

PROPOSAL PROYEK II

“DOTA II PREDICTION”

UNTUK MEMENUHI TUGAS MATA KULIAH PROYEK II

Diajukan oleh :

Afif Rana Muhammad NIM 171524001 D-IV Teknik Informatika

Ahmad Aji Naufal Ali NIM 171524002 D-IV Teknik Informatika

Atika Khoirunnisa NIM 171524006 D-IV Teknik Informatika

Fadly Triansyah Rahmat NIM 171524008 D-IV Teknik Informatika

Muhamad Alvie Irvansyah NIM 171524018 D-IV Teknik Informatika

**JURUSAN TEKNIK KOMPUTER DAN INFORMATIKA**

**POLITEKNIK NEGERI BANDUNG**

**BANDUNG**

**2019**

**Daftar isi**

[**BAB I**](#_6zkb6r73g8de)2

[**PENDAHULUAN**](#_6zkb6r73g8de)2

[**Latar Belakang**](#_6w3oy1c91foe)2

[**Tujuan**](#_szeosjp6nroh)2

[**Deskripsi Permasalahan**](#_dtc9tv31fm1g)2

[**Batasan Masalah**](#_lf9hajhtjjta)3

[**BAB II**](#_6eqwetn4onyr)4

[**STUDI PUSTAKA**](#_6eqwetn4onyr)4

[**Deskripsi singkat Dota II**](#_gysq9v35az12)4

[**Gameplay Dota II**](#_vnt9ofm1zlfx)4

[**Kompetisi Dota II**](#_mibdmm65otog)5

[**Hero Dota II**](#_govpvibl43xi)6

[**Algoritma Penentu Keputusan Sistem**](#_kp6sckl6qqc7)6

[**Hasil Penelitian Sebelumnya**](#_qvve8rfnqaw1)8

[**BAB III**](#_ce7e1tzfaf9l)10

[**METODOLOGI**](#_ce7e1tzfaf9l)10

[**Dataset**](#_iwqjwbyu2ka5)10

[**Model Pengembangan Software**](#_1xf788psk7b)10

[**BAB IV**](#_igxssg5d11cm)12

[**HASIL DAN PEMBAHASAN**](#_wbyc39212rlq)13

[**Analisis Data**](#_x9rqidp6729a)13

[**Problem Domain**](#_4duo273zbnad)13

[**Teknologi**](#_ygtcrybwfyh6)14

[**Model Solusi**](#_4nhfxz3gw0e1)15

[**BAB V**](#_5htuo7fed9tz)16

[**RUBRIK**](#_5htuo7fed9tz)16

[**DAFTAR PUSTAKA**](#_cpvlrn3ti1d1)19

# **BAB I**

# **PENDAHULUAN**

## **Latar Belakang**

Perkembangan teknologi begitu cepat membuat banyak perubahan pada hidup kita, mulai dari perubahan cara berkomunikasi, belajar, maupun cara bermain. Jika 10 tahun yang lalu kita biasa melihat orang-orang bermain permainan yang mengandalkan fisik, kini tren dunia telah berubah. Masyarakat lebih sering terlihat bermain permainan berbasis komputer, khususnya permainan online. Hal ini dibuktikan dengan jumlah pengguna permainan *online* yang semakin meningkat setiap tahunnya. Salah satu permainan online yang populer adalah Dota II.

Kepopuleran Dota II tidak berhenti sampai tingkatan banyak dimainkan. Tetapi telah berubah menjadi salah satu cabang *e-sport* terpopuler di dunia. Kompetisi *e-sport* Dota II pun mempunyai beragam tingkatan, mulai dari tingkatan kota, nasional, maupun internasional. Bahkan pihak penyelenggara e-sport Dota II tingkat internasional, menyediakan hadiah lebih dari 250 miliar rupiah untuk para pemenangnya.

Setiap tahun, pemenang kompetisi Dota II selalu sulit untuk diprediksi. Hal ini dikarenakan terdapat banyak faktor yang mempengaruhi suatu pertandingan dalam permainan Dota II. Oleh sebab itu kelompok kami tertarik membuat suatu sistem yang dapat memprediksi kemenangan pertandingan pada kompetisi Dota II berdasarkan faktor yang banyak tersebut.

## **Tujuan**

Adapun tujuan dari pembuatan aplikasi ini adalah :

1. Memenuhi tugas mata kuliah Proyek 2.
2. Memprediksi hasil pertandingan pada kompetisi *The International* Dota II.
3. Menyajikan hasil prediksi pertandingan.
4. Mempelajari cara penggunaan *Logistic Regression* pada bahasa R untuk memprediksi hasil pertandingan Dota II.

## **Deskripsi Permasalahan**

Permainan Dota II bukanlah sebuah permainan yang mudah untuk dimainkan. Dikarenakan cukup banyak variabel yang harus diperhatikan untuk memenangkan sebuah pertandingan. Setiap variabel yang ada saling berkaitan satu sama lain, contohnya dalam penggunaan hero, setiap hero memiliki hero lain yang mudah untuk dilawannya dan hero lain yang sulit untuk dilawannya, selain itu terdapat pula hero lain yang cocok sebagai teman satu tim. Selain penggunaan hero, terdapat variabel lainnya yang dapat mempengaruhi kemenangan suatu tim. Oleh karena itu, untuk memprediksi suatu pertandingan Dota II tidaklah mudah.

