

Dalam dunia pengembangan perangkat lunak, ada peran yang disebut sebagai pengembang full stack. Ini berarti pengembangan perangkat lunak dilakukan dari awal hingga akhir. Ini mencakup bagian depan (front-end), bagian belakang (back-end), dan sisi klien (client-side). Dalam pengembangan full stack, ada beberapa lingkup penting yang perlu diperhatikan, termasuk pengembangan front-end, pengembangan back-end, manajemen basis data, integrasi front-end ke back-end, kontrol versi dan kolaborasi, serta pengembangan mobile. Dasar-dasar pengembangan front-end web melibatkan bahasa seperti HTML, CSS, dan JavaScript. Beberapa kerangka kerja (framework) yang populer untuk pengembangan full stack web termasuk React, Vue.js, dan Angular.js.

Sementara itu, pengembangan back-end melibatkan pemrograman dalam bahasa seperti JavaScript, Python, PHP, dan lainnya. Beberapa kerangka kerja server yang digunakan adalah Express.js dan Flask. Dalam hal manajemen basis data, Anda akan menemui SQL (seperti MySQL) dan NoSQL (seperti MongoDB). Selain itu, dalam pengembangan mobile full stack, Anda akan menggunakan lingkungan pengembangan terintegrasi (IDE) untuk mengedit kode, dan kerangka kerja seperti React dan Flutter.

Dalam siklus pengembangan aplikasi end-to-end, terdapat beberapa tahapan, seperti perencanaan dan analisis, desain, pengembangan front-end, pengembangan back-end, integrasi dan pengujian, serta pemeliharaan dan peningkatan. Kolaborasi adalah elemen penting dalam rekayasa perangkat lunak, oleh karena itu, alat kendali versi (version control) digunakan untuk memfasilitasi kolaborasi yang efektif. Salah satu aplikasi version control yang populer adalah GitHub.

Alat-alat yang digunakan dalam pengembangan perangkat lunak meliputi editor kode (seperti Visual Studio Code), version control (seperti GitHub), manajemen basis data (seperti MySQL), pengujian API (seperti POSTMAN), pengujian dan debugging (seperti Junit), pengembangan mobile (seperti React), komputasi awan (seperti AWS), CI/CD (Continuous Integration/Continuous Deployment, seperti Jenkins), dan desain antarmuka pengguna (UI/UX) menggunakan alat seperti Figma.

SDLC (Software Development Life Cycle) adalah serangkaian proses terstruktur dan metodologi yang digunakan dalam pengembangan perangkat lunak dari awal hingga selesai. SDLC terdiri dari empat tahap utama, yaitu perencanaan dan analisis, desain, pengembangan, pengujian, penerapan, dan pemeliharaan. Manfaat SDLC meliputi prediktabilitas, peningkatan kualitas perangkat lunak, manajemen risiko yang lebih baik, efisiensi tim dan kolaborasi, pemenuhan kebutuhan pengguna, penghematan biaya dan waktu, perbaikan dokumentasi, serta peningkatan pengawasan dan evaluasi.

Dalam SDLC, ada beberapa model pengembangan yang dapat digunakan, seperti model waterfall, model v-shaped, prototipe, spiral, iteratif inkremental, big bang, dan agile. Dalam proyek perangkat lunak, penggunaan alat kendali versi (version control) adalah suatu keharusan. Git adalah salah satu alat kendali versi yang sangat kuat dan populer. Dasar-dasar perintah Git meliputi git init, git clone, git status, git add, git commit, git push, git pull, git branch, git checkout, git merge, git log, git remote, git fetch, git diff, dan git reset. Alat kendali versi digunakan untuk mengurangi konflik dalam pengembangan perangkat lunak dan memastikan kelancaran proses.