# Requirements Analysis – coPlay Distributed System

This document presents the requirement analysis for the coPlay distributed system project. It includes functional and non-functional requirements for two scenarios: (1) a large organisation with 100 users, and (2) an internet-scale application with 1 billion users. This analysis will also be a guide for design and testing phases.

## 1. Functional Requirements

* Users must be able to send and receive real-time chat messages.
* Users must be able to interact with a Tower of Hanoi game interface (e.g. clicking towers).
* Each connected peers in the game should receive correctly broadcasted messages.
* Web clients must check the backend periodically to fetch updates (chat/game updates specifically).
* Webapp must respond to POST messages and GET requests for tower clicks.
* Consistent updates must be shown on each webapp for messages and tower states.
* The system should support a smooth shutdown during development.

## 2. Non-functional Requirements

* **Latency Tolerance:** The system should handle unbounded but non-Byzantine message delays.
* **Scalability:** The architecture should allow adding more peers without significant slowdown of the system.
* **Fault Detection:** Webapps should be able to detect missing or delayed updates.
* **Consistency:** All connected clients should see the same game/chat state.
* **Reliability:** The system should recover from temporary lags as quickly as possible.

## 3. Scenario 1 – Organisation (100 users)

This scenario considers a LAN-based deployment for a large organisation hosting around 100 users. These users are likely in a close geographic area, which signifies low network latency and high availability.

Requirements:

* Fast message delivery with minimal lag using ZMQ sockets.
* Consistent state update across peers using simple PUSH/PULL sockets.
* Handling of message reordering via already known systematic broadcasting or controlled delays.
* Testing for faults like network lag and order shuffling.
* Persistent state recovery is not required after a crash (restart not required).

## 4. Scenario 2 – Internet (1 billion users)

This scenario envisions internet-scale deployment accessible globally. It requires much higher fault tolerance, message routing intelligence, and coordination mechanisms.

Requirements:

* Use of coordination service like Zookeeper to manage the game state across the available nodes.
* Support for ZNodes for tracking user sessions.
* Leader election or broadcast trees to avoid flooding.
* Support for clients from all over the world with different latency levels (delays).
* Partial tolerance to message duplication, delay, and loss.

## 5. Conclusion

The requirements outlined here will also coordinate the system design and test planning. While all requirements may not be implemented in the prototype due to technical or time constraints, this analysis defines the expectations for both small- and large-scale deployments of the coPlay system.