Journal of Behavioral and Applied Management
2026, Vol. 26, No. 2, 91–99
https://jbam.scholasticahq.com/

ARTICLE

Improving Product Testing with the Use of Artificial Intelligence for a Card Game Company

Stephen Penn
University of North Texas

Deepeka Gurunathan
University of North Texas

Vaishnavi Govind Bhamare
University of North Texas

Small game companies must test new products carefully, particularly when releasing expansions to existing games. As the product line grows, testing demand outpaces what small teams can absorb through hiring, creating a scaling problem. AI augmentation is a potential solution, but its effects on designer and tester behavior in small game companies are not well understood. This paper investigates how AI augmentation in card game testing reshapes designer and tester behavior, using an exploratory case study (Yin, 2018) of an existing company. Python code was developed to simulate matches between AI-controlled decks, and the resulting data were presented through two dashboards to one of the company's owners. Qualitative analysis of the meeting transcript surfaced three propositions describing what designers and testers need from AI-augmented testing, alongside a three-step Validation-Interpretation-Internalization (V-I-I) cognitive cycle through which the dashboard user comes to trust, understand, and incorporate the information. This V-I-I cycle is itself the behavioral change AI introduction produces, and the resulting framework offers small specialty firms a planning tool for phased AI adoption. The findings contribute to the limited academic literature on AI in product testing for small firms.

The global game industry generates multibillion-dollar revenues annually. The tabletop segment that uses cards as its principal mechanic was valued at six billion USD in 2022 and is projected to grow at eight percent per year through 2028 (Cognitive Market Research, 2025), and the broader trading card industry is projected to exceed $21 billion by 2034 (Custom Market Insights, 2025). Card games are sold either as complete, ready-to-play decks or as small packs of randomized cards, each card carrying its own name, artwork, rules, and power values. Because a larger personal library affords more strategic options in play, competitive players continually purchase newly released cards to expand their collections (Ham, 2010).

To meet this demand, designing and testing new cards is a critical operation for any card game company. Within a given year, a company may release several card sets, or expansions, which serve as the primary engine of revenue. Each new set is also an opportunity to refresh the strategic landscape, since unique card capabilities and their interactions allow players to devise novel winning strategies (Bradley, 2025; Ham, 2010). Designing, testing, and approving every card in every expansion is therefore essential (Varaksina & Dyshuk, 2025), and for a small game company the workload can be daunting.

Because players have a strong incentive to purchase newly released cards, companies are obligated to maintain high quality across all aspects of game play (Rosewater, 2011). Product development for these companies typically unfolds across early and later phases: in the early phases, designers establish the theme, artwork, setting, and core game mechanics; in the later phases, individual cards are refined for clarity, balance, and power level. The latter phase is iterative and time-consuming (Miller & Sherwood, 2024), and developers may play through a set many times before release, adjusting cards between plays (Edwards, 2012).

In practice, designing and testing cards requires developers to play the game with newly designed expansion cards as often as the production schedule allows, iterating between play, modification, and re-testing (Miller & Sherwood, 2024). Because expansion sets contain many cards and each card may be modified during testing, ensuring overall quality is laborious (Ham, 2010). Methods for conducting comprehensive testing at lower cost are therefore central to a small company's long-term viability and to its ability to meet, or exceed, customer expectations.

The production cycle of a typical card game, as described in trade publications and academic sources (Edwards, 2012; Jonsson & Tonegran, 2013), can be partitioned into two phases for the purposes of this paper: a design phase and a testing phase. In the design phase, theme, storyline, mechanics (the special rules unique to the game), card text, and artwork are developed. In the testing phase, testers cycle between playing the game and revising card text, mechanics, and numeric values (Jaffe et al., 2012). Because a single card can interact with and modify the effect of another, each change requires re-

Article citation: Penn, S., Gurunathan, D., & Bhamare, V. G. (2026). Improving product testing with the use of artificial intelligence for a card game company. *Journal of Behavioral and Applied Management, 26*(2), 91–99.

testing the affected interactions. The iterative nature of this work is sufficiently demanding that companies frequently recruit volunteers from their existing customer base to assist (Bradley, 2025).

An improved testing process could speed up the process for the tester, which would reduce overhead expenses and reduce time to publication, but could also help the tester understand why a deck is either winning or losing. An effective testing process can improve efficiency and accuracy in producing a high-quality product (Miller & Sherwood, 2024). Capturing the results of many games between two opponents creates an opportunity to present statistics to the tester. The tester would then gain insight into how the deck behaves in typical games. In a testing process that uses AI, the tester assumes that 1) the computer code representing the rules of the game are accurate, 2) the models are making relatively accurate and competent choices representing skilled human players, and 3) the statistics being captured accurately reflect what happened in the game. This research includes validation steps for these three assumptions prior to presenting the dashboards to a Varia designer and tester. The research question asked here is, How does AI augmentation in card game testing change the behavior of the designers and testers?

