

## Problem

- Organizations need secure, well-documented IT systems.
  - Without a structured approach, systems risk being unprepared for real-world operations
- Often, there's a struggle to turn initial designs into operational solutions.
- This project addresses the challenge of refining security controls, backup procedures, and policies into a cohesive documentation suite.
- Key issues include: aligning technical systems with governance policies, ensuring security and resilience

## Our Solution

- The team will take a structured approach to create a unified documentation suite that will make it easy to disseminate enterprise policies to users, while securely keeping features limited on as-needed basis, and with preventive security measures in already in place. To accomplish this:
- Detailed artifacts for security controls, backup and recovery, and organizational policies will be designed
  - Standardized templates and a shared repository ensure consistency and accessibility
  - A compliance matrix aligns controls with policies using MFA, onboarding checks, VPG-gating, and role-based access

## Demo & Screenshots

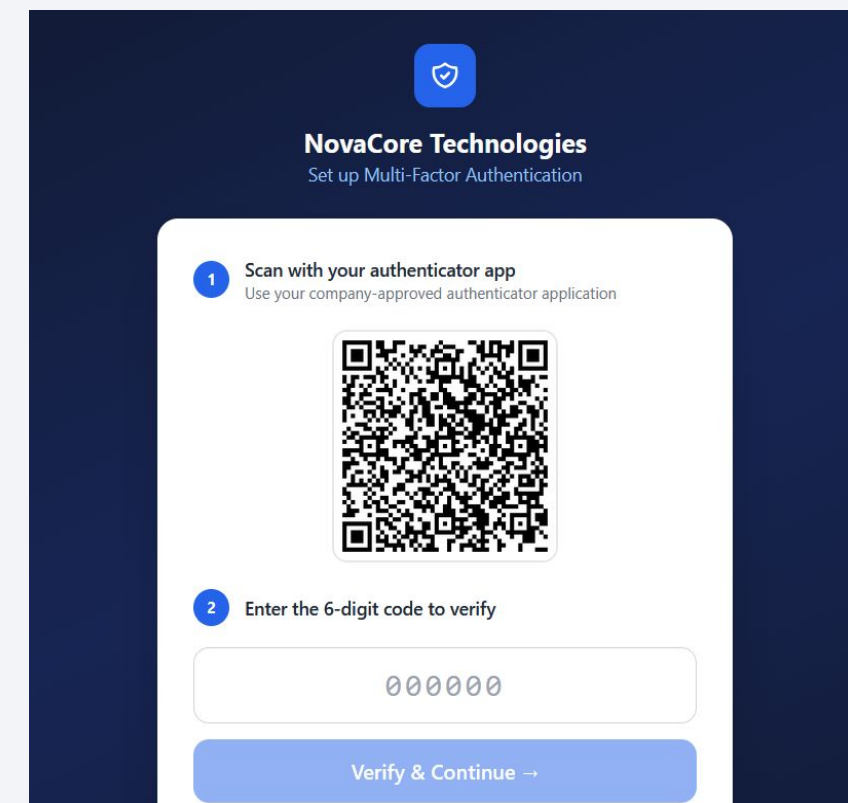


Figure 1 (Top): First-Time MFA Set-up Page

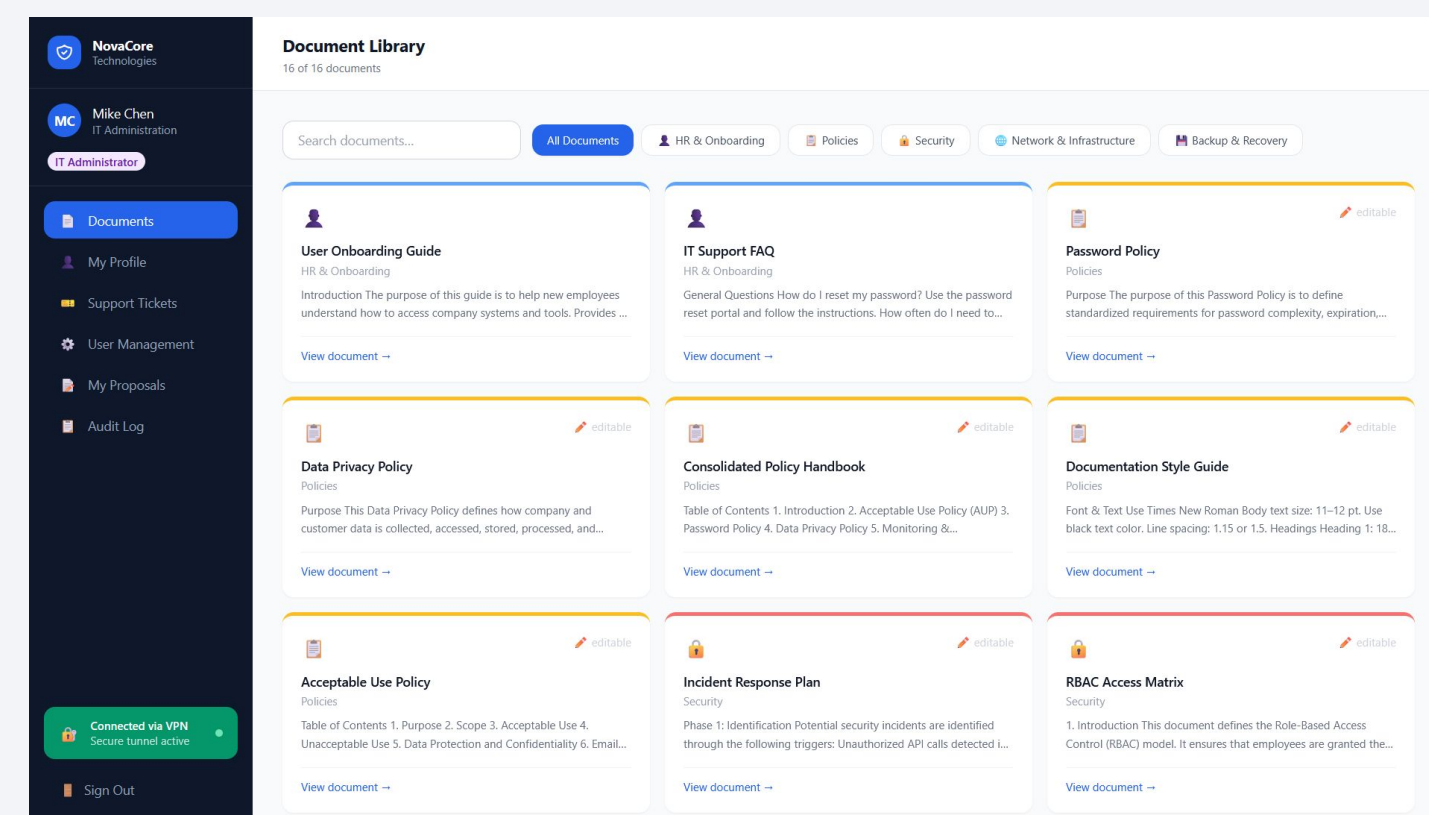


Figure 2 (Middle): Employee Document Library, IT Admin Access Level

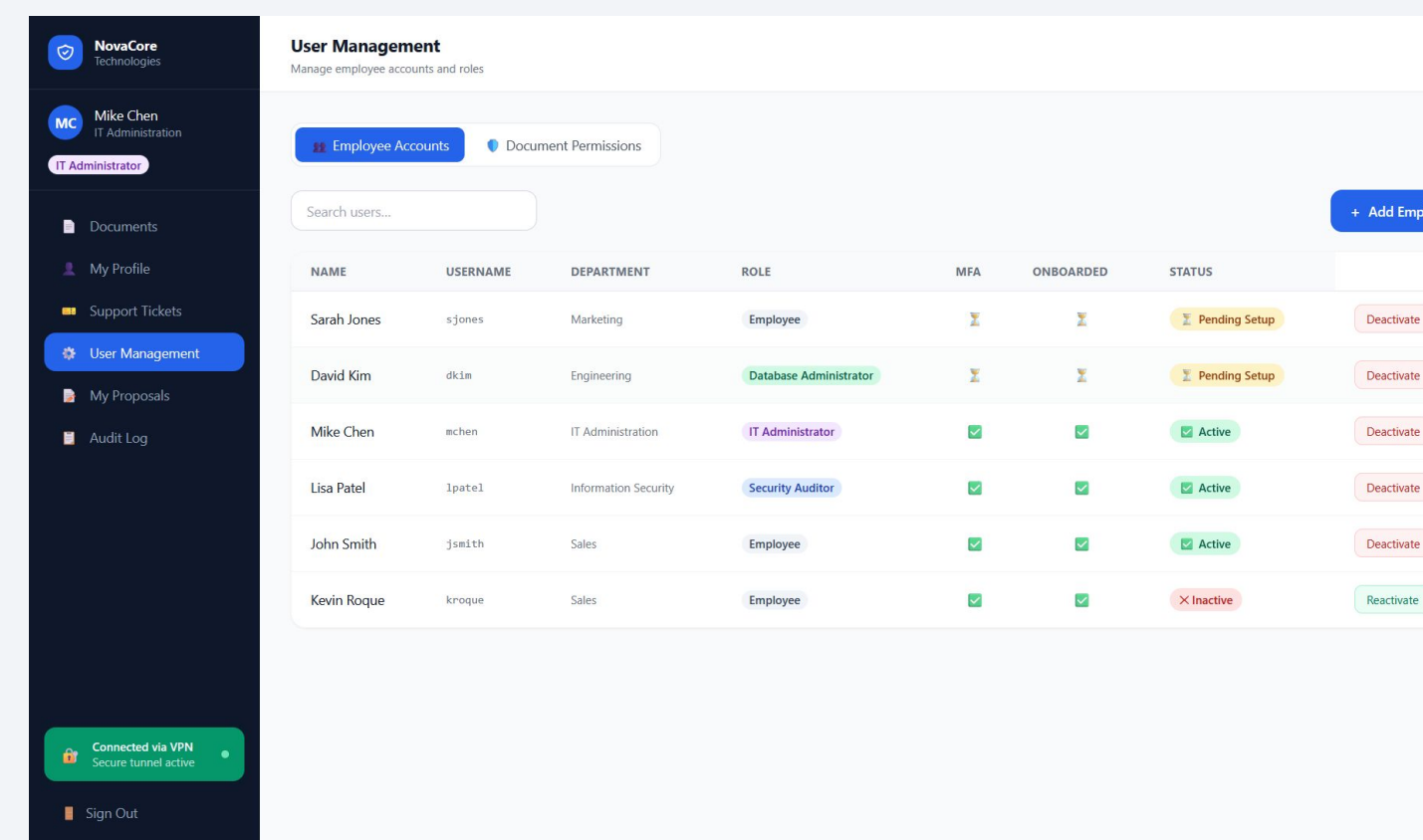
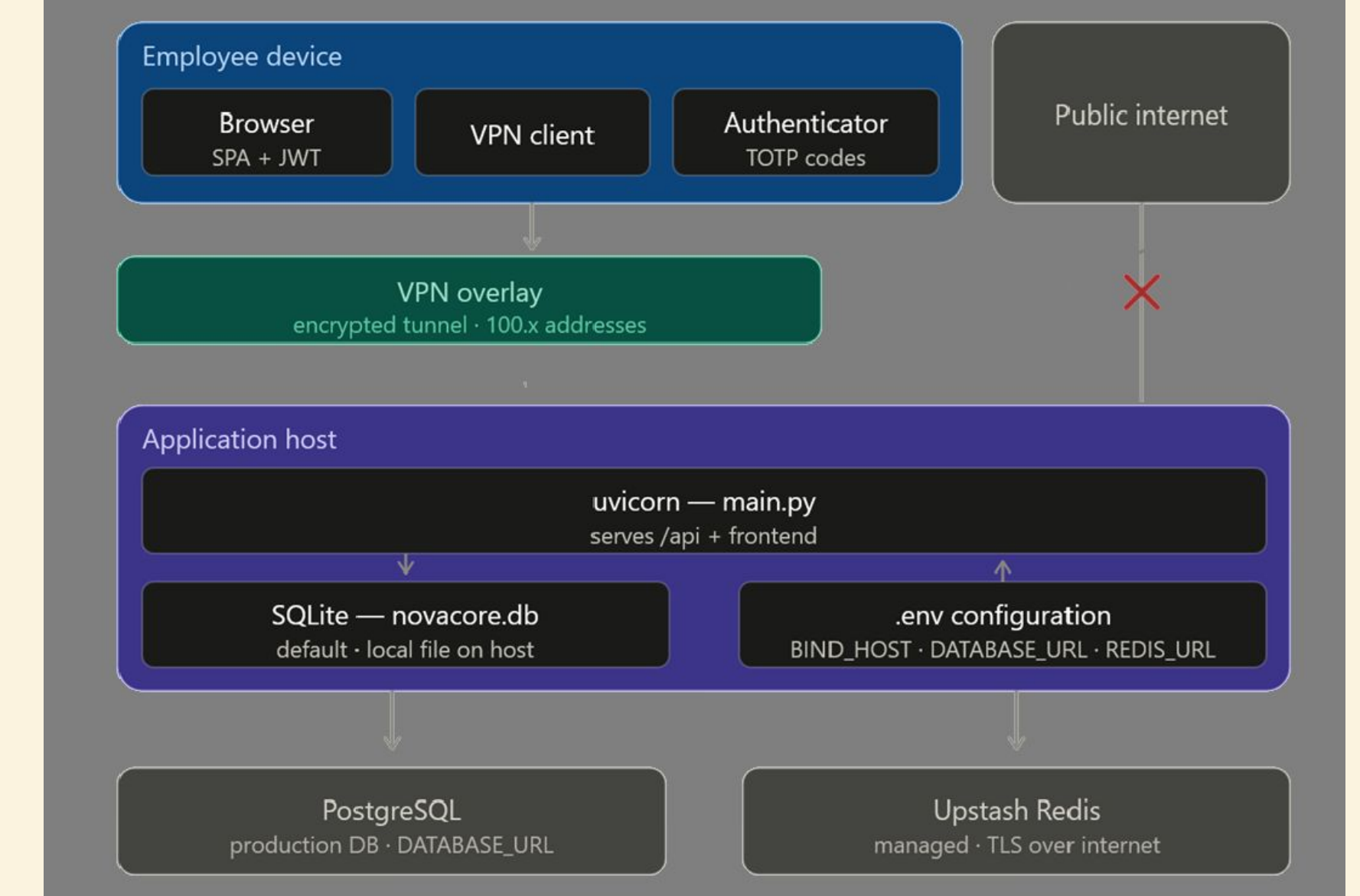


