

Studi Independen Bersertifikat

Dicoding Indonesia X Kampus Merdeka

Batch 3

PT. Dicoding Akademi Indonesia

Profil Dicoding

Dicoding secara resmi diluncurkan tanggal 5 Januari 2015 untuk menjembatani developer Indonesia dengan kebutuhan dan permintaan pasar yang semakin kompetitif. Dicoding hadir sebagai platform pendidikan teknologi yang membantu menghasilkan talenta digital berstandar global. Semua demi mengakselerasi Indonesia agar menjadi yang terdepan. Saat ini, lebih dari 570 ribu developer dan calon developer telah tergabung di Dicoding. 360 ribu individu pembelajar telah dan sedang terdaftar dalam lebih dari 100 kelas yang disediakan oleh Dicoding

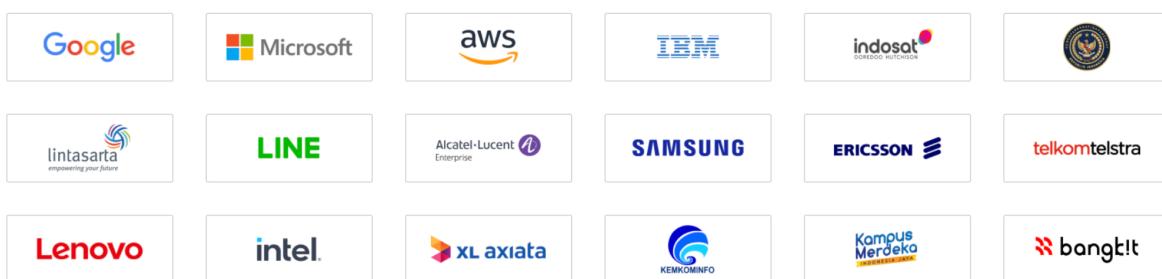
 578.962 Developer aktif yang telah bergabung	 111 Kelas yang telah tersedia	 1250 Event terselenggara di seluruh Indonesia	 488 Job untuk developer yang telah dibuka	 203 Challenge yang telah diselenggarakan
---	--	--	--	---

Saat ini, Dicoding bermitra dengan perusahaan teknologi kelas dunia. Dicoding juga merupakan [Google Authorized Training Partner](#) dan memiliki komitmen kemitraan dengan pemilik teknologi, perusahaan multinasional, Kementerian/Lembaga Pemerintahan, serta perusahaan dengan skala nasional. Dicoding juga adalah mitra penyelenggara:

- Bangkit by Google, Goto, Traveloka,
- Indosat Ooredoo Hutchinson Digital Camp (IDCamp),
- Lintasarta Digischool,
- Kemenparekraf/Baparekraf Digital Talent,
- Cloud and Back-End Developer Scholarship Program with content from AWS,

Serta telah menjadi mitra program Studi Independen Kampus Merdeka dari Kemendikbudristek sejak tahun 2021 (batch pertama).

Telah dipercaya oleh



Program Studi Independen Bersertifikat Batch Ketiga - 2022

Dicoding Indonesia berkomitmen untuk menjalankan program studi independen berkualitas tinggi dengan topik/tema:

1. Pengembang Aplikasi Android (kuota: 400 siswa)
2. Pengembang Aplikasi Android dan Back-End (kuota: 200 siswa)
3. Pengembang Front-End Web dan Back-End (kuota: 600 siswa)
4. Pengembang Front-End Web dan React (kuota: 300 siswa)
5. Pengembang Machine Learning dan Front-End Web (kuota: 500 siswa)
6. Pengembang Machine Learning dan DevOps (kuota: 100 siswa)
7. Pengembang Multi-Platform dan Back-End (kuota: 300 siswa)
8. Pengembang Back-End dan DevOps (kuota: 200 siswa)
9. Pengembang Back-End dan React (kuota: 200 siswa)
10. Pengembang Multi-Platform dan Aplikasi Android (kuota: 200 siswa)

pada program Studi Independen Kampus Merdeka 2022 Batch 3 meliputi proses seleksi, pengelolaan siswa, serta manajemen proyek akhir dengan total:

- 3000 Siswa
- 500 Proyek Akhir (proyeksi 6 siswa per grup)
- 120 Orang Pembimbing/mentor (20 jam / minggu)
Rasio 1 Pembimbing : 25 mahasiswa

Proses Seleksi

1. Registrasi di platform SIB MBKM (<https://kampusmerdeka.kemdikbud.go.id>)
2. Mendaftar ulang melalui form yang telah disediakan oleh tim Dicoding (Data yang perlu diinput mencakup Nama, Asal Universitas, Email aktif, dan nomor telepon yang aktif).
3. Ujian Seleksi / Filtering di Dicoding, meliputi:
 - a. Pengetahuan Teknologi Dasar (Tech), dengan nilai 1-100
 - b. Kepribadian dan pola pikir (Traits), dengan nilai 1-100
4. Siswa yang memenuhi persyaratan harus melengkapi tahapan berikut
 - a. Melengkapi identitas, cakupan pengetahuan dan validasi data yang telah diisi di Kampus Merdeka (data diri, mata kuliah yang telah diambil, validasi semester, validasi prodi, dll)
 - b. Mengumpulkan surat persetujuan dosen pembimbing terkait dengan komitmen konversi SKS dari universitas dan status mahasiswa saat ini (atau surat rekomendasi)
 - c. Mengumpulkan Surat Komitmen Individu Mahasiswa (atau SPTJM)

Peserta juga dapat mengkonversi atau mendapatkan pengakuan hingga 20 SKS, yang bervariasi untuk setiap peserta - bergantung sisa SKS peserta dan persetujuan dosen pembimbing. Peserta yang mengikuti SIB Dicoding 2022 Batch 3, tidak diperkenankan untuk yudisium/dinyatakan lulus program pendidikan diploma 3-4 atau strata 1 sebelum 31 Desember 2022 atau sedang mengikuti kegiatan lain (kampus merdeka, merdeka mengajar, magang, kerja praktek, KKN, program pascasarjana atau pekerjaan part/full-time).

Sesuai arahan tim SIB MBKM, **SKS yang didapatkan dari program ini wajib dikonversi dan jumlah SKS dikonversi harus semaksimal mungkin** (mendekati 20 SKS). Hal ini disarankan, mengingat dari program serupa **dengan konversi rendah, peserta tidak maksimal dalam mengikuti program Kampus Merdeka**. Namun demikian, tim Dicoding masih memberikan toleransi untuk pengambilan SKS reguler di kampus dalam jumlah terbatas (1-6 SKS) mengingat adanya kebutuhan SKS Wajib, atau SKS yang harus diulang oleh peserta.

Pembiayaan

Program Studi Independen ini bersifat **tidak berbayar** terhadap mahasiswa/peserta. Seluruh biaya atas program ini telah ditanggung oleh DIKTI, Kemdikbudristek RI. **SPP/UKT peserta tetap dibayarkan ke Universitas asal**.

Prasyarat Keikutsertaan

Prasyarat Administratif

Seluruh paket

1. Memenuhi ketentuan umum program Studi Independen Kampus Merdeka pada saat pelaksanaan program.
2. Mahasiswa aktif D3/D4/S1 semester 5/7/9/11/13 pada saat program dilaksanakan (Juli-Desember 2022).
3. Tidak mengambil program Kampus Merdeka lainnya pada saat pelaksanaan program.
4. Tidak mengambil KKN/PKL/internship/magang/pekerjaan apapun (part-time ataupun full-time) pada saat pelaksanaan program.
5. Tidak memiliki komitmen paruh/penuh waktu terkait organisasi, volunteership, leadership, atau aktivitas program lainnya pada saat pelaksanaan program.
6. Tidak sedang mengambil program pascasarjana.
7. Telah mendapatkan persetujuan dosen pembimbing untuk mendapatkan pengakuan/konversi SKS melalui program ini.
8. Mengambil 6 SKS atau kurang pada universitas asal (kuliah reguler) pada saat pelaksanaan program.
9. Belum akan lulus dari universitas pada tanggal 31 Desember 2022.

Prasyarat Pengetahuan/Pengalaman

Paket	Prasyarat
<ul style="list-style-type: none"> ● Paket Pengembang Aplikasi Android ● Paket Pengembang Multi-Platform dan Aplikasi Android ● Paket Pengembang Aplikasi Android dan Back-End ● Paket Pengembang Multi-Platform dan Back-End 	<ol style="list-style-type: none"> 1. Memiliki pengalaman dengan object-oriented programming (OOP) pada bahasa pemrograman apapun, dibuktikan melalui mata kuliah yang diambil, transkrip, atau sertifikat kompetensi/kelulusan.
<ul style="list-style-type: none"> ● Paket Pengembang Machine Learning dan Front-End Web ● Paket Pengembang Machine Learning dan DevOps 	<ol style="list-style-type: none"> 1. Lulus mata kuliah terkait Matematika Dasar, Kalkulus, dan Statistika, dibuktikan melalui mata kuliah yang diambil atau transkrip. 2. Memiliki pengalaman pemrograman dengan bahasa pemrograman apapun, dibuktikan melalui mata kuliah yang diambil, transkrip, atau sertifikat kompetensi/kelulusan
<ul style="list-style-type: none"> ● Paket Pengembang Front-End Web dan Back-End ● Paket Pengembang Front-End Web dan React ● Paket Pengembang Back-End dan DevOps ● Paket Pengembang Back-End dan React 	<ol style="list-style-type: none"> 1. Memiliki pengalaman pemrograman dengan bahasa pemrograman apapun, dibuktikan melalui mata kuliah yang diambil, transkrip, atau sertifikat kompetensi/kelulusan.

Prasyarat Teknis

Paket	Prasyarat
Semua Paket	<p>1. Koneksi internet kabel/wifi yang memadai atau selular minimal 4G - cukup untuk melaksanakan pembelajaran tatap muka dalam kondisi kamera menyala sepanjang pembelajaran.</p>
<ul style="list-style-type: none"> • Paket Pengembang Aplikasi Android • Paket Pengembang Aplikasi Android dan Back-End, • Paket Pengembang Multi-Platform dan Aplikasi Android • Paket Pengembang Multi-Platform dan Back-End 	<p>1. Perangkat komputer atau laptop dengan spesifikasi minimal:</p> <ol style="list-style-type: none"> Prosesor setara Core i3 generasi kedua dan RAM / Memory 8GB dengan dukungan 64-Bit (disarankan prosesor setara Core i5 dengan RAM / Memory 8GB atau lebih tinggi). Sistem Operasi Linux, Windows 8/10/11, atau Mac OS X dengan dukungan 64 bit. Dapat menjalankan Android Studio Arctic Fox (2020.3.1) atau lebih baru dan IntelliJ Idea IDE. <p>2. Ponsel Android dengan OS Android 8 Oreo atau lebih baru, atau laptop/komputer harus dapat menjalankan emulator Android.</p>
<ul style="list-style-type: none"> • Paket Pengembang Machine Learning dan Front-End Web • Paket Pengembang Machine Learning dan DevOps • Paket Pengembang Front-End Web dan Back-End • Paket Pengembang Front-End Web dan React • Paket Pengembang Back-End dan DevOps • Paket Pengembang Back-End dan React 	<p>1. Perangkat komputer atau laptop dengan spesifikasi minimal:</p> <ol style="list-style-type: none"> Prosesor setara Dual Core dan RAM / Memory 2GB (disarankan prosesor setara Core i3 dengan RAM / Memory 4GB atau lebih tinggi). Sistem Operasi Linux, Windows, atau Mac OS.

Mekanisme Pembelajaran

Studi Independen Bersertifikat ini bertujuan untuk menghasilkan talenta berstandar tinggi yang sesuai dengan standar Industri. Proses pembelajaran yang dilakukan adalah online learning, dimana peserta harus mengimplementasikan materi yang diperolehnya secara langsung melalui project dan tugas-tugas yang harus diselesaikan untuk menyelesaikan setiap materinya.

Materi diberikan secara asynchronous (online melalui modul belajar di Dicoding Academy) dan akan di-review setiap interval waktu tertentu oleh pembimbing non-akademik dan expert. Selain project dan tugas, pemberian materi juga akan dilengkapi dengan kuis dan atau ujian pilihan ganda untuk memastikan pemahaman peserta.

Selain hard skill di bidang pengembangan teknis, soft skill juga menjadi target kompetensi peserta studi independen yaitu untuk penyiapan karir sebagai developer.

Studi independen akan ditutup dengan project akhir, dimana peserta akan bekerja dalam kelompok dan mengembangkan solusi dengan berbasis permasalahan yang terjadi dalam keseharian. Paket yang disusun oleh Dicoding telah memasukkan aspek teknis maupun non-teknis (soft-skill, penyiapan karir, dll). Pembelajaran akan berlangsung dengan dukungan tim Pembimbing dan Expert dari Dicoding.

Seluruh Pembimbing adalah tim internal yang bekerja di Dicoding dan akan menjadi wali dari peserta SIB MBKM dengan rasio 1 pembimbing akan membimbing 25 mahasiswa.

1. Aspek Teknis

362-385 jam aktivitas belajar selama durasi program

a. Persiapan

34 jam, untuk semua paket, detail & CPL tersedia pada [lampiran 1](#)

- i. [Memulai Dasar Pemrograman untuk Menjadi Pengembang Software](#)
(Pengenalan ke Dasar Pemrograman)
- ii. [Pengenalan ke Logika Pemrograman](#)
- iii. [Belajar Dasar Git dengan GitHub](#)

b. Materi Inti

- i. Pengembang Aplikasi Android

338 jam, detail & CPL tersedia pada [lampiran 2](#)

1. [Memulai Pemrograman Dengan Kotlin](#)
2. [Belajar Fundamental Aplikasi Android](#)
3. [Belajar Pengembangan Aplikasi Android Intermediate](#)
4. [Belajar Prinsip Pemrograman SOLID](#)
5. [Belajar Dasar UX Design](#)

- ii. Pengembang Aplikasi Android dan Back-End

328 jam, detail & CPL tersedia pada [lampiran 3](#)

1. [Memulai Pemrograman Dengan Kotlin](#)
2. [Belajar Fundamental Aplikasi Android](#)
3. [Belajar Prinsip Pemrograman SOLID](#)
4. [Belajar Dasar Pemrograman JavaScript](#)
5. [Belajar Membuat Aplikasi Back-End untuk Pemula](#)

- 6. [Belajar Dasar UX Design](#)
- iii. Pengembang Front-End Web dan Back-End
351 jam, detail & CPL tersedia pada [lampiran 4](#)
 - 1. [Belajar Dasar Pemrograman Web](#)
 - 2. [Belajar Membuat Front-End Web untuk Pemula](#)
 - 3. [Belajar Fundamental Front-End Web Development](#)
 - 4. [Menjadi Front-End Web Developer Expert](#)
 - 5. [Belajar Dasar Pemrograman JavaScript](#)
 - 6. [Belajar Membuat Aplikasi Back-End untuk Pemula](#)
- iv. Pengembang Front-End Web dan React
351 jam, detail & CPL tersedia pada [lampiran 5](#)
 - 1. [Belajar Dasar Pemrograman Web](#)
 - 2. [Belajar Membuat Front-End Web untuk Pemula](#)
 - 3. [Belajar Fundamental Front-End Web Development](#)
 - 4. [Belajar Membuat Aplikasi Web dengan React](#)
 - 5. [Belajar Fundamental Aplikasi Web dengan React](#)
 - 6. [Menjadi Front-End Web Developer Expert](#)
- v. Pengembang Machine Learning dan Front-End Web
332 jam, detail & CPL tersedia pada [lampiran 6](#)
 - 1. [Belajar Dasar Visualisasi Data](#)
 - 2. [Memulai Pemrograman Dengan Python](#)
 - 3. [Belajar Machine Learning untuk Pemula](#)
 - 4. [Belajar Pengembangan Machine Learning](#)
 - 5. [Machine Learning Terapan](#)
 - 6. [Evaluasi Penguasaan Machine Learning](#)
 - 7. [Belajar Dasar Pemrograman Web](#)
 - 8. [Belajar Membuat Front-End Web untuk Pemula](#)
 - 9. [Belajar Fundamental Front-End Web Development](#)
- vi. Pengembang Machine Learning dan DevOps
341 jam, detail & CPL tersedia pada [lampiran 7](#)
 - 1. [Belajar Dasar Visualisasi Data](#)
 - 2. [Memulai Pemrograman Dengan Python](#)
 - 3. [Belajar Machine Learning untuk Pemula](#)
 - 4. [Belajar Pengembangan Machine Learning](#)
 - 5. [Machine Learning Terapan](#)
 - 6. [Evaluasi Penguasaan Machine Learning](#)
 - 7. [Machine Learning Operations](#)
 - 8. [Belajar Dasar-Dasar DevOps](#)
 - 9. [Belajar Jaringan Komputer untuk Pemula](#)
 - 10. [Menjadi Linux System Administrator](#)
 - 11. [Belajar Implementasi CI/CD](#)
 - 12. [Belajar Membangun Arsitektur Microservices](#)
- vii. Pengembang Aplikasi Multi-Platform dan Back-End
343 jam, detail & CPL tersedia pada [lampiran 8](#)
 - 1. [Memulai Pemrograman Dengan Dart](#)
 - 2. [Belajar Membuat Aplikasi Flutter untuk Pemula](#)
 - 3. [Belajar Fundamental Aplikasi Flutter](#)
 - 4. [Belajar Prinsip Pemrograman SOLID](#)

