ASR (Architecturally Significant Requirements)

Purpose and Scope 

Purpose of the Stock Tracker application is to track the stock listed in the National Stock Exchange of India (NSE). The application user here after called as “user” should be able to view the stock which are preconfigured in the application by an admin or super user hereafter called as “admin”. Admin will configure the stock symbols from the list of the stocks listed in NSE.  The stocks available in the Stock Tracker application will be a subset of stocks available in NSE.

All stocks which are configured by the admin is available to any user of the application. User will not be able to add, edit or delete any stocks listed.

Users will be able to view the current value of the stock in near real time during NSE market hours. If the market is closed the last value of stock available will be displayed to user.

Intended Audience

The audience of this application is stock traders, market analysts, financial advisors, and researchers with the “company-name” domain.

Business Context 

* Stake Holders

Main stake holders of the Stock Tracker applications are traders inside the company, market analysts, financial advisors and researchers. Other stakeholders also include top management who is responsible for strategic economic decisions.  

Business Goal   
Goal is to provide realtime stock values to application users. This enables the users to realtime access to stock data and help to make better decisions. The app also helps in prioritization of strategy, policies and action plans in the area of economy, resourcing and future product roadmap.

Capture Constraints

|  |  |  |  |
| --- | --- | --- | --- |
| **Constraint** | **Origin** | **Type** | **Context** |
| React JS, Node JS, FaunaDB | Developer | Technical |  |
| The platform data backup should happen daily. | Stack Holder | Technical |  |
| Portal and API should not be accessible outside Organization domain freely | Stack Holder | Technical |  |
| The site should be responsive to Chrome 85 & above, IOS and Android | Stack Holder | Technical |  |
| The pipeline should automate the code, quality checks, test automation, performance test and security test. | Stack Holder | Technical |  |
| Planform much also follow CI/CD process and follow Blue/Green deployment strategy | Stack Holder | Technical |  |
| Feature Toggle | Stack Holder | Technical |  |
| Failure and Retry (should start from failover) | Stack Holder | Technical |  |
| Use oath for Microsoft identity provider | Stack Holder | Technical |  |
| The Delivered code should have  0 Blocker and critical Sonar Bugs vulnerability  0P1 &P2 defect  90% unit test coverage  90% Test Automation | Stack Holder | Technical |  |

Define the Quality Attributes (Non-Functional Requirements)

|  |  |  |
| --- | --- | --- |
| **Quality Attribute** | **Scenario** | **Priority** |
| Performance | Platform should support 500 symbols with pricing update coming every 10 seconds | High |
| Scalability | During a scheduled maintenance time, new servers can be added. | Low |
| Performance | The 90-percentile page response time should be under 2 seconds |  |
| Performance | The API response time should be under 200 ms |  |
| Scalability | The site should be scalable to take a load 0 to 1000 parallel connections and site should be sustainable to the load |  |
| Availability | The platform should support 99.95% up time |  |
| Security | The platform must be adhere to security compliance for the OWASP top 10 vulnerabilities |  |
| Availability | Automated test data creation & clean-up (should support million rows of data) |  |

Functional Requirements

**Admin user Functionalities**

* Pricing Configuration
  + Third party api config manager (api url, endpoint, country, pricing symbols)
* Create Configuration
* View Existing Configuration
* Save /Update Configuration
* Authenticate User

