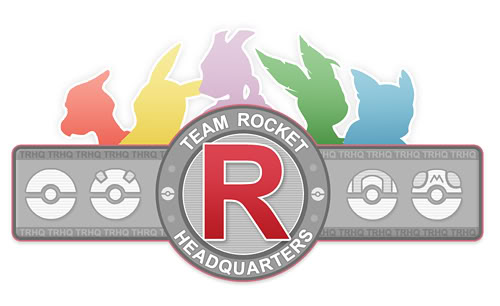
|  |  |
| --- | --- |
|  | **2016 – 2C** |
|  | Sistemas Operativos [-] TP\_GRUPAL  Universidad Tecnológica Nacional  http://actividades.frba.utn.edu.ar/imagenes/logo-con-margenes.jpg |

****

|  |
| --- |
| **[TRABAJO PRÁCTICO so]**  **GRUPO: TEAM ROCKET | 2C2016**  Beghe, Eduardo Jose [-] 150.107-0 [-]  Funti, Lucas [-] 149.899-0 [-]  Santos, Maximiliano [-] 142.559-6 [-] |
| Suchecki, Emiliano Gabriel [-] 143.538-3 [-]  Tobio, Maximiliano [-] 120.421-0 [-] |

****

Índice

Introducción…………..……………………..…………………………………………………………………………………………2

ANEXO PLAN DE ACCION ..…………..………………………………………………………………………………………….3

ANEXO TEST BENEFICIO DE PUBLICACION……………………………………….……………………………………….4

ANEXO TEST ABM VISIBILIDAD….…………………………….……………………………………………………………….6

**Introducción:**

El presente documento tiene como objetivo documentar el proyecto llevado a cabo por nuestro grupo, durante el transcurso del cuatrimestre. En el mismo se podrá encontrar una información más detallada acerca del plan de acción que elegimos (Estrategia), las estructuras de datos elegidas, información acerca de cómo se resolvió las partes de sockets, sincronización, semáforos, file system y otras decisiones de diseño consideradas relevantes.

**Plan de Acción (Estrategia):**

Etapa 1:

* Leer enunciado y hacer una puesta en común de las interpretaciones.
* Tener un feedback de la interpretación en las charlas introductorias y ajustar ideas.
* Hacer un esquema (borrador) de cómo vamos a comunicar los distintos procesos.

Etapa 2:

* Comenzar a codear los procesos cumpliendo las funcionalidades básicas (moverse, dibujar, leer los archivos de configuración, loguear sucesos, etc.)
* Cumplir con los hitos mencionados para el primer CheckPoint.

Etapa 3:

* Establecer un protocolo de comunicación por sockets, homogéneo a todos los procesos, que será usado como base.
* Cumplir con los hitos mencionados para el segundo CheckPoint.
* Empezar a interconectar los procesos por sockets y validar su correcto funcionamiento.

Etapa 4:

* Estudiar y readaptar las nuevas consideraciones debidas al cambio de enunciado.
* Intentar cumplir Hitos del checkpoint 3 y 4, en tiempo.

Etapa 5:

* Hacer el desafío OSADA F.S. y chequear nuestra implementación.
* Hacer los testeos finales del TP, ajustándolo a las pruebas solicitadas.

Etapa 6:

* Entregar.

**Nota: Transversalmente se buscó lograr una documentación del proyecto continua y actualizada, además de buscar una participación activa de todos los integrantes del grupo.**

**Decisiones de diseño**

Debido a la poca disponibilidad de tiempo que tenía el equipo, se decidió posponer el uso de las herramientas recomendadas, solo ante una situación de extrema necesidad o para la etapa final de testeo, en lugar de utilizarlas recurrentemente. Entre las herramientas mencionadas tenemos a:

* Para los casos de prueba ( cUnit ).
* Para el manejo de la memoria Valgrind.
* Para el manejo de los hilos Helgrind.