5. [Menjadi Flutter Developer Expert](#)
 6. [Belajar Dasar Pemrograman JavaScript](#)
 7. [Belajar Membuat Aplikasi Back-End untuk Pemula](#)
 8. [Belajar Dasar UX Design](#)
 - viii. Pengembang Back-End dan DevOps
328 jam, detail & CPL tersedia pada [lampiran 9](#)
 1. [Cloud Practitioner Essentials](#)
 2. [Belajar Dasar Pemrograman JavaScript](#)
 3. [Belajar Membuat Aplikasi Back-End untuk Pemula](#)
 4. [Belajar Fundamental Aplikasi Back-End](#)
 5. [Belajar Dasar-Dasar DevOps](#)
 6. [Belajar Jaringan Komputer untuk Pemula](#)
 7. [Menjadi Linux System Administrator](#)
 8. [Belajar Implementasi CI/CD](#)
 9. [Belajar Membangun Arsitektur Microservices](#)
 - ix. Pengembang Back-End dan React
351 jam, detail & CPL tersedia pada [lampiran 10](#)
 1. [Belajar Dasar Pemrograman Web](#)
 2. [Belajar Membuat Front-End Web untuk Pemula](#)
 3. [Belajar Dasar Pemrograman JavaScript](#)
 4. [Belajar Membuat Aplikasi Back-End untuk Pemula](#)
 5. [Belajar Membuat Aplikasi Web dengan React](#)
 6. [Belajar Fundamental Aplikasi Web dengan React](#)
 7. [Menjadi React Web Developer Expert](#)
 - x. Pengembang Multi-Platform dan Aplikasi Android
343 jam, detail & CPL tersedia pada [lampiran 11](#)
 1. [Memulai Pemrograman Dengan Dart](#)
 2. [Belajar Membuat Aplikasi Flutter untuk Pemula](#)
 3. [Belajar Fundamental Aplikasi Flutter](#)
 4. [Belajar Prinsip Pemrograman SOLID](#)
 5. [Menjadi Flutter Developer Expert](#)
 6. [Belajar Dasar UX Design](#)
 7. [Memulai Pemrograman Dengan Kotlin](#)
 8. [Belajar Membuat Aplikasi Android untuk Pemula](#)
2. Aspek Non-teknis
541 jam aktivitas selama durasi program, sama untuk semua paket, detail & CPL tersedia pada [lampiran 12](#).
- a. Persiapan karir melalui pembelajaran daring, **9 Jam**
 - i. Kelas [Meniti Karir sebagai Software Developer](#)
 - ii. Sesi Webinar Kesiapan Karier,
2 sesi masing-masing 60 menit.
 - b. Softskill (**total 70 jam**)
 - i. Sesi Softskill bersama Pembimbing non Akademik
14 Jam, 7 sesi @ 2 Jam
 1. Personal Growth - Task and Time Management - Individual Development Goals

2. Leadership - Thinking Model - Critical and Creative - Problem Solving
 3. Effective Communication - Presentation Tips - Collaboration
 4. Career Orientation - Digital-Self Branding & Hiring Preparation
- CV, Linkedin and Social media Upgrade
- ii. Tugas Softskill - **56 Jam, 7 tugas @ 8 Jam**
- c. Sesi review dan tatap muka bersama expert
12 jam, 2 jam per 2 minggu
- d. Monitoring (**total 50 jam**)
 - i. Sesi progress report dan konsultasi bersama pembimbing
40 jam, 2 jam per minggu
 - ii. Sesi team meeting dengan tim SIB Dicoding
10 jam, 2 jam per bulan
- e. Final/Capstone Project bersama Expert dan Pembimbing
250 jam
 - i. Tema Proyek
 - ii. Desain dan Manajemen Proyek
 - iii. Kerja sama Tim
 - iv. Pelaporan dan Presentasi
- f. Sesi refleksi dan koordinasi dengan pembimbing di Universitas asal serta pengisian logbook di sistem kampus merdeka dan Universitas asal.
150 jam, 7-8 jam per minggu

Batasan Aktivitas

Aktivitas Studi Independen Bersertifikat meliputi pembelajaran individu dan project akhir dalam bentuk tim. Pada pembelajaran individu, setiap peserta akan mengikuti kelas dalam bentuk asynchronous dimana peserta dapat berkonsultasi dengan expert pada sesi tatap muka.

Selain itu, setiap peserta akan memiliki pembimbing/fasilitator sebagai tempat konsultasi jika ditemui kesulitan dalam mengikuti pembelajaran. Peserta wajib berkomunikasi dengan dosen pembimbing di Universitas asal dan mengisi logbook atau dokumen lain sesuai ketentuan pengelola SIB dan dari Universitas asal.

Peserta akan memperoleh sertifikat kompetensi di setiap kelas jika peserta berhasil lulus dari setiap ujian/penilaian yang diadakan untuk setiap kompetensi.

Pada project akhir, peserta akan dibagi menjadi kelompok, dimana satu kelompok terdiri atas beberapa orang dengan tema yang ditentukan oleh masing-masing kelompok dan harus mendapatkan persetujuan dari Tim Dicoding.

Periode Program

Program akan berlangsung Juli 2022 - Desember 2022 (5 bulan). Tanggal-tanggal penting adalah sebagai berikut:

- | | |
|---|-----------------------------------|
| 1. Registrasi dan Filtering | : 6 Juni 2022 - 10 Juli 2022 |
| 2. Pengumuman peserta diterima | : Paling lambat 20 Juli 2022 |
| 3. Persiapan | : 20 - 25 Juli 2022 |
| 4. Konsolidasi dengan Mitra PT | : 22 Juli 2022 |
| 5. Pemberian akses credential dan matrikulasi | : 25 Juli 2022 atau sebelumnya |
| 6. Onboarding oleh Mendikbudristek | : 25 Juli 2022 (tentatif) |
| 7. Briefing Teknis oleh tim Dicoding | : 25 Juli 2022 |
| 8. Masa pembelajaran | : 25 Juli 2022 - 23 Desember 2022 |
| 9. Pelaporan ke dosen pembimbing | : Minggu ketiga di setiap bulan. |
| 10. Final / Capstone Project | : November - Desember 2022 |
| 11. Penyelesaian Administrasi dan Pelaporan | : 12 - 31 Desember 2022 |

Kewajiban Mitra

1. Memberikan akses dan menyelenggarakan program sesuai ketentuan dan proposal SIB MBKM yang disampaikan kepada DIKTI.
2. Mengadakan kegiatan sesuai dengan proposal program yang disetujui DIKTI, termasuk proses seleksi, pengembangan program, dan proyek akhir.
3. Menyediakan pembimbing untuk membimbing peserta.
4. Memberikan laporan kepada DIKTI maupun Kampus asal terkait progres peserta.
5. Menyediakan tim customer service yang mampu menjawab pertanyaan stakeholder.
6. Menyampaikan informasi dan follow up dengan stakeholder untuk memberikan pembaruan informasi.

Hak dan Kewajiban Peserta

1. Hak Peserta
 - a. Mendapatkan akses pembelajaran ke platform Dicoding sesuai dengan kelas yang diberikan.
 - b. Bertanya dan mendapatkan bimbingan selama durasi program
 - c. Mendapatkan hasil penilaian (transkrip) yang memuat seluruh hasil pembelajaran pada akhir program.
 - d. Mendapatkan sertifikat penyelesaian program (untuk peserta yang lulus) dan surat keikutsertaan (untuk peserta yang tidak lulus) pada akhir program.
 - e. Mendapatkan SKS konversi sesuai dengan rekomendasi hasil akhir, berdasarkan kesepakatan dengan jurusan di awal program.
2. Kewajiban Peserta
 - a. Mengikuti aturan dan kode etik pembelajaran SIB Dicoding 2022.
 - b. Tidak melakukan plagiarisme. Peserta yang melakukan plagiarisme akan ditindak sesuai dengan ketentuan yang berlaku
 - c. Menyelesaikan program pembelajaran, tugas, dan ujian sesuai dengan timeline dan ketentuan yang ditetapkan oleh penyelenggara.

- d. Hadir dalam sesi yang diwajibkan oleh penyelenggara dengan kamera menyala. Kecuali sakit atau alasan darurat lainnya.
- e. Melaporkan progres dan detil pembelajaran kepada DIKTI dan dosen pembimbing di kampus asal sesuai dengan ketentuan dan mekanisme dari DIKTI dan dari kampus masing-masing

Peran, Hak, dan Kewajiban Dosen Pembimbing

1. Peran Dosen Pembimbing
 - a. Menjadi penghubung antara Dicoding dengan mahasiswa apabila terjadi mahasiswa inaktif, kesulitan dalam program, dsb.
 - b. Melakukan filtering awal keseriusan mahasiswa dalam mengikuti program.
 - c. Memantau progress dan kemajuan peserta dalam program
 - d. Melakukan konversi SKS pada akhir program.
2. Hak Dosen Pembimbing
 - a. Mendapatkan akses dan laporan untuk memantau progres siswa, setidaknya satu kali setiap bulan dalam durasi program.
 - b. Mendapatkan hasil penilaian (transkrip) mahasiswa yang memuat seluruh hasil pembelajaran dan rekomendasi SKS pada akhir program.
 - c. Mendapatkan nilai sementara dari tim SIB apabila batas input nilai adalah sebelum tanggal berakhirnya program.
3. Kewajiban Dosen Pembimbing
 - a. Memberikan persetujuan untuk peserta sebelum dimulainya program
 - b. Tetap berkomunikasi dan memantau peserta selama program dan meneruskan informasi dari peserta ke tim SIB apabila diperlukan.
 - c. Melaporkan kepada tim Dicoding apabila terjadi perubahan status peserta di kampus asal (misalnya cuti, yudisium, lulus, DO, dsb).

Mekanisme Pelaporan dan Komunikasi

1. Tim Dicoding akan menyediakan form untuk menginput tanggal batas pengisian nilai di universitas Bapak/Ibu.
2. Tim akan mengirimkan **surat penerimaan (acceptance letter)** sebelum program dimulai.
3. Transkrip (sementara/final) akan berisi:
 - a. Identitas Mahasiswa dan nama Pembimbing
 - b. Status transkrip (sementara/final)
 - c. Status proyek
 - d. Detail paket yang diambil peserta, meliputi
 - i. Kode Kelas/Aktivitas
 - ii. Nama Kelas/Aktivitas
 - iii. Jumlah Jam yang diselesaikan oleh peserta
 - iv. Rekomendasi SKS
 - v. Nilai Angka (1-100) dan Nilai Huruf (A-E)
- e. Absensi peserta

4. Bapak/Ibu dapat bertanya kepada tim Dicoding melalui email atau WA yang tercantum pada bagian kontak. Kami akan membalas selambatnya pada hari kerja berikutnya.
5. Apabila ada dokumen legalitas yang perlu ditandatangani (misalnya laporan magang, dsb), Bapak/Ibu dan mahasiswa dapat menggunakan detail berikut:

Nama Mitra	Dicoding Indonesia (PT. Dicoding Akademi Indonesia)
Nama Program	Dicoding Indonesia X Kampus Merdeka 2022
Penanggung Jawab	Deti Anggraini Ekawati (Program Manager) HP 0813 5727 3280
Alamat Korespondensi:	Dicoding Space, Jalan Batik Kumeli No 50, Kel. Sukaluyu, Kec. Cibeunying Kaler Bandung 40123
Durasi Program	25 Juli - 23 Desember 2022
Tanda tangan	Mohon di email ke mbkm@dicoding.org untuk dibubuhkan tanda tangan digital (beserta stempel apabila diperlukan)

Dokumen akan tersedia dalam 3 (tiga) hari kerja

Kontak tim Dicoding

Email Team SIB : mbkm@dicoding.org

Program Manager

Nama : Deti Anggraini Ekawati
Nomor Handphone : 081357273280
Alamat email : deti@dicoding.com

Senior Education Program Manager

Nama : Adrianus Yoza Aprilio
Nomor Handphone : 085931190140
Alamat email : yoza@dicoding.com

Chief Operations Officer

Nama : Kevin Kurniawan
Nomor Handphone : 08562068389
Alamat email : kevin@dicoding.com

Email dan Telepon/WA akan direspon pada hari dan jam kerja.

Lampiran 1. Silabus Persiapan

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
Memulai Dasar Pemrograman untuk Menjadi Pengembang Software	Di akhir kelas, siswa mampu memodifikasi aplikasi perangkat lunak menggunakan panduan diagram alur dan pemrograman dengan teknologi HTML, CSS, dan JavaScript tingkat dasar secara tepat sesuai persyaratan spesifikasi dan fungsionalitas aplikasi.	1. Siswa mampu meneliti, menganalisis, dan mengevaluasi persyaratan untuk aplikasi perangkat lunak dengan memahami kebutuhan aplikasi dari sisi pengguna dan spesifikasi teknis aplikasi. 2. Siswa mampu membuat perencanaan modifikasi aplikasi perangkat lunak dengan pembuatan requirement aplikasi dan diagram alur. 3. Siswa mampu memodifikasi aplikasi perangkat lunak menggunakan pemrograman HTML, CSS, dan JavaScript tingkat dasar. 4. Siswa mampu mengarahkan dokumentasi pemrograman dan pengembangan perangkat lunak menggunakan metode pengarsipan.	13 Jam (1 hari)	Materi bacaan elektronik, video, kuis, exam yang tersedia di dalam LMS Dicoding (www.dicoding.com/academies/237).	Penilaian dilakukan melalui 8 kali exam yang dilakukan di platform LMS Dicoding.
Pengenalan Ke Logika Pemrograman	Di akhir kelas, siswa dapat memahami logika pemrograman dasar dan menerapkannya dalam pemecahan masalah yang ada di bidang pekerjaan Software Developer.	1. Mengerti apa itu logika pemrograman. 2. Mengetahui apa itu gerbang logika beserta jenis-jenisnya. 3. Memahami cara pemecahan masalah dengan computational thinking.	6 Jam (1 hari)	Materi bacaan elektronik, video, kuis, exam yang tersedia di dalam LMS Dicoding (www.dicoding.com/academies/302).	Penilaian dilakukan melalui 4 kali exam yang dilakukan di platform LMS Dicoding.
Belajar Dasar Git dengan Github	Di akhir kelas, siswa dapat mengelola kumpulan data/kode mereka sendiri dalam repository Github. Serta dapat berkolaborasi dengan developer lain pada repository yang sama.	1. Memahami git sebagai version control system. 2. Memahami Github sebagai tools untuk mengelola kumpulan data/kode. 3. Memahami cara mengelola kumpulan data/kode, mulai dari membuat repository, melakukan perubahan, membuat branch lain, hingga melakukan pull request. 4. Memahami cara berkolaborasi dengan developer lain pada repository yang sama. 5. Memahami penggunaan GitHub sebagai portfolio.	15 Jam (1 hari)	Materi bacaan elektronik, video, kuis, exam yang tersedia di dalam LMS Dicoding (www.dicoding.com/academies/317).	Penilaian dilakukan melalui 1 kali exam yang dilakukan di platform LMS Dicoding.
Subtotal Matrikulasi		34 jam, 3 hari			

