



CARÁTULA

Tema	Grupo	Ayudante		Evaluación Grupal del TP
A	1-3-5-7-9-11		Nota	
			Fecha	
			Firma	

INTEGRANTES

	Padrón	Apellido y Nombre	Asistencia a Entrega	Asistencia a Revisión	Evaluación Individual Final
1					
2					
3					
4					
5					
6					

PLANILLA DE EVALUACIÓN

PRUEBA _____	Hora Inicio		Hora Fin		Fecha:	
	Aprobado	A Revisión	No posee	Responsable, Observaciones		
Trajo Carpeta?						
Caratula completa? Dos?						
Hojas sueltas?						
Indice, Pie y nro de hoja??						
Hipótesis? Problemas?						
Esquema de ejecución?						
Nuevas funciones/ archivos?						
Datos Impresos para seg?						
Documentó el camino feliz?						
Trajo dispositivo listo?						
Readme ok?						
Arranca el Instalar_TP?						
Permite cambiar el default?						
Recuerda cambios default?						
Hace todos los chequeos?						
Crea directorios 1 sola vez?						
Graba log? Graba Conf?						
Reinstala?						
Arranca Iniciar_A?						
Verifica todo ok?						
Setea permisos?						



Setea variables?				
Reinicia?				
Graba log?				
Arranca aut. Recibir_A?				
Acepta reservas x sala?				
Acepta reservas x obra?				
Acepta invitados?				
Rechaza x _____ ?				
Rechaza x _____ ?				
Rechaza x _____ ?				
Rechaza x _____ ?				
Graba Log?				
Arranca aut- Reservar_A?				
Rechaza x fecha inv. ?				
Rechaza x fecha ant. ?				
Rechaza x fech.venc. ?				
Rechaza x fech tardia ?				
Rechaza x hora ?				
Valida sala-fech-hora?				
Valida obra-fech-hora?				
Rechaza x disp ?				
Genera OK?				
Genera NOK ?				
Actualiza DISP ?				
Mueve a Procesados?				
Detecta duplicados?				
Graba Log?				
Imprimir_A				
-a (ayuda)				
-i (invitados)				
-d (disponibilidad)				
-r (ranking)				
-t (ticket)				
-w (grabar)				
Completó el camino feliz?				
Funciones				
Start_A -Stop_A				
Mover_A				
Grabar_L				



Índice

Enunciado – Tema A	4
Introducción	4
Archivos y Estructuras.....	9
Archivo de Configuración	9
Archivos de Log	9
Archivo Maestros	9
Archivo de Disponibilidad	10
Directorio de Arribos	11
Directorio de Rechazos.....	11
Archivos de Reservas	11
Archivos de Invitados	13
Listados	13
Especificación de Comandos	15
Instalación: Instalar_TP	15
Inicialización: Iniciar_A	23
Detección y Recepción de Archivos Externos: Recibir_A	25
Confirmación de Reservas: Reservar_A	28
Impresión de Listados, Tickets, Vales: Imprimir_A	32
Especificación de Funciones.....	35
Start_A	35
Stop_A	35
Mover_A	36
Grabar_L	37



Enunciado – Tema A

Introducción

Los empresarios teatrales desean unificar el sistema de reservas de entradas sin valor comercial, para las funciones de prensa, ensayos, entradas para cartelera, productores, etc.

Para ello emplean el sistema [RESER_A](#) que les permite ir confirmando las reservas solicitadas por los distintos integrantes del negocio teatral.

Instalación del Sistema

El sistema software desarrollado debe tener un comando para su instalación.

Esta actividad debe llevarse a cabo con el comando [Instalar_TP](#)

Inicialización del entorno de ejecución del Sistema

El Proceso se inicia con el aseguramiento de la disponibilidad de la información para llevar adelante el proceso total: Es indispensable contar con el archivo de configuración, los comandos, archivos maestros, disponibilidad, con los permisos adecuados.

Continúa con la asignación de valor a un conjunto de variables de ambiente y luego ofrece arrancar automáticamente el comando de recepción de archivos.

Todas estas actividades deben llevarse a cabo con el comando [Iniciar_A](#).

Detección y Recepción de Archivos Externos

Luego tenemos el proceso de recepción de los archivos externos. Cada empresario o productor envía archivos solicitando reservas para cierto evento o listados con los nombres de los invitados.

Si el nombre del archivo (filename) es del tipo esperado, acepta el archivo, de lo contrario, lo rechaza.

También en el caso de contar con archivos de reservas solicitadas “aceptadas”, arranca el siguiente comando para que las confirme

Todas estas actividades deben llevarse a cabo con el comando [Recibir_A](#).

Confirmación de Reservas

Las reservas solicitadas “aceptadas” deben ser confirmadas si se cumplen ciertas condiciones (oportunidad, disponibilidad, etc.)

Las reservas confirmadas se graban en una nueva estructura con mas datos dejándolas listas para el siguiente comando.

Las reservas rechazadas se graban en una nueva estructura que además de los datos originales tiene el motivo del rechazo.

Todas estas actividades deben llevarse a cabo con el comando [Reservar_A](#).

Impresión de Listados y Tickets

Por último tenemos un proceso que nos permite realizar consultas de disponibilidad, emitir el ranking de los 10 principales, generar la lista de invitados para las reservas confirmadas, generar los archivos para imprimir los tickets de entrada.

Todas estas actividades deben llevarse a cabo con un comando PERL denominado [Imprimir_A](#).



Desarrollar el sistema software RESER_A con todos sus componentes

Se requiere elaborar una serie de comandos que trabajen en forma integrada, no deben ser comandos independientes ya que la naturaleza del TP es que desarrollen UN SISTEMA.

También deben elaborar un comando para instalar todo y varias funciones que centralizan utilidades compartidas por varios comandos.

Los elementos a desarrollar son:

- Shell script denominado que **Iniciar_A** efectúa la inicialización de ambiente para la correcta ejecución del TP
- Shell script denominado **Recibir_A** que detecta el arribo de reservas
- Shell script denominado **Reservar_A** que confirma las reservas
- Programa Perl denominado **Imprimir_A** que genera los listados para impresión.
- Función (en Shell o en Perl) denominada **Mover_A** que se emplea para mover archivos
- Función (en Shell o en Perl) denominada **Grabar_L** que se emplea para grabar los archivos de log
- Función en Shell script denominada **Stop_A** que se emplea para detener procesos y otra complementaria **Start_A** que permite disparar procesos
- Shell script denominado **Instalar_TP** que permite la instalación de todo lo anterior

Elaborar la documentación del sistema RESER_A

Se requiere elaborar la documentación del sistema y armar una carpeta para entregar el día de la corrección al ayudante designado con los siguientes elementos:

1. Carátula

La entregada en este mismo documento con los datos completos en 2 COPIAS una para el grupo y otra para el docente.

2. Planillas de Evaluación

Las entregadas en este mismo documento

3. Índice del Contenido de la Carpeta.

El número de página puede ser incorporado manualmente luego de numerar las hojas

4. Hipótesis y Aclaraciones Globales

Describe las hipótesis que ha considerado para la resolución del TP. Documente cualquier otra aclaración que se considere necesaria. Todas las hipótesis deben presentarse en este punto, puede agruparlas por comando.

5. Problemas relevantes

Describe los problemas relevantes que se hayan presentado durante el desarrollo, la integración y/o la prueba del sistema. Explique cómo fueron solucionados

6. Archivo README

Incluya la impresión del README en la carpeta. Ver detalles en "Recomendaciones para el equipo de integración y testing" y en la especificación del instalador.

7. Esquema de Ejecución

Brinde un esquema de secuencia de los comandos principales del sistema que explique cómo se vinculan entre ellos y que input/output manejan

8. Listado de Nuevas Funciones y/o Comandos Auxiliares

Brinde un listado de las nuevas funciones y/o comandos auxiliares creados por Ustedes, es decir, que no figuran en el enunciado original del TP.



Indique: Nombre de la función, quienes la usan, para que la usan.

Si no crea ninguna, indique: NINGUNA

9. Listado de Nuevos Archivos

Brinde un listado de los nuevos archivos creados por Ustedes, es decir, que no figuran en el enunciado original del TP.

Indique: Nombre del archivo, si es temporal o permanente, donde lo almacenan, quienes lo usan, para que lo usan.

Si no crea ninguno, indique: NINGUNO

10. Listado de DATOS

Brinde un listado de los datos de los archivos salas, obras, combos, reservas (algunos que vaya a emplear en la prueba del camino feliz).

11. Hoja de ruta de prueba “camino feliz”

Instale el tp

Imprima el log de instalación y el contenido del archivo de configuración

Inicialice el ambiente, y permita que el demonio arranque

Imprima el log de [Iniciar_A](#)

Tome un par de archivos de reservas solicitadas y de lista de invitados que tengan nombres aceptables y deposítelos en ARRIDIR

Imprima el log de [Recibir_A](#) y el contenido de los archivos aceptados

Imprima el log de [Reservar_A](#) y el contenido de los archivos combos.dis, reservas.ok, reservas.nok

Ejecute [Imprimir_A](#) con todas las variantes que considere adecuadas, imprima las invocaciones y los distintos resultados obtenidos.

12. Apéndice

Incluya el ENUNCIADO completo del TP (sin la caratula ni las planillas de evaluación dado que ya fueron incluidas al principio de la carpeta)

Evaluación

El día de vencimiento del TP, cada ayudante convocará a los integrantes de un grupo e iniciará la corrección mediante una entrevista grupal. Es imprescindible la presencia de todos los integrantes del grupo el día de la corrección. El objetivo de esto es comprender la dinámica de trabajo del equipo y los roles que ha desempeñado cada integrante del equipo.

Se evaluará el trabajo grupal y a cada integrante en forma individual.

Para que el alumno apruebe el trabajo práctico debe estar aprobado en los dos aspectos.

Dentro de los ítems a chequear el ayudante evaluará aspectos formales (como ser la forma de presentación de la carpeta), aspectos funcionales: que se resuelva el problema planteado y aspectos operativos: que el TP funcione integrado.

