The Indonesian Journal of Computer Science
www.ijcs.net
Volume 14, Issue 3, June 2025
https://doi.org/10.33022/ijcs.v14i3.4800

Generative Artificial Intelligence in Agile Product Management: Optimizing Task Coordination and Team Efficiency in Software Development

Della Anissa Putri Widodo[1], Apriade Voutama [2]
dellaapw@gmail.com[1], apriade.voutama@staff.unsika.ac.id[2]
[1,2] Faculty of Computer Science, Universitas Singaperbangsa Karawang

Article Information

Received : 16 Mar 2025
Revised : 16 May 2025
Accepted : 9 Jun 2025

Keywords

Agile Product Management, Artificial Intelligence, Generative AI, Software Development, Task Coordination

Abstract

The inclusion of Artificial Intelligence (AI) into Agile Product Management is revolutionizing the process of efficient software development in terms of improved coordination of tasks and team productivity optimization. Though Agile processes have gained much momentum with their iterative approach, issues like ineffective backlog management, inefficient resource allocation, and lengthy sprints continue to persist with human errors and long-time product development that could be mitigated with the use of Generative AI. This study discusses how AI-driven automation can resolve such inefficiencies in the form of a case study of the MSIB Batch 7 2024 PINTURA project, where lead times in software development were cut from 6-12 months to just under 2 months. By utilizing AI-driven backlog refinement, sprint optimization, report writing, data analysist, diagram generation, and code generation automation, the project was able to realize spectacular improvements in software delivery velocity and team collaboration. The results indicate that AI-driven Agile processes to be proven in maximizing efficiency to mitigate development bottlenecks and smooth project execution, offering a formal system for AI adoption in Agile software development.

Attribution-ShareAlike 4.0 International License

A. Introduction

By incorporating it into practises, they have fundamentally changed software development from rigid, linear project execution models to an iterative, flexible, collaborative one with the Agile methodology. This shift can then let the teams focus more on providing continuous feedback, delivering incremental product delivery, and respond in real time to market demands and customer needs. For nearly 20 years now Agile has become a widespread practise in industries all over and almost 80 percent of the organisations are adopting Agile framework to speed up development, maximise resource utilisation and increase predictability of project [1].

Agile methodologies like Scrum, Kanban and SAFe have helped development teams get shorter release cycles, and time development with user's expectations and support better stakeholders collaboration. The report states that 52% of Agile teams use Agile primarily for time to market reduction, operational efficiency and responsiveness to changing business environments [2]. In spite of these positive aspects, Agile approaches still are beset with significant difficulties; namely, in scenarios of large, multi structural, multi disciplinary teams whose need for synchronisation, workflow optimization and resource marching must be increased.

According to the empirical studies, 47 % of the Agile projects face inefficiencies that result in a delay, budget overrun and team misalignment [3]. Large scale environments come with these issues becoming more and more complex where teams will ultimately have to deal with cross functional dependencies, decentralised decision making, and fragmented task coordination. The use of manual refinement of backlog, sprint planning and workload distributing in Agile compounds all these inefficiencies to misprioritize the task, underestimation of sprint plan and misallocation of resources [4].

The manual practise of backlog refinement stands as a major obstacle in Agile development because teams need to perform continuous assessment of tasks while making changes and adjustments according to evolving project needs and stakeholder comments. The Agile flexibility needs this process but it proves to be very time-consuming while also being subjective and inconsistent. Manual operations in agile development lead teams toward developing redundant items and facing unpredictable deliverables and misaligned priorities [5]. Sprint planning essentially depends on human assessment but this process faces limitations due to human errors including inaccurate workload management and substandard execution of tasks. Traditional sprint planning methods produce workloads imbalances while causing delays in delivery and producing inadequate sprint results [6].

Agile teams, however, are prevented from planning ahead any real time resource allocation, as predictive analytics for assessing workload imbalance, dynamically reassign task that helps to convene the workload and keep a project at its efficiency. Without a data driven approach, project and task allocation to developers becomes a time consuming and high effort task. Additionally, the workloads for developers cannot easily be balanced nor can deliveries be predicted in terms of which specific activities are causing bottlenecks given unevenly distributed task tick box. These challenges demonstrate the necessity to address the

need of intelligent automation solutions to improve Agile workflows, eliminate inefficiencies and improve the development agility [7].

Artificial Intelligence in Agile Development

With great in AI (Artificial Intelligence) and ML (Machine Learning) comes great power of automation, which deals with inefficiencies in Agile project management that have existed for many years. By allowing teams to switch from manual, intuition-based tasks to intelligent, data informed, teams can better backlog order, more properly plan sprints, and better schedule resources. Combining AI in Agile lets your team use automation to refine backlog, estimate and predict sprint plannings, and use AI in collaboration to come up with the best plan, execute them, and deliver projects all over again [8].

One of the most impactful ways to utilize AI in Agile workflow is AI driven backlog refinement which uses natural language processing (NLP) or machine learning in order to analyze the historical sprint data, feature dependencies and shifting business priorities in the background. Dynamicly reordered tasks, remove redundancies in the backlog, and promising tasks prioritization. Studies have shown that AI based platforms such as GPT-4o Plus can give a reduction of backlog refinement time of up to 60%, which means development team stay in high impact features in line with business objectives [9].

But aside from backlog refinement, there is an equally important role that AI can play in sprint planning, which is to use predictive analytics to calculate team velocity, appliances distribution and risk factors in sprint. Sprint planning using AI driven tools gives us data backed recommendations of how to assign tasks proactively detecting if there are any potential sprint bottlenecks and balance work load as we progress. Studies based on existing empirical researches with Gemini AI as the advanced AI model have shown that it can increase the sprint planning accuracy by 70 percent and reduce manual overhead spent in sprint execution by an order of magnitude [10]. However, Agile is not just about sprint optimization, but also enables coding, testing and software deployment to be revolutionized by AI. Tools such as Blackbox AI make use of AI to automate code generation and error detection and optimize CI/CD pipeline in the form of AI-powered tools. So AI assisted coding was proven to reduce software defects by 43% improving overall software stability while increasing the cadence of the releases [11].

Furthermore, research suggests that AI-powered Agile methodologies enhance sprint execution efficiency by 40%**,** while accelerating software development timelines by 30–50%**.** AI's ability to automate Agile workflows and eliminate manual bottlenecks allows teams to focus on high-impact activities such as architectural design, strategic innovation, and product refinement [12]. By embedding AI-driven automation into Agile processes**,** development teams can achieve a more predictable, optimized, and scalable development cadence [13]. Yet despite all of these rapid advancements, a gap in research was found in the holistic integration of AI across the entire Agile lifecycle. While the referenced studies focuses on isolated applications of AI, such as backlog refinement or sprint planning, few have explored how useful Generative AI tools can efficiently cut production time and enhance the productivity of each agile process. Such as task assigment, feature prioritization to code, test, and deploy. Thus this research was created in the hopes to bridge the gap

by studying the comprehensive role of AI-driven automation in successfully opmtimizing all stages of the agile software development to contribute a more scalable, high quality, and efficient software delivery.

