Revolutionizing Game Development: An AI Strategy for Enhanced Cost-Efficiency and Time-Saving using Procedural Content Generation and Machine Learning

1st Aaron Goldstein
College of Computing, Engineering and Construction
University of North Florida
Jacksonville, United States
n01421643@unf.edu

***Abstract*—Game development is a complex and resource-intensive process, necessitating significant human capital and computational resources. Considering recent advancements in gaming technology, development costs, and extended production timelines will only continue to grow if improvements are not made to streamline the creation process. Moreover, there is a growing gap between small and large (AAA) game studios as aspiring developers may struggle to complete and produce games promptly that can meet modern customer expectations. Consequently, if this gap widens, there may be less competition for large game studios, providing less incentive for them studios to innovate in new titles, leading to worse customer experiences. This study aims to explore how, utilizing artificial intelligence (AI) methodologies such as machine learning (ML), procedural content generation (PCG), and automated testing, game developers can optimize game development processes to make them more time and cost-efficient.**

***Index Terms*—Artificial Intelligence, Game Development, Cost Efficiency, Time Efficiency, Procedural Content Generation, Machine Learning, Video Games**

I. Introduction

The problem this research addresses is the substantial work and resources required to create modern games, which can inhibit innovation and limit market competition. Through the employment of AI technologies, we can streamline the game development process, reduce reliance on manual labor, and empower more individuals to pursue their creative endeavors within the game development industry, fostering innovation and creativity. Coupled with the power to enhance brainstorming processes and game design aspects, all these advantages can significantly reduce time and monetary investments.

The motivation for this study lies in the potential of reducing the large gap between small game studios and large triple AAA companies by making the development process more efficient. In this paper, the author will survey the current research on the field, compare similar games that utilize PCG and those that don't, as well as the sizes of those respective studios, the amount of innovation these games show according to the gaming public, and as researchers and developers what lessons we can learn and apply in new projects. In addition, due to the reduced cost of development when utilizing AI, the author hopes studios will be driven to focus more on creative endeavors, be apt to take risks and innovate to create a superior overall gaming experience for the public.

Existing research and papers on this topic were sourced from Google Scholar and Phind, an artificial intelligence tool specialized in technical tasks. Papers were selected based on their relevance to critical terms such as Artificial Intelligence, Procedural Content Generation, Video Games, and Video Game Development, and the reputation of the publishing journal. Essential definitions pertinent to this article are listed below, followed by a literature review.

A. Definitions

1) **Procedural Content Generation (PCG):** Procedural Content Generation (PCG) is a method used in game development where game content, such as levels, landscapes, characters, or objects, is generated algorithmically rather than being manually created by designers. PCG automatically creates vast and varied content, enabling more diverse and dynamic game environments and experiences.
2) **Triple AAA Companies (AAA):** Triple AAA companies, or simply "AAA," refer to the largest video game developers and publishers that have higher development and marketing budgets. AAA games, produced by these companies, are high-quality, big-budget games that offer profound, immersive gaming experiences, often featuring cutting-edge graphics and technology.
3) **Streamlining:** In the context of game development, streamlining refers to the process of making game design, production, or mechanics more efficient and user-friendly. It involves simplifying or optimizing various aspects of game development or the gameplay itself, such as interfaces, controls, or processes, to improve the overall player experience or development workflow.
4) **Artificial Intelligence (AI) in Game Development:** AI in game development refers to the simulation of

human intelligence processes by machines to enhance gaming experiences. This can include controlling non-player character (NPC) behavior, optimizing game mechanics, and adapting in-game elements based on player behavior.

5) **Machine Learning (ML) in Game Development:** ML in game development involves enabling computers to learn and improve from experience, adjusting in-game elements without being explicitly programmed. ML can be used for various applications, such as personalizing player experiences and improving NPC behaviors.
6) **Automated Testing in Game Development:** Automated testing in game development involves using tools and scripts to automatically execute test cases and evaluate the results, ensuring that the game runs as expected and identifying bugs or areas for improvement.
7) **Innovation in Game Development:** Innovation refers to introducing new ideas, devices, or methods. In game development, this can involve creating unique game mechanics, narratives, technologies, or design approaches to enhance player experience and engagement.
8) **Small Development Studios:** Small development studios typically operate with limited resources and smaller teams, focusing on niche markets or innovative game ideas. They might lack the extensive budgets of AAA studios but often offer unique and creative game titles.
9) **Cost Efficiency and Time Efficiency in Game Development:** These terms refer to optimizing game development processes to reduce expenses and shorten production timelines, making the industry more sustainable and allowing for the production of games that meet player expectations within reasonable budgets and schedules.
10) **Video Game Development:** Video game development encompasses the entire process of creating a video game, from initial concept and design to programming, testing, and release. It involves various disciplines like design, art, programming, and project management.

B. Overview

The key investigations of this paper are the following.

1) **Literature Review:**
    - Conduct a comprehensive exploration of existing academic and industry resources.
    - Focus on established AI technologies and their applications in game development.
2) **Data Analysis:**
    - Create a new dataset, collecting quantitative data such as development time, cost, and game performance for each game studied.
    - Group data into two PCG groups, uses PCG or not PCG depending on the level of PCG usage. Some PCG games may not have used as much PCG as others in the PCG group. In contrast, some non-PCG games may have used some form of PCG, like, for example, "Horizon Zero Dawn, which used GPU-based Procedural Placement for vegetation and objects despite the majority of the quests and content being handcrafted.
    - Compare and contrast these metrics across both groups.

In the results and conclusion, the paper will:

- Present the data findings detailing the effectiveness of PCG usage across the different games.
- Summarize the data analysis, emphasizing key metrics that showcase the impact of AI on game development.
- Provide suggestions for further usage of AI technologies in video games and directions for future research.

C. Literature Review

The first paper the author reviewed was a "Rule-Based Procedural Content Generation System" by Oliveira et al. [1]. This paper proposed a PCG system for utilizing biome seeds to create multiple biomes and densely populate them with specific assets. As a result of this paper's work, the time and resources taken to create massive open worlds may be drastically reduced while also being flexible enough to meet the requirements of a level designer.

