sistema ya fue inicializado exitosamente (no se deberá reinicializar) * El sistema no se puede inicializar porque falta algún componente (se deberá explicar el motivo, como reparar la instalación y terminar) | |
| Inicializar el sistema | |
| Leer el archivo /conf/tpconfig.txt y verificar que todos esos directorios existen  Si se detecta que alguno de estos directorios no existe:  Informa al usuario de la situación, que es lo que está faltando  le brinda una explicación de los pasos que debe seguir para reparar la instalación  termina | |
| los archivos del directorio maestro deben tener permiso de lectura, si no lo tiene corregirlo  los ejecutables deben tener permiso de lectura y ejecución, si no lo tiene, corregirlo | |
| Setear variables de ambiente  Todos los identificadores de directorio definidos en /conf/tpconfig.txt deben definirse como variables de ambiente con el contenido adecuado.  Estas variables deben permanecer durante toda la ejecución del sistema y ser accesibles por todos los comandos desencadenados a partir de éste.  El único que lee el archivo de configuración es este script, el resto de los comandos trabajan con las variables de ambiente directamente. | |
| Arrancar el proceso | |
| Invocar al script PROCESO e indicar por pantalla y en el log el process id  ADVERTENCIA: no invocar el proceso si ya hay uno corriendo. Avisar cuando pasa eso | |
| Grabar log mientras se ejecuta el script | |
| Todo lo que va sucediendo durante la ejecución se debe grabar en el log y mostrarse por pantalla  El nombre del log debe ser distinto a los otros que se crean | |
| Prueba | |
| 1. ¿Hay Indicación correcta en el readme de como se ejecuta este script? |  |
| 1. Test: ¿el comando detecta cuando el sistema nunca fue inicializado y lo inicializa? |  |
| 1. Test: ¿detecta cuando el sistema ya fue inicializado exitosamente y no reinicializa? |  |
| 1. Test: ¿detiene la inicialización cuando falta algún componente? Directorios / archivos |  |
| 1. Test: ¿corrige permisos? |  |
| 1. Test: ¿setea las variables de ambiente y estas se exportan al siguiente script? |  |
| 1. Test: ¿arranca el proceso? ¿Se muestra el process id? |  |
| 1. En el log, ¿hay líneas del progreso de la inicialización?   *Ejemplo:* “la verificación de directorios es exitosa” “corrección de permisos efectuada” |  |
| 1. ¿En el log, graba el pid cuando arranca el proceso? ESTO ES MUY IMPORTANTE PARA REALIZAR LOS TESTEOS |  |
| 1. ¿En el log, hay líneas de “inicialización cancelada” con el motivo de cancelación? |  |

|  |  |
| --- | --- |
| Proceso | |
| Propósito | |
| Leer las novedades y generar los archivos de salida cierre de lote y transacciones | |
| Tipo de proceso | |
| Este comando es un proceso del tipo demonio, daemon o dæmon (de sus siglas en inglés Disk And Execution Monitor).  Se ejecuta en segundo plano en vez de ser controlado directamente por el usuario (es un proceso no interactivo).  Cada vez que termina su trabajo duerme un tiempo “x” y vuelve a empezar, es decir, que a menos que se detenga con Stop, este proceso no tiene condición de fin. | |
| NOTA6: los pasos enumerados a continuación son a modo indicativo, una guía, el grupo puede alterar el orden en el cual realiza los pasos siempre y cuando no se altere el resultado final esperado.  NOTA7: considerar que el contenido del archivo de novedades es siempre “liviano” “con pocos registros” | |
| Precondiciones | |
| Determinar antes de empezar a trabajar si el sistema esta inicializado, porque si no lo está el comando no puede ejecutar. Sin ambiente no hay proceso. | |
| Para un correcto testeo del script en el directorio de arribos no debe haber ningun archivo  Los archivos se depositan manualmente simulando la llegada por ftp | |
| Contar ciclos | |
| Inicializar el ciclo del proceso en 1. Ir sumando uno cada vez que se repite un ciclo. Grabar en log “voy por el ciclo xx” | |