Nota: Entendemos que esto es una mala decisión.

Recursos Críticos:

Se definió que el proceso mapa utiliza semáforos únicamente para las colas de planificación, evitando así problemas de concurrencia.

Sockets:

Se definió un protocolo de comunicación básico a partir del cual se comenzó a implementar la comunicación por sockets. Por defecto, se utilizan estructuras ENUM para identificar los mensajes y se decidió implementar una solución para “serializar” los mensajes. La misma consiste en leer primero el HEADER y en base al tipo de mensaje leer el siguiente byte, el cual contiene el tamaño total del PIPELINE y en tercer lugar leer los datos que se decidieron transmitir.

**Estructuras de datos**

**Proceso Mapa:**





Proceso Entrenador:

asfdas

**ANEXO CONSTANTES**

En el presente anexo se pretende aclarar que simbolizan cada una de las macros tipo objetos que se utilizaron en el tp, para crear los nombres simbólicos de las constantes.

1. **Árbol de directorios (Estructura PokeDex)**

|  |  |
| --- | --- |
| **Config\_Example!**  ├── Entrenadores -  │ │  │   └── Ash -  │ │  │   ├── Dir de Bill -  │   └── metadata  │  │  └── Mapas -  │  └── CiudadPaleta -  ├── medalla-CiudadPaleta.jpg  ├── metadata  │  └── PokeNests -  │  ├── squirtle -  │   ├── squirtle001.dat  │   └── metadata  │  └── pikachu -  ├── metadata  └── pikachu001.dat  **8 directories, 7 files** | **Config implemented in C!**  ├── \_\_ubicacionEntrenadores -  │ │  │   └── Ash -  │ │  │   ├── \_\_ubicacionDirDeBill -  │   └── \_\_ubicacionMetadataEntrenadores  │  │  └── \_\_ubicacionMapas -  │  └── CiudadPaleta -  ├── medalla-CiudadPaleta.jpg  ├── \_\_ubicacionMetadataMapas  │  └── \_\_ubicacionDirPokeNest -  │     ├── squirtle -     │   ├── \_\_ubicacionMetadataPokeNest     │   └── \_\_ubicacionPokemonesPokeNest  │     └── pikachu -     ├── \_\_ubicacionMetadataPokeNest     └── \_\_ubicacionPokemonesPokeNest      **8 directories, 7 files** |

