

PEMBELAJARAN DASAR-DASAR PENGEMBANGAN PERANGKAT LUNAK DAN GIM

Hari, Tanggal: Jumat, 24 Juli 2026

Topik : Budaya Kerja, Pengenalan DDPPLG, dan Profesi IT

Tujuan Pembelajaran : Menganalisis ruang lingkup pembelajaran Dasar-Dasar Pengembangan Perangkat Lunak & Gim, serta profesi bidang PPLG.

A. Budaya Kerja di Bidang IT

Budaya kerja di bidang IT (Teknologi Informasi) sangat lekat dengan **kolaborasi tim, fleksibilitas tinggi, dan siklus inovasi yang cepat**. Praktiknya menekankan pada transparansi, otonomi individu, pembelajaran berkelanjutan, serta penggunaan metodologi tangkas (*Agile*) untuk menyelesaikan proyek secara iteratif dan efektif.

Beberapa karakteristik utama dan pilar pembentuk budaya kerja IT meliputi:

- **Metodologi Agile dan Scrum:** Mengubah cara kerja konvensional melalui rapat harian singkat (*Daily Stand-up*), perencanaan mingguan/dua mingguan (*Sprint*), dan evaluasi berkelanjutan. Hal ini memastikan setiap masalah teknis ditangani cepat dan adaptif terhadap perubahan.
- **Fleksibilitas (*Remote & Hybrid Work*):** Pekerjaan IT sering kali berorientasi pada hasil (keluaran) bukan kehadiran fisik. Banyak perusahaan menerapkan kerja dari rumah (WFH) atau *hybrid* yang memungkinkan talenta bekerja dari mana saja, bahkan lintas negara.
- **Kolaborasi dan Keterbukaan (*Open Communication*):** Tim teknis dan non-teknis bekerja sama dengan batasan hierarki yang cenderung cair. Keterbukaan sangat krusial, misalnya dalam melakukan *Code Review* atau bertukar ide saat *Brainstorming* untuk menyelesaikan *bug* maupun fitur produk.
- **Pembelajaran Tanpa Henti (*Continuous Learning*):** Mengingat tren teknologi seperti AI, *Cloud*, dan bahasa pemrograman terus berkembang, pekerja IT dituntut untuk selalu menguasai *skill* baru dan adaptif.

Dalam praktiknya, budaya kerja ini didukung oleh penggunaan berbagai *tools* kolaborasi seperti Jira untuk pelacakan proyek dan Slack untuk komunikasi internal yang lebih efisien. Selain itu, terdapat penekanan yang kuat pada **etika profesi**, khususnya mengenai privasi dan keamanan data pengguna.

B. Mengenal Mata Pelajaran Dasar-Dasar Pengembangan Perangkat Lunak dan Gim (DDPPLG)

Awal Perjalanan Menjadi Pengembang Perangkat Lunak

Selamat di DDPPLG. Dengan belajar mata pelajaran ini, kita telah memulai langkah untuk mempelajari cara teknologi diciptakan dan dimanfaatkan untuk membantu kehidupan manusia. Setiap hari kita menggunakan berbagai aplikasi, seperti WhatsApp, Google Maps, YouTube, Instagram, aplikasi perbankan, toko daring, hingga sistem informasi sekolah. Pernahkah Anda bertanya, *siapa yang membuat aplikasi-aplikasi tersebut? Bagaimana proses pembuatannya?*

Jawabannya adalah para pengembang perangkat lunak (*software developer*). Mereka bekerja sama dalam sebuah tim untuk merancang, membangun, menguji, dan mengembangkan aplikasi yang digunakan oleh jutaan orang. Melalui mata pelajaran **Dasar-Dasar Pengembangan Perangkat Lunak dan Gim (DDPPLG)**, Anda akan mempelajari fondasi pengetahuan dan keterampilan yang diperlukan untuk menjadi bagian dari profesi tersebut.

Namun, perlu dipahami sejak awal bahwa belajar di jurusan PPLG **bukan hanya belajar menulis kode program (coding)**. Coding memang penting, tetapi itu hanyalah salah satu bagian dari keseluruhan proses pengembangan perangkat lunak. Seorang pengembang perangkat lunak juga harus mampu memahami kebutuhan pengguna, berpikir logis, bekerja sama dalam tim, mengelola data, menguji aplikasi, menjaga keamanan sistem, dan terus belajar mengikuti perkembangan teknologi.