Lampiran 2. Silabus Pengembang Aplikasi Android

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
Memulai Pemrograman Dengan Kotlin	Di akhir kelas, siswa paham tentang konsep dasar bahasa pemrograman, functional programming, serta object-oriented programming (OOP) dengan menggunakan Kotlin.	<p>1. Mempelajari bahasa Kotlin dan karakteristiknya, serta bagaimana Kotlin berperan dalam pengembangan aplikasi.</p> <p>2. Memahami tentang teori dasar Kotlin.</p> <p>Dengan mempelajari konsep dasar dari Kotlin, seperti menggunakan tipe data, membuat fungsi dasar, menangani tipe data null, dan membuat String Template.</p> <p>3. Mempelajari perbedaan antara expression dan statement, cara menggunakan enumeration, cara mengontrol aliran program menggunakan ekspresi if dan when, dan bagaimana menerapkan perulangan menggunakan while, do-while, dan break and continue.</p> <p>4. Mengetahui bagaimana mengelola data di Kotlin dengan menggunakan data class dan collection.</p> <p>5. Memahami konsep function programing dengan mempelajari extension, lambda, higher-order function, recursion serta melihat contoh penerapannya pada collection operator dan scope function.</p> <p>6. Memahami konsep OOP (Object-Oriented Programming) dengan mempelajari inheritance, abstract class, interface, visibility modifier, dan overloading.</p> <p>7. Mengetahui konsep Generic untuk membuat kelas yang dapat menerima berbagai jenis tipe data.</p> <p>8. Mempelajari Coroutine sebagai solusi untuk menjalankan concurrency pada Kotlin.</p>	50 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/80).	Penilaian dilakukan melalui 1 kali exam dan 1 kali submission project yang dilakukan di platform LMS Dicoding.
Belajar Fundamental Aplikasi Android	Di akhir kelas, siswa dapat membuat aplikasi yang dapat mengambil data dari Web API dan menyimpan data	1. Memahami konsep dasar mengembangkan aplikasi Android dengan menggunakan Activity, Intent, Fragment, View & View Group, dan Style & Theme.	150 jam	Materi bacaan elektronik, video, kuis, exam yang tersedia di dalam LMS Dicoding	Penilaian dilakukan melalui 8 kali exam dan 3 kali submission project yang dilakukan di platform LMS

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
	secara lokal.	<p>2. Memahami cara membuat UI/UX aplikasi yang bagus dengan menerapkan Material Design Component.</p> <p>3. Memahami bagaimana melakukan navigasi dalam sebuah aplikasi.</p> <p>4. Memahami berbagai mekanisme menjalankan proses secara asynchronous di background thread.</p> <p>5. Memahami cara koneksi ke Web API dengan menggunakan networking library dan mem-parsing JSON.</p> <p>6. Memahami Android Architecture Component sebagai architecture pattern yang digagas oleh Google.</p> <p>7. Memahami konsep ViewModel untuk mempertahankan data ketika aplikasi dirotasi.</p> <p>8. Memahami konsep LiveData untuk menampilkan data secara real-time ketika ada perubahan data di database.</p> <p>9. Memahami berbagai cara untuk menyimpan data di local, salah satunya menggunakan SQLite dan Room.</p> <p>10. Memahami cara menguji sebuah aplikasi yang akan dibuat, baik menggunakan unit testing maupun instrumental testing.</p> <p>11. Memahami cara membuat reminder dengan mengkolaborasikan fitur setting, AlarmManager, BroadcastReceiver, dan Notification.</p>		(https://www.dicoding.com/academies/14)	Dicoding.
Belajar Pengembangan Aplikasi Android Intermediate	Di akhir kelas, siswa dapat membuat aplikasi dengan menerapkan berbagai komponen yang ada, seperti UI, Animation, Localization, Service, Media, Location, Testing, dan Database.	<p>1. Memahami bagaimana membangun Custom View, baik dari pewarisan kelas tertentu atau membangunnya dari awal.</p> <p>2. Memahami bagaimana membangun object 2D menggunakan Canvas untuk kebutuhan tampilan yang lebih spesifik.</p> <p>3. Memahami bagaimana membuat Widget dalam Aplikasi Android.</p> <p>4. Memahami bagaimana menampilkan Website dalam Aplikasi Android melalui WebView.</p> <p>5. Memahami bagaimana cara membuat animasi,</p>	100 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/352).	Penilaian dilakukan melalui 10 kali exam dan 2 kali submission project yang dilakukan di platform LMS Dicoding.

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
		<p>baik menggunakan Properti Animation maupun Motion Layout.</p> <p>6. Memahami bagaimana menerapkan Localization untuk mendukung multi-bahasa.</p> <p>7. Memahami bagaimana menerapkan Accessibility untuk mendukung kebutuhan pengguna, terutama bagi pengguna penyandang disabilitas.</p> <p>8. Memahami bagaimana Service berjalan untuk melakukan sesuatu aksi di Background Thread.</p> <p>9. Memahami cara mengelola media menggunakan SoundPool, Media Player, dan ExoPlayer.</p> <p>10. Memahami bagaimana mengakses file media dari shared storage menggunakan MediaStore, serta menelusuri dan membuka dokumen, gambar, dan file lain menggunakan Storage Access Framework.</p> <p>11. Memahami bagaimana mengunggah file ke server dengan Multipart dalam retrofit.</p> <p>12. Memahami bagaimana mendapatkan posisi pengguna (latitude dan longitude) menggunakan Location Service, mendapatkan informasi menari dari radius dan posisi tertentu menggunakan Geofences, serta menampilkan peta/maps menggunakan Google Maps API.</p> <p>13. Mengetahui berbagai pendekatan dalam menguji aplikasi, baik secara unit testing menggunakan Test Double, maupun ui testing menggunakan Idling Resource maupun MockWebServer.</p>			
Belajar Prinsip Pemrograman SOLID	Di akhir kelas, siswa dapat menerapkan konsep-konsep dasar OOP dan relasinya untuk menyelesaikan masalah pada software design dengan menggunakan 5 prinsip SOLID.	<p>1. Memahami 4 pilarnya OOP, yakni Inheritance, Encapsulation, Abstraction, dan Polymorphism.</p> <p>2. Mengetahui macam-macam hubungan antar objek di OOP, seperti association, aggregation, composition, dependency, generalization,</p>	15 jam	<p>Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/169).</p>	<p>Penilaian dilakukan melalui 5 kali exam yang dilakukan di platform LMS Dicoding.</p>

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
		<p>specialization, dan implementation.</p> <p>3. Mempelajari 3 karakteristik penting dari design buruk yang perlu dihindari, yakni rigidity, fragility, dan immobility.</p> <p>4. Memahami tujuan 5 prinsip SOLID dan studi kasusnya, yaitu Single Responsibility, Open Closed, Liskov Substitution, Interface Segregation, dan Dependency Inversion.</p>			
Belajar Dasar UX Design	Di akhir kelas, siswa membuat sebuah portofolio berupa UX case study yang berisi keseluruhan proses, mulai dari mencari permasalahan pengguna, mendefinisikan masalah, melakukan eksplorasi, membuat prototype, sampai menguji cobanya.	<p>1. Peserta mampu memahami konsep dasar UX Design dan faktor yang mempengaruhinya, serta memahami proses desain dengan menggunakan pendekatan Design Thinking.</p> <p>2. Peserta mampu memahami bagaimana cara untuk mengetahui masalah utama (pain point) pengguna dengan membuat Empathy Map, User Persona, dan User Journey Map. Dilanjutkan dengan mendefinisikan masalah dalam Problem Statement dan melakukan eksplorasi untuk mencari solusi dengan menggunakan metode How Might We (HMW) dan Crazy Eight.</p> <p>3. Peserta mampu mendefinisikan solusi dalam bentuk Goal Statement dan mengetahui cara membuat rancangan desain dengan membuat User Flow, Storyboard, dan Wireframe.</p> <p>4. Peserta mampu memahami cara membuat Mockup dan High-Fidelity Prototype yang menarik dan interaktif dengan menerapkan prinsip desain visual.</p> <p>5. Peserta mampu merencanakan dan melakukan usability study, mengidentifikasi insight, memodifikasi desain berdasarkan hasil riset, dan membagikan hasil desain kepada tim.</p>	23 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/335)	Penilaian dilakukan melalui 7 kali exam dan 1 kali submission project yang dilakukan di platform LMS Dicoding.
Subtotal Pembelajaran Pengembang Aplikasi Android		338 jam			

Lampiran 3. Silabus Pengembang Aplikasi Android dan Back-End

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
Memulai Pemrograman Dengan Kotlin	Di akhir kelas, siswa paham tentang konsep dasar bahasa pemrograman, functional programming, serta object-oriented programming (OOP) dengan menggunakan Kotlin.	<p>Tujuan Khusus Pelatihan:</p> <ol style="list-style-type: none"> 1. Mempelajari bahasa Kotlin dan karakteristiknya, serta bagaimana Kotlin berperan dalam pengembangan aplikasi. 2. Memahami tentang teori dasar Kotlin. Dengan mempelajari konsep dasar dari Kotlin, seperti menggunakan tipe data, membuat fungsi dasar, menangani tipe data null, dan membuat String Template. 3. Mempelajari perbedaan antara expression dan statement, cara menggunakan enumeration, cara mengontrol aliran program menggunakan ekspresi if dan when, dan bagaimana menerapkan perulangan menggunakan while, do-while, dan break and continue. 4. Mengetahui bagaimana mengelola data di Kotlin dengan menggunakan data class dan collection. 5. Memahami konsep function programing dengan mempelajari extension, lambda, higher-order function, recursion serta melihat contoh penerapannya pada collection operator dan scope function. 6. Memahami konsep OOP (Object-Oriented Programming) dengan mempelajari inheritance, abstract class, interface, visibility modifier, dan overloading. 7. Mengetahui konsep Generic untuk membuat kelas yang dapat menerima berbagai jenis tipe data. 8. Mempelajari Coroutine sebagai solusi untuk menjalankan concurrency pada Kotlin. 	50 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/80).	Penilaian dilakukan melalui 1 kali exam dan 1 kali submission project yang dilakukan di platform LMS Dicoding.
Belajar Fundamental Aplikasi Android	Di akhir kelas, siswa dapat membuat aplikasi yang dapat	<ol style="list-style-type: none"> 1. Memahami konsep dasar mengembangkan aplikasi Android dengan menggunakan Activity, 	150 jam	Materi bacaan elektronik, video, kuis, exam yang	Penilaian dilakukan melalui 8 kali exam dan 3 kali submission

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
	mengambil data dari Web API dan menyimpan data secara lokal.	<p>Intent, Fragment, View & View Group, dan Style & Theme.</p> <p>2. Memahami cara membuat UI/UX aplikasi yang bagus dengan menerapkan Material Design Component.</p> <p>3. Memahami bagaimana melakukan navigasi dalam sebuah aplikasi.</p> <p>4. Memahami berbagai mekanisme menjalankan proses secara asynchronous di background thread.</p> <p>5. Memahami cara koneksi ke Web API dengan menggunakan networking library dan mem-parsing JSON.</p> <p>6. Memahami Android Architecture Component sebagai architecture pattern yang digagas oleh Google.</p> <p>7. Memahami konsep ViewModel untuk mempertahankan data ketika aplikasi dirotasi.</p> <p>8. Memahami konsep LiveData untuk menampilkan data secara real-time ketika ada perubahan data di database.</p> <p>9. Memahami berbagai cara untuk menyimpan data di local, salah satunya menggunakan SQLite dan Room.</p> <p>10. Memahami cara menguji sebuah aplikasi yang akan dibuat, baik menggunakan unit testing maupun instrumental testing.</p> <p>11. Memahami cara membuat reminder dengan mengkolaborasikan fitur setting, AlarmManager, BroadcastReceiver, dan Notification.</p>		tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/14).	project yang dilakukan di platform LMS Dicoding.
Belajar Prinsip Pemrograman SOLID	Di akhir kelas, siswa dapat menerapkan konsep-konsep dasar OOP dan relasinya untuk menyelesaikan masalah pada software design dengan menggunakan 5 prinsip SOLID.	<p>1. Memahami 4 pilar OOP, yakni Inheritance, Encapsulation, Abstraction, dan Polymorphism.</p> <p>2. Mengetahui macam-macam hubungan antar objek di OOP, seperti association, aggregation, composition, dependency, generalization, specialization, dan implementation.</p> <p>3. Mempelajari 3 karakteristik penting dari design buruk yang perlu</p>	15 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/169).	Penilaian dilakukan melalui 5 kali exam yang dilakukan di platform LMS Dicoding.

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
		dihindari, yakni rigidity, fragility, dan immobility. 4. Memahami tujuan 5 prinsip SOLID dan studi kasusnya, yaitu Single Responsibility, Open Closed, Liskov Substitution, Interface Segregation, dan Dependency Inversion.			
Belajar Dasar Pemrograman JavaScript	Di akhir kelas, siswa dapat membuat program dengan JavaScript menggunakan Node.js dan Text Editor, seperti Visual Studio Code.	1. Mengenal apa itu JavaScript, bagaimana sejarahnya, dan alasan untuk mempelajarinya. 2. Mengenal dan bisa mengimplementasikan komponen-komponen dasar di JavaScript. 3. Mengerti dan bisa mengelola data yang lebih kompleks. 4. Mengerti dan bisa mengimplementasikan function. 5. Mengerti dan bisa mengimplementasikan object oriented programming. 6. Mengerti dan bisa mengimplementasikan functional programming. 7. Mengerti dan bisa menyiapkan lingkungan pengembangan di komputer siswa. 8. Mengerti bagaimana beberapa berkas JavaScript dapat saling berkomunikasi satu sama lain. 9. Mengerti dan bisa menangani eror yang mungkin muncul dalam aplikasi. 10. Mengerti dan bisa mengimplementasikan concurrency. 11. Mengerti dan bisa memanfaatkan NPM untuk mengelola package eksternal di dalam project. 12. Mengerti dan bisa mengimplementasikan automated testing menggunakan package Jest.	45 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/256).	Penilaian dilakukan melalui 12 kali exam yang dilakukan di platform LMS Dicoding.
Belajar Membuat Aplikasi Back-End untuk Pemula	Di akhir kelas, siswa dapat membuat aplikasi back-end sederhana berupa RESTful API menggunakan Node.js.	1. Mengerti dasar pengetahuan back-end, seperti server dan client, Web server dan Web Service, REST, dan konsep di dalamnya. 2. Mengerti dasar-sasar Node.js untuk back-end, seperti Node.js, Global Object, Process Object, Modularization, event module, Filesystem, dan teknik Stream.	45 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/261).	Penilaian dilakukan melalui 1 kali submission project yang dilakukan di platform LMS Dicoding.

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
		3. Membangun Web Services Menggunakan Node.js secara native maupun menggunakan framework Hapi 4. Membangun RESTful api untuk fungsionalitas aplikasi nyata. 5. Mampu deploy web services melalui Amazon EC2 6. Mampu mengonsumsi dan menguji RESTful API menggunakan Postman			
Belajar Dasar UX Design	Di akhir kelas, siswa membuat sebuah portofolio berupa UX case study yang berisi keseluruhan proses, mulai dari mencari permasalahan pengguna, mendefinisikan masalah, melakukan eksplorasi, membuat prototype, sampai menguji cobanya.	1. Peserta mampu memahami konsep dasar UX Design dan faktor yang mempengaruhinya, serta memahami proses desain dengan menggunakan pendekatan Design Thinking. 2. Peserta mampu memahami bagaimana cara untuk mengetahui masalah utama (pain point) pengguna dengan membuat Empathy Map, User Persona, dan User Journey Map. Dilanjutkan dengan mendefinisikan masalah dalam Problem Statement dan melakukan eksplorasi untuk mencari solusi dengan menggunakan metode How Might We (HMW) dan Crazy Eight. 3. Peserta mampu mendefinisikan solusi dalam bentuk Goal Statement dan mengetahui cara membuat rancangan desain dengan membuat User Flow, Storyboard, dan Wireframe. 4. Peserta mampu memahami cara membuat Mockup dan High-Fidelity Prototype yang menarik dan interaktif dengan menerapkan prinsip desain visual. 5. Peserta mampu merencanakan dan melakukan usability study, mengidentifikasi insight, memodifikasi desain berdasarkan hasil riset, dan membagikan hasil desain kepada tim.	23 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/335).	Penilaian dilakukan melalui 7 kali exam dan 1 kali submission project yang dilakukan di platform LMS Dicoding.
Subtotal Pembelajaran Pengembang Android Back-end		328 jam			

Lampiran 4. Silabus Pengembang Front-End Web dan Back-End

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
Belajar Dasar Pemrograman Web	Di akhir pelatihan, siswa dapat membuat sebuah website sederhana menggunakan kode pemrograman yang sesuai standar global.	<ol style="list-style-type: none"> 1. Mengenal apa itu Website, serta kebutuhan dalam mengembangkan website (front-end). 2. Mengerti peran, fungsi, dan cara menuliskan struktur HTML. 3. Mengetahui berbagai macam tags elemen dan penggunaannya. 4. Dapat mengelompokkan konten dengan menggunakan element semantik. 	55 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (www.dicoding.com/academies/123).	Penilaian dilakukan melalui 9 kali exam dan 1 kali submission project yang dilakukan di platform LMS Dicoding.
Belajar Membuat Front-End Web untuk Pemula	Di akhir kelas, siswa dapat membuat aplikasi front-end web yang interaktif serta memiliki fitur penyimpanan menggunakan web storage.	<ol style="list-style-type: none"> 1. Mengerti peran dan fungsi dari HTML dan CSS pada pengembangan Website. 2. Mengenal JavaScript dan Fungsinya. 3. Menguasai Dasar Pemrograman JavaScript, seperti statement dan expression, variabel, komentar, tipe data, operator, percabangan, perulangan, objek dan array. 4. Mengetahui Document Object Model dan Browser Object Model 5. Mampu membuat website menjadi interaktif dengan teknik Manipulation DOM dan Eventing. 6. Mampu menggunakan Web Storage untuk media penyimpanan data pada Website. 	46 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (www.dicoding.com/academies/315).	Penilaian dilakukan melalui 1 kali submission project yang dilakukan di platform LMS Dicoding.
Belajar Fundamental Front-End Web Development	Di akhir kelas, siswa dapat membuat aplikasi front-end web dengan kode JavaScript standar ES6, menerapkan Web Components, Webpack, dan Menampilkan data dinamis dari Web API menggunakan AJAX.	<ol style="list-style-type: none"> 1. Mampu menuliskan kode JavaScript menggunakan standar ES6. 2. Mampu membuat dan menggunakan Web Component dalam pengembangan aplikasi Web. 3. Mampu membuat proyek JavaScript dan menggunakan package pihak ketiga melalui Node Package Manager (NPM). 4. Mampu menggunakan Webpack Sebagai Module Bundler. 5. Mampu menampilkan data secara dinamis dari Web API menggunakan AJAX. 	70 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (www.dicoding.com/academies/163).	Penilaian dilakukan melalui 3 kali exam dan 1 kali submission project yang dilakukan di platform LMS Dicoding.