Dan dikarenakan banyaknya variabel, maka diperlukan bantuan komputer sebagai alat bantu komputasi dalam menyelesaikan permasalahan ini. Tentunya hasil prediksi yang dihasilkan dalam proyek ini belum tentu 100% benar. Hasil prediksi yang dihasilkan tentunya harus dikaji ulang, agar hasil prediksinya mendekati 100%.

## **Batasan Masalah**

Dota II merupakan permainan yang memiliki banyak sekali pemain. Permainan ini selalu masuk peringkat 20 besar untuk peringkat permainan yang paling banyak dimainkan. Selain itu juga cukup banyak orang yang tertarik mengenai strategi atau kombinasi serangan yang ada pada permainan ini.

Dota II sudah memasuki dunia *e-sport* semenjak tahun 2011. *E-sport* merupakan salah satu cabang olahraga yang sedang ramai-ramainya dibicarakan terutama untuk kalangan remaja. Pada awalnya hanya terdapat 8 tim *e-sport* (tim profesional) untuk Dota II, dan hanya terdapat 1 turnamen resmi internasional. Namun dengan pesatnya pertumbuhan permainan dan semakin banyaknya peminat, jumlah tim *e-sport* professional menjadi tidak terhitung banyaknya. Selain itu jumlah turnamen internasional pun semakin banyak.

Karena semakin banyaknya tim profesional yang bermunculan, proyek ini tidak akan menggunakan data seluruh tim profesional yang ada, proyek ini hanya akan mengambil data dari tim yang mengikuti turnamen *The international* yang dimana turnamen ini merupakan turnamen terbesar pada permainan ini. Dan untuk rentang waktunya hanya tim yang mengikuti turnamen *The international* 2012 hingga *The international* 2018.

# **BAB II**

# **STUDI PUSTAKA**

## **Deskripsi singkat Dota II**

Dota II merupakan game *multiplayer* dimana dua tim yang terdiri dari 5 pemain berkompetisi dalam mengumpulkan *gold* dan menghancurkan struktur markas musuhnya. Dalam bermain, diperlukan strategi bergerak, kombinasi hero, serta kombinasi serangan yang baik agar suatu tim dapat meraih kemenangan.

## **Gameplay Dota II**

Dota II mengadu 2 tim berisikan masing-masing 5 pemain, tim sebelah kiri disebut ”Radiant” dan tim sebelah kanan disebut ”Dire, dimana keduanya memiliki markas utama yang disebut *Ancient.* Tim pertama yang berhasil menghancurkan *ancient* lawan adalah pemenangnya. Terdapat 3 jalan atau jalur (disebut "*lanes”*) yaitu *offlane* (*toplane* untuk *radiant* dan *bottom lane* untuk *dire*) *midlane* dan *safe lane* (*bottom lane* untuk *radiant* dan *top lane* untuk *dire*) yang menghubungkan kedua *ancient* tersebut dan pasukan (‘creeps’) yang spawn atau muncul secara berkala setiap 30 detik di jalur-jalur tersebut, Setiap *lanes* memiliki 3 menara yang akan menyerang hero atau creeps didekatnya (disebut ’tower’). Pada tower terakhir terdapat 2 barracks yaitu melee barracks dan ranged barracks. Apabila melee barracks atau ranged barracks hancur, Maka creeps musuh akan bertambah kuat (disebut ’super melee creeps atau super ranged creeps’, dan akan menjadi ’mega creep’ apabila kedua barracks dari ketiga lane telah dihancurkan), bertarung dengan ‘creeps’ lawan di sepanjang pertempuran menuju Ancient lawan.

Selain creeps yang ada di lane, untuk menambah pengalaman guna meningkatkan level serta mengumpulkan gold, sebagian pemain juga dapat ’farming’ ataupun ’creeping’ dihutan dengan membunuh *natural creeps* ataupun ancient creeps.

Untuk memperoleh keabadian (kesempatan hidup satu kali lagi saat terbunuh) dengan menggunakan aegis immortal, pemain harus bisa mengalahkan roshan, monster hutan yang berada di jalur parit yang memiliki armor yang sangat kuat.

Dijalur parit juga terdapat rune (seperti double damage, haste, bounty, invisible dan arcane), yang muncul secara berkala setiap 2 menit.

Masing-masing pemain dapat mengendalikan sebuah unit/ karakter yang disebut ‘Hero’ yang dipilih pada awal permainan. Terdapat lebih dari 100 Hero dengan type yang berbeda : Strength, Agility dan Intelligence. Setiap Hero memiliki 4 skill (beberapa memiliki lebih) aktif maupun pasif. Dengan membunuh creeps, menaikkan level, membeli item dan membuat keputusan strategi, pemain akan berusaha menjelajah map untuk menghancurkan Ancient lawannya.

## **Kompetisi Dota II**

Terdapat beberapa kompetisi besar tahunan di Dota 2. Namun pada makalah ini, kami hanya membahas kompetisi Dota 2 yang paling bergengsi di dunia yaitu *The International*.

*The International* Dota 2 adalah ajang turnamen Dota 2 bergengsi yang diselenggarakan oleh *valve corporation* setiap tahunnya. Tim-tim dari berbagai negara berkumpul dalam satu tempat untuk mendapatkan gelar juara *The International*, uang tunai, dan piala Aegis of the Champion. The International pertama kali diselenggarakan di Jerman pada bulan agustus tahun 2011. Total hadiah dari tahun ke tahun semakin bertambah. Hal ini dikarenakan adanya produk yang dikeluarkan oleh Valve, Dimana 0,25 sen dari hasil penjualan produk ini dimasukkan ke total hadiah turnamen International. Produk ini disebut compendium. Pemain Dota II dari tahun ke tahun semakin bertambah, sehingga penjualan compendium dari tahun ke tahun bertambah pula.

