

BAB 2

LANDASAN TEORI

2.1 Game Development

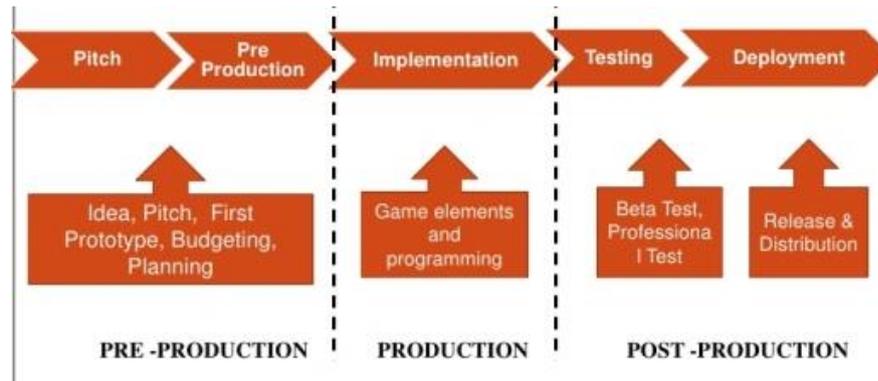
Game merupakan salah satu kegiatan yang digunakan untuk bermain, hal ini biasanya digunakan untuk hiburan atau kesenangan. Video game adalah permainan yang berbentuk elektronik yang melibatkan interaksi antara pengguna dan permainan menggunakan antarmuka dan perangkat masukan untuk menghasilkan umpan balik visual. *Game Development* adalah aktifitas membangun sebuah video game mulai dari konsep sampai dengan selesai yang dilakukan sendiri atau bersama didalam sebuah tim. *Game Development* adalah proses kooperasi yang bergantung pada motivasi, kebutuhan dan batasan yang diberikan oleh satu sama lain [2].

Pada dasarnya game diproduksi seperti memproduksi sebuah software akan tetapi terdapat aspek kreative yang lebih dalam memproduksi sebuah game. Jadi game dapat didefinisikan seperti software yang memiliki seni, musik dan permainan [6]. Dan seperti software *Game Development* adalah sebuah bisnis, game dibuat untuk dijual kepada user dimana akan menghasilkan keuntungan yang stabil jika dilakukan dengan benar.

Untuk menghasilkan sebuah game diperlukannya sebuah metodologi yang tepat untuk mencapai tujuan dengan efektif dan efisien. Oleh karena itu melalukan perencanaan sebelum melakukan *Game Development* itu penting karena tidak semua game dapat menguntungkan. Jika metode yang digunakan benar maka akan meminimalisir kegagalan dan mengurangi kerugian [6].

2.1.1 Tahap Game Development

Secara umum proses *Game Development* dibagi menjadi 3 bagian, antara lain: *Pre-Production*, *Production* dan *Post-Production*. Tahap inilah yang paling sering digunakan oleh game developer untuk menentukan jalannya sebuah *Game Development* [7].



Gambar 2.1 Tahap Game Development

2.1.1.1 Pre-Production

Pre-production atau tahap desain adalah tahapan merencanakan bagaimana jalannya sebuah proyek berdasarkan ide dan konsep yang sudah ditentukan. Tahap ini biasanya berupa riset atau mengumpulkan dan membuat dokumen awal game. Tujuan dari tahapan ini adalah untuk memproduksi arahan yang jelas dalam suatu dokumentasi yang menjelaskan pekerjaan, jadwal dan bentuk awal game. Dalam *pre-production* terdapat tahap *planning*, *concepting* dan *prototyping* dimana tahap ini bisa berbeda sesuai dengan metode yang digunakan, tetapi tiga tahap ini biasanya yang paling sering digunakan dalam *development* sebuah game.

