
Full Stack Enterprise Application Developer

Materi Pembelajaran

Dr. Bambang Purnomosidi D. P.



Daftar Isi

1	Tentang Buku Ini	3
2	Minggu 01	5
	Hari 1	5
	Tujuan	5
	Pembahasan	5
	Pembelajaran	5
	Hari 2	7
	Tujuan	7
	Pembahasan	7
	Pembelajaran	7
	Hari 3	8
	Tujuan	8
	Pembahasan	9
	Pembelajaran	9
	Hari 4	9
	Tujuan	9
	Pembahasan	9
	Pembelajaran	9
	Hari 5	10
	Tujuan	10
	Pembahasan	10
	Pembelajaran	10
3	Minggu 02	11
	Hari 1	11
	Tujuan	11
	Pembahasan	11
	Pembelajaran	11
	Hari 2	11
	Tujuan	11

Pembahasan	11
Pembelajaran	11
Hari 3	12
Tujuan	12
Pembahasan	12
Pembelajaran	12
Hari 4	12
Tujuan	12
Pembahasan	13
Pembelajaran	13
Hari 5	13
Tujuan	13
Pembahasan	13
Pembelajaran	13
4 Minggu 03	14
Hari 1	14
Tujuan	14
Pembahasan	14
Pembelajaran	14
Hari 2	14
Tujuan	14
Pembahasan	14
Pembelajaran	14
Hari 3	15
Tujuan	15
Pembahasan	15
Pembelajaran	15
Hari 4	15
Tujuan	15
Pembahasan	16
Pembelajaran	16
Hari 5	16
Tujuan	16
Pembahasan	16
Pembelajaran	16

5 Minggu 04	17
Hari 1	17
Tujuan	17
Pembahasan	17
Pembelajaran	17
Hari 2	17
Tujuan	17
Pembahasan	18
Pembelajaran	18
Hari 3	18
Tujuan	18
Pembahasan	18
Pembelajaran	18
Hari 4	18
Tujuan	18
Pembahasan	19
Pembelajaran	19

1 Tentang Buku Ini



Versi

1 1.0.0-RC--28-Mei-2019--21:58:44

Buku ini merupakan buku pegangan program “**Full Stack Enterprise Application Developer**” untuk para mentor, siswa, serta pihak-pihak lain yang berkepentingan dengan proses pendidikan di Praxis Academy. Dengan panduan buku ini, diharapkan proses belajar menjadi lebih efektif dan efisien serta mempunyai pedoman yang jelas. Lisensi buku ini adalah Creative Commons Attributiona ShareAlike 4.0 International License - CC-BY-SA 4.0. Secara umum, penggunaan lisensi ini mempunyai implikasi bahwa pengguna materi:

1. Harus memberikan atribusi ke penulis dan sponsor untuk penulisan materi ini (Praxis Academy).
2. Boleh menggunakan produk yang ada disini untuk keperluan apapun jika point 1 di atas terpenuhi.
3. Boleh membuat produk derivatif dari produk yang ada disini sepanjang perubahan-perubahan yang dilakukan diberitahukan ke kami dan di-share dengan menggunakan lisensi yang sama.

Untuk penggunaan selain ketentuan tersebut, silahkan menghubungi:

1 Praxis Academy
2 Jl.Garuda no 67, Manukan
3 Condong Catur
4 Sleman
5 Yogyakarta 55283
6 Indonesia
7 Email: hello@praxisacademy.id

8 Web: <https://praxisacademy.id>

2 Minggu 01

Hari 1

Tujuan

1. Memahami teknologi informasi secara umum dan kaitannya dengan software.
2. Memahami ranah pendidikan yang terkait dengan teknologi informasi: istilah teknik informatika / *informatics*, ilmu komputer / *computer science*.
3. Memahami software paling mendasar: *operating system*
4. Memahami ekosistem sistem operasi saat ini
5. Memahami komponen-komponen sistem operasi - khususnya berbasis Linux
6. Memahami UI di Linux (*GUI* dan *text mode / shell*)
7. Memahami dan bisa menggunakan berbagai utilitas dasar dalam Linux
8. Memahami dasar-dasar dari *shell script*.

Pembahasan

1. Teknologi informasi dan teknologi software
2. Sistem Operasi: UI, shell, utilities, shell script

Pembelajaran

1 Materi dan Penjelasan

1. Pelajari dan pahami halaman Wikipedia - Information Technology, secara khusus kuasai komponen-komponen dari TI dengan penekanan lebih pada software.
2. Pelajari bagaimana software menempati komputer kita serta bagaimana user berinteraksi dengan *tasks* komputasi. Tekankan pemahaman pada bagaimana sistem operasi bekerja.
3. Pelajari overview dari daftar sistem operasi yang ada di dunia ini.
4. Pelajari dan pahami komponen-komponen SO Linux.
5. Pelajari Windowing System yang berfungsi sebagai antarmuka grafis (GUI) di Linux serta implementasinya di Linux menggunakan X Window System.