The second paper reviewed by the author was "Procedural Content Generation for Games: A Survey" by Hendrikx et al. [2]. In this paper, the authors thoroughly review the different classes of game content that can be generated procedurally: game bits (textures, sounds), game space (indoor, outdoor maps), game systems (ecosystems, road networks, entity behavior), game scenarios (puzzles, stories), game design (game rules, setting, story, and theme), and derived content (news broadcasts of user actions in games, or leaderboards). Additionally, they provide an overview of games currently using PCG, such as Elite, a space exploration and trading game, and directions for future research in PCG, such as more detailed generators to improve realism and semi-automatic evaluation of generated content to ensure it meets game requirements or developer standards.

The third paper reviewed by the author was Experience-Driven Procedural Content Generation by Yannakakis et al. [3]. This paper focuses on utilizing affective and cognitive modeling alongside real-time adjustment of game content to personalize the user experience. They additionally provide a taxonomy of PCG algorithms and a framework for PCG, which computational models of user experience drive; they call their approach Experience-Driven Procedural Content Generation (EDPCG). In this paper, the authors discuss some of the challenges around this approach, such as the quantification of the user experience and the assessment of content quality, a point also made by Hendrikx et al. [2]. The authors also discuss how the usage of this technology also expands beyond that of games, referring to its potential applications in web applications, interface design, and art. [3]

The fourth article the author reviewed is "Deep Learning for Procedural Content Generation" by Liu et al. [4]. This article focuses on how deep learning can be applied to content generation and the range of inventions it has powered in content production. The article talks about how deep learning has been used for game levels, think of (2D Super Mario Bros maps), generating game text (AI Dungeon 2), character models, and even music and sound with quite a few games now involving procedural soundtracks. It also details the various neural architectures used for game development, such as GANS, and the different methods used: supervised learning, standard unsupervised learning, reinforcement learning, adversarial learning, and evolutionary computation. The article suggests a learning method for training generators, to train them on content from other similar games; they provide an example of the FPS genre, mentioning how it should be possible to train on levels from Quake, Halo, and Call of Duty to learn how to generate new levels for a Half-Life Game, as all these human-designed content generally share the same functionality constraints (i.e, it's complete, usually free of major of game-breaking bugs that destroy immersion, floating buses, cars, or buildings, etc). They apply the same logic to generating character models, which can be done by training generators on characters from other games, as they will also meet the same functionality constraints.

The fifth and final paper analyzed for this review was "Recent Research on AI in Games" from Xia et al. [5], who performed a systematic survey of recent research on AI in games with a particular focus on believable agents in non-playable characters, game level generation with procedural content generation, and player profiling in player modeling. For more intelligent game NPCs, they discussed existing research involving reinforcement learning through a framework to evolve believable agents. With this framework, NPC agents can explore huge state spaces and exhibit behavioral diversity while still being able to react to different users, according to the framework creators. For PCG, the survey authors discuss the commonly seen games used for PCG research; again, Super Mario Bros is discussed. The authors also discuss a level generator for Angry Birds that can produce stable levels by selecting pre-defined segments. In discussing player experience modeling, they point out how a big challenge is to gain access to the game engine or log information to retrieve player game data. They also mention a potential solution provided by a researcher: convolutional neural networks and transfer learning to derive game logs from a game video. The researcher who proposed this solution explored three different deep CNNs to evaluate player interest in a game through videos. They evaluated their methods through an annotated gameplay video dataset and determined an overall accuracy rate of 75%. Xia et al. also discuss current research utilizing a player's heart rate data to evaluate a player's state while playing HeartStone or Dota 2. The survey notes how the current field of game AI is limited to a few types of games, and much expansion is needed to realize the generalization of game AI. Given the information above, this paper's study, "Revolutionizing Game Development: An AI Strategy for Enhanced Cost-Efficiency and Time-Saving using Procedural Content Generation and Machine Learning," will focus on analyzing how companies utilize these techniques to improve the efficiency of the game development process with a focus on reducing costs and allowing each developer to maximize their development potential. The author hypothesizes through the effective usage of AI, developers can see substantial improvements in game design, potential world scale, and cost savings in time, money, and human resources.

II. Methodology

The study will use a different method from those papers discussed in the review, focusing on real-world industry data revolving around the usage of PCG. The author will research existing games that have utilized cutting-edge PCG and AI techniques in their game development process and analyze how those techniques have impacted their success, the quality of their game, and resource usage.

As discussed briefly in the introduction, a new dataset was created for this study to quantify the differences between two key groups to distinguish games in the industry, PCG and non-PCG. The author found overall, there was a lack of publicly available information on that data that illustrates the tangible effects on efficiency and reduced resource usage that PCG provides. The metrics used are listed in the subsections below, briefly explaining each. The metrics can be distinguished between data that could be retrieved from publicly available resources and that which could not be retrieved due to lack of availability and time constraints. The data that could not be retrieved was generated using ChatGPT 4 for analysis completeness.

A. Metrics for which data was available:

- Average Dev time (Average time in months between full release and early release, or simply the full release if the game was not released in early access)
- Budget (Approximate cost to develop game, easier to find as exact numbers are not made publicly available)
- Revenue (Sometimes forced to use speculations from third parties, actual numbers not always publicly available)
- Profit (Approximated as Revenue - Budget)
- User satisfaction (Use Metacritic scores for PC to ensure consistency, or the only available console itself if an exclusive title)
- Team Size (Size of the game studio)
- PCG (Does the game primarily utilize PCG? Sometimes it's difficult to quantify a game as PCG. Listing Starfield and Horizon Zero Dawn as primary examples) Just because a game is classified as PCG does not mean there is no handcrafted content, and if it is not classified as PCG that it does not use PCG at all! (Horizon Zero Dawn did use GPU-based Procedural Placement for vegetation and objects despite the majority of the quests and content being handcrafted)

B. Metrics for data which was generated using ChatGPT 4

- Performance (How well is the game optimized compared to other games of the same genre.)