Solo existen dos fechas disponibles por grupo para la corrección del TP, la fecha de vencimiento y otra adicional. Para aprovecharlas, efectúen la prueba integrada ANTES de la fecha de vencimiento del TP en el entorno de la facultad

Carpeta

La documentación debe entregarse en una carpeta con TODAS las hojas numeradas y enganchadas. Las hojas sueltas no se considerarán como parte de la misma

El pie de página de cada hoja debe tener: Número de Grupo y Tema (en el margen izquierdo) y Número de Hoja (en el margen derecho). **La numeración puede ser manual**



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”. El instalable deberá contener:

- El archivo README
- Los scripts desarrollados
- Los Archivos de prueba, maestros y tablas de configuración
- Los Archivos de salida de las Pruebas ejecutadas
- Otros casos de prueba
- La documentación
- Cualquier otro archivo que se considere necesario

MUY IMPORTANTE: cuando se efectúa el “tar” se debe usar la opción de “directorio relativo” para evitar problemas cuando se proceda a la instalación en los laboratorios de la facultad.

Recomendaciones para el equipo de desarrollo

1. Se deberá tener en cuenta para la resolución TODAS las condiciones que se enuncian.
2. Se deben respetar los formatos de archivos especificados
3. Se debe respetar la estructura de directorios planteada
4. Se debe evitar el uso de archivos auxiliares permanentes
5. Si se emplean archivos auxiliares temporales, se deben eliminar ANTES de finalizar la ejecución del comando
6. Código de Retorno

Toda invocación desde un comando a otro debe devolver un código de retorno cero (0) si fue exitoso o distinto de cero si tuvo errores. Siempre al finalizar el comando se debe loguear si finalizó correctamente o con errores.

7. Movimiento de Archivos

En líneas generales no se borra ningún archivo de datos, se los mueve de un lugar a otro para asegurar la integridad de la información original. Se solicita una función de librería [Mover_A](#) para el movimiento de archivos de datos la cual debe ser empleada por todos los comandos que la requieran

8. Manejo de errores, logueo

Todo evento que genera algún tipo de error debe ser logueado. Si el comando no graba en un log específico, muestre el error por pantalla. Evite detener el proceso de evaluación del TP por no poder detectar que está pasando. También puede incorporar al comando una señal que permita dejar pistas de auditoría y se enciende si es necesario.

Recomendaciones para el equipo de integración y testing

1. Prevean efectuar la prueba de integración varios días antes de la entrega del TP dado que usualmente surgen errores de comunicación entre los comandos encadenados. Recuerden que la entrega del TP INCLUYE que EN LA DOCUMENTACION muestren el resultado de una prueba de integración exitosa
2. El TP debe correr en las instalaciones de la FIUBA, por lo tanto una integración en sus equipos personales no garantiza que dentro del entorno de la FIUBA la integración muestre los mismos resultados. Ejemplo de esto es el retorno que da el comando ps.
3. Recuerde que dentro del entorno de ejecución de la FIUBA su usuario NUNCA puede ser ROOT
4. Lotes de Prueba



Se proveerán los archivos maestros, el archivo de disponibilidad, algunos archivos de reservas solicitadas y otros de listas de invitados todos ellos con un alto porcentaje de información libre de error. El grupo debe traer un juego extra de archivos de reservas con casos lo suficientemente heterogéneos como para contemplar todas las variantes de ejecución, en particular las de rechazo o error

5. Archivo README, responsabilidad del equipo de integración

Como parte de la documentación del sistema se debe proveer un archivo README en donde se brinden indicaciones de instalación y de ejecución de los comandos, como ser:

- a. Una explicación de cómo copiar desde un medio externo el instalable, a modo de ejemplo:
 - Insertar el dispositivo de almacenamiento con el contenido del tp
 - Crear en el directorio corriente un directorio de trabajo
 - Copiar el archivo *.tgz en ese directorio
 - Descomprimir el *.tgz de manera de generar un *.tar
 - Extraer los archivos del tar.
 - ... etc.
- b. Instrucciones de instalación
- c. Que se requiere para poder instalar, Que nos deja la instalación y donde
- d. Cuáles son los primeros pasos para poder correr el paquete una vez instalado
- e. Que comprobaciones se pueden hacer para asegurar que todo está en condiciones para empezar
- f. Como frenar la ejecución de comandos



Archivos y Estructuras

Archivo de Configuración

CONFDIR/Instalar_TP.conf

Campos	Descripción	Fuente/Valor
Variable	Caracteres	Valores posibles: GRUPO, CONFDIR, BINDIR, MAEDIR, ARRDIR, DATASIZE, ACEPDIR, RECHDIR, PROCDIR, REPODIR, LOGDIR, LOGEXT, LOGSIZE
Valor	Caracteres	Contenido de la variable al momento de la grabación.
Usuario	Caracteres	Es el login del usuario que graba el registro
Fecha	Fecha y hora Formato a Elección	Es la fecha y hora en el momento de grabación del registro.

Separador de campos: = igual

Ejemplo: GRUPO=/usr/alumnos/temp/grupo01=alumnos=30/09/2013 10:03 p.m

Se debe grabar un registro por cada variable. Luego de los registros requeridos, puede agregar todos los registros que desee, es decir que a partir de LOGSIZE, el uso del archivo de configuración es de libre disponibilidad para los desarrolladores.

Archivos de Log

Log de la Instalación: CONFDIR/Instalar_TP.log

Logs de Comandos: LOGDIR/<comando>.logext

Campo	Descripción	Fuente/Valor
Cuando	Fecha y hora	Formato a Elección Es la fecha y hora en el momento de grabación del registro.
Quien	Caracteres	Es el login del usuario que graba el registro
Donde	Caracteres	Nombre del Comando, función o rutina en donde se produce el evento que se registra en el log
Que	Caracteres	Lo determina el programador. Valores Posibles: I,W,E,SE
Porque	Hasta 120 Caracteres	Lo determina el programador.

Separador de campos: - guion

Ejemplo: 20130509 10:53:22-alumnos-Iniciar_A-W-No se pudo arrancar [Recibir_A](#).

Archivo Maestros

Maestro de Salas: MAEDIR/salas.mae

Estructura		
campo1	ID DE LA SALA	numérico, siempre es un número par
campo2	NOMBRE DE LA SALA	caracteres



campo3	CAPACIDAD de la SALA	numérico
campo4	DIRECCIÓN	caracteres
campo5	TELEFONO	caracteres
campo6	CORREO DEL TEATRO	caracteres

Separador de campos: ; punto y coma

Ejemplo:

ID DE LA SALA	NOMBRE DE LA SALA	CAPACIDAD de la SALA	DIRECCIÓN	TELEFONO	CORREO DEL TEATRO
2	Teatro El Fino	840	Paraná 673 primer piso	4307-1965	Teatro.El.Fino@reservas.com.ar
6	Teatro Corrientes Azul	860	Av. Corrientes 5965	4307-1967	Teatro.Corrientes.Azul@reservas.com.ar
8	Teatro Belisario	870	Av. Corrientes 1624	4373-3465	Teatro.Belisario@reservas.com.ar
12	Teatro Espacio Abierto	890	Pasaje Carabelas 255	4328-1903	Teatro.Espacio.Abierto@reservas.com.ar
14	Teatro La tertulia	900	Gallo 826	4865-0303	Teatro.La.tertulia@reservas.com.ar
18	Teatro Payro	920	San Martin 766	4312-5922	Teatro.Payro@reservas.com.ar

Maestro de Obras: MAEDIR/obras.mae

Estructura

campo1	ID DE LA OBRA	numérico, siempre es un número impar
campo2	NOMBRE DE LA OBRA	caracteres
campo3	CORREO PRODUCCION GENERAL	caracteres
campo4	CORREO PRODUCCION EJECUTIVA	caracteres

Separador de campos: ; punto y coma

Ejemplo:

ID DE LA OBRA	NOMBRE DE LA OBRA	CORREO PRODUCCION GENERAL	CORREO PRODUCCION EJECUTIVA
1	Una historia para cartucho	MORA_ANTONIO@HOTMAIL.COM	LEYER_DEMIAN@GMAIL.COM
3	En busca de Noé	ALONSO_EDITH@GMAIL.COM	BEIRO_CLAUDIO@GMAIL.COM
5	Carkalata	PEREZ_JORGE@GMAIL.COM	BEIRO_CLAUDIO@GMAIL.COM
11	Purolucro rompe todo	VERNIS_VICTOR@HOTMAIL.COM	FERREYRA_PAULA@GMAIL.COM
13	Tokitos	VERNIS_VICTOR@HOTMAIL.COM	DANIELAN_HERNAN@GMAIL.COM

Archivo de Disponibilidad

PROCDIR/combo.dis

Estructura

campo1	ID DEL COMBO	Numérico (clave)
campo2	ID DE LA OBRA	numérico
campo3	FECHA DE FUNCIÓN	formato: día/mes/año
campo4	HORA DE FUNCIÓN	formato: hh:mm
campo5	ID DE LA SALA	numérico
campo6	BUTACAS HABILITADAS	numérico
campo7	BUTACAS DISPONIBLES	numérico
campo8	REQUISITOS ESPECIALES	caracteres

Separador de campos: ; punto y coma

Ejemplo:



ID DEL COMBO	ID DE LA OBRA	FECHA DE FUNCIÓN	HORA DE FUNCIÓN	ID DE LA SALA	BUTACAS HABILITADAS	BUTACAS DISPONIBLES	REQUISITOS ESPECIALES
C00010018	1	17/10/2013	17:00	18	920	113	NINGUNO
C00010019	1	18/10/2013	17:00	18	920	0	NINGUNO
C00010020	1	19/10/2013	19:00	18	920	0	NINGUNO
C00010021	1	20/10/2013	16:00	18	920	0	NINGUNO
C00010022	1	24/10/2013	17:00	18	920	0	NINGUNO
C00010023	1	25/10/2013	17:00	18	920	500	NINGUNO
C00010024	1	26/10/2013	19:00	18	920	0	NINGUNO
C00010025	1	27/10/2013	16:00	18	920	0	NINGUNO
C00010026	3	02/11/2013	18:00	12	850	20	NINGUNO
C00010027	3	02/11/2013	21:00	12	850	0	NINGUNO
C00010028	3	03/11/2013	18:00	12	850	0	NINGUNO
C00010029	5	03/11/2013	16:00	6	800	50	NINGUNO

