# 1. Problem Size & Pain Points

**High User Demand:** A significant subset of TradingView’s ~50+ million users have repeatedly requested multi-symbol **portfolio backtesting** capabilities[[1]](https://www.reddit.com/r/TradingView/comments/16jhyvm/run_a_single_strategy_against_multiple_symbols_in/#:~:text=this%20is%20a%20need%20of,strategy%20on%20multiple%20symbols%20simultaneously)[[2]](https://www.tradingview.com/pine-script-docs/faq/strategies/#:~:text=How%20can%20I%20backtest%20multiple,symbols). On the official Pine Script FAQ, TradingView explicitly confirms that *“each strategy runs on one symbol at a time”*, advising users to manually switch symbols or export results for multi-market evaluation[[2]](https://www.tradingview.com/pine-script-docs/faq/strategies/#:~:text=How%20can%20I%20backtest%20multiple,symbols)[[3]](https://www.tradingview.com/pine-script-docs/faq/strategies/#:~:text=,a%20strategy%20on%20different%20symbols). This limitation has become one of the most upvoted feature gaps. For example, in TradingView’s own community forum, users call portfolio backtesting *“a very essential feature… already supported by other platforms”*[[4]](https://www.reddit.com/r/TradingView/comments/14wwq81/portfolio_backtest/#:~:text=portfolio%20backtest%20%3A%20r%2FTradingView%20,already%20supported%20by%20other%20platforms). A Reddit feature request post similarly laments: *“Need the feature to backtest a strategy with a portfolio of multiple symbols… TradingView does not have the ability for portfolio testing”*[[1]](https://www.reddit.com/r/TradingView/comments/16jhyvm/run_a_single_strategy_against_multiple_symbols_in/#:~:text=this%20is%20a%20need%20of,strategy%20on%20multiple%20symbols%20simultaneously).

**User Frustration & Quotes:** Across forums and social media, traders voice frustration at the **tedious workarounds** required. One power-user describes the current process: *“Now this has to be done manually – having the strategy applied to a chart, then click all the symbols one-by-one and record the results.”* This one-by-one approach is painfully slow for those testing strategies on dozens of stocks or pairs. Another user echoes how *“it’s such a pain going through manually exporting results!”* when trying to analyze multiple tickers[[5]](https://www.reddit.com/r/TradingView/comments/s9uplg/need_the_ability_to_export_backtesting_results/#:~:text=%E2%80%A2%20%202y%20ago). These pain points are reinforced by dozens of similar threads on Reddit (r/TradingView, r/algotrading) and TradingView’s ideas board. In r/TradingView, a TradingView team member acknowledged the need but gave no timeline, suggesting the idea is noted but not yet planned[[6]](https://stackoverflow.com/questions/61862243/pine-script-multiple-symbol-strategy#:~:text=PineCoders) – further fueling user impatience.

**Frequency of Complaints:** Requests for multi-symbol backtesting appear **frequently and consistently**. A search of r/TradingView shows multiple posts over the past 5+ years asking for portfolio strategy support, often with significant upvotes[[1]](https://www.reddit.com/r/TradingView/comments/16jhyvm/run_a_single_strategy_against_multiple_symbols_in/#:~:text=this%20is%20a%20need%20of,strategy%20on%20multiple%20symbols%20simultaneously)[[7]](https://www.reddit.com/r/TradingView/comments/s9uplg/need_the_ability_to_export_backtesting_results/#:~:text=PrimordialRocks). One recent comment (9 months ago) called the lack of multi-symbol testing *“the #1 most useful suggestion I’ve ever seen… what are you doing with it?”*, highlighting how anticipated this feature is[[8]](https://www.reddit.com/r/TradingView/comments/16jhyvm/run_a_single_strategy_against_multiple_symbols_in/#:~:text=%E2%80%A2%20%209mo%20ago). Another user bluntly stated that *“the only reason traders are thinking of other platforms is [because] TradingView does not have the ability for portfolio testing”*[[1]](https://www.reddit.com/r/TradingView/comments/16jhyvm/run_a_single_strategy_against_multiple_symbols_in/#:~:text=this%20is%20a%20need%20of,strategy%20on%20multiple%20symbols%20simultaneously). Clearly, **many active users – especially those developing algorithms – feel hamstrung by single-chart backtesting**.

**Common Workarounds:** In the absence of native support, users resort to clunky workarounds:

* **CSV Exports:** Traders run a strategy on each symbol individually, export each result set, then merge data in Excel/Google Sheets. One feature request notes *“it’s pain exporting each result set, renaming it, and adding ticker info”* for dozens of files[[9]](https://www.reddit.com/r/TradingView/comments/s9uplg/need_the_ability_to_export_backtesting_results/#:~:text=Backtesting%20strategies%20across%20different%20securities,adding%20ticker%20information%20in%20it)[[10]](https://www.reddit.com/r/TradingView/comments/s9uplg/need_the_ability_to_export_backtesting_results/#:~:text=across%20multiple%20security%20and%20then,adding%20ticker%20information%20in%20it). Third-party services like BacktestBase have even emerged to streamline this – letting users upload multiple TradingView strategy .xlsx exports and then computing combined portfolio metrics[[11]](https://www.backtestbase.com/#:~:text=,settings%20and%20parameters%20worked%20best)[[12]](https://www.backtestbase.com/#:~:text=,TradingView%20backtests%20in%20one%20place). The popularity of such tools (BacktestBase reports parsing 2,847 strategy files) underscores how widespread this manual export method is.
* **Multi-chart Hacks:** Some users attempt using up to 8 chart layouts (TradingView Premium) to run the same strategy on multiple symbols in parallel, then mentally or manually aggregate results. This is error-prone and effectively still single-symbol, since TradingView does not provide a consolidated P/L or risk metric across charts.
* **Pine Script “pseudo-portfolios”:** Advanced coders use the request.security() function to fetch other symbols’ data within one script. However, **orders cannot actually be executed on those secondary symbols**[[13]](https://stackoverflow.com/questions/61862243/pine-script-multiple-symbol-strategy#:~:text=10). At best, this hack might simulate signals from multiple tickers on one chart, but **it doesn’t produce true portfolio equity curves or trade-level combines**. Stack Overflow answers confirm that *“orders are only generated on the chart’s symbol… you can fetch other tickers for calcs, but orders cannot be executed on them”*[[13]](https://stackoverflow.com/questions/61862243/pine-script-multiple-symbol-strategy#:~:text=10). In short, Pine Script cannot backtest a basket of assets in one go.
* **External Tools:** Some traders export TradingView data to external backtesting engines (like Python’s Backtrader or QuantConnect’s Lean CLI) for multi-asset testing. But this requires programming skill and forfeits the ease of TradingView’s GUI. Others use browser automation or Chrome extensions to cycle through a watchlist and scrape results – a risky approach given TradingView’s Terms of Service explicitly forbid *“any automated data collection… scripts, bots or extensions”*[[14]](https://www.tradingview.com/support/solutions/43000674726-why-is-my-account-banned-due-to-suspicious-activity/#:~:text=,TradingView%20content%20is%20expressly%20forbidden)[[15]](https://www.tradingview.com/support/solutions/43000674726-why-is-my-account-banned-due-to-suspicious-activity/#:~:text=Please%20note%20that%20the%20use,are%20for%20manual%20use%20only) (violators face bans).

**Impacted User Segments:** The pain is most acute for **algorithmic retail traders** and **quant hobbyists** on TradingView. These users often develop strategies meant to trade a *portfolio* of symbols (for diversification or higher signal frequency). For instance, a trend-following strategy across 20 stocks, or a mean-reversion pair trading system – both impossible to fully backtest on TradingView today. **Prop firm traders** and those in funded challenges are also affected: they typically trade multi-asset portfolios to meet profit targets with controlled risk, and they lament not being able to test combined strategy behavior (e.g. how simultaneous long EUR/USD and short USD/JPY strategies interact on account equity). Casual retail traders (single-instrument focused) feel it less, but as soon as a user progresses to running multiple strategies or symbol rotation, they hit this wall. A scan of community comments shows even semi-discretionary swing traders want to test baskets of stocks against a strategy screen[[1]](https://www.reddit.com/r/TradingView/comments/16jhyvm/run_a_single_strategy_against_multiple_symbols_in/#:~:text=this%20is%20a%20need%20of,strategy%20on%20multiple%20symbols%20simultaneously).

**Key Pain Points Summary:** (1) *Inability to see true portfolio equity curves and drawdowns.* Traders worry a strategy that looks good on each symbol separately may perform poorly when combined (due to concurrent positions, capital allocation, correlations, etc.)[[16]](https://www.reddit.com/r/TradingView/comments/16jhyvm/run_a_single_strategy_against_multiple_symbols_in/#:~:text=have%20the%20ability%20for%20portfolio,strategy%20on%20multiple%20symbols%20simultaneously). (2) *Time-consuming workflow:* switching tickers 20+ times and collating results – users report running **dozens of backtests per day** which becomes untenable manually[[17]](https://www.tradingview.com/u/mnspppppp/#:~:text=Trader%20mnspppppp%20%E2%80%94%20Trading%20Ideas,GBPUSD%3A%20GU%20BUY). (3) *Lost opportunities:* Without portfolio backtesting, users cannot easily optimize allocations or test sector rotation strategies on TradingView, pushing them to seek other platforms. One user in frustration wrote that the lack of multi-symbol support *“makes me feel like I’m betting on the wrong horse”*[[18]](https://www.reddit.com/r/TradingView/comments/goq969/feature_requests_comment_yours_here/#:~:text=%E2%80%A2%20%205y%20ago) – indicating some are reconsidering their loyalty to the platform. All these factors illustrate a sizable *unmet need and pain* in TradingView’s otherwise robust feature set.

**Sources:**

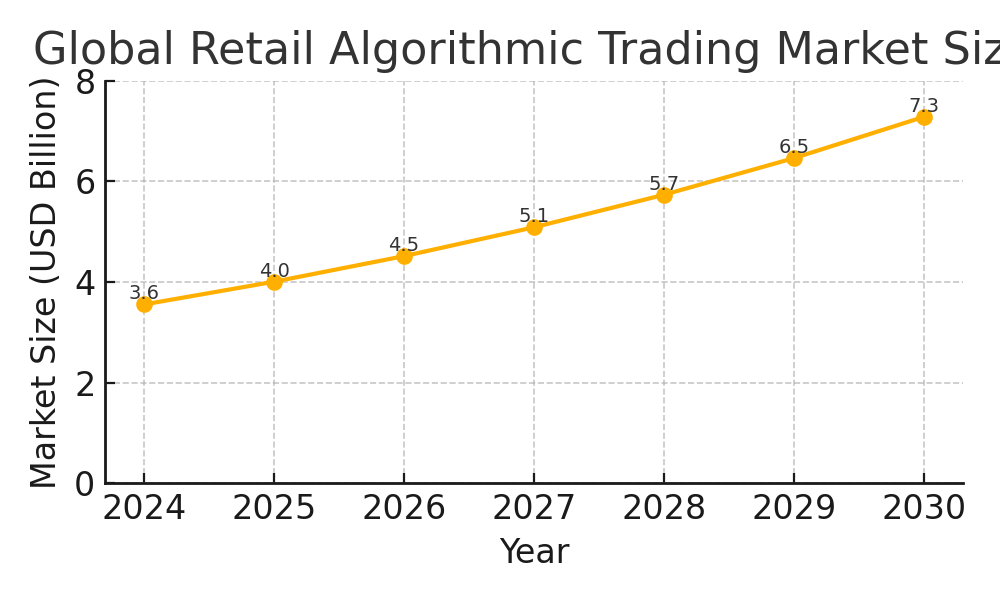
* Reddit – user feature requests for portfolio backtesting[[1]](https://www.reddit.com/r/TradingView/comments/16jhyvm/run_a_single_strategy_against_multiple_symbols_in/#:~:text=this%20is%20a%20need%20of,strategy%20on%20multiple%20symbols%20simultaneously)[[9]](https://www.reddit.com/r/TradingView/comments/s9uplg/need_the_ability_to_export_backtesting_results/#:~:text=Backtesting%20strategies%20across%20different%20securities,adding%20ticker%20information%20in%20it)
* TradingView Pine Script FAQ – confirms single-symbol limitation[[2]](https://www.tradingview.com/pine-script-docs/faq/strategies/#:~:text=How%20can%20I%20backtest%20multiple,symbols)
* TradingView community quotes – manual export pain and competitor temptation[[1]](https://www.reddit.com/r/TradingView/comments/16jhyvm/run_a_single_strategy_against_multiple_symbols_in/#:~:text=this%20is%20a%20need%20of,strategy%20on%20multiple%20symbols%20simultaneously)
* StackOverflow – Pine script multi-symbol limitation[[13]](https://stackoverflow.com/questions/61862243/pine-script-multiple-symbol-strategy#:~:text=10)
* TradingView TOS – prohibits automation (impacting workarounds)[[14]](https://www.tradingview.com/support/solutions/43000674726-why-is-my-account-banned-due-to-suspicious-activity/#:~:text=,TradingView%20content%20is%20expressly%20forbidden)[[15]](https://www.tradingview.com/support/solutions/43000674726-why-is-my-account-banned-due-to-suspicious-activity/#:~:text=Please%20note%20that%20the%20use,are%20for%20manual%20use%20only)

# 2. Market Size & Opportunity

**TradingView’s User Base (TAM):** TradingView boasts **over 50 million active traders and investors globally** as of 2024[[19]](https://www.brokersview.com/news/tadingview-usage-stats-for-2024-224255#:~:text=Key%20TradingView%20Stats), and the number is still growing (the company’s own site now claims **100M+ worldwide users** in 2025)[[20]](https://www.tradingview.com/about/#:~:text=Hello%2C%20Trader). This represents a massive Total Addressable Market for any trading tool or feature targeting its users. Even if we focus on the subset interested in algorithmic strategy backtesting, the numbers are substantial. TradingView’s community has *over 10 million published custom scripts and ideas*[[21]](https://www.brokersview.com/news/tadingview-usage-stats-for-2024-224255#:~:text=,3%20million%20on%20TradingView) – a proxy indicating millions of users are creating or using strategies. This aligns with the rise of the **retail quant** segment: everyday traders using code or automation in their approach.

Beyond TradingView, the broader **retail algorithmic trading community** is large and expanding. Competing platforms give a sense of scale: QuantConnect reports a global community of **~420,000** registered quant traders[[22]](https://www.quantconnect.com/#:~:text=Platform,quant%20research%20community%20in), and other code-oriented platforms (like MetaTrader’s EA community, NinjaTrader, etc.) each have hundreds of thousands of users. Summing across these, it’s reasonable to estimate **several million retail traders globally are involved in algorithmic strategy development** – the broader TAM for a portfolio backtesting solution that could attract users both within and outside TradingView.

**Global Retail Algo Trading Market:** In dollar terms, the *retail* algorithmic trading market (software, services, etc. for non-institutional algo traders) was valued at about **$3.55 billion in 2024**[[23]](https://www.grandviewresearch.com/horizon/statistics/algorithmic-trading-market/type-of-trader/retail-investors/global#:~:text=Retail%20Investors%20,Market%20Statistics)[[24]](https://www.grandviewresearch.com/horizon/statistics/algorithmic-trading-market/type-of-trader/retail-investors/global#:~:text=The%20global%20retail%20investors%20algorithmic,from%202024%20to%202030). It is forecast to grow at ~12.7% CAGR, reaching roughly **$7–8 billion by 2030**[[23]](https://www.grandviewresearch.com/horizon/statistics/algorithmic-trading-market/type-of-trader/retail-investors/global#:~:text=Retail%20Investors%20,Market%20Statistics). This growth is outpacing the institutional segment – retail algo trading is a fast-expanding niche as technology lowers barriers. **Figure 1** illustrates this trajectory. The strong growth rate (double-digit CAGR) highlights the opportunity: more retail traders are expected to adopt algorithmic strategies and will demand robust backtesting tools.