Dalam *planning* sebuah tim developer menentukan tujuan dari proyek mereka, hal ini dapat berupa riset, *brainstorming* dan *pitching*. Tujuan dari *planning* ini untuk menghasilkan ide dan menentukan hasil dari game tersebut. Dalam melakukan *planning* selain dari ide kita perlu menentukan *audience* dan pasar yang hendak ditargetkan karena game adalah sebuah bisnis. Salah satu proses ini adalah melakukan *Market Research* dimana seorang developer atau *business analyst* melakukan proses menjelajahi pasar game untuk mendapatkan informasi yang dapat digunakan untuk menentukan game yang akan dibuat berdasarkan angka, fakta dan opini

Hasil akhir dalam *pre-production* biasanya berupa dokumen desain game (*Game design document*) ataupun sebuah *vertical-slice* atau *prototype* sebuah game yang akan digunakan.

2.1.1.2 Production

Tahap ini adalah tahap yang paling panjang dan sulit dalam sebuah *Game Development*. Tahap production adalah tahap dimana developer mulai mengerjakan game sesuai dengan rencana dan jadwal yang sudah dibuat di tahap sebelumnya.

2.1.1.3 Post-Production

Tahap ini adalah tahap akhir dalam sebuah *Game Development*. Biasanya tahap ini dilakukan setelah game sudah dikembangkan hampir sepenuhnya. Pada tahap ini developer melakukan pengujian dan melakukan perbaikan berdasarkan dari hasil pengujian tersebut. Setelah dilakukannya pengujian dan game telah dicap “selesai” maka game dapat dirilis ke pasar dan mulai dari sana para developer melakukan support dan maintenance.

2.1.2 Market Research

Market Research atau riset pasar adalah teknik untuk mengumpulkan informasi melalui dan tentang pengguna untuk mendukung rencana bisnis [3]. Hal ini digunakan bukan hanya didalam *Game Development* melainkan dibanyak hal mulai dari *software development* sampai ke pembangunan sebuah bisnis. Dalam penelitian ini akan membahas *Market Research* didalam konteks *Game Development* dan berusaha memahaminya.

Dalam *research market* game ada beberapa aspek penting yang harus diperhatikan, bukan hanya dari jumlah penjualan dan jumlah pengguna saja, diantaranya yaitu:

1. Geografi

Game didistribusikan tidak hanya untuk satu daerah saja melainkan secara internasional. Oleh karena itu ketika melakukan *Market Research* seorang developer harus menentukan pasar apa yang ingin dituju dan apa saja yang populer disana.

2. Distribusi penghasilan

Tidak semua keuntungan akan diambil oleh tim developer bahkan terkadang merekapun tidak mendapat setengah dari penghasilan tersebut. Contohnya

pada pasar PC yang populer *Steam*, mengambil potongan sekitar 13%-30% sesuai dengan ketentuannya.

3. Besar game/Tipe game

Budget adalah penentu tipe gamenya. Biasanya tipe game ini dibagi menjadi tiga macam, yaitu:

- a) AAA Games - game yang memiliki budget diatas 100 juta dollar dan diproduksi oleh studio ternama.
- b) AA Games – Game yang memiliki budget dibawah 100 juta dollar dan diproduksi oleh studio yang besar
- c) Indie Games – Game yang diproduksi dengan budget yang minim, biasanya developernya mulai dari satu orang hingga satu tim yang kecil.

4. Platform

Untuk menentukan *target audience* dapat ditentukan dengan platform. Contohnya game premium yang berukuran AAA biasanya ditargetkan kepada platform PC ataupun Console, sedangkan game indie bertipe casual biasanya ditargetkan kepada mobile ataupun handheld console.

5. Model monetisasi

Monetisasi digunakan untuk menentukan tipe bisnis apa game yang ingin gunakan, ada beberapa tipe monetisasi yang ada dipasar sekarang diantaranya:

- a) *Buy to own* – Membeli game seperti biasanya sebanyak satu kali.
- b) *Subscription* – Menggunakan sistem langganan perbulan untuk mengakses gamenya.
- c) *Free to Play* – Game yang gratis dimainkan tetapi memiliki monetisasi yang lain.
- d) *In App Purchases* – Biasanya digunakan di *free to play*, pembelian didalam game yang dapat berupa *booster* ataupun *currency ingame*.
- e) *Ads* – Iklan yang ditampilkan didalam game, biasanya digunakan di *free to play*.

