

# Revolutionizing Product Development: The Agile UX Revolution

Kiran Kher<sup>1</sup>; Hetvi Kotak<sup>2</sup>  
Institute of Technology, Nirma University

**Abstract:-** This study delves into the fusion of agile methodologies with user experience (UX) design principles aimed at elevating product usability and enhancing customer satisfaction. It tackles prevalent challenges such as misconceptions and communication barriers while offering remedies for fostering effective collaboration. Central to the discussion are key principles of integrating agile with UX, including cross-functional collaboration and iterative design, complemented by case studies drawn from prominent industry players such as Spotify and Google. These cases serve to underscore the tangible benefits of agile UX on product usability and customer satisfaction, with a particular focus on measurement techniques and prospective trajectories within the domain. By juxtaposing practical illustrations with theoretical underpinnings, this study furnishes valuable guidance tailored for both researchers and practitioners seeking to harness agile methodologies for optimizing UX outcomes in software development.

## I. INTRODUCTION

In contemporary software development, the synergy between Agile methodologies and User Experience (UX) design principles is paramount for crafting successful products. Agile methodologies such as Scrum, Kanban, and Extreme Programming underscore iterative development, collaboration, and adaptability, embracing a customer-centric ethos. These methodologies prioritize delivering user value through frequent releases and active feedback solicitation, enabling teams to swiftly adapt to evolving requirements. Conversely, UX design principles center on crafting intuitive, usable, and delightful experiences for users. Leveraging techniques like user research, prototyping, and usability testing, UX designers ensure products align with user needs and expectations.

Integrating Agile methodologies with UX design principles empowers teams to develop products that not only fulfill functional requirements but also deliver meaningful and satisfying user experiences, driving market success. This integration ensures that development processes prioritize user-centered design, fostering collaboration among cross-functional teams. Through iterative design, development, and refinement informed by real user feedback, teams create technically robust products that meet evolving user expectations. This collaborative approach enhances product quality, elevates customer satisfaction, and cultivates loyalty. Ultimately, the integration of Agile and UX methodologies enables teams to deliver value more effectively, securing business success and competitive advantage in today's dynamic market landscape.

This paper's objectives entail a comprehensive exploration of the integration between Agile methodologies and UX design principles in software development. It seeks to tackle prevalent challenges like misconceptions and communication barriers while proposing solutions to foster collaboration between development and design teams. By examining key principles of Agile UX integration, such as cross-functional collaboration and iterative design, alongside real-world case studies from industry leaders like Spotify and Google, the paper aims to underscore the positive impact on product usability and customer satisfaction. Additionally, it aims to highlight measurement techniques for assessing this impact and offer insights into future directions in the field. Through these objectives, the paper aims to provide valuable guidance for researchers and practitioners aiming to leverage Agile methodologies for enhanced UX outcomes in software development.

## II. AGILE METHODOLOGIES AND UX: UNDERSTANDING THE BASICS



Fig 1 Agile Methodologies

### ➤ *Scrum:*

Scrum stands out as a framework within Agile software development, highlighting collaboration, responsibility, and incremental advancement. Central to its methodology are brief, time-limited phases referred to as sprints, generally spanning 1-4 weeks. The framework assigns distinct roles to team members, notably the Product Owner, Scrum Master, and Development Team. Critical components encompass the Product Backlog, Sprint Backlog, and Increment. Key events include daily stand-up meetings, sprint planning sessions, sprint reviews, and sprint retrospectives.

### ➤ *Kanban:*

Kanban serves as a visual system for work management, frequently employed in Agile software development. Its core principles revolve around visualizing workflow, constraining work in progress (WIP), and perpetually refining efficiency. Tasks are depicted as cards on a Kanban board, traversing stages like "To Do," "In Progress," and "Done." Unlike Scrum, Kanban prioritizes adaptability and responsiveness, devoid of fixed roles or time-bound iterations.

### ➤ *Extreme Programming (XP):*

Extreme Programming (XP) represents an Agile methodology placing significant emphasis on engineering practices aimed at enhancing software quality and adaptability to evolving requirements. Key practices within XP encompass Test-Driven Development (TDD), pair programming, continuous integration, and frequent deployments. Central to XP is the principle of customer engagement, facilitating regular feedback loops and the presence of on-site customer representatives.

### ➤ *Lean Software Development:*

Lean Software Development derives its principles from lean manufacturing, aiming to reduce waste, enhance efficiency, and amplify customer value. It underscores practices like value stream mapping, diminishing batch sizes, and granting teams autonomy in decision-making. Lean principles prioritize delivering optimal value with minimal effort, all while fostering continuous process enhancement.

### ➤ *Feature-Driven Development (FDD):*

Feature-Driven Development (FDD) stands as an Agile methodology centered around the incremental construction of software features. It encompasses steps such as compiling a feature inventory, strategizing, designing, constructing, and scrutinizing features within concise iterations. FDD places particular emphasis on domain modeling, feature proprietorship, and recurrent progress updates.