Selain peraturan bermain DotA secara umum, *The International* mempunyai 2 tahapan yaitu *group stage* dan *main event*. Masing-masing tahapan memiliki aturan yang berbeda pula. Berikut rinciannya.

**Group Stage**

18 Tim dibagi menjadi 2 grup, masing-masing 9 tim dalam setiap grup, dimana mereka akan bermain dengan format round-robin (setiap tim bertemu tim lainnya tepat 1 kali).

* Dalam setiap pertemuan terdapat 2 permainan (skor = 2 jika menang, 1 jika imbang, 0 jika kalah).
* Tim dengan poin terendah akan tereliminasi.
* Jika terjadi kesamaan skor, beberapa pertandingan baru akan dimulai.
* Jika setelah beberapa pertandingan tersebut skor masih sama, beberapa faktor pertimbangan akan dilihat: (1. Hasil *head-to-head* dari setiap tim, 2. Kemenangan terhadap tim yang lebih rendah skornya (semakin tinggi skor dari tim yang lebih rendah tersebut, skor semakin besar), 3. Coin toss)

**Main Event**

* 16 tim disimpan dalam dua *bracket*, upper dan lower.
* *Grand finals* dimainkan dalam format BO5 (siapapun yang berhasil mendapat 3 kemenangan tercepat akan menang).
* Upper Bracket
* Empat tim terbaik dari tiap grup ditempatkan di upper bracket.
* Tim pertama dari tiap grup memilih musuhnya, yaitu tim ranking 3 atau 4 dari grup lainnya.
* Tim kedua dari tiap grup mendapat musuh yang tidak dipilih tim pertama.
* Setiap pertandingan pada upper bracket merupakan pertandingan Best Of 3 (siapapun yang berhasil mendapat 2 kemenangan tercepat akan menang).
* Lower Bracket
* Tim dengan ranking 5 - 8 ditempatkan di lower bracket.
* Tim pertama dari tiap grup memilih musuhnya, yaitu tim ranking 7 atau 8 dari grup lainnya.
* Tim ke-enam dari tiap grup mendapat musuh yang tidak dipilih tim pertama.
* Ronde pertama akan dimainkan dengan format BO1 (siapapun yang berhasil mendapat 1 kemenangan tercepat menang), dan ronde-ronde selanjutnya dalam format BO3.

## **Hero Dota II**

Hero merupakan karakter yang dimainkan tiap player dalam setiap match dari Dota 2. Setiap player memilih 1 hero yang akan dimainkan sebelum memulai game. Hero yang sudah dipilih tim atau musuh tidak dapat lagi digunakan oleh siapapun (10 hero yang dipilih tiap player unik). Setiap hero memiliki kelebihan dan kekurangan masing-masing. Untuk meraih kemenangan, player dituntut untuk memilih suatu hero dengan tepat, yaitu sesuai dengan tim dan kuat melawan hero musuh.

## **Algoritma Penentu Keputusan Sistem**

Pada "Dota II prediction" ini, akan digunakan algoritma logistic regression. Logistic regression merupakan algoritma yang menghasilkan kemungkinan dari suatu kejadian (pada kasus ini, kemungkinan menang dari suatu tim) berdasarkan 1 atau lebih variabel. Ide dari algoritma logistic regression serupa dengan linear regression, yaitu mencari fungsi yang sesuai berdasarkan data-data yang diberikan, yaitu variabel input dan hasilnya. Untuk itu disini akan dibahas sekilas tentang linear regression untuk pemahaman dari algoritma logistic regression.

**Linear Regression**

Jika kita mendefinisikan variabel input sebagai x-axis dan hasil sebagai y-axis, maka algoritma linear regression akan mencari garis yang merepresetasikan "trend" hasil nilai y berdasarkan x nya. Garis f(x) akan semakin merepresentasikan "trend" pemetaan x ke y jika nilai



semakin kecil. Setelah mendapat garis yang ditentukan, kita dapat mendapat hasil y berdasarkan nilai x apapun dengan persamaan y = mx + c. Catat bahwa variabel input bisa lebih dari 1, untuk itu kita bisa tambahkan z-axis dan seterusnya.

**Logistic Regression**

Sama halnya dengan linear regression, kita definiskan x-axis sebagai input dan y-axis sebagai output. Namun terdapat beberapa perbedaan:

* Nilai y antara 0 dan 1, melambangkan probabilitas suatu kejadian.
* Tidak dapat menggunakan garis sebagai representasi "trend" karena ketika nilai x mendekati tak hingga, nilai y akan bernilai tak hingga juga, disini kita dapat mengasumsikan nilai y jadi 1. Namun pada kasusnya tidak begitu, karena semakin besar faktor x seharusnya lebih besar pula y nya. Jika menggunakan representasi garis, nilai x dan semua nilai pada rentang [x + 1, INF) akan diperlakukan sama, yaitu menghasilkan nilai y = 1.

Oleh karena itu kita butuh mentransformasikan nilai y menjadi antara 0 dan 1. Kita dapat mendefinisikan nilai y yang dihasilkan sebagai score, nilai score positif berarti kemenangan bagi tim, dan kalah sebaliknya. Lebih besar score maka kemungkinan menang semakin besar, nilai score ini tidak lain merupakan nilai log(odds) dimana:



Persamaan diatas dapat ditransformasikan untuk menghasilkan p berdasarkan nilai log(odds) atau score, menjadi



Akurasi kandidat garis ditentukan oleh nilai likelihood nya, yaitu persamaan berikut: (didefinisikan value(xi) = 1 jika dengan parameter xi, nilai yi = 1, dan sebaliknya). Semakin besar nilai persamaan tersebut, maka kandidat garis semakin baik.

