Czech University of Life Sciences Prague

Faculty of Economics and Management

Department of Systems Engineering

Master's Thesis

Designing a Gamification Model for Scrum across an Entire Epic

Bc. Pavlína Šimonová

© 2026 CZU Prague

CZECH UNIVERSITY OF LIFE SCIENCES PRAGUE

Faculty of Economics and Management

DIPLOMA THESIS ASSIGNMENT

Bc. Pavlína Šimonová

Project Management

Thesis title

Designing a Gamification Model for Scrum across an Entire Epic

---

Objectives of thesis

The aim of this thesis will be to design and apply a gamification model for the Scrum framework across an entire epic with the goal of increasing the motivation, engagement and productivity of agile team members. This aim comprises the following objectives:

1. Study the theoretical foundations of gamification and its current use in agile methodologies.
2. Identify gamification elements suitable for each phase of the Scrum process.
3. Develop a comprehensive gamification model tailored to the needs of agile teams, applicable across an entire epic.
4. Describe the conditions, steps, and practical approach for implementing the model in real Scrum environments.

Methodology

The methodology of the thesis will be based on a review of professional literature on gamification and agile methodologies, through which the key gamification elements used to date in Scrum will be identified and described. On the basis of this theoretical analysis, the selected elements will be adapted to the individual phases of the Scrum process and integrated into the design of a gamification model covering the entire epic.

The proposed extent of the thesis

60-80 pages

Keywords

Scrum, Sprint, Epic, Gamification, Project management, Agile

Recommended information sources

MYSLÍN, Josef. Scrum: průvodce agilním vývojem softwaru. 1. vyd. Brno: Computer Press, 2016. 168 s. ISBN 978-80-251-4650-7

PROJECT MANAGEMENT INSTITUTE. *The standard for project management and A guide to the project management body of knowledge : (PMBOK GUIDE).* Newton Square, Pennsylvania: Project Management Institute, Inc., 2021. ISBN 978-1-62825-664-2.

ŠOCHOVÁ, Zuzana and KUNCE, Eduard. *Agilní metody řízení projektů.* Brno: Computer Press, 2019. ISBN 978-80-251-4961-4.

ŠOCHOVÁ, Zuzana. *The great ScrumMaster : #ScrumMasterWay.* Boston: Addison-Wesley, 2017. ISBN 978-0-13-465711-0.

VERHEIJEN, Gunther. Scrum – A Pocket Guide: A Smart Travel Companion. 2. vyd. Utrecht: Van Haren Publishing, 2020

Expected date of thesis defence

2025/26 SS – FEM

Thesis supervisor

Ing. Petra Pavlíčková, Ph.D.

Supervising department

Department of Systems Engineering

Electronic approval: 13. 03. 2026

doc. Ing. Milan Houška, Ph.D.

Head of department

Electronic approval: 16. 03. 2026

prof. Ing. Lukáš Čechura, Ph.D.

Dean

Prague on 16. 03. 2026

Declaration

I declare that I have worked on my master's thesis titled "Designing a Gamification Model for Scrum across an Entire Epic" by myself and I have used only the sources mentioned at the end of the thesis. As the author of the master's thesis, I declare that the thesis does not break any copyrights.

I declare that I have used AI tools in accordance with the internal regulations of the university and the principles of academic integrity and ethics. The use of these tools in the thesis is referenced in an appropriate manner

In Prague on 31.3.2026 _____

Acknowledgement

I would like to thank Ing. Petra Pavlíčková, Ph.D., for their advice and support during my work on this thesis.

Designing a Gamification Model for Scrum across an Entire Epic

Abstract

Sustaining team motivation, engagement, and delivery quality across multiple successive iterations represents a persistent challenge within the Scrum framework. Existing gamification approaches in agile environments are predominantly Sprint-scoped, focused on narrowly tool-traceable behaviours, and insufficiently attentive to the motivational heterogeneity of cross-functional teams. This thesis responds by designing a structured gamification model intended for use across an entire Epic. The framework integrates Self-Determination Theory, the HEXAD user typology, and Scrum operational logic around four pillars of Agile Integrity: Process Flow, Technical Excellence, Product Clarity, and Team Synergy, which collectively monitor project health as a shared team responsibility. A dual incentive architecture combining individual Experience Points with collective Group Gold, a behavioural archetype system, and the Spotlight Mechanism are proposed as core operational components. Voluntary participation through Invisible Mode and exclusive use of non-monetary rewards are embedded as ethical safeguards. The framework targets agile teams with sufficient Scrum maturity and openness to gamified engagement. As a conceptual contribution, it establishes a theoretically grounded design foundation from which future empirical implementations can proceed.

Keywords: gamification, Scrum, agile, motivation, Self-Determination Theory, Epic, KPI, team engagement, game mechanics, framework design

Návrh gamifikačního modelu pro Scrum napříč celým Epicem

Abstrakt

Udržení motivace, angažovanosti a kvality dodávek v průběhu více po sobě jdoucích iterací představuje přetrvávající výzvu v rámci Scrumu. Stávající přístupy ke gamifikaci v agilním prostředí jsou převážně zaměřeny na izolované Sprinty, omezují se na snadno měřitelné chování a nedostatečně reflektují motivační heterogenitu cross-funkčních týmů. Tato práce reaguje návrhem strukturovaného gamifikačního modelu určeného pro použití napříč celým Epikem. Navržený rámec integruje teorii sebeurčení, typologii HEXAD a operační logiku Scrumu do čtyř pilířů agilní integrity: Process Flow, Technical Excellence, Product Clarity a Team Synergy, které společně monitorují zdraví projektu jako sdílenou týmovou odpovědnost. Jako klíčové komponenty jsou navrženy duální incentivní architektura kombinující individuální body zkušeností s kolektivním skupinovým zlatem, systém behaviorálních archetypů a mechanismus Spotlight. Dobrovolná účast prostřednictvím neviditelného režimu a výhradní využití nefinančních odměn jsou zabudovány jako etické pojistky. Rámec předpokládá agilní týmy s dostatečnou Scrum vyspělostí a otevřeností vůči gamifikovanému zapojení. Jako konceptuální příspěvek práce poskytuje teoreticky podložený návrhový základ, od kterého mohou vycházet budoucí empirické implementace.

Klíčová slova: gamifikace, Scrum, agilní metodiky, motivace, teorie sebeurčení, Epik, KPI, angažovanost týmu, herní mechaniky, návrh frameworku

Table of content

1 Introduction 12

2 Objectives and Methodology 14

2.1 Objectives 14

2.2 Methodology 14

3 Project management fundamentals 16

3.1 Definition of and core concepts 16

3.2 Project life cycle 16

3.3 Traditional vs. Agile Project Management 17

4 Agile Project Management 19

4.1 Origins and Rationale 19

4.2 Agile Manifesto 19

4.3 Core principles of Agile project management 21

4.4 Agile Project Life Cycle 22

4.5 Roles and Responsibilities in Agile project management 23

4.6 Benefits and Limitations of Agile project management 24

5 Scrum 24

5.1 Origins and rationale of Scrum 24

5.2 Definition and Characteristics of Scrum 25

5.3 Scrum Roles 26

5.4 Scrum Artifacts 28

5.5 Scrum Events 29

5.6 Scrum Life Cycle and Iterative Development 31

5.7 Advantages and Limitations of Scrum 32

5.8 Performance Metrics and Key Performance Indicators in Scrum 33

6 Gamification 34

6.1 Definition of Gamification 34

6.2 Motivational Foundations of Gamification 35

6.2.1 Game Mechanics and Game Dynamics 36

6.3 Benefits of Gamification 37

6.4 Risks and Limitations of Gamification 39

7 Psychological Aspects and Player Typologies 40

7.1 Motivation in Work and Team Environments 40

7.2 Individual Differences in Motivation and Behaviour 41

7.3 Player Typologies (Bartle, HEXAD) 43

7.4 Implications for Motivation in Scrum Teams 44

8 Gamification in Agile environments 46

8.1 Gamification in Agile Teams 46

8.2 Existing Approaches to Gamification in Scrum 47

8.3 Identified Limitations of Current Approaches 48

8.4 Research Gap and Justification for a New Gamification Model 50

9 Design of the Adaptive Agile Gamification Framework 52

9.1 Conceptual Framework and Strategic Alignment 52

9.1.1 Principles of the Adaptive Agile Gamification Framework 52

9.1.2 Integration with Scrum Events and the Epic Lifecycle 53

9.2 Team Diagnostics and Behavioural Archetype Mapping 54

9.2.1 Design of the Character Charter 54

9.2.2 Mapping Psychological Profiles to Hero Archetypes 55

9.3 The Four Pillars of Agile Integrity 58

9.3.1 Definition of Strategic Modules 59

9.3.2 Selection of Operational Metrics and KPIs 62

9.4 The Dynamic Adaptation Engine 65

9.4.1 Mathematical Model for Incentive Calculation 65

9.4.2 The Spotlight Mechanism and Priority Logic 70

9.5 Reward Economy and Participation Strategies 72

9.5.1 Individual progression, Shared Treasury and Non-monetary Loot System 72

9.5.2 Handling Resistance: The Opt-out and Invisible Mode 74

9.6 Conceptual Data Model of the Framework 75

9.6.1 Entity Structure and Attribute Specification 76

9.6.2 Relationship Logic of the Conceptual Model 79

10 Results and Discussion 80

10.1 Evaluation of the Proposed Framework 80

10.1.1 Strengths and Contributions of the Framework 80

10.1.2 Limitations and Risks of the Framework 82

10.2 Recommendations and Future Directions 84

10.2.1 Recommendations for Practical Application 84

10.2.2 Recommendations for Future Research 86

11 Conclusion 88

12 References 91

13 List of pictures and tables 97

List of pictures 97

List of tables 97

1 Introduction

Agile project management has established itself as the dominant delivery paradigm in knowledge-intensive and software development environments, with the Scrum framework representing its most widely adopted implementation. Scrum structures work through short, iterative cycles called Sprints, each designed to produce a usable product increment and to support continuous inspection and adaptation. While this iterative model is effective at managing short-term delivery risk and maintaining stakeholder alignment, it does not inherently address the challenge of sustaining team motivation, engagement, and consistent delivery quality across a longer delivery horizon. In practice, Scrum teams often operate not only within individual Sprints but across sequences of related iterations unified by a shared product objective, commonly referred to as an Epic. This longitudinal dimension introduces motivational and quality management challenges that Sprint-level mechanisms alone are insufficient to address.

Gamification, the application of game design elements in non-game contexts, has attracted growing interest as a mechanism for sustaining motivation and engagement in professional environments. In agile settings, gamified interventions have been proposed and studied as a means of increasing team participation, reinforcing desirable delivery behaviours, and supporting adherence to agile practices. However, the existing body of research reveals significant limitations: most implementations are narrow in scope, operationalise gamification through a limited set of tool-traceable behaviours within individual Sprints, and fail to account for the motivational heterogeneity of cross-functional teams whose members contribute in qualitatively different ways. No established model exists that integrates motivational theory, Scrum operational logic, and adaptive governance into a single framework designed to operate continuously across an entire Epic.

This thesis addresses that gap. The primary aim is to design a gamification model for the Scrum framework applicable across an entire Epic, with the goal of increasing the motivation, engagement, and productivity of agile team members while maintaining multi-dimensional project integrity. The model draws on Self-Determination Theory as its primary motivational foundation, integrates the HEXAD user typology for individual profiling, and introduces a four-pillar structure for monitoring project health as a shared team responsibility across the Epic lifecycle.

The thesis proceeds from a review of the theoretical foundations in project management, agile methodology, Scrum, and gamification, through a critical analysis of existing approaches and their limitations, to the full design of the proposed framework and its conceptual formalisation in the form of an entity-relationship and information model. The final chapter evaluates the model, acknowledges its limitations, and formulates recommendations for practical application and future research.

2 Objectives and Methodology

2.1 Objectives

The aim of this thesis will be to design and apply a gamification model for the Scrum framework across an entire epic with the goal of increasing the motivation, engagement, and productivity of agile team members. This aim comprises the following objectives:

1. Study the theoretical foundations of gamification and its current use in agile methodologies.
2. Identify innovative gamification elements suitable for each phase of the Scrum process.
3. Develop a concrete gamification model tailored to the needs of agile teams, with an emphasis on implementation across the entire epic.
4. Evaluate the model and propose further steps.

2.2 Methodology

The methodology of this thesis combines literature review, comparative analysis, conceptual design, conceptual modelling, and structured evaluation.

In the first phase, relevant academic and professional literature on project management, agile methodologies, Scrum, gamification, and work motivation was reviewed in order to establish the theoretical foundation of the study and to identify the principal gamification elements and design principles discussed in relation to agile environments. AI tools were used solely as supporting instruments during the literature search and source identification phase in order to assist with keyword refinement, retrieval, and preliminary orientation within relevant sources. The final selection, verification, interpretation, and incorporation of all sources into the thesis were carried out by the author.

In the second phase, existing approaches to gamification in agile and Scrum environments were analysed and compared with regard to their scope, motivational assumptions, temporal horizon, and suitability for cross-functional Scrum teams. This analysis served to identify recurring limitations of current approaches, particularly their focus on short-term and tool-traceable behaviours within individual Sprints.

On the basis of this synthesis, a gamification model was designed for use within the Scrum framework across an entire epic. The proposed model was subsequently formalised at the conceptual level through the specification of its motivational logic, KPI architecture, reward structure, adaptive governance mechanisms, and entity-relationship structure.

Finally, the proposed framework was evaluated through conceptual and critical assessment focusing on its internal coherence, alignment with Scrum principles, grounding in motivational theory, practical applicability, and potential risks and limitations. Since the thesis does not include empirical implementation, the behavioural and performance effects of the model are not measured directly, and their verification is left for future research.

3 Project management fundamentals

3.1 Definition of and core concepts

Project management is broadly defined across major professional frameworks as the structured application of knowledge, skills, tools, and techniques to achieve project objectives. The Project Management Institute (PMI) defines it as "the application of knowledge, skills, tools, and techniques to project activities to meet project requirements" (Project Management Institute, 2021), while IPMA emphasises the role of individual competences (International Project Management Association, 2015) and PRINCE2 focuses on governance and control structures (PeopleCert, 2023). Despite differences in emphasis, all three frameworks converge on the understanding of project management as coordinated effort aimed at delivering unique outcomes within constraints of time, cost, and quality.

A project itself is defined by PMI as "a temporary endeavour undertaken to create a unique product, service, or result" (Project Management Institute, 2021), a definition echoed by IPMA and PRINCE2, which add dimensions of multidisciplinarity and organisational purpose. Project objectives are ideally formulated using the SMART framework (specific, measurable, achievable, relevant, time-bound), and governed by the classical triple constraint of time, cost, and scope, later expanded to include value delivery, stakeholder satisfaction, and risk governance (Svozilová, 2011; Doležal, 2016).

3.2 Project life cycle

The project life cycle structures delivery into sequential phases: Initiation (objective definition and stakeholder identification), Planning (scope, schedule, and resource planning), Execution (delivery of outputs), Monitoring and Controlling (performance tracking and change management), and Closure (formal acceptance and documentation) (Project Management Institute, 2021; Rosenau, 2007; Doležal, 2016).

Three foundational roles underpin project delivery. The Project Manager leads planning, coordinates resources, manages stakeholder communication, and drives delivery across all lifecycle phases (Svozilová, 2011). The Project Sponsor holds strategic authority, ensures alignment with organisational goals, and legitimises resource allocation (Svozilová, 2011). The Project Team executes defined tasks; its effectiveness depends on role clarity, competency mix, and communication structure (Svozilová, 2011).

3.3 Traditional vs. Agile Project Management

In current project management practice, two fundamentally different practices are commonly used, traditional model (predictive) and agile model (adaptive). While traditional project management model focuses on thorough planning and predictability, agile methodologies represent a fundamental change in thinking towards adaptability and empiricism.

Traditional project management relies on a predictive lifecycle in which scope, schedule, and cost are defined in detail at the outset and changes are tightly controlled through formal processes (Project Management Institute, 2021). In contrast, agile methods emerged in response to rapidly changing business environments, formalized by the Agile Manifesto in 2001, and prioritize iterative development, customer collaboration, and flexibility in the face of changing requirements (Šochová, 2024).

The comparison between these two approaches reveals fundamental structural, procedural, and philosophical differences. The following dimensions illustrate the most significant contrasts:

1. **Control logic:**

Traditional project management relies on predictive planning and structured phase progression. Projects are expected to follow a linear path, with minimal deviation from the initial plan (Project Management Institute, 2021). The project manager occupies a central role in coordinating these processes and enforcing compliance with the baseline (Svozilová, 2011). Agile methodologies, in contrast, are based on iterative cycles, frequent inspection, and continuous adaptation. They work on the assumption that not all requirements can be known upfront and that flexibility is necessary in dynamic environments (Šochová, 2024).

2. **Scope, Time, and Cost:**

Predictive models define scope at the beginning of the project and derive time and cost estimates from that scope. Any changes to scope typically require formal approval processes and impact the baseline (Doležal, 2016). The project is complete only when all planned deliverables are produced (Project Management Institute, 2021). Agile reverses this constraint structure, time and cost are fixed while scope remains flexible. Features are delivered incrementally and prioritized based on business value and stakeholder feedback (Šochová, 2024).

3. **Change Management:**

Changes in traditional projects are treated as disruptions to the plan and are often discouraged unless formally justified and approved. (Project Management Institute, 2021). In contrast, agile teams embrace change as a natural and often beneficial part of the project lifecycle. Requirements are expected to evolve, and agile teams adjust deliverables accordingly throughout iterations (Šochová, 2024).

4. **Quality Assurance:**

Traditional quality control is typically performed in dedicated phases, often after development is complete. This can result in late discovery of defects and costly corrections (Svozilová, 2011). Agile methodologies emphasize continuous quality assurance, where testing, validation, and stakeholder review are integrated into each iteration. This reduces the risk of late-stage rework and supports a culture of continuous improvement (Šochová, 2024).

5. **Governance and Roles:**

The traditional model places decision-making and coordination in the hands of a project manager, who holds formal authority over the team and project plan (Svozilová, 2011; Rosenau, 2007). Agile methodologies distribute responsibilities among roles or team functions, promoting collaboration and shared ownership. Teams are often cross-functional and self-organizing, with minimal hierarchical separation between planning and execution (Šochová, 2024).

6. **Risk and Feedback:**

Traditional methods manage risk by identifying and analysing it early, assigning reserves, and attempting to mitigate it before execution begins (Project Management Institute, 2021). Agile methods address risk through early validation, short delivery cycles, and constant stakeholder interaction. This allows issues to be detected early and corrected before they escalate, an approach sometimes referred to as “failing fast” (Šochová, 2024).

7. **Success Criteria:**

Success in traditional projects is typically defined by adherence to the original scope, budget, and timeline (Doležal, 2016). Agile focuses on delivering business value and customer satisfaction. Working outputs and stakeholder feedback take priority over formal documentation or conformance to initial predictions (Šochová, 2024).

8. **Context of Application:**

Traditional project management remains suitable for stable environments where requirements are known upfront and changes are costly or undesirable, such as in engineering, construction, or regulated industries (PeopleCert, 2023; Rosenau, 2007). Agile is better suited to dynamic and innovative environments where requirements evolve during the project, such as software development or product research (Šochová, 2024).

4 Agile Project Management

4.1 Origins and Rationale

Agile project management emerged at the turn of the 21st century as a response to the growing inadequacy of traditional plan-driven approaches in environments characterised by rapid change and uncertainty. In technology and innovation-driven sectors, large-scale waterfall projects frequently failed to meet stakeholder needs or adapt to late-emerging requirements, creating pressure for a fundamentally different approach to project delivery (Svozilová, 2011; Dolezal, 2016; Project Management Institute, 2021).

Rather than attempting to eliminate uncertainty through exhaustive upfront planning, Agile frameworks assume that change is inevitable and must be managed continuously. Planning becomes an iterative, revisable process; success depends not on plan conformance but on the team's ability to respond to feedback, incorporate new insights, and adapt deliverables in real time (Šochová, 2024; Project Management Institute, 2021). Building on earlier incremental and spiral models, Agile unified these concepts into a coherent philosophy oriented around empirical control, stakeholder collaboration, and rapid delivery of value (Project Management Institute, 2021; Šochová, 2024).

This shift is now reflected in major international standards. The PMBOK Guide (2021) explicitly recognises a full spectrum of delivery modes from predictive to adaptive, treating Agile not as a rejection of classical project management but as its contextual evolution in response to the demands of modern, knowledge-intensive delivery environments (Project Management Institute, 2021; PeopleCert, 2023).