### ➤ *Fundamental Principles of UX (User Experience) Design:*

#### • *User-Centric Approach:*

Prioritize understanding and addressing user needs, behaviors, and goals throughout the design process to create effective solutions.

#### • *Usability:*

Strive to make the product easy to use and navigate, employing clear navigation, consistent layout, and intuitive interactions.

- **Accessibility:**  
Ensure inclusivity by designing products that accommodate users with diverse abilities, considering factors like color contrast, text size, and keyboard navigation.
- **Consistency:**  
Maintain uniformity in design elements, interactions, and terminology to facilitate user navigation and comprehension.
- **Simplicity:**  
Aim for simplicity by eliminating unnecessary elements and minimizing cognitive load, facilitating user task completion.
- **Feedback and Responsiveness:**  
Provide timely feedback to users for their actions and interactions, enhancing user understanding and engagement.

- **Visual Hierarchy:**  
Utilize visual cues such as color, typography, and spacing to prioritize information and guide user attention effectively.
- **Emotional Design:**  
Consider the emotional impact of the product on users, incorporating elements like aesthetics and tone to enhance the overall user experience.
- **Iterative Design Process:**  
Embrace an iterative approach to design, incorporating continuous testing and feedback to refine and improve the user experience over time.
- **Context Sensitivity:**  
Design experiences that adapt to user context, including device type, location, and preferences, to deliver personalized and relevant interactions.

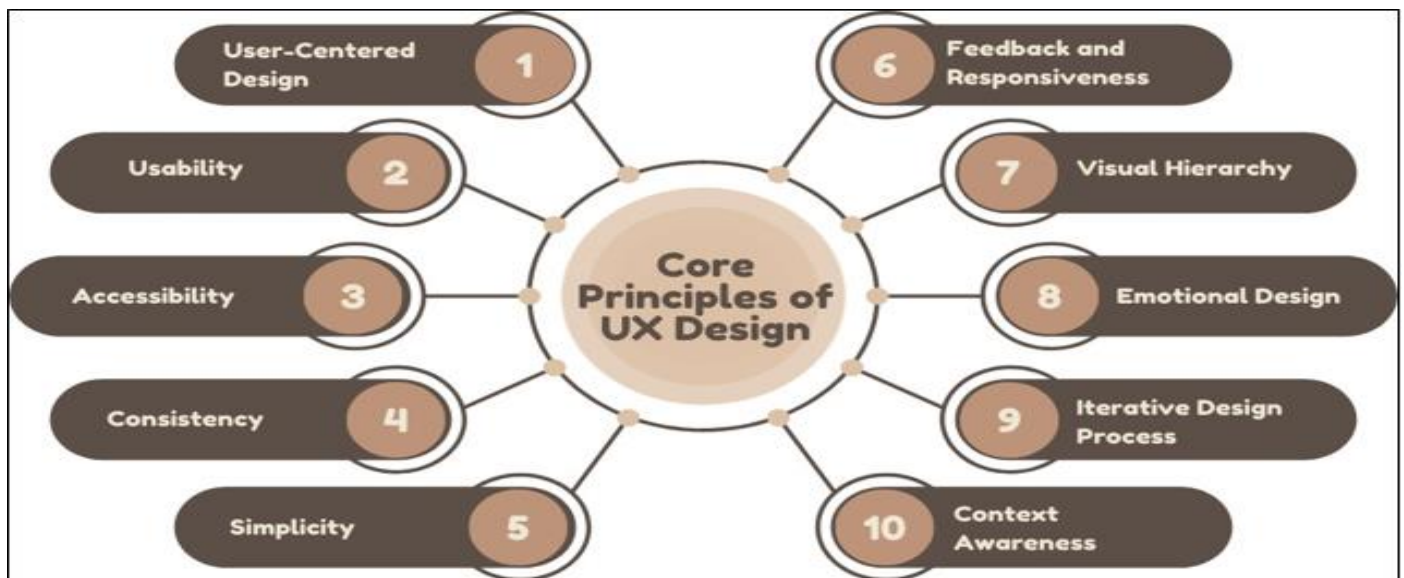


Fig 2 Core Principles of Agile UX Design

### III. THE CONVERGENCE OF AGILE AND UX

The convergence of Agile and UX occurs where their principles and methodologies intersect or complement each other.