Dengan kata lain, DDPPLG akan mengajarkan Anda cara berpikir dan cara bekerja sebagai seorang profesional di bidang teknologi informasi.

Mengapa Tidak Cukup Hanya Bisa Coding?

Bayangkan Anda diminta membuat sebuah rumah. Apakah pekerjaan tersebut hanya membutuhkan tukang yang mampu memasang batu bata? Tentu tidak. Sebelum rumah dibangun, seseorang harus mengetahui kebutuhan pemilik rumah, membuat gambar rancangan, menghitung bahan bangunan, menyusun jadwal pekerjaan, melakukan pembangunan, memeriksa hasilnya, hingga memastikan rumah tersebut aman untuk ditempati.

Membuat perangkat lunak juga memiliki proses yang hampir sama. Menulis kode hanyalah salah satu tahap pembangunan. Sebelum itu, seorang pengembang harus mengetahui apa yang dibutuhkan pengguna. Setelah aplikasi selesai dibuat, aplikasi tersebut masih harus diuji, diperbaiki, dipelihara, bahkan dikembangkan lagi sesuai kebutuhan.

Oleh karena itu, selama belajar di DDPPLG, Anda akan mempelajari seluruh proses tersebut secara bertahap.

Berpikir Komputasional: Belajar Menyelesaikan Masalah

Kemampuan pertama yang akan dipelajari adalah **berpikir komputasional (Computational Thinking)**. Berpikir komputasional bukan berarti berpikir seperti komputer, melainkan berpikir secara runtut, logis, dan sistematis dalam menyelesaikan masalah.

Sebagai contoh, bayangkan Anda diminta membuat secangkir teh. Kemungkinan besar Anda akan melakukan langkah-langkah berikut.

1. Menyiapkan gelas.
2. Memasukkan teh.
3. Memanaskan air.
4. Menuangkan air panas.
5. Menambahkan gula.
6. Mengaduk teh.

Urutan langkah tersebut merupakan cara berpikir yang sistematis. Jika urutannya diubah, misalnya menuangkan air sebelum menyiapkan gelas, tentu hasilnya tidak akan sesuai. Dalam dunia perangkat lunak, setiap masalah juga harus diselesaikan dengan langkah-langkah yang jelas. Kemampuan inilah yang menjadi dasar bagi semua pengembang perangkat lunak.

Algoritma: Menyusun Langkah Penyelesaian

Setelah memahami cara berpikir yang sistematis, Anda akan mempelajari **algoritma**. Algoritma adalah urutan langkah-langkah untuk menyelesaikan suatu masalah. Sebagai contoh, ketika ingin mencari buku di perpustakaan, Anda mungkin melakukan langkah berikut.

1. Datang ke perpustakaan.
2. Mencari nomor rak.
3. Menuju rak tersebut.
4. Mencari judul buku.
5. Mengambil buku.

Urutan langkah tersebut merupakan sebuah algoritma. Dalam pengembangan aplikasi, algoritma digunakan sebelum menulis program agar solusi yang dibuat lebih terstruktur dan mudah dipahami.

Pemrograman: Menerjemahkan Ide Menjadi Program

Setelah memiliki algoritma, langkah berikutnya adalah **pemrograman (programming)**. Pemrograman merupakan proses menerjemahkan algoritma ke dalam bahasa yang dapat dipahami komputer.

Bahasa pemrograman dapat diibaratkan sebagai bahasa komunikasi antara manusia dan komputer. Jika manusia menggunakan bahasa Indonesia atau bahasa Inggris, komputer menggunakan bahasa pemrograman seperti Python, Java, JavaScript, C#, PHP, atau bahasa lainnya. Pada tahap ini Anda akan belajar membuat program sederhana hingga mampu membangun aplikasi yang lebih kompleks secara bertahap.

