



SILABUS

Coding for Non-Coders: Desain Aplikasi Sederhana Low-code

Materi

- 01** Dasar Computational Thinking
- 02** Analisis Masalah, Algoritma & Flowchart
- 03** Pengenalan Python untuk Logika Aplikasi
- 04** Variabel, Struktur Data & Input/Output
- 05** Percabangan & Perulangan
- 06** Fungsi, Modularitas & Debugging
- 07** Project Development: Blueprint Aplikasi
- 08** Prototipe Aplikasi Low-code & Presentasi

I. INFORMASI DIKLAT

Nama Diklat	: Coding for Non-Coders: Desain Aplikasi Sederhana Low-code
Moda Pembelajaran	: Daring
Lama Waktu Diklat	: 48 JP
Kuota Peserta	: Maksimal 100 peserta per kelas
Jenjang Program	: Sertifikasi Berstandar Internasional
Tujuan Pelatihan	: Pelatihan ini dirancang untuk membekali Guru SMK dari berbagai program keahlian dengan kemampuan dasar computational thinking dan logika pemrograman menggunakan Python sebagai fondasi untuk merancang aplikasi sederhana berbasis low-code. Program ini mengacu pada standar industri IT Specialist: Computational Thinking dan IT Specialist: Python, sehingga peserta memahami cara memecahkan masalah secara terstruktur, membangun alur logika, serta menyiapkan blueprint aplikasi sebelum dikembangkan melalui platform low-code seperti AppSheet, Glide, atau Power Apps. Pelatihan menekankan pada kemampuan guru untuk menerapkan logika pemrograman ke dalam solusi nyata di sekolah melalui pembuatan prototipe aplikasi low-code.

II. PERSYARATAN PESERTA

1. Guru SMK dari seluruh bidang keahlian.
2. Tidak diwajibkan memiliki pengalaman coding atau pemrograman sebelumnya.
3. Mampu mengoperasikan komputer dasar (navigasi file, browser, mengetik).
4. Memiliki laptop dan koneksi internet stabil.
5. Memiliki akun Google atau Microsoft untuk prototipe low-code.

III. FASILITAS PESERTA

Setiap peserta program berhak mendapatkan:

1. Akses Lengkap Pelatihan Kompetensi
2. Akses Gratis Ujian Sertifikasi Resmi Berstandar Internasional
3. Akses Gratis Latihan Soal Simulasi Ujian Sertifikasi
4. Bantuan paket internet

IV. OUTCOME DIKLAT

Setelah mengikuti pelatihan ini, peserta mampu:

1. Memahami konsep computational thinking dan langkah-langkah pemecahan masalah digital.
2. Menyusun flowchart dan algoritma dasar untuk solusi aplikasi.
3. Menggunakan Python untuk memahami logika dasar seperti variabel, percabangan, perulangan, dan fungsi.

4. Menguji logika pemrograman menggunakan Python sebagai simulasi sebelum dibawa ke platform low-code.
5. Mendesain struktur data sederhana.
6. Membangun prototipe aplikasi sederhana berbasis low-code.
7. Menghasilkan blueprint logika dan aplikasi low-code sederhana yang dapat digunakan untuk kebutuhan sekolah.
8. Menjelaskan konsep dan proses pembangunan aplikasi kepada peserta didik SMK.

V. SERTIFIKASI INTERNASIONAL

Setiap program pelatihan dilengkapi dengan ujian sertifikasi industri berstandar global. Sertifikasi ini membuktikan bahwa peserta tidak hanya mengikuti pelatihan, tetapi juga lulus verifikasi kompetensi oleh lembaga sertifikasi resmi. Ujian dilaksanakan secara daring melalui *virtual machine* dengan *proctoring*, dengan hasil ujian muncul secara real-time, dan peserta mendapatkan pendampingan mulai dari persiapan belajar, latihan simulasi, hingga diberikan 1 kali kesempatan retake apabila belum mencapai skor minimum 700 (skor maksimum 1000).