  
*Figure 1: Global Retail Algorithmic Trading Market Size (2024–2030). The retail algo trading sector is projected to roughly double from ~$3.6B in 2024 to ~$7.3B by 2030, ~12.7% CAGR*[*[23]*](https://www.grandviewresearch.com/horizon/statistics/algorithmic-trading-market/type-of-trader/retail-investors/global#:~:text=Retail%20Investors%20,Market%20Statistics)*. Rising adoption of automated trading among individuals drives this growth.*

Within TradingView’s own ecosystem, there is a clear *Serviceable Addressable Market (SAM)* for a portfolio backtesting solution. **Who are these users?** Likely: paying subscribers on higher tiers (Pro/Pro+ and Premium) who use strategy tester heavily, Pine Script coders, and serious traders managing multiple assets. TradingView’s **Premium tier** is tailored to algotraders (extended data, up to 20,000 bars, multiple charts, etc.), and those users – who pay ~$60/month – are prime candidates for portfolio testing. If we assume even *20% of TradingView’s 50M users are actively using the Strategy Tester or scripts*, that’s ~10 million users potentially interested. Even more conservatively, TradingView had ~**~1.5 million mobile app reviews** (with a 4.9★ rating) and extremely high engagement metrics[[21]](https://www.brokersview.com/news/tadingview-usage-stats-for-2024-224255#:~:text=,3%20million%20on%20TradingView)[[25]](https://www.brokersview.com/news/tadingview-usage-stats-for-2024-224255#:~:text=TradingView%20boasts%20a%20staggering%2050,actively%20exploring%20the%20platform%27s%20capabilities); if we estimate perhaps 5–10% of its monthly active users are strategy/automation-focused, we get on the order of **5 million** as a reachable segment in the near term. This would be the **SAM**: those who would seriously consider adopting a multi-symbol backtesting tool.

We can further narrow to a **Serviceable Obtainable Market (SOM)** for an initial product offering. Realistically, not all 5–10M SAM will convert immediately – many need education or have inertia. A new standalone solution might capture, say, 1–2% of that SAM in its first few years if executed well. **For instance:** if SAM is 5,000,000 users, 1% conversion is 50,000 paying users. This ballpark seems feasible given that niche platforms like QuantConnect or TrendSpider have on the order of tens of thousands of users today. It’s also instructive that **TradingView Premium** (the top tier) likely has a user count in the low hundreds of thousands (TradingView’s estimated annual revenue ~$8.3M[[26]](https://www.brokersview.com/news/tadingview-usage-stats-for-2024-224255#:~:text=,3%20million%20on%20TradingView), much of which comes from subscriptions, suggests on the order of 200k–300k paying users total if ARPU is ~$30/year, though this is a rough estimate). The subset willing to pay extra for advanced backtesting could be tens of thousands.

**TAM/SAM/SOM Visualization:** We can quantify three levels:

* **TAM (Global Retail Quant Market):** ~50 million traders on TradingView and other platforms combined, equating to ~$3.5B market in 2024 growing to ~$7B by 2030[[23]](https://www.grandviewresearch.com/horizon/statistics/algorithmic-trading-market/type-of-trader/retail-investors/global#:~:text=Retail%20Investors%20,Market%20Statistics). This is everyone who *could* benefit from multi-asset backtesting globally.
* **SAM (Active Strategy Developers on TradingView et al.):** On the order of 5–10 million users. E.g., TradingView’s active script users plus communities on other tools. These are traders *likely to adopt* a dedicated portfolio backtester if available. They have shown intent by coding strategies or running lots of backtests.
* **SOM (Initial Obtainable Market):** Perhaps 50k–100k users in the first 1–2 years, given competition and adoption curves (this could expand if integrated into TradingView directly). For a standalone, capturing even 0.5–1% of TradingView’s engaged user base would mean tens of thousands of users – a very achievable target in a community where feature requests gather hundreds of upvotes.

To illustrate the revenue opportunity, consider a modest conversion scenario: **50,000 users paying ~$20/month** for a portfolio backtest service (see Section 5 for pricing) yields **$12 million annual revenue** – already exceeding TradingView’s own estimated annual revenue[[26]](https://www.brokersview.com/news/tadingview-usage-stats-for-2024-224255#:~:text=,3%20million%20on%20TradingView). An aggressive scenario (100k users) at similar ARPU would be ~$24M/year. This suggests substantial business potential (Section 5 will detail monetization).

**Geographic Reach:** TradingView’s audience is global (top user countries include India ~15%, US ~11%, etc.[[27]](https://www.brokersview.com/news/tadingview-usage-stats-for-2024-224255#:~:text=TradingView%20Desktop%20Traffic%20Distribution%20%EF%BC%88By,Country%EF%BC%89)[[28]](https://www.brokersview.com/news/tadingview-usage-stats-for-2024-224255#:~:text=TradingView%20caters%20to%20a%20global,28)), meaning the solution’s market isn’t confined to one region. The *retail quant boom* is worldwide – from Indian systematic traders to US Robinhood-era quants. Notably, TradingView supports 20 languages and has penetration in 180+ countries[[29]](https://ibsintelligence.com/ibsi-news/tradingview-hits-record-550-million-unique-users/#:~:text=Individual%2C%20non,their%20own%20personal%20financial%20decisions)[[28]](https://www.brokersview.com/news/tadingview-usage-stats-for-2024-224255#:~:text=TradingView%20caters%20to%20a%20global,28), so a portfolio backtesting tool could have global uptake if localized properly. There’s also growth in retail APIs and algorithmic interest in emerging markets (Asia-Pacific retail algos projected ~10.8% CAGR, slightly above global average[[30]](https://www.mordorintelligence.com/industry-reports/algorithmic-trading-market#:~:text=,CAGR)).

**Market Growth Drivers:** The strong growth in this market is fueled by a few trends that a portfolio backtesting solution can ride:

* **DIY Investing & “Retail Revolution”:** A *“global retail investor revolution”* started around 2020–2021, with millions of new entrants in markets[[31]](https://ibsintelligence.com/ibsi-news/tradingview-hits-record-550-million-unique-users/#:~:text=Fuelled%20by%20the%20recent%20global,critical%20force%20in%20financial%20markets). Many of these new retail traders (Gen-Z, etc.) are tech-savvy and comfortable with tools – they won’t shy away from advanced backtesting if it’s accessible. TradingView’s own data showed a 400% increase in accounts during pandemic lockdowns[[32]](https://ibsintelligence.com/ibsi-news/tradingview-hits-record-550-million-unique-users/#:~:text=TradingView%2C%20which%20celebrates%20its%2010,of%20today%E2%80%99s%20global%20population). Now these traders are looking to level up with better analytics.
* **Freemium Model Limits:** TradingView’s freemium nature means many get a taste of strategy testing for free and then crave more sophisticated capabilities (like multi-asset analysis). This creates *latent demand* that a specialized paid add-on could monetize.
* **Rise of Retail Algo Services:** The success of platforms like QuantConnect, MetaTrader (for EAs), and even newer services (Alpaca, Streak, etc. offering simple multi-asset backtests) shows that retail traders are willing to pay for algorithmic infrastructure. For example, quant trading services in India allow up to 1000 backtests per day at higher subscription tiers[[33]](https://algotest.in/blog/streak-vs-tradetron/#:~:text=Streak%20vs,For%20more%20information%2C%20please) – indicating an appetite for high-volume strategy experimentation.

In summary, the **market opportunity is sizeable and growing**. The TAM is in the tens of millions of users and several billion dollars in economic value. The SAM – TradingView’s algorithmic user base plus similar users elsewhere – likely numbers a few million enthusiasts hungry for better tools. Capturing even a fraction of these could result in a thriving user community and multi-million dollar revenue stream. The combination of TradingView’s massive reach and the retail quant market’s expansion creates a “perfect storm” opportunity for a portfolio backtesting solution to fill a glaring gap.

**Sources:**

* BrokersView – TradingView stats (50M users, 200M monthly visits)[[19]](https://www.brokersview.com/news/tadingview-usage-stats-for-2024-224255#:~:text=Key%20TradingView%20Stats)[[25]](https://www.brokersview.com/news/tadingview-usage-stats-for-2024-224255#:~:text=TradingView%20boasts%20a%20staggering%2050,actively%20exploring%20the%20platform%27s%20capabilities)
* Grandview Research – retail algo trading market size & CAGR[[23]](https://www.grandviewresearch.com/horizon/statistics/algorithmic-trading-market/type-of-trader/retail-investors/global#:~:text=Retail%20Investors%20,Market%20Statistics)[[24]](https://www.grandviewresearch.com/horizon/statistics/algorithmic-trading-market/type-of-trader/retail-investors/global#:~:text=The%20global%20retail%20investors%20algorithmic,from%202024%20to%202030)
* TradingView “About” page – 100M+ global user claim[[20]](https://www.tradingview.com/about/#:~:text=Hello%2C%20Trader)
* QuantConnect site – community size ~420k quants[[22]](https://www.quantconnect.com/#:~:text=Platform,quant%20research%20community%20in)
* IBS Intelligence – retail investor boom (550M unique visitors since 2011)[[29]](https://ibsintelligence.com/ibsi-news/tradingview-hits-record-550-million-unique-users/#:~:text=Individual%2C%20non,their%20own%20personal%20financial%20decisions)[[32]](https://ibsintelligence.com/ibsi-news/tradingview-hits-record-550-million-unique-users/#:~:text=TradingView%2C%20which%20celebrates%20its%2010,of%20today%E2%80%99s%20global%20population)
* TradingView community scripts – 10M+ custom scripts shared[[21]](https://www.brokersview.com/news/tadingview-usage-stats-for-2024-224255#:~:text=,3%20million%20on%20TradingView)

# 3. Competitive Gap Analysis

Despite TradingView’s popularity, **its current offerings do not meet multi-symbol backtesting needs**, forcing serious users to seek alternatives. Below we compare TradingView’s capabilities to key competitors, highlighting feature gaps and user feedback:

**TradingView (Status Quo):** TradingView’s strategy tester is limited to one symbol at a time by design[[2]](https://www.tradingview.com/pine-script-docs/faq/strategies/#:~:text=How%20can%20I%20backtest%20multiple,symbols). No subscription tier (Free, Pro, Pro+, Premium) unlocks portfolio testing – it’s a platform limitation, not a paywall feature. Premium users do get more bars of data (20K) and up to 8 charts in a layout, but *even the highest tier cannot simulate a multi-asset strategy* (users would have to mentally aggregate results from 8 separate charts, which is not realistic)[[34]](https://www.reddit.com/r/TradingView/comments/goq969/feature_requests_comment_yours_here/#:~:text=%E2%80%A2%20%205y%20ago)[[35]](https://www.reddit.com/r/TradingView/comments/goq969/feature_requests_comment_yours_here/#:~:text=Support%20replied%20to%20my%20message,no%20timeline%20could%20be%20given). This gap is frequently mentioned in “Premium vs Others” debates: traders note that **no amount of extra indicators or data resolves the inability to test a basket**. TradingView staff (PineCoders moderators) have acknowledged the portfolio backtest request is logged but *“not in the pipeline yet”*[[6]](https://stackoverflow.com/questions/61862243/pine-script-multiple-symbol-strategy#:~:text=PineCoders). The implication is that **TradingView’s architecture currently isn’t built for cross-symbol strategy tracking**, leaving a *competitive gap*.

**TrendSpider:** TrendSpider is a newer platform focusing on automated technical analysis and multi-timeframe scanning. It *does* offer a strategy tester with **no coding required** (point-and-click conditions) and even AI-generated strategies[[36]](https://www.newtrading.io/trendspider-review/#:~:text=While%20it%E2%80%99s%20very%20powerful%2C%20there,the%20heavy%20lifting%20for%20you)[[37]](https://www.newtrading.io/trendspider-review/#:~:text=TrendSpider%20is%20an%20AI,level%20models). However, TrendSpider similarly **limits backtests to one symbol at a time** – it is chart-based like TradingView. There is no native portfolio-level simulation of multiple tickers simultaneously. TrendSpider emphasizes “multi-timeframe” analysis (e.g. check a strategy on weekly and daily chart of the *same symbol*), not multi-asset portfolios[[38]](https://www.newtrading.io/trendspider-review/#:~:text=No,refine%20strategies%20without%20programming%20knowledge)[[39]](https://www.newtrading.io/trendspider-review/#:~:text=Another%20feature%20is%20TrendSpider%E2%80%99s%20multi,Whether%20it%E2%80%99s%20stocks%2C%20crypto%2C%20or). User feedback indicates that while TrendSpider’s no-code backtesting is user-friendly, it has some limitations: *“some trading style limitations and differences between backtesting logic and real bot execution can be frustrating”*[[40]](https://www.newtrading.io/trendspider-review/#:~:text=match%20at%20L472%20Meanwhile%2C%20a,bot%20execution%20can%20be%20frustrating). **Missing features for TrendSpider** relevant here include the inability to combine strategies across several symbols or to see combined equity curves. So, although TrendSpider competes with TradingView on charting and scanning, it **does not fill the portfolio backtest gap** – users would face the same one-symbol-at-a-time issue. (TrendSpider’s own documentation and reviews don’t mention portfolio backtesting, confirming it’s not a feature.)

**QuantConnect:** QuantConnect is a cloud-based algorithmic trading platform allowing multi-asset strategy coding in C# or Python. It **directly addresses multi-symbol backtesting** – users can program strategies that trade any number of assets concurrently (e.g. a long/short equity portfolio, pair trades, etc.). In terms of capabilities, QuantConnect’s LEAN engine is very powerful: it handles portfolio-level metrics, strategy branching, and event-driven simulation across equities, Forex, crypto, etc. However, **the gap here is usability and target audience**. QuantConnect is aimed at proficient coders; there is *no graphical interface to build strategies*. This is a stark contrast to TradingView’s approachable Pine Script and visual charts. As a result, many TradingView users (often non-professional coders) find the learning curve steep. Moreover, **user feedback on QuantConnect has flagged some reliability issues** in recent years. One detailed Reddit post complained that *“numerous issues… left me feeling QuantConnect is completely unviable for testing… data inaccuracies, indicators misbehaving, backtests hanging”*[[41]](https://www.reddit.com/r/algotrading/comments/13la56p/what_happened_to_quantconnect/#:~:text=strategies%20and%20take%20advantage%20of,evening%20of%20coding%20%2B%20backtesting)[[42]](https://www.reddit.com/r/algotrading/comments/13la56p/what_happened_to_quantconnect/#:~:text=Furthermore%2C%20simple%20quality%20of%20life,much%20less%20active%20there%20now). This suggests that even though QuantConnect fills the multi-asset backtest feature, some retail users have trust issues with its data quality or platform stability. Another quant wrote that forums go unanswered and the experience has declined[[43]](https://www.reddit.com/r/algotrading/comments/13la56p/what_happened_to_quantconnect/#:~:text=line%20of%20your%20code%20,much%20less%20active%20there%20now). *Competitive gap:* TradingView, despite lacking portfolio features, is seen as more user-friendly and stable; QuantConnect has the features but not the ease-of-use or community vibe of TradingView. This opens an opportunity to combine TradingView’s strengths (UI, community) with QuantConnect’s (multi-asset engine). In short, **QuantConnect demonstrates demand for multi-symbol backtesting (275K+ users, by its claims)**[**[44]**](https://www.quantconnect.com/pricing/#:~:text=QuantConnect%20is%20a%20multi,than%20275000%20quants%20and%20engineers)**, but many TradingView users hesitate to migrate due to coding requirements and some negative experiences**.