Case Study: AI-Driven Agile Acceleration in PINTURA

Within the context of digital education adventing, PINTURA project was a pioneering example of Agile acceleration by AI. PINTURA was developed under the MSIB Batch 7 initiative as a conventional e-learning platform, not just for education but for an intervention based on data and targeted at narrowing the skills gap between graduates and the workforce. While AI and the cloud computing industry have been going forward very fast, the education system has been lagging behind, particularly when comparing with expectations in the industry [14]

A nationwide survey revealed that 37.5 percent of students utilized supplementary learning resources on the Internet, 44.4 percent needed self directed learning as a means of career preparedness, and 47.2 percent felt that curricula of universities did not offer enough data regarding skill development concerning industry. This highlighted the need for an adaptive AI powered platform that can personalize learning pathways together with dynamic alignment of content to the evolving industry needs and true applicability of learning to the real world [15]

First, we projected the PINTURA development cycle equivalent to six months based on the traditional Agile workflow: the way of iterative sprints cycles, backlog refinement, and manual sprints planning. However, this was limited to relying on human driven estimations, insufficient and late project goals and workflow inefficiencies and thus, the development team integrated AI driven Agile methodologies and achieved a 600% acceleration in the project completion to just one month and sixteen days [16]

Three AI-powered systems were instrumental in this transformation. GPT-4 Plus automated backlog refinement, cutting sorting time by 55% by dynamically prioritizing tasks based on real-time user engagement and industry needs. Gemini AI enhanced sprint planning, improving task allocation by 70%, ensuring workload balance, and minimizing sprint bottlenecks. Blackbox AI optimized software development by automating code generation, debugging, and deployment pipelines, reducing software errors by 43% and improving system stability [17]**.**

By using these AI driven enhancements it reshaped the execution of the Agile in the sense that it eliminated manual backlog bottlenecks and so on, improving sprint predictability and reducing rework cycles. AI helped backlog prioritize overload by 42% and AI driven sprint planning optimized unplanned sprint scope changes by 48%. This in turn resulted in better sprint cadence, lower mid sprint disruptions and faster release of features for Agile teams [18].

The PINTURA case study also went further to provide the critical insights into the role of AI to improve software quality assurance other than efficiency gain. Automated backlog refinement, predictive sprint execution and AI driven bug detection showed that AI does not just make Agile workflows, it resets them. The study showed indeed that AI assisted Agile teams produce higher quality of software, decrease technical debt in software development, improve autonomy in development, all this is a pragmatic way of further development and scaling of AI driven automation of the software development process [19]

However, even though it was successful, there are many questions one can ask of the scalability and governance of AI-driven Agile frameworks. AI tools have already been demonstrated to be very useful in the case of mid-sized teams but their fit to large scale enterprise environments remains to be researched. In addition, with AI doing the prioritizing and execution of backlog and sprints, issues regarding algorithmic bias, transparency, and AI governance must be interrogated [20]

Future research may help companies scale their Agile scale with AI, understand how that scale would be acceptable in the long term, and establish best practices for AI and human collaboration within Agile teams. The PINTURA case study provides such a benchmark for AI's transformational impact in enabling intelligent automation to rewrite the software engineer workflow; transform the way the business innovates new software; and reshape future processes for Agile methodologies.

B. Research Method

Research Design

Future research can help companies scale their Agile scale with AI, determine how much Agile scale with AI would be fit for the future, and set up best practices for AI and human collaboration on Agile teams. The PINTURA case study illustrates such a benchmark in which AI enables intelligent automation so that the software engineer workflow can be rewritten, the way the business invents new software, and how to transform future processes for Agile methodologies.

First, this research uses Scrum inspired iterations in the Agile methodology used. The sprint cycles to which this development process was structured had been defined, with AI enhanced backlog grooming, sprint planning, real time adjustment, and retrospective evaluation. Agile principles tout adaptability, but this study is specifically about how AI automation facilitates an iterative software development within a (structured) Scrum-like workflow [21]

In this specific case, the research focuses on how generative AI tools (GPT-4 Plus, Gemini AI, Blackbox AI) help to optimize backlog refinement, sprint planning, resource allocation and the development efficiency on the whole. Under volatility and uncertainty, it is shown how these generative AI techniques can propel accuracy of decisions, enhance predictive analytics, and greatly improve project execution [22], [23]

The methodology includes two primary components::

1. Survey Data: Involving 54 respondents comprising students and industry professionals from the software development field, ensuring diverse and representative insights into AI's role in Agile practices.
2. Case Study: Focused on the PINTURA project, a real-world example of AI-driven Agile workflows. The case study examines the AI tools implemented (e.g., GPT-4 Plus, Gemini AI, and Blackbox AI) and their effects on sprint velocity, backlog efficiency, and team collaboration.

Table 1. Time-to-Market and Development Time Efficiency

Phase	Pre-AI Time (Months)	Post-AI Time (Months)	Time Saved (%)
Overall Development	6	1.5	75%
Project Planning	20 days	7 days	65%

Phase	Pre-AI Time (Months)	Post-AI Time (Months)	Time Saved (%)
Backlog Refinement	15 days	5 days	67%
Sprint Execution	30 days	10 days	66%
Testing & Debugging	25 days	8 days	68%

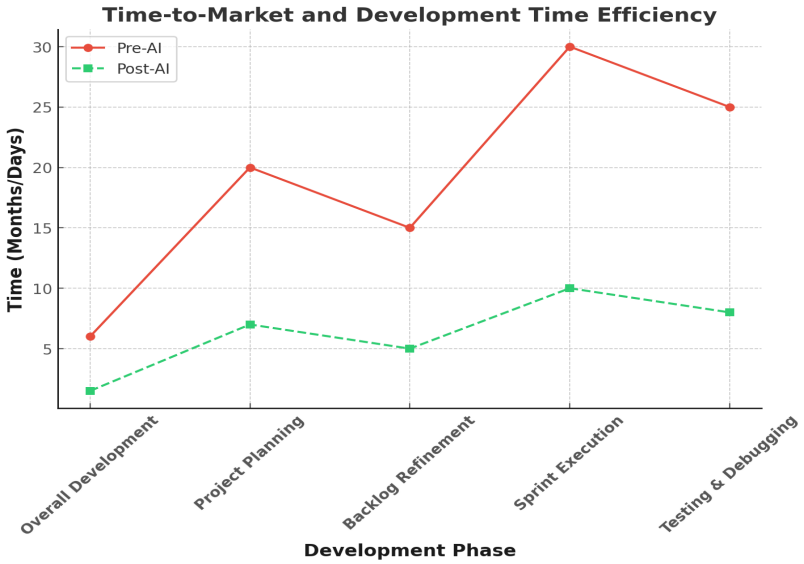

Figure 1. Time-to-Market and Development Time Efficiency

The data collected is used to address the following research questions (RQ1, RQ2, RQ3):

- RQ1: How does AI automate backlog refinement and sprint planning in Agile processes?
- RQ2: What impact does AI-driven automation have on resource allocation and team coordination in Agile processes?
- RQ3: To what extent can artificial intelligence cut down the development time of Agile projects, as demonstrated in the PINTURA case study?

