

BAB 2

LANDASAN TEORI

2.1 Git

Git merupakan sebuah *distributed version control system* gratis dan *open source* yang dirancang untuk menangani segala sesuatu mulai dari proyek kecil hingga sangat besar dengan kecepatan dan efisiensi [5]. Git diciptakan pada tahun 2005 oleh Linus Torvalds beserta pengembang lainnya yang awalnya ditujukan untuk pengembangan *kernel* Linux. Git memiliki tiga keadaan utama pada tempat penyimpanan berkas [2], yaitu sebagai berikut:

1. *Modified*

Modified merupakan keadaan dimana berkas telah diubah tetapi belum dimasukkan ke dalam *database*.

2. *Staged*

Staged merupakan keadaan dimana berkas yang dimodifikasi telah ditandai dalam versi saat ini untuk masuk ke *commit snapshot* berikutnya.

3. *Committed*

Committed merupakan keadaan dimana data telah disimpan dengan aman pada *database* lokal.

Dari keadaan-keadaan tersebut diperoleh tiga bagian utama dari proyek Git [2], yaitu sebagai berikut:

1. *Working tree*

Working tree merupakan checkout tunggal dari satu versi proyek. Berkas-berkas ini ditarik keluar dari *database* pada *Git directory* dan ditempatkan pada *disk* untuk digunakan atau dimodifikasi.

2. *Staging area*

Staging area merupakan sebuah berkas yang menyimpan informasi tentang apa yang akan masuk ke *commit* selanjutnya.

3. *Git directory*

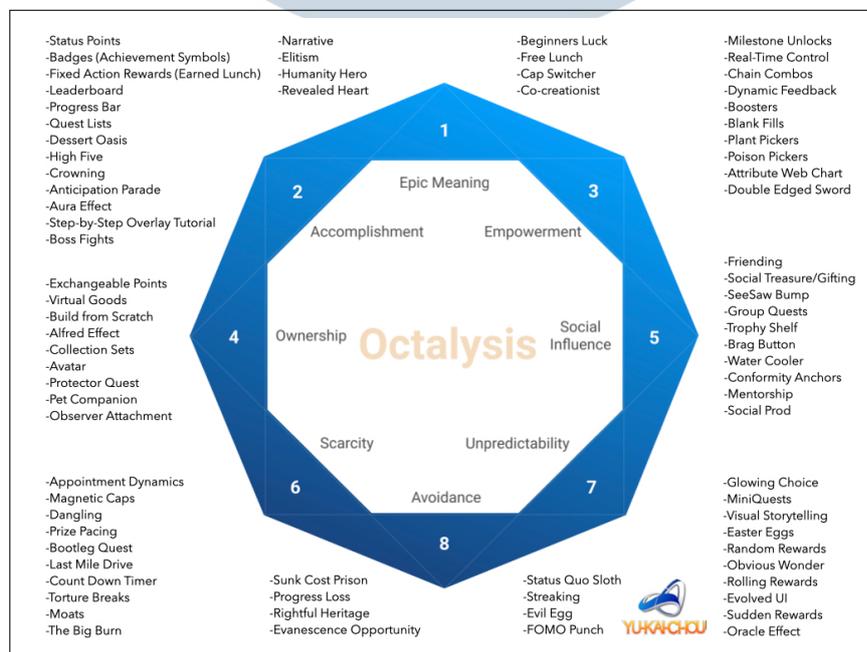
Git directory merupakan tempat Git menyimpan *metadata* dan *database* objek untuk sebuah proyek. Ini adalah bagian terpenting dari Git, dan inilah yang disalin saat pengguna mengkloning repositori dari komputer lain.

