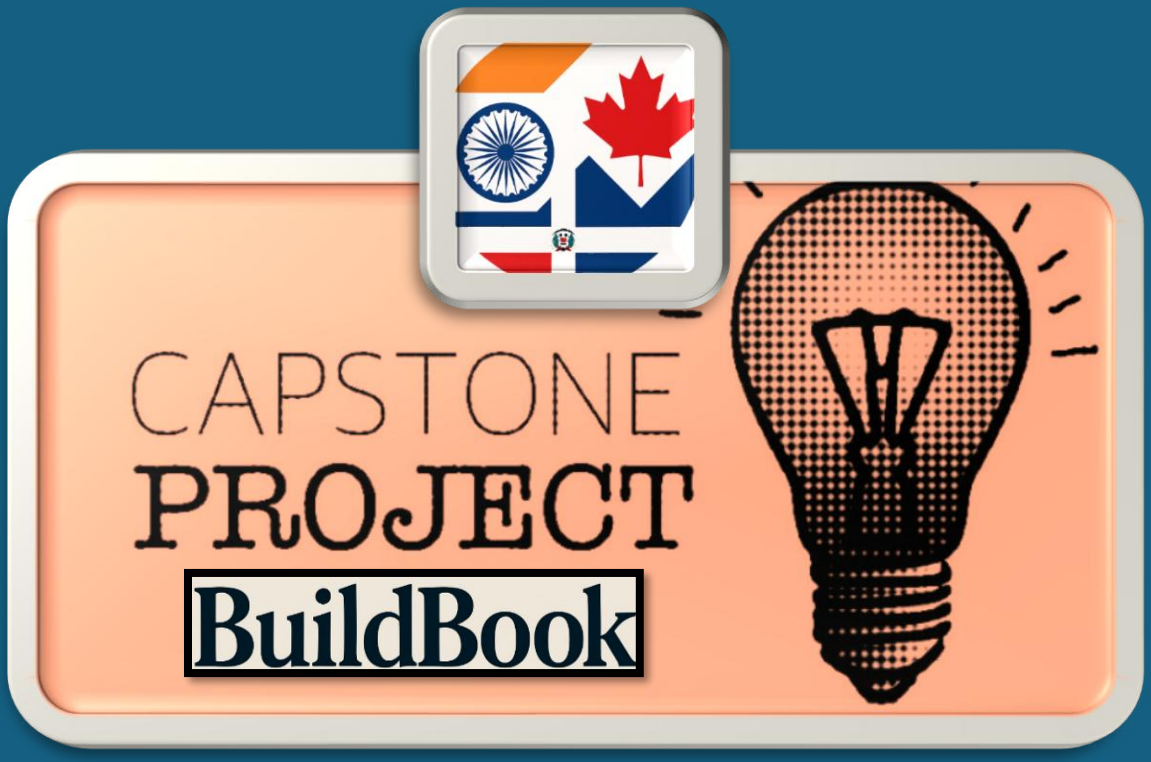
CONESTOGA COLLEGE



GROUP 3

GENAO **CHRISTOPHER**

SHUKLA **HARSHUL**

PATEL **MEET**

PATEL **NISHITKUMAR**

PROFESSOR

HALLMAN **DREW**

APRIL **2025**

"THE FEAR THAT WE WILL LOSE IS DIFFERENT FROM THE FEAR OF HAVING LOST."

TABLE OF CONTENTS

INTRODUCTION

Expanding **INDOCAN**, a web development company, into **New York** and **Hong Kong** requires a **well-structured IT infrastructure** to support efficient business operations. This project aims to design and implement a **scalable, secure, and high-performance** network that ensures seamless collaboration between both locations.

The IT framework will focus on:

- **Server Virtualization & Deployment** – Implementing a **highly efficient** virtualized environment for optimized resource management.
- **Network Infrastructure** – Designing a **resilient, high-speed** network to support critical business operations.
- **Secure Interoffice Connectivity** – Establishing a **VPN-based** secure communication channel between New York and Hong Kong.
- **End-User Computing** – Providing **tailored computing solutions** for developers and office staff.

This project ensures **future scalability, security, and operational efficiency**, enabling INDOCAN to **seamlessly expand and optimize its global presence**.

PROJECT OVERVIEW

The successful expansion of INDOCAN into **New York and Hong Kong** requires a **robust and well-planned IT framework**. The project will involve:

1. **Hardware & Virtualization** – Selecting **high-performance** server components, including **multi-core processors, large memory capacity, and SSD storage** for virtualization.
2. **Network Architecture & Security** – Implementing **firewalls, VLAN segmentation, and QoS** to ensure **optimal performance and security**.
3. **Interoffice Communication** – Configuring **VPN tunnels** and **failover mechanisms** to maintain **secure and reliable** communication between the offices.
4. **Testing & Optimization** – Conducting **extensive testing** for **performance, reliability, and scalability** before full deployment.

REQUIREMENTS AND DESIGN

PLANNING

For a successful expansion, **INDOCAN** must carefully consider multiple factors, including **budget, connectivity, infrastructure, security, and resource management**. The planning phase involves defining the overall **network architecture, hardware requirements, software configurations, and security measures** necessary to support efficient business operations across the **New York and Hong Kong** offices.

SCOPE AND OBJECTIVES

Scope:

The project will cover **planning, implementation, and ongoing support** to establish a scalable and efficient IT infrastructure for INDOCAN's new offices.

Objectives:

1. **End-User Computing** – Recommend and deploy reliable **hardware solutions** tailored for developers and non-developers.
2. **Network Infrastructure** – Design a **secure, high-speed network** that seamlessly connects the two locations with **VPN and firewall protection**.
3. **Server Virtualization** – Implement **scalable virtual environments** with optimized resource allocation.
4. **Security & Compliance** – Enforce **enterprise-level security** using **firewalls, encryption, and multi-factor authentication**.
5. **Business Continuity** – Establish **redundant systems, failover mechanisms, and regular data backups**.

CONFIGURATION AND CUSTOMIZATION

Server Hardware Customization & Configuration:

- Deploy **high-performance servers** with **multi-core processors, large RAM capacity, SSD storage, and redundant power supplies**.
- Configure **virtualized environments** to optimize resource allocation and enhance **scalability**.

Network Architecture & IP Design:

- Assign **subnet ranges** and configure **routing protocols, firewalls, and VLANs**.
- Implement **Quality of Service (QoS)** techniques to prioritize business-critical applications.

Interoffice Connectivity:

- Establish **VPN tunnels** between **New York and Hong Kong** to ensure **secure and efficient** communication.
- Deploy **firewalls and intrusion detection systems (IDS)** to protect against cyber threats.

TESTING AND OPTIMIZATION

Before full deployment, a **comprehensive testing phase** will ensure **stability, performance, and security**:

1. **Hardware & Virtualization Testing:**

- Conduct **load testing, failover testing, and resource benchmarking** to validate performance.

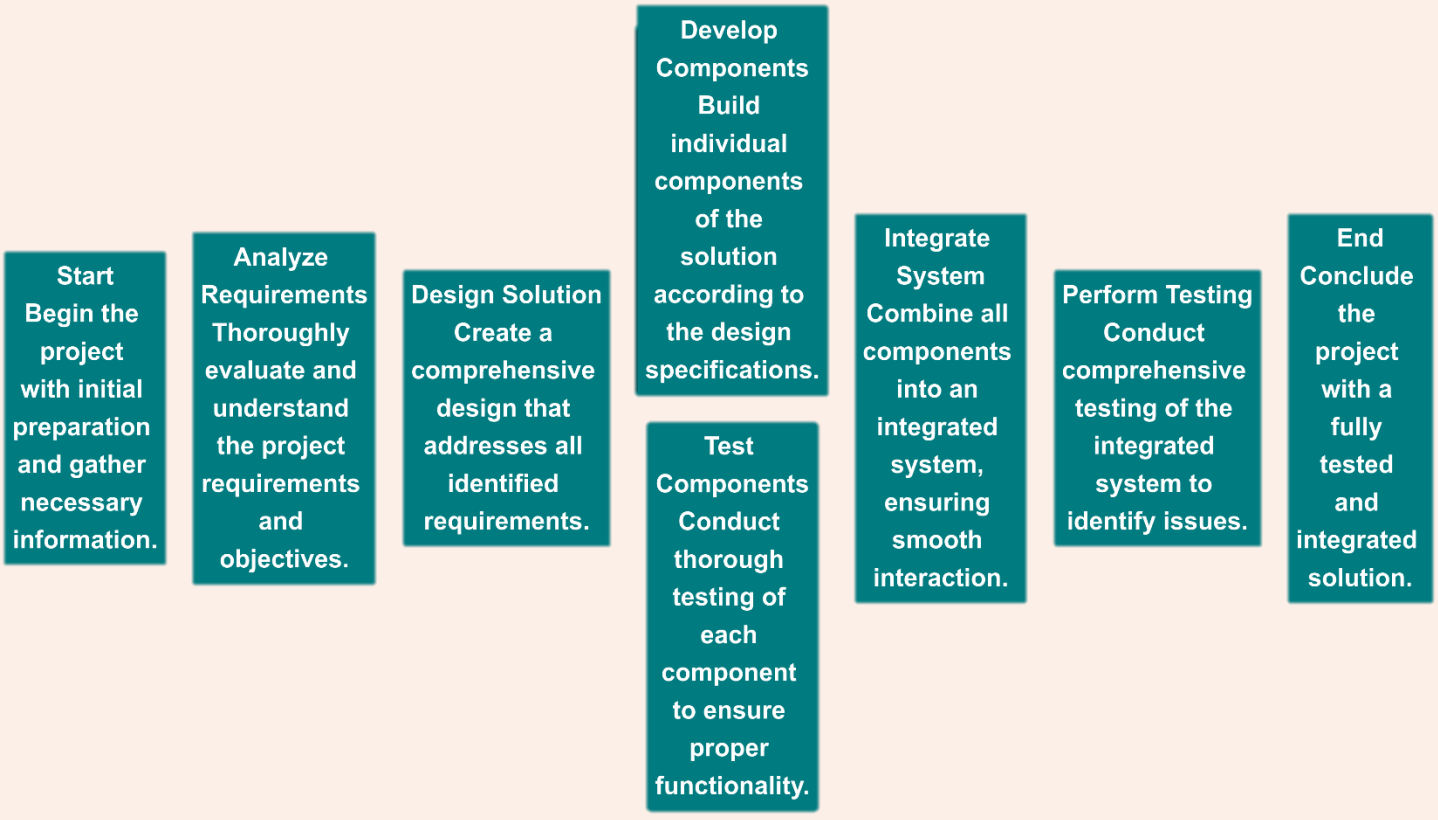
2. **Network Testing:**
   - Perform **latency, bandwidth, and packet loss assessments** to optimize **network traffic flow**.
3. **Security Testing:**
   - Implement **penetration testing and vulnerability assessments** to strengthen **cybersecurity defenses**.

MAINTENANCE AND UPGRADING

To maintain **long-term efficiency**, a structured **maintenance and upgrade plan** will be followed:

1. **Routine Maintenance:**
   - Conduct **regular system updates, patch management, and performance monitoring**.
2. **Security Enhancements:**
   - Continuously monitor **firewalls, antivirus, and intrusion detection systems (IDS)**.
3. **Scalability Planning:**
   - Ensure infrastructure is **future-proof** by implementing **scalable cloud and virtualization solutions**.

By implementing these strategies, **INDOCAN** will establish a **robust, scalable, and secure** IT environment to support seamless expansion and operational success.



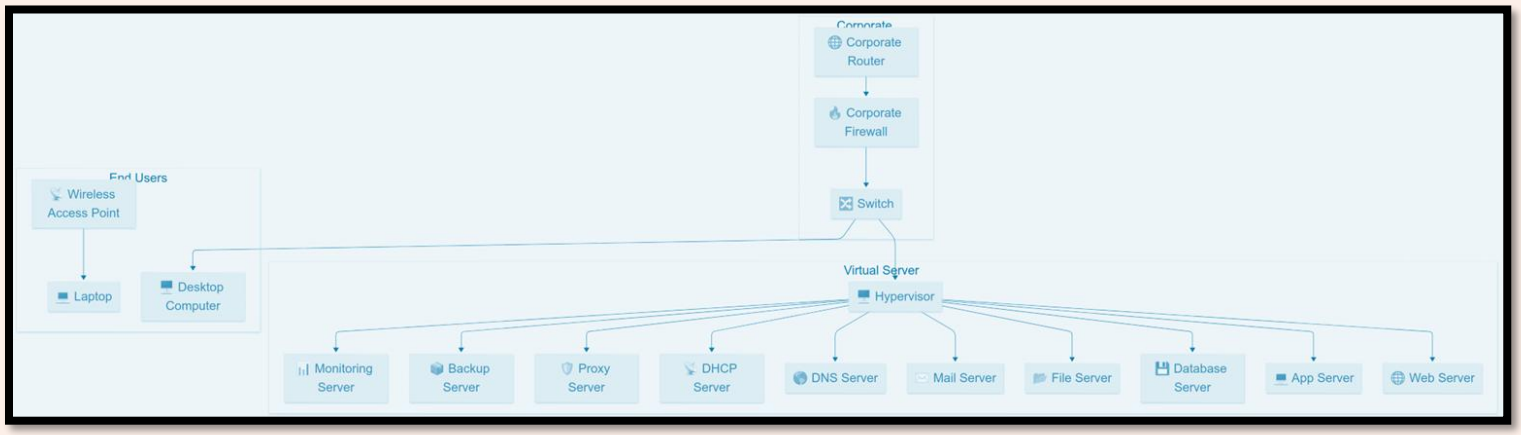
INFRASTRUCTURE

Expanding **INDOCAN** into **New York** and **Hong Kong** requires a robust IT infrastructure to ensure seamless operations and connectivity between both locations. The infrastructure will support network connectivity, server deployment, virtualization, security, and end-user computing needs.

NETWORK INFRASTRUCTURE

Network Design & Implementation

- **Modular and Scalable Architecture** – The network is designed to support future expansion while maintaining reliability.
- **High-Speed Connectivity** – Implementation of **redundant 10 Gbps connections** between offices to ensure fast and secure communication.
- **VLAN Segmentation** – Logical segmentation to separate departments, ensuring optimized traffic flow and security.
- **Load Balancing & Failover Mechanisms** – Redundant links and failover routing to prevent downtime.



HARDWARE & VIRTUALIZATION

Server Deployment

- **High-Performance Servers** – Multi-core processors, 128GB+ RAM, SSD storage.
- **Virtualization** – VMware ESXi 7.0 for resource-efficient infrastructure.
- **Hyper-V Clustering** – Ensuring high availability for critical workloads.

Storage Architecture

- **SAN (Storage Area Network)** – High-speed storage to support virtualization.
- **RAID Configuration** – RAID-10 for redundancy and performance.
- **Data Replication** – Between **New York** and **Hong Kong** for failover capability.

SECURITY INFRASTRUCTURE

Network Security

- **Firewall Protection** – Next-Gen Firewalls (Palo Alto, Fortinet) for deep packet inspection.
- **VPN Connectivity** – Secure tunnels for interoffice communication.
- **Intrusion Prevention System (IPS)** – Preventing unauthorized access and threats.

Access Controls & Compliance

- **Active Directory & Role-Based Access Control (RBAC)** – Ensuring only authorized personnel access resources.
- **Multi-Factor Authentication (MFA)** – Securing sensitive data.
- **Data Encryption** – AES-256 encryption for data at rest and in transit.

END-USER COMPUTING

User Device Allocation

Department	Device Type	Quantity
Developers	High-end Workstations	20
Admin Staff	Business Laptops	10
IT Team	Servers & Workstations	5

Peripheral & Software

- **Microsoft 365, Adobe Suite** – Productivity tools.
- **Cloud-Based Collaboration** – Microsoft Teams, Zoom.

DATA BACKUP & DISASTER RECOVERY

Backup Strategy

- **Automated Backups** – Daily snapshots stored in multiple locations.
- **Off-Site Storage** – Cloud backup with AWS S3.
- **Disaster Recovery Site** – Redundant failover in case of outages.

MONITORING & MAINTENANCE

System Monitoring

- **SNMP & Syslog** – Continuous network and system health monitoring.
- **Performance Dashboards** – Real-time tracking via PRTG & SolarWinds.

Regular Maintenance & Updates

- **Firmware Patching** – Ensuring security vulnerabilities are addressed.
- **Periodic Performance Audits** – Identifying and resolving bottlenecks.

IP Addressing Scheme

Virtual Machine	Assigned IP	Subnet Mask	Gateway	CPU	RAM (GB)	Storage (GB)	Operating System
Hong Kong DC	172.24.13.50	255.255.255.0	172.24.13.1	4	16	200	Windows Server 2022 (64-bit)
Citadel Server Hong Kong	172.24.13.51	255.255.255.0	172.24.13.1	4	16	500	Windows Server 2022 (64-bit)
Failover Cluster (HK)	172.24.13.52	255.255.255.0	172.24.13.1	8	32	500	VMware ESXi 7.0 U3
File Server (HK)	172.24.13.53	255.255.255.0	172.24.13.1	2	8	1000	Windows Server 2022 (64-bit)
Web Server (HK)	172.24.13.54	255.255.255.0	172.24.13.1	4	16	250	Ubuntu Server 22.04 LTS
Load Balancer (HK)	172.24.13.55	255.255.255.0	172.24.13.1	2	8	50	Ubuntu Server 22.04 LTS
Exchange Server1 (HK)	172.24.13.56	255.255.255.0	172.24.13.1	8	32	500	Windows Server 2022 (64-bit)
Exchange Server2 (HK)	172.24.13.57	255.255.255.0	172.24.13.1	8	32	500	Windows Server 2022 (64-bit)
(HK) Database Availability Group	172.24.13.58	255.255.255.0	172.24.13.1	8	64	2000	Windows Server 2022 (64-bit)
New York DC	172.24.13.59	255.255.255.0	172.24.13.1	4	16	200	Windows Server 2022 (64-bit)
Citadel Server New York	172.24.13.60	255.255.255.0	172.24.13.1	4	16	500	Windows Server 2022 (64-bit)
Failover Cluster (NY)	172.24.13.61	255.255.255.0	172.24.13.1	8	32	500	VMware ESXi 7.0 U3
File Server (NY)	172.24.13.62	255.255.255.0	172.24.13.1	2	8	1000	Windows Server 2022 (64-bit)
Web Server (NY)	172.24.13.63	255.255.255.0	172.24.13.1	4	16	250	Ubuntu Server 22.04 LTS
Load Balancer (NY)	172.24.13.64	255.255.255.0	172.24.13.1	2	8	50	Ubuntu Server 22.04 LTS
Exchange Server1 (NY)	172.24.13.65	255.255.255.0	172.24.13.1	8	32	500	Windows Server 2022 (64-bit)
Exchange Server2 (NY)	172.24.13.66	255.255.255.0	172.24.13.1	8	32	500	Windows Server 2022 (64-bit)
(NY) Database Availability Group	172.24.13.67	255.255.255.0	172.24.13.1	8	64	2000	Windows Server 2022 (64-bit)
vCenter Server	172.24.13.68	255.255.255.0	172.24.13.1	4	16	200	VMware vCenter Server Appliance
VMware ESXi Host1	172.24.13.69	255.255.255.0	172.24.13.1	16	128	2000	VMware ESXi 7.0 U3
VMware ESXi Host 2	172.24.13.70	255.255.255.0	172.24.13.1	16	128	2000	VMware ESXi 7.0 U3
GitLab CE Server	172.24.13.72	255.255.255.0	172.24.13.1	8	16	500	Ubuntu Server 22.04 LTS
pfSense Firewall	172.24.13.73	255.255.255.0	172.24.13.1	2	4	50	pfSense 2.6
Zammad Monitoring Server	172.24.13.74	255.255.255.0	172.24.13.1	4	16	200	Ubuntu Server 22.04 LTS
Mattermost Communication Server	172.24.13.75	255.255.255.0	172.24.13.1	4	16	200	Ubuntu Server 22.04 LTS

DOMAIN CONFIGURATION FOR INDOCAN

To support INDOCAN's expansion into New York and Hong Kong, a **centralized Active Directory (AD) structure** will be implemented. This will provide **authentication, resource management, and security** across both locations.

DOMAIN STRUCTURE & NAMING CONVENTION

- **Forest Name:** INDOCAN.LOCAL
- **Domains:**
  - **New York:** INDOCAN-NY.LOCAL
  - **Hong Kong:** INDOCAN-HK.LOCAL
- **Organizational Units (OUs):**
  - Users
  - Computers
  - Groups
  - Servers

Each branch office will have its **own domain controller (DC)** for local authentication while synchronizing with the **primary DC** at headquarters.

DOMAIN CONTROLLER DEPLOYMENT

Location	Server Name	Role	IP Address
New York	INDOCAN-DC1	Primary Domain Controller	172.24.13.59
Hong Kong	INDOCAN-DC2	Secondary Domain Controller	172.24.13.50

Configuration Steps:

1. Install **Windows Server 2022** on each DC.
2. Add **Active Directory Domain Services (ADDS)** role.
3. Promote servers to **domain controllers**.
4. Configure **DNS and Global Catalog** settings.
5. Enable **Directory Replication** between locations.

DOMAIN TRUST CONFIGURATION

To facilitate **seamless access between New York and Hong Kong**, a **two-way forest trust** will be established between INDOCAN-NY.LOCAL and INDOCAN-HK.LOCAL.

Steps to Create the Trust:

1. Open **Active Directory Domains and Trusts**.
2. Right-click INDOCAN-NY.LOCAL and select **Properties**.
3. Navigate to the **Trusts** tab and select **New Trust**.

4. Enter **INDOCAN-HK.LOCAL** as the trust name.
5. Select **Forest Trust → Two-Way Trust**.
6. Authenticate with **administrator credentials** for both domains.
7. Confirm the trust relationship.

GROUP POLICY & ACCESS CONTROL

Group Policies (GPOs) will be enforced to manage user permissions, security settings, and resource access:

- **User Restrictions:** Limit non-admin access to critical systems.
- **Password Policies:** Enforce complex passwords and expiration policies.
- **Mapped Drives & Shared Folders:** Assign network drives based on user roles.
- **Software Deployment:** Centralized installation of required applications.

DNS & CONDITIONAL FORWARDERS

To resolve domain names across locations, **Conditional Forwarders** will be set up for DNS resolution between INDOCAN-NY.LOCAL and INDOCAN-HK.LOCAL.

Steps to Configure:

1. Open **DNS Manager** on INDOCAN-DC1.
2. Navigate to **Conditional Forwarders → New Conditional Forwarder**.
3. Enter **INDOCAN-HK.LOCAL** as the domain name.
4. Add the **IP address** of the Hong Kong DC (172.24.13.50).
5. Repeat the process on INDOCAN-DC2 for INDOCAN-NY.LOCAL.

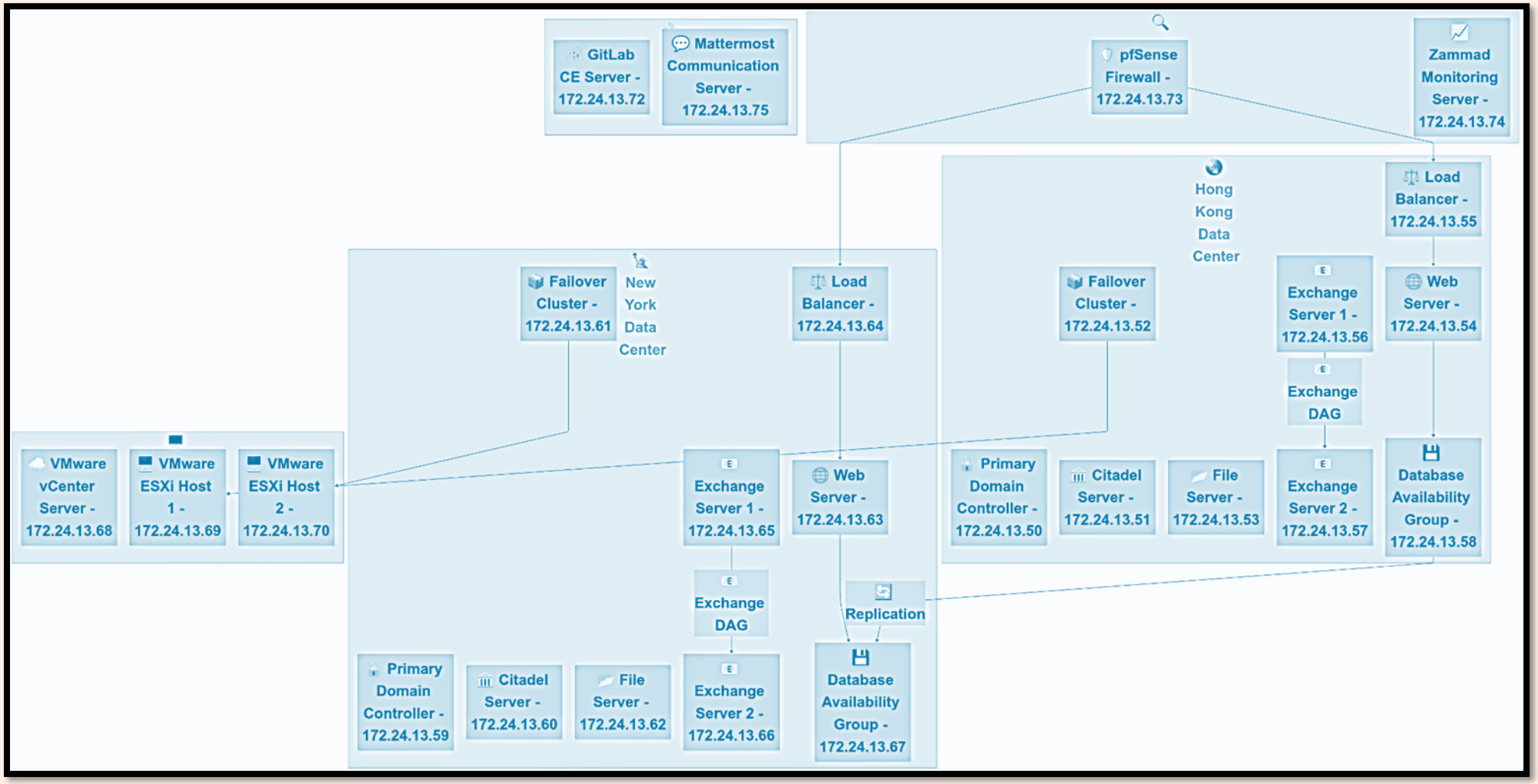
ACTIVE DIRECTORY REPLICATION

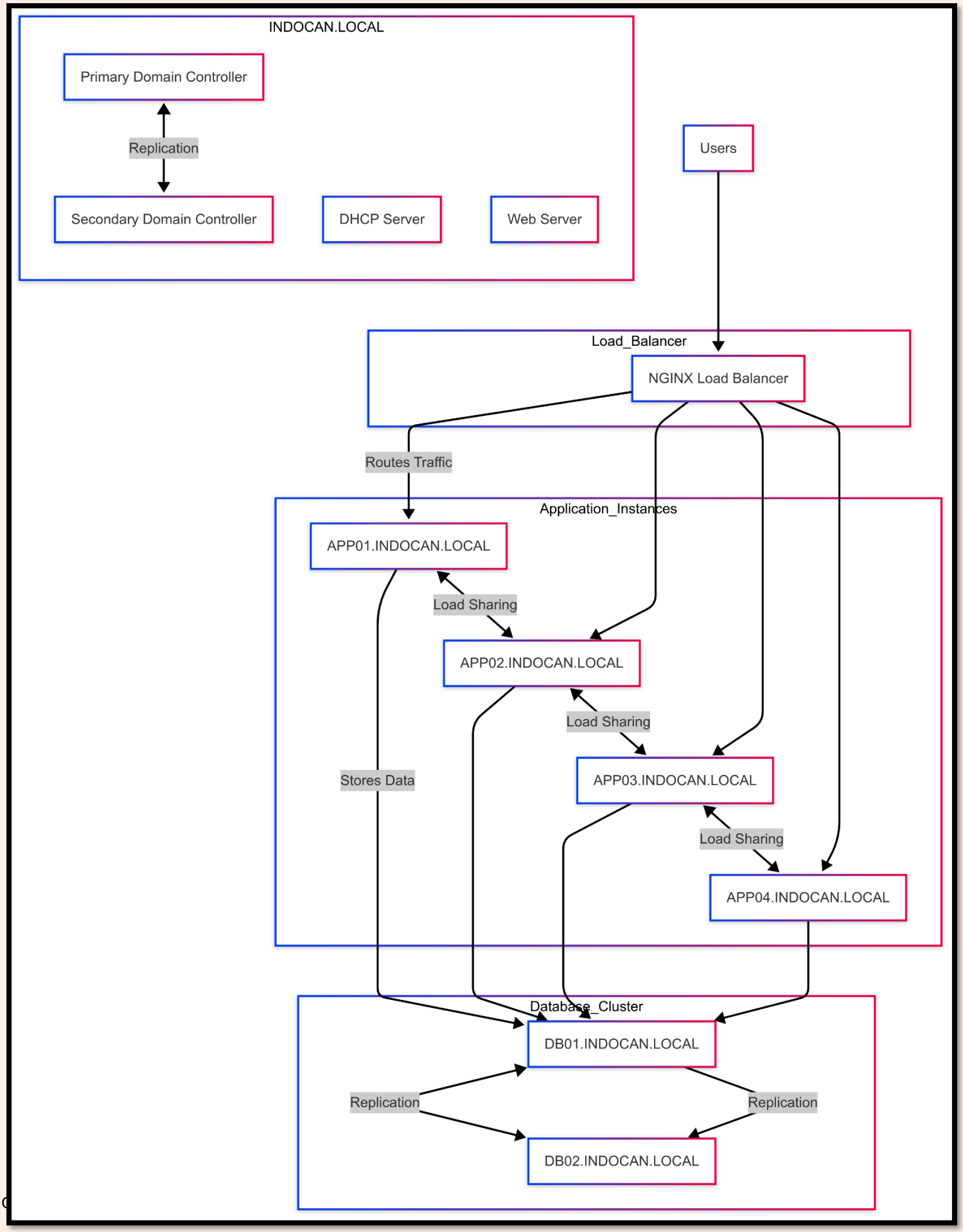
A **site-to-site VPN** will be used to facilitate **Active Directory replication** between New York and Hong Kong. This ensures that **user credentials, security policies, and directory changes** are synchronized across locations.

- **Replication Interval:** Every 15 minutes.
- **Replication Topology:** Hub-and-Spoke (Primary DC in New York as Hub, Secondary DC in Hong Kong as Spoke).
- **Replication Type:** Multi-Master for fault tolerance.

By implementing this domain architecture, INDOCAN will ensure a **secure, scalable, and efficient** identity management system across its global offices. This setup provides **centralized control, improved security, and seamless resource access** between locations.

ARCHITECTURE




INDOCAN.LOCAL
Primary Domain Controller
Replication
Secondary Domain Controller
DHCP Server
Web Server
Users
Load_Balancer
NGINX Load Balancer
Routes Traffic
Application_Instances
APP01.INDOCAN.LOCAL
Load Sharing
APP02.INDOCAN.LOCAL
Load Sharing
Stores Data
APP03.INDOCAN.LOCAL
Load Sharing
APP04.INDOCAN.LOCAL
Database_Cluster
DB01.INDOCAN.LOCAL
Replication
Replication
DB02.INDOCAN.LOCAL

DOMAIN TRUST

To ensure seamless authentication and resource sharing between **INDOCAN's** offices in **New York** and **Hong Kong**, a **two-way transitive trust** will be established. This setup will allow users from both domains to access shared resources efficiently while maintaining **security and centralized management**.

TRUST TYPE & CONFIGURATION

The following **trust types** will be implemented:

- **Two-Way Trust** – Enables users in both locations to authenticate and access permitted resources in each domain.
- **Transitive Trust** – Ensures that any additional child domains created in the future automatically trust each other.
- **Forest-Wide Trust** – Allows **INDOCAN.LOCAL** to manage all authentication requests efficiently.

DOMAIN TRUST IMPLEMENTATION

Domain Structure:

Domain Name	Location	Trust Type	Trust Direction
INDOCAN-NY.LOCAL	New York	Two-Way Trust	Bidirectional
INDOCAN-HK.LOCAL	Hong Kong	Two-Way Trust	Bidirectional

Configuration Steps:

1. Open **Active Directory Domains and Trusts** on **INDOCAN-DC1 (New York)**.
2. Right-click **INDOCAN-NY.LOCAL** → Select **Properties**.
3. Navigate to the **Trusts** tab and click **New Trust**.
4. Enter **INDOCAN-HK.LOCAL** as the trust name.
5. Select **Forest Trust** → **Two-Way Trust**.
6. Authenticate using **Domain Admin Credentials** from both locations.
7. Confirm **Trust Verification** and test connectivity.

DNS CONFIGURATION FOR TRUST RESOLUTION

To ensure both domains resolve each other's names, **Conditional Forwarders** will be created:

1. Open **DNS Manager** on **INDOCAN-DC1**.
2. Navigate to **Conditional Forwarders** → Click **New Conditional Forwarder**.
3. Enter **INDOCAN-HK.LOCAL** and assign its **Domain Controller's IP (172.24.13.51)**.
4. Repeat the process on **INDOCAN-DC2** for **INDOCAN-NY.LOCAL**.

SECURITY & ACCESS CONTROL

- **Role-Based Access Control (RBAC)** will be enforced, ensuring only authorized users can access cross-domain resources.
- **Group Policy Objects (GPOs)** will be implemented to restrict access to sensitive resources.
- **Multi-Factor Authentication (MFA)** will be enabled for cross-domain logins.

TRUST VALIDATION & TESTING

Once the trust is established, the following verification steps will be performed:

1. **Use `nltest /domain_trusts`** to verify trust relationships.
2. **Perform cross-domain authentication tests** using test accounts.
3. **Check event logs** on both **Domain Controllers** for trust errors.

FILE SERVER

A **high-availability file server** is crucial for INDOCAN's operations, ensuring seamless access, redundancy, and security across New York and Hong Kong offices. This section outlines the **installation, failover cluster setup, role configuration, and file sharing** while integrating **security best practices**.

INSTALLATION

Prerequisites:

- **Windows Server 2022** installed on designated file servers.
- **Active Directory Domain Services (ADDS)** configured.
- **Failover Clustering feature enabled.**
- **NTFS-formatted storage volumes available for data storage.**

Installation Steps:

1. Open **Server Manager** → Click **Manage** → Select **Add Roles and Features**.
2. Choose **Role-based or Feature-based Installation**.
3. Select the target server (INDOCAN-NY-FS01 / INDOCAN-HK-FS01).
4. Under **Server Roles**, check **File and Storage Services** → Expand and select **File Server**.
5. Click **Next** → Confirm selections → Click **Install**.
6. Restart the server after installation.

FAILOVER CLUSTER SETUP

To ensure **high availability**, a **File Server Failover Cluster** will be deployed.

Cluster Nodes:

Node Name	IP Address	Role
INDOCAN-NY-FS01	172.24.13.62	Primary File Server (New York)
INDOCAN-NY-FS02	172.24.13.162	Secondary File Server (New York)
INDOCAN-HK-FS01	172.24.13.53	Primary File Server (Hong Kong)
INDOCAN-HK-FS02	172.24.13.153	Secondary File Server (Hong Kong)

Failover Cluster Configuration:

1. Open **Failover Cluster Manager** → Click **Create Cluster**.
2. Add the servers **(INDOCAN-NY-FS01, INDOCAN-NY-FS02, INDOCAN-HK-FS01, INDOCAN-HK-FS02)**.
3. Run **Validation Tests** to ensure compatibility.
4. Assign a **Cluster Name** (**INDOCAN-FS-CLUSTER**).
5. Assign a **Cluster IP** for network communication.
6. Configure **Cluster Shared Volumes (CSV)** for shared file storage.
7. Confirm settings → Click **Create Cluster**.

FILE SERVER ROLE CONFIGURATION

After cluster setup, the **File Server Role** needs to be configured for **High Availability (HA)**.

1. Open **Failover Cluster Manager** → Expand **Roles**.
2. Click **Configure Role** → Select **File Server** → Click **Next**.
3. Choose **File Server for General Use** (for basic file sharing) or **Scale-Out File Server** (for enterprise workloads).
4. Assign a **Client Access Name** (e.g., \INDOCAN-FILES) and **Cluster IP Address**.
5. Select the **Cluster Shared Volume (CSV)** as the storage location.
6. Click **Next** → Confirm settings → Click **Finish**.
7. Verify that the **File Server Role** is running in **Failover Cluster Manager**.

FILE SHARE CREATION

File shares will be created to provide structured **department-based storage**.

Departmental Shares:

Share Name	Path	Access Control
HR Files	\INDOCAN-FILES\HR	HR Group Only
Finance Files	\INDOCAN-FILES\Finance	Finance Group Only
Development Files	\INDOCAN-FILES\Development	Developers Only
Public Share	\INDOCAN-FILES\Public	All Employees

Steps to Create a File Share:

1. Open **Server Manager** → Navigate to **File and Storage Services**.
2. Click **Shares** → Select **New Share**.
3. Choose **SMB Share – Quick** (for standard access) or **SMB Share – Applications** (for high-performance workloads).
4. Select **Cluster Shared Volume (CSV)** as the storage location.
5. Define **Share Name** (e.g., HR, Finance, Development, Public).
6. Configure **NTFS Permissions**:
    - **HR Group** → Full Control on HR Share.
    - **Finance Group** → Full Control on Finance Share.
    - **Developers** → Read/Write Access on Development Share.
    - **All Users** → Read/Write Access on Public Share.
7. Click **Create** → Verify accessibility via \INDOCAN-FILES\.

FILE SHARE ACCESS BY USERS

To ensure **seamless access**, Group Policy (GPO) will be used to **map network drives** for users.

Steps to Map File Shares via GPO:

1. Open **Group Policy Management** → Create a **New GPO**.
2. Navigate to **User Configuration** → **Preferences** → **Windows Settings** → **Drive Maps**.
3. Right-click → **New Mapped Drive**.
4. Set **Location** as \INDOCAN-FILES\HR (for HR Group, repeat for other shares).
5. Assign a **Drive Letter** (e.g., H: for HR, F: for Finance).
6. Under **Common Tab**, check **Reconnect at logon**.
7. Apply **Security Filtering** to specific **AD groups** (HR, Finance, Developers, etc.).
8. Click **OK** → Link GPO to relevant **Organizational Units (OUs)**.
9. Run **gpupdate /force** on client machines to apply changes.

Validation & Testing:

- Users should be able to access their respective **departmental shares**.
- **Unauthorized access should be denied** due to NTFS & Share Permissions.
- Verify logs in **Event Viewer** for failed access attempts.

BACKUP & DISASTER RECOVERY

To **protect against data loss**, an **automated backup and disaster recovery plan** is implemented.

Backup Strategy:

- **Daily Snapshots** → Stored on **local and cloud storage**.
- **Weekly Full Backups** → Replicated between **New York & Hong Kong**.
- **Azure Backup Vault** → Secure off-site data storage.

Disaster Recovery Plan:

- **Automatic Failover** in case of **File Server failure**.
- **Replication between locations** to restore operations quickly.
- **Quarterly DR Testing** to ensure **business continuity**.

WEB SERVER

The Web Server infrastructure for **INDOCAN** is designed to provide **high availability, scalability, and secure access** for internal and external users across **New York and Hong Kong** offices. This section covers the **installation, load balancing, role configuration, and user access**, along with **security enhancements** and **backup strategies** to ensure continuous availability.

INSTALLATION

Prerequisites:

- **Ubuntu Server 22.04 LTS** installed on designated Web Servers.
- **Static IP Assigned** (New York: 172.24.13.63, Hong Kong: 172.24.13.54).
- **Domain Controller (DC) Integration** for centralized authentication.
- **Firewall rules configured** to allow HTTP (80) and HTTPS (443) traffic.

Installation Steps:

1. Update the system:

```
sudo apt update && sudo apt upgrade -y
```

2. Install the **Apache (or Nginx) Web Server**:

```
sudo apt install apache2 -y  # For Apache
# OR
sudo apt install nginx -y    # For Nginx
```

3. Enable and start the web server:

```
sudo systemctl enable apache2 && sudo systemctl start apache2  # Apache
sudo systemctl enable nginx && sudo systemctl start nginx  # Nginx
```

4. Allow HTTP & HTTPS traffic through the firewall:

```
sudo ufw allow 'Apache Full'  # If using Apache
sudo ufw allow 'Nginx Full'   # If using Nginx
```

5. Verify installation by accessing http://server-ip in a browser.

NETWORK LOAD BALANCER

To **distribute traffic** between the **New York and Hong Kong Web Servers**, an **Nginx Load Balancer** will be configured.

Load Balancer Installation:

1. Install Nginx on the Load Balancer servers (172.24.13.55 in HK & 172.24.13.64 in NY):

```
sudo apt install nginx -y
```

2. Edit the Nginx configuration file:

```
sudo nano /etc/nginx/nginx.conf
```

3. Add the following **load balancing configuration**:

```
upstream indocan_web {
    server 172.24.13.54;  # Web Server HK
    server 172.24.13.63;  # Web Server NY
}

server {
    listen 80;
    server_name INDOCAN.LOCAL;

    location / {
  proxy_pass http://indocan_web;
        proxy_set_header Host $host;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    }
}
```

4. Restart Nginx to apply changes:

```
sudo systemctl restart nginx
```

5. Test by accessing **http://INDOCAN.LOCAL** to verify load balancing.

WEB SERVER ROLE CONFIGURATION

After installation, the **Web Server Role** will be configured for **hosting applications**.

Steps for Configuration:

1. Set up a **default document root**:

```
sudo mkdir -p /var/www/INDOCAN.LOCAL/html
sudo chown -R www-data:www-data /var/www/INDOCAN.LOCAL/html
```

2. Create a sample index.html file:

```
echo '<h1>Welcome to INDOCAN Web Server</h1>' | sudo tee
/var/www/INDOCAN.LOCAL/html/index.html
```

3. Create a new virtual host configuration:

```
sudo nano /etc/apache2/sites-available/indocan.conf
```

   Add the following configuration:

```
<VirtualHost *:80>
    ServerAdmin admin@INDOCAN.LOCAL
    ServerName INDOCAN.LOCAL
    DocumentRoot /var/www/INDOCAN.LOCAL/html

    <Directory /var/www/INDOCAN.LOCAL/html>
        Options -Indexes +FollowSymLinks
        AllowOverride All
        Require all granted
    </Directory>
</VirtualHost>
```

4. Enable the new site and restart Apache:

```
sudo a2ensite indocan.conf
sudo systemctl restart apache2
```

ACCESS BY USERS

Users will access the web server via **internal (corporate network)** and **external (public domain)** access.

Internal Access (Within INDOCAN Network)

- Users can access the web portal using **http://INDOCAN.LOCAL**.
- Group Policy will be used to set this as the **default homepage** for company devices.

External Access (Public Users)

- **Public DNS records** will map INDOCAN.LOCAL to the Load Balancer's IP.
- An **SSL certificate (Let's Encrypt or DigiCert)** will be installed for secure HTTPS access.
- A **Web Application Firewall (WAF)** will protect against external threats.

SECURITY HARDENING

To **enhance security**, the following **best practices** will be implemented:

- **Disable directory listing:**

  ```
  sudo nano /etc/apache2/apache2.conf
  ```

  Change Options Indexes FollowSymLinks to Options -Indexes +FollowSymLinks.

- **Enable mod_security for intrusion detection:**

  ```
  sudo apt install libapache2-mod-security2 -y
  sudo systemctl restart apache2
  ```

- **Rate limit requests to prevent DDoS attacks:**

  ```
  limit_req_zone $binary_remote_addr zone=one:10m rate=30r/m;
  ```

BACKUP & DISASTER RECOVERY

To **ensure uptime**, automated **backups and disaster recovery** measures will be implemented:

Backup Strategy:

- **Daily backups** of web server files (/var/www/) to an **off-site storage**.
- **Weekly snapshots** of the entire web server.
- **Failover setup** with automatic traffic rerouting to the secondary site.

Disaster Recovery Plan:

- If the **primary web server** fails, the **Load Balancer automatically reroutes traffic**.
- The backup web server in **New York/Hong Kong** will take over within **5 minutes**.
- **Quarterly DR tests** to validate failover mechanisms.

EXCHANGE SERVER

INDOCAN requires a **high-availability Exchange Server environment** to ensure **secure, reliable, and efficient email communication** between its offices in **New York and Hong Kong**. This section covers the **installation, prerequisite setup, Database Availability Group (DAG) configuration, and email exchange setup**, along with additional enhancements for **security, backup, and disaster recovery**.

INSTALLATION

Prerequisites:

- **Windows Server 2022** installed on designated Exchange Servers.
- **Active Directory (AD) Integrated** with domain trust established between New York and Hong Kong.
- **Static IP Assigned** (New York: 172.24.13.65, Hong Kong: 172.24.13.56).
- **Microsoft Exchange Server 2019 ISO** available for installation.
- **Firewall rules configured** to allow SMTP (25), IMAP (143), POP3 (110), and HTTPS (443) traffic.

Installation Steps:

1. Mount the Exchange Server ISO and run the setup file.
2. Choose **Mailbox Role** and **Edge Transport Role** (for security and mail filtering).
3. Accept the license agreement and proceed.
4. Set the **organization name as INDOCAN**.
5. Assign a static **send and receive connector**.
6. Configure **mail routing between New York and Hong Kong**.
7. Install the required roles and restart the server.
8. Verify installation by logging into the **Exchange Admin Center (EAC)**.

INSTALLING PREREQUISITES

Before configuring Exchange Server, **Windows Server roles and features** must be installed.

Steps:

1. Open **PowerShell as Administrator** and install required features:

```
Install-WindowsFeature RSAT-ADDS, Web-Server, NET-Framework-Features, RSAT-Clustering -IncludeManagementTools
```

2. Install the **Exchange prerequisites package**:

```
Install-WindowsFeature Server-Media-Foundation
```

3. Set up **Unified Communications Managed API 4.0**:

```
Install-Module -Name ExchangeOnlineManagement -Force
```

4. Reboot the server before proceeding to **DAG configuration**.

DATABASE AVAILABILITY GROUP (DAG)

To ensure **email redundancy and failover protection**, a **DAG** will be created.

DAG Configuration Steps:

1. Open **Exchange Admin Center (EAC)** → Navigate to **Servers** → **Database Availability Groups**.
2. Click **New DAG**, enter the DAG name (e.g., INDOCAN-DAG01).
3. Assign a **witness server** (a separate server to maintain quorum).
4. Add both Exchange Servers (INDOCAN-EX01 & INDOCAN-EX02).
5. Configure **network replication settings**.
6. Verify DAG health using PowerShell:

```
Get-DatabaseAvailabilityGroup -Identity INDOCAN-DAG01 | Format-List
```

7. Enable **automatic failover** between the two Exchange Servers.

EMAIL EXCHANGE CONFIGURATION

Mail Flow Setup:

1. Open **EAC → Mail Flow → Send Connectors**.
2. Create a **new send connector** named INDOCAN-Mailflow.
3. Set **SMTP routing between Hong Kong and New York**.
4. Enable **TLS encryption** for secure mail exchange.
5. Add **SPF, DKIM, and DMARC** records to protect against phishing attacks.