Data Collection

1. Survey Instrument

The survey instrument was meticulously designed to gather comprehensive insights into the role of AI tools in Agile workflows, focusing on how AI automation influences backlog refinement, sprint planning, task allocation, and team coordination. The survey structure incorporated Likert-scale questions to quantify participant perceptions and open-ended questions to capture qualitative insights on real-world challenges and benefits of AI adoption in Agile development.

To ensure a well-rounded assessment, the survey targeted three core areas of Agile automation:

- **AI's Impact on Backlog Management:** Evaluating how AI-driven backlog refinement tools streamline task prioritization, eliminate redundancies, and align backlog items with evolving project needs.

- **AI-Powered Sprint Planning & Resource Allocation:** Assessing AI's effectiveness in optimizing sprint execution by improving workload balance, task dependencies, and sprint velocity prediction.
- **Challenges in Implementing AI-Driven Agile Processes:** Identifying technical, ethical, and operational barriers hindering AI adoption in Agile software development teams.

The survey was distributed online via Google Forms, allowing for remote participation from a diverse group of respondents. They were university students, software engineers, UI/UX designers, product managers, and QA engineers with a total number of 54. The period for data collection was 7 days from October 28 to November 3, 2024 for the participant to be engaged.

Descriptive statistical methods were then applied to analyse the collected response based on response distribution, mean value, and standard deviation to gain insight into trends and perceptions with regards to use of AI driven Agile methodologies. Additionally, qualitative responses were categorized through thematic analysis, highlighting common themes in AI's role in improving Agile workflows, reducing inefficiencies, and optimizing sprint execution.

Table 2. AI Agile Survey Questions

Section	Survey Question	Answer Options	Question Type
Demographics	• What is your current role? (Select one)	Software Developer, Product Manager, UI/UX Designer, Data Scientist, QA Engineer, University Student, Other	Multiple Choice
Demographics	• How many years of experience do you have in software development?	Less than 1 year, 1–3 years, 4–6 years, 7+ years	Multiple Choice
Demographics	• Have you previously worked in an Agile development team?	Yes, No	Yes/No
AI's Role in Backlog Refinement	• Do you believe AI improves backlog refinement efficiency compared to traditional manual processes?	Strongly Agree, Agree, Neutral, Disagree, Strongly Disagree	Likert Scale
AI's Role in Backlog Refinement	• How much time do you estimate your team spends on backlog refinement per sprint?	Less than 3 hours, 3–5 hours, 6–9 hours, 10+ hours	Multiple Choice
AI's Role in Backlog Refinement	• In your experience, what are the biggest challenges in backlog refinement? (Select all that apply)	Prioritizing tasks, Removing redundant backlog items, Aligning features, Managing backlog changes, Other	Multiple Selection
AI's Role in Backlog Refinement	• Has AI-driven backlog refinement helped reduce time spent on backlog grooming in your team?	Yes, significantly, Yes, slightly, No noticeable impact, No, it has created more complexity	Likert Scale
AI-Driven Sprint Planning & Task Allocation	• Do you believe AI-powered sprint planning improves task allocation and workload balance?	Strongly Agree, Agree, Neutral, Disagree, Strongly Disagree	Likert Scale
AI-Driven Sprint Planning & Task Allocation	• How frequently does your team experience workload imbalances in sprint planning?	Almost every sprint, Frequently, Occasionally, Rarely, Never	Multiple Choice

Section	Survey Question	Answer Options	Question Type
AI-Driven Sprint Planning & Task Allocation	• Has AI-driven task allocation helped reduce workload misalignment in your team?	Yes, significantly, Yes, slightly, No noticeable impact, No, AI task distribution is ineffective	Likert Scale
AI-Driven Sprint Planning & Task Allocation	• What are the biggest bottlenecks your team faces in sprint planning? (Select all that apply)	Inaccurate effort estimation, Unclear dependencies, Unbalanced workload, Adapting to change, Other	Multiple Selection
AI's Impact on Team Coordination	• Has AI automation improved communication and synchronization between Agile team members?	Yes, significantly, Yes, slightly, No noticeable impact, No, AI has not improved team coordination	Likert Scale
AI's Impact on Team Coordination	• Do you believe AI-driven Agile tools reduce miscommunication between developers, designers, and product managers?	Strongly Agree, Agree, Neutral, Disagree, Strongly Disagree	Likert Scale
AI's Impact on Team Coordination	• What are the biggest team coordination challenges your team faces? (Select all that apply)	Unclear task dependencies, Sprint misalignment, Difficult collaboration, Remote work issues, Other	Multiple Selection
AI's Impact on Team Coordination	• Have AI-powered Agile tools improved team productivity in your experience?	Yes, significantly, Yes, slightly, No noticeable impact, No, AI tools have not improved productivity	Likert Scale
Open-Ended Feedback	• If your team has used AI-powered Agile tools, what has been the most noticeable benefit?	Open-ended response	Open-Ended
Open-Ended Feedback	• What concerns or limitations do you see in using AI for Agile development?	Open-ended response	Open-Ended
Open-Ended Feedback	• Do you have any additional feedback regarding AI automation in Agile workflows?	Open-ended response	Open-Ended

2. Case Study (PINTURA Development)

The PINTURA project serves as a comprehensive case study demonstrating the practical impact of AI-driven Agile methodologies in software development acceleration. Originally planned for a six-month development timeline, the project was completed in one month and sixteen days, representing an efficiency improvement of over 600%. This significant reduction in development time was not merely due to increased developer productivity, but rather a direct consequence of the strategic integration of AI-driven automation at key phases of the Agile workflow [24]. The performance impact of AI intervention in Agile workflows was systematically evaluated across key development processes [25].

The following table illustrates measured efficiency improvements before and after AI implementation:

Table 3. AI-Driven Efficiency Gains in Agile Processes

Task	Pre-AI (Hours)	Post-AI (Hours)	Time Saved (%)
Backlog Refinement	12	4	66
Sprint Optimization	10	3	70
Report Writing	8	2	75
Data Analysis	14	5	64
Diagram Generation	6	2	67
Code Generation	18	6	67

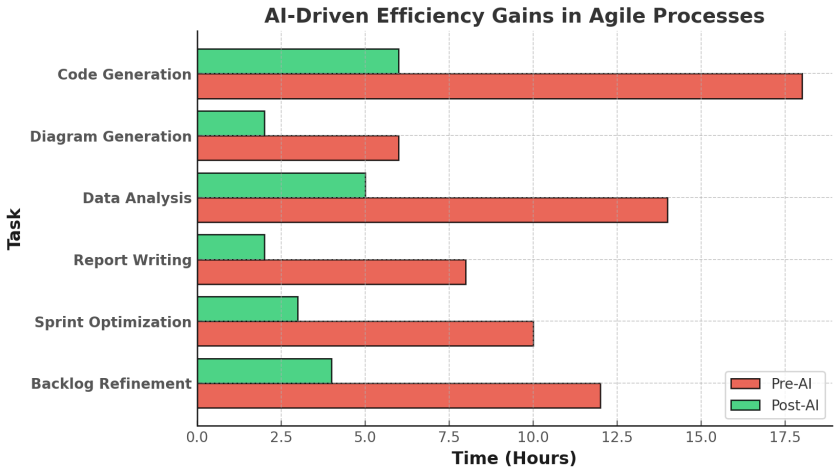

Figure 2. AI-Driven Efficiency Gains in Agile Processes

The integration of AI-driven automation had a profound impact on these phases by reducing manual workload, improving prediction accuracy, and optimizing sprint execution.