Ada beberapa kelebihan dan kekurangan logistic, berikut penjabarannya.

**Kelebihan**

1. Simpel, hanya 1 operasi konstan (O(number of parameters)) untuk mendapatkan hasil (setelah dilakukan data training).
2. Dapat mendapat persentase kemenangan, bukan sekedar 0 atau 1 seperti *decision tree* atau *support vector machine*, tapi 0 sampai 1.

**Kekurangan**

1. Tidak seperti *neural network* atau *decision tree*, *logistic regression* tidak dapat memprediksi berdasarkan urutan kejadian.

## **Hasil Penelitian Sebelumnya**

Penelitian yang membahas bagaimana cara memprediksi kemenangan pada game DotA 2 ini sebenernya sudah dilakukan oleh beberapa peneliti sebelumnya. Di antara penelitian tersebut adalah sebagai berikut :

1. **DOTA 2 Win Prediction (2015)**

Penelitian ini dilakukan oleh Nicholas Kinkade dan Kyung yul Kevin Lim yang berasal dari University of California di San Diego. Pada penelitian ini dibahas bahwa mereka memiliki dua buah *predictor*. Yang pertama menggunakan keseluruhan *post-match data*. Dan yang kedua, menggunakan *Hero Selection Data*.

Penelitian ini juga dikembangkan dengan merujuk pada penelitian sebelumnya yang berjudul *How Does He Saw Me?* dan *To Win or Not to Win*. Selain itu, dalam mendukung pencarian data, mereka menggunakan website “DotA Buff”. Serta menjadikan salah satu aplikasi sebelumnya, “DotA Picker”, sebagai salah satu rujukan dalam mengembangkan predictor mereka.

Pada paper *How Does He Saw Me?* menjelaskan dua jenis model yang digunakan. Model pertama menggunakan *logistic regression* yang memberikan akurasi prediksi mencapai 69%. Dan model kedua menggunakan *K-nearest neighbor classification* yang dapat memberikan akurasi hingga 68%.

Pada paper “To Win or Not to Win“ juga menjelaskan dua jenis model yang dikembangkan. Model pertama menggunakan *logistic regression* yang memberikan akurasi mencapai 69%. Dan model kedua mengkombinasikan *predictor* dengan *genetic fitness metric* yang dapat memberikan akurasi hingga 74%.

Pada penelitian yang dilakukan oleh Nicholas Kinkade dan Kyung yul Kevin Lim, dataset terdiri dari : *Winning Side*, *Duration*, dan atribut tiap pemain yang berupa *XP per minute*, *Gold per minute* dan KDA (*Kill, Deaths, Assists*). Data ini dikumpulkan dari 62.00 pertandingan dalam rentang waktu 20 November sampai 22 November 2015.

1. **Predicting Winning Team and Probabilistic Ratings in “Dota 2" and “Counter-Strike Global Offensive" Video Games (2018)**

Penelitian ini dilakukan oleh Ilya Marakov, Dmitry Savostyanov, Boris Litvyakov, Dmitry I. Ignatov dari National Research University Higher School of Economics, Moscow, Russia. Penelitian ini mencoba melakukan prediksi kemenangan pada game yang bersifat team match dengan menjadikan DotA 2 dan Counter Strike : Global Offensive sebagai uji cobanya. Penelitian ini mencoba menggabungkan antara *Bayesian ranking/rating system* dan *machine learning* untuk mengoptimalkan analisis perilaku pada *game* maupun dari histori yang ada guna mendapatkan data percobaan prediksi.

Dalam penentuan prediksinya, peneliti mencoba teknik *logistic regression*, *support vector machines*, *gradient boosting*, dan *random forest*. Dan teknik terakhir berhasil mencapai akurasi 88.8%.

Untuk data yang digunakan tidak jauh berbeda dengan penelitian sebelumnya (no.1). Yaitu ada atribut individual dan team. Di antaranya Xp dan Gold yang didapat *per minute*, *logs for kills*, *purchases* dan lainnya. Semua data itu diakses melalui API.

1. **Performance of Machine Learning Algorithms in Predicting Game Outcome from Draft in Dota 2 (2016)**

Dalam penelitian yang merupakan kolaborasi dari beberapa peneliti dari sejumlah universitas terkemuka di Rusia, dicoba sebuah pengujian prediksi pada game DotA 2 yang cukup kompleks dibanding yang lainnya. Karena beberapa teknik penentuan keputusan dikombinasikan dalam penelitian ini.

Dalam paper tersebut disebutkan bahwa tujuan dari penelitian ini adalah untuk meningkatkan kemampuan AI (untuk bot), mengoptimalkan strategi permainan, memantau keseimbangan permainan, dan mengidentifikasi permainan seorang pemain.

Data untuk keperluan penelitian ini didapat dari Steam API yang mengambil data dari 11 Februari 2016 sampai 2 Maret 2016. Mode yang diambil adalah *captain mode*, *random draft*, dan *ranking all picks*. Terhitung terdapat 5.071.858 pertandingan. Data-data individu dari pemain pun ikut dikumpulkan.

