PROJECT PROPOSAL

BIG DATA MANAGEMENT

POST GRADUATE DIPLOMA IN DATA ENGINEERING

PROJECT TITLE

MAPPING TOMORROW: INNOVATIVE APPROACHES TO URBAN INFRASTRUCTURE WITH OPENSTREETMAP DATA

SUBMITTED BY:

NIRAJ BHAGCHANDANI [G23AI2087]
SHUBHAM RAJ [G23AI2028]
BHAVESH ARORA [G23AI2126]

SUBMISSION DATE: 20TH OCTOBER, 2024

DEPARTMENT OF AIDE
INDIAN INSTITUTE OF TECHNOLOGY, JODHPUR

MAPPING TOMORROW: INNOVATIVE APPROACHES TO URBAN INFRASTRUCTURE WITH OPENSTREETMAP DATA

Team Composition:

NIRAJ BHAGCHANDANI [G23AI2087]
SHUBHAM RAJ [G23AI2028]
BHAVESH ARORA [G23AI2126]

Application Overview:

Our project focuses on creating a powerful geospatial application that takes advantage of OpenStreetMap (OSM) data to help cities plan smarter. The goal is to analyze traffic patterns, and public transport accessibility, and suggest optimal routes, especially for emergency services. By using big data analytics and machine learning, we aim to offer tools that city planners can use to reduce congestion, improve public transport, and better plan infrastructure development. Some key features will include traffic heatmaps, predictions on future congestion spots, and recommendations for expanding public transport based on real-time and historical data.

Key Technology Challenges:

1. Managing Large Amounts of Data:

OpenStreetMap data is massive and detailed, with millions of roads, buildings, and locations. Handling this large dataset efficiently and running complex queries quickly is one of the big challenges. We plan to use big data tools like Hadoop or Spark to process and store this data, ensuring that the system can handle large-scale requests smoothly.

2. Processing Real-Time Data:

Since traffic patterns and public transport data change rapidly, we need to process and display this information in real time. This means we'll need powerful backend infrastructure to keep up with constant updates and perform calculations on the fly, like optimizing routes and predicting traffic jams. Python libraries like Pandas and Dask will help us with this real-time data processing.

3. Ensuring Scalability:

As the application will potentially serve many users and handle large datasets, we need to ensure that it can scale without slowing down. AWS will be our go-to cloud provider to help us scale the app dynamically as demand increases. We'll use containerization with Docker and orchestration with Kubernetes to manage the deployment smoothly.

4. Creating a User-Friendly Experience:

City planners and decision-makers need to access this data intuitive and simple way. So, we must build a user interface that makes it easy for them to explore traffic data, generate insights, and simulate possible infrastructure improvements. React.js will allow us to build a dynamic and responsive UI, while Leaflet.js will power our interactive maps to visualize traffic and routes effectively.

Technology Stacks:

Frontend:

- React.js: For building a responsive and interactive UI.
- Leaflet.js: For rendering dynamic and interactive maps.
- Material-UI: To create a modern and accessible user interface.

Backend:

- **Node.js + Express.js: To** manage server-side logic and APIs.
- **PostgreSQL with PostGIS:** For geospatial data storage and querying, essential for running advanced map-related queries.
- **Python (Pandas, Dask, Scikit-learn):** For handling machine learning models, traffic predictions, and real-time data processing.
- **Big Data Tools (Hadoop, Spark):** To efficiently process and store the large volumes of OSM data.

Cloud Infrastructure:

AWS (Amazon Web Services): Our cloud provider for hosting, scaling, and managing infrastructure. Services like EC2 for compute power, S3 for data storage, and Kubernetes for managing containerized deployments will be critical for making the app scalable and robust.

APIs:

- OpenStreetMap API: For fetching real-time map data.
- Google Maps API (if needed): To augment the OSM data with additional geolocation services.

Deliverables:

1. Application Demonstration:

We'll deliver a fully working web application that allows users to interact with maps, explore traffic and public transport patterns, and generate predictions based on real-time data. The app will be hosted on AWS, providing access to live data and functionality.

2. Benchmarking Study:

We'll also deliver a detailed benchmarking study, comparing the performance of different tools like PostGIS, MongoDB's geospatial features, and big data processing frameworks like Hadoop and Spark. This will give insights into the best technologies to use for handling large-scale geospatial data.

3. Codebase and Deployment:

All of the code will be made available on GitHub, fully documented for easy setup and deployment. The app will be containerized using Docker and deployed on AWS. We'll also set up a CI/CD pipeline to automate testing and deployment, ensuring the app remains up-to-date and bug-free.