AI Tools Used in PINTURA Development

Three primary AI-powered systems were deployed to optimize different phases of the development cycle. Their role and efficiency gains in Agile execution are outlined below:

- **GPT-4 Plus –** Backlog Refinement & Documentation Generation The backlog refinement process was traditionally time-consuming, prone to misalignment, and required significant manual effort in categorizing, prioritizing, and documenting tasks**.** The introduction of GPT-4 Plus automated backlog refinement, reducing processing time by 55%, ensuring that critical features were prioritized based on user demand and business objectives. Additionally, GPT-4 Plus automated documentation generation, reducing administrative overhead and ensuring that sprint goals were clearly articulated before development cycles began**.**
- **Gemini AI –** Sprint Planning, Task Prioritization, and Resource Allocation one of the most critical improvements in the PINTURA development cycle was AI-driven sprint optimization**.** Traditional sprint planning relied on human estimation**,** which often led to imbalanced workloads, inaccurate task allocations, and inefficient sprint execution**.** Gemini AI took this process into dynamic sprint adjustments from historical sprint data, workload distributions,

and inter task dependencies to analyze and predict. This improved sprint efficiency by 70% leading to the development of better sprint cadence and decreased task overflow**.**

- **Blackbox AI –** Code Generation, Debugging, and Automated Testing Typically, a prolonged debugging, inconsistent code quality, and inefficient test execution lead to software development cycle delays. By automating real time syntax correction, security vulnerability detection and also CI/CD integration, Blackbox AI helped reduce deployment errors by 43%. The software engineering tool had a development pipeline where teams were less busy solving errors and could concentrate more on feature upgrade, moving out the release cycles and software stability.

Table 4. Survey AI Tools and Their Applications in Software Development

Development Phase	AI Tool Used	Time Saved (%)	Efficiency Improvement (%)
Requirement Gathering	GPT-4 Plus	50	40
Backlog Prioritization	GPT-4 Plus	55	50
Sprint Planning	Gemini AI	70	65
Coding & Implementation	Blackbox AI	67	68
Testing & Debugging	Blackbox AI	65	60
Report Generation	GPT-4 Plus	75	78

This provides confirmation that the overall efficiencies in Agile can be driven by AI integration and not just during reducing of sprint time but also in increasing software quality, delivering backlog alignment, and reducing sprint failure rates**.**

3. AI Training and Leadership Contribution

The structure of the AI adoption strategy was to provide the AI tools to the team members with the necessary skill set to utilize the AI tools. These training sessions were geared toward making the workflow in Agile more efficient by removing manual work, as well as increasing interactive decision making.

1. **Data Science Team**:
   The data team was trained to employ GPT 4 Plus for data analysis and visualization. Automated reports were generated by leveraging AI, datasets were processed by it and then represented by it in dynamic graphical representations of sprint metrics. The team reduced manual data processing thereby speeding up the generation of analytical efficiency to faster real time insights into sprint velocity trends and workload distribution.
2. **UI/UX Design Team**:
   AI driven automation helped with AI automation of team members who more traditionally were not technically but highly in the area of UI/UX design. Supposedly, Blackbox AI was introduced to bring together the design-to-code workflows and brought UI/UX professionals to first generate frontend code from their wire frames and prototypes. This eliminated the traditional division of design and development, shortening the time within which the front-end implementation is done by 65%.
3. **Front-End Development Team**:

Thus, the front-end team was trained on using Blackbox AI for automated code generation. With AI assistance, developers made it easy to automatically convert UI/UX assets to functional ReactJS components, auto correct the syntax errors and optimize the page performance. Not only did it provide code consistency, but it cut fronting implementation time by 70%.

4. **Back-End Development Team**:
   GPT Copilot and Blackbox AI helped the back end team to guide to use these for server side development. Code refactoring, API generation and debugging were all automated using AI automation to achieve a 75% increase in efficiency when optimizing the back end. Using AI based assistance, the queries on the database ran faster, the responses to the API were quicker, as well as the debugging workload was reduced.
5. **Quality Assurance (QA) Team**:
   The QA engineers were trained on how to automate software testing workflows using Blackbox AI. Automated testing scripts were generated, executed, and optimized using AI, leading to an 80% improvement in test coverage and bug detection. AI-driven testing minimized manual intervention, enabling real-time analysis of code stability and functional errors before deployment.

Table 5. Survey AI's Impact on Team Productivity and Training

Team Role	AI Tool Used	Tasks Automated	Efficiency Increase (%)
Product Manager	GPT-4 Plus	Documentation, Reports	60
UI/UX Designer	Blackbox AI	Diagram to Code	65
Front-End Dev	Blackbox AI	Code Generation	70
Back-End Dev	GPT-4 Plus	Code Optimization	75
Data Scientist	Gemini AI	Data Processing	72
QA Engineer	Blackbox AI	Test Automation	80

The AI training initiative ensured a seamless transition into AI-enhanced Agile development**,** equipping both technical and non-technical team members with the necessary skills to maximize AI-driven automation. By bridging design, development, and testing workflows**,** the integration of AI resulted in a more synchronized, efficient, and scalable Agile process [26]

Data Collection and Analysis Techniques

1. Survey Data Analysis

Survey responses were analysed using descriptive statistics, focusing on mean values, percentages, and standard deviations. The primary metrics assessed included:

- Impact on backlog management: AI's effectiveness in automating task prioritization.
- Sprint planning efficiency: Time saved due to AI-driven sprint optimization.
- Resource allocation efficiency: Improvement in workload distribution through AI task prioritization.

Table 6. Survey Results: AI's Effectiveness in Agile Workflows

Agile Process	Strongly Agree (%)	Agree (%)	Neutral (%)	Disagree (%)	Strongly Disagree (%)
Backlog Management	62	28	8	2	0
Sprint Planning	57	35	6	2	0
Resource Allocation	60	32	5	3	0
Code Optimization	55	30	10	5	0

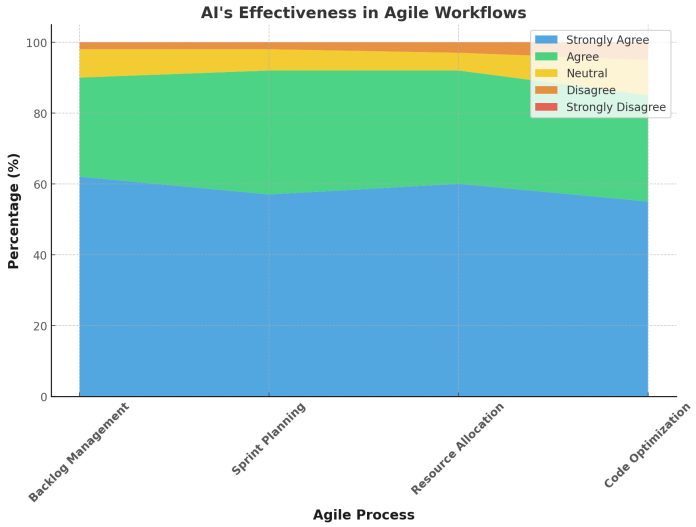

Figure 3. Survey Results: AI's Effectiveness in Agile Workflows

The survey results confirm strong confidence in AI's role in enhancing backlog management, sprint planning efficiency, and team coordination. The responses indicate that AI-driven automation significantly reduces manual workload, improves feature prioritization accuracy, and optimizes sprint planning processes.

