**¿En qué consiste nuestra idea?**

En una aplicación distribuida que comprime archivos usando un cluster formado por 3 raspberrys pi como servidores.

**Funcionamiento de nuestra aplicación:**

**Listado de opciones a implementar:**

**Listado de opciones que no estuvieron al alcance:**

**¿Qué es RabbitMQ?**

Es un software de negociación de mensajes de código abierto que implementa el estándar AMQP, está considerado como middleware de mensajería. La capacidad para operar de forma distribuida y conmutar ante errores son posibles gracias al framework OTP, su servidor está escrito en Erlang

**¿Por qué escogimos RabbitMQ?**

Porque al ser código abierto, no requerimos de una licencia para trabajar con dicho software, aparte es fácil de configurar y los nodos en la red pueden ser fácilmente escalados, por último, posee varias interfaces de programación debido a que soporta múltiples protocolos de mensajes.

**Librerías usadas:**

**Celery:**

Aplicación que permite crear tareas de trabajo asíncronas gestionadas por un gestor de colas con el fin de enviar mensajes de forma distribuida. Celery puede trabajar con operaciones en tiempo real y con la calendarización de tareas, esto último consiste en trabajar con tareas que se ejecutarán en un momento determinado o de manera periódica. Cada tarea es una unidad de ejecución, y estas se ejecutan de forma concurrente en uno o más nodos de trabajo.

**Implementación del código usando Celery:**

**Manejo de errores:**

**Mecanismos de sincronización:**

**Repositorio de GitHub:**

**Bibliografía:**

<https://es.wikipedia.org/wiki/RabbitMQ>

<https://www.rabbitmq.com/>

<https://www.apsl.net/blog/2011/01/14/introduccion-a-celery/>