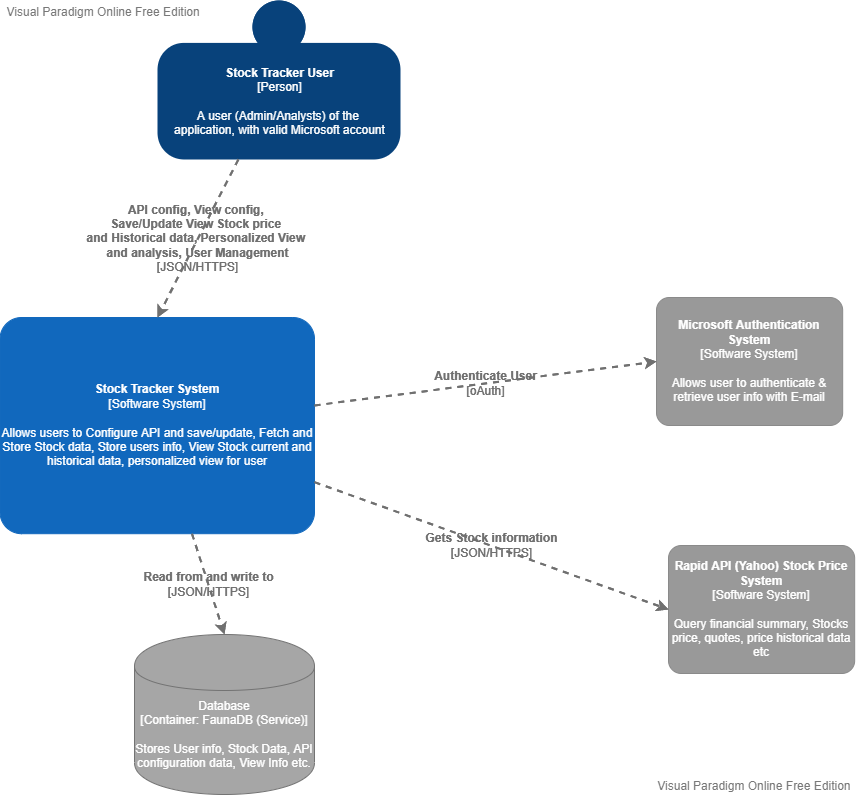
**Analyst user Functionalities**

* Authenticate User
* View market data (sorting, filtering, searching, grouping)
* Analyst should be able to save view (personalization/Fav)
* Export to excel/csv
  + Third party api config manager (api url, endpoint, country, pricing symbols)