Access by Users:

- **Internal Employees**: Access emails via **Outlook or Outlook Web Access (OWA)**.
- **External Clients**: Access emails using **IMAP, POP3, or webmail**.
- **Mobile Access**: Configured via **Exchange ActiveSync (EAS)** for iOS and Android.

SECURITY & COMPLIANCE ENHANCEMENTS

To **secure email communications**, the following security measures are implemented:

- **Enforce Multi-Factor Authentication (MFA)** for user logins.
- **Enable Anti-Spam & Anti-Malware policies** in Exchange Admin Center.
- **Apply Transport Rules** to prevent sensitive data from leaving the organization.
- **Restrict External Email Forwarding** to prevent data leaks.

BACKUP & DISASTER RECOVERY

To **ensure uptime and email data integrity**, automated **backup and disaster recovery measures** will be implemented.

Backup Strategy:

- **Daily email database backups** to off-site storage.
- **Incremental backups** every 6 hours.
- **Exchange-aware snapshots** stored in the cloud.

Disaster Recovery Plan:

- If the **primary Exchange Server** fails, the **DAG automatically reroutes email traffic**.
- The backup Exchange Server in **New York/Hong Kong** will take over within **2 minutes**.
- **Quarterly DR tests** to validate failover mechanisms.

ZAMMAD TICKETING & MONITORING SYSTEM

INDOCAN requires a **centralized ticketing and monitoring system** to streamline **IT support, incident management, and real-time monitoring** of infrastructure components. **Zammad**, an open-source helpdesk solution, will be deployed for handling user requests, issue tracking, and automated reporting. This section covers **installation, configuration, integration, security, and backup strategies** to ensure **optimal IT service management (ITSM).**

INSTALLATION

Prerequisites:

- **Ubuntu Server 22.04 LTS** installed on 172.24.13.74.
- **8 vCPUs, 16GB RAM, 200GB SSD** allocated.
- **Database Server (PostgreSQL)** installed for Zammad data storage.
- **Elasticsearch** installed for advanced search capabilities.
- **Firewall rules** configured to allow ports **80 (HTTP), 443 (HTTPS), 5432 (PostgreSQL), and 9200 (Elasticsearch).**

Installation Steps:

1. **Update & Install Dependencies**:

```
sudo apt update && sudo apt upgrade -y
sudo apt install -y wget curl gnupg2 software-properties-common
```

2. **Add Zammad Repository & Install**:

```
wget -qO- https://dl.packager.io/srv/zammad/zammad/key | sudo apt-key add -
echo "deb https://dl.packager.io/srv/deb/zammad/zammad/stable/ubuntu $(lsb_release -cs) main" | sudo tee /etc/apt/sources.list.d/zammad.list
sudo apt update && sudo apt install -y zammad
```

3. **Start and Enable Zammad**:

```
sudo systemctl enable zammad
sudo systemctl start zammad
```

4. **Verify Installation** by accessing http://server-ip in a browser.

DATABASE CONFIGURATION

Zammad requires **PostgreSQL** for structured ticket storage.

Steps:

1. **Install PostgreSQL:**

```
sudo apt install postgresql postgresql-contrib -y
```

2. **Create Database & User:**

```
sudo -u postgres psql
CREATE DATABASE zammad;
CREATE USER zammad_user WITH PASSWORD 'securepassword';
ALTER DATABASE zammad OWNER TO zammad_user;
GRANT ALL PRIVILEGES ON DATABASE zammad TO zammad_user;
```

3. **Configure Database in Zammad**:

```
sudo zammad config:set DATABASE_ADAPTER=postgresql
```

4. **Restart Services:**

```
sudo systemctl restart zammad postgresql
```

WEB SERVER CONFIGURATION

To enable **secure remote access**, an **NGINX reverse proxy** will be configured.

Steps:

1. **Install NGINX & SSL Support:**

```
sudo apt install nginx -y
sudo apt install certbot python3-certbot-nginx -y
```

2. **Create Zammad NGINX Config:**

```
sudo nano /etc/nginx/sites-available/zammad
```

3. **Add Configuration:**

```
server {
   listen 80;
    server_name zammad.INDOCAN.LOCAL;
    location / {
        proxy_pass http://127.0.0.1:3000;
        proxy_set_header Host $host;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    }
}
```

4. **Enable Configuration & Restart NGINX:**

```
sudo ln -s /etc/nginx/sites-available/zammad /etc/nginx/sites-enabled/
sudo systemctl restart nginx
```

5. **Secure with SSL Certificate:**

```
sudo certbot --nginx -d zammad.INDOCAN.LOCAL
```

USER MANAGEMENT & ROLES

Zammad allows user authentication via **Active Directory (LDAP), SAML, or local user accounts.**

User Roles:

Role	Permissions
Admin	Full access, user management, settings
IT Support	Manage & resolve tickets, track performance
User	Submit and track tickets

LDAP Integration (Active Directory Authentication):

1. **Navigate to** Admin Panel → Integrations → LDAP.
2. **Enter AD Server Details** (e.g., ldap://ad.indocan.local).
3. **Map User Groups** to Roles (e.g., IT-Support → IT Support Role).
4. **Enable Auto-Sync** every **15 minutes**.

TICKETING WORKFLOW CONFIGURATION

To ensure efficient IT support, custom workflows are implemented:

1. **Ticket Categorization** → Assigns priorities based on severity.
2. **SLA Policies** → Defines response times (e.g., Critical Issues: **1-hour resolution**).
3. **Automated Escalations** → Escalates unresolved issues to senior engineers.
4. **Canned Responses** → Predefined responses for common issues.

ALERTS & MONITORING INTEGRATION

To improve IT operations, **Zammad will be integrated with Zabbix** for proactive issue resolution.

1. **Install Zabbix Agent on Zammad Server:**

```
sudo apt install zabbix-agent -y
```

2. **Configure Agent to send alerts:**

```
sudo nano /etc/zabbix/zabbix_agentd.conf
```

```
Server=zabbix.indocan.local
Hostname=zammad.INDOCAN.LOCAL
```

3. **Restart Zabbix Agent:**

```
sudo systemctl restart zabbix-agent
```

4. **Set Alert Triggers in Zabbix Dashboard.**

BACKUP & DISASTER RECOVERY

To ensure uptime and data security, **automated backups and DR strategies** are implemented.

Backup Strategy:

- **Daily database backups** stored in **remote NAS.**
- **Weekly snapshots of Zammad application files.**
- **Encryption enabled** for all backup files.

Disaster Recovery Plan:

- **Failover setup** with standby Zammad instance in **New York & Hong Kong**.
- **Automated restoration scripts** to bring up Zammad in under **10 minutes**.
- **Quarterly DR testing** to verify backup integrity.

GITLAB REPOSITORY BROWSER

INDOCAN requires a self-hosted GitLab CE (Community Edition) server to facilitate source code management, CI/CD pipelines, and collaborative software development. This section covers installation, configuration, integration, security measures, and backup strategies to ensure a secure, efficient, and scalable DevOps environment.

INSTALLATION

Prerequisites:

- **Ubuntu Server 22.04 LTS** installed on 172.24.13.72.
- **8 vCPUs, 16GB RAM, 500GB SSD** allocated.
- **PostgreSQL Database Server** for GitLab data storage.
- **Firewall rules** configured to allow ports **80 (HTTP), 443 (HTTPS), 22 (SSH for Git operations)**.

Installation Steps:

1. **Update System & Install Dependencies**:

```
sudo apt update && sudo apt upgrade -y
sudo apt install -y curl openssh-server ca-certificates tzdata perl
```

2. **Add GitLab Repository & Install GitLab CE**:

```
curl -fsSL https://packages.gitlab.com/install/repositories/gitlab/gitlab-ce/script.deb.sh | sudo bash
sudo apt install -y gitlab-ce
```

3. **Configure GitLab & Set External URL**:

```
sudo nano /etc/gitlab/gitlab.rb

Set:

external_url 'https://gitlab.INDOCAN.LOCAL'
```

4. **Reconfigure & Start GitLab**:

```
sudo gitlab-ctl reconfigure
sudo gitlab-ctl restart
```

5. **Access GitLab Web Interface** via https://gitlab.INDOCAN.LOCAL.

DATABASE CONFIGURATION

GitLab requires **PostgreSQL** for structured data storage.

Steps:

1. **Switch to PostgreSQL User & Access Database:**

```
sudo -i -u gitlab-psql
psql -d gitlabhq_production
```

2. **Create a Secure Database User:**

```
CREATE USER gitlab_user WITH PASSWORD 'securepassword';
ALTER DATABASE gitlabhq_production OWNER TO gitlab_user;
GRANT ALL PRIVILEGES ON DATABASE gitlabhq_production TO gitlab_user;
```

3. **Restart GitLab Services:**

```
sudo gitlab-ctl restart
```

USER & REPOSITORY MANAGEMENT

User Roles:

Role	Permissions
Admin	Full access, manage users and repositories
Developer	Push & merge code, manage CI/CD pipelines
Reporter	Read-only access, issue tracking

Steps to Create Users & Repositories:

1. **Navigate to GitLab Admin Panel** → https://gitlab.INDOCAN.LOCAL.
2. **Go to Users** → Add New User (Assign roles as needed).
3. **Create New Repository** via New Project.
4. **Set Up SSH Keys** for Secure Git Operations:

```
ssh-keygen -t rsa -b 4096 -C "user@INDOCAN.LOCAL"
cat ~/.ssh/id_rsa.pub
```

Add the SSH key in **GitLab User Settings**.

CI/CD PIPELINE CONFIGURATION

GitLab CI/CD automates **build, test, and deployment** processes.

Steps to Set Up CI/CD Pipeline:

1. **Create .gitlab-ci.yml in the repository root:**

```
stages:
  - build
  - test
  - deploy
build-job:
  stage: build
  script:
    - echo "Building Project..."
test-job:
  stage: test
  script:
    - echo "Running Tests..."
deploy-job:
  stage: deploy
  script:
    - echo "Deploying to Production..."
```

2. **Push the changes & monitor pipeline execution in GitLab.**

SECURITY HARDENING

To **enhance security**, the following best practices are implemented:

- **Enable Two-Factor Authentication (2FA)** for all users.
- **Restrict Public Repository Access**:

```
sudo gitlab-rails runner "ApplicationSetting.first.update(default_project_visibility: 'private')"
```

- **Automate Security Scanning**:

```
security:
  script:
    - echo "Running security checks..."
```

- **Enable Firewall Rules** to limit access only to INDOCAN IP ranges.

BACKUP & DISASTER RECOVERY

To **ensure uptime and data protection**, **automated backup and disaster recovery plans** are implemented.

Backup Strategy:

- **Daily repository backups** stored in **secure off-site storage.**
- **Weekly GitLab configuration snapshots.**
- **Automated encryption** of all backups.

Disaster Recovery Plan:

- **Standby GitLab Instance** set up in **New York & Hong Kong.**
- **Automated failover to secondary GitLab server** within **5 minutes**.
- **Quarterly DR testing** to validate recovery processes.

LOAD BALANCER

To ensure **high availability, fault tolerance, and optimized traffic distribution**, INDOCAN is implementing a **load balancing solution** for its **web servers, email servers, and internal applications** across its **New York and Hong Kong** data centers. This section covers **installation, configuration, traffic management, security, and monitoring** for the Load Balancer.

INSTALLATION

Prerequisites:

- **Ubuntu Server 22.04 LTS** installed on 172.24.13.55 (Hong Kong) and 172.24.13.64 (New York).
- **2 vCPUs, 8GB RAM, 50GB SSD** allocated.
- **Firewall rules** configured to allow traffic on **HTTP (80), HTTPS (443), and SMTP (25) for mail balancing.**
- **Nginx or HAProxy installed for Layer 4/7 load balancing.**

Installation Steps:

1. **Update System & Install Nginx**:

```
sudo apt update && sudo apt upgrade -y
sudo apt install -y nginx
```

2. **Enable and Start Nginx**:

```
sudo systemctl enable nginx
sudo systemctl start nginx
```

3. **Verify installation by accessing** http://server-ip in a browser.

LOAD BALANCER CONFIGURATION

Configuring Nginx for Load Balancing

For Web Server Load Balancing (Round-Robin Method)

1. **Open Nginx configuration file:**

```
sudo nano /etc/nginx/nginx.conf
```

2. **Add the following configuration:**

```
upstream indocan_web {
    server 172.24.13.54;  # Web Server HK
    server 172.24.13.63;  # Web Server NY
}

server {
    listen 80;
    server_name INDOCAN.LOCAL;

    location / {
        proxy_pass http://indocan_web;
        proxy_set_header Host $host;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    }
}
```

3. **Restart Nginx to apply changes:**

```
sudo systemctl restart nginx
```

4. **Test by accessing**

```
http://INDOCAN.LOCAL, and verify traffic is distributed across the two web servers.
```

For Mail Server Load Balancing (HAProxy - Layer 4 Load Balancing)

1. **Install HAProxy:**

```
sudo apt install -y haproxy
```

2. **Configure HAProxy for Exchange Server balancing:**

```
sudo nano /etc/haproxy/haproxy.cfg
```

3. **Add the following configuration:**

```
frontend smtp_front
    bind *:25
    mode tcp
    default_backend smtp_servers

backend smtp_servers
   balance roundrobin
    server mail1 172.24.13.56:25 check
    server mail2 172.24.13.65:25 check
```

4. **Restart HAProxy to apply changes:**

```
sudo systemctl restart haproxy
```

SECURITY HARDENING

To **enhance security**, the following **best practices** will be implemented:

- **Enable TLS Termination** to encrypt traffic at the load balancer level.
- **Rate Limiting to prevent DDoS attacks**:

```
limit_req_zone $binary_remote_addr zone=one:10m rate=30r/m;
```

- **Restrict access to the Load Balancer to authorized IPs**:

```
allow 192.168.1.0/24;
deny all;
```

- **Enable Web Application Firewall (WAF) for HTTP filtering**.

MONITORING & PERFORMANCE OPTIMIZATION

To monitor **load balancer performance** and **track traffic distribution**, we will integrate monitoring tools.

Installing Prometheus & Grafana for Load Balancer Monitoring

1. **Install Prometheus**:

```
sudo apt install prometheus -y
```

2. **Add Nginx Metrics Exporter**:

```
sudo apt install prometheus-nginx-exporter -y
```

3. **Install Grafana for visualization**:

```
sudo apt install grafana -y
```

4. **Set up dashboards to monitor traffic, request distribution, and errors.**

PFSENSE FIREWALL

To ensure **network security, traffic filtering, and VPN access**, INDOCAN is implementing **pfSense**, an open-source firewall solution. This section covers **installation, configuration, security policies, VPN setup, monitoring, and backup strategies** to provide a **secure, scalable, and highly available** network infrastructure.

INSTALLATION

Prerequisites:

- **pfSense 2.6** installed on 172.24.13.73.
- **2 vCPUs, 4GB RAM, 50GB SSD** allocated.
- **WAN and LAN interfaces properly assigned.**
- **Firewall rules configured for essential services.**

Installation Steps:

1. **Download pfSense ISO** from the official website.
2. **Create a bootable USB drive** using Rufus or Etcher.
3. **Boot the server from USB** and start the installation.
4. **Select "Install pfSense"** and configure basic network settings.
5. **Assign WAN (em0) and LAN (em1) interfaces**.
6. **Set LAN IP Address (172.24.13.1/24) and DHCP range (172.24.13.100-200)**.
7. **Reboot the system and access the web interface** at https://172.24.13.1.

FIREWALL RULES & SECURITY POLICIES

Inbound & Outbound Traffic Rules:

Rule	Source	Destination	Protocol	Action
Allow Web Access	Any	Web Server (172.24.13.54, 172.24.13.63)	HTTP/HTTPS	Allow
Allow Mail Traffic	Any	Exchange Servers (172.24.13.56, 172.24.13.65)	SMTP	Allow
Allow VPN Traffic	Remote VPN Users	Internal Network	OpenVPN	Allow
Block Unauthorized Access	Any	Internal Network	All	Block

IPS/IDS Configuration:

1. **Install Suricata for Intrusion Prevention:**

```
pkg install suricata
```

2. **Enable Suricata on WAN Interface.**
3. **Configure Snort rules for anomaly detection.**
4. **Enable Auto-Block for detected threats.**

VPN CONFIGURATION : REMOTE ACCESS FOR INDOCAN USERS

To enable **secure remote access**, **OpenVPN** will be configured.

OpenVPN Setup:

1. **Navigate to VPN → OpenVPN → Wizards.**
2. **Select Remote Access (User Auth).**
3. **Configure Tunnel Network (10.8.0.0/24).**
4. **Enable 256-bit AES Encryption.**
5. **Download OpenVPN Client Profile for Users.**

Site-to-Site VPN (New York ↔ Hong Kong):

- **New York pfSense Firewall (172.24.13.73)** will establish a **permanent VPN tunnel** with **Hong Kong (172.24.13.74)**.
- **BGP Routing** will be used to dynamically route traffic between sites.
- **Tunnel Encryption:** AES-256 with SHA-512 authentication.

NETWORK ADDRESS TRANSLATION (NAT) CONFIGURATION

1. **Navigate to Firewall → NAT → Outbound.**
2. **Select Hybrid NAT Mode.**
3. **Map Internal IPs to Public IP for external communication.**

LOAD BALANCING & HIGH AVAILABILITY

To ensure **continuous operation**, a **pfSense failover cluster** is implemented.

CARP Configuration (High Availability Failover):

1. **Primary Firewall:** 172.24.13.73
2. **Secondary Firewall:** 172.24.13.74
3. **Shared Virtual IP:** 172.24.13.1
4. **Enable State Synchronization for seamless failover.**

MONITORING & LOGGING

Setting Up pfSense Logging & Alerts

1. **Enable Syslog Server** for remote logging.
2. **Integrate with Zabbix for real-time monitoring.**
3. **Set Up Email Alerts** for firewall rule violations.

MATTERMOST COMMUNICATION SERVER

To ensure **secure, real-time internal communication**, INDOCAN is deploying **Mattermost**, an open-source collaboration and messaging platform. This section details **installation, configuration, user management, security policies, integration, and backup strategies** to enhance productivity and streamline communication between teams.

INSTALLATION

Prerequisites:

- **Ubuntu Server 22.04 LTS** installed on 172.24.13.75.
- **4 vCPUs, 16GB RAM, 200GB SSD** allocated.
- **PostgreSQL Database** for storing Mattermost data.
- **Firewall rules** configured to allow traffic on **HTTP (80), HTTPS (443), and WebSocket (8065).**

Installation Steps:

1. **Update System & Install Dependencies:**

```
sudo apt update && sudo apt upgrade -y
sudo apt install -y curl unzip tar
```

2. **Download & Extract Mattermost:**

```
wget https://releases.mattermost.com/8.0/mattermost-8.0-linux-amd64.tar.gz
tar -xvzf mattermost-8.0-linux-amd64.tar.gz
sudo mv mattermost /opt/
```

3. **Create a Mattermost User & Set Permissions:**

```
sudo useradd --system --user-group mattermost
sudo chown -R mattermost:mattermost /opt/mattermost
```

4. **Configure Database Connection in Mattermost Config:**

```
sudo nano /opt/mattermost/config/config.json

Set:

"DriverName": "postgres",
"DataSource": "postgres://mmuser:password@localhost/mattermost?sslmode=disable"
```

5. **Start the Mattermost Server:**

```
sudo -u mattermost /opt/mattermost/bin/mattermost
```

6. **Access Mattermost Web Interface** via http://server-ip:8065.

DATABASE CONFIGURATION

Mattermost requires **PostgreSQL** for structured data storage.

Steps:

1. **Install PostgreSQL:**

```
sudo apt install -y postgresql postgresql-contrib
```

2. **Create Database & User:**

```
CREATE DATABASE mattermost;
CREATE USER mmuser WITH PASSWORD 'securepassword';
ALTER DATABASE mattermost OWNER TO mmuser;
GRANT ALL PRIVILEGES ON DATABASE mattermost TO mmuser;
```

3. **Restart Services:**

```
sudo systemctl restart postgresql mattermost
```

USER MANAGEMENT & ROLES

Mattermost allows user authentication via **Active Directory (LDAP), SAML, or OAuth.**

User Roles:

Role	Permissions
Admin	Full access, manage users & channels
Manager	Create teams, configure notifications
User	Join channels, send messages

LDAP Integration for Authentication:

1. **Navigate to Admin Panel → Authentication → LDAP.**
2. **Enter AD Server Details** (e.g., ldap://ad.indocan.local).
3. **Set Search Base** to CN=Users,DC=indocan,DC=local.
4. **Enable Auto-Sync every 30 minutes.**

SECURITY HARDENING

To **enhance security**, the following best practices are implemented:

- **Enable Two-Factor Authentication (2FA)** for all users.
- **Enforce SSL/TLS for secure communication:**

```
sudo certbot --nginx -d mattermost.INDOCAN.LOCAL
```

- **Restrict API Access** to internal IP ranges.
- **Implement End-to-End Encryption (E2EE) for messaging.**

MONITORING & LOGGING

To **monitor Mattermost performance and security**, we integrate monitoring tools.

Steps:

1. **Enable Syslog Logging for Mattermost Events.**
2. **Integrate with Prometheus & Grafana** for dashboard monitoring.
3. **Set Up Email Alerts** for system errors & unusual activity.

DASHBOARD WITH GUACAMOLE RDP

INSTALLATION

To enable secure remote access to all servers within our infrastructure, Apache Guacamole was deployed as a remote desktop gateway. Guacamole is a clientless remote desktop solution that supports RDP, VNC, and SSH, allowing seamless access to multiple systems through a web-based interface.

1. **System Requirements:**

   Operating System: Ubuntu 22.04 LTS

   RAM: Minimum 4GB (8GB recommended)

   CPU: 2 vCPUs (4 vCPUs recommended for concurrent sessions)

   Disk Space: 20GB minimum

   Network: 1Gbps Ethernet connection

2. **Installing Prerequisites:**

   Before installing Guacamole, several dependencies must be installed:

```
sudo apt update && sudo apt upgrade -y
sudo apt install -y build-essential libcairo2-dev libjpeg-turbo8-dev \
libpng-dev libtool-bin uuid-dev freerdp2-dev libssh2-1-dev \
libtelnet-dev libvncserver-dev libpulse-dev libssl-dev \
libvorbis-dev libwebp-dev
```

3. **Downloading and Compiling Guacamole:**

   The latest version of Guacamole and its components are downloaded and compiled:

```
wget https://downloads.apache.org/guacamole/1.5.0/source/guacamole-server-1.5.0.tar.gz
tar -xvzf guacamole-server-1.5.0.tar.gz
cd guacamole-server-1.5.0
./configure --with-init-dir=/etc/init.d
make
sudo make install
sudo ldconfig
```

4. **Starting the Guacamole Server:**

```
sudo systemctl enable guacd
sudo systemctl start guacd
```

5. **Installing Guacamole Client (Web Application):**

```
wget https://downloads.apache.org/guacamole/1.5.0/binary/guacamole-1.5.0.war
sudo mv guacamole-1.5.0.war /var/lib/tomcat9/webapps/guacamole.war
sudo systemctl restart tomcat9
```

6. **Configuring Guacamole Properties:**

   The configuration file /etc/guacamole/guacamole.properties is created to define authentication and connection parameters:

```
guacd-hostname: localhost
guacd-port: 4822
auth-provider: net.sourceforge.guacamole.net.basic.BasicFileAuthenticationProvider
basic-user-mapping: /etc/guacamole/user-mapping.xml
```

DATABASE CONFIGURATION

To store authentication details and session logs securely, Guacamole is integrated with a MySQL database. This allows centralized user management and connection tracking.

1. Installing MySQL and Creating the Database

First, install MySQL and set up a dedicated database for Guacamole:

```
sudo apt update && sudo apt install -y mysql-server
sudo systemctl start mysql
sudo systemctl enable mysql
```

After installation, secure MySQL using:

```
sudo mysql_secure_installation
```

Set a root password.

Remove anonymous users.

Disallow remote root login.

Remove test databases.

2. Creating a Guacamole Database and User

Log in to MySQL and create a database for Guacamole:

```
sudo mysql -u root -p
CREATE DATABASE guacamole_db;
CREATE USER 'guac_user'@'localhost' IDENTIFIED BY 'StrongPassword';
GRANT ALL PRIVILEGES ON guacamole_db.* TO 'guac_user'@'localhost';
FLUSH PRIVILEGES;
EXIT;
```

3. Importing the Guacamole Schema

Guacamole provides an SQL schema that needs to be imported into the database:

```
cat /usr/share/guacamole/schema/mysql/schema.sql | sudo mysql -u guac_user -p guacamole_db
```

4. Configuring Guacamole to Use MySQL

Modify the Guacamole configuration file (/etc/guacamole/guacamole.properties) to use MySQL authentication:

```
mysql-hostname: localhost

mysql-database: guacamole_db

mysql-username: guac_user

mysql-password: StrongPassword

Ensure the changes are applied:

sudo systemctl restart tomcat9 guacd
```

USER MANAGEMENT & ROLES

Guacamole provides a robust user management system that enables role-based access control (RBAC) for secure and efficient remote access management. This section details the setup of user authentication, role assignment, and permission management.

1. Creating and Managing Users

To create and manage users in Guacamole, follow these steps:

1. Log in as an Administrator

Navigate to the Guacamole web interface.

Log in with the admin credentials.

Click on "Settings" > "Users" to manage users.

2. Adding New Users

Click "New User" and fill in the following details:

Username (e.g., admin, support_user, devops)

Password (Set a secure password)

Account Expiry (Optional)

Multi-Factor Authentication (MFA) Settings (If enabled)

3. Assigning Roles

Guacamole allows assigning users to different roles based on access requirements:

Administrator – Full control over system settings and user management.

IT Support – Can manage connections and troubleshoot issues.

Developers – Limited access to test environments.

End Users – Can only access assigned remote desktops.

2. Configuring Group-Based Access Control

Instead of managing individual permissions, Guacamole allows creating user groups for easier access control:

- Navigate to "Settings" > "Groups" and create groups such as:

IT-Support

Developers

Management

General Users

- Assign users to appropriate groups and define access control policies.

3. Enforcing Security Policies

For enhanced security, apply the following policies:

Require Strong Passwords

Enforce password complexity rules (minimum length, special characters, numbers).

Use password expiration policies.

Enable Multi-Factor Authentication (MFA)

Guacamole supports TOTP-based MFA.

Users can link their accounts with an authenticator app

Restrict IP-Based Access

Configure allowlists/denylists to restrict unauthorized IP addresses.

Implement VPN-based access for remote users.

Limit Concurrent Sessions

Prevent multiple logins per user session to avoid session hijacking.

CONFIGURING REMOTE DESKTOP ACCESS

Guacamole supports various remote desktop protocols, including RDP (Windows), SSH (Linux), and VNC. This section details how to configure and optimize remote desktop access for seamless operations.

1. Adding a New Remote Desktop Connection

Log in to Guacamole as an administrator.

Navigate to "Settings" > "Connections" and click "New Connection".

Provide the following details:

- Name: Assign a unique name to the connection
- Protocol: Select RDP, SSH, or VNC depending on the target system.
- Hostname/IP Address: Enter the private or public IP of the target server.
- Port: Use default ports (3389 for RDP, 22 for SSH, 5900 for VNC), or specify custom ports.
- Username & Password: Store login credentials securely in Guacamole or enable user login prompt.

2. Optimizing RDP Performance

To ensure a smooth user experience, optimize RDP connections by:

- Enabling Network-Level Authentication (NLA) for secure authentication.
- Reducing Color Depth to improve performance on low-bandwidth connections.

- Disabling Wallpaper & Animations to save bandwidth.
- Adjusting Compression Settings to prioritize speed over quality.
- Enabling Audio Redirection if remote sound output is needed.

3. Configuring SSH for Secure CLI Access

For Linux server management, configure SSH connections with Guacamole:

- Use Key-Based Authentication instead of passwords for added security.
- Restrict root logins and enforce sudo-only access.
- Enable session logging for auditing and compliance.
- Implement IP allowlists to limit SSH access to specific locations.

4. Managing Remote Access Policies

To control who can access remote desktops:

- Assign specific users/groups to each connection.
- Configure session timeouts to automatically disconnect idle users.
- Enforce multi-factor authentication (MFA) for added security.
- Monitor active sessions via the admin panel to detect unauthorized access.

SECURITY HARDENING & BEST PRACTICES

To ensure secure and reliable remote desktop access, Guacamole must be configured with robust security measures. This section details best practices for hardening the Guacamole environment.

1. Enforcing Secure Authentication

To protect against unauthorized access, implement the following authentication methods:

- Multi-Factor Authentication (MFA): Enable TOTP-based 2FA (Google Authenticator, Authy).
- LDAP & Active Directory Integration: Centralize authentication with existing domain credentials.
- RADIUS/TACACS+ Authentication: Use enterprise-grade authentication services.
- Fail2Ban Configuration: Set up intrusion detection to block brute-force attempts.
- Session Expiry & Auto Logout: Enforce idle session timeouts to prevent unauthorized persistence.

2. Securing the Guacamole Server

To minimize vulnerabilities, apply the following security measures:

- Enable HTTPS (SSL/TLS): Configure Guacamole with Let's Encrypt or an internal CA certificate.
- Restrict Public Access: Allow only specific IPs to access the Guacamole web interface.
- Use Strong Password Policies: Enforce minimum length, complexity, and rotation policies.
- Regularly Update Guacamole & Dependencies: Keep the software up to date to patch vulnerabilities.
- Disable Unused Protocols: If only RDP is required, disable VNC and SSH to reduce attack surface.

3. Network Security & Firewall Rules

To prevent unauthorized access:

- Place Guacamole Behind a Reverse Proxy (NGINX/Apache) for additional security layers.
- Use VPN or Zero Trust Network Access (ZTNA) to restrict external access.
- Apply Firewall Rules (pfSense, IPTables) to limit access based on location/IP.
- Log & Monitor Sessions using SIEM tools (Splunk, Graylog) for anomaly detection.

4. Role-Based Access Control (RBAC)

Restrict user access based on roles:

- Admin Users: Full access to manage Guacamole and create new connections.
- IT Support/Helpdesk: Access to pre-defined machines but no administrative privileges.
- Regular Users: Restricted access to only their assigned desktops/servers.

MONITORING & PERFORMANCE OPTIMIZATION

To ensure Guacamole operates efficiently and remains secure, continuous monitoring and performance optimizations are essential. This section covers strategies to track system health, optimize resource usage, and troubleshoot performance issues.

1. System Monitoring & Logging

To maintain visibility over the Guacamole server and detect potential issues:

- Enable Syslog Logging: Configure Guacamole logs to be forwarded to centralized logging servers for real-time monitoring.
- Guacamole Session Logs: Track user logins, remote session activity, and session duration for auditing.
- Enable Debug Mode: In cases of troubleshooting, enable debugging but disable it afterward to prevent unnecessary performance impact.
- Integrate with SIEM Solutions: Link Guacamole logs with Security Information and Event Management (SIEM) tools for anomaly detection.

2. Resource Optimization & Performance Tuning

To optimize performance and provide a smooth remote access experience:

Allocate Adequate CPU & Memory:

- Assign at least 4 vCPUs and 8GB RAM to the Guacamole server, scaling based on concurrent users.
- Enable resource limits (cgroups) to prevent resource starvation.

Optimize Remote Protocols (RDP/VNC/SSH):

- Enable Adaptive Compression to reduce bandwidth usage.
- Adjust Frame Rate based on connection type (higher for LAN, lower for WAN).
- Use Efficient Video Encoding (H.264 for RDP) to improve performance over slow networks.

Load Balancing for High Availability:

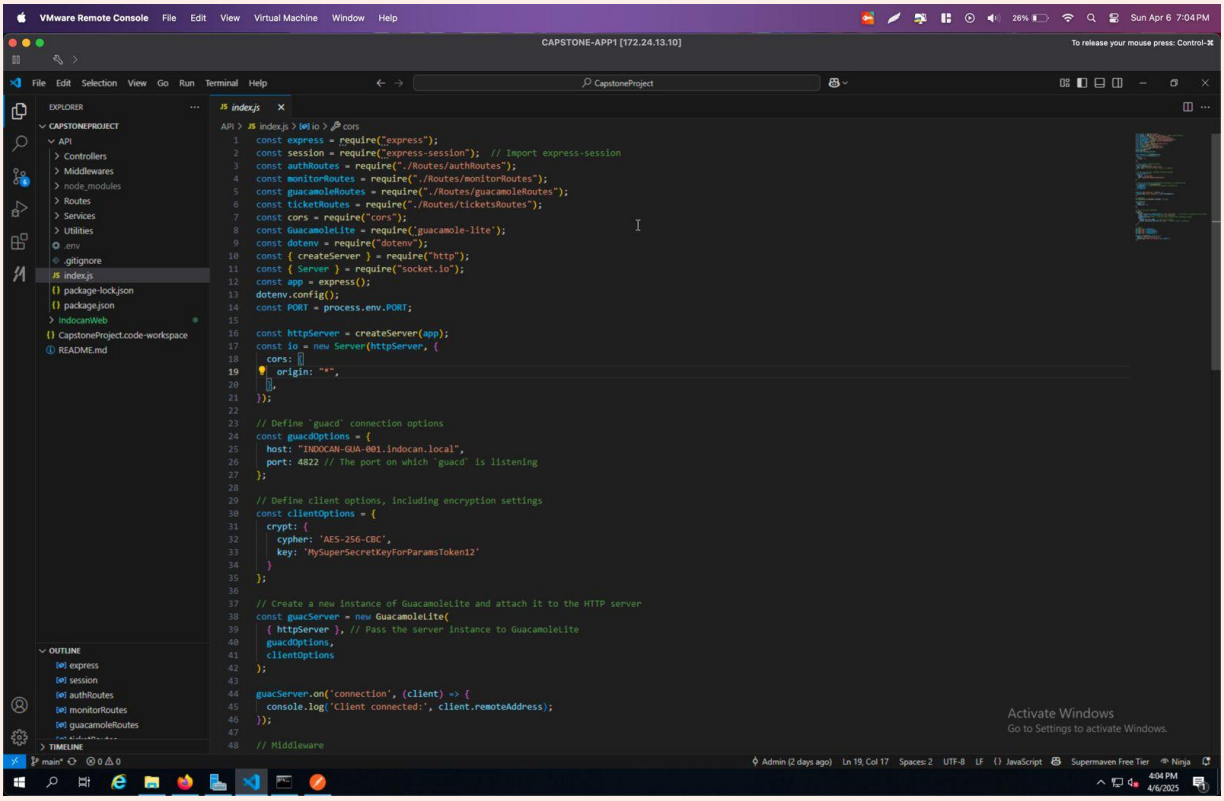
- Deploy Guacamole with NGINX or HAProxy for load balancing across multiple servers.
- Implement Guacamole Database Replication to ensure redundancy.

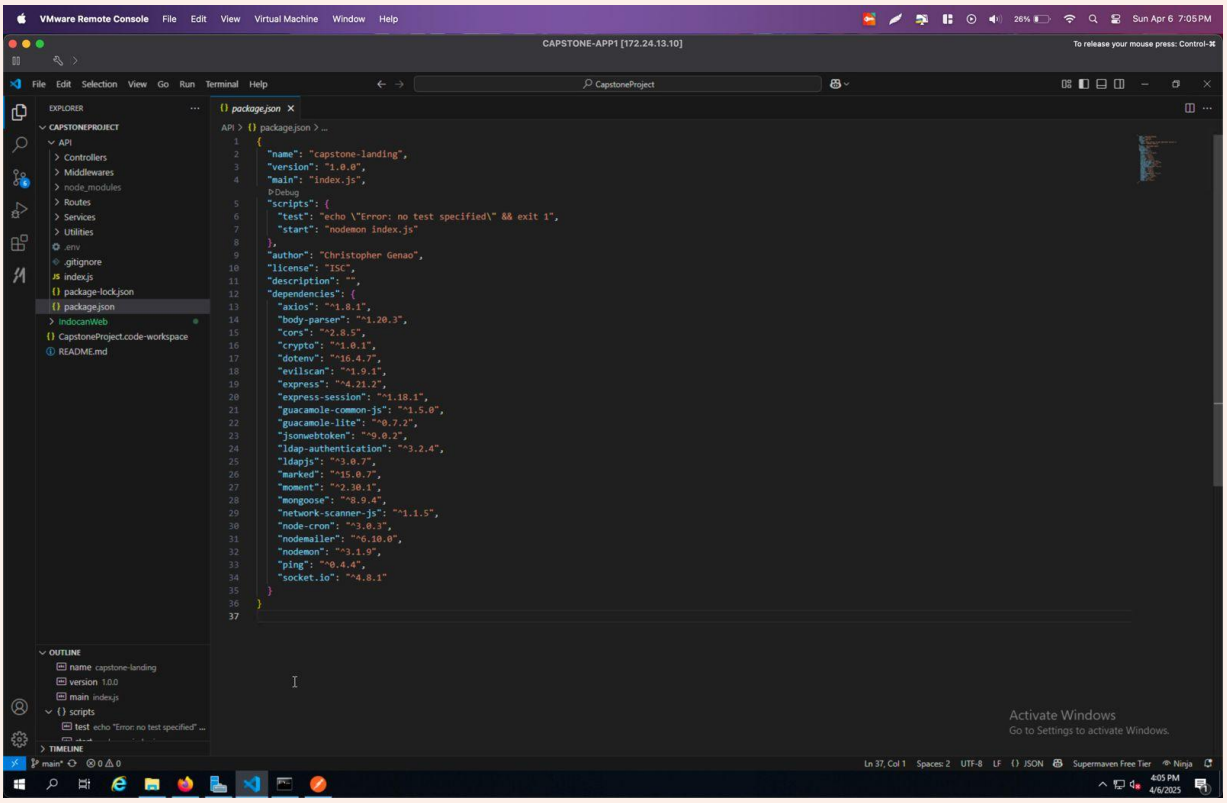
3. Troubleshooting Performance Issues

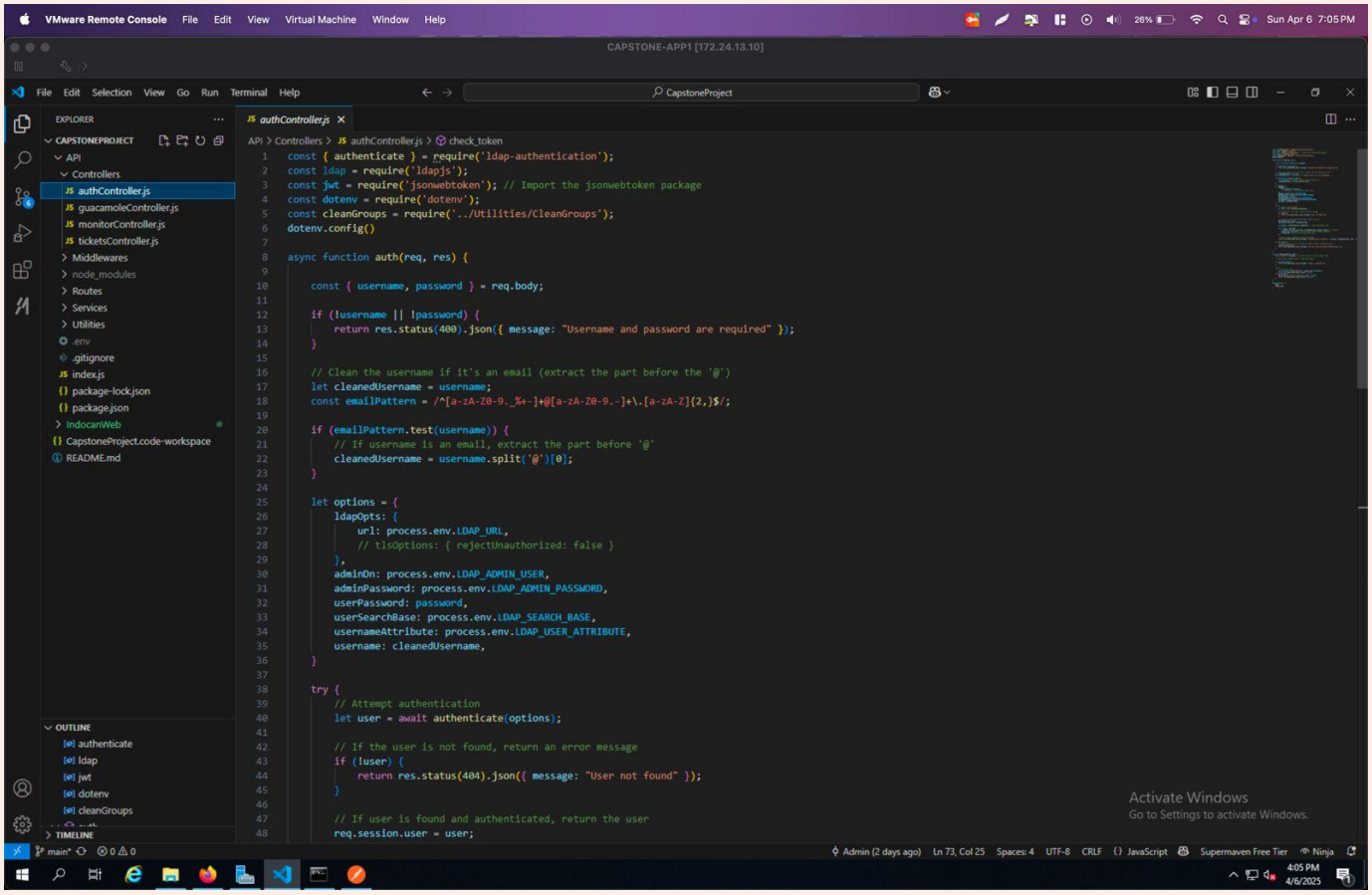
- To diagnose and resolve slow remote access:
- Check Server Load using htop, top, or Glances.
- Monitor Network Latency & Bandwidth Usage via Wireshark or NetFlow analyzers.
- Optimize JVM Heap Size
- Audit Logs for Errors: Identify excessive failed login attempts or unauthorized access.