Basis Data: Tempat Menyimpan Informasi

Sebuah aplikasi tidak akan berguna jika tidak dapat menyimpan data. Misalnya, aplikasi perpustakaan harus menyimpan data anggota dan buku. Aplikasi sekolah harus menyimpan data peserta didik, guru, nilai, serta presensi. Aplikasi toko harus menyimpan data barang dan transaksi.

Semua informasi tersebut disimpan di dalam **basis data (database)**. Basis data dapat diibaratkan seperti lemari arsip yang tersusun rapi. Dengan penyimpanan yang baik, data dapat dicari, diperbarui, maupun dihapus dengan mudah ketika diperlukan.

Desktop, Website, dan Aplikasi Mobile

Saat ini perangkat lunak hadir dalam berbagai bentuk. **Aplikasi desktop** adalah aplikasi yang dipasang langsung pada komputer, misalnya Microsoft Word, Excel, atau aplikasi kasir di sebuah toko. **Website** merupakan aplikasi yang dijalankan melalui peramban (*browser*), seperti portal berita, media sosial, sistem informasi sekolah, atau toko daring.

Sementara itu, **aplikasi mobile** dirancang khusus untuk telepon pintar dan tablet, misalnya aplikasi perbankan, layanan transportasi daring, atau aplikasi pembelajaran. Walaupun bentuknya berbeda, ketiganya memiliki tujuan yang sama, yaitu membantu pengguna menyelesaikan suatu pekerjaan dengan lebih mudah dan efisien.

UI dan UX: Membuat Aplikasi yang Nyaman Digunakan

Pernahkah Anda menggunakan aplikasi yang tampilannya membingungkan sehingga sulit mencari menu yang dibutuhkan? Sebaliknya, mungkin Anda juga pernah menggunakan aplikasi yang tampilannya sederhana tetapi sangat mudah digunakan. Hal tersebut berkaitan dengan **UI (User Interface)** dan **UX (User Experience)**.

UI adalah tampilan yang dilihat pengguna, seperti warna, ikon, tombol, dan tata letak. UX adalah pengalaman pengguna ketika menggunakan aplikasi, misalnya apakah menu mudah ditemukan, prosesnya cepat, dan aplikasi nyaman digunakan. Aplikasi yang baik tidak hanya memiliki banyak fitur, tetapi juga mudah dipahami oleh penggunanya.

Pengujian Aplikasi: Memastikan Semua Berjalan Baik

Setelah aplikasi selesai dibuat, pekerjaan belum selesai. Aplikasi harus diuji terlebih dahulu untuk memastikan semua fitur berjalan sesuai dengan kebutuhan.

Sebagai contoh, jika sebuah aplikasi presensi digunakan di sekolah, pengembang harus memastikan bahwa peserta didik dapat melakukan presensi, guru dapat melihat laporan, dan data tersimpan dengan benar. Jika ditemukan kesalahan, pengembang akan memperbaikinya sebelum aplikasi digunakan oleh banyak orang. Proses inilah yang disebut **pengujian aplikasi (software testing)**.

Version Control: Bekerja Bersama dalam Tim

Di dunia kerja, sebuah aplikasi umumnya tidak dibuat oleh satu orang saja. Sebuah tim dapat terdiri atas puluhan bahkan ratusan pengembang. Agar pekerjaan setiap anggota tim tidak saling bertabrakan, digunakan **Version Control**.

Version Control dapat diibaratkan seperti riwayat revisi pada sebuah dokumen. Setiap perubahan dicatat sehingga apabila terjadi kesalahan, perubahan tersebut dapat ditelusuri atau dikembalikan ke versi sebelumnya. Kemampuan bekerja sama menggunakan Version Control merupakan salah satu keterampilan penting yang harus dimiliki oleh seorang pengembang perangkat lunak.

Cloud Computing: Komputasi Tanpa Batas Tempat

Saat ini banyak aplikasi tidak lagi bergantung pada satu komputer saja. Data disimpan pada server di internet sehingga dapat diakses dari mana saja menggunakan berbagai perangkat.

Sebagai contoh, ketika Anda menyimpan dokumen di Google Drive, dokumen tersebut tetap dapat dibuka melalui komputer lain atau telepon pintar selama terhubung ke internet. Teknologi seperti ini disebut **Cloud Computing**. Cloud Computing memungkinkan aplikasi menjadi lebih fleksibel, mudah diakses, dan mampu melayani banyak pengguna secara bersamaan.