Directorio de Arribos

ARRIDIR/ nombre del archivo

Directorio de Rechazos

RECHDIR/ nombre del archivo

Archivos de Reservas

Reservas Aceptados: ACEPDIR/ nombre del archivo

Reservas Procesados: PROCDIR/ nombre del archivo

Reservas solicitadas por producción: ID de la OBRA-CORREO-xxx

Estructura

campo1	REFERENCIA INTERNA DEL SOLICITANTE	caracteres	opcional
campo2	FECHA DE FUNCIÓN	formato: día/mes/año	obligatorio
campo3	HORA DE FUNCIÓN	formato: hh:mm	obligatorio
campo4	NRO. DE FILA	caracteres	opcional
campo5	NRO. DE BUTACA	caracteres	opcional
campo6	CANTIDAD DE BUTACAS SOLICITADAS	numérico	obligatorio
campo7	SECCION	caracteres	opcional

Separador de campos: ; punto y coma

Ejemplos:

Archivo ejemplo a) 1-MORA_ANTONIO@HOTMAIL.COM-UnaHistoria_del34al40

REFERENCIA INTERNA DEL SOLICITANTE	FECHA DE FUNCIÓN	HORA DE FUNCIÓN	NRO. DE FILA	NRO. DE BUTACA	CANTIDAD DE BUTACAS SOLICITADAS	SECCION
1234	18/10/2013	17:00			2	Platea
1235	18/10/2013	17:00			2	Platea
1236	18/10/2013	17:00			2	Platea
1237	18/10/2013	17:00			2	Platea
1238	18/10/2013	17:00			2	Platea
1239	18/10/2013	17:00			2	Platea
1240	18/10/2013	17:00			2	Platea

Archivo ejemplo b) 1-LEYER_DEMIAN@GMAIL.COM-EstrenoenelPayro333



REFERENCIA INTERNA DEL SOLICITANTE	FECHA DE FUNCIÓN	HORA DE FUNCIÓN	NRO. DE FILA	NRO. DE BUTACA	CANTIDAD DE BUTACAS SOLICITADAS	SECCION
333	18/10/2013	17:00			4	
333	18/10/2013	17:00			4	
333	18/10/2013	17:00			4	
333	18/10/2013	17:00			4	
333	18/10/2013	17:00			4	

Reservas solicitadas por Teatros: ID de la SALA-CORREO-xxx

Estructura

campo1	REFERENCIA INTERNA DEL SOLICITANTE	caracteres	opcional
campo2	FECHA DE FUNCIÓN	formato: día/mes/año	obligatorio
campo3	HORA DE FUNCIÓN	formato: hh:mm	obligatorio
campo4	NRO. DE FILA	caracteres	opcional
campo5	NRO. DE BUTACA	caracteres	opcional
campo6	CANTIDAD DE BUTACAS SOLICITADAS	numérico	obligatorio
campo7	SECCION	caracteres	opcional

Separador de campos: ; punto y coma

Ejemplo: archivo 18-Teatro.Payro@reservas.com.ar-PAyro181013a

REFERENCIA INTERNA DEL SOLICITANTE	FECHA DE FUNCIÓN	HORA DE FUNCIÓN	NRO. DE FILA	NRO. DE BUTACA	CANTIDAD DE BUTACAS SOLICITADAS	SECCION
	18/10/2013	17:00			4	Platea
	18/10/2013	17:00			4	Platea
	18/10/2013	17:00			4	Platea

Reservas Confirmadas: PROCDIR/reservas.ok

Estructura

campo1	ID de la OBRA
Campo2	NOMBRE DE LA OBRA
Campo3	FECHA DE FUNCIÓN
Campo4	HORA DE FUNCIÓN
Campo5	ID de la SALA
Campo6	NOMBRE DE LA SALA
Campo7	CANTIDAD DE BUTACAS CONFIRMADAS
Campo8	ID del COMBO
Campo9	REFERENCIA INTERNA DEL SOLICITANTE
Campo10	CANTIDAD DE BUTACAS SOLICITADAS
Campo11	CORREO DEL SOLICITANTE
campo12	USUARIO GRABACION
campo13	FECHA GRABACION

Separador de campos: ; punto y coma



Reservas Rechazadas: PROCDIR/reservas.nok

Estructura

campo1	REFERENCIA INTERNA DEL SOLICITANTE
campo2	FECHA DE FUNCIÓN
campo3	HORA DE FUNCIÓN
campo4	NRO. DE FILA
campo5	NRO. DE BUTACA
campo6	CANTIDAD DE BUTACAS SOLICITADAS
campo7	SECCION
Campo8	MOTIVO
campo9	ID de la SALA
campo10	ID de la OBRA
campo11	CORREO DEL SOLICITANTE
campo12	USUARIO GRABACION
campo13	FECHA GRABACION

Separador de campos: ; punto y coma

Archivos de Invitados

Invitados Aceptados: REPODIR/<Referencia Interna Del Solicitante>.inv

Estructura

campo1	INVITADO	caracteres	obligatorio
campo2	EMPRESA	caracteres	opcional
campo3	CANTIDAD DE ACOMPAÑANTES	numérico	opcional

Separador de campos: ; punto y coma

Ejemplo:

INVITADO	EMPRESA	CANTIDAD DE ACOMPAÑANTES
YOLI GABRIEL	PEUGEOT	1
ILVA MARIANO	PEUGEOT	1
REJO EDGARDO	PEUGEOT	1
ALLE DANIEL		
GRECO EZEQUIEL		
ABIKU JAVIER		

Listados

Listado de Invitados Confirmados: REPODIR/<id del combo>.inv

Campo	Descripción	Fuente
Línea	Caracteres	Proceso Imprimir_A opción -i

Separador de campos: no posee



Listado de Disponibilidad: REPODIR/<nombre ingresado>.dis

Campo	Descripción	Fuente
Línea	Caracteres	Proceso Imprimir_A opción -d

Separador de campos: no posee

Ranking: REPODIR/ranking.nnn

Campo	Descripción	Fuente
Línea	Caracteres	Proceso Imprimir_A opción -r

Separador de campos: no posee

Tickets: REPODIR/<Id Del Combo>.tck

Estructura		
Campo1	TIPO de COMPROBANTE	Proceso Imprimir_A opción -t
Campo2	NOMBRE DE LA OBRA	Del archivo de reservas confirmadas
Campo3	FECHA DE FUNCIÓN	Del archivo de reservas confirmadas
Campo4	HORA DE FUNCIÓN	Del archivo de reservas confirmadas
Campo5	NOMBRE DE LA SALA	Del archivo de reservas confirmadas
Campo7	REFERENCIA INTERNA DEL SOLICITANTE	Del archivo de reservas confirmadas
Campo8	CORREO DEL SOLICITANTE	Del archivo de reservas confirmadas

Separador de campos: , coma



Especificación de Comandos

A continuación se brinda una descripción de cada comando solicitado.

En esta descripción se sugieren los pasos de ejecución solo a los efectos de ordenar la explicación.

Estos pasos deben considerarse indicativos. Si el equipo de desarrollo lo considera pertinente, puede modificarlos tanto sea en el orden de ejecución como en la forma de resolverlo, siempre y cuando esto no afecte el resultado final esperado.

Instalación: *Instalar_TP*

Opciones y Parámetros

- A especificar por el desarrollador
- Explicar claramente su invocación y uso en el README.

Descripción

El propósito de este comando es efectuar la instalación del sistema [RESER_A](#)

Cuando este script interactúa con el usuario, los valores por default propuestos se deberán mostrar entre paréntesis ().

Si durante la instalación el hilo de ejecución obliga a reiniciar los pasos anteriores, los valores default propuestos inicialmente deben ser reemplazados por los nuevos valores ingresados por el usuario anteriormente en ese mismo paso, es decir, el script debe tener “memoria” durante su ejecución.

Hay un único usuario disponible para la evaluación del TP, debido a esto cada grupo deberá emplear su propio subdirectorio de trabajo llamado grupoxx (donde xx es el nro de grupo).

Antes de comenzar la instalación asegúrese que:

- la última versión del TP está grabada en el dispositivo externo
- que la terminal que se usara para la evaluación posea interfaz para la conexión del dispositivo

Una vez que conecte el dispositivo externo, deberá copiar su contenido a su subdirectorio de trabajo.

En general los directorios pueden ubicarse en cualquier lugar luego de \$grupo, excepto el subdirectorio de configuración: La ubicación del directorio de configuración no puede ser variable, debe estar fija para permitirle al comando de inicialización de ambiente encontrar el archivo de configuración, por lo cual se pide que la extracción genere AL MENOS el siguiente directorio:
...../grupoxx/conf

Para simplificar esta narrativa de aquí en mas a todo el camino (path) que va desde la raíz hasta [grupoxx](#) lo llamaremos genéricamente “\$grupo”, a \$grupo/conf lo llamaremos **CONFDIR**, etc

Pasos sugeridos

1. Inicializar archivo de log

Este comando graba un archivo de Log cuyo nombre es [Instalar_TP](#).log en el directorio **CONFDIR**.

Si el archivo de log no existe, crearlo. Si existe, agregarle los nuevos registros.

Un log de instalación debe permitir recrear en forma completa el proceso de instalación, es decir: las condiciones antes de la instalación, las decisiones tomadas durante la instalación, las condiciones finales.