2.2 Gamifikasi

Gamifikasi adalah suatu proses untuk meningkatkan layanan dengan keterjangkauan untuk pengalaman menyenangkan yang mendukung penciptaan nilai pengguna secara keseluruhan [16]. Pada dasarnya, gamifikasi adalah desain yang paling menekankan pada motivasi manusia dalam prosesnya [17]. Gamifikasi memberikan solusi praktis dengan cara membangun ketertarikan suatu kelompok menggunakan unsur mekanik game. Gamifikasi juga dapat didefinisikan sebagai konsep yang menggunakan mekanika berbasis permainan, tindakan memotivasi, estetika dan permainan berpikir untuk mengikat seseorang, mempromosikan pembelajaran, dan menyelesaikan masalah [18].

2.3 Octalysis Framework

Octalysis merupakan sebuah *framework* atau kerangka kerja gamifikasi yang diciptakan oleh Yu-kai Chou setelah melakukan penelitian gamifikasi dan studi perilaku desain selama lebih dari 17 tahun.



Gambar 2.1. Delapan *Core Drives* dari *Octalysis framework*

Sumber: [17]

Pada Gambar 2.1 dapat dilihat bahwa terdapat delapan *core drives* dalam *Octalysis framework* yang difokuskan terhadap perilaku manusia, yaitu sebagai

berikut:

1. *Epic Meaning & Calling*

Epic Meaning & Calling merupakan dorongan dimana pengguna percaya bahwa pengguna tersebut melakukan sesuatu yang lebih besar dari dirinya sendiri atau merasa dirinya terpilih untuk melakukan sesuatu. Beberapa contoh teknik permainan yang terdapat dalam *core drive* ini yaitu *narrative*, *beginners' luck*, *free lunch*, *humanity hero*, dan *destiny child*.

2. *Development & Accomplishment*

Development & Accomplishment merupakan dorongan dari dirinya sendiri untuk mengatasi tantangan, mengembangkan keterampilan, dan membuat kemajuan. Tantangan merupakan hal yang penting, karena hadiah tanpa tantangan tidak berarti sama sekali. Beberapa contoh teknik permainan yang terdapat dalam *core drive* ini yaitu *badges*, *leaderboard*, *progress bar*, *high five*, dan *crowning*.

3. *Empowerment of Creativity & Feedback*

Empowerment of Creativity & Feedback merupakan dorongan ketika pengguna terlibat dalam proses kreatif dimana pengguna tersebut harus mencari tahu dan mencoba kombinasi yang berbeda secara berulang-ulang. Beberapa contoh teknik permainan yang terdapat dalam *core drive* ini yaitu *milestone unlocks*, *evergreen combos*, *instant feedback*, *boosters*, dan *blank fills*.

4. *Ownership & Possession*

Ownership & Possession merupakan dorongan dimana pengguna termotivasi karena pengguna tersebut merasa memiliki sesuatu. Ketika pengguna merasa memiliki, maka secara tidak langsung ingin membuat apa yang dirinya miliki menjadi lebih baik dan memiliki lebih banyak lagi. Beberapa contoh teknik permainan yang terdapat dalam *core drive* ini yaitu *exchangable points*, *virtual goods*, *alfred effect*, *avatar*, dan *protection*.

5. *Social Influence & Relatedness*

Social Influence & Relatedness merupakan dorongan yang mencakup semua elemen sosial yang memotivasi seseorang, seperti tanggapan sosial, penerimaan, bimbingan, persahabatan, serta kecemburuan dan persaingan. Ketika pengguna melihat seorang teman yang ahli dalam suatu keahlian atau

memiliki sesuatu yang luar biasa, pengguna tersebut menjadi termotivasi untuk mencapai level yang sama. Beberapa contoh teknik permainan yang terdapat dalam *core drive* ini yaitu *friending*, *group quests*, *tout flags*, *water cooler*, dan *mentorship*.