How to Play a Collectible Card Game

Deck sizes vary by game, ranging from 30 to 100 cards, with most games using decks of around 50. Companies sell cards both as ready-to-play decks and as randomized packs of 10 to 15 cards (Jonsson & Tonegran, 2013), and they often facilitate community play through partnerships with local game stores.

At these matches, players take turns making both tactical and strategic decisions until one side meets a victory condition; most card games simulate combat, with a player winning once the opponent's health reaches zero. Each player draws only from the deck they have brought to the table, never from the opponent's. A typical match begins with each player drawing a starting hand from their own deck—usually around six cards—and skilled players will know their own deck thoroughly and begin to infer the contents of the opponent's deck within the first few turns (Maisenhölder, 2018).

This research uses a commercially available card game, Varia, developed by Guildhouse Games (GHG). Unlike traditional trading card games, GHG sells only complete, ready-to-play decks rather than randomized packs intended for trading; players who acquire two decks may begin play immediately, though the game also supports custom deck-building rules for those who wish to construct their own (Guildhouse Games, 2025).

In a match of Varia, players alternate roles each turn between active player and reactive player. The active player initiates a turn by placing one to five cards face up on the table; this set of cards is called a play. The reactive player must respond by playing one card against each of the active player's cards, and each resulting pair is called a Moment. After both players have placed and, where the rules permit, adjusted their cards, the outcome of the turn is determined by comparing the values and special conditions on the cards within each Moment, with the players' health typically reduced as a result. The match ends when one player's health reaches zero, at which point the opposing player is declared the winner (Guildhouse Games, 2021).

GHG currently uses traditional card game testing in its design and development process, working at the level of complete decks rather than individual cards (A. Tessitore, personal communication, 2025). To remain consistent with the company's testing practice, this study also operates at the deck level. Three published decks were used: the Volcanic Warrior, the Shadow Assassin, and the Divine Paladin, with the Volcanic Warrior serving as the primary subject for power analysis against the other two. All matches in this study were played AI against AI, simulating the proposed testing process under controlled conditions (Nelson & Mateas, 2009).

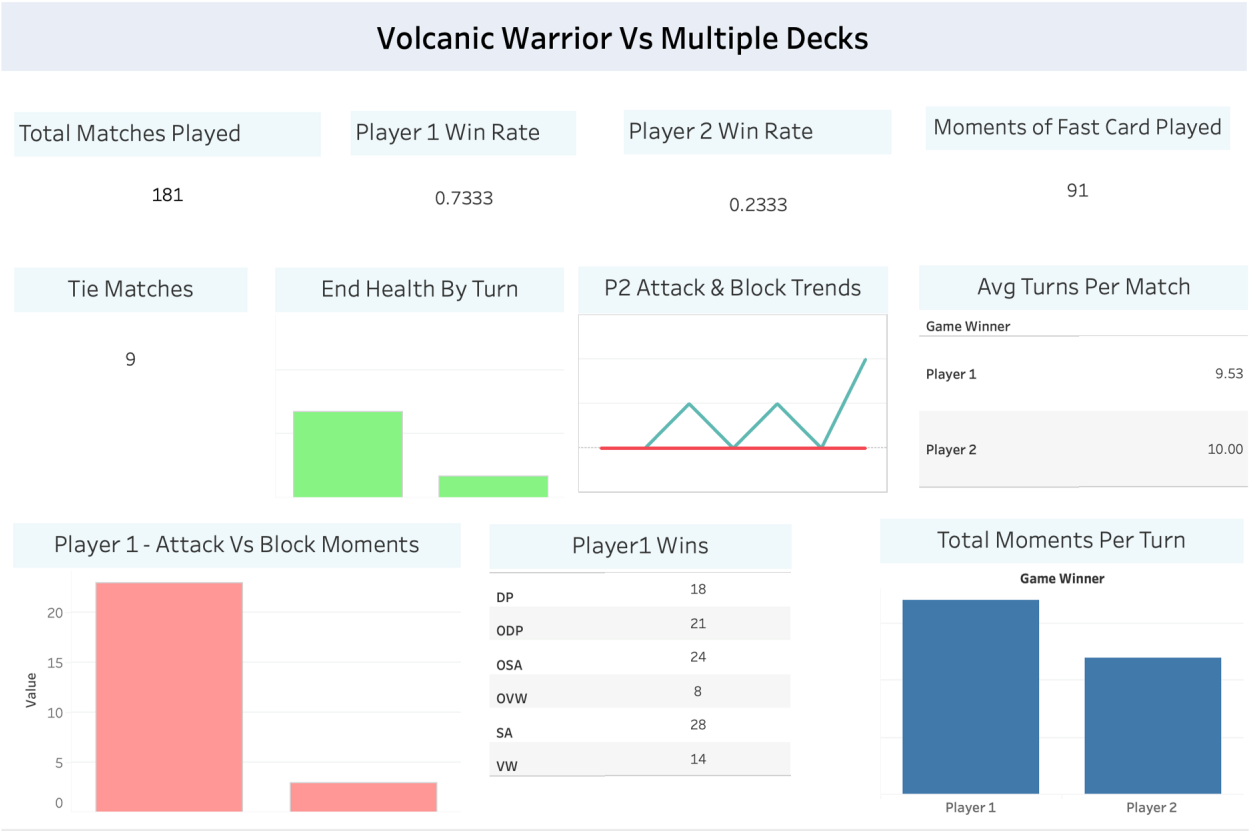
The research was organized into four phases. Phase I proposed a new testing process incorporating AI for the benefit of GHG. Phase II implemented and validated the supporting infrastructure: writing and testing the Python code that executes the game's rules, generating the data sets needed to build predictive models, and verifying the accuracy of code, data, and models. Phase III used this validated infrastructure to run matches between the selected decks, simulating the proposed testing process. This phase also included taking the statistics from the AI-played games, building dashboards, and presenting those dashboards to one of the owners of GHG. Phase IV started with a meeting with one of the owners. The feedback gathered from this presentation meeting was then analyzed in a qualitative manner to develop propositions that will be further investigated in future work.

