



Entwicklungsumgebung

Projekt „Tipp1“

Gruppe 04

Stand: 10.04.2023

Versionierung

| | git | svn | GitLab |
|-------------------------|---|--|---|
| Kosten | Kostenlos | Kostenlos | Kostenlos |
| Merkmale | <ul style="list-style-type: none"> • Unterstützung verteilter Arbeitsabläufe • Sehr hohe Sicherheit gegen sowohl unbeabsichtigte als auch böswillige Verfälschung • Hohe Effizienz | <ul style="list-style-type: none"> • Nutzung mehr im professionellen Bereich, da mehr Kontrolle und Zentralverwaltung möglich ist | <ul style="list-style-type: none"> • Wird als Alternative zu GitHub gesehen • Webanwendung zur Versionskontrolle |
| Netzwerk- verbindung | <ul style="list-style-type: none"> • Keine dauerhafte Netzwerkverbindung notwendig, jeder Nutzer hat lokale Kopie des Codes | <ul style="list-style-type: none"> • Dauerhafte Netzwerkverbindung notwendig, Code und Daten werden zentral abgespeichert | <ul style="list-style-type: none"> • Keine dauerhafte Netzwerkverbindung notwendig, jeder Nutzer hat lokale Kopie des Codes |
| Vorteile | <ul style="list-style-type: none"> • in vielen IDEs integriert • Nutzung von branches lässt Mehrbenutzer Bearbeitung zu | <ul style="list-style-type: none"> • Vergabe von pfadbasierten Zugangsrechten • Verwaltung von Metadaten und Verzeichnissen möglich • Relativ leicht zu erlernen • Daten sind nie komplett gelöscht und können bei Fehlanwendungen sicher wiederhergestellt werden | <ul style="list-style-type: none"> • Keine Einschränkungen an der Anzahl der Repositionen • Integration von mehreren APIs und Drittanbieter Plugins möglich • Einfach einzurichten |
| Nachteile | <ul style="list-style-type: none"> • Schwierig zu erlernen • Verschieben von Daten kann zum Teil zu Chaos im Projekt führen | <ul style="list-style-type: none"> • Dauerhafte Netzwerkverbindung notwendig • Es ist nicht möglich, gleichzeitig an Dateien zu arbeiten | <ul style="list-style-type: none"> • Projekte, an denen nicht mehr regelmäßig gearbeitet wird, werden nach einem Jahr automatisch archiviert • UI ist etwas unintuitiv für Neulinge |

Wir haben uns für Git entschieden. Git ist bereits in vielen gängigen IDEs integriert und wir müssen uns nicht erst um die Implementation kümmern. Ein weiterer Vorteil ist die kostenlose Nutzung.

Die nicht dauerhafte benötigte Netzwerkverbindung ist ein weiterer Vorteil, da bei jedem Teammitglied eine lokale Version als Backup auf dem Rechner existiert und wir somit unabhängig von Servern jederzeit arbeiten können.

UML-Tool mit Quellcodegenerierung

| | visual paradigm | StarUML | Papyrus |
|-----------------------------|---|---|---|
| Kosten / Lizenz | Version Floating für \$449 | Version Commercial für \$129 | kostenlose Erweiterung für Eclipse |
| UML-Version | 2.x | 2.x | 2.x |
| Betriebssystem | Windows, Linux, macOS | Windows, Linux, macOS | Windows, Linux, macOS |
| Quellcodegenerierung in ... | C#, Java, DLL, .NET, VB.NET, PHP, Python, XSD, XML, ActionScript, Objective-C, Ada95, ODL, IDL, Delphi, PERL & Ruby | Java, C#, C++, Python (durch open source Erweiterungen) | Java |
| Sonstige Zusatzinhalte | bietet zusätzlich Tools für: <ul style="list-style-type: none"> • Process Mapping; • UX Design; • Database Engineering; • Dokumentation; • Tabellenanalyse | <ul style="list-style-type: none"> • Lizenz für mehr als 10 User → ab \$10 Rabatt • E-Mail-Support • benutzerdefinierbare Icons der UML-Elemente | <ul style="list-style-type: none"> • Support SysML 1.1 & 1.4 |
| Erfahrung des Teams | Nein | Nein | Ja |

Wir haben uns für die kostenlose Erweiterung unsere IDE Eclipse entscheiden und nutzen Papyrus. Dies bietet uns den Vorteil, erstellte Klassen sofort in unser Projekt einzubinden. Zudem konnten alle Teammitglieder schon Erfahrung mit diesem Tool sammeln.

