# Slide 1

Problem Statement and Team Details

Problem Statement: Stock Market Consultant Agent  
Team Name: Debug Devils  
Team Leader Name : Mallepula Varshitha  
Institute Name: KL University  
Team Leader Email ID: mallepulavarshithareddy@gmail.com

# Slide 2

Problem and Solution

Problem: Stock Market Consultant Agent  
Problem Solution: A Stock Consultant Agent to guide beginners in simple terms.  
Portfolio Input: Users enter stocks they own or plan to buy.  
Actionable Recommendations: Buy / Sell / Hold / Diversify suggestions.   
Example: TCS holding is risky — consider reducing.  
Example: Portfolio lacks IT — add Infosys to diversify.  
User Tracking: Monitors queries and advice reports for progress tracking.  
Impact: Bridges gap between raw market data and actionable insights, helping beginners build confidence and make smarter decisions.

# Slide 3

Methodology & Implementation

Problem Understanding: Beginners see stock prices but don’t know what actions to take.  
Portfolio Input: Users upload holdings via CSV or form (symbol, quantity, buy price).  
Data Collection: Stock prices and sector data fetched from mock CSV/API and updated in real-time using Pathway.  
Rule-Based Analysis:  
Calculate sector concentration, profit/loss %, portfolio balance.  
Apply simple rules:   
If stock > 35% of portfolio → Suggest Reduce. If sector missing → Suggest Add. If profit > 20% → Suggest Book gains. If loss > 15% → Suggest Exit. Else → Suggest Hold.   
Advice Generation:  
Rules are converted into simple English one-liners (advice cards).

# Slide 4

Technology Used

Frontend: Built with Streamlit for an interactive dashboard.  
Features: portfolio upload, advice display, usage summary.  
Backend Logic: Implemented in Python. pandas used for portfolio calculations. Custom rule engine (if–else logic) for generating buy/sell/hold/diversify signals.  
Data Handling: Stock and sector data managed via CSV files and mock APIs. Pathway integrated for fetching and syncing real-time stock prices.   
Billing & Usage Tracking: Simulated system using Python dictionaries or SQLite. Tracks portfolios analysed and advice generated.  
Storage (Optional): Advice history and portfolios can be stored in SQLite or CSV logs.

# Slide 5

Flowchart & Supporting Images

# Slide 6

Feasibility and Market Use

Feasibility  
Easy to build with Python, Streamlit, and stock APIs. Low cost since it uses open-source tools.  
Beginner-friendly: users just upload portfolios, advice comes in plain English.  
Scalable: can later add AI/ML for smarter insights.  
Market Use  
Helpful for students and small investors new to stock markets. Gives simple portfolio advice (reduce risk, diversify, book profits). Fills a gap trading apps show data but don’t guide beginners. Can be extended as a standalone app or integrated into platforms.

# Slide 7

Conclusion

Provides beginners with simple, actionable stock advice.  
Uses portfolio input + real-time data + rules to generate insights.  
Suggests buy, sell, hold, or diversify in plain English.  
User-friendly dashboard with advice cards and usage tracking.  
Bridges the gap between complex stock data and practical guidance.  
Valuable for students and small investors, with scope for future growth.