La escritura en el log de instalación debe respetar las condiciones indicadas en la función [Grabar_L](#)

El primer mensaje que se debe grabar es el mensaje de inicio de ejecución (todos los mensajes que se muestran son a modo de ejemplo)



Inicio de Ejecución

2. Mostrar (y grabar en el log) donde se graba el log de la instalación

Log del Comando Instalar_TP: **CONFDIR**/Instalar_TP.log

3. Mostrar (y grabar en el log) el nombre del directorio de configuración

Directorio de Configuración: **CONFDIR**

4. Detectar si el paquete RESER_A o alguno de sus componentes ya está instalado

Este script de instalación se debe preparar de manera tal que detecte si el paquete o alguno de sus componentes ya se encuentran instalados

Una forma de hacer esto es verificando la existencia del archivo Instalar_TP.conf en el directorio

CONFDIR

- Si Instalar_TP.conf no existe, asumir que el paquete no fue instalado. El proceso sigue en el siguiente punto
- Si Instalar_TP.conf existe, asumir que el paquete ya fue instalado y verificar que la instalación este completa, o indicar los faltantes

4.1. Si esta completo, mostrar y grabar en el log los siguientes mensajes y luego ir a FIN

```
TP SO7508 Segundo Cuatrimestre 2013. Tema A Copyright © Grupo xx
Librería del Sistema: CONFDIR (mostrar path y listar archivos)
Ejecutables: BINDIR (mostrar path y listar archivos)
Archivos maestros: MAEDIR (mostrar path y listar archivos)
Directorio de arriba de archivos externos: ARRIDIR
Archivos externos aceptados: ACEPDIR
Archivos externos rechazados: RECHDIR
Reportes de salida: REPODIR
Archivos procesados: PROCDIR
Logs de auditoría del Sistema: LOGDIR/<comando>.LOGEXT
Estado de la instalación: COMPLETA
Proceso de Instalación Cancelado
```

4.2. Si falta instalar algún componente, mostrar y grabar en el log los siguientes mensajes:

```
TP SO7508 Segundo Cuatrimestre 2013. Tema A Copyright © Grupo xx
Librería del Sistema: CONFDIR (mostrar path y listar archivos)
Ejecutables: BINDIR (mostrar path y listar archivos)
Archivos maestros: MAEDIR (mostrar path y listar archivos)
Directorio de arriba de archivos externos: ARRIDIR
Archivos externos aceptados: ACEPDIR
Archivos externos rechazados: RECHDIR
Reportes de salida: REPODIR
```




```
Archivos procesados: PROCDIR  
Logs de auditoria del Sistema: LOGDIR/<comando>.LOGEXT  
Componentes faltantes: listado de los componentes faltantes  
Estado de la instalación: INCOMPLETA  
Desea completar la instalación? (Si-No)
```

4.3. Si el usuario indica Si:

4.3.1. Comprobar que se dispone en algun lado de los “faltantes”. Indicar en las hipótesis dónde preservan el paquete original para usar en estas circunstancias.

4.3.2. Si por alguna circunstancia no poseen lo necesario para completar la instalación, mostrar un mensaje explicativo, brindar indicaciones, ir a FIN

4.3.3. Chequear que Perl esté instalado como se explica en el punto 6

4.3.4. Brindar, si es necesario, indicaciones al usuario instalador, explicar en qué lugar se instalarán los componentes (el usuario no podrá cambiar estos valores), Mostrar los valores de instalación

```
TP SO7508 Segundo Cuatrimestre 2013. Tema A Copyright © Grupo xx  
Librería del Sistema: CONFDIR  
Ejecutables: BINDIR  
Archivos maestros: MAEDIR  
Directorio de arribo de archivos externos: ARRIDIR  
Espacio mínimo libre para arribos: DATASIZE Mb  
Archivos externos aceptados: ACEPDIR  
Archivos externos rechazados: RECHDIR  
Archivos procesados: PROCDIR  
Reportes de salida: REPODIR  
Logs de auditoria del Sistema: LOGDIR/<comando>.LOGEXT  
Tamaño máximo para los archivos de log del sistema: LOGSIZE Kb  
Estado de la instalacion: LISTA
```

4.3.5. Continuar en el paso: “Confirmar Inicio de Instalación”

4.4. Si el usuario indica No, ir a **FIN**

4.5. Si el paquete no fue instalado, continuar en el siguiente punto

5. Aceptación de términos y condiciones

Mostrar en consola el siguiente mensaje:

```
TP SO7508 Segundo Cuatrimestre 2013. Tema A Copyright © Grupo xx
```



A T E N C I O N: Al instalar TP SO7508 Segundo Cuatrimestre 2013 UD. expresa aceptar los términos y Condiciones del "ACUERDO DE LICENCIA DE SOFTWARE" incluido en este paquete.

Acepta? Si - No

- 5.1. Si el usuario indica No, ir a **FIN**
- 5.2. Si el usuario indica Si, continuar
- 5.3. Cualquier otra cosa, seguir en este punto

6. Chequear que Perl esté instalado

Para la instalación de TP SO7508 Segundo Cuatrimestre 2013 es necesario chequear que Perl versión 5 o superior este instalado.

- 6.1. Si al chequear esto da error, entonces mostrar y grabar en el log:

```
TP SO7508 Segundo Cuatrimestre 2013. Tema A Copyright © Grupo xx
Para instalar el TP es necesario contar con Perl 5 o superior
instalado. Efectúe su instalación e inténtelo nuevamente.
Proceso de Instalación Cancelado
```

- 6.1.1. ir a **FIN**

- 6.2. Si Perl está instalado, mostrar y grabar en el log un mensaje informativo con la versión de perl que se encuentra instalada y continuar.

```
TP SO7508 Segundo Cuatrimestre 2013. Tema A Copyright © Grupo xx
Perl Version:
```

7. Definir el directorio de instalación de los ejecutables

Mostrar y grabar en el log el siguiente mensaje con su respuesta:

```
Defina el directorio de instalación de los ejecutables ($grupo/bin):
```

Proponer /bin y si el usuario lo desea cambiar, permitiréelo.

El usuario puede ingresar un nombre simple como "bin" o un subdirectorio como /tp/sistemas/bin

Reservar este path en la variable **BINDIR**

8. Definir el directorio de instalación de los archivos maestros

Mostrar y grabar en el log el siguiente mensaje con su respuesta:

```
Defina el directorio de instalación de los archivos maestros
($grupo/mae):
```

Proponer /mae y si el usuario lo desea cambiar, permitiréelo.

El usuario puede ingresar un nombre simple como "mae" o un subdirectorio como /data/mae

Reservar este path en la variable **MAEDIR**

9. Definir el directorio de arribo de archivos externos

Se debe solicitar al usuario que indique cual es el directorio que permite el arribo de archivos externos. Mostrar y grabar en el log el siguiente mensaje con su respuesta:



```
Defina el directorio de arribo de archivos externos ($grupo/arribos):
```

Proponer /arribos y si el usuario lo desea cambiar, permitirle.

El usuario puede ingresar un nombre simple como "/arribos" o un subdirectorio como /archivos/externos/arribos

Reservar este path en la variable **ARRIDIR**

10. Definir el espacio mínimo libre para el arribo de archivos externos

Se debe solicitar al usuario que indique cual es el espacio mínimo libre en el directorio **ARRIDIR**.

Mostrar y grabar en el log el siguiente mensaje con su respuesta:

```
Defina el espacio mínimo libre para el arribo de archivos externos en Mbytes (100):
```

Proponer 100 Mb, Si el usuario lo desea cambiar, debe ingresar una cantidad que se interpreta como Mb. Reservar este valor en la variable **DATASIZE**.

11. Verificar espacio en disco

Chequear si en **ARRIDIR** hay disponibles por lo menos **DATASIZE** Mb. Si esto da error mostrar y grabar en el log el siguiente mensaje:

```
Insuficiente espacio en disco.  
Espacio disponible: xx Mb.  
Espacio requerido $DATASIZE Mb  
Cancele la instalación e inténtelo mas tarde o vuelva a intentarlo con otro valor.
```

Volver a verificar espacio en disco.

12. Definir el directorio de grabación de los archivos aceptados

Se debe solicitar al usuario que indique cual es el directorio que desea usar para mover los archivos externos aceptados. Mostrar y grabar en el log el siguiente mensaje con su respuesta:

```
Defina el directorio de grabación de los archivos externos aceptados ($grupo/aceptados):
```

Proponer /aceptados y si el usuario lo desea cambiar, permitirle.

El usuario puede ingresar un nombre simple como "/aceptados" o un subdirectorio como /archivos/externos/ aceptados

Reservar este path en la variable **ACEPDIR**

13. Definir el directorio de grabación de los archivos rechazados

Se debe solicitar al usuario que indique cual es el directorio que desea usar para mover los archivos externos rechazados. Mostrar y grabar en el log el siguiente mensaje con su respuesta:

```
Defina el directorio de grabación de los archivos externos rechazados ($grupo/rechazados):
```

Proponer /rechazados y si el usuario lo desea cambiar, permitirle.

El usuario puede ingresar un nombre simple como "/rechazados" o un subdirectorio como /archivos/externos/rechazados

Reservar este path en la variable **RECHDIR**



14. Definir el directorio de trabajo principal del proceso [Reservar_A](#)

Se debe solicitar al usuario que indique cual es el directorio de trabajo principal que desea usar para mover los archivos procesados. Mostrar y grabar en el log el siguiente mensaje con su respuesta:

```
Defina el directorio de grabación de los archivos procesados ($grupo/procesados):
```

Proponer /procesados y si el usuario lo desea cambiar, permitiréelo.

El usuario puede ingresar un nombre simple o un subdirectorio como /archivos/externos/procesados/reservas

Reservar este path en la variable **PROCDIR**

15. Definir el directorio de grabación de los LISTADOS de salida

Mostrar y grabar en el log el siguiente mensaje con su respuesta:

```
Defina el directorio de grabación de los listados de salida ($grupo/listados):
```

Proponer /listados y si el usuario lo desea cambiar, permitiréelo.