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
Menjadi Front-End Web Developer Expert	Di akhir kelas, siswa dapat membuat aplikasi front-end web yang responsif, memiliki aksesibilitas yang baik, mudah di-maintenance, memiliki sifat native, dapat diuji, dan memiliki performa yang baik.	1. Mengerti teknik membuat website dengan pendekatan mobile first approach. 2. Paham pentingnya aksesibilitas dan mampu membuat website yang memiliki aksesibilitas keyboard maupun konten yang baik. 3. Mampu menuliskan kode JavaScript sesuai dengan prinsip Clean Code serta dengan gaya yang konsisten. 4. Mampu membangun web yang modern dengan memiliki sifat, seperti aplikasi native. 5. Mampu membangun website yang teruji melalui pengujian otomatis. 6. Mampu membangun website dengan kinerja yang baik.	90 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/219).	Penilaian dilakukan melalui 2 kali exam dan 3 kali submission project yang dilakukan di platform LMS Dicoding.
Belajar Dasar Pemrograman JavaScript	Di akhir kelas, siswa dapat membuat program dengan JavaScript menggunakan Node.js dan Text Editor, seperti Visual Studio Code.	1. Mengenal apa itu JavaScript, bagaimana sejarahnya, dan alasan untuk mempelajarinya. 2. Mengenal dan bisa mengimplementasikan komponen-komponen dasar di JavaScript. 3. Mengerti dan bisa mengelola data yang lebih kompleks. 4. Mengerti dan bisa mengimplementasikan function. 5. Mengerti dan bisa mengimplementasikan object oriented programming. 6. Mengerti dan bisa mengimplementasikan functional programming. 7. Mengerti dan bisa menyiapkan lingkungan pengembangan di komputer siswa. 8. Mengerti bagaimana beberapa berkas JavaScript dapat saling berkomunikasi satu sama lain. 9. Mengerti dan bisa menangani eror yang mungkin muncul dalam aplikasi. 10. Mengerti dan bisa mengimplementasikan concurrency. 11. Mengerti dan bisa memanfaatkan NPM untuk mengelola package eksternal di dalam project. 12. Mengerti dan bisa	45 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/256).	Penilaian dilakukan melalui 12 kali exam yang dilakukan di platform LMS Dicoding.

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
		mengimplementasikan automated testing menggunakan package Jest.			
Belajar Membuat Aplikasi Back-End untuk Pemula	Di akhir kelas, siswa dapat membuat aplikasi back-end sederhana berupa RESTful API menggunakan Node.js.	1. Mengerti dasar pengetahuan back-end, seperti server dan client, Web server dan Web Service, REST, dan konsep di dalamnya. 2. Mengerti dasar-sasar Node.js untuk back-end, seperti Node.js, Global Object, Process Object, Modularization, event module, Filesystem, dan teknik Stream. 3. Membangun Web Services Menggunakan Node.js secara native maupun menggunakan framework Hapi 4. Membangun RESTful api untuk fungsionalitas aplikasi nyata. 5. Mampu deploy web services melalui Amazon EC2 6. Mampu mengonsumsi dan menguji RESTful API menggunakan Postman	45 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/261).	Penilaian dilakukan melalui 1 kali submission project yang dilakukan di platform LMS Dicoding.
Subtotal Pembelajaran Front-end Back-end			351 jam		

Lampiran 5. Silabus Pengembang Front-End Web dan React

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
Belajar Dasar Pemrograman Web	Di akhir pelatihan, siswa dapat membuat sebuah website sederhana menggunakan kode pemrograman yang sesuai standar global.	1. Mengenal apa itu Website, serta kebutuhan dalam mengembangkan website (front-end). 2. Mengerti peran, fungsi, dan cara menuliskan struktur HTML. 3. Mengetahui berbagai macam tags elemen dan penggunaannya. 4. Dapat mengelompokkan konten dengan menggunakan element semantik.	55 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (www.dicoding.com/academies/123).	Penilaian dilakukan melalui 9 kali exam dan 1 kali submission project yang dilakukan di platform LMS Dicoding.
Belajar Membuat Front-End Web untuk Pemula	Di akhir kelas, siswa dapat membuat aplikasi front-end web yang interaktif serta memiliki fitur penyimpanan menggunakan web storage.	1. Mengerti peran dan fungsi dari HTML dan CSS pada pengembangan Website. 2. Mengenal JavaScript dan Fungsinya. 3. Menguasai Dasar Pemrograman JavaScript, seperti statement dan expression, variabel, komentar, tipe data, operator, percabangan, perulangan, objek dan array. 4. Mengetahui Document Object Model dan Browser Object Model 5. Mampu membuat website menjadi interaktif dengan teknik Manipulation DOM dan Eventing. 6. Mampu menggunakan Web Storage untuk media penyimpanan data pada Website.	46 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (www.dicoding.com/academies/315).	Penilaian dilakukan melalui 1 kali submission project yang dilakukan di platform LMS Dicoding.
Belajar Fundamental Front-End Web Development	Di akhir kelas, siswa dapat membuat aplikasi front-end web dengan kode JavaScript standar ES6, menerapkan Web Components, Webpack, dan Menampilkan data dinamis dari Web API menggunakan AJAX.	1. Mampu menuliskan kode JavaScript menggunakan standar ES6. 2. Mampu membuat dan menggunakan Web Component dalam pengembangan aplikasi Web. 3. Mampu membuat proyek JavaScript dan menggunakan package pihak ketiga melalui Node Package Manager (NPM). 4. Mampu menggunakan Webpack Sebagai Module Bundler. 5. Mampu menampilkan data secara dinamis dari Web API menggunakan AJAX.	70 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (www.dicoding.com/academies/163).	Penilaian dilakukan melalui 3 kali exam dan 1 kali submission project yang dilakukan di platform LMS Dicoding.

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
Belajar Membuat Aplikasi Web dengan React	Di akhir kelas siswa dapat membuat aplikasi web dengan React yang memanfaatkan functional component, stateful component, dan controlled component.	1. Mengetahui pengertian, alasan, dan ekosistem pada React. 2. Memahami konsep dasar dari React, seperti element dan component, compositional model, declarative code, unidirectional data flow, dan React is Just JavaScript. 3. Mampu membuat React element menggunakan createElement ataupun JSX. 4. Mampu membuat React component melalui fungsi (functional). 5. Mampu memberikan dan mengonsumsi nilai properti pada component. 6. Mampu membuat React Stateful Component melalui Class 7. Mampu menerapkan event handling pada Component 8. Mampu membuat, menggunakan dan menangani nilai pada element Form (Controlled Element) 10. Mampu men-debug guna memantau nilai dari properti dan state pada komponen.	35 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/403).	Penilaian dilakukan melalui 5 kali exam dan 1 kali submission project yang dilakukan di platform LMS Dicoding.
Belajar Fundamental Aplikasi Web dengan React	Di akhir kelas siswa dapat membuat aplikasi web dengan React yang menerapkan validasi properti, routing, component lifecycle, context, dan hooks.	1. Memahami cara mengimplementasikan validasi property component menggunakan PropTypes. 2. Memahami cara membuat routing yang mudah dengan memanfaatkan React Router. 3. Memahami cara memanfaatkan lifecycle component untuk menampilkan data dari API. 4. Memahami cara memanfaatkan React Context dalam berbagai macam kasus yang membutuhkan global state. 5. Memahami cara menggunakan Hooks API untuk memanfaatkan state di dalam functional component.	55 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/413).	Penilaian dilakukan melalui 6 kali exam dan 2 kali submission project yang dilakukan di platform LMS Dicoding.
Menjadi Front-End Web Developer Expert	Di akhir kelas, siswa dapat membuat aplikasi front-end web yang responsif, memiliki aksesibilitas yang	1. Mengerti teknik membuat website dengan pendekatan mobile first approach. 2. Paham pentingnya aksesibilitas dan mampu	90 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding	Penilaian dilakukan melalui 2 kali exam dan 3 kali submission project yang dilakukan di

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
	baik, mudah di-maintenance, memiliki sifat native, dapat diuji, dan memiliki performa yang baik.	membuat website yang memiliki aksesibilitas keyboard maupun konten yang baik. 3. Mampu menuliskan kode JavaScript sesuai dengan prinsip Clean Code serta dengan gaya yang konsisten. 4. Mampu membangun web yang modern dengan memiliki sifat, seperti aplikasi native. 5. Mampu membangun website yang teruji melalui pengujian otomatis. 6. Mampu membangun website dengan kinerja yang baik.		(https://www.dicoding.com/academies/219).	platform LMS Dicoding.
Subtotal Pembelajaran Pengembang Front-end React		351 jam			

Lampiran 6. Silabus Pengembang Machine Learning dan Front-End Web

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
Belajar Dasar Visualisasi Data	Di akhir kelas, siswa dapat membuat sebuah visualisasi data yang efektif menggunakan Google Sheets sesuai teknik praktik terbaik industri sehingga meningkatkan nilainya dalam berkarier sebagai Software Developer.	1. Mengidentifikasi macam-macam media untuk visualisasi data. 2. Menyimpulkan jenis visualisasi data yang terbaik berdasarkan situasi dan kebutuhannya. 3. Membangun visualisasi data menggunakan aplikasi spreadsheet Google Sheets. 4. Menerapkan praktik terbaik, seperti prinsip Gestalt, konsep Preattentive Attributes, dan konsep teori desain, dalam membangun visualisasi data.	16 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/177).	Penilaian dilakukan melalui 5 kali exam yang dilakukan di platform LMS Dicoding.
Memulai Pemrograman Dengan Python	Di akhir kelas, siswa dapat membuat program Python dengan menggunakan IDE Jupyter Notebook, Google Colab, atau IBM Watson Studio.	1. Mengenal apa itu Python serta filosofi-filosofi yang mendasari bahasa Python. 2. Mengetahui cara melakukan instalasi Python pada berbagai OS, beberapa IDE untuk Python, serta dapat melakukan pemrograman dengan Python pada Jupyter Notebook, Google Colab, atau IBM Watson Studio. 3. Mengerti cara penulisan code Python dengan benar. 4. Memahami basic data type dan data structure, seperti list, tuple, set, dan dictionary, mampu melakukan transformasi tipe karakter dan string, serta memahami mode operasi pada Python. 5. Memahami kegunaan setiap tipe variabel, Mengerti tentang Dynamic Typing, Operator, Operands, Expression, serta Input/Output pada bahasa Python. 6. Memahami tentang style guide pada Python. 7. Memahami standar perulangan dan percabangan pada Python. 8. Memahami bagaimana menangani kesalahan (error handling) dan pengecualian (exception) pada Python.	20 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/86).	Penilaian dilakukan melalui 10 kali exam yang dilakukan di platform LMS Dicoding.

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
		<p>9. Mempelajari fungsi-fungsi dan pustaka yang sering dipergunakan untuk menunjang produktivitas Anda menggunakan bahasa Python.</p> <p>10. Memahami tentang object dan class, serta paham tentang pemrograman berorientasi objek dengan Python.</p> <p>11. Mengerti tentang unit testing pada Python.</p> <p>12. Mengetahui library populer pada Python.</p>			
Belajar Machine Learning untuk Pemula	Di akhir kelas, siswa dapat belajar materi dasar dan algoritma-algoritma Machine Learning, serta mampu mengimplementasikannya dalam membuat model Machine Learning untuk memproses data.	<p>1. Memahami berbagai hal tentang data dan bagaimana mempersiapkannya untuk diproses dengan Machine Learning.</p> <p>2. Memahami 2 jenis Machine Learning, yaitu Supervised dan Unsupervised, dengan contoh model Regresi Linear dan Decision Tree untuk Supervised Learning dan K-Means Clustering untuk Unsupervised Learning. Pada modul ini peserta juga akan belajar tentang System Recommendation.</p> <p>3. Mengenal Support Vector Machine, sebuah algoritma Machine Learning yang sangat populer berserta implementasinya untuk kasus Klasifikasi dan Regresi.</p> <p>4. Memahami beberapa hal dasar tentang Machine Learning, seperti Machine Learning Workflow, Use Case, Overfitting dan Underfitting, dan Model Selection. Pada modul ini peserta juga akan belajar bagaimana menggunakan Grid Search untuk mencari parameter terbaik sebuah model, serta bagaimana menguji kualitas model Machine Learning.</p> <p>5. Mengenal dasar dari Neural Network. Pada modul ini akan diterangkan juga mengenai Multi Layer Perceptron serta Convolutional Neural Network dalam Image Classification.</p> <p>6. Mengenal library</p>	30 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/184).	Penilaian dilakukan melalui 1 kali submission yang dilakukan di platform LMS Dicoding.

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
		<p>TensorFlow dan Keras, dua jenis library powerful yang dipakai untuk mengembangkan project Machine Learning.</p> <p>7. Untuk menguji pemahaman peserta terhadap materi kelas ini, peserta akan membuat sebuah proyek Machine Learning sederhana untuk mengklasifikasi gambar menggunakan CNN dan TensorFlow sebagai Submission/Project Akhir</p>			
Belajar Pengembangan Machine Learning	Di akhir kelas, siswa dapat belajar mengenai implementasi Machine Learning pada industri mulai dari Computer Vision, Natural Language, serta deployment proyek Machine Learning.	<p>1. Mempelajari bagaimana merumuskan solusi Machine Learning untuk suatu masalah dengan teknik Problem Framing.</p> <p>2. Belajar mengembangkan model dengan format data berbeda, melakukan Pre-Processing terhadap data, berlatih membuat model klasifikasi, membuat Plot Akurasi dan Loss dari model, penggunaan Callback, TensorFlow Dataset, dan Batch Loading.</p> <p>3. Mengenal teknik yang dipakai dalam sistem rekomendasi, seperti Collaborative dan Content Based Filtering, serta sistem rekomendasi menggunakan jaringan saraf.</p> <p>4. Mengembangkan model untuk klasifikasi gambar dan teknik yang umum dipakai di industri.</p> <p>5. Belajar tentang teknik Pemrosesan Bahasa Alami (Natural Language Processing), berlatih mengolah data teks menggunakan TensorFlow dan praktik membuat model Machine Learning untuk klasifikasi teks.</p> <p>6. Mengenal data time series dan belajar mengembangkan model untuk prediksi data time series.</p> <p>7. Mengenal teknik Reinforcement Learning dan belajar melatih model berdasarkan teori hadiah dan hukuman menggunakan Keras, Game Playing Agent, serta algoritma Minimax.</p> <p>8. Memahami teknik deployment dan berlatih.</p>	40 jam	<p>Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/185).</p>	<p>Penilaian dilakukan melalui 3 kali submission yang dilakukan di platform LMS Dicoding.</p>

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
		<p>men-deploy model Machine Learning pada website dengan TensorFlow.js serta pada Android dengan TensorFlowLite.</p> <p>9. Untuk menguji pemahaman peserta terhadap materi kelas ini, peserta diminta membuat tiga buah project Machine Learning dengan kriteria tertentu menggunakan TensorFlow sebagai Tugas Submission, yaitu membuat model Machine Learning dengan data teks sebagai submission pertama, model Machine Learning dengan data time series sebagai submission kedua, dan model Machine Learning dengan data gambar sebagai submission akhir</p>			
Machine Learning Terapan	Di akhir kelas, siswa dapat belajar mengenai terapan machine learning mulai dari membuat model predictive analytics untuk bisnis dan marketing, sentiment analysis, computer vision untuk pengenalan gambar dan deteksi objek pada gambar, serta sistem rekomendasi.	<ol style="list-style-type: none"> 1. Paham mengenai tahapan dalam machine learning system design dan mampu melakukan proses desain sistem machine learning. 2. Memahami bagaimana tahapan dalam menyusun proyek machine learning serta mampu mengomunikasikan dan mempublikasikan proyek machine learning. 3. Mampu membuat repository proyek machine learning pada GitHub sebagai portofolio. 4. Memahami algoritma K-Nearest Neighbor, random forest, dan gradient boosting untuk kasus predictive analytics serta mampu membuat proyek machine learning pada kasus predictive analytics. 5. Memahami dan mampu membuat model analisis sentimen. 6. Memahami teknik pengenalan dan deteksi objek serta mampu membuat model machine learning untuk deteksi objek. 7. Paham mengenai metode Content based Filtering dan Collaborative Filtering dalam Sistem Rekomendasi serta mampu membuat proyek sistem rekomendasi. 	40 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/319).	Penilaian dilakukan melalui 7 kali exam dan 2 kali submission yang dilakukan di platform LMS Dicoding.