1. **Estructura interna de un archivo config**

├── \_\_nombreMetadataEntrenadores -

**│   ├── (desestimadoElNombreDelEntrenador)**

**│   ├── \_\_nombreEnConfig\_HojaViajes**

**│   ├── \_\_nombreEnConfig\_ObjMapa**

**│   ├── \_\_nombreEnConfig\_Reintentos**

**│   ├── \_\_nombreEnConfig\_Simbolo**

**│   └── \_\_nombreEnConfig\_Vidas**

**│**

├── \_\_nombreMetadataMapas -

**│   ├── \_\_nombreEnConfig\_Algoritmo**

**│   ├── \_\_nombreEnConfig\_Batalla**

**│   ├── \_\_nombreEnConfig\_Deadlock**

**│   ├── \_\_nombreEnConfig\_IP**

**│   ├── \_\_nombreEnConfig\_Puerto**

**│   ├── \_\_nombreEnConfig\_Quantum**

**│   └── \_\_nombreEnConfig\_Retardo**

**│**

├── \_\_nombreMetadataPokeNest -

**│   ├── (desestimadoElTipoDePokeNest)**

**│   ├── \_\_nombreEnConfigIdPokeNest**

**│   └── \_\_nombreEnConfigPosicionPokeNest**

**│   └── \_\_separadorEnConfigPosicionPokeNest**

**│**

└── \_\_ubicacionPokemonesPokeNest -

**├── \_\_valorFinalPokemonesPokenest**

**└── \_\_valorInicialPokemonesPokenest**

**└── \_\_nombreEnConfigNivelPkmn**

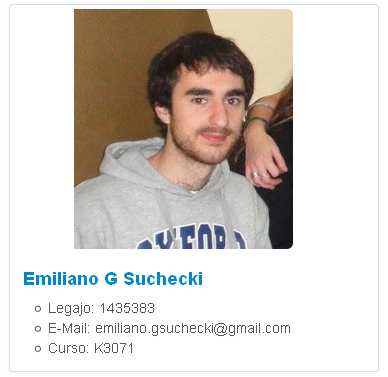
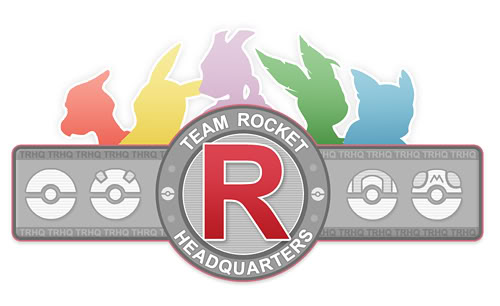
**5 config files, 19 parameters**

