# CS 410/510
# Large Scale Systems

## Assignment 4:
## Implementing Consensus Protocols

**Overview:**
In this course, our goal is to create **MiniSpanner**. MiniSpanner takes inspiration from **Google's Spanner** database. Despite being created nearly two decades ago, Spanner still acts as the backbone for all Google's applications.

In this course, over **a span of four assignments**, you will be creating MiniSpanner. Thus, all the assignments are **incremental** as you will get an opportunity to use your existing code in your current assignment.

**Goal:**
The goal of your final assignment is to add a consensus protocol to your sharded replicated architecture from previous assignment

**CS 410** students will add to their sharded replicated architecture **Paxos** consensus protocol.
**CS 510** students will add to their sharded replicated architecture **Proof-of-Execution (PoE)** protocol.

**Key Challenges:**

- In the last assignment, you created a sharded-replicated architecture where within each shard you were PBFT protocol and across shards you ran 2PC/3PC protocol. Each shard has a leader and the leader is responsible on reaching consensus via Paxos protocol. To run the 2PC/3PC protocol, you made the leaders of the shards communicate with each other.

**Key Expectations:**

- In this assignment, you are **expected** to modify the PBFT protocol into your Paxos/PoE protocol. You can also replace PBFT with your own protocol design!
- Such a change may require adding some additional functions to ResilientDB.
- The overall sharded-replicated architecture should **remain unchanged**! Remember all the goals of Assignment 3 still apply.

- You are **expected** to test your implementation on this updated design. Ideally, you should observe a throughput increase.
- You are **not expected** to implement any failures, failure-recovery, checkpointing, or garbage collection. We will only test for the good case—when there are no failures.

**Assignment Tasks:**
Following are the tasks that you need to perform as part of this assignment.

1. Implement your architecture.
2. Measure the throughput of your commit protocol at the coordinating shard.

**Deliverables:**
As part of this assignment, you are expected to provide the following deliverables.

1. A latex report explaining how you implemented Paxos/PoE in your sharded-replicated architcture. Please provide details of what functions you modified or added to the existing codebase.
2. In your latex report include snapshots of any function that you modified. These snapshots should show your changes.
3. Please state your machine configuration and the observed throughput/latency.