**CS240B Class Project Ideas:**

1. Remote Desktop Services (RDS - backend) and Virtual Desktop Infrastructure (VDI - frontend) - (aka Terminal Services)
2. Hyper-V Replication/Live Migration
3. Server/Service Clustering
4. Multi-tree Active Directory (minimum 2 trees and 1 child domain)
5. IIS service (HTTPS with a fully functioning service)
6. Microsoft SCCM (System Center Config Mgr) – requires MS SQL
7. Containers, Docker, & Orchestrators

**Short Description:**

1. RDS/VDI (Virtual Windows desktop):
   1. Use RDS & VDI technologies to build, test, and demo a virtual MS Windows desktop environment. Access a hosted Windows environment on a central server from remote devices. Could be fat (MS Widows, Mac, or Linux), or thin (Chrome or thin client device).
   2. Could also host individual MS Windows apps (i.e., MS Word, MS Excel, etc.) from a central server through a browser.
   3. **NOTE:** With remote/virtual everything (telecommunicating, online schooling, etc.), VDI is becoming popular.
2. Hyper-V Live Migration:
   1. Use Hyper-V Live Migration to build, test, and demo a live server service (i.e. AD DS, DNS, etc.) migration from one server to another with no downtime.
   2. **NOTE:** Enterprises love Live Migration. Even in a traditional Data Center, servers are refreshed every 3-5 years. Now, we can move the services to a different server (inside the same DC, across DCs, and/or to cloud) without downtime.
3. Microsoft SCCM:
   1. Use MS System Center Config Manager (SCCM) to build, test, and demo distributing and upgrading apps, as well as HW/SW inventory.
   2. Could also be used for other purposes (i.e., licensing, remote control, network security, etc.)
   3. **NOTE:** One can use GPOs to distribute apps; however, GPOs are not an excellent fit for this job. TSCCM is the preferred enterprise tool. Could be too complex for small companies. The benefits are amazing.
4. Clustering (all three models):
   1. Build, test, and demo all three clustering models:
      1. Active-Standby or Failover model
      2. Load balancing model
      3. High Performance (HPC) model
   2. **NOTE:** With the increasing need for High-Availability, lower computing cost, and virtualization, clustering has become the norm! Fewer and fewer IT dept are still asking “do we cluster?”.
5. Multi-Forest-n-Trees AD (Theron):
   1. Build, test, and demo two forests AD environment:
      1. Forest-1 = has two trees with no children (i.e. americas.com and europe.com)
      2. Forest-2 = has one tree with a child domain (asia.com and japan.asia.com).
   2. From a Widows PC that belongs to one of the above domains, what does the above look like and can be used for.
   3. **NOTE:** Trust within a forest (across trees and children) is transitive/automagic, and trust across forests is manual (can be one-way or two-ways).
6. Containers:
   1. Deploy an app/service using MS Containers {not just spin up and show container - Containerize}
   2. Can use PowerShell, Docker, or both {Docker is highly recommended}
   3. Can use Windows Containers and/or Hyper-V Container types {bare minimum, explain the difference in the slides}
   4. Can use whatever Windows Server 2019 flavor {nano, headless, datacenter, etc.}
   5. **NOTE:** Don’t expect you to write anything – find an app/service that can be Containerized.

**NOTE:** I’ll be happy to meet with each team to help define/refine the project scope.

**FINALLY:** The larger the team, the higher the expectations.