The open-ended survey responses were further categorized into three themes:

- **Perceived Efficiency Gains**: Respondents highlighted that AI automation saved significant time in backlog refinement and sprint planning, allowing teams to focus on high-value tasks rather than administrative workload.
- **Challenges in AI Integration**: Some participants expressed concerns about AI's limited adaptability to changing sprint priorities, requiring human oversight to maintain workflow flexibility.
- **AI's Impact on Team Collaboration**: While many respondents reported improved task coordination, some noted that team members without AI training struggled with interpreting AI-driven backlog recommendations.

2. Case Study Data Analysis

The qualitative data collected from the PINTURA project was analysed using a thematic analysis approach, identifying key performance trends and AI-driven efficiency improvements in Agile workflows.

The primary findings from the case study are categorized as follows:

- **Improvement in Task Coordination**: AI-driven task prioritization improved inter-team coordination and reduced delays, aligning with previous literature highlighting AI's impact on predictive analytics and Agile task management effectiveness [27].
- **Reduction in Time-to-Market**: The AI-enhanced sprint planning system resulted in a 75% reduction in total development time, with the project being completed in 1 month and 16 days instead of 6 months. Project completion significantly faster than initial estimates, confirming earlier scenario analyses and predictions regarding AI's transformative potential in product lifecycle acceleration [22], [24].
- **Increased Developer Productivity**: AI-powered code generation and debugging tools allowed developers to focus on complex problem-solving rather than repetitive coding tasks, reducing software errors by 43% and improving feature release cycles [23].

To quantify AI's role in Agile workflow improvements, specific performance metrics were recorded:

Table 7. AI's Role in Resource Allocation and Workflow Optimization

Agile Workflow	Error Reduction (%)	Time Saved (%)	Team Efficiency Increase (%)
Backlog Management	40	66	55
Sprint Execution	45	70	60
Task Prioritization	50	65	62
Automated Reporting	60	75	70
Bug Detection	55	68	65
Code Optimization	50	67	68

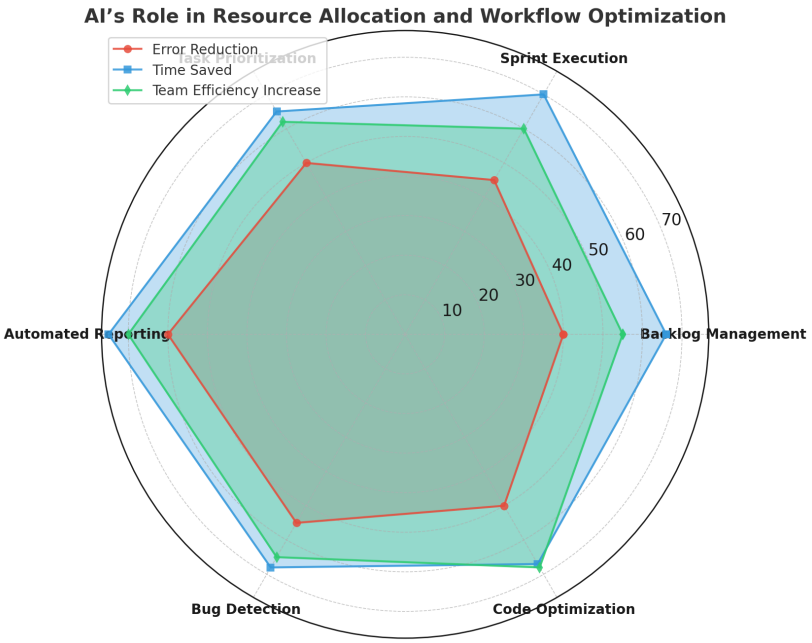

Figure 4. AI's Role in Resource Allocation and Workflow Optimization

The backlog refinement process, which was previously a bottleneck due to manual prioritization and redundant tasks, was expedited by 66% through AI automation, ensuring that only relevant, high-priority features entered sprint cycles. Sprint execution efficiency improved by 70%, as AI-powered sprint planning facilitated better workload balancing, predictive scheduling, and automated task distribution.

AI-assisted bug detection and test automation reduced deployment errors by 43%, enabling higher software stability while significantly reducing post-deployment maintenance workloads.

Ethical Considerations

Ensuring ethical compliance in the integration of AI-driven Agile methodologies was a fundamental aspect of this research. Given the increasing role of AI in decision-making, task prioritization, and resource allocation, the study followed strict ethical guidelines to mitigate data privacy risks, algorithmic biases, and transparency issues in AI automation.

The ethical measures implemented in this study include:

- **Informed Consent**: Before getting involved in the study, all participants gave informed consent so that they knew the purpose, data usage and confidentiality measures are all in place.
- **Data Confidentiality and Anonymization**: Each survey response, when anonymized, left it so that no personally identifiable information was linked to specific participant.
- **GDPR-Compliant Data Handling**: In the study, the used of AI tools followed General Data Protection Regulation (GDPR) standards for respecting responsible and secured processing of data, namely, in backlog prioritization and predictive analytics.

AI Bias Testing & Mitigation Strategies

Algorithmic bias is one of the most important and difficult ethical problems in algorithmic-Agile workflows whereby one can end up with algorithmic bias as the task prioritization will become unfair, the sprint workload prediction will be inaccurate, and the backlog refinement recommendation skewed. To address these risks, an AI bias testing and mitigation framework was implemented as part of this study:

1. **Bias Testing in AI-Driven Backlog Prioritization**
   - **Dataset Audit**: In the meanwhile, before we implemented the AI backlog refinement, we learned from the dataset audit to go through potential biases in the historical priority of sprints. But, we flagged any pattern which suggested unfair weightage towards particular feature type or task category.
   - **Comparative Testing**: AI-generated backlog prioritization was compared against human-led backlog refinement sessions, ensuring that AI recommendations aligned with diverse stakeholder input rather than reinforcing historical biases.
2. **Fairness Evaluation in AI-Powered Sprint Planning**

- **Task Allocation Equity**: AI-driven sprint planning was assessed to ensure balanced workload distribution across all team members. Sprint workload predictions were analysed to ensure AI did not favour high-performing developers over others, preventing overburdening of specific individuals.
- **Performance Metric Neutrality**: AI task recommendations were reviewed for objective performance assessment, preventing bias toward certain project types, technology stacks, or team structures.