El usuario puede ingresar un nombre simple como "listados" o un subdirectorio como archivos/salida/listados

Reservar este path en la variable **REPODIR**.

16. Definir el directorio de logs para los comandos

Se debe solicitar al usuario el directorio donde desea dejar los archivos de log que dejan los comandos principales (no el log de instalación que queda fijo en **CONFDIR**, para ello mostrar en pantalla el siguiente mensaje:

```
Defina el directorio de logs ($grupo/log):
```

Si el usuario acepta el nombre propuesto, continuar

Si el usuario lo desea cambiar, debe ingresar un nombre de directorio

Reservar este path en la variable **LOGDIR**.

17. Definir la extensión para los archivos de log

Mostrar y grabar en el log el siguiente mensaje con su respuesta:

```
"Ingrese la extensión para los archivos de log: (.log)"
```

Proponer .log y si el usuario lo desea cambiar, permitiréelo.

Reservar ese valor en la variable **LOGEXT**.

18. Definir el tamaño máximo para los archivos de log

Mostrar y grabar en el log el siguiente mensaje con su respuesta:

```
Defina el tamaño máximo para los archivos $LOGEXT en Kbytes (400):
```

Proponer 400 KB, Si el usuario lo desea cambiar, debe ingresar una cantidad que se interpreta como Kb.

Reservar este valor en la variable **LOGSIZE**.



19. Mostrar estructura de directorios resultante y valores de parámetros configurados

19.1. Limpiar la pantalla

19.2. Mostrar y grabar en el log el siguiente mensaje con su respuesta:

```
TP S07508 Segundo Cuatrimestre 2013. Tema A Copyright © Grupo xx
Librería del Sistema: CONFDIR
Ejecutables: BINDIR
Archivos maestros: MAEDIR
Directorio de arribo de archivos externos: ARRIDIR
Espacio mínimo libre para arribos: DATASIZE Mb
Archivos externos aceptados: ACEPDIR
Archivos externos rechazados: RECHDIR
Archivos procesados: PROCDIR
Reportes de salida: REPODIR
Logs de auditoria del Sistema: LOGDIR/<comando>.LOGEXT
Tamaño máximo para los archivos de log del sistema: LOGSIZE Kb
Estado de la instalacion: LISTA
```

19.3. Si el usuario indica Si, Continuar en el paso: “Confirmar Inicio de Instalación”

19.4. Si el usuario indica No, Limpiar la pantalla y continuar después del chequeo de Perl

En este caso, los valores default propuestos deben ser los recientemente ingresados por el instalador, es decir los contenidos en las variables: **BINDIR**, **ARRIDIR**, **DATASIZE**, **LOGDIR**, **LOGEXT**, **LOGSIZE**, etc.

20. Confirmar Inicio de Instalación

Mostrar y grabar en el log el siguiente mensaje con su respuesta:

```
Iniciando Instalación. Esta Ud. seguro? (Si-No)
```

20.1. Si el usuario indica Si, Continuar en el paso: “Instalación”

20.2. Si el usuario indica No, ir a **FIN**

21. Instalación

21.1. Crear las estructuras de directorio requeridas mostrando el siguiente mensaje:

```
Creando Estructuras de directorio. . . .
```

\$BINDIR

\$MAEDIR

\$ARRIDIR



\$ACEPDIR

\$RECHDIR

\$PROCDIR

\$REPODIR

\$LOGDIR

- 21.2. Mover los archivos maestros al directorio **MAEDIR** mostrando el siguiente mensaje

Instalando Archivos Maestros

- 21.3. Mover el archivo de disponibilidad al directorio **PROCDIR** mostrando el siguiente mensaje

Instalando Archivo de Disponibilidad

- 21.4. Mover los ejecutables y funciones al directorio **BINDIR** mostrando el siguiente mensaje

Instalando Programas y Funciones

- 21.5. Actualizar el archivo de configuración mostrando el siguiente mensaje

Actualizando la configuración del sistema

Se debe almacenar la información de configuración del sistema en el archivo Instalar_TP.conf en **CONFDIR**

Si el archivo de configuración no existe, crearlo, si existe actualizar los valores que correspondan.

Se debe grabar un registro para cada una de las siguientes variables:

GRUPO, CONFDIR, BINDIR, MAEDIR, ARRIDIR, **DATASIZE**, ACEPDIR, RECHDIR, PROCDIR, **REPODIR**, LOGDIR, LOGEXT, LOGSIZE

22. Borrar archivos temporarios, si los hubiese generado

23. Mostrar mensaje de fin de instalación

Instalación **CONCLUIDA**

24. FIN

Cerrar el archivo Instalar_TP.log

Terminar el proceso



Inicialización: [Iniciar_A](#)

Descripción

El propósito de este comando es preparar el entorno de ejecución del TP (ambiente).

- Es el primero en orden de ejecución
- Se dispara manualmente
- Graba en el archivo de Log a través del [Grabar_L](#)
- Invoca, si corresponde, al siguiente paso [Recibir_A](#)

El Proceso se inicia con el aseguramiento de la disponibilidad de la información para llevar adelante el proceso total: Es indispensable contar con el archivo de configuración, los comandos, archivos maestros, disponibilidad, con los permisos adecuados.

Continúa con la asignación de valor a un conjunto de variables de ambiente y luego ofrece arrancar automáticamente el comando de recepción de archivos.

El resto de los comandos JAMAS deben acceder al archivo de configuración para conocer directorios, deben usar las variables de ambiente que define este proceso.

Pasos Sugeridos

1-Inicializar el archivo de log

```
Comando Iniciar_A Inicio de Ejecución
```

2-Verificar que la instalación está completa

Si se detecta algún faltante en la instalación, explicar la situación, indicar los componentes faltantes, dar indicaciones para que el usuario ejecute el instalador, reiniciar el sistema, etc...

Grabar siempre en el log y terminar la ejecución.

3- Verificar si el ambiente ya ha sido inicializado.

[Iniciar_A](#) debe setear las variables de ambiente una sola vez por cada sesión de usuario.

Si se intenta ejecutar nuevamente este comando, explicar la situación, (por ejemplo indicar "ambiente ya inicializado, si quiere reiniciar termine su sesión e ingrese nuevamente" Grabar siempre en el log y terminar la ejecución.

4- Si Iniciar_A nunca fue ejecutado en esta sesión de usuario

Setear la variable PATH y cualquier otra variable de ambiente que considere necesarias, como ser: **GRUPO, ARRDIR, RECHDIR, BINDIR, MAEDIR, REPODIR, LOGDIR LOGEXT, etc.**

Si se detecta que algún archivo no tiene los permisos adecuados, setarlos correctamente y continuar en el siguiente paso

5-Ver si se desea arrancar [Recibir_A](#)

[Iniciar_A](#) debe ofrecer la posibilidad de arrancar el demonio [Recibir_A](#) por ejemplo mostrando el siguiente mensaje:

```
"Desea efectuar la activación de Recibir\_A?" Si - No
```



5.1 Si el usuario no desea arrancar el demonio **Recibir_A**, entonces explicar cómo hacerlo con el comando **Start_A**

5.2 Si el usuario desea arrancar el demonio **Recibir_A**, activarlo (SOLO SI NO EXISTE OTRO **Recibir_A** CORRIENDO) y explicar cómo detenerlo usando el comando **Stop_A**.

FINAL:

Si el ambiente está inicializado correctamente, mostrar el contenido de las variables, el process id del demonio (si es que está corriendo) y el estado del sistema, por ejemplo:

```
TP SO7508 Segundo Cuatrimestre 2013. Tema A Copyright © Grupo xx

Librería del Sistema: CONFDIR (mostrar path y listar archivos)

Ejecutables: BINDIR (mostrar path y listar archivos)

Archivos maestros: MAEDIR (mostrar path y listar archivos)

Directorio de arriba de archivos externos: ARRIDIR

Archivos externos aceptados: ACEPDIR

Archivos externos rechazados: RECHDIR

Reportes de salida: REPODIR

Archivos procesados: PROCDIR

Logs de auditoria del Sistema: LOGDIR/<comando>.LOGEXT

(listar todas las otras variables que deseen mostrar)

Estado del Sistema: INICIALIZADO

Demonio corriendo bajo el no.: <Process Id de Recibir_A>
```

Cerrar el archivo de log- Terminar el proceso



Detección y Recepción de Archivos Externos: [Recibir_A](#)

Input

- Archivos (cualesquiera) que arriban al directorio **\$ARRIDIR**/[<nombre del archivo>](#)
- Maestro de Salas **\$MAEDIR**/salas.mae
- Maestro de Obras **\$MAEDIR**/obras.mae

Output

- Archivos de Reservas Solicitadas **\$ACEPDIR**/[<nombre del archivo>](#)
- Archivos de Invitados **\$REPODIR**/[<nombre del archivo>](#)
- Archivos (cualesquiera) Rechazados **\$RECHDIR**/[<nombre del archivo>](#)
- Log **\$LOGDIR**/[Recibir_A](#).**\$LOGEXT**

Opciones y Parámetros

- A especificar por el desarrollador

Descripción

El propósito de este comando es detectar la llegada de archivos al directorio **\$ARRIDIR**, efectuar la validación del nombre del archivo que detecta y ponerlo a disposición del siguiente paso.

- Es el segundo en orden de ejecución
- Es un proceso del tipo “Demonio” :
- Se dispara automáticamente o a través del [Start_A](#)
- Se detiene a través del [Stop_A](#)
- Graba en el archivo de Log a través del [Grabar_L](#)
- Invoca, si corresponde, al siguiente paso [Reservar_A](#)

Cada empresario o productor envía archivos solicitando reservas para cierto evento o listados con los nombres de los invitados.

Si el nombre del archivo (filename) es del tipo esperado, acepta el archivo, de lo contrario, lo rechaza.

También en el caso de contar con archivos de reservas solicitadas “aceptadas”, arranca [Reservar_A](#) para que las confirme

¿Qué es un “Demonio”?

