



Program Studi Teknik Informatika
FAKULTAS ILMU KOMPUTER
UNIVERSITAS DIAN NUSWANTORO

STLC - Apa itu V Model dalam Software Testing ?

FAKULTAS ILMU KOMPUTER

PROGRAM STUDI
Teknik Informatika

MATA KULIAH
Software Quality & Testing

Istilah

- **SDLC**: adalah Siklus Hidup Pengembangan Perangkat Lunak. Tahapan kegiatan dilakukan secara berurutan, yang dilakukan oleh Pengembang untuk merancang dan mengembangkan perangkat lunak berkualitas tinggi.
- **STLC**: adalah Siklus Hidup Pengujian Perangkat Lunak. Terdiri dari serangkaian aktivitas yang dilakukan oleh Penguji secara metodologis untuk menguji produk perangkat lunak.
- **Waterfall Model**: Model air terjun adalah model sekuensial yang dibagi menjadi beberapa tahapan kegiatan pengembangan perangkat lunak. Setiap tahap dirancang untuk melakukan aktivitas tertentu. Tahap pengujian dalam model waterfall dimulai hanya setelah implementasi sistem selesai.

FAKULTAS ILMU KOMPUTER

Tahap SDLC	Aktivitas yang dilakukan pada setiap tahap
Requirement Gathering stage	Mengumpulkan sebanyak mungkin informasi tentang detail dan spesifikasi software dari klien.
Design Stage	Merencanakan Bahasa pemrograman, spt Oracle, MySQL, dll. Yang cocok untuk proyek tersebut, juga beberapa fungsi dan arsitektur tingkat tinggi.
Build Stage	Setelah tahap desain, berikutnya adalah tahapan pembangunan, yang tidak lain adalah coding software
Test Stage	Kemudian, menguji perangkat lunak untuk memverifikasi bahwa apa yang dibuat telah sesuai spesifikasi yang diberikan oleh client
Deployment Stage	Terapkan aplikasi di lingkungan masing-masing
Maintenance Stage	Setelah system kita sudah siap digunakan, kamu mungkin nantinya perlu untuk merubah code sesuai dengan permintaan pelanggan.

FAKULTAS ILMU KOMPUTER

PROGRAM STUDI
Teknik Informatika

MATA KULIAH
Software Quality & Testing

V Model ?

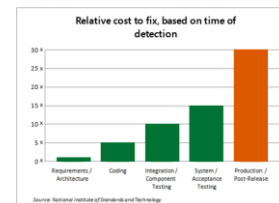
- Model V adalah model Software Development Life Cycle (SDLC) yang sangat ketat/ disiplin, dimana di setiap fase pengembangan terdapat fase pengujian yang paralel.
- Model V merupakan perpanjangan dari model waterfall dimana pengujian dilakukan pada setiap tahapan yang paralel dengan pengembangan secara sekuensial.
- Model V dikenal sebagai **Model Validasi** atau **Verifikasi**.

FAKULTAS ILMU KOMPUTER

Permasalahan menggunakan Model Waterfall

- Seperti yang kita amati, bahwa pengujian pada model waterfall dimulai setelah tahapan implementasi selesai. Tetapi bila kita bekerja dalam proyek besar, dengan sistem yang kompleks, hal tersebut dapat dengan mudah melewati detail utama dalam tahap requirement itu sendiri.
- Contoh kasus.
- Produk yang jelas-jelas salah akan dikirim ke klien, dan mungkin Anda harus memulai proyek itu lagi ATAU jika Anda berhasil mencatat kebutuhan dengan benar tetapi membuat kesalahan serius dalam perancangan dan arsitektur dari software tersebut, Anda harus mendesain ulang seluruh perangkat lunak untuk memperbaiki kesalahan tersebut.
- Berdasarkan penilaian dari ribuan proyek, bahwa cacat yang ditimbulkan selama fase requirement dan desain mencapai hampir setengah dari jumlah total cacat.
- Penilaian ribuan proyek telah menunjukkan bahwa **cacat yang ditimbulkan selama persyaratan & desain mencapai hampir setengah dari jumlah total cacat**.