4.2 Agile Manifesto

The Agile Manifesto was introduced in 2001 by a group of experienced practitioners as a direct response to the inadequacy of rigid, plan-driven approaches in dynamic

development environments (Beck et al., 2001; Project Management Institute, 2021; Šochová, 2024). Although initially conceived within software development, its core ideas have since found broader application across project management disciplines (Dong, 2024; Carlos, 2023). Rather than prescribing a specific methodology, the Manifesto presents a flexible mindset that reframes how project objectives are pursued under conditions of uncertainty (Doležal, 2016; Project Management Institute, 2021).

The Manifesto is grounded in four fundamental values (Beck et al., 2001):

1. **Individuals and interactions over processes and tools**

   Emphasis is placed on effective communication, mutual trust, and collaboration among team members as the main drivers of project success. Processes and tools are viewed as supporting elements, not primary determinants of performance (Doležal, 2016; Project Management Institute, 2021).

2. **Working deliverables over comprehensive documentation**

   The primary objective is to deliver functional results rather than producing extensive formal documentation. Although documentation remains important, it should be concise, relevant, and supportive of delivery rather than an administrative burden (Doležal, 2016).

3. **Customer collaboration over contract negotiation**

   Agile emphasises continuous and transparent cooperation with stakeholders throughout the entire project lifecycle. Instead of rigid adherence to contracts, agile teams foster dynamic dialogue to align deliverables with actual stakeholder needs (Doležal, 2016).

4. **Responding to change over following a plan**

   Agile approaches treat change not as a disruption but as a normal part of evolving project conditions. Project plans are adaptable and subject to frequent revision based on feedback and new insights (Doležal, 2016).

These values are further elaborated through twelve principles that serve as operational guidelines for Agile project management (Beck et al., 2001). Among the most relevant for general project contexts are:

- Early and continuous delivery of valuable outcomes, typically through short iterations or phases, which enable timely feedback and better risk management.
- Acceptance of changing requirements, even in late stages of execution, to maintain alignment with stakeholder priorities.

- Frequent delivery of partial outputs, which supports transparency and facilitates adaptive planning.
- Close and ongoing cooperation with stakeholders, enabling faster decision-making and better prioritisation of work.
- Self-organising and empowered teams, trusted to manage their own processes and deliver results autonomously.
- Face-to-face communication as the preferred method for sharing information, supporting faster resolution of issues.
- Progress measured through delivered functionality, rather than administrative indicators.
- Sustainable work pace, preventing team burnout and preserving long-term productivity.
- Technical excellence and sound design, supporting flexibility and reducing the cost of future change.
- Simplicity, meaning the elimination of non-essential tasks, documents, or features.
- Decentralised decision-making, enhancing responsiveness and team ownership.
- Regular reflection and improvement, institutionalised through retrospectives at the end of each cycle.

  (Beck et al., 2001)

Together, these values and principles establish a conceptual foundation for modern, adaptive project management. The role of the project manager shifts from directive oversight to facilitation, while teams assume greater autonomy in how objectives are achieved (Schwaber, 2020; Dong, 2024).

4.3 Core principles of Agile project management

Agile project management is governed by five interconnected principles that structure both delivery and team behaviour throughout the lifecycle (Project Management Institute, 2021; Svozilová, 2011).

Iterative and incremental delivery organises work into short, repeatable cycles, within each of which the team completes a reviewable portion of the final output. This enables continuous feedback, clarifies stakeholder expectations progressively, and supports

early value delivery while reducing the likelihood of late-stage corrections (Project Management Institute, 2021; Doležal, 2016).

Adaptive planning treats planning not as a one-time activity but as an ongoing process that evolves alongside the project. Only immediate next steps are defined in detail; future plans are revised after each iteration based on what has been delivered and learned. This preserves flexibility in the face of uncertainty without sacrificing directional clarity (Šochová, 2024; PeopleCert, 2023; Project Management Institute, 2021).

Empirical process control places decisions on actual results rather than theoretical predictions. Each iteration concludes with a structured review and retrospective grounded in three mutually reinforcing pillars: transparency, inspection, and adaptation. Together, these ensure that deviations are identified early and that the project trajectory remains focused on measurable value (Schwaber, 2020; Project Management Institute, 2021).

Self-organising and empowered teams challenge traditional hierarchical models by distributing operational decision-making to the team level. Cross-functional members are jointly accountable for planning, execution, and adjustment. This distribution of authority supports faster responses, fosters ownership, and encourages innovative problem-solving, contributing significantly to project resilience (Šochová, 2024; Project Management Institute, 2021).

Continuous stakeholder engagement ensures that stakeholders participate not only at project boundaries but throughout its entire duration. Feedback collected after each iteration feeds directly into the next planning cycle, keeping the evolving solution aligned with business needs and supporting faster, more informed decision-making (Šochová, 2024; Svozilová, 2011).

4.4 Agile Project Life Cycle

The agile project life cycle is not a linear sequence of predefined activities but a responsive, iterative loop. At the outset, the team defines a high-level product vision, which is decomposed into smaller deliverable units addressed through successive time-boxed iterations. Each iteration follows a consistent rhythm: planning (prioritisation, effort estimation, and scope adjustment based on progress), execution (production of a working and independently verifiable increment), review (stakeholder validation and direct

feedback), and retrospective (process reflection and identification of improvements for the next cycle) (Project Management Institute, 2021; Šochová, 2024; PeopleCert, 2023).

Rolling-wave planning reflects the principle that future details should emerge progressively rather than be fixed upfront, allowing the team to continuously incorporate current information and revise earlier assumptions (PeopleCert, 2023; Doležal, 2016). Feedback gathered during validation feeds directly into the next planning cycle rather than being deferred, making validation a routine embedded activity rather than an isolated milestone. The retrospective at the end of each iteration ensures that the team's way of working evolves alongside the product, institutionalising continuous improvement as a structural feature of the lifecycle. The defining characteristic of the agile life cycle is the integration of short, goal-oriented iterations that structure both delivery and learning, ensuring that progress is visible, testable, and directionally accurate at every step.

4.5 Roles and Responsibilities in Agile project management

Agile environments transform traditional project roles to support flexibility, team autonomy, and shared accountability. The emphasis is on motivated, multi-skilled individuals who are given the authority to make decisions, with leadership becoming supportive and facilitative rather than directive (Šochová, 2024; Project Management Institute, 2021; Doležal, 2016).

The Project Manager shifts from directive oversight to facilitation, focusing on maintaining a shared project vision, coordinating iterative planning, and removing obstacles to progress rather than managing individual tasks directly (Project Management Institute, 2021; Doležal, 2016; Šochová, 2024). Agile Teams function as self-organising, cross-functional units with joint responsibility for planning, execution, and quality; role boundaries are intentionally fluid and adapt to project phase and technical complexity (Šochová, 2024). Stakeholders are integrated as continuous partners, providing feedback and reprioritising work after each iteration to ensure that deliverables remain aligned with current business needs and user expectations (PeopleCert, 2023; Project Management Institute, 2021).

4.6 Benefits and Limitations of Agile project management

Agile project management offers significant advantages in dynamic environments but also introduces challenges that must be actively managed. The key benefits and limitations are presented in Table 1.

Table 1: Benefits and limitations of Agile Project Management

Aspect	Potential Benefit	Potential Limitation
Planning and scope	Continuous reprioritisation and flexible scope adaptation	Lower predictability of total cost and timeline
Delivery	Early value delivery through iterative outputs	Dependence on active stakeholder involvement
Quality	Continuous improvement through regular feedback, testing, and retrospectives	Quality may become inconsistent if agile practices are applied by immature teams
Transparency and coordination	Shared visibility supports coordination and shared responsibility	Agile requires intensive and continuous communication
Stakeholder alignment	Ongoing stakeholder engagement helps keep outcomes relevant to current business needs	Limited stakeholder availability may weaken prioritisation and reduce alignment
Documentation	Focus on working outcomes may accelerate progress	Reduced formal documentation may hinder maintenance and knowledge transfer

Source: author's own elaboration based on: Project Management Institute (2021); Svozilová (2011); Doležal (2016); Dong (2024); Carlos (2023); Šochová (2024); PeopleCert (2023); Turley (2018).

Agile is well suited to change-driven environments, but its effectiveness depends on team maturity, structured coordination, and active stakeholder engagement. Only when supported by a disciplined and reflective implementation can agile fulfil its potential to deliver focused and resilient project outcomes (Dong, 2024).

5 Scrum

5.1 Origins and rationale of Scrum

The theoretical foundations of Scrum precede its formal introduction. In their 1986 Harvard Business Review article *The New New Product Development Game*, Hirotaka Takeuchi and Ikujiro Nonaka critically examined conventional product development approaches based on strictly sequential logic, often described as a relay race in which tasks are handed over between specialised functional units. Such an arrangement was shown to

limit flexibility, slow down learning, and reduce the ability of teams to respond effectively to change (Takeuchi, 1986).

As an alternative, Takeuchi and Nonaka proposed a holistic development model inspired by rugby, in which a cross-functional team advances together toward a common goal, with phases overlapping, information flowing continuously, and responsibility shared collectively rather than fragmented across organisational silos. Although the authors did not define Scrum as a formal framework, the principles articulated in their work, namely self-organising teams, intensive collaboration, and iterative learning, later became core pillars of Scrum and agile thinking more broadly (Takeuchi, 1986; Verheyen, 2019).

Scrum was formally introduced by Ken Schwaber and Jeff Sutherland in 1995 at the OOPSLA conference (Object-Oriented Programming, Systems, Languages & Applications). The term was deliberately drawn from rugby terminology, reflecting the emphasis on teamwork, collective movement, and frequent interaction rather than linear task execution (Verheyen, 2019). As adoption grew throughout the late 1990s and early 2000s, the absence of a unified authoritative description led to inconsistent interpretations in practice. In response, Schwaber and Sutherland published the first official Scrum Guide in 2010, establishing a common reference by defining the framework's roles, artifacts, and events. The most recent version, published in 2020, explicitly positions Scrum as applicable beyond software development, addressing complex problems across a wide range of domains (Schwaber, 2020).

5.2 Definition and Characteristics of Scrum

The Scrum Guide defines Scrum as a lightweight framework whose operation is grounded in empirical process control. Empiricism posits that knowledge is acquired primarily through experience and that decisions should be based on observed reality rather than predictive speculation (Schwaber, 2020). In the context of complex product development, where requirements and technologies are frequently volatile, empiricism enables teams to navigate uncertainty through evidence-based management and iterative learning cycles (Verheyen, 2019). To ensure that empirical process control functions effectively, three mutually reinforcing pillars must be upheld: transparency, inspection, and adaptation. They represent an interlocking system in which the failure of anyone compromises the integrity of the entire framework.

In addition to the three pillars, Scrum emphasises five core values: commitment, focus, openness, respect, and courage. These values are intended to shape team culture and behaviour, supporting the trust, collaboration, and responsible self-management on which effective empiricism depends (Schwaber, 2020).

Transparency requires that significant aspects of the process be visible to all those responsible for the outcome. Artifacts such as the Product Backlog and Sprint Backlog serve as information radiators that expose the true state of the project, while common standards such as the Definition of Done establish a unified agreement on what constitutes complete work, preventing ambiguity regarding the quality of the Increment (Schwaber, 2020; Turley, 2018). Svozilová notes that concealing problems or reporting inaccurate status updates destroys transparency and renders the subsequent steps of inspection and adaptation impossible (Svozilová, 2011).

Inspection involves the frequent assessment of Scrum artifacts and progress toward a goal to detect undesirable variances. Scrum incorporates specific events, such as the Sprint Review and Daily Scrum, to serve as formal and regular opportunities for this assessment (Doležal, 2016). Effective inspection requires a diligent examination of the work by skilled professionals; however, inspection without transparency is misleading, as it involves the examination of flawed or hidden data. The Scrum Guide further emphasises that inspection is functionally useless if it is not followed by adaptation (Schwaber, 2020).

Adaptation is the adjustment of the process or the product in response to the data gathered during inspection. If inspection reveals that one or more aspects of a process fall outside acceptable limits and that the resulting product may become unacceptable, corrective action must be taken without delay in order to prevent further deviation (Beck, 2001). For adaptation to be effective, the team requires the authority to manage their own work. Šochová argues that if a team identifies a necessary change during inspection but lacks the power to alter their course of action without external permission, the empirical loop is broken, and the adaptability of Scrum is negated (Šochová, 2024).

5.3 Scrum Roles

Scrum defines a minimal and clearly qualified role structure designed to support self-management, accountability, and empiricism. The Scrum Team represents the fundamental unit of Scrum and is responsible for all activities related to the product extending beyond

development to include stakeholder collaboration, validation, experimentation, and any other work necessary to deliver value. No hierarchical structure exists among the three roles, and the recommended team size is ten or fewer members to preserve agility and communication effectiveness (Myslín, 2016; Šochová, 2024; Schwaber, 2020).

Product Owner is accountable for maximizing the value of the product and serves as the single authority for product priorities (Schwaber, 2020). This role is responsible for effective Product Backlog management, including defining and ordering backlog items according to business value, risk, dependencies, and urgency, and ensuring that the backlog remains transparent, visible, and understandable to all relevant parties. By acting as the sole authority for priorities, the Product Owner prevents conflicting requirements and reduces decision paralysis (Schwaber, 2020). A key responsibility is to establish and communicate the **Product Goal**, which provides long-term direction for product development and ensures strategic coherence across Sprints. The Product Owner continuously balances stakeholder expectations, market conditions, and organisational strategy when refining the backlog, although accountability for its quality and ordering remains solely with this role (Verheyen, 2019; Šochová, 2024). Undermining Product Owner authority directly compromises Scrum's empirical foundation and transparency (Schwaber, 2020; Project Management Institute, 2021).

Scrum Master is responsible for ensuring that both the Scrum Team and the organisation understand and correctly apply Scrum principles and practices. Unlike traditional project management roles, the Scrum Master does not manage people or assign tasks; instead, the role is grounded in servant leadership and focuses on facilitation, coaching, and enabling the team to function effectively (Schwaber, 2020). The Scrum Master ensures that Scrum events take place as prescribed, are productive, and adhere to their intended timeboxes; identifies and removes impediments hindering the Developers' progress; and assists the team in strengthening internal collaboration and self-management capabilities (Turley, 2018). At the organisational level, the Scrum Master promotes Scrum adoption and supports systemic improvements that foster agility. By reinforcing the three pillars, the Scrum Master safeguards the empirical nature of the framework. Ineffective or misunderstood Scrum Master roles are consistently identified in practitioner literature as a frequent cause of failed Scrum implementations, particularly in organisations transitioning from predictive management models (Verheyen, 2019; Project Management Institute, 2021).

Developers are the cross-functional, self-managing members of the Scrum Team directly responsible for creating the product Increment each Sprint. They collectively possess all skills necessary to plan, build, test, and deliver a usable Increment meeting the agreed quality standards, without reliance on external resources (Schwaber, 2020). Developers are collectively accountable for planning the Sprint, producing an Increment that complies with the Definition of Done, and continuously adapting their work to achieve the Sprint Goal. Scrum explicitly grants Developers full autonomy over how the work is performed, reinforcing shared ownership and professional accountability (SBOK Guide, 2022). The cross-functional nature of this role is a critical structural element: by ensuring that all necessary competencies are available within the team, Scrum minimises handovers, accelerates feedback loops, and strengthens ownership of outcomes, factors identified in empirical research as key drivers of improved productivity, quality, and responsiveness (Dong, 2024).

These three roles are intentionally designed to create a balance between strategic direction, process integrity, and execution autonomy. Clear accountability boundaries among them are critical for maintaining transparency and supporting empirical decision-making; deviations such as reintroducing hierarchical control or allowing role overlap undermine Scrum's foundational principles and reduce its effectiveness (Schwaber, 2020; Project Management Institute, 2021).

5.4 Scrum Artifacts

Scrum artifacts make the current state of work, progress toward goals, and product quality visible to both the Scrum Team and stakeholders, thereby enabling empirically grounded decision-making. Each artifact carries a specific commitment that reinforces transparency and prevents ambiguity (Schwaber, 2020; SBOK Guide, 2022).

Product Backlog is an evolving, ordered list of all items currently identified as necessary for the product and represents the single source of work for the Scrum Team. Items may take various forms, such as Epics, User Stories, or technical tasks, and are refined progressively as knowledge about the product and stakeholder needs develops (Schwaber, 2020). Responsibility lies with the Product Owner, who orders items by business value, risk, dependencies, and urgency. The Product Backlog is never complete; it reflects a dynamic understanding of the product, continuously adapting to feedback, market changes, and Sprint

Review outcomes (Verheyen, 2019; Šochová, 2024). The commitment associated with the Product Backlog is the **Product Goal**, which describes a long-term product objective ensuring strategic coherence across Sprints and providing direction for individual backlog items (Schwaber, 2020).

Sprint Backlog is a subset of the Product Backlog selected for a specific Sprint, together with a concrete plan for delivering the selected items and achieving the Sprint Goal. Unlike the Product Backlog, ownership of the Sprint Backlog belongs exclusively to the Developers, reflecting Scrum's emphasis on self-management and team autonomy (Schwaber, 2020). Created during Sprint Planning, the Sprint Backlog evolves throughout the Sprint as the team gains new insights or encounters impediments, making it a living artifact that reflects the real-time state of work and enables inspection during the Daily Scrum (Turley, 2018). Its associated commitment is the **Sprint Goal**, which articulates the purpose of the Sprint and provides flexibility in how the goal is achieved, allowing scope negotiation within Sprint boundaries as long as quality and objectives are preserved (Schwaber, 2020).

Increment represents the sum of all Product Backlog items completed during the current Sprint combined with the cumulative value of all previous Sprints. Each Increment must be usable and meet the quality standards defined by the Scrum Team, ensuring it is potentially releasable at the end of the Sprint (SBOK Guide, 2022). Its commitment is the **Definition of Done**, a formal shared understanding of what it means for work to be complete and of what quality, which eliminates ambiguity about completion and ensures that inspection of the Increment is meaningful and consistent (Schwaber, 2020; Verheyen, 2019). The Increment is primarily inspected during the Sprint Review, where stakeholder feedback directly influences future Product Backlog refinement, linking delivery, inspection, and adaptation in a closed feedback loop (Schwaber, 2020; SBOK Guide, 2022).

5.5 Scrum Events

Scrum defines five prescribed, timeboxed events that establish regularity, ensure transparency, and create formal opportunities for inspection and adaptation. By occurring in a fixed sequence within each Sprint, they reduce the need for ad-hoc meetings and form a closed feedback loop enabling iterative value delivery (Schwaber, 2020; Turley, 2018; SBOK Guide, 2022).

Sprint is the central event of Scrum and acts as a container for all other events. It represents an iteration with fixed length, typically lasting four or less weeks, during which a usable and potentially releasable Increment is created (Schwaber, 2020). Sprints provide a predictable cadence for planning, execution, and evaluation, supporting both transparency and risk reduction. During a Sprint, no changes are permitted that would endanger the Sprint Goal, and the Definition of Done must be upheld throughout. If the Sprint Goal becomes obsolete due to changing circumstances, the Sprint may be cancelled by the Product Owner. The Sprint thus balances short-term stability with strategic adaptability, protecting focus within the iteration while allowing directional change between iterations (Schwaber, 2020).

Sprint Planning initiates the Sprint and addresses three key questions: why the Sprint is valuable, what can realistically be completed, and how the selected work will be done. The entire Scrum Team attends, the Product Owner proposes the most valuable Product Backlog items, while the Developers forecast the work they can complete. The outcome is a Sprint Goal and a Sprint Backlog reflecting a collective commitment rather than a top-down assignment (Schwaber, 2020; Šochová, 2024; Turley, 2018).

Daily Scrum is a short, timeboxed event held every working day of the Sprint. Its primary purpose is to inspect progress toward the Sprint Goal and adapt the Sprint Backlog as necessary (Schwaber, 2020). This event is intended for the Developers and typically lasts no longer than 15 minutes. Rather than serving as a status meeting, the Daily Scrum functions as a coordination and planning mechanism. Developers synchronize their activities, identify impediments, and adjust their plan for the next 24 hours. By providing frequent inspection points, the Daily Scrum significantly increases transparency and reduces the risk of late discovery of problems (Turley, 2018).

Sprint Review takes place at the end of the Sprint and focuses on inspecting the Increment and adapting the Product Backlog. The Scrum Team presents the completed work to stakeholders, who provide feedback based on actual product outcomes rather than assumptions or documentation (Schwaber, 2020). This event is a collaborative working session rather than a formal acceptance meeting. Stakeholders and the Scrum Team discuss progress toward the Product Goal, changes in the environment, and possible next steps. The Product Backlog is updated accordingly, ensuring that future Sprints reflect the most current understanding of value and priorities (SBOK Guide, 2022).