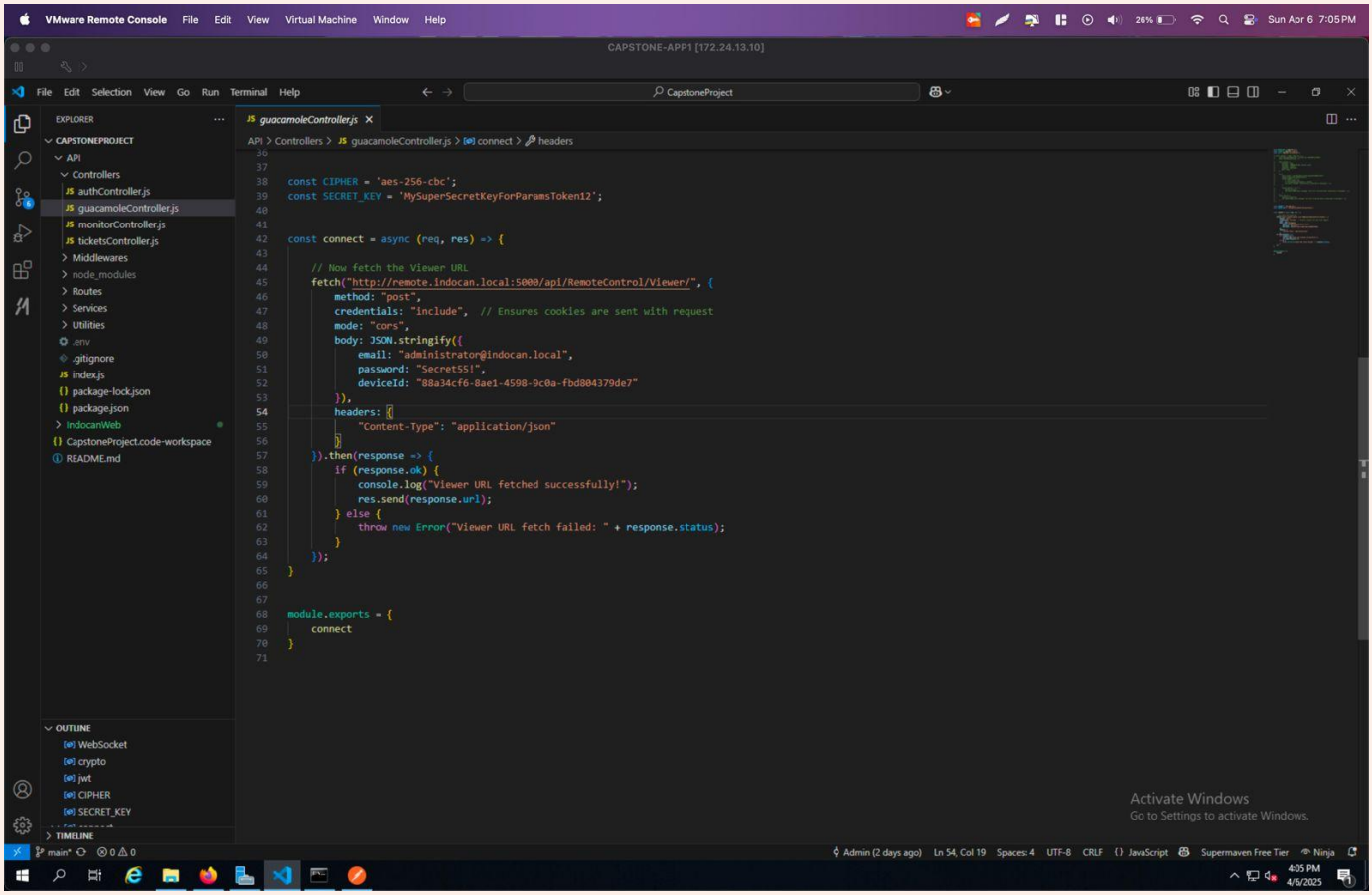
WORKING PROJECT

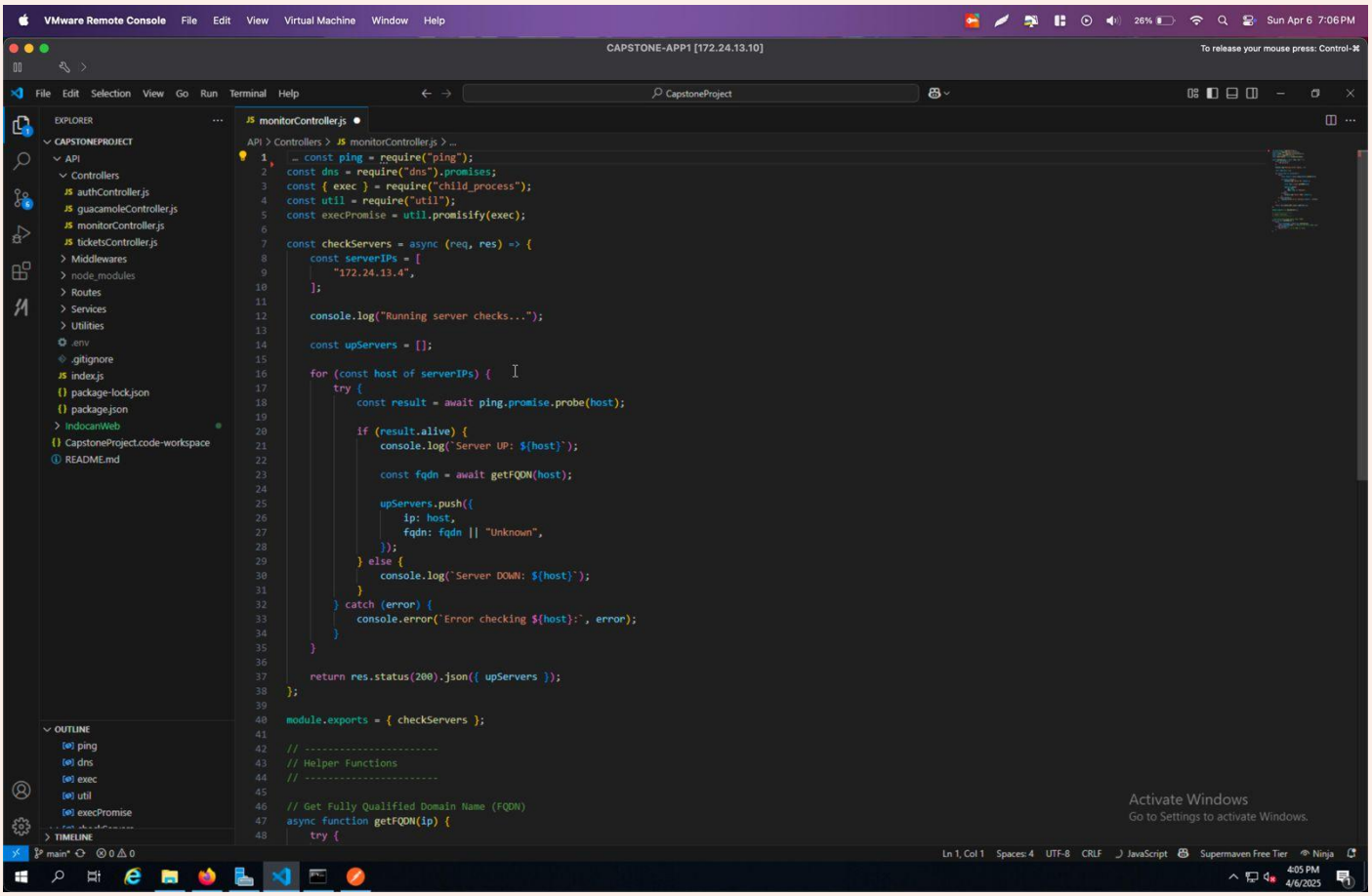
API CODE

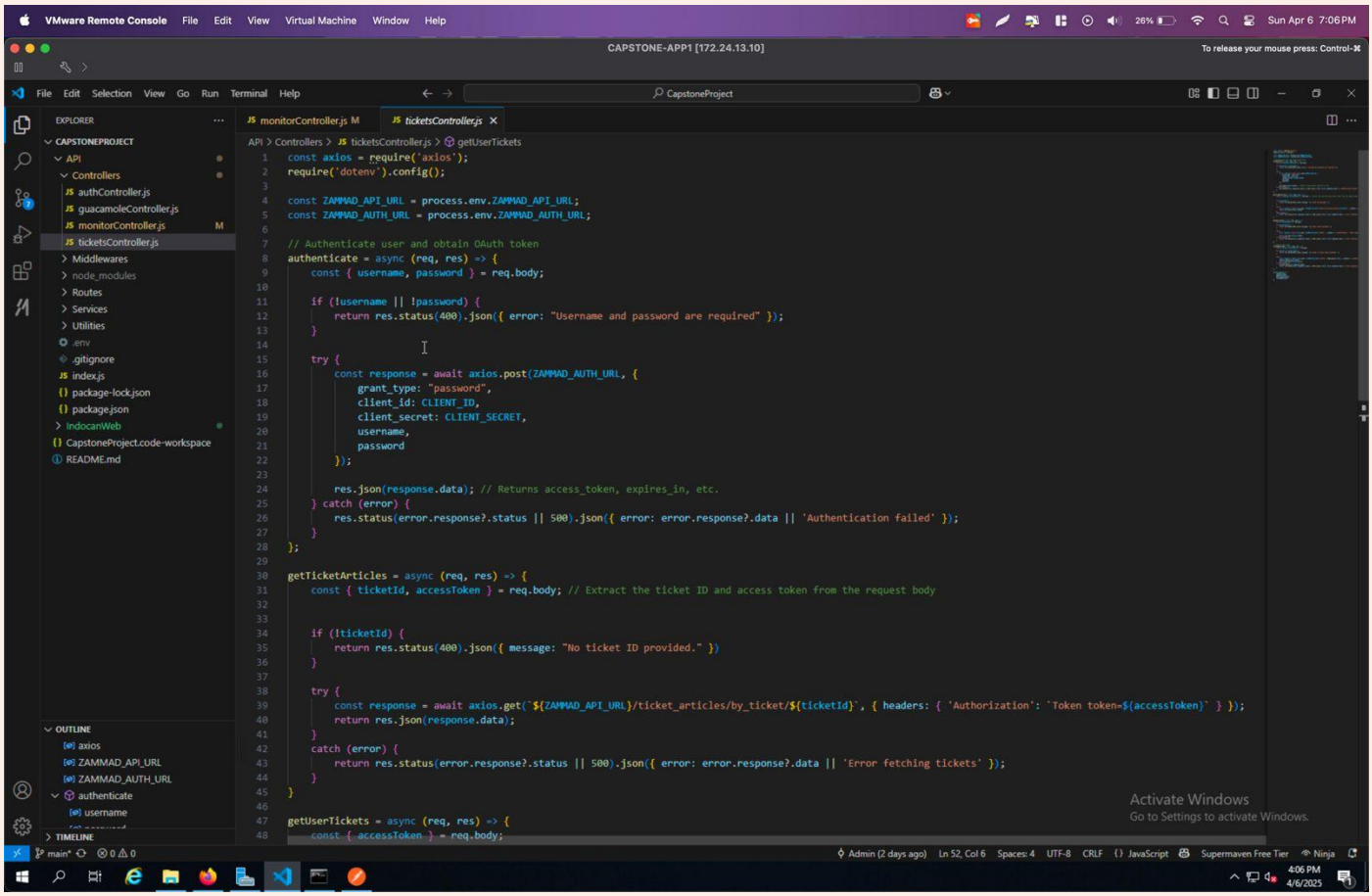


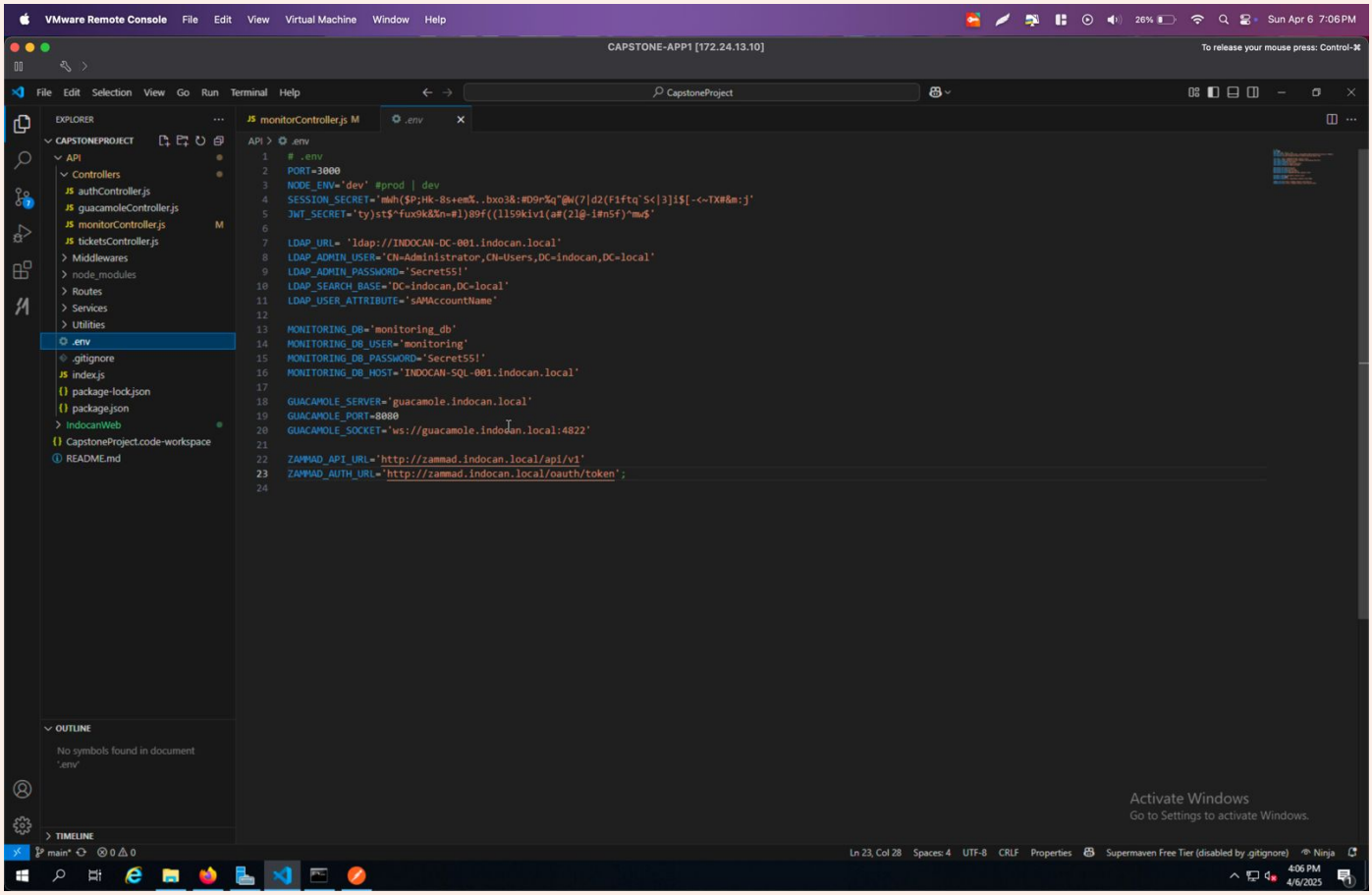


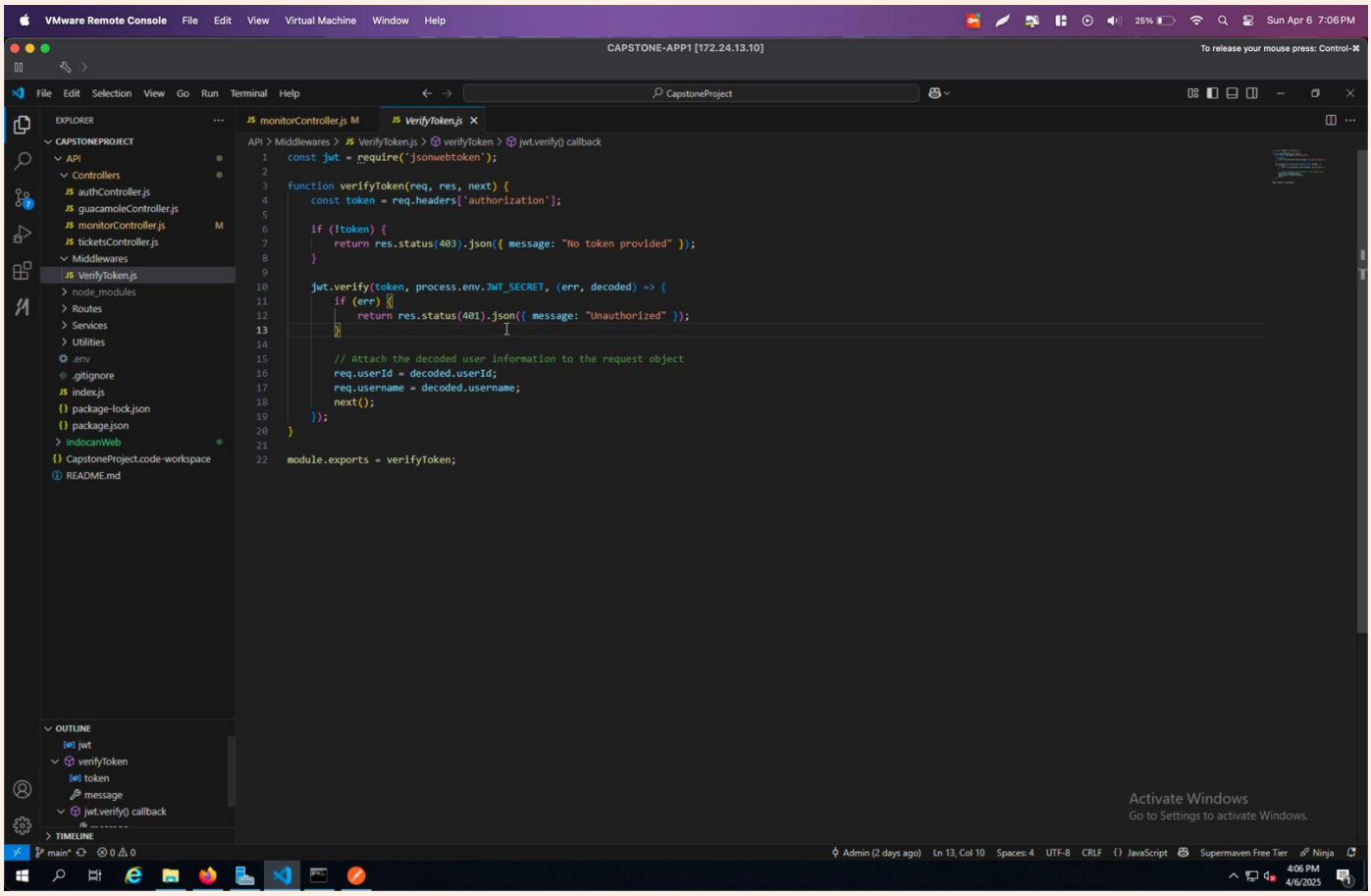

VMware Remote Console File Edit View Virtual Machine Window Help
CAPSTONE-APP1 [172.24.13.10]
File Edit Selection View Go Run Terminal Help
CapstoneProject
authController.js
API > Controllers > authController.js > check_token
const { authenticate } = require('ldap-authentication');
const ldap = require('ldapjs');
const jwt = require('jsonwebtoken'); // Import the jsonwebtoken package
const dotenv = require('dotenv');
const cleanGroups = require('../Utilities/CleanGroups');
dotenv.config()
async function auth(req, res) {
const { username, password } = req.body;
if (!username || !password) {
return res.status(400).json({ message: "Username and password are required" });
}
// Clean the username if it's an email (extract the part before the '@')
let cleanedUsername = username;
const emailPattern = /^[a-zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}$/;
if (emailPattern.test(username)) {
// If username is an email, extract the part before '@'
cleanedUsername = username.split('@')[0];
}
let options = {
ldapOpts: {
url: process.env.LDAP_URL,
// tlsOptions: { rejectUnauthorized: false }
},
adminDn: process.env.LDAP_ADMIN_USER,
adminPassword: process.env.LDAP_ADMIN_PASSWORD,
userPassword: password,
userSearchBase: process.env.LDAP_SEARCH_BASE,
usernameAttribute: process.env.LDAP_USER_ATTRIBUTE,
username: cleanedUsername,
}
try {
// Attempt authentication
let user = await authenticate(options);
// If the user is not found, return an error message
if (!user) {
return res.status(404).json({ message: "User not found" });
}
// If user is found and authenticated, return the user
req.session.user = user;
Activate Windows
Go to Settings to activate Windows.

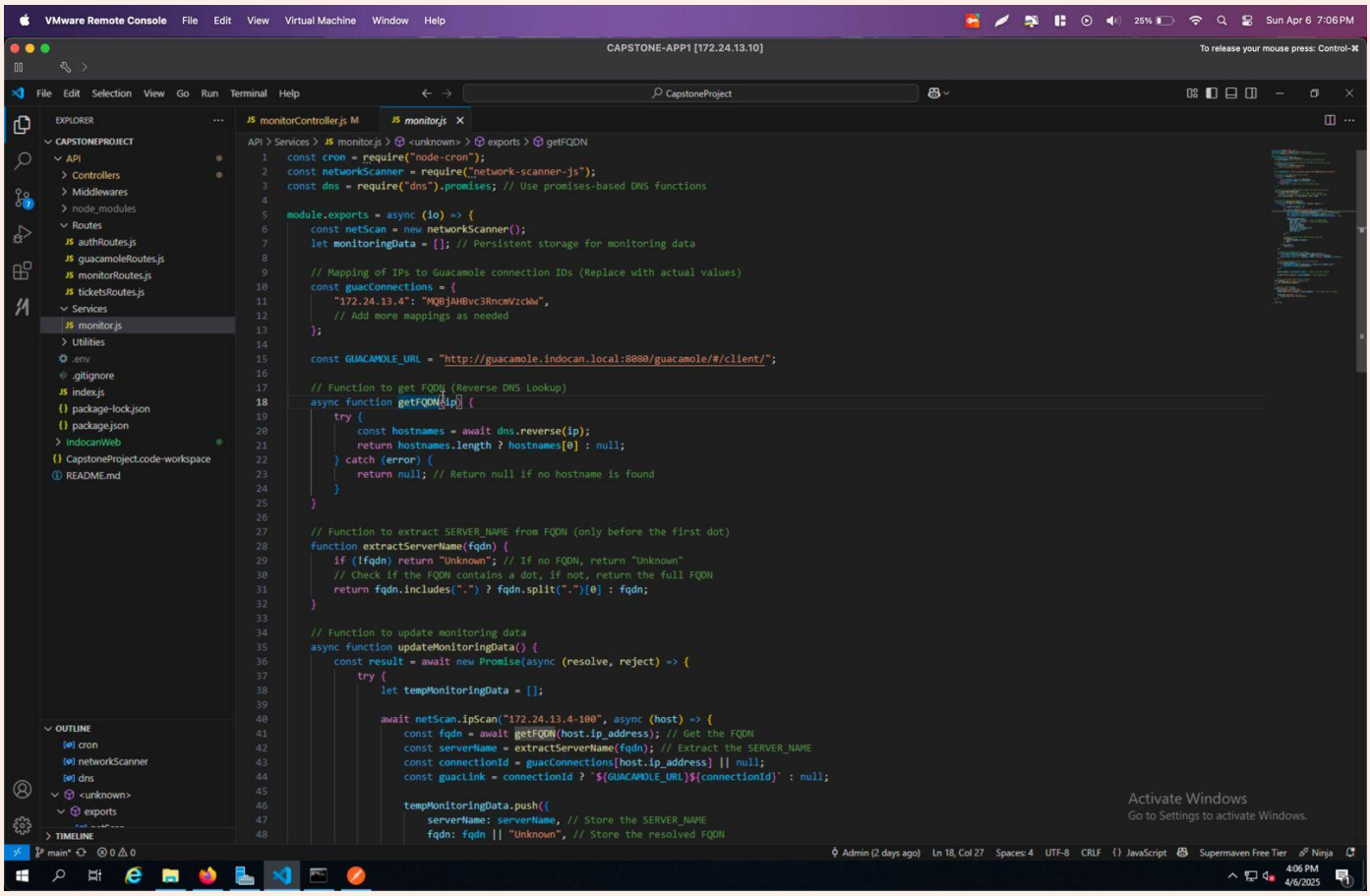

VMware Remote Console File Edit View Virtual Machine Window Help
CAPSTONE-APP1 [172.24.13.10]
File Edit Selection View Go Run Terminal Help
CapstoneProject
guacamoleController.js
API > Controllers > guacamoleController.js > connect > headers
const CIPHER = 'aes-256-cbc';
const SECRET_KEY = 'MySuperSecretKeyForParamsToken12';
const connect = async (req, res) => {
// Now fetch the Viewer URL
fetch("http://remote.indocan.local:5000/api/RemoteControl/Viewer/", {
method: "post",
credentials: "include", // Ensures cookies are sent with request
mode: "cors",
body: JSON.stringify({
email: "administrator@indocan.local",
password: "Secret55!",
deviceId: "88a34cf6-8ae1-4598-9c0a-fbd804379de7"
}),
headers: {
"Content-Type": "application/json"
}
}).then(response => {
if (response.ok) {
console.log("Viewer URL fetched successfully!");
res.send(response.url);
} else {
throw new Error("Viewer URL fetch failed: " + response.status);
}
});
}
module.exports = {
connect
}
Activate Windows
Go to Settings to activate Windows.

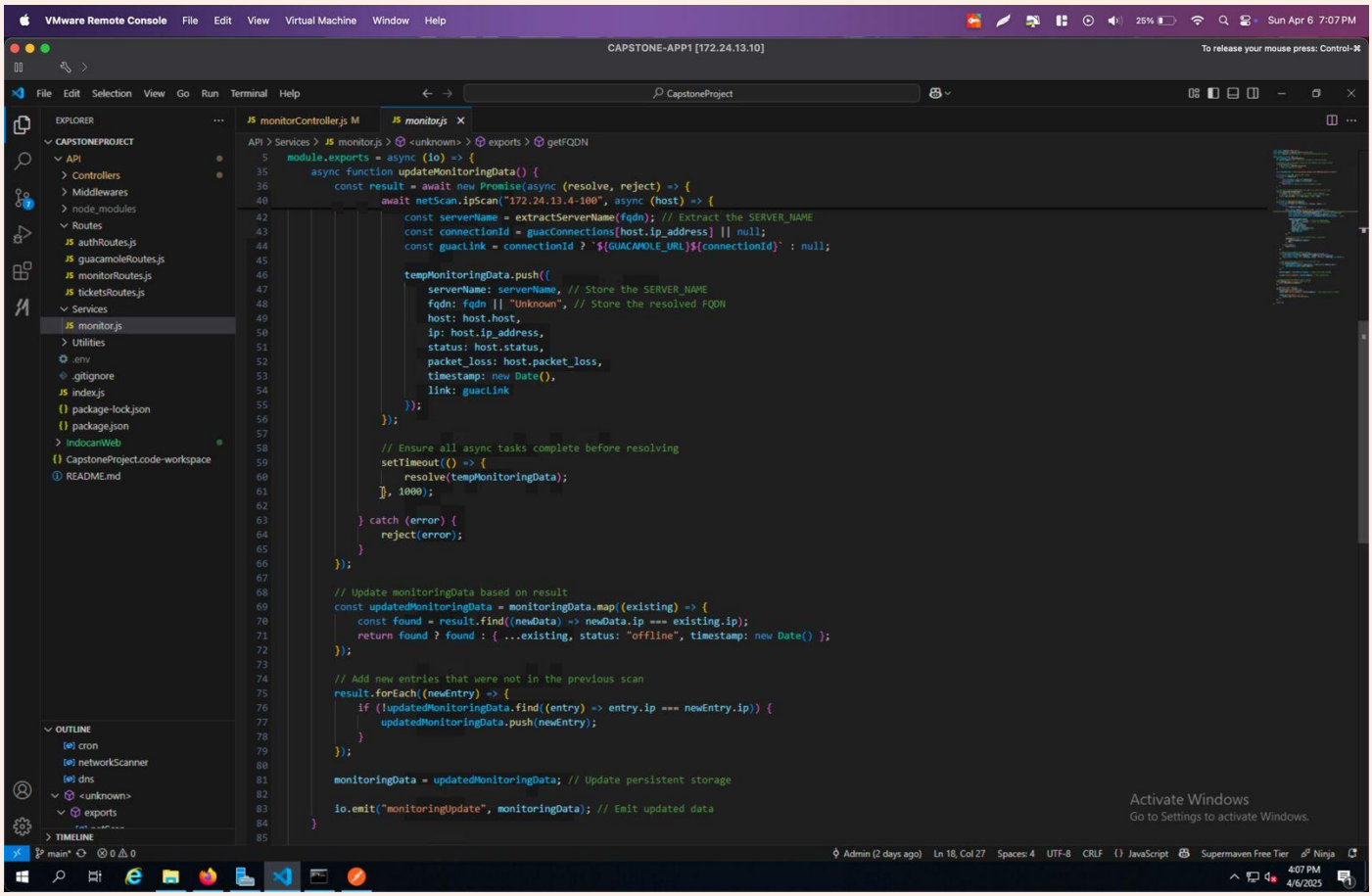

VMware Remote Console File Edit View Virtual Machine Window Help
CAPSTONE-APP1 [172.24.13.10]
File Edit Selection View Go Run Terminal Help
CapstoneProject
EXPLORER
CAPSTONEPROJECT
API
Controllers
authController.js
guacamoleController.js
monitorController.js
ticketsController.js
Middlewares
node_modules
Routes
Services
Utilities
.env
.gitignore
index.js
package-lock.json
package.json
IndocanWeb
CapstoneProject.code-workspace
README.md
OUTLINE
ping
dns
exec
util
execPromise
TIMELINE
monitorController.js
API > Controllers > monitorController.js > ...
const ping = require("ping");
const dns = require("dns").promises;
const { exec } = require("child_process");
const util = require("util");
const execPromise = util.promisify(exec);
const checkServers = async (req, res) => {
const serverIPs = [
"172.24.13.4",
];
console.log("Running server checks...");
const upServers = [];
for (const host of serverIPs) {
try {
const result = await ping.promise.probe(host);
if (result.alive) {
console.log(`Server UP: ${host}`);
const fqdn = await getFQDN(host);
upServers.push({
ip: host,
fqdn: fqdn || "Unknown",
});
} else {
console.log(`Server DOWN: ${host}`);
}
} catch (error) {
console.error(`Error checking ${host}:`, error);
}
}
return res.status(200).json({ upServers });
};
module.exports = { checkServers };
// ----------------------
// Helper Functions
// ----------------------
// Get Fully Qualified Domain Name (FQDN)
async function getFQDN(ip) {
try {
Activate Windows
Go to Settings to activate Windows.
Ln 1, Col 1 Spaces: 4 UTF-8 CRLF JavaScript Supermaven Free Tier Ninja
4:05 PM
4/6/2025

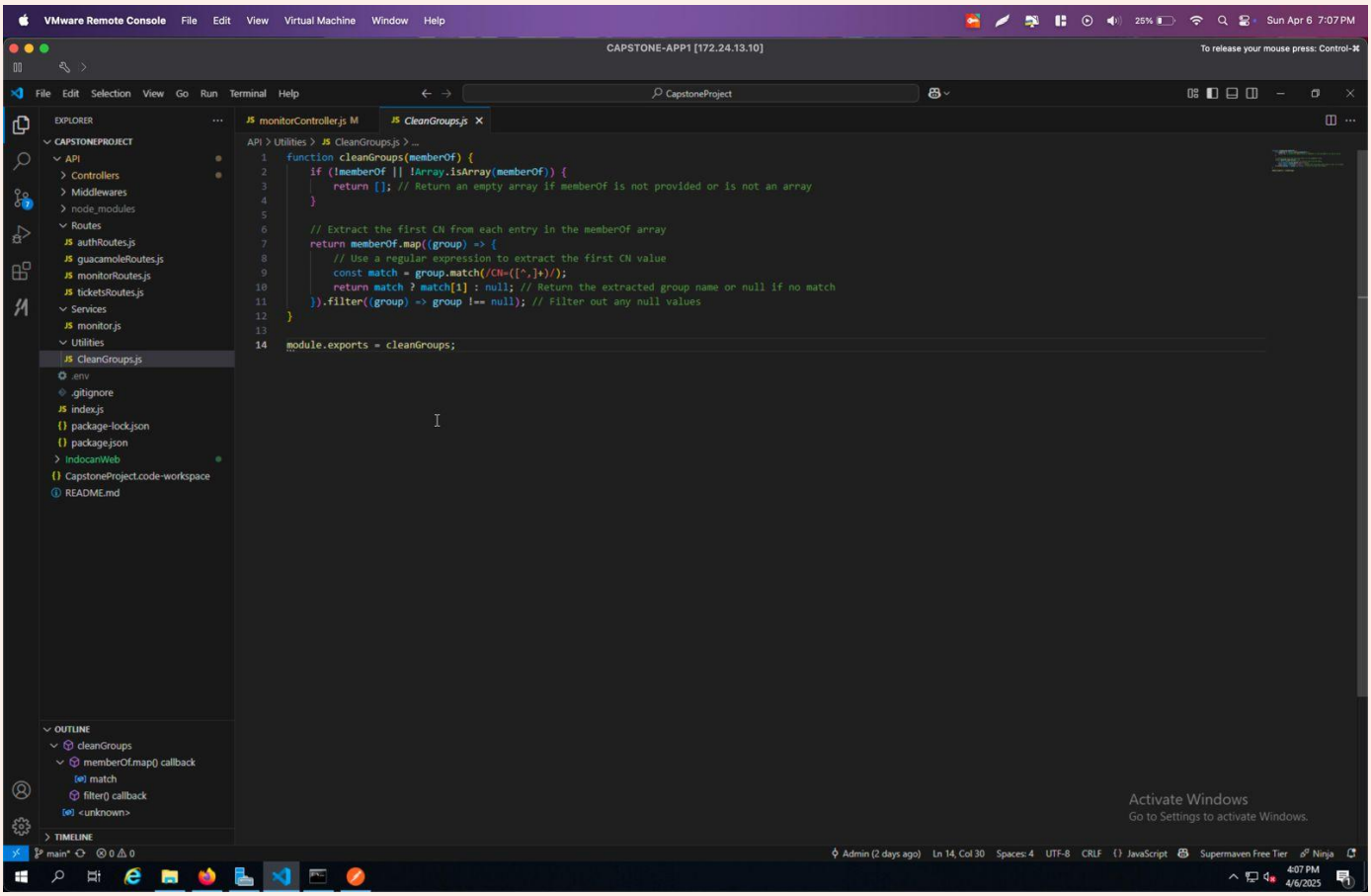

VMware Remote Console File Edit View Virtual Machine Window Help
CAPSTONE-APP1 [172.24.13.10]
File Edit Selection View Go Run Terminal Help
CapstoneProject
EXPLORER
CAPSTONEPROJECT
API
Controllers
authController.js
guacamoleController.js
monitorController.js
ticketsController.js
Middlewares
node_modules
Routes
Services
Utilities
.env
.gitignore
index.js
package-lock.json
package.json
IndocanWeb
CapstoneProject.code-workspace
README.md
OUTLINE
axios
ZAMMAD_API_URL
ZAMMAD_AUTH_URL
authenticate
username
TIMELINE
monitorController.js M
ticketsController.js
API > Controllers > ticketsController.js > getUserTickets
const axios = require('axios');
require('dotenv').config();
const ZAMMAD_API_URL = process.env.ZAMMAD_API_URL;
const ZAMMAD_AUTH_URL = process.env.ZAMMAD_AUTH_URL;
// Authenticate user and obtain OAuth token
authenticate = async (req, res) => {
const { username, password } = req.body;
if (!username || !password) {
return res.status(400).json({ error: "Username and password are required" });
}
try {
const response = await axios.post(ZAMMAD_AUTH_URL, {
grant_type: "password",
client_id: CLIENT_ID,
client_secret: CLIENT_SECRET,
username,
password
});
res.json(response.data); // Returns access_token, expires_in, etc.
} catch (error) {
res.status(error.response?.status || 500).json({ error: error.response?.data || 'Authentication failed' });
}
};
getTicketArticles = async (req, res) => {
const { ticketId, accessToken } = req.body; // Extract the ticket ID and access token from the request body
if (!ticketId) {
return res.status(400).json({ message: "No ticket ID provided." })
}
try {
const response = await axios.get(`${ZAMMAD_API_URL}/ticket_articles/by_ticket/${ticketId}`, { headers: { 'Authorization': `Token token=${accessToken}` } });
return res.json(response.data);
}
catch (error) {
return res.status(error.response?.status || 500).json({ error: error.response?.data || 'Error fetching tickets' });
}
}
getUserTickets = async (req, res) => {
const { accessToken } = req.body;
Activate Windows
Go to Settings to activate Windows.
Admin (2 days ago) Ln 52, Col 6 Spaces: 4 UTF-8 CRLF JavaScript Supermaven Free Tier Ninja
4:06 PM
4/6/2025

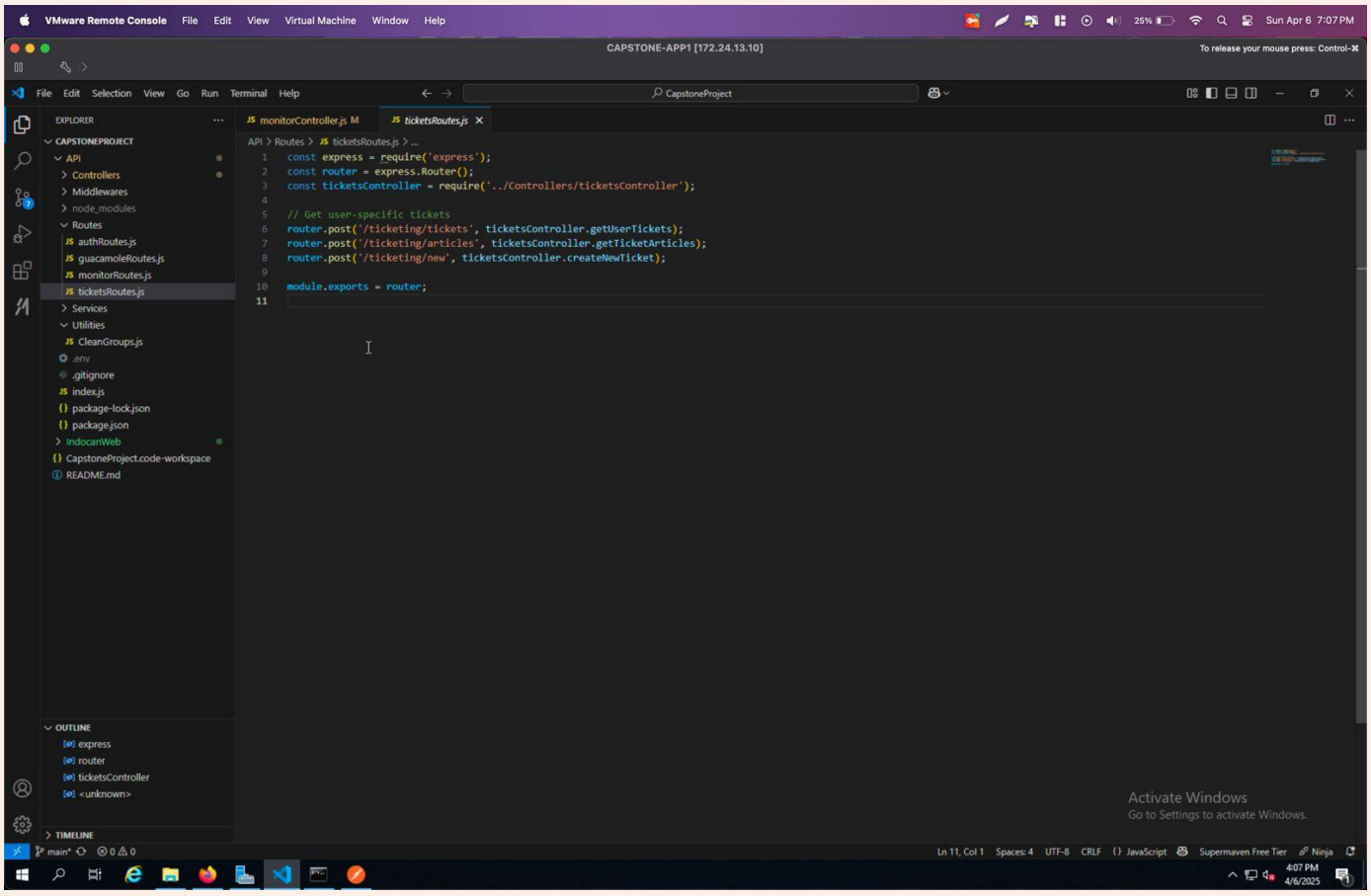

VMware Remote Console File Edit View Virtual Machine Window Help
CAPSTONE-APP1 [172.24.13.10]
File Edit Selection View Go Run Terminal Help
CapstoneProject
monitorController.js M
.env
# .env
PORT=3000
NODE_ENV='dev' #prod | dev
SESSION_SECRET='mWh($P;Hk-8s+em%..bxo3&:#D9r%q"@W(7|d2(F1ftq`S<|3]i$[-<~TX#&m:j'
JWT_SECRET='ty)st$^fux9k&%n~#l)89f((1l59kiv1(a#(2l@-i#n5f)^m<$'
LDAP_URL= 'ldap://INDOCAN-DC-001.indocan.local'
LDAP_ADMIN_USER='CN=Administrator,CN=Users,DC=indocan,DC=local'
LDAP_ADMIN_PASSWORD='Secret55!'
LDAP_SEARCH_BASE='DC=indocan,DC=local'
LDAP_USER_ATTRIBUTE='sAMAccountName'
MONITORING_DB='monitoring_db'
MONITORING_DB_USER='monitoring'
MONITORING_DB_PASSWORD='Secret55!'
MONITORING_DB_HOST='INDOCAN-SQL-001.indocan.local'
GUACAMOLE_SERVER='guacamole.indocan.local'
GUACAMOLE_PORT=8080
GUACAMOLE_SOCKET='ws://guacamole.indocan.local:4822'
ZAMMAD_API_URL='http://zammad.indocan.local/api/v1'
ZAMMAD_AUTH_URL='http://zammad.indocan.local/oauth/token';
Activate Windows
Go to Settings to activate Windows.

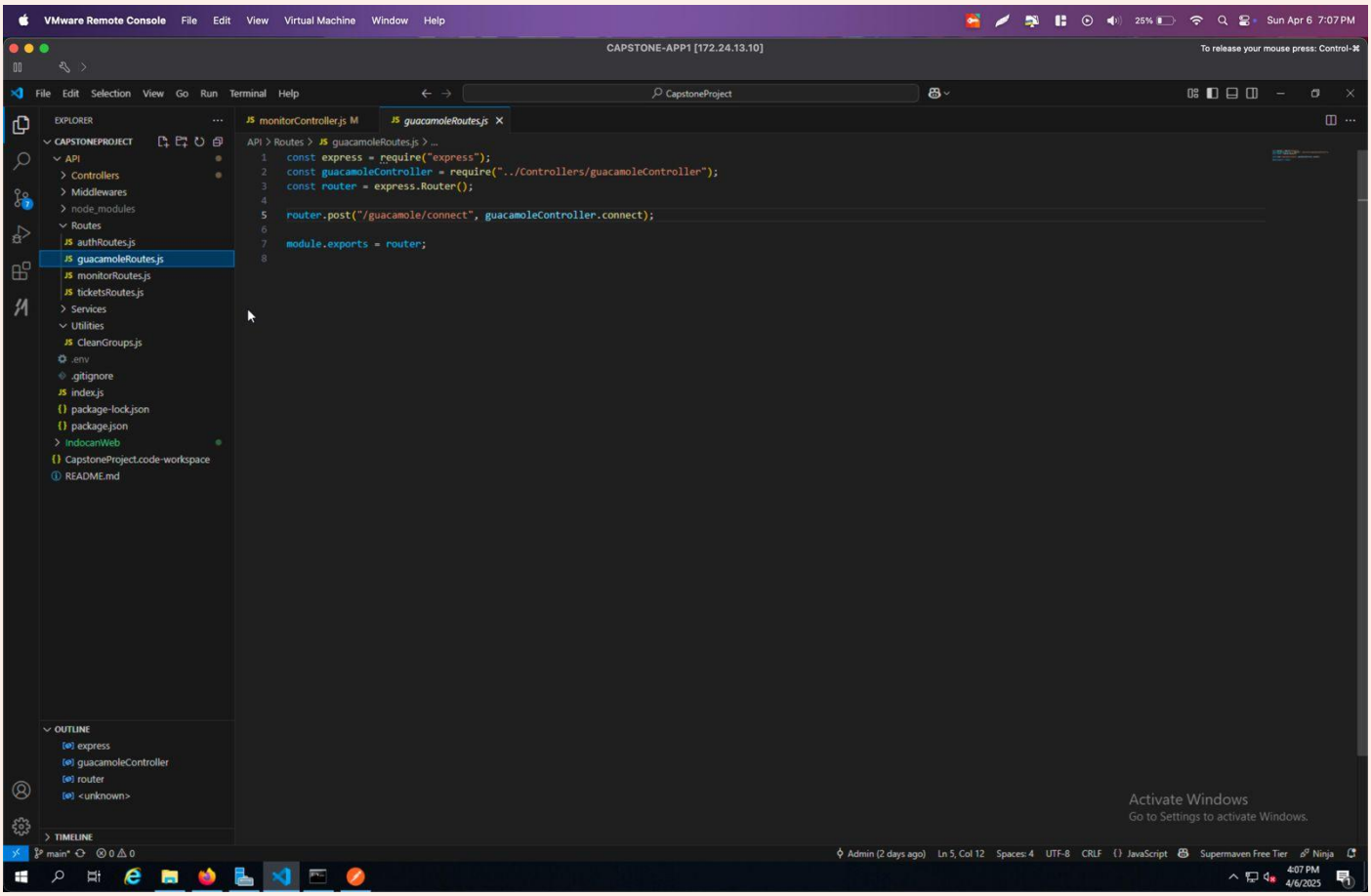

VMware Remote Console File Edit View Virtual Machine Window Help
CAPSTONE-APP1 [172.24.13.10]
File Edit Selection View Go Run Terminal Help
CapstoneProject
monitorController.js M
VerifyToken.js
const jwt = require('jsonwebtoken');
function verifyToken(req, res, next) {
const token = req.headers['authorization'];
if (!token) {
return res.status(403).json({ message: "No token provided" });
}
jwt.verify(token, process.env.JWT_SECRET, (err, decoded) => {
if (err) {
return res.status(401).json({ message: "Unauthorized" });
}
// Attach the decoded user information to the request object
req.userId = decoded.userId;
req.username = decoded.username;
next();
});
}
module.exports = verifyToken;
Activate Windows
Go to Settings to activate Windows.