Artificial Intelligence: Ketika Komputer Dapat Belajar

Beberapa tahun terakhir, perkembangan **Artificial Intelligence (AI)** berlangsung sangat pesat. AI merupakan teknologi yang memungkinkan komputer melakukan tugas yang biasanya membutuhkan kecerdasan manusia, seperti mengenali wajah, menerjemahkan bahasa, memahami suara, atau memberikan rekomendasi.

Saat ini AI telah digunakan pada berbagai bidang, termasuk pendidikan, kesehatan, industri, pertanian, dan transportasi. Sebagai calon pengembang perangkat lunak, Anda tidak hanya akan menjadi pengguna AI, tetapi juga berpeluang mengembangkan aplikasi yang memanfaatkan teknologi tersebut.

Keamanan Aplikasi: Melindungi Data Pengguna

Bayangkan jika data nilai sekolah dapat diubah oleh orang yang tidak bertanggung jawab, atau akun pengguna dapat diakses tanpa izin. Situasi seperti itu menunjukkan pentingnya **keamanan aplikasi (Application Security)**.

Keamanan aplikasi bertujuan melindungi data, menjaga kerahasiaan informasi, serta memastikan aplikasi tetap berjalan dengan baik meskipun menghadapi berbagai ancaman. Di era digital, keamanan menjadi salah satu aspek yang tidak dapat dipisahkan dari pengembangan perangkat lunak.

Semua Saling Berhubungan

Mungkin Anda bertanya, “Mengapa saya harus mempelajari begitu banyak materi?” Jawabannya sederhana. Semua materi tersebut merupakan bagian dari satu proses yang utuh.

Sebuah aplikasi biasanya diawali dengan adanya suatu masalah yang ingin diselesaikan. Pengembang kemudian menggunakan berpikir komputasional untuk memahami masalah tersebut. Setelah itu disusun algoritma sebagai solusi, lalu diterjemahkan menjadi program menggunakan bahasa pemrograman. Program tersebut memanfaatkan basis data untuk menyimpan informasi, kemudian disajikan dalam bentuk aplikasi desktop, website, atau aplikasi mobile.

Agar mudah digunakan, aplikasi dirancang dengan prinsip UI dan UX yang baik, kemudian diuji sebelum digunakan. Selama proses pengembangan, tim menggunakan Version Control untuk bekerja sama. Ketika aplikasi sudah digunakan oleh banyak pengguna, Cloud Computing membantu menyediakan layanan yang stabil, AI dapat ditambahkan untuk memberikan fitur yang lebih cerdas, dan keamanan aplikasi menjaga agar data pengguna tetap terlindungi.

Dengan memahami hubungan antarbagian tersebut, Anda akan melihat bahwa pengembangan perangkat lunak bukan sekadar kegiatan menulis kode, melainkan sebuah proses yang melibatkan banyak pengetahuan dan keterampilan.

C. Pengenalan Profesi Bidang Rekayasa Perangkat Lunak

Pendahuluan: Menjelajahi Dunia Kerja di Era Digital

Setelah memahami berbagai aspek dalam mata pelajaran Dasar-Dasar Pengembangan Perangkat Lunak dan Gim (DDPPLG), langkah selanjutnya adalah mengenali **profesi-profesi yang tersedia** di bidang rekayasa perangkat lunak. Memahami berbagai profesi ini penting agar kamu, sebagai siswa pemula, dapat merencanakan arah belajar dan menentukan minat yang ingin kamu tekuni di masa depan.

Bidang rekayasa perangkat lunak merupakan salah satu sektor yang paling dinamis dan menjanjikan di abad ke-21. Setiap aplikasi, website, atau sistem digital yang kamu gunakan sehari-hari adalah hasil kerja kolaboratif dari berbagai profesi di baliknya. Tidak ada satu orang yang mengerjakan semuanya sendiri. Sebaliknya, sebuah tim yang terdiri dari spesialis berbeda bekerja sama untuk menciptakan produk yang berkualitas.

