

Jayesh Das

Software Engineer

✉ jayeshdas786@gmail.com

☎ 8898523904

📍 Kandivali East, Mumbai 400101, India

in LinkedIn

🐙 GitHub

Profile

I am a dedicated professional with more than 3.3 years of experience in the fintech industry, specializing in handling real-time market data with an emphasis on effective performance. My expertise lies in Microservice architecture design and implementation, allowing me to develop robust and scalable solutions for complex financial systems

Education

SSC From Maharashtra Board, Sheth N.L High School

Mar 2014

Percentage : 80%

MUMBAI, INDIA

HSC From Maharashtra Board, KES Shrof College

Feb 2016

Percentage : 74.46%

MUMBAI, INDIA

Bsc in Information and Technology From Mumbai University,

May 2019

Ghanshyamdas Saraf College

MUMBAI, INDIA

CGP : 6.76/10

Certificates

- Full Stack Developer Certification from SDAC INFO TECH
- Digital Image Processing [🔗](#)
- Data Visualization With Power BI [🔗](#)
- React JS
- Advanced SQL [🔗](#)
- Python for Data Science [🔗](#)

Skills

- Python
- Javascript
- C++
- Django REST API
- React JS
- MSSQL
- PostgreSQL
- MongoDB
- JAVA

Professional Experience

Software Engineer, Fin Rise Softech Pvt. Ltd

Oct 2020 – present
Mumbai, India

Projects

1. Web Trading With Motilal, XTS API Intigration

Summary-The web trading portal is a dynamic platform that seamlessly integrates with the XTS API, providing traders with access to real-time market data and trade execution capabilities. The portal offers a range of powerful features, including strategy trading, a price alert system, and live profit and loss updates, efficiently delivered to clients via a Telegram bot.

Technology/framework/Programming Language which we used: Django REST API , Python , Javascript,ReactJS, Postgress SQL,Websocket,ZMQ,Cython,Message queues (SysvIPC), Shared Memory,

2. Strategy Monolith Application

Summary - Strategy monolith application is a powerful and versatile trading platform developed in Python. It enables traders to create and implement a wide range of trading strategies, including **fractal, stochastic, wait and trade, straddle, jobber, arbitrage spread, and many others**. With seamless integration, the application allows users to place orders with select brokers such as **AngleOne and Zerodha**, enhancing the efficiency and ease of executing trades.

3. Telegram BOT Monolith Application, Integrate TelegramBOT API With Our services

Summary - Our Telegram bot is a powerful tool designed to provide developer and traders with full control over their trading system. The bot offers a range of functionalities that ensure smooth operations and continuous monitoring of the system's health and microservice running status. Additionally, it facilitates real-time updates on traders' live profit and loss, as well as their current positions. The bot's user-friendly interface makes it accessible to everyone, empowering traders to stay connected and informed on the go.

4. Web Trading With Binance API

Summary - The web trading portal is a robust platform that integrates with Binance API, providing users with a seamless trading experience. It offers a wide range of features, including strategy trading and a price alert system, efficiently powered by a Telegram bot.

Programming language which we used: Django Template , Python , Javascript, MSSQL,Websocket

Team size : 4

5. HFT Web Trading(API and MICROSERVICE Development),

High frequency trading with NSE/SGX Co-location server

Summary - we has developed a high-frequency trading platform that leverages a co-location server with the National Stock Exchange (NSE) and Singapore Exchange (SGX). This co-location server ensures ultra-low latency access to market data and order execution, enabling lightning-fast trading capabilities.

The core of our trading infrastructure is a set of C++ microservices meticulously designed by our senior developers. These microservices serve as the backbone of our trading system, efficiently handling various aspects of the trading process.

Key components of our microservices architecture include:

- Feed Broadcaster,Strategy Reader/Sender,Profit and Loss,Alerting System,REST API
- Involved in full life cycle of the project (SDLC) from requirements gathering,
- analyzing, designing and developing App using AGILE Methodology.

Technology/framework/Programming Language which we used: Django REST API , Python , Javascript,ReactJS, Postgress SQL,Websocket,ZMQ,Cython,Message queues (SysvIPC), Shared Memory,

Team size : 12

6. WEB RMS(V2.0), Risk Management System Web Application

Summary -The Risk Management System is a web application developed using python monolith application and django rest api. Its primary objective is to assist organizations in effectively managing risks by providing a user-friendly interface to handle large amounts of data. The application incorporates various features and functionalities that facilitate risk analysis, mitigation, and reporting.It is easy to monitor risk anytime and anyway. this is improved version of our RMS desktop application(v1.0).handalling Live data using websocket and live graphical representation .

Technology which we used: Django Template , Python , Javascript, MSSQL,Websocket

Team size : 6

7. RMS Desktop Application (v1.0), Risk Management System application

Summary -The Risk Management System is a desktop application developed using Java Swing and AWT (Abstract Window Toolkit) libraries. Its primary objective is to assist organizations in effectively managing risks by providing a user-friendly interface to handle large amounts of data. The application incorporates various features and functionalities that facilitate risk analysis, mitigation, and reporting.

Technology which we used: **JAVA Swing AWT,MSSQL**

Team size : 4

Interests

Swimming

Badminton