- Innovation (How creative is the game, or how well does it stand out compared to similar titles? What new things does it contribute?)

C. Conducting the Analysis and Generating the results

The data analysis was conducted using a Python script utilizing various third-party libraries. The key ones are:

- Pandas: Store CSV data in a format that Python can analyze and derive results from
- Matplotlib: Visualize the results of the analysis
- Seaborn: Make the visual results easier to understand and derive conclusions
- Scipy: Test the normality of the data and perform various statistical tests

In running the script through the command line, the user can choose whether or not to use the data generated using ChatGPT 4 in the analysis by including the optional '–include_generated' tag. This data is not included in the analysis by default to help ensure the accuracy of the results and conclusions. The script begins by testing the normality for each metric using the Shapiro-Wilk Test to determine which statistical Test to use for measuring the statistical difference across the two PCG groups in a specific area. In the study, the author selected 0.05 as the significance threshold. The null hypothesis is the data is normally distributed, and if $p < 0.05$, we reject that null hypothesis. In that case, we would determine that our data for that metric is not normally distributed. If our data is normally distributed for a metric, we use an independent samples t-test and the means for the two groups to measure their statistical differences. However, if the data is not normally distributed for a metric, we use the Mann-Whitney U test and the medians for the two groups to measure the statistical difference. Finally, we also perform linear regression to analyze critical relationships between the metrics. The author used the condition $R > 0.5$ to determine whether a correlation was strong and whether the correlation was positive or negative based on whether $R >= 0$.

Key Correlations Measured

- Team Size - PCG
- Budget - PCG
- Team Size - Budget
- Team Size - Average Dev Time

In addition, we will generate three types of visualizations to help us better understand the data and results retrieved from the analysis.

- Histograms: Showcase the distribution of the data across the different metrics
- Bar Charts: Showcase the difference in medians or averages across the two PCG groups depending on the normality of the data distribution for a given metric
- Scatterplots: Used to show the results of linear regression for key relationships

An acknowledged limitation of this study is the limited sample size of the dataset, containing 16 games, 8 PCG, and eight non-PCG. Therefore, outliers like Minecraft's revenue or Dwarf Fortress's dev time may significantly affect the data distribution, reducing the likelihood of a normal distribution. The table of studied games and the dataset on which the analysis was performed can be seen in Table 1.

III. Evaluation of Analysis

Now that we have discussed the methodology used, the author will briefly explain how we will evaluate the results of the statistical tests and different visualizations. To understand the result of our T-test or Mann-Whitney U test, the null hypothesis for both types of Tests, regardless of which is used, is that the difference is not statistically significant. The difference is statistically significant if the $p < 0.05$. This is a different null hypothesis from our normality test. A higher T statistic value for the result of a T-test means a more significant difference between group means, while a lower T statistic represents the opposite. A low U-value for the Mann-Whitney U test represents a higher difference between the data distribution, while a higher U means a significant overlap of the distributions. We will start by visualizing the distributions for the different metrics, PCG and non-PCG, using histograms and then analyzing the bar plots for each metric to investigate the difference across the two groups. Finally, we do a ratio analysis between PCG and non-PCG for budget, revenue, and profit to better understand the financial implications of using PCG.

A. Average Development Time

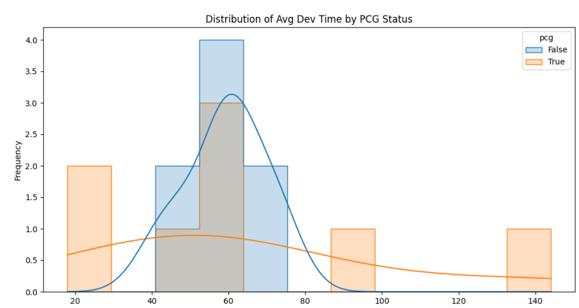

Fig. 1. Histogram of Average Development Time

Normality Test for Average Development Time:

- Shapiro-Wilk Test Statistic: 0.83901
- P-Value: 0.00943
- Conclusion: Data is Not Normally Distributed

Median Values:

- PCG: 58.75
- Non-PCG: 60.00

Statistical Test: Mann-Whitney U test

- Test Statistic: 25.50000
- P-value: 0.52180
- Conclusion: Not Statistically Significant

Explanation: A high U value means a more significant overlap of the distributions. Combined with a p-value greater than

Game Name	PCG	Early Access Time	Full Release Time	Avg Dev Time	Budget	Revenue	Profit	Performance	Innovation	User Satisfaction	Team Size
No Man's Sky	TRUE		60	60	1,248,000	50,000,000	48,752,000	85	90	61	17
Elite Dangerous	TRUE	20	32	26	12,000,000	125,000,000	113,000,000	88	85	80	40
Starfield	TRUE		96	96	200,000,000	600,000,000	400,000,000	90	92	86	500
Minecraft	TRUE	6	30	18	100,000	3,000,000,000	2,999,900,000	95	95	93	1
Stardew Valley	TRUE		48	48	120,000	225,000,000	224,880,000	90	88	88	1
Terraria	TRUE	4	112	58	100,000	349,000,000	348,900,000	88	85	83	11
Valheim	TRUE	48	71	59.5	150,000	140,000,000	139,850,000	92	90	90	5
Dwarf Fortress	TRUE	48	240	144	200,000	7,200,000	7,000,000	85	80	93	2
The Legend of Zelda: Breath of the Wild	FALSE		60	60	120,000,000	1,890,000,000	1,770,000,000	95	95	97	300
The Witcher 3: Wild Hunt	FALSE		42	42	81,000,000	2,448,000,000	2,367,000,000	95	90	93	250
Red Dead Redemption	FALSE		60	60	80,000,000	3,400,000,000	3,320,000,000	90	88	95	1000
Grand Theft Auto V	FALSE		60	60	137,500,000	6,000,000,000	5,862,500,000	95	90	97	1000
God of War (2018)	FALSE		60	60	100,000,000	500,000,000	400,000,000	95	92	94	300
Horizon Zero Dawn	FALSE		72	72	47,000,000	400,000,000	353,000,000	90	88	90	250
Spider-Man (PS4)	FALSE		48	48	229,000,000	1,287,000,000	1,058,000,000	92	90	87	400
Ghost of Tsushima	FALSE		72	72	60,000,000	583,000,000	523,000,000	90	88	83	160