Build-Tool

| | ant | maven | gradle |
|---------------|--|--|--|
| Kosten | Kostenlos | Kostenlos | Kostenlos |
| Type | Build-Tool | Build-Management-Tool | Build-Management-Tool |
| Sprachen | Java (mit entsprechenden Plugins: C++, C#, Ruby, Python) | Java, C#, Ruby, Python, Scala | Java, Rust, C/C++, Swift, Rust |
| Voraussetzung | Java Version 1.8 oder neuer muss auf Betriebssystem installiert sein | Java Development Kit Version 8 oder neuer muss auf dem Betriebssystem installiert sein | Java Development Kit Version 8 oder neuer muss auf dem Betriebssystem installiert sein |

Wir werden bei unserem Projekt Gradle verwenden, da im direkten Vergleich zu Maven Gradle eine bessere Performance und UI aufweist.
Gegenüber Ant hat Gradle den Vorteil, dass es nicht nur ein Build-Tool ist, sondern auch gleichzeitig ein Build-Management-Tool liefert.

Prototyping

| | Figma | Sketch | Moqups | Pencil |
|--------------------|---------------------|-----------------|-----------------|-----------------------|
| Kosten | Kostenlos | 8,26 €/Monat | 15 €/Monat | Kostenlos |
| Erfahrung | Einfach | Fortgeschritten | Fortgeschritten | Einfach |
| Kollaboration | Ja | Ja | Ja | Nein |
| Entwickler Handoff | Ja | Ja | Ja | Nein |
| Plugins | Ja | Ja | Ja | Ja |
| Betriebssystem | Web, macOS, Windows | macOS | Web | Windows, Linux, macOS |
| Dateiformate | Verschiedene | Verschiedene | Verschiedene | Verschiedene |
| HTML Export | Ja | Ja | Ja | Nein |

Wir werden Figma benutzen, da es übersichtlich gestaltet und leicht zu handhaben ist. Zudem kann man dank der Webanwendung zu jedem Zeitpunkt von überall darauf zugreifen. Was des Weiteren dafür spricht sind die Kosten, da es komplett kostenlos ist.

IDE

| | Eclipse | IntelliJ IDEA | NetBeans |
|---------------------|---|---|--|
| Kosten | Kostenlos (Open Source) | Kostenlos (Community Edition), Kommerzielle Lizenzen verfügbar (Ultimate Edition) | Kostenlos (Open Source) |
| Betriebssystem | Windows, Linux, MacOS | Windows, Linux, MacOS | Windows, Linux, MacOS |
| für Java optimiert | Ja | Ja | Ja |
| Vorteile | <ul style="list-style-type: none"> • Große Community, • zahlreiche Plugins, • gute Integration | <ul style="list-style-type: none"> • Hervorragende Performance, • umfangreiches Plugin-Ökosystem, • viele integrierte Funktionen | <ul style="list-style-type: none"> • Starker Fokus auf Java, • gute Community-Unterstützung, • gute Integration |
| Nachteile | <ul style="list-style-type: none"> • Gelegentliche Performance-Probleme, • weniger Funktionen als IntelliJ IDEA | <ul style="list-style-type: none"> • Kostenpflichtige Ultimate Edition für erweiterte Funktionen | <ul style="list-style-type: none"> • Gelegentliche Performance-Probleme, • weniger Funktionen als IntelliJ IDEA |
| Geeignet für | <ul style="list-style-type: none"> • Java-Projekte • Einsteiger • Open-Source-Projekte | <ul style="list-style-type: none"> • Java-Projekte • große, aufwendige Projekte | <ul style="list-style-type: none"> • Java-Projekte • Einsteiger • Open-Source-Projekte |
| Erfahrung des Teams | Ja | Nein | Nein |

Nachdem wir die verschiedenen IDEs verglichen haben, haben wir uns für Eclipse entschieden, weil es kostenlos und Open Source ist, eine große Community hat und zahlreiche Plugins zur Verfügung stehen. Zudem haben wir bereits Erfahrung mit Eclipse.