1. **Gráficos en mapa**

**ANEXO PLANILLA DE CONSULTAS**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ITEM | FECHA | DESCRIPCION / PREGUNTA | PROCESO | OBSERVACIONES | RTA |
| \*\*\* INIT \*\*\* | | | | | |
| 1 | 25-ago | ¿Cómo son los movimientos en el mapa? ¿ Que pasa si no se puede mover en diagonal? | M | ¿ X ; Y ; X ; Y ; X ; Y ? | 27/08-> Santiago: SI. Si no puede alternar se mueve solo en un eje. |
| 2 | 25-ago | ¿Quién valida la metadata del Entrenador? El proceso entrenador | E |  | 27/08 -> Emiliano: Se indució que si. |
| 3 | 25-ago | ¿En que posicion inicia el jugador en un mapa? | M |  | 27/08 -> Santiago: En la 0;0. 10/09 -> Se actualizo a la pos 1;1 (Issues) |
| 4 | 25-ago | ¿Hay que aplicar los 2 algoritmos ? | M | RR / SDRF | 27/08 -> Matias: Si. |
| 5 | 25-ago | Si el modo batalla no esta activo, que sucede? | M |  | 27/08 -> Matias: Quedan inter bloqueados hasta killear a uno (p.ej.) |
| 6 | 25-ago | Dudas varias de interpretacion de enunciado y secuencia | E / M / P | - | - |
| \*\*\* CHECKPOINT #1 \*\* | | | | | |
| 7 | 1-sep | Los identificadores (entrenador y pokenest) deben ser unicos por mapa? | M | Pokenest (si x enunciado) y los entrenadores [SI]? | 10/09 -> Martin: Si. |
| 8 | 2-sep | Chequear las keys de los archivos de config | E / M | <commons/config.h> es sensible a mayus./minus. | 10/09 -> Martin: Esperar a los archivos de test y ajustar. 03/10: Ya estan!! |
| 9 | 3-sep | Chequear los nombres de los procesos | M / P | Chequear sus argumentos y el orden tambien | 10/09 -> Martin: Esperar a los archivos de test y ajustar. 03/10: Ya estan!! |
| 10 | 3-sep | El "tipo=electrico" en la metadata de la pokenest, que utilidad tiene? Puedo tener un pikachu y un raichu en una pokenest? | M | (ambos tipo electrico). [NO] | 10/09 -> Martin: NS/NC 25/09: Emiliano: (Se desestimó) |
| 11 | 4-sep | Argumento "nombreEntrenador" y "nombreMapa" pueden contener espacios? | E / M | [No, ya que C lo interpreta como 2 parametros] | 10/09 -> Martin: Si, los argumentos con espacios se pasan con comillas " " 25/09: (Se desestimó) |
| 12 | 6-sep | Los nombres de las pokenest, respetan los usados en la pokemon factory? | M | [SI] | 10/09 -> Martin: Si, Esperar a los archivos de test y ajustar. 03/10: Ya estan!! |
| 13 | 6-sep | Es correcto recorrer todas las pokenest de un mapa según la pkmon\_factory? O debería recorrer el directorio y cargar lo que encuentro? | M |  | 10/09 -> Emiliano: Ya se resolvió usando iteración de directorios. |
| 14 | 6-sep | Cuando se le envía la señal SIGUSR2 al mapa, también se pueden modificar las pokenest? | M | [Solo Metadata] | 10/09 -> Martin: Solo relee la metadata y solo cambia los algoritmos de planif. |
| 15 | 6-sep | Si E le dice a M, moverme y llega a la pokenest, tiene que esperar otro turno para capturar? | E |  | 10/09 -> Martin: Si para el RR. 27/09 -> Hay nuevo enunciado. |
| 16 | 7-sep | Si en una pokenest tengo 5 pokemon, cuando un E lo captura se puede copiarse uno random? (entre los 5? -> sería posible que 2 entrenadores tengan el pkmn 3 en simultaneo) o les tengo que asignar el 1/2/../5? | M | Me evito validar que Pokemon se capturo y cual no. | 10/09 -> Martin: No, hay que asignar un pokemon de manera univoca a cada entrenador. |
| 17 | 7-sep | Solo los entrenadores involucrados en el deadlock hacen la batalla pokemon? O todos? | M | [Solo involucrados] | 10/09 -> Martin: Solo los involucrados. |
| 18 | 7-sep | En caso de DeadLock, solo se muestra msj en pantalla en el E, o tmb en M? | E / M | [Solo Entrenador] | 10/09 -> Martin: Buena pregunta, consultar en Issues. |
| 19 | 7-sep | ¿Cómo Debuggear el mapa en eclipse? (Por la GUI / ncurses) | M | 16/07 -> Emiliano: Encontre el Issue pero no me funciono.. | 10/09 -> Martin: Hay un Issues que habla sobre eso. |
| 20 | 7-sep | Como pasar los parametros que recibiria por consola (argv, argc), pero desde el eclipse al darle a Run o la araña del Debug. | M | - | 11/09 -> Emiliano: Listo!! |
| 21 | 10-sep | Dudas varias de File System OSADA. | P | - | - |
| \*\*\* CHARLA FUSE \*\* | | | | | |
| 22 | 16-sep | Para el manejo de señales, deberia tener una variable global? Es preferible que sea el mapa, o una generica? | M |  | 17/09 -> Damian: Si usar var\_global tipo vida =1. |
