Group E Exercise2 Plan

The Service We Are Going to Implement:
A distributed memory cache service.

The basic idea is to store key-value data in multiple machines' memory and give access to other services in the whole system. So our service acts as a simple version of Redis with distributed features. Imagine a configuration file in a server. When the whole system is distributed, we need a kind of middleware to store crucial data (like configs) and can be accessed or even updated by other parts in the system.

Key-value pairs in the system will try to be duplicated and spread among the nodes. Nodes in the system will send a heartbeat to others at periodic intervals so that the status of each node can be tracked in the system. Any write or update will be an ACID operation and will use Write-ahead logs to achieve the same.

A similar tool (the 'perfect' version of our project) exists:
etcd github.com/etcd-io/etcd

Communication Protocol:
TCP/IP

Failure Scenarios:

- Media Failure / Data Store Corruption
- Network Partition
- Single Point Failure (leader/not leader)
- Adding New Nodes
- Data Management (Update/Delete)

There are many complicated situations not mentioned. We will try to identify and come up with solutions during the implementation.

Programming Language:
Java, Shell

The specific design of our system is subject to change.

Group Member Signatures:
SINGH CHAUDHARY GUROOSH GABRIEL

MENON SUJESH SURESH

HYNES ANDREW

CHEN CHAO