- f) *Play to Pay* – Mirip seperti subscription, tetapi pengguna membeli jumlah permainan yang dia akan mainkan. Contohnya 10 ronde game dihargai 1 dollar.
- g) *Cloud Platform* – Platform yang disediakan oleh perusahaan ternama yang memberikan akses untuk pengguna yang membelinya. Mirip seperti subscription tapi game akan dirotasi pada tiap bulannya dan pengguna dapat mengakses game yang tersedia.

6. *Genre game*

Genre game adalah kategori sebuah game. Contohnya *Action*, *Adventure*, dll. *Genre game* biasanya terdiri dari beberapa macam dengan variasi dan kombinasi yang berbeda tiap gamenya. *Genre game* biasanya yang menentukan kearah mana game akan didevelop berdasarkan dari hasil riset yang lainnya.

7. Tipe pemain

Ada beberapa cara untuk mengkategorikan pemain, tetapi tidak ada cara langsung. Biasanya pengkategorian player berdasarkan pembelian player. Mulai dari pembelian game sampai kebiasaaa pembelian game. Serta tipe pemain bisa juga dikategorikan berdasarkan *genre game* yang dia mainkan, contohnya casual, hardcore, competitive, dll.

Setelah melakukan *market research* maka developer dapat membangun game berdasarkan hasilnya untuk menghasilkan game yang akan terjual dan menghasilkan keuntungan.

2.2 Definisi *Data mining*

Data mining adalah kegiatan mencari dan mengekstrasi informasi yang belum diketahui sebelumnya (bersifat implisit dan dianggap tidak berguna) dari sebuah data [5]. Definisi lainnya adalah proses untuk menekstrak pola dan tren didalam sebuah dataset yang besar. Pada dasarnya, *data mining* dapat dikategorikan menjadi dua jenis, yaitu:

1. Descriptive

Data mining descriptive merupakan kegiatan untuk mengali nilai penting dari sebuah database yang tersembunyi dan menemukan pola data baru yang belum diketahui sebelumnya.

2. Predictive

Data mining predictive merupakan proses pencarian pola dari data dengan menggunakan beberapa atribut lain untuk di masa akan datang. Metode *data mining* klasifikasi termasuk yang terdapat dalam prediktif mining.

Secara umum terdapat 6 proses yang sering digunakan dalam *data mining* [5], yaitu:

1. Deskripsi (*Description*)

Deskripsi merupakan proses *data mining* untuk mengidentifikasi pola yang tersembunyi dalam sebuah data yang muncul secara berulang dan mengubah pola tersebut menjadi aturan yang baru.

2. Estimasi (*Estimation*)

Estimasi adalah proses *data mining* yang digunakan untuk memprediksi hal yang akan datang. Data didalam estimasi berupa numerik.

3. Prediksi (*Prediction*)

Prediksi merupakan proses untuk memperkirakan kondisi yang akan datang, proses ini sama seperti estimasi tetapi prediksi memiliki nilai yang diperikaran pada masa yang akan datang.

4. Klasifikasi (*Classification*)

Klasifikasi adalah prose menemukan sebuah model atau fungsi yang mendeskripsikan dan membedakan data kedalam kelas-kelas.

5. Klastering (*Clustering*)

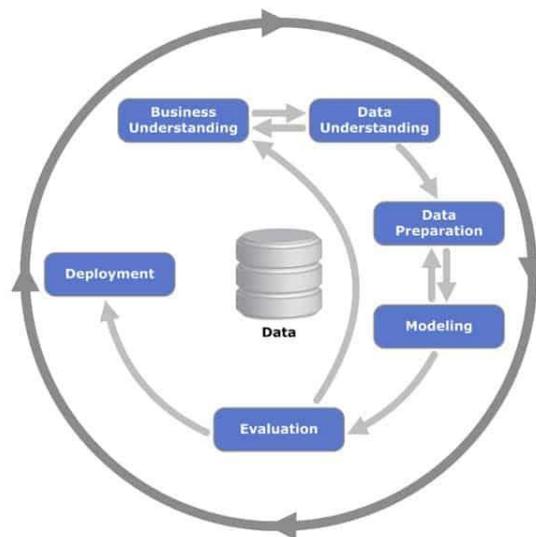
Klastering merupakan proses mengelompokkan data berdasarkan kelas data tertentu.