TABLE I
GAME DATA

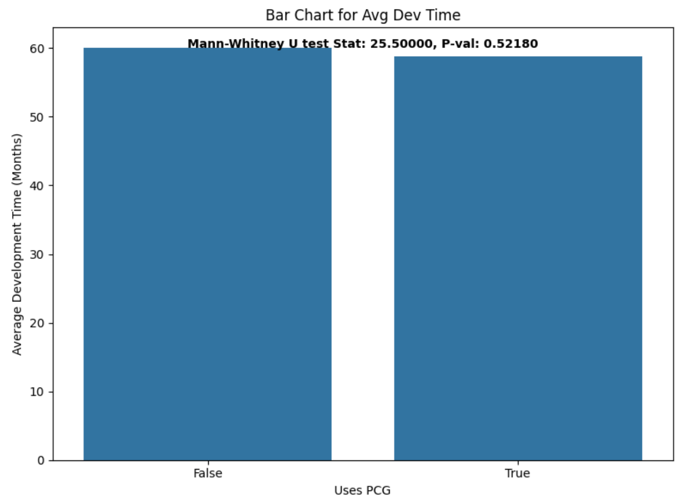

Fig. 2. Bar Chart for Average Development Time Results

0.05, we can derive that between the two PCG groups, there is not a statistically significant difference in Average Development Time (the average between the time taken to early access release and the full release in months for a specific game, or simply full release if a game did not have early access).

B. Budget

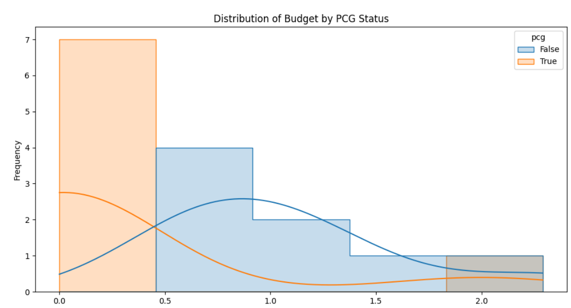

Fig. 3. Histogram of Budget

Normality Test for Budget:

- Shapiro-Wilk Test Statistic: 0.85016
- P-Value: 0.01370
- Conclusion: Data is Not Normally Distributed

Median Values:

- PCG: $175,000.00
- Non-PCG: $90,500,000.00

Statistical Test: Mann-Whitney U test

- Test Statistic: 7.00000
- P-value: 0.01003
- Conclusion: Statistically Significant

Explanation: The significant difference in the median values for the budget, as indicated by the low p-value (0.01003) from

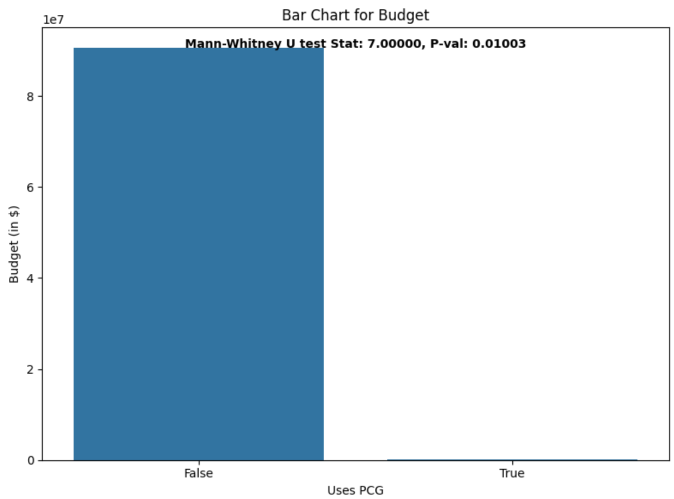

Fig. 4. Bar Chart for Budget Results

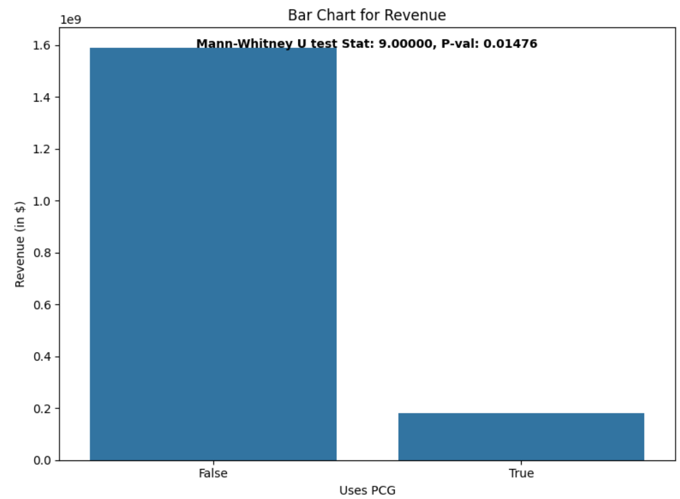

Fig. 6. Bar Chart for Revenue Results

the Mann-Whitney U test, suggests a statistically significant difference in the budget allocation between PCG and non-PCG games. PCG games tend to have a much smaller median budget than non-PCG games. This considerable variance highlights how PCG games are generally developed with much fewer financial resources.