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
Evaluasi Penguasaan Machine Learning	Di akhir pelatihan, peserta dapat memahami konsep-konsep, implementasi dan studi kasus Machine Learning pada dunia nyata.	1. Memahami konsep-konsep dasar machine learning. 2. Memahami alur kerja machine learning, mulai dari terbentuknya data, preprocessing, sampai dengan implementasi. 3. Mengenal contoh implementasi dan studi kasus terkait konsep-konsep machine learning.	15 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/327).	Penilaian dilakukan melalui 1 kali exam yang dilakukan di platform LMS Dicoding.
Belajar Dasar Pemrograman Web	Di akhir pelatihan, siswa dapat membuat sebuah website sederhana menggunakan kode pemrograman yang sesuai standar global.	1. Mengenal apa itu Website, serta kebutuhan dalam mengembangkan website (front-end). 2. Mengerti peran, fungsi, dan cara menuliskan struktur HTML. 3. Mengetahui berbagai macam tags elemen dan penggunaannya. 4. Dapat mengelompokkan konten dengan menggunakan element semantik.	55 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (www.dicoding.com/academies/123).	Penilaian dilakukan melalui 9 kali exam dan 1 kali submission project yang dilakukan di platform LMS Dicoding.
Belajar Membuat Front-End Web untuk Pemula	Di akhir kelas, siswa dapat membuat aplikasi front-end web yang interaktif serta memiliki fitur penyimpanan menggunakan web storage.	1. Mengerti peran dan fungsi dari HTML dan CSS pada pengembangan Website. 2. Mengenal JavaScript dan Fungsinya. 3. Menguasai Dasar Pemrograman JavaScript, seperti statement dan expression, variabel, komentar, tipe data, operator, percabangan, perulangan, objek dan array. 4. Mengetahui Document Object Model dan Browser Object Model 5. Mampu membuat website menjadi interaktif dengan teknik Manipulation DOM dan Eventing. 6. Mampu menggunakan Web Storage untuk media penyimpanan data pada Website.	46 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (www.dicoding.com/academies/315).	Penilaian dilakukan melalui 1 kali submission project yang dilakukan di platform LMS Dicoding.
Belajar Fundamental Front-End Web Development	Di akhir kelas, siswa dapat membuat aplikasi front-end web dengan kode JavaScript standar ES6, menerapkan Web Components, Webpack, dan Menampilkan data dinamis dari Web API menggunakan AJAX.	1. Mampu menuliskan kode JavaScript menggunakan standar ES6. 2. Mampu membuat dan menggunakan Web Component dalam pengembangan aplikasi Web. 3. Mampu membuat proyek JavaScript dan menggunakan package pihak ketiga melalui Node Package Manager (70 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (www.dicoding.com/academies/163).	Penilaian dilakukan melalui 3 kali exam dan 1 kali submission project yang dilakukan di platform LMS Dicoding.

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
		NPM). 4. Mampu menggunakan Webpack Sebagai Module Bundler. 5. Mampu menampilkan data secara dinamis dari Web API menggunakan AJAX.			
Subtotal Pembelajaran Pengembang Machine Learning & Front-end		332 jam			

Lampiran 7. Silabus Pengembang Machine Learning dan DevOps

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
Belajar Dasar Visualisasi Data	Di akhir kelas, siswa dapat membuat sebuah visualisasi data yang efektif menggunakan Google Sheets sesuai teknik praktik terbaik industri sehingga meningkatkan nilainya dalam berkarier sebagai Software Developer.	1. Mengidentifikasi macam-macam media untuk visualisasi data. 2. Menyimpulkan jenis visualisasi data yang terbaik berdasarkan situasi dan kebutuhannya. 3. Membangun visualisasi data menggunakan aplikasi spreadsheet Google Sheets. 4. Menerapkan praktik terbaik, seperti prinsip Gestalt, konsep Preattentive Attributes, dan konsep teori desain, dalam membangun visualisasi data.	16 Jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/177).	Penilaian dilakukan melalui 5 kali exam yang dilakukan di platform LMS Dicoding.
Memulai Pemrograman Dengan Python	Di akhir kelas, siswa dapat membuat program Python dengan menggunakan IDE Jupyter Notebook, Google Colab, atau IBM Watson Studio.	1. Mengenal apa itu Python serta filosofi-filosofi yang mendasari bahasa Python. 2. Mengetahui cara melakukan instalasi Python pada berbagai OS, beberapa IDE untuk Python, serta dapat melakukan pemrograman dengan Python pada Jupyter Notebook, Google Colab, atau IBM Watson Studio. 3. Mengerti cara penulisan code Python dengan benar. 4. Memahami basic data type dan data structure, seperti list, tuple, set, dan dictionary, mampu melakukan transformasi tipe karakter dan string, serta memahami mode operasi pada Python. 5. Memahami kegunaan setiap tipe variabel, Mengerti tentang Dynamic Typing, Operator, Operands, Expression, serta Input/Output pada bahasa Python. 6. Memahami tentang style guide pada Python. 7. Memahami standar perulangan dan percabangan pada Python. 8. Memahami bagaimana menangani kesalahan (error)	20 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/86).	Penilaian dilakukan melalui 10 kali exam yang dilakukan di platform LMS Dicoding.

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
		<p>handling) dan pengecualian (exception) pada Python.</p> <p>9. Mempelajari fungsi-fungsi dan pustaka yang sering dipergunakan untuk menunjang produktivitas Anda menggunakan bahasa Python.</p> <p>10. Memahami tentang object dan class, serta paham tentang pemrograman berorientasi objek dengan Python.</p> <p>11. Mengerti tentang unit testing pada Python.</p> <p>12. Mengetahui library populer pada Python.</p>			
Belajar Machine Learning untuk Pemula	Di akhir kelas, siswa dapat belajar materi dasar dan algoritma-algoritma Machine Learning, serta mampu mengimplementasikannya dalam membuat model Machine Learning untuk memproses data.	<p>1. Memahami berbagai hal tentang data dan bagaimana mempersiapkannya untuk diproses dengan Machine Learning.</p> <p>2. Memahami 2 jenis Machine Learning, yaitu Supervised dan Unsupervised, dengan contoh model Regresi Linear dan Decision Tree untuk Supervised Learning dan K-Means Clustering untuk Unsupervised Learning. Pada modul ini peserta juga akan belajar tentang System Recommendation.</p> <p>3. Mengenal Support Vector Machine, sebuah algoritma Machine Learning yang sangat populer beserta implementasinya untuk kasus Klasifikasi dan Regresi.</p> <p>4. Memahami beberapa hal dasar tentang Machine Learning, seperti Machine Learning Workflow, Use Case, Overfitting dan Underfitting, dan Model Selection. Pada modul ini peserta juga akan belajar bagaimana menggunakan Grid Search untuk mencari parameter terbaik sebuah model, serta bagaimana menguji kualitas model Machine Learning.</p> <p>5. Mengenal dasar dari Neural Network. Pada modul ini akan diterangkan juga mengenai Multi Layer</p>	30 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/184).	Penilaian dilakukan melalui 1 kali submission yang dilakukan di platform LMS Dicoding.

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
		<p>Perceptron serta Convolutional Neural Network dalam Image Classification.</p> <p>6. Mengenal library TensorFlow dan Keras, dua jenis library powerful yang dipakai untuk mengembangkan project Machine Learning.</p> <p>7. Untuk menguji pemahaman peserta terhadap materi kelas ini, peserta akan membuat sebuah proyek Machine Learning sederhana untuk mengklasifikasi gambar menggunakan CNN dan TensorFlow sebagai Submission/Project Akhir.</p>			
Belajar Pengembangan Machine Learning	Di akhir kelas, siswa dapat belajar mengenai implementasi Machine Learning pada industri mulai dari Computer Vision, Natural Language, serta deployment proyek Machine Learning.	<p>1. Mempelajari bagaimana merumuskan solusi Machine Learning untuk suatu masalah dengan teknik Problem Framing.</p> <p>2. Belajar mengembangkan model dengan format data berbeda, melakukan Pre-Processing terhadap data, berlatih membuat model klasifikasi, membuat Plot Akurasi dan Loss dari model, penggunaan Callback, TensorFlow Dataset, dan Batch Loading.</p> <p>3. Mengenal teknik yang dipakai dalam sistem rekomendasi, seperti Collaborative dan Content Based Filtering, serta sistem rekomendasi menggunakan jaringan saraf.</p> <p>4. Mengembangkan model untuk klasifikasi gambar dan teknik yang umum dipakai di industri.</p> <p>5. Belajar tentang teknik Pemrosesan Bahasa Alami (Natural Language Processing), berlatih mengolah data teks menggunakan TensorFlow dan praktik membuat model Machine Learning untuk klasifikasi teks.</p> <p>6. Mengenal data time series dan belajar mengembangkan model untuk prediksi data time series.</p> <p>7. Mengenal teknik Reinforcement Learning</p>	40 jam	<p>Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/185).</p>	Penilaian dilakukan melalui 3 kali submission yang dilakukan di platform LMS Dicoding.

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
		<p>dan belajar melatih model berdasarkan teori hadiah dan hukuman menggunakan Keras, Game Playing Agent, serta algoritma Minimax.</p> <p>8. Memahami teknik deployment dan berlatih men-deploy model Machine Learning pada website dengan TensorFlow.js serta pada Android dengan TensorFlowLite.</p> <p>9. Untuk menguji pemahaman peserta terhadap materi kelas ini, peserta diminta membuat tiga buah project Machine Learning dengan kriteria tertentu menggunakan TensorFlow sebagai Tugas Submission, yaitu membuat model Machine Learning dengan data teks sebagai submission pertama, model Machine Learning dengan data time series sebagai submission kedua, dan model Machine Learning dengan data gambar sebagai submission akhir.</p>			
Machine Learning Terapan	<p>Di akhir kelas, siswa dapat belajar mengenai terapan machine learning mulai dari membuat model predictive analytics untuk bisnis dan marketing, sentiment analysis, computer vision untuk pengenalan gambar dan deteksi objek pada gambar, serta sistem rekomendasi.</p>	<p>1. Paham mengenai tahapan dalam machine learning system design dan mampu melakukan proses desain sistem machine learning.</p> <p>2. Memahami bagaimana tahapan dalam menyusun proyek machine learning serta mampu mengomunikasikan dan mempublikasikan proyek machine learning.</p> <p>3. Mampu membuat repository proyek machine learning pada GitHub sebagai portofolio.</p> <p>4. Memahami algoritma K-Nearest Neighbor, random forest, dan gradient boosting untuk kasus predictive analytics serta mampu membuat proyek machine learning pada kasus predictive analytics.</p> <p>5. Memahami dan mampu membuat model analisis sentimen</p> <p>6. Memahami teknik pengenalan dan deteksi</p>	40 jam	<p>Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/319).</p>	<p>Penilaian dilakukan melalui 7 kali exam dan 2 kali submission yang dilakukan di platform LMS Dicoding.</p>

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
		objek serta mampu membuat model machine learning untuk deteksi objek. 7. Paham mengenai metode Content based Filtering dan Collaborative Filtering dalam Sistem Rekomendasi serta mampu membuat proyek sistem rekomendasi.			
Evaluasi Penguasaan Machine Learning	Di akhir pelatihan, peserta dapat memahami konsep-konsep, implementasi dan studi kasus Machine Learning pada dunia nyata.	1. Memahami konsep-konsep dasar machine learning. 2. Memahami alur kerja machine learning, mulai dari terbentuknya data, preprocessing, sampai dengan implementasi. 3. Mengenal contoh implementasi dan studi kasus terkait konsep-konsep machine learning.	15 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/327).	Penilaian dilakukan melalui 1 kali exam yang dilakukan di platform LMS Dicoding.
Machine Learning Operations	Di akhir kelas, siswa dapat menerapkan model machine learning kedalam tahap produksi dengan membuat machine learning pipeline dan mengimplementasikan prinsip machine learning operations (MLOps).	1. Paham mengenai konsep dasar MLOps dan machine learning pipeline dalam sistem produksi. 2. Memahami tahapan dalam pengembangan data pipeline sebagai bagian awal dari machine learning pipeline. 3. Memahami teknik lanjutan dalam pengembangan model dan mampu memvalidasi model yang telah dibuat sebelum masuk tahap produksi. 4. Mampu menerapkan model yang telah dibuat ke dalam tahap produksi serta mengimplementasikan teknik monitoring dan feedback loop untuk menjaga kualitas sistem machine learning yang telah dibuat. 5. Mampu membuat end-to-end machine learning pipeline yang sesuai dengan prinsip MLOps 6. Mampu menerapkan dan menjalankan machine learning pipeline di Google Cloud Platform atau HEROKU.	45 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/443).	Penilaian dilakukan melalui 7 kali exam dan 2 kali submission project yang dilakukan di platform LMS Dicoding.
Belajar Dasar-Dasar DevOps	Di akhir kelas, siswa dapat memahami bagaimana kultur dan praktik DevOps mampu meningkatkan nilai	1. Mengetahui pengertian dan manfaat dari DevOps. 2. Memahami prinsip-prinsip yang ada pada DevOps. 3. Memahami kultur	15 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding	Penilaian dilakukan melalui 6 kali exam yang dilakukan di platform LMS Dicoding.

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
	teknis serta bisnis perusahaan.	DevOps melalui CALMS Framework. 4. Mengerti praktik DevOps dengan mengenal DevOps Pipeline dan DevOps Tools. 5. Mengetahui bagaimana kisah pengimplementasian DevOps pada kasus nyata dengan mengambil contoh perusahaan Amazon.		(https://www.dicoding.com/academies/382).	
Belajar Jaringan Komputer untuk Pemula	Di akhir kelas, siswa mampu memahami konsep dan praktik jaringan komputer, baik secara lokal maupun menggunakan teknologi AWS.	1. Siswa mampu memahami dasar-dasar jaringan komputer, seperti tipe-tipe jaringan, cara kerja internet, hingga model jaringan yang menjadi referensi global. 2. Siswa mampu mengenal protokol di setiap layer pada model jaringan TCP/IP sehingga lebih mudah memahami cara kerja internet. 3. Siswa mampu membedah segala aspek tentang IP address, mulai dari anatomi, subnetting, hingga menggunakan beberapa tools untuk konfigurasi IP address. 4. Siswa mampu menelaah DNS, mulai dari anatomi DNS, cara kerja DNS, hingga memanfaatkan serangkaian tools untuk konfigurasi DNS. 5. Siswa mampu mengidentifikasi pentingnya keamanan jaringan guna menghalau serangan yang kerap terjadi sekaligus praktik menggunakan beberapa tools untuk konfigurasi keamanan jaringan. 6. Siswa mampu menyebutkan layanan-layanan jaringan yang tersedia di AWS sekaligus mempraktikkannya.	25 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/387).	Penilaian dilakukan melalui 7 kali exam dan 1 kali submission project yang dilakukan di platform LMS Dicoding.
Menjadi Linux System Administrator	Di akhir kelas, siswa mampu mengoperasikan seluk-beluk sistem operasi Linux untuk menjadi seorang Linux System Administrator.	1. Mengenal sistem operasi Linux lebih dalam. 2. Mengerti bagaimana cara berinteraksi dengan sistem operasi Linux. 3. Mengetahui bagaimana cara menulis shell scripting. 4. Memahami bagaimana cara bekerja dengan filesystem. 5. Mengerti cara mengelola service	25 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/423).	Penilaian dilakukan melalui 8 kali exam dan 2 kali submission project yang dilakukan di platform LMS Dicoding.