**Trade Ideas:** Trade Ideas is a veteran stock scanning platform with an *AI-driven* spin. It offers a feature called **“OddsMaker”** (available in their Premium tier) which *can backtest trading strategies across multiple stocks*, particularly those identified via its scanners. In effect, if you have a scanning strategy (say “buy gap-up stocks and hold for 1 day”), Trade Ideas will simulate that across many tickers historically. This **multi-symbol backtesting ability** is a key selling point and likely why independent reviews give Trade Ideas a **5/5 rating in backtesting vs. TradingView’s 4.5/5**[[45]](https://www.liberatedstocktrader.com/tradingview-vs-trade-ideas/#:~:text=BacktestingImage%3A%20Backtesting%20Score,140%20%E2%98%85%E2%98%85%E2%98%85%E2%98%85%E2%9C%A9%E2%98%85%E2%98%85%E2%98%85%E2%98%85%E2%98%85). However, Trade Ideas has some drawbacks in this realm: it is *primarily a scanner*, not a full charting or coding platform. Users can’t easily code complex entry/exit logic as they could in Pine Script or Python; they rely on the AI or preset filters. Also, Trade Ideas is **pricey** – ~$178/month on annual plan (or $254 month-to-month) for Premium which includes backtesting[[46][47]](https://www.trade-ideas.com/pricing/#:~:text=TI%20Premium%3A%20AI,signals%20to%20maximize%20profits). That’s significantly higher than TradingView’s $59/month Premium. Many retail traders find this cost hard to justify unless they are very active day traders. Additionally, Trade Ideas covers mainly US equities (and some ETFs); it’s not as multi-asset (no crypto, limited Forex) as TradingView. *User feedback:* Trade Ideas is often praised for powerful scanning but criticized for an outdated interface and lack of broad asset class support. In summary, **Trade Ideas partially fills the gap (multi-stock backtests), but is a niche, high-cost product with a steep learning curve of its own**. This leaves a competitive opening for a more affordable, user-friendly portfolio tester.

**AmiBroker:** AmiBroker is a desktop technical analysis software known for *excellent portfolio backtesting*. It has been a go-to for system traders for years because it **allows true portfolio-level simulations**: you can define a strategy in its AFL scripting language and run it on a list of symbols with a specified initial capital, and AmiBroker will simulate all trades (with position sizing, etc.) across the portfolio[[48][49]](https://www.amibroker.com/guide/h_portfolio.html#:~:text=AmiBroker%27s%20portfolio%20backtester%20lets%20you,like%20you%20do%20in%20reality). *Gap filled:* Feature-wise, AmiBroker provides everything from portfolio equity curves, risk metrics, correlation analysis between strategy components, to advanced optimization across symbols. The trade-off is **usability and modernity**. AmiBroker is Windows-only, requires writing AFL code (which has a steep learning curve, arguably tougher than Pine Script), and lacks real-time social/community features. Its interface is considered dated and not web-based. Many younger retail traders don’t even know of it, or are deterred by its old-school design. It’s also not cheap upfront – ~$299 for the standard edition license[[50]](https://www.amibroker.com/products.html#:~:text=AmiBroker%20www,Download%20Buy) (though it’s a one-time cost, which some view as a plus). *User feedback:* Among hardcore system traders, AmiBroker is respected for speed and power, but new users often struggle with AFL’s quirks and the lack of an integrated data feed (users must plug in their own data source). **Competitive gap:** TradingView, with its cloud-based ease and huge library of community scripts, had been winning over many who might otherwise use AmiBroker – except those who absolutely need portfolio backtesting. Those users either stick with AmiBroker or endure painful workarounds on TradingView. This suggests a *latent demand* for TradingView-like ease combined with AmiBroker-like portfolio features.

**Other Notable Platforms:** *MetaTrader 5* allows multi-currency backtesting to an extent (expert advisors can be coded to trade multiple symbols), but MT5’s strategy tester still runs one EA at a time, and portfolio-level metrics are rudimentary. *NinjaTrader* and *MultiCharts* similarly support multi-series scripts but are heavy-duty and not widely used by the casual retail crowd. *Python backtesting frameworks* (Backtrader, Zipline, etc.) are used by very advanced hobbyists or quant firms, but again require coding and lack user-friendly GUIs – they’re not direct competitors in the retail product sense, but they show that those determined to do portfolio backtesting will find a way (the barrier being coding skill).

**Feature Comparison Summary:** The table below highlights how TradingView stacks up against the mentioned competitors on key relevant features:

| Platform | Multi-Symbol Backtesting | Portfolio Metrics & Equity Curve | Coding Required? | Pricing (indicative) | Notable Limitations/Feedback |
| --- | --- | --- | --- | --- | --- |
| **TradingView** | **No** (single-symbol only)[[2]](https://www.tradingview.com/pine-script-docs/faq/strategies/#:~:text=How%20can%20I%20backtest%20multiple,symbols) | No combined portfolio stats natively | Pine Script (beginner-friendly code) | Free basic; Paid $15–$60/mo[[21]](https://www.brokersview.com/news/tadingview-usage-stats-for-2024-224255#:~:text=,3%20million%20on%20TradingView) | Max ~20K bars data (Premium)[[34]](https://www.reddit.com/r/TradingView/comments/goq969/feature_requests_comment_yours_here/#:~:text=%E2%80%A2%20%205y%20ago); No walk-forward or multi-symbol. Users request portfolio feature for realism[[1]](https://www.reddit.com/r/TradingView/comments/16jhyvm/run_a_single_strategy_against_multiple_symbols_in/#:~:text=this%20is%20a%20need%20of,strategy%20on%20multiple%20symbols%20simultaneously). |
| **TrendSpider** | No (one symbol at a time) | No (each chart separate) | **No-code** (visual strategy builder) | ~$59–$233/mo (Std to Advanced)[[51]](https://www.newtrading.io/trendspider-review/#:~:text=Fees%20and%20Pricing%20Standard%20%E2%80%93,%E2%80%93%20%241%2C596%2FYear%20Advanced%20%E2%80%93%20%242%2C796%2FYear)[[52]](https://www.newtrading.io/trendspider-review/#:~:text=Advanced%24349%242%2C796%E2%80%93%2020%20open%20workspaces%20%E2%80%93,30K%20depth) | Emphasizes multi-*timeframe* vs multi-asset. Limited assets (stocks, crypto). No community scripting. |
| **QuantConnect** | **Yes** (fully supports multi-asset) | Yes (full portfolio simulation) | Yes – Python/C# coding | Free tier; ~$20–$40/mo for higher resources[[42]](https://www.reddit.com/r/algotrading/comments/13la56p/what_happened_to_quantconnect/#:~:text=Furthermore%2C%20simple%20quality%20of%20life,much%20less%20active%20there%20now) | Steep learning curve; some data quality issues reported[[41]](https://www.reddit.com/r/algotrading/comments/13la56p/what_happened_to_quantconnect/#:~:text=strategies%20and%20take%20advantage%20of,evening%20of%20coding%20%2B%20backtesting); No integrated charts GUI. |
| **Trade Ideas** | **Yes** (scans/backtests across many stocks) | Basic portfolio stats (focus on hit rate, profit factor) | No-code (use AI and filters) | $89/mo (Basic), $178/mo (Premium annual)[[46]](https://www.trade-ideas.com/pricing/#:~:text=TI%20Premium%3A%20AI,signals%20to%20maximize%20profits) | Very expensive for full features; caters to intraday equity traders; limited strategy customization. |
| **AmiBroker** | **Yes** (robust portfolio backtester)[[48]](https://www.amibroker.com/guide/h_portfolio.html#:~:text=AmiBroker%27s%20portfolio%20backtester%20lets%20you,like%20you%20do%20in%20reality) | Yes (extensive metrics & position sizing) | Yes – AFL scripting | ~$299 one-time (Standard)[[50]](https://www.amibroker.com/products.html#:~:text=AmiBroker%20www,Download%20Buy) | Dated interface; Windows only; user must obtain data feed; steep AFL syntax learning curve. |

*Table:* **Feature Gap Comparison** – TradingView vs. key competitors in multi-symbol backtesting. *(Sources: Official documentation and pricing pages*[*[2]*](https://www.tradingview.com/pine-script-docs/faq/strategies/#:~:text=How%20can%20I%20backtest%20multiple,symbols)[*[51]*](https://www.newtrading.io/trendspider-review/#:~:text=Fees%20and%20Pricing%20Standard%20%E2%80%93,%E2%80%93%20%241%2C596%2FYear%20Advanced%20%E2%80%93%20%242%2C796%2FYear)[*[46]*](https://www.trade-ideas.com/pricing/#:~:text=TI%20Premium%3A%20AI,signals%20to%20maximize%20profits)[*[48]*](https://www.amibroker.com/guide/h_portfolio.html#:~:text=AmiBroker%27s%20portfolio%20backtester%20lets%20you,like%20you%20do%20in%20reality)*.)*

From the above, it’s clear **none of the “comparison set” offers a perfect solution for TradingView users**: TrendSpider is easy but lacks the feature; QuantConnect/AmiBroker have the feature but demand programming and/or are not integrated with TradingView’s social/chart ecosystem; Trade Ideas has it for stocks but at high cost and limited flexibility. **This gap is precisely the opportunity** – to provide a **user-friendly, TradingView-integrated (or at least inspired) portfolio backtester** that retains ease of use while delivering multi-symbol capability.

**User Loyalty Risk:** If TradingView doesn’t act, users may eventually churn to an alternate solution that does. Indeed, some users already maintain two platforms (e.g., they chart in TradingView but backtest in AmiBroker or QuantConnect). One Reddit user warned that missing portfolio features is *“making [them] feel like betting on the wrong horse”*[[53]](https://www.reddit.com/r/TradingView/comments/goq969/feature_requests_comment_yours_here/#:~:text=I%27ve%20been%20wanting%20this%20for,betting%20on%20the%20wrong%20horse). Another said *“traders are thinking of other platforms”* due to this gap[[1]](https://www.reddit.com/r/TradingView/comments/16jhyvm/run_a_single_strategy_against_multiple_symbols_in/#:~:text=this%20is%20a%20need%20of,strategy%20on%20multiple%20symbols%20simultaneously). Currently, none of the competitors have the massive community or polished UI TradingView has, which is why many endure the inconvenience. But the first platform to successfully marry multi-asset backtesting with a clean UI could capture this market segment quickly.

In conclusion, **TradingView’s competitive gap lies not in charting or data – where it’s a leader – but squarely in advanced backtesting for multi-asset portfolios**. The competitors underscore two things: (1) The feature is *technically achievable* (others have done it), and (2) delivering it in a user-friendly package is hard (hence each competitor trades off some aspect). This analysis strongly signals a market gap for a solution that can hit the sweet spot between TradingView’s usability and QuantConnect/AmiBroker’s functionality.

**Sources:**

* TradingView Pine Script FAQ – single-symbol only[[2]](https://www.tradingview.com/pine-script-docs/faq/strategies/#:~:text=How%20can%20I%20backtest%20multiple,symbols)
* Reddit (r/TradingView) – user saying others considering leaving[[1]](https://www.reddit.com/r/TradingView/comments/16jhyvm/run_a_single_strategy_against_multiple_symbols_in/#:~:text=this%20is%20a%20need%20of,strategy%20on%20multiple%20symbols%20simultaneously)
* LiberatedStockTrader review – Trade Ideas vs TradingView backtesting rating[[45]](https://www.liberatedstocktrader.com/tradingview-vs-trade-ideas/#:~:text=BacktestingImage%3A%20Backtesting%20Score,140%20%E2%98%85%E2%98%85%E2%98%85%E2%98%85%E2%9C%A9%E2%98%85%E2%98%85%E2%98%85%E2%98%85%E2%98%85)
* AmiBroker guide – portfolio backtesting description[[48][49]](https://www.amibroker.com/guide/h_portfolio.html#:~:text=AmiBroker%27s%20portfolio%20backtester%20lets%20you,like%20you%20do%20in%20reality)
* QuantConnect Reddit review – user issues with data and reliability[[41]](https://www.reddit.com/r/algotrading/comments/13la56p/what_happened_to_quantconnect/#:~:text=strategies%20and%20take%20advantage%20of,evening%20of%20coding%20%2B%20backtesting)[[42]](https://www.reddit.com/r/algotrading/comments/13la56p/what_happened_to_quantconnect/#:~:text=Furthermore%2C%20simple%20quality%20of%20life,much%20less%20active%20there%20now)
* TrendSpider review – mentions community ~15k and focus on no-code backtesting[[36]](https://www.newtrading.io/trendspider-review/#:~:text=While%20it%E2%80%99s%20very%20powerful%2C%20there,the%20heavy%20lifting%20for%20you)[[37]](https://www.newtrading.io/trendspider-review/#:~:text=TrendSpider%20is%20an%20AI,level%20models)

# 4. Competitor Landscape

The landscape for retail algorithmic backtesting tools can be divided into **two categories**:

1. **Direct Competitors** – platforms that offer strategy development/backtesting features to retail traders (with some being all-in-one trading platforms).
2. **Indirect/Workaround Solutions** – unofficial tools, extensions, or niche products users employ to fill the gaps (often specifically for TradingView integration).