Figure 3 (Bottom) - Role-Based Access Control Menu, IT Admin Access Level

## Systems & Operations (SO6)



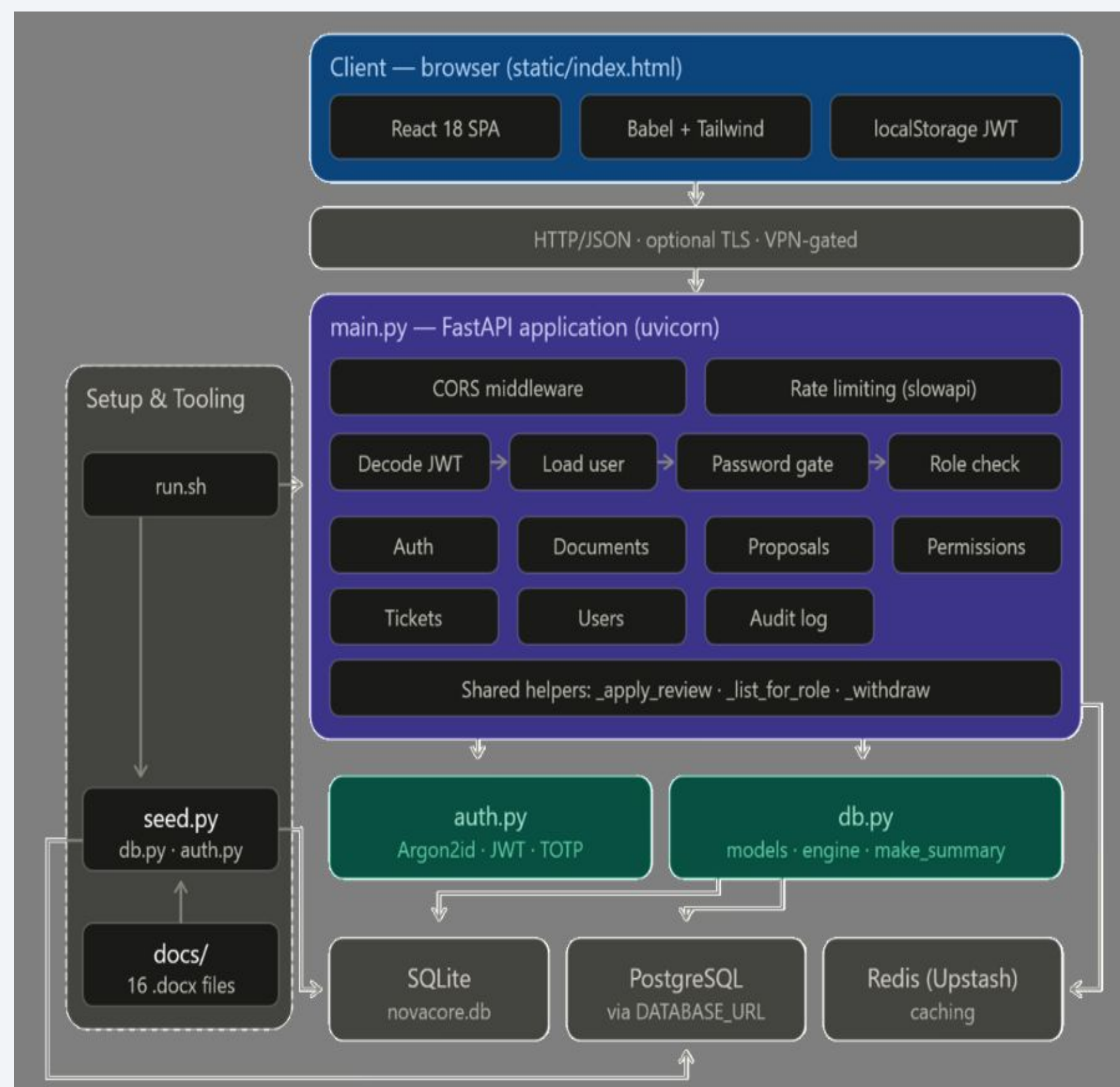
### NovaCore Deployment Architecture

- Designed with minimum maintenance in mind, IT admins outfit user devices with a VPN and MFA application of enterprise choice to connect to the NovaCore platform
  - NovaCore gives access and clear instruction for a simple cloud hosting (supabase), document caching service (Redis), and VPN (Tailscale) that can optionally be scaled to suit their needs with .env configuration

## Stakeholders & Requirements

- NovaCore was made with small enterprises in mind, to provide a scalable and easy to navigate infrastructure that could be used according to their needs even if they can't maintain or develop their own.
- Our team decided to address these core needs for our intended stakeholders:
  - a unified documentation suite
  - customizable role-based access to features and documentation
  - robust security features
  - a centralized workflow for policy changes and access control
- To this end, functional requirements included:
  - MFA authentication for all users
  - A support ticket system
  - Auditing and role-based access management, among others.
- Non-functional needs met are done through
  - Password hashing (Argon2)
  - Caching to prevent database strain from document retrieval
  - VPN-gated portal access
  - Database abstraction for easy database upgrades