6. Asosiasi (*Association*)

Tugas asosiasi dalam *data mining* adalah menemukan atribut yang muncul dalam suatu waktu yang saling berkaitan dan mencari hubungan antara atribut tersebut. Asosiasi berusaha untuk mengungkapkan aturan untuk mengukur hubungan antara dua atau lebih atribut.

2.3 Metode *Cross-Industry Standard Process for Data mining (CRISP-DM)*

Cross-Industry Standard Process for Data mining (CRISP-DM) dikembangkan pada tahun 1996 dari analisa beberapa industri seperti Daimler Chrysler, SPSS, dan NCR. CRISP-DM menjadikan standar proses *data mining* sebagai strategi pemecahan masalah secara umum dari bisnis atau unit penelitian. CRISP-DM merupakan siklus hidup pada sebuah projek *data mining* yang terbagi menjadi enam fase. Biasanya fase tersebut bersifat adaptif, dimana fase berikutnya bergantung pada fase sebelumnya dan sebaliknya, fase yang sudah dikerjakan mungkin dapat kembali pada fase sebelumnya apabila diperlukannya evaluasi atau kesalahan pada fase sebelumnya [4].



Gambar 2.2 Skema Proses CRISP-DM

Enam Fase pada CRISP-DM [4], yaitu:

1. Pemahaman Bisnis (*Business Understanding*)

Fase ini bertujuan untuk memahami objektif dan persyaratan dalam sebuah projek. Adapun objektif yang ingin dicapai, yaitu:

- a) Menentukan tujuan dan kebutuhan secara detail dalam lingkup bisnis atau penelitian.
- b) Menentukan tujuan dari *data mining*.
- c) Memproduksi strategi awal untuk mencapai tujuan.

2. Pemahaman Data (*Data Understanding*)

Fase ini bertujuan untuk mengumpulkan data menanalisis data. Objektifnya yaitu:

- a) Mengumpulkan data awal.
- b) Melakukan analisis dan mengenali lebih lanjut data.
- c) Mengevaluasi kualitas data.

3. Pengolahan Data (*Data Preperation*)

Fase ini bertujuan untuk menyiapkan data untuk digunakan nantinya saat pemodelan. Objektifnya yaitu:

- a) Memilih data yang tepat yang dapat digunakan.
- b) Membersihkan data.
- c) Membangun data dengan menambahkan atribut yang diperlukan.
- d) Menggabungkan dataset dengan dataset yang lain jika dibutuhkan.

4. Pemodelan (*Modeling*)

Fase ini bertujuan untuk membangun dan menguji dengan beberapa teknik modeling yang ada. Objektifnya yaitu:

- a) Memilih teknik permodelan yang sesuai.
- b) Mengkalibrasi aturan model untuk mengoptimalkan hasil.
- c) Menentukan apakah permasalahan yang sama dapat menggunakan teknik yang berbeda.

5. Evaluasi (*Evaluation*)

Fase ini bertujuan untuk menilai hasil dari model yang telah dikerjakan. Objektifnya yaitu:

- a) Mengevaluasi hasil apakah model yang dikerjakan sudah sesuai dengan kriteria bisnis.
- b) Menentukan apakah terdapat permasalahan dari bisnis atau penelitian yang tidak tertangani dengan baik.
- c) Mengambil keputusan selanjutnya yang berkaitan dengan penggunaan hasil *data mining*.

6. Implementasi (*Deployment*)

Fase ini bertujuan untuk mengunnakan hasil dari *data mining* untuk digunakan oleh pengguna. Objektifnya yaitu:

- a) Menggunakan model yang dihasilkan. Terbentuknya sebuah model tidak menandakan terselainya sebuah proyek
- b) Membuat laporan hasil akhir

2.4 Metode *Association Rules*

Association Rules atau aturan asosiasi adalah teknik didalam *data mining* untuk menemukan pola kombinasi dari suatu *item* dan dapat digunakan untuk menemukan nilai *support* dan *confidence*. *Association rule* berbentuk “Jika mendahului, maka berakibat” [8][10]. *Association rules* memiliki dua tahap pengerjaan, yaitu [9]:

1. Mencari kombinasi yang sering terjadi didalam sebuah itemset.
2. Mendefinisikan *Condition* dan *Result* (untuk conditional *association rule*).

