



HOJA CARATULA

Primera hoja de la carpeta, complete con grupo e integrantes

Tema	Grupo	Ayudante	Correcciones			
			Fecha	Hora Inicio	Hora Fin	Resultado
			Entrega parte I 27/09			
			Entrega parte II 18/10			

INTEGRANTES

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

Incorpore a continuación el índice de la carpeta y luego todos los ítems solicitados en la documentación con hojas numeradas en el margen derecho y número de grupo en el margen izquierdo

Incorpore como apéndice al final de la carpeta el enunciado Primera Parte del TP

Sujete las hojas de manera tal que se pueda hojear la carpeta sin que las hojas se caigan ni mezclen (no dentro de folios que impiden su lectura, carpeta con ganchos o broches tipo nepaco es lo adecuado) Las hojas sueltas no se considerarán como parte de la entrega.

El día de la entrega debe traer: carpeta, planilla de evaluación impresa para entregar al docente, maquina con SO instalado adecuado para ejecutar el TP, pen drive con paquete de instalación y datos entregados por la cátedra.

27 de Septiembre o 29 de Septiembre según el turno asignado	Vencimiento Primera Parte del TP – Asistencia obligatoria de los responsables de la resolución de la primera parte del TP. Asistencia opcional del resto del grupo.
18 de Octubre o 20 de Octubre según el turno asignado	Vencimiento Segunda y ultima Parte del TP – Asistencia obligatoria de los responsables de la resolución. Asistencia opcional del resto del grupo.

Se evaluará el trabajo grupal y a cada integrante en forma individual. El objetivo de esto es comprender la dinámica de trabajo del equipo y los roles que ha desempeñado cada integrante del grupo.

Para que el alumno apruebe el trabajo práctico debe estar aprobado en los dos aspectos: grupal e individual.

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.

Entrega final, una vez evaluado satisfactoriamente el TP, se debe remitir el paquete de instalación vía correo electrónico a so7508@gmail.com. En el asunto del correo indicar Nro. de Grupo y Ayudante asignado. En el cuerpo del correo se debe indicar la versión de Sistema Operativo usada para la ejecución.



Apéndice I

Enunciado primera parte – Tema M

Índice del enunciado del TP

Enunciado primera parte – Tema M	2
Introducción Narrativa – ejecución planificación presupuestaria	3
Documentación Solicitada	3
1. Contenido de la Carpeta – Primera Entrega	3
2. Contenido de la Carpeta – Segunda Entrega	4
Especificación de comandos y Funciones	4
Recomendaciones para el equipo de desarrollo	4
Indicaciones para el equipo de integración y testing	5
Comandos Solicitados	5
1. Contenido de la Primera Entrega	5
2. Contenido de la Segunda Entrega	5
Instalación: Instalep	6
Initep	11
Demonep	12
Función Logep	14
Función Movep	15
Procep	16
Listep	16
Estructuras y Archivos	16
Maestro de Centros Presupuestarios: DIRMAE /centros.csv	16
Maestro de Provincias: DIRMAE /provincias.csv	17
Tabla de Trimestres: DIRMAE /trimestres.csv	17
Archivos de Log: DIRLOG / <nombre del comando> .log	18
Archivo de Configuración /dirconf/ Instalep.conf	18



Introducción Narrativa – ejecución planificación presupuestaria

Al sistema lo llamaremos **EPLAM** y el mismo estará compuesto por:

- La documentación del sistema **EPLAM**
- Un comando Shell **Instalep** para la instalación de sistema
- Un comando Shell **Initep** para la configuración del entorno de ejecución
- Un comando shell **Demonep** para la recepción de los archivos de novedades
- Un comando Shell **Procep** para determinación de imputación de gastos a cada partida presupuestaria
- Un comando PERL **Listep** para la generación de listados y reportes de gestión
- Una Función (en Shell o en Perl) denominada **Logep** que se emplea para grabar los archivos de log
- Una Función (en Shell o en Perl) denominada **Movep** que se emplea para mover archivos

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

Documentación Solicitada

1. Contenido de la Carpeta – Primera Entrega

- 1.1. Carátula (primer hoja de este documento)
- 1.2. Nuevo Índice del Contenido de la Carpeta.

OBLIGATORIO contar con un índice con número de página en cada ítem, el número de página puede ser incorporado manualmente.

- 1.3. Aclaraciones Globales e Hipótesis por comando

Documente cualquier aclaración que considere necesaria para asegurar el éxito de la corrección

Documente las hipótesis que ha considerado para la resolución en cada comando (opcional)

Si no hay aclaraciones ni hipótesis, igualmente incorpore esta hoja con solo su título

- 1.4. 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.