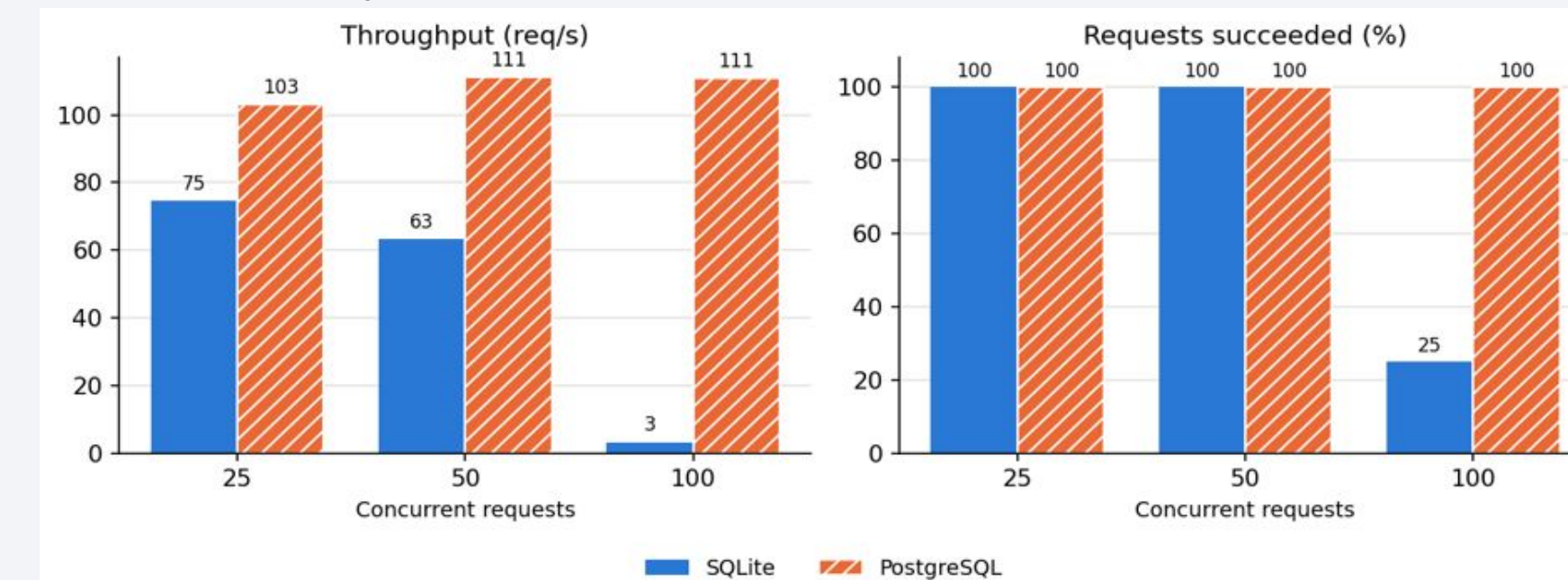
## Architecture



NovaCore Architecture Diagram

## Results & Evaluation

- There were several important evaluations made, such as:
  - Performance and scalability under load for chosen databases (SQLite initially, later PostgreSQL)
  - Validity of security checks like VPN gated access or MFA
- NovaCore is primarily a document access based platform
  - As such, write-heavy performance was evaluated, with results in the graph below
  - The initial SQLite database failed at around 100 concurrent requests, and so we incorporated PostgreSQL to compensate.



| Test case                                       | Result |
|-------------------------------------------------|--------|
| Unauthenticated document access blocked         | PASS   |
| Unauthenticated admin endpoint blocked          | PASS   |
| Unauthenticated audit log access blocked        | PASS   |
| Pre-MFA token rejected as bearer                | PASS   |
| Standard user -> admin user list denied         | PASS   |
| Standard user -> submit content proposal denied | PASS   |
| Standard user -> audit log denied               | PASS   |
| If admin cannot approve own proposal (SPD)      | PASS   |
| Standard user denied role-restricted document   | PASS   |
| Weak password rejected by policy                | PASS   |
| Current password reuse rejected (history)       | PASS   |
| Tampered JWT signature rejected                 | PASS   |
| Invalid password rejected at login              | PASS   |

- To the left is the results of boundary testing for user access based on roles or feature-sets, like MFA or failure to adhere to password policy

## What's Next

- In future, we'd like to see more resources made available for NovaCore to provide:
  - Increased server capacity for cloud storage or caching
  - Freely available load-balancing architecture
  - Alternative backup recovery methods for MFA
  - Iterating on the documentation and user policies supplied by NovaCore
  - Built-in test-suite so that enterprises can test their NovaCore environment health on-premises
  - Currently, documentation upload is back-end only, but a front-end, audit approved upload system would be beneficial

## People & Acknowledgments

**Team Leader:** Tereck Lindsey

**Team Members:** Tereck Lindsey, Alian Rodriguez, Kevin Roque, Aymanul Haque Sadaf

**Product Owner:** <name>

**Mentor:** <name>

**Sponsor:** <company / organization, if any>

**Acknowledgments:** Product owner, mentors, and: "KFSCIS Capstone Program, Prof. Masoud Sadjadi."