مجلة جامعة دمشق للعلوم الهندسية
المجلد41 العدد(2) – 2025: 307-348

Damascus University Journal for engineering sciences
Vo 41 number (2) -2025: 307-348

التحقق النظري من مقاييس إنتاجية الفريق وتقدم المشروع في سياق المنهجيات الرشيقة لتطوير البرمجيات

محمد سعيد أبو طراب*[1]

*[1]. عضو هيئة تدريسية في قسم هندسة البرمجيات ونظم المعلومات– كلية الهندسة المعلوماتية – جامعة دمشق(m.s.aboutrab@damascusuniversity.edu.sy)

الملخص:

يعتبر دور المقاييس البرمجية في مختلف إجرائيات التطوير البرمجي ولا سيما في المنهجيات الرشيقة مهماً في عمليات الإدارة والملاحقة والجودة على مستوى المشروع والإجرائية والمنتج

واتخاذ القرارات ذات الصلة في الوقت المناسب وخاصةً إذا كانت الواصفة المستهدفة في القياس هي إنتاجية الفريق Team Productivity وتقدم المشروع Project Progress. أجريت العديد من الدراسات لتطوير مقاييس لقياس التقدم والإنتاجية في الإجرائيات الرشيقة Agile Development، ولكن هذه المقاييس على قلتها نسبياً عانت من عدم وجود تحقق منهجي من عملها كما عانت من صعوبة استخدامها نتيجة احتياجاتها إلى تحليلات رياضية معقدة مما أبعدها عن الاستخدام العملي بسبب أن الاستخدام غير المضبوط لهذه المقاييس لن يحقق الفائدة المرجوة، بل على العكس سيزيد من خطورة فشل المشروع لاعتماده على نتائج ومؤشرات قياس غير دقيقة.

يهدف هذا البحث إلى تقييم عدد من مقاييس مقترحة للإنتاجية والتقدم وهي مقياس إنتاجية الفريق ضمن تكرار Productivity of the Team Within an Iteration (PTI) ومقياس زيادة التقدم ضمن تكرار Increase in Progress Within an Iteration (IPWI) ومقياس التقدم الفعلي ضمن تكرار Actual Progress in an Iteration (API)، وذلك وفق منهجية تحقق نظرية للتأكد من فعالية ودقة استخدامها والاستفادة منها في المشاريع البرمجية التي تتبع المنهجيات الرشيقة في تطوير البرمجيات.

بينت نتيجة التحقق إمكانية نسبية لاستخدام كل من مقاييس (PTI, API) بعكس مقياس (IPWI) والذي يحتاج إلى إعادة نظر في بنيته حيث لا يمكن الوثوق فعلياً في استخدامه أو الاعتماد على النتائج التي يمكن الوصول إليها إلا في إطار ضيق ومحدود حسب المزية المستهدفة من عملية القياس.

تاريخ الايداع: 2023/1/22
تاريخ القبول: 2023/5/29

حقوق النشر: جامعة دمشق –سورية، يحتفظ المؤلفون بحقوق النشر بموجب CC BY-NC-SA

http://journal.damascusuniversity.edu.sy/index.pHp/index/index
ISSN (online): 2789-6854

الكلمات المفتاحية: المقاييس البرمجية، المنهجيات الرشيقة في تطوير البرمجيات، تقييم المقاييس البرمجية.

A Theoretical Validation of Team Productivity and Project Progress Metrics with the context of Agile Software Development

Mohammad Saeed Abou Trab*[1]

*[1]. Lecturer -Department of Software Engineering and Information Systems – Faculty of Information Technology Engineering – Damascus University. (Email: m.s.aboutrab@damascusuniversity.edu.sy)

Received: 22/1/2023
Accepted: 29/5/2023

Copyright: Damascus University- Syria, The authors retain the copyright under a CC BY- NC-SA

Abstract:

Software metrics has an increasing role within various software development process, especially in agile development context. This role comes from its contribution to measuring, tracking, and auditing quality at the project, process, and product levels. In addition, they support practitioners in making decisions in a timely manner especially if the attribute being measured related to team productivity and project progress.

Several studies have been conducted to develop metrics to measure team productivity and project progress in Agile Development. However, these metrics suffered from the lack of systematic validation, the difficulty of using them since they rely on complex mathematical analyses. As a result, their direct use will increase the risk of project failure due to its reliance on untrusted measurement results and inaccurate interpretation of their indicators.

This research aims at validating several proposed team productivity and progress metrics; Productivity of the Team Within an Iteration (PTI), Increase in Progress Within an Iteration (IPWI), Actual Progress in an Iteration (API), according to a theoretical validation methodology to ensure their use in agile software development approaches.

The validation results showed the relative possibility of using each of the PTI and API metrics, unlike the another metric (IPWI), which need to be reconsidered.

Keywords: Software Metrics, Agile Development, Metrics Validation

المقدمة: Introduction

يتبع تطوير التطبيقات البرمجية إجرائيات واضحة يتم من خلالها الوصول إلى منتجات برمجية تحقق متطلباتها الوظيفية وغير الوظيفية ضمن حدود الزمن والتكلفة المتفق عليها. تعتبر المنهجيات الرشيقة في تطوير التطبيقات البرمجية من المنهجيات التي تحاول تحقيق هذه الأهداف عن طريق مجموعة من المبادئ والقيم التي تعزز مبادئ التطوير التكراري والاستجابة السريعة للتغيير وتعزيز دور الفريق في الإنتاجية. وهذا ما يجب أن تضمنه الإدارة الناجحة للمشروع البرمجي من خلال إجراء القياسات المختلفة للحصول على المعلومات التي تساهم في تقدير التكاليف والزمن اللازم لتنفيذ المشروع وضبط إنتاجية الفريق والتحقق من أن التطور في المشروع يسير ضمن الخطة الموضوعة [1].

سلطت العديد من الأوراق البحثية الضوء على أهمية استخدام المقاييس البرمجية في المنهجيات الرشيقة لتطوير البرمجيات حيث تم تطوير العديد من المقاييس الموجهة لقياس واصفات المشروع أو الإجرائية أو المنتج البرمجي مثل مقاييس التقدم المشروع progress metrics، أو مقاييس الإنتاجية productivity metrics، أو مقاييس التكلفة cost metrics وغيرها من المقاييس التي تساعد الإدارة على تتبع المشروع واتخاذ القرارات ذات الصلة في الوقت المناسب والتي تسهم في نجاحه. من ناحية أخرى، يؤدي الاستخدام المبكر للمقاييس البرمجية على اختلاف أنواعها قبل التحقق من صحتها نظريًا أو تجريبيًا (أو كليهما) إلى زيادة خطورة فشل المشروع بسبب الاعتماد على معلومات قياس غير دقيقة مما سيؤثر سلباً على أي قرار متخذ [2].

أجريت العديد من الدراسات لتطوير مقاييس لقياس التقدم progress والإنتاجية productivity في الإجرائيات الرشيقة Agile Development، ولكن عانت هذه المقاييس على قلتها نسبياً من عدم وجود تحقق منهجي من عملها أو صعوبة استخدامها نتيجة احتياجاتها إلى تحليلات رياضية معقدة مما أبعدها عن الاستخدام العملي. يهدف هذا البحث إلى تقييم عدة مقاييس مقترحة للإنتاجية [3] وفق منهجية تحقق نظرية معتمدة مما يساعد على تحديد فعالية ودقة استخدامها والاستفادة منها في المشاريع البرمجية التي تتبع المنهجيات الرشيقة في تطوير البرمجيات.

تم تقسيم هذه الورقة البحثية وفق ما يلي: يقدم القسم الثاني التعاريف الأساسية لبعض المفاهيم الأساسية، يليه لمحة عن أهم تصنيفات المقاييس البرمجية في القسم الثالث، ويناقش القسم الرابع الدراسة المرجعية، بينما يعرف القسم الخامس المقاييس البرمجية المستهدفة من هذا البحث، كما تم استعراض طرق التحقق من المقاييس البرمجية في القسم السادس، يليه تقديم منهجية التحقق المتبعة في هذه الورقة في القسم السابع، ويقدم القسم الثامن دراسة التحقق للمقاييس البرمجية المستهدفة في هذا البحث ويناقش النتائج التي تم التوصل إليها ليتم استعراض الملخص والتحديات والآفاق المستقبلية في القسم التاسع.

1. تعاريف أساسية Definitions:

الواصفة Attribute: خاصية قابلة للقياس لمكون. ويمكن أن تكون الواصفة داخلية متعلقة بالمنتج نفسه مثل التعقيد Complexity أو خارجية متعلقة بسلوك النظام الخارجي مثل موثوقية النظام Reliability [4].

المكون Component: أحد الأجزاء التي يتألف منها نظام برمجي، ويمكن تقسيمها إلى مكونات أخرى [4].

المقياس البرمجي Software Metric: وهو الذي يقوم بعملية قياس كمية بطريقة محددة للحصول على قيمة تعبر عن الواصفة التي يقيسها ضمن تطبيق أو إجرائية أو مشروع برمجي [4].

عامل الجودة Quality factor: ويعبر عن مجموعة من الميزات المرغوب تواجدها في المكون البرمجي [5].

التحقق Validation: هو التأكد من أن المكون البرمجي المراد التحقق منه يحقق مجموعة من عوامل الجودة [4].

2. تصنيفات المقاييس البرمجية Software Metrics Classification

يتم عادة تصنيف المقاييس البرمجية software metrics حسب طريقة عملها إلى مقاييس مباشرة Direct metrics وأخرى غير مباشرة Indirect metrics. من ناحية أخرى يمكن تصنيفها حسب الهدف الذي تقيسه إلى مقاييس المؤسسة organizational metrics ومقاييس المشروع project metrics ومقاييس الإجرائية process metrics ومقاييس المنتج product metrics [6].

المقاييس المباشرة: وهي المقاييس الأساسية التي يمكن الحصول على البيانات منها بشكل مباشر مثل قياس الوقت والجهد والأخطاء وغيرها.

المقاييس غير المباشرة: وهي المقاييس المشتقة من المقاييس المباشرة عن طريق علاقة محددة الأبعاد مثل قياس الإنتاجية والجودة وغيرها.

مقاييس المؤسسة: تفيد هذه المقاييس في التخطيط الشامل للمشاريع التي تقوم بها المؤسسة أو الشركة وتساعد على تحديد أهداف جديدة على مستوى الأعمال وقياس مدى تحققها.

مقاييس المشروع: وهي المقاييس التي تستهدف قياس واصفات المشروع التي تكون مفيدة في مراقبته والتحكم فيه مثل قياس الجهد الفعلي لاختبار التطبيق المراد تطويره. من خلال إجراء هذا القياس بشكل مستمر يستطيع مدير المشروع معرفة نسبة الجهد المقدر على الجهد الفعلي المستهلك مما سوف يساعده على تقدير الجهد المتبقي وتخصيص الموارد اللازمة لتحقيق الهدف.

مقاييس الإجرائية: تعتبر الإجرائيات المتبعة في تطوير المنتجات البرمجية العامل الحاسم في الوصول إلى هدف المشروع في تسليم المنتج البرمجي في الوقت المحدد والجودة المحددة. تهدف مقاييس الإجرائية إلى ضمان نضجها وجودتها وتحسين فعالية أدائها باستمرار من خلال معرفة المهام المراد إنجازها وتحديد المسؤول عنها ومتى.

مقاييس المنتج: تتعلق مقاييس المنتج بمنتج معين يصف خصائصه مثل الحجم والتعقيد وميزات التصميم والأداء ومستوى الجودة. يتم تصنيف مقاييس المنتج إلى مقاييس ثابتة ومقاييس ديناميكية. على سبيل المثال، (متوسط عدد المختبرين العاملين في مشروع) هو مقياس ثابت للمشروع. يمكن التعامل مع (عدد العيوب المتبقية ليتم إصلاحها) على أنها مقياس ديناميكي حيث لا يمكن حسابها بدقة إلا بعد إجراء تغيير الرمز وإعادة اختبار المنتج.

3.الدراسة المرجعية Related Work

مازالت الأوراق البحثية تسلط الضوء على أهمية استخدام المقاييس البرمجية في المنهجيات الرشيقة لتطوير البرمجيات، حيث أجرت إحدى الدراسات ورشات عمل في أربع شركات تعتمد المنهجية الرشيقة لبيان ماهي المقاييس التي يمكن أن تستخدم من قبل قادة الفرق لقياس إجرائية التطوير المستخدمة. استخدمت عينة محددة من مدراء الفرق لشرح آلية الدراسة وعقد المقابلات وتسجيل بيانات المقاييس المختارة. بينت الدراسة اهتمام الشركات بمقاييس مثل سرعة الفريق team velocity واختبار الأداء performance test ودقة التقدير estimation accuracy. أظهرت الدراسة أن أهم التحديات الأساسية التي تعيق استخدام هذه المقاييس هي عدم توفر بينات مدخلات هذه المقاييس، وعدم ضمان فعاليتها ودقة نتائجها، وعدم إمكانية تفسير مخرجات هذه

المقاييس بشكل دائم لاتخاذ القرارات المتعلقة بتنمية هذه الشركات [2].

يعتبر تقدير الجهد المبذول effort estimation من أحد أكثر الجوانب صعوبة في التطوير الرشيق. لذلك تم تطوير العديد من المقاييس التي تهدف إلى ذلك مثل مقياس تقدير الجهد الذي يعتمد في حسابه على العديد من العوامل مثل عدد وحجم وتعقيد قصص المستخدم في التكرار iteration، وعدد أيام العمل في الشهر، وسرعة التكرار iteration velocity (عدد القصص المنجزة في التكرار)، وراتب الفريق الشهري، ومستوى الثقة في التقديرات. تم التحقق من هذا المقياس عن طريق القيام بدراسة تجريبية على بيانات سابقة لإحدى الشركات وبينت أن نتيجة تقدير الجهد يمكن أن يستخدم حسب معياري متوسط حجم الخطأ النسبي Mean Magnitude of Relative Error (MMRE) والتنبؤ بالخطأ Prediction of Error (PRED) [7].

يعد قياس الإنتاجية productivity measurement أمرًا ضروريًا لتحسين أداء فرق التطوير في المشاريع البرمجية وخصوصاً عند استخدام المنهجيات الرشيقة لتحقيق نتائجاً أفضل في تقدير وقت وتكلفة إنجاز هذه المشاريع. تم تطوير العديد من نماذج المقاييس لقياس الإنتاجية، ومع ذلك لا يمكن الادعاء أنه تم الوصول إلى النموذج النهائي لها سواء في الاعتماد على تعريف متفق عليه للإنتاجية أو معايرة هذه النماذج للحصول منها على نتائج موثوقة يمكن الاعتماد عليها في اتخاذ القرارات الصحيحة لتنفيذ هذه المشاريع. اقترحت إحدى الدراسات نموذجاً إحصائياً للمشاريع التي تتبع المنهجية الرشيقة لقياس إنتاجية الفريق اعتماداً على ستة عوامل داخلية للفريق (رؤية الفريق، قائد الفريق، وعلاقة أعضاء الفريق فيما بينهم، والمتطلبات التي ينجزها الفريق، وجودة مطابقة المتطلبات، وسرعة إنجاز الفريق). تم دراسة فعالية هذا النموذج عن طريق إجراء دراسة إحصائية شملت أربعين شركة برمجية حيث أشارت النتائج أن 80% من الإنتاجية تتأثر بعوامل الفريق وتم التحقق من النتائج عن طريق استخدام معامل التحديد المصحح، والخطأ المعياري في التقدير بين عوامل فعالية الفريق والإنتاجية حيث بينت النتائج قيم معقولة ومقبولة لهذين العاملين [8]. ومع ذلك، لم تبين الدراسة طريقة قياس كل عامل ووحدات قياسه وكيف يمكن تفسير نتائجهم، ومدى ملائمة قيم هذه العوامل مع بعضها البعض، وكيف يمكن أن تؤثر قيم الإنتاجية على تقدير التكلفة والجهد الخاصين بالمشروع.

بطريقة مشابهة، تم دراسة تأثير العوامل المتعلقة بالفرق الرشيقة على إنتاجيتها من خلال استبيان أنجز على 52 شركة برمجية في الباكستان حيث تم إيجاد درجة من الترابط الإحصائي بين هذه العوامل (عضو الفريق، قائد الفريق، علاقة أعضاء الفريق فيما بينهم، المتطلبات التي ينجزها الفريق، جودة مطابقة المتطلبات، سرعة إنجاز الفريق، رؤية الفريق) بشكل عام ولو بصورة متباينة وبين إنتاجية الفرق [9]. على الرغم من محاولة الدراسة ضبط علاقة كل عامل مع الإنتاجية بشكل إحصائي ولكن لم تبين كيفية الحساب وتأثير النتائج. بمعنى آخر بينت الدراسة أهمية هذه العوامل دون أن تبين الطريقة التي سوف تحسب من خلالها أو يمكن أن تفسر نتائجها بطريقة تساهم في اتخاذ القرار الصحيح، كما اقتصر التحقق من تأثير هذه العوامل على دراسة الترابط دون الخوض في التحقق من المقاييس وطريقة القياس مما أفقد الدراسة بعدها التطبيقي.

بمنهجية مشابهة، صنفت دراسة أخرى عن طريق مراجعة منهجية لـ 57 بحثاً متعلقاً بهذا المجال العوامل التي تؤثر على إنتاجية الفرق الرشيقة إلى عوامل متعلقة بالمنتج مثل تعقيد المنتج ولغة البرمجة المستخدمة ومدة المشروع، وعوامل متعلقة بالفريق مثل خبرة الفريق ودرجة تحفيزهم ودرجة تكاملهم ومعدل مغادرتهم الفريق، وعوامل متعلقة بالمؤسسة مثل بيئة العمل وإدارة المعرفة وبرامج التدريب، وعوامل متعلقة بطبيعة المشاريع مفتوحة المصدر مثل طبيعة التعاقد ودرجة الاستثمار. تم دراسة مدى فعالية هذه العوامل عن طريق استبيان الكتروني أجيب عليه من قبل 47 مطور

ليبين ضرورة تحسين الإنتاجية من خلال النظر في مزيج من العوامل البشرية والمؤسساتية [10]. تعاني هذه الدراسة من عدة إشكاليات مثل عدم ضبط عملية اختيار الدراسة المرجعية وضمان شموليتها في اختيار العوامل أو من خلال عدم شمولية الاستبانة من ناحية العينة التي أجابت على أسئلته. من ناحية أخرى لم تبين الدراسة طرق حساب هذه العوامل وكيفية تأثيراتها الرقمية على قياس الإنتاجية ودرجة تأثيرها على مسار تطور المشروع.