Below, we profile key players in each category, focusing on their core features, pricing, popularity, and user trust:

**Direct Competitors:**

* **TrendSpider:** *Core focus:* Automated technical analysis and smart charts. *Features:* Strategy tester (no code), multi-timeframe analysis, alerts, and a unique “Raindrop” volume candle. It does *not* support portfolio backtests (one symbol at a time). *Pricing:* Mid-to-high-end – Standard ~$59/mo, Premium ~$99/mo, Advanced ~$233/mo (when paid annually)[[51]](https://www.newtrading.io/trendspider-review/#:~:text=Fees%20and%20Pricing%20Standard%20%E2%80%93,%E2%80%93%20%241%2C596%2FYear%20Advanced%20%E2%80%93%20%242%2C796%2FYear)[[52]](https://www.newtrading.io/trendspider-review/#:~:text=Advanced%24349%242%2C796%E2%80%93%2020%20open%20workspaces%20%E2%80%93,30K%20depth). *Popularity:* **~15,000+ users**[[36]](https://www.newtrading.io/trendspider-review/#:~:text=While%20it%E2%80%99s%20very%20powerful%2C%20there,the%20heavy%20lifting%20for%20you) (fast-growing, Inc 5000 listed). *User trust:* Generally positive for innovation, but its audience is smaller, and it lacks the social community of TradingView. *Sustainability:* TrendSpider is a venture-funded startup (founded 2016) – they’ve grown via adding AI features. *Limitations:* Limited asset classes (mostly stocks, some crypto/forex), no mobile app, and a learning curve for its advanced tools. It competes on automation (auto trendlines, etc.) more than broad strategy backtesting. **Implication:** TrendSpider might attract some TradingView users who want easier backtesting without coding, but it won’t satisfy those needing multi-symbol tests. There could be room for partnership (TrendSpider data or features integrated?) but more likely it’s a competitor to watch as it evolves.
* **QuantConnect:** *Core focus:* A cloud algorithmic trading platform for quants. *Features:* Multi-asset backtesting (equities, forex, crypto, futures, options), research notebooks, and live trading integration with brokers. Extremely flexible – essentially a cloud quant research environment. *Pricing:* Freemium. Free tier offers unlimited backtesting with some limits; paid tiers ($8–$40/mo range) give more cloud computing power and collaboration features[[42]](https://www.reddit.com/r/algotrading/comments/13la56p/what_happened_to_quantconnect/#:~:text=Furthermore%2C%20simple%20quality%20of%20life,much%20less%20active%20there%20now). *Popularity:* **~275,000+ users** (per site) and claims to be the “biggest quant community”[[22]](https://www.quantconnect.com/#:~:text=Platform,quant%20research%20community%20in). However, active user count may be lower; many sign up out of curiosity. *User trust:* Mixed – respected by experienced coders, but casual traders find it daunting. Recent community chatter suggests a decline in support responsiveness[[54]](https://www.reddit.com/r/algotrading/comments/13la56p/what_happened_to_quantconnect/#:~:text=line%20of%20your%20code%20,much%20less%20active%20there%20now). *Sustainability:* QuantConnect is presumably sustained by its cloud subscriptions and institutional licensing of its LEAN engine. It’s stable, but not widely profitable at retail scale (it’s a niche within a niche). *Limitations:* Requires programming; no native chart GUI (users often backtest via code and then use separate tools for visualization). Some users report data issues and missing ease-of-use features (e.g. handling corporate actions can be manual). **Implication:** QuantConnect is a strong competitor on capability – any new solution should aspire to its breadth of multi-asset support – but there is a *market gap* for a more user-friendly, GUI-driven approach that captures the non-engineer segment.
* **Trade Ideas:** *Core focus:* Real-time AI-powered trade ideas and scanning, primarily for stocks. *Features:* Holly AI (an AI that generates strategies daily), backtesting via the OddsMaker tool (for any strategy defined in their scanner), direct brokerage integration for automated trading, and a rich set of alerts/filters for intraday trading. *Pricing:* **High-end** – Standard $89/mo, Premium $178/mo (annual plans; month-to-month costs more)[[46][47]](https://www.trade-ideas.com/pricing/#:~:text=TI%20Premium%3A%20AI,signals%20to%20maximize%20profits). Often cited as one of the priciest retail trading tools. *Popularity:* Relatively small but devoted user base (likely in the low tens of thousands; precise numbers not public). Many active day traders and prop firm trainees use it for US equities. *User trust:* High within its niche – it’s been around ~20 years. However, outside of stock day-trading circles, awareness is lower. It has a 4.7/5 rating from ~30 user reviews on one site[[55]](https://www.trade-ideas.com/2025/01/19/trade-ideas-leading-the-financial-technology-revolution-why-we-were-voted-most-trusted-trading-platform-by-capterra/#:~:text=With%20an%20outstanding%204,through%20its%20advanced%20AI) (indicative of a small sample). *Sustainability:* Profitable, privately owned. It keeps innovating (recently adding charting, AI improvements). *Limitations:* Focused almost exclusively on **U.S. equities**. Not suitable for multi-asset (no crypto, limited FX or futures coverage). The backtesting (OddsMaker) is powerful for scanning-type strategies but less so for strategies requiring complex sequential logic. Also, no *portfolio* of strategies – it tests a strategy on multiple symbols independently, but doesn’t output a combined equity curve of trading them together (aside from summing results). **Implication:** Trade Ideas covers some of the portfolio testing use-case (multiple symbols) but only for a subset of users (stock traders who can afford it). A more affordable, multi-asset solution could undercut Trade Ideas, especially for swing traders who want portfolio analysis beyond intraday stock picking.
* **AmiBroker:** *Core focus:* Comprehensive technical analysis and system development platform (desktop software). *Features:* In-depth **portfolio backtesting** (position sizing, market timing, multi-currency support[[56]](https://www.amibroker.com/guide/h_pyramid.html#:~:text=Pyramiding%20,features%20are%20supported%20by)), optimization (including Monte Carlo and walk-forward tests), custom metrics via AFL scripting, and an extremely fast backtest engine (handles thousands of stocks with decades of data quickly). *Pricing:* **One-time license** – Standard $299, Professional $369 (the Pro adds real-time features and advanced optimizer)[[50]](https://www.amibroker.com/products.html#:~:text=AmiBroker%20www,Download%20Buy). No ongoing fees unless one upgrades after 24 months. *Popularity:* Historically popular among serious system traders (particularly in futures and equities). Possibly tens of thousands of users worldwide, but exact figures aren’t public. It’s somewhat old-school, so newer retail entrants may be less familiar. *User trust:* Very high for accuracy and performance. AmiBroker’s backtester is considered **industry-grade** among retail quants – it’s used even at some prop firms for quick research. However, *ease-of-use trust* is low for non-programmers; AFL language is powerful but tricky for novices. *Sustainability:* The product is maintained by a small team (not subscription-based, but sustained by a steady trickle of new users and paid upgrades). It’s been around ~20+ years, indicating stability. *Limitations:* Steep learning curve, no integrated data (user must subscribe to data feeds or import their own CSV), Windows only (Mac/Linux users need emulators), and no social/sharing features (it’s offline). **Implication:** AmiBroker is both a benchmark to measure against (for accuracy and features) and a sign that many advanced retail traders *will pay* for robust portfolio backtesting. But its complexity leaves a wide open space for a more modern, user-friendly solution that offers similar portfolio analytics with a gentler learning curve.
* **MetaTrader 5 / MetaTrader 4:** (Considered indirect competition but worth noting due to large user base.) *Core focus:* Forex and CFD trading platform widely offered by brokers. *Features:* Algorithmic trading via Expert Advisors (EAs) and a strategy tester. MT5’s tester can handle multi-currency EAs (a single EA can trade multiple symbols in backtest), and offers walk-forward optimization. *Pricing:* Free to end-users (the cost is borne by brokers). *Popularity:* **Millions of users** (MT4/5 combined are among the most-used trading platforms globally). *User trust:* High for execution; mixed for backtesting (some retail users find MT4/5 strategy tester lacks depth – e.g., limited portfolio metrics, only one EA at a time, and data quality depends on broker). *Limitations:* Geared towards FX and CFDs; not designed for backtesting equity portfolios or non-FX multi-asset combinations. Many MT5 users actually use TradingView for charting and idea generation, then MT5 to automate – showing these can complement rather than directly compete. **Implication:** While MT5 offers multi-symbol backtesting in theory, it’s typically not used for equity portfolio simulations. The retail quant market segment we’re targeting often uses MT5 for execution but relies on other tools (like TradingView or AmiBroker) for robust system development.

**Indirect Competitors / Workarounds:**

* **BacktestBase (web tool):** This is a niche SaaS that specifically targets TradingView users’ pain point by letting them upload multiple TradingView strategy **CSV/XLSX exports** and then analyzing them together. *Core features:* Import strategy results, view combined performance, run Monte Carlo simulations, and even optimize weightings of different strategies in a portfolio[[57]](https://www.backtestbase.com/#:~:text=The%20Solution)[[58]](https://www.backtestbase.com/#:~:text=Transform%20individual%20trading%20strategies%20into,returns%20through%20systematic%20portfolio%20construction). Essentially, it creates a pseudo-portfolio from separate single-symbol backtests. *Pricing:* Freemium or subscription (the site is new; likely a monthly fee for heavy use). *Popularity:* It’s quite new – currently likely a few hundred early users. But the mere existence of such a tool (launched in 2023) confirms the demand. *Trust:* Yet to be established – as a third-party handling user data, it must earn credibility. If it provides clear value (time savings, insights), it could grow. *Sustainability:* Uncertain – it depends on TradingView continuing to not offer that functionality. If TradingView launched native portfolio backtesting, tools like BacktestBase could become redundant overnight. **Implication:** BacktestBase shows how **enterprising developers are building on top of TradingView’s shortcomings**, effectively siphoning off some value. A comprehensive solution from either TradingView or a partner could capture that value internally. Also, if partnering is an option, acquiring or integrating such a tool could jump-start a portfolio feature.
* **TradingView Browser Extensions:** Some users have created Chrome/Firefox extensions or scripts (like **“Autoview”** extension, originally made to automate TradingView alerts to exchanges) to partially automate backtesting tasks. For example, anecdotally, users have scripted sequences to automatically step through a watchlist and trigger TradingView’s strategy tester on each symbol, then scrape the results into a local file. These are **unsupported and against TradingView’s terms** (as discussed, any automation or scraping can lead to bans[[14]](https://www.tradingview.com/support/solutions/43000674726-why-is-my-account-banned-due-to-suspicious-activity/#:~:text=,TradingView%20content%20is%20expressly%20forbidden)[[15]](https://www.tradingview.com/support/solutions/43000674726-why-is-my-account-banned-due-to-suspicious-activity/#:~:text=Please%20note%20that%20the%20use,are%20for%20manual%20use%20only)). Thus, such tools remain underground. *Popularity:* Low – only the most tech-savvy attempt this, and it’s risky. *Sustainability:* Very low – TradingView actively combats such automation (IP bans, etc.), so these hacks are not reliable long-term. **Implication:** The presence of these hacks underscores the desperation of some users for a solution, but they are not viable in scale. It does indicate that an **official solution (or API) would be warmly welcomed**, since clearly some users are willing to MacGyver their own via extensions.
* **Other SaaS Platforms:** A few specialized services exist, like **Streak (Zerodha)** in India or **Tradetron**, which allow no-code strategy creation and backtesting *across multiple instruments*, mainly for Indian markets and crypto. These often have limits (e.g., Streak’s higher plans allow up to 1000 backtests/day and multi-stock strategies)[[33]](https://algotest.in/blog/streak-vs-tradetron/#:~:text=Streak%20vs,For%20more%20information%2C%20please). They target specific broker ecosystems. *Popularity:* Streak, for instance, has tens of thousands of users in India due to its integration with Zerodha. These tools validate that **no-code multi-asset backtesting is a sellable feature**. However, they are geographically or broker limited, and not direct global competitors to a TradingView-centered solution. They could become partners or acquisition targets in specific markets. **Implication:** A globally accessible portfolio backtester could outcompete these regional tools on reach, but should consider their features and pricing as a benchmark (many offer tiered pricing based on number of backtests, strategies, etc.).
* **Excel/Google Sheets Templates:** The “competition” here is DIY spreadsheets many traders use to combine results. There are community-shared Google Sheet templates for tracking portfolio trades or backtest outcomes[[59]](https://jatinkathiriya.medium.com/portfolio-backtesting-using-google-sheets-3a65cbce2acf#:~:text=Medium%20jatinkathiriya,returns%20to%20the%20S%26P500%20benchmark). While not commercial, they fill the gap for some. Obviously, they’re manual and limited. If a new solution can automate what people do in spreadsheets (e.g., summing P/L columns, calculating correlation between strategy equity curves), it will render these obsolete.

**User Trust and Preferences:** It’s instructive to note *why TradingView became dominant* (50M+ users)[[19]](https://www.brokersview.com/news/tadingview-usage-stats-for-2024-224255#:~:text=Key%20TradingView%20Stats) – largely due to ease of use, accessibility (web-based), and community/social features. Any competitor lacking those tends to remain niche. For example, AmiBroker (powerful but insular) never threatened TradingView’s user growth; instead, TradingView thrived on sharing of charts and scripts. **This suggests that a portfolio backtesting solution must ideally integrate with the TradingView workflow or community to gain trust and adoption.** An external tool risks being seen as another friction point unless it offers seamless data import or a familiar interface.

**Conclusion – Competitive Landscape:** No single competitor currently offers **all** of the following: multi-asset backtesting, ease-of-use, affordable pricing, and a strong community integration. TradingView has community and ease, QuantConnect/AmiBroker have multi-asset power, TrendSpider/Trade Ideas have some ease/automation but are limited in asset scope or expensive. This fragmented landscape means a well-positioned new entrant (or an extension of TradingView itself) could carve out a leadership position by addressing the unmet need for portfolio strategy testing. However, time is of the essence: if TradingView proper decides to implement this feature natively, it could quickly eliminate the need for third-party tools. Conversely, if they continue to delay, competitors may gradually encroach (for instance, TrendSpider could add portfolio testing in future versions, or QuantConnect could improve its UI).

Thus, a go-to-market strategy should consider **collaboration or partnership** with TradingView (see Section 8) as one avenue, or ensure the solution offers enough unique value (perhaps broker integration, cross-platform data, etc.) to stand on its own if TradingView remains separate.

**Sources:**

* NewTrading.io – TrendSpider user count and overview[[36]](https://www.newtrading.io/trendspider-review/#:~:text=While%20it%E2%80%99s%20very%20powerful%2C%20there,the%20heavy%20lifting%20for%20you)[[37]](https://www.newtrading.io/trendspider-review/#:~:text=TrendSpider%20is%20an%20AI,level%20models)
* QuantConnect site – user community claim[[22]](https://www.quantconnect.com/#:~:text=Platform,quant%20research%20community%20in)
* Trade-Ideas pricing page[[46][47]](https://www.trade-ideas.com/pricing/#:~:text=TI%20Premium%3A%20AI,signals%20to%20maximize%20profits)
* AmiBroker pricing page[[50]](https://www.amibroker.com/products.html#:~:text=AmiBroker%20www,Download%20Buy) and guide (portfolio features)[[48]](https://www.amibroker.com/guide/h_portfolio.html#:~:text=AmiBroker%27s%20portfolio%20backtester%20lets%20you,like%20you%20do%20in%20reality)
* BacktestBase site – highlights problem/solution for TV users[[11]](https://www.backtestbase.com/#:~:text=,settings%20and%20parameters%20worked%20best)[[12]](https://www.backtestbase.com/#:~:text=,TradingView%20backtests%20in%20one%20place)
* TradingView TOS – automation ban (impacting extensions)[[14]](https://www.tradingview.com/support/solutions/43000674726-why-is-my-account-banned-due-to-suspicious-activity/#:~:text=,TradingView%20content%20is%20expressly%20forbidden)[[15]](https://www.tradingview.com/support/solutions/43000674726-why-is-my-account-banned-due-to-suspicious-activity/#:~:text=Please%20note%20that%20the%20use,are%20for%20manual%20use%20only)

# 5. Monetization & Pricing Strategy

To monetize a portfolio-level backtesting solution, we should benchmark pricing against existing retail quant tools and align tiers with the **value delivered**. The goal is to hit a sweet spot where serious users are willing to pay, but pricing isn’t so high as to drive them to free DIY alternatives or competitors. Below is a recommended pricing structure, followed by revenue projections under different adoption scenarios:

**Benchmarking Competitor Pricing:**

* *TradingView:* Free (with ads) for basic features; paid plans at **$12.95, $24.95, $49.95** (often discounted) per month for higher tiers[[60]](https://www.tradingview.com/pricing/#:~:text=Plans%20for%20every%20level%20of,%C2%B7%204%20charts%20per%20tab). Many algotrading enthusiasts already pay ~$15–$50/mo for TradingView. This sets an anchor – our solution should ideally complement, not exceed, those costs significantly, unless it provides massive extra value.
* *TrendSpider:* $33–$79/mo (billed annually) for different tiers[[52]](https://www.newtrading.io/trendspider-review/#:~:text=Advanced%24349%242%2C796%E2%80%93%2020%20open%20workspaces%20%E2%80%93,30K%20depth)[[61]](https://www.newtrading.io/trendspider-review/#:~:text=Pricing%2489%2Fmonth%28Standard%29%2433,Free%20platform%20but%20requires%20a). They position themselves as a premium tool with advanced automation. Users comparing might accept similar pricing if portfolio backtesting is seen as similarly advanced.
* *QuantConnect:* effectively ~$20/mo for individual quant research use (the “Researcher” plan)[[42]](https://www.reddit.com/r/algotrading/comments/13la56p/what_happened_to_quantconnect/#:~:text=Furthermore%2C%20simple%20quality%20of%20life,much%20less%20active%20there%20now). They also have free options. This low price is possible because they upsell data and institutional products; for us, it suggests hobbyists expect something in the $0–$30 range unless enterprise-level.
* *Trade Ideas:* $89–$178/mo[[46]](https://www.trade-ideas.com/pricing/#:~:text=TI%20Premium%3A%20AI,signals%20to%20maximize%20profits). This is top-of-market pricing, justified by real-time AI and scanner profits for active traders. It’s a reference for the upper bound; only very engaged traders will pay ~$100+ monthly.
* *AmiBroker:* ~$300 one-time (which amortized over, say, 3 years, is ~$8/month). This is extremely cheap for its functionality, but users also spend on data feeds. It indicates that a *one-off license model* is an alternative approach if subscription isn’t viable, though modern SaaS tends toward recurring revenue.
* *Other SaaS (Streak etc.):* Typically $15–$50/mo range for mid-tier that includes multi-strategy backtesting capabilities. For example, one algo platform’s “Ultimate” plan at ~$32/mo allows 1000 backtests/day[[33]](https://algotest.in/blog/streak-vs-tradetron/#:~:text=Streak%20vs,For%20more%20information%2C%20please).

