

Panduan Uji Kompetensi  
**Skema Sertifikasi Programmer**

[www.lsptik.or.id](http://www.lsptik.or.id)

## Daftar Isi

1.	Latar Belakang .....	3
2.	Persyaratan Dasar Pemohon Sertifikasi.....	3
3.	Hak Pemohon Sertifikasi dan Kewajiban Pemegang Sertifikat .....	3
4.	Persyaratan Sertifikasi .....	4
5.	Proses Sertifikasi .....	4
6.	Rincian Unit Kompetensi.....	5

# 1. Latar Belakang

Sertifikasi profesi merupakan upaya untuk memberikan pengakuan atas kompetensi yang dikuasai seseorang sesuai dengan Standard Kompetensi Kerja Nasional Indonesia (SKKNI), standar internasional atau standar khusus. Standar Kompetensi adalah pernyataan yang menguraikan keterampilan, pengetahuan dan sikap yang harus dilakukan saat bekerja serta penerapannya, sesuai dengan persyaratan yang ditetapkan oleh tempat kerja (industri).

Kompeten diartikan kemampuan dan kewenangan yang dimiliki oleh seseorang untuk melakukan suatu pekerjaan yang didasari oleh pengetahuan, ketrampilan dan sikap sesuai dengan unjuk kerja yang ditetapkan. Sertifikasi dilaksanakan dengan uji kompetensi melalui beberapa metode uji oleh asesor yang memiliki lisensi dari BNSP. Uji kompetensi dilaksanakan di Tempat Uji Kompetensi (TUK). TUK LSP TIK Indonesia merupakan tempat kerja atau lembaga yang dapat memberikan fasilitas pelaksanaan uji kompetensi yang telah diverifikasi oleh LSP TIK Indonesia.

## 2. Persyaratan Dasar Pemohon Sertifikasi

- 2.1. Minimal telah menyelesaikan pendidikan Sekolah Menengah Kejuruan (SMK) atau sederajat; Atau
- 2.2. Memiliki sertifikat pelatihan berbasis kompetensi yang sesuai dengan Skema Sertifikasi Programmer; Atau
- 2.3. Telah berpengalaman kerja pada lingkup yang sesuai dengan Skema Sertifikasi Programmer minimal 1 tahun secara berkelanjutan;

## 3. Hak Pemohon Sertifikasi dan Kewajiban Pemegang Sertifikat

- 3.1. Hak Pemohon
  - 3.1.1. Memperoleh penjelasan tentang gambaran proses sertifikasi sesuai dengan skema sertifikasi.
  - 3.1.2. Mendapatkan hak bertanya berkaitan dengan kompetensi.
  - 3.1.3. Memperoleh pemberitahuan tentang kesempatan untuk menyatakan, dengan alasan, permintaan untuk disediakan kebutuhan khusus sepanjang integritas asesmen tidak dilanggar, serta mempertimbangkan aturan yang bersifat Nasional.
  - 3.1.4. Memperoleh hak banding terhadap keputusan Sertifikasi.
  - 3.1.5. Memperoleh sertifikat kompetensi jika dinyatakan kompeten.
  - 3.1.6. Menggunakan sertifikat untuk promosi diri sebagai ahli dalam Skema Sertifikasi Programmer.
- 3.2. Kewajiban Pemegang Sertifikat
  - 3.2.1. Melaksanakan keprofesian sesuai dengan Skema Sertifikasi Programmer.
  - 3.2.2. Menjaga dan mentaati kode etik profesi secara sungguh-sungguh dan konsekuen.
  - 3.2.3. Menjamin bahwa sertifikat kompetensi tidak disalahgunakan.
  - 3.2.4. Menjamin terpelihara kompetensi yang sesuai dengan sertifikat kompetensi.

- 3.2.5. Menjamin bahwa seluruh pernyataan dan informasi yang diberikan adalah terbaru, benar dan dapat dipertanggung jawabkan.
- 3.2.6. Melaporkan rekaman kegiatan yang sesuai Skema Sertifikasi Programmer setiap 6 bulan sekali.
- 3.2.7. Membayar biaya sertifikasi.