- 1.5. Archivo README

OBLIGATORIO que en la carpeta se incluya la impresión del README, el docente seguirá estos pasos para instalar y ejecutar el TP

- 1.6. Listado de Nuevas Funciones y/o Comandos Auxiliares

Si agrega funciones o comandos auxiliares, Indique: Nombre de la función, quienes la usan, para que la usan.

- 1.7. Listado de Nuevos Archivos

Si agrega archivos NO TEMPORALES, Indique: Nombre del archivo, donde lo almacenan, quienes lo usan, para que lo usan.

- 1.8. Listado de DATOS

Imprima un listado con todos los nombres de los archivos de datos entregados por la cátedra y su cantidad de registros

- 1.9. Apéndice, enunciado del TP



2. Contenido de la Carpeta – Segunda Entrega

- 2.1. Carátula (heredada de la primera entrega)
- 2.2. Índice de la Segunda Parte
- 2.3. Aclaraciones Globales e Hipótesis por comando
- 2.4. Problemas relevantes
- 2.5. Archivo README
- 2.6. Listado de Nuevas Funciones y/o Comandos Auxiliares
- 2.7. Listado de Nuevos Archivos
- 2.8. Casos de Prueba

Imprima un listado con todos los casos de prueba: nombre del archivo y que lo caracteriza (que es lo que se prueba con ese archivo)

- 2.9. Apéndice A, todo el Contenido de la Carpeta – Primera Entrega
- 2.10. Apéndice 2, el enunciado del TP segunda entrega

Especificación de comandos y Funciones

Recomendaciones para el equipo de desarrollo

1. Se deberá tener en cuenta para la resolución TODAS las condiciones que se enuncian.

Se deben respetar los formatos de archivos especificados y la estructura de directorios planteada

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. También pueden:

- Crear nuevos scripts
- Aumentar la funcionalidad de los comandos solicitados

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

2. Archivos Auxiliares

Se debe evitar el uso de archivos auxiliares permanentes, los archivos auxiliares temporales, se deben eliminar ANTES de finalizar la ejecución del comando.

3. 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 [Movep](#) para el movimiento de archivos

4. Manejo de errores, logueo

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.

Todo evento que genera algún tipo de error debe ser grabado en el log y mostrado por pantalla

EVITE retrasar el proceso de evaluación/corrección del TP debido a la falta de rastreo de eventos. Es por ello que se recomienda dejar en los scripts las pistas de rastreo que crea convenientes condicionadas a un flag que se enciende si es necesario. Esto evitara que en la ejecución estándar se inunde de mensajes sin interés para el usuario.

La escritura de los archivos de log de los comandos debe ser homogénea, es por ello que se centraliza en la función [Logep](#)

5. Archivo de Configuración

La instalación deja un archivo de configuración con varios registros con el contenido de variables usadas en el sistema. Los desarrolladores pueden agregar más registros si lo consideran necesario.



Indicaciones para el equipo de integración y testing

Siempre realizar la integración antes de cualquier entrega, para poder subsanar los errores de comunicación que surjan entre los comandos encadenados.

La cátedra provee los archivos de prueba

- El primer registro de cada archivo contiene los nombres de cada campo, pueden remover este registro y dejar solo los datos o bien contemplar el no uso del primer registro.

Indique en el punto hipótesis y aclaraciones globales de la carpeta que política emplea en todo el sistema, la misma política debe aplicarse a todos los comandos.

- Si los archivos remitidos por la cátedra contienen caracteres incompatibles con su configuración (por ejemplo, el carácter de fin de registro o fin de archivo), puede realizar la conversión que necesite siempre que sea homogénea (igual para todos los archivos) y sin modificar los datos

Si lo hace, indíquelo en el punto hipótesis y aclaraciones globales de la carpeta

Se proveen archivos de prueba (novedades) con un alto porcentaje de información libre de error, es responsabilidad del equipo de testing generar otros archivos de prueba con casos lo suficientemente heterogéneos como para contemplar todas las variantes de ejecución, en particular las de rechazo o error

Comandos Solicitados

1. Contenido de la Primera Entrega

- 1.1. La documentación del sistema [EPLAM, primera entrega](#)
- 1.2. Un comando Shell [Instalep](#) para la instalación de sistema
- 1.3. Un comando Shell [Initep](#) para la configuración del entorno de ejecución
- 1.4. Un comando shell [Demonep](#) que pueda ser invocado por [Initep](#), duerma un tiempo X (largo) y vuelva a empezar. Eventualmente pueden hacer mas desarrollo dentro de este script, pero no es necesario que este completo para la primera entrega.
- 1.5. Una Función (en Shell o en Perl) denominada [Logep](#) que se emplea para grabar los archivos de log

Se requiere que funcione el instalador, el configurador de entorno grabando en el log con la función [Logep](#) y llamando al comando [Demonep](#)

2. Contenido de la Segunda Entrega

- 2.1. La documentación del sistema [EPLAM, segunda entrega](#)
- 2.2. Todo el TP completo y funcionando



Instalación: *Instalep*

Descripción

El propósito de este comando es efectuar la instalación del sistema **EPLAM**.