C. Revenue

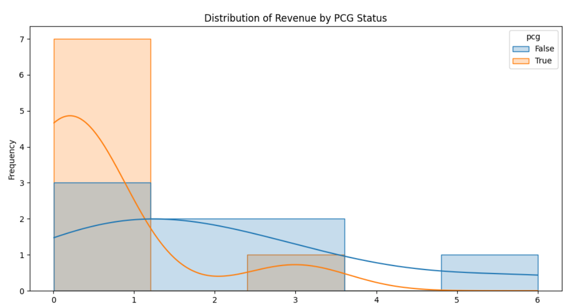

Fig. 5. Histogram of Revenue

Normality Test for Revenue:

- Shapiro-Wilk Test Statistic: 0.76952
- P-Value: 0.00110
- Conclusion: Data is Not Normally Distributed

Median Values:

- PCG: $182,500,000.00
- Non-PCG: $1,588,500,000.00

Statistical Test: Mann-Whitney U test

- Test Statistic: 9.00000
- P-value: 0.01476
- Conclusion: Statistically Significant

Explanation: The analysis of the revenue data reveals a statistically significant difference between PCG and non-PCG games, as shown by the Mann-Whitney U test (p-value: 0.01476). This difference is highlighted by the substantial gap in the median revenue figures, with non-PCG games earning considerably more. The results underscore the significant financial impact of using non-PCG methods in terms of revenue generation. It is important to note, however, that marketing may play a significant role in this difference; in correlations, we see that large studios (large team sizes) are typically associated with not using PCG, and such studios will likely have higher marketing budgets to help their games reach a wider audience.

D. Profit

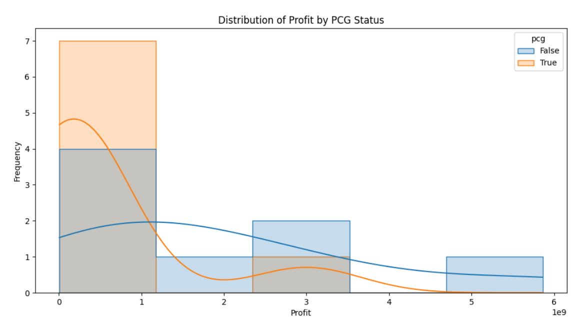

Fig. 7. Histogram of Profit

Normality Test for Profit:

- Shapiro-Wilk Test Statistic: 0.75035
- P-Value: 0.00064
- Conclusion: Data is Not Normally Distributed

Median Values:

- PCG: $182,365,000.00
- Non-PCG: $1,414,000,000.00

Statistical Test: Mann-Whitney U test

- Test Statistic: 7.50000

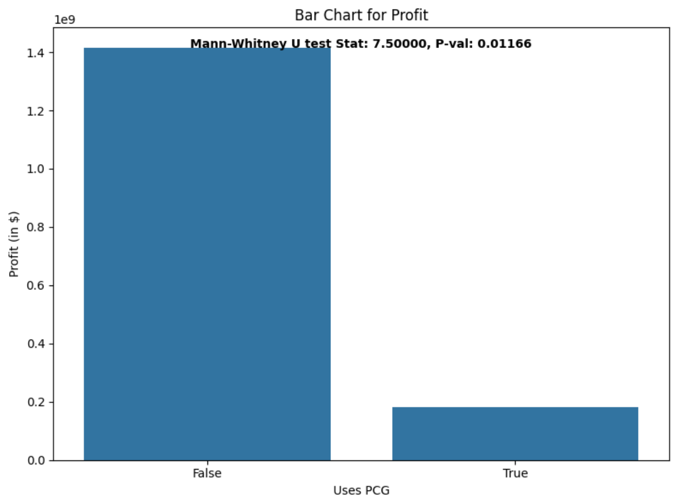

Fig. 8. Bar Chart for Profit Results

- P-value: 0.01166
- Conclusion: Statistically Significant

Explanation: The profit data, showing a statistically significant difference (p-value: 0.01166) between PCG and non-PCG games, indicates that non-PCG games generally yield higher profits. In addition to budgets, PCG games do not match the profit levels of their non-PCG counterparts, possibly reflecting market preferences or the larger scale of non-PCG game projects; again, as discussed in revenue, it's also important to consider that marketing may play a factor in influencing this metric.

E. User Satisfaction

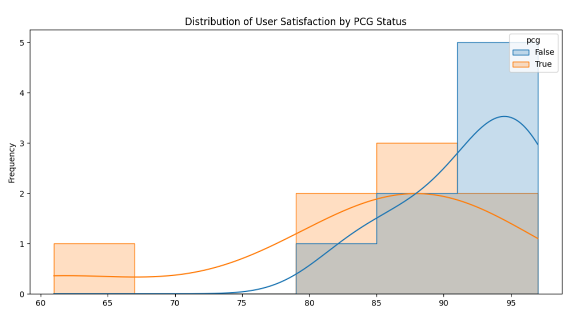

Fig. 9. Histogram of User Satisfaction

Normality Test for User Satisfaction:

- Shapiro-Wilk Test Statistic: 0.80896
- P-Value: 0.00359
- Conclusion: Data is Not Normally Distributed

Median Values:

- PCG: 87.00
- Non-PCG: 93.50

Statistical Test: Mann-Whitney U test

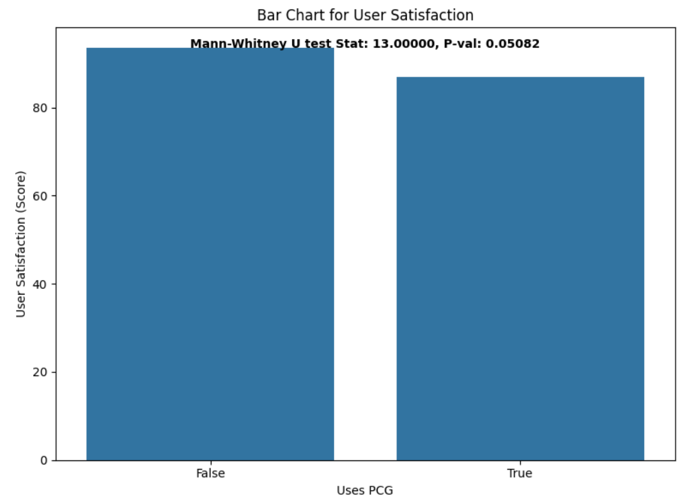

Fig. 10. Bar Chart for User Satisfaction Results

- Test Statistic: 13.00000
- P-value: 0.05082
- Conclusion: Not Statistically Significant

Explanation: When analyzed through the Mann-Whitney U test (p-value: 0.05082), the median values for user satisfaction suggest no statistically significant difference between PCG and non-PCG games. This indicates that using PCG does not necessarily impact user satisfaction adversely, and both types of games can achieve similar levels of player satisfaction.

