

Inhaltsverzeichnis

0.1	IntelliJ IDEA Companion	2
0.1.1	Architektur und Integration	2
0.1.2	Speicherung der Einstellungen	3
0.1.3	Benutzeroberfläche	4

0.1 IntelliJ IDEA Companion

Die Implementierung nutzt die von JetBrains bereitgestellten IntelliJ-Plugin-APIs, um sich direkt in den Entwicklungsablauf der IDE einzuklinken. Diese APIs ermöglichen es, Ereignisse wie einen erfolgreichen Code-Commit abzufangen, eigene Dialogfenster oder Benachrichtigungen anzuzeigen und auf gespeicherte Plugin-Einstellungen zuzugreifen. Dadurch kann das Yappi Companion-Plugin nahtlos in die bestehende IntelliJ-Oberfläche integriert werden, ohne den gewohnten Workflow der Entwickler zu unterbrechen.

Der IntelliJ Companion erweitert Yappi um die Möglichkeit, direkt aus der Entwicklungsumgebung heraus Feedback zu Code-Commits zu erfassen. Ziel ist es, die Erfassung von Stimmungsdaten möglichst nahtlos in den Entwickler-Workflow zu integrieren, ohne den Arbeitsfluss zu unterbrechen. Nach jedem erfolgreichen Commit in IntelliJ IDEA öffnet sich automatisch ein Dialogfenster, in dem der Entwickler seine Stimmung sowie optional Notizen zum Commit angeben kann. Diese Daten werden anschliessend an Yappi übertragen und dort gespeichert.

0.1.1 Architektur und Integration

Die Abbildung 1 zeigt den Aufbau des Yappi IntelliJ IDEA Companion.

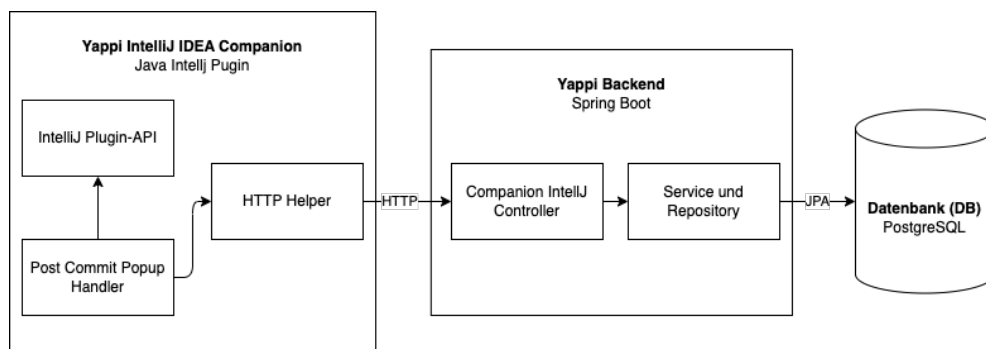


Abbildung 1: Architektur des IntelliJ IDEA Companion

Die Implementierung basiert auf den IntelliJ-Plugin-APIs. Ein (`PostCommitPopupHandler`) registriert sich auf erfolgreiche Commit-Events. Nach Bestätigung des Com-

mits wird der Post-Commit-Dialog asynchron angezeigt, um die Haupt-UI nicht zu blockieren. Der API-Key, der zur Authentifizierung gegenüber Yappi benötigt wird, wird über die Plugin-Einstellungen eingegeben und in einer persistenten Komponente (`CompanionStorage`) gespeichert. Die Kommunikation mit dem Backend erfolgt über eine dedizierte Hilfsklasse (`HttpHelper`), welche die Daten an den entsprechenden Endpoint überträgt.

0.1.2 Speicherung der Einstellungen

Damit sich die Companion App Authentifizieren kann muss ein gültiger API Key hinterlegt werden. Die Eingabe des API-Keys erfolgt über den Menüpunkt Settings → Tools → Yappi Companion in IntelliJ. Die Abbildung 2 zeigt den entsprechenden Menüpunkt.

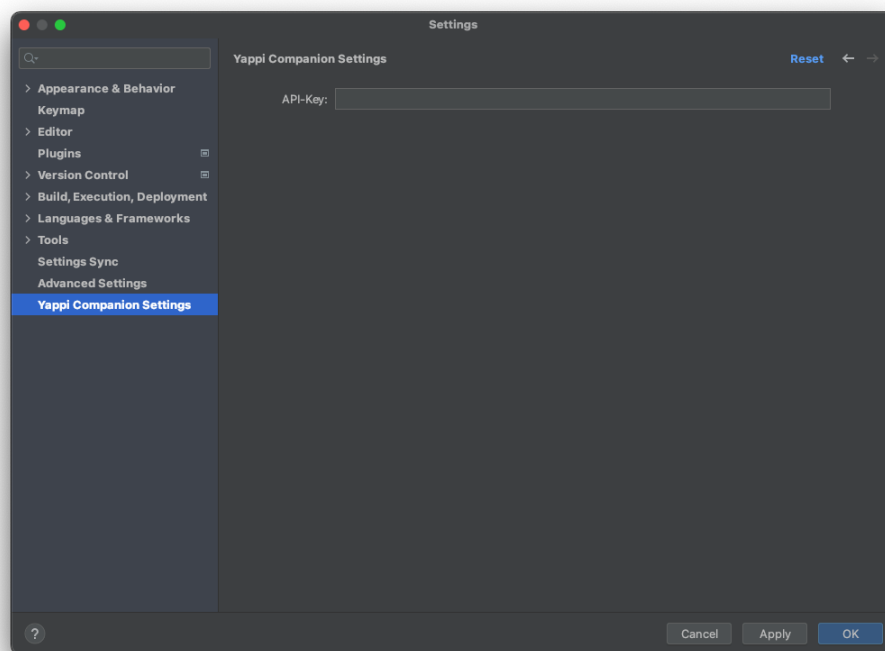


Abbildung 2: Eingabe des API-Keys in den IntelliJ Companion-Einstellungen

Der Key wird in der Klasse `CompanionStorage` mithilfe der IntelliJ-Persistenzmechanismen

gespeichert und beim nächsten Start automatisch geladen.

0.1.3 Benutzeroberfläche

TODO: neues design + text

Die Benutzeroberfläche besteht aus:

- Anzeige der Commit-Nachricht im oberen Bereich
- Auswahlfeld für die Stimmung mit vordefinierten Emojis (Happy, Neutral, Frustriert, Gestresst)
- Eingabefeld für zusätzliche Anmerkungen

Nach Bestätigung werden die Daten an Yappi gesendet, ohne dass weitere Aktionen des Nutzers nötig sind.

TODO: backend