بينت دراسة أخرى مدى إمكانية الاستفادة من المقاييس التي تعتمد على الكود المصدري metric-based metrics والمقاييس التي تعتمد على الالتزام بالمهام task-committed metrics في دعم تصور قادة الفرق الذاتي عن إنتاجية فرقهم وما يتبعها من قرارات متعلقة بتقدير وقت وتكلفة المشاريع. تم أخذ المقاييس التالية التي تعتمد على الكود المصدر بعين الاعتبار في هذه الدراسة: (1) عدد أسطر الرماز المصدري المنجزة من قبل المطور في الزمن Source Line of Code over time (SLOC/Time) (2) تعقيد الكود المنجز من قبل المطور في الزمن Halstead Effort by Time (HalsteadEff/Time) (3) عدد ملفات الكود المصدري المنجز من قبل المطور في الزمن Code Owned by Time (CodeOwned/Time). من جهة أخرى تم أخذ المقاييس التالية التي تعتمد على الالتزام بالمهام بعين الاعتبار في هذه الدراسة: (1) عدد العمليات التي يلتزم فيها المطور في الزمن Commits Performed by Time (Commits/Time) (2) عدد أسطر الكود المصدري التي يلتزم فيها المطور في الزمن Committed Source Lines of Code by Time (CommittedSLOC/Time) (3) عدد المحارف التي يلتزم فيها المطور في الزمن Committed Characters by Time (CommittedChars/ Time). أجريت الدراسة بشكل تجريبي على شركتين برمجيتين لتسعة مشاريع منجزة حيث تم الطلب من قادة الفرق إعطاء تصور ذاتي عن إنتاجية أعضاء فرقهم، من ناحية أخرى تم حساب المقاييس سابقة الذكر لقياس مدى الترابط بين قيم هذه المقاييس وتصور قادة الفرق. بينت نتائج الدراسة وجود ارتباط بين تصورات قادة الفرق عن إنتاجية فرقهم والمقاييس القائمة على الكود المصدري عند مقارنتها بالمقاييس القائمة على الالتزام. بينت الدراسة أن تطبيقها في المجال العملي مازال يعتبر تحدياً وخصوصاً أن مرجع الترابط الصحيح كان تصور قادة الفرق، كما بينت قصورها عن عكس نتائج مقاييس الإنتاجية على قرارات إدارة الفريق [11].

لا يكمن تحدي إدارة المشاريع الرشيقة في توقع زمن إنجازها وتكلفتها اعتماداً على قياس إنتاجية الفرق فحسب وإنما قياس تقدم مسار المشروع للتعامل بشكل فعال مع التغيير المتأصل في طبيعة هذه المشاريع مع تحقيق التوافق مع مهمة المشروع ورؤيته. تقيم هذه الدراسة فعالية النموذج المقترح لخريطة تقدم المشاريع البرمجية الرشيقة [12]. اعتمدت عملية التحقق والتقييم على تطبيق Tobi Board لبناء خريطة التقدم لـ 24 فريق مؤلف من طلبة ماجستير لإنجاز تقييم بيئات عمل محددة، تقييم أفكار العمل، تحديد متطلبات الزبائن وغيرها. اتضح أن Tobi Board يملك قيمة بتوثيق مسارات التقدم ويوفر طريقة منظمة لإعطاء نظرة عامة للجميع عن أنشطة المشروع المنجزة والمستقبلية، لكنها تفتقر إلى وجود تخطيط واضح للإنجاز في كثير من الحالات. تم إثبات صحة مفهوم خريطة التقدم بشكل تجريبي إلى حد ما عن طريق التماس فوائد الاستخدام التجريبية من قبل الفرق حتى ولو كانت غير تجارية [13]. تفتقد الدراسة إلى إمكانية التعميم أو اقتراح مقاييس قادرة على قياس مقدار التقدم بشكل كمي مضبوط يؤمن المدخل اللازم لقادة الفرق لاتخاذ القرارات المناسبة فيما يتعلق بتقدم المشروع.

4.مقاييس التقدم والإنتاجية Progress and Productivity Metrics:

تم مؤخراً اقتراح مقاييس لقياس التقدم progress والإنتاجية productivity في الإجرائيات الرشيقة Agile Development دون تقديم تحقق منهجي لهم. يقدم هذا القسم هذه المقاييس [3] وهي مقاييس غير مباشرة تعتمد بشكل أساسي في حسابها على المقاييس التالية:

- مقياس الحجم Size لقصة مستخدم

يعتمد مقياس الحجم على عدد أسطر الكود المصدري Line of Code (LOC) في تنفيذ قصة المستخدم User Story.

- مقياس قيمة الأعمال المكتسبة Earned Business Value (EBV)

يستهدف هذا المقياس [14] التطوير الرشيق لقياس تطور الإنجاز نسبة لتحقيق قيمة الأعمال المكتسبة من القيمة الكلية لإنجاز المشروع البرمجي. يعتمد هذا المقياس على مبدأ أن قيمة الأعمال تعتمد على مدى إنجاز الأعمال التي لها قيمة من وجهة نظر الزبون. لا يمكن تحديد هذه القيم بشكل مطلق وإنما يتم تحديدها بشكل نسبي على اعتبار أن القيمة الكلية للأعمال هي (1) وتتحقق بإنجاز كامل أعمال المشروع. لتحديد مدى التقدم على مستوى قيمة الأعمال نتيجة إنجاز المهام يتم استخدام Work Breakdown Structure (WBS) والذي يهدف إلى تشكيل شجرة من الأعمال ناتجة عن التقسيم المتكرر لها ومن ثم إعطاء قيم وزن لكل منها حسب درجة أهميتها نسبة للأعمال من نفس الدرجة لهذه الشجرة (Siblings). من هذا المنطلق تحسب قيمة الأعمال لعمل ما BV(Bucket) بشكل عودي عن طريق حساب نسبة وزن هذا العمل wt(bucket) إلى أوزان جيرانه مضروباً بقيمة الأعمال للمستوى الأعلى منه BV(parent) وهكذا، حيث يعبر عنه بالمعادلة التالية:

$$BV(Bucket) = BV(Parent) \times \left[\frac{wt(Bucket)}{wt(Bucket) + \sum_{sibilings} wt(sibiling)}\right]$$

وبالتالي تحسب قيمة الأعمال المنجزة بالنسبة للمشروع EBV (project) عن طريق مجموع قيم الأعمال لقصص المستخدم المنجزة:

$$EBV(Project) = \sum_{completed} BV(UserStory)$$

- مقياس وظيفة قصة مستخدم Functionality (F)

لقياس وظيفة F_i متطلب من وجهة نظر المستخدم من حيث الوظائف التي يمكن يستفيد منها نتيجة تنفيذ هذا المتطلب، يمكن استخدام مقياس Function Point [15].

$$FP = TC \times [0.65 + 0.01 \times \sum_{i=1}^{14} X_i\ ]$$

حيث:

TC: هو مجموع عمليات الضرب الفردية لعدد العوامل التالية بوزن كل معامل حسب الجدول 1.

الجدول (1) أوزان معاملات مقياس الوظيفة

العامل		عوامل التثقيل		
	العدد	منخفض	متوسط	مرتفع
المدخلات الخارجية External Inputs(EI)		3	4	6
المخرجات الخارجية External (EO) Outputs.		4	5	7
الاستعلامات الخارجية External (EI) Inquiries.		3	4	6
ملفات المنطق الداخلية Internal (ILF) Logical Files		7	10	15
ملفات المنطق الخارجية External (ELF) Logical Files		5	7	10

X_i: قيمة السؤال i وفق مقياس تتراوح قيمته من '0' إذا كان السؤال ليس له أي تأثير إلى '5' في حال كون السؤال يملك تأثيراً كبيراً.

4-1 مقياس إنتاجية الفريق ضمن تكرار PTI:

$$PTI = K \times \frac{T_{SI}}{T_{PI}}$$

حيث:

T_{SI}: عدد قصص المستخدم user stories الكلي المنفذة بشكل صحيح ضمن التكرار 'I'.

T_{PI}: عدد قصص المستخدم user stories الكلي المخطط لتنفيذها ضمن التكرار 'I'.

K: ثابت يعبر عن نسبة عدد أعضاء الفريق المشاركين فعلياً في عمليات التطوير ضمن التكرار 'I' إلى عدد الأعضاء ضمن الفريق.

4-2 مقياس زيادة التقدم ضمن تكرار IPWI:

يتم قياس زيادة التقدم في التكرار الواحد عن طريق حساب مجموع الزيادات في نسبة صحة الكود المستخدم في إنجاز كل قصة المستخدم 'i' user story قبل المباشرة في عمليات التصحيح وبعد الانتهاء منها.= $IPWI$

$$\sum_{i=1}^{n}\left(\frac{L_{i_1} - L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0} - L_{Ei_0}}{L_{i_0}}\right)$$

حيث:

n: عدد قصص المستخدم المنفذة بنجاح في التكرار 'I'.

L_{i_0}: عدد أسطر الكود LOC (Line of Code) الكلي لتنفيذ قصة المستخدم 'i' قبل الشروع بأي عملية تصحيح.

L_{Ei_0}: عدد أسطر الكود LOC (Line of Code) لتنفيذ قصة المستخدم 'i' والتي تحتوي أخطاء قبل الشروع بأي عملية تصحيح.

L_{i_1}: عدد أسطر الكود LOC (Line of Code) الكلي لتنفيذ قصة المستخدم 'i' بعد الانتهاء من عمليات التصحيح.

L_{Ei_1}: عدد أسطر الكود LOC (Line of Code) لتنفيذ قصة المستخدم 'i' والتي تحتوي أخطاء بعد الانتهاء من عمليات التصحيح.

4- 3 مقياس التقدم الفعلي ضمن تكرار API

$$API = \frac{\sum_{i=1}^{n}[EBV_i \times F_i]}{\sum_{j=1}^{N}[EBV_j \times F_j]}$$

حيث:

n: عدد قصص المستخدم المنفذة بنجاح في التكرار 'I'.

N: عدد قصص المستخدم المخطط تنفيذها في التكرار 'I'.

EBV_i: قيمة الأعمال المكتسبة نتيجة التنفيذ الصحيح لقصة المستخدم 'i'.

EBV_j: قيمة الأعمال المكتسبة المتوقع الحصول عليها فيما لو أنجز التنفيذ الصحيح لقصة المستخدم 'j'.

F_i: قيمة وظيفة قصة المستخدم 'i' نتيجة التنفيذ الصحيح لها.

F_j: قيمة وظيفة قصة المستخدم 'i' المتوقع الحصول عليها فيما لو أنجز التنفيذ الصحيح لها.

5.طرق التحقق النظري من مقاييس البرمجيات Software Metrics Theoretical Validation Approaches

تستخدم المقاييس البرمجية Software metrics عادة من قبل مجتمع هندسة البرمجيات لفهم وتطوير المنتجات البرمجية وإجراءات التطوير البرمجي عن طريق تحويل العناصر التي يتم قياسها إلى أرقام لها معنى على المستوى الداخلي للتطبيق أو الإجرائية أو من ناحية أخرى لربطهما مع معايير الجودة. ولكن يبقى السؤال عن مدى قدرة هذه المقاييس إعطاء صورة حقيقية عن العناصر التي يتم قياسها لعدم وجود آليات وقواعد واضحة للتحقق من المقاييس المطورة. تم مسح الأبحاث السابقة من خلال استبيان بحثي [16] قام بتجميع جميع معايير التحقق المعتمدة في الأبحاث وربط هذه المعايير بتصنيفات أساسية وأخرى فرعية.

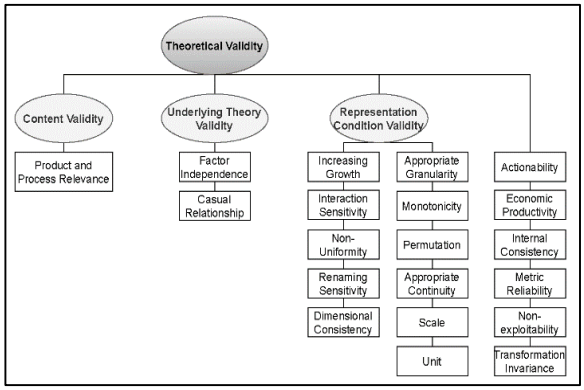
تم تحديد ثلاث تصنيفات رئيسية للتحقق من المقاييس البرمجية وهي التحقق الداخلي والخارجي والبناء.

التحقق الداخلي Internal Validity ويهتم بالتأكد من أن المقياس يقيس بالشكل الصحيح الواصفات التي من المفروض أن يقيسها [17]، ومن أهم تصنيفاته الفرعية هي التحقق النظري Theoretical Validation.

التحقق الخارجي External Validity ويركز على التأكد بأن المقياس مرتبط بطريقة أو بأخرى سواء بالتوقع أو السببية مع أحد عوامل الجودة الخارجيين [17]، ومن أهم تصنيفاته الفرعية هي التحقق التجريبي Empirical Validation

تحقق البناء Construct Validity ويتم من خلاله التأكد من أن القياسات التي يقوم بها المقياس مناسبة حسب تعريف الواصفات التي يستهدفها هذا المقياس. ويتم التركيز هنا بالبناء على التحقق من أي أداة أو جهاز أو إجراء يستخدم لجمع هذه القياسات [18].

يجب عدم هدر الوقت في الشروع بعمليات التحقق الخارجي أو البناء إلا بعد التحقق من أن المقياس يقيس بشكل صحيح الواصفات التي من المفروض أن يقيسها، أي بعد إتمام عمليات التحقق الداخلي والنظري، لذلك تم التركيز في هذه الورقة البحثية على عمليات التحقق الداخلي Internal validation وبشكل خاص التحقق النظري Theoretical Validationوما يتضمنه من تصنيفات فرعية حسب الشكل1.



الشكل (1) تصنيفات التقييم النظري المقاييس البرمجية [16]

يندرج تحت صنف التحقق النظري مجموعة من معايير التحقق التابعة له مباشرة وثلاثة تصنيفات أخرى فرعية وهي: التحقق من المحتوى Content Validity، التحقق من النظرية Underlying-Theory Validity، التحقق من شرط العرض Representation Condition Validity.

5-1 معايير التحقق التابعة مباشرة للصنف

باعتبار أن U هي الوحدة القابلة للقياس، M هو المقياس و(M(U هي قيمة المقياس M التي تقيس الواصفة P في الوحدة القابلة للقياس U:

المعيار 1: الفعل Actionability

يحقق أي مقياس هذا المعيار في حال كونه مفيد لمستخدمه باتخاذ قرار فعال متعلق بالتطبيق بناء على القيمة التي يعطيها [19].

المعيار 2: الانتاجية الاقتصادية Economic Productivity

يتحقق هذا المعيار في حال أخذ المقياس بالحسبان التوازن بين التكلفة والفائدة التي يحققها، أي يجب على المقياس أن يهتم بالإضافة إلى تحقيق قياسات جيدة دعم التكلفة المنخفضة لعملية القياس [20].

المعيار 3: الاتساق الداخلي Internal Consistency

يجب على القياسات المبدئية التي يحتاجها المقياس أن تستهدف نفس الوحدة وأن تكون متكاملة مع بعضها بطريقة منضبطة للحصول على قيمة القياس النهائية [18].

المعيار4: وثوقية المقياس Metric Reliability

يجب على المقياس في حال تحقيقه هذا المعيار أن يكون دقيقاً أي يعطي نتائج مع هامش خطأ قليل بغض النظر عن آلية تنفيذ عملية القياس [18].

المعيار 5: عدم الاستغلال Non- exploitability

يحقق المقياس هذا المعيار في حال لم يستطع المطورون تطويع المقياس بحيث تبدو النتائج أفضل صورياً [21].

المعيار6: ثبات التحويل Transformation Invariance

يمكن القول أن المقياس يحقق هذا المعيار إذا أعطى نفس قيمة القياس لوحدتين متكافئتين بالسياق semantically- equivalent [22].

5-2 تصنيف التحقق من المحتوى Content validity:

يمكن أن يحقق مقياس ما تصنيف التحقق من المحتوى في حال إمكانية شموليته لكامل مجال الواصفات التي يقوم بقياسها. يندرج تحت هذه التصنيف مجموعة من المعايير.

المعيار7: علاقة المقياس بالمنتج أو الإجراء Product and Process Relevance

يجب على المقياس أن يكون مخصصاً لمنتجاً برمجياً في مجال محدد أو إجراء تطويري محدد. ومع ذلك إذا تم إثبات فعالية مقياس في مجال ما يمكن أن يكون فاعلاً في مجالات أخرى [5]، [23]، [24].

5-3 تصنيف التحقق من النظرية Underlying Theory Validity:

يركز هذا الصنف على التحقق من أن المقياس البرمجي يجب أن يكون مبنياً اعتماداً على نظرية محددة ومثبتة في المجال الذي يطبق عليه المقياس، أي يجب أن يعتمد المقياس على نموذج قياس محدد يصف العلاقة بين الواصفات التي تستهدف بعملية القياس [23]، [25]. يندرج تحت هذا الصنف معيارين (استقلالية العامل والعلاقة السببية).

المعيار8: استقلالية العامل Factor independence

يمكن القول أن المقياس يحقق هذا المعيار إذا كانت القياسات الفردية التي تتم من خلال هذا المقياس مستقلة عن بعضها البعض [23].

المعيار9: العلاقة السببية Causal Relationship validity

يمكن الحكم أن المقياس يحقق العلاقة السببية في حال تم إثبات وجود علاقة سببية بين هذا المقياس وبين إحدى عوامل الجودة الخارجية [18].

5-4 تصنيف شرط العرض representation condition:

يركز هذا الصنف على التحقق من أن الواصفات المقاسة تحافظ على خصائص النظام العددي الذي تقابله [4]، [25]. يندرج تحت هذا الصنف مجموعة من أحد عشر معيار.

المعيار10: درجة التقسيم المناسبة Appropriate Granularity

يحقق المقياس هذا المعيار في حال أن عملية التحويل من الواصفة المقاسة إلى قيمة رقمية ليست شديدة النعومة finely-grained أو شديدة الخشونة coarsely-grained [25]، [26]. يمكن تمييز ثلاث درجات لهذا المعيار:

- التقسيم الناعم Fine granularity ويتحقق حال وجود عدد منته من الوحدات القابلة للقياس التي تعطي نفس القيمة عند تطبيق المقياس.
- التقسيم المتوسط Medium granularity:
- $(\exists U1)(\exists U2)\big(U1 \equiv U2\ and\ M(U1) \neq M(U2)\big)$

هذا يعني أنه في حال وجود وحدتين لهما نفس الوظيفة فهذا لا يؤدي بالضرورة لأن يكون لهما نفس القيمة عند تطبيق المقياس وذلك بسبب اختلافهما تصميمياً. بمعنى آخر يجب على المقياس أن يميز الحالة التصميمية للوحدة المقاسة وليس فقط الوظيفة التي تنجزها.

- التقسيم الخشن coarse granularity:
- $(\exists U1)(\exists U2)\big(M(U1) = M(U2)\big)$

هذا يعني أنه يمكن أن يوجد وحدتين لهما نفس القيمة عند تطبيق المقياس.

المعيار11: زيادة النمو Increasing growth validity

$$(\exists U1)(\exists U2)(U1 + U2 = U3) \Rightarrow M(U1) + M(U2) \geq M(U3)$$

هذا يعني أنه في حال تم إنشاء وحدة عن طريق الجمع بين وحدتين قابلتين للقياس فإن قيمة المقياس على الوحدة الناتجة يجب أن تكون مساوية بالحد الأعلى مجموع قيمتي المقياس على الوحدتين المشكلتان له في حال كون الوحدتين غير مترابطتين وأصغر من مجموع قيمتي المقياس على الوحدتين المشكلتين له في حال كون الوحدتين مترابطتين

وذلك بسبب اندماج بعض الخصائص مع بعضها نتيجة دمج الوحدتين [24]، [26].