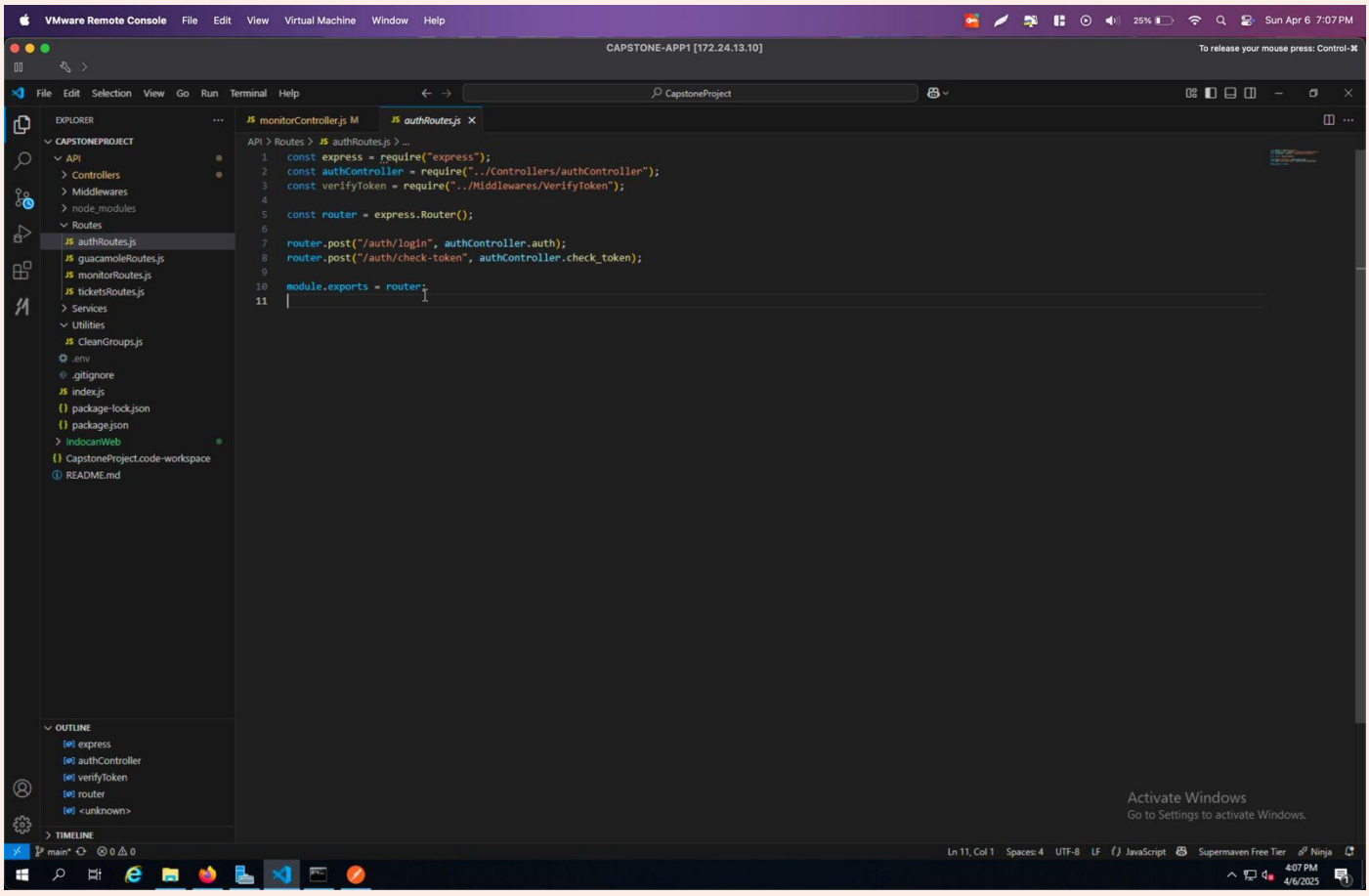

VMware Remote Console File Edit View Virtual Machine Window Help
CAPSTONE-APP1 [172.24.13.10]
To release your mouse press: Control-⌘
File Edit Selection View Go Run Terminal Help
CapstoneProject
EXPLORER
CAPSTONEPROJECT
API
Controllers
Middlewares
node_modules
Routes
authRoutes.js
guacamoleRoutes.js
monitorRoutes.js
ticketsRoutes.js
Services
monitor.js
Utilities
.env
.gitignore
index.js
package-lock.json
package.json
IndocanWeb
CapstoneProject.code-workspace
README.md
monitorController.js M
monitor.js
API > Services > monitor.js > <unknown> > exports > getFQDN
const cron = require("node-cron");
const networkScanner = require("network-scanner-js");
const dns = require("dns").promises; // Use promises-based DNS functions
module.exports = async (io) => {
const netScan = new networkScanner();
let monitoringData = []; // Persistent storage for monitoring data
// Mapping of IPs to Guacamole connection IDs (Replace with actual values)
const guacConnections = {
"172.24.13.4": "MQBjAHBvc3RncmVzcwBw",
// Add more mappings as needed
};
const GUACAMOLE_URL = "http://guacamole.indocan.local:8080/guacamole/#/client/";
// Function to get FQDN (Reverse DNS Lookup)
async function getFQDN(ip) {
try {
const hostnames = await dns.reverse(ip);
return hostnames.length ? hostnames[0] : null;
} catch (error) {
return null; // Return null if no hostname is found
}
}
// Function to extract SERVER_NAME from FQDN (only before the first dot)
function extractServerName(fqdn) {
if (!fqdn) return "Unknown"; // If no FQDN, return "Unknown"
// Check if the FQDN contains a dot, if not, return the full FQDN
return fqdn.includes(".") ? fqdn.split(".")[0] : fqdn;
}
// Function to update monitoring data
async function updateMonitoringData() {
const result = await new Promise(async (resolve, reject) => {
try {
let tempMonitoringData = [];
await netScan.ipScan("172.24.13.4-100", async (host) => {
const fqdn = await getFQDN(host.ip_address); // Get the FQDN
const serverName = extractServerName(fqdn); // Extract the SERVER_NAME
const connectionId = guacConnections[host.ip_address] || null;
const guacLink = connectionId ? `${GUACAMOLE_URL}${connectionId}` : null;
tempMonitoringData.push({
serverName: serverName, // Store the SERVER_NAME
fqdn: fqdn || "Unknown", // Store the resolved FQDN
OUTLINE
cron
networkScanner
dns
<unknown>
exports
TIMELINE
Activate Windows
Go to Settings to activate Windows.
main*
Admin (2 days ago) Ln 18, Col 27 Spaces: 4 UTF-8 CRLF JavaScript Supermaven Free Tier Ninja
4:06 PM
4/6/2025


VMware Remote Console File Edit View Virtual Machine Window Help
CAPSTONE-APP1 [172.24.13.10]
To release your mouse press: Control-⌘
File Edit Selection View Go Run Terminal Help
CapstoneProject
EXPLORER
CAPSTONEPROJECT
API
Controllers
Middlewares
node_modules
Routes
authRoutes.js
guacamoleRoutes.js
monitorRoutes.js
ticketsRoutes.js
Services
monitor.js
Utilities
.env
.gitignore
index.js
package-lock.json
package.json
IndocanWeb
CapstoneProject.code-workspace
README.md
monitorController.js M
monitor.js
API > Services > monitor.js > <unknown> > exports > getFQDN
module.exports = async (io) => {
async function updateMonitoringData() {
const result = await new Promise(async (resolve, reject) => {
await netScan.ipScan("172.24.13.4-100", async (host) => {
const serverName = extractServerName(fqdn); // Extract the SERVER_NAME
const connectionId = guacConnections[host.ip_address] || null;
const guacLink = connectionId ? `${GUACAMOLE_URL}${connectionId}` : null;
tempMonitoringData.push({
serverName: serverName, // Store the SERVER_NAME
fqdn: fqdn || "Unknown", // Store the resolved FQDN
host: host.host,
ip: host.ip_address,
status: host.status,
packet_loss: host.packet_loss,
timestamp: new Date(),
link: guacLink
});
});
// Ensure all async tasks complete before resolving
setTimeout(() => {
resolve(tempMonitoringData);
}, 1000);
} catch (error) {
reject(error);
}
});
// Update monitoringData based on result
const updatedMonitoringData = monitoringData.map((existing) => {
const found = result.find((newData) => newData.ip === existing.ip);
return found ? found : { ...existing, status: "offline", timestamp: new Date() };
});
// Add new entries that were not in the previous scan
result.forEach((newEntry) => {
if (!updatedMonitoringData.find((entry) => entry.ip === newEntry.ip)) {
updatedMonitoringData.push(newEntry);
}
});
monitoringData = updatedMonitoringData; // Update persistent storage
io.emit("monitoringUpdate", monitoringData); // Emit updated data
}
OUTLINE
cron
networkScanner
dns
<unknown>
exports
TIMELINE
Activate Windows
Go to Settings to activate Windows.
main*
Admin (2 days ago) Ln 18, Col 27 Spaces: 4 UTF-8 CRLF JavaScript Supermaven Free Tier Ninja
4:07 PM
4/6/2025

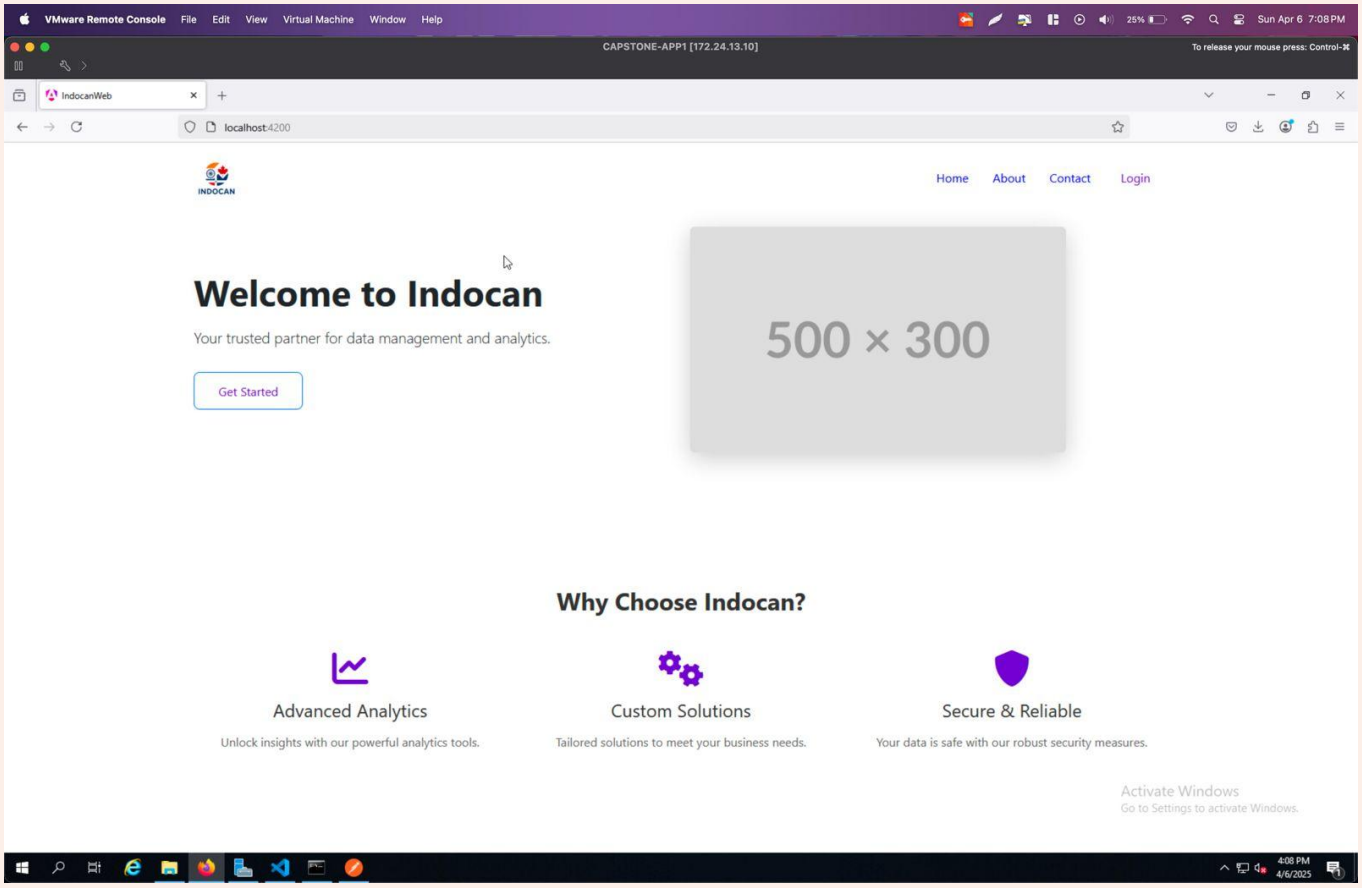

CAPSTONE-APP1 [172.24.13.10]
monitorController.js M
CleanGroups.js
function cleanGroups(memberOf) {
if (!memberOf || !Array.isArray(memberOf)) {
return []; // Return an empty array if memberOf is not provided or is not an array
}
// Extract the first CN from each entry in the memberOf array
return memberOf.map((group) => {
// Use a regular expression to extract the first CN value
const match = group.match(/CN=([^,]+)/);
return match ? match[1] : null; // Return the extracted group name or null if no match
}).filter((group) => group !== null); // Filter out any null values
}
module.exports = cleanGroups;
Activate Windows
Go to Settings to activate Windows.

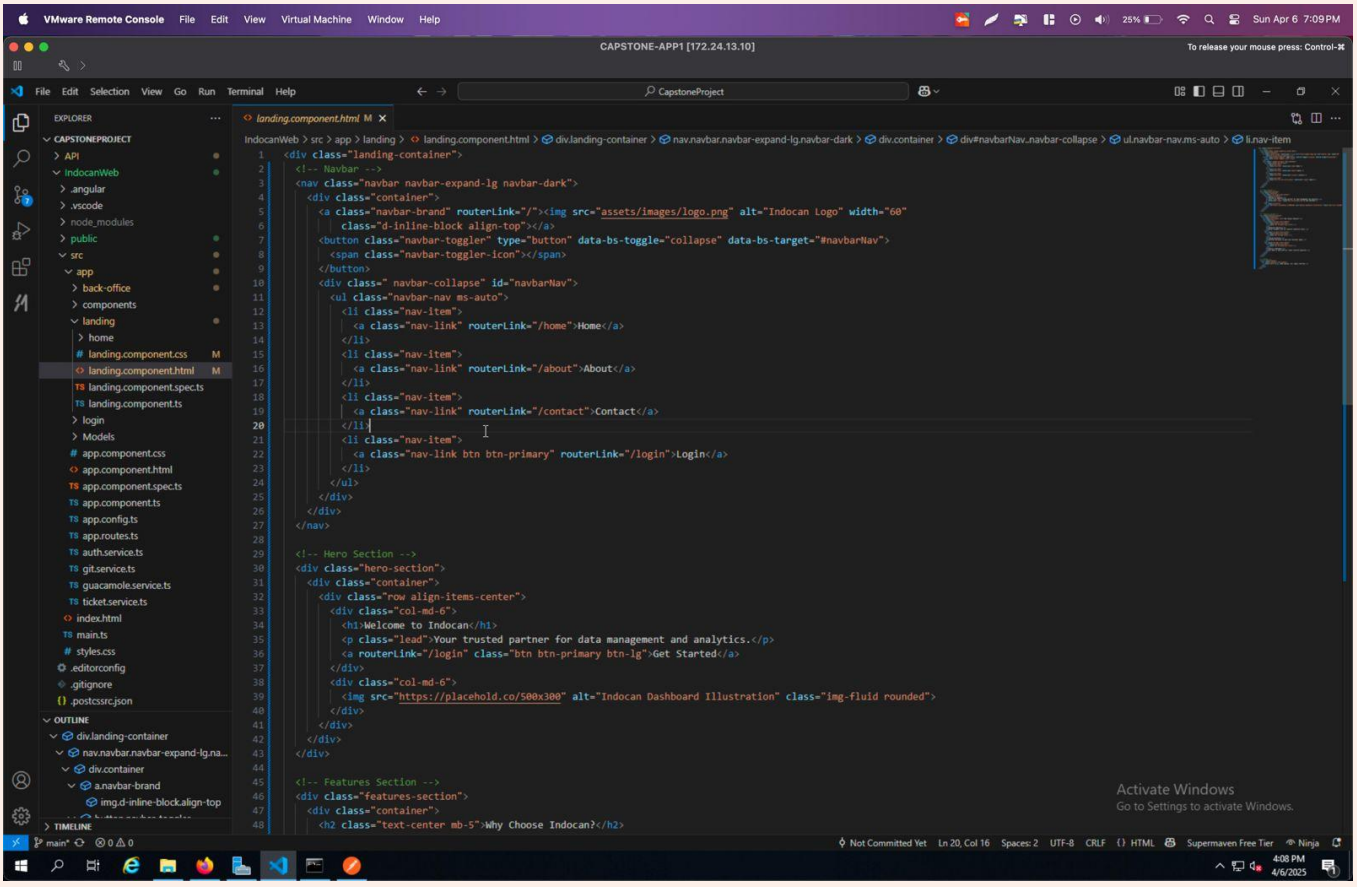

CAPSTONE-APP1 [172.24.13.10]
monitorController.js M
ticketsRoutes.js
const express = require('express');
const router = express.Router();
const ticketsController = require('../Controllers/ticketsController');
// Get user-specific tickets
router.post('/ticketing/tickets', ticketsController.getUserTickets);
router.post('/ticketing/articles', ticketsController.getTicketArticles);
router.post('/ticketing/new', ticketsController.createNewTicket);
module.exports = router;
Activate Windows
Go to Settings to activate Windows.


VMware Remote Console File Edit View Virtual Machine Window Help
CAPSTONE-APP1 [172.24.13.10]
To release your mouse press: Control-⌘
File Edit Selection View Go Run Terminal Help
CapstoneProject
EXPLORER
CAPSTONEPROJECT
API
Controllers
Middlewares
node_modules
Routes
authRoutes.js
guacamoleRoutes.js
monitorRoutes.js
ticketsRoutes.js
Services
Utilities
CleanGroups.js
.env
.gitignore
index.js
package-lock.json
package.json
IndocanWeb
CapstoneProject.code-workspace
README.md
monitorController.js M
guacamoleRoutes.js
API > Routes > guacamoleRoutes.js > ...
1 const express = require("express");
2 const guacamoleController = require("../Controllers/guacamoleController");
3 const router = express.Router();
4
5 router.post("/guacamole/connect", guacamoleController.connect);
6
7 module.exports = router;
8
OUTLINE
express
guacamoleController
router
<unknown>
TIMELINE
Activate Windows
Go to Settings to activate Windows.
Admin (2 days ago) Ln 5, Col 12 Spaces: 4 UTF-8 CRLF JavaScript Supermaven Free Tier Ninja
4:07 PM
4/6/2025

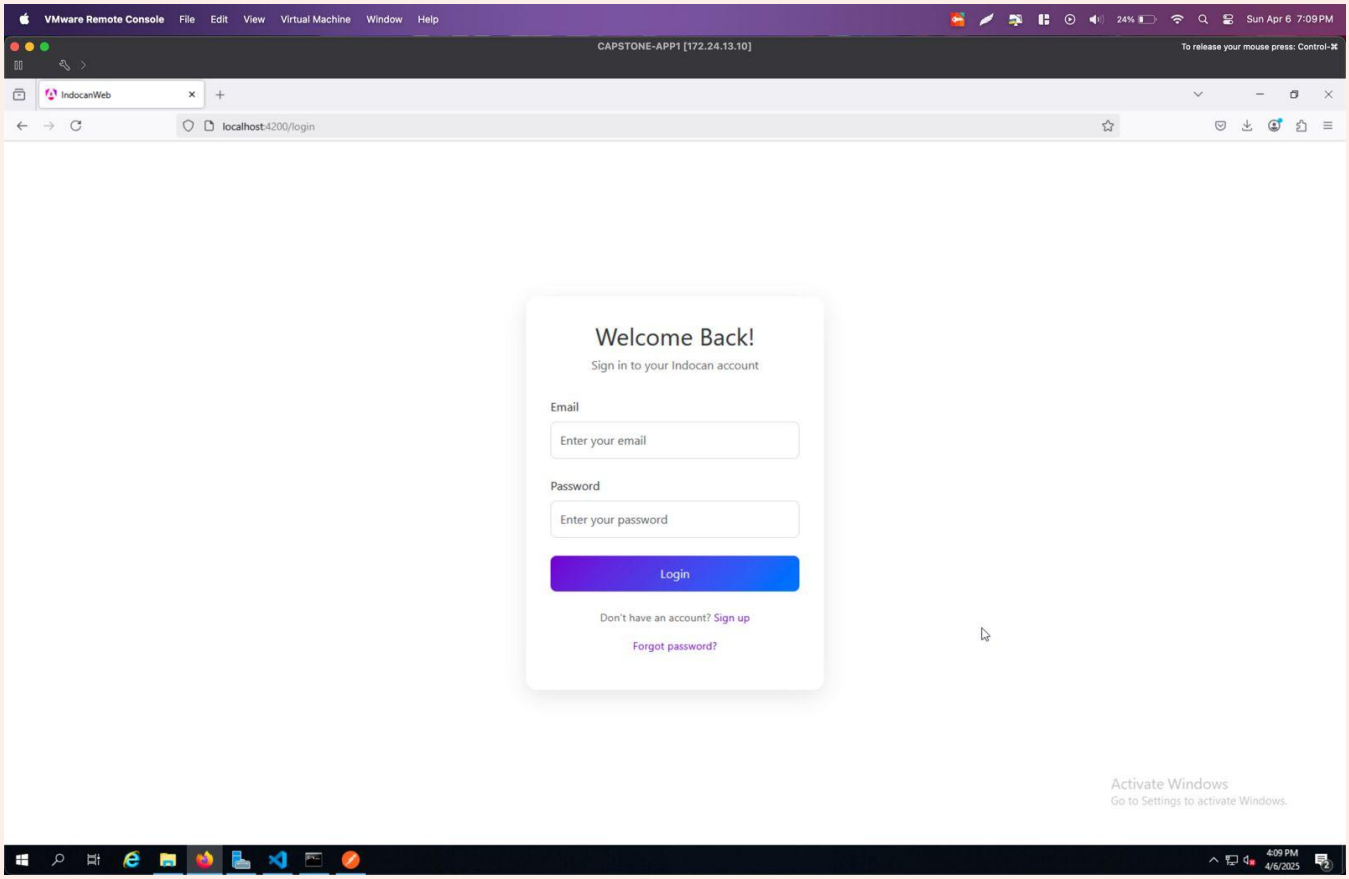

VMware Remote Console File Edit View Virtual Machine Window Help
CAPSTONE-APP1 [172.24.13.10]
To release your mouse press: Control-⌘
File Edit Selection View Go Run Terminal Help
CapstoneProject
EXPLORER
CAPSTONEPROJECT
API
Controllers
Middlewares
node_modules
Routes
authRoutes.js
guacamoleRoutes.js
monitorRoutes.js
ticketsRoutes.js
Services
Utilities
CleanGroups.js
.env
.gitignore
index.js
package-lock.json
package.json
IndocanWeb
CapstoneProject.code-workspace
README.md
monitorController.js M
authRoutes.js
API > Routes > authRoutes.js > ...
1 const express = require("express");
2 const authController = require("../Controllers/authController");
3 const verifyToken = require("../Middlewares/VerifyToken");
4
5 const router = express.Router();
6
7 router.post("/auth/login", authController.auth);
8 router.post("/auth/check-token", authController.check_token);
9
10 module.exports = router;
11
OUTLINE
express
authController
verifyToken
router
<unknown>
TIMELINE
Activate Windows
Go to Settings to activate Windows.
Ln 11, Col 1 Spaces: 4 UTF-8 LF JavaScript Supermaven Free Tier Ninja
4:07 PM
4/6/2025

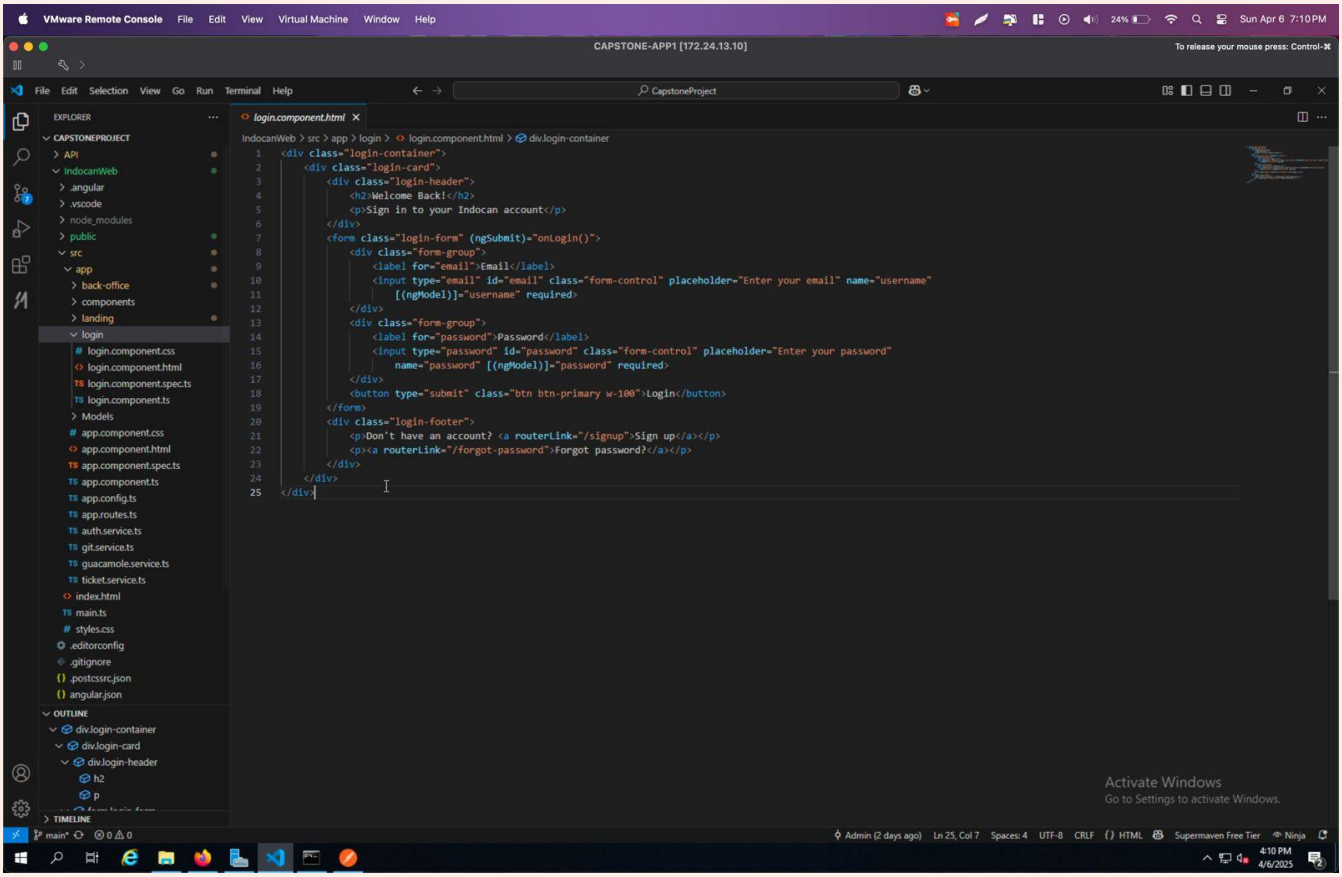
LANDING PAGE





LOGIN


Welcome Back!
Sign in to your Indocan account
Email
Enter your email
Password
Enter your password
Login
Don't have an account? Sign up
Forgot password?


login.component.html

<h2>Welcome Back!</h2>
Sign in to your Indocan account

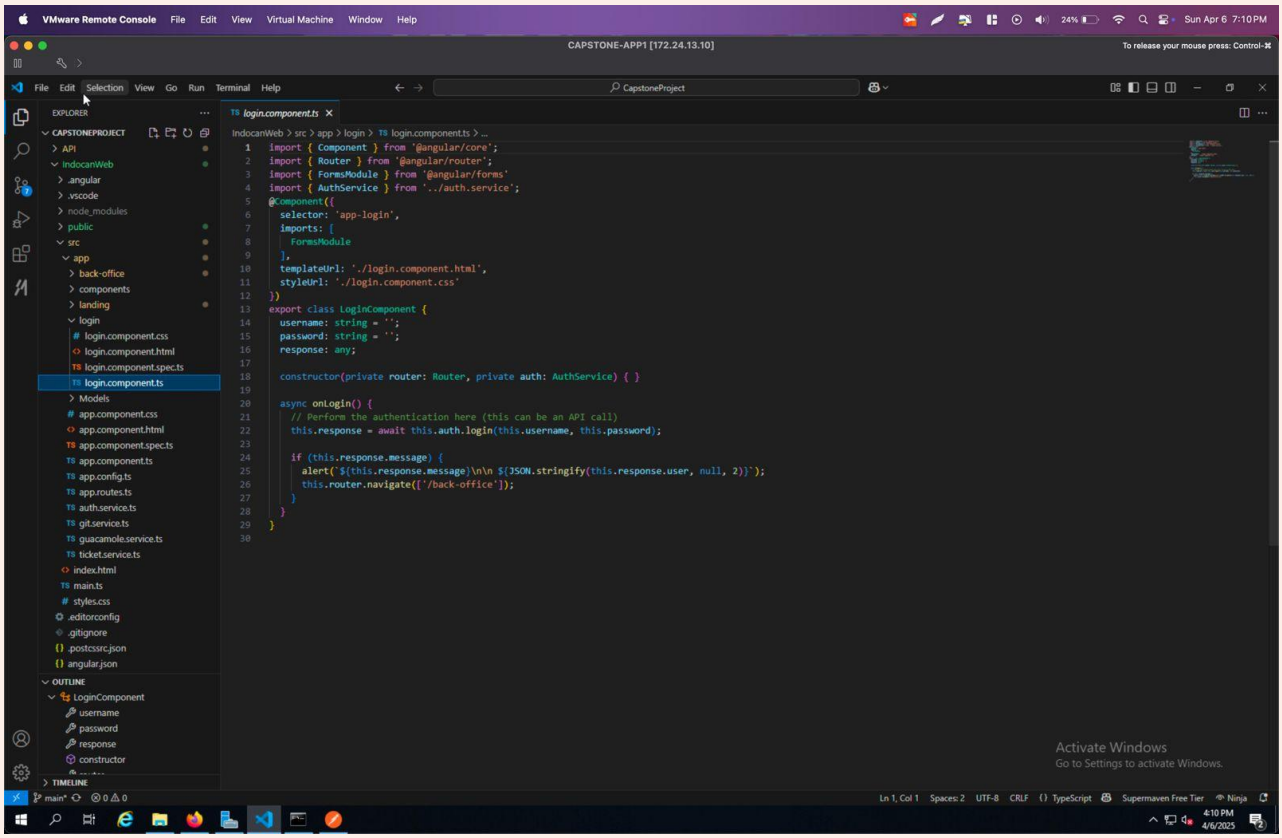
<form class="login-form" (ngSubmit)="onLogin()">

<label for="email">Email</label>
<input type="email" id="email" class="form-control" placeholder="Enter your email" name="username"
[(ngModel)]="username" required>

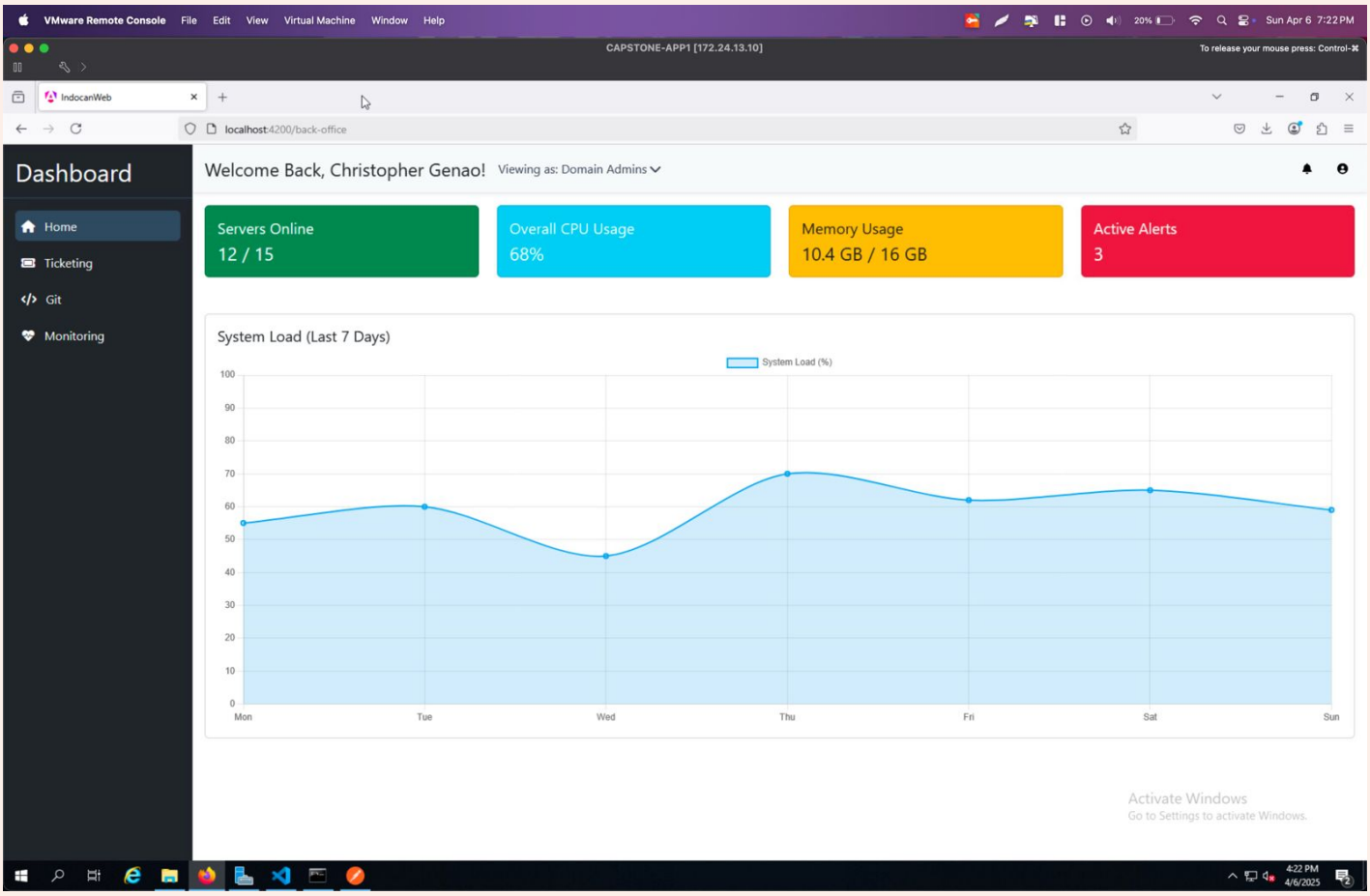
<label for="password">Password</label>
<input type="password" id="password" class="form-control" placeholder="Enter your password"
name="password" [(ngModel)]="password" required>

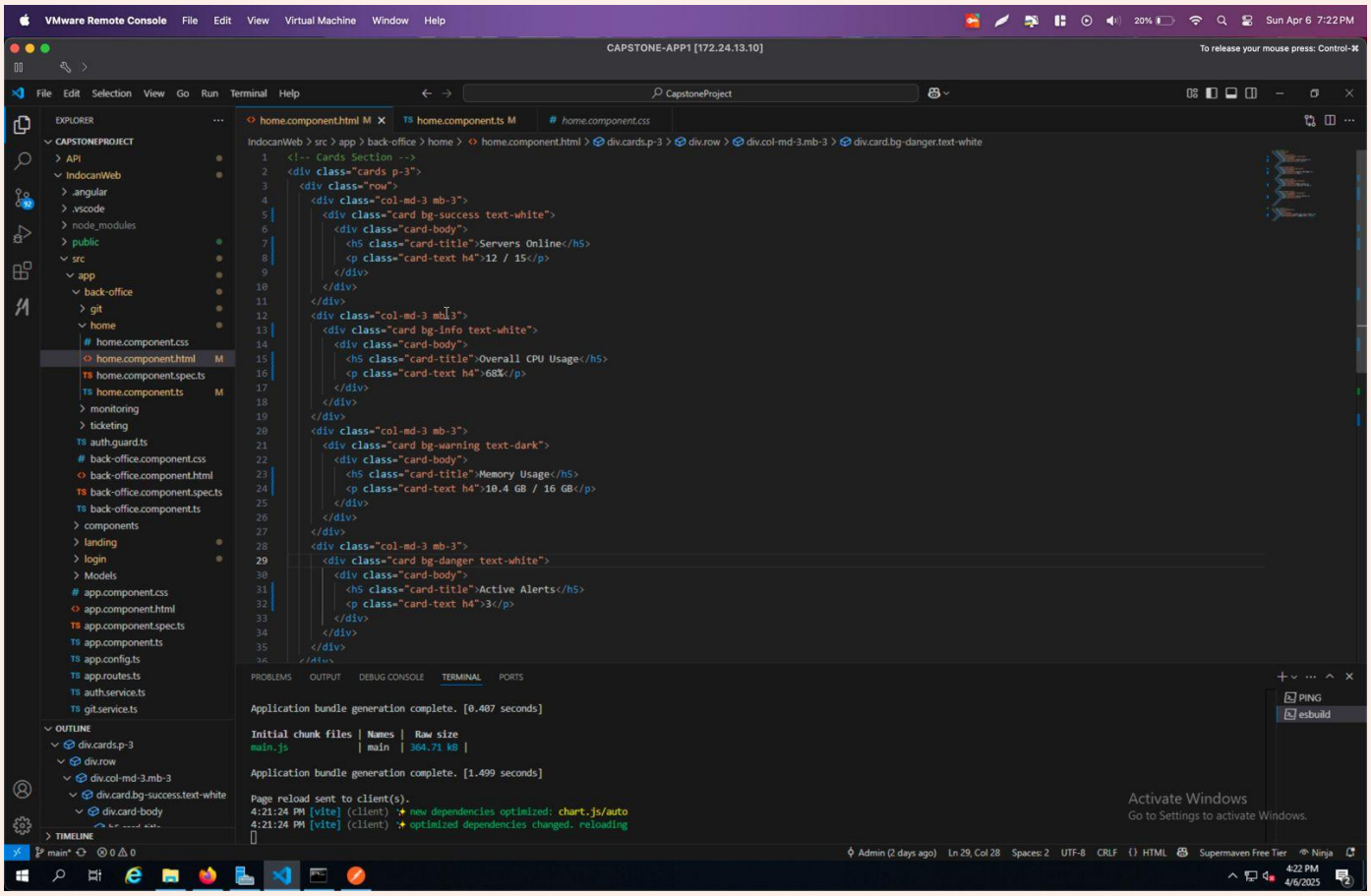
<button type="submit" class="btn btn-primary w-100">Login</button>
</form>

Don't have an account? <a routerLink="/signup">Sign up
<a routerLink="/forgot-password">Forgot password?


CAPSTONE-APP1 [172.24.13.10]
login.component.ts
import { Component } from '@angular/core';
import { Router } from '@angular/router';
import { FormsModule } from '@angular/forms'
import { AuthService } from '../auth.service';
@Component({
selector: 'app-login',
imports: [
FormsModule
],
templateUrl: './login.component.html',
styleUrl: './login.component.css'
})
export class LoginComponent {
username: string = '';
password: string = '';
response: any;
constructor(private router: Router, private auth: AuthService) { }
async onLogin() {
// Perform the authentication here (this can be an API call)
this.response = await this.auth.login(this.username, this.password);
if (this.response.message) {
alert(`${this.response.message}\n\n ${JSON.stringify(this.response.user, null, 2)}`);
this.router.navigate(['/back-office']);
}
}
}
Activate Windows
Go to Settings to activate Windows.

HOME


CAPSTONE-APP1 [172.24.13.10]
IndocanWeb
localhost:4200/back-office
Dashboard
Welcome Back, Christopher Genao!
Viewing as: Domain Admins
Home
Ticketing
Git
Monitoring
Servers Online
12 / 15
Overall CPU Usage
68%
Memory Usage
10.4 GB / 16 GB
Active Alerts
3
System Load (Last 7 Days)
System Load (%)
100
90
80
70
60
50
40
30
20
10
0
Mon
Tue
Wed
Thu
Fri
Sat
Sun
Activate Windows
Go to Settings to activate Windows.

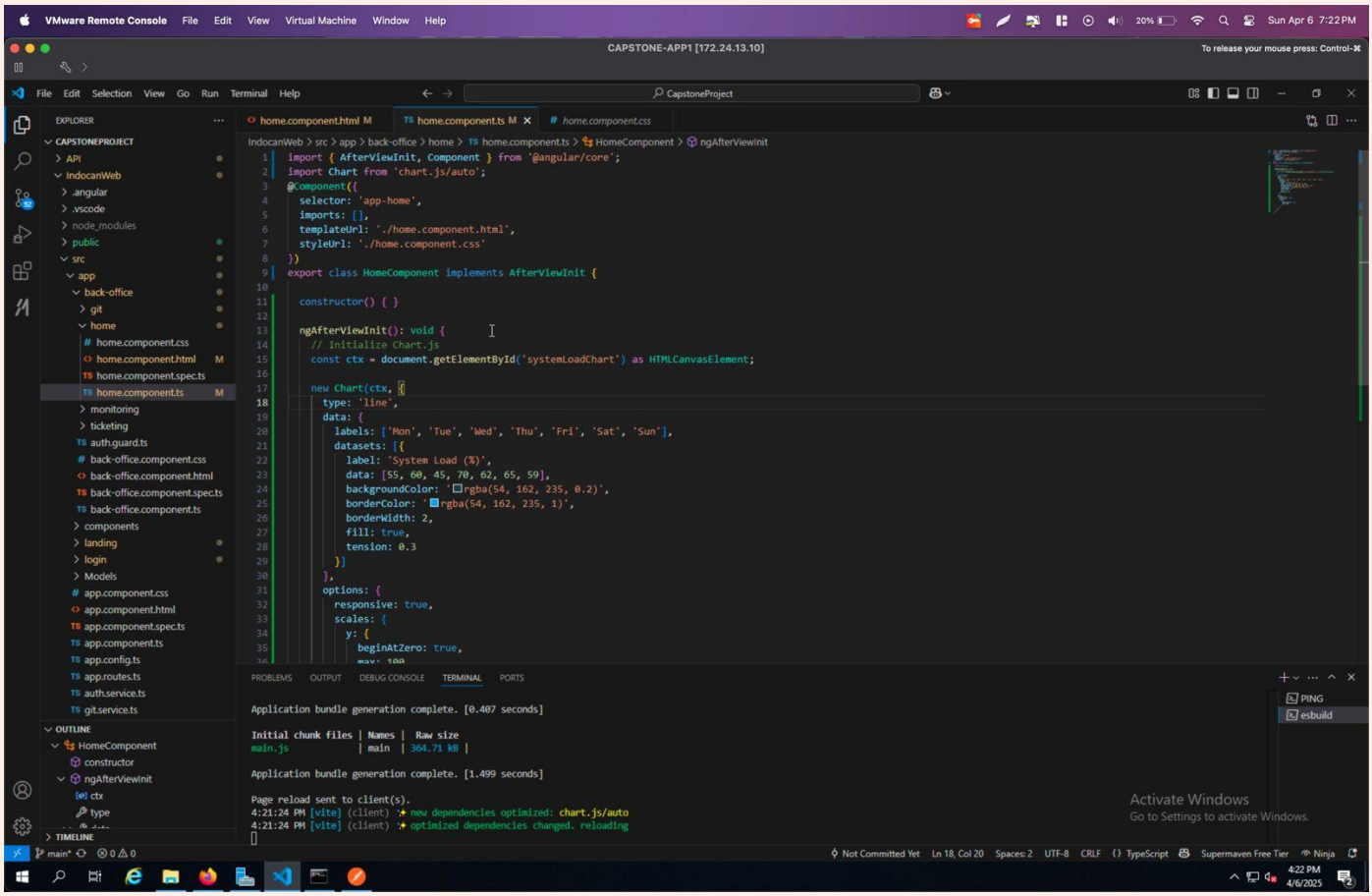

VMware Remote Console File Edit View Virtual Machine Window Help
CAPSTONE-APP1 [172.24.13.10]
home.component.html M
home.component.ts M
home.component.css
<!-- Cards Section -->

<h5 class="card-title">Servers Online</h5>
12 / 15

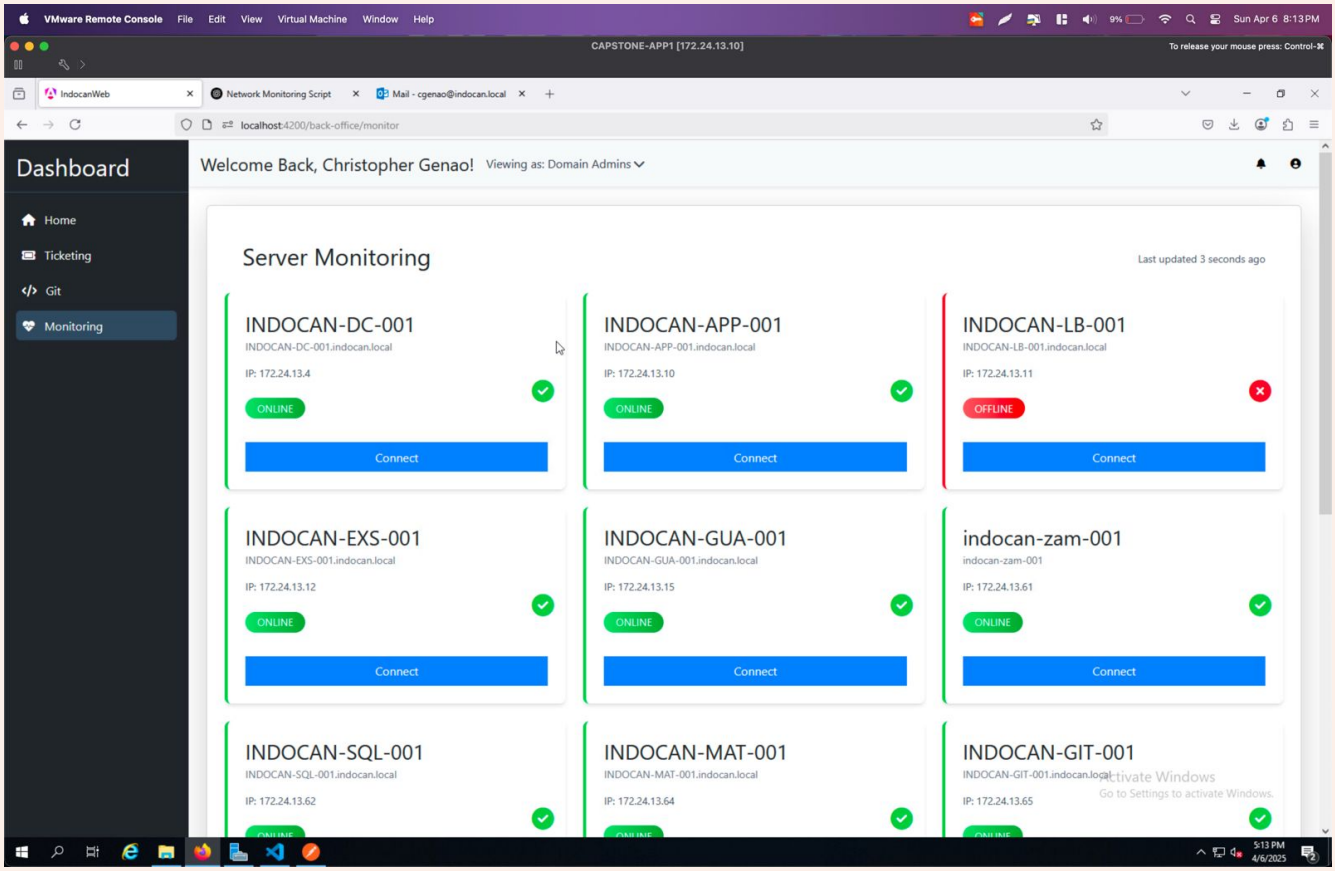
<h5 class="card-title">Overall CPU Usage</h5>
68%

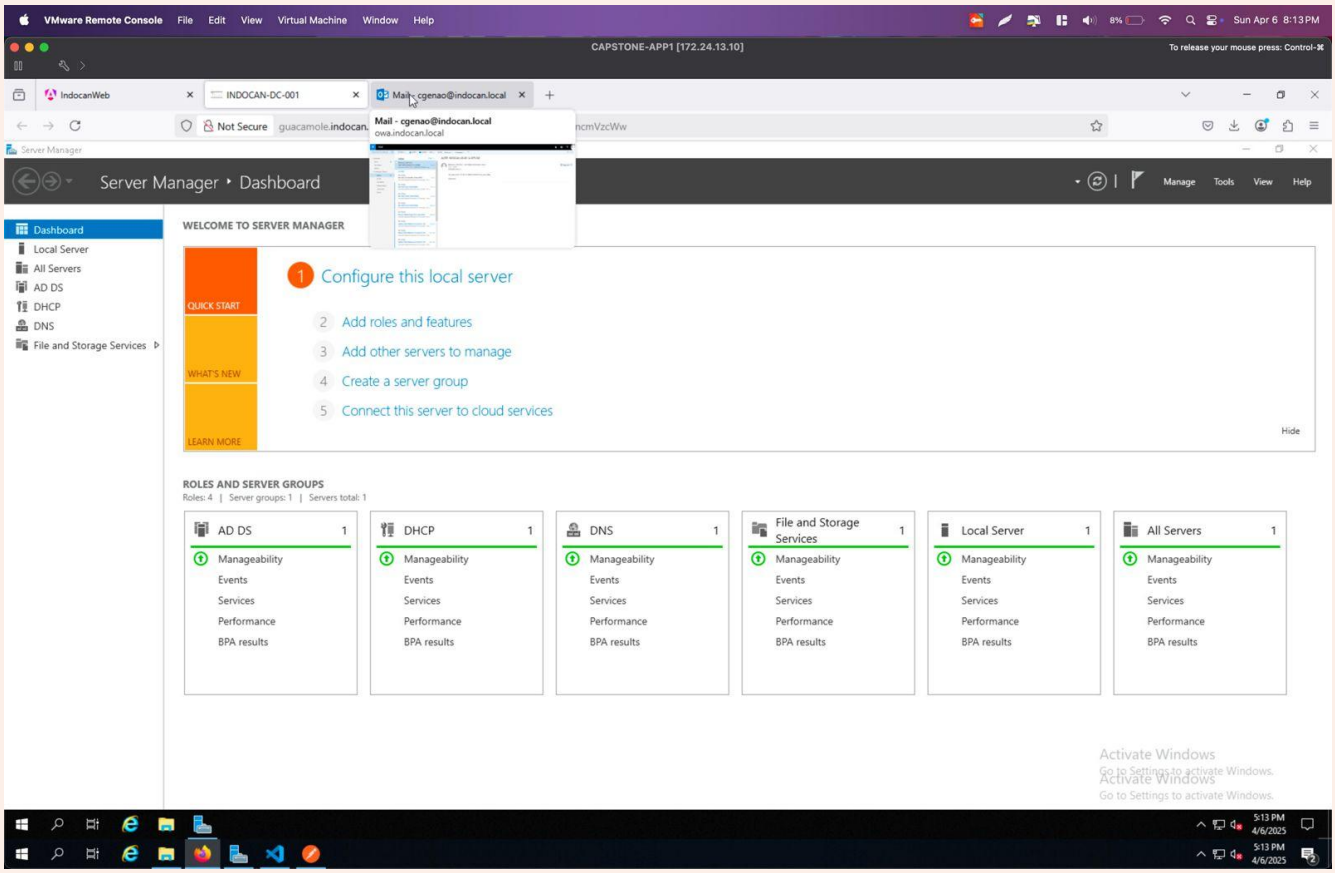
<h5 class="card-title">Memory Usage</h5>
10.4 GB / 16 GB

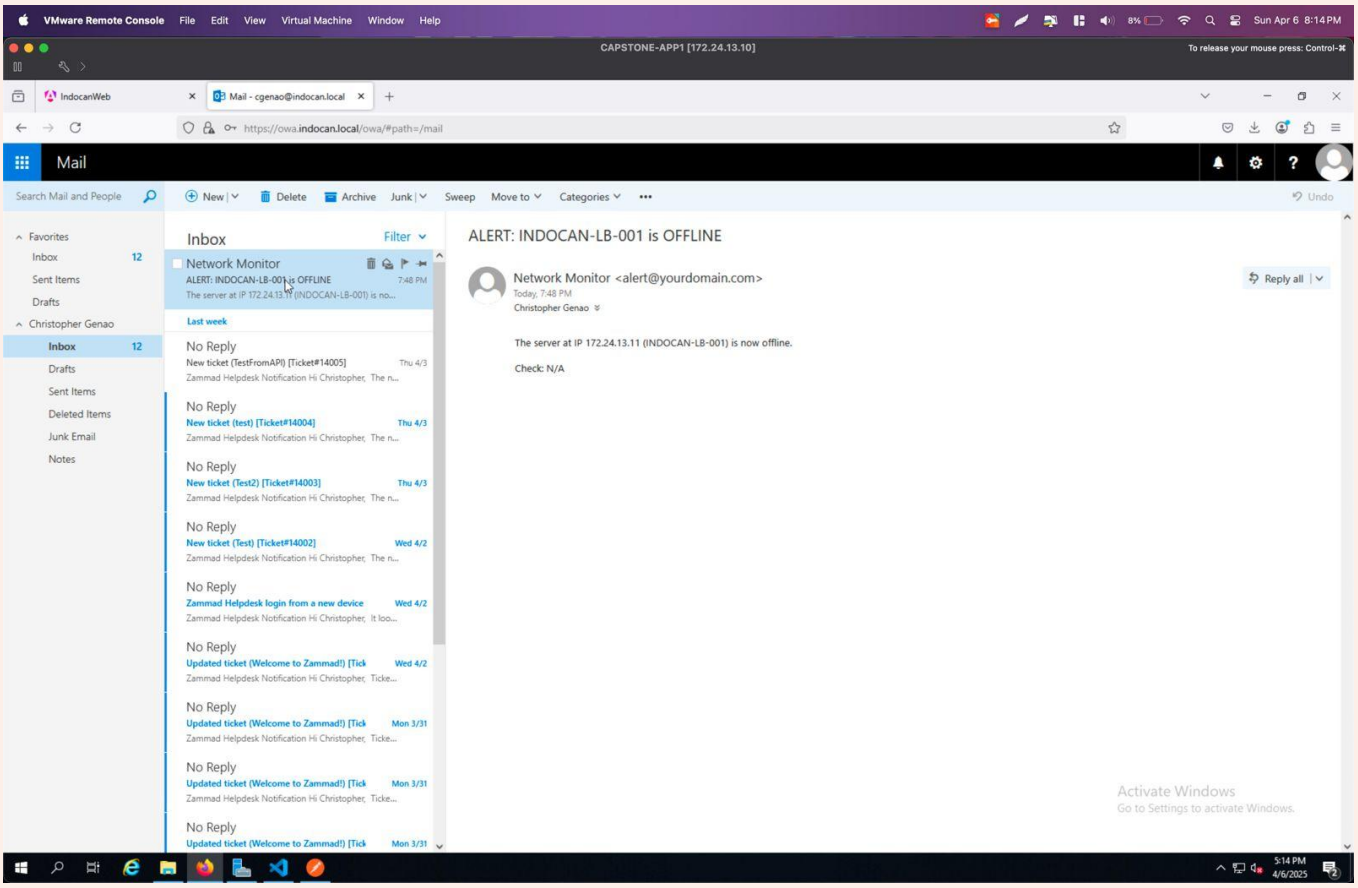
<h5 class="card-title">Active Alerts</h5>
3
Application bundle generation complete. [0.407 seconds]
Initial chunk files | Names | Raw size
main.js | main | 364.71 kB |
Application bundle generation complete. [1.499 seconds]
Page reload sent to client(s).
4:21:24 PM [vite] (client) new dependencies optimized: chart.js/auto
4:21:24 PM [vite] (client) optimized dependencies changed. reloading
Activate Windows
Go to Settings to activate Windows.

