

Nayla Putri Cahya Ramadani

2024071020

Mata Kuliah Software Engineering

Tugas Evolution of Software

1. 1960s – Batch Processing Systems

Pada tahun 1960-an, software digunakan dengan sistem batch processing, yaitu program dijalankan secara berkelompok tanpa interaksi langsung dari pengguna. User harus memasukkan data terlebih dahulu, lalu komputer memproses semuanya sekaligus, dan hasilnya baru bisa dilihat setelah proses selesai. Sistem ini lambat dan tidak fleksibel karena pengguna tidak bisa langsung berinteraksi dengan komputer.

2. 1980s – Personal Computing

Pada tahun 1980-an, muncul komputer pribadi (PC) yang bisa digunakan oleh individu, bukan hanya perusahaan besar. Software mulai dibuat agar bisa digunakan langsung oleh user, seperti aplikasi pengolah kata dan spreadsheet. Hal ini membuat komputer menjadi lebih umum digunakan di rumah, sekolah, dan kantor.

3. 1990s – Client–Server Systems

Pada tahun 1990-an, software mulai menggunakan sistem client–server, yaitu ada komputer server yang menyediakan layanan dan komputer client yang mengakses layanan tersebut. Contohnya seperti sistem database di kantor, di mana banyak komputer terhubung ke satu server pusat. Sistem ini memungkinkan banyak user menggunakan software secara bersamaan.

4. 2000s – Web-Based Applications

Pada tahun 2000-an, software mulai berbasis web dan bisa diakses melalui browser menggunakan internet. User tidak perlu menginstal aplikasi, cukup membuka website untuk menggunakan software tersebut. Contohnya seperti email berbasis web dan sistem informasi online.

5. 2010s – Mobile & Cloud Computing

Pada tahun 2010-an, software berkembang ke perangkat mobile seperti smartphone dan menggunakan teknologi cloud computing. Data dan aplikasi tidak lagi hanya disimpan di satu komputer, tetapi bisa diakses dari mana saja melalui internet. Hal ini membuat software lebih fleksibel dan mudah diakses kapan saja dan di mana saja.

6. 2020s – AI-Driven Intelligent Systems

Pada tahun 2020-an, software mulai menggunakan teknologi Artificial Intelligence (AI) yang membuat sistem menjadi lebih pintar. Software dapat belajar dari data, mengenali

pola, dan membantu mengambil keputusan secara otomatis. Contohnya seperti chatbot, sistem rekomendasi, dan sistem pengenalan wajah.

Kesimpulan

Seiring waktu, software menjadi semakin kompleks dan canggih, dari yang awalnya hanya memproses data secara sederhana hingga menjadi sistem pintar berbasis AI. Karena kompleksitasnya meningkat, diperlukan software engineering agar software dapat dibuat dengan terstruktur, stabil, dan sesuai kebutuhan pengguna.

Tugas Sizing The Problem

Velocity estimate may need to be adjusted (based on observed progress)

Velocity adalah perkiraan kecepatan tim dalam menyelesaikan pekerjaan dalam periode tertentu. Pada awalnya, velocity hanya berupa estimasi, sehingga bisa saja tidak sesuai dengan kondisi nyata di lapangan. Oleh karena itu, velocity perlu disesuaikan berdasarkan progress yang benar-benar terjadi, agar perencanaan waktu menjadi lebih akurat.

However, the total duration can be computed quickly

Meskipun velocity bisa berubah, total waktu yang dibutuhkan untuk menyelesaikan proyek tetap bisa dihitung dengan cepat. Hal ini karena kita sudah memiliki estimasi ukuran pekerjaan dan perkiraan kecepatan tim. Dengan informasi tersebut, kita bisa memperkirakan kapan proyek akan selesai.

Provided that the relative size estimates of parts are accurate

Perhitungan durasi akan lebih akurat jika estimasi ukuran setiap bagian pekerjaan sudah tepat. Jika kita bisa memperkirakan mana tugas yang kecil, sedang, atau besar dengan benar, maka perhitungan waktu penyelesaian proyek akan lebih mendekati kenyataan.

Easier to achieve if the parts are small and similar-size

Estimasi akan lebih mudah dan akurat jika pekerjaan dibagi menjadi bagian-bagian kecil dengan ukuran yang hampir sama. Bagian kecil lebih mudah dipahami, diperkirakan, dan dikerjakan dibandingkan tugas yang terlalu besar. Hal ini juga membantu tim memantau progress dengan lebih jelas.