Metode yang digunakan untuk menyelesaikan permasalahan di penelitian ini adalah dengan menggunakan : *naive bayes*, *logistic regression*, *factorization machines*, dan *gradient boosting of decision tree*.

# **BAB III**

# **METODOLOGI**

## **Dataset**

Pada proyek ini dataset yang akan digunakan berkaitan dengan variabel-variabel yang terdapat pada suatu pertandingan. Data pertandingan yang diambil merupakan pertandingan yang terjadi pada kompetisi Dota 2 yaitu kompetisi The International. Kompetisi The international merupakan kompetisi terbesar Dota 2 dan tim-tim yang dapat mengikuti kompetisi ini hanyalah tim-tim terbaik dunia, sehingga kemampuan masing-masing anggota tim sudah dipastikan pemain yang handal.

Untuk mengumpulkan dataset yang ada, pada proyek ini akan digunakan metode *scraping* menggunakan aplikasi yang terdapat pada pengayaan *browser* dan menggunakan API yang ada. Namun, API yang tersedia belum tentu dapat mengambil seluruh dataset yang dibutuhkan, karena terdapat beberapa API yang berbayar.

Berdasarkan API yang ada (API tidak berbayar), maka dari setiap pertandingan Dota 2, data yang dapat diambil adalah data hero yang digunakan setiap pemain, data mengenai perkembangan *exp* dan *gold* per menit dari setiap pemain, banyaknya *kill*, *assist*, *death*, *last hits*, *denies*, *gold spent*, *hero damage*, *tower damage*, *hero healing* dan *level* akhir hero dari suatu pertandingan, *winning* tim, fraksi tim, durasi pertandingan dan data lainnya.

Dari sekian banyak data yang dapat diambil dari sebuah pertandingan, hanya data yang berkaitan dengan data hero yang digunakan oleh seorang pemain, data mengenai perkembangan *exp* dan *gold* per menit dari seorang pemain, banyaknya *kill, assist*, *death*, *last hits* dari seorang pemain, fraksi tim, dan *winning* tim lah yang menjadi variabel utama dalam proyek ini.

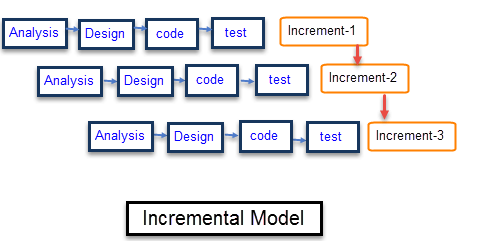
Dan berdasarkan perkiraan, data yang akan diambil dengan menggunakan API dan *scraping* untuk mendapatkan data pertandingan pada kompetisi The international, maka total data pertandingan yang diambil ada sebanyak 2216 pertandingan dan data ini yang nantinya akan diproses pada proyek ini. Dan dari 2216 data, sekitar 80%-nya merupakan data train dan sekitar 20% merupakan data test.

## **Model Pengembangan Software**

Pada proyek ini, model pengembangan *software* yang digunakan adalah Model *Incremental*. Model *Incremental* ini merupakan proses pengembangan *software* yang semua kebutuhan *software* dipecah menjadi beberapa proyek pengembangan kecil. Karena ini lah, kebutuhan utama *software* harus sudah jelas Namun, pada pengerjaannya detail-detail tambahan dapat ditambahkan seiringnya waktu. Pada model ini terdapat empat tahap utama, yaitu :

1. *Analysis.*
2. *Design.*
3. *Code.*
4. *Testing*.

Dalam menggunakan model ini, setiap proyek pengembangan akan melalui keempat tahap utama tersebut. Ini menyebabkan terjadinya beberapa siklus empat tahap utama untuk menyelesaikan semua proyek-proyek kecil tersebut. Hasil dari setiap siklus nantinya akan ditambahkan (*increment*) ke hasil siklus sebelumnya sampai semua proyek-proyek kecil tadi sudah berhasil dibuat dan sudah disatukan.

