

Yappi - Developer Happiness

IP5 Project

Windisch, August 2025

Studenten: Xeno Isenegger, Gideon Monterosa

Fachbetreuer: Norbert Seyff, Nitish Patkar

Abstract

Inhaltsverzeichnis

1	Einleitung	6
1.1	Hintergrund und Motivation	6
1.2	Ziele und Vision	7
1.3	Fragestellungen	8
2	Hintergrund	9
3	State of the Art	10
3.1	Definition von Entwicklerzufriedenheit	10
3.2	Stand der Forschung und verwandte Arbeiten	10
3.3	Bestehende Lösungen und Wettbewerbsanalyse	10
4	Methoden	12
4.1	Projektmethodik	12
4.2	Prototypen	12
4.3	Proof of Concepts	12
5	Konzeptentwurf	13
5.1	Zugriffskontrolle über API Keys	13
5.2	Companion Apps	13
5.2.1	Integration in die Entwicklungsumgebung	13
5.2.2	Integration von Kalenderdaten	13
5.2.3	Integration von Gesundheitsdaten	13
5.3	Yappi Coach	14
5.4	Konzeptevaluation	14
6	Implementierung	15
6.1	Zugriffskontrolle über API Keys	15
6.2	Companion Apps	15
6.2.1	IntelliJ IDEA Companion	15
6.2.2	Calendar Companion	15

6.2.3	Health Companion	15
6.3	Deployment	15
7	Evaluation	16
8	Quellen	17

Abbildungsverzeichnis

Tabellenverzeichnis

Kapitel 1

Einleitung

1.1 Hintergrund und Motivation

TODO: evt Ausgangslage

TODO: aus Project Agreement: zu überarbeiten

Die Zufriedenheit von Entwicklerinnen und Entwicklern wird, wenn überhaupt, meist nur anhand der Menge ihrer geleisteten Arbeit gemessen. Dabei entstehen zwangsläufig Defizite, und ein halbjährliches Mitarbeitergespräch erweist sich oft als wenig wirksame Massnahme zur Problemlösung.

Dieses Projekt baut auf einer bestehenden Arbeit auf, in der eine Plattform zur Erfassung der Entwicklerzufriedenheit entwickelt wurde. Die Webapplikation Yappi ermöglicht es Entwicklerinnen und Entwicklern, ihre Zufriedenheit mit ihrer Arbeit und ihrer aktuellen Situation fortlaufend zu bewerten. Yappi erfasst emotionale Faktoren wie Happiness sowie weitere Zufriedenheitsindikatoren. Zusätzlich können spezifische Aufgaben und Arbeitstypen individuell bewertet werden. Die erhobenen Daten werden anonym auf Teamebene analysiert, um ein fundiertes Verständnis für die Stimmung innerhalb der Teams zu gewinnen.

Entwicklerinnen und Entwickler haben die Möglichkeit, ihre Zufriedenheit für verschiedene Teams zu erfassen, wodurch gezielte Analysen ermöglicht werden. Unternehmen erhalten dadurch wertvolle Einblicke, um das Arbeitsumfeld gezielt zu verbessern.

Unser Projekt baut auf Yappi auf und zielt darauf ab, die Erfassung der Zufriedenheit weiter zu optimieren. Es wird untersucht, wie die Daten noch präziser erfasst und ausgewertet werden können, um langfristige Verbesserungen zu unterstützen. Diese Arbeit dient als Grundlage für ein weiterfüh-

rendes Forschungsprojekt, das sich vertieft mit der Entwicklerzufriedenheit auseinandersetzt und zusätzliche Erkenntnisse gewinnen soll.

1.2 Ziele und Vision

TODO: Text aus dem Project Agreement noch zu überarbeiten

Yappi wird zu einer umfassenden Plattform weiterentwickelt, die nicht nur die Zufriedenheit misst, sondern sich nahtlos in den Arbeitsalltag integriert und wertvolle Handlungsempfehlungen liefert. Dazu werden folgende Kernaspekte umgesetzt:

Produktivitätsfaktoren identifizieren

Durch eine tiefere Analyse von Zufriedenheitsindikatoren sollen zentrale Faktoren ermittelt werden, die sich positiv oder negativ auf die Produktivität und das Wohlbefinden von Entwicklerinnen und Entwicklern auswirken. Diese Erkenntnisse werden genutzt, um Vorschläge zu Verbesserungsmassnahmen abzuleiten.

Integration in den Arbeitsprozess

Yappi soll sich direkt in bestehende Arbeitsabläufe einfügen, um die Erfassung der Zufriedenheit möglichst intuitiv und effizient zu gestalten. Dies kann durch verschiedene Schnittstellen und Erweiterungen erfolgen, die eine nahtlose Interaktion ermöglichen.

Erweiterung um kontextbezogene Daten

Um ein umfassenderes Bild der Arbeitszufriedenheit zu erhalten, können weitere Einflussfaktoren berücksichtigt werden. Dazu gehören beispielsweise arbeitsbezogene Rahmenbedingungen oder individuelle Gesundheits- und Belastungsindikatoren. Diese Daten sollen helfen, ein besseres Verständnis für langfristige Trends und Zusammenhänge zu entwickeln.