Sprint Retrospective concludes the Sprint cycle and is dedicated to process improvement. The Scrum Team reflects on the previous Sprint with regard to collaboration,

tools, practices, and adherence to Scrum principles, identifying concrete actions to improve effectiveness in subsequent Sprints (Schwaber, 2020). Improvements may relate to technical practices, communication patterns, or team dynamics. By institutionalising reflection, the Sprint Retrospective supports continuous improvement and reinforces Scrum's empirical foundation (Verheyen, 2019; Project Management Institute, 2021).

5.6 Scrum Life Cycle and Iterative Development

The Scrum life cycle applies empirical process control through successive Sprints, each producing a usable Increment. Although Scrum incorporates all fundamental phases of project management, these phases are not executed sequentially or only once; they are distributed across the entire product lifecycle and recur within each Sprint. Planning takes place continuously through Product Backlog refinement, Sprint Planning, and daily replanning; monitoring and control are embedded in the Daily Scrum, Sprint Review, and Retrospective (Schwaber, 2020; Turley, 2018).

In addition to the Sprint-based rhythm, agile product delivery may be viewed through broader planning horizons extending across multiple iterations. One such horizon is the Epic, a large body of related work unified by a shared product objective and delivered incrementally across a sequence of Sprints. The Epic is not a formal construct of the Scrum Guide, which deliberately operates at the Product Backlog level without prescribing internal hierarchies; it is instead established through scaled agile frameworks such as SAFe and through industry tooling conventions, most notably Atlassian's Jira, where it has become a de facto standard for organising work above the User Story level (Leffingwell, 2011; Atlassian, 2024).

The lifecycle of an Epic follows the iterative logic of Scrum itself. An Epic originates as a high-level product objective identified during roadmap planning and is added to the Product Backlog in its unrefined state. As the team approaches the work, the Epic is progressively decomposed through backlog refinement into User Stories of appropriate granularity for Sprint Planning. Stories are then delivered across successive Sprints, each contributing an increment toward the Epic's completion. The Epic is considered closed when all constituent stories have been delivered and accepted, or when a strategic decision to discontinue the work is made (Leffingwell, 2011; Cohn, 2010). Throughout this lifecycle,

the Epic serves as the unit of continuity that preserves coherence of intent across Sprint boundaries without altering the underlying empirical structure of Scrum.

This longitudinal dimension has direct implications for performance measurement and team motivation. While individual Sprints provide short-cycle feedback, the Epic represents the horizon over which systemic patterns in delivery quality, flow efficiency, and team collaboration become observable. Metrics evaluated only at the Sprint level may obscure gradual degradation or improvement that manifests only across a longer sequence of iterations. The Epic therefore constitutes a meaningful unit of analysis for any framework that aims to sustain and improve team performance over time, a consideration that directly motivates the Epic-scoped design of the framework proposed in this thesis.

5.7 Advantages and Limitations of Scrum

The advantages and limitations of Scrum are closely linked to the same empirical and iterative principles that define the framework. Table 2 summarises the most significant characteristics.

Table 2: Advantages and limitations of Scrum

Aspect	Potential Benefit	Potential Limitation
Adaptability	Scrum supports continuous reprioritisation based on feedback and emerging information	In highly stable contexts, iterative planning may create unnecessary overhead
Transparency	Shared roles, events, and artefacts improve visibility and coordination	Transparency alone does not ensure effective adoption in low-maturity teams
Learning and improvement	Regular feedback and retrospectives support continuous improvement	Benefits depend on organisational readiness and team maturity
Delivery	Incremental delivery reduces the risk of delivering outdated solutions	Scrum offers limited explicit guidance for budgeting, formal risk management, and long-term scheduling
Organisational fit	Scrum works well in dynamic and collaborative environments	Coordination becomes more demanding in large and highly interdependent settings

Source: author's own elaboration based on: Project Management Institute (2021); Šochová (2024); Turley (2018); Schwaber (2020); Verheyen (2019).

Scrum should be understood as a flexible framework whose effectiveness depends on organisational readiness, team maturity, and context-aware application, rather than as a universally applicable solution.

5.8 Performance Metrics and Key Performance Indicators in Scrum

Effective Scrum implementation requires a continuous ability to assess whether the team is delivering value sustainably and with improving capability over time. Because Scrum operates through iterative, incremental delivery under evolving requirements, its metrics must capture flow efficiency, quality consistency, and team health rather than deviation from a static plan (Forsgren, 2018). The Scrum Guide does not prescribe specific metrics, but establishes the empirical pillars of transparency, inspection, and adaptation as the epistemological foundation for any measurement practice, with metrics serving to make the state of the system visible and enable informed adaptation at both the Sprint and Epic level (Schwaber, 2020).

A widely adopted conceptual distinction in agile measurement separates leading indicators from lagging indicators. Lagging indicators, such as defect counts or release frequency, reflect outcomes that have already occurred and are useful for retrospective evaluation. Leading indicators, such as backlog readiness or work-in-progress levels, signal the current health of the delivery process and enable proactive intervention before problems compound across iterations (Reinertsen, 2009). Effective measurement systems combine both types to support retrospective learning and real-time adaptation simultaneously.

Scrum metrics can be organised according to the dimension of team performance they address. Flow metrics measure the speed and predictability of work moving through the system, with Cycle Time and Lead Time representing the most widely used indicators of delivery responsiveness. Quality metrics capture the correctness and sustainability of delivered increments, addressing defect rates and adherence to the Definition of Done. Process metrics evaluate the stability and efficiency of team practices, including backlog readiness and the degree to which Sprint scope remains stable from commitment to completion (Forsgren, 2018; Reinertsen, 2009). Sustainable agile delivery cannot be reduced to velocity alone; teams that optimise exclusively for throughput risk accumulating technical debt, degrading requirement clarity, or eroding collaborative practices. A balanced metric system therefore requires coverage across all three dimensions.

A further consideration concerns the risk of measurement dysfunction. When individuals or teams are evaluated against narrow output metrics, behaviour tends to shift toward optimising the metric rather than the underlying goal it represents. In management literature, this phenomenon is described as Goodhart's Law: when a measure becomes a

target, it ceases to be a good measure (Goodhart, 1981). In Scrum contexts, this may manifest as artificially inflated story point estimates, premature task closure, or reduced investment in quality in favour of throughput. Robust KPI frameworks therefore pair output measures with quality and process indicators, creating a structure in which optimising one metric at the expense of another becomes visible and self-correcting.

Finally, the temporal scope of measurement matters. Sprint-level metrics provide short-cycle feedback suitable for immediate adaptation, while Epic-level aggregation reveals systemic trends invisible within a single iteration. A team may perform adequately within individual Sprints while exhibiting gradual degradation in quality or collaboration across the full Epic lifecycle. Measurement frameworks operating exclusively at the Sprint level therefore risk missing longitudinal patterns that only become apparent at a larger delivery horizon (Reinertsen, 2009).

6 Gamification

6.1 Definition of Gamification

In modern context gamification refers to systematic application of selected game design elements within environments that are not games. These elements include mechanisms such as points, badges, levels, feedback systems and challenges. Their purpose is to influence behaviour, strengthen engagement and enhance motivation. This understanding is broadly accepted in academic research and builds on the influential definition introduced in the conference paper *From Game Design Elements to Gamefulness: Defining "Gamification"*, which describes gamification as the application of game design elements in contexts that are not games (Deterding, 2011). The term itself was first introduced by British inventor Nick Pelling in 2002, originally to describe game-like user interface design intended to make digital transactions more efficient and enjoyable (Burke, 2014). Although his definition was limited in scope and impact at the time, it marked the linguistic origin of a concept that would later develop into a structured interdisciplinary methodology. The behavioural logic behind gamification, however, predates the term and can be traced to long-standing cultural and psychological practices (Chou, 2016; Burke, 2014).

Despite its digital framing, the behavioural logic behind gamification predates the term itself. For instance, the Boy Scouts' merit badge structure or mid-20th-century customer

loyalty programmes operated on mechanisms of symbolic reward and visible progress, relying on voluntary participation, social recognition, and goal-oriented progression. These systems anticipated modern gamification by employing motivational strategies closely resembling those later formalised in game design theory (Burke, 2014; Chou, 2016).

This perspective has since been expanded by various authors who emphasise different dimensions of gamification. One widely discussed definition conceptualises gamification as a process that enhances a service by enabling gameful experiences and supporting user value creation (Huotari, 2012), placing emphasis on the user's subjective perception rather than on the mere presence of game elements. In contrast, definitions emerging from business contexts highlight its instrumental function, describing it as the application of game mechanics and experience design to engage individuals and motivate them toward achieving specific objectives (Burke, 2014; Werbach, 2020). Motivational theory has also shaped how gamification is understood: the Octalysis framework introduces the term human-focused design to emphasise how gamification activates core human drives such as purpose, mastery, and social recognition, redirecting attention from surface-level incentives toward the psychological conditions required for sustainable engagement (Chou, 2016).

Although these perspectives differ in scope and emphasis, they contribute to a broader understanding of gamification as a flexible methodological framework applicable across diverse domains. Clarifying its conceptual boundaries in relation to serious games, game-based learning, and playful design remains essential for both theoretical precision and practical implementation (Deterding, 2011). Gamification does not aim to replicate games, but to harness their motivational architecture in support of specific real-world objectives (Burke, 2014; Werbach, 2020).

6.2 Motivational Foundations of Gamification

The effectiveness of gamification depends on the extent to which its design elements align with current motivational theory and empirical evidence. Rather than functioning as surface-level rewards, game components such as progress tracking, achievements, and adaptive challenges generate engagement when they respond to core psychological drivers such as autonomy, competence, purpose, and social recognition (Sailer, 2020).

Contemporary gamification research draws heavily from Self-Determination Theory (SDT), which explains how intrinsic motivation arises when individuals experience

autonomy, mastery, and relatedness in an activity. Game elements that support voluntary choice, offer clear feedback on skill improvement, and create opportunities for meaningful interaction contribute directly to these psychological needs. Specific gamified features such as badges and leaderboards have been shown to increase perceived competence and social relatedness, while personalisation and optional challenges support autonomy. Although intrinsic motivation is essential for long-term engagement, gamified systems often combine these mechanisms with extrinsic motivators, and their effectiveness depends largely on user interpretation. Rewards experienced as informative and autonomy-supportive can enhance motivation, whereas those perceived as controlling may diminish internal drive. Cumulative evidence confirms that the motivational impact of external incentives varies according to how they are structured and contextualised (Sailer, 2017; Ryan, 2017).

Flow theory also informs gamification design, particularly in contexts where performance and concentration are critical. When users face tasks with clear goals, immediate feedback, and optimal challenge relative to their skill, they are more likely to enter a state of focused immersion. Gamified systems that use progressive difficulty, level-based advancement, and real-time feedback reflect this structure and have been associated with increased task involvement and perceived enjoyment (Csikszentmihalyi, 2018).

The Expectancy-Value framework further helps explain motivational effects of gamification. Motivation tends to increase when users believe that effort leads to success, that success yields valued outcomes, and that those outcomes are personally meaningful. Progress indicators, milestone structures, and reward transparency contribute to reinforcing these links, particularly when gamified systems make success visible and the outcomes relevant (Sailer, 2020).

To support these mechanisms in practice, designers often refer to comprehensive models such as the Octalysis framework, which categorises eight core motivational drives, including achievement, ownership, unpredictability, and social influence, into practical strategies for gamified environments. These drives have been widely adopted in gamification practice and are increasingly integrated into empirical research (Chou, 2016).

6.2.1 Game Mechanics and Game Dynamics

One of the core building blocks of gamification is the use of game mechanics and game dynamics. Although these terms are sometimes used interchangeably, they refer to different levels of system design. Game mechanics are the specific rules, feedback structures, and

reward systems implemented within an application. Game dynamics, on the other hand, refer to the psychological and behavioural responses that emerge from how users interact with those mechanics.

Commonly used game mechanics in gamified systems include points, badges, leaderboards, levels, challenges, achievements, and progress indicators. These elements are used to structure user activity, provide feedback, and highlight achievement. For example, points quantify effort, levels represent stages of advancement, and badges function as symbolic recognition for specific accomplishments (Werbach, 2020).

Game dynamics arise when users respond to these mechanics in a way that shapes their motivation, attention, or emotional experience. Dynamics such as competition, progression, scarcity, and social comparison are not directly programmed but develop from how users interpret the system. A leaderboard may encourage competition, while a time-limited challenge may create urgency or focus. Unlike mechanics, dynamics depend on user perception and can differ across individuals or contexts (Chou, 2016).

Understanding the relationship between mechanics and dynamics is essential for designing effective gamified systems. If mechanics are misaligned with user needs or poorly contextualised, they may lead to disengagement, stress, or unintended negative behaviours. For example, a public ranking system without opportunities for self-paced progress may discourage less successful participants. Conversely, when mechanics provide appropriate challenge, timely feedback, and visible progress, they are more likely to support intrinsic motivation and sustained engagement (Sailer, 2017). Design approaches such as the MDA framework and Werbach's classification of game elements, encompassing feedback, rewards, goals, progression, resources, and social interaction, offer practical bases for designing cohesive and motivating systems (Werbach, 2020).

6.3 Benefits of Gamification

Gamification has attracted widespread attention due to its ability to influence user motivation, behaviour, and engagement across various domains. When applied thoughtfully, it can enhance participation, increase task performance, and support long-term goal adherence. These effects have been demonstrated in areas such as education, health promotion, organisational development, and user experience design (Koivisto, 2019).

One of the key benefits of gamification is its capacity to improve intrinsic motivation. By incorporating elements such as clear goals, immediate feedback, and a sense of progression, gamified systems can satisfy users' psychological needs for autonomy, competence, and relatedness. These needs are fundamental to sustained motivation and are strongly associated with greater effort, persistence, and satisfaction (Sailer, 2017). Well-designed gamification encourages users to focus on mastery and self-improvement rather than purely external rewards (Deci, 2017).

Gamification also increases engagement by introducing variety, structure, and challenge into otherwise routine or repetitive tasks, transforming low-interest activities into experiences perceived as enjoyable or purposeful. In digital environments, gamified features often lead to longer session durations, higher completion rates, and greater user retention (Hamari, 2014). In software development and IT project environments, gamification can improve developer focus, visualise sprint progress, and support consistent participation in recurring agile routines such as stand-ups or backlog updates.

In team or organisational contexts, gamification can support collaboration and recognition. Features such as collective challenges, contribution tracking, and social feedback encourage cooperative behaviour and make team effort more visible. Within agile teams, this may take the form of peer recognition, shared goals for sprint completion, or mechanisms that highlight cross-functional support during iterative delivery cycles. Additionally, gamification can help clarify expectations and objectives, which is particularly valuable in complex project environments where team motivation may fluctuate (Burke, 2014).

From a learning perspective, gamification supports active participation and knowledge retention through formative feedback, achievement tracking, and adaptive challenges. This is particularly relevant in software teams adopting new technologies or methods, where structured progression and immediate reinforcement can accelerate skill acquisition (Sailer, 2020). The effectiveness of gamification nonetheless depends on contextual factors such as user preferences, system design, and implementation quality; when aligned with user needs and task goals, it can produce measurable improvements in both individual and group outcomes (Landers, 2015).

6.4 Risks and Limitations of Gamification

A common limitation lies in the overuse of extrinsic rewards. When gamification relies too heavily on points, badges, or leaderboards without supporting autonomy and competence, users may become dependent on superficial incentives. This can diminish intrinsic motivation, especially if rewards are perceived as controlling rather than informative (Ryan, 2017; Pink, 2009). Once the extrinsic stimulus is removed or loses novelty, users often experience a drop in engagement or even aversion to the underlying task (Chou, 2016).

Another risk concerns the uniform application of game elements without consideration for individual differences. Users respond differently to competition, social comparison, or public ranking. What motivates one team member may demotivate another, particularly in mixed-ability teams. If such systems are not adapted to varying user profiles, they may foster disengagement or perceived unfairness (Zichermann, 2013).

Gamification can also lead to unintended behaviour if users begin to optimise for points rather than meaningful progress. In agile teams, this might manifest as a focus on rapid task closure at the expense of quality or cooperation (Zichermann, 2013). When performance metrics are rewarded uncritically, there is a risk of gaming the system or distorting team behaviour.

In project and IT environments, a further limitation is sustainability. Gamification often triggers an initial increase in motivation, but this effect can fade as novelty wears off or when users begin to predict the system's feedback. Without variation, new challenges, or dynamic reinforcement, gamification can become monotonous and lose its impact (Zichermann, 2013).

Poorly implemented gamification may also appear manipulative or trivialising in professional settings. Where game mechanics are perceived as coercive rather than empowering, users may lose trust in the system or reduce their willingness to collaborate. Long-term reliance on external control or pressure-driven systems can lead to fatigue or burnout, particularly when based on manipulative design principles (Chou, 2016). Gamification is therefore not universally beneficial; its success depends on thoughtful integration, sensitivity to context, and continuous adaptation.

7 Psychological Aspects and Player Typologies

7.1 Motivation in Work and Team Environments

Work motivation refers to the psychological processes that explain why individuals initiate work activities, how they distribute effort, and why they persist when tasks are challenging. In organizational contexts, motivation is reflected not only in observable effort and performance, but also in the reasons that guide work behaviour. This distinction matters because similar outcomes can stem from different forms of motivation that vary in stability and links to employee well-being (Gagné, 2005; Ryan, 2000).

A widely used framework for describing motivational regulation in work settings is the Self-Determination Theory (SDT), which conceptualizes motivation on a continuum from controlled to autonomous regulation. Controlled motivation is driven by external demands and pressures, such as rewards, punishments, or compliance with expectations. On the other hand, autonomous regulation, is self-endorsed and occurs when one perceives the work as personally meaningful and valuable. An important aspect of this approach is internalization, which explains how work demands that originate externally can become more self-determined if individuals integrate them into their own value system, and why the same organizational requirement can be experienced differently by different employees (Gagné, 2005; Ryan, 2000).

According to the SDT, the quality of motivation is associated with the satisfaction of the three innate psychological needs for **autonomy**, **competence**, and **relatedness**. In the organizational context, these needs function as a bridge connecting work conditions to motivation, because structures such as managerial style, evaluation practices, and social interaction can support or frustrate need satisfaction. A key implication is that tools such as rewards, targets, or monitoring do not have uniform motivational effects, their impact depends on whether they are experienced as informational and autonomy-supportive or as controlling (Deci, 1985; Ryan, 2017; Gagné, 2005).

Alongside need-based explanations, organizational motivation is also described through goal-directed self-regulation. Goal-setting theory explains how goals and feedback shape motivated behaviour by directing attention, mobilizing effort, supporting persistence, and encouraging strategy development. In work settings, goals help structure action and

make progress interpretable, while feedback supports adjustment of effort and approach over time (Locke, 2002).

In team environments, motivation is shaped not only by individual regulation and goals but also by social conditions. Psychological safety captures whether people perceive the team as safe for interpersonal risk-taking, such as admitting uncertainty, raising concerns, or discussing mistakes. When psychological safety is present, members are more likely to engage in learning behaviours such as speaking up and reflecting on mistakes, which are important for sustained engagement in complex collaborative work (Edmondson, 1999).

7.2 Individual Differences in Motivation and Behaviour

Even within the same team and under the same working conditions, employees can differ in what they experience as motivating, how they interpret feedback, and how they cope with pressure. In organizational psychology, these differences are treated as a predictable source of variation that helps explain why identical work conditions can produce different levels of engagement, persistence, and adaptive behaviour across individuals (Deci, 2017).

One way to describe these differences is to focus on how employees typically regulate their behaviour in work settings. In SDT terms, people differ in the extent to which they adopt autonomous versus controlled reasons for acting, and this shapes how they interpret external influences such as goals, evaluation, or rewards. Accordingly, the same intervention may be experienced as supportive by one person and as pressuring by another, depending on how the person internalizes or resists external demands (Gagné, 2005; Ryan, 2000; Ryan, 2017).

A second way to capture individual differences is through motivation profiles, rather than assuming that each employee has one dominant motive, motivation profiles describe common combinations of motives than can co-occur within individuals. This is approach is useful because it allows work motivation to be described as a configuration within individuals, which aligns better with the complexity of real workplace and these configurations can be linked to different patterns of behaviour and well-being (Gagné, 2015; Howard, 2016).

Differences in work behaviour are also connected to relatively stable dispositions and self-related beliefs. Personality describes enduring tendencies that influence how people

approach achievement, responsibility, and effort, which can affect their typical work behaviour and persistence. In addition, self-efficacy refers to beliefs about one's capability to perform tasks, and it is closely linked to persistence and coping when work becomes challenging. People with higher self-efficacy typically approach difficult tasks with greater confidence and are more likely to continue after setbacks, while lower self-efficacy is more likely to be associated with avoidance and faster withdrawal (Judge, 2002; Bandura, 1997).