Phase I: Assisting GHG in Card Game Testing with AI

In testing new expansions, Varia testers need to ensure that the deck in question is not winning too often or too seldom. Winning too often indicates the deck is too powerful and customers will lose interest in playing against such a deck. Winning too seldom indicates the deck is underpowered and players will not want to purchase the deck (Jaffe et al., 2012). A card game tester needs to first detect the power level of a deck when playing against other previously published decks and collect information on potential reasons why the deck is too over- or underpowered (Bradley, 2025; Varaksina & Dyshuk, 2025). With that information, a tester can then decide how to proceed in modifying the cards and eventually resume testing. With the importance of testing for a company to produce high-quality products, the iterative nature of testing can lead to delays and high expenses. Therefore, an improved testing process using AI could assist the card game designers and testers in meeting the production schedule and, most importantly, customer expectations (Jaffe, 2013; Nelson & Mateas 2009). See Figure 1 for the existing testing process.

Figure 1

Dashboard for a Single Deck Against Several Decks



The proposed testing process includes one new step, which is playing several matches between two bots (AI models) before other testing is performed. See Figure 2 for the proposed testing process. This new step would use the deck under development against existing known decks, such as decks previously published. Testing against previously published decks would give insight into the power level of the deck compared to decks known by current customers. Testing could also include using decks that have never been published but are known to be underpowered or overpowered. Testing against multiple decks with a range of power would provide more insight and validation of the power level of the tested deck. The person reading the statistical summary would need to understand why opposing decks were chosen to successfully interpret the information presented on the dashboards.

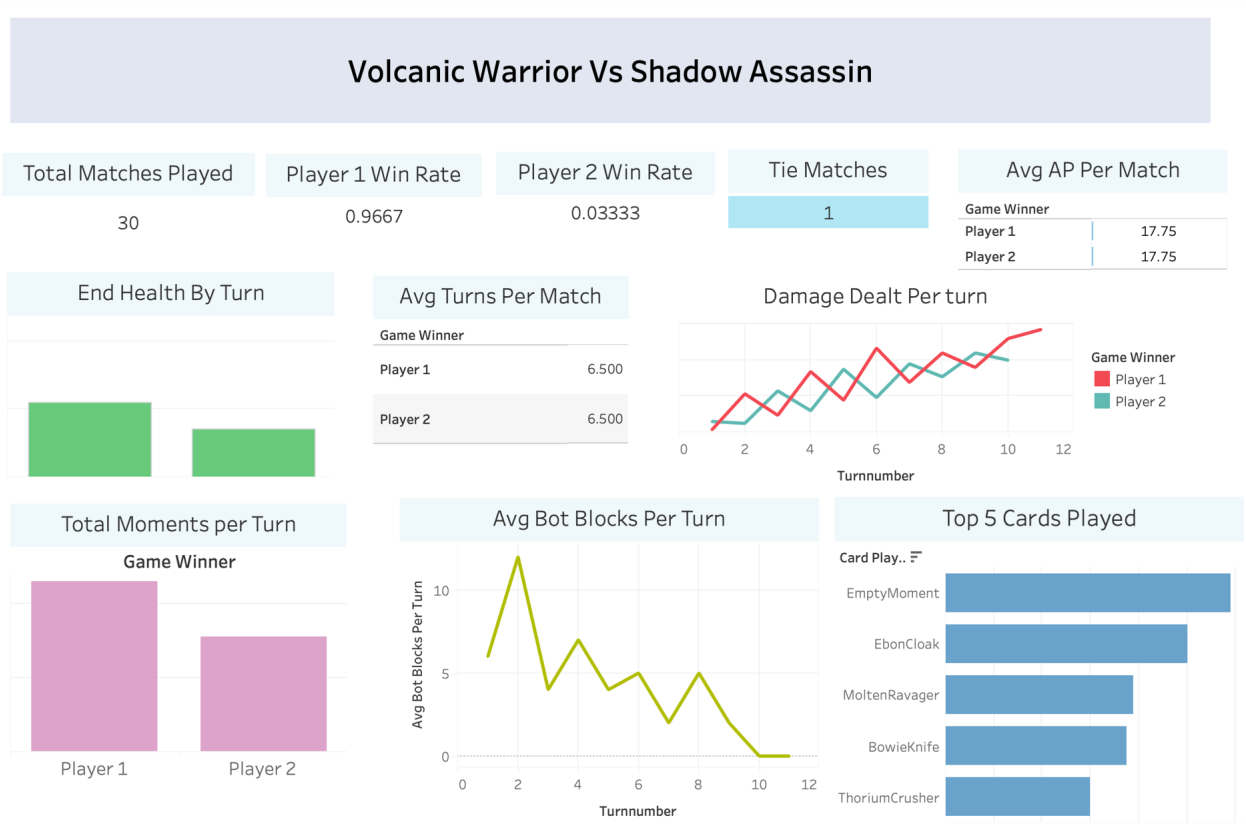
Phase II: Validating the Data, Code, and Models