## 4. Persyaratan Sertifikasi

Peserta uji kompetensi harus melengkapi persyaratan yang sesuai dengan Skema Sertifikasi Programmer yang meliputi:

- 4.1. Melengkapi isian formulir permohonan (FR-APL01) dan formulir asesmen mandiri (FR-APL02)
- 4.2. Menyerahkan persyaratan uji kompetensi
  - a. Pas foto 3x4 (3 lembar).
  - b. Copy identitas diri KTP/KK (1 lembar).
  - c. Copy ijazah terakhir (1 lembar).
  - d. Copy sertifikat yang relevan dengan Skema Sertifikasi Programmer, bila ada.
  - e. CV pengalaman / keterangan kerja yang relevan dengan Skema Sertifikasi Programmer, bila ada.
  - f. Portofolio yang relevan dengan Skema Sertifikasi Programmer, bila ada.

## 5. Proses Sertifikasi

- 5.1. Calon peserta uji kompetensi mengajukan permohonan sertifikasi melalui TUK (Tempat Uji Kompetensi) yang telah diverifikasi oleh LSP TIK Indonesia atau langsung melalui LSP TIK Indonesia.
- 5.2. Calon peserta uji kompetensi melengkapi isian formulir permohonan (FR-APL01) dan formulir asesmen mandiri (FR-APL02) serta menyerahkan persyaratan uji kompetensi.
- 5.3. Calon peserta uji kompetensi akan disetujui sebagai peserta uji kompetensi apabila persyaratan dan bukti-bukti yang disertakan telah memadai sesuai dengan skema sertifikasi.
- 5.4. Asesor dan peserta uji kompetensi menentukan tempat dan waktu pelaksanaan uji kompetensi yang telah disepakati oleh kedua belah pihak.
- 5.5. Setelah proses uji kompetensi, Asesor merekomendasikan kompeten (K) atau belum kompeten (BK) berdasarkan bukti-bukti yang telah dikumpulkan selama proses uji kompetensi.
- 5.6. LSP TIK Indonesia mengadakan rapat pleno untuk memberikan keputusan hasil uji kompetensi berdasarkan rekomendasi dari Asesor Kompetensi dan bukti-bukti yang telah dikumpulkan selama proses uji kompetensi.
- 5.7. LSP TIK Indonesia menerbitkan Sertifikat Kompetensi Skema Sertifikasi Programmer bagi peserta uji kompetensi yang dinyatakan **Kompeten** di semua unit kompetensi yang diujikan.
- 5.8. LSP TIK Indonesia menerbitkan Surat Keterangan telah mengikuti proses uji kompetensi bagi peserta uji kompetensi yang dinyatakan **Belum Kompeten**.

## 6. Rincian Unit Kompetensi

No	Kode Unit	Judul Unit
1	J.620100.009.02	Menggunakan Spesifikasi Program
2	J.620100.016.01	Menulis Kode Dengan Prinsip Sesuai Guidelines dan Best Practices
3	J.620100.017.02	Mengimplementasikan Pemrograman Terstruktur
4	J.620100.018.02	Mengimplementasikan Pemrograman Berorientasi Objek
5	J.620100.019.02	Menggunakan Library atau Komponen Pre-Existing
6	J.620100.021.02	Menerapkan Akses Basis Data
7	J.620100.023.02	Membuat Dokumen Kode Program
8	J.620100.025.02	Melakukan Debugging
9	J.620100.033.02	Melaksanakan Pengujian Unit Program

**Kode Unit : J.620100.009.02**

**Judul Unit : Menggunakan Spesifikasi Program**

**Deskripsi Unit :** Unit kompetensi ini berhubungan dengan sikap, pengetahuan, dan keterampilan yang dibutuhkan untuk dapat memahami spesifikasi hasil perancangan program (termasuk Context Diagram (DCD), Data Flow Diagram (DFD), Entity Relationship Diagram (ERD), diagram objek, diagram komponen, class/modul program, properti class, tabel dan deskripsinya) dan menggunakannya.