F. Team Size

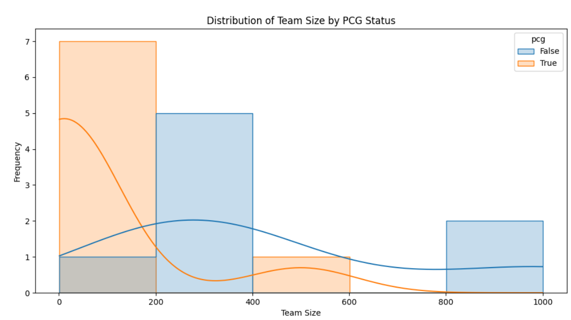

Fig. 11. Histogram of Team Size

Normality Test for Team Size:

- Shapiro-Wilk Test Statistic: 0.77486
- P-Value: 0.00129
- Conclusion: Data is Not Normally Distributed

Median Values:

- PCG: 8.00
- Non-PCG: 300.00

Statistical Test: Mann-Whitney U test

- Test Statistic: 6.00000
- P-value: 0.00723

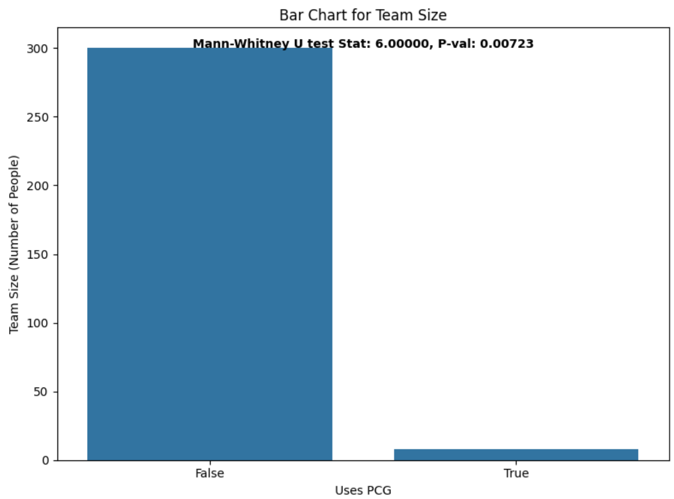

Fig. 12. Bar Chart for Team Size Results

- Conclusion: Statistically Significant

Explanation: A statistically significant difference in team sizes for PCG versus non-PCG games is observed (p-value: 0.00723). The median team size for PCG games is markedly smaller, highlighting the effectiveness of PCG in reducing the need for larger development teams due to the apparent benefits of allowing an individual developer to complete more tasks promptly through automation. Combined with similar user satisfaction scores, this bodes well for the future of PCG despite the high revenues of non-PCG titles.

G. *Performance*

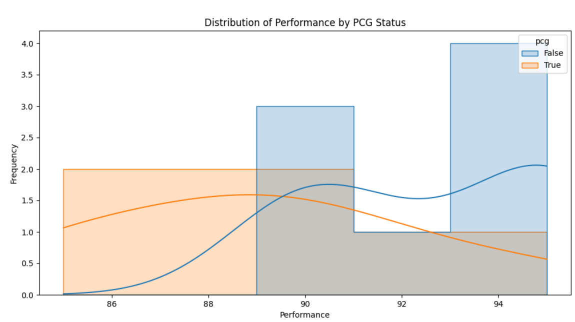

Fig. 13. Histogram of Performance

Normality Test for Performance:

- Shapiro-Wilk Test Statistic: 0.88478
- P-Value: 0.04609
- Conclusion: Data is Not Normally Distributed

Median Values:

- PCG: 89.00
- Non-PCG: 93.50

Statistical Test: Mann-Whitney U test

- Test Statistic: 12.50000

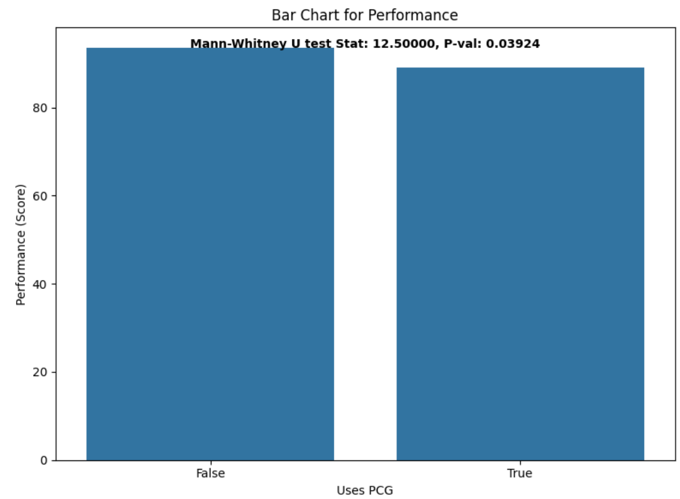

Fig. 14. Bar Chart for Performance Results

- P-value: 0.03924
- Conclusion: Statistically Significant

Explanation: The performance metric analyzed by the Mann-Whitney U test shows a statistically significant difference between PCG and non-PCG games (p-value: 0.03924). Despite this, the difference in median values is not as pronounced given the relatively high U value, suggesting that while there are measurable differences in-game performance, they are not highly divergent between the two categories. This analysis should be taken cautiously as the data for this metric was generated using ChatGPT 4 and does not come from not real-world data.