المعيار12: حساسية التفاعل Interaction sensitivty

$$(\exists U1)(\exists U2)(\exists U3)(M(U1) = M(U2) \Rightarrow (M(U1 + U2) \neq M(U1 + U3))$$

هذا يعني أن التكامل بين كل من وحدتين لهما نفس قيمة المقياس مع وحدة برمجية أخرى لا يؤدي بالضرورة للحصول على نفس قيمة المقياس [26] .

المعيار13: الرتابة Monotonicity

$$U1 \subseteq U2 \Rightarrow M(U1) \leq M(U2)$$

هذا يعني أن قيمة المقياس عند تطبيقه على وحدة لا يزيد على قيمة المقياس عند تطبيقه على كامل برنامج مؤلف من هذه الوحدة ووحدات أخرى [24]، [26].

المعيار14: عدم الرسمية Non-uniformity

$$(\exists U)(\exists Q)(M(U) \neq M(Q))$$

هذا يعني أنه يجب على المقياس ألا يعطي نفس القيمة على الأقل لوحدتين مختلفتين [26].

المعيار15: تحقق التعلق Permutation validity

يجب على المقياس أن يعكس تسلسل تكامل الوحدات القابلة للقياس ففي حال كون وحدة هي تعديل لوحدة برمجية أخرى، فيجب أن يكون لهما قيميتين مختلفتين عند تطبيق المقياس[26].

المعيار 16: عدم حساسية تعديل الاسم Renaming Insensitivity

يحقق المقياس هذا المعيار في حال لم تتأثر نتيجة القياس بعد تعديل اسم الوحدة القابلة للقياس. أي إن تم إنتاج وحدة من وحدة أخرى عن طريق تعديل اسمها فقط، فيجب أن يكون لهما قيمتين متساويتين عند تطبيق المقياس [26].

المعيار17: الاستمرارية Appropriate continuity

يجب ألا يعاني المقياس الجيد من أي انقطاعات غير متوقعة في مجال القيم حسب الواصفة التي يقوم بقياسها [25].

المعيار18: اتساق الأبعاد Dimensional Consistency

يجب على نموذج القياس أن يكون متسقاً بأبعاده. يمكن أن يحقق المقياس لهذه الخاصة إذا كانت تشكيل المقياس المركب والمؤلف من عدة مقاييس يعتمد على تابع رياضي مفهوم ومعلوم يحقق الاتساق بين وحدات المقاييس المشكلة للمقياس المركب [25].

المعيار19: تحقق المقدار Scale validity

يحقق المقياس هذا المعيار في حال تم تعريف المقياس ونطاقه بشكل واضح بحيث تكون جميع التحويلات المنطقية التي يمكن أن تجري عليه مقبولة. بالإضافة يجب أن تكون قيمة المقياس عند تطبيقه على الوحدة القابلة للقياس موجبة وفي عدم تواجد الواصفة المقاسة ضمن الوحدة القابلة للقياس فإن قيمة قياسها هي قيمة صفرية [24]، [25].

$$M(U) \geq 0$$
$$P(U) = \varphi \Rightarrow M(U) = 0$$

المعيار20: تحقق الوحدة Unit validity

يحقق المقياس هذه المعيار في حال تم استخدام واحدت مناسبة لها معاني مفيدة لنتائج القياس التي يقوم بها [25].

6.منهجية التحقق Validation Approach

تعتمد منهجيات التحقق على وجهتين فلسفيتين مختلفتين بالطبيعة الفلسفة المقادة بالهدف والفلسفة المقادة بالنظرية. تعتمد منهجية الفلسفة المقادة بالهدف على قاعدة أن فائدة وفعالية وصحة أي مقياس تكمن في التحقق من أنه يلبي الغرض أو الهدف الرئيسي من استخدامه، ولا فائدة من كون المقياس صالحاً في ناحية بنيته الداخلية مالم يحقق هدف استخدامه، بينما ترى منهجية الفلسفة المقادة بالنظرية أن صحة أي مقياس بجب أن يعتمد على صحة بنيانه الداخلي بالدرجة الأولى واتباعه منهجاً نظرياً معرفاً.

نلاحظ درجة من التعارض بين منهجيتي التحقق، ولكن من ناحية أخرى يمكن النظر إليهما من منظور تكاملي بحيث يمكن للباحث أو الممارس أن يختار بعض الجوانب من كليهما بما يتوافق مع الدوافع والأهداف المنطقية لاستخدام

المقياس تحت الاختبار. من هذا المنطلق تم تحديد مجموعة من المزايا advantages المرتبطة بالأهداف التي يمكن أن يستهدفها أي مقياس [16]:

السلامة الرياضية Mathematical Soundness: تمكن هذه المزية في حال التأكد منها مستخدمي المقياس من أداء العمليات الحسابية المحددة دون الشك أن ترجمة النتائج الرقمية له لن تعبر بشكل سليم عن الواصفة التي يستهدف قياسها.

التطبيق العملي Practicality: تمكن هذه المزية في حال التأكد منها من الوثوق بإمكانية تطبيق المقياس بشكل مباشر للحصول على نتائج مجدية.

الصحة Correctness: تمكن هذه المزية في حال تحققها التأكد أنه من النادر أن يعطي المقياس نتائج غير صحيحة.

الكفاءة Efficiency: تمكن هذه المزية في حال تحققها توفير الوقت والجهد باختيار مقياس واحد لهدف قياس معين.

الكشف عن الفروق Difference-detecting: تتيح ميزة اكتشاف الفروق للمقاييس أن تكتشف بعناية الاختلافات البسيطة في الواصفة المراد قياسها.

تعزيز الفرضيات Hypothesis-strengthening: تتحقق هذه المزية في حال التأكد من الوصول إلى ثقة أكبر في الفرضيات أو الاستنتاجات أو النظرية المعلنة من خلال القياسات المتكررة التي يجريها المقياس.

الجدوى Meaningfulness: تتحقق هذه المزية في حال وجود علاقة مفهومة وبديهية بين المقياس والواصفة التي يقيسها.

إعلام القرار Decision-informing: تتحقق هذه المزية في حال ساعد المقياس الباحثين والممارسين على اتخاذ قرارات بناء على القيم الناتجة عنه.

التركيز على الجودة Quality-focused: تتحقق هذه المزية في حال ثبت أن المقياس يمكن أن يطور برامج عالية الجودة مما يعود بالنفع على الناس والمجتمع.

بناء النظرية Theory-building: تمكن ميزة بناء النظرية الباحثين أو الممارسين من الوصول لاستنتاجات نظرية حول طبيعة البرمجيات بشكل عام وتطورها.

مساهمة توافقية Consensus Contribution: تتحقق هذه المزية في وجود حالة اتفاق بين الممارسين والباحثين حول كيفية جمع بيانات وتفسير المقياس بنفس الطريقة.

يبين الجدول 2 والمستنتج من [16] معايير التحقق المقابلة للمزايا المستهدفة للمقياس تحت الاختبار.

للتحقق من فعالية المقاييس المستخدمة، يمكن اتباع الخطوات التالية [16]:

a. تحديد الهدف من استخدام المقياس وعملية القياس التي يقوم بها.
b. ربط الأهداف التي تم تحديدها مع مجموعة من المزايا المحددة في الجدول والتي يركز عليها مجتمع البحث أو فريق التطوير البرمجي.
c. اختيار معايير التحقق النظرية التي توافق المزايا المحددة في الخطوة السابقة حسب الجدول 2.
d. مناقشة مدى اتباع المقياس تحت الاختبار للمعايير المختارة في الخطوة السابقة.
e. مناقشة نتائج التقييم بشكل كيفي يبرز المزايا المحققة من قبل المقياس تحت الاختبار والنقط السلبية التي تقيد استخدامه مما يشكل وجهة واضحة للباحثين لتطويره وتحسين أدائه.

الجدول (2) معايير التحقق المقابلة للمزايا المستهدفة للمقياس تحت الاختبار

رقم المعيار	المزايا / معايير التحقق	السلامة الرياضية	التطبيق العملي	الصحة	الكفاءة	الكشف عن الفروق	تعزيز الفرضيات	الجدوى	إعلام القرار	التركيز على الجودة	بناء النظرية	مساهمة وظيفية
1	الفعل		×						×			
2	الانتاجية الاقتصادية		×						×			
3	الاتساق الداخلي							×				
4	وثوقية المقياس			×								
5	عدم الاستغلال		×									
6	ثبات التحويل					×						
7	علاقة المقياس بالمنتج أو الإجراء		×									
8	استقلالية العامل						×	×				
9	العلاقة السببية		×						×		×	
10	درجة التقسيم المناسبة					×						
11	زيادة النمو	×										×
12	حساسية التفاعل	×										×
13	الرتابة	×										
14	عدم الرسمية	×				×		×				
15	تحقق التعلق							×				
16	عدم حساسية تعديل الاسم							×				
17	الاستمرارية	×						×				×
18	اتساق الأبعاد	×						×				
19	تحقق المقدار	×										
20	تحقق الوحدة							×				

c. بناءً على اختيار المزايا التي سوف يتم التحقق من أن مقاييس الإنتاجية والتقدم تلبيها وباستخدام الجدول 2، تم تحديد واختيار معايير التحقق النظرية التي توافقها.

d. مناقشة مدى اتباع المقياس تحت الاختبار للمعايير المختارة في الخطوة السابقة حسب ما هو مبين في الملحق 1 للمقياس PTI والملحق 2 للمقياس IPWI والملحق 3 للمقياس API.

e. مناقشة مدى تحقيق المقاييس تحت الاختبار المزايا المستهدفة. يبين الجدول 3 نتائج تقييم كل من المقاييس الستة من حيث تحقيقها للمزايا من خلال مناقشة جميع معايير التحقق المرتبطة بكل من المزايا التسع المستهدفة وفق منهجية التحقق النظري، حيث يمكن الاطمئنان بتحقيق المقياس تحت الاختبار مزية محددة في حال تلبية هذا المقياس لأغلبية معايير التحقق المرتبطة بهذه الميزة وكذلك الأمر يمكن الحكم بعدم تحقيق المقياس تحت الاختبار مزية محددة في حال كانت أغلبية معايير التحقق المرتبطة بها غير محققة.

7.النتائج والمناقشة Results and Discussion:

بتطبيق منهجية التحقق الموضحة في الفقرة السابقة لتقييم مقاييس التقدم والإنتاجية المقترحة للإجرائيات الرشيقة في تطوير البرمجيات (PTI, IPWI, API)، نجد مايلي:

a. تهدف المقاييس المقترحة إلى قياس إنتاجية الفريق ومراقبة مدى التقدم في التكرار الواحد حسب الخطة الموضوعة في منهجيات تطوير برمجية رشيقة للمساعدة في اتخاذ القرارات ذات الصلة والتي تساعد في تقدير زمن وتكلفة المشروع عن طريق تقدير الموارد اللازمة والسعي نحو أمثلة الأداء لتخفيض التكلفة وتسريع الوصول إلى المنتج البرمجي المراد تطويره وفق خصائص متوافقة مع احتياجات الزبون.

b. يمكن ربط هدف مقاييس الإنتاجية والتقدم المقترحة والمبينة في البند (a) مع جميع المزايا المحددة في الجدول وهي: السلامة الرياضية والتطبيق العملي والصحة والكشف عن الفروقات وتعزيز الفرضيات والجدوى وإعلام القرار وبناء النظرية ومساهمة توافقية. في المقابل لم يتم التركيز على التحقق من توفر المزايا التالية (جودة المقياس والكفاءة) وذلك لعدم وجود معايير تحقق تقابل هذه المزايا وفق المنهجية النظرية للتحقق من المقاييس.

7-1التحقق من مقياس إنتاجية الفريق ضمن تكرار PTI:

يمكن الاطمئنان أن مقياس إنتاجية الفريق PTI يعطي مؤشراً يمكن الاعتداد به عند قياس الإنتاجية وذلك بسبب تلبيته معظم المزايا المستهدفة مع ضرورة الأخذ بعين الاعتبار المزايا غير المحققة وضرورة تجنب استخدام المقياس في حال كانت هذه المزايا ذات أولوية مطلقة.

تم التأكد أولاً من أداء المقياس للعمليات الحسابية المحددة دون الشك بمؤشرات النتائج الرقمية له من خلال دراسة موافقته لجميع معايير التحقق الخاصة بمزية السلامة الرياضية وهم (11) زيادة النمو و (12) حساسية التفاعل و (13) الرتابة و(14) عدم الرسمية و (17) الاستمرارية و (18) اتساق الأبعاد و (19) تحقق المقدار.

من ناحية ثانية، تم التأكد من إمكانية تطبيق المقياس بشكل مباشر للحصول على نتائج مجدية من خلال دراسة موافقته لمعايير التحقق الخاصة بمزية التطبيق العملي حيث وافق المقياس كل من (1) الفعل و(2) الإنتاجية الاقتصادية و(5) عدم الاستغلال و(7) علاقة المقياس بالمنتج أو الإجراء بينما فشل وفق معيار (9) العلاقة السببية حيث لا يوجد تعريف يوضح المقصود بالإنجاز الناجح لقصص المستخدم من جهة ولا يوجد علاقة تربط بينها وبين إنتاجية الفريق التي يستهدفها المقياس.

من ناحية ثالثة، تم التأكد من استمرارية إعطاء المقياس لنتائج مستقرة وصحيحة من خلال دراسة موافقته لمعيار التحقق الخاص بمزية الصحة وهو (4) وثوقية المقياس.

من ناحية رابعة، فشل المقياس في اكتشاف الفروق البسيطة في الواصفة المراد قياسها من خلال عدم موافقته لأغلب معايير التحقق الخاصة بمزية الكشف عن الفروقات حيث لم يوافق المقياس معياري (6) ثبات التحويل بسبب عدم إعطائه نتيجة واحدة لمجموعتين متكافئتين سياقياً لقصص المستخدم و(10) درجة التقسيم المناسبة حيث لم يتم تحديد نعومة درجة التقسيم المناسبة له بينما اكتفى بموافقته لمعيار (14) عدم الرسمية.

من ناحية خامسة، تم تأكيد الثقة في الفرضيات التي يعتمد عليها هذا المقياس من خلال دراسة موافقته لمعيار التحقق الخاص بمزية تعزيز الفرضيات حيث وافق المقياس معيار (8) استقلالية العامل.

من ناحية سادسة، تم تأكيد وجود علاقة مفهومة وبديهية بين المقياس والواصفة التي يقيسها من خلال دراسة موافقته لمعايير التحقق الخاصة بمزية الجدوى حيث وافق المقياس كل من (3) الاتساق الداخلي و(8) استقلالية العامل و(14) عدم الرسمية و(16) عدم حساسية تعديل الاسم و(17) الاستمرارية و(18) اتساق الأبعاد و(20) تحقق الوحدة، بينما لم يوافق المعيار (15) تحقق التعلق حيث لم يستطع المقياس أن يميز بين حالتين مدروستين بالرغم من اختلاف قصص المستخدم في كل حالة.

من ناحية سابعة، تم التأكد من مساعدة المقياس المستخدمين على اتخاذ قرارات بناء على القيم الناتجة عنه من خلال دراسة موافقته لمعايير التحقق الخاصة بمزية إعلام القرار حيث وافق المقياس كل من (1) الفعل و (2) الإنتاجية الاقتصادية ولم يوافق المعيار (9) العلاقة السببية كما تمت الإشارة من قبل.

من ناحية ثامنة، فشل المقياس من الاعتماد على نظرية تمكن الباحثين والممارسين من الوصول إلى استنتاجات مهمة اعتماداً عليها من خلال عدم موافقته لمعيار (9) العلاقة السببية.

من ناحية تاسعة، تم التأكد من الاتفاق على كيفية جمع بيانات المقياس وتفسير نتائجه من خلال دراسة موافقته لمعايير التحقق الخاصة بالمساهمة التوافقية حيث وافق المقياس كل من المعايير (11) زيادة النمو و(12) حساسية التفاعل و (17) الاستمرارية.

7-2 التحقق من مقياس زيادة التقدم ضمن تكرار IPWI:

بعد التحقق من مقياس زيادة التقدم ضمن تكرار IPWI الفريق PTI حسب المزايا التسع الممكنة نلاحظ أن المقياس لا يلبي معظمها مما يعطي مؤشراً واضحاً بضرورة تطوير المقياس وعدم الاطمئنان الى النتائج التي يقيسها.

في التفصيل، تم التأكد أولاً من أن المقياس يدعم العمليات الحسابية من خلال دراسة موافقته لمعظم معايير التحقق الخاصة بمزية السلامة الرياضية وهم (11) زيادة النمو و (13) الرتابة و(14) عدم الرسمية و (18) اتساق الأبعاد. بينما لم يوافق المقياس معايير (12) حساسية التفاعل بسبب عدم قدرته على اكتشاف الفروق بين تكاملات قصص المستخدم و (17) الاستمرارية بسبب وجود انقطاعات

حسابية و (19) تحقق المقدار حيث لا يمكن التأكد أن جميع التحويلات المنطقية يمكن تطبيقها على هذا المقياس.

من ناحية ثانية، فشل المقياس في إمكانية تطبيقه بشكل مباشر للحصول على نتائج مجدية من خلال عدم موافقته لأغلب لمعايير التحقق الخاصة بمزية التطبيق العملي حيث لم يوافق المقياس كل من (1) الفعل بسبب عدم إمكانية عكس نتائجه في مساعدة قائد الفريق على اتخاذ أي قرار و(2) الإنتاجية الاقتصادية بسبب الجهد والتكلفة العالية التي يتطلبها تطبيق هذا المقياس و(5) عدم الاستغلال بسبب إمكانية التلاعب بطريقة القياس للحصول على نتائج جيدة للتقدم في العمل و(9) العلاقة السببية حيث لا توجد علاقة موضحة تربط بين نتيجة هذا المقياس مع جودة المنتج النهائي، بينما اكتفى المقياس فقط بموافقته المعيار (7) علاقة المقياس بالمنتج أو الإجراء.

من ناحية ثالثة، لم يتم التأكد من استمرارية إعطاء المقياس لنتائج مستقرة وصحيحة من خلال عدم موافقته لمعيار التحقق الخاصة بمزية الصحة وهو (4) وثوقية المقياس حيث لا يمكن ضمان الحصول على نفس نتيجة القياس إذا تمت إعادة عملية القياس من قبل شخصين مختلفين.

من ناحية رابعة، فشل المقياس في اكتشاف الفروق البسيطة في الواصفة المراد قياسها من خلال عدم موافقته لأغلب معايير التحقق الخاصة بمزية الكشف عن الفروقات حيث لم يوافق المقياس معياري (6) ثبات التحويل بسبب عدم إعطائه نتيجة واحدة لمجموعتين متكافئتين سياقياً لقصص المستخدم و(10) درجة التقسيم المناسبة بسبب أن المقياس شديد النعومة لأن نتيجته تتأثر بعدد قصص المستخدم المنفذة وعدد اسطر الكود المنجزة لكل قصة مستخدم وعدد الأخطاء المكتشفة مع الأخذ بعين الاعتبار مدى التقدم في الإنجاز لحظة القياس، بينما اكتفى المقياس بموافقته لمعيار (14) عدم الرسمية.