| Lectura | |
| **Leer el directorio de arribos** y si existen novedades; verificarlas (determinar que el nombre del archivo es correcto, no este vacío, sea legible, no venga duplicado, etc)  Si no hay nada dormir un tiempo x y volver a empezar | |
| Ver si la novedad es procesable | |
| **Verificar que el nombre externo de los archivos sea Lote\_nn** donde nn es un numero de 01 al 99  Ejemplo:  Lote\_01  Lote\_02 … etc. | |
| Verificar que el archivo no este vacío | |
| Verificar que el archivo sea un archivo regular | |
| Verificar si está duplicado; es decir; si ya se procesó un archivo con idéntico nombre (esto se logra inspeccionando el directorio de procesados) | |
| Si todas las verificaciones precedentes fueron OK, **el archivo se acepta** y se lo mueve al directorio de aceptados.  Grabar en el log el nombre del archivo aceptado | |
| Si alguna de las verificaciones precedentes falla, el archivo no puede ser aceptado, entonces se lo manda completo al directorio de rechazados.  Grabar en el log el nombre del archivo rechazado y **bien en claro el motivo del rechazo** | |
| Cuando se revisaron todos los nombres de los archivos de arribos se inicia la lectura y procesamiento de cada archivo  Aun cuando no se hayan aceptado archivos en el ciclo si quedo pendiente algun archivo de otro ciclo, también hay que procesarlo | |
| Registro de Input | |
| **El archivo de novedades contiene dos** tipos de registros  Registros de transacciones  Registro de Cierre de Lote o Trailer  Un archivo puede venir solo con trailer y 0 transacciones  Un archivo no puede venir sin el registro de cierre de lote  La cantidad de transacciones debe se igual al total trx que se informa en el registro de cierre de lote  Los registros de cierre de lote se identifican con el tipo de operación CI  **Nº de Campo Descripcion Ejemplo de Transaccion Ejemplo de Cierre (a modo informativo)**  Campo 1 **Tipo de Operación** CO CI  Campo 2 **Descripcion de la Operación** Compra de 11$ en 1 cuota Cierre de Lote  Campo 3 **Total Trx**   0  Campo 4 **Fecha de Cierre de Lote**   4-->0912  Campo 5 **Año** 5-->2019 5-->2019  Campo 6 **Fecha y hora** 6-->0912134245 6-->0912133920  Campo 7 **Numero de Tarjeta** 7-->4545454500000000  Campo 8 **Vencimiento** 8-->2003  Campo 9 **Importe** 9-->000000001100  Campo 10 **Cuotas** 10-->001  Campo 11 **Trace Nº** 11-->000523 11-->000522  Campo 12 **Código de Respuesta ISO 8583** 12-->00 12-->00  Campo 13 **Retrieval Reference Number** 13-->001303000523 13-->000389133751  Campo 14 **Ticket Nº** 14-->0280  Campo 15 **Autorizacion** 15-->040303  Campo 16 **IdTrx** 1  Campo 17 **Trx relacionada**  Campo 18 **Mensaje del Host**   18-->05800000000000000000000 0000000000000000000000000000 | |
| Acerca de los datos | |
| NOTAR que algunos datos están precedidos por nro de campo guion guion mayor  Ejemplo:  4-->0912 5-->2019 6-->0912133920  Para usar la información se debe remover ese prefijo | |
| CONTROL del TRAILER (registro de cierre de Lote) | |
| Si el archivo se acepta porque su nombre es correcto ver si la cantidad de registros de transacciones coincide con lo indicado en el campo 3 del trailer (registro tipo CI)  Si la cantidad de transacciones es igual al campo 3 ir a **PROCESAR LAS TRANSACCIONES y a GRABAR EL CIERRE DE LOTE**  Si la cantidad de transacciones NO es igual al campo 3 ir a **RECHAZAR el ARCHIVO COMPLETO** y continuar con el siguiente archivo  Cualquiera sea el caso el archivo debe ser removido de aceptados y depositado en donde corresponda:   * Los que vienen OK se pasan a procesados * Los que se rechazan se pasan a rechazados | |
| **RECHAZAR el ARCHIVO COMPLETO** | |
| mover el archivo completo de aceptados a rechazados  Grabar en el log el nombre del archivo rechazado y bien en claro el motivo del rechazo: cantidad de transacciones informadas en el cierre, cantidad en el lote  Continuar con el siguiente archivo | |