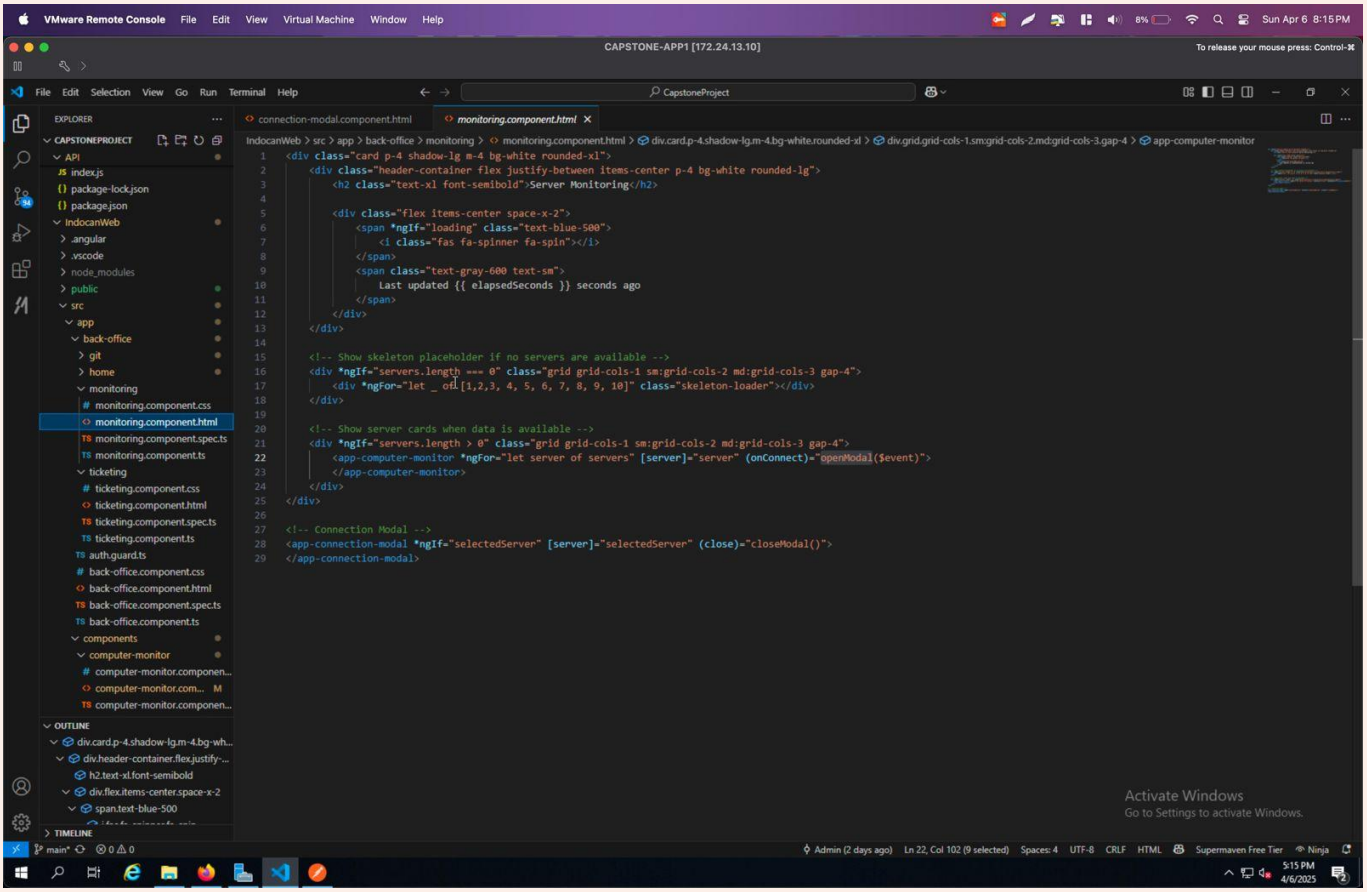

VMware Remote Console File Edit View Virtual Machine Window Help
CAPSTONE-APP1 [172.24.13.10]
home.component.html M
home.component.ts M
home.component.css
import { AfterViewInit, Component } from '@angular/core';
import Chart from 'chart.js/auto';
@Component({
selector: 'app-home',
imports: [],
templateUrl: './home.component.html',
styleUrl: './home.component.css'
})
export class HomeComponent implements AfterViewInit {
constructor() { }
ngAfterViewInit(): void {
// Initialize Chart.js
const ctx = document.getElementById('systemLoadChart') as HTMLCanvasElement;
new Chart(ctx, {
type: 'line',
data: {
labels: ['Mon', 'Tue', 'Wed', 'Thu', 'Fri', 'Sat', 'Sun'],
datasets: [{
label: 'System Load (%)',
data: [55, 60, 45, 70, 62, 65, 59],
backgroundColor: 'rgba(54, 162, 235, 0.2)',
borderColor: 'rgba(54, 162, 235, 1)',
borderWidth: 2,
fill: true,
tension: 0.3
}]
},
options: {
responsive: true,
scales: {
y: {
beginAtZero: true,
Application bundle generation complete. [0.407 seconds]
Initial chunk files | Names | Raw size
main.js | main | 364.71 kB |
Application bundle generation complete. [1.499 seconds]
Page reload sent to client(s).
4:21:24 PM [vite] (client) new dependencies optimized: chart.js/auto
4:21:24 PM [vite] (client) optimized dependencies changed. reloading
Activate Windows
Go to Settings to activate Windows.

MONITORING


Dashboard
Welcome Back, Christopher Genao!
Viewing as: Domain Admins
Home
Ticketing
Git
Monitoring
Server Monitoring
Last updated 3 seconds ago
INDOCAN-DC-001
INDOCAN-DC-001.indocan.local
IP: 172.24.13.4
ONLINE
Connect
INDOCAN-APP-001
INDOCAN-APP-001.indocan.local
IP: 172.24.13.10
ONLINE
Connect
INDOCAN-LB-001
INDOCAN-LB-001.indocan.local
IP: 172.24.13.11
OFFLINE
Connect
INDOCAN-EXS-001
INDOCAN-EXS-001.indocan.local
IP: 172.24.13.12
ONLINE
Connect
INDOCAN-GUA-001
INDOCAN-GUA-001.indocan.local
IP: 172.24.13.15
ONLINE
Connect
indocan-zam-001
indocan-zam-001
IP: 172.24.13.61
ONLINE
Connect
INDOCAN-SQL-001
INDOCAN-SQL-001.indocan.local
IP: 172.24.13.62
INDOCAN-MAT-001
INDOCAN-MAT-001.indocan.local
IP: 172.24.13.64
INDOCAN-GIT-001
INDOCAN-GIT-001.indocan.local
IP: 172.24.13.65


Server Manager • Dashboard
Dashboard
Local Server
All Servers
AD DS
DHCP
DNS
File and Storage Services
WELCOME TO SERVER MANAGER
1 Configure this local server
2 Add roles and features
3 Add other servers to manage
4 Create a server group
5 Connect this server to cloud services
ROLES AND SERVER GROUPS
AD DS
DHCP
DNS
File and Storage Services
Local Server
All Servers
Manageability
Events
Services
Performance
BPA results


Mail - cgenao@indocan.local
https://owa.indocan.local/owa/#path=/mail
Mail
Search Mail and People
New
Delete
Archive
Junk
Sweep
Move to
Categories
Undo
Favorites
Inbox 12
Sent Items
Drafts
Christopher Genao
Inbox 12
Drafts
Sent Items
Deleted Items
Junk Email
Notes
Inbox
Filter
Network Monitor
ALERT: INDOCAN-LB-001 is OFFLINE
The server at IP 172.24.13.11 (INDOCAN-LB-001) is no...
7:48 PM
Last week
No Reply
New ticket (TestFromAPI) [Ticket#14005]
Thu 4/3
Zammad Helpdesk Notification Hi Christopher, The n...
No Reply
New ticket (test) [Ticket#14004]
Thu 4/3
Zammad Helpdesk Notification Hi Christopher, The n...
No Reply
New ticket (Test2) [Ticket#14003]
Thu 4/3
Zammad Helpdesk Notification Hi Christopher, The n...
No Reply
New ticket (Test) [Ticket#14002]
Wed 4/2
Zammad Helpdesk Notification Hi Christopher, The n...
No Reply
Zammad Helpdesk login from a new device
Wed 4/2
Zammad Helpdesk Notification Hi Christopher, It loo...
No Reply
Updated ticket (Welcome to Zammad!) [Tick
Wed 4/2
Zammad Helpdesk Notification Hi Christopher, Ticke...
No Reply
Updated ticket (Welcome to Zammad!) [Tick
Mon 3/31
Zammad Helpdesk Notification Hi Christopher, Ticke...
No Reply
Updated ticket (Welcome to Zammad!) [Tick
Mon 3/31
Zammad Helpdesk Notification Hi Christopher, Ticke...
No Reply
Updated ticket (Welcome to Zammad!) [Tick
Mon 3/31
ALERT: INDOCAN-LB-001 is OFFLINE
Network Monitor <alert@yourdomain.com>
Today, 7:48 PM
Christopher Genao
Reply all
The server at IP 172.24.13.11 (INDOCAN-LB-001) is now offline.
Check: N/A
Activate Windows
Go to Settings to activate Windows.


CAPSTONE-APP1 [172.24.13.10]
connection-modal.component.html
monitoring.component.html
IndocanWeb > src > app > back-office > monitoring > monitoring.component.html > div.card.p-4.shadow-lg.m-4.bg-white.rounded-xl > div.grid.grid-cols-1.sm:grid-cols-2.md:grid-cols-3.gap-4 > app-computer-monitor

<h2 class="text-xl font-semibold">Server Monitoring</h2>

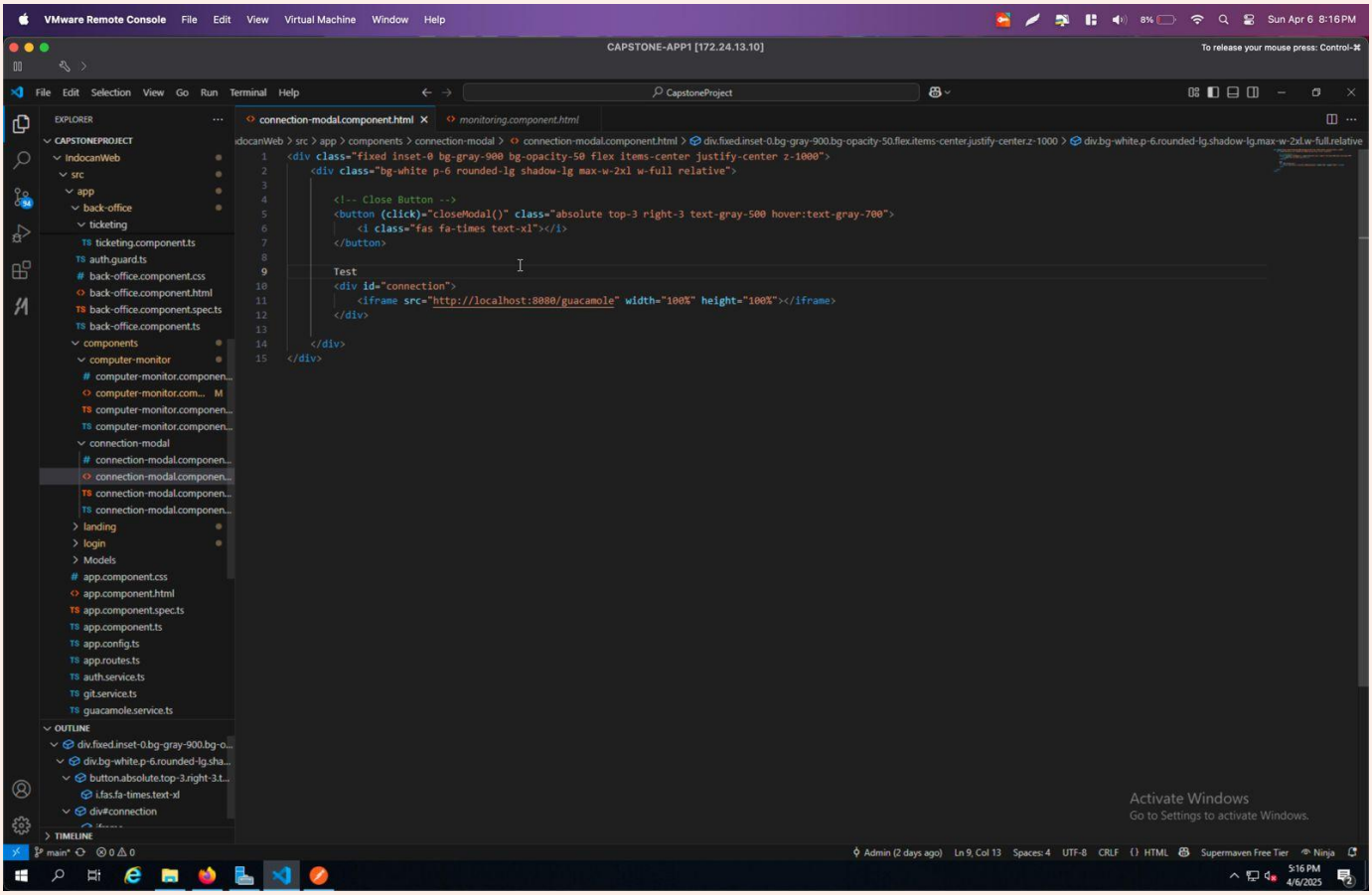
<span *ngIf="loading" class="text-blue-500">

Last updated {{ elapsedSeconds }} seconds ago

<!-- Show skeleton placeholder if no servers are available -->
<div *ngIf="servers.length === 0" class="grid grid-cols-1 sm:grid-cols-2 md:grid-cols-3 gap-4">
<div *ngFor="let _ of [1,2,3, 4, 5, 6, 7, 8, 9, 10]" class="skeleton-loader">

<!-- Show server cards when data is available -->
<div *ngIf="servers.length > 0" class="grid grid-cols-1 sm:grid-cols-2 md:grid-cols-3 gap-4">
<app-computer-monitor *ngFor="let server of servers" [server]="server" (onConnect)="openModal($event)">
</app-computer-monitor>

<!-- Connection Modal -->
<app-connection-modal *ngIf="selectedServer" [server]="selectedServer" (close)="closeModal()">
</app-connection-modal>
Activate Windows
Go to Settings to activate Windows.

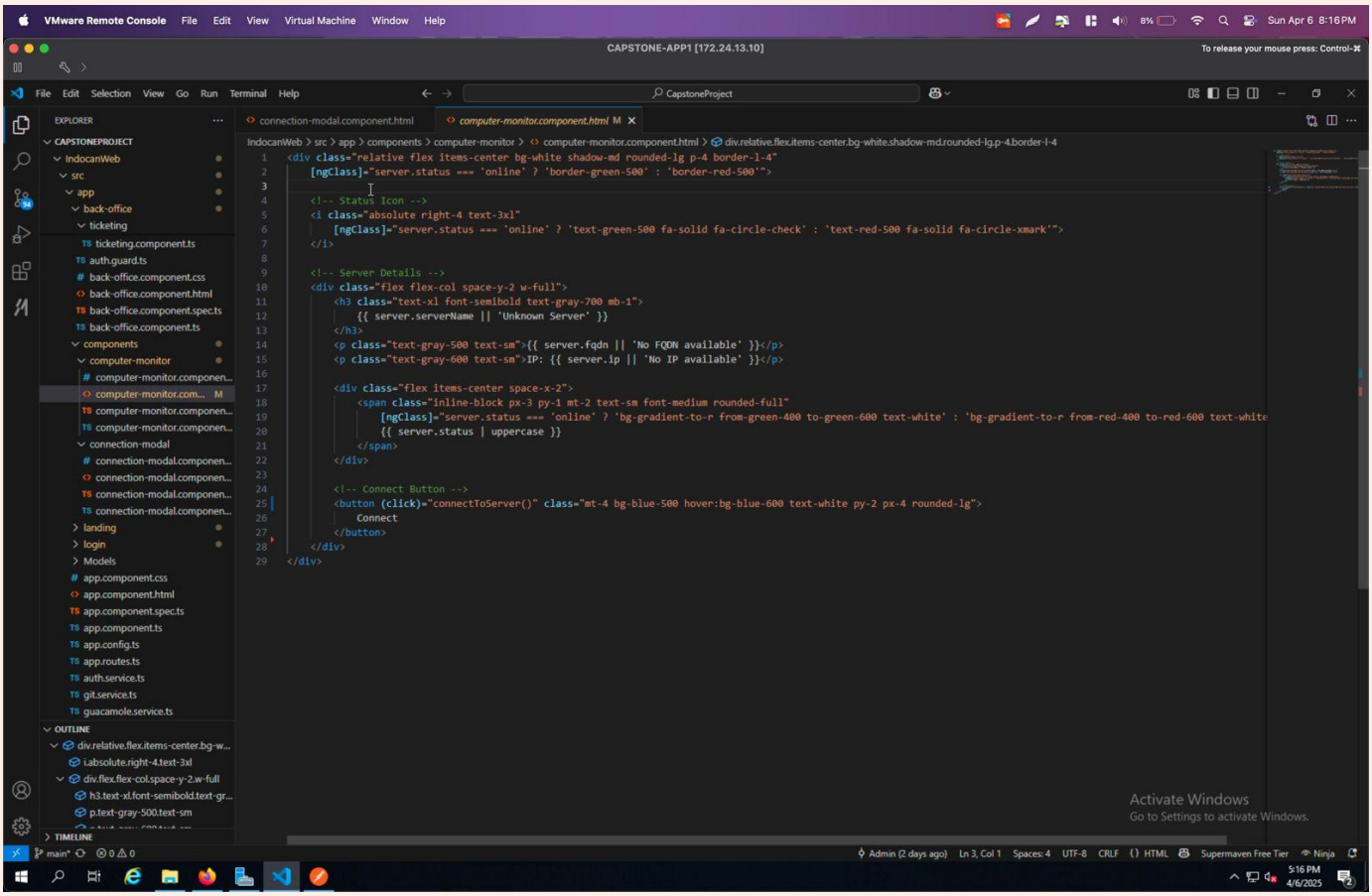

VMware Remote Console File Edit View Virtual Machine Window Help
CAPSTONE-APP1 [172.24.13.10]
connection-modal.component.html
monitoring.component.html

<!-- Close Button -->
<button (click)="closeModal()" class="absolute top-3 right-3 text-gray-500 hover:text-gray-700">

</button>
Test

<iframe src="http://localhost:8080/guacamole" width="100%" height="100%"></iframe>

Activate Windows
Go to Settings to activate Windows.


VMware Remote Console File Edit View Virtual Machine Window Help
CAPSTONE-APP1 [172.24.13.10]
connection-modal.component.html
computer-monitor.component.html M
<div class="relative flex items-center bg-white shadow-md rounded-lg p-4 border-l-4"
[ngClass]="server.status === 'online' ? 'border-green-500' : 'border-red-500'">
<!-- Status Icon -->
<i class="absolute right-4 text-3xl"
[ngClass]="server.status === 'online' ? 'text-green-500 fa-solid fa-circle-check' : 'text-red-500 fa-solid fa-circle-xmark'">

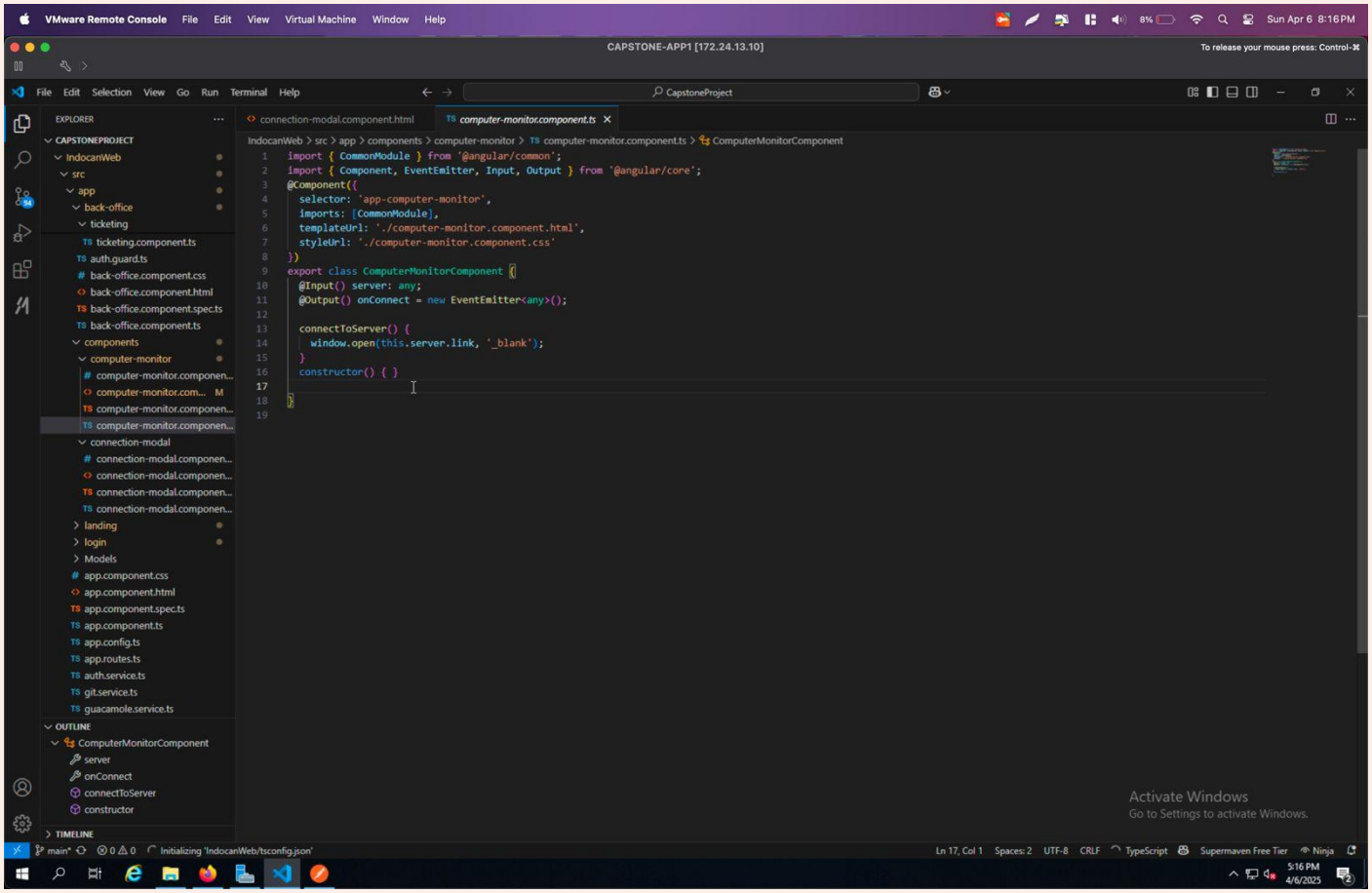
<!-- Server Details -->

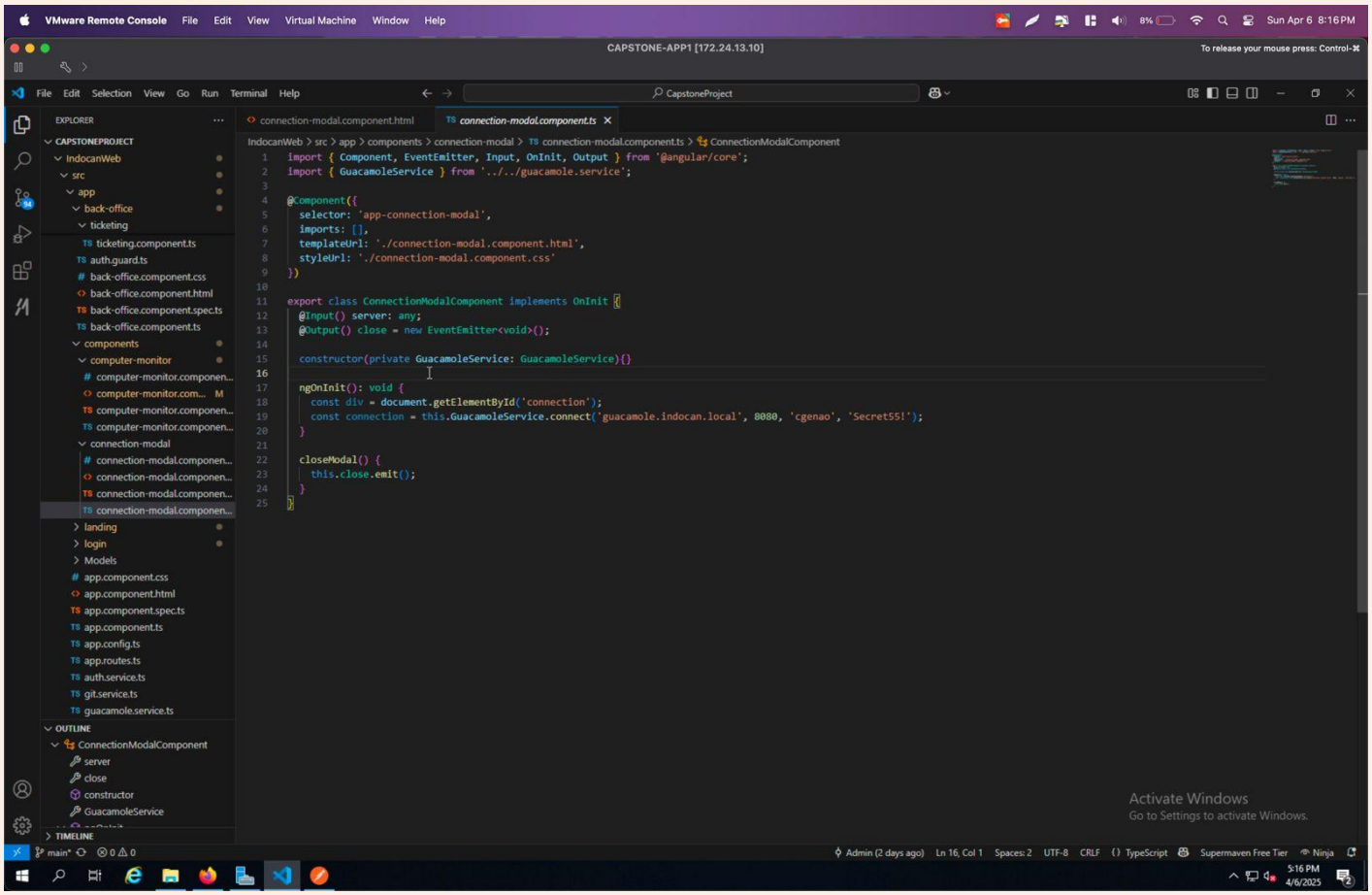
<h3 class="text-xl font-semibold text-gray-700 mb-1">
{{ server.serverName || 'Unknown Server' }}
</h3>
{{ server.fqdn || 'No FQDN available' }}
IP: {{ server.ip || 'No IP available' }}

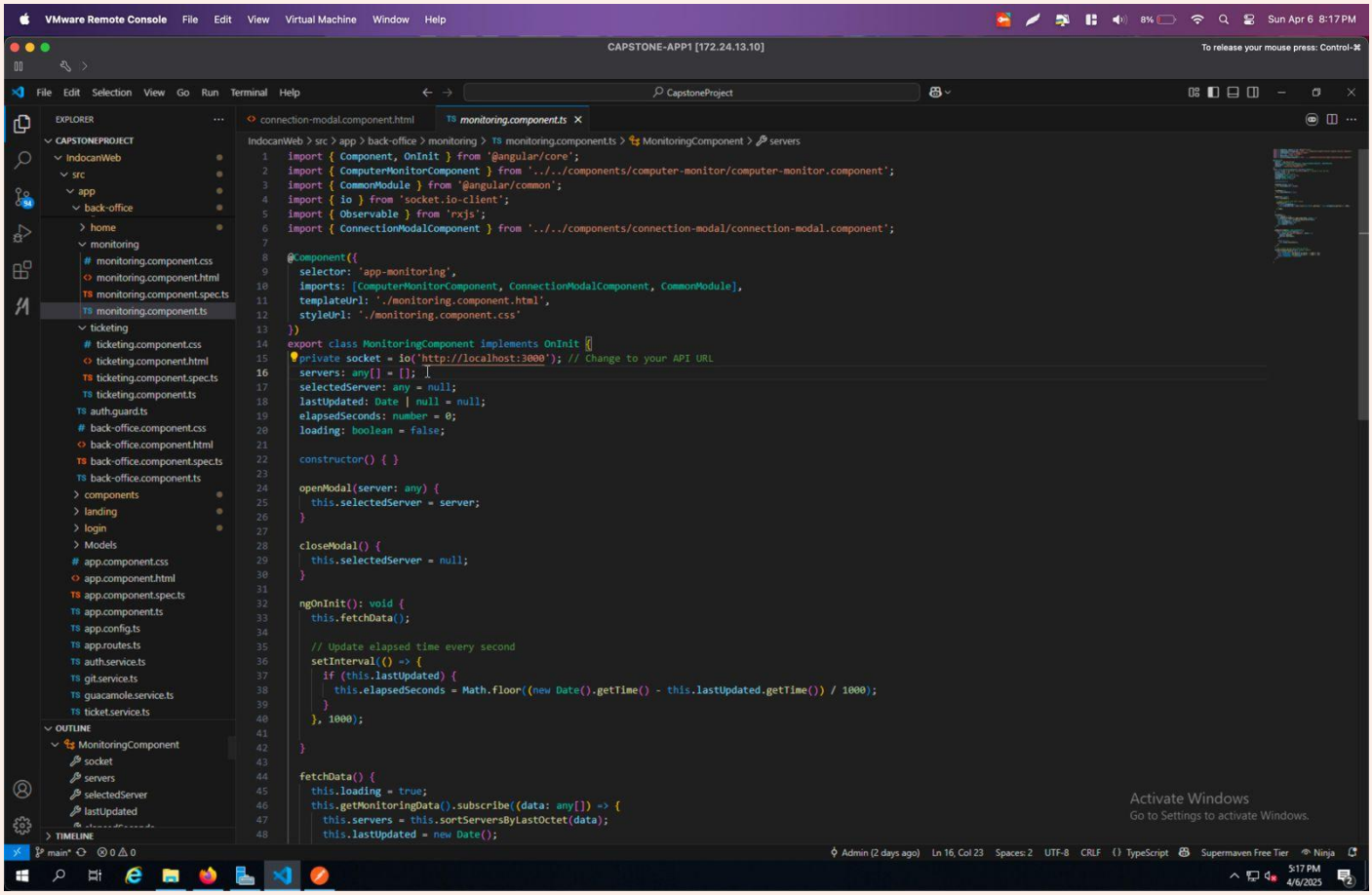
<span class="inline-block px-3 py-1 mt-2 text-sm font-medium rounded-full"
[ngClass]="server.status === 'online' ? 'bg-gradient-to-r from-green-400 to-green-600 text-white' : 'bg-gradient-to-r from-red-400 to-red-600 text-white
{{ server.status | uppercase }}

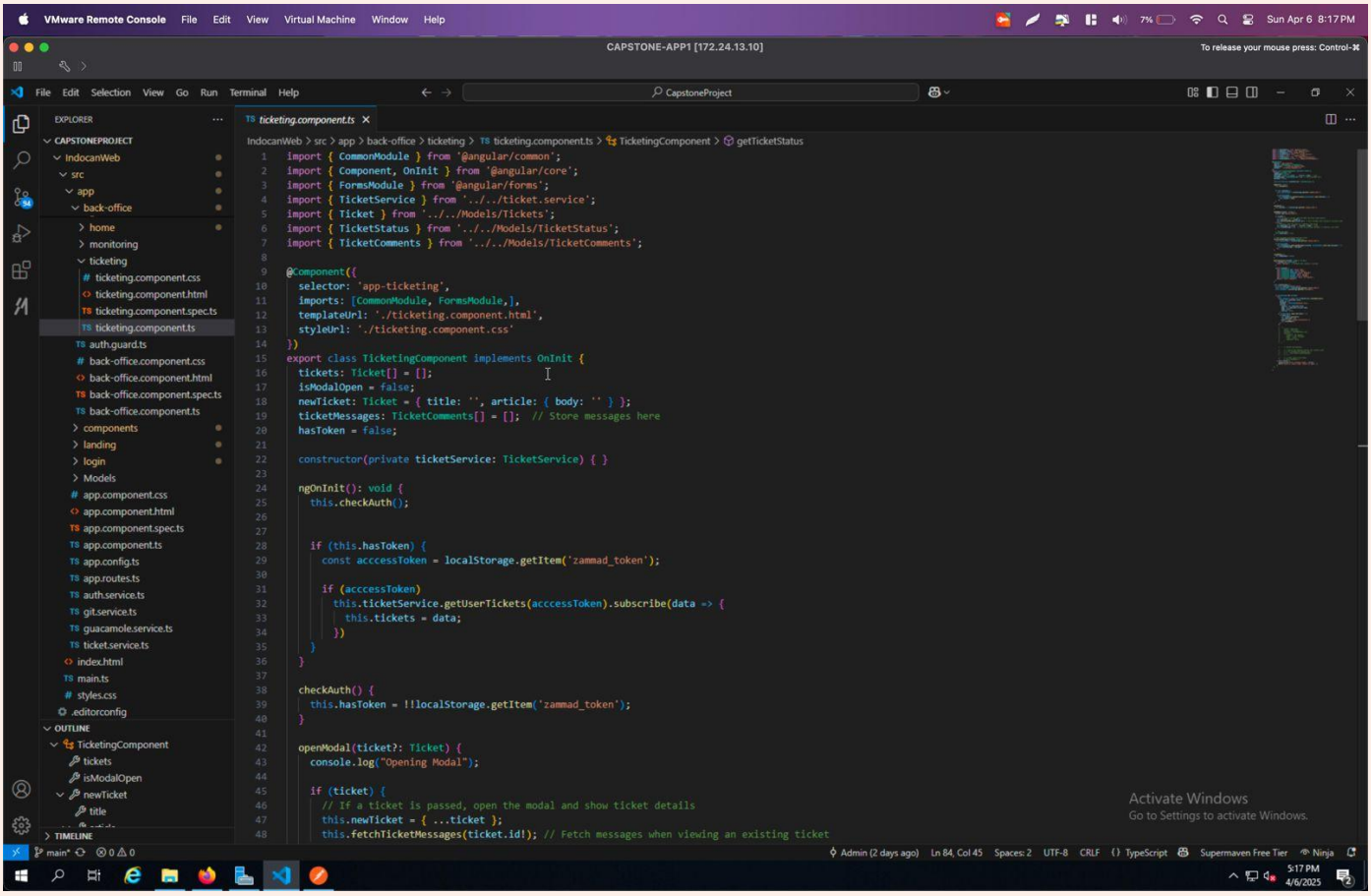
<!-- Connect Button -->
<button (click)="connectToServer()" class="mt-4 bg-blue-500 hover:bg-blue-600 text-white py-2 px-4 rounded-lg">
Connect
</button>

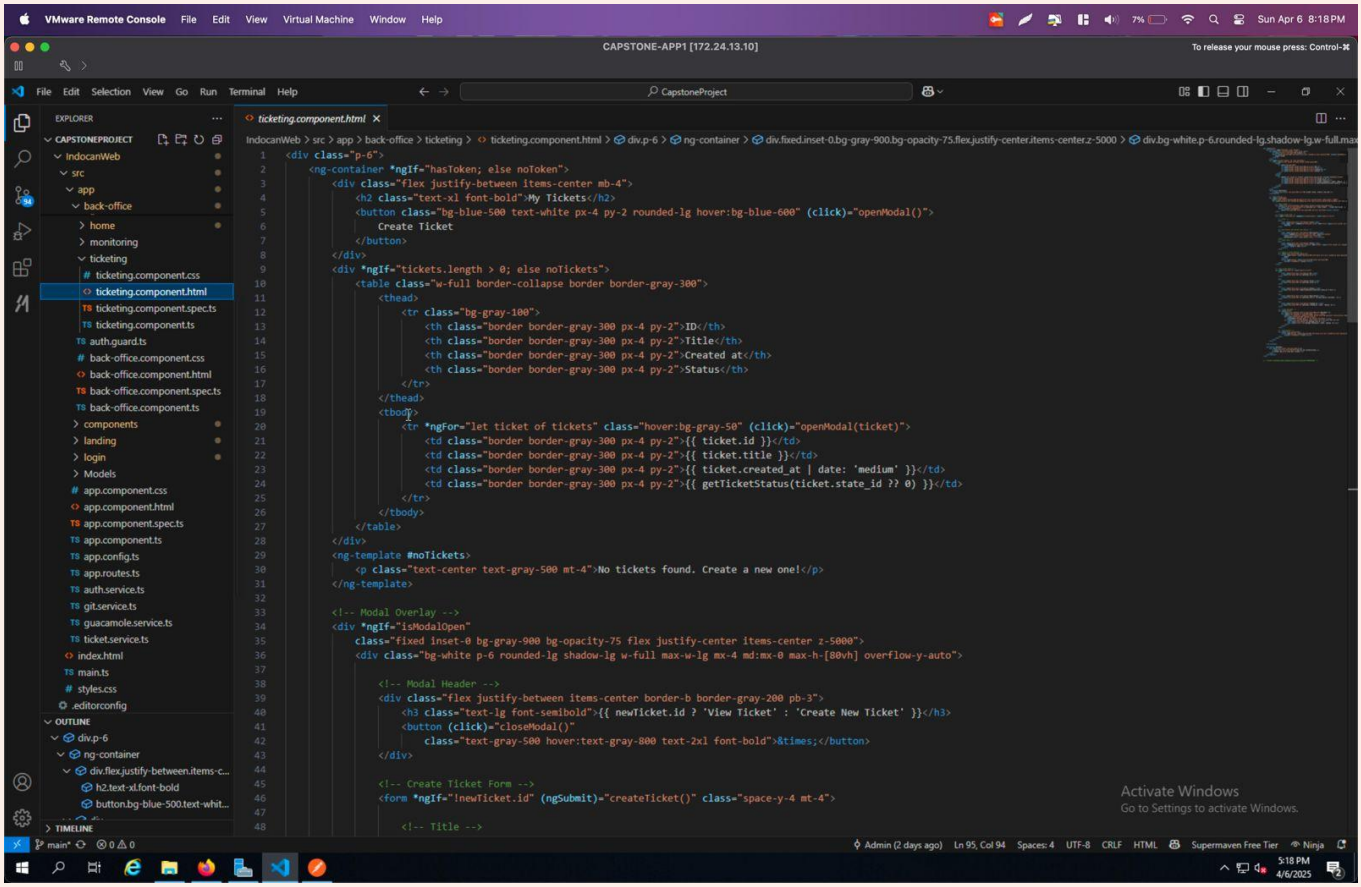
Activate Windows
Go to Settings to activate Windows.