										
EXAM SCORE REPORT										
CANDIDATE					EXAM					
[Redacted]					Microsoft Excel (Office 2019)					
[Redacted]					Exam reference # [Redacted]					
[Redacted]@gmail.com					Date: January 16, 2023					
[Redacted]					ID: [Redacted]					
RESULTS	100	200	300	400	500	600	700	800	900	1000
Required Score										
Your Score										
SECTION ANALYSIS					FINAL SCORE					
Manage Worksheets and Workbooks					Required Score					
					Your Score					
Manage Data Cells and Ranges										
Manage Tables and Table Data										
Perform Operations by using Formulas and Functions										
Manage Charts										
					OUTCOME					
					Pass					

Gambar 1. Contoh Score Report



Gambar 2. Contoh Sertifikat Kompetensi Internasional

Peserta yang lulus akan menerima sertifikat internasional serta score report sebagai bukti kompetensi yang dapat ditampilkan pada CV, portofolio, atau profil profesional. Lulusan ujian juga dinyatakan memiliki capaian yang setara dengan 150 jam teori dan praktik. Sertifikasi ini memberikan keunggulan kompetitif, karena memvalidasi keterampilan secara objektif sesuai standar industri. Berikut contoh sertifikat dan score report yang akan diperoleh peserta.

VI. ALUR PELAKSANAAN DIKLAT

Seluruh rangkaian berlangsung secara daring melalui Zoom dan dirancang agar peserta dapat belajar secara efektif hanya dalam 2 jam per hari.

No.	Nama Kegiatan	Tanggal Pelaksanaan
1.	Onboarding Peserta oleh Mitra	5 Desember 2025
2.	Pembelajaran Pertemuan ke-1	8 Desember 2025
3.	Pembelajaran Pertemuan ke-2	9 Desember 2025
4.	Pembelajaran Pertemuan ke-3	10 Desember 2025
5.	Pembelajaran Pertemuan ke-4	11 Desember 2025
6.	Pembelajaran Pertemuan ke-5	12 Desember 2025
7.	Pembelajaran Pertemuan ke-6	15 Desember 2025
8.	Pembelajaran Pertemuan ke-7	16 Desember 2025
9.	Pembelajaran Pertemuan ke-8	17 Desember 2025
10.	Finalisasi dan Presentasi Tugas Akhir	18-19 Desember 2025
11.	Penjadwalan & Periode Ujian Sertifikasi Internasional	Desember 2025

VII. KURIKULUM DIKLAT

No	Mata Diklat	Pengetahuan (JP)	Keterampilan (JP)	Total JP
1	Dasar Computational Thinking	2	3	5
2	Analisis Masalah, Algoritma & Flowchart	1	4	5
3	Pengenalan Python untuk Logika Aplikasi	1	4	5
4	Variabel, Struktur Data & Input/Output	1	4	5

No	Mata Diklat	Pengetahuan (JP)	Keterampilan (JP)	Total JP
5	Percabangan & Perulangan	1	6	7
6	Fungsi, Modularitas & Debugging	1	4	5
7	Project Development: Blueprint Aplikasi	-	5	5
8	Prototipe Aplikasi Low-code & Presentasi	-	6	6
TOTAL		7 JP	36 JP	48 JP

VIII. SILABUS DIKLAT

UNIT 1: Dasar Computational Thinking

Total = 6 JP

Elemen Kompetensi	KUK	Indikator	Materi	JP
Memahami prinsip CT	Menjelaskan konsep CT	Mengidentifikasi dekomposisi, pola, abstraksi, algoritma	CT Concepts & Components	2
Menganalisis masalah	Mendesripsikan masalah	Menentukan input–proses–output	Problem Analysis	2
Mengidentifikasi pola	Membandingkan data	Menemukan pola berulang	Pattern Recognition	1
Membuat solusi terstruktur	Menyusun langkah logis	Membuat algoritma sederhana	Algorithm Design	1

UNIT 2: Analisis Masalah, Algoritma & Flowchart

Total = 6 JP

Elemen Kompetensi	KUK	Indikator	Materi	JP
Membuat representasi masalah	Mengembangkan algoritma	Menyusun langkah-langkah solusi	Algorithmic Thinking	2
Membuat flowchart	Menggambar alur proses	Menentukan simbol & urutan logis	Flowcharting Basics	2
Menentukan keputusan	Mengidentifikasi kondisi	Memilah percabangan	Conditional Logic	1
Menguji algoritma	Validasi alur	Menemukan error logika	Algorithm Evaluation	1

UNIT 3: Pengenalan Python untuk Logika Aplikasi

Total = 6 JP

Elemen Kompetensi	KUK	Indikator	Materi	JP
Memahami Python sebagai alat logika	Menjelaskan fungsi Python	Interpreter, script, syntax dasar	Python Basics	2
Mengoperasikan Python	Menjalankan kode sederhana	Print, komentar, eksekusi	Running & Testing Code	1
Memahami tipe data dasar	Mengidentifikasi tipe	Integer, float, string, boolean	Data Types	2
Menggunakan variabel	Membuat variabel	Manipulasi nilai	Variables	1

