- Introduction (10 points). **Why** are you conducting this research? Identify potential users of the knowledge base and application(s) that you intend to develop.

This research is driven by the goal of establishing an actively traded fund centered on a swing trading investment strategy, which aims to capture short-to-medium term price movements typically lasting days to weeks, or potentially months. If an especially strong company is identified, then a buy and hold strategy may be employed. The fund's technical foundation will be a hybrid system, utilizing Python for efficient data loading and preprocessing, and Go for the high-performance core algorithmic implementation. Key data sources will include Yahoo! Finance for historical prices, Google Trends and StockTwits for sentiment analysis, and Polygon.io for specific company data. The investment philosophy is adaptive, with preliminary results iteratively shaping the algorithm's evolution. Potential users of the knowledge base and algorithm are individual investors, small to medium-sized hedge funds, and quantitative researchers who seek simulated automated trade execution, dynamic portfolio management, real-time risk monitoring, and comprehensive strategy backtesting.

- Literature review (10 points).. **Who** else has conducted research like this?

Existing research provides a strong foundation for this project, defining swing trading as a speculative approach focused on short-to-medium term price fluctuations, distinct from long-term investing or day trading, and employing strategies like trend-following, breakout, and reversal, all while emphasizing robust risk management (Fernando, 2024). This article highlights the coupled role of both technical analysis (using indicators like moving averages) and fundamental analysis (evaluating economic and company factors) in swing trading. Furthermore, academic studies consistently demonstrate the effectiveness of algorithmic trading, particularly when enhanced by machine learning and reinforcement learning models, which can predict stock prices and optimize trading policies, often yielding significantly higher risk-adjusted returns (Zhang et al., 2020) when augmented with market trend and sentiment data (Bollen et al., 2011). Sentiment analysis, derived from sources like social media (Twitter, StockTwits) and Google Trends, has been shown to correlate with market movements and improve prediction accuracy, although challenges related to data accuracy and noise filtering persist.

- Methods (10 points). **How** are you conducting the research? Make sure you address the issues that are the focus of this checkpoint assignment.

The research methodology centers on a robust data acquisition strategy, employing ETL and API integration to consolidate diverse financial data into a unified platform, which is crucial for enhanced data accuracy, improved reporting, and faster decision-making. However, the initial reliance on Yahoo Finance via yfinance is problematic due to its unofficial, unreliable web-scraping nature, necessitating a shift to more robust, officially supported data providers like FinancialModelingPrep (FMP), Polygon.io, Alpha Vantage, and Finnhub for historical and real-time market data. Similarly, Google Trends lacks an official API, making scraping unreliable; thus, the Glimpse API is identified as a reliable alternative for accurate, real-time search trend data (Glimpse, 2025). StockTwits, a dedicated platform for investor sentiment, offers valuable data via its API, though rate limits and privacy considerations must be managed (StockTwits, n.d.; The impact of social media sentiment, 2025).

The core trading algorithm will be implemented in Go, chosen for its superior speed, efficiency, and concurrency features, which are essential for low-latency execution in a dynamic trading environment. Rigorous backtesting will be a fundamental step, simulating real-world conditions by accounting for transaction costs (slippage, commissions), market impact, and liquidity risks, while actively avoiding data biases such as over-optimization. Performance will be evaluated using key metrics like Sharpe Ratio, Volatility, Maximum Drawdown, and Compound Annual Growth Rate (CAGR) to ensure risk-adjusted returns.

- Results (10 points).. **What** did you learn from your research so far?

The research so far confirms that swing trading is a viable strategy for short-to-medium term gains, provided robust risk management is in place. I am still in the early phases of the project and I'm currently working on data collection. I'm running into some issues with APIs in regards to connection and rate limits for free tiers. Specifically, a critical insight gained is the unreliability of free data sources like yfinance and scraped Google Trends data for a professional trading fund, underscoring the necessity of investing in reliable, paid API providers. Furthermore, operating an actively traded fund introduces substantial legal and regulatory compliance requirements that must be addressed if it were to go into production, and rigorous backtesting, which accounts for real-world costs and biases, is paramount for validating strategy performance before live deployment

- Conclusions (10 points).. **So, what** does it all mean? Do you have any concerns about the term project at this point?

