

ISSN 2809-8501 (Online)

UTSAHA: Journal of Entrepreneurship

<https://journal.jfpublisher.com/index.php/joe>

Vol. 4, Issue 4, October 2025

doi.org/10.56943/joe.v4i4.882

Prototyping Model for Designing a Desktop-Based Trading Robot Application

Ali Rahmad¹, Fatchur Rozi², Andy Rachman³, Anggi Yhurinda Perdana Putri⁴

¹ali.rahmad@rocketmail.com, ²itsozie77@gmail.com, ³andy.rach1910@itats.ac.id,

⁴anggi@itats.ac.id

Institut Teknologi Adhi Tama Surabaya

*Corresponding Author: Ali Rahmad

E-mail: ali.rahmad@rocketmail.com

ABSTRACT

The rapid development of the digital economy has stimulated increasing public awareness and literacy regarding investment and trading activities, particularly following the COVID-19 pandemic. This phenomenon has triggered the emergence of various new investment instruments, including cryptocurrency, foreign exchange, and commodity trading such as gold. To minimize trading chaos and failure resulting from emotional and psychological factors, automated trading applications known as Expert Advisors represent viable alternative solutions. This research aims to develop a desktop-based trading robot application using the prototyping model methodology. The research methodology encompassed field surveys, interviews, literature reviews, iterative application development, and comprehensive feasibility testing. The prototyping model consisted of sequential processes: initial requirement gathering, design, prototype construction, customer evaluation, review and modification, customer satisfaction assessment, development, testing, and maintenance. The developed trading robot application was successfully implemented and evaluated through the System Usability Scale (SUS), achieving an aggregate score of 88.21%, indicating excellent usability and substantial user convenience facilitation. Furthermore, the application underwent quality assessment using ISO 9126-3 framework, evaluating Functionality and Usability dimensions. The Functionality factor attained an average score of 89%, confirming that the application functions exceptionally well, while the Usability factor achieved an average score of 95%, demonstrating high usefulness. These findings validate the effectiveness of the prototyping model in developing automated trading systems that balance technical sophistication with user accessibility.

Keywords: Automated Trading System, Expert Advisor, ISO 9126-3, Prototyping Model, System Usability Scale

INTRODUCTION

In the contemporary era, the digital economy has experienced rapid development alongside increasing public awareness and financial literacy regarding the importance of investment and trading activities. Primary data obtained through questionnaire surveys from 120 respondents revealed that the majority of retail investors who decided to invest in stocks during the pandemic period were individuals under 30 years of age, with monthly incomes below Rp 4.8 million, and a preference for short-term investment strategies (trading) (Lubis & Kusuma, 2022). This phenomenon has stimulated the emergence of various new investment instruments circulating within society, including cryptocurrency, foreign exchange, commodity trading, and similar vehicles. Among these, gold trading has gained prominence as a significant component of commodity markets. Gold trading represents a commercial entity that has recently gained widespread attention among retail investors. When executed properly by users (commonly referred to as traders), gold trading presents substantial business opportunities (Syaputra & Voutama, 2021). The desire to generate significant profits remains the most prevalent motivation among traders engaging in this business. However, this ambition is frequently not accompanied by adequate knowledge or psychological preparedness regarding trading practices (Syaputra & Voutama, 2021).

Within the trading domain, three fundamental pillars determine trader success: psychology, market analysis, and risk management (Elder, 2014). The success or failure of traders depends heavily on their emotional regulation. A trader may possess a sophisticated trading system, yet if dominated by emotions such as fear, greed, frustration, or arrogance, they will inevitably experience disorder and potential failure in their trading endeavours. Conversely, successful traders must maintain maximum objectivity throughout their trading activities (Elder, 2014). To minimize the chaos and failure in trading resulting from these psychological factors, one viable alternative solution involves implementing automated trading applications, commonly known as Expert Advisors. An Expert Advisor (EA), also referred to as a Forex Robot, constitutes supplementary software or script within a trading application that functions as an automated trading engine capable of executing trades independently (Alqodri et al., 2015; Jamilah & Mardiana, 2024).

The proliferation of automated trading systems has attracted substantial academic attention in recent years. Recent evidence indicates that algorithmic trading has achieved significant market penetration, with approximately 53% of hedge fund trades and 58% of foreign exchange transactions being executed through algorithmic systems (Garg, 2023). Furthermore, market projections suggest that the algorithmic trading sector will expand from \$11.1 billion in 2019 to \$18.8 billion by 2024, underscoring the rapid growth trajectory of this technology within financial markets. These developments highlight the increasing reliance on

automated systems across various trading domains, including commodity markets such as gold trading.

Previous research has explored multiple dimensions of automated trading system development and implementation. Studies investigating the integration of machine learning techniques with algorithmic trading have demonstrated promising results in terms of predictive accuracy and profitability optimization (Qureshi et al., 2025). For instance, recent work on deep reinforcement learning applications has shown that algorithms such as Advantage Actor-Critic (A2C), Deep Deterministic Policy Gradient (DDPG), Proximal Policy Optimization (PPO), Soft Actor-Critic (SAC), and Twin Delayed Deep Deterministic Policy Gradient (TD3) can effectively learn and adapt to volatile market conditions, thereby enhancing portfolio performance (Lu, 2025). However, these studies have also identified persistent challenges, including overfitting tendencies, model instability, limited interpretability of complex algorithms, and difficulties in managing slippage and latency issues. These limitations underscore the necessity for continued refinement in automated trading system design.

Furthermore, empirical investigations into automated trading performance have yielded mixed findings regarding profitability and risk management effectiveness. While some studies demonstrate that properly designed automated systems can outperform human traders in specific market environments, other research suggests that algorithm performance is highly dependent on market conditions and the fundamental characteristics of traded assets (Calvez & Cliff, 2018). The literature emphasizes that successful implementation requires robust risk control mechanisms, comprehensive backtesting procedures, and careful parameter optimization. Additionally, Wu et al. (2022) has highlighted the critical importance of money management strategies in mitigating excessive risk exposure during automated trading operations.

Despite the growing body of research on automated trading systems, several gaps remain in the literature. Most existing studies focus primarily on stock and foreign exchange markets, with limited attention devoted to commodity trading applications, particularly gold trading. Moreover, many commercial Expert Advisors lack transparency in their operational mechanisms and fail to provide users with adequate control features. The issue of unauthorized usage and inadequate licensing mechanisms has also received insufficient attention in prior research. These gaps motivate the current study's focus on developing a desktop-based trading robot with integrated licensing and user control features specifically designed for gold trading applications.