Both Agile and UX stress the importance of iterative development. Agile divides projects into manageable increments (sprints), while UX promotes continuous iteration based on user feedback. This shared emphasis allows for swift prototyping, testing, and refinement of user experiences throughout the development cycle. Agile fosters cross-functional collaboration among developers, designers, product managers, and stakeholders. Likewise, UX design thrives on collaboration between designers, researchers, developers, and other stakeholders to address user needs effectively. By closely working together, Agile teams seamlessly integrate UX design activities into their development processes. Both methodologies prioritize user needs and preferences. Agile teams capture user

requirements through user stories and prioritize features based on user value. UX design entails understanding user behaviors, goals, and pain points to craft intuitive and valuable experiences. By aligning user-centric principles, Agile teams ensure their products effectively meet user needs. Agile methodologies promote continuous feedback loops through practices like sprint reviews, daily stand-ups, and retrospectives. Similarly, UX design relies on feedback from usability testing, user research, and analytics to refine the user experience. Integrating user feedback into each iteration allows Agile teams to validate design decisions and make necessary adjustments early in the development process. Both Agile and UX advocate for an empirical approach to development, prioritizing data and evidence over assumptions. Agile teams leverage metrics such as velocity, burn-down charts, and customer satisfaction to gauge progress and make informed decisions. UX designers collect data through user research, usability testing, and analytics to understand user behavior and inform design choices.

**IV. INTEGRATING SCRUM WITH LEAN UX**

➤ *Using Gamification*

The Agile team begins by creating a list of hypotheses regarding the needs of potential users. These assumptions are refined through discussions with the product owner (PO) during product back- log refinement sessions. This initial step sets the stage for the team to align their efforts with user- centric objectives. As the sprint commences, the team convenes for a design studio session. Here, they select one of the hypotheses as the sprint’s guiding theme. This hypothesis serves as the focal point for the team’s collaborative efforts as they sketch and discuss design ideas aimed at addressing user needs. Following the design studio session, the team transitions seamlessly into the sprint planning meeting. In addition to selecting user stories for

development, the team allocates time and resources for the weekly user experiment—a critical component of the Lean UX process. During the weekly user experiment, the team presents a version of their developing product, often referred to as a Minimum Viable Product (MVP), to users for feedback. This feedback serves as valuable input for refining the product and validating assumptions, ensuring that the team remains aligned with user needs and preferences. Throughout the sprint, the team is incentivized to apply Lean UX tactics through gamification elements. Each Lean UX tactic implemented earns the team rewards, with additional incentives offered for collaborative efforts. Special rewards are also provided for employing Lean UX tactics for the first time and for successfully addressing specific UX challenges identified by the team.