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
		dengan baik. 6. Memahami praktik-praktik dalam memelihara sistem. 7. Menguasai topik-topik lanjutan seperti VM dan Container.			
Belajar Implementasi CI/CD	Di akhir kelas, siswa dapat memahami bagaimana cara mengimplementasikan CI/CD sebagai salah satu praktik dalam penerapan DevOps	1. Mengetahui pengertian dan manfaat dari praktik CI/CD. 2. Mengetahui beberapa strategi dalam melakukan deployment. 3. Memahami bagaimana cara membuat CI/CD pipeline menggunakan GitHub Actions. 4. Memahami bagaimana cara membuat CI/CD pipeline menggunakan layanan-layanan AWS. 5. Mengetahui bagaimana cara mengamankan CI/CD pipeline yang dibuat di AWS.	30 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/428).	Penilaian dilakukan melalui 6 kali exam dan 2 kali submission project yang dilakukan di platform LMS Dicoding.
Belajar Membangun Arsitektur Microservices	Di akhir kelas, siswa dapat memahami bagaimana cara mengimplementasikan CI/CD sebagai salah satu praktik dalam penerapan DevOps	1. Mengetahui pengertian dan manfaat dari praktik CI/CD. 2. Mengetahui beberapa strategi dalam melakukan deployment. 3. Memahami bagaimana cara membuat CI/CD pipeline menggunakan GitHub Actions. 4. Memahami bagaimana cara membuat CI/CD pipeline menggunakan layanan-layanan AWS. 5. Mengetahui bagaimana cara mengamankan CI/CD pipeline yang dibuat di AWS.	40 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/433).	Penilaian dilakukan melalui 6 kali exam dan 2 kali submission project yang dilakukan di platform LMS Dicoding.
Subtotal Pembelajaran Pengembang Machine Learning & DevOps			341 jam		

Lampiran 8. Silabus Pengembang Multi-Platform dan Back-End

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
Memulai Pemrograman Dengan Dart	Di akhir kelas, siswa dapat membuat program Dart dengan menggunakan IDE IntelliJ IDEA atau IDE Online seperti DartPad.	1. Mengenal apa itu Dart, karakteristiknya, dan platform apa saja yang didukung. 2. Menyiapkan tools apa saja yang diperlukan untuk membuat program pertama menggunakan Dart. 3. Memahami tentang teori dasar Dart dengan mempelajari konsep seperti menggunakan variabel, tipe data, operator, dan membuat fungsi dasar. 4. Memahami tentang bagaimana mengatur alur dari suatu program Dart, seperti percabangan dan perulangan. 5. Mempelajari tipe data dari Dart yang berguna untuk menyimpan beberapa objek data. 6. Memahami mengenai konsep OOP pada Dart, bagaimana membuat sebuah class, properti dan fungsi dari class, serta mensimulasikan problematika dunia nyata dalam paradigma Object-Oriented Programming (OOP). 7. Memahami konsep functional programming seperti perilaku fungsi, lambda, high-order, dan closures. 8. Memahami tentang type system atau aturan yang digunakan pada Dart. 9. Mempelajari bagaimana membuat dan menangani proses asynchronous pada Dart. 10. Mengenal panduan bagaimana menulis kode Dart yang sesuai dengan coding convention.	20 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/191).	Penilaian dilakukan melalui 11 kali exam yang dilakukan di platform LMS Dicoding.
Belajar Membuat Aplikasi Flutter untuk Pemula	Di akhir kelas, siswa dapat membuat aplikasi Flutter dengan memanfaatkan widget untuk menyusun layout-nya.	1. Mengenal tentang Flutter dan instalasi alat-alat yang dibutuhkan untuk pengembangan. 2. Mempelajari dasar-dasar dalam membuat aplikasi Flutter dan pengenalan tentang package manager. 3. Mengenal widget-widget dasar yang umum digunakan dalam	40 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/159).	Penilaian dilakukan melalui 4 kali exam dan 1 kali submission yang dilakukan di platform LMS Dicoding.

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
		<p>Flutter, navigasi antar halaman, dan bagaimana menambahkan asset serta gambar.</p> <p>4. Mampu melakukan build aplikasi Flutter ke dalam bentuk berkas APK, AAB, dan folder web.</p>			
Belajar Fundamental Aplikasi Flutter	Di akhir kelas, siswa dapat membuat aplikasi Flutter dengan mengimplementasikan fitur-fitur umum seperti mengambil data dari internet, penyimpanan lokal, notifikasi, dan testing.	<p>1. Mengenal tentang dasar Flutter, seperti widget, state, navigation, dan devtools.</p> <p>2. Mempelajari desain layout menggunakan berbagai jenis widget yang disediakan Flutter.</p> <p>3. Mempelajari bagaimana mengelola state dan data dalam lingkungan deklaratif UI seperti Flutter.</p> <p>4. Mempelajari bagaimana menerapkan localizations dan accessibility untuk menjangkau pengguna lebih luas.</p> <p>5. Mempelajari bagaimana mengambil data dari internet/API dan menampilkannya.</p> <p>6. Mempelajari bagaimana menjalankan tugas secara terjadwal meskipun aplikasi tidak dalam kondisi terbuka.</p> <p>7. Mempelajari pendekatan untuk menyimpan data secara lokal pada perangkat.</p> <p>8. Mempelajari bagaimana membagi satu basis kode aplikasi flutter menjadi beberapa versi aplikasi.</p> <p>9. Mempelajari bagaimana melakukan pengujian otomatis pada aplikasi Flutter.</p> <p>10. Mempelajari integrasi antara Flutter dan Firebase.</p>	70 jam	<p>Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/195).</p>	<p>Penilaian dilakukan melalui 10 kali exam dan 3 kali submission yang dilakukan di platform LMS Dicoding.</p>
Belajar Prinsip Pemrograman SOLID	Di akhir kelas, siswa dapat menerapkan konsep-konsep dasar OOP dan relasinya untuk menyelesaikan masalah pada software design dengan menggunakan 5 prinsip SOLID.	<p>Tujuan Akhir:</p> <p>1. Memahami 4 pilar OOP, yakni Inheritance, Encapsulation, Abstraction, dan Polymorphism.</p> <p>2. Mengetahui macam-macam hubungan antar objek di OOP, seperti association, aggregation, composition, dependency, generalization, specialization, dan implementation.</p> <p>3. Mempelajari 3 karakteristik penting dari</p>	15 jam	<p>Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/169).</p>	<p>Penilaian dilakukan melalui 5 kali exam yang dilakukan di platform LMS Dicoding.</p>

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
		design buruk yang perlu dihindari, yakni rigidity, fragility, dan immobility. 4. Memahami tujuan 5 prinsip SOLID dan studi kasusnya, yaitu Single Responsibility, Open Closed, Liskov Substitution, Interface Segregation, dan Dependency Inversion.			
Menjadi Flutter Developer Expert	Di akhir kelas, siswa dapat membuat aplikasi yang siap rilis dengan kode yang berkualitas dan mudah untuk dikembangkan terkait dengan feedback dari pengguna ke depannya.	1. Mampu mengembangkan aplikasi Flutter dengan menerapkan clean architecture untuk memudahkan pemeliharaan dan pengembangan. 2. Mampu menganalisis kebutuhan fitur untuk menghasilkan skenario pengujian. Kemudian menerapkan praktik TDD dalam membuat sebuah fitur baru. 3. Mampu membuat kustomisasi UI. 4. Mampu membuat modul-modul pengembangan pada proyek untuk aplikasi berskala besar. 5. Mampu memahami konsep Reactive Programming serta implementasinya menggunakan BLoC. 6. Mampu memahami praktik Continuous Integration sebagai proses otomatisasi untuk mempermudah pengujian aplikasi. 7. Mampu memahami praktik terbaik pengembangan aplikasi Flutter dalam aspek performa. 8. Mampu memahami beberapa cara yang bisa dilakukan untuk meningkatkan keamanan pada aplikasi, seperti Encryption, Obfuscation, dan SSL Pinning. 9. Mampu mengerti tahap akhir dalam sebuah pengembangan aplikasi ketika akan rilis.	85 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/199).	Penilaian dilakukan melalui 10 kali exam dan 2 kali submission yang dilakukan di platform LMS Dicoding.
Belajar Dasar Pemrograman JavaScript	Di akhir kelas, siswa dapat membuat program dengan JavaScript menggunakan Node.js dan Text Editor, seperti Visual Studio Code.	1. Mengenal apa itu JavaScript, bagaimana sejarahnya, dan alasan untuk mempelajarinya. 2. Mengenal dan bisa mengimplementasikan komponen-komponen dasar di JavaScript. 3. Mengerti dan bisa mengelola data yang	45 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/256).	Penilaian dilakukan melalui 12 kali exam yang dilakukan di platform LMS Dicoding.

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
		<p>lebih kompleks.</p> <p>4. Mengerti dan bisa mengimplementasikan function.</p> <p>5. Mengerti dan bisa mengimplementasikan object oriented programming.</p> <p>6. Mengerti dan bisa mengimplementasikan functional programming.</p> <p>7. Mengerti dan bisa menyiapkan lingkungan pengembangan di komputer siswa.</p> <p>8. Mengerti bagaimana beberapa berkas JavaScript dapat saling berkomunikasi satu sama lain.</p> <p>9. Mengerti dan bisa menangani eror yang mungkin muncul dalam aplikasi.</p> <p>10. Mengerti dan bisa mengimplementasikan concurrency.</p> <p>11. Mengerti dan bisa memanfaatkan NPM untuk mengelola package eksternal di dalam project.</p> <p>12. Mengerti dan bisa mengimplementasikan automated testing menggunakan package Jest.</p>			
Belajar Membuat Aplikasi Back-End untuk Pemula	Di akhir kelas, siswa dapat membuat aplikasi back-end sederhana berupa RESTful API menggunakan Node.js.	<p>1. Mengerti dasar pengetahuan back-end, seperti server dan client, Web server dan Web Service, REST, dan konsep di dalamnya.</p> <p>2. Mengerti dasar-sasar Node.js untuk back-end, seperti Node.js, Global Object, Process Object, Modularization, event module, Filesystem, dan teknik Stream.</p> <p>3. Membangun Web Services Menggunakan Node.js secara native maupun menggunakan framework Hapi</p> <p>4. Membangun RESTful api untuk fungsionalitas aplikasi nyata.</p> <p>5. Mampu deploy web services melalui Amazon EC2</p> <p>6. Mampu mengonsumsi dan menguji RESTful API menggunakan Postman</p>	45 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/261).	Penilaian dilakukan melalui 1 kali submission project yang dilakukan di platform LMS Dicoding.
Belajar Dasar UX Design	Di akhir kelas, siswa membuat sebuah portofolio berupa UX case study yang berisi keseluruhan	1. Peserta mampu memahami konsep dasar UX Design dan faktor yang mempengaruhinya, serta memahami proses desain dengan	23 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding	Penilaian dilakukan melalui 7 kali exam dan 1 kali submission project yang dilakukan di

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
	proses, mulai dari mencari permasalahan pengguna, mendefinisikan masalah, melakukan eksplorasi, membuat prototype, sampai menguji cobanya	menggunakan pendekatan Design Thinking. 2. Peserta mampu memahami bagaimana cara untuk mengetahui masalah utama (pain point) pengguna dengan membuat Empathy Map, User Persona, dan User Journey Map. Dilanjutkan dengan mendefinisikan masalah dalam Problem Statement dan melakukan eksplorasi untuk mencari solusi dengan menggunakan metode How Might We (HMW) dan Crazy Eight. 3. Peserta mampu mendefinisikan solusi dalam bentuk Goal Statement dan mengetahui cara membuat rancangan desain dengan membuat User Flow, Storyboard, dan Wireframe. 4. Peserta mampu memahami cara membuat Mockup dan High-Fidelity Prototype yang menarik dan interaktif dengan menerapkan prinsip desain visual. 5. Peserta mampu merencanakan dan melakukan usability study, mengidentifikasi insight, memodifikasi desain berdasarkan hasil riset, dan membagikan hasil desain kepada tim.		(https://www.dicoding.com/academies/335).	platform LMS Dicoding.
Subtotal Pembelajaran Pengembang Multiplatform - Backend		343 jam			

Lampiran 9. Silabus Pengembang Back-End dan DevOps

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
Cloud Practitioner Essentials (Belajar Dasar AWS Cloud)	Di akhir kelas, siswa dapat memahami AWS Cloud dengan segala jenis layanan, infrastruktur global, hingga harganya.	1. Memahami dasar-dasar komputasi cloud. 2. Mengetahui berbagai layanan AWS, dari komputasi, penyimpanan, jaringan, keamanan, pemantauan, hingga arsitektur. 3. Memahami konsep dasar dari Infrastruktur Global AWS. 4. Mengetahui konsep harga dan dukungan di AWS. 5. Mengetahui contoh arsitektur cloud di AWS	13 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/251).	Penilaian dilakukan melalui 12 kali exam yang dilakukan di platform LMS Dicoding.
Belajar Dasar Pemrograman JavaScript	Di akhir kelas, siswa dapat membuat program dengan JavaScript menggunakan Node.js dan Text Editor, seperti Visual Studio Code.	1. Mengenal apa itu JavaScript, bagaimana sejarahnya, dan alasan untuk mempelajarinya. 2. Mengenal dan bisa mengimplementasikan komponen-komponen dasar di JavaScript. 3. Mengerti dan bisa mengelola data yang lebih kompleks. 4. Mengerti dan bisa mengimplementasikan function. 5. Mengerti dan bisa mengimplementasikan object oriented programming. 6. Mengerti dan bisa mengimplementasikan functional programming. 7. Mengerti dan bisa menyiapkan lingkungan pengembangan di komputer siswa. 8. Mengerti bagaimana beberapa berkas JavaScript dapat saling berkomunikasi satu sama lain. 9. Mengerti dan bisa menangani eror yang mungkin muncul dalam aplikasi. 10. Mengerti dan bisa mengimplementasikan concurrency. 11. Mengerti dan bisa memanfaatkan NPM untuk mengelola package eksternal di dalam project. 12. Mengerti dan bisa mengimplementasikan automated testing menggunakan package Jest.	45 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/256).	Penilaian dilakukan melalui 12 kali exam yang dilakukan di platform LMS Dicoding.

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
Belajar Membuat Aplikasi Back-End untuk Pemula	Di akhir kelas, siswa dapat membuat aplikasi back-end sederhana berupa RESTful API menggunakan Node.js.	1. Mengerti dasar pengetahuan back-end, seperti server dan client, Web server dan Web Service, REST, dan konsep di dalamnya. 2. Mengerti dasar-sasar Node.js untuk back-end, seperti Node.js, Global Object, Process Object, Modularization, event module, Filesystem, dan teknik Stream. 3. Membangun Web Services Menggunakan Node.js secara native maupun menggunakan framework Hapi 4. Membangun RESTful api untuk fungsionalitas aplikasi nyata. 5. Mampu deploy web services melalui Amazon EC2 6. Mampu mengonsumsi dan menguji RESTful API menggunakan Postman	45 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/261).	Penilaian dilakukan melalui 1 kali submission project yang dilakukan di platform LMS Dicoding.
Belajar Fundamental Aplikasi Back-End	Di akhir kelas, siswa dapat membuat aplikasi back-end berupa RESTful API yang menerapkan Database, Authentication dan Authorization, Message Broker, Storage, dan Caching sesuai studi kasus yang ada dalam dunia nyata.	1. Dapat mengorganisir struktur kode menggunakan Hapi Plugin. 2. Dapat melakukan validasi data menggunakan Joi. 3. Dapat menggunakan database PostgreSQL pada AmazonRDS. 4. Dapat menerapkan Authentication dan Authorization pada RESTful API. 5. Dapat melakukan database normalisasi untuk membangun fitur yang membutuhkan kueri kompleks. 6. Dapat menggunakan message broker RabbitMQ pada Amazon MQ. 7. Dapat menggunakan storage menggunakan Amazon S3. 8. Dapat menerapkan caching menggunakan Redis dan Amazon ElastiCache.	90 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/271).	Penilaian dilakukan melalui 3 kali submission project yang dilakukan di platform LMS Dicoding.
Belajar Dasar-Dasar DevOps	Di akhir kelas, siswa dapat memahami bagaimana kultur dan praktik DevOps mampu meningkatkan nilai teknis serta bisnis perusahaan.	1. Mengetahui pengertian dan manfaat dari DevOps. 2. Memahami prinsip-prinsip yang ada pada DevOps. 3. Memahami kultur DevOps melalui CALMS Framework. 4. Mengerti praktik DevOps dengan mengenal DevOps Pipeline dan DevOps	15 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/382).	Penilaian dilakukan melalui 6 kali exam yang dilakukan di platform LMS Dicoding.