This research aims to design a desktop-based trading robot utilizing the prototyping model. Design and development represents the initial stage of creating conceptual frameworks and structural sketches that have not been previously constructed, which are then processed into functional designs serving the desired purposes (Mukramin & Hadi, 2022). Given the rapid evolution of software

technology, measurement standards must be adapted to accommodate these changes (Yhurinda et al., 2019). The prototyping model approach was selected because it enables the researcher, acting as the system developer, to clearly identify and interpret consumer requirements. This occurs because system developers, clients, and end-users can observe and test computer system components from the earliest stages of the development process (Putra & Jibril, 2024).

One distinctive advantage of this trading robot application lies in its integrated licensing mechanism, which legitimizes client ownership and exclusive access rights to the trading robot application. Consequently, the application cannot be arbitrarily used by unauthorized individuals. Additionally, this trading robot application incorporates control button features that enable the robot to determine trading positions (buy/sell) to be executed.

LITERATURE REVIEW

Robot Trading / Expert Advisor

The term “robot” originated as an acronym for Residents Official Board of Technology. Robots have significantly impacted various industries, particularly in processing operations where they have enhanced productivity and efficiency in industrial production through their capacity to perform repetitive tasks without experiencing fatigue or committing errors (Macpherson et al., 2022). A robot constitutes a complex and intelligent machine that integrates knowledge from multiple disciplines, including mechanics, electronics, control systems, computer science, sensors, and artificial intelligence. Robots represent highly sophisticated and versatile machines possessing the capability to perform diverse tasks with precision and efficiency (Khanesar & Branson, 2022). Trading robots prove particularly suitable for traders seeking comfortable trading experiences, as these systems can execute trades with relatively greater speed and accuracy compared to human traders. Expert Advisors represent robotic systems or applications implemented within specific platforms. The platform in question is MetaTrader5, which was developed specifically using MQL5, supported by C programming language, to facilitate trader transactions (Dinata, 2018).

The evolution of Expert Advisors has progressed considerably from simple rule-based systems to sophisticated algorithms incorporating advanced technical indicators and pattern recognition capabilities. Contemporary Expert Advisors can simultaneously monitor multiple currency pairs or commodities, execute complex trading strategies, and implement dynamic position sizing based on real-time market volatility. However, the effectiveness of these systems remains contingent upon several critical factors, including proper parameter optimization, adequate backtesting across diverse market conditions, and continuous monitoring to ensure alignment with evolving market dynamics. Research has demonstrated that while Expert Advisors eliminate emotional decision-making and ensure consistent strategy execution, they may struggle during unprecedented market events or

structural shifts that fall outside their training parameters. Therefore, the integration of manual oversight mechanisms and user control features represents a crucial design consideration in modern Expert Advisor development.

Money Management

Money management strategies constitute methods for minimizing excessive risk during trading activities (Yulianti, 2019). The substantial losses of considerable fortunes due to the absence of simple risk control procedures provides sufficient evidence that money management represents a critical component of trading success (Kadir & Musdiana, 2024; Mcdowell, 2008). Inadequate money management can result in trades that are disproportionately large or small relative to the capital at risk (Young, 2013).

Effective money management extends beyond simple position sizing to encompass comprehensive risk-reward ratio analysis, drawdown limitations, and portfolio diversification principles. Professional traders typically risk no more than one to two percent of their total capital on any single trade, thereby ensuring that consecutive losses do not substantially erode their trading accounts. Furthermore, sophisticated money management approaches incorporate concepts such as the Kelly Criterion for optimal position sizing, fixed fractional methods, and dynamic risk adjustment based on account performance. The integration of these principles within automated trading systems presents unique challenges, as algorithms must balance aggressive profit-seeking behaviour with conservative capital preservation objectives. Research indicates that automated trading systems equipped with robust money management protocols demonstrate significantly improved long-term sustainability compared to systems focused solely on maximizing short-term returns.

Prototyping Model

The defining characteristic of this methodology is that system designers and clients can observe and experiment with computer system components from the initial development stages (Susanto & Meiryani, 2019). The prototyping approach offers distinct advantages in software development contexts characterized by evolving requirements and the need for continuous stakeholder feedback. Unlike traditional waterfall methodologies that demand comprehensive upfront specifications, prototyping facilitates iterative refinement through successive approximations of the final system. This methodology proves particularly valuable in developing specialized applications such as trading robots, where user requirements may not be fully articulated at project inception and where functional validation requires hands-on experimentation with live market data. The prototyping process typically encompasses several iterations, each incorporating user feedback to progressively enhance system functionality, usability, and performance. However, potential drawbacks include the risk of scope creep,

inadequate documentation, and the temptation to deploy preliminary prototypes as production systems without sufficient optimization. Therefore, disciplined project management and clear transition criteria from prototype to production remain essential for successful implementation.

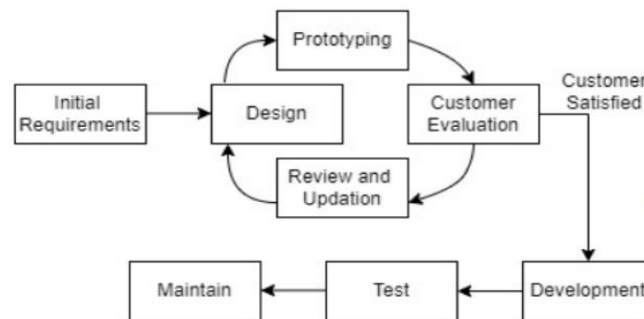


Figure 1 Prototyping Model
Source: Author's Database

SUS (System Usability Scale)

The System Usability Scale represents a widely employed methodology for measuring user satisfaction and evaluating product and system usability (M. L. Hamzah et al., 2021). The System Usability Scale functions as an instrument that assesses customer satisfaction levels based on multiple-choice questionnaire responses (Othman & Harun, 2021). This scale is typically structured using a Likert-type format, wherein respondents rate their satisfaction on a scale ranging from 1 to 5 or alternative numerical ranges (Todorovic et al., 2017).