| **GRABAR EL CIERRE DE LOTE** | |
| Nombre del Archivo: Cierre\_de\_Lote  Estructura y fuente  Descripción Fuente  Tipo de Operación Idem al Registro Original  Descripción de la Operación Idem al Registro Original  Total Trx Idem al Registro Original  Fecha de Cierre de Lote Idem al Registro Original  Año Idem al Registro Original  Fecha y hora Idem al Registro Original  Trace Nº Idem al Registro Original  Código de Respuesta ISO 8583 Idem al Registro Original  Retrieval Reference Number Idem al Registro Original  Mensaje del Host Idem al Registro Original  Nº Batch Campo Calculado a partir del campo 18  Cantidad de Compras Campo Calculado a partir del campo 18  Monto de compras Campo Calculado a partir del campo 18  Cantidad de Devoluciones Campo Calculado a partir del campo 18  Monto Devoluciones Campo Calculado a partir del campo 18  Cantidad de Anulaciones Campo Calculado a partir del campo 18  Monto de Anulaciones Campo Calculado a partir del campo 18 | |
| Campo 18 del registro original: Mensaje del Host; Cuando es cierre de lote este campo contiene lo siguiente:  **Descripción Atributo Longitud**  Codigo ISO an..5 5  Nº Batch an..3 3  Cantidad de Compras an..4 4  Monto de compras an..12 12  Cantidad de Devoluciones an..4 4  Monto Devoluciones an..12 12  Cantidad de Anulaciones an..4 4  Monto de Anulaciones an..12 12  Parsearlo para obtener cada uno de los campos del registro de salida. | |
| Ejemplo de registro de salida “cierre de lote”  Descripcion Ejemplo  Tipo de Operación CI  Descripcion de la Operación Cierre de Lote  Total Trx 0  Fecha de Cierre de Lote 4-->0912  Año 5-->2019  Fecha y hora 6-->0912133920  Trace Nº 11-->000522  Código de Respuesta ISO 8583 12-->00  Retrieval Reference Number 13-->000389133751  Mensaje del Host 18-->058000000000000000000000000000000000000000000000000  Nº Batch 058  Cantidad de Compras 0000  Monto de compras 0  Cantidad de Devoluciones 0000  Monto Devoluciones 0  Cantidad de Anulaciones 0000  Monto de Anulaciones 0 | |
| Grabar en el log “Batch Nº xxx grabado en cierre de lote" | |
| **PROCESAR LAS TRANSACCIONES**  Nombre del Archivo: TRX-aaaammdd  Donde aaaa= se obtiene del campo 5  Mmdd = se obtiene del campo 4  Estructura y fuente  **Descripción Fuente**  **Tipo de Operación** Idem al Registro Original  **Descripción de la Operación** Idem al Registro Original  **Año** Idem al Registro Original  **Fecha y hora** Idem al Registro Original  **Número de Tarjeta** Idem al Registro Original  **Vencimiento** Idem al Registro Original  **Importe** Idem al Registro Original  **Cuotas** Idem al Registro Original  **Trace Nº** Idem al Registro Original  **Código de Respuesta ISO 8583** Idem al Registro Original  **Retrieval Reference Number** Idem al Registro Original  **Ticket Nº** Idem al Registro Original  **Autorización** Idem al Registro Original  **IdTrx** Idem al Registro Original  **Trx relacionada** Idem al Registro Original  **Mensaje del Host** Si viene informado, idem al registro original (campo 18) Si no viene informado debe ser calculado usando el Campo 12 (Código de Respuesta ISO 8583) y la tabla CodigosISO8583: Con el código ir a la tabla y obtener la Descripción y Referencia y grabar ambos campos concatenados  **Mes del campo 6** Campo Calculado a partir del campo 6 con formato mm  **Dia del campo 6** Campo Calculado a partir del campo 6 con formato dd  **Hora del campo 6** Campo Calculado a partir del campo 6 con formato hh:mm:ss  **Monto del Campo 9** Campo Calculado a partir del campo 9: en el campo 9 (importe) los dos últimos dígitos corresponden los decimales, por lo tanto si en este campo se informa 000000001100 el monto de la transaccion es 11 pesos  Formato a elección  Ejemplo de registro de salida “transacción”  **Descripcion Ejemplo**  **Tipo de Operación** CO  **Descripcion de la Operación** Compra de 11$ en 1 cuota  **Año** 5-->2019  **Fecha y hora** 6-->0912134245  **Numero de Tarjeta** 7-->4545454500000000  **Vencimiento** 8-->2003  **Importe** 9-->000000001100  **Cuotas** 10-->001  **Trace Nº** 11-->000523  **Código de Respuesta ISO 8583** 12-->00  **Retrieval Reference Number** 13-->001303000523  **Ticket Nº** 14-->0280  **Autorizacion** 15-->040303  **IdTrx** 1  **Trx relacionada**  **Mensaje del Host** APROBADA (authno) |Operación aprobada, emitir cupón (cargo o ticket).  **Mes del campo 6** 09  **Dia del campo 6** 12  **Hora del campo 6** 13:42:45  **Monto del Campo 9** 11 | |
| **Cálculo del Mensaje del Host**  Este campo puede o no venir informado  Si viene informado, idem al registro original (campo 18) Si no viene informado debe ser calculado usando el Campo 12 (Código de Respuesta ISO 8583) y la tabla CodigosISO8583: Con el código ir a la tabla y obtener la Descripción y Referencia y grabar ambos campos concatenados  (con algún separador)  **Código de Respuesta ISO 8583** 12-->00  **Mensaje del Host** APROBADA (authno) |Operación aprobada, emitir cupón (cargo o ticket). | |
| **Cálculo de Mes Dia y hora**  A partir del Campo 6 (Fecha y hora) obtener el Mes el Dia y la hora en formato hh:mm:ss  Ejemplo:  6-->0912134245  **Mes del campo 6** 09  **Dia del campo 6** 12  **Hora del campo 6** 13:42:45 | |
| **Cálculo de Monto**  A partir del Campo 9 (Importe) obtener el Monto teniendo en cuenta que los dos ultimos digitos del importe corresponden a los decimales, por lo tanto si en este campo se informa 000000001100 el monto de la transacción es 11 pesos  Ejemplo:  **Importe** 9-->000000001100  **Monto del Campo 9** 11 | |
| Grabar en el log “Batch Nº xxx y cuantas transacciones se grabaron” | |
| Grabar log mientras se ejecuta el script | |
| Recuerde que todo lo que va sucediendo durante la ejecución se debe grabar en el log  El nombre del log debe ser distinto a los otros que se crean | |
| Evitar Reprocesos | |
| Recuerde que Cuando se procesa un archivo aceptado se lo mueve a procesados para evitar reproceso | |
| Fin de ciclo | |
| Cuando se termina el ciclo, el proceso duerme un minuto y se reinicia. | |
| Prueba | |
| 1. ¿Hay hipótesis indicando que separador de campos se emplea en el input / output ? |  |
| 1. Test: ¿el proceso no ejecuta cuando el sistema nunca fue inicializado? |  |
| 1. ¿Aclara en Hipótesis donde se realiza este control? |  |
| 1. Test: ¿se previene la ejecución de dos procesos simultáneos? |  |
| 1. ¿Aclara en Hipótesis como se realiza este control? |  |
| 1. Test: ¿cuenta los ciclos? |  |
| 1. Test: ¿detecta las novedades? |  |
| 1. Test: ¿si el nombre es incorrecto porque tiene espacios el filename lo mueve a rechazados? |  |
| 1. Test: ¿si el nombre es incorrecto por cualquier otro motivo lo mueve a rechazados? |  |
| 1. Test: ¿si el nombre está bien pero el archivo esta vacío, lo mueve a rechazados? |  |
| 1. Test: si el nombre está bien pero el archivo no es un archivo regular, de texto (es una imagen, un pdf, etc) ¿lo mueve a rechazados? |  |
| 1. Test: ¿si el nombre esta bien pero ya existe en /procesados un archivo con idéntico nombre (duplicado) lo mueve a rechazados? |  |
| 1. Test: ¿si e nombre esta bien y no viene vacio, es regular y no esta duplicado, lo mueve a aceptados? |  |
| 1. Test: ¿Cuándo procesa archivos? Siempre que haya algo para procesar (esto es lo correcto) o solo cuando el script acepta algo |  |
| 1. Test: ¿acepta un archivo con solo el tráiler y 0 transacciones cuando esto es lo que indica el cierre de lote? |  |
| 1. Test: ¿rechaza un archivo sin tráiler? |  |
| 1. Test: ¿rechaza un archivo con tráiler incorrecto? |  |
| 1. Test: si rechaza el archivo ¿lo saca de aceptados y lo mueve a rechazados? |  |
| 1. Test: ¿arma el registro de cierre de lote correctamente? |  |
| 1. Test: ¿graba cierre de lote? |  |