Test-Automatisierung

| | JUnit | TestNG | SikuliX |
|------------------------|--|---|---|
| Typ | Unit-Testing | Unit-Testing, Integration-Testing, Functional Testing | GUI-Testing (automatisiertes Testen von grafischen Benutzeroberflächen) |
| Integration | Nahtlose Integration in alle gängigen IDEs | Nahtlose Integration in alle gängigen IDEs | Eher aufwändige Integration |
| Syntax | Einfach und leicht verständlich | Wie JUnit aber mehr Funktionen | etwas komplexer |
| Erweiterbarkeit | Gut, zahlreiche Erweiterungen verfügbar | Gut, zahlreiche Erweiterungen verfügbar | Begrenzt |
| Kosten | Kostenlos (Open Source) | Kostenlos (Open Source) | Kostenlos (Open Source) |
| Sprache | Java | Java | Java |
| Erfahrung des Teams | Ja | Nein | Nein |

Für unser Projekt haben wir uns für JUnit entschieden, weil es ein bewährtes, leicht verständliches Framework für Unit-Tests in Java ist. Es ist kostenlos, Open Source und bietet eine einfache Teststruktur mit Annotationen und Assertions. Im Vergleich zu TestNG ist JUnit leichtgewichtiger und einfacher in der Handhabung, während SikuliX aufgrund seiner Fokussierung auf GUI-Tests für unser Projekt nicht geeignet ist.

Dokumentationstool

| | JavaDoc | Doxygen |
|---------------------|---|---|
| Kosten | Kostenlos | Kostenlos |
| Betriebssystem | Windows, Linux, MacOS | Windows, Linux, MacOS |
| Sprache | Java | C/C++, Java, Lex, Python, C#, PHP, etc... |
| Vorteile | <ul style="list-style-type: none">• Integration mit Entwicklungsumgebungen• Erzeugung von HTML, RTF, XML, PDF und anderen Formaten• Einfache Navigation• Automatische Dokumentation & Überprüfung• HTML 5 Unterstützung | <ul style="list-style-type: none">• Automatische Dokumentation & Überprüfung• Unterstützung für mehrere Programmiersprachen• Cross-Referenzierung• Anpassungsmöglichkeiten für das Aussehen, den Inhalt und die Struktur der generierten Dokumentation |
| Nachteile | <ul style="list-style-type: none">• Erfordert die Verwendung von Kommentaren im Quellcode• Kann nur für Java-Code verwendet werden• Eingeschränkte Anpassungsmöglichkeiten für das Aussehen der generierten Dokumentation | <ul style="list-style-type: none">• PDF erzeugung Problematisch• Schwierigkeiten bei der Erstellung von Dokumentation für komplexe Algorithmen oder Designs• Generierte Dokumentation kann aufgrund von begrenzter Kontrolle über die Kommentare im Code unvollständig oder unklar sein |
| Erfahrung des Teams | Nein | Nein |

Wir werden das JavaDoc Tool nutzen, da dies bereits in Eclipse integriert und leicht zu bedienen ist.

Obfuscator

| | JavaGuard | ProGuard | yGuard |
|-------------|--|--|--|
| Kosten | Kostenlos | Kostenlos | Kostenlos |
| Plattformen | Windows, Mac, Linux | Windows, Android, Linux | Windows, Mac, Linux |
| Sprache | Java | Java, Kotlin | Java |
| Vorteile | <ul style="list-style-type: none">• Schutz vor Reverse Engineering | <ul style="list-style-type: none">• optimiert Code mit bis zu 20% kürzerer Laufzeit• komprimiert Code bis zu 90%• Schutz vor Reverse Engineering• Open source | <ul style="list-style-type: none">• Open source• reduziert Code• Schutz vor Reverse Engineering• Leistungsstarke und einfach zu nutzende XML-Syntax |
| Nachteile | <ul style="list-style-type: none">• letztes Update 2016• vermutlich kein Support der neueren Java-Versionen• kein Online-Support | <ul style="list-style-type: none">• Hauptfunktion ist nicht Schutz vor Reverse Engineering, sondern verkleinern von Apps | <ul style="list-style-type: none">• schlechtere Leistung als ProGuard |

Da wir nicht viel Erfahrung mit Obfuscator haben, entschieden wir uns für ProGuard, da dieses Tool im Vergleich am besten abschneidet. Es hat die besten Vorteile und die wenigsten Nachteile gegenüber den anderen Tools.