Since its development in 1986, the System Usability Scale has emerged as one of the most extensively validated and reliable usability assessment instruments in human-computer interaction research. The ten-item questionnaire alternates between positive and negative statements to minimize response bias and enhance measurement reliability. Scores are calculated through a standardized formula that converts raw responses into a scale ranging from 0 to 100, with scores above 68 generally considered above average usability. The SUS demonstrates several methodological advantages, including its technology-agnostic nature, ease of administration, cost-effectiveness, and robust psychometric properties validated across diverse domains and cultures. In the context of trading software evaluation, the SUS provides quantitative metrics for assessing user interface intuitiveness, learning curve steepness, and overall user experience quality, thereby facilitating evidence-based design improvements and comparative benchmarking against industry standards.

ISO 9126-3

ISO 9126-3 constitutes an ISO management standard that plays a critical role in evaluating software quality and performance (N. Hamzah et al., 2021). This standard provides guidelines and criteria for assessing various software aspects,

including functionality, reliability, efficiency, maintainability, and usability (Capiluppi et al., 2009). By adhering to ISO 9126-3 standards, organizations can ensure their software meets the highest industry standards and delivers superior quality levels to end-users (Rwiza et al., 2020).

The ISO 9126 framework decomposes software quality into six primary characteristics, each further subdivided into measurable sub-characteristics that facilitate comprehensive evaluation. Functionality encompasses suitability, accuracy, interoperability, security, and compliance aspects. Reliability addresses maturity, fault tolerance, and recoverability dimensions. Efficiency considers time behaviour and resource utilization. Maintainability evaluates analysability, changeability, stability, and testability. Usability examines understandability, learnability, operability, and attractiveness. Finally, portability assesses adaptability, installability, co-existence, and replaceability attributes. The ISO 9126-3 component specifically addresses internal metrics that can be applied to non-executable software deliverables during development phases, enabling early quality assessment and defect prevention. For trading robot applications, adherence to ISO 9126-3 principles ensures systematic evaluation of critical attributes such as calculation accuracy, system stability under stress conditions, response time performance, and maintainability for future enhancements, thereby establishing a foundation for delivering reliable and professionally engineered software solutions.

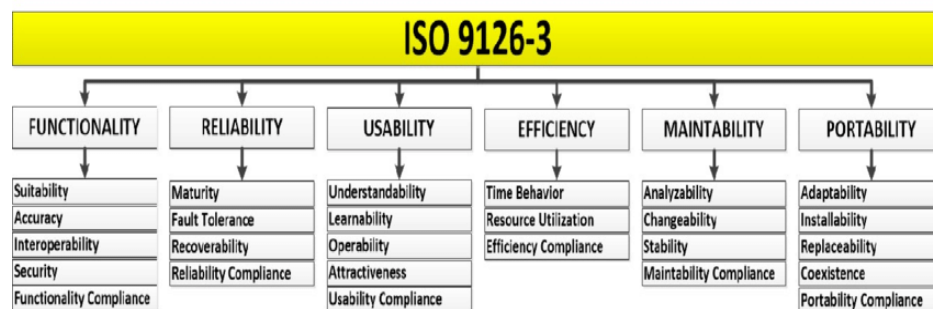


Figure 2 ISO 9126-3
Source: Author's Database

RESEARCH METHODOLOGY

This research employs the prototyping model as its primary methodological framework. According to research published in "Using Prototyping Method for Analysis and Design of Information Systems for Student Registration in Sekolah Master," the prototyping model represents an optimal approach for addressing misunderstandings between users and system analysts that arise when users cannot clearly articulate system requirements. Recent empirical investigations have validated the effectiveness of prototyping-oriented development methodologies in ameliorating weaknesses inherent in traditional life cycle-oriented approaches (Bjarnason et al., 2023). Furthermore, prototyping has been established as a critical

practice within requirements engineering, particularly in contexts demanding iterative stakeholder collaboration (Bjarnason et al., 2023). The system design aims to provide comprehensive explanations of the proposed system architecture. The application development follows the prototyping model, which encompasses sequential processes including initial requirement gathering, design conceptualization, prototype construction, customer evaluation, review and modification, customer satisfaction assessment, full-scale development, rigorous testing, and ongoing maintenance. The research implementation adhered to a structured workflow specifically tailored for trading robot application development.

The initial phase involved conducting a comprehensive field survey at the research site, located at the residence of Mr. Irfan Taufiq, B.Com., MCompStud., at Perum Karah Indah, Jl. Jambangan Baru Sel. Raya Block H No.20, Karah, Jambangan District, Surabaya City, East Java 60232. Following the preliminary survey, mutual agreement was established between the researcher and the client to conduct intensive development activities at the client's residence. This arrangement facilitated continuous collaboration and immediate feedback integration throughout the development process, ensuring alignment between system capabilities and user expectations. Such collaborative approaches have been demonstrated to increase the identification of system requirements and user needs significantly compared to traditional demonstration-only methodologies (Gómez & Figueroa, 2024).

The second phase comprised in-depth interviews designed to establish the informant's profile and elicit detailed information regarding strategic considerations for trading robot application development. Through systematic interviewing, the researcher determined that the indicators and benchmarks for position-taking and entry strategy formulation would utilize multiple moving averages in conjunction with specialized indicators for identifying market trend direction (bullish or bearish). These technical specifications formed the foundation for the robot's decision-making algorithms and trading logic implementation. The development of algorithmic trading systems necessitates careful consideration of quantitative analysis frameworks, programming proficiency, and comprehensive understanding of market mechanisms (Sevastjanov et al., 2024).

The literature review phase was undertaken to augment knowledge, broaden perspectives, and establish theoretical foundations essential for trading robot application development. This scholarly investigation involved comprehensive examination of relevant textbooks and scientific publications addressing automated trading systems, algorithmic trading strategies, software engineering methodologies, and related computational finance topics. Recent systematic reviews of algorithmic trading literature have emphasized the technology-driven advancements and temporal progression of automated trading methodologies, providing essential context for contemporary system development (Horobet et al., 2024). The literature review provided crucial insights into best practices, common pitfalls, and emerging trends in trading automation technology, particularly

regarding statistical and machine learning methods employed in automated trading systems (Shavandi & Khedmati, 2022).