FAKULTAS ILMU KOMPUTER

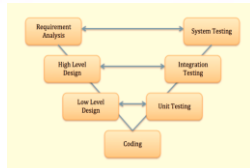


Selain itu, biaya perbaikan cacat meningkat sepanjang siklus hidup pengembangan. Semakin awal cacat/ kesalahan pada siklus hidup dideteksi, biayanya semakin murah untuk memperbaikinya. Seperti yang mereka katakan,

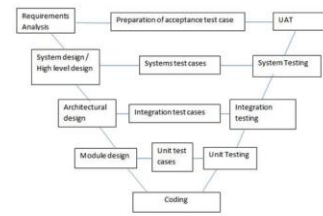
Solusi : V- Model

Pengujian Model V telah dikembangkan dimana untuk setiap fase, dalam siklus hidup pengembangan ada fase pengujian yang saling berhubungan/ sesuai.

Selain model V, ada model pengembangan berulang, di mana pengembangan dilakukan secara bertahap, dengan setiap tahap menambahkan fungsionalitas ke perangkat lunak. Setiap fase terdiri dari serangkaian aktivitas pengembangan dan pengujian yang independen. Contoh dari siklus hidup Pengembangan mengikuti metode iteratif adalah Rapid Application Development (RAD), Agile Development



- Sisi sebelah kiri dari model adalah **SDLC**
- Sisi sebelah kanan dari model adalah **STLC**
- Gambar keseluruhan seperti V, dinamakan **V-Model**



STLC - V Model - © www.SoftwareTestinghelp.com

Kelebihan dari V-Model

- Sederhana dan mudah digunakan.
- Aktivitas pengujian seperti perencanaan, perancangan pengujian terjadi dengan baik sebelum coding. Hal ini menghemat banyak waktu. Maka, peluang keberhasilannya lebih tinggi dibandingkan dengan model waterfall.
- Pelacakan cacat/ defect proaktif – yaitu cacat ditemukan pada tahap awal.
- Untuk proyek kecil yang kebutuhannya mudah dipahami, model ini berfungsi dengan baik.
- Menghindari aliran ke bawah dari cacat.

Kelemahan dari V-Model

- Sangat kaku dan paling tidak fleksibel.
- Perangkat lunak dikembangkan selama tahap implementasi, jadi tidak ada prototipe awal perangkat lunak yang dihasilkan.
- Jika ada perubahan yang terjadi di tengah jalan, maka dokumen tes beserta dokumen kebutuhan (requirement document) harus diperbarui.

Kapan menggunakan V-model ?

- V-model sebaiknya digunakan untuk proyek berukuran kecil hingga menengah dimana kebutuhan telah didefinisikan dengan jelas dan telah ditetapkan.
- V-model seharusnya dipilih jika tersedia sumber daya teknis yang cukup dengan keahlian teknis yang dibutuhkan.
- Kepercayaan yang tinggi dari pelanggan dibutuhkan untuk memilih pendekatan model bentuk V ini, karena tidak ada prototipe yang diproduksi, ada resiko sangat tinggi yang terlibat dalam memenuhi harapan pelanggan.
- Teknologi dan tools yang digunakan tidak dinamis

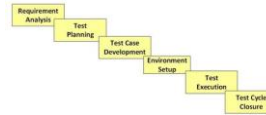
Pros and Cons of using V-model

Pros	Cons
Perkembangan dan kemajuan sangat terorganisir dan sistematis.	Tidak cocok untuk proyek yang lebih besar dan kompleks
Bekerja baik untuk project ukuran kecil sampai menengah.	Tidak sesuai jika requirement tidak konsisten
Pengujian dimulai dari awal sehingga ambiguitas dikenali/ diidentifikasi sejak awal.	Tidak ada perangkat lunak yang berfungsi yang diproduksi pada tahap perantara.
Mudah dikelola karena setiap fase memiliki tujuan dan sasaran yang jelas.	Tidak ada ketentuan untuk melakukan analisis risiko sehingga ada ketidakpastian dan risiko.