6. *Scarcity & Impatience*

Scarcity & Impatience merupakan dorongan untuk mengharapkan sesuatu yang tidak dapat dimiliki oleh pengguna. Fakta bahwa pengguna tidak dapat memperoleh sesuatu saat ini memotivasi dirinya untuk memikirkannya sepanjang hari. Beberapa contoh teknik permainan yang terdapat dalam *core drive* ini yaitu *appointment dynamics*, *magnetic caps*, *prize pacing*, *countdown timer*, dan *torture breaks*.

7. *Unpredictability & Curiosity*

Unpredictability & Curiosity merupakan dorongan untuk mencari tahu apa yang akan terjadi selanjutnya. Jika pengguna tersebut tidak tahu apa yang akan terjadi, maka otak dari pengguna tersebut akan bekerja dan sering memikirkannya. Beberapa contoh teknik permainan yang terdapat dalam *core drive* ini yaitu *glowing choice*, *easter egg*, *random rewards*, *rolling rewards*, dan *oracle effect*.

8. *Loss & Avoidance*

Loss & Avoidance merupakan dorongan yang didasarkan pada harapan untuk menghindari hal buruk. Dalam skala kecil, kemungkinan untuk menghindari kehilangan *progress* dari pekerjaan sebelumnya. Dalam skala yang lebih besar, kemungkinan untuk menghindari bahwa semua yang pengguna lakukan hingga saat ini tidak berguna karena saat ini pengguna tidak melanjutkannya kembali. Beberapa contoh teknik permainan yang terdapat dalam *core drive* ini yaitu *progress loss*, *rightful heritage*, *status quo sloth*, *visual grave*, dan *FOMO punch*.

2.4 Algoritma Fisher-Yates Shuffle

Fisher-Yates Shuffle (dinamakan berdasarkan penemunya, yaitu Ronald Fisher dan Frank Yates) merupakan sebuah algoritma untuk menghasilkan permutasi acak dari suatu himpunan terhingga, dengan kata lain untuk mengacak suatu himpunan tersebut. Algoritma Fisher-Yates Shuffle dipilih karena merupakan metode pengacakan yang lebih baik dan sesuai untuk mengacak angka dengan

waktu eksekusi yang cepat [19]. Penggunaan algoritma Fisher-Yates Shuffle modern yang diimplementasikan oleh Richard Durstenfeld dapat mengurangi kompleksitas algoritma dari $O(N^2)$ menjadi $O(N)$ [20]. Metode dasar yang diberikan untuk menghasilkan permutasi acak yaitu dari angka 1 sampai N [21]. Terdapat langkah-langkah yang digunakan untuk menghasilkan suatu permutasi acak untuk angka 1 sampai N , yaitu sebagai berikut:

1. Tuliskan angka dari 1 sampai N .
2. Pilih sebuah angka acak X antara 1 sampai dengan jumlah angka yang belum dicoret.
3. Hitung dari bawah, coret angka X yang belum dicoret, dan tuliskan angka tersebut di tempat lain.
4. Ulangi langkah 2 dan 3 hingga semua angka tercoret.
5. Urutan angka yang dituliskan pada langkah 3 adalah permutasi acak dari angka awal.

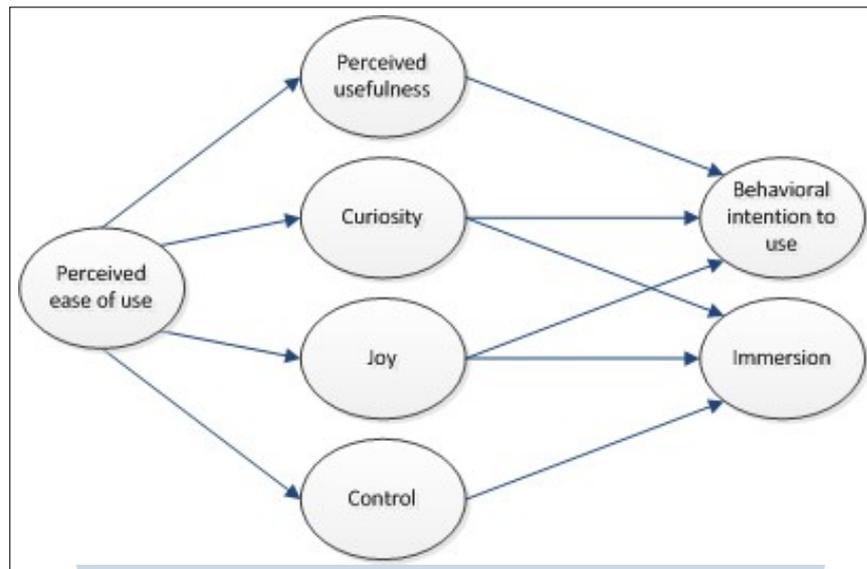
Terdapat contoh perhitungan menggunakan algoritma Fisher-Yates Shuffle yang dapat dilihat pada Tabel 2.1.