Un demonio, daemon o dæmon (de sus siglas en inglés Disk And Execution Monitor), es un tipo especial de proceso informático que se ejecuta en segundo plano en vez de ser controlado directamente por el usuario (es un proceso no interactivo).

Este tipo de programas se ejecutan de forma continua.

En este caso el proceso arranca, contabiliza el ciclo, verifica el directorio **\$ARRIDIR**, verifica los archivos, acepta o rechaza según corresponda, determina si debe invocar [Reservar_A](#), duerme un tiempo x y vuelve a empezar

Pasos sugeridos:

No se puede ejecutar este comando si la inicialización de ambiente no fue realizada



1. Grabar en el Log el nro de ciclo: Ej: "Ciclo Nro 1". Indicar en las hipótesis como realizan la contabilidad del ciclo.
2. Chequear si hay archivos en el directorio **\$ARRIDIR**. Si existen archivos, por cada archivo que se detecta
 - 2.1. Verificar que el archivo sea un archivo común, de texto. Los archivos de cualquier otro tipo, se rechazan.
 - 2.2. Verificar que el formato del nombre del archivo sea correcto, los archivos con nombres que no se correspondan con el formato esperado, se rechazan.

En este directorio se reciben dos tipos de archivos:

- Los archivos de reservas solicitadas
- Los archivos de invitados

Validación del nombre de los archivos de reservas solicitadas:

- Si la reserva es solicitada por la producción general o la producción ejecutiva, el nombre del archivo de reservas solicitadas tiene el siguiente formato: **ID de la OBRA-CORREO-xxx**
 - El id de la OBRA siempre es un número impar
 - Luego del id de la Obra viene un guion "-"
 - Luego del guión viene el correo electrónico de la producción general o de la producción ejecutiva. La combinación Id de la OBRA-correo debe existir en el archivo de OBRAS
 - Luego del correo viene un guion "-"
 - Y finalmente xxx que es cualquier combinación de caracteres sin guiones ni espacios.
- Si la reserva es solicitada por la sala teatral, el nombre del archivo de reservas solicitadas tiene el siguiente formato: **ID de la SALA-CORREO-xxx**
 - El id de la SALA siempre es un número par
 - Luego del id de la Sala viene un guion "-"
 - Luego del guión viene el correo electrónico del teatro. La combinación Id de la Sala-correo debe existir en el archivo de SALAS
 - Luego del correo viene un guion "-"
 - Y finalmente xxx que representa una cadena caracteres sin guiones ni espacios.

Validación del nombre de los archivos de invitados:

- El nombre de los archivos de invitados tienen el siguiente formato: **Referencia Interna del Solicitante.inv**
 - La referencia Interna del solicitante es cualquier combinación de caracteres sin guiones ni espacios
 - Luego de la referencia debe seguir obligatoriamente ".inv" de lo contrario el archivo no será identificado como archivo de invitados

- 2.3. Si el nombre del archivo es válido y es de reservas mover el archivo aceptado a **\$ACEPDIR** empleando la función **Mover_A** y grabar en el log el mensaje de éxito



2.4. Si el nombre del archivo es válido y es de invitados mover el archivo aceptado a **\$REPODIR** empleando la función **Mover_A** y grabar en el log el mensaje de éxito

2.5. Si el nombre del archivo no es válido mover el archivo rechazado a **\$RECHDIR** empleando la función **Mover_A**, grabar en el log el mensaje de rechazo aclarando cual es el motivo:

- Tipo de archivo invalido
- nombre del archivo con formato invalido
- Sala inexistente
- Obra inexistente
- Correo inexistente
- cualquier otro error que considere pertinente indicar.

3. Una vez que se hayan procesado todos los archivos que existen en **\$ARRIDIR** se debe chequear la existencia de archivos en el directorio **\$ACEPDIR** (ya sean del ciclo actual o de ciclos anteriores).

4. Si existen archivos en **\$ACEPDIR**

4.1. Invocar al Comando **Reservar_A** siempre que éste no se esté ejecutando.

Si arranca correctamente se debe mostrar por pantalla el process id de **Reservar_A**

Si da algún tipo de error se debe mostrar por pantalla el mensaje explicativo

5. Dormir x minutos y Volver al punto 1

Queda a consideración de cada grupo el valor que se asigna a la variable de tiempo x, para la prueba del TP se solicitará modificar el valor usando vi.



Confirmación de Reservas: **Reservar_A**

Input

- Archivos de Reservas Solicitadas “aceptadas” **\$ACEPDIR**/**<nombre del archivo>**
- Maestro de Salas **\$MAEDIR**/salas.mae
- Maestro de Obras **\$MAEDIR**/obras.mae
- Archivo de Disponibilidad **\$PROCDIR**/combos.dis

Output

- Archivos Procesados **\$PROCDIR**/**<nombre del archivo>**
- Archivo de Reservas Confirmadas **\$PROCDIR**/reservas.ok
- Archivo de Reservas No Confirmadas **\$PROCDIR**/reservas.nok
- Archivo de Disponibilidad **\$PROCDIR**/combos.dis
- Archivos duplicados **\$RECHDIR**/**<nombre del archivo>**
- Log **\$LOGDIR/Reservar_A.\$LOGEXT**

Opciones y Parámetros

- A especificar por el desarrollador

Descripción

El propósito de este comando es leer los archivos que se encuentran en el directorio **\$ACEPDIR**, que contienen las reservas solicitadas y grabar sus registros en reservas confirmadas (reservas.ok) o en reservas no confirmadas (reservas.nok), según la disponibilidad de butacas y validaciones efectuadas en cuanto a formato, oportunidad y disponibilidad.

- Es el tercero en orden de ejecución
- Se dispara automáticamente
- Graba en el archivo de Log a través del **Grabar_L**

No debe procesar dos veces un mismo archivo

A efectos de simplificar el desarrollo del TP, no será tomada en cuenta la ubicación solicitada (la reserva puede venir con fila, butaca y sección). Solo tendremos en cuenta la cantidad de butacas solicitadas.

Pasos sugeridos

No se puede ejecutar este comando si la inicialización de ambiente no fue realizada

1. Inicializar el Log

Inicializar el log grabando: Inicio de **Reservar_A** y Cantidad de Archivos de input. Los archivos de input se encuentran en **\$ACEPDIR**

2. Procesar Un Archivo

Grabar en el log “Archivo a procesar: <nombre del archivo a procesar>

3. Verificar que no sea un archivo duplicado



Cada vez que se procesa un archivo, se lo mueve tal cual fue recibido y con el mismo nombre a **\$PROCDIR**

Es por ello que antes de intentar procesar un archivo se debe analizar si ya fue procesado

Si ya fue procesado, rechazar el archivo completo y grabar en el log un mensaje aclaratorio, como ser: "Se rechaza el archivo por estar DUPLICADO".

El archivo duplicado se lo mueve a **\$RECHDIR** empleando la función **Mover_A**.

4. Realizar otras validaciones a nivel de archivo si lo considera necesario

Es responsabilidad del desarrollador prever otras circunstancias de error, como ser que el formato del registro no se corresponde con lo esperado, que el archivo esta vacio, etc.

Si ello ocurre, rechazar el archivo completo y grabar en el log un mensaje aclaratorio, como ser: "Archivo Vacio".

El archivo rechazado se lo mueve a **\$RECHDIR** empleando la función **Mover_A**.

5. Por cada Registro del Archivo

5.1 Verificar la fecha de función solicitada - OPORTUNIDAD

a) No se toman reservas que tengan la fecha invalida

Para realizar esta verificación validar que la fecha de la función sea una fecha válida (formato dia/mes/año)

- Si es una fecha invalida, rechazar el registro con Motivo = Fecha invalida. Ir a RECHAZAR ESTA RESERVA

b) No se toman reservas de funciones vencidas, ni reservas para el mismo día de la función

Para realizar esta verificación comparar la fecha de la función con la fecha del día.

- Si está vencida o es para el mismo día: rechazar el registro con Motivo = Reserva vencida o Motivo = Reserva tardía. Ir a RECHAZAR ESTA RESERVA

c) No se toman reservas con más de 45 días de anticipación.

Para realizar esta verificación comparar la fecha de la función con la fecha del día + 45 días

- Si es una fecha con más de 45 dias de anticipación entonces rechazar el registro con Motivo = Reserva anticipada. Aun no se pueden confirmar reservas para esta función. Ir a RECHAZAR ESTA RESERVA

5.2 Verificar la hora de función solicitada

a) No se toman reservas que tengan la hora invalida

Para realizar esta verificación validar que la hora de la función sea una hora válida (formato hh:mm)

- Si es una hora invalida, rechazar el registro con Motivo = hora invalida. Ir a RECHAZAR ESTA RESERVA

5.3 Verificar que existe el evento

Para verificar la existencia del evento cuya reserva es solicitada, buscar si existe un combo en el archivo de disponibilidad **\$PROCDIR/combo.dis** que sea de la obra-fecha-hora ó de la sala-fecha-hora según los datos que se dispongan

Por obra: **ID DE LA OBRA** (del nombre del archivo) + **FECHA DE FUNCIÓN** (del registro) + **HORA DE FUNCIÓN** (del registro)

Por sala: **ID DE LA SALA** (del nombre del archivo) + **FECHA DE FUNCIÓN** (del registro) + **HORA DE FUNCIÓN** (del registro)



- Si existe, continuar
- Si no existe, entonces Motivo de Rechazo = No existe el evento solicitado. Ir a RECHAZAR ESTA RESERVA

5.4 Determinar si hay DISPONIBILIDAD

El DISPONIBLE se obtiene:

- Si es la primera reserva de ese combo en este ciclo de proceso del comando [Reservar_A](#), entonces la cantidad de butacas disponibles se toma del archivo de disponibilidad (**BUTACAS DISPONIBLES**)
- Si en este ciclo de proceso ya se confirmaron reservas para ese combo, entonces la cantidad de butacas disponibles se toma del valor calculado (y que está en memoria) hasta este momento.