Validating the code included code walk-throughs and tests of selected cards. Code walk-throughs consisted of explaining the Python code to other researchers on the team, unit testing, and functional testing. Validation also consisted of monitoring which cards were in both players' hands and considering the play chosen by the AI model. The card tests consisted of a human, a member of the research team, playing against the AI model. In playing these games, the game code shuffled the cards, dealt the hands, allowed for reactive plays, determined the interaction of cards for each Moment, calculated damage, and determined the winner. During validation of the model, the models were rebuilt numerous times with different variables, different amounts of data, and different dependent variables. Eventually, one model became a challenge to beat and was deemed validated.

A large data set was needed to build the model. However, the models were needed to run a match, which created a conundrum in that the models could not be built until a very high number of matches were completed. Without a validated model, playing the game and recording turn outcomes was impossible. The initial data sets used to build the models were restricted to only the results of the first turn of a match. The researchers decided to create a list of all potential plays for each of the three decks. These lists of all potential plays, about 25,000 to 30,000 plays per deck, became the list of plays to execute for the active player. The plan was to run each of these plays against randomly selected plays from an opposing deck. Thus, no model was needed to select the active player's play; all plays were eventually used via a loop. A Python routine looped through each of these active player plays and se-

Figure 2

Dashboard for a Single Deck Against a Single Deck



lected a random play from the reactive player's deck that matched the number of Moments from the opposing deck. Once the two plays were selected, the turn was executed and the results recorded. One hundred opposing plays were chosen at random for each of the active player's plays. This process created roughly 2,500,000 or more rows of input data per unique combinations of decks. Three unique decks, Volcanic Warrior, Shadow Assassin, and Divine Paladin, gave six combinations. Each deck could be played against itself, called a mirror match, or against one of the other two decks. This list of combinations of decks played against each other created roughly a couple of million rows of output data. Eventually, over 10,000,000 rows of input data were used for fitting the models.

The model was written in Python with the packages Tensorflow and Keras using Jupyter Lab. The Sequential model class was chosen for its simplicity. It had five layers with array lengths of 53, 26, 13, 6, and 1 respectively. The last layer was the predicted value of damage delivered to the opposing hero. The activation function used Rectified Linear Unit (RELU). Relu removes negative values, which should reflect the fact that damage inflicted would always be zero or greater. The optimizer was Root Mean Squared Propagation (RMSPROP). The loss function was the Mean Squared Error (MSE) and the metrics were Mean Absolute Error (MAE; Chollet, 2021).

Validating the model included comparing the planned cards with a human's choice of cards. The researchers verified that the plays matched or came close to the what the human would choose given the same hand of cards. When the planned cards did not match, model accuracy was consistently the culprit. Accuracy was determined by comparing the predicted amount of damage delivered from a chosen play to actual damage inflicted. Multiple models were developed using different independent variables, hyperparameters, and turn results. Eventually one model was chosen that provided both accuracy of prediction and consistently effective choices in cards played. The chosen model used one-hot-key with each column being a card. Different combinations of variables were tried, but having one column for each unique card in all three decks optimized the predictive models sufficiently. The models predicted the amount of damage done for the turn. Using only damage as the predicted value meant that the AI model used an aggressive, damage-oriented strategy.

Validating the statistics included conversations with the owner of the game company prior to development of the dashboards. These conversations verified that the results

were typical of games played and even the effects of the game overall. For example, the collected data of playing close to 100 games showed that the active player, which alternates each turn, scored significantly higher damage than the reactive player.

Discussing this finding of the difference in damage inflicted with the owner verified that that effect was intended in the game. The active player has little advantage in showing the turn's planned cards first. However, the active player also chooses the number of cards to play, which defines the number of Moments in the turn. Since the reactive player cannot change the number Moments, the reactive player has a smaller, more restricted set of choices in choosing how to respond, thus leading to suboptimal plays. These suboptimal plays lead to reducing damage threatened and inflicted. These game outcomes confirmed that the rules and the models used were leading to typical and expected game outcomes (A. Tessitore, personal communication, 2025).