Tabel 2.1. Contoh Perhitungan Algoritma Fisher-Yates Shuffle

Range	Roll	Scratch	Result
		1 2 3 4 5 6	
1-6	4	1 2 3 6 5	4
1-5	2	1 5 3 6	2 4
1-4	3	1 5 6	3 2 4
1-3	1	6 5	1 3 2 4
1-2	1	5	6 1 3 2 4
Hasil Pengacakan			5 6 1 3 2 4

2.5 Hedonic Motivation System Adoption Model (HMSAM)

HMSAM merupakan sebuah model untuk mengukur sebuah sistem yang mengadaptasi motivasi hedonis.



Gambar 2.2. Model HMSAM

Sumber: [22]

Pada Gambar 2.2 dapat dilihat bahwa terdapat beberapa faktor yang menjadi fokus pengukuran pada HMSAM, yaitu sebagai berikut:

1. *Perceived ease of use* merupakan pengukur kemudahan penggunaan suatu sistem.
2. *Perceived usefulness* merupakan pengukur peningkatan kinerja ketika menggunakan suatu sistem.
3. *Curiosity* merupakan sejauh mana suatu sistem dapat meningkatkan rasa ingin tahu dalam aspek kognitif.
4. *Control* merupakan persepsi pengguna bahwa dirinya yang diajak berinteraksi oleh sistem.
5. *Joy* merupakan aspek kesenangan yang diperoleh dari interaksi dengan sistem.
6. *Behavioral intention to use* merupakan keinginan pengguna untuk menggunakan aplikasi.
7. *Immersion* merupakan total keterlibatan yang dilakukan antara pengguna dengan aplikasi yang mengukur seberapa dalam pengguna terfokus dalam menggunakan sistem.

Terdapat skala pengukuran pada HMSAM [23], yang dapat dilihat pada Tabel 2.2.

Tabel 2.2. Skala Pengukuran HMSAM

Gagasan	Kriteria
<i>Perceived Ease of Use</i>	<p>PEOU1. Interaksi dengan permainan ini jelas dan mudah dimengerti.</p> <p>PEOU2. Berinteraksi dengan permainan ini tidak membutuhkan banyak usaha.</p> <p>PEOU3. Permainan ini bebas dari masalah.</p> <p>PEOU4. Mudah untuk membuat permainan ini melakukan apa yang diinginkan.</p> <p>PEOU5. Mudah untuk menjadi terampil dalam menggunakan permainan ini.</p>
<i>Perceived Usefulness</i>	<p>PU1. Permainan ini mengurangi stres.</p> <p>PU2. Permainan ini membantu menghabiskan waktu dengan lebih baik.</p> <p>PU3. Permainan ini memberikan alternatif yang berguna.</p> <p>PU4. Permainan ini membantu berpikir lebih jernih.</p> <p>PU5. Permainan ini membantu untuk merasa diremajakan.</p>
<i>Curiosity</i>	<p>CUR1. Pengalaman ini membangkitkan rasa ingin tahu.</p> <p>CUR2. Pengalaman ini membuat penasaran.</p> <p>CUR3. Pengalaman ini membangkitkan imajinasi.</p>
<i>Control</i>	<p>CTL1. Memiliki banyak kendali.</p> <p>CTL2. Dapat memilih dengan bebas apa yang ingin dilihat atau dilakukan.</p> <p>*CTL3. Tidak memiliki kendali atas interaksi.</p> <p>CTL4. Diizinkan untuk mengontrol interaksi.</p>