The application development phase encompassed multiple sequential stages integral to the prototyping methodology. The initial requirement gathering stage involved systematic documentation of functional and non-functional system requirements through collaborative sessions with the client. The design phase translated these requirements into architectural blueprints, including system flow diagrams, database schemas, and user interface mockups. Subsequently, the prototyping stage produced a functional preliminary version of the trading robot, incorporating core features and basic trading logic. This prototype underwent iterative review and modification cycles based on customer feedback and performance evaluation against predefined criteria. The modular approach to prototyping enables demonstration of software tool concepts at both operational principle and implementation resource levels to various stakeholders (Bois & Gerritsen, 2013). Upon achieving customer satisfaction with prototype functionality and usability, the development phase commenced, involving complete system implementation with refined algorithms, comprehensive error handling, and optimized performance characteristics. The testing phase employed systematic validation procedures to ensure system reliability, accuracy, and stability under various market conditions. Finally, the maintenance phase established protocols for ongoing system monitoring, bug fixes, performance optimization, and feature enhancements based on evolving user requirements and market dynamics.

RESULT AND DISCUSSION

Prototyping Iterations

Following the completion of the design phase and user interface development, the researcher presented and demonstrated the prototyping design outcomes to the client and end-users to facilitate observation of the trading robot application's operational flow and functional capabilities. The prototyping iteration process was conducted three times to achieve results that aligned with client expectations and requirements. These iterative refinements occurred during the customer evaluation phase, where feedback was systematically collected and incorporated into subsequent prototype versions. This iterative approach enabled progressive enhancement of system functionality, interface design, and overall user experience through continuous stakeholder engagement (Bjarnason et al., 2023).

System Design

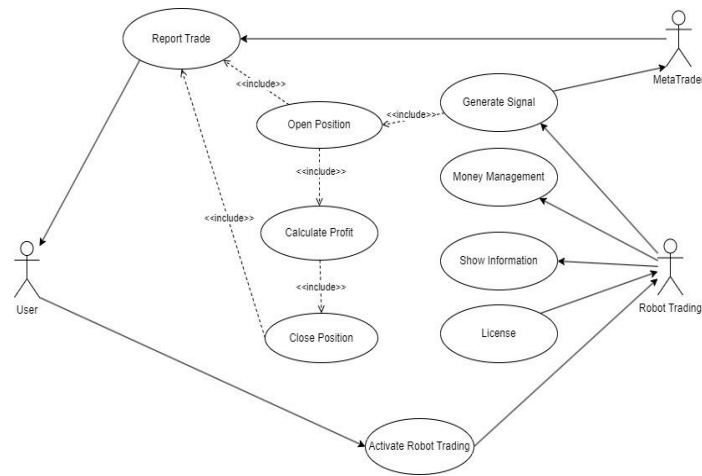


Figure 3 Use Case Diagram

Source: Author's Database

The system architecture was comprehensively documented through Unified Modeling Language diagrams to illustrate structural and behavioural aspects of the trading robot application. The use case diagram, presented in Figure 3, delineates the primary actors and their interactions with system functionalities, encompassing user authentication, trading parameter configuration, automated trade execution, position monitoring, and report generation capabilities. This visual representation facilitates clear communication of system requirements and functional boundaries between developers and stakeholders (Wang et al., 2025).

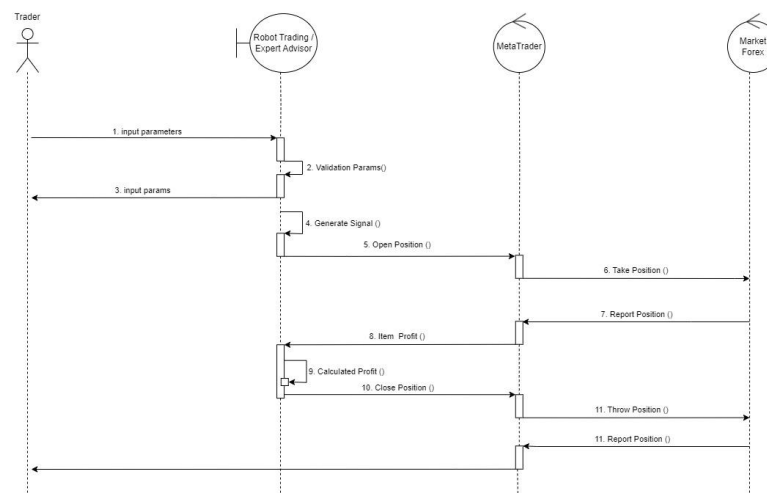


Figure 4 Sequence Diagram

Source: Author's Database

The sequence diagram, illustrated in Figure 4, demonstrates the temporal ordering of interactions between system components during typical operational scenarios. This diagram captures the message exchanges between the user interface, trading logic module, MetaTrader5 platform interface, and data management components throughout the trading workflow. The sequential visualization clarifies the control flow during critical operations such as signal generation, position entry validation, risk management checks, and order execution processes. Such behavioral modeling ensures comprehensive understanding of system dynamics and facilitates identification of potential bottlenecks or failure points.

Implementation and Operational Results

The trading robot application was successfully implemented and integrated within the MetaTrader5 platform, as depicted in Figure 5. The implementation demonstrates seamless interoperability with the MetaTrader5 environment, leveraging the MQL5 programming language to access market data, execute trades, and manage positions programmatically. The user interface provides intuitive controls for activating and deactivating the automated trading system, adjusting risk parameters, and monitoring real-time performance metrics. The integration architecture ensures low-latency communication between the Expert Advisor and the trading server, minimizing execution delays that could adversely impact trading performance.



Figure 5 Display of Trading Robot Application Results

Source: Author's Database

The application underwent extensive testing using historical market data to evaluate its trading performance across different timeframes and market conditions. Figure 6 presents the detailed transaction report generated during testing on a 30-minute timeframe, documenting individual trade entries, exits, position sizes, profit/loss outcomes, and cumulative performance metrics. The report demonstrates the robot's capacity to execute trades according to the programmed strategy, maintaining consistent application of entry and exit rules based on multiple moving

average signals and trend direction indicators. Over the testing period, the system executed trades with systematic adherence to predefined risk management protocols, limiting individual position exposure in accordance with money management principles.