To demonstrate the difference between an overpowered deck and a balanced deck against published decks, a mechanism was developed to force a published deck to be overpowered. Following discussions with the GHG owner, the overpowered decks used in this study were published decks with adjusted Action Point costs. Every card in Varia has an Action Point cost ranging from one to nine. Most cards cost three or four action points. One restraint in making a play in Varia is that the sum of the play's Action Point costs cannot exceed ten. In other words, every turn each player has ten Action Points to spend on chosen cards for a play. This restriction reduces the options a player has in choosing which and how many cards to play. To create an overpowered deck all Action Point costs for the cards in the deck were lowered by one. This adjustment gives the deck an advantage in creating more options in legal plays every turn.

To validate the code, the models, and the output, a hypothesis was defined and tested. A deck with lower Action Point cost cards will have a proportion of wins greater than 50% when played against decks with unadjusted Action Point costs. The proportion of wins is typically called the win rate. Confidence Intervals and Z tests of proportions were performed with the null hypothesis stating that the win rate should be 50%. The published baseline deck achieved a win rate of 54.4% (sample size = 68; 95% confidence interval: 0.42–0.66), which was not significantly different from the expected 50% win rate ($Z = 0.73$, $p > .2$). Thus, a win rate of 54% could be due to chance and we fail to reject the null hypothesis. In contrast, the overpowered deck produced a win rate of 70.8% (sample size = 96; 95% confidence interval: 0.61–0.81), showing a clear and statistically significant advantage ($Z = 4.08$, $p < 0.05$). Thus, the null hypothesis is rejected for the overpowered deck. These quantitative results confirm that randomness alone does not explain the higher win rate. When the only known change between the decks is the Action Point costs, we conclude that new Action Point costs directly caused the imbalance. These results validate that the AI testing framework is sensitive enough to detect rule-driven imbalances and can reliably flag overpowered decks for further human review.

Once the code, models, data, and overpowered decks were validated, everything was ready for playing games. We chose to use one model for both players to ensure that the skill levels of the AI players were equal.

Phase III: Research Design and Meeting Set Up

Exploratory Case Study Approach

The intention of this exploratory case study (Yin, 2018) was to inductively generate theory about the cognitive work involved in AI-augmented testing for small specialty firms. The Volcanic Warrior was played against itself, the Shadow Assassin, and the Divine Paladin, both with and without modifications. Modified decks lowered each card's Action Point cost by one point, which enabled more cards per turn, giving the modified deck an advantage and producing the contrast needed to evaluate the framework's sensitivity to power-level differences. This change was suggested by the designer of the game for testing the Python code and the ability to detect overpowered decks. In total, there were six unique decks: Volcanic Warrior, Shadow Assassin, Divine Paladin, overpowered Volcanic Warrior, overpowered Shadow Assassin, and overpowered Divine Paladin. Thus, the Volcanic Warrior deck was played against each of the other six decks roughly thirty matches each for a total of 181 matches. Data was recorded at the end of each turn for each of these matches where each match lasted between three to twelve game-turns.

Dashboard Development and Meeting Protocol

GHG continued using their human-vs-human games for playtesting during this research, but added AI as an additional step (Jaffe et al., 2012), which comprising two substeps: collecting data from matches and then displaying that data on dashboards. The addition of AI only involved one person from GHG to review the statistics. Since the designer has a game effect in mind when designing each card, the intention was to compare the collected statistics, displayed graphically on dashboards, to expected match results to develop ideas for modifying and improving the cards (Nelson, 2011). AI models used in this research simulated a low to mid-skilled player because there is simply too much for this first generation of AI models for Varia to do in considering the number of options, future turns, randomness, and opponent's potential plays (de Mesentier Silva et al., 2017).

In an online meeting, the dashboards containing the collected statistics were presented to the owner of GHG, who is also a designer and tester of their products The three researchers presented the dashboards and recorded the owner's response. The response included initial impression, expected values in the chart, interpretation of the values, and why such a chart may or may not be useful for a tester. All the information was then compared to the expected results to evaluate the three hypotheses. See Figures 1 and 2 for the dashboards that were presented.

This approach had clear limits being a single meeting with one person from the company. Even so, the qualitative method is sufficient for exploratory research aimed at understanding how tester and designer behavior changes when these roles gain access to data generated from many AI-played games.

Phase IV: Qualitative Analysis Findings

Three propositions emerged from qualitative analysis of the meeting transcript, categorized by the perspectives the GHG owner adopted as he spoke. At various points he spoke as a designer, raising what a designer would want from AI augmentation, for example, comparing the strategy the AI employed against the strategy he had intended when designing the cards. At other points he spoke as a tester, raising concerns about card power and deck balance.