من ناحية خامسة، تم تأكيد الثقة في الفرضيات التي يعتمد عليها هذا المقياس من خلال دراسة موافقته لمعيار التحقق الخاص بمزية تعزيز الفرضيات حيث وافق المقياس معيار (8) استقلالية العامل.

من ناحية سادسة، تم تأكيد وجود علاقة مفهومة وبديهية بين المقياس والواصفة التي يقيسها من خلال دراسة موافقته لأغلب معايير التحقق الخاصة بمزية الجدوى حيث وافق المقياس كل من (3) الاتساق الداخلي و(8) استقلالية العامل و(14) عدم الرسمية و(16) عدم حساسية تعديل الاسم و(18) اتساق الأبعاد، بينما لم يوافق المعيار (15) تحقق التعلق حيث لم يستطع المقياس أن يميز بين حالتين مدروستين بالرغم من اختلاف قصص المستخدم في كل حالة و(17) الاستمرارية بسبب وجود انقطاعين في الحساب و(20) تحقق الوحدة بسبب عدم إمكانية ترجمة محددة للقيم الناتجة.

من ناحية سابعة، فشل المقياس أيضاً في مساعدة المستخدمين على اتخاذ قرارات بناء على القيم الناتجة عنه من خلال عدم موافقته لجميع معايير التحقق الخاصة بمزية إعلام القرار حيث لم يوافق المقياس كل من (1) الفعل و (2) الإنتاجية الاقتصادية و(9) العلاقة السببية كما تمت الإشارة من قبل.

من ناحية ثامنة، فشل المقياس من الاعتماد على نظرية تمكن الباحثين والممارسين من الوصول إلى استنتاجات مهمة اعتماداً عليها من خلال عدم موافقته لمعيار (9) العلاقة السببية كما تمت الإشارة من قبل.

من ناحية تاسعة، لم يتم حسم الاتفاق على كيفية جمع بيانات وتفسير نتائج المقياس من خلال عدم موافقته لأغلب معايير التحقق الخاصة بالمساهمة التوافقية حيث لم يوافق المقياس معياري (12) حساسية التفاعل و(17) الاستمرارية كما تمت الإشارة من قبل بينما اكتفى بموافقته لمعيار (11) زيادة النمو.

8-1 التحقق من مقياس التقدم الفعلي ضمن تكرار API:

يمكن الاطمئنان أن مقياس التقدم الفعلي ضمن تكرار يمكن أن يعطي مؤشراً جيداً يعبر عن مدى تقدم الإنجاز الحاصل وذلك بسبب تلبيته ستة مزايا من أصل تسعة مستهدفة مع ضرورة الأخذ بعين الاعتبار المزايا غير المحققة وضرورة تجنب استخدام المقياس في حال كانت هذه المزايا ذات أولوية مطلقة.

تم التأكد أولاً من قدرة المقياس على إنجاز العمليات الحسابية المحددة دون الشك بمؤشرات النتائج الرقمية له من خلال دراسة موافقته لأغلب معايير التحقق الخاصة بمزية السلامة الرياضية وهم (11) زيادة النمو و (12) حساسية التفاعل و (13) الرتابة و(14) عدم الرسمية و (17) الاستمرارية، بينما لم يوافق المقياس معياري (18) اتساق الأبعاد بسبب عدم تفسير قيم الجداء لأبعاد غير متناسبة و(19) تحقق المقدار بسبب عدم التأكد من إمكانية إنجاز كل العمليات المنطقية على هذا المقياس.

من ناحية ثانية، تم التأكد من إمكانية تطبيق المقياس بشكل مباشر للحصول على نتائج مجدية من خلال دراسته موافقته لمعايير التحقق الخاصة بمزية التطبيق العملي حيث وافق المقياس كل من (1) الفعل و(5) عدم الاستغلال و(7) علاقة المقياس بالمنتج أو الإجراء بينما لم يحقق المقياس معياري (2) الإنتاجية الاقتصادية بسبب الجهد والوقت الكبيرين في إنجاز عمليات القياس و(9) العلاقة السببية حيث لا توجد علاقة موضحة تربط بين نتيجة هذا المقياس وجودة المنتج النهائي.

من ناحية ثالثة، تم التأكد من استمرارية إعطاء المقياس لنتائج مستقرة وصحيحة من خلال دراسة موافقته لمعيار التحقق الخاص بمزية الصحة وهو (4) وثوقية المقياس.

من ناحية رابعة، فشل المقياس في اكتشاف الفروق البسيطة في الواصفة المراد قياسها من خلال عدم موافقته لأغلب معايير التحقق الخاصة بمزية الكشف عن الفروقات حيث لم يوافق المقياس معياري (6) ثبات التحويل بسبب عدم إعطائه نتيجة واحدة لمجموعتين متكافئتين سياقياً لقصص المستخدم و(10) درجة التقسيم المناسبة بسبب أن المقياس شديد النعومة لأنه نتيجة القياس ستتأثر بعدد قصص المستخدم المنفذة والكلية وقيمة الأعمال المكتسبة ونقط الوظيفة الخاصة بكل منها ومدى التقدم في الإنجاز لحظة القياس، بينما اكتفى المقياس بموافقته لمعيار (14) عدم الرسمية.

من ناحية خامسة، تم تأكيد الثقة في الفرضيات التي يعتمد عليها هذا المقياس من خلال دراسة موافقته لمعيار التحقق الخاص بمزية تعزيز الفرضيات حيث وافق المقياس معيار (8) استقلالية العامل.

من ناحية سادسة، تم تأكيد وجود علاقة مفهومة وبديهية بين المقياس والواصفة التي يقيسها من خلال دراسة موافقته لمعظم معايير التحقق الخاصة بمزية الجدوى حيث وافق المقياس كل من (8) استقلالية العامل و(14) عدم الرسمية و(16) عدم حساسية تعديل الاسم و(17) الاستمرارية و(15) تحقق التعلق و(20) تحقق الوحدة، بينما لم يوافق المعيار (3) الاتساق الداخلي بسبب اعتماده على مقياسين غير مباشرين أحدهما داخلي والآخر خارجي و(18) اتساق الأبعاد بسبب عدم إمكانية تفسير قيمة جداء بعدين غير منسجمين.

من ناحية سابعة، فشل المقياس في مساعدة المستخدمين على اتخاذ قرارات بناء على القيم الناتجة عنه من خلال عدم موافقته لمعظم معايير التحقق الخاصة بمزية إعلام القرار حيث لم يوافق المقياس كل من (2) الإنتاجية الاقتصادية و(9) العلاقة السببية كما تمت الإشارة من قبل، بينما اكتفى بموافقة معيار (1) الفعل.

من ناحية ثامنة، فشل المقياس من الاعتماد على نظرية تمكن الباحثين والممارسين من الوصول إلى استنتاجات مهمة اعتماداً عليها من خلال عدم موافقته لمعيار (9) العلاقة السببية كما تمت الإشارة مسبقاً.

من ناحية تاسعة، تم التأكد من الاتفاق على كيفية جمع بيانات وتفسير نتائج المقياس من خلال دراسة موافقته

لجميع معايير التحقق الخاصة بالمساهمة التوافقية حيث وافق المقياس كل من المعايير (11) زيادة النمو و(12) حساسية التفاعل و (17) الاستمرارية.

الجدول (3) نتائج تقييم كل من المقاييس الستة

المقياس/ الخاصية	السلامة الرياضية	التطبيق العملي	الصحة	الكشف عن الفروقات	تعزيز الفرضيات	الجدوى	إعلام القرار	بناء النظرية	مساهمة توافقية
PTI	✓	✓	✓	✗	✓	✓	✓	✗	✓
IPWI	✓	✗	✗	✗	✓	✓	✗	✗	✗
API	✓	✓	✓	✗	✓	✓	✗	✗	✓

9.الخاتمة والتوصيات المستقبلية Conclusion and Future Work:

يعتبر دور المقاييس البرمجية في مختلف إجرائيات التطوير البرمجي ولا سيما المنهجيات الرشيقة مهماً في عمليات الإدارة والملاحقة والجودة على مستوى المشروع والإجرائية والمنتج واتخاذ القرارات ذات الصلة في الوقت المناسب. من ناحية أخرى، يؤدي الاستخدام غير المضبوط لهذه المقاييس إلى زيادة خطورة فشل المشروع بسبب الاعتماد على معلومات قياس غير دقيقة مما سيؤثر سلباً على أي قرار متخذ.

لهذا الغرض، قدمت هذه الدراسة تقييم عدد من مقاييس مقترحة للإنتاجية والتقدم وهي مقياس إنتاجية الفريق ضمن تكرار (PTI) ومقياس زيادة التقدم ضمن تكرار (IPWI) ومقياس التقدم الفعلي ضمن تكرار (API)، وذلك وفق منهجية تحقق نظرية للتأكد من فعالية ودقة استخدامها والاستفادة منها في المشاريع البرمجية التي تتبع المنهجيات الرشيقة في تطوير البرمجيات.

بينت نتيجة التحقق فعالية استخدام كل من مقاييس PTI و API وذلك لاتفاقها في موافقة معظم المزايا المستهدفة وهي السلامة الرياضية والتطبيق العملي والصحة وتعزيز الفرضيات والجدوى والمساهمة التوافقية مع ضرورة الأخذ بعين الاعتبار التريث في استخدام هذه المقاييس عندما يكون الهدف من عمليات القياس هي الكشف عن الفروقات أو بناء نظريات ذات علاقة. من ناحية أخرى اختلف هذين المقياسين من جهة موافقة خاصية إعلام القرار حيث يمكن الاستناد إلى مقياس PTI عند اتخاذ أي قرار متعلق بالمشروع في حين لا يمكن استخدام مقياس API لذلك.

من جانب آخر، تقترح الدراسة إعادة النظر في بنية مقياس IPWI وذلك لعدم تلبيتها لمعظم المزايا المستهدفة وبالتالي لا يمكن الوثوق فعلياً في استخدامها أو الاعتماد على النتائج التي يمكن الوصول إليها إلا في إطار ضيق ومحدود حسب المزية المستهدفة من عملية القياس.

تشكل هذه الدراسة نقطة انطلاق للمتابعة في مجال المقاييس البرمجية والاستفادة من تقييم المقاييس المقترحة للوصول إلى مقاييس أكثر نضجاً واستقراراً وفعالية في هذه المجال عن طريق حل معظم الإشكالات التي بينتها نتائج التقييم حسب معايير التحقق.

التمويل: هذا البحث ممول من جامعة دمشق وفق رقم التمويل (501100020595).

References:

1. Beck, K., Beedle, M., Bennekum, A.v., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R.C., Mellor, S., Schwaber, K., Sutherland, J., and Thomas, D. Manifesto for Agile Software Development. 2001 12.01.2023]; Available from: https://agilemanifesto.org/.

2. Ram, P., Rodriguez, P., and Oivo, M. Software Process Measurement and Related Challenges in Agile Software Development: A Multiple Case Study. 2018. Cham: Springer International Publishing.

3. Misra, S., Omorodion, F.M., and Damasevicius, R. Metrics for Measuring Progress and Productivity in Agile Software Development. in International Conference on Innovations in Bio-Inspired Computing and Applications. 2021. Cham: Springer International Publishing.

4. Fenton, N. and Bieman, J., Software Metrics: A Rigorous and Practical Approach, Third Edition. 2014: CRC Press, Inc.

5. Schneidewind, N.F., Methodology for validating software metrics. IEEE Transactions on Software Engineering, 1992. 18(5): p. 410-422.DOI: 10.1109/32.135774.

6. Gupta, M. and Kalia, A., Empirical Study of Software Metrics. Research Journal of Science and Technology, 2017. 9(1): p. 17-24.

7. Butt, S.A., Misra, S., Piñeres-Espitia, G., Ariza-Colpas, P., and Sharma, M.M. A Cost Estimating Method for Agile Software Development. in International Conference on Computational Science and Its Applications. 2021. Cham: Springer International Publishing.

8. Mashmool, A., Khosravi, S., Joloudari, J.H., Inayat, I., Gandomani, T.J., and Mosavi, A. A Statistical Model to Assess the Team's Productivity in Agile Software Teams. in 2021 IEEE 4th International Conference and Workshop Óbuda on Electrical and Power Engineering (CANDO-EPE). 2021.DOI: 10.1109/CANDO-EPE54223.2021.96.67902

9. Iqbal, J., Omar, M., and Yasin, A. An Empirical Analysis of the Effect of Agile Teams on Software Productivity. in 2019 2nd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET). 2019.DOI: 10.1109/ICOMET.2019.8.673413.

10. Canedo, E.D. and Santos, G.A., Factors Affecting Software Development Productivity: An empirical study, in Proceedings of the XXXIII Brazilian Symposium on Software Engineering. 2019, Association for Computing Machinery: Salvador, Brazil. p. 3.316–07

11. Oliveira, E., Fernandes, E., Steinmacher, I., Cristo, M., Conte, T., and Garcia, A., Code and commit metrics of developer productivity: a study on team leaders perceptions. Empirical Software Engineering, 2020. 25(4): p. 2519-2549.DOI: 10.1007/s10664-020-09820-z.

12. Schmidt, T.S., Gerdzhikov, G., and Paetzold, K. Set-based Design in Agile Development: Reinterpreting the Repository Tree. in 2018 IEEE International Conference on Engineering, Technology and Innovation

(ICE/ITMC). 2018.DOI: 10.11/09ICE.2018.8436330.

13. Schmidt, T.S., Behrenbeck, J., Burger, K., Hostettler, R., Paetzold, K., and Zimmermann, M., Mapping the Progress in Agile Product Development: A Multi-Case Study. Proceedings of the Design Society: International Conference on Engineering Design, 2019. 1(1): p. 1215-1224.DOI: 10.1017/dsi.2019.127.

14. Rawsthorne, D., Calculating earned business value for an agile project. Agile Journal, 2006.

15. Albrecht, A. Measuring application development productivity. in Proceedings of Joint Application Development Symposium. 1979.

16. Meneely, A., Smith, B., and Williams, L., Validating software metrics: A spectrum of philosophies. ACM Transactions on Software Engineering Methodology, 2013. 21(4): p. 1–28.DOI: 10.1145/2377656.2377661.

17. Emam, K.E. and Schneidewind, N.F. Methodology for Validating Software Product Metrics. 2002.

18. Curtis, B., Measurement and experimentation in software engineering. Proceedings of the IEEE, 1980. 68(9): p. 1144-1157.DOI: 10.1109/PROC.1980.11813.

19. Fenton, N.E. and Neil, M., Software metrics: roadmap, in Proceedings of the Conference on The Future of Software Engineering. 2000, Association for Computing Machinery: Limerick, Ireland. p. 357–370.

20. Jones, C., Software metrics: good, bad and missing. Computer, 1994. 27(9): p. 98-100.DOI: 10.1109/2.312055.

21. Cavano, J.P. and McCall, J.A., A framework for the measurement of software quality, in Proceedings of the software quality assurance workshop on Functional and performance issues. 1978, Association for Computing Machinery. p. 133–139.

22. Harman, M. and Clark, J. Metrics are fitness functions too. in 10th International Symposium on Software Metrics, 2004. Proceedings. 2004.DOI: 10.1109/METRIC.2004.1357891.

23. Roche, J.M., Software metrics and measurement principles. ACM SIGSOFT Software Engineering Notes, 1994. 19(1): p. 77–85.DOI: 10.1145/181610.181625.

24. Briand, L.C., Morasca, S., and Basili, V.R., Property-based software engineering measurement. IEEE Transactions on Software Engineering, 1996 :(1)22 .p. 68-86.DOI: 10.1109/32.481535.

25. Kitchenham, B., Pfleeger, S.L., and Fenton, N., Towards a framework for software measurement validation. IEEE Transactions on Software Engineering, 1995. 21(12): p. 929-944.DOI: 10.1109/32.489070.

26. Weyuker, E.J., On Testing Non-Testable Programs. The Computer Journal, 1982. 25(4): p. 465-470.DOI: 10.1093/comjnl/25.4.465.