H. *Innovation*

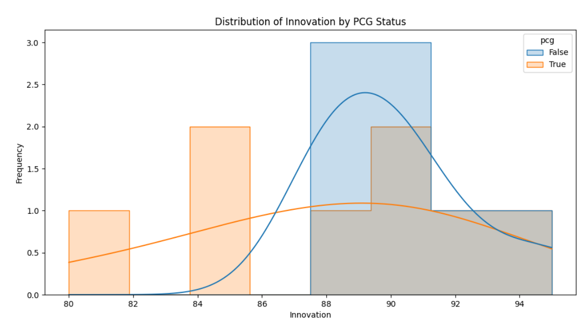

Fig. 15. Histogram of Innovation

Normality Test for Innovation:

- Shapiro-Wilk Test Statistic: 0.92877
- P-Value: 0.23310
- Conclusion: Data is Normally Distributed

Average Values:

- PCG: 88.12
- Non-PCG: 90.12

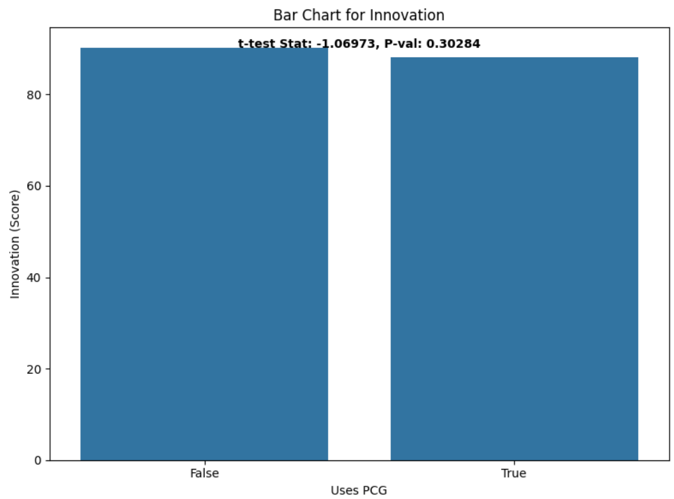

Fig. 16. Bar Chart for Innovation Results

Fig. 17. Scatter Plot for Team Size vs Development Cost Correlation

Statistical Test: t-test

- Test Statistic: -1.06973
- P-value: 0.30284
- Conclusion: Not Statistically Significant

Explanation: The t-test results for innovation (p-value: 0.30284) indicate that there is no statistically significant difference in the level of innovation between PCG and non-PCG games. Both types of games, on average, exhibit similar levels of innovation, suggesting that the use of PCG does not inherently limit or enhance a game's innovative aspects. As with the performance conclusion, the analysis of this metric should be taken with a grain of salt, as it relies on generated rather than real-world data.

I. Team Size vs Development Cost Correlation

Correlation Results:

- Slope: 144556.86
- Intercept: 28495661.98
- R: 0.6375452910087153
- R-squared: 0.41
- P-value: 0.00789
- Nature: Positive Correlation
- Strength: High Correlation

Explanation: A moderate r-squared value of 0.63 means a sizable proportion of variance in development cost can be explained by team size.

J. Team Size vs Development Time Correlation

Correlation Results:

- Slope: 0.00
- Intercept: 60.47
- R: 0.04367404654924944
- R-squared: 0.00
- P-value: 0.87241
- Nature: Positive Correlation
- Strength: Low Correlation

Explanation: This low correlation is likely explained by indie developers using PCG to offset human capital requirements and increased bureaucracy within larger game studios.

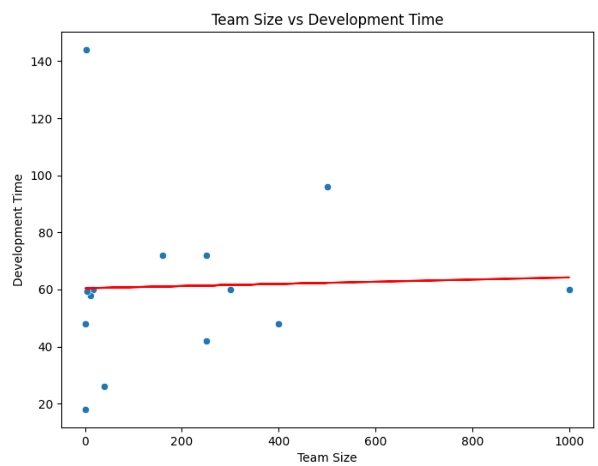

Fig. 18. Scatter Plot for Team Size vs Development Time Correlation

K. Team Size vs. PCG Correlation

Correlation Results:

- Slope: -0.00
- Intercept: 0.75
- R: -0.6054405958117982
- R-squared: 0.37
- P-value: 0.01294
- Nature: Negative Correlation
- Strength: High Correlation

Explanation: Larger teams tend not to use PCG; an R squared of 0.37 means a decent proportion of the variance in PCG usage can be explained by team size.

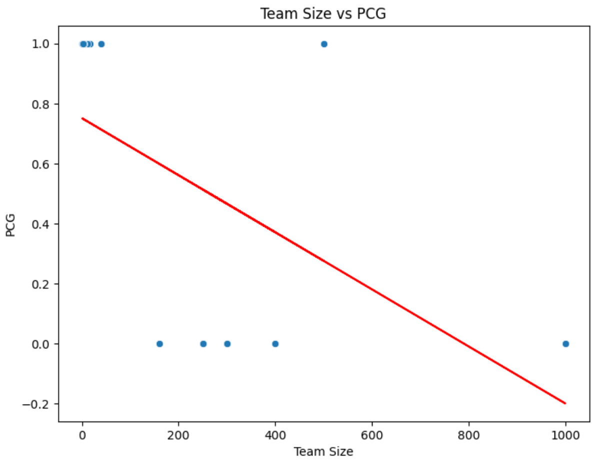

Fig. 19. Scatter Plot for Team Size vs PCG Correlation

L. Development Cost vs PCG Correlation

Correlation Results:

- Slope: -0.00
- Intercept: 0.76
- R: -0.5548110289339537
- R-squared: 0.31
- P-value: 0.02571
- Nature: Negative Correlation
- Strength: High Correlation

Explanation: Higher development costs are associated with a lower likelihood of using PCG. With an R-squared of 0.31, this metric also accounts for a fair proportion of variance in PCG usage.