Punto 1. 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:

- La documentación, los scripts desarrollados y los archivos con datos (maestros y novedades) entregados por la cátedra

La extracción del paquete debe generar en **\$GRUPO** un subdirectorio denominado **/dirconf**

Resguarde siempre el paquete original incluidos los datos de prueba en algún lugar de \$grupo

Brinde indicaciones precisas en el README para realizar la descarga del paquete

- Una explicación de cómo descargar el paquete
- Una explicación de cómo descomprimir, crear directorio del grupo, etc
- Una explicación de lo que sea crea a partir de la descompresión

Punto 2. INSTALADOR

Brinde indicaciones precisas en el README para realizar la instalación del paquete

- Una explicación sobre que se requiere para poder instalar y/o ejecutar el sistema
- Instrucciones de instalación del sistema **EPLAM**
- Que nos deje la instalación y dónde
- Cuáles son los primeros pasos para poder ejecutar el sistema
- Como arrancar o detener comandos
- Cualquier otra indicación que considere necesaria

Punto 3. DIRECTORIO DE TRABAJO

Para realizar la instalación el directorio de trabajo debe ser Grupoxx, donde xx es su número de grupo

Todo el camino (path) que va desde la raíz hasta Grupoxx lo denominaremos genéricamente en esta explicación **\$GRUPO**

Toda la instalación debe depender del directorio de trabajo **\$GRUPO** (creado en la descompresión del paquete)

Ningún comando puede leer o grabar información del sistema fuera de esta rama

Punto 4. INTERACCIÓN CON EL USUARIO

Se Interactúa con el usuario para solicitarle que defina subdirectorios, espacio requerido, etc.

Siempre que se interactúa con el usuario proponer el valor por default mostrándolo entre paréntesis. Ejemplo:

Defina el directorio de ejecutables (\$grupo/bin):

Si durante el mismo hilo (o instancia) de ejecución de **Instalep** se debe reiniciar desde un paso previo (volver a un punto anterior), los valores default propuestos inicialmente por el script deben ser reemplazados por los valores recientemente ingresados por el usuario, es decir, que el script debe tener “memoria” de los valores definidos por el usuario durante el mismo hilo de ejecución.

Punto 5. PREVENIR USO DE DIRCONF



El directorio `$GRUPO/dirconf` (ya fue creado en la descompresión del paquete) se considera nombre reservado, es decir que el usuario no puede escoger este nombre para un directorio de instalación.

Si Uds necesitan ampliar la lista de nombres reservados [aclárenlo en las hipótesis](#) y [programen la validación en este script](#).

Punto 6. LOG

El log de la instalación `Instalep.log` debe grabarse en el directorio de configuración `$GRUPO/dirconf`

Grabar en el log todos los mensajes que se muestran por pantalla, los valores ingresados por el usuario. Si se ejecuta muchas veces el script, se agregan las nuevas líneas.

Grabar en el log el siguiente mensaje

```
Inicio del proceso. Usuario Fecha y Hora
```

Punto 7. DETECTAR SISTEMA YA INSTALADO

Detectar si el paquete **EPLAM** ya está instalado

Este script de instalación se debe preparar de manera tal que detecte si el paquete ya se encuentra instalado

- Si `Instalep.conf` no existe, asumir que el paquete no fue instalado. Ir a definir los nombres de los directorios
- Si `Instalep.conf` existe, asumir que el paquete ya fue instalado. Mostrar y grabar en el log el siguiente mensaje

```
Directorio de Configuración: $GRUPO/dirconf (mostrar path y listar archivos)
Directorio de Ejecutables: ($GRUPO/DIRBIN mostrar path y listar archivos)
Directorio de Maestros y Tablas: ($GRUPO/DIRMAE mostrar path y listar archivos)
Directorio de Recepción de Novedades: ($GRUPO/DIRREC mostrar path)
Directorio de Archivos Aceptados: ($GRUPO/DIROK mostrar path)
Directorio de Archivos Procesados: ($GRUPO/DIRPROC mostrar path)
Directorio de Archivos de Reportes: ($GRUPO/DIRINFO mostrar path)
Directorio de Archivos de Log: ($GRUPO/DIRLOG mostrar path)
Directorio de Archivos Rechazados: ($GRUPO/DIRNOK mostrar path)
```

Ir a FIN

Punto 8. DEFINIR LOS NOMBRES DE LOS DIRECTORIOS

1. Ejecutables

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

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

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

El usuario puede ingresar un nombre simple como "bin" o un subdirectorio como `/so7508/tp/binarios`

Reservar este path en la variable `DIRBIN`

2. Maestros y tablas

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

```
Defina el directorio de Maestros y Tablas ($grupo/mae):
```

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



El usuario puede ingresar un nombre simple o un subdirectorio

Reservar este path en la variable **DIRMAE**

3. Recepción de novedades

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

Defina el directorio de recepción de novedades (\$grupo/nov):

Proponer /nov y si el usuario lo desea cambiar, permitiré.