Elemen Kompetensi	Kriteria Unjuk Kerja
1. Menggunakan metode pengembangan program	1.1. Metode pengembangan aplikasi ( <i>software development</i> ) didefinisikan. 1.2. Metode pengembangan aplikasi ( <i>software development</i> ) dipilih sesuai kebutuhan.
2. Menggunakan diagram program dan deskripsi program	2.1. Diagram program dengan metodologi pengembangan sistem didefinisikan. 2.2. 2.2 Metode pemodelan, diagram objek dan diagram komponen digunakan pada implementasi program sesuai dengan spesifikasi.
3. Menerapkan hasil pemodelan ke dalam pengembangan program	3.1. Hasil pemodelan yang mendukung kemampuan metodologi dipilih sesuai spesifikasi. 3.2. Hasil pemrograman ( <i>Integrated Development Environment-IDE</i> ) yang mendukung kemampuan metodologi bahasa pemrograman dipilih sesuai spesifikasi.

**Kode Unit : J.620100.016.01**

**Judul Unit : Menulis Kode Dengan Prinsip Sesuai *Guidelines* dan *Best Practices***

**Deskripsi Unit :** Unit ini menentukan kompetensi, pengetahuan dan Sikap kerja yang diperlukan dalam menerapkan prinsip penulisan kode yang baik agar kode tersebut dapat dirawat (*maintainability*).

Elemen Kompetensi	Kriteria Unjuk Kerja
1. Menerapkan <i>coding-guidelines</i> dan <i>best practices</i> dalam penulisan program (kode sumber)	1.1. Kode sumber dituliskan mengikuti <i>coding-guidelines</i> dan <i>best practices</i> . 1.2. Struktur program yang sesuai dengan konsep paradigmanya dibuat. 1.3. Galat/ <i>error</i> ditangani.
2. Menggunakan ukuran performansi dalam menuliskan kode sumber.	2.1. Efisiensi penggunaan <i>resources</i> oleh kode dihitung. 2.2. Kemudahan interaksi selalu diimplementasikan sesuai standar yang berlaku.

**Kode Unit : J.620100.017.02**

**Judul Unit : Mengimplementasikan Pemrograman Terstruktur**

**Deskripsi Unit :** Unit kompetensi ini berhubungan dengan sikap, pengetahuan, dan keterampilan yang dibutuhkan untuk membuat program terstruktur atau prosedural.

Elemen Kompetensi	Kriteria Unjuk Kerja
1. Menggunakan tipe data dan <i>control program</i>	1.1. Tipe data yang sesuai standar ditentukan. 1.2. <i>Syntax program</i> yang dikuasai digunakan sesuai standar. 1.3. Struktur kontrol program yang dikuasai digunakan sesuai standar
2. Membuat program sederhana	2.1. Program baca tulis untuk memasukkan data dari <i>keyboard</i> dan menampilkan ke layar monitor termasuk variasinya sesuai standar masukan/keluaran telah dibuat. 2.2. Struktur kontrol percabangan dan pengulangan dalam membuat program telah digunakan.
3. Membuat program menggunakan prosedur dan fungsi	3.1. Program dengan menggunakan prosedur dibuat sesuai aturan penulisan program. 3.2. Program dengan menggunakan fungsi dibuat sesuai aturan penulisan program. 3.3. Program dengan menggunakan prosedur dan fungsi secara bersamaan dibuat sesuai aturan penulisan program. 3.4. Keterangan untuk setiap prosedur dan fungsi telah diberikan.
4. Membuat program menggunakan <i>array</i>	4.1. Dimensi <i>array</i> telah ditentukan. 4.2. Tipe data <i>array</i> telah ditentukan. 4.3. Panjang <i>array</i> telah ditentukan. 4.4. Pengurutan <i>array</i> telah digunakan
5. Membuat program untuk akses <i>file</i>	5.1. Program untuk menulis data dalam media penyimpan telah dibuat. 5.2. Program untuk membaca data dari media penyimpan telah dibuat.
6. Mengkompilasi Program	6.1. Kesalahan program telah dikoreksi. 6.2. Kesalahan <i>syntax</i> dalam program telah dibebaskan.

**Kode Unit : J.620100.018.02**

**Judul Unit : Mengimplementasikan Pemrograman Berorientasi Objek**

**Deskripsi Unit :** Unit kompetensi ini berhubungan dengan sikap, pengetahuan, dan keterampilan yang dibutuhkan dalam membuat perangkat lunak aplikasi dalam bahasa pemrograman berorientasi objek.

