

## Entwicklung eines Plugins zum automatischen Verfolgen von sicherheitskritischen Anforderungen in Softwareprojekten

### Hintergrund

IT-Sicherheit ist bei der Softwareentwicklung ein zentrales Thema und wird, nach einigen Studien und wissenschaftlichen Umfragen, häufig vernachlässigt. Eine bekannte Möglichkeit sicherheitskritische Anforderungen zu verfolgen, ob diese auch eingehalten wurden, bildet die Methode des *Tracings* aus dem Requirements Engineering. Beim Tracing existieren verschiedene Ansätze Anforderung zu verfolgen, wobei diese sich bisher überwiegend auf manuelle oder semi-automatische Ansätze fokussieren. Dabei ist es notwendig, dass Entwickelnde aktiv selbst Anforderungen in Dokumenten oder externer Software einpflegen. Aufgrund dessen geraten Anforderungen (vor allem Sicherheitsanforderungen) häufig in Vergessenheit.

### Aufgabe

In dieser Masterarbeit soll ein semi- bzw. vollautomatischer Ansatz als Prototyp entwickelt werden, welcher es ermöglichen soll, sicherheitskritische Anforderungen zu verfolgen. Dabei soll dieser Prototyp in Form eines Plugins direkt in eine IDE eingesetzt und eine Verknüpfung zwischen Quellcode und Anforderungen erzeugt werden. Die Datenbasis für das (semi-)automatische Verfolgen von Anforderungen soll dabei Opensource Projekte oder ein bereits erstellter Datensatz bilden. Für die Erprobung des Prototyps soll somit ein Datensatz (KI oder analog) erstellt werden, wodurch beim Programmieren automatisch bei Erfassung einer sicherheitskritischen Anforderung das Plugin darauf aufmerksam macht, ob eine sicherheitskritische Anforderung erfüllt ist. Existieren Änderungen während des Programmierens, so soll es möglich Anforderungen abzuändern und wahlweise mit einer Begründung zu erklären. Bsp.: Ein Entwickelnder programmiert derzeit die Verschlüsselung von Nutzerdaten für eine Anwendung. Nachdem dies abgeschlossen wurde, erkennt das Plugin die ursprüngliche Anforderung wieder und fragt, ob die Anforderung dabei abgeschlossen wurde.

1. Einarbeitung in die Literatur zu IT-Sicherheit und Plugin Programmieren
2. Erstellung eines Datensatzes auf Basis eines bestehenden Datensatzes oder eines bestehenden Projekts
3. Entwicklung des Plugins
4. Evaluation und/oder Validierung des Plugins anhand eines definierten Projekts

### Organisatorisches

Betreuer:	Alexander Specht
Erstprüfer:	Prof. Dr. Kurt Schneider
Zweitprüferin:	Dr. Jil Klünder
Beginn:	ab sofort möglich
Randbedingungen:	Kenntnisse von Plugin-Programmierung vom Vorteil