A. Berbagai Profesi di Bidang Rekayasa Perangkat Lunak

Berikut adalah tabel profesi-profesi utama di bidang rekayasa perangkat lunak beserta tugas utamanya, dilengkapi dengan penjelasan dan analogi sederhana untuk memudahkan pemahaman.

No.	Profesi	Tugas Utama
1	Software Developer	Membuat aplikasi berbasis desktop maupun sistem perangkat lunak secara umum.
2	Web Developer	Membuat dan mengembangkan website yang dapat diakses melalui browser.
3	Mobile Developer	Membuat aplikasi yang berjalan pada perangkat mobile, seperti Android dan iOS.
4	Game Developer	Membuat permainan digital (game) untuk berbagai platform.

No.	Profesi	Tugas Utama
5	UI Designer	Mendesain tampilan visual antarmuka aplikasi agar menarik dan konsisten.
6	UX Designer	Mendesain pengalaman pengguna agar aplikasi mudah dan nyaman digunakan.
7	Database Administrator	Mengelola, merawat, dan mengamankan basis data organisasi atau perusahaan.
8	Quality Assurance (QA)	Menguji aplikasi untuk memastikan kualitas dan menemukan kesalahan (bug).
9	System Analyst	Menganalisis kebutuhan pengguna dan merancang spesifikasi sistem yang akan dibangun.
10	DevOps Engineer	Mengelola proses deployment (peluncuran) aplikasi dan infrastruktur pendukungnya.
11	AI Engineer	Mengembangkan sistem kecerdasan buatan yang mampu belajar dari data.
12	Cloud Engineer	Mengelola layanan dan infrastruktur komputasi awan (<i>cloud computing</i>).
13	Cyber Security Specialist	Mengamankan sistem dan aplikasi dari ancaman serta serangan siber.

B. Penjelasan Detail Setiap Profesi

1. Software Developer

Software Developer adalah profesi yang paling umum dikenal. Tugasnya adalah **membangun perangkat lunak** yang berjalan di komputer desktop atau sistem operasi tertentu. Perangkat lunak yang dibuat bisa berupa aplikasi akuntansi, sistem manajemen sekolah, atau program pengolah data.

Analogi: Bayangkan kamu adalah arsitek dan tukang bangunan sekaligus yang membangun sebuah rumah tinggal dari nol—mulai dari membuat denah, membangun fondasi, hingga menyelesaikan interior rumah tersebut.

2. Web Developer

Web Developer fokus pada **pembuatan dan pengembangan website**. Website yang dibuat bisa bersifat statis (hanya menampilkan informasi) maupun dinamis (dapat berinteraksi dengan pengguna dan basis data). Web Developer terbagi menjadi *Front-End Developer* (yang menangani tampilan) dan *Back-End Developer* (yang menangani logika dan data di balik layar).

Analogi: Web Developer ibarat pemilik toko online yang tidak hanya mendesain etalase toko agar menarik, tetapi juga mengelola gudang barang, sistem pembayaran, dan pengiriman di belakangnya.

3. Mobile Developer

Mobile Developer membuat **aplikasi yang berjalan di perangkat genggam**, seperti ponsel pintar dan tablet. Aplikasi yang dibuat bisa untuk sistem operasi Android (menggunakan Kotlin atau Java) atau iOS (menggunakan Swift).

Analogi: Jika Web Developer membangun toko yang bisa dikunjungi siapa saja melalui jalan raya (browser), Mobile Developer membangun toko kecil yang bisa dibawa pengguna ke mana saja di dalam saku celananya.

4. Game Developer

Game Developer adalah profesi yang menggabungkan **seni, teknologi, dan kreativitas** untuk menciptakan permainan digital. Game Developer tidak hanya menulis kode, tetapi juga bekerja sama dengan desainer grafis, komposer musik, dan penulis naskah untuk menciptakan pengalaman bermain yang menyenangkan.

Analogi: Game Developer ibarat sutradara film yang tidak hanya menulis skenario, tetapi juga mengatur tata lighting, suara, akting, dan efek visual agar penonton (pemain) terhibur dan terlibat secara emosional.

5. UI Designer (User Interface Designer)

UI Designer bertanggung jawab atas **tampilan visual** aplikasi atau website. Mereka menentukan warna, tipografi, ikon, tombol, dan tata letak setiap elemen di layar. Tujuannya adalah membuat antarmuka yang estetik, konsisten, dan sesuai dengan identitas merek produk.