Tabel 2.2. Skala pengukuran HMSAM (lanjutan)

Gagasan	Kriteria
<i>Joy</i>	<p>JOY1. Merasa bermain permainan ini menyenangkan.</p> <p>JOY2. Bersenang-senang menggunakan permainan ini.</p> <p>*JOY3. Menggunakan permainan ini membosankan.</p> <p>*JOY4. Permainan ini membuat kesal.</p> <p>JOY5. Pengalaman bermainnya menyenangkan.</p>
<i>Behavioral Intention to Use</i>	<p>BIU1. Berencana untuk menggunakan kembali di masa depan.</p> <p>BIU2. Berniat untuk terus menggunakannya di masa depan.</p> <p>BIU3. Berharap penggunaannya akan terus berlanjut di masa depan.</p>
<i>Immersion</i>	<p>IME1. Dapat menghiraukan sebagian besar gangguan lainnya.</p> <p>IME2. Tenggelam dalam permainan.</p> <p>*IME3. Mudah terganggu oleh perhatian lain.</p> <p>IME4. Perhatian tidak mudah teralihkan.</p>

Kriteria HMSAM yang memiliki konotasi negatif (dengan tanda bintang) diskalakan terbalik. Contohnya pada skala Likert urutan skornya dibalik, dimulai dari skor 5 untuk kategori Sangat Tidak Setuju (STS) hingga skor 1 untuk kategori Sangat Setuju (SS).

2.6 Skala Likert

Skala Likert adalah skala yang paling banyak digunakan dalam riset berupa survei dan skala psikometrik yang umum digunakan pada kuesioner [24]. Skala Likert digunakan untuk mengukur pendapat, persepsi, dan sikap seseorang atau sekelompok orang mengenai fenomena sosial [25]. Dalam skala Likert, terdapat 5 kategori beserta skor dari masing-masing kategori yang dapat dilihat pada Tabel 2.3.

Tabel 2.3. Indikator Kategori Skala Likert

Kategori	Skor
Sangat Setuju (SS)	5
Setuju (S)	4
Netral (N)	3
Tidak Setuju (TS)	2
Sangat Tidak Setuju (STS)	1

Untuk mencari persentase dari hasil skala Likert, dapat menggunakan Persamaan 2.1 untuk kriteria HMSAM yang memiliki konotasi positif (tanpa tanda bintang) dan menggunakan Persamaan 2.2 untuk kriteria HMSAM yang memiliki konotasi negatif (dengan tanda bintang).

$$Persentase = \frac{(SS * 5) + (S * 4) + (N * 3) + (TS * 2) + (STS * 1)}{5 * TotalResponden} * 100 \quad (2.1)$$

$$Persentase = \frac{(STS * 5) + (TS * 4) + (N * 3) + (S * 2) + (SS * 1)}{5 * TotalResponden} * 100 \quad (2.2)$$

Dari persentase tersebut, dapat diperoleh tingkat penilaian responden yang sudah dikategorikan secara keseluruhan berdasarkan interval persentase yang dapat dilihat pada Tabel 2.4.

Tabel 2.4. Interval Persentase terhadap Kategori Penilaian Responden

Interval Persentase	Kategori
80% - 100%	Sangat Setuju (SS)
60% - 79,99%	Setuju (S)
40% - 59,99%	Netral (N)
20% - 39,99%	Tidak Setuju (TS)
0% - 19,99%	Sangat Tidak Setuju (STS)