| 23 | 16-sep | ¿Cómo conozco el tiempo de ejecucion de un proceso? | E |  | 17/09 -> Damian: Usar librería <time.h> |
| 24 | 16-sep | ¿Qué se recomienda hacer para no abrir un mapa duplicado? | M | Socketes / Consultar PIDs / Mediante archivos | 17/09 -> Damian: Utilizar Sockets y consultar x puerto ocupado. |
| 25 | 16-sep | ¿Hay un solo planificador, no? | M | [SI] | 17/09 -> Damian: Si. |
| 26 | 16-sep | ¿El mapa deberia borrarle los pokemones al entrenador? | E / M | [NO]. | 17/09 -> Damian: No, cuando inicialice el entrenador que los borre. |
| 27 | 16-sep | ¿Qué pasa si un entrenador esta en un deadlock y no tiene pokemones?  ¿Como batalla? | M |  | 17/09 -> Damian: Tecnicamente no esta en un deadlock, esta en starvation. |
| 28 | 16-sep | ¿Qué significa exactamente lo de reaccionar de manera favorable? | M |  | 17/09 -> Damian: Desconectar el entenador, liberar pkmn, salir del mapa. |
| 29 | 16-sep | ¿Estaria bien implementar los pokemons disponibles como listas? | M |  | 17/09 -> Damian: Si. Utilizar listas. |
| 30 | 16-sep | ¿Cuál es la utilidad de la cola de finalizados? | M |  | 17/09 -> Damian: Verificar el correcto funcionamiento del planificador. |
| 31 | 16-sep | ¿Cuánto tiempo deben permanecer en la cola de finalizados? | M |  | 17/09 -> Damian: Lo que uds consideren. Lo pueden dejar ∞ o borrarlo. |
| 32 | 16-sep | ¿El entrenador tambien debe leer la metadata del mapa para conocer la IP y el Puerto? | E | [SI] | 17/09 -> Damian: Buena pregunta, consultar en Issues. En principio Si. |
| \*\*\* CHECKPOINT #2 \*\* | | | | | |
| 33 | 30-sep | Según el nuevo enunciado, ¿es correcto que el entrenador todo el tiempo manda msjs y el mapa solo atiende al que a el le parece? | M |  | 01/10 -> Martin: No. El mapa le otorga turno. |
| 34 | 30-sep | El entrenador debe tener algun retardo para retransmitir el msj? | E |  | 01/10: Emiliano: (Se desestimo) |
| 35 | 30-sep | Según el nuevo enunciado, ¿es correcto que cuando el E finaliza los objetivos se desconecta y EL busca la medalla y se la copia? | E / M | Se elimino el msj fin de objetivos. | 01/10 -> Martin: Buena pregunta, consultar en Issues. |
| 36 | 30-sep | Chequear entender el planificador | M | - | - |
| 37 | 30-sep | ¿Cómo es la captura de pkmn? Quantum consume 1 unidad y SRDF lo manda a bloqueados, pero si habia disponibles que pasa? | M |  | 01/10 -> Martin: Quantum Si. SRDF pasa a bloqueados hasta que se chequee que haya disponibles y los asigne. |
| 38 | 30-sep | En SRDF, hay un entenador que solicita capturar pkmn (y hay) -> Se lo otorga. SRDF lo vuelve a priorizar (x paso 1) y es el más cercano (x paso 2), entonces sigue ejecutando él? | M |  | 01/10 -> Martin: Si. Puede darse. |
| 39 | 30-sep | Estamos trabajando con varios hilos, para el caso de una señal CTRL + C, ¿Cómo me aseguro que todos los hilos se cierren? | M | Capturo el PID y le digo que se cierre? | 01/10 -> Martin: Si. Pthreads tmb lo gestiona. |
| 40 | 30-sep | Actualmente no estamos bloqueando las señales, ¿Qué problema puedo tener? | E / M / P | Por los sockets, El sleep, etc. | 01/10 -> Martin: En principio ninguno. Continua prox. ejecucion. |
| 41 | 30-sep | Es conveniente utilizar "volatile sig\_atomic\_t" para un flag de señal? | E / M / P |  | 01/10 -> Martin: En principio no. |
| 42 | 30-sep | Para la batalla pokemon es correcto crearme un nuevo pokemon factory a partir del nombre del archivo y luego consultar el nivel? | E / M |  | 01/10 -> Martin: Si. |
| 43 | 30-sep | Saben cuándo va a estar el lote de pruebas OSADA? | E / M / P |  | 01/10 -> Martin: No.  03/10: Ya estan!! |
| 44 | 30-sep | El pokedex cliente debe ser un proceso que lo ejecutas y se cierra (quedando en background) o queda abierto mostrando printfs? | P |  | 01/10 -> Martin: Background |
| 45 | 30-sep | Para que se utilizan los comandos MD5SUM y TOUCH sugeridos? Hay que implementarlos en c? | P | Para el server sin interfaz gráfica? | 01/10 -> Martin: Para el server. |
| 46 | 30-sep | Dudas varias de File System OSADA. | P | - | - |
| \*\*\* CHECKPOINT #3 \*\* | | | | | |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

**ANEXO INTEGRANTES GITHUB**

****