Analogi: UI Designer ibarat dekorator interior yang memilih warna cat dinding, model sofa, dan penataan lampu agar ruangan terlihat indah dan harmonis.

6. UX Designer (User Experience Designer)

UX Designer fokus pada **pengalaman pengguna** saat berinteraksi dengan aplikasi. Mereka memastikan bahwa alur penggunaan aplikasi intuitif, tidak membingungkan, dan mencapai tujuan pengguna dengan efisien. UX Designer sering melakukan riset, membuat *wireframe*, dan menguji prototipe dengan pengguna nyata.

Analogi: UX Designer ibarat perancang taman bermain anak-anak. Bukan hanya taman yang indah dilihat, tetapi juga aman, mudah dijangkau, dan setiap wahana bisa dinikmati tanpa instruksi yang rumit.

7. Database Administrator (DBA)

Database Administrator bertugas **mengelola basis data** organisasi. Mereka memastikan data tersimpan dengan aman, dapat diakses dengan cepat, dan terlindungi dari kerusakan atau kehilangan. DBA juga bertanggung jawab atas *backup* (cadangan data) dan pemulihan data jika terjadi masalah.

Analogi: DBA ibarat kepala perpustakaan yang mengatur sistem katalog, memastikan setiap buku ada di tempatnya, melindungi koleksi dari kerusakan, dan memastikan siapa saja yang berhak bisa meminjam buku tertentu.

8. Quality Assurance (QA)

Quality Assurance adalah profesi yang bertugas **menguji aplikasi** sebelum dirilis ke pengguna. QA mencari *bug* (kesalahan), menguji berbagai skenario penggunaan, dan memastikan aplikasi berjalan sesuai spesifikasi. Mereka adalah "penjaga gerbang" kualitas produk.

Analogi: QA ibarat inspektur kendaraan yang memeriksa mobil baru sebelum dijual ke konsumen. Mereka menguji rem, lampu, mesin, dan berkendara di berbagai kondisi untuk memastikan mobil aman dan layak jalan.

9. System Analyst

System Analyst adalah jembatan antara **pengguna (klien) dan tim pengembang**. Mereka mengumpulkan kebutuhan pengguna, menganalisis proses bisnis yang ada, dan menerjemahkannya menjadi spesifikasi teknis yang bisa dipahami programmer. System Analyst harus memahami baik dunia bisnis maupun teknologi.

Analogi: System Analyst ibarat penerjemah profesional yang duduk di antara dua orang yang berbicara bahasa berbeda. Mereka memastikan pesan dari satu pihak tersampaikan dengan tepat kepada pihak lain.

10. DevOps Engineer

DevOps Engineer bertugas **mengelola proses peluncuran (deployment) aplikasi** dari lingkungan pengembangan ke lingkungan produksi (yang digunakan pengguna nyata). Mereka juga mengelola infrastruktur server, otomatisasi proses, dan memastikan aplikasi dapat diakses secara stabil dan cepat.

Analogi: DevOps Engineer ibarat koordinator logistik konser musik. Mereka memastikan peralatan tiba tepat waktu, panggung siap, sound system berfungsi, dan seluruh pertunjukan berjalan lancar dari awal hingga akhir.

11. AI Engineer

AI Engineer mengembangkan **sistem kecerdasan buatan** yang mampu belajar dari data dan membuat keputusan. Contoh hasil kerja AI Engineer adalah sistem pengenalan wajah, chatbot pintar, rekomendasi produk, dan kendaraan otonom.

Analogi: AI Engineer ibarat guru privat yang mengajarkan komputer untuk "belajar" dari contoh-contoh yang diberikan. Semakin banyak contoh yang diberikan, semakin pintar komputer dalam mengenali pola dan membuat prediksi.

12. Cloud Engineer

Cloud Engineer mengelola **layanan dan infrastruktur komputasi awan**, seperti Amazon Web Services (AWS), Google Cloud Platform (GCP), atau Microsoft Azure. Mereka membantu perusahaan menggunakan sumber daya komputer melalui internet tanpa perlu membeli perangkat keras sendiri.