الملحق(1) التحقق من مقياس PTI

اسم المقياس:	PTI	الخاصية:	السلامة الرياضية	نتيجة التحقق:	محقق
الرقم المميز	معيار التحقق	النتيجة	السبب		
11	زيادة النمو	محقق	ليكن لدينا two sprint backlogs ضمن Sprint I (التكرار): SBL1 يتألف من T_{PI_1} قصص مستخدم وSBL2 يتألف من T_{PI_2} قصص مستخدم مختلفين تماماً وبفرض k هي ثابت من أجل هذا التكرار وبالتالي ليس له تأثير على المعادلة. $SBL1 + SBL2 \Longrightarrow T_{PI_1} + T_{PI_2} = T_{PI} \wedge T_{SI_1} + T_{SI_2} = T_{SI}$ $PTI = \frac{T_{SI}}{T_{PI}} = \frac{T_{SI_1} + T_{SI_2}}{T_{PI_1} + T_{PI_2}}$ بفرض أن المقياس طبق على كل backlog على حدى بنفس sprint: $PTI_1 = \frac{T_{SI_1}}{T_{PI_1}}, PTI_2 = \frac{T_{SI_2}}{T_{PI_2}} \Longrightarrow PTI_1 + PTI_2 = \frac{T_{SI_1}}{T_{PI_1}} + \frac{T_{SI_2}}{T_{PI_2}} \Longrightarrow$ $\frac{T_{SI_1}}{T_{PI_1}} \geq \frac{T_{SI_1}}{T_{PI_1} + T_{PI_2}} \wedge \frac{T_{SI_2}}{T_{PI_2}} \geq \frac{T_{SI_2}}{T_{PI_1} + T_{PI_2}} \Longrightarrow PTI_1 + PTI_2 \geq PTI$	
12	حساسية التفاعل	محقق	ليكن لدينا ثلاثة sprint backlogs ضمن Sprint I: SBL1 يتألف من T_{PI_1} قصص مستخدم وSBL2 يتألف من T_{PI_2} قصص مستخدم مختلفين وSBL3 يتألف من T_{PI_3} قصص مستخدم مختلفين، وبفرض k هي ثابت من أجل هذا التكرار وبالتالي ليس له تأثير على المعادلة. $SBL1 + SBL3 \Longrightarrow T_{PI_1} + T_{PI_3} = T_{PI_{13}} \wedge T_{SI_1} + T_{SI_3} = T_{SI_{13}}$ $SBL2 + SBL3 \Longrightarrow T_{PI_2} + T_{PI_3} = T_{PI_{23}} \wedge T_{SI_2} + T_{SI_3} = T_{SI_{23}}$ $PTI_{13} = \frac{T_{SI_{13}}}{T_{PI_{13}}} = \frac{T_{SI_1} + T_{SI_3}}{T_{PI_1} + T_{PI_3}}$ $PTI_{23} = \frac{T_{SI_{23}}}{T_{PI_{23}}} = \frac{T_{SI_2} + T_{SI_3}}{T_{PI_2} + T_{PI_3}}$ بفرض أن المقياس طبق على كل backlog على حدى بنفس التكرار: $PTI_1 = \frac{T_{SI_1}}{T_{PI_1}}, PTI_2 = \frac{T_{SI_2}}{T_{PI_2}}, PTI_3 = \frac{T_{SI_3}}{T_{PI_3}}$ $\because PTI_1 = PTI_2 \Longrightarrow \frac{T_{SI_1} + T_{SI_3}}{T_{PI_1} + T_{PI_3}} \neq \frac{T_{SI_2} + T_{SI_3}}{T_{PI_2} + T_{PI_3}} \Longrightarrow PTI_{13} \neq PTI_{23}$	
13	الرتابة	محقق	ليكن لدينا two sprint backlogs ضمن Sprint I: SBL1 يتألف من T_{PI_1} قصص مستخدم وSBL2 يتألف من T_{PI_2} قصص مستخدم، وبفرض k هي ثابت من أجل هذا التكرار وبالتالي ليس له تأثير على المعادلة. $SBL1 \subset SBL2 \Longrightarrow T_{PI_1} \leq T_{PI_2} \wedge T_{SI_1} \leq T_{SI_2}$		

			بفرض أن المقياس طبق على كل backlog على حدى بنفس التكرار: $PTI_1 = \frac{T_{SI_1}}{T_{PI_1}}, PTI_2 = \frac{T_{SI_2}}{T_{PI_2}} \implies PTI_1 \leq PTI_2$
14	عدم الرسمية	محقق	ليكن لدينا two sprint backlogs ضمن Sprint I: SBL1 يتألف من T_{PI_1} قصص مستخدم و SBL2 يتألف من T_{PI_2} قصص مستخدم. $\exists SBL1, \exists SBL2 : SBL1 \not\equiv SBL2 \wedge T_{PI_1} \neq T_{PI_2} \wedge T_{SI_1} \neq T_{SI_2}$ بفرض أن المقياس طبق على كل backlogعلى حدى بنفس التكرار: $PTI_1 = \frac{T_{SI_1}}{T_{PI_1}}, PTI_2 = \frac{T_{SI_2}}{T_{PI_2}} \implies PTI_1 \neq PTI_2$
17	الاستمرارية	محقق	لا يعاني المقياس $PTI = k \times \frac{T_{SI}}{T_{PI}}$ من أي انقطاعات غير متوقعة في مجال القيم لأنه يقيس نسبة عدد قصص المستخدم المنجزة إلى عدد قصص المستخدم الكلية. الحالة الوحيدة التي يمكن أن يعاني منها هذه المقياس من انقطاع في القيم هي في حال $T_{PI} = 0$. وهذا لا يتحقق إلا في حال عدم احتواء Sprint Backlog أي قصة مستخدم بمعنى عدم وجود تكرار Sprint على الإطلاق.
18	اتساق الأبعاد	محقق	يحقق المقياس $PTI = k \times \frac{T_{SI}}{T_{PI}}$ هذا المعيار بحكم اعتماده على نسبة عدد إلى عدد مضروباً بنسبة عدد إلى عدد، والتي تفترض أن نسبة الإنجاز المحققة تعتمد على مدى استغلال كامل أعضاء الفريق المخصص لإنجاز هذه المهام.
19	تحقق المقدار	محقق	ليكن لدينا sprint backlog ضمن Sprint I مؤلفة من T_{PI_1} قصص مستخدم. $PTI = 0 \implies \frac{T_{SI}}{T_{PI}} = 0 \implies T_{SI} = 0$ أي في حال لم يتم إنجاز أي من قصص المستخدم فهذا يعني أن قيمة المقياس ستكون صفرية.

اسم المقياس:	PTI	الخاصية:	التطبيق العملي	نتيجة التحقق:	محقق
الرقم المميز	معيار التحقق	النتيجة	السبب		
1	الفعل	محقق	يحقق المقياس $PTI = k \times \frac{T_{SI}}{T_{PI}}$ هذا المعيار كونه يفيد مستخدمه في لحظة القياس معرفة نسبة إنجاز قصص المستخدم وبالتالي يعطي مؤشراً واضحاً لإمكانية الانتهاء من إنجاز كافة قصص المستخدم في الوقت المحدد من عدمه مما يمكن قائد الفريق من اتخاذ القرارات المناسبة اعتماداً على قيمة هذا المقياس.		
2	الإنتاجية الاقتصادية	محقق	يحقق المقياس $PTI = k \times \frac{T_{SI}}{T_{PI}}$ هذا المعيار حيث لا تعتبر تكلفة استخدام هذا المقياس تكلفة صغيرة تعتمد على عد مباشر لقصص المستخدم المنجزة بالنسبة إلى عدد قصص المستخدم الكلي مع عدد أعضاء الفريق العاملين في التكرار الواحد.		
5	عدم الاستغلال	محقق	يحقق المقياس $PTI = k \times \frac{T_{SI}}{T_{PI}}$ هذا المعيار حيث لا يمكن تطويع هذا المقياس للحصول على نتائج أفضل مما يعكسه الواقع لأنه يعتمد عد مباشر لقصص المستخدم المنجز نسبة إلى عدد قصص المستخدم الكلي مع عدد أعضاء الفريق العاملين في التكرار الواحد.		

التحقق النظري من مقاييس إنتاجية الفريق وتقدم المشروع في سياق ..... طراب

7	علاقة المقياس بالمنتج أو الإجراء	محقق	يحقق المقياس $PTI = k \times \frac{T_{SI}}{T_{PI}}$ هذا المعيار حيث تم تطويره لإجراءات التطوير الرشيقة ومع ذلك فإن المفهوم البسيط والمباشر الذي يعتمده يمكن تطويعه لإجراءات تطويرية أخرى.
9	العلاقة السببية	غير محقق	لا يحقق المقياس $PTI = k \times \frac{T_{SI}}{T_{PI}}$ هذا المعيار حيث لا توجد علاقة موضحة تربط بين إنتاجية قصص المستخدم وإنجازهم بنجاح مع جودة المنتج النهائي حيث كذلك الأمر يوجد عدم وضوح بتعريف الإنجاز الناجح لقصص المستخدم.

اسم المقياس:	PTI	الخاصية:	الصحة	نتيجة التحقق:	محقق
الرقم المميز	معيار التحقق	النتيجة	السبب		
4	وثوقية المقياس	محقق	يحقق المقياس $PTI = k \times \frac{T_{SI}}{T_{PI}}$ هذا المعيار حيث يمكن بإعادة عملية القياس في نفس اللحظة الزمنية الحصول على نفس نتيجة القياس بغض النظر عمن يجري عملية القياس بشرط تثبيت النسبة k في التكرار الواحد.		

اسم المقياس:	PTI	الخاصية:	الكشف عن الفروقات	نتيجة التحقق:	غير محقق
الرقم المميز	معيار التحقق	النتيجة	السبب		
6	ثبات التحويل	غير محقق	ليكن لدينا two sprint backlogs ضمن Sprint I: SBL1 يتألف من T_{PI_1} قصص مستخدم وSBL2 يتألف من T_{PI_2} قصص مستخدم، وبفرض k هي ثابت من أجل هذا التكرار وبالتالي ليس له تأثير على المعادلة. $\exists SBL1, \exists SBL2: (SBL1 \equiv SBL2) \wedge (T_{PI_1} \neq T_{PI_2} \vee T_{SI_1} \neq T_{SI_2})$ التكافؤ السياقي بين SBL1 و SBL2 هنا يعني أن كليهما يوصف نفس الوظائف عن طريق مجموعات مختلفة من قصص المستخدم. بفرض أن المقياس طبق على كل backlog على حدى بنفس التكرار: $PTI_1 = \frac{T_{SI_1}}{T_{PI_1}}, PTI_2 = \frac{T_{SI_2}}{T_{PI_2}} \nRightarrow PTI_1 = PTI_2$	
10	درجة التقسيم المناسبة	غير محقق	ليكن لدينا two sprint backlogs ضمن Sprint I: SBL1 يتألف من T_{PI_1} قصص مستخدم وSBL2 يتألف من T_{PI_2} قصص مستخدم، وبفرض k هي ثابت من أجل هذا التكرار وبالتالي ليس له تأثير على المعادلة. يمكن القول أن المقياس ليس شديد النعومة لأنه يوجد عدد كبير منته من backlogs المختلفين مع إمكانية الحصول على نفس نتيجة المقياس بينما يعتبر من المقاييس ذات التقسيم الخشن حيث يمكن ل two backlogs مختلفين تماماً أن يعطوا نفس نتيجة المقياس	

			$\exists SBL1, \exists SBL2: : SBL1 \not\equiv SBL2 \wedge T_{PI_1} = T_{PI_2} \wedge T_{SI_1} = T_{SI_2}$ على بفرض أن المقياس طبق على كل backlog على حدى بنفس التكرار: $PTI_1 = \frac{T_{SI_1}}{T_{PI_1}}, PTI_2 = \frac{T_{SI_2}}{T_{PI_2}} \Longrightarrow PTI_1 = PTI_2$
14	عدم الرسمية	محقق	ليكن لدينا two sprint backlogs ضمن Sprint I: SBL1 يتألف من T_{PI_1} قصص مستخدم و SBL2 يتألف من T_{PI_2} قصص مستخدم. $\exists SBL1, \exists SBL2: SBL1 \not\equiv SBL2 \wedge T_{PI_1} \neq T_{PI_2} \wedge T_{SI_1} \neq T_{SI_2}$ بفرض أن المقياس طبق على كل backlogعلى حدى بنفس التكرار: $PTI_1 = \frac{T_{SI_1}}{T_{PI_1}}, PTI_2 = \frac{T_{SI_2}}{T_{PI_2}} \Longrightarrow PTI_1 \neq PTI_2$

اسم المقياس:	PTI	الخاصية:	تعزيز الفرضيات	نتيجة التحقق:	محقق
الرقم المميز	معيار التحقق	النتيجة	السبب		
8	استقلالية العامل	محقق	يحقق المقياس $PTI = k \times \frac{T_{SI}}{T_{PI}}$ هذه الخاصية لأن القياسات الفردية ستكون مستقلة عن بعضها بحكم أن عدد قصص المستخدم المنجزة T_{SI} ستختلف مع مرور الوقت حكماً طالما المشروع مازال قائماً.		

اسم المقياس:	PTI	الخاصية:	الجدوى	نتيجة التحقق:	محقق
الرقم المميز	معيار التحقق	النتيجة	السبب		
3	الاتساق الداخلي	محقق	يستهدف المقياس $PTI = k \times \frac{T_{SI}}{T_{PI}}$ عدد قصص المستخدم المنجزة ضمن تكرار محدد نسبة إلى عدد قصص المستخدم الكلية في SBL وبالتالي يستهدف نفس الوحدة (كعدد) وهي متكاملة مع بعضها بطريقة منضبطة للحصول على قيمة المقياس النهائية.		
8	استقلالية العامل	محقق	يحقق المقياس $PTI = k \times \frac{T_{SI}}{T_{PI}}$ هذه الخاصية لأن القياسات الفردية ستكون مستقلة عن بعضها بحكم أن عدد قصص المستخدم المنجزة T_{SI} ستختلف مع مرور الوقت حكماً طالما المشروع مازال قائماً.		
14	عدم الرسمية	محقق	ليكن لدينا two sprint backlogs ضمن Sprint I: SBL1 يتألف من T_{PI_1} قصص مستخدم و SBL2 يتألف من T_{PI_2} قصص مستخدم. $\exists SBL1, \exists SBL2: SBL1 \not\equiv SBL2 \wedge T_{PI_1} \neq T_{PI_2} \wedge T_{SI_1} \neq T_{SI_2}$ بفرض أن المقياس طبق على كل backlogعلى حدى بنفس التكرار: $PTI_1 = \frac{T_{SI_1}}{T_{PI_1}}, PTI_2 = \frac{T_{SI_2}}{T_{PI_2}} \Longrightarrow PTI_1 \neq PTI_2$		
15	تحقق التعلق	غير محقق	لا يعكس المقياس $PTI = k \times \frac{T_{SI}}{T_{PI}}$ تسلسل إنجاز قصص المستخدم. بفرض لدينا sprint backlog ضمن Sprint I: SBL مؤلفة من T_{PI} قصص مستخدم. في حالة الاختبار الأولى، تم إنجاز T_{SI} قصة مستخدم.		

			في حالة الاختبار الثانية، تم إنجاز T_{SI} قصة مستخدم، ولكن الاختلاف يكمن بين الحالتين في القصص نفسها. لا يستطيع المقياس أن يميز بين الحالتين بالرغم من اختلاف القصص.
16	عدم حساسية تعديل الاسم	محقق	يحقق المقياس $PTI = k \times \frac{T_{SI}}{T_{PI}}$ هذه الخاصة بحكم اعتماده على نسبة عدد قصص المستخدم المنجزة إلى عدد قصص المستخدم الكلية الموجودة في SBL ولا تتأثر قيمة المقياس بتعديل اسم قصة المستخدم المنجزة طالما أنها أنجزت.
17	الاستمرارية	محقق	لا يعاني المقياس $PTI = k \times \frac{T_{SI}}{T_{PI}}$ من أي انقطاعات غير متوقعة في مجال القيم لأنه يقيس نسبة عدد قصص المستخدم المنجزة إلى عدد قصص المستخدم الكلية. الحالة الوحيدة التي يمكن أن يعاني منها هذه المقياس من انقطاع في القيم هي في حال $T_{PI} = 0$. وهذا لا يتحقق إلا في حال عدم احتواء Sprint Backlog بمعنى عدم وجود تكرار Sprint على الإطلاق.
18	اتساق الأبعاد	محقق	يحقق المقياس $PTI = k \times \frac{T_{SI}}{T_{PI}}$ هذه الخاصة بحكم اعتماده على نسبة عدد إلى عدد مضروباً بنسبة عدد إلى عدد، والتي تفترض أن نسبة الإنجاز المحققة تعتمد على مدى استغلال كامل أعضاء الفريق المخصص لإنجاز هذه المهام.
20	تحقق الوحدة	محقق	يحقق المقياس $PTI = k \times \frac{T_{SI}}{T_{PI}}$ على الرغم أن المقياس هو نسبة مئوية لا واحدة لها ناتجة عن ضرب نسبتين ببعضهما البعض ولكنه من خلال حساب هذه النسبة يمكن معرفة عدد قصص المستخدم المنجزة نسبة إلى عدد قصص المستخدم الكلي في التكرار الواحد في لحظة القياس وبالتالي معرفة الإنتاجية.

اسم المقياس:	PTI	الخاصية:	إعلام القرار	نتيجة التحقق:	محقق
الرقم المميز	معيار التحقق	النتيجة	السبب		
1	الفعل	محقق	يحقق المقياس $PTI = k \times \frac{T_{SI}}{T_{PI}}$ هذا المعيار كونه يفيد مستخدمه في لحظة القياس معرفة نسبة إنجاز قصص المستخدم وبالتالي يعطي مؤشراً واضحاً لإمكانية الانتهاء من إنجاز كافة قصص المستخدم في الوقت المحدد من عدمه مما يمكن قائد الفريق من اتخاذ القرارات المناسبة اعتماداً على قيمة هذا المقياس.		
2	الإنتاجية الاقتصادية	محقق	يحقق المقياس $PTI = \frac{T_{SI}}{T_{PI}}$ هذا المعيار حيث لا تعتبر تكلفة استخدام هذا المقياس تكلفة صغيرة تعتمد على عد مباشر لقصص المستخدم المنجزة بالنسبة إلى عدد قصص المستخدم الكلي مع عدد أعضاء الفريق العاملين في التكرار الواحد.		
9	العلاقة السببية	غير محقق	لا يحقق المقياس $PTI = \frac{T_{SI}}{T_{PI}}$ هذا المعيار حيث لا توجد علاقة موضحة تربط بين إنتاجية قصص المستخدم وإنجازهم بنجاح مع جودة المنتج النهائي حيث كذلك الأمر يوجد عدم وضوح بتعريف الإنجاز الناجح لقصص المستخدم مع عدد أعضاء الفريق العاملين في التكرار الواحد.		

اسم المقياس:	PTI	الخاصية:	بناء النظرية	نتيجة التحقق:	غير محقق
الرقم المميز	معيار التحقق	النتيجة	السبب		
9	العلاقة السببية	غير محقق	لا يحقق المقياس $PTI = \frac{T_{SI}}{T_{PI}}$ هذا المعيار حيث لا توجد علاقة موضحة تربط بين إنتاجية قصص المستخدم وإنجازهم بنجاح مع جودة المنتج النهائي حيث كذلك الأمر يوجد عدم		

التحقق النظري من مقاييس إنتاجية الفريق وتقدم المشروع في سياق ..... طراب

			وضوح بتعريف الإنجاز الناجح لقصص المستخدم مع عدد أعضاء الفريق العاملين في التكرار الواحد.

اسم المقياس:	PTI	الخاصية:	مساهمة توافقية	نتيجة التحقق:	محقق
الرقم المميز	معيار التحقق	النتيجة	السبب		
11	زيادة النمو	محقق	ليكن لدينا two sprint backlogs ضمن Sprint I (التكرار): SBL1 يتألف من T_{PI_1} قصص مستخدم وSBL2 يتألف من T_{PI_2} قصص مستخدم مختلفين تماماً وبفرض k هي ثابت من أجل هذا التكرار وبالتالي ليس له تأثير على المعادلة. $SBL1 + SBL2 \Longrightarrow T_{PI_1} + T_{PI_2} = T_{PI} \wedge T_{SI_1} + T_{SI_2} = T_{SI}$ $PTI = \frac{T_{SI}}{T_{PI}} = \frac{T_{SI_1} + T_{SI_2}}{T_{PI_1} + T_{PI_2}}$ بفرض أن المقياس طبق على كل backlog على حدى بنفس sprint: $PTI_1 = \frac{T_{SI_1}}{T_{PI_1}}, PTI_2 = \frac{T_{SI_2}}{T_{PI_2}} \Longrightarrow PTI_1 + PTI_2 = \frac{T_{SI_1}}{T_{PI_1}} + \frac{T_{SI_2}}{T_{PI_2}} \Longrightarrow$ $\frac{T_{SI_1}}{T_{PI_1}} \geq \frac{T_{SI_1}}{T_{PI_1} + T_{PI_2}} \wedge \frac{T_{SI_2}}{T_{PI_2}} \geq \frac{T_{SI_2}}{T_{PI_1} + T_{PI_2}} \Longrightarrow PTI_1 + PTI_2 \geq PTI$
12	حساسية التفاعل	محقق	ليكن لدينا ثلاثة sprint backlogs ضمن Sprint I: SBL1 يتألف من T_{PI_1} قصص مستخدم وSBL2 يتألف من T_{PI_2} قصص مستخدم مختلفين وSBL3 يتألف من T_{PI_3} قصص مستخدم مختلفين، وبفرض k هي ثابت من أجل هذا التكرار وبالتالي ليس له تأثير على المعادلة. $SBL1 + SBL3 \Longrightarrow T_{PI_1} + T_{PI_3} = T_{PI_{13}} \wedge T_{SI_1} + T_{SI_3} = T_{SI_{13}}$ $SBL2 + SBL3 \Longrightarrow T_{PI_2} + T_{PI_3} = T_{PI_{23}} \wedge T_{SI_2} + T_{SI_3} = T_{SI_{23}}$ $PTI_{13} = \frac{T_{SI_{13}}}{T_{PI_{13}}} = \frac{T_{SI_1} + T_{SI_3}}{T_{PI_1} + T_{PI_3}}$ $PTI_{23} = \frac{T_{SI_{23}}}{T_{PI_{23}}} = \frac{T_{SI_2} + T_{SI_3}}{T_{PI_2} + T_{PI_3}}$ بفرض أن المقياس طبق على كل backlog على حدى بنفس التكرار: $PTI_1 = \frac{T_{SI_1}}{T_{PI_1}}, PTI_2 = \frac{T_{SI_2}}{T_{PI_2}}, PTI_3 = \frac{T_{SI_3}}{T_{PI_3}}$ $\because PTI_1 = PTI_2 \Longrightarrow \frac{T_{SI_1} + T_{SI_3}}{T_{PI_1} + T_{PI_3}} \neq \frac{T_{SI_2} + T_{SI_3}}{T_{PI_2} + T_{PI_3}} \Longrightarrow PTI_{13} \neq PTI_{23}$
17	الاستمرارية	محقق	لا يعاني المقياس $PTI = k \times \frac{T_{SI}}{T_{PI}}$ من أي انقطاعات غير متوقعة في مجال القيم لأنه يقيس نسبة عدد قصص المستخدم المنجزة إلى عدد قصص المستخدم الكلية. الحالة الوحيدة التي يمكن أن يعاني منها هذه المقياس من انقطاع في القيم هي في حال $T_{PI} = 0$. وهذا لا يتحقق إلا في حال عدم احتواء Sprint Backlog بمعنى عدم وجود تكرار Sprint على الإطلاق.	