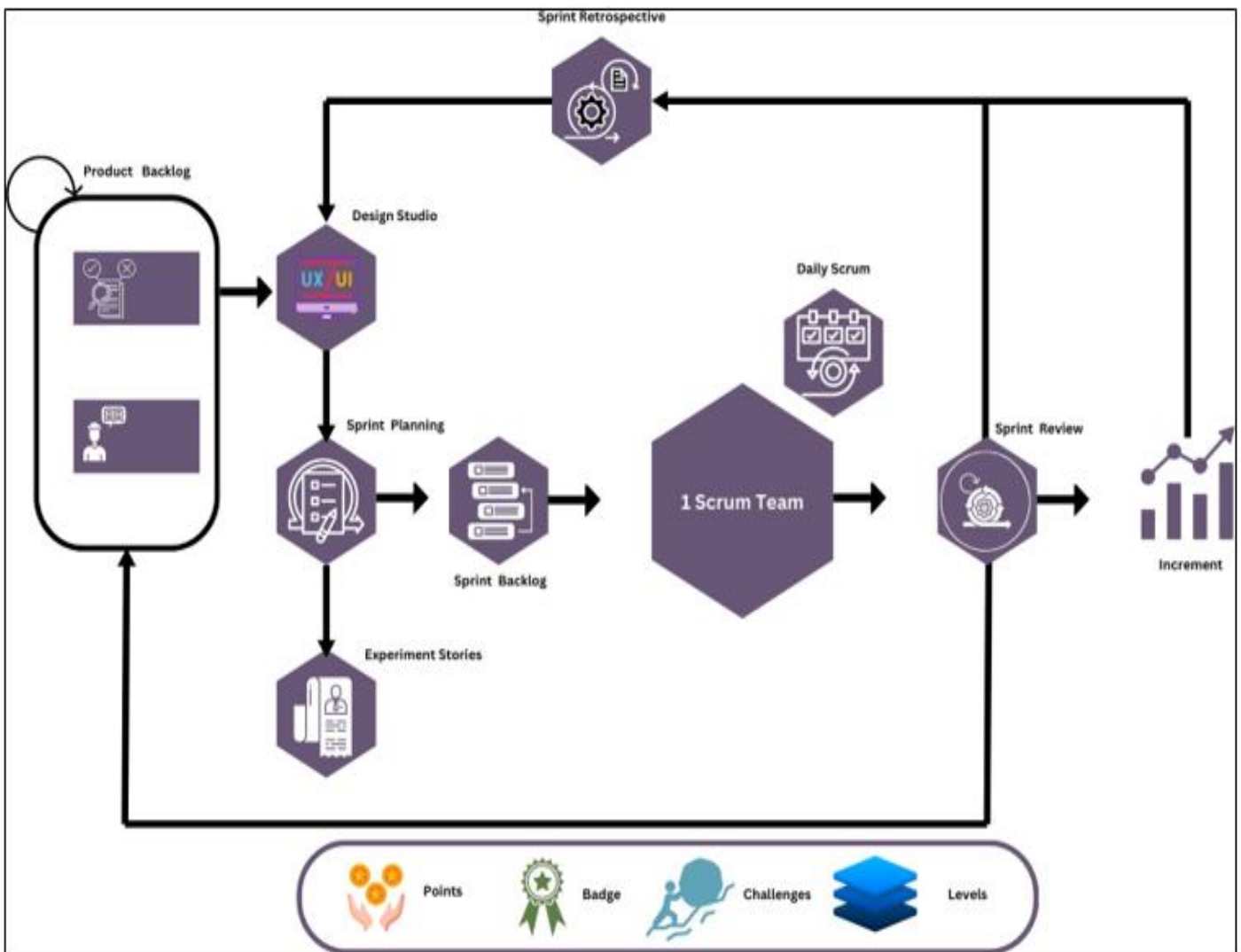


Fig 3 Integrating Scrum with Lean UX using gamification.

## V. CHALLENGES IN AGILE UX INTEGRATION

The integration of Agile and UX encounters various challenges, many rooted in misconceptions and myths surrounding both methodologies.