Analogi: Cloud Engineer ibarat manajer penyewaan gedung perkantoran. Alih-alih perusahaan membeli tanah dan membangun gedung sendiri, mereka menyewa ruang kantor yang sudah lengkap dengan listrik, internet, dan keamanan—dan Cloud Engineer memastikan semua fasilitas itu berjalan dengan baik.

13. Cyber Security Specialist

Cyber Security Specialist bertugas **melindungi sistem, jaringan, dan data** dari ancaman siber. Mereka membangun pertahanan, memantau aktivitas mencurigakan, dan merespons jika terjadi serangan atau kebocoran data.

Analogi: Cyber Security Specialist ibarat penjaga keamanan dan sistem alarm di sebuah museum. Mereka memasang kamera, mengatur akses ruangan, memantau gerak-gerik mencurigakan, dan bertindak cepat jika ada percobaan pembobolan.

C. Profesi Tambahan yang Perlu Dikenali

Selain tiga belas profesi di atas, terdapat beberapa profesi lain yang juga penting dalam ekosistem rekayasa perangkat lunak, yaitu:

No.	Profesi	Tugas Utama
14	Technical Writer	Menulis dokumentasi teknis, panduan pengguna, dan manual aplikasi.
15	Product Manager	Merencanakan roadmap produk, mengelola prioritas fitur, dan memastikan produk sesuai kebutuhan pasar.
16	Data Scientist	Menganalisis data dalam jumlah besar untuk menemukan pola dan wawasan bisnis.

No.	Profesi	Tugas Utama
17	Scrum Master	Memfasilitasi tim pengembang agar bekerja secara efisien menggunakan metodologi Agile/Scrum.
18	IT Support / Helpdesk	Memberikan bantuan teknis kepada pengguna yang mengalami masalah dengan aplikasi atau perangkat.
19	Network Engineer	Merancang dan mengelola infrastruktur jaringan komputer dalam organisasi.
20	Embedded Systems Developer	Mengembangkan perangkat lunak untuk perangkat keras khusus, seperti mesin ATM, alat medis, atau kendaraan.

D. Perbedaan antara UI Designer dan UX Designer

Banyak pemula yang sering keliru membedakan UI Designer dan UX Designer. Padahal, keduanya memiliki fokus yang berbeda meskipun saling melengkapi.

Aspek	UI Designer	UX Designer
Fokus Utama	Tampilan visual dan estetika	Pengalaman dan kenyamanan pengguna
Pertanyaan yang Dijawab	"Apakah aplikasi ini terlihat menarik?"	"Apakah aplikasi ini mudah digunakan?"
Output Kerja	Mockup, desain grafis, style guide	Wireframe, user flow, hasil riset pengguna
Analogi	Mendesain tampilan etalase toko	Merancang alur belanja agar pelanggan tidak bingung

Dalam praktiknya, di perusahaan kecil atau startup, satu orang sering menangani UI dan UX sekaligus. Namun, di perusahaan besar, kedua peran ini biasanya dipegang oleh spesialis yang berbeda.

E. Jalur Karier dan Peningkatan Level

Di dunia rekayasa perangkat lunak, karier tidak berhenti pada level pemula. Setiap profesi memiliki jenjang karier yang jelas, umumnya sebagai berikut:

Level	Jabatan Umum	Deskripsi
Junior	Junior Developer, Junior Designer, Junior QA	Pemula dengan pengalaman 0–2 tahun, masih banyak belajar dan dibimbing.
Middle	Developer, Designer, QA Engineer	Memiliki pengalaman 2–5 tahun, mampu mengerjakan tugas secara mandiri.
Senior	Senior Developer, Senior Designer, Lead QA	Pengalaman 5+ tahun, mampu memimpin fitur kompleks dan membimbing junior.
Lead / Manager	Tech Lead, Engineering Manager, Product Manager	Memimpin tim, membuat keputusan teknis, dan mengelola proyek.
Executive	CTO (Chief Technology Officer), VP of Engineering	Level eksekutif yang menentukan arah teknologi perusahaan.