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
		Tools. 5. Mengetahui bagaimana kisah pengimplementasian DevOps pada kasus nyata dengan mengambil contoh perusahaan Amazon.			
Belajar Jaringan Komputer untuk Pemula	Di akhir kelas, siswa mampu memahami konsep dan praktik jaringan komputer, baik secara lokal maupun menggunakan teknologi AWS.	1. Siswa mampu memahami dasar-dasar jaringan komputer, seperti tipe-tipe jaringan, cara kerja internet, hingga model jaringan yang menjadi referensi global. 2. Siswa mampu mengenal protokol di setiap layer pada model jaringan TCP/IP sehingga lebih mudah memahami cara kerja internet. 3. Siswa mampu membedah segala aspek tentang IP address, mulai dari anatomi, subnetting, hingga menggunakan beberapa tools untuk konfigurasi IP address. 4. Siswa mampu menelaah DNS, mulai dari anatomi DNS, cara kerja DNS, hingga memanfaatkan serangkaian tools untuk konfigurasi DNS. 5. Siswa mampu mengidentifikasi pentingnya keamanan jaringan guna menghalau serangan yang kerap terjadi sekaligus praktik menggunakan beberapa tools untuk konfigurasi keamanan jaringan. 6. Siswa mampu menyebutkan layanan-layanan jaringan yang tersedia di AWS sekaligus mempraktikkannya.	25 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/387).	Penilaian dilakukan melalui 7 kali exam dan 1 kali submission project yang dilakukan di platform LMS Dicoding.
Menjadi Linux System Administrator	Di akhir kelas, siswa mampu mengoperasikan seluk-beluk sistem operasi Linux untuk menjadi seorang Linux System Administrator.	1. Mengenal sistem operasi Linux lebih dalam. 2. Mengerti bagaimana cara berinteraksi dengan sistem operasi Linux. 3. Mengetahui bagaimana cara menulis shell scripting. 4. Memahami bagaimana cara bekerja dengan filesystem. 5. Mengerti cara mengelola service dengan baik. 6. Memahami praktik-praktik dalam memelihara sistem. 7. Menguasai topik-topik lanjutan seperti VM dan	25 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/423).	Penilaian dilakukan melalui 8 kali exam dan 2 kali submission project yang dilakukan di platform LMS Dicoding.

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
		Container.			
Belajar Implementasi CI/CD	Di akhir kelas, siswa dapat memahami bagaimana cara mengimplementasikan CI/CD sebagai salah satu praktik dalam penerapan DevOps	1. Mengetahui pengertian dan manfaat dari praktik CI/CD. 2. Mengetahui beberapa strategi dalam melakukan deployment. 3. Memahami bagaimana cara membuat CI/CD pipeline menggunakan GitHub Actions. 4. Memahami bagaimana cara membuat CI/CD pipeline menggunakan layanan-layanan AWS. 5. Mengetahui bagaimana cara mengamankan CI/CD pipeline yang dibuat di AWS.	30 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/428).	Penilaian dilakukan melalui 6 kali exam dan 2 kali submission project yang dilakukan di platform LMS Dicoding.
Belajar Membangun Arsitektur Microservices	Di akhir kelas, siswa mampu membangun arsitektur microservices dengan memperhatikan praktik-praktik terbaik.	1. Mengetahui pengertian dan manfaat dari container. 2. Menjelaskan apa yang dimaksud dengan microservices beserta praktik-praktik terbaiknya. 3. Memahami apa itu Kubernetes dan bagaimana cara penggunannya. 4. Mengerti bagaimana cara membangun serverless microservices dengan baik. 5. Mengerti bagaimana cara memantau aplikasi yang menggunakan arsitektur	40 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/433).	Penilaian dilakukan melalui 6 kali exam dan 2 kali submission project yang dilakukan di platform LMS Dicoding.
Subtotal Pembelajaran Pengembang Backend-Devops		328 jam			

Lampiran 10. Silabus Pengembang Back-End dan React

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
Belajar Dasar Pemrograman Web	Di akhir pelatihan, siswa dapat membuat sebuah website sederhana menggunakan kode pemrograman yang sesuai standar global.	1. Mengenal apa itu Website, serta kebutuhan dalam mengembangkan website (front-end). 2. Mengerti peran, fungsi, dan cara menuliskan struktur HTML. 3. Mengetahui berbagai macam tags elemen dan penggunaannya. 4. Dapat mengelompokkan konten dengan menggunakan element semantik.	55 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (www.dicoding.com/academies/123).	Penilaian dilakukan melalui 9 kali exam dan 1 kali submission project yang dilakukan di platform LMS Dicoding.
Belajar Membuat Front-End Web untuk Pemula	Di akhir kelas, siswa dapat membuat aplikasi front-end web yang interaktif serta memiliki fitur penyimpanan menggunakan web storage.	1. Mengerti peran dan fungsi dari HTML dan CSS pada pengembangan Website. 2. Mengenal JavaScript dan Fungsinya. 3. Menguasai Dasar Pemrograman JavaScript, seperti statement dan expression, variabel, komentar, tipe data, operator, percabangan, perulangan, objek dan array. 4. Mengetahui Document Object Model dan Browser Object Model 5. Mampu membuat website menjadi interaktif dengan teknik Manipulation DOM dan Eventing. 6. Mampu menggunakan Web Storage untuk media penyimpanan data pada Website.	46 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (www.dicoding.com/academies/315).	Penilaian dilakukan melalui 1 kali submission project yang dilakukan di platform LMS Dicoding.
Belajar Dasar Pemrograman JavaScript	Di akhir kelas, siswa dapat membuat program dengan JavaScript menggunakan Node.js dan Text Editor, seperti Visual Studio Code.	1. Mengenal apa itu JavaScript, bagaimana sejarahnya, dan alasan untuk mempelajarinya. 2. Mengenal dan bisa mengimplementasikan komponen-komponen dasar di JavaScript. 3. Mengerti dan bisa mengelola data yang lebih kompleks. 4. Mengerti dan bisa mengimplementasikan function. 5. Mengerti dan bisa mengimplementasikan object oriented programming. 6. Mengerti dan bisa mengimplementasikan functional programming. 7. Mengerti dan bisa	45 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/256).	Penilaian dilakukan melalui 12 kali exam yang dilakukan di platform LMS Dicoding.

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
		menyiapkan lingkungan pengembangan di komputer siswa. 8. Mengerti bagaimana beberapa berkas JavaScript dapat saling berkomunikasi satu sama lain. 9. Mengerti dan bisa menangani eror yang mungkin muncul dalam aplikasi. 10. Mengerti dan bisa mengimplementasikan concurrency. 11. Mengerti dan bisa memanfaatkan NPM untuk mengelola package eksternal di dalam project. 12. Mengerti dan bisa mengimplementasikan automated testing menggunakan package Jest.			
Belajar Membuat Aplikasi Back-End untuk Pemula	Di akhir kelas, siswa dapat membuat aplikasi back-end sederhana berupa RESTful API menggunakan Node.js.	1. Mengerti dasar pengetahuan back-end, seperti server dan client, Web server dan Web Service, REST, dan konsep di dalamnya. 2. Mengerti dasar-sasar Node.js untuk back-end, seperti Node.js, Global Object, Process Object, Modularization, event module, Filesystem, dan teknik Stream. 3. Membangun Web Services Menggunakan Node.js secara native maupun menggunakan framework Hapi 4. Membangun RESTful api untuk fungsionalitas aplikasi nyata. 5. Mampu deploy web services melalui Amazon EC2 6. Mampu mengonsumsi dan menguji RESTful API menggunakan Postman	45 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/261).	Penilaian dilakukan melalui 1 kali submission project yang dilakukan di platform LMS Dicoding.
Belajar Membuat Aplikasi Web dengan React	Di akhir kelas siswa dapat membuat aplikasi web dengan React yang memanfaatkan functional component, stateful component, dan controlled component.	1. Mengetahui pengertian, alasan, dan ekosistem pada React. 2. Memahami konsep dasar dari React, seperti element dan component, compositional model, declarative code, unidirectional data flow, dan React is Just JavaScript. 3. Mampu membuat React element menggunakan createElement ataupun JSX. 4. Mampu membuat React component melalui	35 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/403).	Penilaian dilakukan melalui 5 kali exam dan 1 kali submission project yang dilakukan di platform LMS Dicoding.

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
		fungsi (functional). 5. Mampu memberikan dan mengonsumsi nilai properti pada component. 6. Mampu membuat React Stateful Component melalui Class 7. Mampu menerapkan event handling pada Component 8. Mampu membuat, menggunakan dan menangani nilai pada element Form (Controlled Element) 9. Mampu men-debug guna memantau nilai dari properti dan state pada komponen.			
Belajar Fundamental Aplikasi Web dengan React	Di akhir kelas siswa dapat membuat aplikasi web dengan React yang menerapkan validasi properti, routing, component lifecycle, context, dan hooks.	1. Memahami cara mengimplementasikan validasi property component menggunakan PropTypes. 2. Memahami cara membuat routing yang mudah dengan memanfaatkan React Router. 3. Memahami cara memanfaatkan lifecycle component untuk menampilkan data dari API. 4. Memahami cara memanfaatkan React Context dalam berbagai macam kasus yang membutuhkan global state. 5. Memahami cara menggunakan Hooks API untuk memanfaatkan state di dalam functional component.	55 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/413).	Penilaian dilakukan melalui 6 kali exam dan 2 kali submission project yang dilakukan di platform LMS Dicoding.
Menjadi React Web Developer Expert	Di akhir kelas ini, siswa mampu membuat aplikasi React yang menerapkan state management baik, testable, cepat untuk di-deploy, serta menerapkan praktik terbaik yang ada.	1. Mampu membuat Custom Hook. 2. Mampu mengelola state yang kompleks dengan menggunakan Redux. 3. Mampu menerapkan pengujian otomatis seperti unit testing, integration testing, dan End-to-End Testing pada aplikasi React. 4. Mampu mendeploy aplikasi React dengan cepat menggunakan praktik CI/CD. 5. Mampu menerapkan praktik dan konsep advance di React seperti high-order component, render props, strict mode, dan linter.	70 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/418).	Penilaian dilakukan melalui 7 kali exam dan 2 kali submission project yang dilakukan di platform LMS Dicoding.
Subtotal Pembelajaran Pengembang React Back-end		351 jam			

Lampiran 11. Silabus Pengembang Multi-Platform dan Aplikasi Android

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
Memulai Pemrograman Dengan Dart	Di akhir kelas, siswa dapat membuat program Dart dengan menggunakan IDE IntelliJ IDEA atau IDE Online seperti DartPad.	1. Mengenal apa itu Dart, karakteristiknya, dan platform apa saja yang didukung. 2. Menyiapkan tools apa saja yang diperlukan untuk membuat program pertama menggunakan Dart. 3. Memahami tentang teori dasar Dart dengan mempelajari konsep seperti menggunakan variabel, tipe data, operator, dan membuat fungsi dasar. 4. Memahami tentang bagaimana mengatur alur dari suatu program Dart, seperti percabangan dan perulangan. 5. Mempelajari tipe data dari Dart yang berguna untuk menyimpan beberapa objek data. 6. Memahami mengenai konsep OOP pada Dart, bagaimana membuat sebuah class, properti dan fungsi dari class, serta mensimulasikan problematika dunia nyata dalam paradigma Object-Oriented Programming (OOP). 7. Memahami konsep functional programming seperti perilaku fungsi, lambda, high-order, dan closures. 8. Memahami tentang type system atau aturan yang digunakan pada Dart. 9. Mempelajari bagaimana membuat dan menangani proses asynchronous pada Dart. 10. Mengenal panduan bagaimana menulis kode Dart yang sesuai dengan coding convention.	20 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/191).	Penilaian dilakukan melalui 11 kali exam yang dilakukan di platform LMS Dicoding.
Belajar Membuat Aplikasi Flutter untuk Pemula	Di akhir kelas, siswa dapat membuat aplikasi Flutter dengan memanfaatkan widget untuk menyusun layout-nya.	1. Mengenal tentang Flutter dan instalasi alat-alat yang dibutuhkan untuk pengembangan. 2. Mempelajari dasar-dasar dalam membuat aplikasi Flutter dan pengenalan tentang package manager. 3. Mengenal widget-widget dasar yang umum digunakan dalam	40 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/159).	Penilaian dilakukan melalui 4 kali exam dan 1 kali submission yang dilakukan di platform LMS Dicoding.

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
		<p>Flutter, navigasi antar halaman, dan bagaimana menambahkan asset serta gambar.</p> <p>4. Mampu melakukan build aplikasi Flutter ke dalam bentuk berkas APK, AAB, dan folder web.</p>			
Belajar Fundamental Aplikasi Flutter	Di akhir kelas, siswa dapat membuat aplikasi Flutter dengan mengimplementasikan fitur-fitur umum seperti mengambil data dari internet, penyimpanan lokal, notifikasi, dan testing.	<p>1. Mengenal tentang dasar Flutter, seperti widget, state, navigation, dan devtools.</p> <p>2. Mempelajari desain layout menggunakan berbagai jenis widget yang disediakan Flutter.</p> <p>3. Mempelajari bagaimana mengelola state dan data dalam lingkungan deklaratif UI seperti Flutter.</p> <p>4. Mempelajari bagaimana menerapkan localizations dan accessibility untuk menjangkau pengguna lebih luas.</p> <p>5. Mempelajari bagaimana mengambil data dari internet/API dan menampilkannya.</p> <p>6. Mempelajari bagaimana menjalankan tugas secara terjadwal meskipun aplikasi tidak dalam kondisi terbuka.</p> <p>7. Mempelajari pendekatan untuk menyimpan data secara lokal pada perangkat.</p> <p>8. Mempelajari bagaimana membagi satu basis kode aplikasi flutter menjadi beberapa versi aplikasi.</p> <p>9. Mempelajari bagaimana melakukan pengujian otomatis pada aplikasi Flutter.</p> <p>10. Mempelajari integrasi antara Flutter dan Firebase.</p>	70 jam	<p>Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/195).</p>	<p>Penilaian dilakukan melalui 10 kali exam dan 3 kali submission yang dilakukan di platform LMS Dicoding.</p>
Belajar Prinsip Pemrograman SOLID	Di akhir kelas, siswa dapat menerapkan konsep-konsep dasar OOP dan relasinya untuk menyelesaikan masalah pada software design dengan menggunakan 5 prinsip SOLID.	<p>1. Memahami 4 pilar OOP, yakni Inheritance, Encapsulation, Abstraction, dan Polymorphism.</p> <p>2. Mengetahui macam-macam hubungan antar objek di OOP, seperti association, aggregation, composition, dependency, generalization, specialization, dan implementation.</p> <p>3. Mempelajari 3 karakteristik penting dari</p>	15 jam	<p>Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/169).</p>	<p>Penilaian dilakukan melalui 5 kali exam yang dilakukan di platform LMS Dicoding.</p>

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
		design buruk yang perlu dihindari, yakni rigidity, fragility, dan immobility. 4. Memahami tujuan 5 prinsip SOLID dan studi kasusnya, yaitu Single Responsibility, Open Closed, Liskov Substitution, Interface Segregation, dan Dependency Inversion.			
Menjadi Flutter Developer Expert	Di akhir kelas, siswa dapat membuat aplikasi yang siap rilis dengan kode yang berkualitas dan mudah untuk dikembangkan terkait dengan feedback dari pengguna ke depannya	1. Mampu mengembangkan aplikasi Flutter dengan menerapkan clean architecture untuk memudahkan pemeliharaan dan pengembangan. 2. Mampu menganalisis kebutuhan fitur untuk menghasilkan skenario pengujian. Kemudian menerapkan praktik TDD dalam membuat sebuah fitur baru. 3. Mampu membuat kustomisasi UI. 4. Mampu membuat modul-modul	85 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/199).	Penilaian dilakukan melalui 10 kali exam dan 2 kali submission yang dilakukan di platform LMS Dicoding.
Belajar Dasar UX Design	Di akhir kelas, siswa membuat sebuah portofolio berupa UX case study yang berisi keseluruhan proses, mulai dari mencari permasalahan pengguna, mendefinisikan masalah, melakukan eksplorasi, membuat prototype, sampai menguji cobanya.	1. Peserta mampu memahami konsep dasar UX Design dan faktor yang mempengaruhinya, serta memahami proses desain dengan menggunakan pendekatan Design Thinking. 2. Peserta mampu memahami bagaimana cara untuk mengetahui masalah utama (pain point) pengguna dengan membuat Empathy Map, User Persona, dan User Journey Map. Dilanjutkan dengan mendefinisikan masalah dalam Problem Statement dan melakukan eksplorasi untuk mencari solusi dengan menggunakan metode How Might We (HMW) dan Crazy Eight. 3. Peserta mampu mendefinisikan solusi dalam bentuk Goal Statement dan mengetahui cara membuat rancangan desain dengan membuat User Flow, Storyboard, dan Wireframe. 4. Peserta mampu memahami cara membuat Mockup dan High-Fidelity Prototype yang menarik dan interaktif dengan	23 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/335).	Penilaian dilakukan melalui 7 kali exam dan 1 kali submission project yang dilakukan di platform LMS Dicoding.