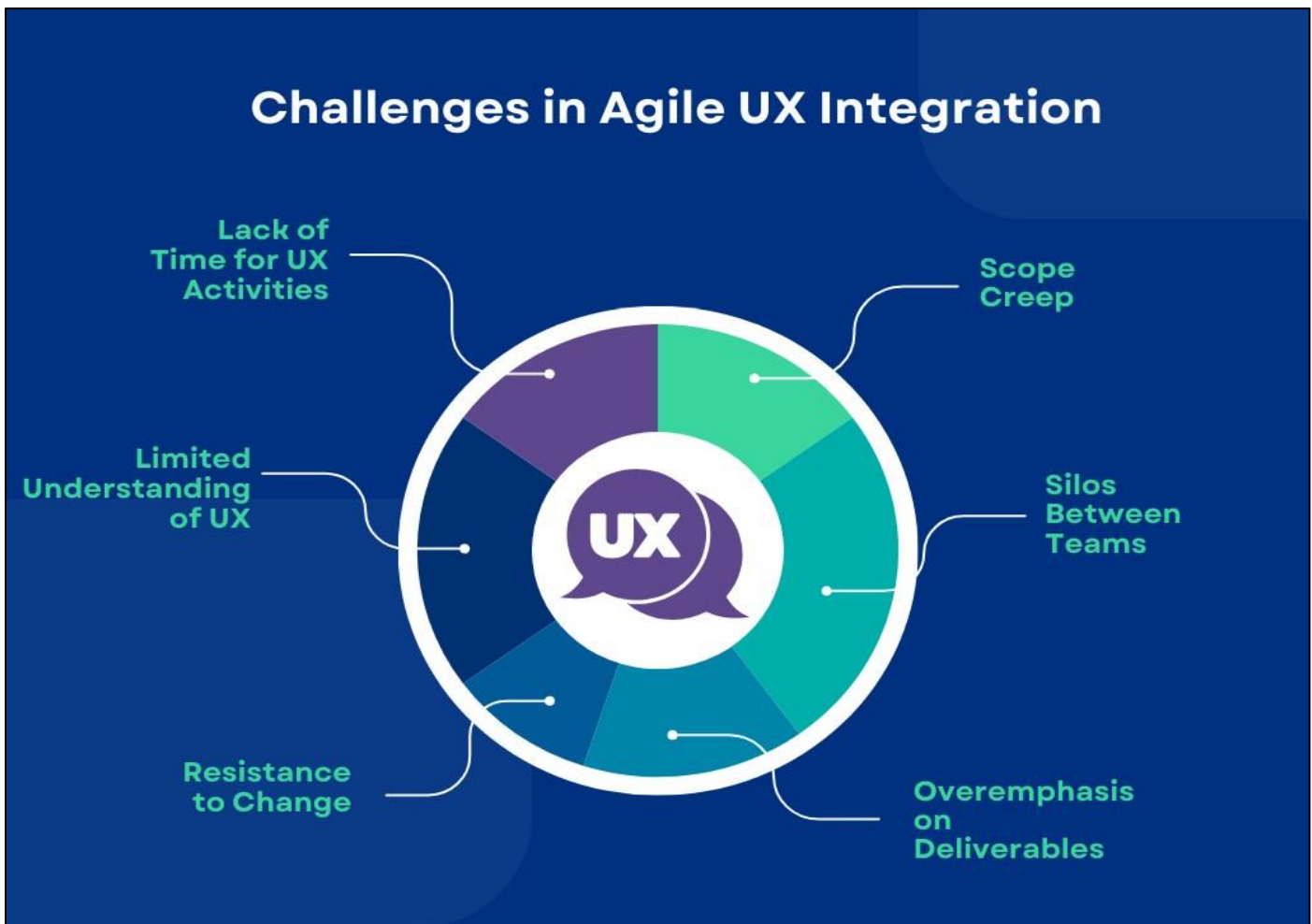


Fig 4 Challenges in Agile UX Integration

Some teams fear that Agile's focus on speed and delivery might sideline UX activities, perceiving UX design as a hindrance to the development pace. However, integrating UX into Agile iterations can ultimately elevate product quality and user satisfaction. Limited understanding of UX principles among Agile teams may lead them to perceive it solely as a development phase rather than an ongoing mindset. Consequently, the significance of UX research, usability testing, and iterative design might be overlooked. Resistance to change within Agile teams, accustomed to established workflows, may hinder the incorporation of UX practices. Overcoming this resistance demands effective communication, education about UX benefits, and illustrating how it aligns with Agile principles. Agile's flexibility can sometimes foster scope creep, as new features are introduced without considering their impact on user experience. Striking a balance between flexibility and user-centric focus is vital to curb scope creep. In organizations where Agile development and UX/design teams operate separately, miscommunication, duplicated efforts, and conflicts can arise. Breaking down these silos

and promoting collaboration is essential for successful integration. Agile methodologies' preference for working software over extensive documentation may undervalue UX artifacts like wireframes and prototypes. Educating Agile teams on the value of these artifacts in informing design decisions and meeting user needs is crucial. Measuring the impact of UX activities in Agile projects can be challenging due to the absence of clear metrics and KPIs. Defining and tracking UX-specific metrics such as usability scores and user satisfaction rates can help illustrate the value of UX in Agile projects. Addressing these misconceptions and challenges necessitates collaborative efforts among Agile practitioners, UX designers, and stakeholders to cultivate a shared understanding of UX's importance in delivering successful products. Providing training and resources to support Agile teams in effectively integrating UX into their development processes can also prove beneficial.

## VI. SOLUTIONS FOR EFFECTIVE COLLABORATION

### ➤ *Cross-Functional Collaboration Strategies*

Effective collaboration between development and design teams is paramount for successful product delivery. Initiate collaborative planning sessions wherein both teams collaborate to prioritize features, refine user stories, and synchronize project goals and timelines. Implement design sprints, where cross-functional teams collaboratively tackle specific design challenges by ideating, prototyping, and validating solutions within a condensed timeframe. Promote pairing and shadowing opportunities between developers and designers, fostering knowledge exchange, diverse perspectives, and skill-sharing to cultivate mutual understanding and empathy. Hold daily stand-up meetings inclusive of both development and design teams, facilitating progress updates, issue discussions, and task alignment for the day. Establish regular feedback mechanisms such as design reviews, code reviews, and retrospectives to foster continuous improvement and mutual learning between development and design teams.

### ➤ *Co-Location and Remote Collaboration Techniques*

Whenever feasible, co-locate development and design teams in a shared physical space to facilitate spontaneous communication, collaboration, and idea exchange. Establish dedicated collaboration areas, whether physical or virtual, where teams can convene for meetings, brainstorming sessions, and design reviews. Leverage video conferencing tools for remote collaboration, allowing face-to-face communication, screen sharing, and virtual whiteboarding sessions to bridge geographical barriers. Implement real-time collaboration platforms like shared document editors, virtual whiteboards, and collaborative prototyping tools to enable synchronous collaboration and idea generation irrespective of team locations. Utilize instant messaging and chat platforms for swift communication, informal discussions, and real-time resource sharing, nurturing a sense of unity among team members.

### ➤ *Tools and Technologies for Facilitating Collaboration*

Utilize project management tools like Jira, Trello, or Asana to efficiently organize tasks, monitor progress, and facilitate collaboration within Agile projects. Employ design software such as Figma, Sketch, or Adobe XD for crafting and distributing design mockups, prototypes, and interactive wireframes, fostering collaborative design iteration and feedback exchange. Implement version control systems like Git or SVN to collectively manage code and design assets, enabling multiple team members to collaborate on the same files concurrently and track modifications over time. Utilize collaborative documentation platforms like Confluence or Google Docs to document project requirements, design

choices, meeting summaries, and other pertinent project details, ensuring transparency and accessibility for all team members. Explore integration platforms and plugins to seamlessly connect various tools and systems utilized by development and design teams, streamlining workflows and minimizing manual efforts in information sharing and updates dissemination.