Codeconventions

| | Mozilla | Google | Oracle |
|---------------------|---|--|--|
| Lesbarkeit | hoch | mittel - hoch | hoch |
| Benennungen | Klassen - CamelCase mit großem Anfangsbuchstaben Interfaces - wie Klasse Methoden - CamelCase mit großem Anfangsbuchstaben ab zweitem Wortteil Variablen - wie Methoden, Klassenvariablen mit "m" und statische Variablen mit "s" beginnen Konstanten - werden nicht geschrieben | Klassen - CamelCase mit großem Anfangsbuchstaben Interfaces - wie Klassen Methoden - CamelCase mit großem Anfangsbuchstaben ab zweitem Wortteil Variablen - wie Methoden Konstanten - alles großgeschrieben, mit Unterstrichen zur Worttrennung | Klassen - CamelCase mit großem Anfangsbuchstaben Interfaces - wie Klasse Methoden - CamelCase mit großem Anfangsbuchstaben ab zweitem Wortteil Variablen - wie Methoden Konstanten - werden nicht geschrieben |
| Kommentare | verständlich und so wenig Kommentare wie möglich | verständlich und so wenig Kommentare wie möglich | verständlich und so wenig Kommentare wie möglich |
| Zeilenformattierung | Zeile nicht länger als 100 Zeichen 4 Leerzeichen zum Einrücken | Zeile nicht länger als 100 Zeichen 2 Leerzeichen zum Einrücken | Zeile nicht länger als 80 Zeichen, 4 Leerzeichen zum Einrücken, 8 Leerzeichen für Tab |

Wir haben uns aufgrund von besserer Lesbarkeit und einfacherer Wartung für die Java Coding Conventions von Oracle entschieden.

Kollaborationstool

| | Discord | Notion (Teamspace) | Trello |
|----------------------------|---|---|--|
| Kosten | kostenlos (Nitro für \$9.99 pro Monat/User) | Version Plus für \$8 pro Monat/User (mit Einschränkungen auch kostenlos) | kostenlos (Version Standard für \$6 pro Monat/User) |
| Chat-Funktion | Ja | Nein | Nein |
| Video/Voice-Call | Ja | Nein | Nein |
| Datenaustausch/-ordnung | limitierter Austausch keine wirkliche Strukturierung | unbegrenzt leichtes Organisieren | unbegrenzt leichtes Organisieren |
| Termin-/Aufgabenverwaltung | unübersichtlich umsetzbar | schnell und übersichtlich | durch App-Integration sehr leicht |

Neben zweiwöchentlichen persönlichen Meetings arbeiten wir bei diesem Projekt mit den Diensten Discord und Trello. Discord bietet uns eine schnelle und einfache Kommunikation via Chat oder Voice-Calls. Trello dagegen wird zur Organisation aller Dokumente sowie die weiterer Termin- /Aufgabenverwaltung verwendet.

Quellen

Versionierung

- <https://de.wikipedia.org/wiki/Git>
- <https://www.simplilearn.com/tutorials/git-tutorial/what-is-gitlab#:~:text=Very%20easy%20to%20set%20up,APIs%20and%20third%2Dparty%20services>
- <https://www.atlassian.com/git/tutorials/why-git#:~:text=One%20of%20the%20biggest%20advantages,every%20change%20to%20your%20codebase.>
- https://de.wikipedia.org/wiki/Apache_Subversion
- <https://de.wikipedia.org/wiki/GitLab>

UML-Tool mit Quellcodegenerierung

- <https://www.visual-paradigm.com/>
- <https://staruml.io/>
- <https://www.eclipse.org/papyrus/index.php>

Build-Tool

- https://en.wikipedia.org/wiki/Apache_Ant
- <https://ant.apache.org/index.html>
- <https://maven.apache.org/index.html>
- https://en.wikipedia.org/wiki/Apache_Maven
- <https://en.wikipedia.org/wiki/Gradle>
- <https://gradle.org/>

Prototyping

- <https://www.figma.com/de/>
- <https://www.sketch.com/>
- <https://pencil.evolus.vn/Features.html>
- <https://moqups.com/de/#tour>

Dokumentationstool

- <https://www.doxygen.nl/manual/index.html><https://www.doxygen.nl/manual/index.html>
- <https://www.baeldung.com/javadoc>

Obfuscator

- <https://www.guardsquare.com/proguard>
- <https://www.yworks.com/products/yguard>
- <https://sourceforge.net/projects/javaquard/>

Codeconventions

- <https://google.github.io/styleguide/javaguide.html>
- <https://www.oracle.com/java/technologies/javase/codeconventions-contents.html>
- https://firefox-source-docs.mozilla.org/code-quality/coding-style/coding_style_java.html

Kollaborationstool

- <https://www.notion.so/de-de/product>
- <https://trello.com/home>