Three Emergent Propositions

The first proposition reflects concerns from the designer. About 45 minutes into the conversation the owner asked, "Hurdle number one … is can you prove to me the designer that you can teach your model to play with my game pieces?" He also added, "The way I think it should." In other words, the owner is asking if the AI can implement a strategy at playing a specific deck that would be the way the designer intended during the design phase. Earlier in the conversation, the company owner asked, "How many times does it win playing the way that I have intended? Maybe 60%, right? But then directly next to that, show me like, how many times does it win when it just plays aggro? So not the way I intended." The GHG owner was specifically asking for a grid of deck and strategy combinations showing win rates. For example, if the Volcanic Warrior used three different strategies, say aggro, mid-range, and defense, against the Shadow Assassin that used 2 different strategies, say stealth and defense, then a 3x2 matrix would show the Volcanic Warrior's win rates for each combination. This matrix would enable the designer to see interactions of opposing strategies that might become apparent depending on the cards in each deck. One of the included strategies should always be the intended strategy. The result is a proposition that a variety of strategies helps the designer.

> Proposition 1. Designers need behavioral variety across strategies; current single-strategy AI doesn't provide it; multi-strategy AI is a candidate next step.
>
> Drawback/Problem. Developing and implementing the strategies either in code or with AI takes time.
>
> Solution. Develop more rule-based strategies founded on expert advice and use Monte Carlo Tree Search for the AI to detect new strategies over many games.
>
> Affected Role. Designers

The second proposition builds on the first proposition but gets into more details. As the owner was describing strategies, he also mentioned variables. The owner stated, "…the biggest volcanic warrior metric that I would care about would be my damage per moment." Instead of stating more strategies are needed, the second proposition says more variables are needed that are specific to each employed strategy. He went on to state, "And then conversely, like the damage per moment of my opponent as the volcanic warrior would be the second thing that I care about because all I want as the warrior is for my DPM to be higher than yours." In this quote, DPM stands for Damage per Moment. For example, for an aggressive strategy, called aggro by players, a variable that records damage threatened is worthwhile. Another could be damage inflicted, with both recorded per Moment and per turn; a third could capture the differential between damage inflicted by the two opposing heroes. For an aggro deck, like the Volcanic Warrior, the first two variables are important, but the most important is the third. The Volcanic Warrior is willing to trade blows if the Volcanic Warrior is inflicting more damage than the opponent, which is the Volcanic Warrior's strategy. Both the tester and designer are interested in variables that are formulated based on the strategy being implemented. Other examples could include number of times blocked divided by the number of blocks in the deck for defensive decks. Thus, the second proposition is about the coverage of the chosen variables displayed in the dashboard.

> Proposition 2. Small firms need behavioral coverage their tester budgets can't fund; broad-coverage AI dashboards address this.
>
> Drawback/Problem. Initial cost of AI could still be steep, such as developing and testing the game code.
>
> Solution. Create specific variables for each deck and strategy combination.
>
> Affected Role. Designers and Testers

The third proposition addresses a play tester's need to verify current understanding about how a deck operates; how it wins and loses. About halfway into the meeting, the owner commented on "getting data-backed power ratings" and stated he could see the value in AI augmented testing. Play testers, more than designers, focus on "… things that are completely broken…what doesn't work at all…what breaks the game engine…weird infinite loop that is not intended." The text on cards can have unintended effects when interacting with text on other cards. Testers want to know when these failures happen – cards and decks need to be tested not just for design and balance issues, but for failure rates. The owner gave an example, "…The 4th Blade every turn is trying to figure … the right … sequenced combo to do within that turn, but the whole time [The 4th Blade is] also…setting up [his] end game combo." Therefore, play testers need statistics from variables that address sequence of cards played within a moment, within a turn, and across turns, as well as the synergies that emerge between card interactions. Some cards, like Jeering Ploy, have limited effectiveness early in a game, such as game turns 1, 2, and 3, and have more potential in later turns. Calculating the power rating of a card across the whole game and in the early, middle, and late game turns is important to fully understand the power of any particular card. The owner offered another exam-

ple, "…Jeering Ploy is a defensive card that lets me disengage, but then it immediately forces you to engage me, right?... it forces you to move, which if I have the ship, that's 3 1/2 damage…" When a card succeeds and fails and what is the difference in effect between success and failure gives the tester understanding of how the cards and deck will feel in a customer's hands. Therefore, the last proposition addresses a tester's needs to verify prior notions about a deck. The constraint is that every time a card goes through redesign, no matter how small a change is made, it must go through full testing again. The dashboards used by the tester need these card specific variables.

Proposition 3. Testers need to verify prior notions about deck behavior; deck-specific AI dashboards address this.

Drawback/Problem. Every time a new deck, strategy, collected variable, or dashboard is introduced, the tester will spend time and effort validating the game code, the AI implementation, and the variables on the dashboard.

Solution. Need deck specific and strategy specific dashboards.