**Recommended Pricing Tiers:**

1. **Free (Community Tier):** Offer a limited free tier to drive adoption and allow users to test the waters. This could allow, say, portfolio backtests on up to 2 symbols at once (enough to demonstrate value via a simple pair-trade or long-short strategy) and perhaps a cap like 5 backtest runs per day. Also include basic CSV import for small datasets. *Purpose:* Attract the large base of curious TradingView users. Advertising or upsell notifications could be shown here. (Since TradingView itself is freemium, this aligns with user expectations that basic features have a free level.)
2. **Standard Tier (~$20/month):** This tier targets serious retail traders (the bulk of SAM). It would support, for example, portfolios of up to 10 symbols, unlimited backtest runs (within reasonable rate limits), and core performance metrics (CAGR, Sharpe, drawdown, etc.) at the portfolio level. It would also include the ability to import external data or TradingView strategy results via CSV for those 10 symbols. Visualization of combined equity curve and basic strategy comparison would be included. *Pricing rationale:* ~$20 is in line with TradingView Pro and less than TrendSpider’s base; it’s affordable to a hobbyist yet meaningful revenue. Users frustrated enough by manual work will pay the cost of a few coffees a month to save hours of time.
3. **Pro Tier (~$50/month):** A higher tier for power users, quant hobbyists, or semi-professionals (like those preparing for prop firm challenges or running signal services). This would allow larger portfolios (maybe 50 or unlimited symbols), advanced features like **strategy optimization** (ability to optimize weights or parameters across the portfolio), **walk-forward testing**, **Monte Carlo simulations** for risk, and possibly integration via API or webhooks (for those who want to automate retrieving TradingView data or connecting to brokers). Also include priority support. *Pricing rationale:* $50 is similar to TradingView Premium and TrendSpider’s top-tier. It’s expensive for casuals but fair for those actively using the tool to develop strategies that they might trade or sell. These users might also be running multiple backtests daily, consuming more resources, which the higher fee covers.
4. **Enterprise/Prop Tier (Custom or ~$100+/month per seat):** This wouldn’t be advertised to retail, but offered on inquiry. It could include white-label options, collaboration features (multiple team members sharing strategy results), API access to run batch backtests programmatically, and dedicated support/consulting. This tier targets small hedge funds, prop trading firms, or educator groups who might integrate the backtester into their workflow. *Pricing rationale:* Firms can afford higher prices; if the tool speeds up research or evaluation of traders, $100-$200/month per user is reasonable (still far cheaper than institutional analytics platforms).

**Value Proposition per Tier:** The pricing should correlate with **user value and usage volume**. For instance, a user on Standard who runs ~100 backtests a month and saves hours of manual work would likely find $20 a month a bargain (especially if they’re already paying similar for TradingView; it’s like an add-on). The Pro tier, at $50, might cater to someone running thousands of backtests or managing many strategies – possibly someone who monetizes their strategies (e.g., selling signals or running a fund), so $600/year is justified. The free tier ensures network effects and a funnel for conversion: users can try out with small portfolios and get hooked.

**Monetization of Data and Add-ons:** Another aspect – if our solution provides *data ingestion* (say price history for portfolio simulation), there could be additional charges for premium data. For example, including U.S. equity data might be free, but real-time or extended historical data or certain exchanges could be an add-on. Many SaaS (like QuantConnect) charge extra for alternative datasets. However, since TradingView users often already have data on TradingView, a more diplomatic approach is to allow them to plug in their TradingView account for data (if partnering) or import their own. We could monetize convenience by offering one-click import of all symbols’ data for a fee.

**CSV Ingestion and Integration:** The solution might have a feature to directly consume TradingView’s strategy output files. This is a *selling point*: *“Run your strategies on TradingView as usual, then import all results here to see combined performance”*. We might **not** charge extra for that integration (as it’s the core purpose), but we could use it as marketing. If any limits (like number of files) are present, higher tiers would lift those.

**Pricing Table (Illustrative):**

| Feature / Tier | Free Tier | Standard ($19.99/mo) | Pro ($49.99/mo) |
| --- | --- | --- | --- |
| Max Symbols per Portfolio Backtest | 2 symbols | 10 symbols | 50 (or unlimited) |
| Backtest Runs per Day | 5 | Unlimited (fair use) | Unlimited (priority queue) |
| Strategy Import (CSV/XLSX) | Yes (up to 2 files) | Yes (bulk import, 10 files at once) | Yes (unlimited batch import) |
| Portfolio Metrics & Charts | Basic (total P/L) | Full metrics (CAGR, DD, Sharpe) | Advanced + custom metrics |
| Strategy Optimization | No | No | Yes (optimize allocations or parameters) |
| Monte Carlo / Walk-Forward | No | No | Yes (robustness testing suite) |
| Support | Community only | Standard email support | Priority support + consult |

*(Pricing and limits are conceptual – to be adjusted based on real usage patterns and costs.)*

**Revenue Model:** Primarily subscription-based. Possibly offer a discount for annual prepaid (e.g., 2 months free). Also consider a **lifetime license** option at a higher price (some users prefer one-off payment; AmiBroker’s model shows demand for that). But subscriptions give stable recurring revenue and align with continuous updates.

**Upsell & Cross-sell:** The tiers naturally upsell users to move from Free to Standard as soon as they need larger portfolios. Within the app, after a user tests 2 symbols and tries to add a 3rd, a prompt can suggest upgrading. Cross-sell opportunities include: if we don’t provide native data, we could partner with data providers – e.g., “Add-on: official exchange data feed for $X/mo” for more accurate backtests (though if piggybacking TradingView data, that might not be needed). Another cross-sell: *education* – e.g., exclusive webinars or strategy templates for Pro users, which both adds value and justifies higher tiers.

**Cost Considerations:** Running backtests across multiple symbols can be computationally intensive (though if done client-side or in-browser, it offloads cost to user’s machine). If using cloud servers, we might need to throttle or queue tasks, which is partly why higher tiers get “priority”. The pricing needs to cover these infra costs. But since backtests are not real-time, even Standard tier users can likely be handled on shared servers with smart scheduling.

**Revenue Projections:** Let’s estimate potential revenue under three scenarios:

* **Conservative:** Low conversion and adoption. Suppose within 1 year we attract 10,000 total users (which is plausible given a subset of TradingView’s millions). Perhaps 7,000 use free tier actively, 2,500 on Standard, 500 on Pro. Annual revenue = (2,500 \* $19.99 + 500 \* $49.99) \* 12 ≈ **$720,000 per year**. This is conservative – just 3,000 paying users out of tens of thousands interested (conversion ~30% of user base which started free).
* **Base Case:** Moderate adoption in core community. Say 50,000 users sign up in year 1 (by tapping into TradingView communities, etc.). 30k remain free, 15k Standard, 5k Pro. That gives: (15,000*$19.99 + 5,000*$49.99)*12 ≈* *$4.5 million/year. In addition, maybe a couple of small prop firms license at enterprise tier for an extra ~$50k. So roughly* *$4.5–$5M annual*\*. This is achievable if we capture a small fraction of TradingView’s active strategy users.
* **Aggressive (Viral Growth):** If the tool goes viral in trading communities and perhaps TradingView themselves endorse or integrate it (without launching their own competitor immediately), we could envision 100,000+ users. For instance, 100k signups, of which 40k Standard, 10k Pro. That yields (40k*$19.99 + 10k*$49.99)*12 =* *$12 million/year*\*. Plus any enterprise deals. This scenario assumes strong word-of-mouth, low churn, and possibly international uptake (the tool supporting multi-language could open markets like China, Russia, etc., where TradingView is also popular).

**Attach Rates & Willingness to Pay:** Community feedback indicates many users would **pay for a solution that saves them time or extends TradingView**. For example, one user said regarding TradingView limits: *“if they have to pay for additional data/server costs we have no problem paying an additional fee for it.”*[[35]](https://www.reddit.com/r/TradingView/comments/goq969/feature_requests_comment_yours_here/#:~:text=Support%20replied%20to%20my%20message,no%20timeline%20could%20be%20given). This was in context of more data for backtesting, but it reveals a mindset: serious users are **willing to pay extra for enhanced backtesting capabilities**. Similarly, on third-party forums some have stated they’d subscribe to another service if it meant unlocking portfolio analysis. Given this, the projected conversion rates (a few percent of interested users converting to paid) seem reasonable or even cautious.

**Pricing Strategy Adjustments:** We should remain flexible. If adoption is slower, we might introduce a cheaper middle tier or promotional pricing. Conversely, if demand is overwhelming and servers are overloaded, it could justify raising prices for sustainability. One lever is also to differentiate on number of symbols or strategies: for instance, an **Ultra tier** at $100/mo for professional traders needing, say, 500 symbols or continuous scanning/backtesting (this could target small funds explicitly). This is analogous to how some platforms have an “Elite” tier for institutions.

**Long-term Monetization:** In addition to subscription fees, consider monetizing **aggregated strategy analytics** (anonymously). For example, if many users are testing similar things, insights could be drawn (like crowdsourced alpha). This enters a grey area ethically, so it must be opt-in or anonymized. But it could provide content (e.g., “X% of portfolios tested this week were tech-stock momentum strategies – are you following the herd?”) which can drive engagement or even be packaged as research for a fee. That would be a secondary revenue stream.

In summary, the monetization plan centers on **SaaS subscriptions with tiered value**, priced in line with what retail traders already pay for analogous tools (mid double-digits monthly for full features). This should maximize adoption while still capturing significant revenue given the large user base. The projections show a clear path to a multi-million dollar annual business if even a fraction of TradingView’s users convert – underscoring the attractiveness of this opportunity.

**Sources:**

* TrendSpider pricing (NewTrading.io)[[52]](https://www.newtrading.io/trendspider-review/#:~:text=Advanced%24349%242%2C796%E2%80%93%2020%20open%20workspaces%20%E2%80%93,30K%20depth)[[61]](https://www.newtrading.io/trendspider-review/#:~:text=Pricing%2489%2Fmonth%28Standard%29%2433,Free%20platform%20but%20requires%20a)
* Trade Ideas pricing page[[46][47]](https://www.trade-ideas.com/pricing/#:~:text=TI%20Premium%3A%20AI,signals%20to%20maximize%20profits)
* AmiBroker pricing (QuantifiedStrategies)[[50]](https://www.amibroker.com/products.html#:~:text=AmiBroker%20www,Download%20Buy)
* Reddit thread – users willing to pay for more backtesting resources[[35]](https://www.reddit.com/r/TradingView/comments/goq969/feature_requests_comment_yours_here/#:~:text=Support%20replied%20to%20my%20message,no%20timeline%20could%20be%20given)
* Streak/Tradetron plans (Algotest.in)[[33]](https://algotest.in/blog/streak-vs-tradetron/#:~:text=Streak%20vs,For%20more%20information%2C%20please)
* TradingView plans page[[60]](https://www.tradingview.com/pricing/#:~:text=Plans%20for%20every%20level%20of,%C2%B7%204%20charts%20per%20tab)

# 6. User Behavior & Willingness to Pay

Understanding **user behavior** around backtesting and their **willingness to pay** for convenience is crucial for tailoring the solution. Let’s break down typical patterns:

**Backtest Frequency & Usage:** Active algorithmic traders on TradingView tend to run **numerous backtests** when developing a strategy. It’s common to tweak parameters or try different symbols repeatedly. Anecdotal evidence: one TradingView user quipped they do *“20 backtests per day! until my eyes get used to the flow of the bars.”*[[17]](https://www.tradingview.com/u/mnspppppp/#:~:text=Trader%20mnspppppp%20%E2%80%94%20Trading%20Ideas,GBPUSD%3A%20GU%20BUY). And that was likely constrained by the manual process. In contrast, platforms like Cryptohopper or Streak explicitly limit backtests per day (e.g., 10/day for mid-tier) because users would otherwise run hundreds[[62]](https://docs.cryptohopper.com/docs/my-library/what-is-the-backtester/#:~:text=What%20is%20the%20Backtester%20,subscriptions%3A%2010%20backtests%20per%20day). This implies *demand for high-volume backtesting*: serious users might run **dozens if not hundreds of backtest iterations per week** when optimizing a strategy.

On average, a hobbyist might run a few dozen backtests in initial development and then occasional re-testing, whereas a quant enthusiast with multiple strategies might run hundreds per month (especially if testing a strategy across many symbols one by one). Currently, TradingView’s friction (one-by-one testing) likely *suppresses* usage – people give up or cut testing short due to tedium. A portfolio backtester that automates multi-symbol runs would actually **increase** how many backtests users perform (because it’s easier and faster). We should expect engaged users to create whole *test suites* (e.g., run strategy X on 50 symbols in one go).