### ➤ *Key Principles of Agile UX Integration*

Integrating Agile and UX effectively requires adhering to key principles that align with both methodologies. Emphasize the importance of continuous feedback loops throughout the development process. This includes gathering feedback from users, stakeholders, and team members at every stage of the project. Incorporate user feedback through methods such as usability testing, user interviews, surveys, and analytics. Use this feedback to iterate on designs and make informed decisions about product features and functionality. Integrate feedback mechanisms into Agile ceremonies such as sprint reviews, daily stand-ups, and retrospectives to ensure that user perspectives are considered and addressed regularly. Adopt an iterative approach to design and prototyping, where designs are developed incrementally and refined based on user feedback and testing. Break down design tasks into small, manageable chunks that can be completed within a single iteration (sprint). This allows for rapid iteration and validation of design ideas. Utilize prototyping tools and techniques to create interactive prototypes that can be tested with users early and often. Prototype feedback helps identify usability issues, validate design assumptions, and refine the user experience iteratively. Prioritize user-centered requirements gathering to ensure that product features and functionality align with user needs, goals, and preferences. Involve UX designers and researchers in the early stages of the project to conduct user research, persona development, and user journey mapping.

Use these insights to inform the creation of user stories and acceptance criteria. Collaborate with stakeholders and development teams to translate user needs into actionable requirements that can be prioritized and implemented iteratively. Ensure that user-centric considerations are integrated into Agile planning processes, such as backlog grooming and sprint planning.

## VII. REAL-WORLD CASE STUDIES

### ➤ *Overview of Spotify's Agile Development Process*

Spotify employs an Agile development methodology, where cross-functional teams collaborate in short iterations, known as sprints, to introduce new features and enhancements. These teams are structured around distinct product domains or functionalities, promoting autonomy, collaboration, and rapid iteration.



Fig 5 Real-World Case Studies

#### ➤ *Integration of UX Design Principles into Agile Sprints*

Spotify places a strong emphasis on user-centric design, ensuring that UX designers collaborate closely with product managers, developers, and stakeholders to prioritize user needs throughout the development process. Design sprints are conducted to brainstorm, prototype, and validate design solutions before implementation, facilitating rapid iteration and user feedback integration. User research, usability testing, and analytics data play integral roles in Spotify's Agile ceremonies such as sprint planning, reviews, and retrospectives, guiding design decisions and feature prioritization. This Agile UX approach has contributed to Spotify's highly intuitive and user-friendly music streaming platform, continuously refining features based on user feedback and data insights. By integrating UX design principles into Agile sprints, Spotify has significantly improved product usability, increased user engagement, and enhanced customer satisfaction levels. Prioritizing user needs has enabled Spotify to maintain a competitive advantage in the competitive music streaming market, attracting and retaining a large user base.

#### ➤ *Google: Agile UX in Web Services*

##### • *Implementation of Agile Methodologies at Google:*

Google has embraced Agile methodologies, including Scrum and Kanban, across its diverse product teams, fostering rapid iteration, adaptability, and collaboration. These teams at Google adhere to Agile principles such as cross-functional collaboration, iterative development, and continuous delivery to ensure the delivery of high-quality web services. User-centric design is a cornerstone of Google's approach, with regular UX research and usability testing conducted to gain insights into user needs, behaviors, and pain points. Design sprints are employed to explore and validate design concepts, with an emphasis on delivering intuitive and seamless user experiences across various Google products like Search, Maps, and Gmail. Google's design philosophy, Material Design, serves as a unified framework for crafting user interfaces that prioritize usability, accessibility, and aesthetic appeal, contributing to a cohesive and consistent user experience across Google's product ecosystem.

• *Lessons Learned and Best Practices:*

Google underscores the significance of collaboration among design, development, and product management teams to achieve success in delivering web services. User feedback is esteemed as a crucial source of insights for steering product enhancements and fostering innovation, with data-driven decision-making steering product development endeavors. Google’s Agile UX approach underscores the significance of adaptability, experimentation, and perpetual learning in addressing evolving user requirements and market dynamics, guaranteeing that its products stay pertinent and competitive amidst the ever-changing digital milieu.

**VIII. IMPACT ON PRODUCT USABILITY AND CUSTOMER SATISFACTION**

➤ *Measuring the Impact of Agile UX on Usability Metrics*

Calculate the proportion of tasks effectively executed by users within the product. Agile UX methodologies like iterative design, continuous feedback loops, and user-centric requirements gathering play roles in enhancing task completion rates over time. Evaluate the duration users invest in accomplishing tasks within the product. Agile UX strategies emphasizing simplicity, efficiency, and intuitiveness may result in decreased time on task, signifying enhanced usability. Monitor the frequency and severity of user errors encountered within the product. Through iterative incorporation of user feedback and regular usability testing, Agile UX teams can identify and rectify usability issues contributing to error rates. Quantify the percentage of users who persist in using the product over time. Favorable user experiences stemming from Agile UX integration can result in increased retention rates, indicating improved usability and customer satisfaction.