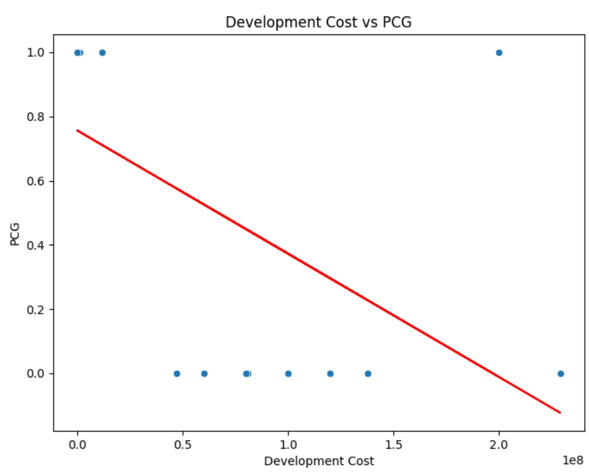

Fig. 20. Scatter Plot for Development Cost vs PCG Correlation

M. Ratios Comparison between PCG and Non-PCG

The study also examined the budget, profit, and revenue ratios between PCG and non-PCG games to understand their

Metric	**Test Used**	**Test Value**	**P-value**	**Normality**	**PCG Median/Mean**	**Non-PCG Median/Mean**
Average Development Time	Mann-Whitney U	25.50000	0.52180	Not Normal	58.75	60.00
Budget	Mann-Whitney U	7.00000	0.01003	Not Normal	$175,000	$90,500,000
Revenue	Mann-Whitney U	9.00000	0.01476	Not Normal	$182,500,000	$1,588,500,000
Profit	Mann-Whitney U	7.50000	0.01166	Not Normal	$182,365,000	$1,414,000,000
User Satisfaction	Mann-Whitney U	13.00000	0.05082	Not Normal	87.00	93.50
Team Size	Mann-Whitney U	6.00000	0.00723	Not Normal	8.00	300.00
Performance	Mann-Whitney U	12.50000	0.03924	Not Normal	89.00	93.50
Innovation	t-test	-1.06973	0.30284	Normal	88.12	90.12

TABLE II
SUMMARY OF STATISTICAL ANALYSIS

Correlation	Slope	R	R-squared	P-value
Team Size vs Dev Cost	144556.86	0.6375	0.41	0.00789
Team Size vs Dev Time	0.00	0.0437	0.00	0.87241
Team Size vs PCG	-0.00	-0.6054	0.37	0.01294
Dev Cost vs PCG	-0.00	-0.5548	0.31	0.02571

TABLE III
SUMMARY OF CORRELATION ANALYSIS

financial implications. The following ratios were calculated:

- **Budget Ratio (PCG / Non-PCG):** 0.002
- **Profit Ratio (PCG / Non-PCG):** 0.129
- **Revenue Ratio (PCG / Non-PCG):** 0.115

These ratios highlight the financial differences between PCG and non-PCG games, indicating a significantly lower budget and yet a proportionally higher profit for PCG games compared to non-PCG games.

IV. RESULTS, CONCLUSIONS, AND FUTURE WORK

A. Key takeaways from statistics

The results of the author's study indicate that smaller teams can effectively utilize procedural content generation (PCG). This approach helps offset human capital requirements, leading to similar user satisfaction and development time results compared to studios with much larger teams.

However, it is essential to note that PCG games typically generate smaller profits, with a ratio of 0.129 (or 12.9%) of non-PCG games. However, this is contrasted by the budget requirements ratio, which is even more remarkable at a mere 0.002 (less than 1% of non-PCG games).

Additionally, the marketing efforts of larger game studios should be considered a potentially significant factor influencing the revenue and profit of non-PCG games, possibly even more than the quality of the games themselves.

This analysis highlights the profitability of utilizing PCG, particularly in cost reduction. This is evidenced by the median PCG budget, which constitutes a tiny fraction of the non-PCG median budget. In contrast, median PCG profits achieve a much more significant relative percentage of the non-PCG median profit, showcasing the potential financial benefits of PCG in the gaming industry.

B. Future Work

None of the non-generated data was normally distributed; hence, the Mann-Whitney U test was used to measure the statistical difference for each. To avoid this limitation in future research and potentially increase the reliability of results, a larger sample size should be used to reduce the effect of outliers on the data. The innovation metric was normally distributed, and a t-test was used to determine the statistical difference between the two groups; however, the data for this metric was generated using ChatGPT 4, so it can not be used conclusively.

To expand the study to use non-generated performance and innovation metric data, actual benchmarking can be done for each game using the same hardware or the only console available if it is an exclusive title to measure the average fps over an hour of normal gameplay. In addition, sentiment analysis may be performed using machine learning to score innovation by scraping text off public sources such as Reddit or Steam reviews.

In addition, performing this sort of analysis during game creation would be helpful, as would heavily monitoring the effect of using PCG on speeding up each stage of the development process. Future plans to expand this study include creating three different types of games, two for each type, one that uses PCG, and one that does not, for a total of 6 games. The study would include measuring the effect of PCG on each stage of the development process and the overall effect on quality according to game testers.

REFERENCES

[1] G. Oliveira, L. Almeida, J. P. Sousa, B. Barroso, and I. Barbedo, "A rule based procedural content generation system," in *International Conference on Optimization, Learning Algorithms and Applications*. Springer, 2022, pp. 107–113.

[2] M. Hendrikx, S. Meijer, J. Van Der Velden, and A. Iosup, "Procedural content generation for games: A survey," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 9, no. 1, pp. 1–22, 2013.

[3] G. N. Yannakakis and J. Togelius, "Experience-driven procedural content generation," *IEEE Transactions on Affective Computing*, vol. 2, no. 3, pp. 147–161, 2011.

[4] J. Liu, S. Snodgrass, A. Khalifa, S. Risi, G. N. Yannakakis, and J. Togelius, "Deep learning for procedural content generation," *Neural Computing and Applications*, vol. 33, no. 1, pp. 19–37, 2021.

[5] B. Xia, X. Ye, and A. O. Abuassba, "Recent research on ai in games," *2020 International Wireless Communications and Mobile Computing (IWCMC)*, pp. 505–510, 2020.