Results			
History Quality:	95%		
Bars:	954	Ticks:	2913564
Total Net Profit:	439.22	Balance Drawdown Absolute:	0.00
Gross Profit:	4 904.22	Balance Drawdown Maximal:	620.68 (30.82%)
Gross Loss:	-4 465.00	Balance Drawdown Relative:	30.82% (620.68)
		Symbols:	1
		Equity Drawdown Absolute:	6.90
		Equity Drawdown Maximal:	673.59 (33.45%)
		Equity Drawdown Relative:	33.45% (673.59)
Profit Factor:	1.10	Expected Payoff:	3.25
Recovery Factor:	0.65	Sharpe Ratio:	2.41
AHPR:	1.0039 (0.39%)	LR Correlation:	0.72
GHPR:	1.0027 (0.27%)	LR Standard Error:	141.99
		Margin Level:	2638.91%
		Z-Score:	0.05 (3.99%)
		OnTester result:	0
Total Trades:	135	Short Trades (won %):	50 (50.00%)
Total Deals:	270	Profit Trades (% of total):	72 (53.33%)
		Largest profit trade:	112.20
		Average profit trade:	68.11
		Maximum consecutive wins (\$):	6 (501.05)
		Maximal consecutive profit (count):	501.05 (6)
		Average consecutive wins:	2
		Long Trades (won %):	85 (55.29%)
		Loss Trades (% of total):	63 (46.67%)
		Largest loss trade:	-100.00
		Average loss trade:	-70.87
		Maximum consecutive losses (\$):	5 (-435.87)
		Maximal consecutive loss (count):	-435.87 (5)
		Average consecutive losses:	2

Figure 6 Trading Results Report on a 30-Minute Timeframe

Source: Author's Database

The equity curve visualization, shown in Figure 7, illustrates the progression of account balance over time on the 30-minute timeframe. The graphical representation reveals periods of consistent profitability interspersed with drawdown phases characteristic of commodity market volatility. The relatively smooth equity curve progression suggests effective risk management implementation, with controlled drawdowns that do not exceed acceptable threshold levels. This performance pattern indicates that the trading strategy exhibits resilience across varying market conditions encountered during the testing period.

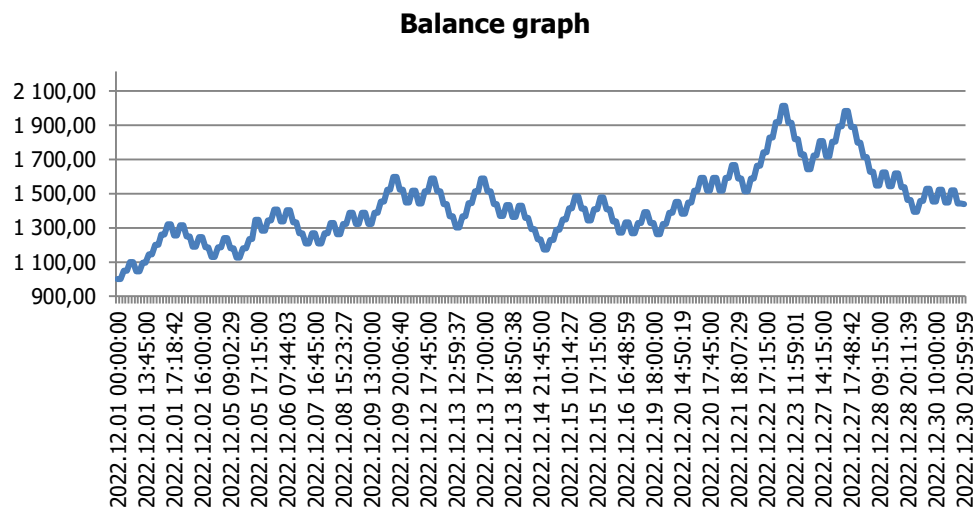


Figure 7 Graph on a 30-Minute Timeframe

Source: Author's Database

Figure 8 displays the temporal distribution of trades executed by the robot, revealing insights into the system's trading frequency and temporal patterns. The time-based analysis demonstrates that the robot maintains consistent trading activity throughout market hours, without exhibiting excessive overtrading or prolonged periods of inactivity. This balanced trading frequency suggests appropriate sensitivity in signal generation, avoiding both overly aggressive trading that could incur excessive transaction costs and overly conservative behavior that might miss valid trading opportunities.

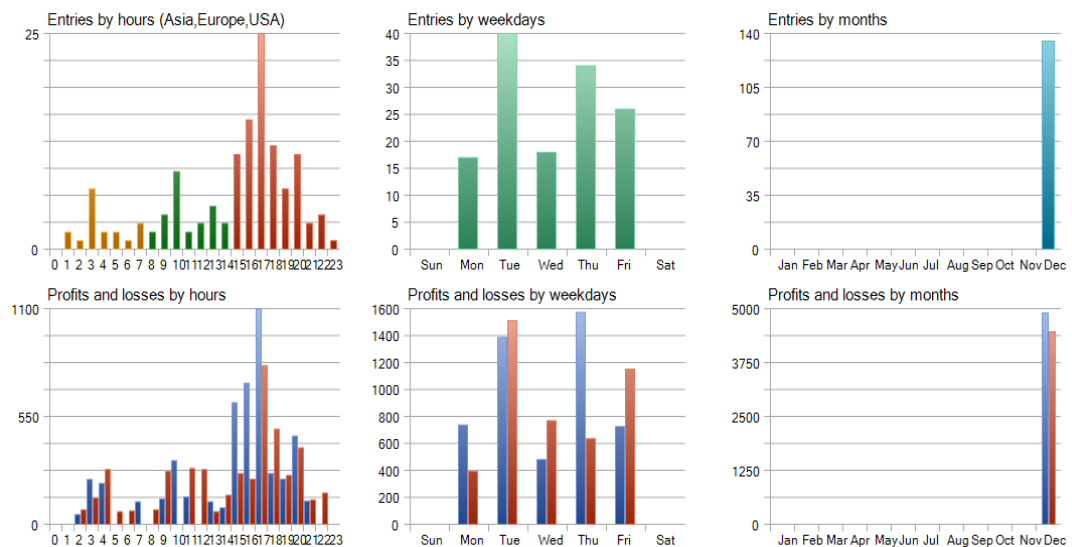


Figure 8 Robot Trading Application Graph Based on Time

Source: Author's Database

Customer Satisfaction Assessment

Customer satisfaction evaluation was conducted through implementation of the System Usability Scale (SUS), a standardized instrument comprising ten questions designed to assess user perceptions of system usability. The questionnaire was administered to seven respondents, including the primary client and representative end-users, to capture diverse perspectives on application usability. Each question employed a five-point Likert scale, with response options ranging from 1 (Strongly Disagree) to 5 (Strongly Agree). The SUS methodology alternates between positively and negatively worded items to minimize response bias and enhance measurement reliability (M. L. Hamzah et al., 2021; Othman & Harun, 2021).

