

## **Issuetracking-Systeme**

1. Welche grundsätzlichen Funktionalitäten bietet ein Issuetracking-System (Vorgangsverwaltung)?
2. Welche Typen von Reports (Berichten) gibt es typischerweise?
3. Wie sieht typischerweise der Status-Workflow eines Vorgangs in einem solchen System aus?

## **Prinzipien der SE**

1. Nennen Sie drei Prinzipien der Software-Entwicklung nach Ghezzi et al. (2003)  
Ghezzi, Jazayeri, Mandrioli. Fundamentals of Software Engineering. Prentice Hall, 2nd ed., 2003
2. Was versteht man unter hoher Kopplung und geringer Kohäsion?
3. Welche Zielsetzung ist bspw. mit Modularität verbunden?
4. Welche Zielsetzung ist bspw. mit Abstraktion verbunden?
5. Welche Zielsetzung ist bspw. mit Inkrementalität und iterativem Vorgehen verbunden?

## **Git**

1. Welche Zustände kann eine Datei innerhalb eines Git-Repositories annehmen (Stichwort: "git status lifecycle")?
2. Wie kann ich Dateien aus der Versionierung ausschließen?
3. Was sind typische Beispiele von solchen Dateien?
4. Durch welche Schritte kann ich den Status einer Datei (innerhalb des lifecycle) ändern? (vgl. Abbildung zum "git status lifecycle")
5. Was bedeutet es in Git, eine Datei zu "stagen"?
6. Was bedeutet es in Git, eine Datei zu "committen"?
7. Warum sollte ich als Entwickler auf separaten Branches arbeiten (und nicht direkt auf dem master/main-Branch)?
8. Was versteht man unter "Fast-forward", wenn die Arbeit eines Branches in einen anderen Branch integriert wird?
9. Was passiert bei einem "three-way-merge" (auch "recursive strategy"), wenn die Arbeit eines Branches in einen anderen Branch integriert wird?