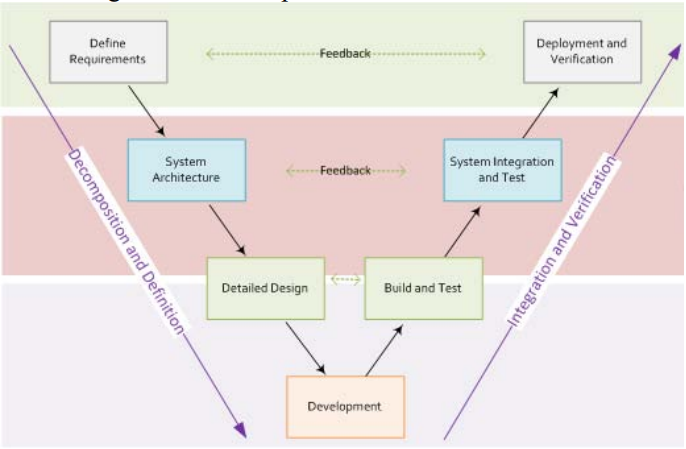
**Metode SDLC V-Model**

Yuda Fatah Kurniawan1

**1. Definisi V-Model**

V-Model adalah variasi dari model *Waterfall* dalam bentuk V dilipat menjadi setengah pada tingkat dekomposisi terendah, seperti yang ditunjukan pada Image 1.1. Bagian kiri pada bentuk V merepresentasikan evolusi kebutuhan pengguna menjadi komponen yang lebih kecil melalui proses dekomposisi dan definisi; bagian kanan mewakili integrasi dan verifikasi komponen sistem ke dalam tingkat implementasi dan perakitan secara berurutan. Sumbu vertikal menggambarkan tingkat dekomposisi dari level sistem, di bagian atas, hingga level terendah detail pada level komponen di bagian bawah. Dengan demikian, semakin kompleks suatu sistem maka semakin dalam bentuk V dengan jumlah tahapan yang lebih besar. V-model, menjadi tiga dimensi, juga memiliki sumbu yang normal terhadap bidang, yaitu sumbu z. Ini mewakili komponen yang terkait dengan beberapa pengiriman. Waktu karenanya diwakili oleh dua sumbu: dari kiri ke kanan dan ke dalam bidang [1].

**2. Gambar V-Model**



**Image 1.1** V-Model

**3. Kelebihan dan Kekurang V-Model**

Kekuatan kunci dari v-model adalah model ini dapat digunakan dalam proyek yang sangat besar di mana sejumlah kontraktor, subkontraktor dan tim yang berbeda terlibat; dekomposisi, integrasi dan verifikasi pada setiap tahap dengan semua pihak yang terlibat dimungkinkan oleh v-model sebelum maju ke tahap berikutnya. Faktor ini diakui oleh parameter waktu yang diwakili oleh sumbu y dan z dari model. Dengan demikian, sejumlah besar pihak dan pemangku kepentingan dalam proyek yang sangat besar menjadi terintegrasi secara inheren dengan menggunakan pendekatan yang digerakkan oleh persyaratan [1].

Kekurangan dari v-model yaitu kurang jika digunakan pada perangkat lunak yang menyediakan layanan kepada pengguna akhir atau ke aplikasi pengguna akhir dan Perangkat lunak yang menyediakan antarmuka visual untuk pengguna akhir.

**4. Kondisi Digunakannya Metode**

(Nayan [1]) mempertimbangkan model untuk tiga kasus penggunaan yang berbeda, atau kategori perangkat lunak, untuk menyediakan konteks untuk aplikasi mereka, sebagai berikut:

1. Perangkat lunak yang menyediakan fungsionalitas *back-end*. Biasanya, ini adalah perangkat lunak yang menyediakan layanan ke aplikasi lain.
2. Perangkat lunak yang menyediakan layanan kepada pengguna akhir atau ke aplikasi pengguna akhir. Biasanya, ini adalah perangkat lunak yang merangkum logika bisnis atau data format untuk membuatnya lebih mudah dipahami oleh pengguna akhir.
3. Perangkat lunak yang menyediakan antarmuka visual untuk pengguna akhir. Biasanya, ini adalah aplikasi *front-end* yang merupakan antarmuka pengguna grafis (*GUI*).

Seperti *waterfall*, v-model lebih cocok untuk sistem kategori 1; pencipta memodifikasi model ke vee + untuk kategori 1 dan 3, dan setelah itu menciptakan model vee ++ untuk semua kategori.

**5. Perbedaan V-Model dengan Variasi yang Lain**

Variasi dari v-model adalah model vee +. Pada model ini menambah keterlibatan pengguna, risiko dan peluang ke sumbu z dari v-model. Dengan demikian, pengguna tetap terlibat sampai dekomposisi tidak menarik bagi mereka. Untuk pengaplikasiannya setiap anomali yang diidentifikasi selama tahap integrasi dan perakitan diberikan kepada pengguna untuk penerimaan atau, dalam kasus penolakan, harus diselesaikan pada tingkat dekomposisi yang tepat; anomali kemudian dapat diselesaikan sebagai kesalahan atau diterima sebagai fitur desain. Sebaliknya, model spiral menambahkan keterlibatan pengguna selama kegiatan pengurangan risiko sedangkan v dan model *waterfall* menggabungkannya selama fase definisi persyaratan awal. Selain itu, risiko dan peluang pada fitur vee + berarti bahwa tahap tertentu dari v pada tingkat dekomposisi yang lebih rendah dapat dipotong untuk mengintegrasikan paket COTS (*commercial, off-the-shelf software*) [2]. Ini berarti bahwa berbagai tingkat produk, pada tingkat dekomposisi yang berbeda, dapat ditambahkan ke siklus hidup dengan cara yang mulus.

Variasi lain dari v-model adalah model vee ++ [2] yang menambahkan proses yang berpotongan ke model vee +; analisis dekomposisi dan proses resolusi ditambahkan ke kaki kiri bentuk v, dan analisis verifikasi dan proses dekomposisi ditambahkan ke yang benar.

# **6. Daftar Pustaka**

|  |  |
| --- | --- |
| [1] | N. B, "Software Development Lifecycle Models," *ACM SIGSOFT Software Engineering Notes,* vol. 35, pp. 8-13, 2010. |
| [2] | H. Mooz and K. Forsberg, "A visual explanation of the development methods and strategies including the waterfall, spiral, vee, vee+, and vee++ metods," pp. 4-6, 2001. |