VMware Remote Console File Edit View Virtual Machine Window Help
CAPSTONE-APP1 [172.24.13.10]
To release your mouse press: Control-⌘
File Edit Selection View Go Run Terminal Help
CapstoneProject
EXPLORER
CAPSTONEPROJECT
connection-modal.component.html
computer-monitor.component.ts
IndocanWeb > src > app > components > computer-monitor > computer-monitor.component.ts > ComputerMonitorComponent
import { CommonModule } from '@angular/common';
import { Component, EventEmitter, Input, Output } from '@angular/core';
@Component({
selector: 'app-computer-monitor',
imports: [CommonModule],
templateUrl: './computer-monitor.component.html',
styleUrl: './computer-monitor.component.css'
})
export class ComputerMonitorComponent {
@Input() server: any;
@Output() onConnect = new EventEmitter<any>();
connectToServer() {
window.open(this.server.link, '_blank');
}
constructor() { }
}
OUTLINE
ComputerMonitorComponent
server
onConnect
connectToServer
constructor
TIMELINE
Activate Windows
Go to Settings to activate Windows.
Ln 17, Col 1 Spaces: 2 UTF-8 CRLF TypeScript Supermaven Free Tier Ninja
5:16 PM
4/6/2025


VMware Remote Console File Edit View Virtual Machine Window Help
CAPSTONE-APP1 [172.24.13.10]
To release your mouse press: Control-⌘
File Edit Selection View Go Run Terminal Help
CapstoneProject
EXPLORER
CAPSTONEPROJECT
connection-modal.component.html
connection-modal.component.ts
IndocanWeb > src > app > components > connection-modal > connection-modal.component.ts > ConnectionModalComponent
import { Component, EventEmitter, Input, OnInit, Output } from '@angular/core';
import { GuacamoleService } from '../../guacamole.service';
@Component({
selector: 'app-connection-modal',
imports: [],
templateUrl: './connection-modal.component.html',
styleUrl: './connection-modal.component.css'
})
export class ConnectionModalComponent implements OnInit {
@Input() server: any;
@Output() close = new EventEmitter<void>();
constructor(private GuacamoleService: GuacamoleService){}
ngOnInit(): void {
const div = document.getElementById('connection');
const connection = this.GuacamoleService.connect('guacamole.indocan.local', 8080, 'cgenao', 'Secret55!');
}
closeModal() {
this.close.emit();
}
}
OUTLINE
ConnectionModalComponent
server
close
constructor
GuacamoleService
TIMELINE
Activate Windows
Go to Settings to activate Windows.
Admin (2 days ago) Ln 16, Col 1 Spaces: 2 UTF-8 CRLF TypeScript Supermaven Free Tier Ninja
5:16 PM
4/6/2025

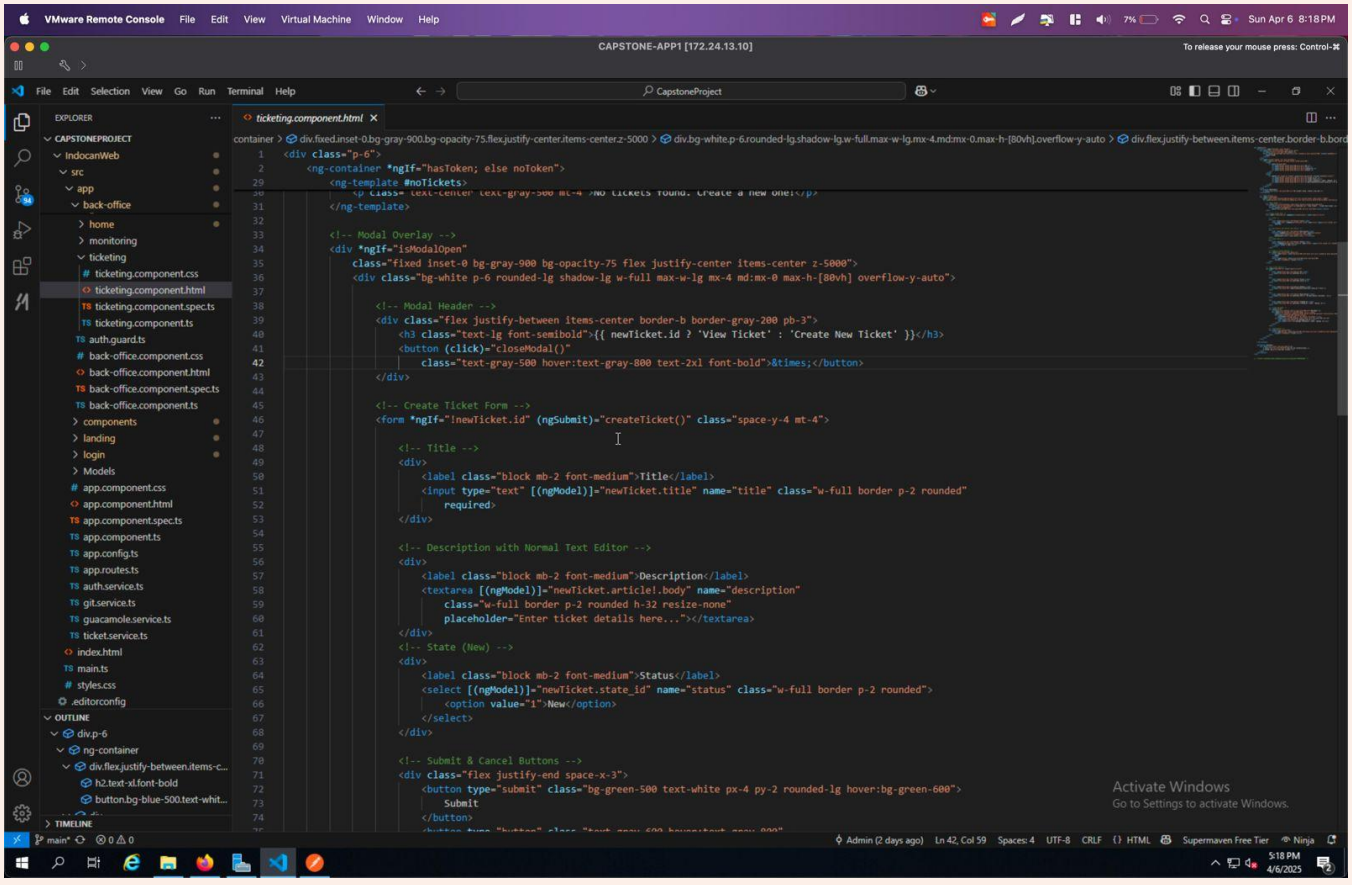

CAPSTONE-APP1 [172.24.13.10]
monitoring.component.ts
import { Component, OnInit } from '@angular/core';
import { ComputerMonitorComponent } from '../../components/computer-monitor/computer-monitor.component';
import { CommonModule } from '@angular/common';
import { io } from 'socket.io-client';
import { Observable } from 'rxjs';
import { ConnectionModalComponent } from '../../components/connection-modal/connection-modal.component';
@Component({
selector: 'app-monitoring',
imports: [ComputerMonitorComponent, ConnectionModalComponent, CommonModule],
templateUrl: './monitoring.component.html',
styleUrl: './monitoring.component.css'
})
export class MonitoringComponent implements OnInit {
private socket = io('http://localhost:3000'); // Change to your API URL
servers: any[] = [];
selectedServer: any = null;
lastUpdated: Date | null = null;
elapsedSeconds: number = 0;
loading: boolean = false;
constructor() { }
openModal(server: any) {
this.selectedServer = server;
}
closeModal() {
this.selectedServer = null;
}
ngOnInit(): void {
this.fetchData();
// Update elapsed time every second
setInterval(() => {
if (this.lastUpdated) {
this.elapsedSeconds = Math.floor((new Date().getTime() - this.lastUpdated.getTime()) / 1000);
}
}, 1000);
}
fetchData() {
this.loading = true;
this.getMonitoringData().subscribe((data: any[]) => {
this.servers = this.sortServersByLastOctet(data);
this.lastUpdated = new Date();
Activate Windows
Go to Settings to activate Windows.


CAPSTONE-APP1 [172.24.13.10]
ticketing.component.ts
import { CommonModule } from '@angular/common';
import { Component, OnInit } from '@angular/core';
import { FormsModule } from '@angular/forms';
import { TicketService } from '../../ticket.service';
import { Ticket } from '../../Models/Tickets';
import { TicketStatus } from '../../Models/TicketStatus';
import { TicketComments } from '../../Models/TicketComments';
@Component({
selector: 'app-ticketing',
imports: [CommonModule, FormsModule,],
templateUrl: './ticketing.component.html',
styleUrl: './ticketing.component.css'
})
export class TicketingComponent implements OnInit {
tickets: Ticket[] = [];
isModalOpen = false;
newTicket: Ticket = { title: '', article: { body: '' } };
ticketMessages: TicketComments[] = []; // Store messages here
hasToken = false;
constructor(private ticketService: TicketService) { }
ngOnInit(): void {
this.checkAuth();
if (this.hasToken) {
const acccessToken = localStorage.getItem('zammad_token');
if (acccessToken)
this.ticketService.getUserTickets(acccessToken).subscribe(data => {
this.tickets = data;
})
}
checkAuth() {
this.hasToken = !!localStorage.getItem('zammad_token');
}
openModal(ticket?: Ticket) {
console.log("Opening Modal");
if (ticket) {
// If a ticket is passed, open the modal and show ticket details
this.newTicket = { ...ticket };
this.fetchTicketMessages(ticket.id!); // Fetch messages when viewing an existing ticket
Activate Windows
Go to Settings to activate Windows.

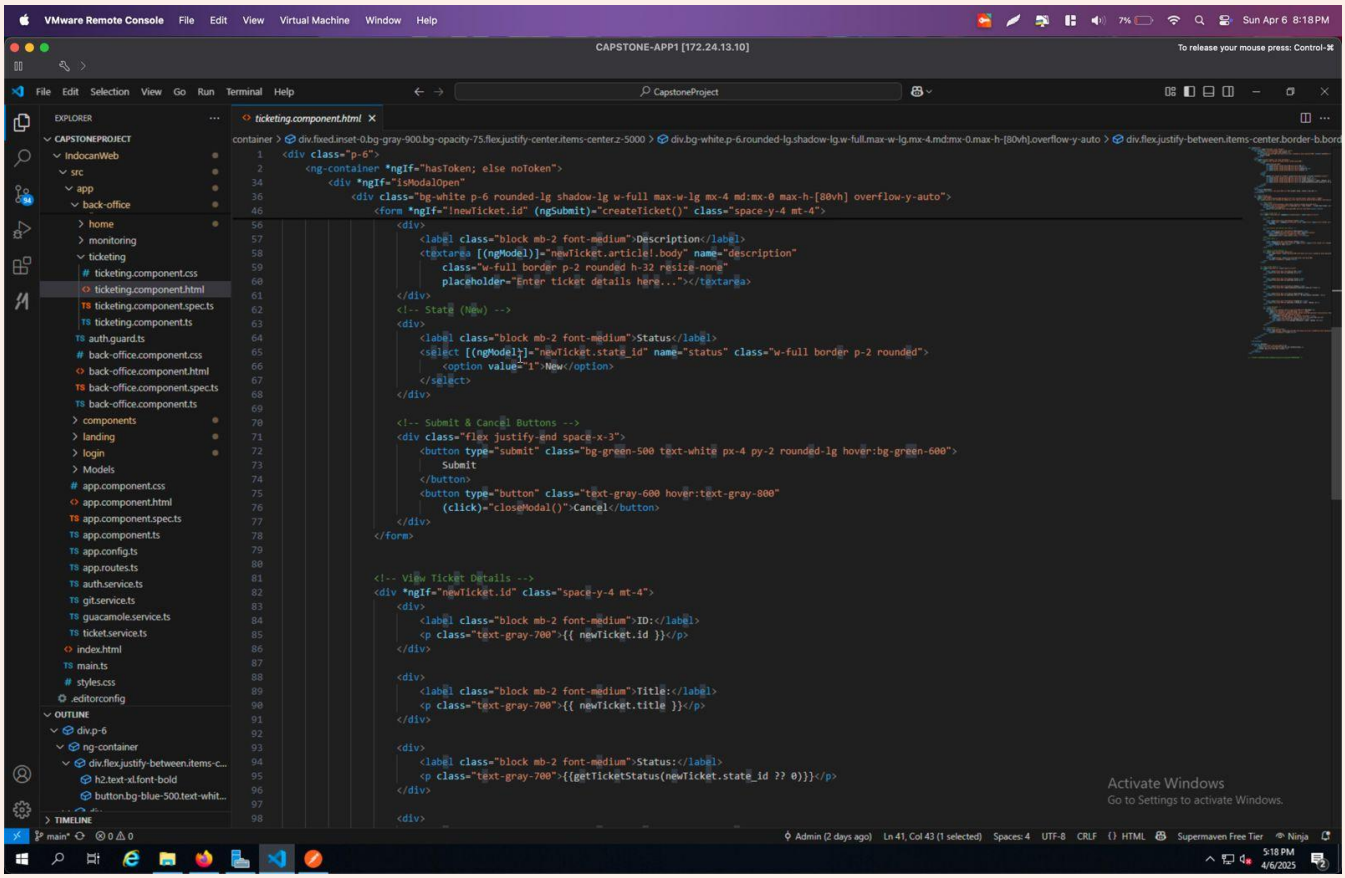

VMware Remote Console File Edit View Virtual Machine Window Help
CAPSTONE-APP1 [172.24.13.10]
To release your mouse press: Control-⌘
File Edit Selection View Go Run Terminal Help
CapstoneProject
EXPLORER
CAPSTONEPROJECT
IndocanWeb
src
app
back-office
home
monitoring
ticketing
ticketing.component.css
ticketing.component.html
ticketing.component.spec.ts
ticketing.component.ts
auth.guard.ts
back-office.component.css
back-office.component.html
back-office.component.spec.ts
back-office.component.ts
components
landing
login
Models
app.component.css
app.component.html
app.component.spec.ts
app.component.ts
app.config.ts
app.routes.ts
auth.service.ts
git.service.ts
guacamole.service.ts
ticket.service.ts
index.html
main.ts
styles.css
.editorconfig
OUTLINE
TIMELINE
ticketing.component.html

<ng-container *ngIf="hasToken; else noToken">
<h2 class="text-xl font-bold">My Tickets</h2>
Create Ticket
<table class="w-full border-collapse border border-gray-300">
<th class="border border-gray-300 px-4 py-2">ID</th>
<th class="border border-gray-300 px-4 py-2">Title</th>
<th class="border border-gray-300 px-4 py-2">Created at</th>
<th class="border border-gray-300 px-4 py-2">Status</th>
<tr *ngFor="let ticket of tickets" class="hover:bg-gray-50" (click)="openModal(ticket)">
<td class="border border-gray-300 px-4 py-2">{{ ticket.id }}</td>
<td class="border border-gray-300 px-4 py-2">{{ ticket.title }}</td>
<td class="border border-gray-300 px-4 py-2">{{ ticket.created_at | date: 'medium' }}</td>
<td class="border border-gray-300 px-4 py-2">{{ getTicketStatus(ticket.state_id ?? 0) }}</td>
No tickets found. Create a new one!
<!-- Modal Overlay -->
<div *ngIf="isModalOpen"
class="fixed inset-0 bg-gray-900 bg-opacity-75 flex justify-center items-center z-5000">

<!-- Modal Header -->
<h3 class="text-lg font-semibold">{{ newTicket.id ? 'View Ticket' : 'Create New Ticket' }}</h3>
<button (click)="closeModal()"
class="text-gray-500 hover:text-gray-800 text-2xl font-bold">×</button>
<!-- Create Ticket Form -->
<form *ngIf="!newTicket.id" (ngSubmit)="createTicket()" class="space-y-4 mt-4">
<!-- Title -->
Activate Windows
Go to Settings to activate Windows.
Admin (2 days ago) Ln 95, Col 94 Spaces: 4 UTF-8 CRLF HTML Supermaven Free Tier Ninja
5:18 PM
4/6/2025

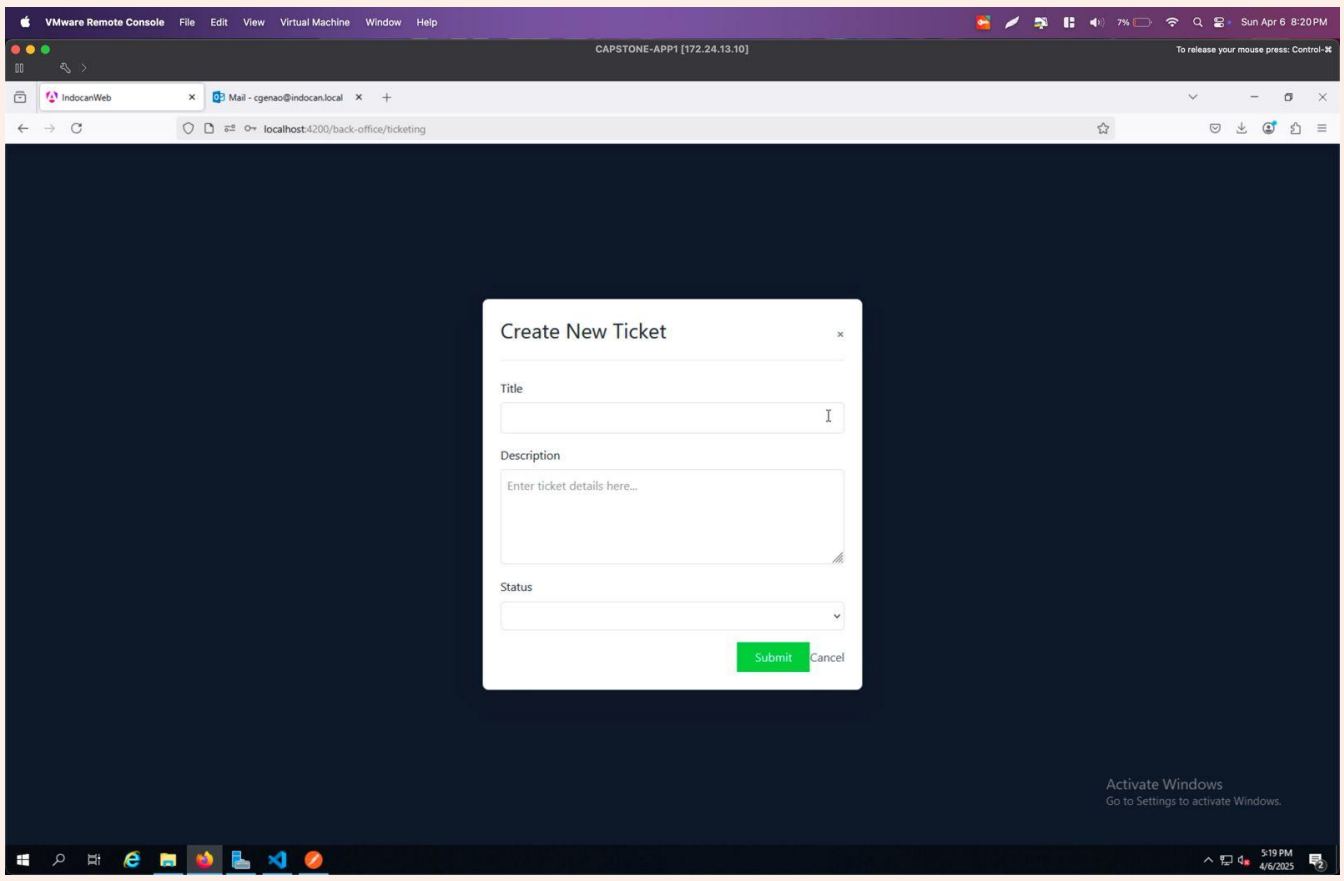

VMware Remote Console File Edit View Virtual Machine Window Help
CAPSTONE-APP1 [172.24.13.10]
To release your mouse press: Control-⌘
File Edit Selection View Go Run Terminal Help
CapstoneProject
EXPLORER
CAPSTONEPROJECT
ticketing.component.html
<label class="block mb-2 font-medium">Title</label>
<input type="text" [(ngModel)]="newTicket.title" name="title" class="w-full border p-2 rounded"
required>
<!-- Description with Normal Text Editor -->
<label class="block mb-2 font-medium">Description</label>
<textarea [(ngModel)]="newTicket.article!.body" name="description"
class="w-full border p-2 rounded h-32 resize-none"
placeholder="Enter ticket details here..."></textarea>
<!-- State (New) -->
<label class="block mb-2 font-medium">Status</label>
<select [(ngModel)]="newTicket.state_id" name="status" class="w-full border p-2 rounded">
<option value="1">New</option>
<!-- Submit & Cancel Buttons -->

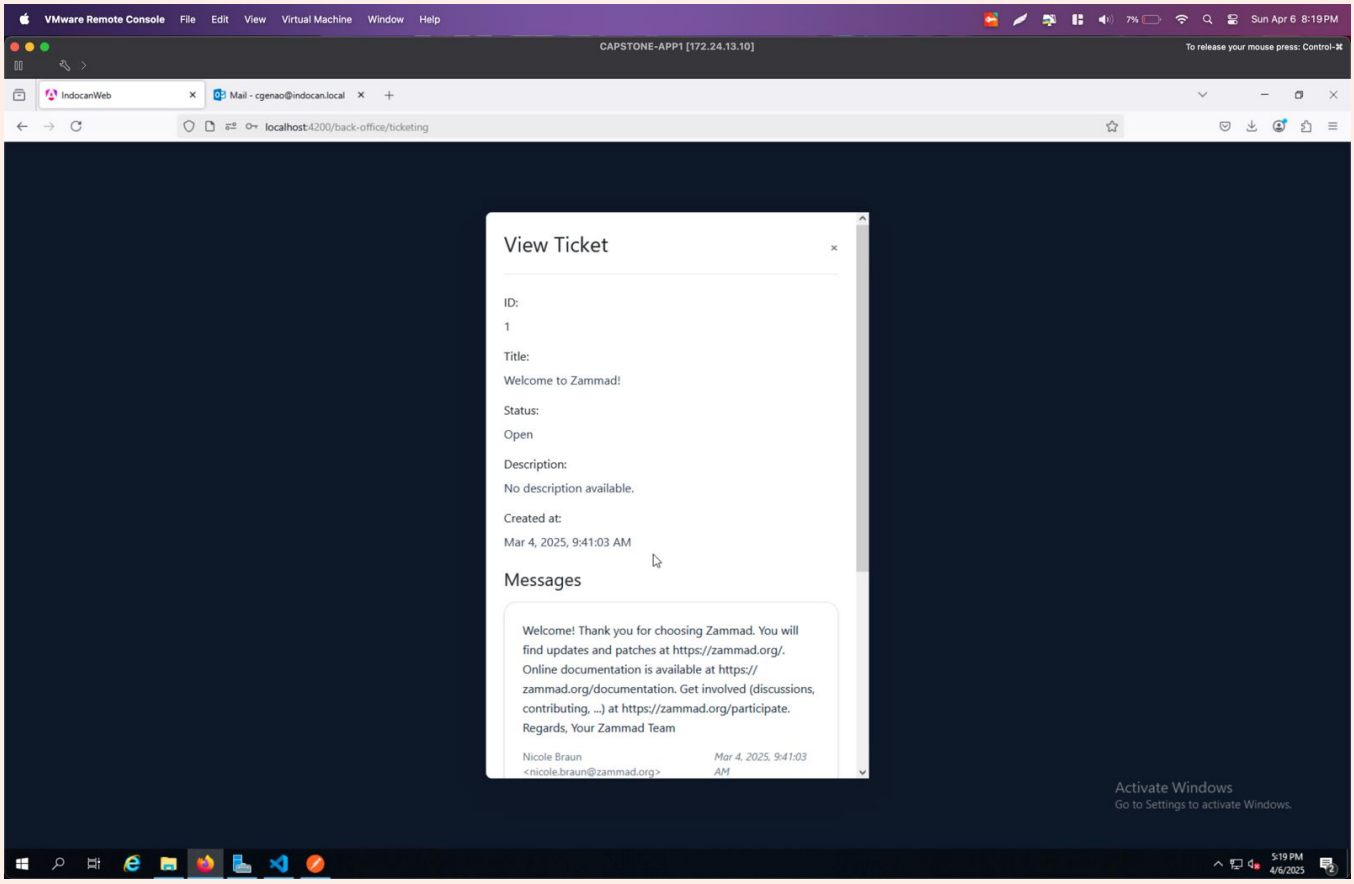
<button type="submit" class="bg-green-500 text-white px-4 py-2 rounded-lg hover:bg-green-600">
Submit
Activate Windows
Go to Settings to activate Windows.
Admin (2 days ago) Ln 42, Col 59 Spaces: 4 UTF-8 CRLF HTML Supermaven Free Tier Ninja
5:18 PM
4/6/2025


VMware Remote Console File Edit View Virtual Machine Window Help
Sun Apr 6 8:18 PM
CAPSTONE-APP1 [172.24.13.10]
To release your mouse press: Control-⌘
File Edit Selection View Go Run Terminal Help
CapstoneProject
EXPLORER
CAPSTONEPROJECT
IndocanWeb
src
app
back-office
home
monitoring
ticketing
ticketing.component.css
ticketing.component.html
ticketing.component.spec.ts
ticketing.component.ts
auth.guard.ts
back-office.component.css
back-office.component.html
back-office.component.spec.ts
back-office.component.ts
components
landing
login
Models
app.component.css
app.component.html
app.component.spec.ts
app.component.ts
app.config.ts
app.routes.ts
auth.service.ts
git.service.ts
guacamole.service.ts
ticket.service.ts
index.html
main.ts
styles.css
.editorconfig
OUTLINE
TIMELINE
ticketing.component.html

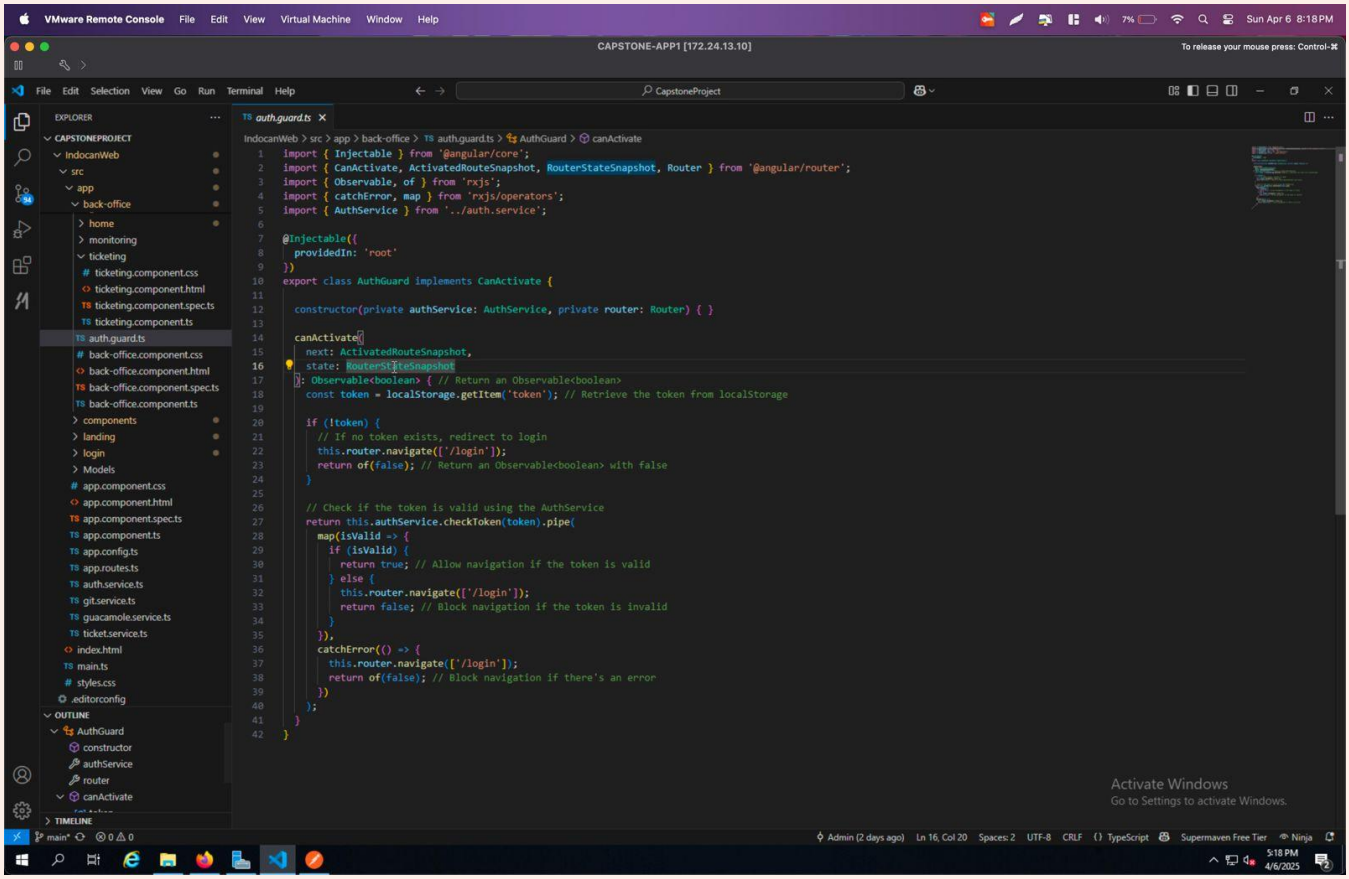
<ng-container *ngIf="hasToken; else noToken">
<div *ngIf="isModalOpen"

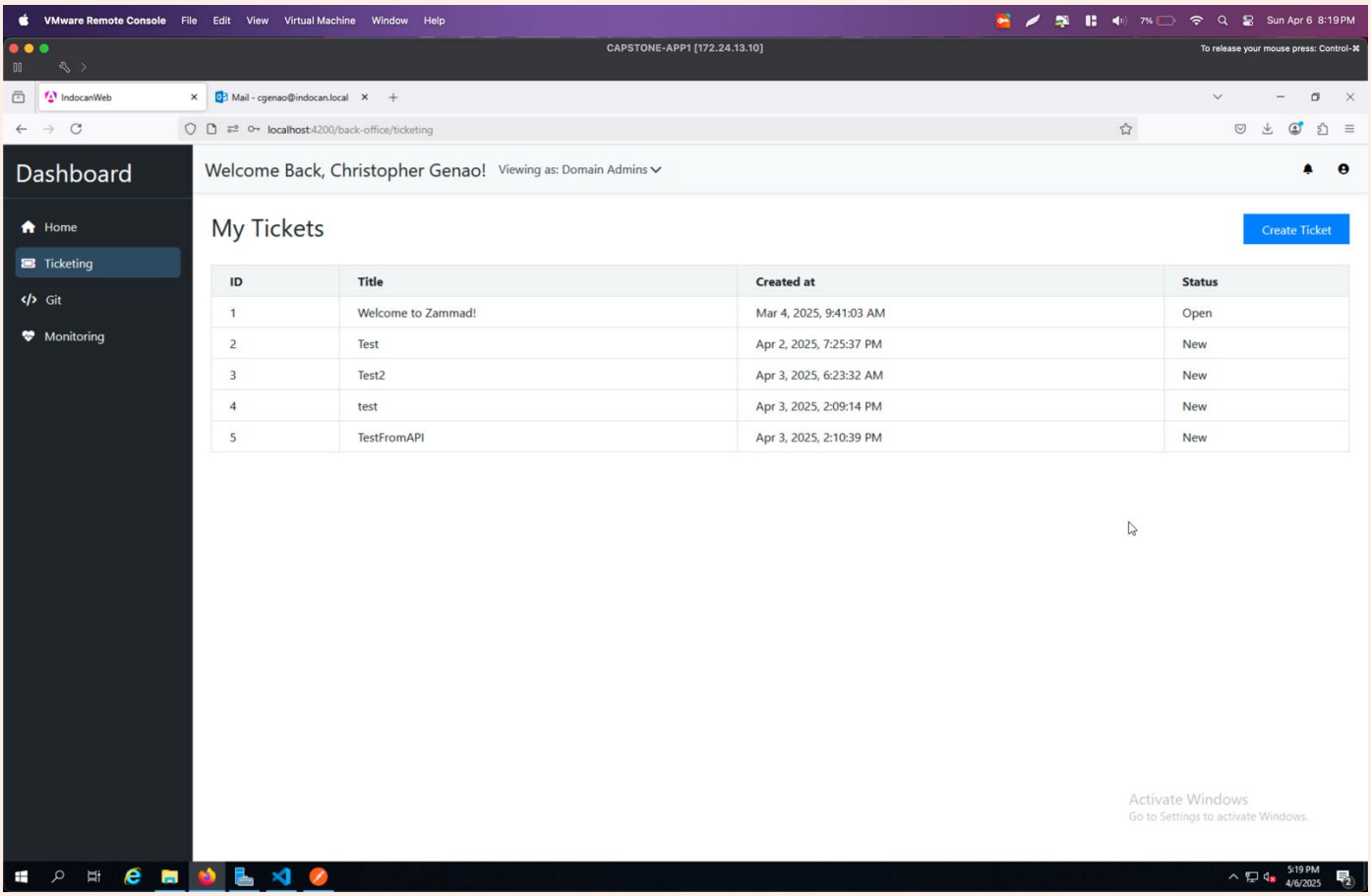
<form *ngIf="!newTicket.id" (ngSubmit)="createTicket()" class="space-y-4 mt-4">
<label class="block mb-2 font-medium">Description</label>
<textarea [(ngModel)]="newTicket.article!.body" name="description"
class="w-full border p-2 rounded h-32 resize-none"
placeholder="Enter ticket details here..."></textarea>
<!-- State (New) -->
<label class="block mb-2 font-medium">Status</label>
<select [(ngModel)]="newTicket.state_id" name="status" class="w-full border p-2 rounded">
<option value="1">New</option>
<!-- Submit & Cancel Buttons -->
<button type="submit" class="bg-green-500 text-white px-4 py-2 rounded-lg hover:bg-green-600">
Submit
<button type="button" class="text-gray-600 hover:text-gray-800"
(click)="closeModal()">Cancel</button>
</form>
<!-- View Ticket Details -->
<div *ngIf="newTicket.id" class="space-y-4 mt-4">
<label class="block mb-2 font-medium">ID:</label>
{{ newTicket.id }}
<label class="block mb-2 font-medium">Title:</label>
{{ newTicket.title }}
<label class="block mb-2 font-medium">Status:</label>
{{getTicketStatus(newTicket.state_id ?? 0)}}
Activate Windows
Go to Settings to activate Windows.
5:18 PM
4/6/2025

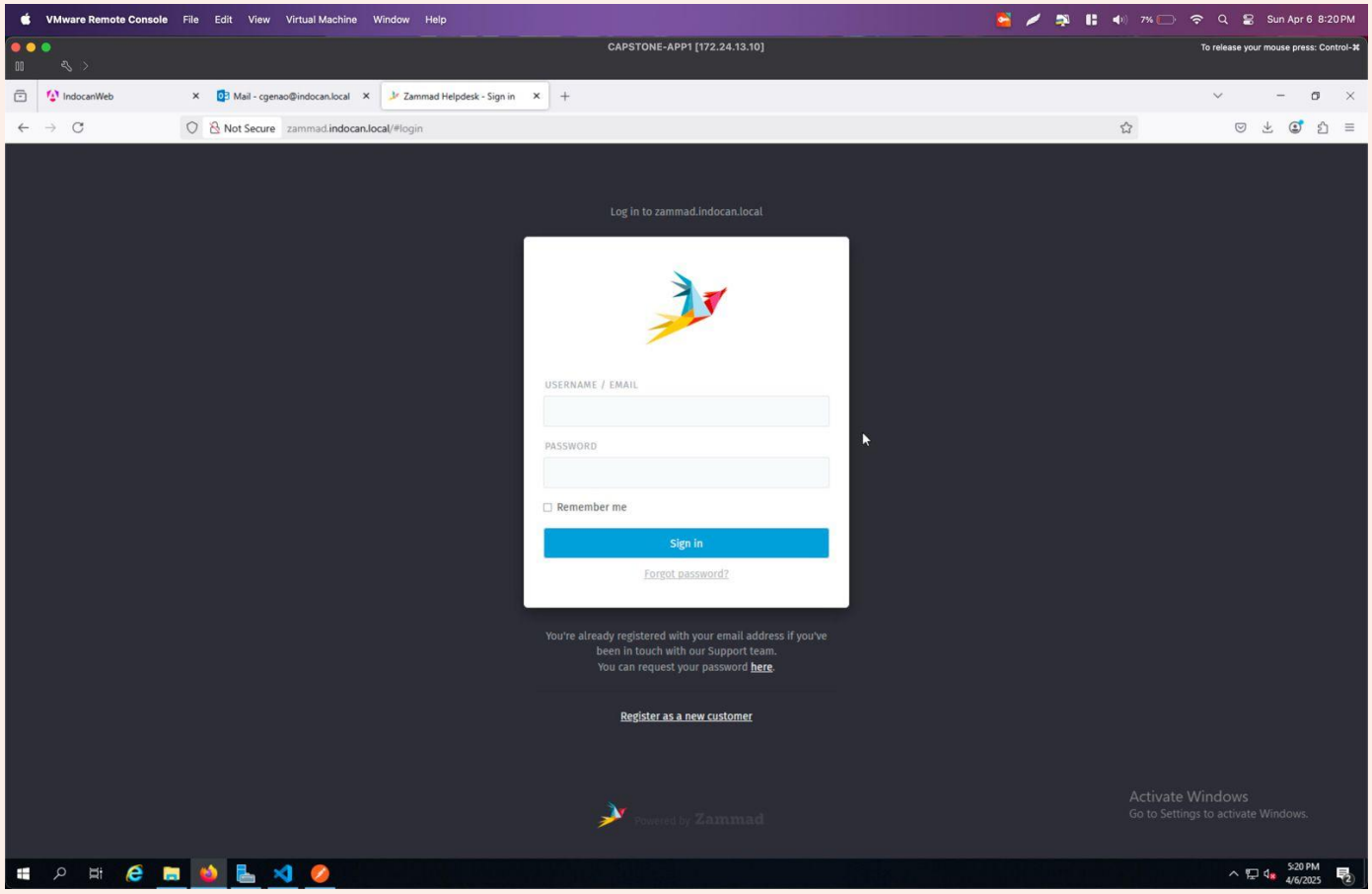

VMware Remote Console File Edit View Virtual Machine Window Help
Sun Apr 6 8:20 PM
CAPSTONE-APP1 [172.24.13.10]
To release your mouse press: Control-⌘
IndocanWeb
Mail - cgenao@indocan.local
localhost:4200/back-office/ticketing
Create New Ticket
Title
Description
Enter ticket details here...
Status
Submit
Cancel
Activate Windows
Go to Settings to activate Windows.
5:19 PM
4/6/2025

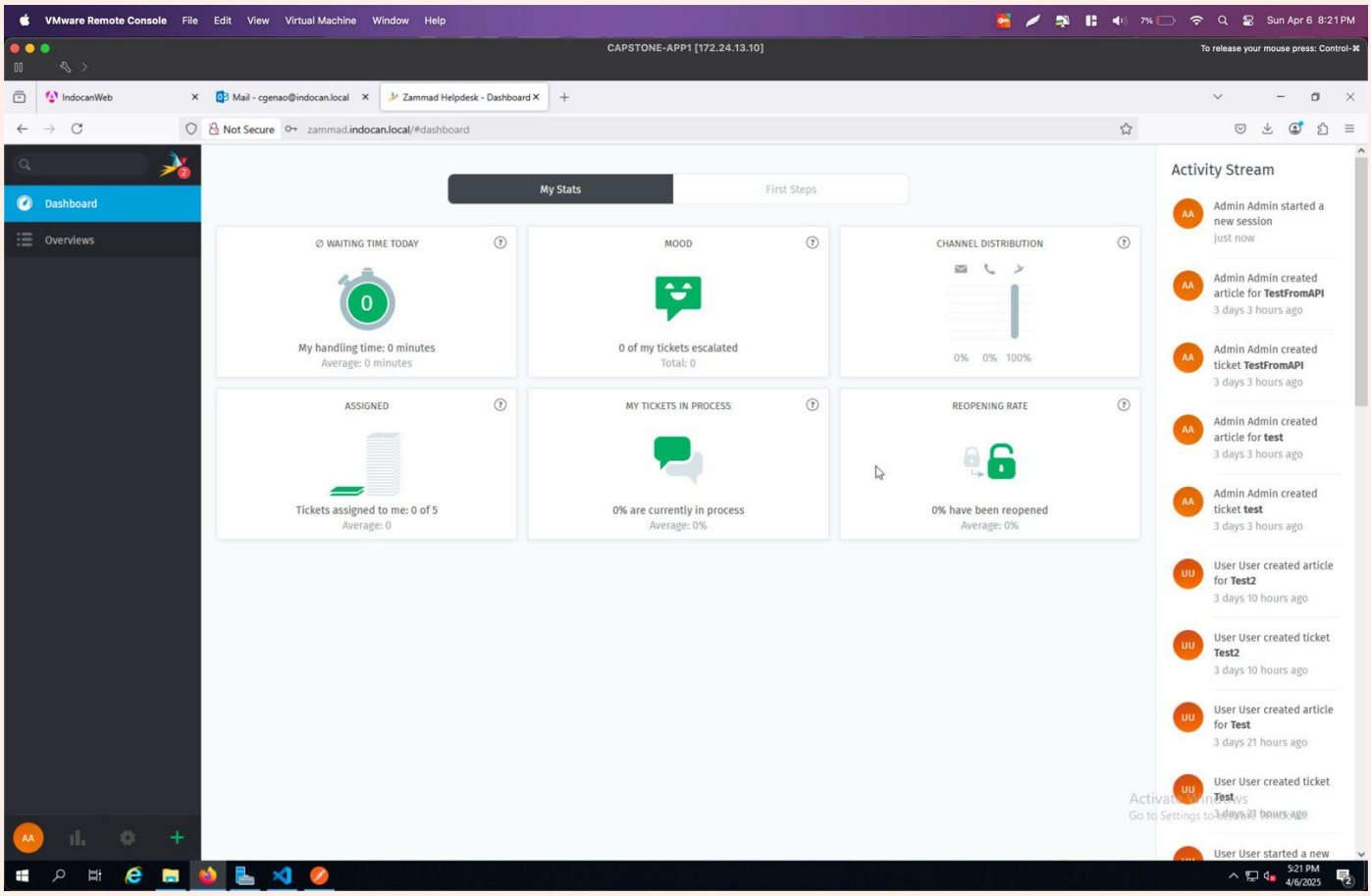


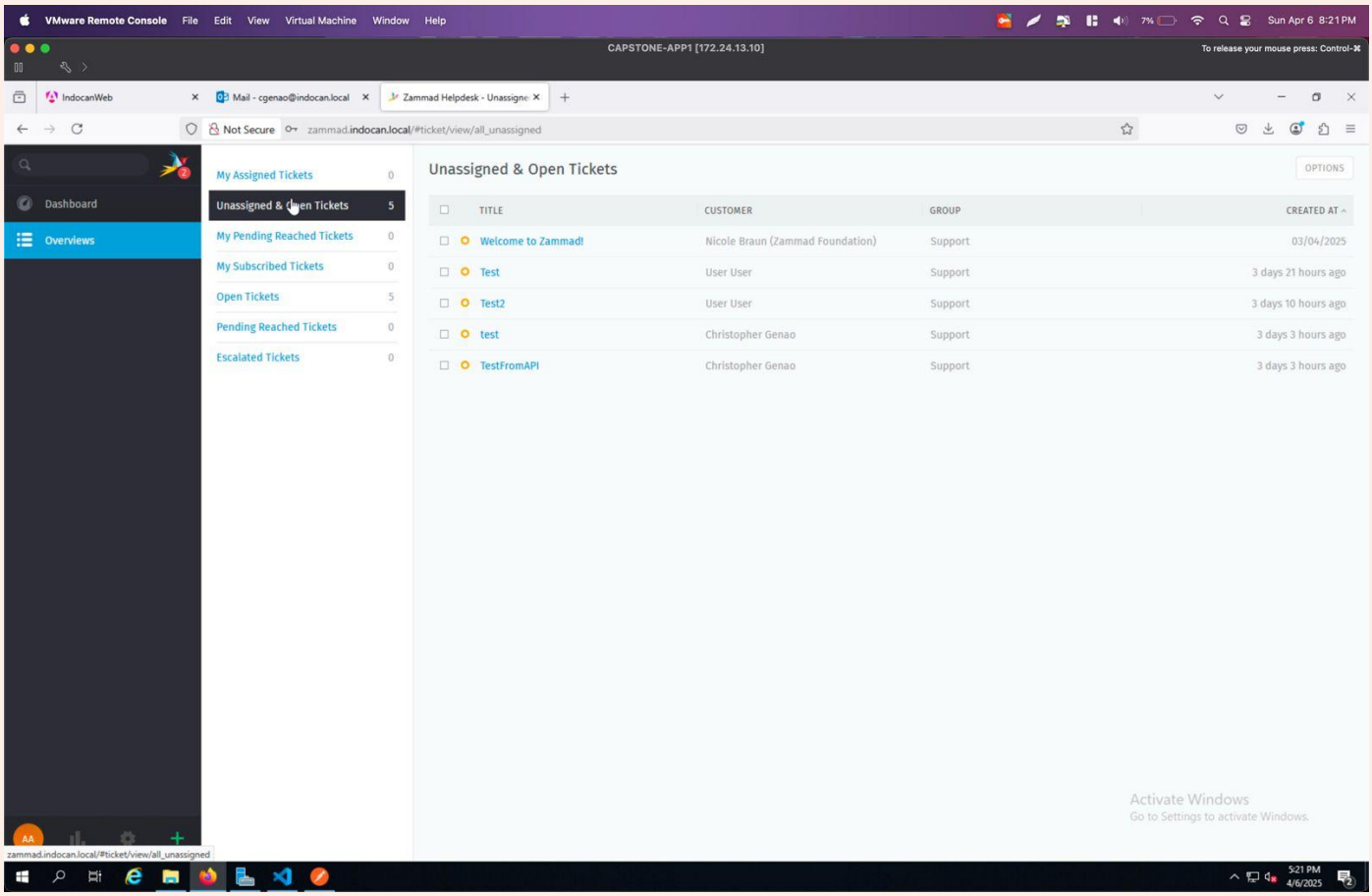
AUTHENTICATION GUARD



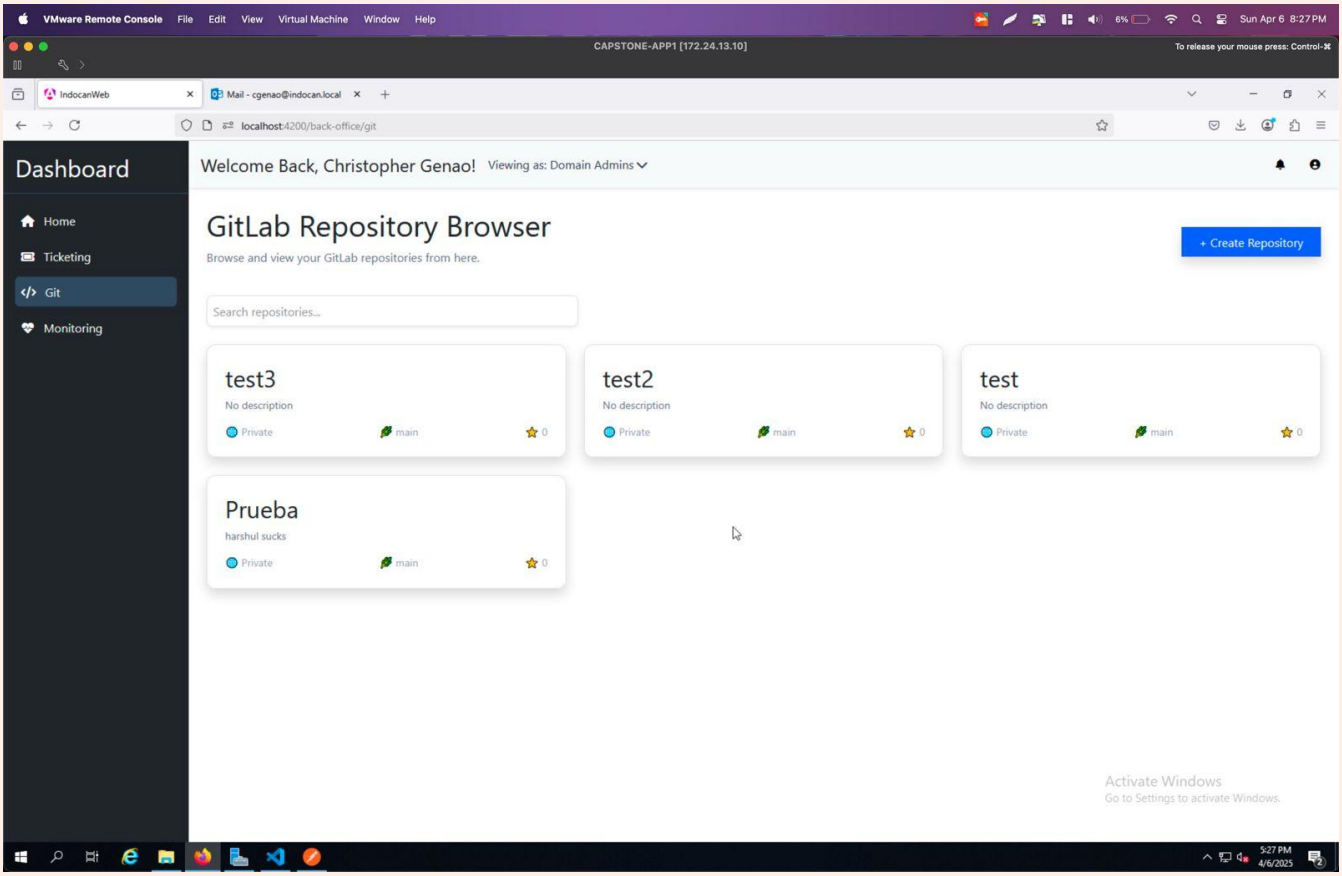

VMware Remote Console File Edit View Virtual Machine Window Help
Sun Apr 6 8:19 PM
CAPSTONE-APP1 [172.24.13.10]
To release your mouse press: Control-⌘
IndocanWeb
Mail - cgenao@indocan.local
localhost:4200/back-office/ticketing
Dashboard
Welcome Back, Christopher Genao!
Viewing as: Domain Admins
Home
Ticketing
Git
Monitoring
My Tickets
Create Ticket
ID
Title
Created at
Status
1
Welcome to Zammad!
Mar 4, 2025, 9:41:03 AM
Open
2
Test
Apr 2, 2025, 7:25:37 PM
New
3
Test2
Apr 3, 2025, 6:23:32 AM
New
4
test
Apr 3, 2025, 2:09:14 PM
New
5
TestFromAPI
Apr 3, 2025, 2:10:39 PM
New
Activate Windows
Go to Settings to activate Windows.
5:19 PM
4/6/2025