Finally, individuals differ in how they define competence and success, which is often described through goal orientation. A mastery orientation focuses on developing competence through learning and improvement, whereas a performance orientation focuses on demonstrating competence and reducing the risk of negative evaluation. These orientations influence how people respond to challenge and feedback, which can translate into different levels of persistence, openness, and learning behaviour at work (Locke, 2002; Vandewalle, 1997).

Within the competence dimension specifically, goal orientation theory identifies a further distinction that is relevant for understanding individual behaviour in performance-driven environments. A mastery-approach orientation, also referred to as a learning orientation, is characterized by the pursuit of deep skill development, thoroughness, and adherence to quality standards, even when this requires additional effort or slower output. In contrast, a performance-approach orientation is characterized by a focus on demonstrating ability through visible achievement, speed of execution, and tangible impact. Both orientations reflect a genuine expression of the competence need as defined by SDT, but they manifest in recognizably different behavioural patterns in collaborative work settings. Individuals with a mastery-approach orientation tend to invest in precision, error prevention, and long-term sustainability of their output, while those with a performance-approach orientation tend to prioritize efficiency, responsiveness, and rapid delivery of results. This distinction is empirically supported and has been shown to predict differences in how individuals approach feedback, challenge, and task selection (Vandewalle, 1997; Elliot and McGregor, 2001). In the context of gamification design, acknowledging this differentiation within the competence need allows for more precise alignment between individual motivational profiles and the incentive structures offered by the system, rather than treating all competence-driven individuals as a homogeneous group.

7.3 Player Typologies (Bartle, HEXAD)

Player typologies are used to describe systematic differences in what people find engaging and rewarding in gameful systems. In a workplace context, their main purpose is not to label employees, but to support the idea that a single gamification design may fit some users while missing others, which is consistent with the broader argument that motivational effects depend on interpretation and person–context fit (Tondello, 2019; Tondello, 2016).

One of the best-known typologies is Bartle's model, originally developed for Multi-User Dungeons. Bartle distinguishes four player types based on two underlying dimensions. a preference for action versus interaction and a preference for player-oriented versus world-oriented goals.

Achievers are characterized by an emphasis on measurable progress, completion of objectives, and accumulation of indicators of success.

Explorers are defined by interest in discovery and understanding, including curiosity about how the system operates and enjoyment of uncovering new content.

Socializers are primarily oriented toward interpersonal contact, placing value on communication, relationship-building, and shared activity.

Killers are associated with competition and influence over other players, including enjoyment of dominance, winning, and impacting others' outcomes.

Although this model was created for online game environments, it is often referenced in later gamification discussions because it provides an intuitive way to think about different motivational interests, such as progress and achievement, curiosity and exploration, social connection, or competitive influence (Bartle, 1996).

For workplace gamification, a more directly targeted model is the HEXAD user typology, proposed as a typology for gamified systems rather than for games. HEXAD describes six user types:

Philanthropists are characterized by motivation related to purpose and contribution, with preference for helping others and supporting meaningful goals.

Socializers are motivated by interaction and connection, valuing communication and social participation.

Free Spirits are associated with autonomy and self-expression, emphasizing freedom of choice, creativity, and exploration.

Achievers are defined by mastery and challenge, valuing progress, learning, and the accomplishment of difficult tasks.

Players are characterized by a strong orientation toward external rewards and incentives, such as points, prizes, and tangible benefits.

Disruptors are associated with change and experimentation, showing preference for influencing systems, testing boundaries, and introducing new approaches.

HEXAD is more compatible with organisational settings than Bartle's model because it explicitly addresses both intrinsic and extrinsic motivational drivers that appear in work-related gamification designs (Marczewski, 2015). It has been operationalised through a standardised questionnaire, the Gamification User Types Hexad Scale, which enables individual tendencies across the six types to be assessed in a consistent and comparable manner (Tondello, 2016; Tondello, 2019).

Player typologies do not predict behaviour deterministically, but they offer a clear vocabulary for discussing user diversity and for explaining why the same gamified intervention may not be equally engaging for all team members (Bartle, 1996; Marczewski, 2015; Tondello, 2016).

7.4 Implications for Motivation in Scrum Teams

In Scrum Teams, motivational quality is especially important because effective work depends on self-regulation, active participation, and sustained engagement across iterations. A useful theoretical lens is Self Determination Theory, which differentiates autonomous and controlled regulation and explains motivational quality through the satisfaction of autonomy, competence, and relatedness. In this context, autonomy is reflected in experienced choice and ownership of work decisions, competence in clear informational feedback and opportunities to improve mastery, and relatedness in respectful, supportive collaboration (Deci, 1985; Gagné, 2005; Deci, 2017).

Goal-focused self-regulation is another relevant mechanism because Scrum work is repeatedly structured around short-term goals and frequent progress evaluation. Goal-setting theory explains how specific goals and feedback direct attention, mobilize effort, support persistence, and influence strategy selection. For motivation in Scrum Teams, this implies that goals and feedback should be designed so that progress is interpretable and meaningful, rather than reduced to narrow metric optimization (Locke, 2002).

Team motivation in Scrum also depends on social conditions that influence whether members contribute actively, share concerns, and learn from mistakes. Psychological safety is a key concept for describing whether interpersonal risk-taking is experienced as acceptable, which affects behaviours such as speaking up, admitting uncertainty, and requesting help. These behaviours are closely linked to learning and improvement, and they shape whether engagement is sustained or becomes cautious and minimal (Edmondson, 1999).

These implications matter directly for workplace gamification, because gamified systems can change perceived autonomy, competence feedback, goal salience, and social comparison. Reviews of gamification research consistently show that outcomes vary across contexts and depend on design choices and user interpretation, meaning gamification cannot be treated as uniformly effective in work settings. Experimental evidence further supports that specific game elements tend to produce specific psychological effects, rather than "gamification" functioning as one generic intervention (Sailer, 2017; Koivisto, 2019; Seaborn, 2015; Hamari, 2014; Hamari, 2017).

For Scrum Teams, this supports cautious handling of competitive elements and careful prioritization of competence-supportive feedback. Individual leaderboards and strong social comparison can increase evaluation pressure and may conflict with psychologically safe interaction, while cooperative or team-based elements are more compatible with a climate that supports learning behaviours. At the same time, progress indicators, meaningful milestones, and informative feedback can strengthen goal clarity and competence experiences when they are framed as developmental rather than controlling (Ryan, 2017; Sailer, 2017; Locke, 2002; Edmondson, 1999).

A further implication is that gamification in Scrum Teams should reflect heterogeneity among team members, because individuals differ in what they find engaging and in how they respond to motivational cues. The HEXAD model provides a structured typology of user preferences in gamification, and its validated scale supports designing a more balanced set of motivational affordances instead of relying on a single engagement style (Tondello, 2016).

Finally, existing Scrum-focused gamification studies show that gamification has been applied to support engagement with Scrum-related practices, but they also indicate that results depend on design quality and fit with team needs. This reinforces the view that gamification in Scrum Teams should be treated as a targeted motivational design problem

aligned with psychological mechanisms and team conditions, not as a generic overlay of points and rewards (Marques, 2020).

8 Gamification in Agile environments

8.1 Gamification in Agile Teams

The integration of gamification into agile software development is frequently proposed as a mechanism to sustain team motivation and engagement, particularly for repetitive or administrative tasks such as project tracking, documentation, and effort estimation. In agile settings, gamification is therefore most commonly introduced as a practical layer intended to support day-to-day execution of agile practices within iterative cycles, rather than as an isolated engagement initiative (Marques, 2023).

This focus is important because Scrum is grounded in empiricism and in a recurring cycle of inspection and adaptation realised through defined events and artifacts within each Sprint. Gamification mechanisms that introduce additional feedback, visibility rules, or recognition signals can therefore influence how teams interpret progress and how they adjust coordination across timeboxed iterations. In practice, published interventions commonly operationalise gamification through tool-traceable behaviours, especially task or user story completion, because these behaviours are readily captured in standard agile toolchains; however, this tendency may bias interventions towards measurable proxies rather than towards behaviours that more directly strengthen inspection and adaptation (Schwaber, 2020; Marques, 2023).

Beyond delivery-oriented activities, research also extends gamification towards agile-adjacent practices that rely strongly on communication and collaboration. Human-centered agile-based requirements engineering studies, for example, examine gamification as a way to support stakeholder involvement and engagement during requirements-related activities, thereby broadening the focus from downstream completion metrics towards upstream interaction quality and shared understanding. Empirical research in software engineering further indicates that outcomes may be mediated by how developers engage with gamified systems rather than by game elements alone, with developer engagement and job satisfaction treated as relevant pathways through which gamification may influence results (Stol, 2021; Fatima, 2024).

Evidence on gamification in agile teams remains limited and methodologically uneven, which restricts strong conclusions about sustained effects in industrial settings. At the same time, the literature increasingly acknowledges the possibility of unintended effects, including undesirable competitive dynamics and "gaming the system." Although some systematic evidence on negative effects originates in adjacent contexts, the underlying risk mechanisms are relevant for agile teams because they can undermine trust and distort process signals that teams rely on when inspecting and adapting their way of working. Agile-specific research therefore also proposes governance-oriented approaches, including structured processes for identifying, assessing, and monitoring gamification risks in Scrum contexts (Schwaber, 2020; Elsalmy, 2025; Almeida, 2023; Pedreira, 2015; Marques, 2023).

8.2 Existing Approaches to Gamification in Scrum

Existing approaches to gamification in Scrum can be meaningfully grouped by the point of integration, tool-integrated approaches embedded in the issue-tracking workflow, workflow-oriented approaches that gamify the management of work across the delivery process, and event-focused approaches that gamify selected Scrum events, especially those aimed at improving team interaction and reflection. This categorisation follows the structure of Scrum, where work is organised through defined events and artifacts and coordinated through recurring feedback cycles (Marques, 2023; Schwaber, 2020).

A tool-integrated approach consists of solutions designed to support Scrum adoption and practice adherence by embedding gamification into the team's existing toolchain. These approaches typically connect feedback and rewards to tool-traceable actions and provide dashboards or progress indicators to reinforce behaviours that the intervention defines as desirable. Empirical evidence from a Jira-integrated gamification solution for improving Scrum adoption illustrates this approach and highlights that the effectiveness and acceptance of such solutions depend strongly on rule design and the perceived fairness of scoring when tool events are used as behavioural proxies (Marques, 2020).

A workflow-oriented approaches gamify how work is managed across the process, sometimes in Scrum-like or hybrid settings. These approaches may use points, badges, or other mechanisms to encourage patterns such as completing work, sharing information, or cooperating, and they can introduce constraints or incentives that shape prioritisation and coordination. A longitudinal case study of a Scrumban-integrated gamification approach

demonstrates this logic in a Scrum-related environment, while also indicating that hybrid process choices affect direct comparability with implementations strictly aligned to Scrum (Yilmaz, 2016).

A third approach targets Scrum events, most notably the Sprint Retrospective, where gamification is operationalised as game-based facilitation rather than as tool scoring. In this approach, the primary objective is to improve participation, structure, and the quality of reflection, which is relevant because the retrospective represents Scrum's formal mechanism for inspection and adaptation of team working practices. Evidence from multiple action research on game-based Sprint retrospectives reports improvements in recurring retrospective difficulties, while also showing that outcomes depend on team context and facilitation quality, which limits the transferability of uniform "plug-and-play" solutions (Przybyłek, 2021; Schwaber, 2020).

Beyond direct gamification of Scrum workflows and events, some approaches address practices that influence Scrum outcomes indirectly, such as requirements-related collaboration. Research on human-centred agile-based requirements engineering examines gamification as a way to increase stakeholder involvement and engagement during requirements activities, which can be relevant for Scrum teams insofar as upstream clarity and shared understanding affect backlog quality and iteration effectiveness (Fatima, 2024).

Across these approaches, the agile-focused evidence indicates substantial variation in design goals, mechanics, and evaluation practices, and it does not converge on a single standard implementation pattern for Scrum. A broader mapping study in software engineering supports this observation by describing fragmentation and methodological heterogeneity in gamification research, but it should be treated as contextual support rather than Scrum-specific evidence (Pedreira, 2015; Marques, 2023).

8.3 Identified Limitations of Current Approaches

One recurring limitation in existing Scrum gamification approaches is the tendency to build interventions around what can be easily captured in the team's tooling. The agile-focused review reports that many initiatives operationalise gamification through tool-traceable activities, which makes implementation straightforward but can shift attention toward measurable proxies rather than behaviours that directly strengthen inspection and adaptation in Scrum practice. As a result, interventions may increase visible activity without

necessarily improving the quality of coordination, learning, or decision-making within the Sprint cycle (Schwaber, 2020; Marques, 2023).

A second limitation concerns acceptance and perceived fairness in cross-functional teams. Tool-integrated designs depend on local workflow conventions and on how different roles interact with the toolchain, so scoring rules can unintentionally advantage some contributions while underrepresenting others. Field evidence from a Jira-integrated gamification solution for Scrum adoption highlights challenges related to sustaining use over time and to the perceived fairness of point allocation under realistic distributions of work, indicating that rule design becomes a critical success factor (Marques, 2020).

A third limitation relates to the scope and transferability of existing approaches. Some interventions are evaluated in Scrum-like or hybrid settings, such as Scrumban, which can be relevant for practice but limits direct comparability with interventions intended for Scrum in its standard form. Other approaches focus on a single Scrum event, most commonly the Sprint Retrospective, and therefore address only one part of the overall feedback system through which Scrum teams coordinate and improve across multiple Sprints (Schwaber, 2020; Przybyłek, 2021; Yilmaz, 2016).

Methodological limitations also remain prominent. The agile-focused systematic review identifies a small number of eligible studies and weaknesses in reporting outcomes and challenges, which constrains generalisation. This is consistent with the broader software engineering mapping literature, which points to fragmentation and heterogeneous evaluation designs and calls for stronger, more comparable empirical studies, particularly with longitudinal evidence in industrial contexts (Pedreira, 2015; Marques, 2023).

Finally, existing approaches often under-specify how negative or unintended effects are identified and mitigated. Systematic mapping work in adjacent contexts reports risks such as undesirable competitive dynamics and "gaming the system," which are relevant mechanisms to consider in agile teams because they can undermine trust and distort the process signals needed for inspection and adaptation. Agile-specific research responds to this issue by proposing explicit risk management processes for Scrum-oriented gamification, which implies that safeguards and monitoring should be treated as integral to design rather than as optional additions (Schwaber, 2020; Elsalmy, 2025; Almeida, 2023).

8.4 Research Gap and Justification for a New Gamification Model

Gamification in Scrum is currently implemented mainly through isolated, narrowly scoped solutions rather than through a unified model that can be applied consistently across an epic. Although Scrum is a frequent target within agile gamification, the empirical basis remains limited and methodologically uneven, which restricts strong claims about durable effects in industrial teams and leaves insufficient guidance for long-term application (Pedreira, 2015; Marques, 2023).

Many implementations operationalise gamification by attaching feedback and rewards to tool-traceable behaviours, particularly task and user story completion, because these behaviours are straightforward to capture in common toolchains. This lowers implementation effort, but it can also steer attention toward measurable proxies instead of behaviours that strengthen transparency, inspection, and adaptation across the Sprint cycle, which are central to Scrum's empiricism. A model intended for use across multiple Sprints therefore needs to specify how gamification supports the full Scrum feedback system rather than reinforcing a limited set of tracked actions (Marques, 2023; Schwaber, 2020).

Sustainable use depends on perceived legitimacy within a cross-functional team. When scoring is derived from tool events, differences in roles and in how work is represented can create systematic bias and fairness concerns, which reduces acceptance and long-term adherence to the intervention. Field evidence from a Jira-integrated solution designed to support Scrum adoption highlights that rule design and fairness issues are central barriers to sustained usage, indicating that a new model should explicitly define role-aware representation of contributions and recognition (Marques, 2020).

Unintended effects provide an additional justification for a new model. Competitive dynamics and "gaming the system" are documented risk mechanisms, and in Scrum these mechanisms can be especially harmful because they can undermine trust and distort the process signals needed for inspection and adaptation. A model intended for practical application therefore requires built-in safeguards, monitoring, and mitigation procedures, and agile-specific work proposes structured risk management processes for gamification in Scrum contexts that treat risk handling as part of an iterative cadence (Schwaber, 2020; Elsalmy, 2025; Almeida, 2023).

A further gap concerns the absence of a systematic way to connect motivational theory to concrete Scrum structures and to adapt the intervention to heterogeneous teams over time.

Existing approaches rarely specify how motivational needs or preferences translate into rule design across different Scrum events and artifacts, and they typically lack adaptive mechanisms that can shift attention, incentives, or feedback to match evolving team conditions across an epic. Addressing these gaps requires a model that is explicitly designed for epic-level continuity, role and contribution diversity, and adaptive adjustment of gamification focus while maintaining safeguards against metric gaming (Elsalmy, 2025; Marques, 2020; Marques, 2023; Schwaber, 2020).

9 Design of the Adaptive Agile Gamification Framework

9.1 Conceptual Framework and Strategic Alignment

The proposed framework is designed not as a superficial layer of entertainment but as a strategic management instrument whose primary objective is to align intrinsic human motivation, as described by Self-Determination Theory, with the operational needs of an Agile development lifecycle. The methodology is platform-agnostic: while it can be visualised in tools such as Notion, Jira, or Azure DevOps, its value lies in the logical rules and psychological triggers it introduces into the team's workflow rather than in any specific software implementation. Its design is adaptive, responding to project health through a data-driven approach that adjusts incentive structures in real time based on observable integrity indicators.

9.1.1 Principles of the Adaptive Agile Gamification Framework

Four foundational principles govern the methodology and together distinguish it from shallow gamification approaches characterised by points-chasing, social fatigue, or short-lived engagement spikes.

The first principle, **Holistic Integrity**, establishes that the framework does not reward speed or output in isolation. Project health is decomposed into four complementary dimensions, each capturing a distinct aspect of delivery quality, and the incentive structure ties collective rewards directly to the aggregate health of all four dimensions. This multi-dimensional approach prevents the common failure mode in which velocity is optimised at the expense of quality, documentation, or team cohesion.

The second principle, **Systemic Inclusivity**, ensures that the framework is applicable to all members of a cross-functional Scrum Team rather than favouring roles whose contributions are easily quantifiable. By grounding incentives in universal psychological drivers and providing multiple pathways for earning recognition, the framework creates meaningful participation opportunities for Product Owners, UX Designers, Scrum Masters, and other roles whose work is often underrepresented in developer-centric gamification models.

The third principle, **Active Orchestration**, positions the Scrum Master as the active governor of the framework. Rather than operating as a static background system, the

framework provides governance mechanisms that allow the Scrum Master to recalibrate the incentive focus based on real-time integrity data, directing the team's attention toward dimensions that require immediate strengthening.

The fourth principle, **Ethical Guardrails**, ensures that participation in the gamified layer is voluntary. Team members who are not motivated by gamified elements can contribute to the project and benefit from collective rewards without engaging with the individual progression system. Diagnostic integrity is maintained by collecting project-wide KPI data regardless of individual participation choices, ensuring that project health is accurately assessed without coercive participation.

9.1.2 Integration with Scrum Events and the Epic Lifecycle

Two temporal layers operate simultaneously within the methodology: the high-frequency feedback loop of the Sprint and the long-term strategic horizon of the Epic lifecycle. This dual-layered scope is essential because many integrity issues, such as gradually accumulating technical debt, slowly declining collaboration quality, or progressively weakening backlog preparation, are difficult to detect within a single iteration but become clearly visible when tracked longitudinally across multiple Sprints. By anchoring major progression milestones and collective reward accumulation in the Epic rather than in individual Sprints, this anchoring provides sustained motivation that extends beyond the two-to-four-week iteration cycle.

Integration with the existing Scrum cadence is achieved without introducing additional ceremonies or administrative overhead. Individual XP is generated automatically at the moment of task completion based on the item's story point estimate and applicable quality conditions, providing immediate feedback that reinforces competence without requiring manual intervention. Pillar data updates continuously as work items move through the Sprint cycle, making the current state of project integrity visible at any point during the iteration.

At Sprint boundaries, the framework's governance layer activates naturally within existing Scrum events. During Sprint Planning, the most recent pillar data informs decisions about where to direct the team's incentive focus for the upcoming iteration. Team members may also identify and commit to supplementary activities outside the primary Sprint Backlog that contribute to integrity dimensions requiring attention, provided these are prioritised only after core delivery commitments are established. At the Sprint Retrospective, collective rewards are calculated based on the Sprint's overall integrity performance and distributed to

the Team Treasury, and team members have the opportunity to reflect on their current behavioural archetypes and adjust them for the next iteration if their goals or the project's needs have shifted. The Daily Scrum remains strictly operational, with the framework functioning as a passive visibility layer unless integrity data indicates a condition requiring immediate attention.

