LAB - 5 Choosing Technical Stacks

Name: Sri Naga Hansi Mamidi

CWID: 20012906

Scenario 1: Logging

Elasticsearch is inherently known for storing huge data with customizable fields and its index-based fast search through the database. Logging of arbitrary pieces of technology implies that huge entries of technological usage are stored and monitored when necessary. Hence, Elasticsearch is a good option for logging. It can also be effectively used to submit the log entries through the REST API endpoints by sending the data in the JSON format or any other suitable format similar to Postman. Log entries can be queried via the REST API using the keywords or custom fields. Splunk and Dynatrace are well-known tools for visualizing the logs based on time filters and memory utilization. An Express.js paired with node.js server is well-suited for Javascript web application.

Scenario 2: Expense Reports

A realtime database like Firestore can be structured to store data i.e., to store the same data structure: id, user, isReimbursed, reimbursedBy, submittedOn, paidOn and amount. It allows users to update the records in real-time when an expense is reimbursed. Express.js with Node.js for web server due to its lightweight nature and the way it handles the server requests. Cloud functions can be used to automatically trigger the generation of PDFs using wkhtmltopdf including all the details when an expense is marked as reimbursed. Email service like MailGun is an option to easily handle emailing and automatically trigger an email based on the update. Templating for this web application can be handled with React to build interactive interfaces.

Scenario 3: A Twitter Streaming Safety Service

Twitter Streaming API is the API which allows access to a real-time stream of tweets. A web server built using Javascript, React along with HTML and CSS styling makes it visually easy to monitor the data for any alert words. For the system to be expandable beyond local precinct and constantly stable, it has to be auto-scalable and it can be achieved by hosting its servers in the cloud. Cloud's kubernetes cluster

creation autoscales its servers based on the traffic i.e., allocates more servers on high traffic and discards the servers not in use for low traffic. A fail-over mechanism setup along with this makes the server in other regions available even when one region is down. Elasticsearch again proves to be handy in this case where the streaming data of tweets is exponentially high with indexing search for both triggers and historical log of tweets. An email alert can be set up using Mailgun to trigger an email for alerting words found. This enables handling of real-time incident reports by making use of the email templates and filling in the incident data. Google Cloud Storage(GCS) is an ideal option to store media data due to its storage capabilities. This media storage can be categorized to be standard, nearline, coldline and archive based on the storage periods and makes it easy to filter, fetch and manage storage costs. An archive storage is perfect for storing the long term data and less cost. React for UI, Node.js for backend along with redis caching and data visualization libraries like chart.js can be used to effectively handle all these.

Scenario 4: A Mildly Interesting Mobile Application

To handle geospatial data, postgreSQL can be used along with the PostGIS extension which can store coordinates and perform geospatial queries to find events within a specific geographic area. Google Cloud Storage is again a suitable option to store images with a long storage period(Archive Storage) that comes with less storage cost. Cloud Content Delivery Network(CDN) can be used for caching viral or most frequently accessed content within a specific region for fast retrieval. Express.js and node.js is an ideal solution again to build a lightweight API and PostgreSQL to accommodate its geospatial nature. An administrative dashboard for managing content can be built using React and Javascript along with Firebase authentication for the users to signup and signin with or without social media accounts.