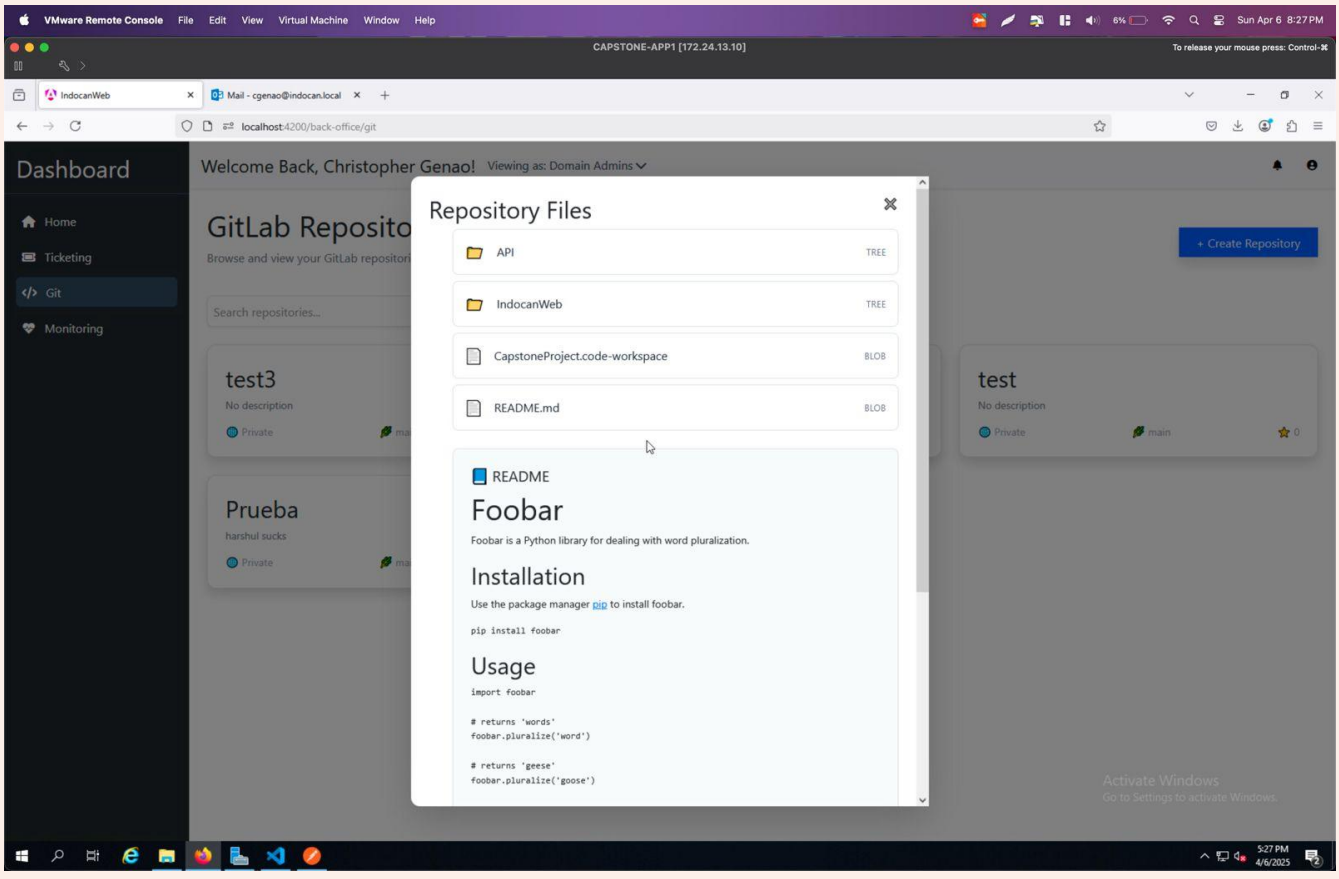

VMware Remote Console File Edit View Virtual Machine Window Help
Sun Apr 6 8:20 PM
CAPSTONE-APP1 [172.24.13.10]
To release your mouse press: Control-⌘
IndocanWeb
Mail - cgenao@indocan.local
Zammad Helpdesk - Sign in
Not Secure
zammad.indocan.local/#login
Log in to zammad.indocan.local
USERNAME / EMAIL
PASSWORD
Remember me
Sign in
Forgot password?
You're already registered with your email address if you've been in touch with our Support team.
You can request your password here.
Register as a new customer
Powered by Zammad
Activate Windows
Go to Settings to activate Windows.
5:20 PM
4/6/2025

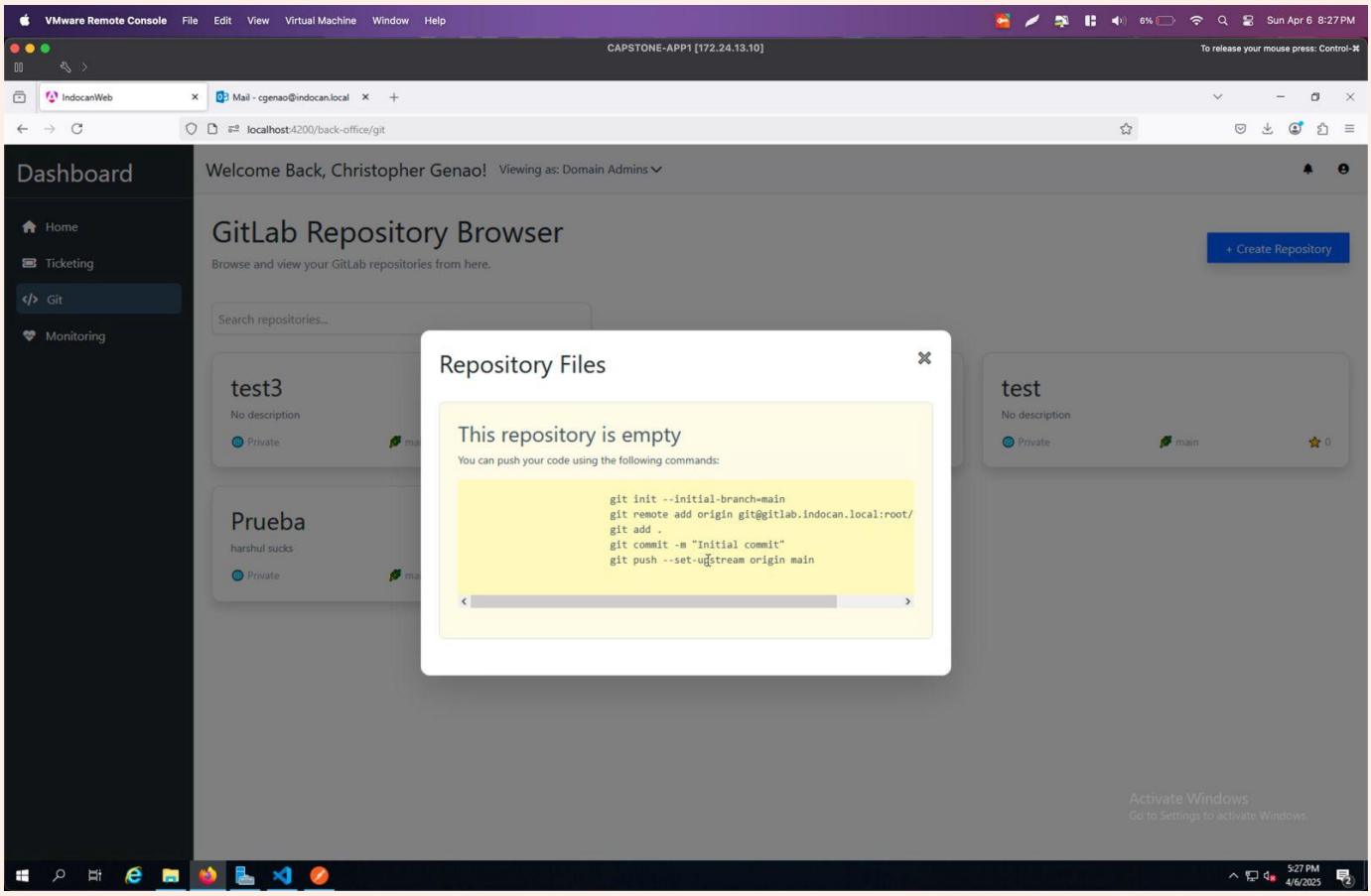

VMware Remote Console File Edit View Virtual Machine Window Help
Sun Apr 6 8:21 PM
CAPSTONE-APP1 [172.24.13.10]
To release your mouse press: Control-⌘
IndocanWeb
Mail - cgenao@indocan.local
Zammad Helpdesk - Dashboard
Not Secure
zammad.indocan.local/#dashboard
Dashboard
Overviews
My Stats
First Steps
Ø WAITING TIME TODAY
My handling time: 0 minutes
Average: 0 minutes
MOOD
0 of my tickets escalated
Total: 0
CHANNEL DISTRIBUTION
0% 0% 100%
ASSIGNED
Tickets assigned to me: 0 of 5
Average: 0
MY TICKETS IN PROCESS
0% are currently in process
Average: 0%
REOPENING RATE
0% have been reopened
Average: 0%
Activity Stream
Admin Admin started a new session
just now
Admin Admin created article for TestFromAPI
3 days 3 hours ago
Admin Admin created ticket TestFromAPI
3 days 3 hours ago
Admin Admin created article for test
3 days 3 hours ago
Admin Admin created ticket test
3 days 3 hours ago
User User created article for Test2
3 days 10 hours ago
User User created ticket Test2
3 days 10 hours ago
User User created article for Test
3 days 21 hours ago
User User created ticket Test
3 days 21 hours ago
Activate Windows
Go to Settings to activate Windows.
5:21 PM
4/6/2025

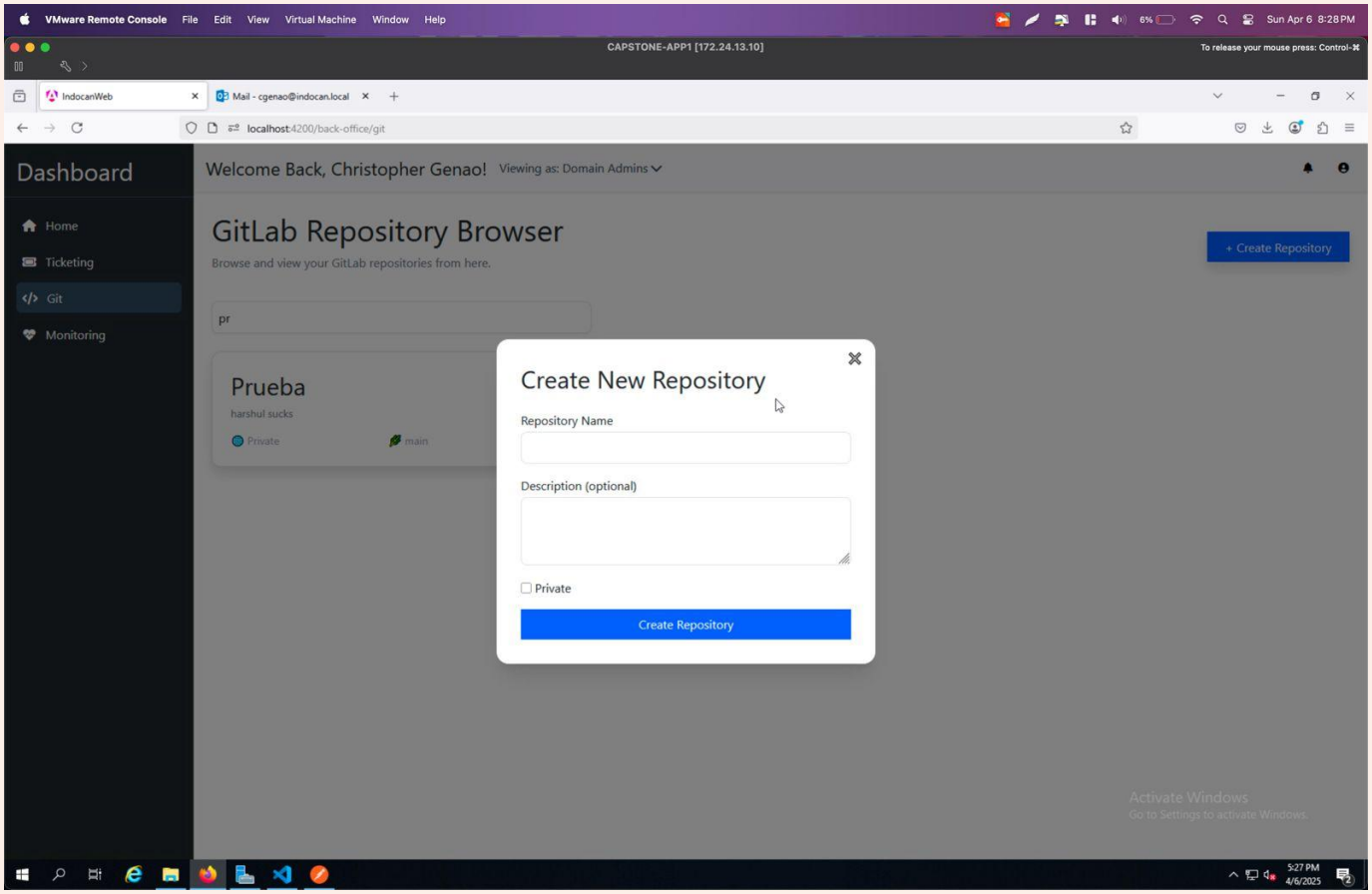

VMware Remote Console File Edit View Virtual Machine Window Help
Sun Apr 6 8:21 PM
CAPSTONE-APP1 [172.24.13.10]
To release your mouse press: Control-⌘
IndocanWeb
Mail - cgenao@indocan.local
Zammad Helpdesk - Unassigne
Not Secure
zammad.indocan.local/#ticket/view/all_unassigned
Dashboard
Overviews
My Assigned Tickets 0
Unassigned & Open Tickets 5
My Pending Reached Tickets 0
My Subscribed Tickets 0
Open Tickets 5
Pending Reached Tickets 0
Escalated Tickets 0
Unassigned & Open Tickets
OPTIONS
TITLE
CUSTOMER
GROUP
CREATED AT
Welcome to Zammad!
Nicole Braun (Zammad Foundation)
Support
03/04/2025
Test
User User
Support
3 days 21 hours ago
Test2
User User
Support
3 days 10 hours ago
test
Christopher Genao
Support
3 days 3 hours ago
TestFromAPI
Christopher Genao
Support
3 days 3 hours ago
Activate Windows
Go to Settings to activate Windows.
zammad.indocan.local/#ticket/view/all_unassigned
5:21 PM
4/6/2025

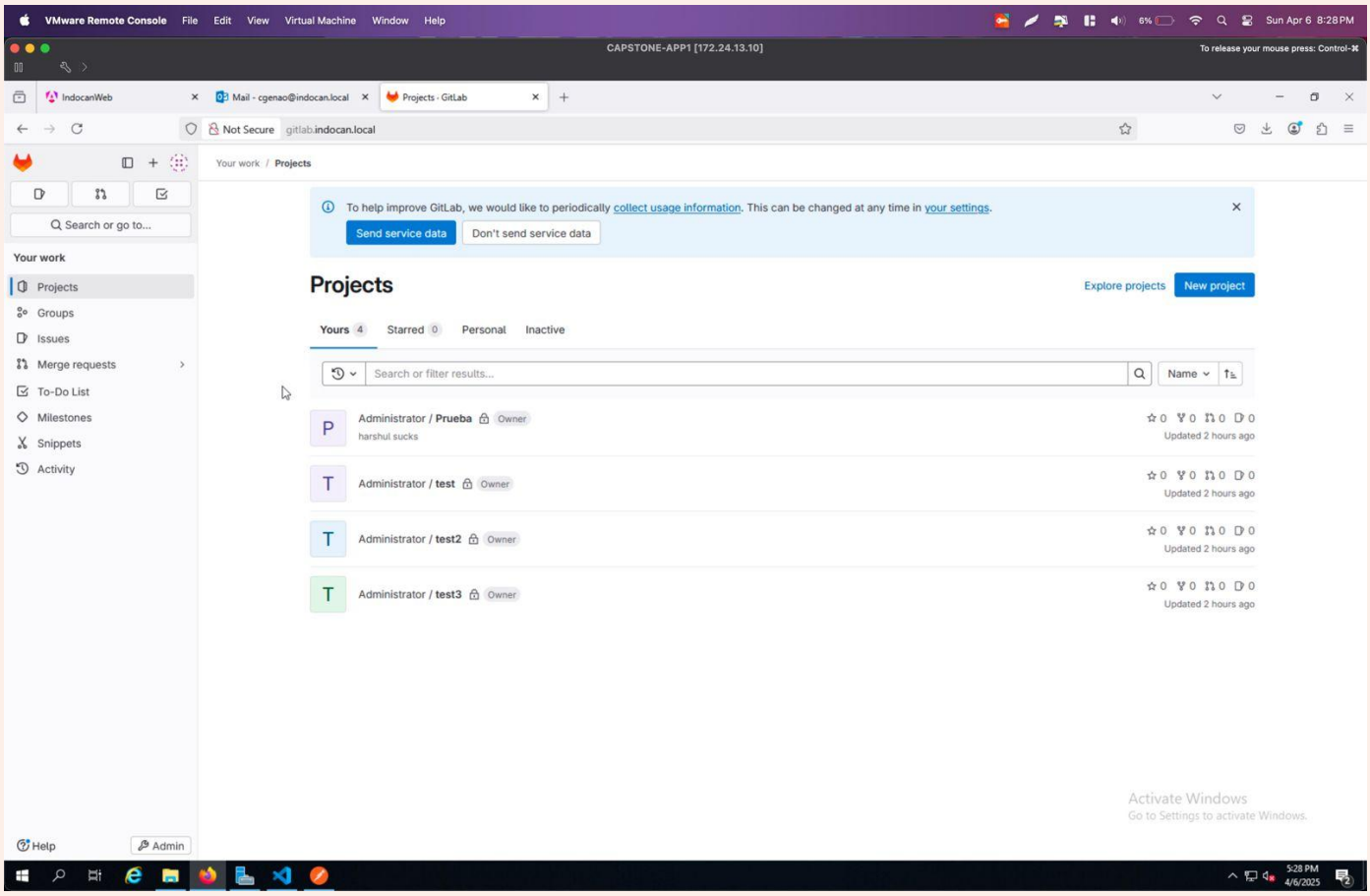
GITLAB

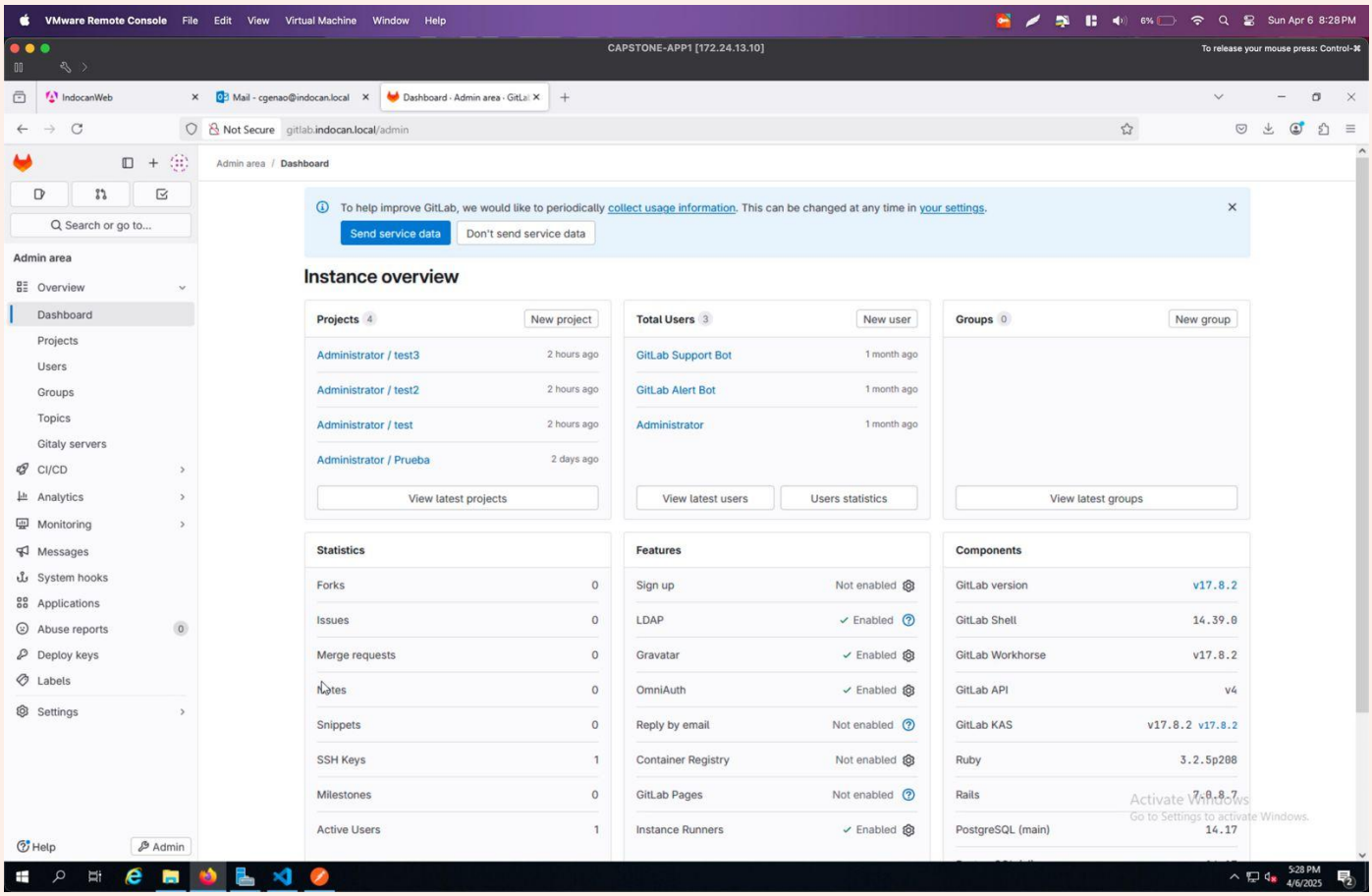

Dashboard
Welcome Back, Christopher Genao!
Viewing as: Domain Admins
Home
Ticketing
Git
Monitoring
GitLab Repository Browser
Browse and view your GitLab repositories from here.
+ Create Repository
Search repositories...
test3
No description
Private
main
test2
No description
Private
main
test
No description
Private
main
Prueba
harshul sucks
Private
main

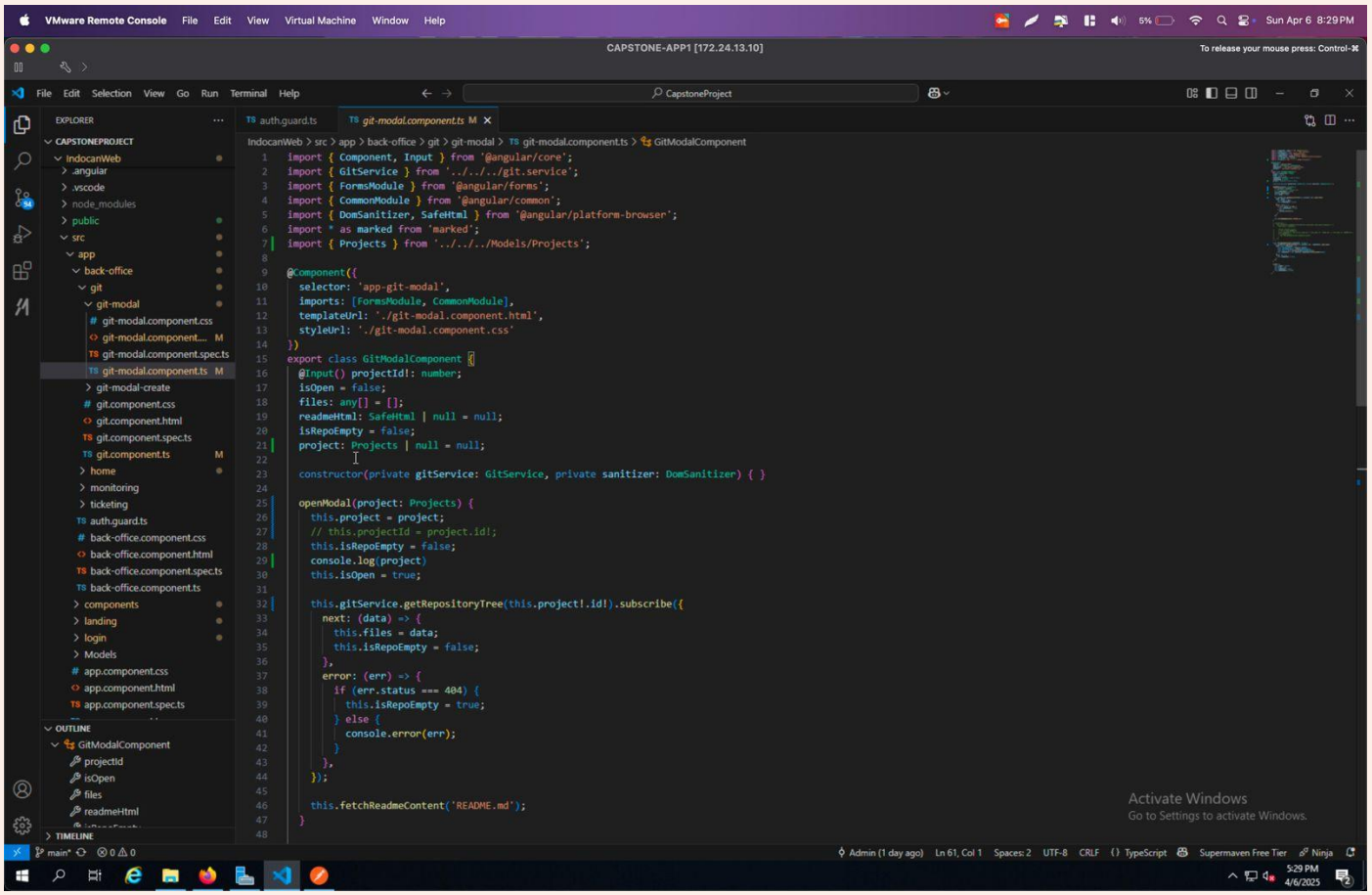

Dashboard
Welcome Back, Christopher Genao!
Viewing as: Domain Admins
Repository Files
API
TREE
IndocanWeb
TREE
CapstoneProject.code-workspace
BLOB
README.md
BLOB
README
Foobar
Foobar is a Python library for dealing with word pluralization.
Installation
Use the package manager pip to install foobar.
pip install foobar
Usage
import foobar
# returns 'words'
foobar.pluralize('word')
# returns 'geese'
foobar.pluralize('goose')

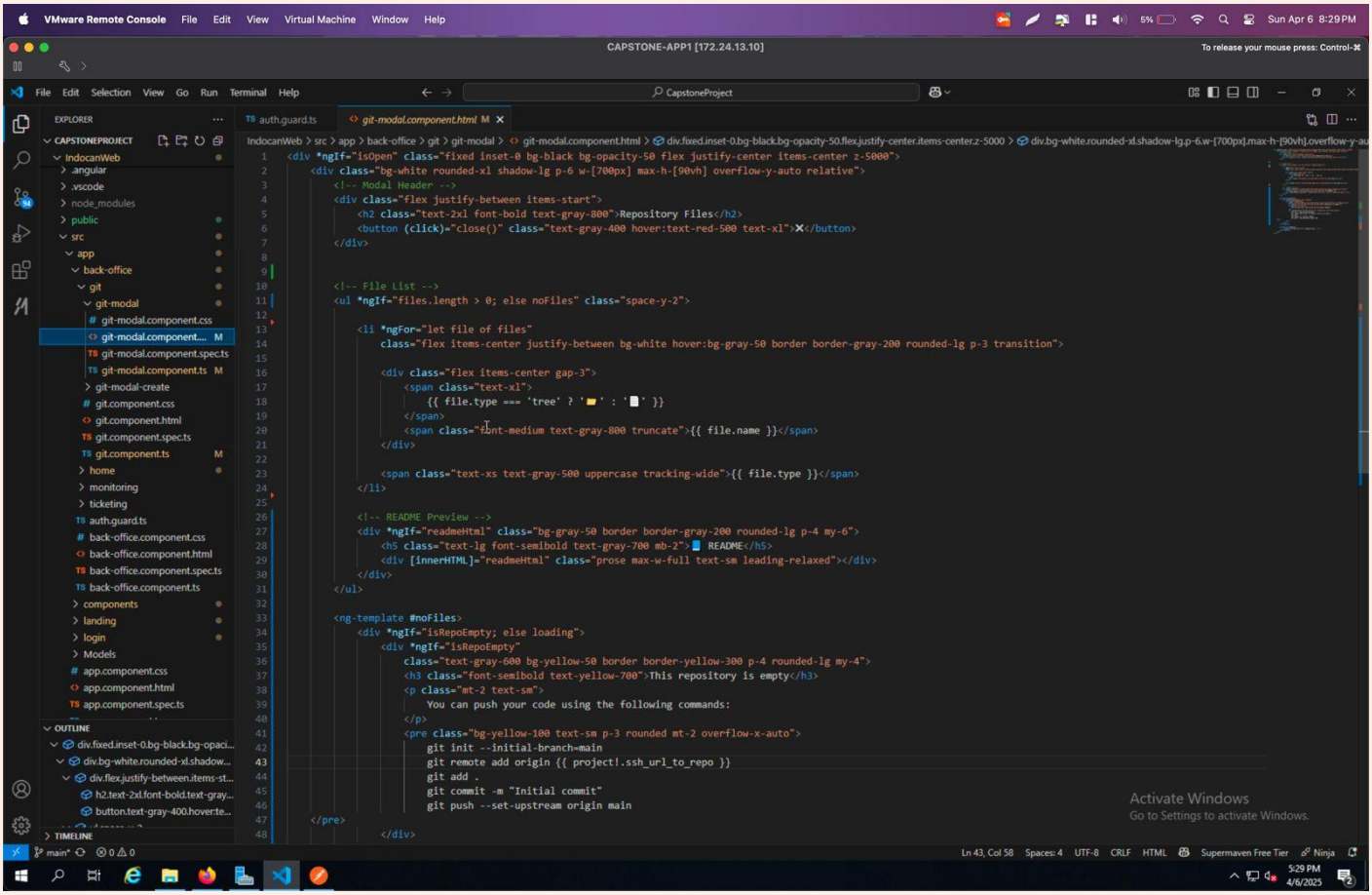

Dashboard
Welcome Back, Christopher Genao!
Viewing as: Domain Admins
GitLab Repository Browser
Browse and view your GitLab repositories from here.
+ Create Repository
Search repositories...
Home
Ticketing
Git
Monitoring
test3
No description
Private
test
No description
Private
main
Prueba
harshul sucks
Private
Repository Files
This repository is empty
You can push your code using the following commands:
git init --initial-branch=main
git remote add origin git@gitlab.indocan.local:root/
git add .
git commit -m "Initial commit"
git push --set-upstream origin main
Activate Windows
Go to Settings to activate Windows.


Dashboard
Welcome Back, Christopher Genao!
Viewing as: Domain Admins
GitLab Repository Browser
Browse and view your GitLab repositories from here.
+ Create Repository
pr
Home
Ticketing
Git
Monitoring
Prueba
harshul sucks
Private
main
Create New Repository
Repository Name
Description (optional)
Private
Create Repository
Activate Windows
Go to Settings to activate Windows.


VMware Remote Console File Edit View Virtual Machine Window Help
Sun Apr 6 8:28 PM
CAPSTONE-APP1 [172.24.13.10]
To release your mouse press: Control-⌘
IndocanWeb
Mail - cgenao@indocan.local
Projects · GitLab
Not Secure gitlab.indocan.local
Your work / Projects
Search or go to...
Your work
Projects
Groups
Issues
Merge requests
To-Do List
Milestones
Snippets
Activity
To help improve GitLab, we would like to periodically collect usage information. This can be changed at any time in your settings.
Send service data
Don't send service data
Projects
Explore projects
New project
Yours 4
Starred 0
Personal
Inactive
Search or filter results...
Name
Administrator / Prueba Owner
harshul sucks
Administrator / test Owner
Administrator / test2 Owner
Administrator / test3 Owner
Updated 2 hours ago
Activate Windows
Go to Settings to activate Windows.
Help
Admin
5:28 PM
4/6/2025


VMware Remote Console File Edit View Virtual Machine Window Help
Sun Apr 6 8:28 PM
CAPSTONE-APP1 [172.24.13.10]
To release your mouse press: Control-⌘
IndocanWeb
Mail - cgenao@indocan.local
Dashboard · Admin area · GitLab
Not Secure gitlab.indocan.local/admin
Admin area / Dashboard
Search or go to...
Admin area
Overview
Dashboard
Projects
Users
Groups
Topics
Gitaly servers
CI/CD
Analytics
Monitoring
Messages
System hooks
Applications
Abuse reports
Deploy keys
Labels
Settings
To help improve GitLab, we would like to periodically collect usage information. This can be changed at any time in your settings.
Send service data
Don't send service data
Instance overview
Projects 4
New project
Administrator / test3 2 hours ago
Administrator / test2 2 hours ago
Administrator / test 2 hours ago
Administrator / Prueba 2 days ago
View latest projects
Total Users 3
New user
GitLab Support Bot 1 month ago
GitLab Alert Bot 1 month ago
Administrator 1 month ago
View latest users
Users statistics
Groups 0
New group
View latest groups
Statistics
Forks 0
Issues 0
Merge requests 0
Notes 0
Snippets 0
SSH Keys 1
Milestones 0
Active Users 1
Features
Sign up Not enabled
LDAP Enabled
Gravatar Enabled
OmniAuth Enabled
Reply by email Not enabled
Container Registry Not enabled
GitLab Pages Not enabled
Instance Runners Enabled
Components
GitLab version v17.8.2
GitLab Shell 14.39.0
GitLab Workhorse v17.8.2
GitLab API v4
GitLab KAS v17.8.2 v17.8.2
Ruby 3.2.5p208
Rails 7.0.8.7
PostgreSQL (main) 14.17
Activate Windows
Go to Settings to activate Windows.
Help
Admin
5:28 PM
4/6/2025


VMware Remote Console File Edit View Virtual Machine Window Help
CAPSTONE-APP1 [172.24.13.10]
To release your mouse press: Control-⌘
File Edit Selection View Go Run Terminal Help
CapstoneProject
EXPLORER
CAPSTONEPROJECT
IndocanWeb
auth.guard.ts
git-modal.component.ts M
IndocanWeb > src > app > back-office > git > git-modal > git-modal.component.ts > GitModalComponent
import { Component, Input } from '@angular/core';
import { GitService } from '../../../git.service';
import { FormsModule } from '@angular/forms';
import { CommonModule } from '@angular/common';
import { DomSanitizer, SafeHtml } from '@angular/platform-browser';
import * as marked from 'marked';
import { Projects } from '../../../Models/Projects';
@Component({
selector: 'app-git-modal',
imports: [FormsModule, CommonModule],
templateUrl: './git-modal.component.html',
styleUrl: './git-modal.component.css'
})
export class GitModalComponent {
@Input() projectId!: number;
isOpen = false;
files: any[] = [];
readmeHtml: SafeHtml | null = null;
isRepoEmpty = false;
project: Projects | null = null;
constructor(private gitService: GitService, private sanitizer: DomSanitizer) { }
openModal(project: Projects) {
this.project = project;
// this.projectId = project.id!;
this.isRepoEmpty = false;
console.log(project)
this.isOpen = true;
this.gitService.getRepositoryTree(this.project!.id!).subscribe({
next: (data) => {
this.files = data;
this.isRepoEmpty = false;
},
error: (err) => {
if (err.status === 404) {
this.isRepoEmpty = true;
} else {
console.error(err);
}
},
});
this.fetchReadmeContent('README.md');
}
OUTLINE
GitModalComponent
projectId
isOpen
files
readmeHtml
TIMELINE
Activate Windows
Go to Settings to activate Windows.
Admin (1 day ago) Ln 61, Col 1 Spaces: 2 UTF-8 CRLF TypeScript Supermaven Free Tier Ninja
5:29 PM
4/6/2025


VMware Remote Console File Edit View Virtual Machine Window Help
CAPSTONE-APP1 [172.24.13.10]
To release your mouse press: Control-⌘
File Edit Selection View Go Run Terminal Help
CapstoneProject
EXPLORER
CAPSTONEPROJECT
auth.guard.ts
git-modal.component.html M
IndocanWeb > src > app > back-office > git > git-modal > git-modal.component.html > div.fixed.inset-0.bg-black.bg-opacity-50.flex.justify-center.items-center.z-5000 > div.bg-white.rounded-xl.shadow-lg.p-6.w-[700px].max-h-[90vh].overflow-y-au
<div *ngIf="isOpen" class="fixed inset-0 bg-black bg-opacity-50 flex justify-center items-center z-5000">

<!-- Modal Header -->

<h2 class="text-2xl font-bold text-gray-800">Repository Files</h2>
<button (click)="close()" class="text-gray-400 hover:text-red-500 text-xl">✕</button>

<!-- File List -->
<ul *ngIf="files.length > 0; else noFiles" class="space-y-2">
<li *ngFor="let file of files"
class="flex items-center justify-between bg-white hover:bg-gray-50 border border-gray-200 rounded-lg p-3 transition">

{{ file.type === 'tree' ? '📁' : '📄' }}

{{ file.name }}

{{ file.type }}

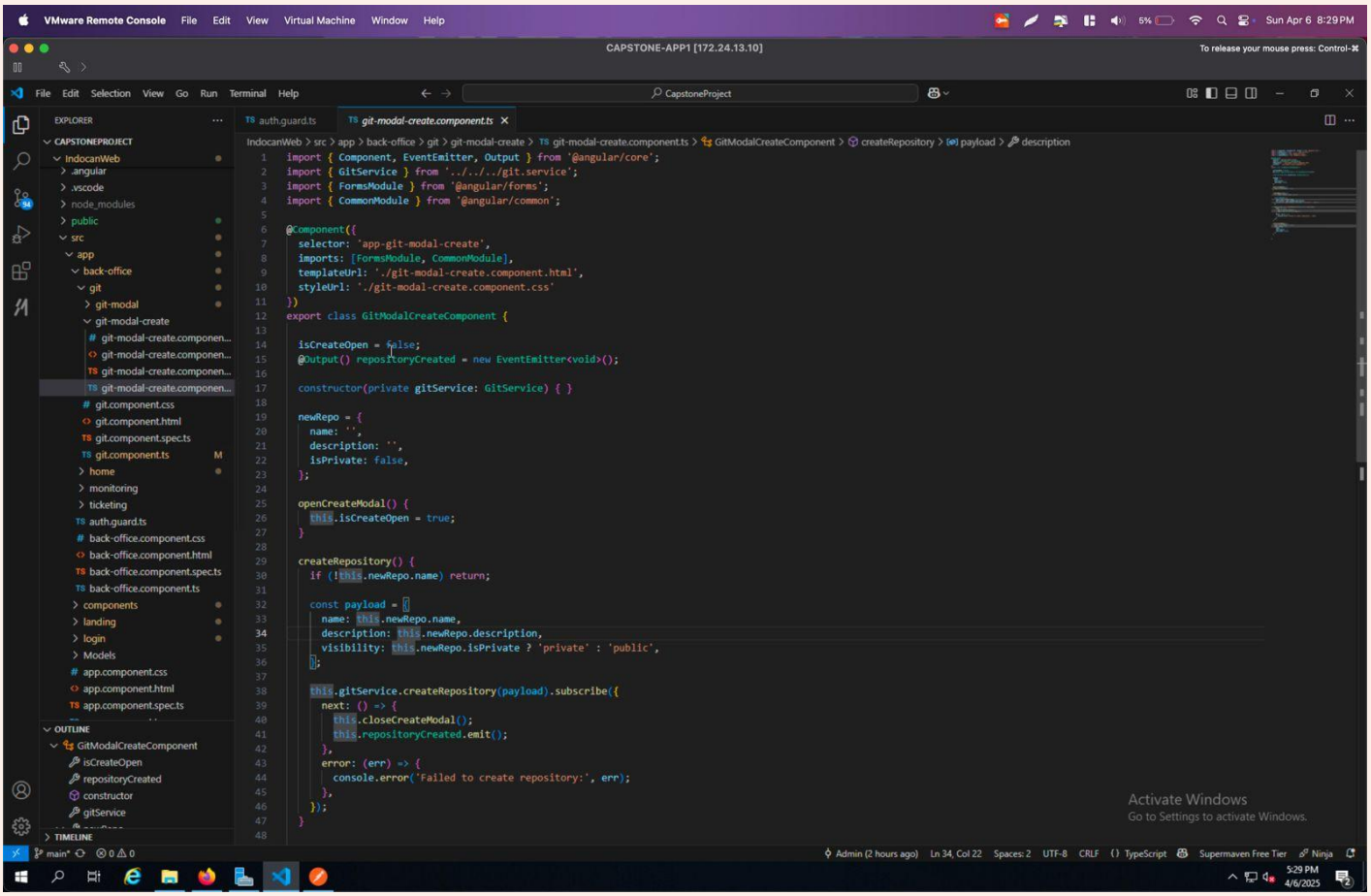
<!-- README Preview -->
<div *ngIf="readmeHtml" class="bg-gray-50 border border-gray-200 rounded-lg p-4 my-6">
<h5 class="text-lg font-semibold text-gray-700 mb-2">📘 README</h5>
<div [innerHTML]="readmeHtml" class="prose max-w-full text-sm leading-relaxed">

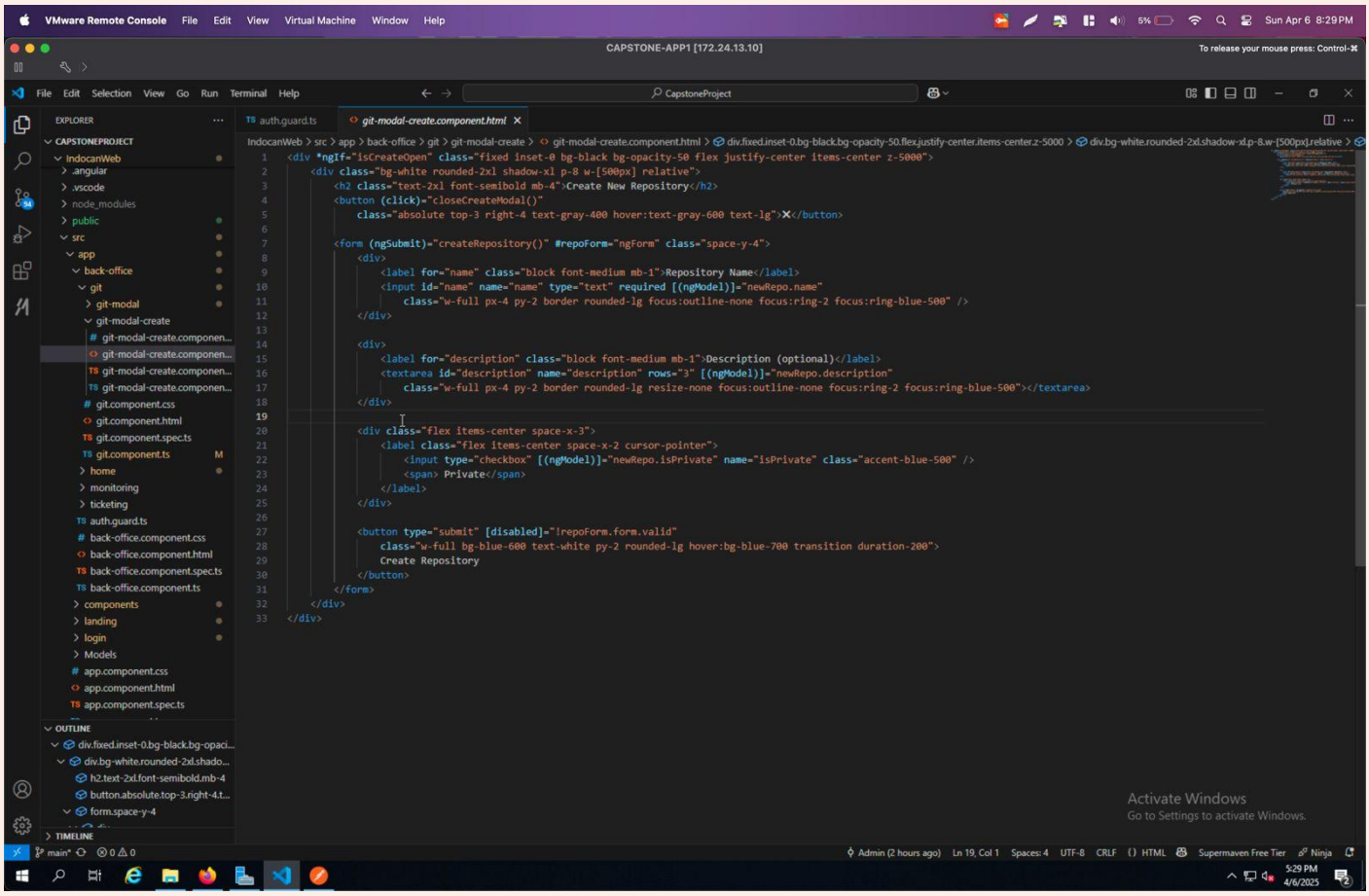
<ng-template #noFiles>
<div *ngIf="isRepoEmpty; else loading">
<div *ngIf="isRepoEmpty"
class="text-gray-600 bg-yellow-50 border border-yellow-300 p-4 rounded-lg my-4">
<h3 class="font-semibold text-yellow-700">This repository is empty</h3>