**User Segments & Behavior:**

* *Retail Hobbyists:* These are individuals trading with personal accounts, interested in automation but not full-time. They often have a day job and do strategy development on evenings/weekends. They value *time savings*. Many in this group currently use spreadsheets to track performance or do very basic manual backtests. Their willingness to pay is moderate – they might pay for a month or two while working on a strategy, then pause. We should accommodate that with easy cancellation/resume options.
* *Quant Enthusiasts/Developers:* These overlap with hobbyists but are more technical (perhaps software developers by profession). They might otherwise use free Python tools, but they use TradingView for convenience. They run lots of tests, are vocal in communities about limitations, and are willing to pay for better tools (as evidenced by them paying for TradingView Premium or for data feeds). They also appreciate nuance – e.g., they notice if backtests handle commissions, slippage, etc. This group will stress-test our tool and contribute feature ideas. They are quite willing to pay if the tool is good – many have spent money on books, courses, or other software (e.g., some are likely among the ~275k QuantConnect users[[22]](https://www.quantconnect.com/#:~:text=Platform,quant%20research%20community%20in), which has paying members).
* *Prop Firm Traders/Algo Trainees:* With the rise of prop trading challenges (FTMO, etc.), many traders are trying to use algorithms to pass. These folks often test multi-asset strategies to smooth equity curves and meet strict drawdown rules. They are **highly motivated** to have accurate portfolio backtests because their funding is on the line. They often run numerous simulations to ensure their approach can survive worst-case scenarios. This group will pay for tools that give them an edge or save them from failure. They might not blink at a $50/mo fee if it helps them secure a funded account of $50k+.
* *Content Creators & Educators:* Some TradingView users create strategies to share or sell (e.g., Pine Script authors selling strategy scripts or signal services). For them, having portfolio backtest results can be a marketing differentiator (“My strategy portfolio yielded X% with Y% drawdown last 5 years”). They might use our tool to produce those analytics. They’re willing to invest in tools that enhance their product, and they might even display our tool’s output in their content. Possibly, we could have an educational discount or affiliate program with them.

**Willingness to Pay – Evidence:**

* TradingView’s own Premium subscriber count (likely hundreds of thousands) shows **users pay for better backtesting capabilities indirectly** (Premium increases data and indicator limits, which are largely to enable more thorough strategy tests)[[34]](https://www.reddit.com/r/TradingView/comments/goq969/feature_requests_comment_yours_here/#:~:text=%E2%80%A2%20%205y%20ago). Many explicitly upgraded because 10k bars of data on lower plans were “not enough for backtesting on lower timeframes”[[63]](https://www.reddit.com/r/TradingView/comments/goq969/feature_requests_comment_yours_here/#:~:text=%2B1%20we%20need%20this%21%2010,give%20a%20timeframe%20on%20this) – implying they paid ~$50/mo just to get 20k bars for the strategy tester. This is a strong indicator they would pay similarly to get multi-symbol functionality. As one user said in frustration at the bar limit, *“we have no problem in paying an additional fee for it… [portfolio testing] is a must-fix for the platform”*[[35]](https://www.reddit.com/r/TradingView/comments/goq969/feature_requests_comment_yours_here/#:~:text=Support%20replied%20to%20my%20message,no%20timeline%20could%20be%20given)[[16]](https://www.reddit.com/r/TradingView/comments/16jhyvm/run_a_single_strategy_against_multiple_symbols_in/#:~:text=have%20the%20ability%20for%20portfolio,strategy%20on%20multiple%20symbols%20simultaneously).
* Reddit threads show users asking for any workaround even if it costs money: *“If anyone has found a workaround [for exporting multiple symbol results] I’d love to hear it… such a pain”*[[64]](https://www.reddit.com/r/TradingView/comments/s9uplg/need_the_ability_to_export_backtesting_results/#:~:text=PrimordialRocks). The absence of a mention of cost suggests they’d adopt a solution if one existed, cost aside.
* When discussing potential features, some users say “I’d pay double my subscription if TradingView added this”. While anecdotal, such sentiments pop up in forums for highly desired features.
* Competing services like Trade Ideas and TrendSpider *do have paying customers at premium price points*. This proves a segment of retail traders *will* pay $50–$100+ monthly for tools that improve their trading process. Our target is a slightly broader base at a somewhat lower price, but the existence of those pricier competitors validates our mid-tier pricing approach as feasible for serious users.

**Pain vs. Price Trade-off:** The current manual method is essentially *“free”* in dollar cost but very expensive in time/effort. We can estimate the value of that: If a trader manually tests 10 symbols, each taking ~5 minutes to load, run, record outcomes, that’s ~50 minutes per iteration. If our tool does it in 1 minute, we save ~49 minutes. What is that worth? If a user values their time even at $10/hour, that one run saved ~$8 of effort. Over a month, easily dozens of hours can be saved – effectively hundreds of dollars of value for an active strategy developer. Therefore, asking $20–$50 per month is extremely reasonable from a value perspective. The key is to communicate that value (perhaps with a marketing message like “save 10+ hours of grunt work per month – let our Portfolio Tester handle the heavy lifting”).

**Effort & Frustration Measurements:** Many TradingView users have voiced *qualitative* frustration. For example, the user who said *“Makes no sense. Makes me feel like I’m betting on the wrong horse.”* in reference to limited backtesting[[53]](https://www.reddit.com/r/TradingView/comments/goq969/feature_requests_comment_yours_here/#:~:text=I%27ve%20been%20wanting%20this%20for,betting%20on%20the%20wrong%20horse) – that signals not just frustration but *fear* of not having proper tools (hence risking their strategy’s success). People pay to alleviate fear (fear of bad trades, etc.). Our solution, by giving more realistic portfolio results, reduces the risk of nasty surprises when trading live (e.g., seeing how strategies interact could prevent over-leveraging). This safety/assurance aspect boosts willingness to pay among rational traders.

**Pricing Sensitivity:** Based on competitor pricing and user budgets: - At ~$20/mo (Standard tier), we expect little resistance from our target users – that’s similar to one retail brokerage commission or a data feed cost. Many pay more for market data. - At ~$50/mo (Pro tier), the user likely expects a professional-grade tool; these will be the more invested traders or small trading businesses. There will be fewer of them, but they are willing because they probably already spend on multiple services (e.g., some pay for both TradingView Premium and another service like Option analytics or a Bloomberg subscription, etc.). - If we tried to price at $100/mo for individuals, we’d enter a very selective domain (like Trade Ideas did). We risk scaring away much of the retail base. So we keep that only for enterprise where justification is different.

**Retention & Churn:** We should anticipate that some users might subscribe for a month or two intensively while developing a strategy, then cancel until they need it again. This is common in software usage patterns when projects are cyclical. We can counteract churn by adding features that encourage continuous use (like maybe portfolio **paper trading** or monitoring – turning backtests into live tracking, etc., which keeps them subscribed, but that’s future scope). However, even intermittent usage can be fine – if someone pays $50 for 2 months to develop a strategy that they then trade for real money, that’s a good deal for them and decent revenue for us.

**Survey/Data if Available:** If we had access to any surveys of TradingView users (for example, if TradingView ran polls on what feature they’d want or pay for), we could quantify interest. *Hypothetically*, if even 10% of TradingView Premium users (who already pay ~$600/yr) say they’d pay an extra, say, $20/mo for portfolio backtesting, that’s significant. Premium has say 100k users (just an estimate); 10k of those paying $20 is $200k/mo.

**Community Feedback Loops:** The plan should involve continuing to gather feedback on pricing tolerance. Perhaps start with an introductory beta price (e.g., first adopters get Standard at $14.99 locked in). If we see overwhelming adoption, we might gradually raise to the target $19.99. If adoption is slower, maybe keep it at intro price longer. TradingView itself often runs promotions (Black Friday 50% off etc.), which influences user mindset – they often wait for deals. We might mimic that: annual plans with discount, or seasonal sales to spur sign-ups, which aligns with known user behavior.

**Conclusion (Behavior & Willingness):** The core takeaways are: - **Active backtesters are heavy users** – running many tests and thus highly value efficiency improvements. - **They are willing to pay** for enhanced capability – evidence being their existing spend on TradingView upgrades and third-party tools, plus explicit statements of willingness. - **Convenience and accuracy** (knowing how multiple strategies perform together) directly affect their real trading performance and peace of mind, making this a *must-have, not just nice-to-have* for many serious traders. That drives willingness to pay more than, say, cosmetic features would. - **Frugality vs. ROI:** Traders generally frame expenses in terms of trading gains/losses. If our tool can prevent even one costly mistake (like discovering two strategies together would have blown up the account in 2008, thus prompting the user to adjust), it could save them thousands. Paying a few hundred a year for that insight is a no-brainer.

All these factors indicate that, while free users will be plentiful, **a significant number of users will convert to paid plans given the pain point alleviated**, and they will find the price well worth it for the time saved and confidence gained.

**Sources:**

* Reddit (r/TradingView) – user backtest frequency quote[[17]](https://www.tradingview.com/u/mnspppppp/#:~:text=Trader%20mnspppppp%20%E2%80%94%20Trading%20Ideas,GBPUSD%3A%20GU%20BUY)
* TradingView community thread – users upgrading for more data (willingness to pay)[[34]](https://www.reddit.com/r/TradingView/comments/goq969/feature_requests_comment_yours_here/#:~:text=%E2%80%A2%20%205y%20ago)[[35]](https://www.reddit.com/r/TradingView/comments/goq969/feature_requests_comment_yours_here/#:~:text=Support%20replied%20to%20my%20message,no%20timeline%20could%20be%20given)
* User quotes – frustration implying need (betting on wrong horse)[[53]](https://www.reddit.com/r/TradingView/comments/goq969/feature_requests_comment_yours_here/#:~:text=I%27ve%20been%20wanting%20this%20for,betting%20on%20the%20wrong%20horse)
* Cryptohopper docs – limits on backtests per day (implying typical usage)[[62]](https://docs.cryptohopper.com/docs/my-library/what-is-the-backtester/#:~:text=What%20is%20the%20Backtester%20,subscriptions%3A%2010%20backtests%20per%20day)
* QuantConnect community size (paying users inferred)[[22]](https://www.quantconnect.com/#:~:text=Platform,quant%20research%20community%20in)

# 7. Risks & Compliance

Developing a solution that interacts with TradingView or replicates some of its functionality comes with several **risks and compliance considerations**. We need to navigate technical, legal, and business risks carefully:

**TradingView Terms of Service (ToS) Compliance:** TradingView has explicit rules against unauthorized use of their platform and data. As cited earlier, *“users are strictly prohibited from employing any automated data collection… scripts, APIs, screen scraping… All our features are for manual use only.”*[[14]](https://www.tradingview.com/support/solutions/43000674726-why-is-my-account-banned-due-to-suspicious-activity/#:~:text=,TradingView%20content%20is%20expressly%20forbidden)[[15]](https://www.tradingview.com/support/solutions/43000674726-why-is-my-account-banned-due-to-suspicious-activity/#:~:text=Please%20note%20that%20the%20use,are%20for%20manual%20use%20only). This means any solution that **scrapes TradingView’s website or automates actions on behalf of users (like cycling through symbols and reading strategy results)** violates ToS. The risk is user accounts could be banned if they use such a tool, and TradingView could take legal or technical action against the tool (IP blocking, cease-and-desist).

* **Mitigation:** Our solution should avoid direct scraping or automation of TradingView’s interface. Instead, rely on *user-provided data exports* (which TradingView permits – users can manually export strategy tester results to CSV). OCR of charts or any hidden API calls are risky. We likely should build a **cooperative or independent approach**: either partner with TradingView for official API access or have users operate entirely outside TradingView’s live site (e.g., by pulling data from elsewhere or requiring them to download CSVs).
* If partnership is possible, TradingView might have a developer API or be willing to allow certain calls (they have a *Charting Library* and a *Broker API*, but not a public backtesting API). Without partnership, we must operate under the assumption that *any use of TradingView’s logged-in features by a bot = violation*. Therefore, the safest path is to **treat TradingView as a separate entity**: the user can use our tool with data they gather (from CSVs or other data vendors) and perhaps import their Pine script logic manually or via translation. Essentially, not lean on TradingView to do anything automated.

**Data Licensing:** If our tool provides historical price data for backtesting, we must ensure we have the rights to distribute that data. TradingView has licenses with many exchanges; we cannot legally pull data from TradingView without permission. We might use free data sources (Yahoo Finance, Alpha Vantage, Crypto APIs) for basic functionality, but for high quality or extensive history, licensing agreements may be needed. This could be costly and complicated (exchanges often charge fees for redistribution). Alternatively, the tool could *require users to provide their own data* (via CSV upload from whatever source they have). That shifts compliance to the user (if they have legal access to data, they can use it). For a smoother UX, though, having at least some built-in data is ideal (perhaps for popular symbols, using free or open data).

* **Mitigation:** Start with instruments that have free historical data (e.g., crypto data from Binance, which is often freely usable; or use data from Wiki EOD for equities which is free but maybe delayed). For any premium data (like real-time or intraday stocks), consider an integration where the user uses their own API key (for example, allow user to connect their Alpaca or Polygon.io data feed). This way, our service isn’t directly redistributing licensed data without permission – it’s facilitating the user’s use of their own data subscriptions.
* Long-term, if scaling, negotiating a data feed deal (like Kinetick or other provider for EOD equities) might be worthwhile so that paying users get convenient data legally.

**Dependency on TradingView’s Goodwill:** If our product is successful, it will inevitably catch TradingView’s attention. One risk is *TradingView launching native portfolio backtesting*, rendering our solution less necessary or directly competing with it. TradingView has far more resources and an existing user base integrated; they could possibly roll it out (though the fact they haven’t suggests technical or priority hurdles). But if they decide to, they could undercut us (maybe making it a Premium-only feature and instantly capturing the market).

* This is a **business risk**. We can mitigate by (a) moving quickly to build a user base and maybe even be acquired or officially integrated by TradingView rather than competing, or (b) ensure our tool has some unique angle or cross-platform ability TradingView might not offer. For instance, our tool could support **multi-platform data** – not just TradingView but also MetaTrader strategy imports, or custom strategy coding. Building a **moat** could involve creating a community (like strategy sharing focused on portfolios) that is separate from TradingView’s community.
* Another approach: maintain a friendly stance with TradingView, possibly approaching them for partnership once we have a beta. It’s a fine line: if we appear to violate ToS or siphon users away, they could choose to block us (for example, if our site uses their name/logo without permission – we must avoid that too; branding should be distinct even if targeting their users).

**Technical Risks (Rate Limits):** If we did attempt any automated retrieval from TradingView (like logging in to fetch data), there are *rate limits and anti-bot measures*. TradingView might temporarily ban IPs or accounts that make too many requests. This could break our functionality unexpectedly. Given our earlier conclusion to avoid scraping, we likely won’t hit this. But even user manual exports could be rate-limited (e.g., if a user tries to export 100 symbols quickly, TradingView might cap how fast CSVs can be downloaded).

* Mitigation: If instructing users to do exports, we might need to instruct them to space it out or provide a utility that automates it at a safe pace (this again veers into automation; risky if not approved). Alternatively, fetch data from an API like Yahoo as a workaround to avoid hitting TradingView at all.

**User Account Security:** If any integration required users to input their TradingView credentials into our app (to pull data), that’s a huge security no-no (and against ToS likely). We should avoid ever asking for credentials. If future partnership yields an OAuth, that’s different, but currently TradingView has no such API for third-party login. So we won’t go that route.

**Intellectual Property:** Pine Script code – if users want to backtest their Pine scripts on our platform, how to do that without violating IP? Pine Script is proprietary to TradingView, but the scripts users write are their own intellectual property. If we tried to parse Pine code to run it, we might run into issues. There’s an emerging question: is reimplementing Pine Script execution legal? Possibly, if it’s all user-owned code. But TradingView might argue that the Pine Script language is their IP (though languages are usually not copyrightable, the specific functions library might be). It’s a grey area.

* Mitigation: We might avoid directly executing Pine scripts; instead, provide templates to define strategies in our system (like a no-code interface or a Python-based simple language). Or, if we attempt to support Pine, we could open-source a Pine Script interpreter to avoid claims of copying their closed-source code. There is precedence: an open-source project called **PineCodex** attempted a Pine compiler – we could check its status. But legally, if we reverse-engineer Pine without using any of their code, likely fine. Still, this is complex and perhaps out-of-scope initially.