This integration model ensures that the gamified layer amplifies the existing Scrum feedback loop rather than competing with it, and that incentive mechanisms remain subordinate to the team's primary delivery objectives at all times.

9.2 Team Diagnostics and Behavioural Archetype Mapping

Bridging the gap between project requirements and individual engagement requires a structured diagnostic approach that identifies each team member's motivational drivers and translates them into a gamified identity that authentically reflects their professional strengths and preferences. The diagnostic process is governed by a hierarchical selection model in which Self-Determination Theory (SDT) serves as the primary engine for functional archetype assignment, while the HEXAD framework provides a secondary layer of motivational personalisation.

9.2.1 Design of the Character Charter

The Character Charter is the foundational diagnostic instrument through which each team member establishes their gamified identity. Rather than functioning as a static profile assigned once and forgotten, the charter is designed as a dynamic interface that documents the user's psychological and professional alignment and supports periodic recalibration to prevent archetype stagnation over the course of the Epic. The diagnostic sequence proceeds through three layers, each building on the previous one to produce a progressively more personalised engagement profile.

The first layer, SDT-Based **Archetype Selection**, forms the primary directive. Team members navigate a deductive decision path rooted in the three core psychological needs identified by Self-Determination Theory: Autonomy, Competence, and Relatedness. This path leads to the assignment of a Hero Archetype that reflects the individual's dominant motivational driver. Those primarily driven by competence are further differentiated into two archetypes based on whether their orientation tends toward speed and impact or toward

precision and craftsmanship, a distinction whose theoretical basis is addressed in the archetype definitions below. Individuals whose primary driver is Autonomy are mapped to the Ranger archetype, while those most motivated by Relatedness adopt the Cleric archetype.

The second layer, **HEXAD Profiling**, adds motivational personalisation to the primary archetype. Once the SDT-based archetype is established, the HEXAD User Type Scale is applied to determine the narrative framing of the individual's engagement with the system. By identifying a secondary user type, the team member personalises how their archetype interacts with feedback and recognition mechanisms. An Achiever orientation, for instance, emphasises status-based feedback and visible progression markers, whereas a Socialiser orientation shifts acknowledgment toward team impact and collaborative contribution.

The third layer, **Utility and Preference Mapping**, completes the charter by documenting which non-monetary rewards resonate most strongly with the individual. By recording preferences for outcomes such as educational opportunities, autonomous time, team recognition, or equipment, the charter ensures that the reward economy can be tailored to sustain intrinsic motivation throughout the Epic lifecycle rather than relying on generic incentives that lose relevance over time.

9.2.2 Mapping Psychological Profiles to Hero Archetypes

The core of the diagnostic methodology lies in the alignment of psychological drivers with professional contributions, achieved by mapping the three fundamental constructs of SDT (Autonomy, Competence, and Relatedness) onto four distinct hero archetypes. These archetypes do not represent job titles such as Developer, Tester, or Product Owner but instead describe the behavioural profile and psychological approach an individual brings to any agile task, ensuring that the mapping is inclusive and applicable to all members of a cross-functional team.

The mapping produces four archetypes from three SDT constructs because the Competence need is differentiated into two distinct orientations. Self-Determination Theory defines competence as a single psychological need, and the framework does not challenge this position. However, within the practical context of agile delivery, the experience of competence manifests in two recognisably different behavioural patterns: one oriented toward efficient execution and rapid impact, and another oriented toward thoroughness, precision, and adherence to quality standards. This differentiation is consistent with the

established distinction between mastery-approach and performance-approach goal orientations, where mastery-approach focuses on developing deep competence through careful learning and performance-approach focuses on demonstrating competence through visible achievement. The framework operationalises this distinction as a deliberate design choice: rather than collapsing both orientations into a single archetype and losing diagnostic resolution, it separates them into the Barbarian (performance-approach, speed and impact) and the Paladin (mastery-approach, quality and craftsmanship), providing the incentive system with a finer-grained understanding of how individual team members experience and express competence in their work.

The four archetypes are defined as follows:

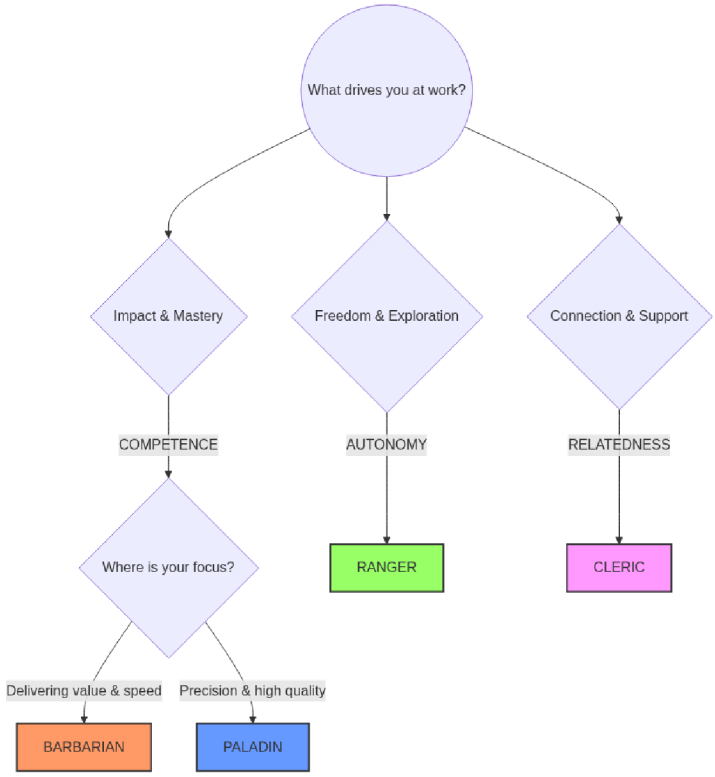

Figure 1: SDT-Based Archetype Selection Model

Source: author's own elaboration

1. **Barbarian** (Competence, performance-approach): Defined by mastery through efficiency, the Barbarian focuses on maximizing throughput and eliminating immediate process bottlenecks. This profile is essential for maintaining project momentum and "flow." In practice, this manifests as a team member closing high-priority tickets or pushing for a rapid MVP launch to gather early feedback while still adhering to the core DoD.

2. **Paladin** (Competence, mastery-approach): Motivated by mastery through craftsmanship, the Paladin prioritizes technical or functional integrity and strict adherence to the Definition of Done (DoD). This role is exemplified by an individual dedicated to refactoring legacy code ensuring security standards, or flawlessly implementing accessibility requirements, even under delivery pressure.
3. **Ranger** (Autonomy): Centred on the need for self-governance and exploration, the Ranger specializes in pathfinding and reducing product uncertainty. They are most effective during discovery phases, prototyping new technologies, or investigating market trends to clear the path for the rest of the team.
4. **Cleric** (Relatedness): Defined by the need for connection and team synergy, the Cleric focuses on knowledge sharing and maintaining a healthy collaborative environment. Their contribution is measured by the "unblocking" of colleagues, facilitating difficult communication, and mentoring junior members to improve collective performance.

Crucially, archetypes are not permanent assignments but behavioural modes that any team member can adopt based on the nature of their current task or the needs of the project at a given point in the Epic. A developer may operate as a Barbarian during a high-pressure delivery phase and switch to a Paladin orientation when the team enters a stabilisation period, while a Product Owner may adopt a Ranger mode during market research and shift to a Cleric mode when facilitating stakeholder alignment. This fluidity is reinforced through the recalibration opportunity built into the Sprint Retrospective, where team members reflect on whether their current archetype still aligns with their goals and the project's evolving needs.

To demonstrate that archetypes are independent of job titles and applicable across all roles, the following matrix illustrates how each archetype manifests in different project roles:

Table 3: Proposed behavioural archetypes for gamification in Scrum teams

Archetype	Core motivational orientation	Typical behaviour in Scrum	Suitable gamification elements
Barbarian	speed, challenge, visible progress	pushes execution, removes friction, seeks fast wins	timed challenges, progress streaks, sprint quests
Paladin	quality, rules, mastery	protects standards, Definition of Done, reliability	mastery badges, quality milestones, completion standards
Ranger	exploration, discovery, experimentation	tests alternatives, seeks innovation, explores options	discovery tasks, experimentation points, innovation quests
Cleric	support, cohesion, collaboration	helps others, connects people, supports team safety	peer recognition, support quests, mentoring rewards

Source: author's own elaboration

9.3 The Four Pillars of Agile Integrity

The structural core of the framework is a system of four integrity pillars that collectively define what it means for an Agile project to be healthy across an entire Epic. Project health cannot be adequately captured by a single metric or a single category of performance, because optimising for one dimension inevitably creates blind spots in others. The framework therefore decomposes project integrity into four complementary dimensions, each responsible for monitoring a distinct aspect of the delivery system: Process Flow, Technical Excellence, Product Clarity, and Team Synergy. These pillars are not independent scores but form a causally interconnected structure, and their aggregate produces the Project Integrity Score (PIS), which serves as the primary input for the framework's collective reward calculation, with each pillar contributing equally so that no single dimension can be neglected without affecting the entire team's outcome. Crucially, all four pillars are defined as shared responsibilities of the entire Scrum Team: no pillar is assigned to or owned by a specific role, and while certain metrics may be more directly influenced by particular roles, the pillar-level score always reflects the aggregate performance of the delivery system as a whole.

9.3.1 Definition of Strategic Modules

Each pillar represents a strategic module that monitors one dimension of project integrity across the Epic lifecycle. The four pillars are defined as follows:

Process Flow monitors whether work items move through the Sprint cycle in a smooth and predictable manner, and its focus on consistency distinguishes it from output-oriented indicators that simply measure how much work a team completes. A team that delivers all committed items in a final-day burst achieves the same aggregate output as one that delivers incrementally throughout the Sprint, yet the former pattern introduces compounding risk across an Epic: when delivery is unreliable, the Scrum Master loses the ability to forecast capacity accurately, bottlenecks remain hidden until they become critical, and decisions about where to direct the team's attention are based on misleading signals. Within the framework, Process Flow serves as an early-warning system for systemic dysfunction, because erratic delivery patterns typically originate from upstream issues such as poor task decomposition, unclear acceptance criteria, or hidden dependencies that may not surface in output metrics until several Sprints have passed. By tracking flow at the pillar level continuously across the Epic, the framework makes it possible to detect these chronic patterns before they escalate into delivery failures that would be far more costly to address.

Technical Excellence tracks whether the team's delivery chain consistently produces outputs that meet defined quality standards, treating quality not as the responsibility of any single role but as an emergent property of the entire delivery system that spans from input specifications through implementation and testing to final verification against the Definition of Done. A defect that reaches production may have originated at any point in this chain: ambiguous specifications written by the Product Owner, insufficient test coverage designed by QA, incomplete implementation by a developer, or a Scrum Master's failure to enforce adherence to agreed standards. Rather than attributing fault to one role, the pillar measures the aggregate quality of the output as a shared consequence of how the entire team has functioned during the Sprint. Within the Epic lifecycle, Technical Excellence serves as a safeguard against the accumulation of technical debt, a risk that is easy to underestimate in any single Sprint but that compounds severely over multiple iterations. Bypassing quality standards or tolerating elevated defect rates may appear to carry minimal consequences in isolation, but across an Epic these compromises accumulate as unresolved defects generate cascading rework, shortcuts in the Definition of Done erode the reliability of future

Increments, and the team's capacity becomes progressively consumed by remediation rather than new value delivery. The pillar therefore protects both the current quality of outputs and the long-term sustainability of the development process.

Product Clarity assesses whether the team operates with a shared and accurate understanding of what is being delivered and why. Where Technical Excellence evaluates the quality of outputs, Product Clarity evaluates the quality of inputs: the precision of requirements, the stability of scope, and the degree to which the team's interpretation of work items aligns with the Product Owner's intent. When this alignment is absent, effort is wasted on rework that could have been prevented entirely through clearer preparation and more effective communication during refinement. The pillar encompasses two distinct levels of understanding. The first is task-level clarity, which concerns whether individual work items are specified with sufficient detail, including well-defined acceptance criteria and resolved dependencies, so that the team can implement them correctly on the first attempt. The second is contextual clarity, which concerns whether the team understands the business rationale and user needs behind each item, since without this deeper understanding developers and testers tend to make assumptions that may satisfy the literal specification while missing its underlying purpose. Scope stability represents a particular concern across the Epic lifecycle, because the Scrum framework prescribes that no changes should endanger the Sprint Goal once a Sprint has begun, yet in practice mid-Sprint scope changes do occur with varying frequency and justification. The framework treats all scope changes as measurable events that affect the pillar score, on the grounds that the Product Owner, as a full member of the Scrum Team, shares responsibility for maintaining Sprint integrity. However, to account for situations where scope changes are genuinely forced by external factors beyond the team's control, the Scrum Master may activate a compensation mechanism for the affected Sprint. In practice, this takes the form of a temporary adjustment to the critical threshold of the Scope Stability KPI, shifting it to reflect the reduced scope control realistically available to the team under the given external conditions. The adjusted threshold applies only to the Sprint in question and reverts to its default value in the following iteration. The decision to activate compensation and the degree of threshold adjustment rest with the Scrum Master, who documents the rationale during the Sprint Retrospective to preserve transparency and prevent routine misuse of the mechanism.

Team Synergy examines the operational quality of collaboration within the Scrum Team by focusing on observable cooperative behaviours that directly affect delivery

capacity: how quickly blockers are resolved, whether knowledge circulates effectively among team members, and whether Scrum events fulfil their intended purpose rather than becoming unproductive routines. A team composed of individually skilled members can still underperform significantly if blockers remain unresolved for extended periods, if knowledge stays siloed within individuals or sub-groups, or if coordination events fail to produce the shared understanding and actionable decisions they are designed to generate, and these failure modes are often invisible in output-oriented metrics even as their cumulative effects compound over multiple Sprints. The pillar also reflects a measurement challenge that the framework addresses explicitly. Unlike the other three pillars, where KPIs can be derived primarily from tool-traceable data, collaborative behaviour is only partially visible in project management tooling, since much of the most valuable knowledge sharing and mutual support happens informally through conversations, quick explanations, or spontaneous help. The framework therefore adopts a dual-source approach, combining tool-traceable indicators that provide continuous mid-Sprint signals with structured peer assessment at the Sprint Retrospective that serves as a corrective layer capturing informal collaboration not recorded in tooling. This ensures both timely visibility for the Scrum Master during the Sprint and measurement accuracy in the final pillar score.

Although each pillar monitors a distinct dimension of project health, they do not operate in isolation. The four pillars form a causal system in which problems in one area propagate into others, often before they become visible in the originating pillar's own metrics. Product Clarity functions as the primary upstream pillar: vague acceptance criteria and unstable scope generate implementation errors that surface as rising rework in Technical Excellence and increasing Cycle Time variance in Process Flow. Process Flow acts as a precondition for the other three, since unstable delivery compresses review and testing windows, limits opportunities for incremental feedback, and creates uneven workload distribution. Technical Excellence reflects the downstream consequences of both Clarity and Synergy failures, making persistent decline in quality a reliable signal that compounding issues exist elsewhere in the system. And Team Synergy functions as an enabling condition for all other pillars, because when collaboration deteriorates, its effects propagate broadly through slower unblocking, isolated decision-making, and degraded communication during refinement. This causal interconnection is a deliberate design property: the framework treats cross-pillar patterns as diagnostic signals that reveal root causes rather than symptoms,

giving the Scrum Master a richer and more actionable picture of project health than any single pillar could provide on its own.

To fulfil this diagnostic function, all four pillars are evaluated at the end of every Sprint, and their scores are tracked as a continuous trend across the entire Epic. This longitudinal perspective is essential because many integrity issues, such as gradually rising technical debt or slowly declining collaboration quality, are difficult to detect within a single iteration but become clearly visible when pillar scores are compared over multiple Sprints. The pillar system is not designed as a passive monitoring layer: the scores serve as direct inputs for the framework's active governance mechanisms, enabling the Scrum Master to adjust the team's focus and incentive structure in response to emerging risks before they escalate into systemic problems.

9.3.2 Selection of Operational Metrics and KPIs

Each pillar is operationalised through three complementary KPIs that together capture the full scope of the dimension it monitors. The selection of exactly three metrics per pillar reflects a deliberate balance: fewer would leave critical aspects of each dimension unmeasured, while a larger set would increase administrative overhead and dilute the diagnostic clarity that the framework depends on. Together, the twelve KPIs form the framework's data layer, referred to as the Golden Twelve, which provides the inputs for pillar score calculation and the incentive model.

The metrics were selected according to three criteria. First, each KPI must capture a distinct aspect of its pillar that the other two do not, so that the three metrics within a pillar provide complementary rather than redundant information. Second, the metrics must be derivable from data that is either automatically generated by standard project management tooling or collectible through lightweight structured assessment, ensuring that the framework remains implementable without specialised infrastructure. Third, each metric must be interpretable at the team level as a shared outcome rather than as an evaluation of a specific role, which is consistent with the framework's role-agnostic design.

Where applicable, metrics are normalised by story point estimates assigned to individual work items. Story points serve as the framework's standard normalisation unit, ensuring that observed variation reflects genuine process characteristics rather than legitimate differences in task scope. The framework treats story point estimates as external inputs provided by the team and does not prescribe how they are determined.

The complete mapping of KPIs to pillars is summarised in the following table:

Table 4: The Golden Twelve – KPI mapping to Agile Integrity Pillars

Process Flow	Cycle Time	Throughput	Lead Time
Technical Excellence	Re-work Rate	Defect Density	DoD Compliance
Product Clarity	First-Pass Rate	Backlog Readiness	Scope Stability
Team Synergy	Unblocking Speed	Knowledge Share	Meeting Efficiency

Source: author's own elaboration

The majority of these metrics are derived from tool-traceable data and are available continuously throughout the Sprint. Three exceptions are worth noting: Backlog Readiness is captured as a single snapshot at Sprint Planning, reflecting its function as a diagnostic indicator of preparation quality rather than a measure of in-Sprint performance. Knowledge Share and Meeting Efficiency adopt a dual-source approach, combining tool-traceable indicators with assessment-based data collected during the Sprint, which is described in their respective definitions below.

Cycle Time measures the elapsed duration from the moment active work on an item begins until the item satisfies the Definition of Done. Raw values are normalised by story point estimates, producing a ratio of actual duration to expected complexity. The framework further incorporates variance through the coefficient of variation over normalised values, because a low average can mask significant instability when individual items vary widely in processing duration. The composite formulation combines both dimensions into a single score, as shown in Equation (1):

$$CTscore = \overline{CTnorm} \times (1 + CVnorm) \quad (1)$$

This ensures that both average processing speed and its predictability are captured in one value.

Throughput captures the volume of completed work within a single Sprint, providing a volumetric counterbalance to Cycle Time. The framework allows it to be measured either as a simple count of completed items or as a sum of completed story points, with the choice left to the implementing team.

Lead Time tracks the total elapsed duration from the moment an item enters the Sprint Backlog until its completion. The difference between Lead Time and Cycle Time represents queue time, the period during which an item is committed but not yet actively worked on, which exposes capacity bottlenecks and prioritisation issues that execution-level metrics cannot detect.

Re-work Rate expresses the proportion of completed stories reopened due to technical defects, as shown in Equation (2):

$$Rework\ Rate = \frac{Stories_{reopened}}{Stories_{completed}} \tag{2}$$

The metric is deliberately not normalised by story points, since it captures how frequently the delivery chain fails rather than the magnitude of individual failures.

Defect Density quantifies the number of defects identified per story point of completed work, encompassing defects found both during in-Sprint testing and after item completion. Normalisation by story points ensures that teams working on complex items are not penalised for the natural relationship between task complexity and defect likelihood.

DoD Compliance represents the average percentage of Definition of Done criteria satisfied across all completed stories, as shown in Equation (3):

$$DoD\ Compliance = \frac{1}{n}\sum_{i=1}^{n}\frac{criteria_{met,i}}{criteria_{total}} \tag{3}$$

This graduated approach distinguishes between minor omissions and systematic non-compliance. The framework recognises that an implementing team may opt for a stricter binary interpretation if that better suits their quality culture.

First-Pass Rate measures the proportion of stories accepted as complete on their first submission, without being returned due to misalignment between the delivered result and the original intent of the requirement. This distinguishes it from Re-work Rate: items returned due to technical defects count under Re-work Rate, while items returned because the result does not match what was requested contribute to First-Pass Rate.