You can push your code using the following commands:

<pre class="bg-yellow-100 text-sm p-3 rounded mt-2 overflow-x-auto">
git init --initial-branch=main
git remote add origin {{ project!.ssh_url_to_repo }}
git add .
git commit -m "Initial commit"
git push --set-upstream origin main
</pre>

OUTLINE
TIMELINE
Activate Windows
Go to Settings to activate Windows.
Ln 43, Col 58 Spaces: 4 UTF-8 CRLF HTML Supermaven Free Tier Ninja
5:29 PM
4/6/2025


VMware Remote Console File Edit View Virtual Machine Window Help
CAPSTONE-APP1 [172.24.13.10]
To release your mouse press: Control-⌘
File Edit Selection View Go Run Terminal Help
CapstoneProject
EXPLORER
CAPSTONEPROJECT
IndocanWeb
auth.guard.ts
git-modal-create.component.ts
IndocanWeb > src > app > back-office > git > git-modal-create > git-modal-create.component.ts > GitModalCreateComponent > createRepository > payload > description
import { Component, EventEmitter, Output } from '@angular/core';
import { GitService } from '../../../git.service';
import { FormsModule } from '@angular/forms';
import { CommonModule } from '@angular/common';
@Component({
selector: 'app-git-modal-create',
imports: [FormsModule, CommonModule],
templateUrl: './git-modal-create.component.html',
styleUrl: './git-modal-create.component.css'
})
export class GitModalCreateComponent {
isCreateOpen = false;
@Output() repositoryCreated = new EventEmitter<void>();
constructor(private gitService: GitService) { }
newRepo = {
name: '',
description: '',
isPrivate: false,
};
openCreateModal() {
this.isCreateOpen = true;
}
createRepository() {
if (!this.newRepo.name) return;
const payload = {
name: this.newRepo.name,
description: this.newRepo.description,
visibility: this.newRepo.isPrivate ? 'private' : 'public',
};
this.gitService.createRepository(payload).subscribe({
next: () => {
this.closeCreateModal();
this.repositoryCreated.emit();
},
error: (err) => {
console.error('Failed to create repository:', err);
},
});
}
OUTLINE
GitModalCreateComponent
isCreateOpen
repositoryCreated
constructor
gitService
TIMELINE
Activate Windows
Go to Settings to activate Windows.
Admin (2 hours ago) Ln 34, Col 22 Spaces: 2 UTF-8 CRLF TypeScript Supermaven Free Tier Ninja
5:29 PM
4/6/2025


VMware Remote Console File Edit View Virtual Machine Window Help
CAPSTONE-APP1 [172.24.13.10]
To release your mouse press: Control-⌘
File Edit Selection View Go Run Terminal Help
CapstoneProject
EXPLORER
CAPSTONEPROJECT
IndocanWeb
auth.guard.ts
git-modal-create.component.html
IndocanWeb > src > app > back-office > git > git-modal-create > git-modal-create.component.html > div.fixed.inset-0.bg-black.bg-opacity-50.flex.justify-center.items-center.z-5000 > div.bg-white.rounded-2xl.shadow-xl.p-8.w-[500px].relative
<div *ngIf="isCreateOpen" class="fixed inset-0 bg-black bg-opacity-50 flex justify-center items-center z-5000">

<h2 class="text-2xl font-semibold mb-4">Create New Repository</h2>
<button (click)="closeCreateModal()"
class="absolute top-3 right-4 text-gray-400 hover:text-gray-600 text-lg">✕</button>
<form (ngSubmit)="createRepository()" #repoForm="ngForm" class="space-y-4">

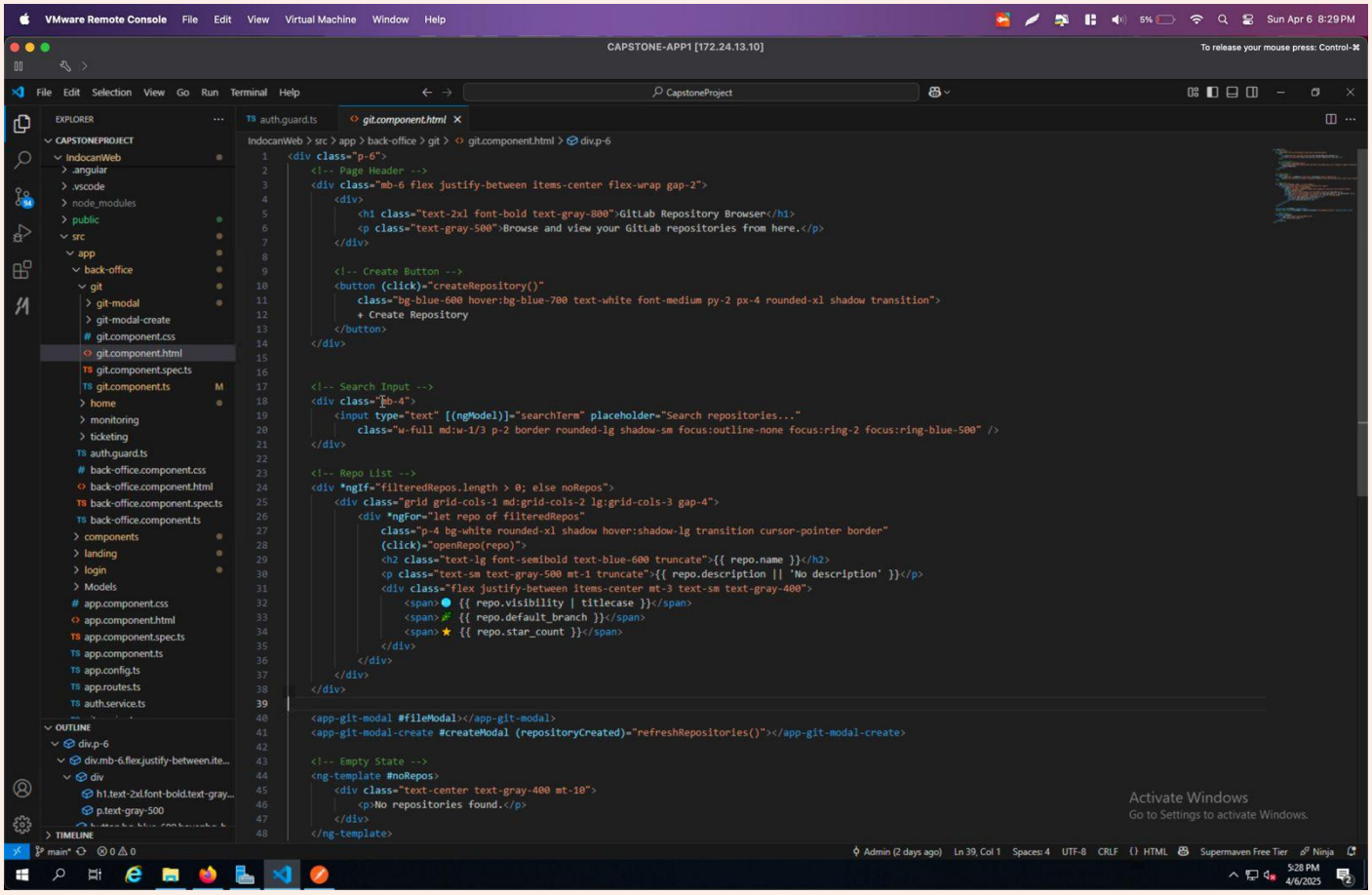
<label for="name" class="block font-medium mb-1">Repository Name</label>
<input id="name" name="name" type="text" required [(ngModel)]="newRepo.name"
class="w-full px-4 py-2 border rounded-lg focus:outline-none focus:ring-2 focus:ring-blue-500" />

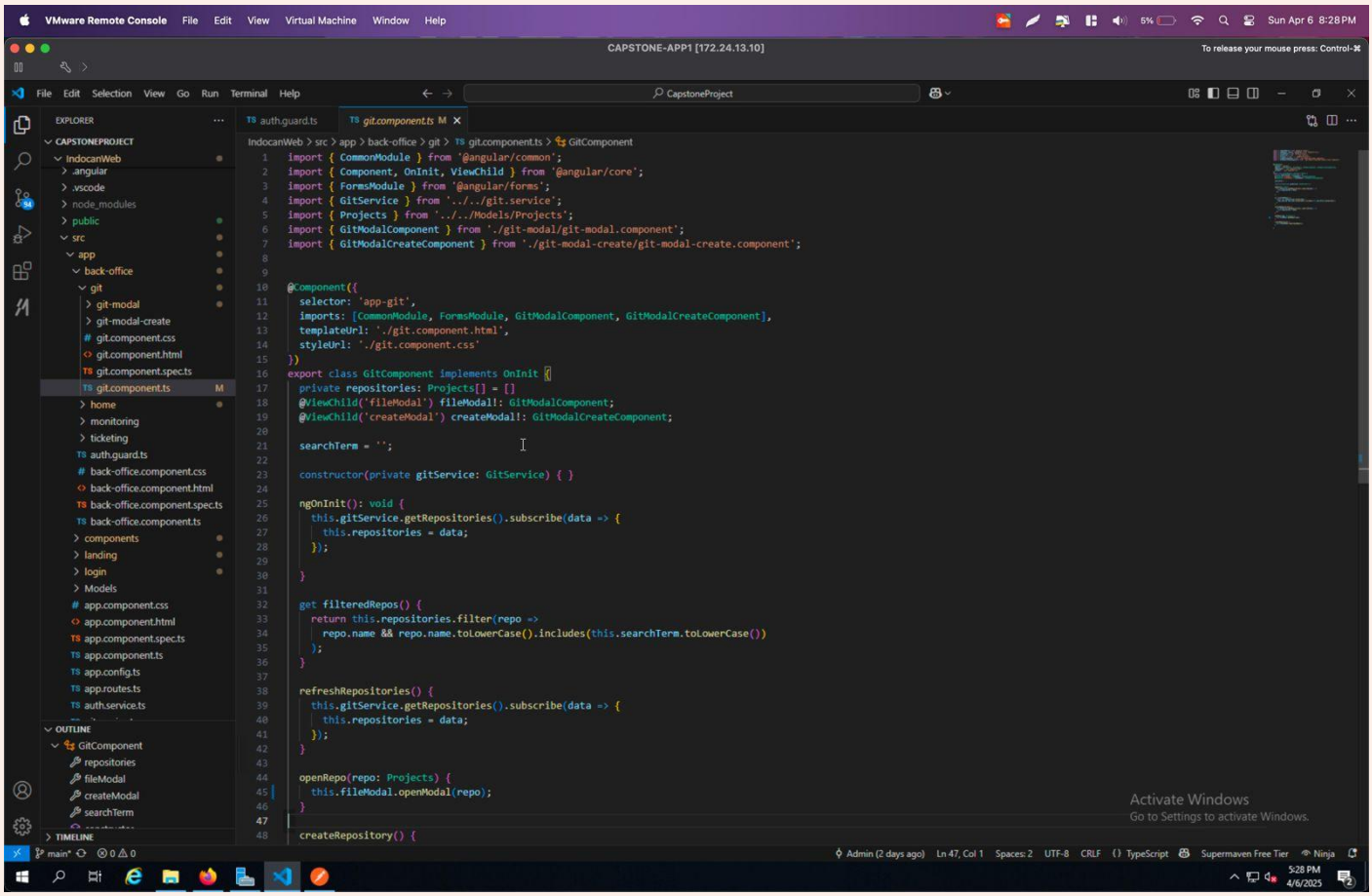
<label for="description" class="block font-medium mb-1">Description (optional)</label>
<textarea id="description" name="description" rows="3" [(ngModel)]="newRepo.description"
class="w-full px-4 py-2 border rounded-lg resize-none focus:outline-none focus:ring-2 focus:ring-blue-500"></textarea>

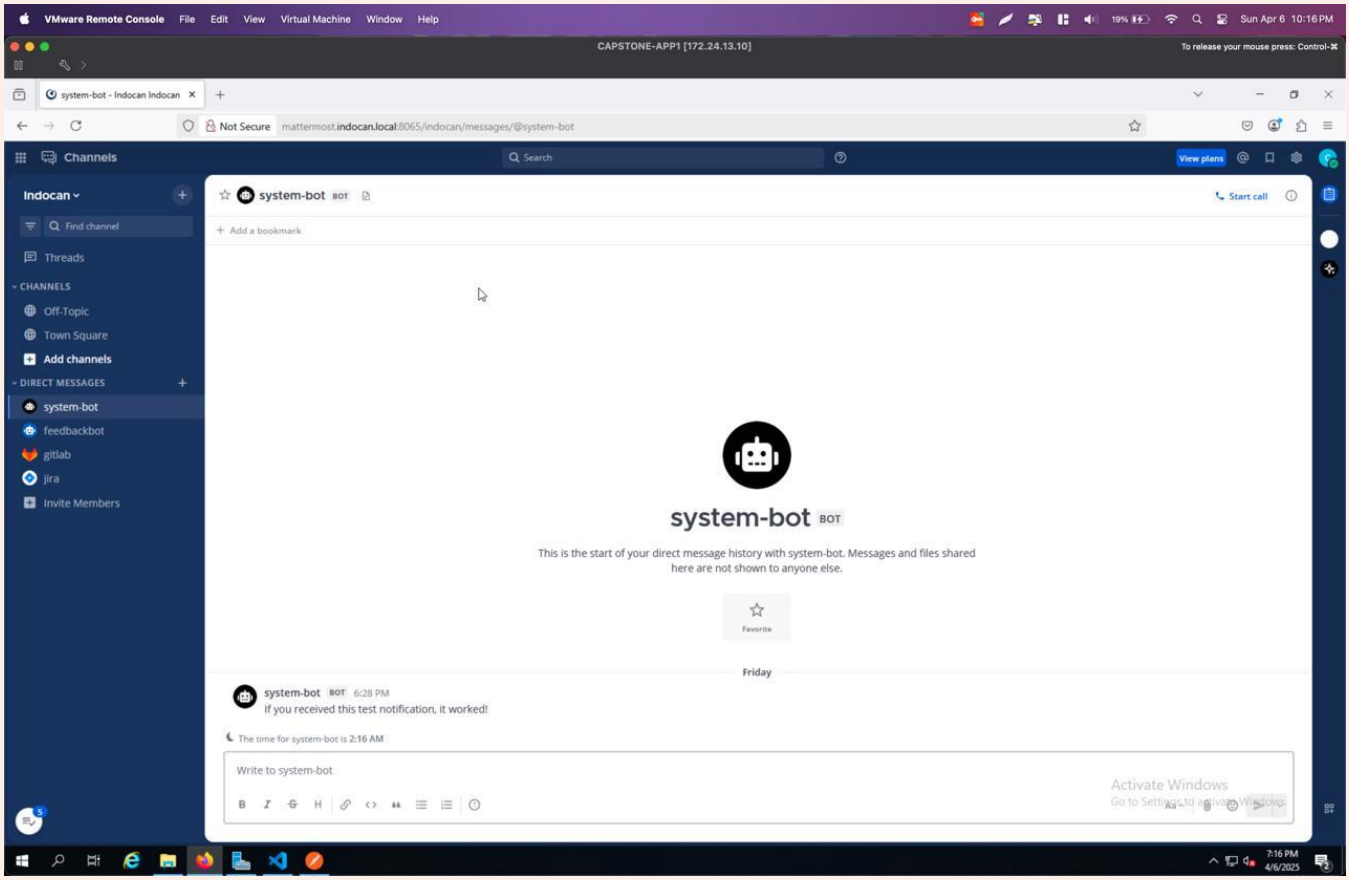
<label class="flex items-center space-x-2 cursor-pointer">
<input type="checkbox" [(ngModel)]="newRepo.isPrivate" name="isPrivate" class="accent-blue-500" />
 Private
</label>

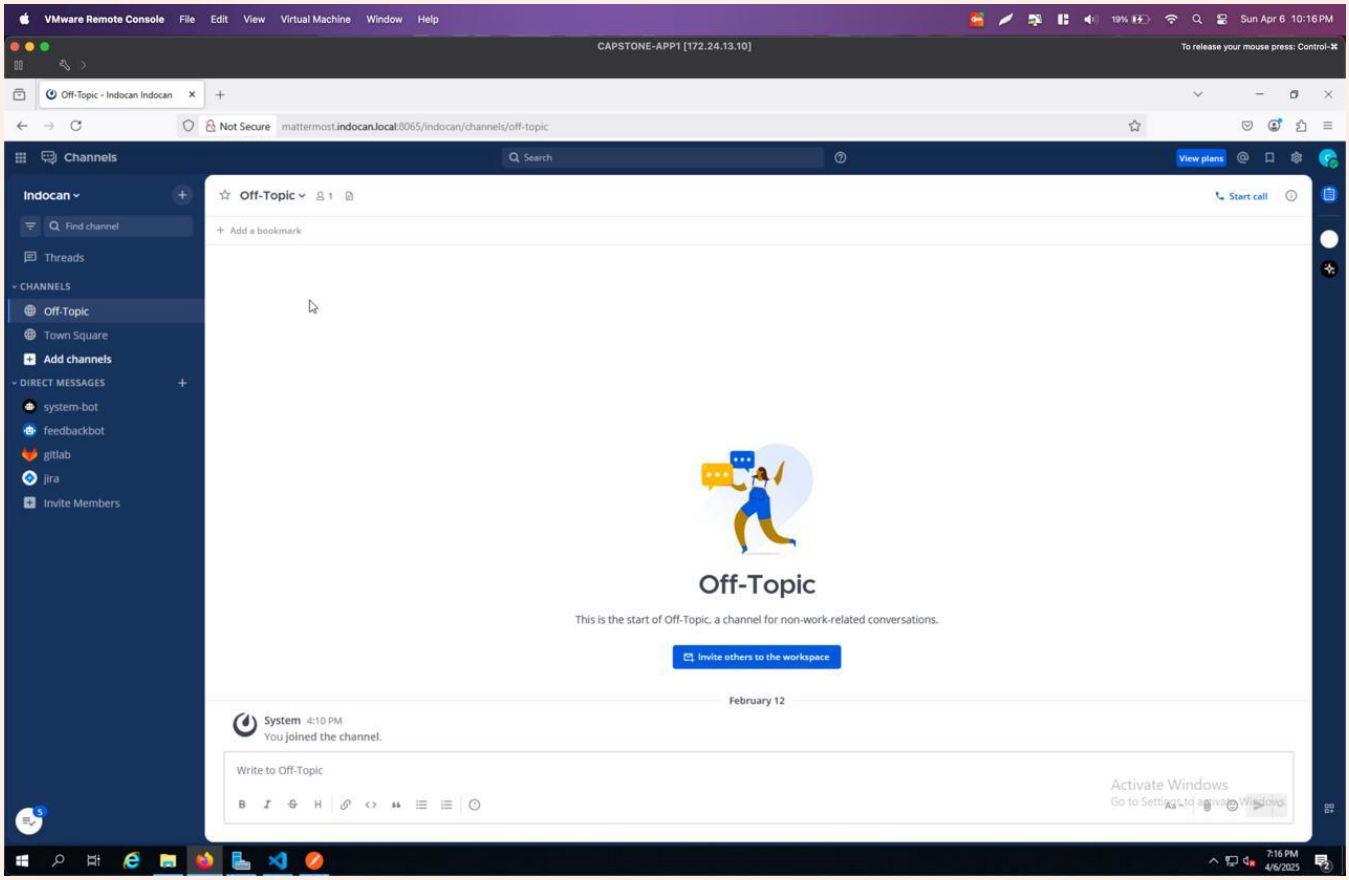
<button type="submit" [disabled]="!repoForm.form.valid"
class="w-full bg-blue-600 text-white py-2 rounded-lg hover:bg-blue-700 transition duration-200">
Create Repository
</button>
</form>

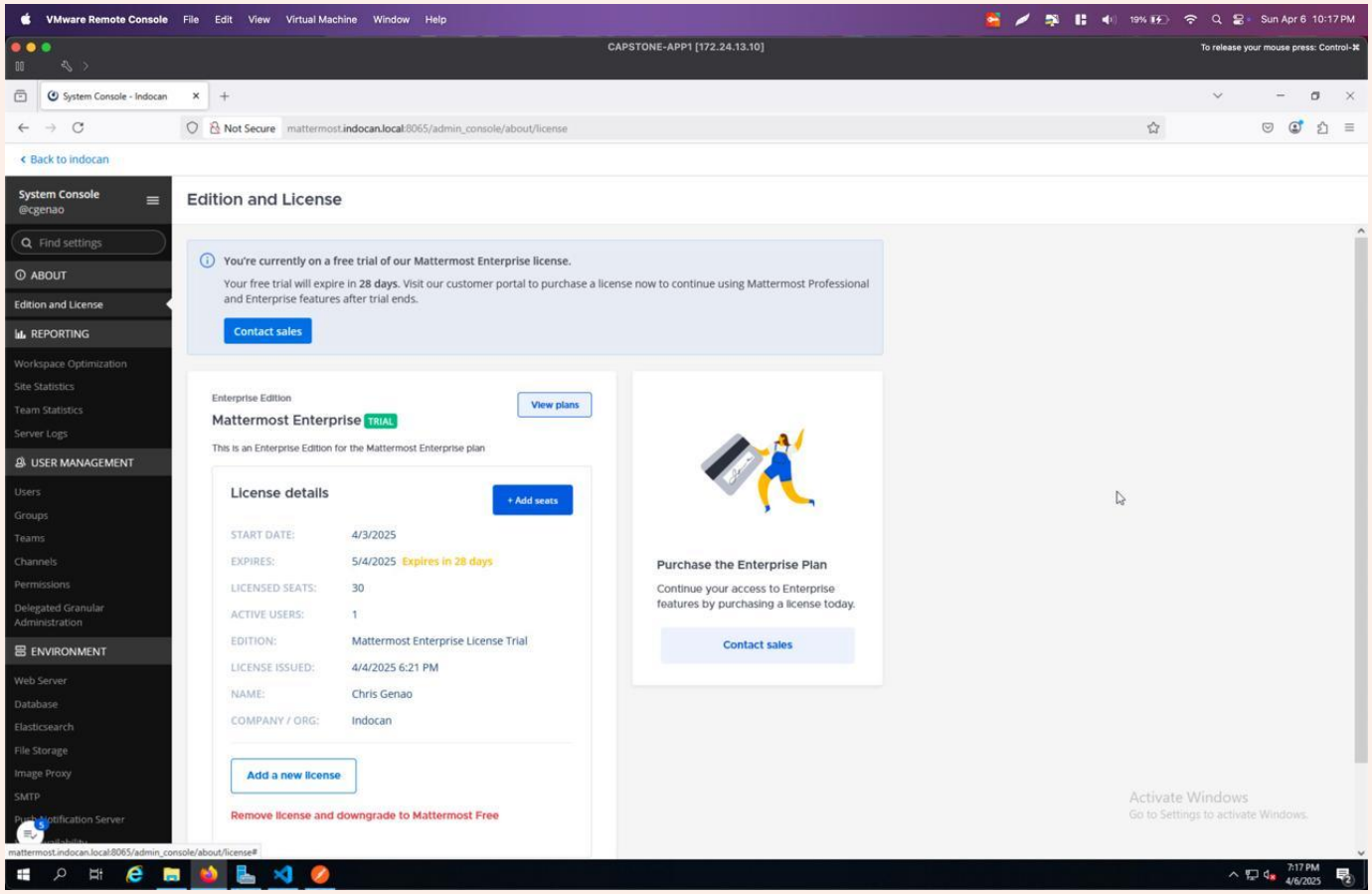
OUTLINE
TIMELINE
Activate Windows
Go to Settings to activate Windows.
Admin (2 hours ago) Ln 19, Col 1 Spaces: 4 UTF-8 CRLF HTML Supermaven Free Tier Ninja
5:29 PM
4/6/2025

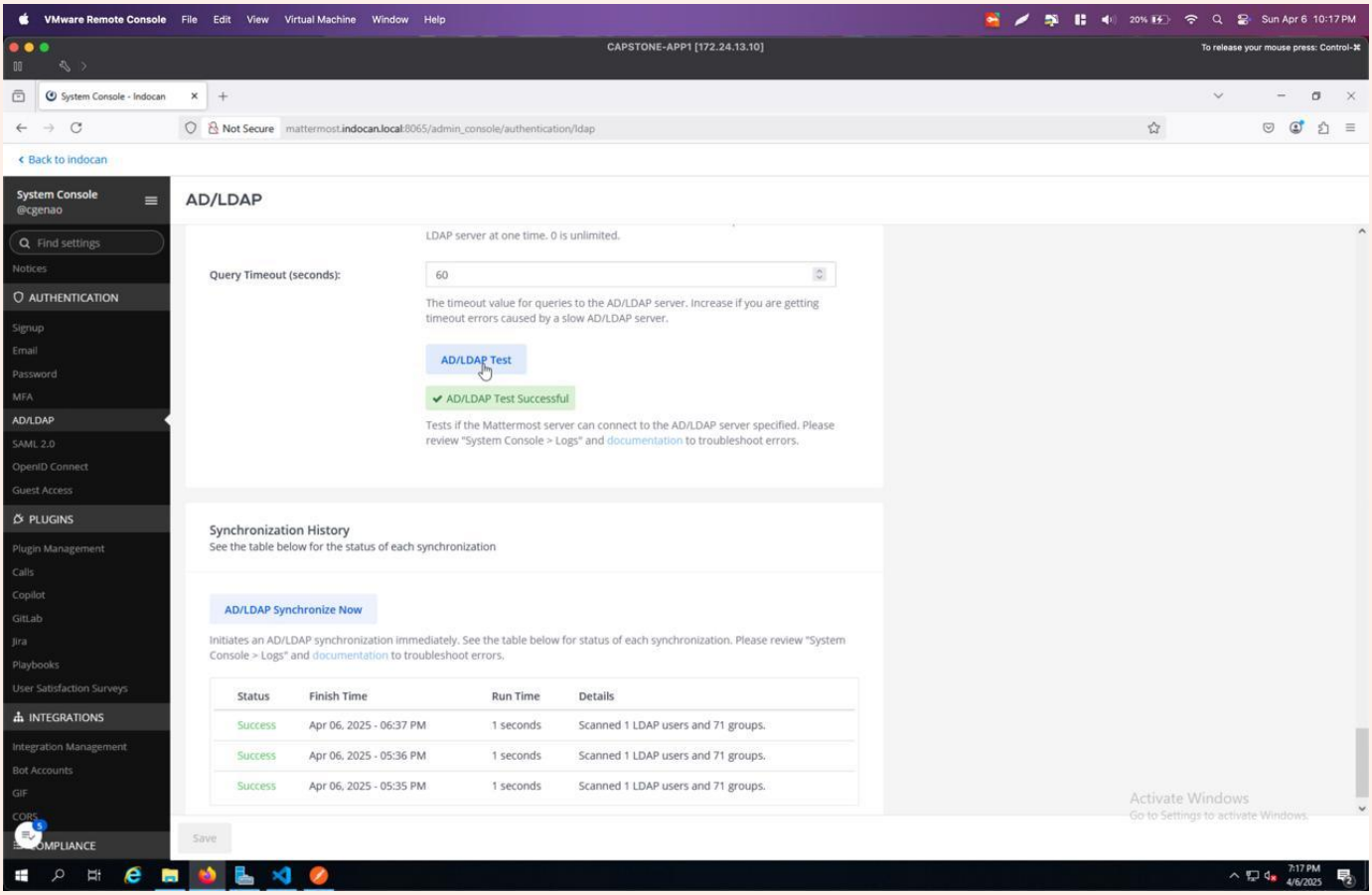

CAPSTONE-APP1 [172.24.13.10]
git.component.html
<h1 class="text-2xl font-bold text-gray-800">GitLab Repository Browser</h1>
Browse and view your GitLab repositories from here.
+ Create Repository
<app-git-modal #fileModal></app-git-modal>
<app-git-modal-create #createModal (repositoryCreated)="refreshRepositories()"></app-git-modal-create>
No repositories found.
Activate Windows
Go to Settings to activate Windows.

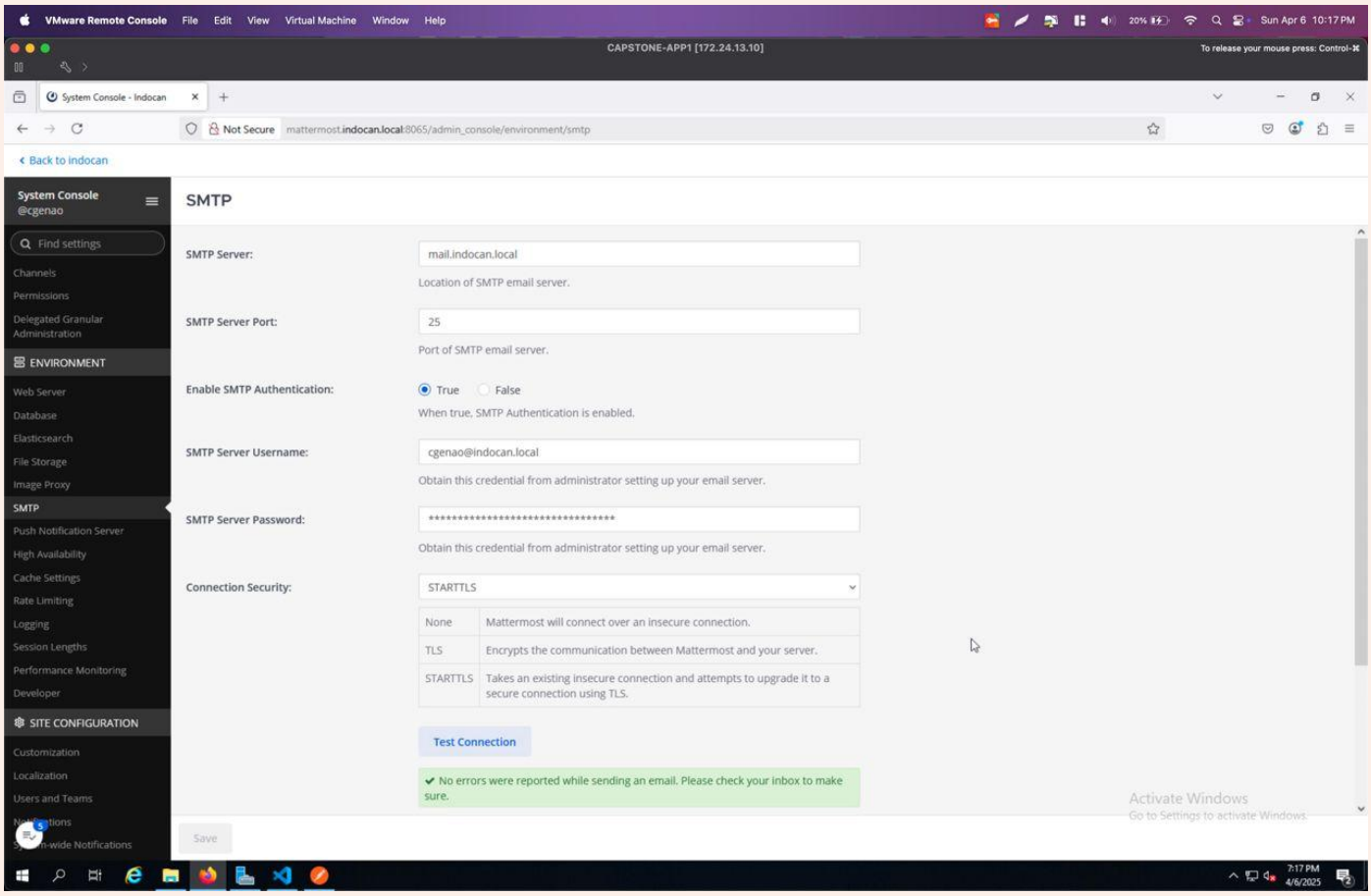

CAPSTONE-APP1 [172.24.13.10]
git.component.ts
import { CommonModule } from '@angular/common';
import { Component, OnInit, ViewChild } from '@angular/core';
import { FormsModule } from '@angular/forms';
import { GitService } from '../../git.service';
import { Projects } from '../../Models/Projects';
import { GitModalComponent } from './git-modal/git-modal.component';
import { GitModalCreateComponent } from './git-modal-create/git-modal-create.component';
@Component({
selector: 'app-git',
imports: [CommonModule, FormsModule, GitModalComponent, GitModalCreateComponent],
templateUrl: './git.component.html',
styleUrl: './git.component.css'
})
export class GitComponent implements OnInit {
private repositories: Projects[] = []
@ViewChild('fileModal') fileModal!: GitModalComponent;
@ViewChild('createModal') createModal!: GitModalCreateComponent;
searchTerm = '';
constructor(private gitService: GitService) { }
ngOnInit(): void {
this.gitService.getRepositories().subscribe(data => {
this.repositories = data;
});
}
get filteredRepos() {
return this.repositories.filter(repo =>
repo.name && repo.name.toLowerCase().includes(this.searchTerm.toLowerCase())
);
}
refreshRepositories() {
this.gitService.getRepositories().subscribe(data => {
this.repositories = data;
});
}
openRepo(repo: Projects) {
this.fileModal.openModal(repo);
}
createRepository() {
Activate Windows
Go to Settings to activate Windows.

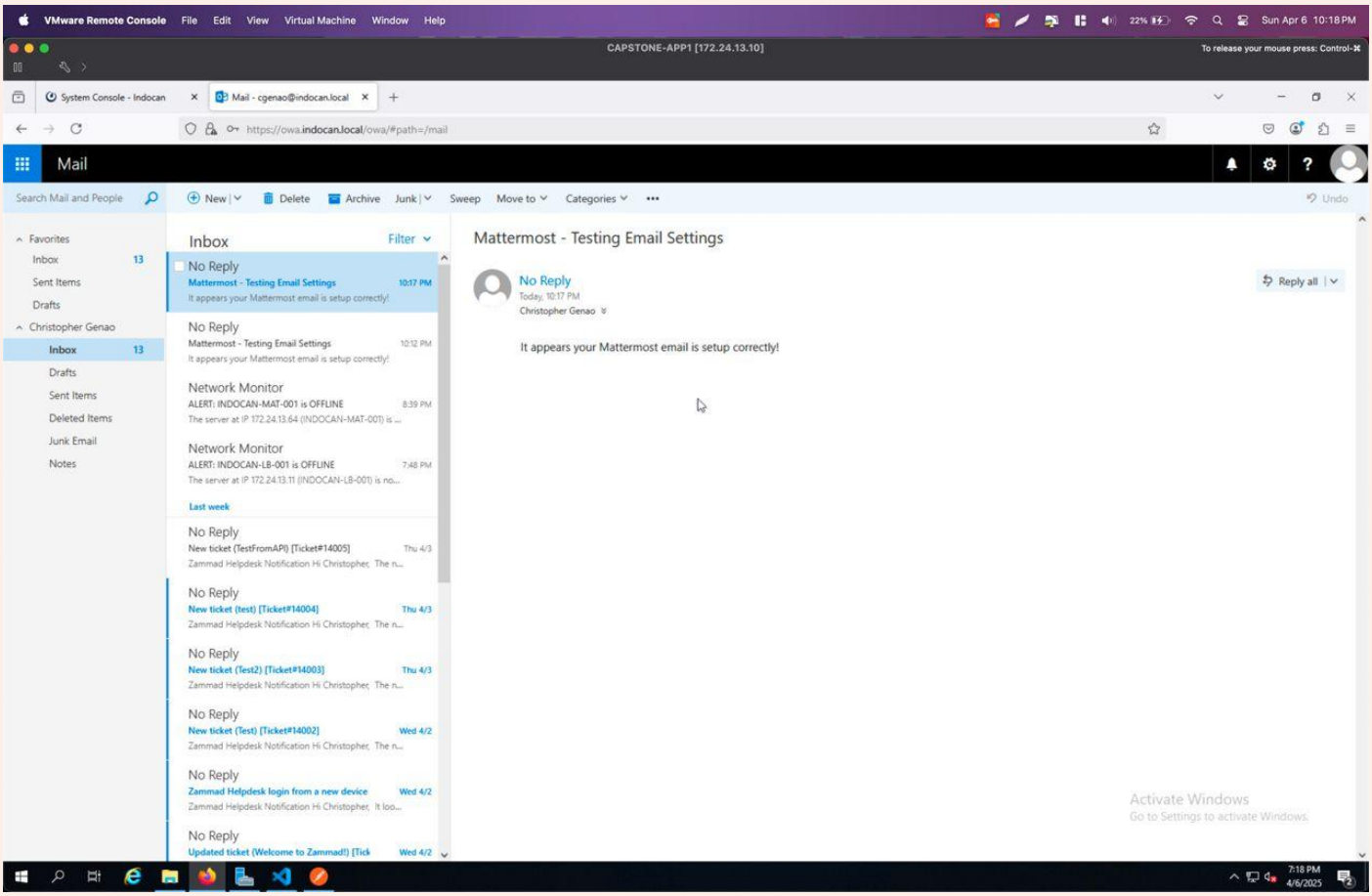

VMware Remote Console File Edit View Virtual Machine Window Help
CAPSTONE-APP1 [172.24.13.10]
system-bot - Indocan Indocan
Not Secure mattermost.indocan.local:8065/indocan/messages/@system-bot
Channels
Search
View plans
Indocan
Find channel
Threads
CHANNELS
Off-Topic
Town Square
Add channels
DIRECT MESSAGES
system-bot
feedbackbot
gitlab
jira
Invite Members
system-bot BOT
Start call
Add a bookmark
system-bot
BOT
This is the start of your direct message history with system-bot. Messages and files shared here are not shown to anyone else.
Favorite
Friday
system-bot BOT 6:28 PM
If you received this test notification, it worked!
The time for system-bot is 2:16 AM
Write to system-bot
Activate Windows
7:16 PM
4/6/2025


VMware Remote Console File Edit View Virtual Machine Window Help
CAPSTONE-APP1 [172.24.13.10]
Off-Topic - Indocan Indocan
Not Secure mattermost.indocan.local:8065/indocan/channels/off-topic
Channels
Search
View plans
Indocan
Find channel
Threads
CHANNELS
Off-Topic
Town Square
Add channels
DIRECT MESSAGES
system-bot
feedbackbot
gitlab
jira
Invite Members
Off-Topic
Start call
Add a bookmark
Off-Topic
This is the start of Off-Topic, a channel for non-work-related conversations.
Invite others to the workspace
February 12
System 4:10 PM
You joined the channel.
Write to Off-Topic
Activate Windows
7:16 PM
4/6/2025


VMware Remote Console File Edit View Virtual Machine Window Help
CAPSTONE-APP1 [172.24.13.10]
To release your mouse press: Control-⌘
System Console - Indocan
Not Secure mattermost.indocan.local:8065/admin_console/about/license
Back to indocan
System Console
@cgenao
Find settings
ABOUT
Edition and License
REPORTING
Workspace Optimization
Site Statistics
Team Statistics
Server Logs
USER MANAGEMENT
Users
Groups
Teams
Channels
Permissions
Delegated Granular Administration
ENVIRONMENT
Web Server
Database
Elasticsearch
File Storage
Image Proxy
SMTP
Edition and License
You're currently on a free trial of our Mattermost Enterprise license.
Your free trial will expire in 28 days. Visit our customer portal to purchase a license now to continue using Mattermost Professional and Enterprise features after trial ends.
Contact sales
Enterprise Edition
Mattermost Enterprise TRIAL
View plans
This is an Enterprise Edition for the Mattermost Enterprise plan
License details
+ Add seats
START DATE: 4/3/2025
EXPIRES: 5/4/2025 Expires in 28 days
LICENSED SEATS: 30
ACTIVE USERS: 1
EDITION: Mattermost Enterprise License Trial
LICENSE ISSUED: 4/4/2025 6:21 PM
NAME: Chris Genao
COMPANY / ORG: Indocan
Add a new license
Remove license and downgrade to Mattermost Free
Purchase the Enterprise Plan
Continue your access to Enterprise features by purchasing a license today.
Contact sales
Activate Windows
Go to Settings to activate Windows.
7:17 PM
4/6/2025


VMware Remote Console File Edit View Virtual Machine Window Help
CAPSTONE-APP1 [172.24.13.10]
To release your mouse press: Control-⌘
System Console - Indocan
Not Secure mattermost.indocan.local:8065/admin_console/authentication/ldap
Back to indocan
System Console
@cgenao
Find settings
Notices
AUTHENTICATION
Signup
Email
Password
MFA
AD/LDAP
SAML 2.0
OpenID Connect
Guest Access
PLUGINS
Plugin Management
Calls
Copilot
GitLab
Jira
Playbooks
User Satisfaction Surveys
INTEGRATIONS
Integration Management
Bot Accounts
GIF
CORS
COMPLIANCE
AD/LDAP
LDAP server at one time. 0 is unlimited.
Query Timeout (seconds): 60
The timeout value for queries to the AD/LDAP server. Increase if you are getting timeout errors caused by a slow AD/LDAP server.
AD/LDAP Test
AD/LDAP Test Successful
Tests if the Mattermost server can connect to the AD/LDAP server specified. Please review "System Console > Logs" and documentation to troubleshoot errors.
Synchronization History
See the table below for the status of each synchronization
AD/LDAP Synchronize Now
Initiates an AD/LDAP synchronization immediately. See the table below for status of each synchronization. Please review "System Console > Logs" and documentation to troubleshoot errors.
Status Finish Time Run Time Details
Success Apr 06, 2025 - 06:37 PM 1 seconds Scanned 1 LDAP users and 71 groups.
Success Apr 06, 2025 - 05:36 PM 1 seconds Scanned 1 LDAP users and 71 groups.
Success Apr 06, 2025 - 05:35 PM 1 seconds Scanned 1 LDAP users and 71 groups.
Save
Activate Windows
Go to Settings to activate Windows.
7:17 PM
4/6/2025


VMware Remote Console File Edit View Virtual Machine Window Help
CAPSTONE-APP1 [172.24.13.10]
System Console - Indocan
Not Secure mattermost.indocan.local:8065/admin_console/environment/smtp
Back to indocan
System Console
@cgenao
Find settings
Channels
Permissions
Delegated Granular Administration
ENVIRONMENT
Web Server
Database
Elasticsearch
File Storage
Image Proxy
SMTP
Push Notification Server
High Availability
Cache Settings
Rate Limiting
Logging
Session Lengths
Performance Monitoring
Developer
SITE CONFIGURATION
Customization
Localization
Users and Teams
Notifications
System-wide Notifications
SMTP
SMTP Server: mail.indocan.local
Location of SMTP email server.
SMTP Server Port: 25
Port of SMTP email server.
Enable SMTP Authentication: True False
When true, SMTP Authentication is enabled.
SMTP Server Username: cgenao@indocan.local
Obtain this credential from administrator setting up your email server.
SMTP Server Password:
Obtain this credential from administrator setting up your email server.
Connection Security: STARTTLS
None Mattermost will connect over an insecure connection.
TLS Encrypts the communication between Mattermost and your server.
STARTTLS Takes an existing insecure connection and attempts to upgrade it to a secure connection using TLS.
Test Connection
No errors were reported while sending an email. Please check your inbox to make sure.
Save
Activate Windows
Go to Settings to activate Windows.


VMware Remote Console File Edit View Virtual Machine Window Help
CAPSTONE-APP1 [172.24.13.10]
System Console - Indocan
Mail - cgenao@indocan.local
https://owa.indocan.local/owa/#path=/mail
Mail
Search Mail and People
New Delete Archive Junk Sweep Move to Categories
Undo
Favorites
Inbox 13
Sent Items
Drafts
Christopher Genao
Inbox 13
Drafts
Sent Items
Deleted Items
Junk Email
Notes
Inbox
Filter
No Reply
Mattermost - Testing Email Settings 10:17 PM
It appears your Mattermost email is setup correctly!
No Reply
Mattermost - Testing Email Settings 10:12 PM
It appears your Mattermost email is setup correctly!
Network Monitor
ALERT: INDOCAN-MAT-001 is OFFLINE 8:39 PM
Network Monitor
ALERT: INDOCAN-LB-001 is OFFLINE 7:48 PM
Last week
No Reply
New ticket (TestFromAPI) [Ticket#14005] Thu 4/3
No Reply
New ticket (test) [Ticket#14004] Thu 4/3
No Reply
New ticket (Test2) [Ticket#14003] Thu 4/3
No Reply
New ticket (Test) [Ticket#14002] Wed 4/2
No Reply
Zammad Helpdesk login from a new device Wed 4/2
No Reply
Updated ticket (Welcome to Zammad!) [Tick Wed 4/2
Mattermost - Testing Email Settings
No Reply
Today, 10:17 PM
Christopher Genao
Reply all
It appears your Mattermost email is setup correctly!
Activate Windows
Go to Settings to activate Windows.

CONCLUSION

In conclusion, this project has effectively delivered a secure, scalable, and high-performance IT infrastructure to support INDOCAN's strategic expansion into New York and Hong Kong. Through the implementation of server virtualization, robust hardware components, and a resilient, segmented network architecture, we've created a cohesive environment that ensures business continuity, operational efficiency, and seamless global collaboration.

A suite of powerful tools has been integrated to enhance productivity and streamline workflows across both offices. Mattermost supports real-time team communication, Zammad automates IT support through efficient ticketing, Guacamole provides secure browser-based remote desktop access, and the self-hosted GitLab offers centralized version control and collaborative development capabilities vital for INDOCAN's web development operations.

Centralized authentication via Active Directory, secure VPN connectivity, and role-based access control further reinforce the infrastructure's reliability and security. Designed with scalability in mind, this infrastructure lays a future-ready foundation that empowers INDOCAN's technical teams and end users alike, enabling innovation and growth in a globally distributed environment.

REFERENCES

Apache guacamole manual — Apache Guacamole Manual V1.5.5. (n.d.). https://guacamole.apache.org/doc/gug/

Configuration settings - Mattermost documentation. (n.d.). https://docs.mattermost.com/configure/configuration-settings.html

Configure the Webserver — Zammad System Documentation documentation. (n.d.). https://docs.zammad.org/en/latest/getting-started/configure-webserver.html

Sophos Ltd. (n.d.). *Configure Exchange and all other clients - Sophos Central Admin*. https://docs.sophos.com/central/customer/help/en-us/ManageYourProducts/EmailSecurity/SophosGateway/ExternalServices/ConfigureExchange/index.html

Virtualizing pfSense Software with VMware vSphere / ESXi | pfSense Documentation. (n.d.). https://docs.netgate.com/pfsense/en/latest/recipes/virtualize-esxi.html

Yang, K., Ellingwood, J., & Juell, K. (2024, February 28). *How to install and configure GitLab on Ubuntu*. DigitalOcean. https://www.digitalocean.com/community/tutorials/how-to-install-and-configure-gitlab-on-ubuntu