F. Peluang Karier Lulusan IT

Lulusan jurusan IT, khususnya RPL, memiliki **peluang karier yang sangat luas**. Kemampuan yang kamu pelajari di SMK tidak hanya dibutuhkan oleh perusahaan teknologi, tetapi juga oleh hampir semua sektor industri di era digital ini.

Berikut adalah berbagai tempat kerja yang dapat menjadi tujuan:

No.	Sektor / Tempat Kerja	Penjelasan
1	Software House	Perusahaan yang khusus mengerjakan proyek pembuatan perangkat lunak untuk klien. Contoh: membuat aplikasi kasir untuk toko, website untuk sekolah, atau sistem informasi untuk pemerintah.
2	Startup	Perusahaan teknologi muda yang berkembang pesat, biasanya fokus pada inovasi produk digital. Bekerja di startup menawarkan pengalaman dinamis dan kesempatan berkontribusi langsung pada pertumbuhan perusahaan.
3	Perusahaan Nasional	Perusahaan besar di Indonesia yang memiliki divisi teknologi informasi, seperti perusahaan telekomunikasi, e-commerce, atau manufaktur.
4	Pemerintahan	Instansi pemerintah membutuhkan tenaga ahli IT untuk mengelola sistem informasi publik, e-government, dan keamanan data nasional.
5	Perbankan	Bank dan lembaga keuangan sangat bergantung pada sistem perangkat lunak untuk transaksi, mobile banking, dan keamanan data nasabah.
6	Rumah Sakit	Fasilitas kesehatan modern menggunakan sistem informasi rumah sakit (SIMRS), rekam medis elektronik, dan aplikasi telemedicine.

No.	Sektor / Tempat Kerja	Penjelasan
7	Sekolah / Perguruan Tinggi	Lembaga pendidikan membutuhkan tenaga IT untuk mengelola sistem informasi akademik, e-learning, dan infrastruktur teknologi.
8	Freelancer	Bekerja secara mandiri menerima proyek dari berbagai klien, baik lokal maupun internasional. Keuntungannya adalah fleksibilitas waktu dan potensi penghasilan yang tinggi.
9	Wirausaha Digital	Membangun produk digital sendiri, seperti aplikasi, game, atau platform online, dan menjalankannya sebagai bisnis mandiri.

G. Mengapa Lulusan RPL Dibutuhkan di Berbagai Sektor?

Era digital telah mengubah cara hampir semua organisasi beroperasi. Tidak peduli apakah sebuah rumah sakit, bank, atau sekolah—semuanya kini membutuhkan sistem perangkat lunak untuk:

1. **Meningkatkan efisiensi operasional** (otomatisasi tugas-tugas manual).
2. **Meningkatkan pelayanan kepada pengguna** (aplikasi yang responsif dan mudah diakses).
3. **Mengamankan data dan informasi** (perlindungan dari ancaman siber).
4. **Mengambil keputusan berbasis data** (analisis data untuk strategi bisnis).
5. **Bersaing di pasar digital** (kehadiran online melalui website dan aplikasi).

Oleh karena itu, keahlian yang kamu pelajari di jurusan PPLG bukan hanya relevan untuk perusahaan teknologi, tetapi menjadi **kebutuhan fundamental di hampir setiap sektor kehidupan modern**.

H. Keterampilan Pendukung yang Perlu Dikembangkan

Selain kemampuan teknis (hard skills), lulusan RPL yang sukses juga perlu mengasah **keterampilan non-teknis (soft skills)** berikut:

Keterampilan	Penjelasan
Komunikasi	Mampu menyampaikan ide teknis dengan bahasa yang mudah dimengerti orang awam.
Kerja Sama Tim	Perangkat lunak dibangun oleh tim, bukan individu. Kemampuan berkolaborasi sangat penting.
Manajemen Waktu	Mampu menyelesaikan tugas sesuai tenggat waktu (deadline).
Pemecahan Masalah	Mampu menganalisis masalah secara kritis dan menemukan solusi yang efektif.
Kemauan Belajar Terus-Menerus	Teknologi berkembang sangat cepat. Yang dipelajari hari ini bisa saja usang dalam beberapa tahun.
Kreativitas	Mampu menciptakan solusi inovatif dan produk yang berbeda dari yang sudah ada.