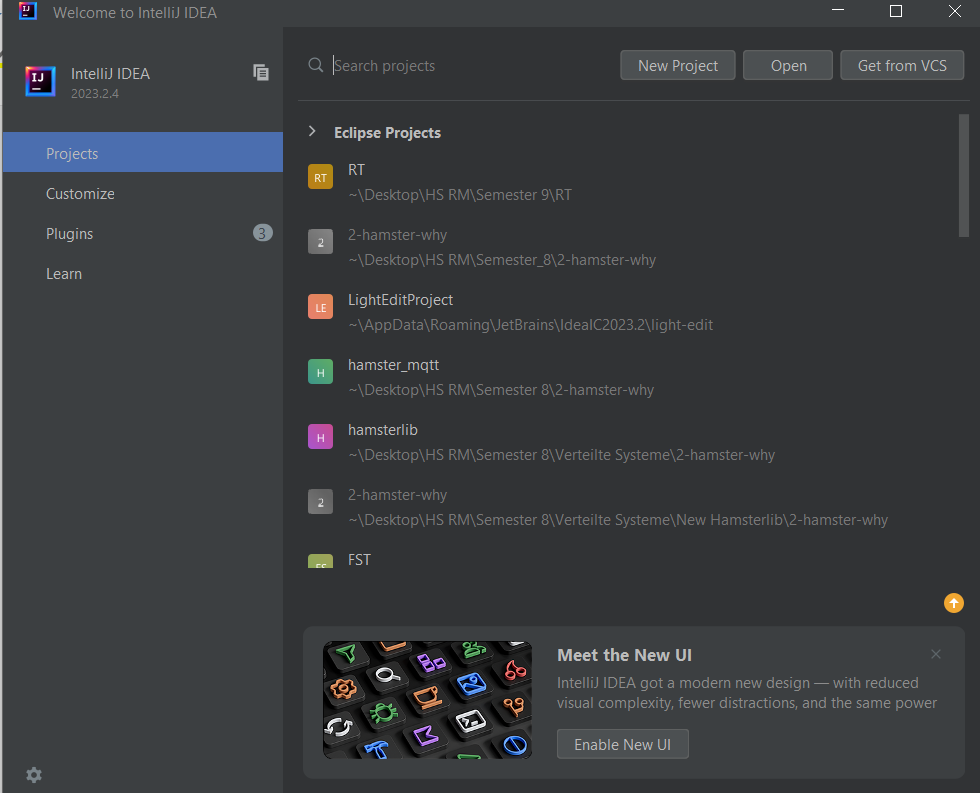
**Nr. 1**



1. Code Auto-Completion

2. Auto-Indent

3. Error Detection

4. Run Code with 1 click

5. Debug with 1 click

6. Easy to switch between classes

7. Version Control Integration

8. Easy refactoring

9. Terminal Commands in DIE

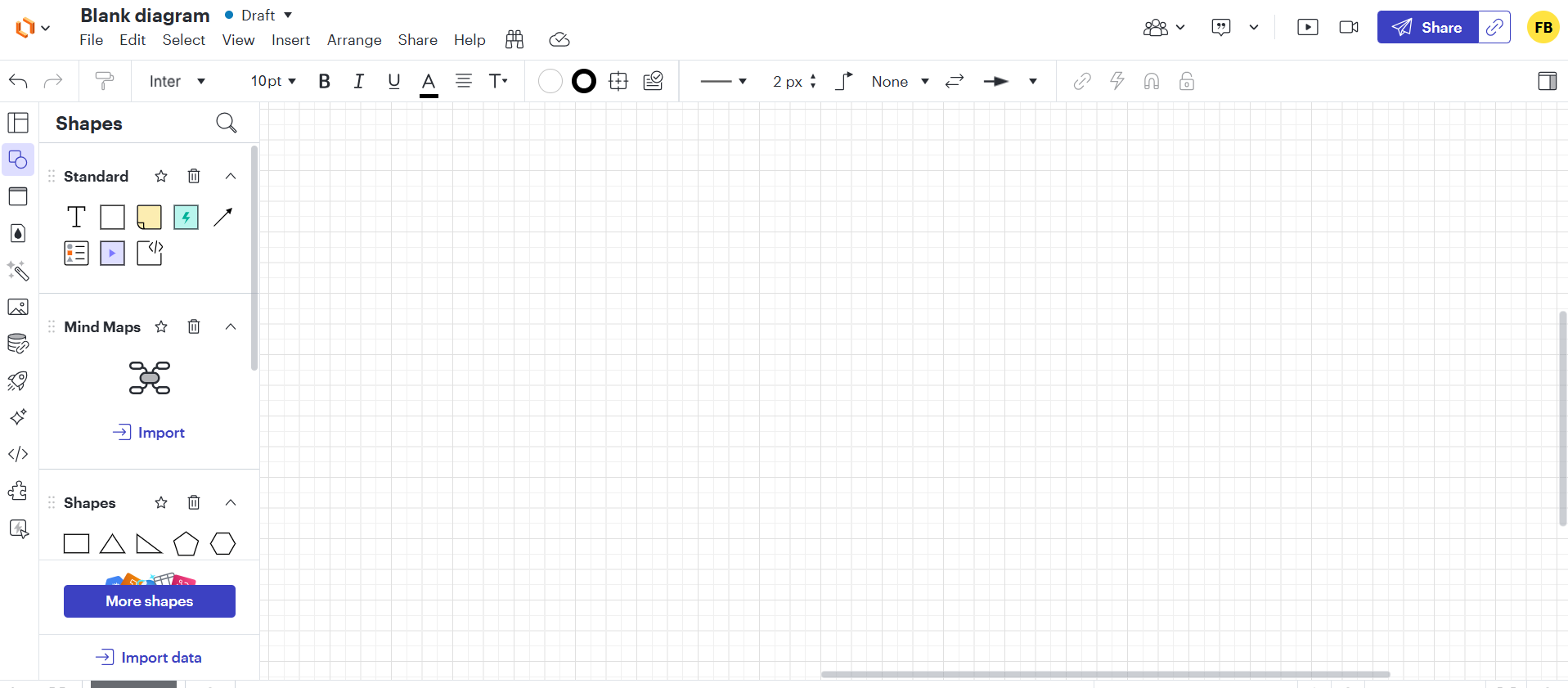
10. Easy Code Navigation

**Nr. 2 + 3**

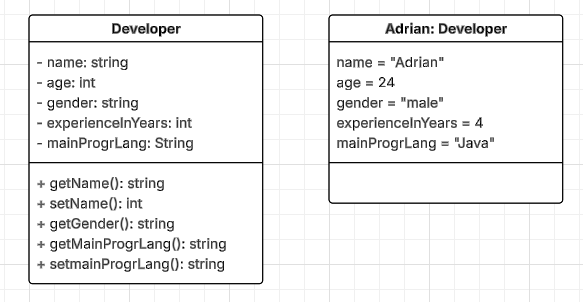
Siehe Github: AdrianHenkelUAS/SWED1

**Nr. 4**

Lucid Chart (Unterstützt UML 2.0)



**Nr. 5**



<https://lucid.app/lucidchart/3eda4911-0192-4dd9-a8c6-ceef8a33c2e6/edit?viewport_loc=-1892%2C-591%2C2558%2C1130%2C0_0&invitationId=inv_3916ee22-1b50-4787-ac64-9681b245df12>

Code: Siehe Java in Github

**Nr. 6**

Software Design: Die Entwicklung des Konzepts der Softwaresystems, welches dann später implementiert wird

Software Analysis: Beobachten von bisher geschriebener Software um Fehler, Performance etc. zu überprüfen

**Nr. 7**

Software-Design wird bei allen größeren Projekten benötigt. Dies vereinfacht die Zusammenarbeit, unterstützt das Verstehen und hilft dabei schnell Fehler zu finden und auch Fehler zu vermeiden.

Wenn man diesen Schritt auslässt, kann man leicht etwas übersehen oder man merkt erst nach der Implementierung, dass der Code erst gar nicht funktioniert.

**Nr. 8**

Diese 4 Bereiche sind abhängig voneinander.

Beispiel: Die Datenbank gibt über eine API Daten her. Diese Daten werden dann von dem Interface aufgegriffen und angezeigt.