

BAB II

TINJAUAN PUSTAKA

II.1. Perancangan

Perancangan adalah penggambaran, perencanaan dalam pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah kedalam suatu kesatuan yang utuh dan berfungsi perancangan sistem dapat dirancang dalam bentuk bagan atau sistem, yang merupakan alat bentuk grafik yang dapat digunakan untuk menunjukkan urutan-urutan proses dari sistem.

(Sumber : Syifaun Nafisah, 2006 : 2)

II.2. Animasi

Animasi mewujudkan ilusi (*illusion*) bagi pergerakan dengan memaparkan atau menampilkan satu urutan gambar yang berubah sedikit demi sedikit (*progressively*) pada kecepatan yang tinggi. Animasi digunakan untuk memberi gambaran pergerakan bagi sesuatu objek. Ia membolehkan sesuatu objek yang tetap atau statik dapat bergerak dan kelihatan seolah-olah hidup.

Animasi multimedia merupakan proses pembentukan gerak dari berbagai media atau objek yang divariasikan dengan efek-efek dan filter, gerakan transisi, suara-suara yang selaras dengan gerakan animasi tersebut.

Animasi di dalam sebuah aplikasi multimedia dapat menjanjikan suatu visual yang lebih dinamik serta menarik kepada penonton karena ia memungkinkan sesuatu yang mustahil atau kompleks berlaku di dalam kehidupan

sebenarnya direalisasikan di dalam aplikasi tersebut. (Sumber : Jurnal Agus Suheri “Animasi Multimedia Pembelajaran” Hal : 28-29).

II.3. Pembelajaran

Pembelajaran yang dilakukan pada saat ini adalah lebih banyak dilakukan dengan menulis, membaca, dan mendengarkan. Menulis, membaca, dan mendengarkan dilakukan ketika terjadi proses belajar mengajar dimana terjadi komunikasi antara pengajar dan peserta ajar yang biasa dilakukan di kelas.

Sumber pembelajaran yang disampaikan Pendidik kepada pengajar berasal dari buku yang berupa Teks dan atau gambar yang disampaikan kepada Peserta Didik dengan narasi dari pengajar tentang teks atau gambar. (Sumber : Jurnal Agus Suheri “ Animasi Multimedia Pembelajaran “ Hal : 30).

II.4. Berhitung

II.4.1. Penjumlahan

Penjumlahan merupakan operasi matematika yang menjumlahkan satu angka dengan angka lain sehingga menghasilkan suatu nilai tertentu yang pasti.

Simbol untuk operasi penjumlahan adalah tanda plus (+) Contoh : $2 + 3 = 5$.

(sumber: <http://ejournal.unp.ac.id/index.php/jupekhu>)

II.4.2. Pengurangan

Pengurangan merupakan salah satu dari empat operasi dasar aritmetika, dan pada prinsipnya merupakan kebalikan dari operasi perjumlahan. Operasi

pengurangan dinyatakan dengan tanda minus dalam notasi infix, dengan bentuk rumus: $c - b = a$. (sumber: <http://ejournal.unp.ac.id/index.php/jupekhu>).

II.5. Taman Kanak-Kanak

Taman kanak-kanak adalah bentuk satuan pendidikan pra sekolah pada jalur pendidikan sekolah, yang menyediakan program pendidikan dini bagi anak usia sekurang-kurangnya empat tahun sampai memasuki sekolah dasar, dengan lama pendidikan satu sampai dua tahun.

(Sumber: <http://staff.uny.ac.id/sites/default/files/132243759/JURNALPERANPENGELOLAANDITK.pdf>).

II.6. *Multimedia*

Multimedia dapat diartikan sebagai penggunaan beberapa media yang berbeda untuk menggabungkan dan menyampaikan informasi dalam bentuk text, audio, grafis, animasi, dan video. *Multimedia* dapat digunakan dalam:

1. Bidang periklanan yang efektif dan interaktif
2. Bidang pendidikan dalam penyampaian bahan pengajaran secara interaktif dan dapat mempermudah pembelajaran karena didukung oleh berbagai aspek: suara, video, animasi, teks, dan grafik
3. Bidang jaringan dan internet yang membantu dalam pembuatan website yang menarik, informatif, dan interaktif .

(Sumber : Jurnal Agus Suheri “Animasi Multimedia Pembelajaran” Hal:29-30).