6. Pelajari juga keterkaitan X Window System dengan Display Manager, Window Manager, serta Desktop Environment.
7. Pelajari Linux console dan keterkaitannya dengan Linux shell.
8. Pelajari ringkasan dari berbagai command shell terutama yang digunakan di Linux.
9. Ada beberapa utilitas yang biasanya digunakan pada saat berada di *console / shell*. Beberapa utilitas tersebut biasanya berada dalam kelompok GNU Core Utilities dan util-linux. Pelajari dan pahami beberapa perintah yang biasanya digunakan. Silahkan coba di Guru99 dan E-Guide - format PDF.
10. Pelajari dan pahami cara membuat shell script.

1 Latihan

1. Boot laptop masing-masing, perhatikan proses booting dari awal, jika muncul Grub, usahakan melihat parameter dari Grub tersebut. Cari informasi di Internet tentang parameter Grub tersebut.
2. Cari informasi tentang software yang ada di laptop anda: display manager yang digunakan, window manager yang digunakan, desktop environment yang digunakan, serta shell apa yang digunakan. Dari mana bisa mengetahui informasi tersebut?
3. Cari lokasi dari kernel Linux, sebutkan file-file yang terkait dan kegunaannya.
4. Masuk ke *terminal / console* dan kerjakan beberapa perintah berikut melalui *command line / shell*:
 - buat direktori `$HOME/praxis/minggu-01/hari-01`
 - silahkan coba beberapa perintah di Guru99 dan E-Guide - PDF.
 - tulis hasil dari masing-masing perintah tersebut ke dalam file `cmdline.txt` (gunakan copy paste dari shell)
5. Silahkan coba 30 contoh shell script, masukkan semua file-file yang dihasilkan di direktori `$HOME/praxis/minggu-01/hari-01/30shellscript`.

1 Kasus

Referensi: 1. Wikibooks - Bash Shell Scripting 2. Bash Reference Manual 3. Bash Handbook

Selesaikan kasus-kasus berikut.

0. Semua hasil disimpan pada `$HOME/praxis/minggu-01/hari-01/kasus`
1. Buat shell script untuk melihat daftar file pada suatu direktori (termasuk direktori anak-anaknya) dan jika terdapat file dengan ekstensi `.java` - tampilkan tulisan “Ada file Java pada direktori {nama direktori}”. Hasil eksekusi (misalnya):


```
1 $ cari.sh $HOME/src
2 Ada file Java pada direktori /home/bdpd/src/hari-01
```

2. Buat shell script untuk menanyakan suatu nama program (misalnya `firefox`), setelah itu mencari PID dari program tersebut dan jika PID program tersebut ada, maka program tersebut akan dimatikan. Saran: gunakan perintah-perintah `ps`, `grep`, `awk`, dan `kill`.

Hari 2

Tujuan

1. Siswa memahami peran Git dalam software engineering serta memahami keterkaitan Git dengan istilah-istilah yang biasanya digunakan:
 - software engineering
 - software configuration management
 - version control system
 - distributed version control system
2. Siswa memahami berbagai vendor yang menyediakan fasilitas untuk *remote repository*.
3. Siswa memahami dasar-dasar penggunaan Git serta GitHub dan mampu menggunakan Git serta GitHub untuk *single person development*.
4. Siswa memahami dan mampu membuat file *markdown* untuk dokumentasi di GitHub.
5. Siswa memahami dan mampu menggunakan Git serta GitHub untuk *team* pengembang aplikasi.

Pembahasan

1. Software engineering, software configuration management, version control. dan distributed version control.
2. Git dan perintah-perintah dasarnya
3. *Markdown* sebagai format untuk dokumentasi
4. Git dan *remote repository* (GitHub, GitLab, Assembla, BitBucket)
5. Git untuk single person development
6. Git untuk tim pengembang aplikasi

Pembelajaran

1 Materi dan Penjelasan

1. Memahami ruang lingkup software engineering.
2. Memahami keterkaitan software engineering dengan SCM - Software Configuration Management.
3. Memahami keterkaitan SCM dengan Version Control dan Distributed Version Control.
4. Memahami Git dan keterkaitannya dengan Distributed Version Control
5. Memahami dan mampu men-*setup* Git - Chapter 1 - Getting Started.
6. Membuat account GitHub serta membuat repo di GitHub.
7. Memahami README.md serta mampu menggunakan pemformatan file markdown untuk menuliskan dokumentasi.
8. Menggunakan `git status`, `git add`, `git commit`, `git push` untuk menyimpan ke *remote repo*
9. Memahami dan menggunakan branching and merging.
10. Membuat pull request untuk repo di GitHub milik sendiri, merging, kemudian sinkronisasi ke lokal repo di komputer.
11. Memahami dan mampu menggunakan Github untuk kolaborasi tim.