| 1. Test: ¿arma el registro de transacciones correctamente? |  |
| 1. Test: el nombre externo de los archivos de transacciones ¿son correctos? |  |
| 1. Test: ¿graba transacciones? |  |
| 1. Test: ¿valida bien que el Código de Respuesta ISO 8583 exista en la tabla? |  |
| 1. Test: si el código no existe en la tabla, ¿explica en la hipotesis que se hace? |  |
| 1. Test: ¿mueve de aceptados a procesados? |  |
| 1. Test: ¿duerme y arranca nuevamente otro ciclo? |  |
| 1. En el log, ¿hay líneas del progreso de la ejecución?   *Ejemplo:* “se acepta archivo xxx” “se rechaza archivo yyy; se procesa archivo zzz”  Nº de batch grabado en cierre de lote, Nº de batch y cantidad de transacciones grabadas y dónde |  |

|  |  |
| --- | --- |
| Start y Stop Puede construir una función que haga los dos trabajos o dos funciones separadas | |
| STOP  Esta función tiene por objeto detener procesos.  Es complementaria a Start  Evidenciar en un log el resultado de la operación (puede ser compartido con start)  Explicar su uso en el README | |
| START  Esta función tiene por objeto arrancar procesos.  Es complementaria a Stop  Puede ser invocada desde la línea de comando o bien desde otro comando.  No se puede arrancar un proceso si éste ya se encuentra corriendo.  No se puede arrancar un proceso si la inicialización de ambiente no fue realizada.  Evidenciar en un log el resultado de la operación  Explicar su uso en el README | |
| Prueba | |
| 1. ¿Hay Indicación correcta en el readme de cómo se ejecuta START? | SI |
| 1. ¿Hay Indicación correcta en el readme de cómo se ejecuta STOP? | SI |
| 1. ¿Funciona el Start? | SI |
| 1. ¿Funciona el Stop? | SI |
| 1. En el log, hay evidencia que realizaron varias pruebas de arranque - detención | SI, ver Evidencia 7 |

|  |  |
| --- | --- |
| Evidencias | |
| Evidencia de Procesamiento de Novedades 1 | |
| 1. ¿En el log, hay evidencia del proceso de los siguientes archivos de novedades? (todos deben contener al menos 1 registro con cualquier formato)  * Loter\_01 * Lote 01 * Lote\_001 * Lote\_00 * Lote\_199 * Lote01 * Lote-01 * lote\_01 * Miarchivo\_Lote\_01 * Lote\_02\_archi | SI, ver evidence 8a |
| Evidencia de Procesamiento de Novedades 2 | |
| 1. ¿En el log, hay evidencia del proceso de los siguientes archivos de novedades? (con el contenido indicado)  * Lote\_02 vacío * Lote\_03 conteniendo un pdf | SI, ver evidence 8b |
| Evidencia de Procesamiento de Novedades 3 | |
| 1. ¿En el log, hay evidencia del proceso de los siguientes archivos de novedades? (con el contenido indicado)  * Lote\_04 conteniendo 3 registros sin trailer * Lote\_05 conteniendo 2 registros de transacciones con tráiler indicando que tiene 3 registros | SI, ver evidence 8c |
| Evidencia de Procesamiento de Novedades 4 | |
| 1. ¿En el log, hay evidencia del proceso de los siguientes archivos de novedades? casos de prueba construidos por el grupo en donde  * Se manda a procesar un lote donde un registro tiene un Código de Respuesta ISO 8583 inexistente * Se manda a procesar un lote donde el campo 4 viene sin datos * Se manda a procesar un lote donde el campo 5 viene sin datos * Se manda a procesar un lote donde el campo 6 viene sin datos * Se manda a procesar un registro con formato incorrecto   NOTA: en hipotesis documente que hacen en estos casos y cualquier otra circunstancia de error. | SI, ver evidence 8d |
| 1. ¿Envió todos los casos de prueba usados en las pruebas 1 2 3 4? | SI |
| Evidencia de Procesamiento de Novedades 5 | |
| 1. ¿En el log, hay evidencia del proceso de los siguientes archivos de novedades? casos de prueba remitidos por la catedra   Lote\_11 al 20 | SI |
| 1. Verificar Líneas de log del procesamiento correcto de estos archivos | SI |
| Volver a depositar los archivos   * Lote\_12 y Lote\_13  1. Verificar líneas de log del NO procesamiento de estos archivos | SI (rechazados por duplicados) |