El usuario puede ingresar un nombre simple o un subdirectorio

Reservar este path en la variable **DIRREC**

4. Aceptados por el demonio

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

Defina el directorio de Archivos Aceptados (\$grupo/ok):

Proponer /ok y si el usuario lo desea cambiar, permitiré.

El usuario puede ingresar un nombre simple o un subdirectorio

Reservar este path en la variable **DIROK**

5. Procesados por el proceso de imputación de gastos

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

Defina el directorio de Archivos Procesados (\$grupo/imp):

Proponer /imp y si el usuario lo desea cambiar, permitiré.

El usuario puede ingresar un nombre simple como "imp" o un subdirectorio como /so7508/tp/presupuesto/imputacion

Reservar este path en la variable **DIRPROC**

6. Reportes

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

Defina el directorio de Reportes (\$grupo/rep):

Proponer /rep y si el usuario lo desea cambiar, permitiré.

El usuario puede ingresar un nombre simple o un subdirectorio

Reservar este path en la variable **DIRINFO**

7. Log

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

Defina el directorio de log (\$grupo/log):

Proponer /log y si el usuario lo desea cambiar, permitiré.

El usuario puede ingresar un nombre simple o un subdirectorio

Reservar este path en la variable **DIRLOG**

8. Rechazados

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

Defina el directorio de rechazados (\$grupo/nok):

Proponer /nok y si el usuario lo desea cambiar, permitiré.

El usuario puede ingresar un nombre simple o un subdirectorio

Reservar este path en la variable **DIRNOK**

Punto 9. VERIFICAR ESPACIO EN DISCO.



Se debe solicitar al usuario que indique cual es el espacio mínimo libre requerido en el directorio para la recepción de novedades

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

```
Defina espacio mínimo libre para la recepción de archivos 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

Chequear si hay disponible el espacio requerido. Si no hay, mostrar y grabar en el log

```
Insuficiente espacio en disco.  
Espacio disponible: xx Mb.  
Espacio requerido $DATASIZE Mb  
Inténtelo nuevamente.
```

Volver a verificar espacio en disco.

Si hay suficiente espacio, mostrar y grabar en el log

```
Suficiente espacio en disco.  
Espacio disponible: xx Mb.  
Espacio requerido $DATASIZE Mb  
De enter para continuar.
```

Punto 10. MOSTRAR VALORES

Limpiar la pantalla y Mostrar los valores de los parámetros configurados

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

```
Directorio de Configuración: $GRUPO/dirconf (mostrar path y listar  
archivos)  
Directorio de Ejecutables: ($GRUPO/DIRBIN mostrar path y listar archivos)  
Directorio de Maestros y Tablas: ($GRUPO/DIRMAE mostrar path y listar  
archivos)  
Directorio de Recepción de Novedades: ($GRUPO/DIRREC mostrar path)  
Directorio de Archivos Aceptados: ($GRUPO/DIROK mostrar path)  
Directorio de Archivos Procesados: ($GRUPO/DIRPROC mostrar path)  
Directorio de Archivos de Reportes: ($GRUPO/DIRINFO mostrar path)  
Directorio de Archivos de Log: ($GRUPO/DIRLOG mostrar path)  
Directorio de Archivos Rechazados: ($GRUPO/DIRNOK mostrar path)  
Estado de la instalación: LISTA  
Desea continuar con la instalación? (Si - No) _
```

Si el usuario indica No, volver atrás

Si el usuario indica Si confirmar inicio de instalación

Punto 11. VOLVER ATRAS

Si el usuario indica No, Limpiar la pantalla y realizar nuevamente la Validación 6: Definir los nombres de los directorios pero esta vez, los valores default propuestos deben cambiarse por los recientemente ingresados por el instalador.



Por ejemplo si para los ejecutables el usuario indicó "mis_comandos_del_tp" entonces ahora el default no debe ser bin sino mis_comandos_del_tp porque es lo último que ingreso el usuario para ese directorio

Punto 12. CONFIRMACIÓN DE LA INSTALACIÓN

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

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

Si el usuario indica Si, Continuar en el paso: "Instalación"

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

Punto 13. INSTALACIÓN

1. crear las estructuras de directorio requeridas

Mostrar y grabar en el log el siguiente mensaje

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

Como resultado de la instalación la estructura de **\$GRUPO** debe ser la siguiente:

- **\$GRUPO/DIRBIN** en donde se depositarán los scripts ejecutables
- **\$GRUPO/DIRMAE** en donde depositarán los archivos maestros y tablas
- **\$GRUPO/DIRREC** sin archivos, lo usaremos para simular la recepción de archivos de novedades
- **\$GRUPO/DIROK** sin archivos, lo usaremos para depositar los archivos aceptados por el demonio
- **\$GRUPO/DIRPROC/proc** sin archivos, lo usaremos para depositar los archivos procesados por el proceso de imputación de gastos
- **\$GRUPO/DIRINFO** sin archivos, lo usaremos para depositar los Reportes que genera el comando Perl luego de la imputación de gastos
- **\$GRUPO/DIRLOG** sin archivos, lo usaremos para depositar los Archivos de Log
- **\$GRUPO/DIRNOK** para depositar los Archivos Rechazados

2. Mover los ejecutables y funciones

Mostrar y grabar en el log el siguiente mensaje