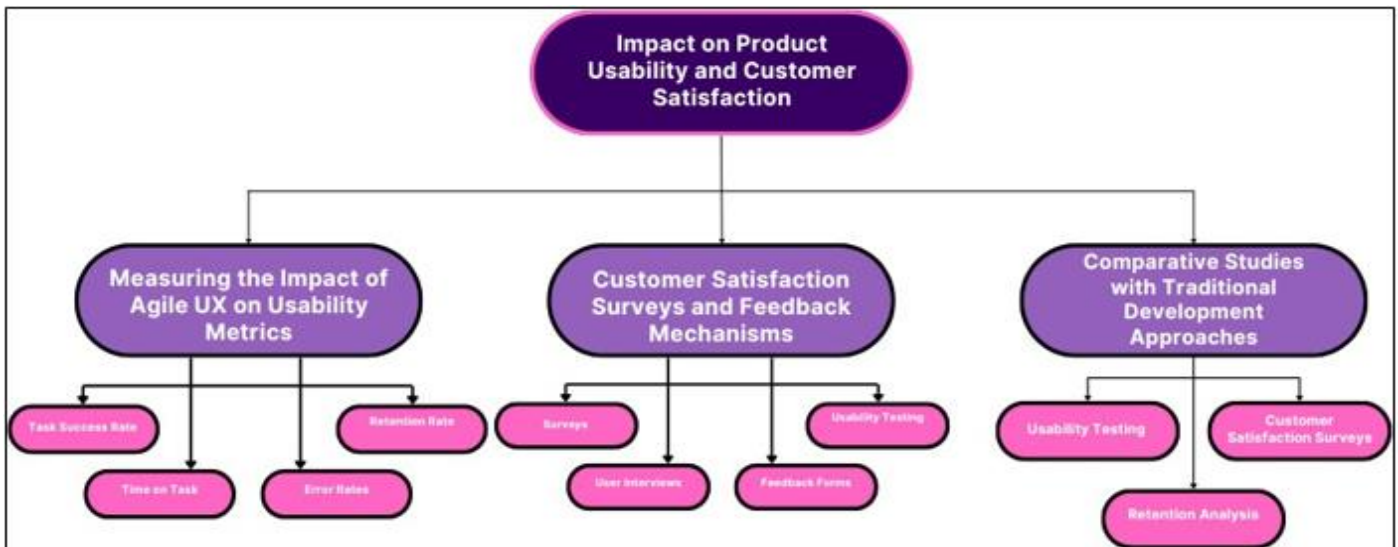


Fig 6 Impact on Product Usability and Customer Satisfaction

➤ *Customer Satisfaction Surveys and Feedback Mechanisms*

Administer customer satisfaction surveys to collect feedback on usability, functionality, and overall satisfaction with the product. Agile teams can leverage survey data to pinpoint areas for enhancement and prioritize features based on user preferences. Conduct one-on-one interviews with users to gather qualitative insights into their experiences, pain points, and preferences. Agile UX teams can utilize interview findings to shape design decisions and validate product assumptions. Integrate feedback forms within the product to enable users to provide real-time feedback on specific features, functionalities, or usability issues. Agile teams can utilize this feedback to iterate on designs and address user concerns promptly. Facilitate usability testing sessions with real users to observe their interactions with the product and pinpoint usability issues. Agile UX teams can leverage testing findings to prioritize design enhancements and validate design decisions.

➤ *Comparative Studies with Traditional Development Approaches*

Conduct usability testing sessions with users to compare the usability of products developed using Agile UX methodologies against those developed using traditional waterfall or sequential development approaches. Evaluate metrics such as task success rates, time on task, and user satisfaction scores to discern disparities in usability between the two methodologies. Administer customer satisfaction surveys to users who have utilized products developed using Agile UX methodologies and those developed using traditional approaches. Compare satisfaction scores and feedback to ascertain which approach elicits higher levels of customer satisfaction. Analyze retention rates of users who have engaged with products developed using Agile UX methodologies versus traditional approaches over time. Contrast retention rates to evaluate the enduring impact of Agile UX integration on user engagement and satisfaction.



## IX. FUTURE DIRECTIONS AND EMERGING TRENDS

### ➤ *Evolution of Agile UX Practices*

Agile UX practices will persist in prioritizing cross-disciplinary collaboration, involving designers, developers, product managers, and other stakeholders closely working together to deliver user-centric products. Additionally, these practices will increasingly integrate lean principles, aiming to minimize waste, optimize flow, and maximize value delivery. This will entail streamlining processes, reducing bureaucracy, and concentrating on delivering features that offer the highest value to users. Moreover, Design Ops, which concentrates on streamlining design processes and workflows, will become more intertwined with Agile UX practices. This will encompass the implementation of design systems, automation of repetitive tasks, and establishment of design governance frameworks to support scalable and efficient design operations.

