Architecture Document

Cloud Service Provider:

The cloud service provider we will be using is Amazon Web Services. It fits our uses because of the ease of scalability and offered services covering our specific needs for this project. The cost aspect is also a positive with the reduction compared to other cloud service providers because of the learner lab accounts we are using for the class. Along with that this is also a provider which the team generally has some experience with. We will be using several of the services they offer over the course of building our project such as EC2, S3, and RDS. EC2 will provide us with the scalable servers we will use to host our project on. S3 is their cloud storage which we will rely on. Then RDS is their database service which we will be utilizing for that aspect of our project as well.

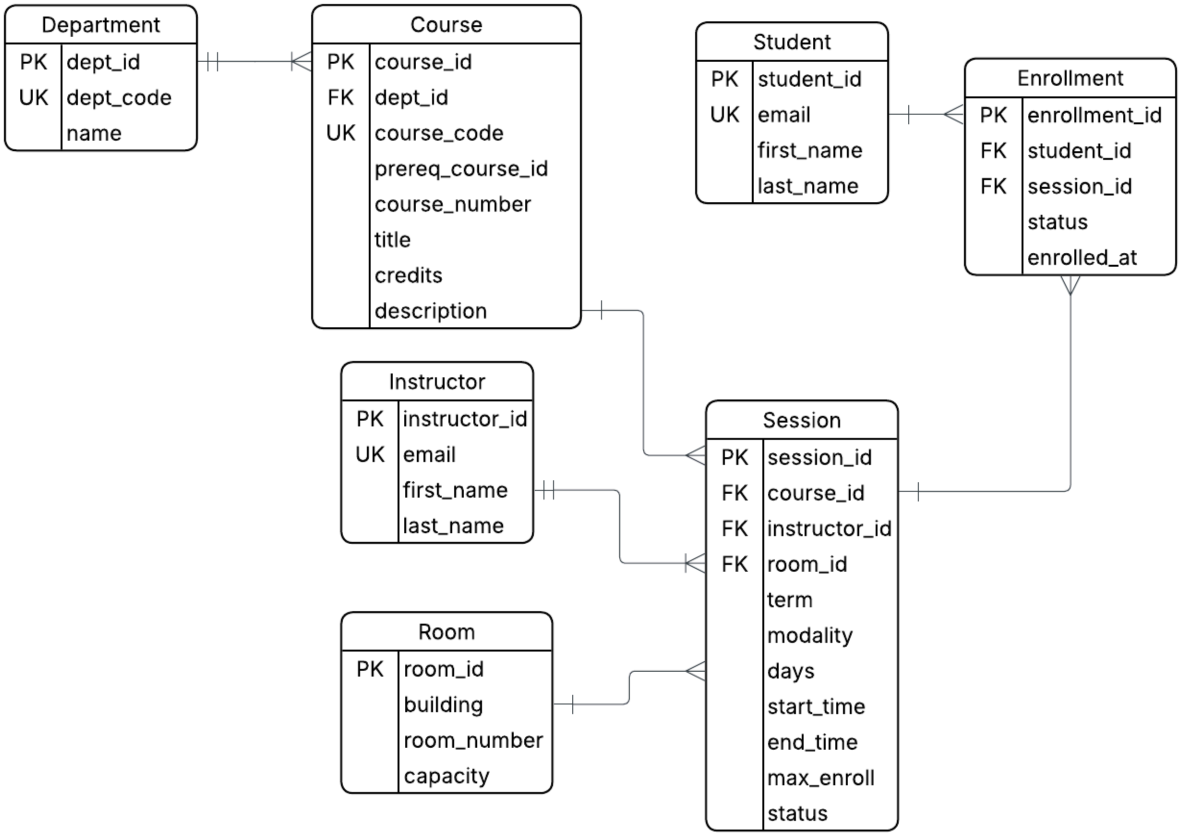
Application Design:

For the building of our application we will be writing using the Javascript programming language as it is something our team has experience with and lends itself to our use case. The run-time environment we will utilize is node. The application API we will utilize is the RESTFul api. These decisions were decided mostly based on experience among the members of the team. For the application framework and middleware we will use Angular and Express. This decision was based on the recommendations provided by the professor along with our own judgement. Express will be helpful for us to create scalable and efficient server-side applications

Operating System and Virtual Servers:

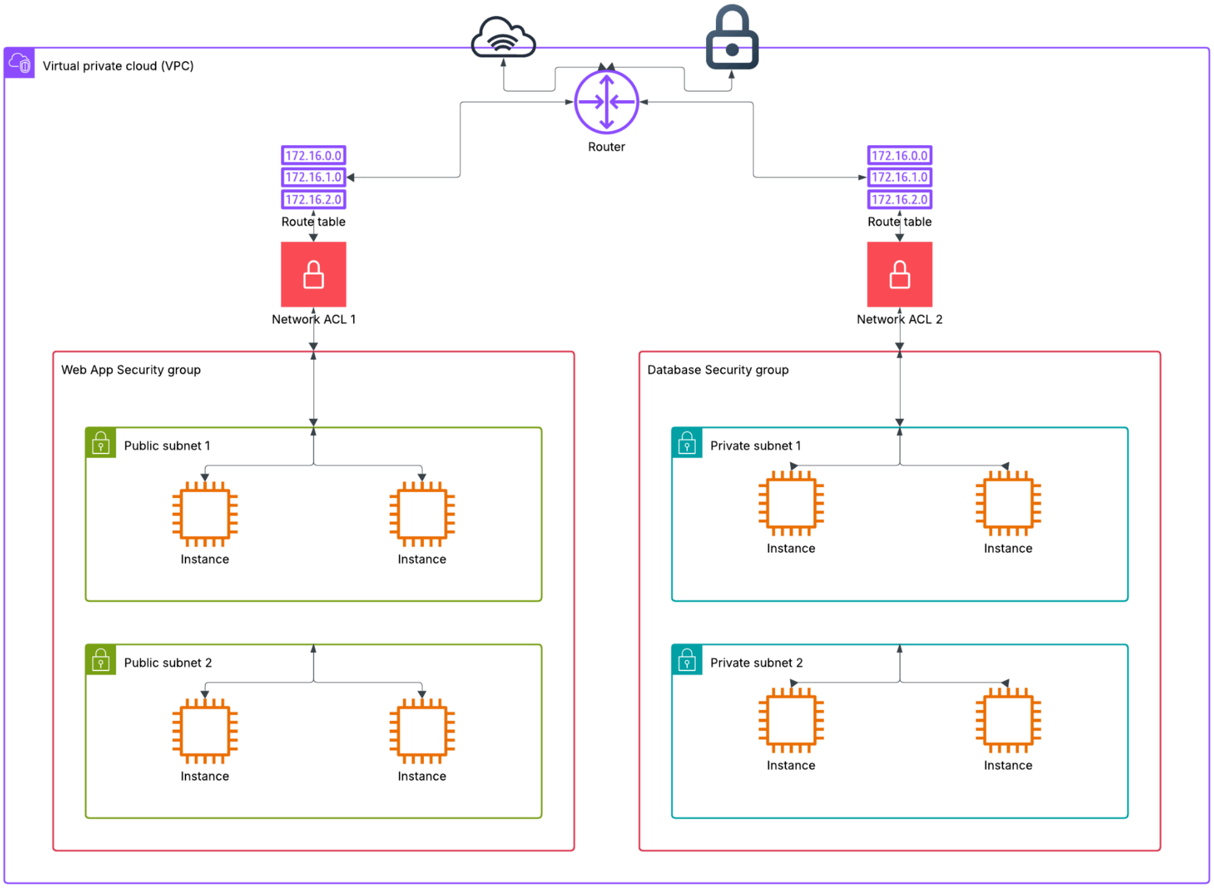
We will be utilizing Windows for our virtual servers on AWS. This was decided based on the group overall having more experience with Windows operating systems than Linux rather than a decision based on any particular functionality. For our use case we will use T3 instances for our servers as they are a general use type designed for applications with moderate CPU usage that experience temporary spikes in use. The temporary spikes being in our case whenever a large number of students try to sign up for classes at the same time. T3.micro instances come with 1 GiB of memory, 2 vCPU and 30 GiB of storage by default which we believe will be effective for our project use case.

Database Design:



The database management system we will be utilizing for this project is MySQL. This was decided primarily based on the recommendations provided, but also because we believe that it will effectively fit our needs.

Network Architecture and Design:



Our network for this project will follow a basic MVC framework architecture. We will have security groups for our web and application servers and another for our database servers. The web application security group will have the ports 80 for HTTP, 443 for HTTPS, 8080 for our application, ICMP, remote desktop protocol, and SSH open. The database security group will need the port 3306 for MySQL. We will not be setting any egress rules while we are controlling the ingress in our use case. The web and application security group will cover our public subnets and the database security group will cover our private subnets. We will have multiple subnets to assure resiliency and availability.

Data Visualization Tool:

The data visualization tool we will use is Tableau, as its dashboard functionality will be beneficial. It was chosen based on the team's general familiarity with the software compared to Power BI. It is a popular tool in the industry and fits our use case perfectly. Along with that there is a license for students which we can use for the project which will make cost not an issue.

Testing and Quality Assurance Process:

We will perform testing and quality assurance processes during our sprints over the course of this project to make sure that our production code is valid. Unit Testing will be performed for individual modules in order to make sure that they work as expected for the greater application. Along with that we will perform integration testing to make sure that these modules are able to function properly when integrated together, making sure they connect and work together in the way we desire. Then we will perform End-to-End Testing to make sure that our application is functioning as a complete system from a user perspective instead of just looking at the different parts.

Authentication and Authorization Process:

Our architecture for our system in this project will require authentication because only students should be able to register for classes and only instructors should be able to see who is in their classes and what classes they are teaching. There will be several roles within the application, such as student, instructor, and then administrator. Students will be able to view, search, and register for classes. Instructors will be able to see what classes they are teaching and see what students are taking their classes. The administrator role should be able to create classes and assign instructors to them.

Team Responsibilities

Evan Arnold

Project Manager: Coordinates team activities and ensures timely submission of Deliverables.

Network Engineer: Designs the network architecture, including firewalls and Subnets.

Christian Owusu-achiaw

Application Developer: Designs the application’s runtime environment, APIs, and port configurations.

Alex Doan

Database Architect: Develops the database schema and ensures compliance with third-normal form

Riyah Miah

Cloud Architect: Leads the selection and configuration of cloud resources.

QA Analyst: Develops the testing plan and tools for validating the system's functionality.