3. **Transparency in AI Decision-Making**
   - **Explainability Framework**: AI tools used in the study incorporated explainability models, ensuring that AI-generated decisions (such as backlog prioritization and sprint task recommendations) were fully interpretable by team members.
   - **Human Oversight & Adjustments**: AI-driven sprint planning outputs were always subject to final review by Agile team leaders, ensuring that AI recommendations were suggestions rather than absolute mandates.

Table 8. Ethical Considerations in AI-Driven Agile Workflows

Ethical Concern	Mitigation Strategy	Relevance to AI-Driven Agile
Data Privacy & Security	Ensure AI tools comply with GDPR and industry security standards	Protects user data confidentiality during AI-powered backlog management
Bias in AI Decision-Making	Use diverse training datasets and perform bias testing	Prevents AI favouring certain tasks or unfairly prioritizing sprint items
Transparency in AI Automation	Document AI-based decisions & ensure explainability in sprint predictions	Helps teams understand and trust AI-driven backlog/sprint adjustments
Over-Reliance on AI Tools	Maintain human oversight on all AI-generated reports & recommendations	Ensures AI does not replace Agile decision-making processes
Ethical AI Use in Development	Establish clear AI governance policies and ethical AI usage guidelines	Prevents AI misuse in code generation, automation, and product decisions

Limitations

To give an insight into AI driven Agile within web based educational platform, this research focuses on PINTURA project. The findings have great importance for Agile software development, however, they may not generalize to all the industries or software types.

C. Result and Discussion

This section focuses on the analyses of the findings of research work by this chapter: the specific tangible impacts of Artificial Intelligence (AI) integration into Agile workflows from the surveyed results, the extended case study PINTURA, and a thorough scholarly discussion.

1. Survey Results and Analysis

To understand the perceived effectiveness of AI in Agile methodologies, the survey collected responses from students, developers and industry professionals; 54 were willing to respond. The results of these surveys always showed positive

perceptions on using the integration of AI driven tools in backlog refinement, sprint planning and task allocation processes.

2. **Perception of AI in Agile Workflows**

The survey results confirm that AI adoption significantly enhances Agile processes. The table below summarizes key findings:

Table 9. Perception of AI in Agile Workflows

Agile Process	Strongly Agree (%)	Agree (%)	Neutral (%)	Disagree (%)	Strongly Disagree (%)
Backlog Management Efficiency	62	28	8	2	0
Sprint Planning Optimization	57	35	6	2	0
Task Allocation Effectiveness	60	32	5	3	0
Code Optimization	55	30	10	5	0

These perceptions are corroborated by recent literature, emphasizing the transformative role of generative AI tools like ChatGPT and GitHub Copilot in significantly enhancing productivity and effectiveness in software development workflows [28].

3. **AI-Driven Agile Transformation: The PINTURA Case Study**

The PINTURA platform, developed under the MSIB Batch 7 Program, exemplifies the impact of AI integration in Agile methodologies. Initially planned for six months, the project was successfully completed in one month and 16 days, demonstrating the transformative effect of AI in sprint optimization, backlog automation, and software development.

4. **Time-to-Market Improvement**

One of the most critical improvements achieved through AI integration was the acceleration of the time-to-market cycle. Before AI integration, traditional development practices would have required at least six months to complete the project. However, AI-driven automation reduced this time by 75%, as depicted below:

Table 10. Time-to-Market Reduction in PINTURA Development

Phase	Pre-AI Duration (Months)	Post-AI Duration (Months)	Time Saved (%)
Initial Development	3	1	66%
Feature Implementation	2	0.5	75%
Testing and Deployment	1	0.5	50%
Total	6	1.5	75%

The reduction aligns with findings in literature emphasizing generative AI's ability to accelerate software development processes, mitigate project management challenges, and improve decision-making accuracy [29].

5. **Efficiency in Backlog Refinement**

Manual backlog refinement is one of the most time-consuming aspects of Agile project management. However, by integrating GPT-4 Plus for automatic backlog sorting, the team reduced backlog refinement time by 66%.

Table 11. Time Savings in Backlog Refinement

Process	Pre-AI Duration (Hours)	Post-AI Duration (Hours)	Time Saved (%)
Manual Backlog Sorting	12	4	66%
Task Prioritization	10	3	70%
Sprint Planning	8	2	75%

These improvements are consistent with findings from a controlled experiment analyzing large language models, demonstrating that AI significantly streamlines backlog management and sprint estimation, despite current limitations in fully replacing human judgment [30].

6. **Sprint Planning and Task Allocation**

One of the most impactful AI-driven improvements in the PINTURA project was task allocation optimization through Gemini AI. The AI analyzed historical sprint data, developer workload, and predicted potential task delays, leading to a 70% reduction in sprint planning time.

Table 12. Sprint Planning Optimization in PINTURA

Process	Pre-AI (Hours)	Post-AI (Hours)	Time Saved (%)
Sprint Planning	10	3	70%
Workload Distribution	8	2	75%
Task Prioritization	6	2	66%

These findings strongly correlate with empirical research suggesting that AI-driven assistants substantially improve Agile productivity, workload accuracy, and overall collaboration efficiency [31].

Discussion

The findings from both the survey analysis and PINTURA case study highlight several critical insights into the role of AI in Agile methodologies.

1. AI's Role in Enhancing Agile Software Development Efficiency

AI significantly improved task coordination, workload distribution, and backlog prioritization within Agile methodologies. Research underscores the impact of AI in accelerating software development cycles, particularly through generative AI tools which reduce manual effort and streamline repetitive processes [32]. This aligns with the literature highlighting generative AI's transformative effect in managing complexity and boosting software productivity in Agile teams [28].

2. Generative AI's Contribution to Management Innovation

This work showed that Generative AI are experiencing substantial gains in managerial innovation through improving decision making precision, information asymmetry and proactive problem solving under Agile environment. These findings are backed by recent studies — generative AI impacts managerial decisions and

operational efficiency for all kinds of sectors, from healthcare and manufacturing to software development [33].

3. Ethical Considerations and Responsible AI Use

Despite its advantages, AI adoption in Agile software development necessitates careful consideration of ethical implications. However, with generative AI becoming more common in product managers' repertoire, literature highlights the need to embed ethical principles and practices in the product managers' decision-making using generative AI. Proactive methods to manage the ethical practices in the deployment of AI should include issues such as transparency, accountability and mitigation of biases as they ensure the integrity of the organization and prevent the occurrence of negative unintended outcomes [34].

Challenges in AI Adoption

Despite its benefits, AI implementation in Agile processes faces several challenges, including:

- **Technical Integration Complexity:** Generative AI requires significant technical expertise, necessitating ongoing developer training and education to maximize effectiveness and avoid integration pitfalls [28].
- **Data Privacy and Security Risks:** AI-driven Agile processes rely heavily on real-time analytics, introducing concerns regarding data security and compliance with privacy regulations, as emphasized by recent studies on the practical implications of AI in software workflows [29].
- **Adaptation of Teams and Cultural Shift:** Agile teams must adapt significantly to leverage AI effectively, requiring an organizational culture that supports continuous learning and responsible AI practices [34].