UNIT 4: Variabel, Struktur Data & Input/Output

Total = 6 JP

Elemen Kompetensi	KUK	Indikator	Materi	JP
Menggunakan operator dasar	Mengolah data	Operator aritmatika & logika	Operators	2
Menggunakan struktur data	Mengelola list & dictionary	Membuat, mengambil, memodifikasi elemen	Lists & Dictionaries	2
Mengambil input	Menggunakan input()	Menerima & memproses input	Input/Output	1
Melakukan format output	Menampilkan hasil	f-string, concatenation	Output Formatting	1

UNIT 5: Percabangan & Perulangan

Total = 10 JP

Elemen Kompetensi	KUK	Indikator	Materi	JP
Menerapkan percabangan	Membuat kontrol kondisi	if, elif, else berjalan benar	Conditional Statements	3
Menggunakan perulangan	Membuat loop	for & while	Loops	3
Menggabungkan kondisi & loop	Mendesain alur logis	Nested conditions & loops	Nested Logic	2
Menguji logika	Debug sederhana	Menemukan kesalahan alur	Logical Debugging	2

UNIT 6: Fungsi, Modularitas & Debugging

Total = 6 JP

Elemen Kompetensi	KUK	Indikator	Materi	JP
Membuat fungsi sederhana	Menyusun <code>def</code>	Parameter & return	Functions Basics	2

Elemen Kompetensi	KUK	Indikator	Materi	JP
Menggunakan modularitas	Memecah program	Mengorganisasi script	Modular Programming	2
Menggunakan library dasar	Mengimpor modul	math, random, datetime	Python Modules	1
Melakukan debugging	Memperbaiki error	Syntax & runtime errors	Debugging Techniques	1

UNIT 7: Project Development Blueprint Aplikasi Low-Code

Total = 4 JP

Elemen Kompetensi	KUK	Indikator	Materi	JP
Mendesain solusi aplikasi	Menyusun requirement	Menentukan fitur, user flow	Application Design	1
Membuat blueprint logika	Menyiapkan algoritma	Flowchart lengkap aplikasi	Logic Blueprint	1
Menyusun struktur data	Menentukan tabel & field	Blueprint database	Data Structure Planning	1
Simulasi logika di Python	Menjalankan fungsi dasar	Python sebagai logic testbed	Logic Simulation	1

UNIT 8: Prototipe Aplikasi Low-code & Presentasi

Total = 4 JP

Elemen Kompetensi	KUK	Indikator	Materi	JP
Membangun prototipe aplikasi	Membuat aplikasi dasar	Form, list, detail, basic logic	Low-code Prototyping	1
Menghubungkan logika Python → low-code	Menggunakan blueprint	Logika berjalan benar di platform	Logic Integration	1
Melakukan testing aplikasi	Meninjau fungsional	Tidak ada error input/output	App Testing	1

Elemen Kompetensi	KUK	Indikator	Materi	JP
Presentasi proyek	Memaparkan aplikasi	Jelaskan masalah, solusi, fitur	Project Presentation	1

IX. METODE PEMBELAJARAN

Pelatihan ini menggunakan pendekatan daring berbasis kompetensi yang memadukan sesi sinkron dan asinkron agar peserta dapat memahami konsep computational thinking serta menerapkan logika pemrograman Python sebagai fondasi dalam merancang aplikasi low-code. Pembelajaran dimulai melalui sesi live di Zoom, di mana instruktur memberikan penjelasan konsep inti, mendemonstrasikan penggunaan Python untuk membangun logika sederhana, serta menunjukkan bagaimana alur dan algoritma dapat diterjemahkan ke dalam platform low-code.

Peserta kemudian melakukan praktik mandiri melalui Learning Management System (LMS), termasuk latihan membuat flowchart, menuliskan algoritma, membangun program Python sederhana (percabangan, perulangan, fungsi), dan menyiapkan blueprint logika aplikasi. Dengan pendekatan project-based learning, peserta didorong untuk mengembangkan prototipe aplikasi low-code berbasis kebutuhan sekolah, seperti aplikasi presensi sederhana, manajemen inventaris, atau input nilai harian.