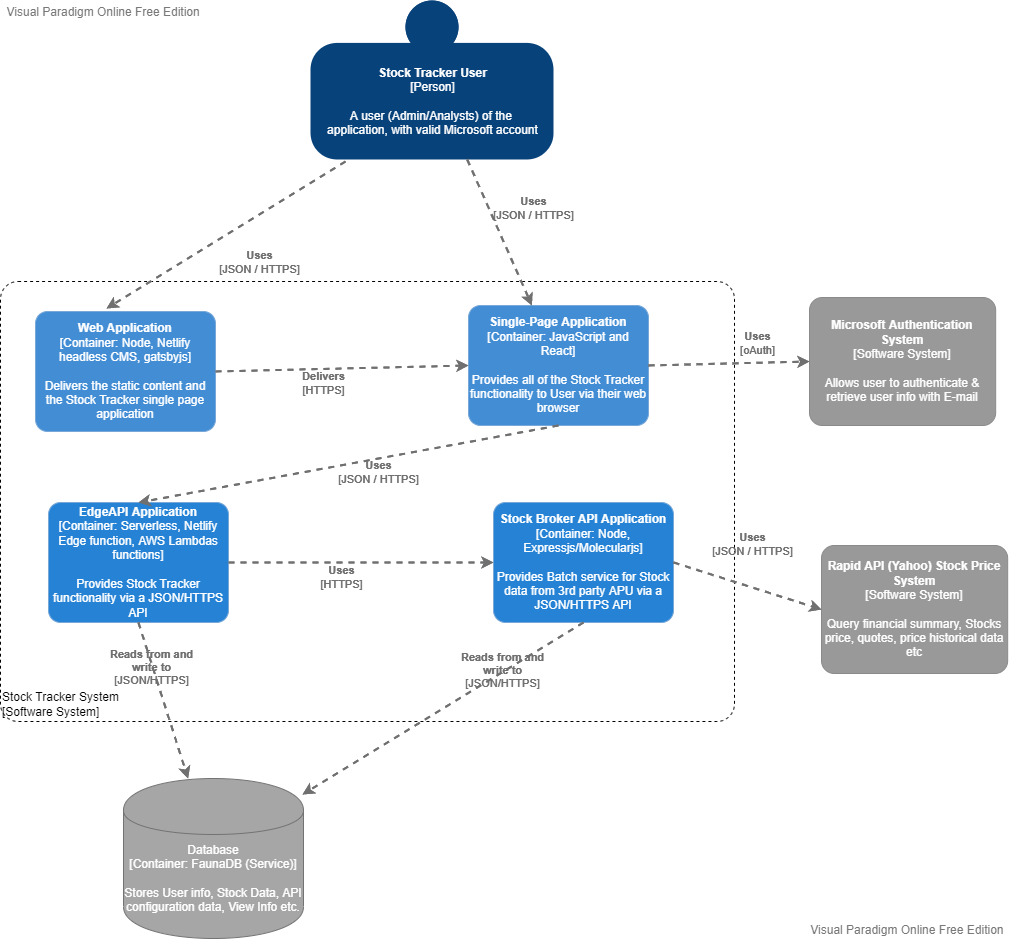
Architecture Diagrams

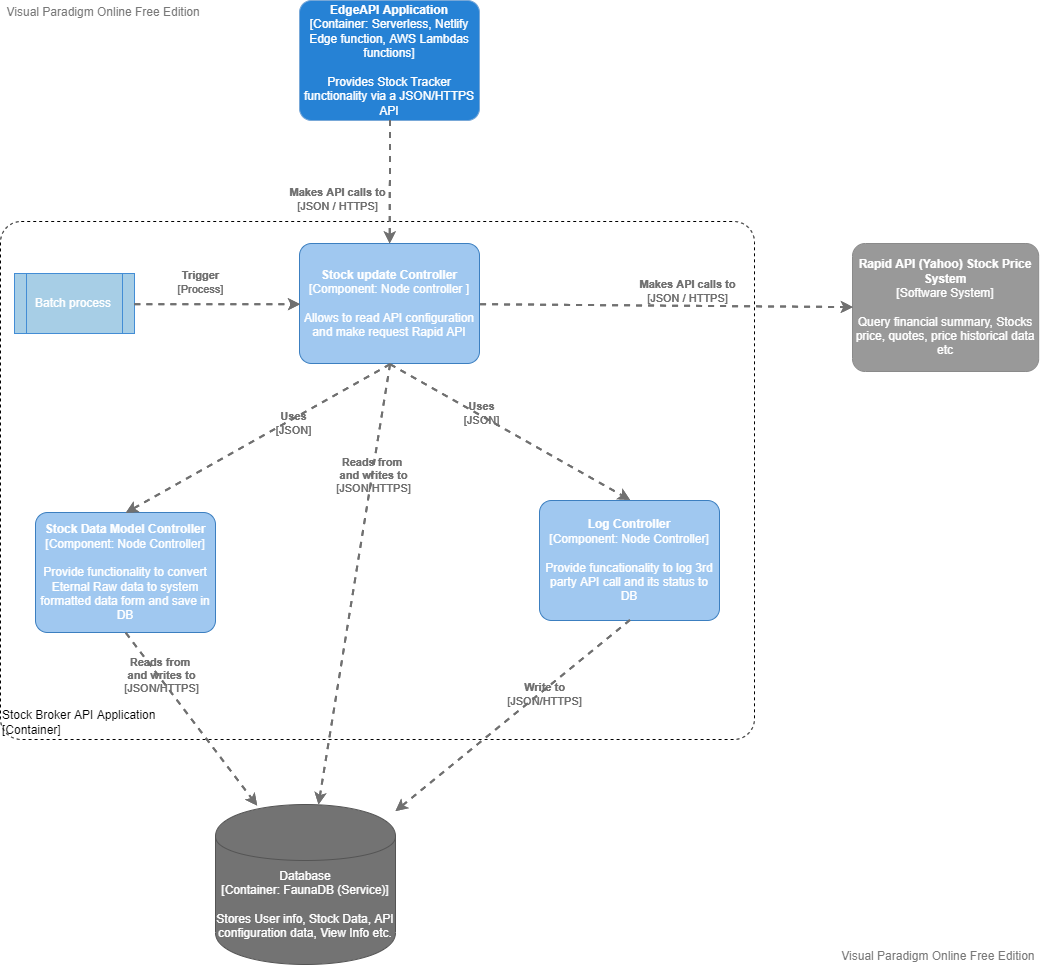
Stock Tracker C4 Model of Software Architecture