Backlog Readiness reflects the proportion of stories selected for the Sprint that satisfy the team's Definition of Ready at the time of Sprint Planning, where the Definition of Ready typically includes conditions such as clearly defined acceptance criteria, resolved dependencies, and completed estimation. A low score signals that the team is entering the iteration with insufficiently prepared inputs, which increases the probability of downstream issues across multiple pillars.

Scope Stability quantifies the degree to which Sprint scope remains unchanged relative to the original commitment, as shown in Equation (4):

$$Scope\ Stability = 1 - \frac{SP_{added} + SP_{removed}}{SP_{committed}} \tag{4}$$

The use of absolute values ensures that scope changes in both directions are captured, because replacing stories with others of equal size still represents significant disruption even though the total volume remains unchanged. The resulting score ranges from 1 (no change) downward, with values approaching 0 indicating near-complete restructuring of the Sprint.

Unblocking Speed tracks, the average elapsed time from the moment a blocker is formally reported until it is resolved, calculated across all reported blockers within the Sprint. It captures the team's collective responsiveness to impediments.

Knowledge Share captures the degree to which expertise circulates among team members rather than remaining concentrated in individuals. The framework adopts a dual-source approach: a tool-traceable component derived from observable activities such as code reviews, documentation contributions, and collaborative sessions provides a continuous approximation, while a peer-assessed component collected through structured anonymous evaluation at the Sprint Retrospective serves as a corrective layer capturing informal knowledge sharing not visible in tooling. The final score is a weighted composite of both components, where the relative weighting between tool-traceable and peer-assessed data is left to the implementing team to determine based on their tooling coverage and team practices.

Meeting Efficiency evaluates whether Scrum events fulfil their prescribed purpose and produce their intended outputs within the allocated timebox. The framework defines this as a composite of a tool-traceable component (timebox adherence) and an assessment-based component (quality of event outputs evaluated after each event). As with Knowledge Share, the relative weighting is left configurable.

9.4 The Dynamic Adaptation Engine

The framework transforms raw KPI data into actionable scores and incentive outputs through a two-component engine: a mathematical model that produces individual and collective rewards from pillar data, and the Spotlight Mechanism through which the Scrum Master can dynamically adjust the framework's focus in response to emerging integrity risks.

9.4.1 Mathematical Model for Incentive Calculation

The mathematical model operates in three layers: normalisation of raw KPI values into comparable scores, aggregation of these scores into pillar-level and project-level integrity

indicators, and translation of integrity data into the framework's two incentive currencies, individual Experience Points (XP) and collective Group Gold.

- **KPI Normalisation:** The twelve KPIs are expressed in heterogeneous units (days, percentages, counts, composite scores) and cannot be aggregated directly. The framework addresses this through threshold normalisation, in which each KPI is mapped onto a common [0, 1] scale using two reference values configured at the beginning of the Epic: an ideal value representing fully healthy performance (score = 1.0) and a critical value representing an unacceptable state (score = 0.0). These reference values are set based on the team's context, historical performance, or industry benchmarks, and they can be recalibrated between Sprints if the team's operating conditions change significantly.

  For metrics where a lower raw value indicates better performance (such as Cycle Time, Lead Time, or Unblocking Speed), the normalised score is calculated, as shown in Equation (5):

$$KPI_{Score} = \frac{critical - ideal}{critical - actual} \tag{5}$$

  For metrics where a higher raw value indicates better performance (such as Throughput, DoD Compliance, or Scope Stability), as shown in Equation (6):

$$KPI_{Score} = \frac{actual - critical}{ideal - critical} \tag{6}$$

  In both cases, the resulting score is clamped to the interval [0, 1], so that values exceeding the ideal threshold produce a score of 1.0 and values worse than the critical threshold produce 0.0. This clamping prevents outlier Sprints from distorting the integrity picture in either direction.

- **Pillar Score and Project Integrity Score**: Once all three KPIs within a pillar have been normalised, the pillar score is computed as their arithmetic mean, as shown in Equation (7):

$$Pillar_{score} = \frac{KPI_1 + KPI_2 + KPI_3}{3} \tag{7}$$

  The default weighting is equal across all three KPIs. The framework permits the implementing team to adjust these weights to reflect the relative importance

of specific metrics within their operating context, provided that the weights sum to 1.

The Project Integrity Score aggregates the four pillar scores into a single indicator of overall project health, as shown in Equation (8):

$$PIS = \frac{Pillar_{PF} + Pillar_{TE} + Pillar_{PC} + Pillar_{TS}}{4} \quad (8)$$

As with individual pillars, the default weighting is equal, and the framework permits reconfiguration. The resulting PIS is a value between 0 and 1, where 1.0 represents full integrity across all four dimensions and values approaching 0 indicate systemic failure.

- **Individual Progression (Experience Points (XP)):** The XP system provides the framework's individual incentive layer, rewarding both direct delivery contributions and the quality with which those contributions are executed. XP is accumulated by each participating team member across the Epic and serves as the input for individual progression through the levelling system defined in the reward economy.

  The total XP earned for completing a work item is calculated as shown in Equation (9):

$$XP_{total} = (SP \times k \times m_{spotlight}) + Mastery_Bonus \quad (9)$$

  Where SP is the story point estimate of the completed item, k is a base coefficient configured by the Scrum Master at the beginning of the Epic and held constant throughout to prevent mid-Epic inflation, and $m_{spotlight}$is the Spotlight multiplier (defaulting to 1.0 when no Spotlight is active, and ranging from 1.25 to 1.5 when the item's activities fall under the currently spotlighted pillar).

  The separation between the base component and Mastery Bonus is deliberate. The base component $(SP \times k \times m_{spotlight})$ rewards effort and volume as shown in Equation (9), while Mastery Bonus rewards the quality of execution. Without this separation, the XP system would incentivise quantity alone, creating pressure to close items rapidly without regard for the integrity standards it is designed to protect.

  Mastery Bonuses are fixed or percentage-based XP supplements awarded when the completed item meets specific quality conditions. They are not affected

by the Spotlight multiplier, ensuring that quality is consistently valued regardless of which pillar is currently prioritised.

Three Mastery Bonus types are defined, each linked to a distinct integrity pillar:

1. **First-Pass Bonus** (Product Clarity) is awarded when a work item is accepted on its first submission without being returned for clarification or revision due to misalignment with the original requirement. This bonus incentivises thorough understanding of the task before work begins, rewarding the investment in upfront communication that directly strengthens the Product Clarity pillar. Any return of the item for requirement-related reasons permanently disqualifies it from receiving this bonus.
2. **Zero-Defect Bonus** (Technical Excellence) is awarded when a completed item passes QA or peer review without any identified defects or violations of technical standards. This bonus reinforces craftsmanship and thoroughness in implementation, motivating team members to verify their own work before submitting it for review rather than relying on downstream testing to catch errors.
3. **DoD Hero Bonus** (Team Synergy) is awarded when a completed item satisfies all criteria in the Definition of Done, including those often neglected under delivery pressure such as documentation, log cleanup, status updates, and knowledge transfer notes. Although DoD Compliance is measured as a KPI within Technical Excellence, the Mastery Bonus for complete DoD fulfilment is deliberately placed under Team Synergy because the Definition of Done functions as a social contract between team members rather than a purely technical standard. Completing documentation and transition materials is an act of collective responsibility: it ensures that colleagues who interact with the work in subsequent Sprints can do so without unnecessary friction. A team member may therefore earn a Zero-Defect Bonus (for clean code) without earning a DoD Hero Bonus (if documentation is incomplete), making visible the distinction between individual technical quality and responsibility toward the team.

All three Mastery Bonuses are available to any role, reinforcing the role-agnostic design principle. A Product Owner who prepares a task so clearly that it passes as

First-Pass earns the same type of recognition as a developer whose code passes without defects.

- **Side Quests:** In addition to base XP from completed work items and Mastery Bonuses, the framework defines Side Quests as a supplementary XP source. Side Quests reward activities that contribute to project integrity but fall outside the standard Sprint Backlog, such as documentation improvements, mentoring, refactoring of legacy components, or process optimisation initiatives. Each Side Quest carries a variable XP value assigned according to its scope and complexity.

  Side Quests serve a specific structural purpose: they create XP-earning opportunities for roles whose primary contributions do not generate story points. A Product Owner who invests time in improving backlog quality or a Scrum Master who develops new facilitation techniques can earn XP through Side Quests that reflect the value of these contributions, ensuring that progression through the levelling system is accessible to all roles rather than disproportionately favouring those who close story-pointed items. Side Quests do not constitute a distinct data source within the KPI layer; their effects on pillar scores are captured only indirectly, to the extent that the activities they incentivise produce observable changes in the standard metrics already defined for each pillar.

  Team members operating in Invisible Mode do not accumulate XP, do not receive Mastery Bonuses, and do not participate in the levelling or archetype systems. Their contributions are reflected exclusively through the collective incentive layer.

- **Collective Reward (Group Gold):** Group Gold provides the framework's collective incentive layer, reinforcing shared accountability for project integrity. Unlike XP, which rewards individual contributions and their quality, Group Gold is generated at the team level and distributed equally among all team members, including those in Invisible Mode.

  The Gold generated in each Sprint is calculated as shown in Equation (10), where PIS is defined in (8):

$$Gold_{sprint} = \max\left(Base_{gold} \times PIS, Base_{gold} \times 0.2\right) \tag{10}$$

  Where $Base_{gold}$ is a fixed value configured at the beginning of the Epic. The floor of 20% ensures that the team receives a minimum reward even in Sprints with

significantly degraded integrity, preventing complete disengagement in difficult periods while preserving the proportional relationship between integrity and reward for the majority of the score range.

The equal distribution of Gold across the entire team, regardless of individual role or participation mode, is grounded in the core principle that all four pillars represent shared responsibilities. When any pillar declines, the Gold reduction affects every team member, creating collective pressure to identify and address the root cause rather than attributing blame to individual roles.

Gold accumulates in a shared Team Treasury across the Epic and is redeemable for non-monetary rewards. The restriction to non-monetary rewards is a deliberate safeguard against the erosion of intrinsic motivation that research on Self-Determination Theory consistently associates with direct financial incentivisation of creative and collaborative work.

9.4.2 The Spotlight Mechanism and Priority Logic

The Spotlight Mechanism is the framework's instrument for dynamic governance, enabling the Scrum Master to redirect the team's attention and incentive structure toward integrity dimensions that require immediate intervention. While the mathematical model operates continuously and automatically, the Spotlight introduces a layer of informed human judgment that adapts the framework to emerging conditions within and between Sprints.

The Spotlight is activated when a pillar score falls below a threshold that signals risk to overall project integrity, or when a pillar exhibits significant instability across consecutive Sprints. The specific activation threshold is configured by the Scrum Master based on the team's context and risk tolerance. The Spotlight can be activated or adjusted at any point during the Sprint, including mid-Sprint, which allows immediate response to acute integrity risks without waiting for a Sprint boundary.

The relationship between individual actions and pillar health operates through two parallel channels. On the diagnostic side, actions generate data that flows into KPIs, which aggregate into pillar scores and ultimately into PIS. On the incentive side, the Spotlight Mechanism requires a mapping of activities to pillars in order to determine which work items qualify for the elevated XP multiplier.

The default activity-to-pillar mapping is defined as follows:

Table 5: Default activity-to-pillar mapping for the Spotlight Mechanism

Process Flow	Completing work items within estimated timeframes, reducing queue time, maintaining consistent delivery pace throughout the Sprint
Technical Excellence	Resolving defects, writing tests, refactoring, conducting thorough code reviews, enforcing technical standards
Product Clarity	Refining backlog items, clarifying acceptance criteria, conducting user research, maintaining stable Sprint scope
Team Synergy	Unblocking colleagues, sharing knowledge through reviews or documentation, facilitating productive Scrum events, completing all DoD criteria for the benefit of downstream collaborators

Source: author's own elaboration

This mapping is configurable by the implementing team to reflect their specific workflow and tooling. The default structure provides a starting point; the team determines which of their concrete activities fall under each category.

When the Spotlight is placed on a pillar, the Spotlight multiplier is applied to the base XP component of all activities mapped to that pillar. The multiplier operates within a suggested default range of 1.25x to 1.5x, providing meaningful incentive to redirect effort without creating excessive distortion of the overall XP economy. Activities mapped to non-spotlighted pillars continue to generate XP at the standard rate ($m_{spotlight} = 1.0$), and Mastery Bonuses remain unaffected by the Spotlight, ensuring that quality incentives are not diluted by the governance mechanism.

The Spotlight does not alter KPI calculations, pillar scores, or the PIS formula (8). It operates exclusively within the XP layer, adjusting the reward weighting for individual contributions without changing the underlying integrity assessment. This separation ensures that the diagnostic accuracy of the pillar system is preserved regardless of which governance interventions are active.

The strategic value of the Spotlight extends beyond simple rebalancing. By making it temporarily more rewarding to invest in a struggling pillar, the mechanism provides a legitimate path for the team to slow down in one area and strengthen another without feeling that they are sacrificing progression. If quality is declining and the Spotlight is placed on Technical Excellence, team members who take the time to write thorough tests or conduct careful reviews are rewarded for doing so rather than penalised by slower XP accumulation compared to those focused purely on throughput.

The Spotlight remains active until the Scrum Master determines that the integrity risk has been sufficiently addressed, typically evidenced by the pillar score returning to or above the activation threshold for at least one full Sprint. The decision to deactivate rests with the Scrum Master, preserving the human judgment element that the framework deliberately incorporates into its governance model.

Only one pillar can be under Spotlight at any given time. This constraint is intentional: attempting to prioritise multiple dimensions simultaneously dilutes the team's focus and undermines the mechanism's motivational effectiveness. If multiple pillars are in decline, the Scrum Master identifies the root cause, which the causal interconnection of pillars is designed to support, and directs the Spotlight toward the pillar where intervention is most likely to produce cascading improvement across the system. Because pillar health is causally linked, addressing the upstream cause of a multi-pillar decline often resolves downstream symptoms without requiring separate intervention.

9.5 Reward Economy and Participation Strategies

Sustained engagement across an entire Epic requires more than accumulating points. XP and Group Gold provide the raw incentive signals, but without meaningful outcomes attached to them, their motivational power fades as novelty wears off. Equally important is the recognition that not every team member responds positively to gamified mechanics, and a system that forces participation risks generating resentment rather than engagement. The reward economy and participation model address both of these challenges.

9.5.1 Individual progression, Shared Treasury and Non-monetary Loot System

XP accumulated across the Epic drives progression through a levelling system that translates cumulative effort and quality into visible milestones. Each level is defined by an XP threshold, and advancement to a new level is permanent within the scope of the Epic. The number of levels and the specific XP thresholds are configured by the implementing team based on the expected duration of the Epic, the team’s size, and the anticipated volume of work, ensuring that progression pacing feels meaningful rather than either trivially fast or unreachably slow.

Levels serve two purposes within the framework. The first is motivational visibility: a level represents a tangible marker of sustained contribution that satisfies the psychological

need for competence by making long-term progress concrete and recognisable. Unlike XP, which increments continuously and can feel abstract, a level transition is a discrete event that provides a sense of achievement and marks a qualitative shift in the individual's standing within the gamified system.

The second purpose is informational. Level data, aggregated and anonymised where appropriate, can serve as an input for organisational decision-making. A team member's level progression across an Epic reflects not only the volume of their contributions but also their consistency and the quality standards they maintained, since Mastery Bonuses accelerate progression relative to pure volume. This information can complement existing performance evaluation processes, provided it is interpreted in context and not reduced to a simplistic ranking metric. The framework does not prescribe how organisations use level data, but it makes this data available as a structured, longitudinal indicator of professional engagement.

Group Gold accumulates in a shared Team Treasury throughout the Epic. The treasury functions as a collective savings mechanism: Gold earned in each Sprint adds to the total, and the accumulated balance becomes redeemable at defined intervals, typically at the end of the Epic or at major milestones within it such as the completion of significant functional blocks. The timing of redemption is configured by the implementing team, though the framework recommends anchoring primary redemption events to the Epic boundary to reinforce the long-term perspective that distinguishes this methodology from Sprint-scoped gamification approaches.

The treasury is redeemable exclusively for non-monetary rewards. This restriction is a fundamental design principle grounded in motivational research, which consistently demonstrates that direct financial incentivisation of creative, knowledge-intensive, and collaborative work tends to undermine the intrinsic motivation that sustains long-term engagement. Non-monetary rewards preserve the symbolic and experiential character of the incentive, maintaining its association with professional growth and team identity rather than with transactional compensation.

Rather than prescribing a fixed reward catalogue, the framework defines principles for constructing one that remains motivationally effective:

1. **Rewards should satisfy at least one of the three SDT needs**. Autonomy-supporting rewards grant the individual or team greater control over their working conditions, such as dedicated time for self-directed learning, flexible

scheduling, or the freedom to select a preferred project or task. Competence-supporting rewards invest in professional growth, such as conference attendance, training budgets, certification programmes, or access to advanced tools and resources. Relatedness-supporting rewards strengthen team bonds, such as team events, shared experiences, or collaborative off-site activities.

2. **Rewards should be non-rivalrous** whenever possible. Because Gold is distributed equally and the treasury is shared, the reward system functions best when the benefits are collective or at least non-competitive. Rewards that create internal competition for limited resources would contradict the cooperative incentive structure that Gold is designed to reinforce.
3. **Rewards should be calibrated to the treasury's expected capacity**. If the reward catalogue contains items that require more Gold than the team can realistically accumulate across an Epic, the system becomes demotivating rather than aspirational. Conversely, if all rewards are easily attainable within the first few Sprints, the treasury loses its function as a long-term motivator. The implementing team should ensure that the catalogue spans a range of values, with some rewards achievable relatively early and others requiring sustained high integrity across most of the Epic.

Individual loot preferences documented in the Character Charter during the diagnostic phase inform how the reward catalogue is constructed and which items are prioritised. This connection between the initial diagnostic and the reward economy ensures that the system remains personalised and relevant throughout the Epic rather than relying on generic incentives that may not resonate with the specific team.

9.5.2 Handling Resistance: The Opt-out and Invisible Mode

Not every team member is motivated by gamified systems, and for some individuals, public progression mechanics such as levels, leaderboards, or archetype identities may feel trivialising, stressful, or simply irrelevant to their professional self-concept. Forcing participation in such cases does not produce engagement; it produces resentment and performative compliance, both of which undermine the collaborative culture that the framework is designed to strengthen. Invisible Mode exists to address this reality.

A team member who activates Invisible Mode opts out of the individual gamification layer entirely. They do not accumulate XP, do not progress through levels, do not adopt a

hero archetype, and do not receive Mastery Bonuses. Their name does not appear on any progression-related visualisations, and they are not subject to any form of individual comparison or ranking within the gamified system.

Crucially, opting out of the individual layer does not exclude the team member from the collective layer. Invisible Mode participants receive their equal share of Group Gold and benefit from the Team Treasury on the same terms as all other team members. This design ensures that choosing not to engage with gamification carries no financial or material penalty, which is essential for the opt-out to be genuinely voluntary rather than formally available but practically punitive.

Invisible Mode does not affect the diagnostic layer of the framework. Work completed by team members in Invisible Mode generates the same KPI data as work completed by active participants: their tasks contribute to Cycle Time, Throughput, Defect Density, and all other metrics in exactly the same way. Pillar scores and PIS reflect the performance of the entire team, including those who have opted out of the gamified interface. This separation between the diagnostic function (which operates universally) and the incentive function (which operates selectively) ensures that project health assessment remains accurate regardless of how many team members choose to participate in the gamified layer.

Invisible Mode can be activated or deactivated at any point during the Epic, typically during the Sprint Retrospective when archetype recalibration also takes place. A team member who initially opts out may choose to join the gamified layer in a later Sprint if their perspective changes, and conversely, an active participant may switch to Invisible Mode if the gamified elements become a source of stress or distraction. XP accumulated before switching to Invisible Mode is preserved but frozen, and resumes accumulating if the team member reactivates their participation. This reversibility removes the pressure of making a permanent decision at the outset of the Epic and allows team members to find their preferred level of engagement organically over time.

9.6 Conceptual Data Model of the Framework

This section formalises the proposed framework into a conceptual entity-relationship model. Its purpose is to make explicit the logical structure of the framework, the dependencies among its core entities, and the information flows required for performance monitoring, incentive allocation, and governance across an epic. The model is organised

across three functional layers, namely operational, diagnostic, and motivational, whose separation reflects the framework's core principle that performance measurement and incentive delivery must remain structurally independent. The complete entity-relationship structure is depicted in Figure 2.