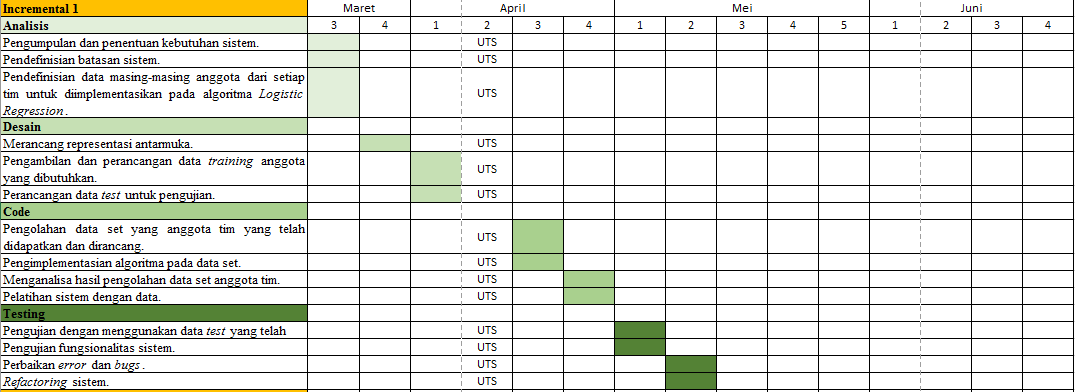


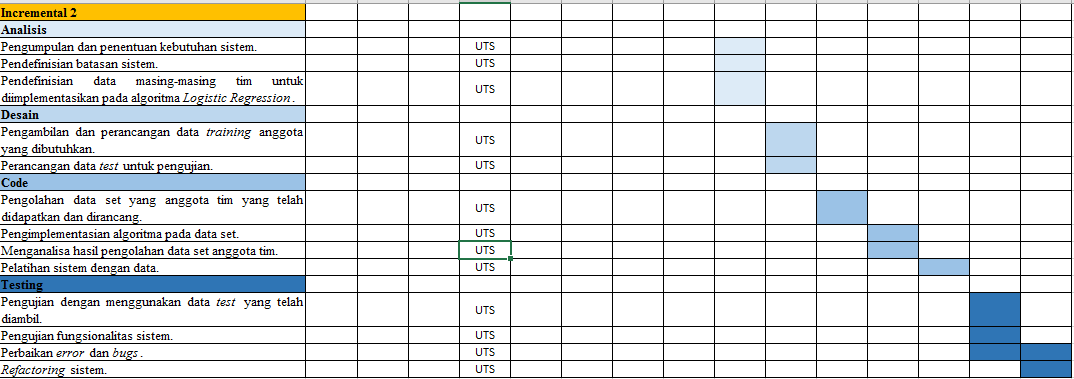
Gambar 3.1 Model Incremental

(Sumber : <https://www.guru99.com/what-is-incremental-model-in-sdlc-advantages-disadvantages.html>)

Digunakannya model *Incremental* sebagai model pengembangan *software* proyek, dikarenakan karakteristik proyek yang sesuai dengan karakteristik proyek-proyek yang menggunakan model ini. Beberapa karakteristik proyek ini yaitu telah memiliki kebutuhan sistem yang jelas, menggunakan teknologi yang baru dipelajari, tim pengerjaan proyek yang masih kurang terlatih, dan waktu pengerjaan proyek yang terbatas.

**Timeline Pengerjaan**

****



# 

# **BAB IV**

# **HASIL DAN PEMBAHASAN**

## **Analisis Data**

### **Problem Domain**

1. Perilaku

Program yang akan dibuat merupakan program yang akan memprediksi hasil setiap pertandingan di kompetisi *The International* Dota II. Program nantinya akan menghitung persentase kemenangan setiap tim di setiap pertandingannya. Tim yang memiliki persentase kemenangan lebih tinggi akan dimasukkan ke pertandingan selanjutnya dan akan dihitung kembali persentase kemenangannya melawan tim di pertandingan tersebut. Program akan terus menghitung sampai mendapatkan satu tim yang diprediksi akan menjadi pemenang turnamen. Sistem turnamen yang akan dipakai pada program akan sama seperti sistem turnamen *The International* Dota II yang asli.

Program dalam menghitung persentase kemenangan akan ada dua langkah yang dilakukan. Pertama, program akan menghitung persentase kemenangan tim berdasarkan komposisi pemain pada kedua tim yang akan bertanding. Selanjutnya, akan dihitung persentase kemenangan tim berdasarkan komposisi *Hero* yang dimainkan di kedua tim. Setelah kedua persentase didapatkan, kedua persentase tersebut akan dikalikan dengan nilai tertentu lalu dijumlahkan sehingga didapat persentase akhir.

1. Data

Untuk melatih sistem menghitung persentase berdasarkan komposisi pemain, dari setiap pemain dalam tim akan digunakan data KDA (*Kill Death Assist*), *winrate hero*, GPM (*Gold Per Minute*), XPM (*Experience Per Minute*), dan *winrate* pemain. *Winrate* pemain terdapat banyak jenisnya, namun yang akan digunakan hanyalah *winrate* premium, karena semua pertandingan pada turnamen *The International* Dota II merupakan pertandingan premium dan *winrate* premium merupakan rata-rata kemenangan pemain di setiap pertandingan premium. Untuk *winrate hero*, jika *hero* tersebut sudah pernah digunakan dalam turnamen *The International*, maka akan dipakai *winrate* pada turnamen *The International*. Namun, jika belum pernah digunakan dalam turnamen *The International*, *winrate* yang akan digunakan merupakan *winrate hero* dari seluruh pertandingan premium yang pernah dimainkan oleh pemain yang sedang menggunakan *hero* tersebut.

Untuk melatih sistem menghitung persentase berdasarkan komposisi *hero,* akan digunakan data *synergy* dan *counter*. *Synergy* adalah data yang menyatakan kecocokannya satu *hero* dengan *hero* lainnya sebagai rekan tim. *Counter* adalah data yang menyatakan pas atau tidaknya *hero* yang digunakan melawan *hero* tim lawan.

1. Metode

Hampir semua data yang dibutuhkan sudah siap untuk diolah. Hanya untuk data GPM dan XPM, akan dicari rata-rata dari keduanya dengan menambahkan nilai keduanya lalu hasil dari penjumlahan tersebut akan dibagi dua sehingga menghasilkan data BPM (*Build Per Minute*). Ini dikarenakan nilai GPM dan XPM merupakan nilai yang dapat menambah kekuatan suatu *hero* dalam permainan. Dengan GPM yang tinggi, pemain dalam membeli *item* yang dibutuhkan untuk menambah kekuatan *hero.* Sedangkan XPM, semakin tinggi XPM, maka semakin cepat level *hero* yang digunakan bertambah, dan semakin tinggi level suatu *hero,* kekuatan *hero* yang dipakai semakin kuat.

