**IMPLEMENTACION SUPERCOMPUTADORA**

**GRUPO SUPERCOMPUTADORA**

**LIDER: EDGAR TORRES COY**

**PROFESOR: JOHN EDWARD CASTRO BERNETH**

**FUNDACION UNIVERSITARIA SAN MATEO**

**INGENIERIA DE SISTEMAS**

**COMPUTACION PARALELA Y DISTRIBUIDA**

**BOGOTA**

**2019**

Contenido

[**PROPUESTAS DEL TRABAJO A REALIZAR** 2](#_Toc20946917)

[**Propuesta 1:** 2](#_Toc20946918)

[**Propuesta 2:** 2](#_Toc20946919)

[**Propuesta 3:** 3](#_Toc20946920)

# **PROPUESTAS DEL TRABAJO A REALIZAR**

## **Propuesta 1:**

Se ejecutará el desarrollo de una súper computadora mediante equipos físicos los cuales estarán interconectados mediante un switch y cables de red, permitiendo su comunicación, los equipos contaran con sistema operativo basado en Linux (se sugiere Debian o Pelican).

La metodología que se usara para la construcción de la súper computadora será Scrum ya que es una metodología ágil que nos permite acoplar los procesos de desarrollo que se requieren para la realización del proyecto.

Su arquitectura será la SOA (Arquitectura orientada a servicios), dado que su meta es aportar la flexibilidad mediante la automatización de infraestructuras y herramientas necesarias para ejecutar la creación de una súper computadora.

Como aporte a la materia este trabajo nos permitirá evidenciar físicamente la forma cómo se comporta una supercomputadora, mediante la estructura propuesta, y compararla con un solo clúster, con la finalidad de verificar su desempeño y sus bondades en la automatización de procesos.

## **Propuesta 2:**

Comprobar mediante el uso de máquinas virtuales el desempeño que tiene una supercomputadora, configurando un equipo como master y los demás como nodos, para brindar al master las fuerzas necesarias para ejecutar más eficientemente los procesos, los cuales usaran un sistema operativo basado en Linux.

La arquitectura bajo la cual se desarrollara será SOA (Arquitectura orientada a servicios), la cual nos brinda flexibilidad en la automatización de los procesos que ejecutan cada uno de los nodos al servicio del equipo Master.

La metodología que se pretende usar para el desarrollo de la supercomputadora es Scrum la cual nos brinda la posibilidad de llevar en mejor forma las actividades para el desarrollo del trabajo.

Como aporte a la materia, la demostración de la virtualización de las maquinas nos permitirá ver en forma más clara como se pueden compartir recursos entre varios equipos para mejorar el desempeño de una máquina.

## **Propuesta 3:**

La metodología Ágil que se pretende implementar es KANBAN ya que esta es similar a la metodología SCRUM, no poseemos historia de usuario, nos permite el trabajo en paralelo y la asignación de tareas con sus respectivos estados (To Do, Doing, Don)

Descripción del desarrollo del trabajo

* La duración de los Sprint será equivalente a 7 días
* Las tareas que se identifiquen serán asignadas dependiendo de las habilidades que tenga cada uno de los integrantes
* Los Dailys se realizaran a través del grupo de wathsapp (si se es requerido) cada uno tendrá duración de 15 minutos aprox. En el cual se expondrá las tareas en las que se está trabajando y se presentaran los posibles impedimentos

Arquitectura (backend)

Teniendo en cuenta la necesidad es viable la arquitectura SOA, que nos permite la distribución de los servicios en diferentes servidores para el trabajo en paralelo. Se pretende crear un bus de servicios, esto con el fin de centralizar y gestionar la comunicación entre los diferentes servicios, estos podrían estar alojados en dockers para poder suministrar los recursos necesarios para que cada servicio funcione de manera satisfactoria y en caso de fallar no afecte el funcionamiento de los otros.

Arquitectura (Infraestructura)

Balanceador, para la distribución de peticiones entre las máquinas que se vayan a implementar.

Si es posible, X cantidad de máquinas con las mismas características. En caso contrario, X cantidad de máquinas virtualizadas.

Tecnologías y Herramientas

• Servicios REST con dotNet Core para garantizar la interoperabilidad entre sistemas (Windows y Linux).

• Dockers, la implementación de estos es viable en Windows y Linux.

El aporte que pretendemos dar a la clase es el de realizar por medio de una comprobación física o virtual las bondades de la creación de supercomputadoras y como al realizar cualquiera de las tres propuestas estamos realizando una comprobación de los procesos educativos y los temas vistos durante el desarrollo de la materia.