### ➤ *Integration with Emerging Technologies (AI, IoT, etc.)*

Agile UX practices will require adaptation to the integration of artificial intelligence (AI) and machine learning (ML) technologies into products. This adaptation encompasses designing intuitive user interfaces for AI-powered features, integrating user feedback to enhance AI algorithms, and addressing ethical considerations concerning AI-driven decision-making. Additionally, Agile UX practices will play a pivotal role in crafting user experiences for IoT devices and ecosystems. This entails designing seamless interactions between physical and digital interfaces, ensuring interoperability across devices, and addressing privacy and security concerns linked with IoT data collection and processing. Furthermore, with the rising prominence of voice interfaces and natural language processing (NLP) technologies, Agile UX practices must concentrate on creating conversational user experiences that are intuitive, context-aware, and responsive to user needs and preferences.

### ➤ *Challenges and Opportunities in Scaling Agile UX*

Scaling Agile UX practices to large, distributed teams poses challenges concerning maintaining alignment, communication, and collaboration. Organizations will necessitate investing in tools, processes, and training to facilitate effective collaboration at scale. Furthermore, adopting Agile UX across an enterprise introduces challenges associated with organizational culture, legacy processes, and stakeholder buy-in. Encouraging a culture of experimentation, embracing change, and offering support and resources for teams transitioning to Agile UX practices become imperative for success.

Moreover, scaling Agile UX practices entails striking the right balance between speed and quality. Organizations must prioritize iterative delivery while ensuring that user needs, usability, and design quality remain uncompromised in the pursuit of speed.

## X. CONCLUSION

In summary, the fusion of Agile methodologies with User Experience (UX) design strategies offers a potent means of delivering user-centric products with efficiency and efficacy. Agile and UX methodologies exhibit shared principles like iterative development, user-centric focus, and continuous improvement, rendering them highly compatible for amalgamation. Cross-functional collaboration, on-going feedback loops, and iterative design emerge as pivotal for seamless Agile UX integration, ensuring the prioritization and fulfillment of user needs throughout the development journey. Vital for evaluating the effectiveness of Agile UX practices and fostering continuous enhancement is the measurement of their impact on usability metrics and the solicitation of customer feedback. The integration of emerging technologies such as AI, IoT, and voice interfaces unveils novel prospects and hurdles for Agile UX, necessitating adaptation and innovation to engineer seamless and intuitive user experiences. The ramifications for both research and practice are profound. Researchers can delve deeper into the efficacy of Agile UX integration across various contexts, pinpoint best practices, and devise frameworks for scaling Agile UX practices. Meanwhile, practitioners can leverage these insights to refine their Agile UX methodologies, accentuate user-centric design, and spur innovation in product development.

### ➤ *Looking Ahead, Several Recommendations for Future Work Emerge:*

- Undertake longitudinal investigations to evaluate the enduring effects of Agile UX integration on product usability, customer satisfaction, and business outcomes over time.
- Examine the involvement of Design Ops in bolstering scalable and streamlined Agile UX practices, encompassing the creation of design systems, streamlining design workflows through automation, and instituting design governance frameworks.
- Delve into the complexities and prospects surrounding Agile UX implementation at the enterprise level, scrutinizing aspects like organizational culture, stakeholder buy-in, and alignment with overarching business objectives.

## REFERENCES

- [1]. WebFX. (n.d.). Agile UX design: Embracing the key principles. Retrieved from
- [2]. Belkhir, L., & Elmeli, A. (2018). Assessing ICT global emissions footprint: trends to 2040 & recommendations. *Journal of Cleaner Production*, 177, 448–463.
- [3]. Calero, C., Angeles Moraga, M., & Garcia, F. (2022). Software, sustainability, and UN sustainable development goals. *IT Professional*, 24(1), 41–48.
- [4]. Calero, C., & Piattini, M. (2017). Puzzling out software sustainability. *Sustainable Computing: Informatics and Systems*, 16, 117–124.

- [5]. Eckstein, J., & Melo, C. de O. (2021). Sustainability: delivering agility's promise. *Software Sustainability*, 215–241.
- [6]. Ochoa-Zambrano, J. (2021). How collective intelligence can gear agility with sustainability. In P. Gregory & P. Kruchten (Eds.), *Agile Processes in Software Engineering and Extreme Programming – Workshops* (pp. 69–77). Springer.
- [7]. Da Silva, T. S., Silveira, M. S., Maurer, F., & Silveira, F. F. (2018). The evolution of agile UXD. *Information and Software Technology*, 102, 1–5.
- [8]. Curcio, K., Santana, R., Reinehr, S., & Malucelli, A. (2019). Usability in agile software development: a tertiary study. *Computer Standards & Interfaces*, 64, 61–77.
- [9]. Kashfi, P., Feldt, R., & Nilsson, A. (2019). Integrating UX principles and practices into software development organizations: a case study of influencing events. *Journal of Systems and Software*, 154, 37–58.