### **Teknologi**

Dalam proyek ini akan menggunakan bahasa R sebagai alat bantu komputasi. Bahasa R merupakan sebuah bahasa yang sudah terkenal untuk melakukan pengolahan data khususnya untuk *machine learning*. Pada bahasa ini sudah tersedia *library* untuk membuat *logistic regression* yang merupakan metode yang dipakai dalam proyek ini. Nama *library* untuk membuat *logistic regression* pada bahasa ini adalah glm(*generalized linear model*). Untuk membuat *logistic regression* pada *library* ini, dibutuhkan sebuah angka 0 atau 1 dan sebuah atau beberapa nilai bilangan riil. Untuk proyek ini angka 0 merupakan representasi dari tim yang kalah dalam pertandingan, dan angka 1 merupakan representasi dari tim yang memenangkan pertandingan. Dan untuk bilangin riil nantinya dapat diisi dari data yang sudah diproses sebelumnya (pre-processing). Selain itu, setiap data yang dimasukkan dapat ditentukan bobotnya, dan *library* ini mampu mengetahui data mana yang tidak berguna dan data yang berguna untuk prediksi. Dan bahasa ini juga dapat menampilkan persentase kemenangan suatu tim (persentase prediksinya) berdasarkan data *train* dan data *test* yang dimasukkan.

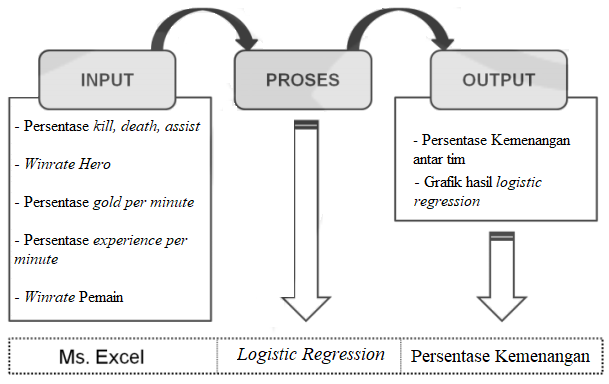
Selain itu, bahasa ini juga dapat merepresentasikan gambar grafik dari *logistic regression* dengan menggunakan *package graphics*-nya. *Package* ini dapat membuat segala macam gambar, mulai dari titik, garis, bidang datar, hingga gambar 3 dimensi. Namun dalam proyek ini, *Package* ini hanya digunakan untuk menggambarkan grafik *logistic regression* dari data *train* dan data *test* yang dimasukkan.

Pada saat melakukan *training*, data *train* akan dimasukkan sesuai dengan data *train* yang sudah diproses sebelumnya. Ketika *training* selesai maka akan dilakukan *test* dengan data *test* yang ada. Namun, tidak semua nilai variabel yang terdapat di data *test* dimasukkan sesuai dengan data *test*-nya. Karena terdapat beberapa variabel yang baru bisa didapatkan nilainya setelah pertandingan selesai. Oleh karenanya nilai dari variabel yang seperti ini diatasi dengan riwayat permainan dari setiap pemain yang bertanding. Dan setiap data *test* yang sudah diprediksi nantinya, akan dijadikan data *train* yang berisikan hasil prediksi beserta data lainnya.

### **Model Solusi**

Model data yang telah disusun bertujuan untuk mencari model grafik yang membentuk pola regresi pada *logistic regression.* Dari pola yang terbentuk akibat berbagai variabel data, maka dapat ditemukan kecenderungan hasil suatu permainan pada DotA 2.

Data diberikan solusi model akan dicari menggunakan *logistic regression* yang diimplementasikan dalam bahasa R. Pencarian solusi ini juga memanfaatkan *spreadsheet* Ms. Excel sebagai *tools* pembantu. Bahasa R akan mengimpor data dari Ms. Excel dan solusi di ekspor menjadi grafik serta nilai persentase kemenangan. Proses ini memudahkan untuk menginput data maupun melihat solusi karena kami menggunakan *scraping* dalam pengumpulan data, serta hasil *scraping* yang umum biasanya berbentuk file excel. Gambar 4.1 menggambarkan alur pencarian solusi model.



