- **By Koushik Rangu email: koushikrangu@gmail.com**

1. What is the core business problem and why does it matter?

The main business problem is that hotel bars are often facing stock management issues. Many times, popular items like specific liquor brands go out of stock, which leads to poor guest experience. At the same time, some slow-moving items are ordered in excess, which causes overstocking and waste. This imbalance affects not just customer satisfaction, but also increases storage cost and wastage. Our goal is to build a system that can predict how much of each item will be consumed and suggest how much stock should be kept ready at each bar. This helps in reducing losses and improving overall operations in hotel bar management.

2. What assumptions did you make? Why?

Since this is a simulation project and not all real-time data is available, we had to make a few assumptions. First, we assumed that the system works on a daily level instead of hourly data. To calculate the current stock, we used the latest consumption and assumed there is stock left for two more days. We also assumed a lead time of 3 days, meaning if you order something today, it will take 3 days to arrive. To be safe, we also included 20% extra stock as a safety buffer. These assumptions are made to make the system practical and close to real-world behavior, even if the live stock tracking system is not available.

3. What model did you use and why did you choose it? Why not others?

We used the **Holt-Winters Exponential Smoothing model** for time series forecasting. This model works well when there are patterns over time, such as daily or weekly consumption trends. It can understand both trend and seasonality, and it doesn’t require heavy tuning or a large dataset. We chose this model because it gives good results for bar consumption data, which is usually consistent and has repeating patterns. Other models like ARIMA are also good, but they may not handle seasonality as easily. More complex models like LSTM require lots of data and time to train, which is not suitable for our current dataset. So Holt-Winters is a good balance between accuracy and simplicity.

4. How does your system perform? What would you improve?

The system performs quite well in providing short-term forecasts. It predicts the consumption for the next 7 days and then calculates a recommended par level (how much stock to keep) and reorder quantity. These recommendations are given for each item at each bar, which makes it useful for real hotel use. The forecasts match the actual trends closely in most cases. However, the system can be improved further. Instead of estimating current stock, we can use actual closing stock data. Also, the model can be trained separately for different brands or bars. In the future, we can include cost optimization, reorder size suggestions, or even include supplier constraints.

5. How would this solution work in a real hotel?

In a real hotel bar, the system can run daily or weekly. Each night, it would check how much of each item was consumed, and based on that, it would give recommendations on whether to reorder and how much. The hotel manager can receive this as an Excel file or see it on a dashboard. This will help them order items in the right quantity and avoid both stockouts and overstocking. The par level tells them the safe quantity to keep on hand, and the reorder logic ensures they don’t run short. This can be implemented across multiple hotel locations, and each manager can view their own bar’s data and take action.

(Optional) What would break at scale? What would you track in production?

If this system is used across hundreds of hotel bars, a few challenges may arise. The forecasting might become slow if there are too many items or very large datasets. Also, if the data is missing or incorrect (like wrong closing balances), the forecast might be wrong. In production, we would need to add monitoring, logging, and alerts. We would track how accurate the forecast is compared to real consumption. We would also track how many times the reorder suggestions helped avoid stockouts or wastage. This would help improve the system over time and make it reliable at scale.