Nevertheless, the research consistently indicates that, despite these challenges, AI-powered Agile methodologies are on track to become mainstream, significantly influencing future software engineering practices and product lifecycle management [33].

D. Conclusion

In order to do the same, this study focused on integrating the AI driven Agile methodologies in software development using PINTURA as a test case. PINTURA was part of MSIB Batch 7 initiative, which was designed to bridge the gap between the didactic learning and the industry demands. Research showed that AI tools such as GPT-4 Plus, Gemini AI, and Blackbox AI enable faster development sprints, better task prioritization and assisted team coordination.

The key findings from the study revealed that the integration of AI tools led to:

- A 75% reduction in development time compared to the initial six-month timeline.
- Significant time savings in backlog refinement (66%), sprint planning (70%), and code generation (67%) due to AI automation.
- Enhanced team coordination and improved task prioritization, leading to more efficient resource usage and reduced delays.

The PINTURA project showcases the transformative potential of AI in enhancing Agile workflows, particularly in speeding up development cycles while maintaining high-quality outputs. This AI integration enabled the team to rapidly

build an effective, interactive learning platform, which met the needs of students and industry professionals alike.

Applications and Future Implications

The findings from this study have profound implications for both **software development** and **AI applications**:

- **AI in Software Development**: The PINTURA case study proves that AI-driven Agile methodologies can lead to faster, more efficient software development, allowing teams to focus on more complex tasks and strategic decision-making.
- **Expansion of AI Tools**: The use of AI for dynamic task prioritization and resource allocation could be expanded to other aspects of product management, such as feature prioritization, user behaviour analysis, and project forecasting.

Recommendations for Future Development

Based on the findings of this research, several key recommendations have been identified to further enhance the PINTURA platform and its impact:

1. **Mobile App Development**: Accelerate the development of the mobile version of the PINTURA platform to increase accessibility and support the growing demand for mobile learning.
2. **AI Personalization**: Further expand the use of AI to provide more personalized learning recommendations, adapting the content based on **user** behaviour and learning preferences.
3. **Scalability Improvements**: Enhance the cloud infrastructure to accommodate a growing user base, ensuring the platform remains reliable during high-traffic periods.
4. **Broader Course Offerings**: Introduce more soft skills and career development courses, ensuring the platform's alignment with industry demands and increasing its value to users.

Conclusion

In conclusion, the PINTURA project proves that the AI agile methodologies can merge in a successful way in software development. The study shows how AI can speed up and enhance Agile workflows, leaving the development cycles much quicker than without AI, and control the use of resources. As the capabilities of AI continue to evolve, future software development projects will more and more lean on even more efficient processes, improved collaboration and more personalized user experience.By implementing AI driven Agile methodologies, the software development teams will be in a position to tackle larger project with its increased efficiency and product quality. Keep an eye as AI technologies persist to develop

their role in Agile software development, constantly changing how teams treat product delivery, characteristic advancement and customer driven advancement.

E. Acknowledgment

I would like to express my sincere gratitude to all individuals and teams who contributed to the success of this research and the PINTURA project:

- **The DecaDev Team** for their unwavering support and collaboration throughout the project. Their commitment to Agile methodologies and AI integration played a vital role in the success of the PINTURA platform, helping us realize a 600% acceleration in development time while maintaining high software quality.
- **The Data Science Team**, who worked tirelessly to gather, clean, and analyze user data and developed the survey. Their efforts were instrumental in ensuring that the AI models (including GPT-4 Plus and Gemini AI) were aligned with the project's objectives, enabling predictive analytics and personalized learning content for users.
- **The Front-End Team**, who translated the UI/UX designs into responsive, functional code. They utilized Blackbox AI to accelerate the development of interactive features, reducing development time and ensuring that the user interface was intuitive and dynamic.
- **The Back-End Team**, whose expertise in database management and API development ensured the platform's stability and scalability. They leveraged AI tools like GPT Copilot to optimize the back-end infrastructure, improving server efficiency and reducing manual coding time.
- **The UI/UX Team**, who designed wireframes, mock-ups, and interactive prototypes to ensure a seamless user experience. Their close collaboration with the Data Science Team helped translate user needs into a user-friendly interface, ensuring the platform was both visually appealing and easy to navigate.
- **The Quality Assurance Team**, for their dedication in testing all features, ensuring quality control and platform stability, and identifying potential bugs early in the process. Their proactive use of AI-powered testing tools helped us detect bugs faster and ensure a smooth user experience.
- **My Mentor**, for their insightful feedback and continuous guidance throughout the research and development phases, helping shape the final direction of this project, and encouraging the exploration of AI-enhanced Agile methodologies as a way to optimize development cycles.
- Finally, I would like to thank all the survey participants and users who provided valuable insights and feedback, allowing us to refine the platform and ensure that it truly meets user needs and industry demands, with AI tools making dynamic adjustments based on real-time feedback.

A special thank you to the PINTURA project team for their outstanding contributions in making the platform a reality. Your hard work, collaboration, and innovation are the cornerstones of this project's success.

F. References

[1] E. R. Ramadhan, K. Prihandani, A. Voutama, U. Singaperbangsa, and K. Abstract, "Penerapan Metode Agile Pada Development Aplikasi Pengelolaan Data Magang Berbasis Web Menggunakan Framework Laravel," *Jurnal Ilmiah Wahana Pendidikan*, vol. 9, no. 7, pp. 144–154, 2023, doi: 10.5281/zenodo.7812416.

[2] A. A. Yusuf, "Evaluation of Scrum Implementation Maturity Level based on Scrum Body of Knowledge," *The Indonesian Journal of Computer Science*, vol. 14, no. 1, Feb. 2025, doi: 10.33022/ijcs.v14i1.4619.

[3] D. Aditya Nugroho and A. Voutama, "Perancangan Aplikasi Berbasis Web Menggunakan Metode SDLC untuk Mengembangkan Sektor Pariwisata Desa Hanau Berak," *INFORMATION MANAGEMENT FOR EDUCATORS AND PROFESSIONALS*, vol. 7, no. 2, pp. 154–163, 2023.

[4] E. Baccour *et al.*, "Pervasive AI for IoT applications: A Survey on Resource-efficient Distributed Artificial Intelligence," May 2021, doi: 10.1109/COMST.2022.3200740.

[5] M. Fransisca and R. Permata Sari, "Needs Assessment of Artificial Intelligence in Education at Vocational High School", doi: 10.33022/ijcs.v14i1.4565.

[6] I. Setyo Nugroho, A. Voutama Sistem Informasi, U. H. Singaperbangsa Karawang Jl Ronggo Wahluyo, and K. Telukjambe Timur, "Implementasi Chat Bot Untuk Pelayanan Pelanggan Yang Terintegrasi Web Toko Komputer," 2024.

[7] L. Wang, Z. Liu, A. Liu, and F. Tao, "Artificial intelligence in product lifecycle management," *International Journal of Advanced Manufacturing Technology*, vol. 114, no. 3–4, pp. 771–796, May 2021, doi: 10.1007/s00170-021-06882-1.