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
		menerapkan prinsip desain visual. 5. Peserta mampu merencanakan dan melakukan usability study, mengidentifikasi insight, memodifikasi desain berdasarkan hasil riset, dan membagikan hasil desain kepada tim.			
Memulai Pemrograman Dengan Kotlin	Di akhir kelas, siswa paham tentang konsep dasar bahasa pemrograman, functional programming, serta object-oriented programming (OOP) dengan menggunakan Kotlin.	1. Mempelajari bahasa Kotlin dan karakteristiknya, serta bagaimana Kotlin berperan dalam pengembangan aplikasi. 2. Memahami tentang teori dasar Kotlin. Dengan mempelajari konsep dasar dari Kotlin, seperti menggunakan tipe data, membuat fungsi dasar, menangani tipe data null, dan membuat String Template. 3. Mempelajari perbedaan antara expression dan statement, cara menggunakan enumeration, cara mengontrol aliran program menggunakan ekspresi if dan when, dan bagaimana menerapkan perulangan menggunakan while, do-while, dan break and continue. 4. Mengetahui bagaimana mengelola data di Kotlin dengan menggunakan data class dan collection. 5. Memahami konsep function programing dengan mempelajari extension, lambda, higher-order function, recursion serta melihat contoh penerapannya pada collection operator dan scope function. 6. Memahami konsep OOP (Object-Oriented Programming) dengan mempelajari inheritance, abstract class, interface, visibility modifier, dan overloading. 7. Mengetahui konsep Generic untuk membuat kelas yang dapat menerima berbagai jenis tipe data. 8. Mempelajari Coroutine sebagai solusi untuk menjalankan concurrency pada Kotlin.	50 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/80).	Penilaian dilakukan melalui 1 kali exam dan 1 kali submission project yang dilakukan di platform LMS Dicoding.

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi / Jam Pembelajaran	Sumber Daya Pembelajaran	Cara Penilaian
Belajar Membuat Aplikasi Android untuk Pemula	Di akhir kelas, siswa dapat membuat aplikasi Android yang dapat menampilkan list dan detail data.	1. Memahami penggunaan Android Studio untuk membuat dan menjalankan project serta mengekspornya menjadi APK. 2. Mempelajari konsep dasar Activity dan lifecycle-nya untuk mengambil dan menampilkan data. 3. Memahami cara untuk berpindah halaman pada aplikasi, baik yang di dalam aplikasi dengan explicit intent maupun di luar aplikasi dengan implicit intent. 4. Mengetahui berbagai macam layout dan komponen yang dapat digunakan untuk mendesain aplikasi. 5. Memahami implementasi style & theme untuk mempersingkat pembuatan desain aplikasi. 6. Memahami konsep dan cara kerja RecyclerView untuk menampilkan data yang berjumlah banyak secara dinamis dan contoh penggunaan Layout Manager.	40 jam	Materi bacaan elektronik, video, exam yang tersedia di dalam LMS Dicoding (https://www.dicoding.com/academies/51).	Penilaian dilakukan melalui 5 kali exam dan 1 kali submission project yang dilakukan di platform LMS Dicoding.
Subtotal Pembelajaran Pengembang Multiplatform Android			343 jam		

Lampiran 12. Silabus Non-Teknis & Soft-skill

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi Pembelajaran (jam)	Sumber Daya Pembelajaran	Cara Penilaian
Inisiatif, Proaktif, Bertanggung jawab	Siswa dapat secara aktif melaporkan progress dan kendala yang dihadapi selama program	Peserta akan melaporkan progress pembelajaran serta mendapatkan masukan dari pembimbing. ±2-3 jam per minggu Peserta akan melakukan team meeting berkala dengan tim SIB Dicoding ±2 jam per bulan	50 jam	Form pelaporan progress dan kendala	Penilaian dilakukan berdasarkan ketepatan waktu pelaporan dan keaktifan siswa melaporkan kendala yang dihadapi.
Mempersiapkan diri, berkontribusi aktif, dan kemauan untuk belajar	Siswa hadir dengan persiapan dan aktif berkontribusi pada sesi live review materi dan pemberian materi praktis dari industri	Peserta akan mereview materi yang telah dipelajari dan dapat menanyakan hal-hal yang ingin diketahui kepada expert. 2 jam per dua minggu	12 jam	Google Meet Google Form	Penilaian dilakukan berdasarkan kualitas dan kuantitas pertanyaan yang diajukan oleh peserta serta kuis yang diadakan dalam sesi
Pembelajaran Soft skills	Siswa mengerti terkait Time Scheduling, Critical/Design Thinking, Effective Communication, dan Digital Branding.	Peserta akan menerima materi softskill dari mentor untuk menunjang kemampuan non-teknis peserta. 7 sesi, masing-masing 2 jam 7 tugas, masing-masing 8 jam	70 jam	Google Meet Google Form	Penilaian dilakukan berdasarkan kualitas tugas serta latihan yang diselenggarakan dalam sesi.
Persiapan Karir	Di akhir kelas, siswa dapat menyimpulkan dan memilih jalur karier pada bidang Software Development yang sesuai dengan diri mereka beserta mengerti hal-hal yang harus mereka persiapkan untuk mencapai dan menjalani karier tersebut.	1. Mengidentifikasi opsi jalur karier yang tersedia di bidang Software Development. 2. Menyimpulkan jalur karier yang tepat di bidang Software Development. 3. Menerapkan praktek terbaik dari developer berpengalaman untuk mengambil jalur karier. 4. Mengetahui segala persiapan yang perlu dilakukan untuk mencapai salah satu jalur karier Software Developer tertentu. 5. Peserta akan menerima 2 sesi, masing-masing 60 menit.	9 jam	1. Materi bacaan elektronik, video, kuis, exam yang tersedia di dalam LMS Dicoding (www.dicoding.com/academies/292). 2. Google Meet/Youtube	Penilaian dilakukan melalui 5 kali exam yang dilakukan di platform LMS Dicoding.
Refleksi diri	Siswa mampu menceritakan kembali dan melaporkan hal yang didapatkan selama proses pembelajaran dalam bentuk lisan dan tulisan.	Siswa mengisi logbook, memberikan laporan ke dosen pembimbing akademik, serta refleksi pembelajaran secara mandiri. ± 8 jam per minggu	150 jam	Platform Kampus Merdeka dan mekanisme pelaporan yang ditentukan oleh dosen pembimbing akademik.	Penilaian dilakukan berdasarkan ketepatan waktu pengisian logbook dan feedback yang diberikan oleh dosen pembimbing akademik.

Learning Objective	Tingkat Kompetensi	Detil Pembelajaran	Durasi Pembelajaran (jam)	Sumber Daya Pembelajaran	Cara Penilaian
Capstone Project / Proyek Akhir	Siswa mampu menyelesaikan proyek akhir, yakni pengembangan aplikasi/solusi yang dikerjakan untuk memvalidasi skill pengembangan produk dan menambah portfolio.	Siswa akan dikelompokkan dalam empat sampai lima peserta per grup untuk mengerjakan proyek tematik pada dunia nyata yang dapat membantu masyarakat. Meliputi proses persiapan, pengajuan topik, pengerjaan, dan evaluasi.	250 jam	Dicoding Challenge	<p>Penilaian dilakukan berdasarkan kualitas produk yang dihasilkan dalam capstone project serta dokumentasi project.</p> <p>Melibatkan Problem Solving, Design Thinking, dan Kolaborasi.</p>
Subtotal Aktivitas Terkait Pembelajaran Non-teknis		541 jam (56 hari)			

Lampiran 13. Gambaran Konversi SKS per Paket

Pengembang Front-End Web dan Back-End

Kelas/Aktivitas	Jam	Rekomendasi SKS
Memulai Dasar Pemrograman untuk Menjadi Pengembang Software ke Dasar Pemrograman	13	1
Pengenalan Ke Logika Pemrograman	6	
Belajar Dasar Git dengan Github	15	
Belajar Dasar Pemrograman Web	55	3
Belajar Membuat Front-End Web untuk Pemula	46	
Belajar Fundamental Front-End Web Development	70	2
Menjadi Front-End Web Developer Expert	90	3
Belajar Dasar Pemrograman JavaScript	45	3
Belajar Membuat Aplikasi Back-End untuk Pemula	45	4
Capstone / Final Project	250	4
Soft skill & Career Development	291	4
Total	926	20

Pengembang Aplikasi Multi-Platform dan Back-End

Kelas/Aktivitas	Jam	Rekomendasi SKS
Memulai Dasar Pemrograman untuk Menjadi Pengembang Software ke Dasar Pemrograman	13	1
Pengenalan Ke Logika Pemrograman	6	
Belajar Dasar Git dengan Github	15	
Memulai Pemrograman Dengan Dart	20	1
Belajar Prinsip Pemrograman SOLID	15	
Belajar Membuat Aplikasi Flutter untuk Pemula	40	2
Belajar Dasar UX Design	23	
Belajar Fundamental Aplikasi Flutter	70	
Menjadi Flutter Developer Expert	85	3
Belajar Dasar Pemrograman JavaScript	45	3
Belajar Membuat Aplikasi Back-End untuk Pemula	45	4
Capstone / Final Project	250	4
Soft skill & Career Development	291	4
Total	918	20

Pengembang Aplikasi Android

Kelas/Aktivitas	Jam	Rekomendasi SKS
Memulai Dasar Pemrograman untuk Menjadi Pengembang Software ke Dasar Pemrograman	13	1
Pengenalan Ke Logika Pemrograman	6	
Belajar Dasar Git dengan Github	15	
Memulai Pemrograman Dengan Kotlin	50	2
Belajar Fundamental Aplikasi Android	150	5
Belajar Pengembangan Aplikasi Android Intermediate	100	3
Belajar Prinsip Pemrograman SOLID	15	1
Belajar Dasar UX Design	23	
Capstone / Final Project	250	4
Soft skill & Career Development	291	4
Total	913	20

Pengembang Aplikasi Android dan Back-End

Kelas/Aktivitas	Jam	Rekomendasi SKS
Memulai Dasar Pemrograman untuk Menjadi Pengembang Software ke Dasar Pemrograman	13	1
Pengenalan Ke Logika Pemrograman	6	
Belajar Dasar Git dengan Github	15	
Memulai Pemrograman Dengan Kotlin	50	2
Belajar Fundamental Aplikasi Android	150	5
Belajar Prinsip Pemrograman SOLID	15	1
Belajar Dasar UX Design	23	
Belajar Membuat Aplikasi Back-End untuk Pemula	45	3
Belajar Dasar Pemrograman JavaScript	45	
Capstone / Final Project	250	4
Soft skill & Career Development	291	4
Total	903	20

Pengembang Front-End Web dan React

Kelas/Aktivitas	Jam	Rekomendasi SKS
Memulai Dasar Pemrograman untuk Menjadi Pengembang Software ke Dasar Pemrograman	13	1
Pengenalan Ke Logika Pemrograman	6	
Belajar Dasar Git dengan Github	15	
Belajar Dasar Pemrograman Web	55	3
Belajar Membuat Front-End Web untuk Pemula	46	
Belajar Fundamental Front-End Web Development	70	2
Belajar Membuat Aplikasi Web dengan React	35	1
Belajar Fundamental Aplikasi Web dengan React	55	2
Menjadi Front-End Web Developer Expert	90	3
Capstone / Final Project	250	4
Soft skill & Career Development	291	4
Total	926	20

Pengembang Machine Learning dan Front-End Web

Kelas/Aktivitas	Jam	Rekomendasi SKS
Memulai Dasar Pemrograman untuk Menjadi Pengembang Software ke Dasar Pemrograman	13	1
Pengenalan Ke Logika Pemrograman	6	
Belajar Dasar Git dengan Github	15	
Belajar Dasar Visualisasi Data	16	1
Memulai Pemrograman Dengan Python	20	
Belajar Machine Learning untuk Pemula	30	1
Belajar Pengembangan Machine Learning	40	4
Machine Learning Terapan	40	
Evaluasi Penguasaan Machine Learning	15	
Belajar Dasar Pemrograman Web	55	3
Belajar Membuat Front-End Web untuk Pemula	46	
Belajar Fundamental Front-End Web Development	70	
Capstone / Final Project	250	4
Soft skill & Career Development	291	4
Total	907	20

Pengembang Machine Learning dan DevOps

Kelas/Aktivitas	Jam	Rekomendasi SKS
Memulai Dasar Pemrograman untuk Menjadi Pengembang Software ke Dasar Pemrograman	13	1
Pengenalan Ke Logika Pemrograman	6	
Belajar Dasar Git dengan Github	15	
Belajar Dasar Visualisasi Data	16	1
Memulai Pemrograman Dengan Python	20	
Belajar Machine Learning untuk Pemula	30	1
Belajar Pengembangan Machine Learning	40	3
Machine Learning Terapan	40	
Evaluasi Penguasaan Machine Learning	15	
Machine Learning Operations	45	2
Belajar Dasar-Dasar DevOps	15	1
Belajar Jaringan Komputer untuk Pemula	25	
Menjadi Linux System Administrator	25	
Belajar Implementasi CI/CD	30	3
Belajar Membangun Arsitektur Microservices	40	
Capstone / Final Project	250	
Soft skill & Career Development	291	4
Total	916	20

Pengembang Back-End dan DevOps

Kelas/Aktivitas	Jam	Rekomendasi SKS
Memulai Dasar Pemrograman untuk Menjadi Pengembang Software ke Dasar Pemrograman	13	1
Pengenalan Ke Logika Pemrograman	6	
Belajar Dasar Git dengan Github	15	
Cloud Practitioner Essentials (Belajar Dasar AWS Cloud)	13	1
Belajar Dasar Pemrograman JavaScript	45	3
Belajar Membuat Aplikasi Back-End untuk Pemula	45	
Belajar Fundamental Aplikasi Back-End	90	3
Belajar Dasar-Dasar DevOps	15	1
Belajar Jaringan Komputer untuk Pemula	25	
Menjadi Linux System Administrator	25	
Belajar Implementasi CI/CD	30	3
Belajar Membangun Arsitektur Microservices	40	
Capstone / Final Project	250	
Soft skill & Career Development	291	4
Total	903	20

Pengembang Back-End dan React

Kelas/Aktivitas	Jam	Rekomendasi SKS
Memulai Dasar Pemrograman untuk Menjadi Pengembang Software ke Dasar Pemrograman	13	1
Pengenalan Ke Logika Pemrograman	6	
Belajar Dasar Git dengan Github	15	
Belajar Dasar Pemrograman Web	55	3
Belajar Membuat Front-End Web untuk Pemula	46	
Belajar Dasar Pemrograman JavaScript	45	
Belajar Membuat Aplikasi Back-End untuk Pemula	45	3
Belajar Membuat Aplikasi Web dengan React	35	
Belajar Fundamental Aplikasi Web dengan React	55	
Menjadi React Web Developer Expert	70	2
Capstone / Final Project	250	4
Soft skill & Career Development	291	4
Total	926	20

Pengembang Multi-Platform dan Aplikasi Android

Kelas/Aktivitas	Jam	Rekomendasi SKS
Memulai Dasar Pemrograman untuk Menjadi Pengembang Software ke Dasar Pemrograman	13	1
Pengenalan Ke Logika Pemrograman	6	
Belajar Dasar Git dengan Github	15	
Memulai Pemrograman Dengan Dart	20	1
Belajar Prinsip Pemrograman SOLID	15	
Belajar Membuat Aplikasi Flutter untuk Pemula	40	
Belajar Dasar UX Design	23	2
Belajar Fundamental Aplikasi Flutter	70	
Menjadi Flutter Developer Expert	85	
Memulai Pemrograman Dengan Kotlin	50	3
Belajar Membuat Aplikasi Android untuk Pemula	40	
Capstone / Final Project	250	
Soft skill & Career Development	291	4
Total	918	20