The project's vision for an algorithmic swing trading fund, leveraging a hybrid Python and Go architecture with a multi-modal data strategy, is technically sound and aligns with modern quantitative finance practices. However, several critical concerns must be proactively addressed for the fund's long-term success and credibility. Firstly, the immediate need to transition from unreliable free data sources to robust, paid APIs is paramount to ensure data quality and system stability. Secondly, the significant legal and regulatory complexities associated with operating an "actively traded fund" demand dedicated planning and compliance efforts. Lastly, the algorithmic strategy must incorporate sophisticated processing for sentiment data to filter noise and correctly interpret potentially contrarian signals from sources like Google Trends, while rigorous backtesting, accounting for real-world costs and biases, is essential to validate the strategy's true viability in live markets.

Update for Checkpoint B:

As stated in the previous checkpoint, I am doing my data collection in Python to use its extensive libraries and open sourced features. I have used yFinance (pivoting from Polygon due to API limitations) to collect data from "all" tickers in the S&P 500 from 01-01-1999 to 06-01-2025 and save the results in CSV files, one for each ticker. The log daily return was also calculated for each asset and is included in the resulting CSV. This timeframe is crucial as it encompasses major market events like the dot-com bubble, the 2008 financial crisis, and the COVID-19 pandemic, allowing for a robust test of my strategy's resilience. This data will be used for the identification of assets as well as for training the model(s).

While I initially planned to use "more robust" solutions than yFinance to collect stock data, it is difficult to surpass its ease of use. I will still (hopefully) look to employ a different API that is able to provide up-to-the-minute data when actually executing the strategy.

Although my buy-and-hold strategy is based on fundamental analysis and long-term trends, I recognize that past performance is not a guarantee of future results. A simple backtest on historical data alone is insufficient to build confidence in the strategy's long-term viability. Therefore, I will employ Monte Carlo simulation to account for market uncertainty and test my strategy across thousands of plausible future scenarios, as recommended in the checkpoint writeup. Furthermore, I will use the historical data from 1999-2025 to estimate the statistical properties of the asset returns (mean, volatility, and correlation). I will then generate synthetic time series data for hundreds of new 26-year periods. This synthetic data will preserve the key statistical dependencies and properties observed in the historical record, ensuring the simulated scenarios are realistic representations of market behavior. This will allow me to evaluate my strategy's performance not just on a single past trajectory, but on a wide distribution of possible market outcomes.

I will perform a walk-forward backtest on the Monte-Carlo-generated data and will "rebalance" the portfolio at the start (e.g., in January 1999) based on my initial buy-and-hold selection criteria and then hold that allocation for the entire period. This will simulate the behavior of a patient investor who is not swayed by short-term market fluctuations. As suggested, I will use Alpha, Beta, and the Sharpe Ratio to assess the viability of my fund and compare it to benchmarks.

Update for Checkpoint C:

I am very happy with the progress that I made as part of this checkpoint. I now have at least the basics of an algorithm that identifies and executes trades and creates an interactive data visualization html to look at the results. As of right now, the "value" of my ETF is updated only when a trade executes; I am currently working on reconfiguring some things so that the actual "real-time" value is represented so the chart is more "sawtooth" and less "stair stepper." The obvious next step is to use Monte-Carlo generated data for me to execute walk-forward backtesting to test the true validity of my trading approach. Currently, the algorithm is using a momentum strategy utilizing 50- and 200-day moving averages, but I will likely tweak it to also factor in mean reversion or swing trading. As of right now, I'm keeping an open mind and would love for the data to guide me to an "ideal" algorithm!

References:

Bollen, J., Mao, H., & Zeng, X. (2011). Twitter mood predicts the stock market. *Journal of Computational Science*, *2*(1), 1-8.

Fernando, J. (2024, June 28). *Swing trading: A short-term strategy*. Investopedia. Retrieved from https://www.investopedia.com/terms/s/swingtrading.asp

Zhang, J., Song, Y., & Chen, J. (2020). Machine learning in algorithmic trading: A survey. *Journal of Quantitative Finance*, *20*(7), 1097-1115.