Si no hay disponibilidad, rechazar ese registro con Motivo de Rechazo = falta de disponibilidad. Ir a RECHAZAR ESTA RESERVA

Si hay disponibilidad, pero no alcanza para cubrir toda la cantidad solicitada, rechazar ese registro con Motivo de Rechazo = falta de disponibilidad. Ir a RECHAZAR ESTA RESERVA

5.5 Efectuar otras validaciones a nivel de campo, en el momento que lo crea conveniente

Efectuar las validaciones a nivel de campo que considere necesarias y en el momento que lo crea conveniente. Se recuerda que lo descripto son sugerencias para el desarrollador, puede llevar a cabo el análisis del registro en el orden que lo desee.

Cualquier error que se detecte y provoque el rechazo del registro, debe quedar documentado en el motivo y grabar el registro en reservas.nok

5.6 Armar el registro de reservas.ok

Estructura		
campo1	ID de la OBRA	Según lo determinado previamente (en 5.3)
Campo2	NOMBRE DE LA OBRA	Del Archivo de Obras
Campo3	FECHA DE FUNCIÓN	Del Archivo de Reservas solicitadas
Campo4	HORA DE FUNCIÓN	Del Archivo de Reservas solicitadas
Campo5	ID de la SALA	Según lo determinado previamente (en 5.3)
Campo6	NOMBRE DE LA SALA	Del Archivo de Salas
Campo7	CANTIDAD DE BUTACAS CONFIRMADAS	= a CANTIDAD DE BUTACAS SOLICITADAS
Campo8	ID del COMBO	Según lo determinado previamente (en 5.3)
Campo9	REFERENCIA INTERNA DEL SOLICITANTE	Del Archivo de Reservas solicitadas
Campo10	CANTIDAD DE BUTACAS SOLICITADAS	Del Archivo de Reservas solicitadas
Campo11	CORREO DEL SOLICITANTE	Del Nombre del Archivo
campo12	USUARIO GRABACION	Usuario de Grabación
campo13	FECHA GRABACION	Fecha y Hora de Grabación, formato a elección

5.7 Grabar el registro en reservas.ok

Grabar el registro en el archivo \$**PROCDIR/reservas.ok**

Si el Archivo ya existe, agregarle los nuevos registros. Si no existe, crearlo.

Contabilizar la cantidad de registros grabados.

Actualizar el DISPONIBLE de ese combo

Repetir el paso 5 hasta que se termine el archivo.

6. RECHAZAR ESTA RESERVA

6.1 Armar el registro de reservas.nok

Estructura		



campo1	REFERENCIA INTERNA DEL SOLICITANTE	Del Archivo de Reservas solicitadas
campo2	FECHA DE FUNCIÓN	Del Archivo de Reservas solicitadas
campo3	HORA DE FUNCIÓN	Del Archivo de Reservas solicitadas
campo4	NRO. DE FILA	Del Archivo de Reservas solicitadas
campo5	NRO. DE BUTACA	Del Archivo de Reservas solicitadas
campo6	CANTIDAD DE BUTACAS SOLICITADAS	Del Archivo de Reservas solicitadas
campo7	SECCION	Del Archivo de Reservas solicitadas
Campo8	MOTIVO	lo determinado previamente
campo9	ID de la SALA	Del Nombre del Archivo o lo determinado previamente (en 5.3) o “falta SALA”, según corresponda
campo10	ID de la OBRA	Del Nombre del Archivo o lo determinado previamente (en 5.3) o “falta OBRA”, según corresponda
campo11	CORREO DEL SOLICITANTE	Del Nombre del Archivo
campo12	USUARIO GRABACION	Usuario de Grabación
campo13	FECHA GRABACION	Fecha y Hora de Grabación, formato a elección

6.2 Grabar el registro en reservas.nok

Grabar el registro en el archivo **\$PROCDIR/reservas.nok**

Si el Archivo ya existe, agregarle los nuevos registros. Si no existe, crearlo.

Contabilizar la cantidad de registros grabados.

Repetir el paso 5 hasta que se termine el archivo.

7. Fin de Archivo

Para evitar el reprocesamiento de un mismo archivo, mover el archivo procesado a: **\$PROCDIR** empleando la función **Mover_A**.

Grabar en el Log la cantidad de Registros grabados en ok y la cantidad de Registros grabados en nok

NOTA: en tiempos de testing verifique que la cantidad de Registros grabados en ok + la cantidad de Registros grabados en nok coincida con el total de registros del archivo

Repetir hasta que se terminen todos los archivos.

8. Actualizar la disponibilidad

De acuerdo a su elección algorítmica, en el momento que lo considere adecuado, PERO SIEMPRE ANTES de terminar la ejecución de este comando, debe actualizar en el archivo de disponibilidad el nuevo DISPONIBLE en todos los registros afectados.

Cada vez que lo hace debe grabar en el log “Actualización del archivo de disponibilidad”

9. Cerrar el Log

Cerrar el log grabando “Fin de Reservar_A”



Impresión de Listados, Tickets, Vales: [Imprimir_A](#)

Input

- Maestro de Salas **\$MAEDIR**/salas.mae
- Maestro de Obras **\$MAEDIR**/obras.mae
- Archivo de Reservas Confirmadas **\$PROCDIR**/reservas.ok
- Archivo de Reservas No Confirmadas **\$PROCDIR**/reservas.nok
- Archivo de Disponibilidad **\$PROCDIR**/compos.dis
- Archivos de Invitados **\$REPODIR**/**<Referencia Interna Del Solicitante>**.inv

Output

- Listado de Invitados Confirmados: **\$REPODIR**/**<id del combo>**.inv
- Listado de Disponibilidad: **\$REPODIR**/**<nombre ingresado>**.dis
- Ranking **\$REPODIR**/ranking.nnn
- Tickets: **\$REPODIR**/**<Id Del Combo>**.tck

Opciones y Parámetros

- -a (ayuda)
- -w (grabar)
- -i (invitados)
- -d (disponibilidad)
- -r (ranking)
- -t (ticket)
- Otros parámetros a especificar por el desarrollador

Descripción

Este proceso nos permite realizar consultas de disponibilidad, emitir el ranking de los 10 principales, generar la lista de invitados para las reservas confirmadas, generar los archivos para imprimir los tickets de entrada.

- Es el cuarto en orden de ejecución
- Se dispara manualmente
- No graba en el archivo de log

Llamaremos evento a la combinación obra-fecha-hora-sala (combo) a los fines de simplificar la explicación

Opción a: opción de ayuda del comando

Se debe presentar un menú amigable y una opción de ayuda del comando

Opción w: opción de grabación

Esta opción es combinable con las opciones -i -d -r -t

Cuando se indica w, se debe grabar el archivo



Si no se indica w, solo mostrar por pantalla

Opción i: generación del listado de invitados a un evento

Para generar este listado se deben cruzar los datos del archivo de reservas.ok y los archivos de invitados a través de la referencia interna del solicitante, teniendo en cuenta que:

- No todos los registros de reservas.ok tienen referencia interna del solicitante
- No todos los registros de reservas.ok con referencia interna del solicitante tienen archivos de invitados
- No todos los archivos de invitados tienen reservas confirmadas

Es por ello que si el usuario elige esta opción se le debe presentar una lista de eventos “candidatos” y solicitar que elija uno de ellos para generar el listado de invitados

Una vez seleccionado el evento mostrar por pantalla y grabar (si se indico opción w) lo siguiente.

Primera línea: **Evento: <ID DEL COMBO> Obra:<ID de la OBRA-NOMBRE DE LA OBRA> Fecha y Hora:<FECHA DE FUNCIÓN-HORA DE FUNCIÓN> Hs. Sala: <ID de la SALA-NOMBRE DE LA SALA>**

Segunda línea: **REFERENCIA INTERNA DEL SOLICITANTE** (corte de control por referencia interna del solicitante)

- Si no existe un listado, mostrar la leyenda “sin listado de invitados” y seguir con la siguiente referencia interna del solicitante
- Si existe un listado de invitados para esa referencia interna, mostrar en líneas sucesivas **NOMBRE DEL INVITADO, CANTIDAD DE ACOMPAÑANTES, TOTAL ACUMULADO** (valor anterior + 1 + cantidad de acompañantes)

Cuando cambia la referencia interna del solicitante mostrar total de Reservas confirmadas: cantidad (suma de todas las butacas confirmadas para esa referencia interna según reservas.ok) y el total acumulado. Antes de seguir con la siguiente referencia, volver a cero los totales.

Seguir con la siguiente referencia interna hasta terminar todas las que corresponden al evento.

El nombre del archivo debe ser <id del combo>.inv. Si ya existe un archivo del mismo nombre, reemplazarlo

Opción d: listado de disponibilidades

Para generar este listado se emplea el archivo combos.dis

Solicitar al usuario que ingrese: un **ID DE LA OBRA** ó un **ID DE LA SALA** ó un rango de **ID DE LA OBRA** ó un rango de **ID DE LA SALA**

Si no existe la clave o el rango de claves mostrar mensaje y volver a solicitar.

Si existe la clave y/o el rango de claves mostrar mostrar por pantalla y grabar (si se indico opción w) lo siguiente.

ID DEL COMBO - ID DE LA OBRA - FECHA DE FUNCIÓN - HORA DE FUNCIÓN - ID DE LA SALA – BUTACAS HABILITADAS - BUTACAS DISPONIBLES

Si se debe grabar el listado, solicitar al usuario un nombre para el listado y grabar \$REPODIR/<nombre ingresado>.dis. Jamás xx puede ser “combos”

Opción r: generar ranking de eventos

El propósito de este listado es mostrar cuales son los 10 eventos con mayor cantidad de reservas.

Para generar este listado se debe emplear el archivo de reservas.ok

Mostrar **CORREO ELECTRÓNICO, CANTIDAD**



Si el usuario solicita grabar, entonces el nombre del archivo debe ser ranking.nnn dónde nnn es un descriptor siempre distinto que asegura no sobrescribir ningún informe previo.