الملحق(2) التحقق من مقياس IPWI

اسم المقياس:	IPWI	الخاصية:	السلامة الرياضية	نتيجة التحقق:	محقق
الرقم المميز	معيار التحقق	النتيجة	السبب		
11	زيادة النمو	محقق	ليكن لدينا two sprint backlogs ضمن Sprint I (التكرار): SBLa يتألف من n قصص مستخدم وSBL$_b$ يتألف من m قصص مستخدم. بضم كلا backlogs سينتج لدينا SBL: $SBL = SBL_a + SBL_b$ يتألف من $o \leq n + m$ قصص مستخدم. عند تطبيق المقياس على كل من backlog ينتج لدينا مايلي: $IPWI_a = \sum_{i=1}^{n} (\frac{L_{i_1} - L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0} - L_{Ei_0}}{L_{i_0}})$ $IPWI_b = \sum_{j=1}^{m} (\frac{L_{j_1} - L_{Ei_1}}{L_{j_1}} - \frac{L_{j_0} - L_{Ej_0}}{L_{j_0}})$ $IPWI = \sum_{k=1}^{o} (\frac{L_{k_1} - L_{Ek_1}}{L_{k_1}} - \frac{L_{k_0} - L_{Ek_0}}{L_{k_0}})$ $IPWI_b + IPWI_a = \sum_{i=1}^{n} (\frac{L_{i_1} - L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0} - L_{Ei_0}}{L_{i_0}}) + \sum_{j=1}^{m} (\frac{L_{j_1} - L_{Ei_1}}{L_{j_1}} - \frac{L_{j_0} - L_{Ej_0}}{L_{j_0}}) \Longrightarrow$ $IPWI_b + IPWI_a = \sum_{i=1}^{n} (\frac{L_{i_1} - L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0} - L_{Ei_0}}{L_{i_0}}) + \sum_{i=n+1}^{n+m} (\frac{L_{j_1} - L_{Ei_1}}{L_{j_1}} - \frac{L_{j_0} - L_{Ej_0}}{L_{j_0}}) \Longrightarrow$ $IPWI_b + IPWI_a = \sum_{i=1}^{n+m} (\frac{L_{i_1} - L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0} - L_{Ei_0}}{L_{i_0}})$ $\because n + m \geq o \ \wedge i = k \ as \ starting \ point \Longrightarrow$ $\sum_{i=1}^{n+m} (\frac{L_{i_1} - L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0} - L_{Ei_0}}{L_{i_0}}) \geq \sum_{i=1}^{o} (\frac{L_{i_1} - L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0} - L_{Ei_0}}{L_{i_0}})$ $IPWI_b + IPWI_a \geq IPWI$			

12	حساسية التفاعل	غير محقق	ليكن لدينا ثلاثة sprint backlogs ضمن Sprint I: SBL1 يتألف من n قصص مستخدم و SBL2 يتألف من m قصص مستخدم و SBL3 يتألف من o قصص مستخدم، $SBL13 = SBL1 + SBL3 \Longrightarrow Size(SBL13) = s_{13} \leq n + o$ $SBL23 = SBL2 + SBL3 \Longrightarrow Size(SBL23) = s_{23} \leq m + o$ $IPWI_1 = \sum_{i=1}^{n} (\frac{L_{i_1} - L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0} - L_{Ei_0}}{L_{i_0}})$ $IPWI_2 = \sum_{j=1}^{m} (\frac{L_{j_1} - L_{Ei_1}}{L_{j_1}} - \frac{L_{j_0} - L_{Ej_0}}{L_{j_0}})$ $IPWI_{13} = \sum_{i=1}^{s_{13}} (\frac{L_{i_1} - L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0} - L_{Ei_0}}{L_{i_0}})$ $IPWI_{23} = \sum_{j=1}^{s_{23}} (\frac{L_{j_1} - L_{Ei_1}}{L_{j_1}} - \frac{L_{j_0} - L_{Ej_0}}{L_{j_0}})$ $IPWI_{13} \leq IPWI_1 + IPWI_3, IPWI_{23} \leq IPWI_2 + IPWI_3$ $\because IPWI_1 = IPWI_2 \nRightarrow IPWI_{13} \neq IPWI_{23}$
13	الرتابة	محقق	ليكن لدينا two sprint backlogs ضمن Sprint I: SBL1 يتألف من n قصص مستخدم و SBL2 يتألف من m قصص مستخدم، $SBL1 \subset SBL2 \Longrightarrow n \leq m$ وقصص مستخدم $SBL1$ متضمنة حكماً في $SBL2$ بفرض أن المقياس طبق على كل backlog على حدى بنفس التكرار: $IPWI_1 = \sum_{i=1}^{n} (\frac{L_{i_1} - L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0} - L_{Ei_0}}{L_{i_0}})$ $IPWI_2 = \sum_{j=1}^{m} (\frac{L_{j_1} - L_{Ei_1}}{L_{j_1}} - \frac{L_{j_0} - L_{Ej_0}}{L_{j_0}})$ $\because n \leq m \Rightarrow IPWI_1 \leq IPWI_2$
14	عدم الرسمية	محقق	ليكن لدينا two sprint backlogs ضمن Sprint I: SBL1 يتألف من n قصص مستخدم و SBL2 يتألف من m قصص مستخدم> $\exists SBL1, \exists SBL2: SBL1 \not\equiv SBL2 \wedge n \neq m$ وقصص مستخدم $SBL1$ وقصص مستخدم$SBL2$ هي قصص مختلفة. بفرض أن المقياس طبق على كل backlog على حدى بنفس التكرار: $IPWI_1 = \sum_{i=1}^{n} (\frac{L_{i_1} - L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0} - L_{Ei_0}}{L_{i_0}})$ $IPWI_2 = \sum_{j=1}^{m} (\frac{L_{j_1} - L_{Ei_1}}{L_{j_1}} - \frac{L_{j_0} - L_{Ej_0}}{L_{j_0}})$ $\Longrightarrow IPWI_1 \neq IPWI_2$

17	الاستمرارية	غير محقق	يعاني المقياس $IPWI = \sum_{i=1}^{n}(\frac{L_{i_1}-L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0}-L_{Ei_0}}{L_{i_0}})$ من انقطاعين في الحساب: $L_{i_1} = 0 \vee L_{i_0} = 0$ يمكن لإحدى هذه الحالات من الحدوث في حال كون إحدى قصص المستخدم لم تنجز بعد عند القياس.
18	اتساق الأبعاد	محقق	يحقق المقياس $IPWI = \sum_{i=1}^{n}(\frac{L_{i_1}-L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0}-L_{Ei_0}}{L_{i_0}})$ هذا المعيار بحكم اعتماده على حساب مجموع نسب مئوية لأعداد أسطر الكود المستخدم في إنجاز قصص المستخدم.
19	تحقق المقدار	غير محقق	لا يمكن التأكد أن جميع التحويلات المنطقية يمكن تطبيقها على هذا المقياس. بما أن هذا المقياس يعبر عن مجموع نسب مئوية فإن النتيجة الكلية لهذا المقياس لا يمكن تعبر بدقة عن مدى التطور الذي تحقق وكذلك التعديلات في قيمه لا يمكن تفسيرها أيضاً بطريقة لها معنى. فعلى سبيل المثال إن احتوت جميع أسطر الكود المستخدم في تنفيذ قصة مستخدم 'i'على أخطاء بعد إجراء عمليات التصحيح: $L_{j_1} = L_{Ei_1}$ وبعد إجراء عمليات التصحيح، مازال هناك عدداً من الأسطر تحتوي أخطاء: $L_{j_0} > L_{Ej_0}$ عندئذ ستكون نتيجة المقياس $IPWI_i$من أجل قصة المستخدم هذه سالبة. من المفترض فهم القيمة السالبة أن تراجعاً في التقدم قد حصل ولكن مقدار هذا التراجع وتأثيره على نتيجة التقدم الكلية لن تكون بالضرورة لها معنى.

اسم المقياس:	IPWI	الخاصية:	التطبيق العملي	نتيجة التحقق:	غير محقق
الرقم المميز	معيار التحقق	النتيجة	السبب		
1	الفعل	غير محقق	يعتمد المقياس $IPWI = \sum_{i=1}^{n}(\frac{L_{i_1}-L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0}-L_{Ei_0}}{L_{i_0}})$ على مجموع النسب المئوية لزيادات صحة الكود لكل من قصص المستخدم المنجزة في التكرار المحدد. وبحكم أن نتيجة المجموع متعلق بعدد قصص المستخدم الموجودة في التكرار فلا يوجد أي حد أعلى معلوم لهذا المجموع وبالتالي ستفقد النتيجة أي ترجمة أو معنى لقيمتها وذلك بسبب عدم وجود وحدة لها تفسر نتيجتها أو عدم كون النتيجة نسبة مئوية. نتيجة لذلك لا يمكن عكس هذه النتيجة على اتخاذ أي قرار واضح لقائد الفريق.	
2	الإنتاجية الاقتصادية	غير محقق	يعتمد المقياس $IPWI = \sum_{i=1}^{n}(\frac{L_{i_1}-L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0}-L_{Ei_0}}{L_{i_0}})$ على مجموع النسب المئوية لزيادات صحة الكود لكل من قصص المستخدم المنجزة في التكرار المحدد. يتطلب ذلك قياس عدد أسطر الكود الكلي لكل قصة مستخدم قبل التصحيح وبعده كما يتطلب قياس عدد أسطر الكود التي تحتوي أخطاء لكل قصة مستخدم قبل التصحيح وبعده. عملية قياس عدد أسطر الكود التي تحتوي أخطاء لقصة مستخدم واحدة ليست بالعملية المباشرة حيث يجب أن يتم إجراء تحليل سكوني للكود أو اختبار الوحدة وهذا يتطلب جهداً ووقتاً من قبل المطور أو الفاحص قبل إمكانية تحديد الأسطر التي تحتوي أخطاء ومن ثم عدها. سيتم تكرار هذه العملية من أجل كل قصة مستخدم قبل المباشرة بعمليات التصحيح وبعدها وهذا يعتبر عملاً مجهداً ويتطلب وقتاً وتكلفة.	
5	عدم الاستغلال	غير محقق	يعتمد المقياس $IPWI = \sum_{i=1}^{n}(\frac{L_{i_1}-L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0}-L_{Ei_0}}{L_{i_0}})$ على مجموع النسب المئوية لزيادات صحة الكود لكل من قصص المستخدم المنجزة في التكرار المحدد. يتطلب ذلك قياس	

			عدد أسطر الكود الكلي لكل قصة مستخدم قبل التصحيح وبعده كما يتطلب قياس عدد أسطر الكود التي تحتوي أخطاء لكل قصة مستخدم قبل التصحيح وبعده. عملية قياس عدد أسطر الكود التي تحتوي أخطاء لقصة مستخدم واحدة ليست بالعملية المباشرة ويمكن أن تحتمل أخطاء حسب خبرة من يقوم بهذه العملية. تزيد قيمة هذا المقياس كلما كانت درجة التصحيح أعلى وبالتالي يمكن للمطور ادعاء وجود أخطاء ومن ثم تصحيحها أي إعطاء عملية تقدم وهمية.
7	علاقة المقياس بالمنتج أو الإجراء	محقق	يحقق المقياس $IPWI = \sum_{i=1}^{n}(\frac{L_{i_1}-L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0}-L_{Ei_0}}{L_{i_0}})$ هذا المعيار حيث تم تطويره لإجراءات التطوير الرشيقة ومع ذلك فإن المفهوم البسيط والمباشر الذي يعتمده يمكن تطويعه لإجراءات تطويرية أخرى.
9	العلاقة السببية	غير محقق	لا يحقق المقياس $IPWI = \sum_{i=1}^{n}(\frac{L_{i_1}-L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0}-L_{Ei_0}}{L_{i_0}})$ هذا المعيار حيث لا توجد علاقة موضحة تربط بين نتيجة هذا المقياس مع جودة المنتج النهائي.

اسم المقياس:	IPWI	الخاصية:	الصحة	نتيجة التحقق:	غير محقق
الرقم المميز	معيار التحقق	النتيجة	السبب		
4	وثوقية المقياس	غير محقق	لا يحقق المقياس $IPWI = \sum_{i=1}^{n}(\frac{L_{i_1}-L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0}-L_{Ei_0}}{L_{i_0}})$ هذا المعيار حيث لا يمكن ضمان الحصول على نفس النتيجة إذا تمت إعادة عملية القياس من قبل شخصين مختلفين بسبب عدم ثبات قياس عدد أسطر الكود التي تحتوي أخطاء من أجل قصة مستخدم ما.		

اسم المقياس:	IPWI	الخاصية:	الكشف عن الفروقات	نتيجة التحقق:	غير محقق
الرقم المميز	معيار التحقق	النتيجة	السبب		
6	ثبات التحويل	غير محقق	ليكن لدينا two sprint backlogs ضمن Sprint I: SBL1 يتألف من n قصص مستخدم و SBL2 يتألف من m قصص مستخدم $\exists SBL1, \exists SBL2: SBL1 \equiv SBL2 \wedge n \neq m$ التكافؤ السياقي بين SBL1 و SBL2 هنا يعني أن كليهما يوصف نفس الوظائف عن طريق مجموعات مختلفة من قصص المستخدم. بفرض أن المقياس طبق على كل backlog على حدى بنفس التكرار: $IPWI_1 = \sum_{i=1}^{n}(\frac{L_{i_1} - L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0} - L_{Ei_0}}{L_{i_0}})$ $IPWI_2 = \sum_{j=1}^{m}(\frac{L_{j_1} - L_{Ei_1}}{L_{j_1}} - \frac{L_{j_0} - L_{Ej_0}}{L_{j_0}})$ $\nRightarrow IPWI_1 = IPWI_2$	
10	درجة التقسيم المناسبة	غير محقق	ليكن لدينا two sprint backlogs ضمن Sprint I: SBL1 يتألف من n قصص مستخدم و SBL2 يتألف من m قصص مستخدم. يمكن القول أن المقياس شديد النعومة لأنه نتيجة المقياس سوف تختلف باختلاف عدد قصص المستخدم المنفذة وعدد اسطر الكود المنجزة لكل قصة مستخدم وعدد الأخطاء		

			المكتشفة مع الأخذ بعين الاعتبار مدى التقدم في الإنجاز لحظة القياس. بفرض أن المقياس طبق على كل backlog على حدى بنفس التكرار: $IPWI_1 = \sum_{i=1}^{n}(\frac{L_{i_1} - L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0} - L_{Ei_0}}{L_{i_0}})$ $IPWI_2 = \sum_{j=1}^{m}(\frac{L_{j_1} - L_{Ei_1}}{L_{j_1}} - \frac{L_{j_0} - L_{Ej_0}}{L_{j_0}})$ $\Longrightarrow IPWI_1 \neq IPWI_2$
14	عدم الرسمية	محقق	ليكن لدينا two sprint backlogs ضمن Sprint I: SBL1 يتألف من n قصص مستخدم و SBL2 يتألف من m قصص مستخدم $\exists SBL1, \exists SBL2: SBL1 \not\equiv SBL2 \wedge n \neq m$ وقصص مستخدم $SBL1$ وقصص مستخدم $SBL2$ هي قصص مختلفة. بفرض أن المقياس طبق على كل backlog على حدى بنفس التكرار: $IPWI_1 = \sum_{i=1}^{n}(\frac{L_{i_1} - L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0} - L_{Ei_0}}{L_{i_0}})$ $IPWI_2 = \sum_{j=1}^{m}(\frac{L_{j_1} - L_{Ei_1}}{L_{j_1}} - \frac{L_{j_0} - L_{Ej_0}}{L_{j_0}})$ $\Longrightarrow IPWI_1 \neq IPWI_2$

اسم المقياس:	IPWI	الخاصية:	تعزيز الفرضية	نتيجة التحقق:	محقق
الرقم المميز	معيار التحقق	النتيجة	السبب		
8	استقلالية العامل	محقق	لتحقيق هذا المعيار، يجب على المقياس أن يقوم بإجراءات قياس مستقلة للعوامل الداخلة في حسابه أي عدد أسطر الكود L_i الكلي وعدد أسطر الكود التي تحتوي أخطاء L_{Ei} من أجل كل قصة مستخدم i سواء قبل إجراء عمليات التصحيح أو بعده. إن إجراء عمليات قياس عدد أسطر الكود هي قياسات فردية مباشرة غير متعلقة ببعضها وبالتالي يعتبر هذا المعيار محققاً في حالة هذا المقياس $IPWI = \sum_{i=1}^{n}(\frac{L_{i_1} - L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0} - L_{Ei_0}}{L_{i_0}})$		

اسم المقياس:	IPWI	الخاصية:	الجدوى	نتيجة التحقق:	محقق
الرقم المميز	معيار التحقق	النتيجة	السبب		
3	الاتساق الداخلي	محقق	يعتمد المقياس $IPWI = \sum_{i=1}^{n}(\frac{L_{i_1} - L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0} - L_{Ei_0}}{L_{i_0}})$ على حساب مجموع زيادات صحة أكواد قصص المستخدم المنجزة ضمن تكرار محدد وطالما النسبة تعتمد في حساب عدد أسطر الكود سواء قبل التصحيح أو بعده بالتالي يستهدف نفس الوحدة (كعدد) وهي متكاملة مع بعضها بطريقة منضبطة داخلياً للحصول على قيمة المقياس النهائية.		