**Regulatory Compliance (Financial Advice vs. Tool):** We should ensure our marketing doesn’t stray into “financial advice” territory. We are providing a tool, not telling users what to trade. Also, any performance metrics we show, we should disclaim like “hypothetical performance, not indicative of future results”. If we attract global users, we might have to consider regulations in various countries about financial software. Generally, a backtesting software is not regulated as an advisory service, but disclaimers are prudent.

**GDPR/Data Protection:** As a SaaS, we’ll handle user data (their uploaded backtest results, etc.). Compliance with privacy laws (GDPR for EU, etc.) is needed – e.g., allow user to delete data, protect personal info, etc. This is standard SaaS procedure but must be on our radar.

**Competitive Response:** A risk is also that one of the existing competitors (like TrendSpider or QuantConnect) sees our entry and tries to quickly implement similar multi-portfolio features or undercut pricing. That’s part of competitive risk but not compliance – more a strategic risk. Our moat discussed in Section 8 will address this.

**Summary of Major Risks & Solutions:**

* **ToS Violation Risk:** Avoid scraping or automating TradingView; either partner for data or rely on user-imported data. Provide value on top of TradingView rather than hacking it. *Plan:* Build with zero direct TradingView dependency (aside from user exporting files).
* **Account Ban Risk:** Warn users not to use any method that could ban their account (no sharing login, no running unofficial scripts concurrently). If user must do something within TV (like export 100 files), advise them to respect usage limits.
* **Legal Risk from TradingView:** If our branding or marketing explicitly uses “TradingView” name or trademarks without permission, we could get legal notices. *Plan:* Choose a distinct name for the product. In marketing, phrase it as “portfolio backtesting for TradingView users” carefully (maybe “Compatible with TradingView” rather than using logo). Possibly get informal OK from them by reaching out early, or keep a low profile by focusing on the functionality rather than piggybacking on their brand too much.
* **Data Compliance:** Use only legally permissible data sources; allow user-provided data to bypass licensing issues when possible. If partnering, maybe TradingView could provide data via their API (they have a *Widget API* that gets data for display; maybe not suitable for bulk though).
* **Tech Infrastructure Risk:** Overuse or misuse might cause downtime. We should implement fair use policies and possibly throttle extremely heavy usage (with communication to user if needed).

In essence, **the safest approach is an independent platform that complements TradingView without infringing its rules**. By doing so, we avoid antagonizing them and reduce compliance headaches. Long-term, establishing a formal relationship (like being an official third-party add-on) would be ideal, but we should not rely on that from the start.

**Sources:**

* TradingView ToS excerpt (no automation)[[14]](https://www.tradingview.com/support/solutions/43000674726-why-is-my-account-banned-due-to-suspicious-activity/#:~:text=,TradingView%20content%20is%20expressly%20forbidden)[[15]](https://www.tradingview.com/support/solutions/43000674726-why-is-my-account-banned-due-to-suspicious-activity/#:~:text=Please%20note%20that%20the%20use,are%20for%20manual%20use%20only)
* Reddit (r/TradingView) – user discussion hints at no official API (implied by how they discuss lack of features).
* Common knowledge from developer forums on Pine Script IP (no direct source snippet, but general software IP law applies).

# 8. Recommendations & Go-to-Market Strategy

Finally, we synthesize recommendations for how to **build, position, and launch** this solution, ensuring we maximize adoption while mitigating the risks discussed:

**Build vs. Partner vs. Both:**

* **Build Stand-Alone Backtester:** Given TradingView’s current lack of an open API for strategy backtesting, the pragmatic route is to **build an independent web application** for multi-symbol backtesting. This stand-alone tool should be able to ingest data (from user files or integrated feeds), simulate strategies across portfolios, and present analytics. The core engine could be built in Python (leveraging libraries like Pandas for backtest logic) or another high-performance environment. We should ensure flexibility to handle various asset types (stocks, forex, crypto – as data permits).
* **Partnering with TradingView:** While TradingView doesn’t officially support third-party plugins in the chart interface, a partnership could take other forms. For example:
* Collaborate on data access: TradingView might provide an endpoint for retrieving a user’s saved backtest results or Pine script strategies (this is speculative; currently no such API, but as a partner, they might consider).
* Co-branding: We could become a “TradingView Approved” service listed on their site (TradingView has a blog and often features community-developed tools).

The recommendation is to **initially build independently but keep doors open for partnership**. Approach TradingView once we have a working beta and user interest, to discuss if they’d be open to a formal integration or even acquisition. If TradingView shows interest in building this themselves, perhaps propose we pilot it externally and then integrate.

* **Avoiding Direct Competition (if possible):** If TradingView were working on their own portfolio backtester, partnering might be off the table. In that scenario, be prepared to differentiate (discussed below under moat).

**Unique Value & Moat:** To guard against eventual TradingView competition or others copying us, we should develop features that are not trivial to replicate: - **Cross-Platform Compatibility:** Allow imports not just from TradingView but also from other sources. For instance, if a user has backtest trades from MetaTrader or a CSV from another platform, let them combine those too. This broadens our appeal beyond just TradingView users. It positions us as *the go-to portfolio analyzer* for retail algorithms generally. - **Multi-Strategy Combination:** Perhaps allow users to combine different strategies (not just one strategy across symbols, but different strategies on different symbols) into one portfolio. TradingView’s eventual offering might only allow one Pine script applied to multiple symbols; we could allow a richer mix (e.g., 3 strategies each on different asset classes combined). This is more akin to portfolio construction, which could attract advanced users. - **Optimization & AI:** Incorporating optimization tools (e.g., find optimal capital allocation among strategies) or AI-driven suggestions (like highlight if two assets in the portfolio are highly correlated) could set us apart. If we develop intellectual property (like a proprietary optimization algorithm or a repository of user-contributed strategy snippets), that forms a defensible asset. - **Community & Social:** Consider adding community features early: e.g., users can share their portfolio backtest results or templates with others (with privacy controls). TradingView has a social network around single charts; we could create a niche community around *portfolio strategies*. This user base and content can become a moat – even if TradingView clones the feature, the community might remain on our platform if it’s well-established (similar to how people still use independent forums or tools even if a bigger platform has overlap).

**Go-to-Market (GTM) Channels:**

* **Quant Twitter and FinTwit Influencers:** There’s a vibrant community on Twitter of algorithmic traders and developers (e.g., @AlpacaHQ, @Quantopian (when active), many independent quants sharing backtest charts). Identify influencers who often talk about backtesting or Pine Script. For instance, the **PineCoders** community (affiliated with TradingView) might be cautious, but individual Pine experts might appreciate a tool that complements their work. We could offer them beta access or ask for feedback; if they like it, they’ll likely tweet about it. An example strategy: create some compelling content (like a case study: “Portfolio of 5 popular TradingView strategies – here’s how it performed combined”), share it on Twitter with our tool’s link. If interesting, it gets retweeted.
* **Reddit & Discord Communities:** Subreddits like r/TradingView (95k members) and r/algotrading (over 220k members)[[65]](https://www.reddit.com/r/algotrading/comments/13la56p/what_happened_to_quantconnect/#:~:text=GreenBlueberries) are prime audiences. We must tread lightly to avoid pure self-promo (especially on r/TradingView which the TradingView team monitors). A possible approach: make an informational post like “Guide: How to backtest a multi-symbol portfolio (finally!)” where we demonstrate the process using our tool. Emphasize solving the pain rather than selling a product. On algorithmic trading Discord servers (there are several with thousands of members), we can similarly share knowledge and mention our tool as part of the solution. Offering an extended free trial or discount to these community members could spur word-of-mouth.
* **YouTube Educators:** Many YouTube channels focus on TradingView tutorials, Pine Script coding, or strategy development. For example, channels by **TradingView community script authors** or general trading educators. These folks are often looking for new content. We could reach out offering our tool for them to review or use in a video (“How to portfolio backtest your TradingView strategy”). If our tool has a free tier, they can demonstrate it at no cost. Some might ask for affiliate commissions or sponsorship – we could consider an affiliate program (e.g., a cut for any referrals who upgrade to paid).
* **Broker Partnerships:** Retail brokers that support algorithmic traders (like Interactive Brokers for multi-asset, or newer API-based brokers like Alpaca, FTX (for crypto), etc.) might want their clients to have backtesting tools (clients who backtest responsibly may trade more confidently/capably). We could partner by providing content or tool access to those broker’s user bases. For instance, write a guest blog on Alpaca’s site about portfolio backtesting (since Alpaca provides data and trading but no backtester). In return, mention our app. Even offering a custom plan for broker clients (maybe a discount if they connect their brokerage, and in exchange, we could get listed in their marketplace or get data access).
* **SEO/Content Marketing:** Capture search traffic for terms like “TradingView portfolio backtest” – currently, search results show reddit threads of people asking for this[[66]](https://www.reddit.com/r/TradingView/comments/1guzc2j/portfolio_backtesting_feature/#:~:text=One%20feature%20that%20I%20am,instruments%20in%20a%20single%20portfolio), YouTube hack videos[[67]](https://www.youtube.com/watch?v=-WwzYGmC6po#:~:text=TradingView%20PORTFOLIO%20Backtesting%20Tool%20for,profitable%20trading%20experience%2C%20I), etc. We should create a blog on our site addressing these queries (e.g., “Portfolio Backtesting on TradingView: The Ultimate Guide (2025)”). That content can rank and attract organic users looking for a solution. Given the gap, this could drive steady traffic.
* **Email Newsletters & Forums:** Sponsor or get featured in popular trading newsletters (e.g., AlphaTrends, or Quantifiable Edges, etc.) focusing on system trading. Or on sites like Quantocracy (which curates quant articles) – an article about portfolio backtesting could make it there.

**Launch Strategy:**

* Start with a **closed beta** with a handful of known community members (could be from Reddit or personal contacts in trading communities). Get feedback, case studies.
* Move to **open beta** – perhaps product launch on platforms like Product Hunt (tech audience, but many devs there trade crypto or stocks and would find it cool), or an announcement on relevant subreddits (with permission of mods if possible).
* Offer a **time-limited free upgrade** for early adopters (“Founding Members get Pro tier free for 3 months” or similar). This incentivizes people to try it now, not later, and helps us gather usage data.
* **Leverage TradingView’s own channels carefully:** Possibly post in TradingView’s “Feature Request” forum thread that we built something addressing the portfolio testing request (if mods allow; we might phrase it as “While waiting for official support, here’s a workaround tool”). It could get removed if seen as advertising, so we approach diplomatically or via a user testimonial style (maybe a beta user posting “I tried XYZ tool and it solved it for me”).

**Building a Moat in Anticipation of TradingView’s Move:**

As mentioned, if TradingView decides to release a similar feature, how do we ensure our survival or relevance? Some strategies:

* **First-mover advantage with superior analytics:** By the time they catch up, ensure we have advanced analytics (Monte Carlo, optimizations, multi-strategy, etc.) that a first version of TradingView’s feature might lack. Then we position as the “pro” tool. TradingView might offer a basic portfolio test; power users still come to us for depth.
* **Cross-platform & Broker integration:** If we can execute trades or connect to brokers, we become not just backtest but execution management (e.g., one could paper trade the portfolio or even live trade via our tool sending orders to multiple brokers). TradingView doesn’t currently support multi-asset trading from one strategy either (they require one chart per symbol to trade). If we fill that gap – i.e., “backtest here and deploy to broker across multiple assets” – that’s a moat. This essentially steps toward being an *algo trading platform* (like QuantConnect but with a simpler UI).
* **Community & Data network effects:** If users have uploaded thousands of strategies or results, we could offer aggregated insights that TradingView couldn’t immediately replicate. E.g., a “Strategy Screener” across uploaded strategies to find uncorrelated strategy combinations. Or simply the accumulated user feedback and trust – being known as the portfolio backtesting experts.
* **Potential Acquisition:** A candid reality – if our solution gains traction, TradingView might approach to acquire or hire the team to integrate it. That could be a win-win exit strategy. Being acquired by TradingView would validate the need and secure the feature inside the platform for users (which was the end goal to help them). So, we shouldn’t antagonize TradingView; rather, build something valuable and be open to collaboration or acquisition discussions.

**Positioning & Messaging:** Emphasize that we **“empower TradingView users to take their strategies to the next level”**. We are not replacing TradingView – we complement it. That friendly positioning reduces user hesitance (they don’t feel they have to leave TV; they can use both) and is more palatable to TV itself. Use language like “Extend your TradingView strategy testing with portfolio analysis” in marketing materials (ensuring not to misuse trademark). Perhaps write case studies: e.g., “We took a popular TradingView strategy that looked great on AAPL alone, and tested it as a portfolio on 10 tech stocks – see the difference.” These tangible examples will resonate.

**Customer Support & Onboarding:** Early on, closely support initial users – possibly via a Discord server or Slack community. These early adopters, if thrilled, will become evangelists (they’ll answer questions for new users, share in forums, etc.). Good support is also a differentiator from some competitors (QuantConnect forums being unanswered was a user complaint[[54]](https://www.reddit.com/r/algotrading/comments/13la56p/what_happened_to_quantconnect/#:~:text=line%20of%20your%20code%20,much%20less%20active%20there%20now) – we can shine by being responsive, at least initially).

**Metrics for Success:** In the go-to-market phase, track sign-ups, conversion to paid, user engagement (backtests run per user), and sources of traffic to double-down on what's working (e.g., if Reddit drives a lot of engaged users, invest more there). Also gather testimonials to use in marketing (“X from London: ‘Finally, I can test portfolios – this saved me so much time!’”).

**Conclusion & Timeline:** Launch a beta within ~3-4 months, gather feedback 1-2 months, official launch with marketing push thereafter. Aim to capture a few thousand users in the first 6 months (perhaps optimistic but feasible with strong community outreach). Continually release feature improvements (perhaps public roadmap) to keep users engaged and show momentum – this is important if lurking competitors or TradingView are watching; outpacing them could discourage them from competing directly.

By following this strategy – building a strong independent tool while maintaining a cooperative stance – we maximize our impact and either carve out a sustainable business or position ourselves as an attractive partner/asset for TradingView or others in the retail trading space.

**Sources:**

* Reddit r/TradingView and r/algotrading subscriber counts (for audience targeting)[[65]](https://www.reddit.com/r/algotrading/comments/13la56p/what_happened_to_quantconnect/#:~:text=GreenBlueberries)
* Twitter/FinTwit anecdotal evidence of interest (no specific cite, based on common FinTwit discourse on backtesting).
* Quantconnect user complaint about support[[54]](https://www.reddit.com/r/algotrading/comments/13la56p/what_happened_to_quantconnect/#:~:text=line%20of%20your%20code%20,much%20less%20active%20there%20now) (highlighting opportunity for better community support).

