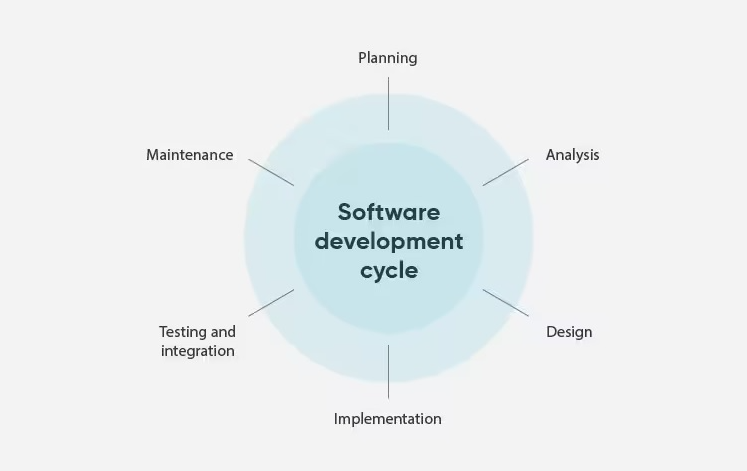
**Exploring Software engineering tools**



**Teorētisks projekts:** E-komercijas tīmekļa lietotne, kur lietotāji var pārlūkot produktus, pievienot tos grozam un veikt drošas maksājumu transakcijas.

**Plānošana:** Izmantot Jira, kas palīdz plānot projektu, ļaujot komandai izveidot uzdevumus, piešķirt tos konkrētiem cilvēkiem, noteikt termiņus un sekot progresam. Tas palīdz saglabāt projekta kārtību un nodrošina, ka katrs komandas loceklis zina savus uzdevumus un laika grafiku.

**Analīze:** Lucidchart vai Draw.io var izveidot plūsmas diagrammas, datu plūsmas diagrammas un sistēmas arhitektūras diagrammas. Tādā veidā var vizualizēt sistēmas datu apstrādes un mijiedarbības procesus, līdz ar to vieglāk atklāt iespējamās kļūdas jau agri.

**Dizains:** Ar Figma vai Adobe XD var veidot prototipus, saskarnes dizainus un interaktīvus maketus, kuras pēc tam ieinteresētas personas var pārskatīt un apstiprināt pirms izstrādes uzsākšanas.

**Izstrāde:** Visual Studio Code ir labs koda redaktors, kas piedāvā paplašinājumus gandrīz visām programmēšanas valodām un integrāciju ar Git. Tas padara koda rakstīšanu, testēšanu un labošanu efektīvāku.

**Testēšana un integrācija:** Selenium vai JUnit automatizēti testēšanas rīki, kas pārbauda tīmekļa funkcionalitāti vai testē programmatūras kodu. Tas nozīmē, ka kļūdas tiek atklātas ātri, veicot atkārtojamus testus automātiski.

**Uzturēšana:** GitHub vai GitLab Rīki, ko izmanto versiju kontrolei, nepārtrauktai integrācijai un sadarbībai. Šie rīki ļauj sekot izmaiņām kodā, kas atvieglo kļūdu labošanas procesu vai atgriešanos pie iepriekšējām versijām.