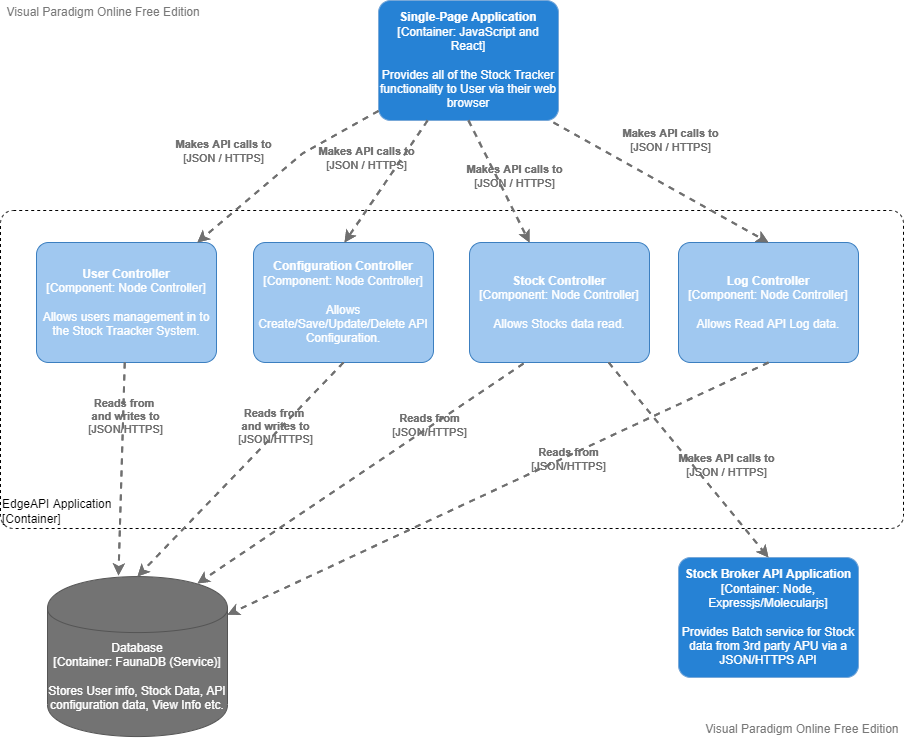
Context Diagram



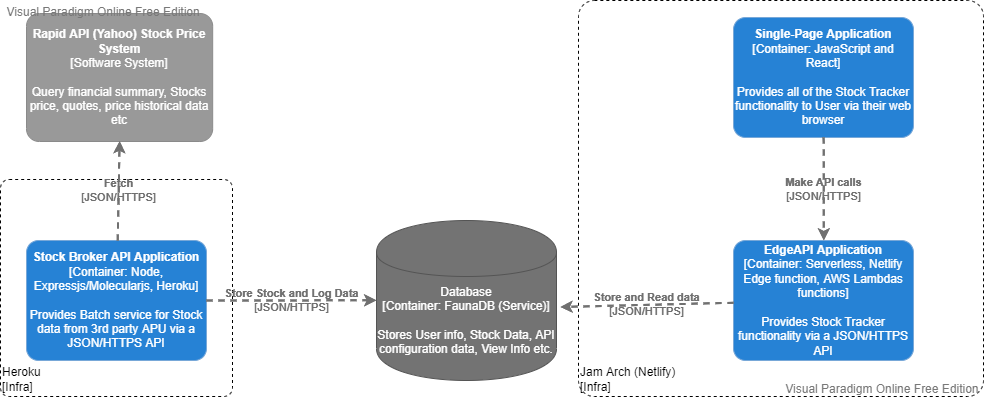
Container Diagram



Component Diagram

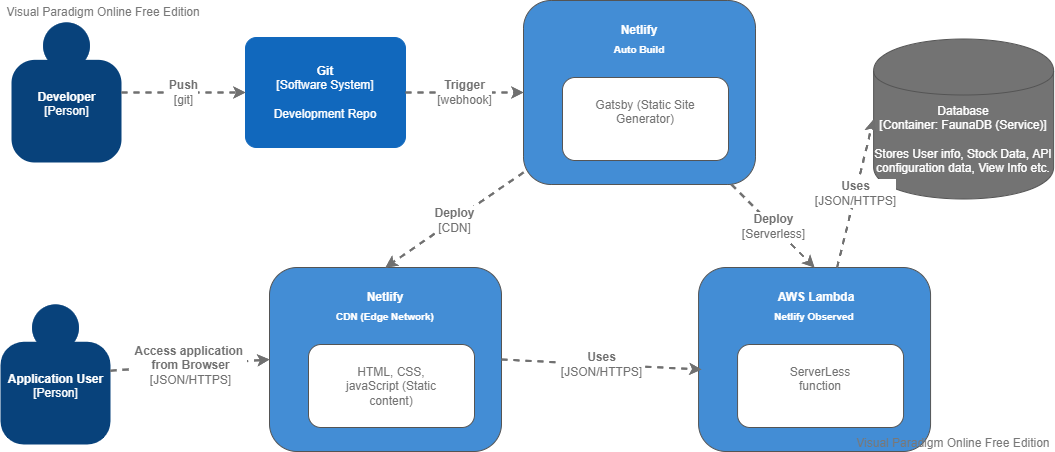