Definisi STLC

- **Software Testing Life Cycle (STLC)** is a sequence of specific activities conducted during the testing process to ensure software quality goals are met. STLC involves both verification and validation activities. Contrary to popular belief, Software Testing is not just a single/isolate activity, i.e. testing. It consists of a series of activities carried out methodologically to help certify your software product.
- **Software Testing Life Cycle (STLC)** adalah serangkaian aktifitas spesifik yang dilakukan selama proses pengujian untuk memastikan sasaran kualitas perangkat lunak terpenuhi. STLC melibatkan kegiatan verifikasi dan validasi. Pengujian perangkat lunak bukanlah aktivitas tunggal. Pengujian terdiri dari serangkaian kegiatan yang dilakukan secara metodologis untuk membantu menjamin produk perangkat lunak anda.

Tahapan STLC



1. Analisis Kebutuhan
2. Perencanaan Pengujian
3. Pengembangan Test Case (Uji Kasus)
4. Pengaturan Lingkungan Pengujian
5. Eksekusi Pengujian
6. Penutupan Siklus Pengujian

Setiap tahapan memiliki kriteria entry dan exit yang pasti, aktivitas, dan hasil kerja yang saling berhubungan

Kriteria Entry dan Exit pada STLC

- Kriteria Entry: memberikan item prasyarat yang harus dilengkapi sebelum pengujian dapat dimulai.
- Kriteria Keluar: mendefinisikan item yang harus dilengkapi sebelum pengujian dapat diselesaikan.
- Pada setiap tahapan STLC memiliki kriteria entry dan exit semua.
- Idealnya, tahapan berikutnya belum bisa dilakukan sebelum kriteria exit tahapan sebelumnya terpenuhi. Namun pada praktiknya, hal ini kadang tidak memungkinkan.

Tahapan STLC (1) – Requirement Analysis

- Tahapan ini sering disebut dengan Requirement Analysis
- Tim penguji mempelajari requirement dari sudut pandang pengujian untuk mengidentifikasi persyaratan yang dapat diuji dan tim QA dapat berinteraksi dengan berbagai pemangku kepentingan untuk memahami kebutuhan secara rinci.
- Requirement dapat berupa fungsional atau non-fungsional.
- Kelayakan otomatisasi untuk proyek pengujian juga dilakukan pada tahap ini.

Tahapan STLC (1) – Requirement Analysis

Aktivitas dalam Pengujian Fase Persyaratan :

- Mengidentifikasi jenis tes yang akan dilakukan.
- Mengumpulkan detail tentang prioritas dan fokus pengujian.
- Menyiapkan Requirement Traceability Matrix (RTM)/ Matriks Keterlacakan Persyaratan (menyimpan catatan persyaratan dan untuk memastikan pemenuhan persyaratan).
- Mengidentifikasi detail lingkungan pengujian tempat pengujian seharusnya dilakukan.
- Menganalisis kelayakan otomatisasi (jika diperlukan).

Hasil Pengujian Fase Requirement berupa :

- RTM
- Laporan kelayakan otomatisasi. (jika ada)

Tahap STLC	Kriteria Entry/Masukan	Aktivitas/Kegiatan	Kriteria Keluar	Kiriman/Hasil
Analisis Kebutuhan	– Dokumen Persyaratan tersedia (baik fungsional maupun non fungsional) – Kriteria penerimaan didefinisikan. – Dokumen arsitektur aplikasi tersedia.	– Menganalisis fungsionalitas bisnis untuk mengetahui modul bisnis dan fungsi spesifik modul. – Identifikasi semua transaksi dalam modul. – Identifikasi semua profil pengguna. – Kumpulkan antarmuka / otentikasi pengguna, persyaratan penyebaran geografis. – Identifikasi jenis tes yang akan dilakukan. – Kumpulkan detail tentang prioritas dan fokus pengujian. – Mempersiapkan Requirements Traceability Matrix (RTM). – Identifikasi detail lingkungan pengujian tempat pengujian seharusnya dilakukan. – Analisis kelayakan otomatisasi (jika diperlukan).	– Menandatangani RTM – Laporan kelayakan otomatisasi uji ditandatangani oleh klien	– RTM – Laporan kelayakan otomatisasi (jika ada)