```
Instalando Programas y Funciones
```

3. Mover los archivos maestros y tablas

Mostrar y grabar en el log el siguiente mensaje

```
Instalando Archivos Maestros y Tablas
```

Punto 14. ARCHIVO DE CONFIGURACIÓN

Se debe almacenar la información de configuración del sistema en el archivo **Instalep.conf** en **\$GRUPO/dirconf** grabando un registro para cada una de las variables indicadas durante este proceso.

Mostrar y grabar en el log el siguiente mensaje

```
Actualizando la configuración del sistema  
Instalación CONCLUIDA.
```

Punto 15. FIN

Borrar archivos temporarios, si los hubiese generado

Grabar en el log el siguiente mensaje

```
Fin del proceso. Usuario Fecha y Hora
```



Cerrar el archivo [Instalep.log](#) y Terminar el proceso

Initep

Input

- Archivo de Configuración [\\$GRUPO/dirconf/EPLAM.conf](#)

Output

- Log del Comando [\\$GRUPO/DIRLOG/Initep.log](#)

Opciones y Parámetros

- A especificar por el desarrollador

Descripción

El propósito de este comando es preparar el ambiente del sistema y dejarlo listo para su ejecución.

- Es el primero en orden de ejecución
- Se dispara manualmente
- Graba en el archivo de Log a través del [Logep](#)
- Invoca, si corresponde, el siguiente proceso: [Demonep](#)

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 y los archivos maestros todos ellos con los permisos adecuados.

Continúa con la asignación de valor a un conjunto de variables de ambiente que van a ser usadas por el resto del sistema y luego ofrece arrancar automáticamente el comando [Demonep](#)

No se puede ejecutar ningún comando si la inicialización de ambiente no fue realizada, es por ello que los comandos JAMAS deben acceder al archivo de configuración para conocer directorios o el contenido de las variables del sistema, sino que deben usar las variables de ambiente que define este proceso.

Pasos Sugeridos

Punto 1. VERIFICAR SI EL AMBIENTE YA HA SIDO INICIALIZADO

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

Si se intenta ejecutar más de una vez en la misma sesión de usuario, no permitirlo y explicar la situación mostrando y grabando en el log el siguiente mensaje:

```
Ambiente ya inicializado, para reiniciar termine la sesión e ingrese nuevamente
```

y terminar la ejecución.

Punto 2. SETEAR LAS VARIABLES DE AMBIENTE

Setear la variable PATH y cualquier otra variable de ambiente que considere necesarias

[GRUPO](#) [DIRBIN](#) [DIRMAE](#) [DIRREC](#) [DIROK](#) [DIRPROC](#) [DIRINFO](#) [DIRLOG](#)

Si no se puede efectuar esta tarea, mostrar mensaje explicativo y Terminar la ejecución

Punto 3. VERIFICAR LOS PERMISOS

Si se detecta que algún archivo no tiene los permisos adecuados explicar la situación con un mensaje, configurar los permisos de los archivos correctamente.

Si no se puede efectuar la corrección, mostrar mensaje explicativo y Terminar la ejecución



Cuando todos los archivos tienen los permisos adecuados, Mostrar y Grabar en el log usando la función **Logep** el siguiente mensaje

Estado del Sistema: INICIALIZADO

Punto 4. VER SI SE ARRANCA EL DEMONIO

Initep debe ofrecer la posibilidad de arrancar el demonio **Demonep**

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

¿Desea efectuar la activación de Demonep? Si - No

Punto 5. NO ARRANCAR

Si el usuario no desea arrancar el demonio **Demonep**, entonces explicar cómo hacerlo manualmente

Punto 6. SI ARRANCAR

Si el usuario desea arrancar el demonio **Demonep**, activarlo y explicar cómo detenerlo manualmente

Mostrar mensaje y grabar en el log

Demonep corriendo bajo el no.: <Process Id de Demonep>

Punto 7. LOG

Cerrar el archivo de log- Terminar el proceso

Punto 8. NUEVA SESIÓN, NUEVO INICIO

Al cerrar la sesión de usuario todos los valores de las variables de ambiente deben perderse, para poder ejecutar los comandos será necesario volver a inicializar el sistema

Demonep

Input

- Maestros y Tablas **\$GRUPO/DIRMAE**
- Novedades **\$GRUPO/DIRREC/<nombre del archivo>**

Output

- Archivos Aceptados **\$GRUPO/DIROK/<nombre del archivo>**
- Archivos Rechazados **\$GRUPO/DIRNOK/<nombre del archivo>**
- Log del Comando **\$GRUPO/DIRLOG/Demonep.log**

Descripción

El propósito de este comando es detectar la llegada de archivos al directorio **\$GRUPO/DIRREC** y aceptar o rechazar estos archivos según corresponda

- Es el segundo en orden de ejecución
- Es un proceso del tipo "Demonio" :
- Se dispara con **Initep** o manualmente
- Se detiene manualmente
- Mueve los archivos a través del **Movep**
- Graba en el archivo de Log a través del **Logep**
- Invoca, si corresponde, el siguiente proceso: **Procep**

El Proceso se inicia con la detección de la presencia de archivos en el directorio **\$GRUPO/DIRREC**



Si el nombre del archivo (filename) cumple con el formato de nombre esperado y el archivo es de texto, el archivo se acepta, de lo contrario se lo rechaza.

También verifica si hay archivos ya aceptados para arrancar automáticamente el proceso [Procep](#)

Luego duerme un tiempo x y vuelve a empezar, es decir, que a menos que se detenga, este proceso no tiene condición de fin.

A este tipo de programas se los denomina demonio, daemon o dæmon (de sus siglas en inglés Disk And Execution Monitor).

Otra característica de los procesos del tipo demonio, es que se ejecutan en segundo plano en vez de ser controlado directamente por el usuario (es un proceso no interactivo).

Pasos Sugeridos:

Punto 1. OBLIGACIÓN DE INICIAR ANTES DE EJECUTAR CUALQUIER COMANDO

Si el ambiente no fue inicializado, no ejecutar

Mostrar mensaje adecuado y terminar el programa

Punto 2. MANTENER UN CONTADOR DE CICLOS DE EJECUCIÓN DE DEMONEP.

Se debe mantener un contador de ciclos de ejecución del demonio, grabar en el log el número de ciclo.

```
Demonep ciclo nro. 1
```

Punto 3. CHEQUEAR SI HAY ARCHIVOS EN EL DIRECTORIO \$GRUPO/DIRREC

Si existen archivos en el directorio [\\$GRUPO/DIRREC](#)

Grabar en el log el nombre del archivo detectado

```
Archivo detectado: < nombre del archivo >
```

Punto 4. VERIFICAR QUE EL ARCHIVO SEA UN ARCHIVO COMÚN, DE TEXTO.

Los archivos que no sean de texto se rechazan, si este fuera el caso, grabar en el log

```
Archivo rechazado, motivo: no es un archivo de texto
```

Punto 5. VERIFICAR QUE EL ARCHIVO NO ESTÉ VACÍO.

Los archivos vacíos se rechazan, si este fuera el caso, grabar en el log

```
Archivo rechazado, motivo: archivo vacío
```

Punto 6. VERIFICAR QUE EL FORMATO DEL NOMBRE DEL ARCHIVO SEA CORRECTO

FORMATO CORRECTO: ejecutado_<año_presupuestario>_<cod_provincia>_<animesdia>.csv

Los archivos con nombres que no se correspondan con el formato de nombre esperado, se rechazan. Si este fuera el caso, grabar en el log

```
Archivo rechazado, motivo: formato de nombre incorrecto.
```

Punto 7. VERIFICAR AÑO

- Año presupuestario debe ser el año corriente
- Los archivos con error en el año, se rechazan. Si este fuera el caso, grabar en el log

```
Archivo rechazado, motivo: año xxxx incorrecto.
```

Punto 8. VERIFICAR PROVINCIA

- Cod_provincia debe existir en el maestro de provincias
- Los archivos con error en la provincia, se rechazan. Si este fuera el caso, grabar en el log

```
Archivo rechazado, motivo: provincia xx incorrecta.
```



Punto 9. VERIFICAR FECHA

- ANIOMESDIA debe ser una fecha válida
- ANIOMESDIA debe ser menor o igual a la fecha del día pero mayor o igual a la fecha de inicio del año presupuestario.
- Los archivos con error en la fecha, se rechazan. Si este fuera el caso, grabar en el log