II.7. Sistem perangkat yang digunakan

II.7.1. *Adobe Flash Pro CS6*

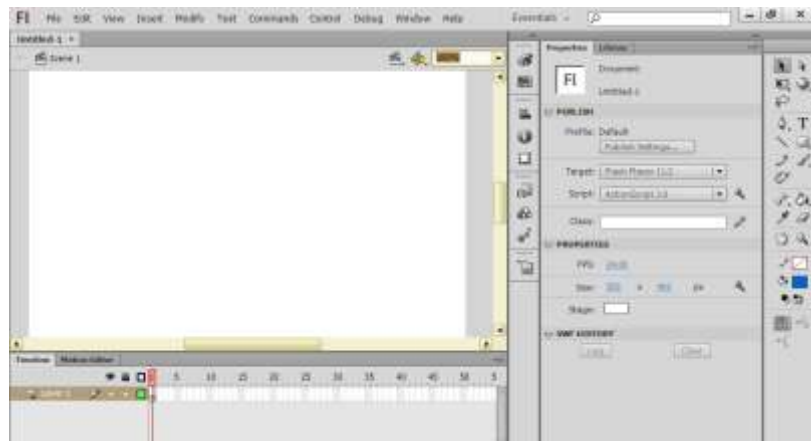
Adobe Flash merupakan sebuah program yang didesain khusus oleh *Adobe* dan program aplikasi standar *authoring tool professional* yang digunakan untuk membuat animasi dan bitmap yang sangat menarik untuk keperluan pembangunan *situs web* yang interaktif dan dinamis. *Flash* didesain dengan kemampuan untuk membuat animasi 2 dimensi yang handal dan ringan sehingga flash banyak digunakan untuk membangun dan memberikan efek animasi pada *website*, CD Interaktif dan yang lainnya. Selain itu aplikasi ini juga dapat digunakan untuk membuat animasi logo, *movie*, *game*, pembuatan navigasi pada *situs web*, tombol animasi, *banner*, menu interaktif, interaktif form isian, *e-card*, *screen saver* dan pembuatan aplikasi-aplikasi web lainnya. Dalam *Flash*, terdapat teknik-teknik membuat animasi, fasilitas *action script*, filter, custom easing dan dapat memasukkan video lengkap dengan fasilitas *playback* FLV. Keunggulan yang dimiliki oleh *Flash* ini adalah ia mampu diberikan sedikit *code* pemograman baik yang berjalan sendiri untuk mengatur animasi yang ada didalamnya atau digunakan untuk berkomunikasi dengan program lain seperti HTML, PHP, dan *Database* dengan pendekatan XML, dapat dikolaborasikan dengan *web*, karena mempunyai keunggulan antara lain kecil dalam ukuran *file outputnya*. (Sumber : Wahana Komputer, 2012 : 2).



Gambar II.1. Tampilan layar saat pertama kali membuka *Adobe Flash CS6*

(Sumber : Madcoms Madiun, 2012 : 2).

1. *Create from Template*, berguna untuk membuka lembar kerja dengan template yang tersedia dalam program *Adobe Flash CS6*.
2. *Open a Recent Item*, berguna untuk membuka kembali file yang pernah anda simpan atau pernah anda buka sebelumnya.
3. *Create New*, berguna untuk membuka lembar kerja baru dengan beberapa pilihan *script* yang tersedia.
4. *Learn*, berguna untuk membuka jendela *Help* yang berguna untuk mempelajari suatu perintah. (Sumber : Madcoms Madiun, 2012 : 2).



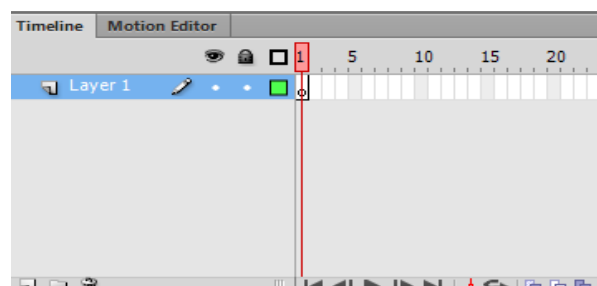
Gambar II.2. Ruang kerja *Adobe Flash CS6*

(Sumber : Madcoms Madiun, 2012 : 7).

1. *Timeline*

Timeline berguna untuk menentukan durasi animasi, jumlah *layer*, *frame*, menempatkan *script* dan beberapa keperluan animasi lainnya. Semua bentuk animasi yang akan dibuat diatur dan ditempatkan pada *layer* dalam *timeline*.