Elemen Kompetensi	Kriteria Unjuk Kerja
1. Membuat program berorientasi objek dengan memanfaatkan <i>class</i>	1.1. Program dengan menggunakan <i>class</i> dibuat. 1.2. Properti <i>class</i> yang akan direalisasikan dalam bentuk prosedur/fungsi dibuat. 1.3. Data didalam <i>class</i> dibuat mandiri.

	1.4. Hak akses dari tipe data ( <i>private</i> , <i>protected</i> , <i>public</i> ) dikelola.
2. Menggunakan tipe data dan <i>control program</i> pada metode atau operasi dari suatu kelas	2.1. Tipe data diidentifikasi. 2.2. Sintaks program dikuasai sesuai dengan bahasa pemrogramannya. 2.3. <i>Control program</i> dikuasai.
3. Membuat program dengan konsep berbasis objek	3.1. <i>Inheritance</i> pada <i>class</i> diterapkan. 3.2. <i>Polymorphism</i> pada <i>class</i> diterapkan. 3.3. <i>Overloading</i> pada <i>class</i> diterapkan.
4. Membuat program <i>object oriented</i> dengan <i>interface</i> dan paket	4.1. <i>Interface class</i> program dibuat. 4.2. Paket dengan program dibuat.
5. Mengkompilasi Program	5.1. Kesalahan dapat dikoreksi. 5.2. Program bebas salah sintaks dihasilkan

**Kode Unit** : J.620100.019.02

**Judul Unit** : Menggunakan *Library* atau Komponen *Pre-Existing*

**Deskripsi Unit** : Unit kompetensi ini berhubungan dengan sikap, pengetahuan, dan keterampilan yang diperlukan untuk menggunakan komponen-komponen *reuse* (yang dapat dipergunakan secara berulang) untuk mendukung pengembangan aplikasi yang efisien.

Elemen Kompetensi	Kriteria Unjuk Kerja
1. Melakukan pemilihan unit-unit <i>reuse</i> yang potensial	1.1. <i>Class</i> unit-unit <i>reuse</i> (dari aplikasi lain) yang sesuai dapat diidentifikasi. 1.2. Keuntungan efisiensi dari pemanfaatan komponen <i>reuse</i> dapat dihitung. 1.3. Lisensi, Hak cipta dan hak paten tidak dilanggar dalam pemanfaatan komponen <i>reuse</i> tersebut.
2. Melakukan integrasi <i>library</i> atau komponen <i>pre-existing</i> dengan <i>source code</i> yang ada	2.1. Ketergantungan antar unit diidentifikasi. 2.2. Penggunaan komponen yang sudah <i>obsolete</i> dihindari. 2.3. Program yang dihubungkan dengan <i>library</i> diterapkan.
3. Melakukan pembaharuan <i>library</i> atau komponen <i>preexisting</i> yang digunakan	3.1. Cara-cara pembaharuan <i>library</i> atau komponen <i>pre-existing</i> diidentifikasi. 3.2. Pembaharuan <i>library</i> atau komponen <i>preexisting</i> berhasil dilakukan



**Kode Unit : J.620100.021.02**

**Judul Unit : Menerapkan Akses Basis Data**

**Deskripsi Unit :** Unit kompetensi ini berhubungan dengan sikap, pengetahuan, dan keterampilan yang dibutuhkan untuk membuat program yang mengakses basis data suatu sistem manajemen basis data.

Elemen Kompetensi	Kriteria Unjuk Kerja
1. Membuat berbagai operasi terhadap basis data	1.1. Data dapat disimpan/diubah ke dalam format basis data. 1.2. Informasi yang diinginkan dapat dihasilkan menggunakan <i>query</i> tersebut. 1.3. Indeks dipergunakan untuk mempercepat akses.
2. Membuat prosedur akses terhadap basis data	2.1. <i>Library</i> akses basis data dapat diterapkan. 2.2. Perintah akses data yang relevan dengan teknologi atau jenis baru data, diterapkan untuk mengakses data.
3. Membuat koneksi basis data	3.1. Teknologi koneksi yang sesuai dipilih. 3.2. Keamanan koneksi ditentukan. 3.3. Hak setiap pengguna ditentukan.
4. Menguji program basis data	4.1. Skenario pengujian disiapkan. 4.2. Logika pemrograman mengacu pada kinerja <i>statement</i> akses data yang akan dibaca. 4.3. Performansi mengacu pada kinerja <i>statement</i> akses data yang akan dibaca data diuji.