```
Archivo rechazado, motivo: fecha aaaammdd incorrecta.
```

NOTA: si detecta otro tipo de error clasificable, puede incluir su validación.

Punto 10. ACEPTAR LOS ARCHIVOS CON NOMBRE VÁLIDO.

Si el nombre del archivo es válido Grabar en el log el mensaje de archivo aceptado

```
Archivo aceptado
```

Invocar la función Movep para mover el archivo aceptado al directorio DIROK

Punto 11. RECHAZAR LOS ARCHIVOS INVÁLIDOS

Invocar la función Movep para mover el archivo rechazado al directorio DIRNOK

Punto 12. VER SI SE ARRANCA EL Procep

Si en el directorio DIROK existen archivos aceptados (del presente ciclo o de ciclos anteriores), **Demonep** debe arrancar el **Procep** siempre que éste no esté corriendo.

Si arranca, grabar en el log

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

Punto 13. NO ARRANCAR

Si correspondía arrancar **Procep** pero se debe posponer la ejecución de **Procep** porque ya hay un **Procep** corriendo, grabar en el log:

```
Invocación de Procep pospuesta para el siguiente ciclo
```

Punto 14. DORMIR UN TIEMPO X Y EMPEZAR UN NUEVO CICLO

El tiempo X para la primera entrega puede ser un tiempo largo, de varios minutos, el tiempo x para la segunda entrega debe ser un tiempo corto, no mayor al minuto.

Función Logep

Opciones y Parámetros

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

Descripción

¿Qué es un Log?

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

Es usado para registrar información sobre cuándo, quién, dónde, qué y por qué un evento ocurre para una aplicación, proceso o dispositivo. Es empleado por los profesionales de IT, auditoría y seguridad informática.

A estos 5 valores se los llama estándar W5, por su origen en ingles: when, who, where, what and why

- WHO: ¿Quién?



- Usuario, es el login del usuario
- WHEN: ¿Cuándo?
 - Fecha y Hora, en el formato que deseen y calculada justo antes de la grabación.
- WHERE: ¿Dónde?
 - Comando (parámetro 1), nombre del comando o función que genera el mensaje.
 - Se apreciará la utilidad de este parámetro por ejemplo cuando la función Movep deba generar mensajes en el log del comando llamador
- WHAT: ¿Qué?
 - Tipo de Mensaje (Parámetro 3) valores posibles:
 - INFO = INFORMATIVO: mensajes explicativos sobre el curso de ejecución del comando. Ej: Inicio de Ejecución
 - WAR = WARNING: mensajes de advertencia pero que no afectan la continuidad de ejecución del comando. Ej: Archivo duplicado
 - ERR = ERROR: mensajes de error Ej: Archivo Inexistente.
- WHY: ¿Por qué?
 - Mensaje (Parámetro 2)

Requerimientos

- El directorio de log está determinado por la variable de ambiente LOGDIR
- El nombre del archivo de log es igual al nombre del comando y extensión .log
- Grabar un archivo distinto para cada comando, en el lugar indicado y con el nombre adecuado.
- La escritura de archivos de log debe ser homogénea para todos los comandos
- Cada registro de log debe responder al estándar W5.
- Si el archivo de log no existe, se debe crear. Si existe se le deben agregar los nuevos registros **siempre**
- Se debe controlar el crecimiento del archivo de log
 - En todo sistema, es importante evitar el crecimiento INDISCRIMINADO de los archivos de Log. Es por ello que esta función debe prever un mecanismo para controlarlo y evitarlo
 - En el log deben permanecer siempre los mensajes mas recientes
 - Cada vez que se hace la reducción del tamaño del archivo, señalar la situación con el mensaje "Log Excedido" en el propio log
 - **aclare en Hipótesis y Aclaraciones Globales cual fue el mecanismo que adoptó.**
 - **Ejemplo sencillo:** cuando un archivo de log supera un tamaño X, el archivo se trunca dejando las últimas xx líneas

Función **Movep**

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 centralizar el movimiento de archivos que deben realizar la mayor parte de los comandos de este sistema para evitar diferentes políticas en el tratamiento de archivos duplicados.

En líneas generales el sistema no borra ningún archivo, los mueve de un lugar a otro, en el contexto de este TP se deben CONSERVAR todos los archivos aun cuando:

- sea un archivo improcesable, roto, dañado, vacío;
- sea un archivo con un nombre incorrecto, con espacios, mal formado;