1 Latihan

1. Buat repo sesuai keterangan pada panduan umum Praxis Academy untuk hari 1, masukkan semua file-file yang dibuat pada pertemuan hari pertama.
2. Praktikkan Getting Started ini bersama rekan.

1 Kasus

1. Praktikkan Team Collaboration with GitHub bersama rekan.

Hari 3

Tujuan

1. Development tools dan ekosistemnya
2. Dasar-dasar Java

Pembahasan**Pembelajaran**

1 Materi dan Penjelasan

1 Latihan

1 Kasus

Hari 4**Tujuan**

1. Package di Java<
2. Dynamic typing dan static typing
3. Variabel, konstanta, operator, ekspresi, statement, dan tipe data di Java
4. Penanganan error / exception

Pembahasan**Pembelajaran**

1 Materi dan Penjelasan

1 Latihan

1 Kasus

Hari 5

Tujuan

1. Dasar-dasar Pemrograman Berorientasi Obyek di Java<
2. Alur kendali program di Java
3. Algoritma, pseudocode, dan implementasi algoritma ke Java

Pembahasan

Pembelajaran

1 Materi dan Penjelasan

1 Latihan

1 Kasus

3 Minggu 02

Hari 1

Tujuan

1. Annotations
2. Object Oriented Programming Lanjut di Java

Pembahasan

Pembelajaran

1 Materi dan Penjelasan

1 Latihan

1 Kasus

Hari 2

Tujuan

1. Generics
2. Functional Programming di Java

Pembahasan

Pembelajaran

1 Materi dan Penjelasan

1 Latihan

1 Kasus

Hari 3

Tujuan

1. Unit Testing di Java

Pembahasan

Pembelajaran

1 Materi dan Penjelasan

1 Latihan

1 Kasus

Hari 4

Tujuan

1. Serialisasi data: XML, JSON, dll
2. Struktur data di Java

Pembahasan**Pembelajaran**

1 Materi dan Penjelasan

1 Latihan

1 Kasus

Hari 5**Tujuan**

1. Concurrent Programming di Java

Pembahasan**Pembelajaran**

1 Materi dan Penjelasan

1 Latihan

1 Kasus

4 Minggu 03

Hari 1

Tujuan

1. Pustaka standar dan pustaka pihak ketiga di Java
2. Proses *build* dan pengelolaan paket pustaka di Java: Gradle

Pembahasan

Pembelajaran

1 Materi dan Penjelasan

1 Latihan

1 Kasus

Hari 2

Tujuan

1. Database management system
2. Akses ke database menggunakan Java: JDBC dan ORM

Pembahasan

Pembelajaran

1 Materi dan Penjelasan

1 Latihan

1 Kasus

Hari 3

Tujuan

1. NoSQL dan NewSQL

Pembahasan

Pembelajaran

1 Materi dan Penjelasan

1 Latihan

1 Kasus

Hari 4

Tujuan

1. RESTful endpoint
2. GraphQL

Pembahasan**Pembelajaran**

1 Materi dan Penjelasan

1 Latihan

1 Kasus

Hari 5**Tujuan**

1. Arsitektur Software: microservices dan *full stack applicatio*
2. Framework: Dasar-dasar Micronaut

Pembahasan**Pembelajaran**

1 Materi dan Penjelasan

1 Latihan

1 Kasus

5 Minggu 04

Hari 1

Tujuan

1. Berpindah development tools: step-by-step
2. Studi kasus perpindahan development tools: dari Java ke peranti pengembangan lainnya (JavaScript, Kotlin, Go, PHP)
3. Dasar-dasar JavaScript dan Node.js

Pembahasan

Pembelajaran

1 Materi dan Penjelasan

1 Latihan

1 Kasus

Hari 2

Tujuan

1. JavaScript

Pembahasan**Pembelajaran**

1 Materi dan Penjelasan

1 Latihan

1 Kasus

Hari 3**Tujuan**

1. Vue.js

Pembahasan**Pembelajaran**

1 Materi dan Penjelasan

1 Latihan

1 Kasus

Hari 4**Tujuan**

1. Aplikasi Full stack: Micronaut + Vue.js
2. Aplikasi Full stack: Micronaut + Vue.js + Database

Pembahasan

Pembelajaran

1 Materi dan Penjelasan

1 Latihan

1 Kasus