**Kode Unit : J.620100.023.02**

**Judul Unit : Membuat Dokumen Kode Program**

**Deskripsi Unit :** Unit kompetensi ini berhubungan dengan sikap, pengetahuan, dan keterampilan yang diperlukan untuk membuat dokumentasi dari kode program yang telah ditulis secara *hardcopy* termasuk identifikasi penjas dari dokumen tersebut.

Elemen Kompetensi	Kriteria Unjuk Kerja
1. Melakukan identifikasi kode program	1.1. Modul program diidentifikasi 1.2. Parameter yang dipergunakan diidentifikasi 1.3. Algoritma dijelaskan cara kerjanya 1.4. Komentar setiap baris kode termasuk data, eksepsi, fungsi, prosedur dan <i>class</i> (bila ada) diberikan
2. Membuat dokumentasi modul program	2.1. Dokumentasi modul dibuat sesuai dengan identitas untuk memudahkan pelacakan 2.2. Identifikasi dokumentasi diterapkan 2.3. Kegunaan modul dijelaskan 2.4. Dokumen direvisi sesuai perubahan kode program
3. Membuat dokumentasi fungsi, prosedur atau method program	3.1. Dokumentasi fungsi, prosedur atau metod dibuat 3.2. Kemungkinan eksepsi dijelaskan 3.3. Dokumen direvisi sesuai perubahan kode program
4. Men- <i>generate</i> dokumentasi	4.1. <i>Tools</i> untuk <i>generate</i> dokumentasi diidentifikasi 4.2. <i>Generate</i> dokumentasi dilakukan

**Kode Unit** : J.620100.025.02

**Judul Unit** : Melakukan *Debugging*

**Deskripsi Unit** : Unit kompetensi ini berhubungan dengan sikap, pengetahuan, dan keterampilan yang dibutuhkan dalam memeriksa kode program dari kesalahan (*bug*).

Elemen Kompetensi	Kriteria Unjuk Kerja
1. Mempersiapkan kode program	1.1. Kode program sesuai spesifikasi disiapkan. 1.2. <i>Debugging tools</i> untuk melihat proses suatu modul dipersiapkan
2. Melakukan <i>debugging</i>	2.1. Kode program dikompilasi sesuai bahasa pemrograman yang digunakan. 2.2. Kriteria lulus <i>build</i> dianalisis. 2.3. Kriteria eksekusi aplikasi dianalisis. 2.4. Kode kesalahan dicatat.
3. Memperbaiki program	3.1. Perbaikan terhadap kesalahan kompilasi maupun <i>build</i> dirumuskan. 3.2. Perbaikan dilakukan.

**Kode Unit** : J.620100.033.02

**Judul Unit** : Melaksanakan Pengujian Unit Program

**Deskripsi Unit** : Unit kompetensi ini berhubungan dengan sikap, pengetahuan, dan keterampilan yang dibutuhkan dalam melakukan uji coba unit program.

Elemen Kompetensi	Kriteria Unjuk Kerja
1. Menentukan kebutuhan uji coba dalam pengembangan	1.1. Prosedur uji coba aplikasi diidentifikasi sesuai dengan <i>software development life cycle</i> . 1.2. <i>Tools</i> uji coba ditentukan. 1.3. Standar dan kondisi uji coba diidentifikasi.
2. Mempersiapkan dokumentasi uji coba	2.1. Kebutuhan untuk uji coba ditentukan. 2.2. Uji coba dengan variasi kondisi dapat dilaksanakan. 2.3. Skenario uji coba dibuat.
3. Mempersiapkan data uji	3.1. Data uji unit tes diidentifikasi. 3.2. Data uji unit tes dibangkitkan.
4. Melaksanakan prosedur uji coba	4.1. Skenario uji coba didesain. 4.2. Prosedur uji coba dalam algoritma didesain. 4.3. Uji coba dilaksanakan.
5. Mengevaluasi hasil uji coba	5.1. Hasil uji coba dicatat. 5.2. Hasil uji coba dianalisis. 5.3. Prosedur uji coba dilaporkan. 5.4. Kesalahan/ <i>error</i> diselesaikan.