8	استقلالية العامل	محقق	لتحقيق هذا المعيار، يجب على المقياس أن يقوم بإجراءات قياس مستقلة للعوامل الداخلة في حسابه أي عدد أسطر الكود L_i الكلي وعدد أسطر الكود التي تحتوي أخطاء L_{Ei} من أجل كل قصة مستخدم i سواء قبل إجراء عمليات التصحيح أو بعده. إن إجراء عمليات قياس عدد أسطر الكود هي قياسات فردية مباشرة غير متعلقة ببعضها وبالتالي يعتبر هذا المعيار محققاً في حالة هذا المقياس $IPWI = \sum_{i=1}^{n}(\frac{L_{i_1}-L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0}-L_{Ei_0}}{L_{i_0}})$
14	عدم الرسمية	محقق	ليكن لدينا two sprint backlogs ضمن Sprint I: SBL1 يتألف من n قصص مستخدم و SBL2 يتألف من m قصص مستخدم $\exists SBL1, \exists SBL2: SBL1 \not\equiv SBL2 \wedge n \neq m$ وقصص مستخدم $SBL1$ وقصص مستخدم$SBL2$ هي قصص مختلفة. بفرض أن المقياس طبق على كل backlog على حدى بنفس التكرار: $IPWI_1 = \sum_{i=1}^{n}(\frac{L_{i_1} - L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0} - L_{Ei_0}}{L_{i_0}})$ $IPWI_2 = \sum_{j=1}^{m}(\frac{L_{j_1} - L_{Ei_1}}{L_{j_1}} - \frac{L_{j_0} - L_{Ej_0}}{L_{j_0}})$ $\Longrightarrow IPWI_1 \neq IPWI_2$
15	تحقق التعلق	غير محقق	إن العامل الأساسي الذي يؤثر على نتيجة المقياس هو قياس عدد الأسطر وليس تسلسل ترتيبها. وبالتالي تعديل ترتيب أسطر الكود لإنجاز قصص المستخدم دون أن يؤدي ذلك إلى تعديل عدد الأسطر لن يؤدي بالضرورة إلى أي تغيير في نتيجة المقياس. من هذا المنطلق يعتبر هذا المعيار غير محقق في حالة هذا المقياس $IPWI = \sum_{i=1}^{n}(\frac{L_{i_1}-L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0}-L_{Ei_0}}{L_{i_0}})$
16	عدم حساسية تعديل الاسم	محقق	يحقق المقياس $IPWI = \sum_{i=1}^{n}(\frac{L_{i_1}-L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0}-L_{Ei_0}}{L_{i_0}})$ هذا المعيار بحكم اعتماد المقياس على عدد أسطر الكود لإنجاز قصص المستخدم ولن يتأثر المقياس سواء بتغيير اسم قصة المستخدم أو بتعديل أسماء المحددات الداخلية لتنفيذ هذه القصص من أسماء متحولات أو دالات وغيرها.
17	الاستمرارية	غير محقق	يعاني المقياس$IPWI = \sum_{i=1}^{n}(\frac{L_{i_1}-L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0}-L_{Ei_0}}{L_{i_0}})$ من انقطاعين في الحساب: $L_{i_1} = 0 \vee L_{i_0} = 0$ يمكن لإحدى هذه الحالات من الحدوث في حال كون إحدى قصص المستخدم لم تنجز بعد عند القياس.
18	اتساق الأبعاد	محقق	يحقق المقياس$IPWI = \sum_{i=1}^{n}(\frac{L_{i_1}-L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0}-L_{Ei_0}}{L_{i_0}})$ هذا المعيار بحكم اعتماده على حساب مجموع نسب مئوية لأعداد أسطر الكود المستخدم في إنجاز قصص المستخدم.
20	تحقق الوحدة	غير محقق	يعتمد المقياس $IPWI = \sum_{i=1}^{n}(\frac{L_{i_1}-L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0}-L_{Ei_0}}{L_{i_0}})$ على مجموع النسب المئوية لزيادات صحة الكود لكل من قصص المستخدم المنجزة في التكرار المحدد. وبحكم أن نتيجة المجموع متعلق بعدد قصص المستخدم الموجودة في التكرار فلا يوجد أي حد أعلى معلوم لهذا المجموع وبالتالي ستفقد النتيجة أي ترجمة أو معنى لقيمتها وذلك بسبب عدم وجود وحدة لها تفسر نتيجتها أو عدم كون النتيجة نسبة مئوية.

اسم المقياس:	IPWI	الخاصية:	إعلام القرار	نتيجة التحقق:	غير محقق
الرقم المميز	معيار التحقق	النتيجة	السبب		
1	الفعل	غير محقق	يعتمد المقياس $IPWI = \sum_{i=1}^{n}(\frac{L_{i_1}-L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0}-L_{Ei_0}}{L_{i_0}})$ على مجموع النسب المئوية لزيادات صحة الكود لكل من قصص المستخدم المنجزة في التكرار المحدد. وبحكم أن نتيجة المجموع متعلق بعدد قصص المستخدم الموجودة في التكرار فلا يوجد أي حد أعلى معلوم لهذا المجموع وبالتالي ستفقد النتيجة أي ترجمة أو معنى لقيمتها وذلك بسبب عدم وجود وحدة لها تفسر نتيجتها أو عدم كون النتيجة نسبة مئوية. نتيجة لذلك لا يمكن عكس هذه النتيجة على اتخاذ أي قرار واضح لقائد الفريق.		
2	الإنتاجية الاقتصادية	غير محقق	يعتمد المقياس $IPWI = \sum_{i=1}^{n}(\frac{L_{i_1}-L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0}-L_{Ei_0}}{L_{i_0}})$ على مجموع النسب المئوية لزيادات صحة الكود لكل من قصص المستخدم المنجزة في التكرار المحدد. يتطلب ذلك قياس عدد أسطر الكود الكلي لكل قصة مستخدم قبل التصحيح وبعده كما يتطلب قياس عدد أسطر الكود التي تحتوي أخطاء لكل قصة مستخدم قبل التصحيح وبعده. عملية قياس عدد أسطر الكود التي تحتوي أخطاء لقصة مستخدم واحدة ليست بالعملية المباشرة حيث يجب أن يتم إجراء تحليل سكوني للكود أو اختبار الوحدة وهذا يتطلب جهداً ووقتاً من قبل المطور أو الفاحص قبل إمكانية تحديد الأسطر التي تحتوي أخطاء ومن ثم عدها. سيتم تكرار هذه العملية من أجل كل قصة مستخدم قبل المباشرة بعمليات التصحيح وبعدها وهذا يعتبر عملاً مجهداً ويتطلب وقتاً وتكلفة.		
9	العلاقة السببية	غير محقق	لا يحقق المقياس $IPWI = \sum_{i=1}^{n}(\frac{L_{i_1}-L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0}-L_{Ei_0}}{L_{i_0}})$ هذا المعيار حيث لا توجد علاقة موضحة تربط بين نتيجة هذا المقياس مع جودة المنتج النهائي.		

اسم المقياس:	IPWI	الخاصية:	بناء النظرية	نتيجة التحقق:	غير محقق
الرقم المميز	معيار التحقق	النتيجة	السبب		
9	العلاقة السببية	غير محقق	لا يحقق المقياس $IPWI = \sum_{i=1}^{n}(\frac{L_{i_1}-L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0}-L_{Ei_0}}{L_{i_0}})$ هذا المعيار حيث لا توجد علاقة موضحة تربط بين نتيجة هذا المقياس مع جودة المنتج النهائي.		

اسم المقياس:	IPWI	الخاصية:	مساهمة توافقية	نتيجة التحقق:	غير محقق
الرقم المميز	معيار التحقق	النتيجة	السبب		
11	زيادة النمو	محقق	ليكن لدينا two sprint backlogs ضمن Sprint I (التكرار): SBLa يتألف من n قصص مستخدم وSBL_b يتألف من m قصص مستخدم. بضم كلا backlogs سينتج لدينا SBL : $SBL = SBL_a + SBL_b$ يتألف من $o \leq n + m$ قصص مستخدم. عند تطبيق المقياس على كل من backlog ينتج لدينا مايلي: $IPWI_a = \sum_{i=1}^{n}(\frac{L_{i_1} - L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0} - L_{Ei_0}}{L_{i_0}})$		

$$IPWI_b = \sum_{j=1}^{m} (\frac{L_{j_1} - L_{Ei_1}}{L_{j_1}} - \frac{L_{j_0} - L_{Ej_0}}{L_{j_0}})$$

$$IPWI = \sum_{k=1}^{o} (\frac{L_{k_1} - L_{Ek_1}}{L_{k_1}} - \frac{L_{k_0} - L_{Ek_0}}{L_{k_0}})$$

$$IPWI_b + IPWI_a = \sum_{i=1}^{n} (\frac{L_{i_1} - L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0} - L_{Ei_0}}{L_{i_0}}) + \sum_{j=1}^{m} (\frac{L_{j_1} - L_{Ei_1}}{L_{j_1}} - \frac{L_{j_0} - L_{Ej_0}}{L_{j_0}}) \implies$$

$$IPWI_b + IPWI_a = \sum_{i=1}^{n} (\frac{L_{i_1} - L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0} - L_{Ei_0}}{L_{i_0}}) + \sum_{i=n+1}^{n+m} (\frac{L_{j_1} - L_{Ei_1}}{L_{j_1}} - \frac{L_{j_0} - L_{Ej_0}}{L_{j_0}}) \implies$$

$$IPWI_b + IPWI_a = \sum_{i=1}^{n+m} (\frac{L_{i_1} - L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0} - L_{Ei_0}}{L_{i_0}})$$

$$\because n + m \geq o \ \wedge i = k \ as \ starting \ point \implies$$

$$\sum_{i=1}^{n+m} (\frac{L_{i_1} - L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0} - L_{Ei_0}}{L_{i_0}}) \geq \sum_{i=1}^{o} (\frac{L_{i_1} - L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0} - L_{Ei_0}}{L_{i_0}})$$

$$IPWI_b + IPWI_a \geq IPWI$$

12	حساسية التفاعل	غير محقق	ليكن لدينا ثلاثة sprint backlogs ضمن Sprint I: SBL1 يتألف من n قصص مستخدم وSBL2 يتألف من m قصص مستخدم وSBL3 يتألف من o قصص مستخدم، $SBL13 = SBL1 + SBL3 \Longrightarrow Size(SBL13) = s_{13} \leq n + o$ $SBL23 = SBL2 + SBL3 \Longrightarrow Size(SBL23) = s_{23} \leq m + o$ $IPWI_1 = \sum_{i=1}^{n} (\frac{L_{i_1} - L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0} - L_{Ei_0}}{L_{i_0}})$ $IPWI_2 = \sum_{j=1}^{m} (\frac{L_{j_1} - L_{Ei_1}}{L_{j_1}} - \frac{L_{j_0} - L_{Ej_0}}{L_{j_0}})$ $IPWI_{13} = \sum_{i=1}^{s_{13}} (\frac{L_{i_1} - L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0} - L_{Ei_0}}{L_{i_0}})$ $IPWI_{23} = \sum_{j=1}^{s_{23}} (\frac{L_{j_1} - L_{Ei_1}}{L_{j_1}} - \frac{L_{j_0} - L_{Ej_0}}{L_{j_0}})$ $IPWI_{13} \leq IPWI_1 + IPWI_3, IPWI_{23} \leq IPWI_2 + IPWI_3$ $\because IPWI_1 = IPWI_2 \nRightarrow IPWI_{13} \neq IPWI_{23}$
17	الاستمرارية	غير محقق	يعاني المقياس($IPWI = \sum_{i=1}^{n} (\frac{L_{i_1} - L_{Ei_1}}{L_{i_1}} - \frac{L_{i_0} - L_{Ei_0}}{L_{i_0}}$ من انقطاعين في الحساب: $L_{i_1} = 0 \vee L_{i_0} = 0$ يمكن لإحدى هذه الحالات من الحدوث في حال كون إحدى قصص المستخدم لم تنجز بعد عند القياس.

الملحق(3) التحقق من مقياس API

اسم المقياس:	API	الخاصية:	السلامة الرياضية	نتيجة التحقق:	محقق
الرقم المميز	معيار التحقق	النتيجة	السبب		
11	زيادة النمو	محقق	ليكن لدينا two sprint backlogs ضمن Sprint I (التكرار): SBL_1 يتألف من N قصص مستخدم كلية أنجز منها n قصص مستخدم وSBL_2 يتألف من M قصص مستخدم كلية أنجز منها m قصص مستخدم. بضم كلا backlogs سينتج لدينا SBL : $SBL = SBL_a + SBL_b$ يتألف من $O \leq N + M$ قصص مستخدم أنجز منها $o \leq n + m$. عند تطبيق المقياس على كل backlog ينتج لدينا مايلي: $API_1 = \frac{\sum_{i=1}^{n} [EBV_i \times F_i]}{\sum_{j=1}^{N} [EBV_j \times F_j]}$ $API_2 = \frac{\sum_{i=1}^{m} [EBV_i \times F_i]}{\sum_{j=1}^{M} [EBV_j \times F_j]}$ $API = \frac{\sum_{i=1}^{o} [EBV_i \times F_i]}{\sum_{j=1}^{O} [EBV_j \times F_j]}$		

			$API \leq \frac{\sum_{i=1}^{n}[EBV_i \times F_i]}{\sum_{j=1}^{N}[EBV_j \times F_j] + \sum_{j=N+1}^{N+M}[EBV_j \times F_j]} + \frac{\sum_{i=n+1}^{n+m}[EBV_i \times F_i]}{\sum_{j=1}^{N}[EBV_j \times F_j] + \sum_{j=N+1}^{N+M}[EBV_j \times F_j]}$ $API_1 + API_2 = \frac{\sum_{i=1}^{n}[EBV_i \times F_i]}{\sum_{j=1}^{N}[EBV_j \times F_j]} + \frac{\sum_{i=1}^{m}[EBV_i \times F_i]}{\sum_{j=1}^{M}[EBV_j \times F_j]} \Longrightarrow$ بمقارنة المعادلتين السابقتين نجد: $API_1 + API_2 \geq API$
12	حساسية التفاعل	محقق	ليكن لدينا ثلاثة sprint backlogs ضمن Sprint I: SBL_1 يتألف من Nقصص مستخدم كلية أنجز منها n قصص مستخدم، وSBL_2 يتألف من M قصص مستخدم كلية أنجز منها m قصص مستخدم، وSBL_3 يتألف من O قصص مستخدم أنجز منها o قصص مستخدم، $SBL13 = SBL1 + SBL3 \Longrightarrow Size(SBL13) = S_{13} \leq N + O$ أنجز منها $s_{13} \leq n + o$ $SBL23 = SBL2 + SBL3 \Longrightarrow Size(SBL23) = S_{23} \leq M + O$ أنجز منها $s_{23} \leq m + o$ $API_1 = \frac{\sum_{i=1}^{n}[EBV_i \times F_i]}{\sum_{j=1}^{N}[EBV_j \times F_j]}$ $API_2 = \frac{\sum_{i=1}^{m}[EBV_i \times F_i]}{\sum_{j=1}^{M}[EBV_j \times F_j]}$ $API_3 = \frac{\sum_{i=1}^{o}[EBV_i \times F_i]}{\sum_{j=1}^{O}[EBV_j \times F_j]}$ $API_{13} = \frac{\sum_{i=1}^{s13}[EBV_i \times F_i]}{\sum_{j=1}^{S13}[EBV_j \times F_j]}$ $API_{23} = \frac{\sum_{i=1}^{s23}[EBV_i \times F_i]}{\sum_{j=1}^{S23}[EBV_j \times F_j]}$ $API_{13} \leq API_1 + API_3, API_{23} \leq API_2 + API_3$ $\because API_1 = API_2 \Longrightarrow API_{13} \neq API_{23}$
13	الرتابة	محقق	ليكن لدينا two sprint backlogs ضمن Sprint I: SBL_1 يتألف من Nقصص مستخدم كلية أنجز منها n قصص مستخدم، وSBL_2 يتألف من M قصص مستخدم كلية أنجز منها m قصص مستخدم،

			$SBL_1 \subset SBL_2 \Longrightarrow N \leq M \wedge n \leq m$ وقصص مستخدم $SBL1$ متضمنة حكماً في $SBL2$ بفرض أن المقياس طبق على كل backlog على حدى بنفس التكرار: $API_1 = \frac{\sum_{i=1}^{n}[EBV_i \times F_i]}{\sum_{j=1}^{N}[EBV_j \times F_j]}$ $API_2 = \frac{\sum_{i=1}^{m}[EBV_i \times F_i]}{\sum_{j=1}^{M}[EBV_j \times F_j]}$ $\because N \leq M \wedge n \leq m \Rightarrow API_1 \leq API_2$
14	عدم الرسمية	محقق	ليكن لدينا two sprint backlogs ضمن Sprint I: SBL_1 يتألف من Nقصص مستخدم كلية أنجز منها n قصص مستخدم، وSBL_2 يتألف من M قصص مستخدم كلية أنجز منها m قصص مستخدم، $\exists SBL_1, \exists SBL_2 : N \leq M \wedge n \leq m$ وقصص مستخدم SBL_1 وقصص مستخدمSBL_2 قصص مختلفة. بفرض أن المقياس طبق على كل backlog على حدى بنفس التكرار: $API_1 = \frac{\sum_{i=1}^{n}[EBV_i \times F_i]}{\sum_{j=1}^{N}[EBV_j \times F_j]}$ $API_2 = \frac{\sum_{i=1}^{m}[EBV_i \times F_i]}{\sum_{j=1}^{M}[EBV_j \times F_j]}$ $\Longrightarrow API_1 \neq API_2$
17	الاستمرارية	محقق	لا يعاني المقياس API من أي انقطاعات في القيم انقطاعين غير متوقعة. الحالة الوحيدة التي يمكن أن يعاني منها هذه المقياس من انقطاع في القيم هي حالة عدم وجود قصص مستخدمين في SBL بمعنى عدم وجود تكرار Sprint على الإطلاق وهنا لا داعي للقياس على الإطلاق.
18	اتساق الأبعاد	غير محقق	لا يحقق المقياس API= $\frac{\sum_{i=1}^{n}[EBV_i \times F_i]}{\sum_{i=1}^{N}[EBV_j \times F_j]}$ هذا المعيار. على الرغم من اعتماد المقياس على الحصول على مقدار نسبي ناتج عن مجموع قيم ناتجة عن جداء بعدين أحدهما هو قيمة الأعمال المكتسبة والثاني هو الوظيفية لكل قصة مستخدم، ولكن قيمة كل جداء لكل قصة مستخدم لا يمكن شرحها أو تفسيرها أو البناء عليها.
19	تحقق المقدار	غير محقق	لا يمكن التأكد أن جميع التحويلات المنطقية يمكن تطبيقها على هذا المقياس ولا يوجد معنى دقيق لفهم الفواصل في قيم هذا المقياس وكيف يمكن تحقيقه. حتى القيمة الصفرية لهذا المقياس لا يمكن تفسيرها بدقة. رياضياً يمكن الحصول على قيمة صفرية للمقياس في حال كون قيمة الأعمال المكتسبة أو الوظيفية لجميع قصص المستخدم هي صفرية. كيف يمكن تفسير عدم تحقق وظيفية قصة مستخدم $(F_i = 0)$ مع وجود قيمة أعمال مكتسبة لها $(EBV_i \neq 0)$.

