CS-554 Lab 3

Scenario 1: Logging

1) I would store the log entries in a NoSQL database like MongoDB. I would use MongoDB because it is a document-based database. It is dynamic, scalable, and provides users high performance and availability.
2) Users can perform a POST request to the server to submit log entries. Logs are then submitted as a document in MongoDB. I would implement this using the Nodejs server and handlebars that provide users with a front-end form to submit the request.
3) Users can query log entries by submitting GET requests to the server and in return can get a response with the logs from the MongoDB database.
4) I would use a front-end library like React that will allow users to see their entries.
5) I would create the web server using Nodejs and Express. Express is a Node framework that is used to build an efficient and effortless web server.

Scenario 2: Expense Reports

1) Since expenses to be stored are of the same data structure, I would use a structured database like MySQL to store the expenses.
2) I would use Nodejs and Express as a web server. Advantages of using express are that it allows defining routes of application based on HTTP methods and URL, it also includes various middleware modules, and it easily connects with databases like MongoDB and MySQL.
3) I would use modules like Nodemailer to send emails to users. Nodemailer can be used to send HTML or plain text email with attachments, and it has built-in SMTP support.
4) Node libraries like PDFKit and pdfmake can be used to generate pdf.
5) The templating for the web application can be done using handlebars. Handlebars is a templating engine that ensures minimum templating and is a logicless engine that keeps the view and the code separated.

Scenario 3: A twitter streaming safety service

1) I would use PowerTrack API to filter tweets as they happen and Search API to search previous tweets using advanced filtering tools.
2) This can be built expandable to beyond the local precinct by using geolocation filtering.
3) To make sure this system is constantly stable, I would ensure that all the security measures are handled properly, that my server is not vulnerable to any cyber-attack and is fully secured and the application is scalable and can be used globally.
4) Nodejs and Express would be used for creating a web server.
5) I would MongoDB as my database because it is fast and highly scalable, and data can be stored in JSON. It can be easily accessed with Nodejs.
6) Redis can be used to store the historical logs of all tweets because it is comparatively faster.
7) I would use express middleware to check for any keywords and trigger an investigation if found in real-time.
8) I would use cloud storage like Google Cloud for the long-term storage of media files. It can handle large media files and ensures the secure sharing of files.
9) I would use Express and Nodejs to create a web server.

Scenario 4: A mildly interesting mobile application

1) Since I would use react native to build the mobile application, google maps API would be a good choice to handle the geospatial nature.
2) I would use Amazon S3 for storing the images for the long term and MongoDB for the short term. I would also use Redis for faster retrieval of images.
3) Nodejs and Express would be used for the API backend and can be easily connected to react-native since both are written in JavaScript.
4) I would use MongoDB as my database because we need a non-structured database in this scenario, it allows geospatial queries. It is versatile, fast, and easily accessible with Nodejs.