Tahapan STLC (2) - Perencanaan Pengujian

- Perencanaan Tes di STLC adalah fase di mana manajer Senior QA menentukan strategi rencana pengujian berikut upaya dan perkiraan anggaran biaya untuk proyek tersebut.
- Perencanaan Pengujian juga dilakukan untuk menentukan sumber daya, lingkungan pengujian, batasan pengujian, dan jadwal pengujian.
- Rencana Tes dipersiapkan dan diselesaikan dalam fase yang sama.

Tahapan STLC (2) - Perencanaan Pengujian

Aktivitas Perencanaan Pengujian :

- Penyusunan dokumen rencana pengujian / strategi untuk berbagai jenis pengujian
- Pemilihan alat uji
- Uji estimasi upaya
- Perencanaan sumber daya dan menentukan peran dan tanggung jawab.
- Kebutuhan pelatihan

Hasil Perencanaan Tes :

- Dokumen rencana pengujian / dokumen strategi.
- Dokumen estimasi upaya.

Tahap STLC	Kriteria Entri/Masukan	Aktivitas/Kegiatan	Kriteria Keluar	Kiriman/Hasil
Perencanaan Tes	<ul style="list-style-type: none"> – Dokumen Persyaratan – Matriks Ketertelusuran Persyaratan. – Dokument Uji kelayakan otomatisasi. 	<ul style="list-style-type: none"> – Menganalisis berbagai pendekatan pengujian yang tersedia. – Finalisasi pada pendekatan yang paling cocok. – Persiapan rencana uji / dokumen strategi untuk berbagai jenis pengujian. – Pemilihan alat uji. – Perkiraan upaya pengujian. – Perencanaan sumber daya dan menentukan peran dan tanggung jawab 	<ul style="list-style-type: none"> – Dokumen rencana uji / strategi yang disetujui. – Dokumen estimasi upaya ditandatangani. 	<ul style="list-style-type: none"> – Rencana uji / dokumen strategi. – Dokumen estimasi upaya.

Tahapan STLC (3) - Tahap Pengembangan Kasus Uji

- Fase ini melibatkan pembuatan, verifikasi, dan pengerjaan ulang kasus uji & skrip uji setelah rencana uji siap.
- Pertama, **data Test diidentifikasi, dibuat dan direview**, lalu **dikerjakan ulang** berdasarkan kebutuhan.
- Kemudian tim QA memulai proses **pengembangan kasus uji** untuk masing-masing unit.

Kegiatan Pengembangan Kasus Uji :

- Buat kasus uji, skrip otomatisasi (jika ada)
- kasus uji direview , sebagai baseline dan skrips/ naskah pengujian
- Buat data pengujian (Jika Test Environment tersedia)

Hasil Pengembangan Kasus Uji :

- Uji kasus / skrip
- Data Uji

Tahap STLC	Kriteria Entri/Masukan	Aktivitas/Kegiatan	Kriteria Keluar	Kiriman/Hasil
Pengembangan uji kasus	<ul style="list-style-type: none"> – Dokumen Persyaratan – RTM dan rencana uji – Laporan analisis otomatisasi 	<ul style="list-style-type: none"> – Buat kotak uji, desain uji, skrip otomatisasi (jika ada) – Review dan uji kasus dasar dan skrip awal – Buat data uji 	<ul style="list-style-type: none"> – Ditinjau dan ditandatangani pada Kasus / skrip uji – Meninjau dan menandatangani data uji 	<ul style="list-style-type: none"> – Uji kasus / skrip – Uji data

Tahapan STLC (4) - Tahap Pengujian Setup Lingkungan

- Tahap ini menentukan kondisi perangkat lunak dan perangkat keras yang digunakan untuk menguji produk kerja.
- Tahapan ini adalah salah satu aspek penting dari proses pengujian dan dapat dilakukan secara paralel dengan Tahap Pengembangan Kasus Uji.
- Tim penguji mungkin tidak terlibat dalam aktivitas ini jika tim pengembangan menyediakan lingkungan pengujian. Tim penguji diharuskan untuk melakukan pemeriksaan kesiapan (smoke testing) dari lingkungan yang diberikan.