# **PROPUESTA SELECCIONADA**

La propuesta seleccionada para realizar el trabajo es la # 1, teniendo en cuenta lo expuesto en la misma se realizaran las siguientes actividades para poder cumplir con la propuesta:

1. Revisión bibliográfica y estudio de las bases de funcionamiento de los clusters.
2. Selección del tipo adecuado para las necesidades del grupo (MOSIX).
3. Selección del equipo mínimo necesario para implementar el cluster elegido.
4. Adquisición y prueba del hardware seleccionado.
5. Instalación de placa de Red adicional en el nodo maestro.
6. Cableado general y puesta a punto de la conexión del servidor a la red local de la FIUNER.
7. Instalación y conexión de los nodos mediante un Switch.
8. Instalación del sistema operativo Linux en ambas PCs
9. Instalación del núcleo del sistema MOSIX en ambas PCs
10. Configuración de todos los servicios y protocolos de red necesarios.
11. Configuración y puesta a punto del MOSIX
12. Instalación y puesta a punto del MPI
13. Instalación y puesta a punto del PVM
14. Instalación y puesta a punto del sistema de colas de procesos (PBS).
15. Instalación y puesta a punto del sistema de administración remota (Webmin).
16. Instalación de paquetes, lenguajes y herramientas específicas para cálculo numérico.
17. Prueba del sistema con diferentes aplicaciones (software de redes neuronales, programas de cálculo pesado, etc.)
18. Instalación de clientes X (entorno gráfico) y SSH para acceso desde las terminales Windows.
19. Instalación de los servicios para el acceso a través de Internet, desde fuera de la FIUNER.
20. Documentación del proceso (versiones preliminares de este informe)

|  |  |  |
| --- | --- | --- |
| **Rol** | **Responsabilidades:** | **Entregable:** |
| Tester (Jhon Jairo López Sáez) | Encargado de realizar diferentes pruebas desde el primer prototipo - modelo hasta el final en lineamiento con la arquitectura y metodología colaborativa propuesta; asi commo el registro de los cambios y mejoras en la evolución en el tiempo. | Diagramas y resultados de pruebas obtenidas. |
| Testing (Brayam Estiven Duran Cardona) | Encargado de realizar pruebas funcionales y no funcionales con el fin de detectar defectos en el uso de la plataforma | Pruebas documentadas realizadas (test configuration, set de pruebas) |
| Implementador (David Gerardo Urrego Camargo) | Encargado de realizar y verificar los componentes del proyecto de acuerdo con la metodología y la arquitectura establecida al inicio del proyecto, al igual será el responsable de implementar un prototipo y generar un reporte respecto. | Documentación Técnica |
| Desarrollador (Jhonatan Stiv Cuadros Cárdenas) | Encargado de implementar todos los requerimientos relacionados en el documento ya sea con un software o con la implementación y/o configuración de la supercomputadora establecida en el documento, siguiendo los parámetros de la metodología. | Implementación |
| Diseñador (Anderson Rubio) | Delegado de la implementación de la metodología, relaciona el diseño con la implementación incluyendo el nivel de detalle esperado en el diseño antes de que proceda la implementación.  Así como la verificación de los requerimientos para el desarrollo del proyecto.  creación de un concepto de sistema que ayude a cumplir los objetivos del proyecto | Diagramas de estructura  Requerimientos de Vista metodología de proyecto |
| Analista (Tatiana Pinzon) | Encargada de analizar la información para el levantamiento de los requerimientos y las necesidades para la ejecución del proyecto. | Matiz de requerimientos |
|  |  |  |

* Parte 2 de la investigación del proyecto realizado.

**PVM (Parallel virtual Machine)**

En su traducción al español Máquina Virtual Paralela, es un sistema o conjunto de bibliotecas que se usa principalmente dentro de la programación paralela, la cual me permite realizar una conexión entre dos o más equipos de cómputo, con el objetivo integrar los recursos de estas computadoras, para finalmente obtener el esquema de una sola máquina.

Este software opera en diferentes plataformas como lo pueden ser (Unix y Windows), aunque para la creación del proyecto se usara o se implementara dentro de la plataforma Linux.

Algunas de las funciones del uso del PVM consiste en la mensajería asincrónica y para el control de procesos entre los diferentes nodos presentes dentro del sistema, además que permite la escalabilidad del mismo.

**Ventajas:**

Entre las diferentes librerías existentes para el envío de mensajes en el paralelismo es uno de los más fáciles de usar.

Control arbitrario de dependencias de estructura, esto permite que la aplicación determine: (Donde y cuando Iniciar o terminar tareas, que maquinas se añaden o se eliminan desde la máquina virtual y finalmente que tareas se puede comunicar y/o sincronizar con otras.)

**Desventajas:**

En gran medida el rendimiento del esquema dependerá de la capacidad de procesamiento de los ordenadores o nodos que se encuentran dentro del esquema.

Lo que es una ventaja también se puede convertir en una desventaja, ya que, aunque es muy fácil de usar, esto conlleva a que sea algo deficiente en cuanto al paso de mensaje entre nodos.