Diskusi kolaboratif melalui breakout rooms digunakan untuk membahas rancangan aplikasi, logika proses, dan solusi pemrograman. Sesi mentoring disediakan untuk membantu peserta melakukan debugging, mengoptimalkan alur logika, serta menerjemahkan blueprint Python ke dalam platform low-code. Dengan kombinasi demonstrasi langsung, praktik terstruktur, diskusi kelompok, dan proyek aplikasi nyata, peserta memperoleh pengalaman belajar yang praktis dan relevan untuk diterapkan di kelas SMK.

X. ASESMEN & EVALUASI

Asesmen dalam pelatihan ini bertujuan memastikan peserta tidak hanya memahami konsep computational thinking, tetapi juga mampu mengimplementasikan logika pemrograman Python sebagai dasar untuk membangun aplikasi low-code. Evaluasi mencakup asesmen formatif dan sumatif yang dilakukan sepanjang pelatihan.

1. Asesmen Formatif bertujuan membantu peserta memperbaiki pemahaman sebelum masuk ke tahap pembuatan aplikasi, dan dilaksanakan selama proses pelatihan melalui:
 - a. Kuis mengenai konsep computational thinking dan logika dasar Python.
 - b. Latihan praktik seperti membuat flowchart, menyusun algoritma, dan membangun program Python sederhana.
 - c. Review progres blueprint aplikasi dan simulasi logika menggunakan Python.

- d. Umpan balik langsung dari instruktur selama sesi sinkron dan melalui tugas yang diunggah di LMS.
2. Asesmen Sumatif dilaksanakan pada akhir program dan mencakup:
 - a. Proyek Akhir: Blueprint Logika & Prototipe Aplikasi Low-code
Peserta merancang aplikasi sederhana (misalnya presensi, inventaris, atau input nilai) yang dilengkapi blueprint logika berbasis flowchart dan algoritma. Blueprint diuji melalui program Python sederhana untuk memvalidasi logika.
 - b. Pembuatan Prototipe Low-code
Peserta mengimplementasikan blueprint tersebut menjadi prototipe aplikasi low-code yang memuat form input, tampilan data, dan alur proses sederhana.
 - c. Testing & Debugging
Peserta mengevaluasi fungsionalitas aplikasi, memperbaiki error input/output, dan memastikan alur logika berjalan sesuai desain.
 - d. Presentasi Proyek
Peserta menjelaskan masalah yang dipecahkan, logika yang dikembangkan, hasil simulasi Python, serta prototipe aplikasi low-code yang dihasilkan.
 - e. Post-Test & Evaluasi Pelatihan
Peserta mengerjakan post-test untuk mengukur perkembangan pemahaman dan memberikan evaluasi pelatihan untuk peningkatan kualitas program.

XI. PERALATAN DAN PERLENGKAPAN PELATIHAN

1. Peralatan yang Harus Disiapkan Peserta
Peserta wajib menyiapkan perangkat berikut:
 - a. Laptop atau Desktop
Dengan spesifikasi minimum:
 - i. Untuk Windows
 1. Sistem operasi: Windows 10 atau lebih baru
 2. Prosesor: Intel Core i5 / AMD Ryzen 5 (≥ 2.6 GHz)
 3. RAM: Minimal 8 GB
 4. Resolusi layar: Minimal 1280×800
 5. Periferal wajib: keyboard lengkap dan mouse dua tombol (left-right click)
 - ii. Untuk macOS
 1. Sistem operasi: macOS Ventura (13) atau lebih baru
 2. Prosesor: Apple Silicon M1/M2 atau lebih baru
 3. RAM: Minimal 8 GB
 4. Resolusi layar: Minimal 1280×800
 5. Periferal wajib: keyboard lengkap dan mouse dua tombol
2. Perangkat Tambahan
 - a. Headset dengan mikrofon (disarankan noise-cancelling)
 - b. Kamera (built-in atau eksternal)

- c. Koneksi internet stabil minimal 10 Mbps
 - d. Browser versi terbaru (Google Chrome atau Microsoft Edge)
- 3. Tools dan Aplikasi yang Digunakan Selama Pelatihan
 - a. Python
 - b. VS Code atau editor teks lainnya
 - c. Google Workspace / Microsoft Office
 - d. Platform low-code (AppSheet / Glide / Power Apps)
 - e. Google Drive / OneDrive
 - f. Zoom untuk sesi live
 - g. Learning Management System (LMS)
- 4. Lingkungan Belajar Tambahan yang Disarankan
 - a. Ruangan yang tenang dengan pencahayaan baik
 - b. Sumber listrik stabil
 - c. Stabilizer/UPS (opsional) untuk mencegah gangguan saat build project