

Pengembangan Full Stack (Full Stack Development) merujuk pada pengembangan seluruh aplikasi secara end-to-end, dari sisi depan (front-end) hingga sisi belakang (back-end) dan, dalam beberapa kasus, hingga sisi klien (client-side)

- Front End : Membangun antarmuka pengguna yang menarik dan interaktif menggunakan HTML, CSS, dan JavaScript. Menggunakan framework dan pustaka front-end, seperti React, Angular, Vue.js, atau jQuery, untuk mempercepat pengembangan dan meningkatkan efisiensi.
- Back End : Membangun server dan aplikasi yang berfungsi sebagai "otak" dari aplikasi, menerima permintaan dari sisi depan, memproses data, dan memberikan respons yang sesuai. Menggunakan bahasa pemrograman server-side seperti Node.js, Python, Ruby, Java, PHP, atau C#
- Database Management : Mendesain dan mengelola basis data untuk menyimpan, mengambil, dan memanipulasi data aplikasi. Menggunakan teknologi database seperti MySQL, PostgreSQL, MongoDB, atau Firebase
- Integration of Front-End and Back-End : Menghubungkan komponen front-end dengan layanan back-end melalui API (Application Programming Interface) untuk berkomunikasi dengan server dan database. Menyelaraskan data dan tampilan antara sisi depan dan sisi belakang aplikasi.
- Version Control and Collaboration : Menggunakan sistem pengendalian versi, seperti Git, untuk mengelola perubahan kode dan kolaborasi dalam tim pengembang. Memastikan kode terus berkembang dengan aman dan sesuai dengan tujuan proyek.
- Mobile Development : Beberapa Pengembang Full Stack juga memiliki kemampuan untuk mengembangkan aplikasi mobile menggunakan framework seperti React Native, Flutter
- Dasar-dasar Frontend web : CSS , HTML , JAVASCRIPT
- Popular Frontend Framework Full Stack Web Development : Vue.js , Angular.js , React
- Dasar-Dasar Backend Development :
 1. Bahasa Pemrograman Server-Side Bahasa pemrograman seperti Node.js (JavaScript), Python, Ruby, Java, PHP, C#, dan lain-lain, digunakan untuk menulis kode di sisi server.
 2. Server Framework Framework seperti Express.js untuk Node.js, Flask untuk Python, Ruby on Rails untuk Ruby, Spring untuk Java, dan Laravel untuk PHP
 3. Database Management Jenis database yang umum digunakan adalah SQL (MySQL, PostgreSQL, SQL Server) dan NoSQL (MongoDB, Firebase)
- Tahap tahap pengembangan aplikasi end-to-end :
 1. Perencanaan dan Analisis
 2. Desain
 3. Pengembangan Front-End
 4. Pengembangan Back-End
 5. Integrasi dan pengujian
 6. Pemeliharaan dan peningkatan
- Manfaat Version Control untuk Berkolaborasi Setiap kali pengembang membuat perubahan pada kode, sistem version control merekam detail perubahan tersebut Rekam Perubahan Version control memungkinkan tim untuk melihat riwayat lengkap dari semua perubahan yang terjadi pada proyek dari awal hingga saat ini. Pencatatan Riwayat Ketika dua atau lebih pengembang melakukan perubahan pada area kode yang sama, version control membantu mengidentifikasi dan menyelesaikan konflik Pemecahan Konflik Version control memungkinkan pengembang untuk memulihkan kode ke versi sebelumnya jika ada masalah atau bug yang terjadi, sehingga mengurangi risiko kehilangan pekerjaan.
- Tools sets Sebagai Full Stack Developer :
 1. IDE -Code Editor
 2. Version Control -Repository
 3. DBMS
 4. API
 5. TEST & DEBUGGING
 6. LAYANAN CLOUD
 7. CI/CD
 8. DESAIN UI/UX(figma,etc)