[[1]](https://www.reddit.com/r/TradingView/comments/16jhyvm/run_a_single_strategy_against_multiple_symbols_in/#:~:text=this%20is%20a%20need%20of,strategy%20on%20multiple%20symbols%20simultaneously) [[8]](https://www.reddit.com/r/TradingView/comments/16jhyvm/run_a_single_strategy_against_multiple_symbols_in/#:~:text=%E2%80%A2%20%209mo%20ago) [[16]](https://www.reddit.com/r/TradingView/comments/16jhyvm/run_a_single_strategy_against_multiple_symbols_in/#:~:text=have%20the%20ability%20for%20portfolio,strategy%20on%20multiple%20symbols%20simultaneously) Run a single Strategy against multiple Symbols in a Watchlist style window : r/TradingView

<https://www.reddit.com/r/TradingView/comments/16jhyvm/run_a_single_strategy_against_multiple_symbols_in/>

[[2]](https://www.tradingview.com/pine-script-docs/faq/strategies/#:~:text=How%20can%20I%20backtest%20multiple,symbols) [[3]](https://www.tradingview.com/pine-script-docs/faq/strategies/#:~:text=,a%20strategy%20on%20different%20symbols) Strategies

<https://www.tradingview.com/pine-script-docs/faq/strategies/>

[[4]](https://www.reddit.com/r/TradingView/comments/14wwq81/portfolio_backtest/#:~:text=portfolio%20backtest%20%3A%20r%2FTradingView%20,already%20supported%20by%20other%20platforms) portfolio backtest : r/TradingView - Reddit

<https://www.reddit.com/r/TradingView/comments/14wwq81/portfolio_backtest/>

[[5]](https://www.reddit.com/r/TradingView/comments/s9uplg/need_the_ability_to_export_backtesting_results/#:~:text=%E2%80%A2%20%202y%20ago) [[7]](https://www.reddit.com/r/TradingView/comments/s9uplg/need_the_ability_to_export_backtesting_results/#:~:text=PrimordialRocks) [[9]](https://www.reddit.com/r/TradingView/comments/s9uplg/need_the_ability_to_export_backtesting_results/#:~:text=Backtesting%20strategies%20across%20different%20securities,adding%20ticker%20information%20in%20it) [[10]](https://www.reddit.com/r/TradingView/comments/s9uplg/need_the_ability_to_export_backtesting_results/#:~:text=across%20multiple%20security%20and%20then,adding%20ticker%20information%20in%20it) [[64]](https://www.reddit.com/r/TradingView/comments/s9uplg/need_the_ability_to_export_backtesting_results/#:~:text=PrimordialRocks) Need the ability to export backtesting results for multiple symbols at once. : r/TradingView

<https://www.reddit.com/r/TradingView/comments/s9uplg/need_the_ability_to_export_backtesting_results/>

[[6]](https://stackoverflow.com/questions/61862243/pine-script-multiple-symbol-strategy#:~:text=PineCoders) [[13]](https://stackoverflow.com/questions/61862243/pine-script-multiple-symbol-strategy#:~:text=10) Pine-Script + Multiple Symbol Strategy - Stack Overflow

<https://stackoverflow.com/questions/61862243/pine-script-multiple-symbol-strategy>

[[11]](https://www.backtestbase.com/#:~:text=,settings%20and%20parameters%20worked%20best) [[12]](https://www.backtestbase.com/#:~:text=,TradingView%20backtests%20in%20one%20place) [[57]](https://www.backtestbase.com/#:~:text=The%20Solution) [[58]](https://www.backtestbase.com/#:~:text=Transform%20individual%20trading%20strategies%20into,returns%20through%20systematic%20portfolio%20construction) TradingView Backtest Analysis | Strategy Comparison & Monte Carlo – BacktestBase

<https://www.backtestbase.com/>

[[14]](https://www.tradingview.com/support/solutions/43000674726-why-is-my-account-banned-due-to-suspicious-activity/#:~:text=,TradingView%20content%20is%20expressly%20forbidden) [[15]](https://www.tradingview.com/support/solutions/43000674726-why-is-my-account-banned-due-to-suspicious-activity/#:~:text=Please%20note%20that%20the%20use,are%20for%20manual%20use%20only) Why is my account banned due to suspicious activity? — TradingView

<https://www.tradingview.com/support/solutions/43000674726-why-is-my-account-banned-due-to-suspicious-activity/>

[[17]](https://www.tradingview.com/u/mnspppppp/#:~:text=Trader%20mnspppppp%20%E2%80%94%20Trading%20Ideas,GBPUSD%3A%20GU%20BUY) Trader mnspppppp — Trading Ideas & Charts — TradingView ...

<https://www.tradingview.com/u/mnspppppp/>

[[18]](https://www.reddit.com/r/TradingView/comments/goq969/feature_requests_comment_yours_here/#:~:text=%E2%80%A2%20%205y%20ago) [[34]](https://www.reddit.com/r/TradingView/comments/goq969/feature_requests_comment_yours_here/#:~:text=%E2%80%A2%20%205y%20ago) [[35]](https://www.reddit.com/r/TradingView/comments/goq969/feature_requests_comment_yours_here/#:~:text=Support%20replied%20to%20my%20message,no%20timeline%20could%20be%20given) [[53]](https://www.reddit.com/r/TradingView/comments/goq969/feature_requests_comment_yours_here/#:~:text=I%27ve%20been%20wanting%20this%20for,betting%20on%20the%20wrong%20horse) [[63]](https://www.reddit.com/r/TradingView/comments/goq969/feature_requests_comment_yours_here/#:~:text=%2B1%20we%20need%20this%21%2010,give%20a%20timeframe%20on%20this) Feature Requests: comment yours here! : r/TradingView

<https://www.reddit.com/r/TradingView/comments/goq969/feature_requests_comment_yours_here/>

[[19]](https://www.brokersview.com/news/tadingview-usage-stats-for-2024-224255#:~:text=Key%20TradingView%20Stats) [[21]](https://www.brokersview.com/news/tadingview-usage-stats-for-2024-224255#:~:text=,3%20million%20on%20TradingView) [[25]](https://www.brokersview.com/news/tadingview-usage-stats-for-2024-224255#:~:text=TradingView%20boasts%20a%20staggering%2050,actively%20exploring%20the%20platform%27s%20capabilities) [[26]](https://www.brokersview.com/news/tadingview-usage-stats-for-2024-224255#:~:text=,3%20million%20on%20TradingView) [[27]](https://www.brokersview.com/news/tadingview-usage-stats-for-2024-224255#:~:text=TradingView%20Desktop%20Traffic%20Distribution%20%EF%BC%88By,Country%EF%BC%89) [[28]](https://www.brokersview.com/news/tadingview-usage-stats-for-2024-224255#:~:text=TradingView%20caters%20to%20a%20global,28) TadingView Usage Stats for 2024 - BrokersView

<https://www.brokersview.com/news/tadingview-usage-stats-for-2024-224255>

[[20]](https://www.tradingview.com/about/#:~:text=Hello%2C%20Trader) About — TradingView

<https://www.tradingview.com/about/>

[[22]](https://www.quantconnect.com/#:~:text=Platform,quant%20research%20community%20in) QuantConnect.com: Open Source Algorithmic Trading Platform.

<https://www.quantconnect.com/>

[[23]](https://www.grandviewresearch.com/horizon/statistics/algorithmic-trading-market/type-of-trader/retail-investors/global#:~:text=Retail%20Investors%20,Market%20Statistics) [[24]](https://www.grandviewresearch.com/horizon/statistics/algorithmic-trading-market/type-of-trader/retail-investors/global#:~:text=The%20global%20retail%20investors%20algorithmic,from%202024%20to%202030) Retail Investors - Algorithmic trading market outlook

<https://www.grandviewresearch.com/horizon/statistics/algorithmic-trading-market/type-of-trader/retail-investors/global>

[[29]](https://ibsintelligence.com/ibsi-news/tradingview-hits-record-550-million-unique-users/#:~:text=Individual%2C%20non,their%20own%20personal%20financial%20decisions) [[31]](https://ibsintelligence.com/ibsi-news/tradingview-hits-record-550-million-unique-users/#:~:text=Fuelled%20by%20the%20recent%20global,critical%20force%20in%20financial%20markets) [[32]](https://ibsintelligence.com/ibsi-news/tradingview-hits-record-550-million-unique-users/#:~:text=TradingView%2C%20which%20celebrates%20its%2010,of%20today%E2%80%99s%20global%20population) TradingView hits record 550 million unique users

<https://ibsintelligence.com/ibsi-news/tradingview-hits-record-550-million-unique-users/>

[[30]](https://www.mordorintelligence.com/industry-reports/algorithmic-trading-market#:~:text=,CAGR) Algorithmic Trading Market Size, Growth, Share & Forecast Report 2030

<https://www.mordorintelligence.com/industry-reports/algorithmic-trading-market>

[[33]](https://algotest.in/blog/streak-vs-tradetron/#:~:text=Streak%20vs,For%20more%20information%2C%20please) Streak vs. Tradetron: A Comprehensive Comparison for Traders

<https://algotest.in/blog/streak-vs-tradetron/>

[[36]](https://www.newtrading.io/trendspider-review/#:~:text=While%20it%E2%80%99s%20very%20powerful%2C%20there,the%20heavy%20lifting%20for%20you) [[37]](https://www.newtrading.io/trendspider-review/#:~:text=TrendSpider%20is%20an%20AI,level%20models) [[38]](https://www.newtrading.io/trendspider-review/#:~:text=No,refine%20strategies%20without%20programming%20knowledge) [[39]](https://www.newtrading.io/trendspider-review/#:~:text=Another%20feature%20is%20TrendSpider%E2%80%99s%20multi,Whether%20it%E2%80%99s%20stocks%2C%20crypto%2C%20or) [[40]](https://www.newtrading.io/trendspider-review/#:~:text=match%20at%20L472%20Meanwhile%2C%20a,bot%20execution%20can%20be%20frustrating) [[51]](https://www.newtrading.io/trendspider-review/#:~:text=Fees%20and%20Pricing%20Standard%20%E2%80%93,%E2%80%93%20%241%2C596%2FYear%20Advanced%20%E2%80%93%20%242%2C796%2FYear) [[52]](https://www.newtrading.io/trendspider-review/#:~:text=Advanced%24349%242%2C796%E2%80%93%2020%20open%20workspaces%20%E2%80%93,30K%20depth) [[61]](https://www.newtrading.io/trendspider-review/#:~:text=Pricing%2489%2Fmonth%28Standard%29%2433,Free%20platform%20but%20requires%20a) TrendSpider Review: Is The Platform Worth It?

<https://www.newtrading.io/trendspider-review/>

[[41]](https://www.reddit.com/r/algotrading/comments/13la56p/what_happened_to_quantconnect/#:~:text=strategies%20and%20take%20advantage%20of,evening%20of%20coding%20%2B%20backtesting) [[42]](https://www.reddit.com/r/algotrading/comments/13la56p/what_happened_to_quantconnect/#:~:text=Furthermore%2C%20simple%20quality%20of%20life,much%20less%20active%20there%20now) [[43]](https://www.reddit.com/r/algotrading/comments/13la56p/what_happened_to_quantconnect/#:~:text=line%20of%20your%20code%20,much%20less%20active%20there%20now) [[54]](https://www.reddit.com/r/algotrading/comments/13la56p/what_happened_to_quantconnect/#:~:text=line%20of%20your%20code%20,much%20less%20active%20there%20now) [[65]](https://www.reddit.com/r/algotrading/comments/13la56p/what_happened_to_quantconnect/#:~:text=GreenBlueberries) What Happened to Quantconnect? : r/algotrading

<https://www.reddit.com/r/algotrading/comments/13la56p/what_happened_to_quantconnect/>

[[44]](https://www.quantconnect.com/pricing/#:~:text=QuantConnect%20is%20a%20multi,than%20275000%20quants%20and%20engineers) QuantConnect Pricing Page

<https://www.quantconnect.com/pricing/>

[[45]](https://www.liberatedstocktrader.com/tradingview-vs-trade-ideas/#:~:text=BacktestingImage%3A%20Backtesting%20Score,140%20%E2%98%85%E2%98%85%E2%98%85%E2%98%85%E2%9C%A9%E2%98%85%E2%98%85%E2%98%85%E2%98%85%E2%98%85) TradingView vs. Trade Ideas: AI vs. Social Trading 2025 Test

<https://www.liberatedstocktrader.com/tradingview-vs-trade-ideas/>

[[46]](https://www.trade-ideas.com/pricing/#:~:text=TI%20Premium%3A%20AI,signals%20to%20maximize%20profits) [[47]](https://www.trade-ideas.com/pricing/#:~:text=TI%20Premium%3A%20AI,signals%20to%20maximize%20profits) Trade Ideas Subscription Plans - Standard to Premium Trading Tools

<https://www.trade-ideas.com/pricing/>

[[48]](https://www.amibroker.com/guide/h_portfolio.html#:~:text=AmiBroker%27s%20portfolio%20backtester%20lets%20you,like%20you%20do%20in%20reality) [[49]](https://www.amibroker.com/guide/h_portfolio.html#:~:text=AmiBroker%27s%20portfolio%20backtester%20lets%20you,like%20you%20do%20in%20reality) Portfolio-level back testing

<https://www.amibroker.com/guide/h_portfolio.html>

[[50]](https://www.amibroker.com/products.html#:~:text=AmiBroker%20www,Download%20Buy) $299 Professional Edition: $369 Ultimate Pack Pro - AmiBroker

<https://www.amibroker.com/products.html>

[[55]](https://www.trade-ideas.com/2025/01/19/trade-ideas-leading-the-financial-technology-revolution-why-we-were-voted-most-trusted-trading-platform-by-capterra/#:~:text=With%20an%20outstanding%204,through%20its%20advanced%20AI) 2024's Most Trusted Trading Platform - Trade Ideas

<https://www.trade-ideas.com/2025/01/19/trade-ideas-leading-the-financial-technology-revolution-why-we-were-voted-most-trusted-trading-platform-by-capterra/>

[[56]](https://www.amibroker.com/guide/h_pyramid.html#:~:text=Pyramiding%20,features%20are%20supported%20by) Pyramiding (scaling in/out) and mutliple currencies in the portfolio ...

<https://www.amibroker.com/guide/h_pyramid.html>

[[59]](https://jatinkathiriya.medium.com/portfolio-backtesting-using-google-sheets-3a65cbce2acf#:~:text=Medium%20jatinkathiriya,returns%20to%20the%20S%26P500%20benchmark) Portfolio Backtesting Using Google Sheets | by Jatin - Medium

<https://jatinkathiriya.medium.com/portfolio-backtesting-using-google-sheets-3a65cbce2acf>

[[60]](https://www.tradingview.com/pricing/#:~:text=Plans%20for%20every%20level%20of,%C2%B7%204%20charts%20per%20tab) TradingView Subscriptions: Pricing and Features

<https://www.tradingview.com/pricing/>

[[62]](https://docs.cryptohopper.com/docs/my-library/what-is-the-backtester/#:~:text=What%20is%20the%20Backtester%20,subscriptions%3A%2010%20backtests%20per%20day) What is the Backtester | Cryptohopper Documentation

<https://docs.cryptohopper.com/docs/my-library/what-is-the-backtester/>

[[66]](https://www.reddit.com/r/TradingView/comments/1guzc2j/portfolio_backtesting_feature/#:~:text=One%20feature%20that%20I%20am,instruments%20in%20a%20single%20portfolio) Portfolio backtesting feature : r/TradingView - Reddit

<https://www.reddit.com/r/TradingView/comments/1guzc2j/portfolio_backtesting_feature/>

[[67]](https://www.youtube.com/watch?v=-WwzYGmC6po#:~:text=TradingView%20PORTFOLIO%20Backtesting%20Tool%20for,profitable%20trading%20experience%2C%20I) TradingView PORTFOLIO Backtesting Tool for Strategy Tester (FREE!)

<https://www.youtube.com/watch?v=-WwzYGmC6po>