اسم المقياس:	API	الخاصية:	التطبيق العملي	نتيجة التحقق: محقق

الرقم المميز	معيار التحقق	النتيجة	السبب
1	الفعل	محقق	إن نتيجة المقياس $API = \frac{\sum_{i=1}^{n}[EBV_i \times F_i]}{\sum_{j=1}^{N}[EBV_j \times F_j]}$ هي نسبة مئوية دون واحدة وبالتالي قيمتها تعبر عن نسبة تقدم الإنجاز في التكرار وبالتالي يعطي مؤشراً واضحاً لإمكانية الانتهاء من إنجاز كافة قصص المستخدم في الوقت المحدد من عدمه مما يمكن قائد الفريق من اتخاذ القرارات المناسبة اعتماداً على قيمة هذا المقياس.
2	الإنتاجية الاقتصادية	غير محقق	لا يحقق المقياس $API = \frac{\sum_{i=1}^{n}[EBV_i \times F_i]}{\sum_{j=1}^{N}[EBV_j \times F_j]}$ هذا المعيار حيث يعتمد على الحصول على مقدار نسبي ناتج عن مجموع قيم ناتجة عن جداء بعدين أحدهما هو قيمة الأعمال المكتسبة والثاني هو الوظيفية لكل قصة مستخدم موجودة في SBL وكليهما مقياسين غير مباشرين يتضمنان مجموعة من القياسات المباشرة وغير المباشرة وهذا يتطلب جهداً ووقتاً من قبل المطور أو الفاحص.
5	عدم الاستغلال	محقق	يعتمد المقياس $API = \frac{\sum_{i=1}^{n}[EBV_i \times F_i]}{\sum_{j=1}^{N}[EBV_j \times F_j]}$ على الحصول على مقدار نسبي ناتج عن مجموع قيم ناتجة عن جداء بعدين أحدهما هو قيمة الأعمال المكتسبة والثاني هو الوظيفية لكل قصة مستخدم موجودة في SBL وكليهما مقياسين غير مباشرين معتمدين.
7	علاقة المقياس بالمنتج أو الإجراء	محقق	يحقق المقياس $API = \frac{\sum_{i=1}^{n}[EBV_i \times F_i]}{\sum_{j=1}^{N}[EBV_j \times F_j]}$ هذا المعيار حيث تم تطويره لإجراءات التطوير الرشيقة ومع ذلك فإن المفهوم البسيط والمباشر الذي يعتمده يمكن تطويعه لإجراءات تطويرية أخرى.
9	العلاقة السببية	غير محقق	لا يحقق المقياس $API = \frac{\sum_{i=1}^{n}[EBV_i \times F_i]}{\sum_{j=1}^{N}[EBV_j \times F_j]}$ هذا المعيار حيث لا توجد علاقة موضحة تربط بين نتيجة هذا المقياس مع جودة المنتج النهائي.

اسم المقياس:	API	الخاصية:	الصحة	نتيجة التحقق:	محقق
الرقم المميز	معيار التحقق	النتيجة	السبب		
4	وثوقية المقياس	محقق	يحقق المقياس $API = \frac{\sum_{i=1}^{n}[EBV_i \times F_i]}{\sum_{j=1}^{N}[EBV_j \times F_j]}$ هذا المعيار بفرض ثبات أوزان أولوية قصص المستخدم في SBL خلال التكرار وثبات البنية الوظيفية لقصص المستخدم خلال التكرار وهذه فرضيات غير متعلقة بأساس المقياس وإنما تعتمد على التخطيط الجيد.		

اسم المقياس:	API	الخاصية:	الكشف عن الفروقات	نتيجة التحقق:	غير محقق
الرقم المميز	معيار التحقق	النتيجة	السبب		
6	ثبات التحويل	غير محقق	ليكن لدينا two sprint backlogs ضمن Sprint I: SBL_1 يتألف من Nقصص مستخدم كلية أنجز منها n قصص مستخدم، وSBL_2 يتألف من M قصص مستخدم كلية أنجز منها m قصص مستخدم، $\exists SBL_1, \exists SBL_2 : SBL1 \equiv SBL2 \wedge N \neq M \wedge n \neq m$ التكافؤ السياقي بين SBL1 و SBL2 هنا يعني أن كليهما يوصف نفس الوظائف عن طريق مجموعات مختلفة من قصص المستخدم.		

			بفرض أن المقياس طبق على كل backlog على حدى بنفس التكرار: $API_1 = \frac{\sum_{i=1}^{n}[EBV_i \times F_i]}{\sum_{j=1}^{N}[EBV_j \times F_j]}$ $API_2 = \frac{\sum_{i=1}^{m}[EBV_i \times F_i]}{\sum_{j=1}^{M}[EBV_j \times F_j]}$ $\not\Rightarrow API_1 = API_2$
10	درجة التقسيم المناسبة	غير محقق	ليكن لدينا two sprint backlogs ضمن Sprint I: SBL_1 يتألف من Nقصص مستخدم كلية أنجز منها n قصص مستخدم، وSBL_2 يتألف من M قصص مستخدم كلية أنجز منها m قصص مستخدم، يمكن القول أن المقياس أنه شديد النعومة لأنه نتيجة المقياس سوف تختلف باختلاف عدد قصص المستخدم المنفذة، والكلية وقيمة الأعمال المكتسبة ونقط الوظيفة الخاصة بكل منها، مدى التقدم في الإنجاز لحظة القياس. بفرض أن المقياس طبق على كل backlog على حدى بنفس التكرار: $API_1 = \frac{\sum_{i=1}^{n}[EBV_i \times F_i]}{\sum_{j=1}^{N}[EBV_j \times F_j]}$ $API_2 = \frac{\sum_{i=1}^{m}[EBV_i \times F_i]}{\sum_{j=1}^{M}[EBV_j \times F_j]}$ $\Longrightarrow API_1 \neq API_2$
14	عدم الرسمية	محقق	ليكن لدينا two sprint backlogs ضمن Sprint I: SBL_1 يتألف من Nقصص مستخدم كلية أنجز منها n قصص مستخدم، وSBL_2 يتألف من M قصص مستخدم كلية أنجز منها m قصص مستخدم، $\exists SBL_1, \exists SBL_2 : N \leq M \wedge n \leq m$ وقصص مستخدم SBL_1 وقصص مستخدمSBL_2 قصص مختلفة. بفرض أن المقياس طبق على كل backlog على حدى بنفس التكرار: $API_1 = \frac{\sum_{i=1}^{n}[EBV_i \times F_i]}{\sum_{j=1}^{N}[EBV_j \times F_j]}$ $API_2 = \frac{\sum_{i=1}^{m}[EBV_i \times F_i]}{\sum_{j=1}^{M}[EBV_j \times F_j]}$ $\Longrightarrow API_1 \neq API_2$

اسم المقياس:	API	الخاصية:	تعزيز الفرضيات	نتيجة التحقق:	محقق
الرقم المميز	معيار التحقق	النتيجة	السبب		
8	استقلالية العامل	محقق	لتحقيق هذا المعيار، يجب على المقياس أن يقوم بإجراءات قياس مستقلة للعوامل الداخلة في حسابه أي قيمة الأعمال المكتسبة والوظيفية لكل قصة من قصص المستخدم المنجزة نسبة إلى قصص المستخدم الكلية، وهي قياسات فردية غير مباشرة وقيمها غير متعلقة ببعضها		

			وبالتالي يعتبر هذا المعيار محققاً في حالة هذا المقياس $API = \frac{\sum_{i=1}^{n}[EBV_i \times F_i]}{\sum_{j=1}^{N}[EBV_j \times F_j]}$

اسم المقياس:	API	الخاصية:	الجدوى	نتيجة التحقق:	محقق
الرقم المميز	معيار التحقق	النتيجة	السبب		
3	الاتساق الداخلي	غير محقق	لا يحقق المقياس هذا المعيار حيث يعتمد في حسابه على مقياسين غير مباشرين يحسبان لكل قصة مستخدم أحدهما مقياس داخلي (يتعلق بالخصائص الداخلية للكود) وهو الوظيفية والثاني مقياس خارجي يعتمد على نسبة أولوية إنجاز قصة المستخدم نسبة لباقي قصص المستخدم. على الرغم من أن كلا المقياسين يستهدفان قصص المستخدم ولكنهما غير مترابطين من الناحية المفاهيمية.		
8	استقلالية العامل	محقق	لتحقيق هذا المعيار، يجب على المقياس أن يقوم بإجراءات قياس مستقلة للعوامل الداخلة في حسابه أي قيمة الأعمال المكتسبة والوظيفية لكل قصة من قصص المستخدم المنجزة نسبة إلى قصص المستخدم الكلية، وهي قياسات فردية غير مباشرة وقيمها غير متعلقة ببعضها وبالتالي يعتبر هذا المعيار محققاً في حالة هذا المقياس $API = \frac{\sum_{i=1}^{n}[EBV_i \times F_i]}{\sum_{j=1}^{N}[EBV_j \times F_j]}$		
14	عدم الرسمية	محقق	ليكن لدينا two sprint backlogs ضمن Sprint I: SBL_1 يتألف من N قصص مستخدم كلية أنجز منها n قصص مستخدم، وSBL_2 يتألف من M قصص مستخدم كلية أنجز منها m قصص مستخدم، $\exists SBL_1, \exists SBL_2 : N \leq M \wedge n \leq m$ وقصص مستخدم SBL_1 وقصص مستخدمSBL_2 قصص مختلفة. بفرض أن المقياس طبق على كل backlog على حدى بنفس التكرار: $API_1 = \frac{\sum_{i=1}^{n}[EBV_i \times F_i]}{\sum_{j=1}^{N}[EBV_j \times F_j]}$ $API_2 = \frac{\sum_{i=1}^{m}[EBV_i \times F_i]}{\sum_{j=1}^{M}[EBV_j \times F_j]}$ $\Longrightarrow API_1 \neq API_2$		
15	تحقق التعلق	محقق	يحقق هذا المقياس $API = \frac{\sum_{i=1}^{n}[EBV_i \times F_i]}{\sum_{j=1}^{N}[EBV_j \times F_j]}$ هذا المعيار بحكم أن أي تعديل في ترتيب قصص المستخدم المنجزة سوف يؤثر على نتيجة المقياس الكلية وذلك لأن لكل قصة مستخدم قيمة الوظيفية وقيمة أعمال مكتسبة مختلفة.		
16	عدم حساسية تعديل الاسم	محقق	يحقق هذا المقياس $API = \frac{\sum_{i=1}^{n}[EBV_i \times F_i]}{\sum_{j=1}^{N}[EBV_j \times F_j]}$ هذا المعيار بحكم أن أسماء قصص المستخدم ليس لها أي دور في المقياس وأي تعديل في أسماء قصص المستخدم المنجزة لن يؤثر على نتيجة المقياس النهائية.		
17	الاستمرارية	محقق	لا يعاني المقياس API من أي انقطاعات في القيم انقطاعين غير متوقعة. الحالة الوحيدة التي يمكن أن يعاني منها هذه المقياس من انقطاع في القيم هي حالة عدم وجود قصص مستخدمين في SBL بمعنى عدم وجود تكرار Sprint على الإطلاق وهنا لا داعي للقياس		

			على الإطلاق.
18	اتساق الأبعاد	غير محقق	لا يحقق المقياس $API = \frac{\sum_{i=1}^{n}[EBV_i \times F_i]}{\sum_{j=1}^{N}[EBV_j \times F_j]}$ هذا المعيار. على الرغم من اعتماد المقياس على الحصول على مقدار نسبي ناتج عن مجموع قيم ناتجة عن جداء بعدين أحدهما هو قيمة الأعمال المكتسبة والثاني هو الوظيفية لكل قصة مستخدم، ولكن قيمة كل جداء لكل قصة مستخدم لا يمكن شرحها أو تفسيرها أو البناء عليها.
20	تحقق الوحدة	محقق	إن نتيجة المقياس $API = \frac{\sum_{i=1}^{n}[EBV_i \times F_i]}{\sum_{j=1}^{N}[EBV_j \times F_j]}$ هي نسبة مئوية دون واحدة وبالتالي قيمتها تعبر عن نسبة تقدم الإنجاز التكرار.

اسم المقياس:	API	الخاصية:	إعلام القرار	نتيجة التحقق:	غير محقق
الرقم المميز	معيار التحقق	النتيجة	السبب		
1	الفعل	محقق	إن نتيجة المقياس $API = \frac{\sum_{i=1}^{n}[EBV_i \times F_i]}{\sum_{j=1}^{N}[EBV_j \times F_j]}$ هي نسبة مئوية دون واحدة وبالتالي قيمتها تعبر عن نسبة تقدم الإنجاز في التكرار وبالتالي يعطي مؤشراً واضحاً لإمكانية الانتهاء من إنجاز كافة قصص المستخدم في الوقت المحدد من عدمه مما يمكن قائد الفريق من اتخاذ القرارات المناسبة اعتماداً على قيمة هذا المقياس.		
2	الإنتاجية الاقتصادية	غير محقق	لا يحقق المقياس $API = \frac{\sum_{i=1}^{n}[EBV_i \times F_i]}{\sum_{j=1}^{N}[EBV_j \times F_j]}$ هذا المعيار حيث يعتمد على الحصول على مقدار نسبي ناتج عن مجموع قيم ناتجة عن جداء بعدين أحدهما هو قيمة الأعمال المكتسبة والثاني هو الوظيفية لكل قصة مستخدم موجودة في SBL وكليهما مقياسين غير مباشرين يتضمنان مجموعة من القياسات المباشرة وغير المباشرة وهذا يتطلب جهداً ووقتاً من قبل المطور أو الفاحص.		
9	العلاقة السببية	غير محقق	لا يحقق المقياس $API = \frac{\sum_{i=1}^{n}[EBV_i \times F_i]}{\sum_{j=1}^{N}[EBV_j \times F_j]}$ هذا المعيار حيث لا توجد علاقة موضحة تربط بين نتيجة هذا المقياس مع جودة المنتج النهائي.		

اسم المقياس:	API	الخاصية:	بناء النظرية	نتيجة التحقق:	غير محقق
الرقم المميز	معيار التحقق	النتيجة	السبب		
9	العلاقة السببية	غير محقق	لا يحقق المقياس $API = \frac{\sum_{i=1}^{n}[EBV_i \times F_i]}{\sum_{j=1}^{N}[EBV_j \times F_j]}$ هذا المعيار حيث لا توجد علاقة موضحة تربط بين نتيجة هذا المقياس مع جودة المنتج النهائي.		

اسم المقياس:	API	الخاصية:	مساهمة توافقية	نتيجة التحقق:	محقق
الرقم المميز	معيار التحقق	النتيجة	السبب		
11	زيادة النمو	محقق	ليكن لدينا two sprint backlogs ضمن Sprint I (التكرار): SBL_1 يتألف من N قصص مستخدم كلية أنجز منها n قصص مستخدم وSBL_2 يتألف من M قصص مستخدم كلية أنجز منها m قصص مستخدم. بضم كلا backlogs سينتج لدينا SBL : $SBL = SBL_a + SBL_b$		

يتألف من $O \leq N + M$قصص مستخدم أنجز منها $o \leq n + m$. عند تطبيق المقياس على كل backlog ينتج لدينا مايلي: $API_1 = \frac{\sum_{i=1}^{n}[EBV_i \times F_i]}{\sum_{j=1}^{N}[EBV_j \times F_j]}$ $API_2 = \frac{\sum_{i=1}^{m}[EBV_i \times F_i]}{\sum_{j=1}^{M}[EBV_j \times F_j]}$ $API = \frac{\sum_{i=1}^{o}[EBV_i \times F_i]}{\sum_{j=1}^{O}[EBV_j \times F_j]}$ $API \leq \frac{\sum_{i=1}^{n}[EBV_i \times F_i]}{\sum_{j=1}^{N}[EBV_j \times F_j] + \sum_{j=N+1}^{N+M}[EBV_j \times F_j]} + \frac{\sum_{i=n+1}^{n+m}[EBV_i \times F_i]}{\sum_{j=1}^{N}[EBV_j \times F_j] + \sum_{j=N+1}^{N+M}[EBV_j \times F_j]}$ $API_1 + API_2 = \frac{\sum_{i=1}^{n}[EBV_i \times F_i]}{\sum_{j=1}^{N}[EBV_j \times F_j]} + \frac{\sum_{i=1}^{m}[EBV_i \times F_i]}{\sum_{j=1}^{M}[EBV_j \times F_j]} \Longrightarrow$ بمقارنة المعادلتين السابقتين نجد: $API_1 + API_2 \geq API$			
ليكن لدينا ثلاثة sprint backlogs ضمن Sprint I: SBL_1 يتألف من Nقصص مستخدم كلية أنجز منها n قصص مستخدم، وSBL_2 يتألف من M قصص مستخدم كلية أنجز منها m قصص مستخدم، وSBL_3 يتألف من O قصص مستخدم أنجز منها o قصص مستخدم، $SBL13 = SBL1 + SBL3 \Longrightarrow Size(SBL13) = S_{13} \leq N + O$ $s_{13} \leq n + o$ أنجز منها $SBL23 = SBL2 + SBL3 \Longrightarrow Size(SBL23) = S_{23} \leq M + O$ $s_{23} \leq m + o$ أنجز منها $API_1 = \frac{\sum_{i=1}^{n}[EBV_i \times F_i]}{\sum_{j=1}^{N}[EBV_j \times F_j]}$ $API_2 = \frac{\sum_{i=1}^{m}[EBV_i \times F_i]}{\sum_{j=1}^{M}[EBV_j \times F_j]}$ $API_3 = \frac{\sum_{i=1}^{o}[EBV_i \times F_i]}{\sum_{j=1}^{O}[EBV_j \times F_j]}$ $API_{13} = \frac{\sum_{i=1}^{s13}[EBV_i \times F_i]}{\sum_{j=1}^{S13}[EBV_j \times F_j]}$ $API_{23} = \frac{\sum_{i=1}^{s23}[EBV_i \times F_i]}{\sum_{j=1}^{S23}[EBV_j \times F_j]}$ $API_{13} \leq API_1 + API_3, API_{23} \leq API_2 + API_3$ $\because API_1 = API_2 \Longrightarrow API_{13} \neq API_{23}$	محقق	حساسية التفاعل	12
لا يعاني المقياس API من أي انقطاعات في القيم انقطاعين غير متوقعة. الحالة الوحيدة التي يمكن أن يعاني منها هذه المقياس من انقطاع في القيم هي حالة عدم وجود قصص مستخدمين في SBL بمعنى عدم وجود تكرار Sprint على الإطلاق وهنا لا داعي للقياس على الإطلاق.	محقق	الاستمرارية	17