- ya haya sido procesado
- al moverlo nos encontremos que ya existe otro archivo del mismo nombre en ese lugar.

Requerimientos

- Mover el archivo solicitado al directorio indicado sin alterar su contenido
- Si en el destino ya existe otro archivo con el mismo nombre (nombre de archivo duplicado), no debe fracasar la operación, la función debe poder conservar ambos.
 - Crear un subdirectorio /dpl para depositar el archivo duplicado y moverlo allí
 - Si también en /dpl ya existe otro archivo con el mismo nombre, emplear un número secuencial (desde 1 hasta n) para modificar el nombre del archivo y ponerlo como complemento final de la extensión: Ejemplo: <nombre del archivo original>.nnn donde nnn es el número de secuencia
- Si esta función es invocada por un comando que graba en un archivo de log, registrar el resultado de su uso en el log del comando

Premisas:

- No se puede definir un archivo auxiliar solo para registrar el número de secuencia, usar el archivo de configuración del sistema para estos propósitos
- Si el origen y el destino son iguales, no mover y registrar en el log el error
- Si el origen no existe, no mover y registrar en el log el error
- Si el destino no existe, no mover y registrar en el log el error
- Indicar en las Hipótesis Globales cuantos números de secuencia emplea: uno por directorio o uno para toda la instalación

Procep

Falta especificar

Listep

Falta especificar

Estructuras y Archivos

Maestro de Centros Presupuestarios: DIRMAE/centros.csv

Centro de Presupuesto	Nombre
0.0.0.1	Unidad Ministro
0.0.0.1-1	Secretaria de Coordinacion Ministerial
0.0.0.1-1-1	SubSubsecretaria de Administracion General
0.0.0.1-1-1.1	Direccion General Alfa



0.0.0.1-1-1.2	Dirección General Prima
0.0.0.1-1-2	SubSecretaría de Infraestructura Ministerial
0.0.0.1-2	Secretaría de Trabajo
0.0.0.1-2-1	SubSecretaría de Fiscalización
0.0.0.1-2-1.1	Dirección General Baires
0.0.0.1-2-1.2	Dirección General Sancor
0.0.0.1-2-1.3	Dirección General Norte
0.0.0.1-2-1.4	Dirección General Sur
0.0.0.1-3	Secretaría de Empleo
0.0.0.1-3-1	SubSecretaría D
0.0.0.1-3-1.1	Dirección General Delta
0.0.0.1-3-2	SubSecretaría E
0.0.0.1-3-2.1	Dirección General Epsilon

Maestro de Provincias: [DIRMAE/provincias.csv](#)

Código de provincia	Provincia	Región
1	CABA	Baires
2	Buenos Aires	Baires
3	Catamarca	Norte
4	Chaco	Norte
5	Chubut	Sur
6	Córdoba	Sancor
7	Corrientes	Norte
8	Entre Ríos	Norte
9	Formosa	Norte
10	Jujuy	Norte
11	La Pampa	Sur
12	La Rioja	Norte
13	Mendoza	Sur
14	Misiones	Norte
15	Neuquén	Sur
16	Rio Negro	Sur
17	Salta	Norte
18	San Juan	Sur
19	San Luis	Sur
20	Santa Cruz	Sur
21	Santa Fe	Sancor
22	Santiago del Estero	Norte
23	Tierra del Fuego	Sur
24	Tucumán	Norte

Tabla de Trimestres: [DIRMAE/trimestres.csv](#)

Año Presupuestario	Trimestre	Desde	Hasta
2016	Primer Trimestre 2016	10/12/2015	9/3/2016
2016	Segundo Trimestre 2016	10/3/2016	9/6/2016
2016	Tercer Trimestre 2016	10/6/2016	9/9/2016



2016	Cuarto Trimestre 2016	10/9/2016	9/12/2016
2016	Anual 2016	10/12/2015	9/12/2016
2017	Primer Trimestre 2017	10/12/2016	9/3/2017
2017	Segundo Trimestre 2017	10/3/2017	9/6/2017
2017	Tercer Trimestre 2017	10/6/2017	9/9/2017
2017	Cuarto Trimestre 2017	10/9/2017	9/12/2017
2017	Anual 2017	10/12/2016	9/12/2017

Archivos de Log: *DIRLOG*/*<nombre del comando>*.log

Campo	Descripción /Fuente/Valor	
Quien	Caracteres	Es el login del usuario que graba el registro
Cuando	Fecha y hora	Formato a Elección Es la fecha y hora en el momento de grabación del 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.
Porque	Caracteres	Lo determina el programador.

Separador de campos: - guion

Ejemplo: 20160905 19:53:22-Sandra-*Demonep*-WAR-No se pudo mover el archivo

En la variable de ambiente LOGSIZE se tiene el tamaño máximo en Kbytes que puede alcanzar un archivo de log

Archivo de Configuración */dirconf/Instalep.conf*

Campo	Descripción/Fuente/Valor	
Variable	Caracteres	Valores posibles: GRUPO, DIRBIN, DIRMAE, etc
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=09/04/2015 10:03 p.m

Se debe grabar un registro por cada variable. Luego de los registros requeridos, puede agregar todos los registros que desee

NOTA: más archivos en la próxima entrega