Gambar 4.1 Alur Pencarian Solusi Model

# **BAB V**

# **RUBRIK**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | **Expected Points** | **Realization Points** |
| Kemampuan untuk mendesain,  dan mengevaluasi suatu sistem software sesuai kebutuhan yang diharapkan dalam batasan yang relevan. | Desain buruk, tidak memenuhi minimal requirement fungsional. Tidak memperhatikan pertanyaan yang berhubungan dengan penggunaan memori, waktu respon, dll. | Desain memenuhi sebagian besar requirement fungsional, membawa beberapa akun dari beberapa kendala utama, desain/implementasi tampak dan tidak dibangun untuk berevolusi. | Desain yang memuaskan dan fleksibel serta memenuhi semua persyaratan fungsional;  Implementasi bug-free yang baik; | Desain dan implementasi yang luar biasa, memenuhi semua requirement fungsional,desain fleksibel bisa mengakomodasi potensial perubahan di masa depan, memperhatikan dengan cermat semua key constraints. | **3** |  |
| Kemampuan untuk mendesain dan melakukan  percobaan untuk menguji sistem software dan menafsirkan hasil untuk mendebug dan memperbaiki sistem. | Tidak ada pengujian sistematis software untuk memastikan cakupan yang wajar dari kemungkinan kasus; Tidak ada pengujian kinerja Sistem. | Pendekatan pengujian tim menyediakan cakupan esensial dasar dari kemungkinan kasus; Tim juga menguji aspek-aspek kunci dari kinerja sistem hingga batas tertentu. | Dengan teliti mendesain test case untuk mencakup berbagai situasi sesuai. Hati-hati menguji kinerja sistem dengan mempertimbangkan faktor-faktor kunci seperti lalu lintas jaringan serta respons pengguna dalam kondisi yang berbeda. | Dirancang dengan baik, test suite yang sistematis menyediakan cakupan yang sangat baik menilai kinerja sistem untuk kasus-kasus khusus dan ekstrim; Suite ini dirancang untuk menguji kinerja sistem sehubungan dengan semua faktor penting, termasuk lalu lintas jaringan, penggunaan memori, dengan hasil yang digunakan untuk menyetel bagian dari sistem. | **4** |  |
| Kemampuan untuk menggunakan teknik dan tools, termasuk sistem kontrol versi, alat komunikasi, dokumentasi standar dan praktik pengujian, dll.,  diperlukan untuk berhasil sebagai profesional CSE. | Penggunaan alat profesional penting yang tidak wajar / tidak biasa; Trivial / tidak ada kontrol versi; Pengujian ad-hoc / haphazard; Dokumentasi samar / miskin dokumentasi. | Penggunaan tools profesional yang penting secara moderat; Komit, cabang, tag, dll. Menambahkan beberapa informasi versioning yang berguna; Adopsi konvensi standar untuk dokumentasi dan testing, tetapi aplikasi tidak konsisten. | Penggunaan tools yang profesional dan standar secara efektif; Banyak fitur kontrol versi yang digunakan untuk mendukung upaya development; Integrasi yang kuat dari alat komunikasi tim; Konvensi standar dihormati untuk dokumentasi dan pengujian. | Sangat baik menggunakan tools professional dan sistem; Sepenuhnya memanfaatkan semua fitur kontrol versi, kerangka pengujian, sistem dokumentasi, dll; Memanfaatkan banyak tools di dalam toolchain pengembangan, misalnya. Integrasi berkelanjutan. | **3** |  |
| Kemampuan untuk memperhitungkan pertimbangan sosial dan etika yang relevan dalam perancangan sistem software. | Tidak ada perhatian yang diberikan pada pertimbangan sosial / etika. Tim tampaknya bahkan tidak mempertimbangkan masalah implikasi sosial dan etika yang potensial dari sistem tersebut. | Pertimbangan minimal dari implikasi sosial dan etika dari sistem yang mungkin timbul dalam kasus / situasi ekstrim. | Perhatian yang wajar diberikan pada implikasi sosial dan etika sistem dengan semua skenario use case umum untuk sistem yang dipertanggungjawabkan; Beberapa pertimbangan potensi bahaya yang mungkin disebabkan dalam kasus-kasus ekstrim. | Analisis yang sangat baik dari masalah etika general yang terkait dengan sistem dan dampaknya terhadap masyarakat serta analisis sistem sehubungan dengan persyaratan ACM / IEEE Code. | **3** |  |
| Bekerja secara efektif dalam tim pengembangan software. | Tim disfungsi; Anggota saling menyalahkan untuk masalah dalam proyek; Semangat tim benar-benar kurang | Tim berfungsi pada tingkat efektivitas minimal; Anggota berkonsentrasi pada bagian-bagian sistem yang berbeda tanpa memperhatikan dampak pada pekerjaan anggota lain. | Umumnya tim efektif; Anggota tertarik untuk menyajikan gambaran positif tentang kerja tim; Anggota tim memiliki gagasan luas tentang pekerjaan anggota lain dalam proyek. | Tim yang sangat efektif; Anggota tim tidak hanya bekerja masing-masing selama desain dan pengembangan sistem tetapi juga memberikan penghargaaan yang sesuai pada setiap anggota untuk kontribusinya di berbagai aspek proyek. | **3** |  |
| Kemampuan untuk terlibat dalam komunikasi tertulis yang efektif. | Dokumentasi terdiri dari sedikit lebih banyak (dikomentari dengan kurang baik) kode sistem. | Sebagian dokumentasi efektif dalam menyampaikan aspek teknis sistem;  Dasar pemikiran untuk pilihan desain, pendekatan testing, dll., Tidak jelas; Panduan pengguna yang minim; Informasi tim masa depan mungkin perlu mengembangkan sistem yang kurang. | Dokumentasi sistem dengan jelas mempresentasikan semua aspek penting dari proyek: detail desain dan implementasi, perincian skrip testing dll. User manual yang ditulis dengan baik. | Dokumentasi yang sangat baik dari semua aspek sistem termasuk pilihan desain dan implementasi, rincian kode yang relevan, proses dan alat yang digunakan, dan skrip uji coba, semuanya dijelaskan secara terstruktur dan terintegrasi; Informasi untuk memungkinkan desainer selanjutnya untuk mengembangkan sistem; user manual yang dirancang dengan baik; Ilustrasi, grafik, dan tata letak dieksekusi dengan efek yang sangat baik. | **3** |  |

# DAFTAR PUSTAKA

<https://en.wikipedia.org/wiki/The_International_(Dota_2)#2019>, [12 Maret 2019]

Guru99. Incremental Model in SDLC: Use, Advantage & Disadvantage. https://www.guru99.com/what-is-incremental-model-in-sdlc-advantages-disadvantages.html. [12 Maret 2019]

TRY QA. What is Incremental model- advantages, disadvantages and when to use it?.. http://tryqa.com/what-is-incremental-model-advantages-disadvantages-and-when-to-use-it/. [12 Maret 2019]