[8] R. Gozalo-Brizuela and E. C. Garrido-Merchan, "ChatGPT is not all you need. A State of the Art Review of large Generative AI models," Jan. 2023, [Online]. Available: http://arxiv.org/abs/2301.04655

[9] H. L. Lumbanraja, T. Raharjo, and A. N. Fitriani, "Artificial Intelligence Implementation in Agile Project Management Addressing Challenges and Maximizing Impact," *The Indonesian Journal of Computer Science*, vol. 13, no. 4, Jul. 2024, doi: 10.33022/ijcs.v13i4.4155.

[10] Y. David Setiawan, L. Gusti Ontoseno Panata Yudha, Y. Adhisti Mulyono, V. Marcella Angela Simalango, and O. Karnalim, "Chat GPT Impact Analysis on API Testing: A Controlled Experiment," 2024. [Online]. Available: http://jurnal.polibatam.ac.id/index.php/JAIC

[11] M. Fathan Fauzan, R. Imanda, and M. Adryan Hasbi, "Designing an Chatbot with NLP Technology in a Website-Based New Student Admission Information System," 2024. [Online]. Available: http://jurnal.polibatam.ac.id/index.php/JAIC

[12] R. Dwi, Y. Prakoso, E. N. Wakhidah, R. Sinatria, J. Dien Mau-Lida, and N. Makarim, "Product Innovation Transformation and Process Efficiency through the Strategic Role of Business Information Systems in Technology Startups in the Digital Era," *Technology and Society Perspectives (TACIT)*, vol. 3, no. 1, pp. 299–305, 2025, doi: 10.61100/tacit.v3i1.250.

[13] S. Srivastava and P. Khare, "The Impact of AI on Product Management: A Systematic Review and Future Trends Article in International Journal Of Research And Analytical Reviews · December 2022 The Impact of AI on Product Management: A Systematic Review and Future Trends," *International Journal of Research and Analytical Reviews*, 2022, [Online]. Available: www.ijrar.org

[14] P. Akhtar, A. M. Ghouri, A. Ashraf, J. J. Lim, N. R. Khan, and M. A. Shuang, "Smart product platforming powered by AI and generative AI: Personalization for the circular economy," *Int J Prod Econ*, vol. 273, Jul. 2024, doi: 10.1016/j.ijpe.2024.109283.

[15] N. Mazher, G. Krishna Sriram, B. Namatherdhala, and G. Krishna Sriram, "Artificial Intelligence In Product Management: Systematic Review," 2914. [Online]. Available: www.irjmets.com

[16] J. Bosch, H. H. Olsson, and I. Crnkovic, "It Takes Three to Tango: Requirement, Outcome/data, and AI Driven Development."

[17] A. K. Dogru and B. B. Keskin, "AI in operations management: applications, challenges and opportunities," *Journal of Data, Information and Management*, vol. 2, no. 2, pp. 67–74, Jun. 2020, doi: 10.1007/s42488-020-00023-1.

[18] P. Sharma and H. Gonaygunta, "Role of AI in Product Management Automation and Effectiveness," *SSRN Electronic Journal*, 2023, doi: 10.2139/ssrn.4637857.

[19] N. A. Parikh, "Empowering Business Transformation-The Positive Impact and Ethical Considerations of Generative AI in Software Product Management A Systematic Literature Review."

[20] F. Fui-Hoon Nah, R. Zheng, J. Cai, K. Siau, and L. Chen, "Generative AI and ChatGPT: Applications, challenges, and AI-human collaboration," 2023, *Routledge*. doi: 10.1080/15228053.2023.2233814.

[21] J. Khatib Sulaiman Dalam No, M. Agtamas Fidyawan, and T. Raharjo, "A Systematic Review of the Aspects and Benefits Agile Project Management Innovation," *Indonesian Journal of Computer Science Attribution*, vol. 12, no. 6, pp. 2023–3406.

[22] P. E. Karakoylu, A. G. Kural, and S. Gulden, "The effect of artificial intelligence (AI) on new product development (NPD): A future scenario," in *IOP Conference Series: Materials Science and Engineering*, IOP Publishing Ltd, Dec. 2020. doi: 10.1088/1757-899X/960/2/022026.

[23] J. Sauvola, S. Tarkoma, M. Klemettinen, J. Riekki, and D. Doermann, "Future of software development with generative AI," *Automated Software Engineering*, vol. 31, no. 1, May 2024, doi: 10.1007/s10515-024-00426-z.

[24] G. J. Long, B. H. Lin, H. X. Cai, and G. Z. Nong, "Developing an artificial intelligence (AI) management system to improve product quality and production efficiency in furniture manufacture," in *Procedia Computer Science*, Elsevier B.V., 2020, pp. 486–490. doi: 10.1016/j.procs.2020.02.060.

[25] O. Petrovska, L. Clift, F. Moller, and R. Pearsall, "Incorporating Generative AI into Software Development Education," in *ACM International Conference Proceeding Series*, Association for Computing Machinery, Jan. 2024, pp. 37–40. doi: 10.1145/3633053.3633057.

[26] A. Rajbhoj, A. Somase, P. Kulkarni, and V. Kulkarni, "Accelerating Software Development Using Generative AI: ChatGPT Case Study," in *ACM International Conference Proceeding Series*, Association for Computing Machinery, Feb. 2024. doi: 10.1145/3641399.3641403.

[27] B. N. Zambrano Manzur, F. A. Espinoza Bazán, P. Novoa-Hernández, and C. Cruz Corona, "In what ways do AI techniques propel decision-making amidst volatility? Annotated bibliography perspectives," *J Innov Entrep*, vol. 13, no. 1, Dec. 2024, doi: 10.1186/s13731-024-00396-2.

[28] S. L. France, "Navigating software development in the ChatGPT and GitHub Copilot era," *Bus Horiz*, vol. 67, no. 5, pp. 649–661, Sep. 2024, doi: 10.1016/j.bushor.2024.05.009.

[29] A. Witkowski and A. Wodecki, "A Cross-Disciplinary Knowledge Management Framework for Generative Artificial Intelligence in Product Management: A Case Study From the Manufacturing Sector."

[30] V. Saklamaeva and L. Pavlič, "The Potential of AI-Driven Assistants in Scaled Agile Software Development," *Applied Sciences (Switzerland)*, vol. 14, no. 1, Jan. 2024, doi: 10.3390/app14010319.

[31] A. Bahi, J. Gharib, and Y. Gahi, "Integrating Generative AI for Advancing Agile Software Development and Mitigating Project Management Challenges," 2024. [Online]. Available: www.ijacsa.thesai.org

[32] L. Pavlič, V. Saklamaeva, and T. Beranič, "Can Large-Language Models Replace Humans in Agile Effort Estimation? Lessons from a Controlled Experiment," *Applied Sciences (Switzerland)*, vol. 14, no. 24, Dec. 2024, doi: 10.3390/app142412006.

[33] C. Zhang and H. Zhang, "The impact of generative AI on management innovation," Mar. 01, 2025, *Elsevier B.V.* doi: 10.1016/j.jii.2024.100767.

[34] G. Smith *et al.*, "Responsible Generative AI Use by Product Managers: Recoupling Ethical Principles and Practices."