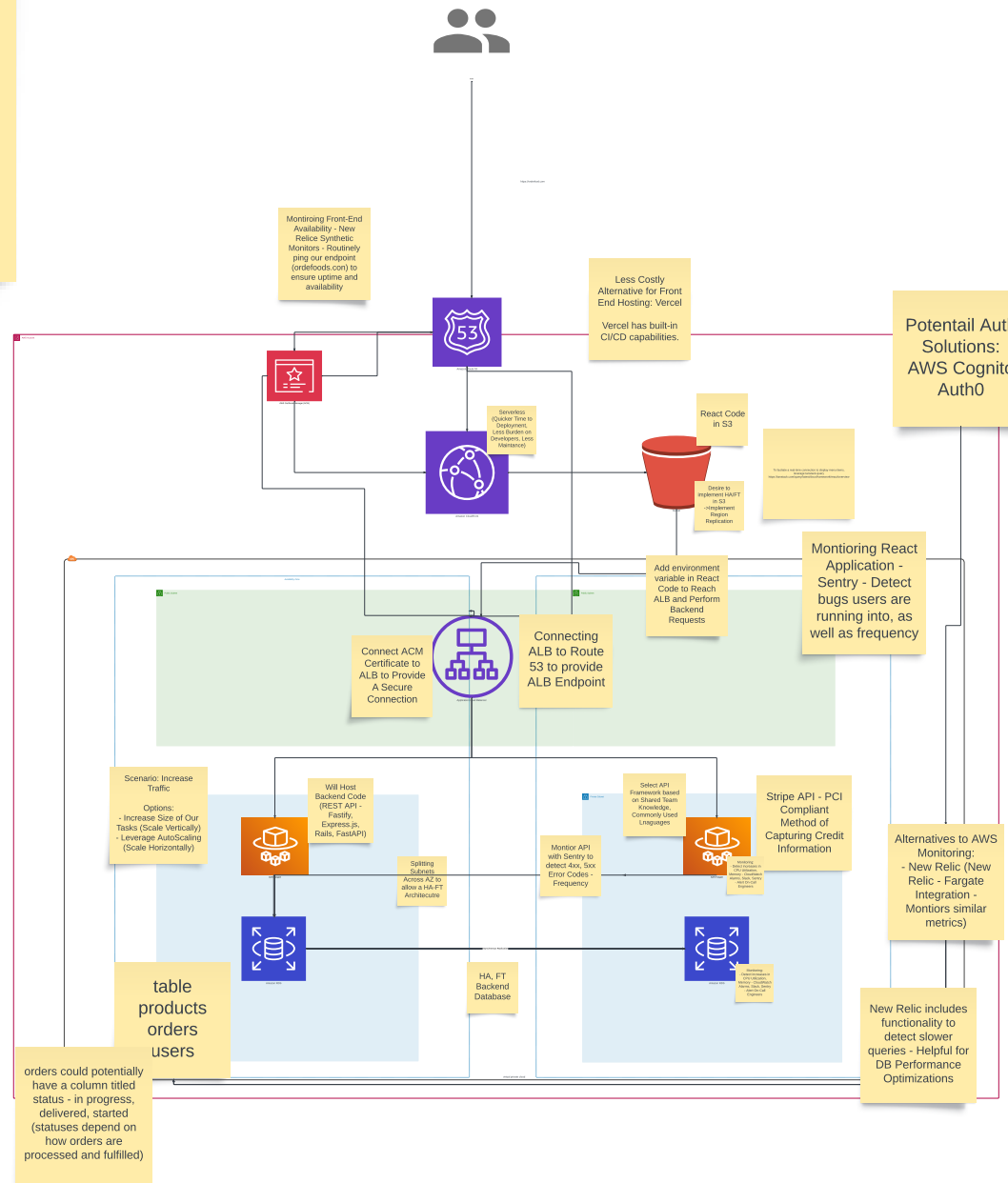Imagine you are tasked with designing a web application for a food delivery service. The web page should display a real-time menu, allow users to place orders, track order status, and view previous orders. Describe how you would handle the frontend structure, backend API design, real-time updates for order tracking, caching, database storage, and user authentication. Consider aspects like performance optimization, scalability, and the specific AWS services or other technologies you would use to support each part of the system.

Action Items

Assumptions

Montiroing Front-End Availability - New Relice Synthetic Monitors - Routinely ping our endpoint (ordefoods.con) to ensure uptime and availability

Less Costly Alternative for Front End Hosting: Vercel

Vercel has built-in CI/CD capabilities.

Potentail Auth Solutions:
AWS Cognito
Auth0

Serverless (Quicker Time to Deployment, Less Burden on Developers, Less Maintance)

React Code in S3

Desire to implement HA/FT in S3 -Implement Region Replication

Montioring React Application - Sentry - Detect bugs users are running into, as well as frequency

CI/CD Options:
- AWS CodeBuild
- Host Jenkins Service
- Github Actions

Add environment variable in React Code to Reach ALB and Perform Backend Requests

Connect ACM Certificate to ALB to Provide A Secure Connection

Connecting ALB to Route 53 to provide ALB Endpoint

Scenario: Increase Traffic

Options:
- Increase Size of Our Tasks (Scale Vertically)
- Leverage AutoScaling (Scale Horizontally)

Will Host Backend Code (REST API - Fastify, Express.js, Rails, FastAPI)

Select API Framework based on Shared Team Knowledge, Commonly Used Lnaguages

Stripe API - PCI Compliant Method of Capturing Credit Information

Montior API with Sentry to detect 4xx, 5xx Error Codes - Frequency

Alternatives to AWS Monitoring:
- New Relic (New Relic - Fargate Integration - Montiors similar metrics)

Splitting Subnets Across AZ to allow a HA-FT Architecutre

table
products
orders
users

HA, FT Backend Database

New Relic includes functionality to detect slower queries - Helpful for DB Performance Optimizations

orders could potentially have a column titled status - in progress, delivered, started (statuses depend on how orders are processed and fulfilled)