Table 1 SUS Questionnaire Results

No	Question	Original Score							Calculated Score							Value (Total x 2.5)
		R1	R2	R3	R4	R5	R6	R7	R1	R2	R3	R4	R5	R6	R7	
1	Q1	5	5	4	5	5	4	4	4	4	3	4	4	3	3	8.93
2	Q2	2	2	1	1	1	2	2	3	3	4	4	4	3	3	8.57
3	Q3	4	5	5	5	5	4	5	3	4	4	4	4	3	4	9.29
4	Q4	2	2	2	2	1	1	1	3	3	3	3	4	4	4	8.57
5	Q5	4	5	4	5	4	5	5	3	4	3	4	3	4	4	8.93
6	Q6	1	1	2	2	2	2	1	4	4	3	3	3	3	4	8.57
7	Q7	4	4	4	5	5	4	4	3	3	3	4	4	3	3	8.21
8	Q8	1	1	1	1	2	1	1	4	4	4	4	3	4	4	9.64
9	Q9	4	5	4	5	5	5	5	3	4	3	4	4	4	4	9.29
10	Q10	2	2	1	2	2	1	2	3	3	4	3	3	4	3	8.21
SUS Score																88.21

Source: Author's Analysis

Table 1 presents the comprehensive SUS questionnaire results, displaying raw scores from seven respondents (R1-R7) across all ten questions. The scoring methodology involves subtracting one from odd-numbered item scores and subtracting even-numbered item scores from five, summing all converted scores, and multiplying by 2.5 to produce a final score ranging from 0 to 100 (Brooke, 1996). Individual question scores ranged from 8.21 to 9.64 on the normalized scale, indicating consistently positive user responses across different usability dimensions. Question 8, addressing the perceived complexity of system usage, achieved the highest score of 9.64, suggesting users found the application straightforward and not unnecessarily complex. Questions 7 and 10, concerning learning curve and confidence in system usage, received scores of 8.21, representing the lowest yet still highly positive ratings.

The aggregate SUS score calculated from all respondent data yielded a value of 88.21, substantially exceeding the established threshold of 68 for above-average

usability. This score places the trading robot application in the "excellent" category according to standardized SUS interpretation guidelines, indicating superior usability characteristics. The high SUS score reflects successful achievement of design objectives related to user interface intuitiveness, functional clarity, ease of learning, and overall user satisfaction. Such positive usability assessment suggests minimal training requirements and high likelihood of user acceptance and continued system usage.

Application Feasibility Testing Using ISO 9126-3

Application quality assessment was conducted utilizing the ISO 9126-3 framework, focusing on two critical quality characteristics: functionality and usability. The evaluation questionnaire employed a five-point Likert scale administered to seven respondents, capturing user perceptions regarding specific quality attributes. The functionality dimension encompassed questions addressing suitability, accuracy, interoperability, security, and compliance aspects of the trading robot application. The usability dimension evaluated understandability, learnability, operability, attractiveness, and user satisfaction (Capiluppi et al., 2009; N. Hamzah et al., 2021).

Table 2 ISO 9126-3 Questionnaire Results

Factor	Question	R1	R2	R3	R4	R5	R6	R7	Total Highest Score	Total Lowest Score	Total Score	Index Score (%)
Functionality	Q1	5	5	5	4	5	5	4	35	7	33	94.3
	Q2	4	5	5	5	5	4	4	35	7	32	91.4
	Q3	4	5	4	4	4	4	4	35	7	29	82.9
	Q4	3	4	5	5	4	4	5	35	7	30	85.7
	Q5	4	5	4	5	5	5	4	35	7	32	91.4
	Total Index Score											89.14
Usability	Q6	5	5	5	5	5	5	5	35	7	35	100
	Q7	5	5	5	5	5	4	5	35	7	34	97.1
	Q8	5	5	5	5	5	5	5	35	7	35	100
	Q9	5	5	4	4	4	4	4	35	7	30	85.7
	Q10	5	5	5	4	5	4	4	35	7	32	91.4
	Total Index Score											94.84

Source: Author's Analysis

Table 2 presents detailed ISO 9126-3 questionnaire results, organized by quality factors and individual questions. For the functionality assessment, five questions (Q1-Q5) were evaluated, with individual scores calculated as percentages of the maximum possible score. Question 1, assessing the application's suitability for intended trading purposes, achieved a score index of 94.3%, indicating strong alignment between system capabilities and user requirements. Questions 2 and 5, evaluating calculation accuracy and functional completeness, both attained score indices of 91.4%, reflecting user confidence in the system's computational

reliability and feature comprehensiveness. Question 3, concerning integration with the MetaTrader5 platform, received a score index of 82.9%, the lowest within the functionality dimension yet still representing satisfactory interoperability. Question 4, addressing security and access control mechanisms, achieved a score index of 85.7%, suggesting adequate but potentially improvable security features. The aggregate functionality score index of 89.14% substantially exceeds the acceptable threshold of 76%, indicating that the application meets high standards for functional quality.

The usability evaluation comprised five questions (Q6-Q10), examining various aspects of user experience and interface design. Questions 6 and 8, assessing interface understandability and ease of operation, both achieved perfect score indices of 100%, indicating unanimous respondent agreement that the application provides clear, intuitive, and easily operable interfaces. Question 7, evaluating learnability and training requirements, attained a score index of 97.1%, suggesting minimal learning curve and rapid user proficiency development. Question 10, addressing overall user satisfaction, received a score index of 91.4%, reflecting strong positive user sentiment toward the application. Question 9, concerning interface aesthetics and visual appeal, achieved the lowest usability score index of 85.7%, suggesting potential opportunities for visual design enhancement while still maintaining satisfactory aesthetic quality. The aggregate usability score index of 94.84% significantly surpasses the 76% acceptability threshold, demonstrating exceptional usability characteristics.