9.6.1 Entity Structure and Attribute Specification

This section formalises the framework's logical structure into a conceptual entity-relationship model. The model specifies the entities, attributes, and relationships required to support performance monitoring, incentive allocation, and governance across an Epic. Its purpose is to demonstrate the internal coherence of the proposed design at the data level, independent of any specific tooling or platform.

Figure 2 presents the conceptual entity-relationship representation of the proposed framework.

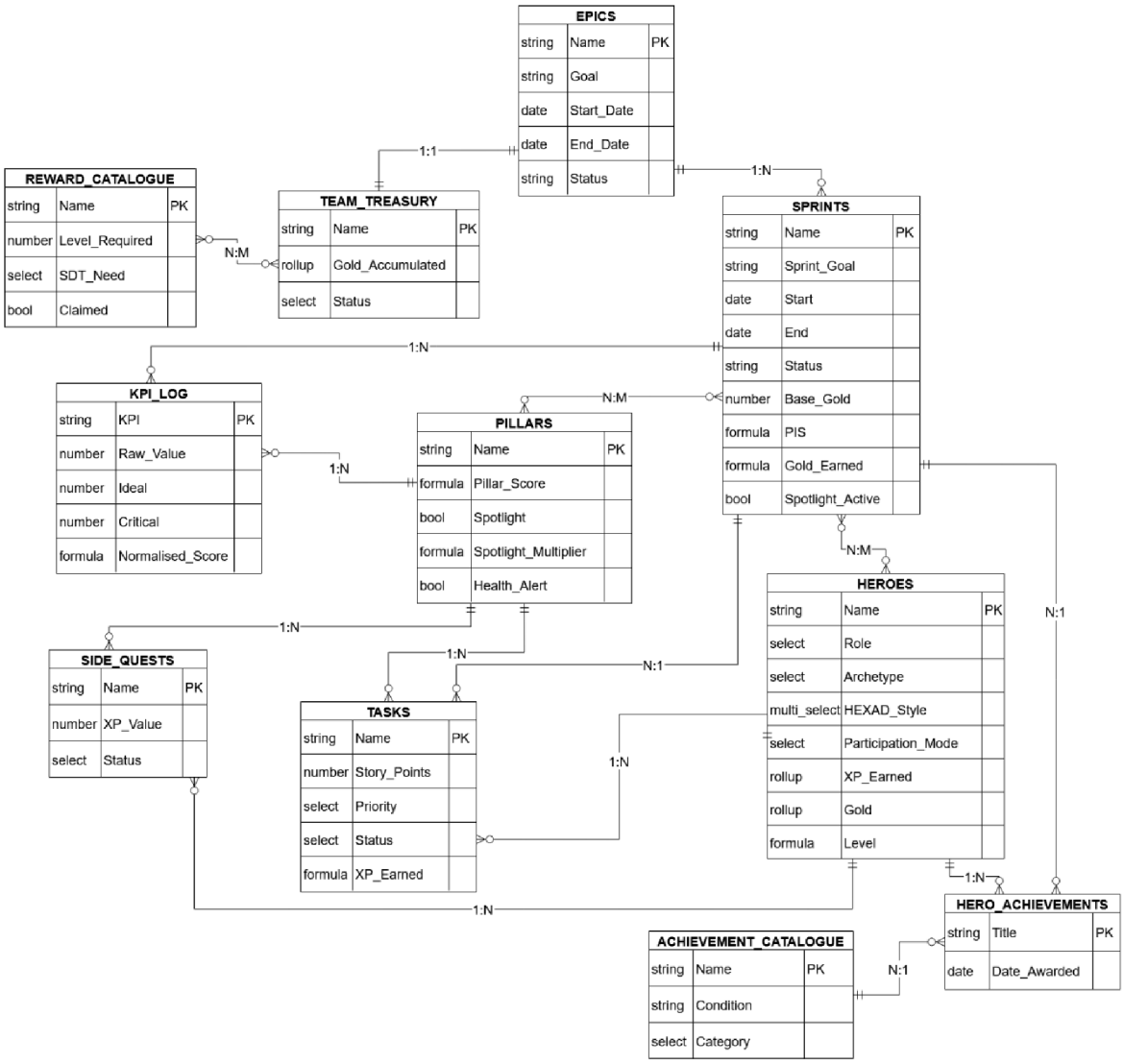

Figure 2: Conceptual entity-relationship diagram of the proposed gamification framework
Source: author's own elaboration

The ERD translates the framework from design logic into a structured conceptual model composed of operational, diagnostic, and motivational layers. Its role is to demonstrate the internal coherence of the proposed framework at the level of entities, attributes, and relationships rather than to represent a software-specific implementation.

Epics is the top-level container of the gamification lifecycle, defining the temporal and strategic scope within which all framework activity is measured. Its derived attributes aggregate collective Gold and task output across all downstream entities.

Sprints is the primary operational unit, capturing the iterative structure of the Scrum process and serving as the central aggregation point for both performance measurement and

reward calculation. Per-pillar integrity scores and the Pillar Integrity Score are computed as per Equation (8), and Gold Earned per Sprint is computed as per Equation (10).

Pillars contains permanent records representing the framework's strategic dimensions. Each pillar aggregates its linked KPI scores into a Pillar Score as per Equation (7) and carries a Spotlight flag and a Health Alert indicator whose threshold is configured by the implementing team at Epic kick-off.

KPI Log holds individual KPI measurement records, each anchored to a specific Sprint and Pillar. Raw values are normalised into a common scale using Equation (5) or Equation (6) depending on the directionality of the metric, enabling direct aggregation into pillar-level scores.

Heroes records individual team member profiles, including their Scrum Role, Archetype, HEXAD Style, and participation mode. Accumulated XP and Gold are derived from linked Tasks, Side Quests, and Sprints respectively. Level is computed from accumulated XP against thresholds configured at Epic kick-off.

Tasks is the atomic unit of individual contribution, linking each work item to its authoring Hero, its Sprint, and its Pillar. XP Earned is computed as per Equation (9).

Side Quests records supplementary contribution activities that fall outside the standard Sprint Backlog. Each record carries a variable XP value assigned according to its scope and complexity, a Pillar reference, and a link to the Hero to whom it is awarded. Side Quests serve as the primary XP-earning mechanism for roles whose standard contributions do not generate story points.

Team Treasury records the collective Gold Balance accumulated across the Epic. Gold Earned in each Sprint flows into the treasury as per Equation (10) and is redeemable for non-monetary rewards at defined milestones configured by the implementing team.

Achievement Catalogue is a static reference entity defining the set of achievements available within the framework. Each record specifies the award condition and its category classification. The catalogue is not modified during the Epic lifecycle.

Hero Achievements is an audit log recording each instance of an achievement being awarded to a Hero, preserving the full history of awards across the Epic with traceability to the Sprint in which each award was made.

Reward Catalogue defines the non-monetary rewards available for redemption from the Team Treasury. Each record carries an SDT Need classification, indicating whether the

reward supports Autonomy, Competence, or Relatedness, and a redemption record linking any claim to the Epic in which it occurred.

9.6.2 Relationship Logic of the Conceptual Model

The relational topology of the data model is organised across three functional layers whose interdependencies are depicted in Figure 2.

The operational layer is anchored by a one-to-many relation between Epics and Sprints, establishing the temporal boundary of the gamification lifecycle. Sprints relate to Heroes through a many-to-many association capturing team composition per iteration. Tasks form a three-way junction entity relating Heroes, Sprints, and Pillars, enabling multi-dimensional aggregation of individual XP within the correct iterative and categorical context. Side Quests relate to Heroes and Pillars through many-to-one associations, extending the XP economy beyond story-pointed delivery items. Team Treasury holds a one-to-one relation with Epics, accumulating Gold across all Sprints within the Epic boundary.

The diagnostic layer is structured around a one-to-many relation between Pillars and KPI Log, propagating Normalised Scores upward into Pillar Score computation as per Equation (7). KPI Log additionally holds a many-to-one relation to Sprints, anchoring each measurement to a specific iteration and enabling longitudinal comparison of KPI trajectories across the Epic lifecycle.

The motivational layer comprises four entities that read from but do not write to the operational and diagnostic layers, preserving the separation between performance measurement and incentive delivery. Hero Achievements relates to Heroes, Achievement Catalogue, and Sprints, recording the full award history. Reward Catalogue relates to Team Treasury and Epics, tracking collective reward redemption across the Epic boundary.

10 Results and Discussion

10.1 Evaluation of the Proposed Framework

The main outcome of this thesis is the design of a gamification framework intended for use in Scrum across an entire epic. Its contribution does not lie in the verification of a deployed intervention, but in the development of a structured conceptual model that responds to several limitations identified in the reviewed literature on agile gamification. Existing approaches have been described as fragmented, methodologically uneven, and frequently centred on narrowly defined, tool-traceable behaviours, which limits their value for long-term use in cross-functional Scrum teams and reduces their ability to address broader dimensions of project health over time (Pedreira, 2015; Marques, 2020; Marques, 2023). In response to these limitations, the proposed framework integrates motivational theory, Scrum operational logic, KPI-based monitoring, and explicit design safeguards into a single model intended to operate across multiple Sprints within a shared epic horizon.

The evaluation of the framework proceeds from two complementary perspectives. The first concerns whether the model provides a meaningful and theoretically grounded response to the shortcomings identified in existing Scrum-oriented gamification approaches, particularly with respect to epic-level continuity, diversity of team contributions, and the treatment of risks such as fairness problems or metric gaming. The second concerns whether the internal structure of the framework forms a coherent and practically viable conceptual basis for use in real Scrum teams. This internal coherence is further strengthened by the conceptual formalisation of the framework through its entity-relationship and information structure, while its actual behavioural effects remain to be verified through future empirical research.

10.1.1 Strengths and Contributions of the Framework

One of the most significant strengths of the proposed framework is its deliberate shift in the temporal logic of gamification, moving from isolated, Sprint-scoped interventions toward the level of the epic as a broader and more meaningful delivery horizon. The reviewed literature consistently indicates that existing agile gamification solutions tend to attach rewards and feedback to selected actions within a limited timeframe, which makes them straightforward to implement but poorly suited to supporting longer-term team

functioning and sustained motivation across multiple iterations (Pedreira, 2015; Marques, 2023). Many of the most operationally significant problems in Scrum, including gradually declining collaboration quality, the accumulation of technical debt, or progressively increasing rework, do not manifest fully within a single Sprint but become clearly visible only as patterns across multiple iterations. By anchoring motivational mechanisms and adaptation logic to the epic horizon, the framework offers a qualitatively different understanding of what gamification in agile environments can accomplish, one in which progression is tied to sustained project integrity rather than to isolated bursts of output.

A further contribution lies in the framework's structural approach to defining project health through four analytically distinct pillars of Agile Integrity: Process Flow, Technical Excellence, Product Clarity, and Team Synergy. Rather than incentivising speed or visible output in isolation, this multi-dimensional structure captures project health through interconnected dimensions that collectively reflect the continuity of delivery, the quality of results, the clarity of work inputs, and the collaborative capacity of the team. In doing so, the framework attempts to remain closer to Scrum's underlying empirical logic of transparency, inspection, and adaptation, treating gamification as a means of directing and sustaining attention toward relevant project conditions. The practical significance of this design is especially apparent in the Spotlight Mechanism, through which the incentive structure can be temporarily redirected toward a weakening pillar, introducing a form of adaptive governance that makes motivational design responsive to changing team conditions.

The framework also makes a meaningful contribution to the problem of role bias in cross-functional Scrum teams. Many current implementations derive recognition primarily from digital traces captured in project management tooling, which systematically privileges story-pointed delivery work while underrepresenting facilitation, backlog clarification, stakeholder coordination, and mentoring (Marques, 2020). The proposed framework responds through a role-agnostic design in which participation is organised around motivational and behavioural orientations expressed through the four hero archetypes rather than around formal job titles. This conceptual separation makes the framework more inclusive and better suited to heterogeneous teams whose members contribute in qualitatively different but equally valuable ways. Inclusiveness is reinforced further through the Character Charter, which translates individual psychological drivers identified via SDT and the HEXAD typology into a personalised engagement profile, and through Side Quests

and collectively distributed Group Gold, which ensure that recognition is not confined to story-pointed delivery alone (Marczewski, 2015; Tondello, 2016).

Although the ethical dimension is not the central contribution of the framework, it constitutes one of its practically important strengths. The reviewed literature identifies competitive dynamics, social comparison pressure, and rule manipulation as recurring mechanisms through which gamified systems can undermine trust and distort the collaborative signals that Scrum teams depend on during inspection and adaptation (Elsalmy, 2025; Almeida, 2023). The proposed model addresses these risks through several structural safeguards: the restriction of collective rewards to non-monetary incentives reflects the consistent finding that direct financial incentivisation of creative and collaborative work tends to undermine intrinsic motivation (Pink, 2009; Deci, 2017); the equal distribution of Group Gold creates collective accountability for integrity failures; and Invisible Mode provides a genuinely voluntary opt-out from the individual gamification layer without material penalty. This last provision is theoretically grounded in the SDT understanding that individuals differ not only in their motivational orientations but also in whether they experience gamified systems as autonomy-supportive or as a form of external pressure (Gagné, 2005; Ryan, 2000). The framework's platform-agnostic architecture further enhances its transferability, since its essential contribution lies in the logical structure of its rules and incentive calculations rather than in any specific toolchain implementation.

10.1.2 Limitations and Risks of the Framework

The most fundamental limitation of the proposed framework is that the model remains conceptual and has not been tested or validated in a real Scrum team operating under industrial conditions. The arguments presented in favour of its design choices are grounded in established motivational theory and in the identified limitations of existing approaches, but theoretical coherence does not constitute empirical validation. It is therefore only defensible to conclude that the framework is theoretically informed, internally structured, and intentionally designed to address several shortcomings identified in the literature. Any stronger claim about its effects on team motivation, engagement, collaboration quality, or delivery performance would require empirical application, systematic data collection, and rigorous evaluation under realistic conditions, which lies beyond the scope of the present thesis and constitutes the most important direction for future research.

A second limitation concerns the framework's dependence on the quality and interpretability of measurement. The four-pillar structure rests on the assumption that complex phenomena such as delivery flow, technical quality, requirement clarity, and collaborative effectiveness can be represented meaningfully through a defined set of quantitative and semi-quantitative indicators. This assumption is not equally sustainable across all twelve KPIs. Metrics such as Cycle Time, Throughput, and Scope Stability can be derived with reasonable reliability from standard tooling. Others, most notably Knowledge Share and Meeting Efficiency, capture dimensions of team behaviour only partially visible in tool-traceable data, and their measurement depends on structured peer assessment that introduces its own sources of bias. There remains a persistent risk that practical implementation would gradually privilege what is easiest to measure, thereby reproducing the very bias toward narrowly traceable outputs that the framework explicitly sets out to overcome.

A third limitation relates to the challenge of ensuring genuine fairness in cross-functional teams. Inclusiveness at the level of design does not automatically translate into perceived fairness under the specific conditions of any given team. Contributions differ not only in their content but also in their visibility, timing, and traceability within tooling, and team members may reasonably perceive the system as inadvertently favouring certain forms of work even where the design intention is the opposite. This perception can undermine the trust and legitimacy that any gamified system requires to function effectively over a long epic horizon. The framework should therefore be understood as a guided model that requires careful contextual calibration, transparent team communication, and continuous reflective review rather than as a self-sufficient mechanism that produces equitable outcomes under all conditions.

A fourth limitation concerns the suitability of gamification for all team compositions. Even a thoughtfully designed system operates on the premise that a meaningful proportion of team members will find the gamified layer engaging, and that their participation will generate the collective dynamics the framework depends on to function as intended. For some individuals, gamified systems are experienced as fundamentally incongruent with their professional identity or preferred mode of collaboration, regardless of design quality (Tondello, 2016; Ryan, 2000). Invisible Mode addresses this partially, but does not resolve whether the collective incentive structure retains sufficient motivational coherence when participation in the individual layer is low or unevenly distributed. The Scrum Master and

team should therefore assess the team's motivational composition using tools such as the HEXAD scale before implementation rather than applying the framework as a universal solution.

The framework is also vulnerable to metric gaming. Any system that makes indicators and visible progression salient creates conditions under which participants may begin to optimise for the metric rather than for the underlying quality or collaborative behaviour it represents. The multi-pillar logic, collective incentive layer, and Spotlight Mechanism collectively reduce the likelihood of simplistic score-chasing, yet they cannot eliminate it entirely, particularly in team environments where trust and psychological safety are already fragile. Across a long epic, gaming patterns that emerge gradually may become normalised if not actively monitored through the governance mechanisms the framework provides.

Finally, the overall complexity of the framework constitutes a genuine practical limitation. The proposed model combines behavioural archetypes, twelve-KPI monitoring, individual and collective incentive calculations, adaptive Spotlight governance, configurable weighting, and voluntary participation modes into a single operational system. This complexity is partly a deliberate consequence of accounting for the multidimensional nature of agile teamwork, but it also makes the framework considerably more demanding to explain, configure, and maintain in practice. Scrum Masters in teams with lower agile maturity or limited capacity for reflective facilitation may find consistent implementation difficult to sustain across an epic of realistic duration. Its practical use should therefore be understood as presupposing a team context in which motivational design is already treated as a legitimate component of reflective agile practice.

10.2 Recommendations and Future Directions

10.2.1 Recommendations for Practical Application

The proposed framework is a structured and theoretically grounded model, but its practical value depends entirely on the conditions under which it is introduced. Based on the limitations identified in the preceding section, the following recommendations are intended to guide teams and organisations considering its adoption.

The most important prerequisite is sufficient Scrum maturity. The framework assumes that the team already operates with functional Sprint cadences, consistent use of a Definition of Done, and an established culture of retrospective reflection. Organisations in early stages

of Scrum adoption should not attempt to layer this gamification model onto an immature process. Instead, a phased introduction is advisable: in the first Epic, only the diagnostic layer should be activated, meaning that KPI data is collected, and pillar scores are calculated but no incentive mechanics are yet introduced. This allows the team to develop familiarity with the measurement framework and to identify which metrics require calibration before rewards become attached to them. Incentive mechanics, including XP, Group Gold, and archetype selection, should be introduced gradually in subsequent Epics once the team has confidence in the reliability of the underlying data.

A second recommendation concerns the role of the Scrum Master as the active governor of the framework. The Spotlight Mechanism and the broader adaptive governance layer depend on the Scrum Master's ability to read pillar trends correctly and intervene at the appropriate moment, a competence that extends beyond standard Scrum facilitation. Organisations should ensure that the Scrum Master receives adequate preparation before the framework is activated, including familiarity with the mathematical model, KPI definitions, and the activation logic of the Spotlight. In teams where the Scrum Master role is filled by someone with limited experience, it may be advisable to involve an agile coach during the first Epic of implementation.

Third, the calibration of reference values for KPI normalisation deserves careful attention. The ideal and critical thresholds must reflect the team's actual operating context rather than abstract industry benchmarks. Where historical data is available from previous Sprints, it should serve as the primary basis for calibration. Where it is not, the team should treat the first one or two Sprints of the Epic as a baseline period during which scores are calculated but collective rewards are not yet distributed, allowing thresholds to be adjusted before the incentive economy becomes active.

Finally, the framework should be introduced through transparent communication about its purpose and limits. Team members should understand from the outset that the system is designed to support collective integrity rather than to evaluate individual performance, that participation in the individual gamification layer is voluntary, and that pillar scores reflect the shared outcome of the delivery system rather than the contribution of any single role. Without this shared understanding, even well-designed gamification mechanics can generate misinterpretation, distrust, or competitive dynamics that contradict the framework's foundational principles.

10.2.2 Recommendations for Future Research

The framework presented in this thesis is a conceptual design whose practical effects remain to be verified, and this limitation defines a clear agenda for future research.

The most immediate priority is empirical validation through a controlled pilot study. The framework should be tested in at least one real Scrum team over the duration of a complete Epic, with quantitative data collected on the KPIs defined in the Golden Twelve and qualitative data gathered through structured interviews and retrospective observations. Primary outcome measures should include team motivation assessed through validated instruments aligned with Self-Determination Theory, as well as self-reported engagement, perceived fairness, and pillar score trajectories across Sprints. A comparison with a control group operating under standard Scrum without gamification would strengthen the validity of any conclusions drawn.

A second direction concerns the automation of the calculation engine. The mathematical model described in this thesis is fully specified but has been implemented only as an illustrative prototype. Future work should develop a production-ready integration with common agile toolchains, particularly Jira and Azure DevOps, in which KPI data is collected automatically, pillar scores are calculated in real time, and the Scrum Master dashboard is populated without manual data entry. Such an implementation would substantially reduce the administrative overhead identified as one of the framework's practical risks.