Intelligente Analyse und Handlungsempfehlungen

Durch die Integration von AI schnittstellen können gezielte Analysen erstellt und individualisierte Empfehlungen abgeleitet werden. Dies kann sowohl auf individueller als auch auf Teamebene erfolgen, um nachhaltige Verbesserungen im Arbeitsumfeld zu fördern.

Fazit

Mit diesen Erweiterungen wird Yappi zu einem essenziellen Bestandteil des Entwickleralltags. Es bietet nicht nur eine präzisere Erfassung der Zufriedenheit, sondern liefert auch wertvolle Einblicke und Handlungsempfehlungen, um die Arbeitsbedingungen nachhaltig zu verbessern. Unternehmen erhalten fundierte Analysen und können gezielt Massnahmen ergreifen, um eine motivierte und produktive Entwicklergemeinschaft zu fördern.

1.3 Fragestellungen

TODO: Text aus dem Project Agreement noch zu überarbeiten

- A. Durch welche Technologien und Schnittstellen kann Yappi erweitert werden, um ein reibungsloses und einfaches Erfassen von Zufriedenheitsdaten zu ermöglichen?
 - a. Entwicklung von Entwickler-Tool-Plugins, die nahtlos in bestehenden Arbeitsumgebungen integriert werden können, um die Nutzung von Yappi angenehmer und effizienter zu gestalten. Diese Plugins sollen Entwicklern ermöglichen, direkt in ihrer bevorzugten Umgebung Feedback zu erfassen, ohne den Arbeitsfluss zu unterbrechen. Integration von Yappi in verschiedene Plattformen und Tools wie Webbrowser, IntelliJ, Microsoft Teams und Outlook.
- B. Wie können Gesundheitsdaten in die Auswertung der Entwicklerzufriedenheit einfließen?
 - a. Direkte Anbindung der Gesundheitsdaten-API, um relevante Gesundheitsmetriken wie Herzfrequenz, Schlafqualität oder Stresslevel automatisch in die Analyse der Entwicklerzufriedenheit zu integrieren. Dies ermöglicht eine genauere Einschätzung des Wohlbefindens und potenzieller Belastungsfaktoren.
- C. Wie kann Yappi Teams und Entwickler dabei unterstützen, aus den erfassten Zufriedenheitsdaten Handlungsempfehlungen abzuleiten, um die Zufriedenheit und Produktivität von Entwicklern zu erhöhen?
 - a. Entwicklung eines Yappi Coach, der anhand einer detaillierten Analyse der erfassten Daten gezielte Tipps zur Verbesserung der Arbeitsweise gibt. Beispielsweise könnte der Coach darauf hinweisen, dass Meetings nicht länger als 1,5 Stunden dauern sollten, da längere Sitzungen die Zufriedenheit und Konzentration der Entwickler negativ beeinflussen können.
 - b. Integration von KI-gestützten Diensten, die auf Basis der gesammelten Gesundheitsdaten sowie Zufriedenheits- und Produktivitätsmetriken individuelle Massnahmen vorschlagen. Diese KI-gestützten Empfehlungen können Teams dabei helfen, gezielt Optimierungen vorzunehmen, um die Arbeitsbedingungen und die Effizienz der Entwickler nachhaltig zu verbessern.

Kapitel 2

Hintergrund

TODO: unterkapitel für den Stand von Yappi vor dem Projekt

Kapitel 3

State of the Art

3.1 Definition von Entwicklerzufriedenheit

Entwicklerzufriedenheit wird in der Literatur als Balance zwischen positiven und negativen Erlebnissen bei der Arbeit definiert (Graziotin & Fagerholm, 2019).

3.2 Stand der Forschung und verwandte Arbeiten

3.3 Bestehende Lösungen und Wettbewerbsanalyse

QUESTION: Braucht es hier quellen?

QUESTION: Sind das zu viele?

Officevibe ist ein SaaS-Tool für wöchentliche Puls-Umfragen, das primär die allgemeine Mitarbeiter bindung und das Engagement im Unternehmen misst. Für Entwicklerteams liefert es zwar Stimmungstrends, doch es fehlen auf Entwickler zugeschnittene Kontextbasierte Daten.

TeamMood verschickt täglich einen kurzen Stimmungs-Prompt per E-Mail, Slack oder Teams und visualisiert die Antworten als Verlaufsdiagramm. Die Lösung ist niedrigschwellig, erfasst jedoch keine Code- oder Prozessmetriken und gibt auch keine automatisierten Empfehlungen, so dass Entwickler selbst interpretieren müssen, was aus den Daten folgt.

Happimeter stammt aus einem Forschungsprojekt der TU Wien und erzeugt einen „Happiness-Score“ auf Basis von Wearable-Sensoren wie Herzfrequenz und Aktivität. Es fehlt jedoch der Kontext zur eigentlichen Entwicklungsarbeit.