The comprehensive ISO 9126-3 assessment reveals that the trading robot application achieves high quality standards across both functionality and usability dimensions. The functionality score of 89.14% confirms that the system reliably performs its intended trading operations with accuracy, appropriate platform integration, and adequate security measures. The usability score of 94.84% validates the effectiveness of the user-centered design approach, demonstrating that users find the application understandable, learnable, and satisfying to operate. These quality metrics collectively indicate successful achievement of development objectives and readiness for deployment in operational trading environments.

The convergence of high SUS scores (88.21) and strong ISO 9126-3 ratings (functionality: 89.14%, usability: 94.84%) provides robust evidence of application quality from multiple validated assessment perspectives. These results suggest that the prototyping methodology effectively captured and implemented user requirements while maintaining technical quality standards. The iterative development process, incorporating continuous user feedback and systematic refinement, contributed to achieving superior usability and functionality outcomes compared to traditional development approaches that might not involve such intensive stakeholder collaboration.

CONCLUSION

Based on the findings and discussions elaborated in the previous chapters and the testing results, it can be concluded that the researcher has successfully developed a desktop-based trading robot application using the prototyping model. The trading robot application has been directly tested by users using the System Usability Scale (SUS) and obtained an average score of 88.21%, which indicates that the developed trading robot application greatly facilitates ease of use for users. In addition, this trading robot has also been tested for feasibility using ISO 9126-3 for the Functionality and Usability factors, with an average score for the Functionality factor of 89%, which means the trading robot application functions very well. Meanwhile, the Usability factor has an average score of 95%, which means this trading robot application is highly useful.

SUGGESTION

Several recommendations emerge from this research for future development and implementation. For future researchers, the trading robot application could be further developed by incorporating other technical indicators such as Stochastic, Parabolic SAR, or other indicators, or by implementing different methodologies such as Fibonacci or martingale strategies to evaluate their effectiveness in various market conditions. Additionally, future studies should not focus solely on the XAU/USD pair but also examine other currency pairs such as EUR/USD, EUR/GBP, GBP/USD, and others, enabling comparative analysis of results across different trading pairs and providing insights into the system's performance across various market instruments. For users, it is essential to maintain active supervision of the trading robot's operations, as the effectiveness of any trading system ultimately depends on how users utilize it. The trading robot serves merely as a tool that represents human traders in executing transactions, and continuous monitoring with appropriate intervention remains crucial for optimal trading outcomes.

REFERENCES

- Algodri, F., Lestari, S., & Wardani, N. E. (2015). TEKNOLOGI TRADING BERBASIS EXPERT ADVISOR (EA) PADA PASAR VALUTA ASING DENGAN TEKNIK KALENDER EKONOMI. *SEMNASTEKNOMEDIA ONLINE*, 3(1), 5-12-21. <https://ojs.amikom.ac.id/index.php/semnasteknomedia/article/view/817>
- Bjarnason, E., Lang, F., & Mjöberg, A. (2023). An empirically based model of software prototyping: a mapping study and a multi-case study. *Empirical Software Engineering*, 28(5), 115. <https://doi.org/10.1007/s10664-023-10331-w>
- Bois, E. Du, & Gerritsen, B. H. M. (2013). Demonstration of software concepts to multiple stakeholders using modular abstract prototyping. *CoDesign*, 9(3), 137-159. <https://doi.org/10.1080/15710882.2013.824485>

- Brooke, J. (1996). SUS -- a quick and dirty usability scale. In *Usability Evaluation in Industry* (pp. 189–194). Taylor & Francis. https://www.researchgate.net/publication/319394819_SUS_--_a_quick_and_dirty_usability_scale
- Calvez, A. le, & Cliff, D. (2018). Deep Learning can Replicate Adaptive Traders in a Limit-Order-Book Financial Market. *2018 IEEE Symposium Series on Computational Intelligence (SSCI)*, 1876–1883. <https://doi.org/10.1109/SSCI.2018.8628854>
- Capiluppi, A., Boldyreff, C., Beecher, K., & Adams, P. J. (2009). Quality Factors and Coding Standards – a Comparison Between Open Source Forges. *Electronic Notes in Theoretical Computer Science*, 233, 89–103. <https://doi.org/10.1016/j.entcs.2009.02.063>
- Dinata, H. (2018). *Implementasi Expert Advisor Dengan Algoritma Fibonacci Pada Analisa Teknikal Untuk Perdagangan Forex* [Universitas Kanjuruhan Malang]. <http://ejournal.unikama.ac.id/index.php/JoISTIC/>
- Elder, D. A. (2014). *The New Trading for a Living*. John Wiley & Sons, Inc.
- Garg, I. (2023). ALGORITHMIC TRADING: A COMPREHENSIVE REVIEW OF TECHNOLOGICAL ADVANCEMENTS AND MARKET IMPLICATIONS. *International Journal of Social Science & Economic Research*, 08(08), 2252–2247. <https://doi.org/10.46609/IJSSER.2023.v08i08.012>
- Gómez, V., & Figueroa, P. (2024). ProtoColVR: Requirements Gathering and Collaborative Rapid Prototyping of VR Training Simulators for Multidisciplinary Teams. *IEEE Transactions on Visualization and Computer Graphics*, 30(5), 2549–2558. <https://doi.org/10.1109/TVCG.2024.3372057>
- Hamzah, M. L., Ambiyar, A., Rizal, F., Simatupang, W., Irfan, D., & Refdinal, R. (2021). Development of Augmented Reality Application for Learning Computer Network Device. *International Journal of Interactive Mobile Technologies (IJIM)*, 15(12), 47. <https://doi.org/10.3991/ijim.v15i12.21993>
- Hamzah, N., Chuprat, S., Handayani, D. O. D., Xiaoxi, K., & Nagappan, S. D. (2021). Evaluating Quality Characteristics of Ubiquitous Application Through Means of Quality Models using Meta-metrics Approach. *Journal of Physics: Conference Series*, 2120(1), 012033. <https://doi.org/10.1088/1742-6596/2120/1/012033>
- Horobet, A., Boubaker, S., Belascu, L., Negreanu, C. C., & Dinca, Z. (2024). Technology-driven advancements: Mapping the landscape of algorithmic trading literature. *Technological Forecasting and Social Change*, 209, 123746. <https://doi.org/10.1016/j.techfore.2024.123746>
- Jamilah, H. M., & Mardiana, L. (2024). THE ROLE OF FINTECH AND FINANCIAL LITERACY ON MSME PERFORMANCE. *Journal of Entrepreneurship*, 3(2), 36–46. <https://doi.org/10.56943/joe.v3i2.560>
- Kadir, S., & Musdiana, A. D. (2024). BUSINESS RISK AND RETURN CONCEPT IN ISLAMIC INVESTMENT. *Journal of Entrepreneurship*, 1–12. <https://doi.org/10.56943/joe.v3i4.695>
- Khanesar, M. A., & Branson, D. (2022). Robust Sliding Mode Fuzzy Control of Industrial Robots Using an Extended Kalman Filter Inverse Kinematic Solver. *Energies*, 15(5), 1876. <https://doi.org/10.3390/en15051876>
- Lu, C. I. (2025). *Evaluation of Deep Reinforcement Learning Algorithms for*