Dalam menentukan *association rule* diperlukannya sebuah item untuk memenuhi syarat minimum dari nilai *support*. *Support* adalah suatu ukuran yang menunjukkan sebara besar tingkat dominasi suatu item dari keseluruhan kejadian. Ukuran inilah yang akan menentukan *confidence* sebuah *item*. Untuk nilai *confidence* dapat ditemukan dengan rumus sebagai berikut:

$$Support (X) = \frac{Frekuensi (X, Y)}{N}$$

$$Support (X, Y) = P(A \cap B) = \frac{Frekuensi X dan Y}{N}$$

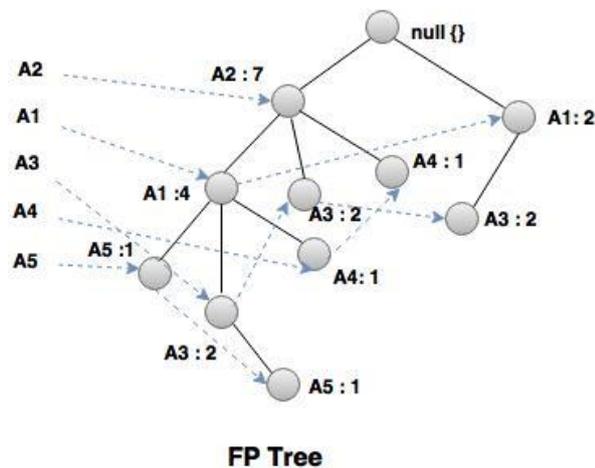
Setelah menentukan support sebuah item maka dicarilah *confidence* untuk menunjukkan hubungan antara 2 item atau lebih. Rumus mencari *confidence* sebagai berikut:

$$Confidence (X, Y) = P(X | Y) = \frac{Frekuensi X dan Y}{Frekuensi X}$$

Kedua ukuran ini nantinya akan digunakan untuk menentukan kekuatan sebuah pola dengan membandingkan pola tersebut dengan nilai minimum kedua parameter tersebut. Bila sebuah pola memenuhi kedua nilai minimum tersebut dapat disebut sebagai *interesting rule* atau *strong rule* [10].

2.4.1 Algoritma *FP-Growth*

Frequent pattern adalah pola yang paling sering terjadi didalam sebuah data. Algoritma *Frequent Pattern-Growth* atau biasa disebut *FP-Growth* adalah algoritma lanjutan dari algoritma Apriori, yang menutupi kelemahan dari algoritma tersebut yaitu menghasilkan kombinasi yang sangat banyak sehingga tidak efisien [11].



Gambar 2.3 Contoh *FP-Tree*

Algoritma ini menggunakan pemetaan data atau melakukan scan database untuk membentuk struktur *FP-tree* (lihat pada gambar). Dengan menggunakan struktur *tree* algoritma *FP-Growth* dapat langsung mengekstrak *frequent itemset* dari susunan *tree* yang sudah dibentuk. Setelah sebuah *FP-Tree* dibentuk maka langkah selanjutnya adalah sebagai berikut [12]:

1. Tahap pembangkitan *conditional pattern base*

Conditional Pattern Base merupakan subdata yang berisi *prefix path* (lintasan awal) dan *suffix pattern* (pola akhiran). Pembangkitan *conditional pattern base* didapatkan melalui *FP-Tree* yang telah dibangun sebelumnya.

2. Tahap pembangkitan *conditional FP-Tree*

Pada tahap ini, *support count* dari setiap item pada setiap *conditional pattern base* dijumlahkan, lalu setiap item yang memiliki jumlah *support count* lebih besar atau sama dengan *minimum support count* akan dibangkitkan dengan *conditional FP-Tree*.

3. Tahap Pencarian *Frequent Itemset*

Apabila *Conditional FP-Tree* merupakan lintasan tunggal (single path), maka didapatkan *frequent itemset* dengan melakukan kombinasi item untuk setiap

conditional *FP-Tree*. Jika bukan lintasan tunggal, maka dilakukan pembangkitan *FP-Growth* secara rekursif (proses memanggil dirinya sendiri).