Opción t: generar el listado de tickets a imprimir

Para generar este listado se debe emplear el archivo de reservas.ok

Solicitar al usuario que ingrese: un **ID DEL COMBO**

Si no existe la clave volver a solicitar.

Por cada registro del combo solicitado:

Si la cantidad de butacas confirmadas es 1: grabar VALE POR 1 ENTRADA en TIPO DE COMPROBANTE

Si la cantidad de butacas confirmadas es 2: grabar VALE POR 2 ENTRADAS en TIPO DE COMPROBANTE

Si la cantidad de butacas confirmadas es MAYOR a 2: grabar:

- Tantos registros con VALE POR 2 ENTRADAS en TIPO DE COMPROBANTE como sea posible (hasta agotar el numero de butacas confirmadas)
- En caso de que quede un remanente de 1 entrada para completar el numero de butacas confirmadas grabar otro registro con VALE POR 1 ENTRADA en TIPO DE COMPROBANTE

El resto de los campos según la estructura solicitada

El nombre del archivo debe ser <id del combo>.tck. Si ya existe un archivo del mismo nombre, reemplazarlo

Estructura

Campo1	TIPO de COMPROBANTE	VALE POR 1 ENTRADA o VALE POR 2 ENTRADAS según corresponda
Campo2	NOMBRE DE LA OBRA	Del archivo de reservas confirmadas
Campo3	FECHA DE FUNCIÓN	Del archivo de reservas confirmadas
Campo4	HORA DE FUNCIÓN	Del archivo de reservas confirmadas
Campo5	NOMBRE DE LA SALA	Del archivo de reservas confirmadas
Campo6	REFERENCIA INTERNA DEL SOLICITANTE	Del archivo de reservas confirmadas
Campo7	CORREO DEL SOLICITANTE	Del archivo de reservas confirmadas

Requisitos

Imprimir_A No debe ejecutar si la inicialización de ambiente no fue realizada

Imprimir_A No debe ejecutar si ya existe otro comando Imprimir_A en ejecución

Se deben emplear estructuras Hash en la resolución (requisito indispensable), indicar en las hipótesis generales del TP que opciones resuelven con esta herramienta.



Especificación de Funciones

A continuación se brinda una descripción de cada función solicitada.

Los pasos de ejecución sugeridos son solo a los efectos de ordenar la explicación, por lo cual deben considerarse meramente indicativos.

Si el equipo de desarrollo lo considera pertinente, puede modificarlos tanto sea en el orden de ejecución como en la forma de resolverlo, siempre y cuando esto no afecte el resultado final esperado y lo aclare debidamente en las hipótesis.

Si el equipo de desarrollo lo considera pertinente puede aumentar la funcionalidad requerida de las funciones o crear nuevas funciones de librería que crea conveniente

Estos cambios deben estar documentados en la carpeta que entrega el día de vencimiento del tp

[Start_A](#)

Opciones y Parámetros

- parámetros u opciones a especificar por el desarrollador

Descripción

Esta función tiene por objeto disparar procesos. Es complementaria a [Stop_A](#).

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. Indicar en las Hipótesis Globales donde se realizan estos controles

- a) en forma centralizada en esta función
- b) el control se delega al comando llamador

Si esta función es invocada por un comando que graba en un archivo de log, actualizarlo indicando el resultado de la operación

Explicar claramente su invocación y uso en el README.

[Stop_A](#)

Opciones y Parámetros

- parámetros u opciones a especificar por el desarrollador

Descripción

Esta función tiene por objeto detener procesos. Es complementaria a [Start_A](#)

Puede ser invocada desde la línea de comando o bien desde otro comando.

Si esta función es invocada por un comando que graba en un archivo de log, actualizarlo indicando el resultado de la operación

Explicar claramente su invocación y uso en el README.



Mover_A

Opciones y Parámetros

- Parámetro 1 (obligatorio): origen
- Parámetro 2 (obligatorio): destino
- Parámetro 3 (opcional): comando que la invoca
- Otros parámetros u opciones a especificar por el desarrollador

Descripción

Esta función tiene por objeto mover un archivo de un directorio a otro contemplando la posibilidad de archivos duplicados.

En caso de archivos duplicados se debe gestionar un numero de secuencia nnn.

Este número de secuencia puede ser centralizado (una única secuencia para todo el sistema) o descentralizado (diferentes secuencias).

EVITE crear un archivo auxiliar solo para registrar el número de secuencia

Si requiere registrar el numero de secuencia en un archivo utilice el archivo de configuración agregando al final un registro con la variable y el valor. Documente esto en las hipótesis generales del sistema.

Si esta función es invocada por un comando que graba en un archivo de log, actualizarlo indicando el resultado de la operación

Mover_A debe devolver un código de retorno cero (0) si fue exitoso el movimiento o distinto de cero si tuvo errores.

Pasos Sugeridos

1. Antes de efectuar el movimiento, debe:
 - 1.1. Verificar si el origen y el destino son iguales. Si este fuera el caso, no mover
 - 1.2. Si el origen no existe o el destino no existe, es un error. Si este fuera el caso, no mover
 - 1.3. Verificar si es un archivo duplicado, es decir, si en el destino ya existe un archivo con ese mismo nombre. En este caso debe:
 - 1.3.1. Ver si existe dentro del directorio destino un subdirectorio /dup
 - 1.3.1.1. si no existe, crearlo
 - 1.3.2. Mover el archivo a ese subdirectorio con el siguiente nombre: <nombre del archivo original>.nnn dónde nnn es un número de secuencia que evita nombres duplicados, es decir, evita "sobreescribir" archivos.
 - 1.3.3. Eliminar el archivo del directorio de origen
- 1.4. si no es un archivo duplicado, entonces hacer el move al destino indicado.

Ejemplo mover arch.txt de /data1 a /data2

Antes de MOVER	Luego de MOVER
/data1/arch.txt	/data1
/data2/arch.txt	/data2/arch.txt
	/data2/dup/arch.txt.1

Ejemplo mover arch.txt de /data1 a /data2

Antes de MOVER	Luego de MOVER
/data1/arch.txt	/data1
/data2	/data2/arch.txt



Grabar_L

Opciones y Parámetros

- Parámetro 1 (obligatorio): comando
- Parámetro 2 (opcional): tipo de mensaje
- Parámetro 3 (obligatorio): mensaje
- Otros parámetros u opciones a especificar por el desarrollador

¿Qué es un Log?

Un log es un registro oficial de eventos durante un periodo de tiempo en particular.

Es usado para registrar datos o información sobre quién, qué, cuándo, dónde y por qué (who, what, when, where y why, también llamado Standard W5) un evento ocurre para un dispositivo en particular o aplicación.

También se le considera como aquel mensaje que genera el programador de un sistema operativo, alguna aplicación o algún proceso, en virtud del cual se muestra un evento del sistema.

Es empleado por los profesionales de IT, auditoria y seguridad informática

¿Cómo debe ser un Log?

La escritura en el archivo de Log debe ser homogénea para todos los comandos por lo tanto debe estar centralizada a través de una función de librería.

Cada registro de log debe responder al estándar W5.

El Archivo de Log no debe crecer indiscriminadamente, debe preverse la forma de generar alertas o evitar su crecimiento sin control.

Descripción

Esta función tiene por objeto grabar en el log correspondiente los mensajes pasados como parámetro contemplando la creación del archivo de log cuando este no existiera.

Con el nombre del comando pasado como parámetro y la extensión indicada en la variable de ambiente **\$LOGEXT** se construye el nombre del **Archivo de Log**, excepto para el log de la instalación que la extensión es siempre .log

El directorio donde se graba el log se indica en la variable de ambiente **\$LOGDIR**, excepto para el log de la instalación que se graba en el directorio **CONFDIR**

Si el archivo de log no existe, se debe crear. Si existe se le deben agregar los nuevos registros.

El tamaño máximo que puede alcanzar un archivo de log se indica en la variable de ambiente **\$LOGSIZE**.

Este tamaño máximo es un valor de referencia, en la práctica es posible encontrar archivos de log que superan el valor **\$ LOGSIZE** dado que depende del momento en que se efectúa este control.

La idea es que cuando un archivo de log supera ese tamaño, se trunque el archivo dejando por ejemplo las últimas 50 líneas (valor arbitrariamente establecido). Aclare en las hipótesis generales del sistema que política de truncamiento ha adoptado, si desea proponer otro mecanismo puede hacerlo aclarándolo debidamente en las hipótesis generales del sistema.

Cada vez que se reduce el tamaño del log, grabar en el log el mensaje: Log Excedido

Esto debe ser aplicado a todos los logs de comandos excepto al log de la instalación, este log nunca debe ser recortado.

Cada línea de log debe responder al estándar W5 (when, who, where, what and why):

¿Cuándo?

Fecha y Hora, en el formato que deseen y calculada justo antes de la grabación.



¿Quién?

Usuario, es el login del usuario

¿Dónde?

Comando, nombre del comando o función que genera el mensaje.

Se apreciará la utilidad de este parámetro cuando la función [Mover_A](#) deba generar mensajes de log

¿Qué?

Tipo de Mensaje: Existen cuatro tipos de mensajes:

- I = INFORMATIVO: mensajes explicativos sobre el curso de ejecución del comando. Ej: Inicio de Ejecución
- W = WARNING (de alerta): mensajes de advertencia pero que no afectan la continuidad de ejecución del comando. Ej: Archivo duplicado
- E = ERROR: mensajes de error Ej: Archivo Inexistente.
- SE = SEVERAL ERROR: mensajes severos de error que provocarán la cancelación del comando: Ej: Archivo Maestro no encontrado

¿Por qué?

Mensaje, string de hasta 120 caracteres.

Todo el mensaje debe registrarse en una sola línea, el separador de registros debe ser new line.

ATENCION: Como el comando de instalación también debe grabar un log respetando la forma de escritura aquí indicada, aclare en las hipótesis generales del sistema que solución adoptaron para evitar duplicar este código

FIN