Affected Role. Testers

Accepting and Using Statistics from AI-run Games: The V-I-I Process

The participating GHG owner gave advice from multiple perspectives, namely the tester and the designer. At one point in the meeting when seeing the second dashboard, he stated, "that one jumped out at me as, I don't, I honestly don't believe that data. I would be shocked to know that was actually true." This reaction signaled that the dashboard required further validation work before it could be trusted. He pressed on whether the statistics were accurate, under what conditions the data were generated, how the statistics were collected, and how they were calculated. Once these questions were answered, he began connecting the numbers to his understanding of the cards and his customers. About halfway through the meeting the owner discussed one deck that was published. "When we released it into the wild, like nobody understood how it worked. It's … like you have to play this card with this card and create these very specific scenarios to get [it] to work. And like just everyone kept losing and saying that it wasn't good, right? Even though that's not true. But as a designer, that data would be valuable to me…" He began to see how the numbers came from many games involving the chosen decks.

At the point at which he was connecting his interpretation of the dashboard to his understanding of the game, he began to explain how the dashboards could be used, how they could be improved, and how the AI augmented process could benefit GHG. The researchers conducted qualitative analysis on these quotes and reduced these instances to nine questions related to the propositions and each stage of understanding. These nine questions represent what the roles of designer and tester need at each stage of understanding. The researchers labeled these three stages as Validation, Interpretation, and Internalization, V-I-I.

Validation Work is the person getting to the point of trusting the source data and the dashboard's information. Interpretation Work is the person making meaning from the trusted information. Internalization Work is when the user's working knowledge changes. The designer and the tester each have a set of questions to answer as they work through the V-I-I cycle. The three propositions, described above, and three steps in V-I-I produce nine questions, some for the designer and some for the tester. AI testing creates new cognitive work for the two roles — designer and tester — who each must traverse a V-I-I cycle to gain benefits from the AI-augmented testing process. This cycle is itself the behavioral change that AI introduction produces. See Table for the nine questions.

Theoretical Positioning

V-I-I resembles Nonaka's Socialization-Externalization-Combination-Internalization (SECI) framework (Farnese et al., 2019; Nonaka, Toyama, & Konno, 2000). SECI de-

Table

Nine Questions

Proposition	Validation	Interpretation	Internalization
P1. Variety	Is the AI actually playing the multiple strategies as designed, rather than collapsing toward one strategy?	What does the variety reveal about how the deck performs under different playing styles? Where does it succeed and where does it break?	How does seeing the deck's behavior across multiple strategies change my mental model of what the deck is and how it should be tuned?
P2. Coverage	Is the AI actually using the deck-specific strategies?	Are some decks regulating win rates as intended to stop runaway decks?	How does this comprehensive coverage picture update my understanding of which decks need rebalancing?
P3. Prior Notions	Are the collected statistics from deck+strategy games accurate?	Are the results what I expected and, if not, are existing statistics enough to figure out why?	What needs to be updated? Car values? Card text? Ratio of car in a deck? AI strategy? The designer's and tester's understanding of the game?

scribes organizational knowledge conversion in general and V-I-I describes the specific cognitive work small-firm practitioners must do to integrate AI-generated explicit data into their tacit professional knowledge. V-I-I therefore offers a focused articulation of the explicit-to-tacit phase of SECI in AI-augmented decision contexts. Empirical operationalization and testing of the framework are reserved for subsequent research.

Conclusion

Theoretical Contributions—V-I-I as an Emergent Framework

The V-I-I cycle describes a cognitive process that aligns partially with Nonaka's theory of tacit and explicit knowledge conversion, but no prior work has named the validation-interpretation-internalization sequence as a distinct pattern in AI-augmented testing contexts. Articulating this three-step cycle is a modest but novel contribution that emerges from close case-study documentation of how a designer-tester-owner, a role configuration typical of small specialty firms, engages with AI-generated dashboards. Such grounded observations are uncommon in the broader AI-adoption literature, which has tended to focus on large organizations and theoretical frameworks.

Practical Contributions—Cost of AI Augmentation in Terms of Work Hours

AI testing creates new cognitive work for both roles, designer and tester, who each must traverse a V-I-I cycle to gain benefits from the AI-augmented testing process. V-I-I offers a practitioner-oriented simplification of how people come to trust, understand, and absorb what a dashboard shows. It builds on Munzner's (2014) nested model of visualization design and validation by collapsing complex, multi-level design and evaluation concerns into three observable user states: first validating the numbers and views, then interpreting the patterns and relationships, and finally internalizing the resulting insights into their own mental models and decisions. The V-I-I cycle also recurs each time the game code, AI strategies, or dashboards change, which means small firms must plan for repeated rounds of cognitive work rather than a single learning curve. The V-I-I table, organized by proposition, gives small specialty firms a planning tool for phased AI adoption. Taken together, these findings push against the common assumption that AI adoption reduces cognitive work; in this case, AI testing redirected the work rather than reducing it.