Infrastructure Architecture Diagram

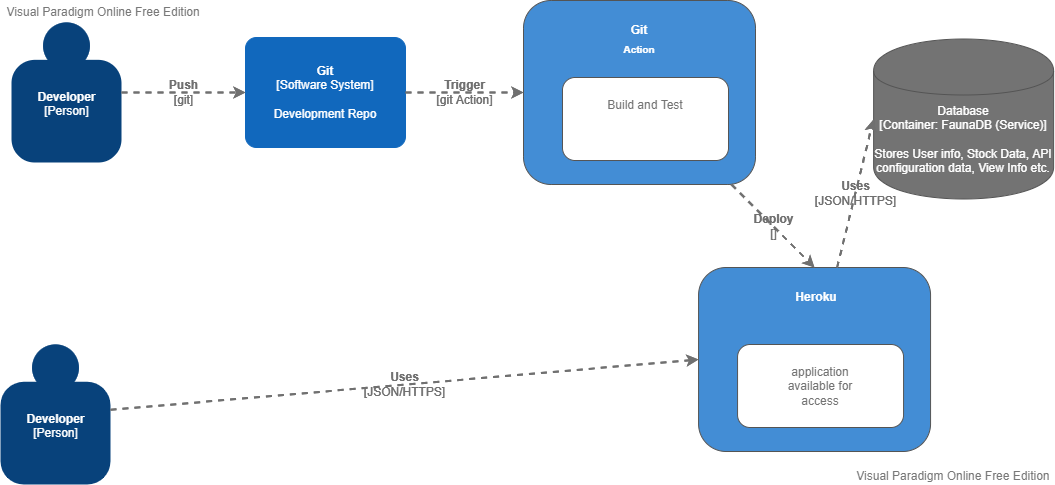


Deployment Diagram

Netlify Deployment Flow



Node Server Deployment Flow



End of Document