Tahapan STLC (4) - Tahap Pengujian Setup Lingkungan

Aktivitas Pengujian Pengaturan Lingkungan :

- Memahami arsitektur yang diperlukan, pengaturan lingkungan, dan menyiapkan daftar kebutuhan perangkat keras dan perangkat lunak untuk Lingkungan Pengujian.
- Menyiapkan Lingkungan pengujian dan data pengujian
- Lakukan pengujian "smoke"

Hasil dari Pengaturan Lingkungan Pengujian :

- Kesiapan lingkungan dengan data pengujian yang siap
- Hasil Pengujian "Smoke"

Tahap STLC	Kriteria Entri/Masukan	Aktivitas/Kegiatan	Kriteria Keluar	Kiriman/Hasil
Pengaturan Lingkungan Uji	<ul style="list-style-type: none"> – Desain Sistem dan dokumen arsitektur tersedia – Rencana pengaturan lingkungan tersedia 	<ul style="list-style-type: none"> – Memahami arsitektur yang diperlukan, pengaturan lingkungan – Menyiapkan daftar persyaratan pengembangan perangkat keras dan perangkat lunak – Finalisasi persyaratan konektivitas – Siapkan daftar periksa pengaturan lingkungan – Tes pengaturan Lingkungan dan data uji – Lakukan smoke testing pada saat build. – Terima / tolak build tergantung pada hasil smoke testing 	<ul style="list-style-type: none"> – Pengaturan lingkungan berfungsi sesuai rencana dan daftar periksa – Pengaturan data uji selesai – smoke testing berhasil 	<ul style="list-style-type: none"> – Lingkungan siap dengan pengaturan data uji – Hasil smoke testing.

Tahapan STLC (5) - Tahap Eksekusi Pengujian

- Tahap ini dilakukan oleh penguji di mana pengujian perangkat lunak dilakukan berdasar pada rencana uji dan kasus uji yang disiapkan.
- Proses pengujian terdiri dari :
 - eksekusi naskah uji,
 - pemeliharaan naskah uji, dan
 - pelaporan bug.
 Jika bug dilaporkan maka akan dikembalikan ke tim pengembangan untuk diperbaiki dan pengujian ulang akan dilakukan.

Tahapan STLC (5) - Tahap Eksekusi Pengujian

Aktivitas Eksekusi Pengujian :

- Jalankan tes sesuai rencana
- Dokumentasikan hasil pengujian, dan kerusakan log untuk kasus yang gagal
- Petakan cacat untuk kasus pengujian di RTM
- Uji ulang perbaikan Cacat
- Lacak cacat untuk penutup

Hasil Eksekusi Uji :

- Melengkapi RTM dengan status eksekusi
- Kasus uji diperbarui dengan hasil
- Laporan cacat

Tahap STLC	Kriteria Entri/Masukan	Aktivitas/Kegiatan	Kriteria Keluar	Kiriman/Hasil
Eksekusi Testing	<ul style="list-style-type: none"> – RTM Baseline, Rencana Tes , Test case / script tersedia – Lingkungan uji siap – Pengaturan data uji telah dilakukan – Laporan uji Unit / Integrasi untuk di build yang akan diuji tersedia 	<ul style="list-style-type: none"> – Jalankan tes sesuai rencana – Dokumentasikan hasil tes, dan log cacat untuk kasus yang gagal – Perbarui rencana pengujian / uji kasus, jika perlu – Mapping Error untuk menguji kasus dalam RTM – Tes ulang perbaikan cacat – Pengujian Regresi aplikasi – Lacak cacat yang akan ditutup 	<ul style="list-style-type: none"> – Semua tes yang direncanakan dijalankan – Cacat dicatat dan dilacak hingga ditutup 	<ul style="list-style-type: none"> – RTM lengkap dengan status eksekusi – Kasus uji diperbarui dengan hasil – Laporan cacat