Beyond validation and tooling, future research should examine the scalability of the framework to larger and more complex delivery environments. The present design targets a single Scrum team operating within a single Epic. In organisations running multiple parallel Scrum teams under a shared programme or product increment, the pillar logic and collective reward structure would require adaptation to account for inter-team dependencies, shared backlogs, and cross-team knowledge flows. Examining how the Adaptive Agile Gamification Framework can be aligned with programme-level agile governance within frameworks such as SAFe or LeSS represents a meaningful contribution to both research and practice.

Finally, longitudinal research examining the sustainability of motivational effects across multiple consecutive Epics would address one of the most persistent open questions in gamification research. The framework is designed to sustain engagement beyond the novelty phase through adaptive mechanics, non-monetary rewards, and voluntary

participation modes, but whether these design choices are sufficient to prevent motivational decline over a period of six months or more remains an empirical question that the present work cannot answer.

11 Conclusion

The primary aim of this thesis was to design a gamification model for the Scrum framework applicable across an entire epic, with the goal of increasing the motivation, engagement, and productivity of agile team members. The work pursued this aim through four interrelated objectives, each of which has been addressed in the course of the preceding chapters.

The first objective, studying the theoretical foundations of gamification and its current use in agile methodologies, was fulfilled through a structured review and analytical synthesis of the relevant literature. The theoretical part of the thesis examined project management fundamentals, the principles and lifecycle of agile project management, and the Scrum framework in detail, before turning to gamification as a discipline grounded in motivational psychology. Key theoretical frameworks were identified and discussed, including Self-Determination Theory, Flow theory, the Expectancy-Value model, and the Octalysis framework, alongside established player typologies such as the Bartle model and the HEXAD user typology. This theoretical foundation established the conditions under which gamification can support sustained motivation in professional team environments and provided the conceptual basis for the subsequent design work.

The second objective, identifying innovative gamification elements suitable for each phase of the Scrum process, was addressed through a critical analysis of existing approaches to gamification in agile environments. The review revealed that current implementations are predominantly narrow in scope, focused on tool-traceable behaviours within individual Sprints, and insufficiently attentive to the motivational heterogeneity of cross-functional teams. These limitations were synthesised into a clearly defined research gap, which justified the need for a new model operating at the level of the epic rather than the isolated iteration.

In the preparation of the theoretical and analytical sections of this thesis, AI tools were used as auxiliary instruments in accordance with the internal regulations of the Czech University of Life Sciences Prague, specifically to support literature search and to review grammar and stylistic clarity. All design work, analytical reasoning, and original contributions presented from this point forward represent exclusively the author's own intellectual output.

The third and central objective, developing a concrete gamification model tailored to the needs of agile teams with an emphasis on epic-level integration and conceptual

operationalisation, represents the primary contribution of this thesis. The proposed Adaptive Agile Gamification Framework integrates motivational theory, Scrum logic, and KPI-based monitoring into a coherent operational model. Its structural core consists of four pillars of Agile Integrity, namely Process Flow, Technical Excellence, Product Clarity, and Team Synergy, which together define project health as a multi-dimensional and shared team responsibility. The framework introduces a dual incentive architecture combining individual Experience Points with collective Group Gold, a behavioural archetype system derived from Self-Determination Theory and supplemented by HEXAD profiling, and a dynamic governance instrument in the form of the Spotlight Mechanism. Ethical design principles, including voluntary participation through Invisible Mode and the exclusive use of non-monetary rewards, are embedded as structural safeguards rather than optional additions. Crucially, the framework is designed for epic-level continuity, which allows motivational mechanisms and integrity monitoring to operate longitudinally across multiple Sprints, a temporal horizon that existing approaches consistently fail to address. This contribution is further strengthened by the conceptual formalisation of the framework through its entity-relationship structure and indicative information logic, which clarify how its core mechanisms can be represented as a consistent system of entities, metrics, and governance rules.

The fourth objective, evaluating the model and proposing further steps, was fulfilled in the results and discussion chapter, where the framework was assessed from the perspectives of both its theoretical coherence and its practical applicability. The evaluation identified the model's principal strengths, including its role-agnostic design, its causal pillar structure, and its adaptive governance capacity, while also acknowledging the limitations that inevitably accompany a conceptual proposal that has not yet been empirically tested. Recommendations for practical application and future research were formulated accordingly.

Taken together, the thesis contributes to the field of agile gamification by proposing a theoretically grounded and methodologically structured model that extends the temporal and motivational scope of existing approaches. It does not claim to resolve the empirical uncertainties that characterise this area of research, but it offers a principled design foundation from which future implementations and evaluations can proceed. The question of whether the Adaptive Agile Gamification Framework produces the intended behavioural

outcomes in real Scrum teams remains open and represents a meaningful direction for subsequent empirical work.

12 References

ALMEIDA, Cláuvin, Marcos KALINOWSKI, Anderson UCHÔA and Bruno FEIJÓ, 2023. *Negative effects of gamification in education software: Systematic mapping and practitioner perceptions*. Information and Software Technology. Elsevier BV, 156, 107142. ISSN 0950-5849. Available from: doi:10.1016/j.infsof.2022.107142

ATLASSIAN, 2024. *Epics, Stories, Themes, and Initiatives* [online]. [cit. 2025-11-05]. Available from: https://www.atlassian.com/agile/project-management/epics-stories-themes

BANDURA, Albert, 1997. *Self-efficacy: The Exercise of Control*. New York: W.H. Freeman. ISBN 978-0-7167-2626-5.

BARTLE, Richard, 1996. *Hearts, clubs, diamonds, spades: Players who suit MUDs*. Journal of MUD Research. 1(1) [online]. [cit. 2025-1-12]. Available from: https://mud.co.uk/richard/hcds.htm

BECK, Kent, Mike BEEDLE, Arie VAN BENNEKUM et al., 2001. *Manifesto for Agile Software Development* [online]. [cit. 2025-11-18]. Available from: https://agilemanifesto.org/

BURKE, Brian, 2014. *Gamify: How Gamification Motivates People to Do Extraordinary Things*. Bibliomotion. ISBN 978-1-937134-86-0.

CARLOS, Antonio and Vinícius ROMEIRO, 2023. *20 Years of the Agile Manifesto: On Agile Project Management*. Management and Production Engineering Review. 37–48. Available from: doi:10.24425/mper.2023.146021

CHOU, Yu-kai, 2016. *Actionable Gamification: Beyond Points, Badges, and Leaderboards*. Packt Publishing. ISBN 978-1-839210-77-8.

CSIKSZENTMIHALYI, Mihaly, 2018. *Flow: The Psychology of Optimal Experience*. ISBN 978-1-717428-51-7.

DECI, Edward L. and Richard M. RYAN, 1985. *Intrinsic Motivation and Self-Determination in Human Behavior*. New York: Plenum Press. Perspectives in Social Psychology. ISBN 978-0-306-42022-1.

DECI, Edward, Anja OLAFSEN and Richard RYAN, 2017. *Self-Determination Theory in Work Organizations: The State of a Science*. Annual Review of Organizational Psychology and Organizational Behavior. 4. Available from: doi:10.1146/annurev-orgpsych-032516-113108

DETERDING, Sebastian et al., 2011. *From Game Design Elements to Gamefulness: Defining Gamification*. Available from: doi:10.1145/2181037.2181040

DOLEŽAL, Jan, 2016. *Projektový management: Komplexně, prakticky a podle světových standardů*. Praha: Grada Publishing. Expert (Grada). ISBN 978-80-247-5620-2.

DONG, Hao, Nicholas DACRE, David BAXTER and Serkan CEYLAN, 2024. *What is Agile Project Management? Developing a New Definition Following a Systematic Literature Review*. Project Management Journal. 55. Available from: doi:10.1177/87569728241254095

EDMONDSON, Amy, 1999. *Psychological Safety and Learning Behavior in Work Teams*. Administrative Science Quarterly. 44, 350–383. Available from: doi:10.2307/2666999

ELSALMY, Fayrouz M., Nada H. SHERIEF, Walid M. ABDEL MOEZ and Hany H. AMMAR, 2025. *Agile Gamification Risk Management Process: A Comprehensive Process for Identifying and Assessing Gamification Risks*. Computers. MDPI, 14(2), 76. ISSN 2073-431X. Available from: doi:10.3390/computers14020076

ELLIOT, Andrew J. and Holly A. McGREGOR, 2001. *A 2 × 2 achievement goal framework*. Journal of Personality and Social Psychology. American Psychological Association, 80(3), 501–519. ISSN 0022-3514. Available from: doi:10.1037/0022-3514.80.3.501

FATIMA, Arsh, Asma SHAHEEN, Sohaib AHMED, Bushra FAZAL, Faizan AHMAD, Tze Wei LIEW and Zeeshan AHMED, 2024. *Exploring the Use of Gamification in Human-Centered Agile-Based Requirements Engineering*. Frontiers in Computer Science. Frontiers Media, 6. ISSN 2624-9898. Available from: doi:10.3389/fcomp.2024.1442081

FORSGREN, Nicole, Jez HUMBLE and Gene KIM, 2018. *Accelerate: The Science of Lean Software and DevOps: Building and Scaling High Performing Technology Organizations*. Portland: IT Revolution Press. ISBN 978-1-942788-33-1.

GAGNÉ, Marylène and Edward L. DECI, 2005. *Self-determination theory and work motivation*. Journal of Organizational Behavior. Wiley, 26(4), 331–362. ISSN 0894-3796. Available from: doi:10.1002/job.322

GAGNÉ, Marylène, Jacques FOREST, Maarten VANSTEENKISTE et al., 2015. *The Multidimensional Work Motivation Scale: Validation Evidence in Seven Languages and Nine Countries*. European Journal of Work and Organizational Psychology. Routledge, 24(2), 178–196. ISSN 1359-432X. Available from: doi:10.1080/1359432X.2013.877892

GOODHART, Charles A. E., 1981. *Problems of monetary management: The U.K. experience*. In: COURAKIS, Anthony S., ed. Inflation, Depression, and Economic Policy in the West. Totowa: Barnes and Noble Books, pp. 111–146.

HAMARI, Juho, 2017. *Do badges increase user activity? A field experiment on the effects of gamification*. Computers in Human Behavior. 71, 469–478. ISSN 0747-5632. Available from: doi:10.1016/j.chb.2015.03.036

HAMARI, Juho, Jonna KOIVISTO and Harri SARSA, 2014. *Does Gamification Work? — A Literature Review of Empirical Studies on Gamification*. Available from: doi:10.1109/HICSS.2014.377

HOWARD, Joshua, Marylène GAGNÉ, Alexandre J. S. MORIN and Anja VAN DEN BROECK, 2016. *Motivation profiles at work: A self-determination theory approach*. Journal of Vocational Behavior. Elsevier BV, 95–96, 74–89. ISSN 0001-8791. Available from: doi:10.1016/j.jvb.2016.07.004

HUOTARI, Kai and Juho HAMARI, 2012. *Defining gamification: A service marketing perspective*. In: Proceeding of the 16th International Academic MindTrek Conference. New York: ACM, pp. 17–22. Available from: doi:10.1145/2393132.2393137

INTERNATIONAL PROJECT MANAGEMENT ASSOCIATION, 2015. *Individual Competence Baseline: For Project, Programme & Portfolio Management*. ISBN 978-94-92338-00-6.

JOHNSON, Daniel, Sebastian DETERDING, Kerri-Ann KUHN, Aleksandra STANEVA, Stoyan STOYANOV and Leanne HIDES, 2016. *Gamification for health and wellbeing: A systematic review of the literature*. Internet Interventions. 6, 89–106. ISSN 2214-7829. Available from: doi:10.1016/j.invent.2016.10.002

JUDGE, Timothy A. and Remus ILIES, 2002. *Relationship of personality to performance motivation: A meta-analytic review*. Journal of Applied Psychology. American Psychological Association, 87(4), 797–807. ISSN 1939-1854. Available from: doi:10.1037/0021-9010.87.4.797

KOIVISTO, Jonna and Juho HAMARI, 2019. *The rise of motivational information systems: A review of gamification research*. International Journal of Information Management. 45, 191–210. ISSN 0268-4012. Available from: doi:10.1016/j.ijinfomgt.2018.10.013

LANDERS, Richard N., 2015. *Developing a theory of gamified learning: Linking serious games and gamification of learning*. Simulation & Gaming. Available from: doi:10.1177/1046878114563660

LANDERS, Richard N., Kristina N. BAUER and Rachel C. CALLAN, 2017. *Gamification of task performance with leaderboards: A goal setting experiment*. Computers in Human Behavior. 71, 508–515. ISSN 0747-5632. Available from: doi:10.1016/j.chb.2015.08.008

LOCKE, Edwin A. and Gary P. LATHAM, 2002. *Building a practically useful theory of goal setting and task motivation: A 35-year odyssey*. American Psychologist. American Psychological Association, 57(9), 705–717. ISSN 1935-990X. Available from: doi:10.1037/0003-066x.57.9.705

MARCZEWSKI, Andrzej Christopher, 2015. *Even Ninja Monkeys Like to Play*. CreateSpace Independent Publishing Platform. ISBN 978-1-5147-4566-3.

MARQUES, Rita, Gonçalo COSTA, Miguel MIRA DA SILVA, Daniel GONÇALVES and Pedro GONÇALVES, 2020. *A gamification solution for improving Scrum adoption*. Empirical Software Engineering. Springer Science and Business Media, 25(4), 2583–2629. ISSN 1382-3256. Available from: doi:10.1007/s10664-020-09816-9

MARQUES, Rita, Miguel MIRA DA SILVA and Daniel GONÇALVES, 2023. *Gamification for agile: a systematic literature review*. International Journal of Agile Systems and Management. Inderscience Publishers, 16(2), 226. ISSN 1741-9174. Available from: doi:10.1504/ijasm.2023.130838

MYSLÍN, Josef, 2016. *Scrum: Průvodce agilním vývojem softwaru*. Brno: Computer Press.

PEDREIRA, Oscar, Félix GARCÍA, Nieves BRISABOA and Mario PIATTINI, 2015. *Gamification in software engineering – A systematic mapping*. Information and Software Technology. Elsevier BV, 57, 157–168. ISSN 0950-5849. Available from: doi:10.1016/j.infsof.2014.08.007

PEOPLECERT, 2023. *PRINCE2 7 Managing Successful Projects*. 7th ed. ISBN 978-9-925344-60-4.

PINK, Daniel H., 2009. *Drive: The Surprising Truth About What Motivates Us*. New York: Riverhead Books. ISBN 978-1-594488-84-9.

PROJECT MANAGEMENT INSTITUTE, 2021. *A Guide to the Project Management Body of Knowledge (PMBOK Guide) – Seventh Edition and The Standard for Project Management*. Project Management Institute. ISBN 978-1-62825-664-2.

PRZYBYŁEK, Adam, Marta ALBECKA, Olga SPRINGER and Wojciech KOWALSKI, 2021. *Game-based Sprint retrospectives: multiple action research*. Empirical Software Engineering. Springer Science and Business Media, 27(1). ISSN 1382-3256. Available from: doi:10.1007/s10664-021-10043-z

REINERTSEN, Donald G., 2009. *The Principles of Product Development Flow: Second Generation Lean Product Development*. Redondo Beach: Celeritas Publishing. ISBN 978-1-935401-00-1.

ROSENAU, Milton D., 2007. *Řízení projektů*. 3rd ed. Praha: Computer Press. Business Books. ISBN 978-80-251-1506-0.

RYAN, Richard M. and Edward L. DECI, 2000. *Self-determination theory and the facilitation of intrinsic motivation, social development, and well-being*. American Psychologist. 55, 68–78. Available from: doi:10.1037/0003-066X.55.1.68

RYAN, Richard M. and Edward L. DECI, 2017. *Self-Determination Theory: Basic Psychological Needs in Motivation, Development, and Wellness*. New York: Guilford Publications. ISBN 978-1-4625-2876-9.

SAILER, Michael and Lisa HOMNER, 2020. *The gamification of learning: A meta-analysis*. Educational Psychology Review. 32, 77–112. Available from: doi:10.1007/s10648-019-09498-w

SAILER, Michael, Jan Ulrich HENSE, Sarah Katharina MAYR and Heinz MANDL, 2017. *How gamification motivates: An experimental study of the effects of specific game design elements on psychological need satisfaction*. Computers in Human Behavior. 69, 371–380. ISSN 0747-5632. Available from: doi:10.1016/j.chb.2016.12.033

SCHWABER, Ken and Jeff SUTHERLAND, 2020. *The Scrum Guide: The Definitive Guide to Scrum: The Rules of the Game* [online]. [cit. 2025-10-17]. Available from: https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-US.pdf

SCRUMSTUDY, 2022. *A Guide to the Scrum Body of Knowledge (SBOK Guide)*. 4th ed. [online]. [cit. 2025-10-17]. Phoenix, Arizona: SCRUMstudy, a brand of VMEdu. ISBN 978-0-989925-20-4. Available from: https://www.scrumstudy.com/

STOL, Klaas-Jan, Mario SCHAARSCHMIDT and Shelly GOLDBLIT, 2021. *Gamification in software engineering: The mediating role of developer engagement and job satisfaction*. Empirical Software Engineering. Springer Science and Business Media, 27(2). ISSN 1382-3256. Available from: doi:10.1007/s10664-021-10062-w

SVOZILOVÁ, Alena, 2011. *Projektový management*. 2nd updated ed. Praha: Grada. Expert (Grada). ISBN 978-80-247-3611-2.

ŠOCHOVÁ, Zuzana and Eduard KUNCE, 2024. *Agilní metody řízení projektů*. 3rd updated ed. Albatros Media. ISBN 978-80-251-5108-2.

TAKEUCHI, Hirotaka and Ikujiro NONAKA, 1986. *The new new product development game*. Harvard Business Review. ISSN 0017-8012. Available from: https://hbr.org/1986/01/the-new-new-product-development-game

TONDELLO, Gustavo F., Rina R. WEHBE, Lisa DIAMOND, Marc BUSCH, Andrzej MARCZEWSKI and Lennart E. NACKE, 2016. *The Gamification User Types Hexad Scale*. In: Proceedings of the 2016 Annual Symposium on Computer-Human Interaction in Play. New York: ACM, pp. 229–243. Available from: doi:10.1145/2967934.2968082

TONDELLO, Gustavo F., Alberto MORA, Andrzej MARCZEWSKI and Lennart E. NACKE, 2019. *Empirical validation of the Gamification User Types Hexad scale in English and Spanish*. International Journal of Human-Computer Studies. Elsevier BV, 127, 95–111. ISSN 1071-5819. Available from: doi:10.1016/j.ijhcs.2018.10.002

TURLEY, Frank and Nader RAD, 2018. *Agile Scrum Handbook*. Van Haren. ISBN 978-94-018-0278-9.

VAN DEN BROECK, Anja, Maarten VANSTEENKISTE and Hans DE WITTE, 2008. *Self-determination theory: A theoretical and empirical overview in occupational health psychology*. In: Occupational Health Psychology: European Perspectives on Research, Education, and Practice. Vol. 3, pp. 63–88.

VANDEWALLE, Don, 1997. *Development and validation of a work domain goal orientation instrument*. Educational and Psychological Measurement. SAGE Publications, 57(6), 995–1015. ISSN 0013-1644. Available from: doi:10.1177/0013164497057006009

VERHEYEN, Gunther, 2019. *Scrum – A Pocket Guide: A Smart Travel Companion*. Van Haren Publishing. ISBN 978-94-018-0375-5.

WERBACH, Kevin and Dan HUNTER, 2020. *For the Win, Revised and Updated Edition: The Power of Gamification and Game Thinking in Business, Education, Government, and Social Impact*. Philadelphia: University of Pennsylvania Press. ISBN 978-1-61363-104-1.

YILMAZ, Murat and Rory O'CONNOR, 2016. *A Scrumban integrated gamification approach to guide software process improvement: A Turkish case study*. Tehnički Vjesnik. 23, 237–244. Available from: doi:10.17559/TV-20140922220409

ZICHERMANN, Gabe and Joselin LINDER, 2013. *The Gamification Revolution: How Leaders Leverage Game Mechanics to Crush the Competition*. New York: McGraw Hill Professional. ISBN 978-0-07-180832-3.

13 List of pictures and tables

List of pictures

Figure 1: SDT-Based Archetype Selection Model 56

Figure 2: Conceptual entity-relationship diagram of the proposed gamification framework 77

List of tables

Table 1: Benefits and limitations of Agile Project Management 24

Table 2: Advantages and limitations of Scrum 32

Table 3: Proposed behavioural archetypes for gamification in Scrum teams 58

Table 4: The Golden Twelve – KPI mapping to Agile Integrity Pillars 63

Table 5: Default activity-to-pillar mapping for the Spotlight Mechanism 71