- Portfolio Optimisation*. <http://arxiv.org/abs/2307.07694>
- Lubis, M. H. R., & Kusuma, I. G. K. C. B. A. (2022). Analisis Pertumbuhan Investor Ritel Pada Masa Pandemi Dan Implikasi Pajak Penghasilan Final Atas Penjualan Saham Di Bursa. *JURNAL PAJAK INDONESIA (Indonesian Tax Review)*, 6(2), 245–264. <https://doi.org/10.31092/jpi.v6i2.1854>
- Macpherson, W., Werner, A., & Mey, M. R. (2022). Industry 4.0: Emerging job categories and associated competencies in the automotive industry in South Africa. *SA Journal of Human Resource Management*, 20. <https://doi.org/10.4102/SAJHRM.V20I0.1916>
- Mcdowell, B. A. (2008). *A Trader's Money Management System: How to Ensure Profit and Avoid the Risk of Ruin*. John Wiley & Sons, Inc.
- Mukramin, M., & Hadi, A. (2022). RANCANGA BANGUN APLIKASI PERPUSTAKAAN SEKOLAH BERBASIS WEB. *KERNEL: Jurnal Riset Inovasi Bidang Informatika Dan Pendidikan Informatika*, 2(2), 69–76. <https://doi.org/10.31284/j.kernel.2021.v2i2.1893>
- Othman, B., & Harun, A. Bin. (2021). The Influence of Service Marketing Mix and Umrah Service Quality on Customer Satisfaction and Customer Loyalty towards Umrah Travel Agents in Malaysia. *Technium Social Sciences Journal*, 22, 553–618. <https://doi.org/10.47577/tssj.v22i1.4330>
- Putra, P., & Jibril, M. (2024). SISTEM INFORMASI PENJUALAN BAKSO DIWARUNG 3 D BERBASIS WEB. *TEKNOFILE: Jurnal Sistem Informasi*, 2(6), 474–484. <https://jurnal.nawansa.com/index.php/teknofile/article/view/256>
- Qureshi, S. M., Saeed, A., Ahmad, F., Khattak, A. R., Almotiri, S. H., Al Ghamdi, M. A., & Shah Rukh, M. (2025). Evaluating machine learning models for predictive accuracy in cryptocurrency price forecasting. *PeerJ Computer Science*, 11, e2626. <https://doi.org/10.7717/peerj-cs.2626>
- Rwiza, S., Kissaka, M., & Kapis, K. (2020). A Metric for Evaluating Security Models based on Implementation of Public Key Infrastructure. *International Journal of Wireless and Microwave Technologies*, 10(6), 27–35. <https://doi.org/10.5815/ijwmt.2020.06.04>
- Sevastjanov, P., Kaczmarek, K., & Rutkowski, L. (2024). A multi-model approach to the development of algorithmic trading systems for the Forex market. *Expert Systems with Applications*, 236, 121310. <https://doi.org/10.1016/j.eswa.2023.121310>
- Shavandi, A., & Khedmati, M. (2022). A multi-agent deep reinforcement learning framework for algorithmic trading in financial markets. *Expert Systems with Applications*, 208, 118124. <https://doi.org/10.1016/j.eswa.2022.118124>
- Susanto, A., & Meiryani. (2019). System Development Method with The Prototype Method. *International Journal of Scientific & Technology Research*.
- Syaputra, R., & Voutama, A. (2021). Perancangan Sistem Expert Advisor Untuk Melakukan Perdagangan Emas Otomatis Pada Software Metatrader 4. *INTECOMS: Journal of Information Technology and Computer Science*, 4(2). <https://journal.ipm2kpe.or.id/index.php/INTECOM/article/view/2773>
- Todorovic, N., Manojlovic, I., & Budovic, A. (2017). Measuring tourist satisfaction and destination image with HOLSAT. *Glasnik Srpskog Geografskog Drustva*, 97(2), 87–118. <https://doi.org/10.2298/GSGD1702087T>
- Wang, L., Wang, M.-C., Zhang, Y.-R., Ma, J., Shao, H.-Y., & Chang, Z.-X. (2025).

- Automated Identification and Representation of System Requirements Based on Large Language Models and Knowledge Graphs. *Applied Sciences*, 15(7), 3502. <https://doi.org/10.3390/app15073502>
- Wu, M.-E., Syu, J.-H., & Chen, C.-M. (2022). Kelly-Based Options Trading Strategies on Settlement Date via Supervised Learning Algorithms. *Computational Economics*, 59(4), 1627–1644. <https://doi.org/10.1007/s10614-021-10226-2>
- Yhurinda, A., Putri, P., & Subriadi, A. P. (2019). Software Cost Estimation Using Function Point Analysis. *IPTEK Journal of Proceedings Series*, 79–83. <https://doi.org/10.12962/j23546026.y2019i1.5115>
- Young, A. R. (2013). *Expert Advisor Programming For Meta Trader 5 : Creating automated trading systems in the MQL5 language*. Edgehill Publishing. <https://archive.org/details/ExpertAdvisorProgrammingForMetaTrader5>
- Yulianti, Y. (2019). *MONEY MANAGEMENT SEBAGAI ALTERNATIF UNTUK MENINGKATKAN PROFIT TRADER PADA PT. VICTORY INTERNATIONAL FUTURES MALANG*. Universitas Muhammadiyah Malang.