Limitations

This work rests on a single case exploring the use of AI in product testing for one small game company, with propositions developed from feedback gathered during a single meeting with one participant. The propositions are therefore intended to guide subsequent research that will operationalize and test formally stated hypotheses, rather than to serve as confirmed claims in their own right.

Future Work

Three lines of future research follow from this work. First, the V-I-I process warrants formal stakeholder analysis. While in small specialty firms the designer, tester, and owner roles often consolidate into one person — a structural observation that likely travels beyond card games — separating these roles analytically would enable deeper investigation of how each contributes distinct cognitive work, with the resulting implications rolled back up for single-person small firms. Second, the AI itself requires further development: a more elaborate method for AI decision-making per card, per Moment, and per turn (for which Monte Carlo Tree Search is a promising candidate), along with additional decks added to the code base to expand the volume of game data and the variables tied to strategy-specific dashboards. Third, the propositions developed here should be operationalized as testable hypotheses and examined across multiple small specialty firms to establish whether the V-I-I pattern generalizes beyond a single case.

References

Bradley, W. (2025). *Creation of a trading card game through the exploration of emergent gameplay*. University of Maine. https://digitalcommons.library.umaine.edu/honors/903/

Chollet, F. (2021). *Deep learning with Python* (2nd ed.). Manning Publications.

Cognitive Market Research. (2025). *Board game and card game market report*. https://www.cognitivemarketresearch.com/board-game-and-card-game-market-report

Custom Market Insights. (2025, August 13). *Global trading card games market size/share worth USD 21.05 billion by 2034*. Gale. https://link.gale.com/apps/doc/A851497157/GBIB

de Mesentier Silva, F., Lee, S., Togelius, J., & Nealen, A. (2017). *AI-based playtesting of contemporary board games.* Proceedings of the 12th International Conference on the Foundations of Digital Games, Article 13. pp. 1–10.

Edwards, R. (2012). *The game production pipeline: Concept to completion.* IGN. http://www.ign.com/articles/2006/03/16/the-game-production-pipeline-concept-to-completion

Farnese, M. L., Barbieri, B., Chirumbolo, A., & Patriotta, G. (2019). Managing knowledge in organizations: A Nonaka's SECI model operationalization. *Frontiers in Psychology, 10*, Article 2730. https://doi.org/10.3389/fpsyg.2019.02730

Guildhouse Games. (2021). *Varia rulebook*. https://www.playvaria.com/_files/ugd/87f56f_043d39c4141b4e5fb683fd651000ec79.pdf

Guildhouse Games. (2025). *Varia: A brand new tabletop XCG experience*. https://www.playvaria.com/

Ham, E. (2010). Rarity and power: Balance in collectible object games. *The International Journal of Computer Game Research, 10*(1). https://gamestudies.org/1001/articles/ham

Jaffe, A. B. (2013). *Understanding game balance with quantitative methods* [Unpublished doctoral dissertation]. University

of Washington. https://digital.lib.washington.edu/researchworks/handle/1773/22797

Jaffe, A., Miller, A., Andersen, E., Liu, Y., Karlin, A., & Popović, Z. (2012). *Evaluating competitive game balance with restricted play*. Proceedings of the 8th AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment, 8(1), 26–31.

Jonsson, J., & Tonegran, L. (2013). *Designing cards as a polymorphic resource for online free-to-play trading card games*. https://www.diva-portal.org/smash/get/diva2:629418/FULLTEXT01.pdf

Maisenhölder, P. (2018). Why should I play to win if I can pay to win: Economic inequality and its influence on the experience of non-digital games. *Well Played, 7*(1), 60–83.

Miller, N., & Sherwood, M. (2024). *How to make a card game: A comprehensive guide from concept to launch*. Play Party Game. https://playpartygame.com/card-games/how-to-make-a-card-game/

Munzner, T. (2014). *Visualization analysis and design*. CRC Press.

Nelson, M. (2011). *Game metrics without players: Strategies for understanding game artifacts*. Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment, 7(3), 19–24.

Nelson, M. J., & Mateas, M. (2009). *A requirements analysis for videogame design support tools*. Proceedings of the 4th International Conference on Foundations of Digital Games, pp. 137–144.

Nonaka, I., Toyama, R., & Konno, N. (2000). SECI, Ba and leadership: A unified model of dynamic knowledge creation. *Long Range Planning, 33*(1), 5-34.

Rosewater, M. (2011). *Ten things every game needs*. Wizards. http://magic.wizards.com/en/articles/archive/making-magic/ten-things-every-game-needs-part-1-part-2-2011-12-19

Varaksina, S., & Dyshuk, I. (2025). *Game development trends 2025: What's shaping the future of gaming*. The Mind Studios. https://games.themindstudios.com/post/game-development-trends/

Yin, R. K. (2018). *Case study research and applications: Design and methods* (6th ed.). Sage.

Stephen Penn (stephen.penn@unt.edu)

Deepeka Gurunathan (deepekagurunathan@my.unt.edu)

Vaishnavi G. Bhamare (vaishnavigovindbhamare@my.unt.edu)