Code Climate Velocity analysiert Pull-Request-Dauer, Cycle-Time und Review-Aktivität, um Teamleistung zu beurteilen. Die Plattform konzentriert sich vollständig auf Prozess- und Code-Metriken. Subjektive Zufriedenheitsdaten bleiben aussen vor, sodass der Einfluss der Stimmung auf die gemessene Performance unsichtbar bleibt.

GitHub Insights bietet Dashboards zu Commits, Pull-Requests und Release-Takt. Das Tool liefert wertvolle Aktivitätsstatistiken, berücksichtigt jedoch keine emotionalen Faktoren.

Microsoft Viva Insights wertet Kalendereinträge, Meetings und Kommunikationsmuster in Microsoft 365 aus, um Arbeitsgewohnheiten zu optimieren. Zwar erhält man Hinweise zu Meeting-Überlastung oder Fokuszeiten, doch wird das Wohlbefinden ausschliesslich indirekt aus Kommunikationsdaten abgeleitet, ohne jede Kopplung an tatsächliche Entwickler-tätigkeiten oder subjektive Gefühlslagen.

Engagement-Tools erfassen Stimmung, Dev-Analytics Leistung, Gesundheitsdaten, doch keine Lösung integriert alle drei Dimensionen und leitet automatisch konkrete Verbesserungen ab. Yappi positioniert sich daher als Brückentechnologie, die

- das Erfassen von Zufriedenheitsdaten in den Arbeitsprozess integriert,
- Daten zu Commits und Kalenderereignissen automatisch als Kontext mit einbezieht,
- zusätzlich Gesundheitsdaten sammelt,
- per KI Coach handlungsorientierte Empfehlungen liefert.

Kapitel 4

Methoden

TODO: Kanban erwähnen

- 4.1 Projektmethodik
- 4.2 Prototypen
- 4.3 Proof of Concepts

TODO: arc42 erwähnen

Kapitel 5

Konzeptentwurf

5.1 Zugriffskontrolle über API Keys

TODO: Kapitel evt umbenennen; Aussage soll sein die das Yappi zu Plattform wird

5.2 Companion Apps

5.2.1 Integration in die Entwicklungsumgebung

5.2.2 Integration von Kalenderdaten

TODO XENO:

5.2.3 Integration von Gesundheitsdaten

Die folgenden Gesundheitsmetriken sind für unsere Arbeit relevant, da sie einen interessanten Einfluss auf die Zufriedenheit von Softwareentwicklern haben.

TODO: Nochmals anschauen

1. Schlafdauer

Untersuchungen zeigen, dass längere Schlafdauer mit höherer Arbeitszufriedenheit einhergeht: Männer mit mehr Schlaf berichten signifikant grössere Zufriedenheit am Arbeitsplatz als solche mit verkürztem Schlafpensum.

2. Ruheherzfrequenz

Eine erhöhte Ruheherzfrequenz spiegelt häufig chronisch erhöhte Stresslevel wider. In einer Querschnittsstudie war hoher Job-Strain mit erhöhter RHR assoziiert, und gleichzeitig berichten Beschäftigte in stark belastenden Jobs über signifikant geringere Zufriedenheit.

3. Stress (HRV)

Die Herzratenvariabilität (HRV) ist ein objektiver Marker für die autonome Balance: Niedrige HRV-Werte korrelieren konsistent mit höheren Stresslevels am Arbeitsplatz. Da hohe Stresslevels nachweislich die Zufriedenheit verringern, eignet sich HRV-Monitoring als indirektes Mass für potenzielle Unzufriedenheit.

4. Aktivitätsminuten und Schritte

Regelmässige moderate Bewegung, erfasst über Schritte und aktive Minuten, steht in direktem Zusammenhang mit höherer Jobzufriedenheit. Eine aktuelle Studie belegt, dass wöchentliche Freizeitaktivität signifikant positive Effekte auf die Zufriedenheit am Arbeitsplatz hat.

TODO: zu viel?

Diese Kennzahlen erlauben es, Zusammenhänge zwischen erholungsbezogenen Faktoren und der subjektiven Arbeitszufriedenheit zu erkennen und so gezielte Massnahmen zur Förderung des Wohlbefindens und der Leistungsfähigkeit abzuleiten.

5.3 Yappi Coach

5.4 Konzeptevaluation

TODO XENO: Fragebogen

Kapitel 6

Implementierung

6.1 Zugriffskontrolle über API Keys

TODO: Quelle für spring Security Architektur

<https://docs.spring.io/spring-security/reference/servlet/architecture.html>

6.2 Companion Apps

6.2.1 IntelliJ IDEA Companion

6.2.2 Calendar Companion

TODO XENO:

6.2.3 Health Companion

6.3 Deployment

TODO XENO:

TODO: nicht sicher ob es auch ein entsprechendes Kapitel im Konzeptsdesign benötigt

TODO: UML Deployment Diagramm

Kapitel 7

Evaluation

TODO XENO:

Kapitel 8

Quellen

Literatur

Graziotin, D., & Fagerholm, F. (2019). Happiness and the Productivity of Software Engineers. In C. Sadowski & T. Zimmermann (Hrsg.), *Rethinking Productivity in Software Engineering* (S. 109–124). Apress. https://doi.org/10.1007/978-1-4842-4221-6_10