**Project 1:**

Initially, we operated from a single Data Center, but we logically segregated it into four environments: User Acceptance Testing (UAT), System Integration Testing (SIT), End-to-End (E2E), and Production. As we plan and prepare for the construction of the new Data Center, our aim is to implement more advanced and token-based authentication.

Having the necessary hardware in place, we conducted thorough testing of Openshift, Openstack, and related applications for our clients and customers. We meticulously documented every endpoint and communication matrix on the Confluence page, capturing details and creating diagrams for the flow of each application.

To optimize traffic distribution, we configured a mechanism in the Load Balancer (LTM) that evenly distributes traffic in a round-robin fashion between DC1 and DC2. This strategy ensures high availability (HA) and enables effective management of peak traffic flows. We also rigorously tested failover scenarios within the same setup.

This entire project spanned over a year to accomplish, taking into account the complexities of setting up a new Data Center, implementing advanced authentication methods, and ensuring the reliability and scalability of our infrastructure.

**Project 2:**

Large-scale web application programming and deployment are executed following the DevOps lifecycle (SDLC) process.

The source code is developed within the developer's IDE and then committed to "Gitlab," serving as our Source Code Management (SCM) and Continuous Integration (CI) tool. The Jenkins Automation Server is employed to retrieve the source code, progressing through various CI/CD phases, including checkout, build, release, testing, and deployment.

Testing is conducted in Test environments configured through "Ansible," a Configuration Management Tool. These environments, specifically designed for testing, are constructed using "Docker" containers. The containers, in turn, are orchestrated by the "Kubernetes" engine, ensuring large-scale redundancy, scalability, monitoring, and recovery.

This infrastructure aligns with industry-standard Development (DEV), User Acceptance Testing (UAT), System Integration Testing (SIT), and Production (PRD) environments. The setup is crafted with Ansible and Docker, orchestrated by the Jenkins Automation Server.

Monitoring of all these clusters and infrastructures is carried out using the "Nagios Monitoring Tool," providing a comprehensive report accessible via a centralized console or dashboard.

The entire process adheres to industry standards of the CI/CD pipeline, covering phases from code checkout to deployment, with automation triggers seamlessly integrated from Git to Jenkins.