Tahap STLC (6) - Tahap Penutupan siklus Pengujian

- Tahap ini adalah penyelesaian pelaksanaan uji yang melibatkan beberapa aktivitas seperti
 - pelaporan penyelesaian uji,
 - pengumpulan matriks penyelesaian uji, dan hasil uji.
- Anggota tim pengujian bertemu, berdiskusi, dan menganalisis artefak pengujian untuk mengidentifikasi strategi yang harus diterapkan di masa depan, mengambil pelajaran dari siklus pengujian saat ini.
- Idennya adalah untuk menghilangkan kemacetan proses untuk siklus pengujian di masa mendatang.

Tahap STLC (6) - Tahap Penutupan siklus Pengujian

Aktivitas Penutupan Siklus Pengujian :

- Evaluasi kriteria penyelesaian siklus berdasarkan :
 - Waktu
 - Cakupan Tes
 - Biaya,
 - Perangkat Lunak,
 - Tujuan Bisnis Penting,
 - Kualitas
- Siapkan metrik pengujian berdasarkan parameter di atas.
- Dokumentasikan pembelajaran dari proyek
- Siapkan laporan uji penutupan
- Pelaporan kualitatif dan kuantitatif kualitas produk pekerjaan kepada pelanggan.
- Analisis hasil pengujian untuk mengetahui distribusi kerusakan menurut jenis dan tingkat keparahan.

Hasil Penutupan Siklus Uji :

- Laporan Penutupan Uji
- Metrik pengujian

Tahap STLC	Kriteria Entri/Masukan	Aktivitas/Kegiatan	Kriteria Keluar	Kiriman/Hasil
Penutupan siklus uji	<ul style="list-style-type: none"> Pengujian telah selesai Hasil tes tersedia Log yang cacat tersedia 	<ul style="list-style-type: none"> Mengevaluasi kriteria penyelesaian siklus berdasarkan Waktu, Cakupan pengujian, Biaya, Kualitas Perangkat Lunak, Tujuan Bisnis Kritis Siapkan metrik pengujian berdasarkan parameter di atas. Dokumentasikan pembelajaran dari proyek Mempersiapkan laporan penutupan Tes Pelaporan kualitatif dan kuantitatif kualitas produk kerja kepada pelanggan. Uji hasil analisis untuk mengetahui distribusi cacat berdasarkan jenis dan tingkat keparahan 	<ul style="list-style-type: none"> Laporan Uji Penutupan ditandatangani oleh 	<ul style="list-style-type: none"> Laporan Uji Penutupan Metrik uji

Kesimpulan

- Ada banyak model siklus hidup pengembangan. **Model pengembangan yang dipilih untuk suatu proyek bergantung pada maksud dan tujuan proyek itu.**
- Pengujian bukanlah aktivitas yang berdiri sendiri, dan harus menyesuaikan model pengembangan yang dipilih untuk proyek tersebut.
- Dalam model apa pun, pengujian harus dilakukan di semua tingkatan, yaitu mulai dari requirement sampai maintenance/ pemeliharaan.
- Menggunakan V-model untuk proyek berukuran kecil hingga menengah, dan sumber daya teknikal cukup memadai.

PR (untuk bahan kuis minggu depan)

- Carilah beberapa model pengembangan Perangkat lunak dan jelaskan masing-masing model tersebut! (Proses, Kekurangan dan kelebihan)
- Carilah beberapa model pengujian perangkat lunak dan jelaskan masing-masing model tersebut! (Proses, Kekurangan dan kelebihan)
- Lihat tutorial pada aktivitas siklus STLC berikut untuk pengayaan.
- <https://www.youtube.com/watch?list=UU19i1XD6k88KqHfET8atqFQ&v=Dq5IYYqnnGQ&feature=emblogo>

Terimakasih