



# PENGUJIAN PERANGKAT LUNAK

## -ACCEPTANCE TEST-

Eka Widhi Yunarso (EWD)  
2017-2

*Digunakan dilingkungan internal prodi  
D3 Manajemen Informatika, Fakultas  
Ilmu Terapan, Universitas Telkom*

# RINGKASAN

Dalam Bab *Acceptance Test* ini, menjelaskan bahwa Terdapat Proses *Testing* yang merupakan bagian tahapan model Proses Pengembangan V- System. Dalam Proses Pengujian perlu adanya Perencanaan Pengujian (*Test Planning*) dengan dokumentasi-dokumentasi yang dibutuhkan. Hal ini juga perlu dengan Strategi Pengujian yang mendasari tahapan Pengujian P/L ini. Dalam ***Acceptance Test*** ini ditujukan agar **Verifikasi** dan **validasi** terhadap **P/L** apakah **diterima atau tidak sebelum direlease** kepada pengguna. Perlu kiranya adanya dokumen template untuk Accptance Test Plan untuk kebutuhan Pengujian.

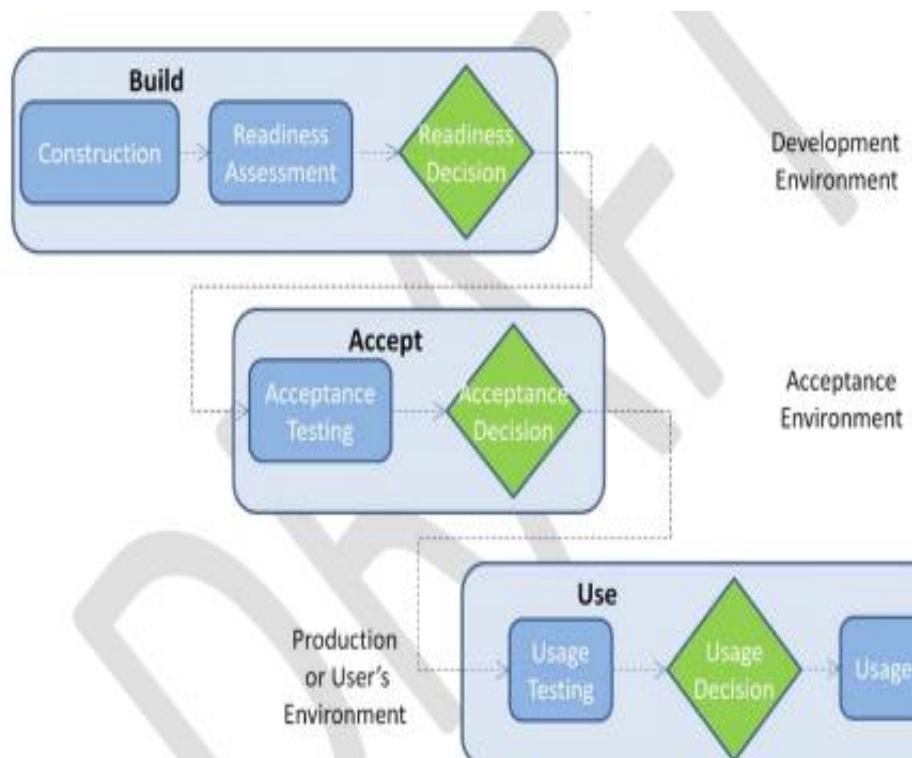


# ISTILAH ACCEPTANCE TEST

1. **Acceptance Decision**, Keputusan terhadap penerimaan terhadap perangkat lunak yang telah dibangun atau dikembangkan berdasarkan Acceptance Criteria
2. **Acceptance Testing**, Proses Pengujian dengan melibatkan sekumpulan data untuk seseorang membuat Acceptance Decision sebagai keputusan terhadap penerimaan perangkat lunak
3. **Readiness Decision**, Keputusan terhadap suatu produk perangkat lunak, apakah siap untuk dibaca, dilihat atau digunakan orang yang terlibat dalam Acceptance Decision atau Acceptance Testing



# ACCEPTANCE PROCESS



Proses diawali dengan Construction dari Produk P/L, yang memungkinkan terdapat banyak bug, maka perlu di uji sebelum diterima oleh konsumen/pengguna

Untuk meminimalisir jumlah konsumen menanyakan produk ini untuk diterima, maka suplier, yang akan me-release produk, maka melakukan self-assesment dengan aktifitas verifikasi P/L terhadap kebutuhan, hal ini disebut readiness Assesment

Keputusan melibatkan readiness Assesment pada produk P/L tsb disebut readiness Decision

Pengujian dilakukan untuk diterima sebagai kandidat release pengguna, hal ini melakukan Acceptance Testing

Keputusan untuk menerima produk P/L. Kondisi ini disebut Acceptance Decision

Keputusan tiap pengguna membuat produk diterima atau tidak secara aktual digunakan. Hal ini disebut Usage Decision

Tiap keputusan bisa menghasilkan keluaran negatif atau positif



# LATIHAN

No	Studi Kasus
1.	Suppose you are writing a simple calculator program. This program can handle positive integer calculation, including addition, subtraction, multiplication, and division. The input is a string composed of digits (0, 1, 2, ..., 9) and operators (+, -, *, /). No space is allowed. The input string can be at most 100 characters long, and each number can compose of at most 10 digits. Division of two integers produces one integer by truncation. If the answer contains more than 10 digits, this program simply outputs an overflow error message. Using the <b>equivalence partitioning and boundary value analysis methods</b> , derive a set of <b>acceptance test cases</b> for the program.



# REFERENSI

- Boriz Beizer, Control Flow Testing.  
Cognizant Technology Solutions, Software Testing, 2010.
- Desikan, Srinivasan, dan Gopalaswamy Ramesh. Software Testing: Principles and Practices. Dorling Kindersley, 2006.
- Hoyle, David. ISO9000 Handbook 6th Ed. Elsevier Science and Technology, 2009.
- IEEE, "IEEE Standard 610.12-1990, IEEE Standard Glossary of Software Engineering Terminology," 1990.
- Koirala, S, dan S Sheikh. Software Testing: Interview Questions. Infinity Science Press, 2008.
- Limaye, MG. Software Testing: Principles, Techniques and Practice. Tata McGraw-Hill, 2009.
- Malik, Kamma. Software Quality, A Practitioner's Approach . New Delhi: Tata McGraw-Hill, 2008.
- Patton, Ron. Software Testing (2nd Edition). Sams, 2005.
- Pressman, Roger R. Software Engineering, A Practitioner's Approach. Singapore: McGraw-Hill, 2005.
- Sharon Robson, White Box Testing, STANZ, 2009.
- Telles, Matt, dan Yuan Hsieh. The Science Of Software Debugging. Dreamtech Press, 2004.
- Yunarso, Eka Widhi. Student Workbook Jaminan Mutu Sistem Informasi. DeePublish, 2013.