(Sumber : Madcoms Madiun, 2012 : 7).



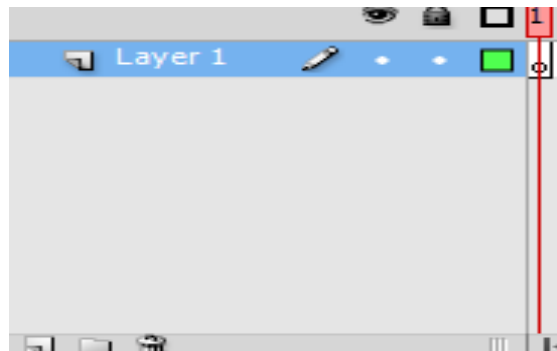
Gambar II.3. Tampilan *timeline*

(Sumber : Madcoms Madiun, 2012 : 7).

2. *Layer*

Layer berfungsi untuk menampung objek dalam lembar kerja. Dalam pembuatan animasi, sebaiknya setiap jenis objek yang digerakkan diletakkan pada

layer yang berbeda agar mempermudah pengeditan objek dalam *layer* tanpa mempengaruhi objek yang berbeda dalam *layer* lainnya. Semakin panjang animasi semakin banyak *layer* yang digunakan dan folder kerja yang dibutuhkan untuk mengatur objek animasi tersebut. (Sumber : Madcoms Madiun, 2012 : 21).

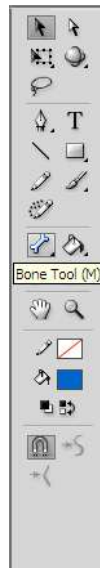


Gambar II.4. Tampilan *layer*

(Sumber : Madcoms Madiun, 2012 : 22).

4. *Panel Toolbox*

Panel ini banyak membuat *tool* yang akan digunakan mengolah dan membuat objek. Tiap *tool* memiliki fungsi yang berbeda tetapi saling mendukung. Jika pada tampilan program belum tampak *panel toolbox* maka langkah untuk menampilkannya dengan memilih menu *Window > Tools*. (Sumber : Wahana Komputer, 2012 : 18).



Gambar II.5. Tampilan *Panel Tools*

(Sumber : Wahana Komputer, 2012 : 19).

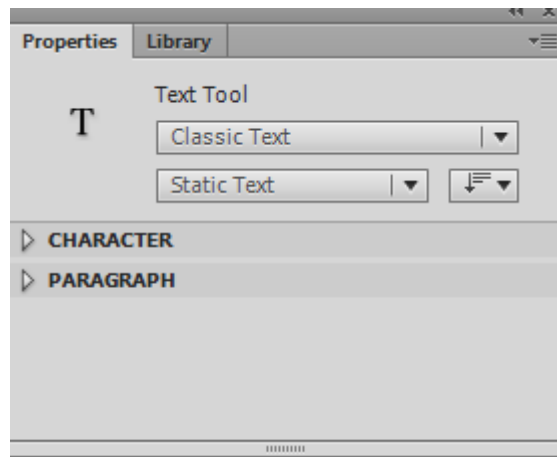
3. Teks

Teks merupakan salah satu komponen penting dalam membuat sebuah animasi untuk memperkuat informasi yang ditampilkan. *Adobe Flash Pro CS6* menyediakan 2 jenis teks yaitu : *Tlf text* dan *Classic text*. *Tlf text* mempunyai 3 tipe teks yaitu :

- *Read Only*, teks tidak dapat diseleksi atau diedit saat dipublish sebagai file SWF.
- *Selectable*, teks dapat diseleksi dan dapat disalin, tetapi tidak dapat diedit saat animasi dipublish sebagai file SWF.
- *Editable*, teks dapat diseleksi dan diedit saat animasi dipublish sebagai SWF.

Untuk jenis *Classic Text* mempunyai 3 tipe teks yaitu :

- *Static text*, adalah tipe teks yang mempunyai sifat statis tidak mengalami perubahan isi teks pada saat animasi dijalankan.
- *Dynamic text*, adalah jenis teks yang dapat berubah secara dinamis pada saat animasi dipublish sebagai file SWF.
- *Input text*, adalah jenis teks yang digunakan untuk proses *input* data yang dikaitkan dengan perintah *script* tertentu. (Sumber : Madcoms Madiun, 2012 : 73-74).



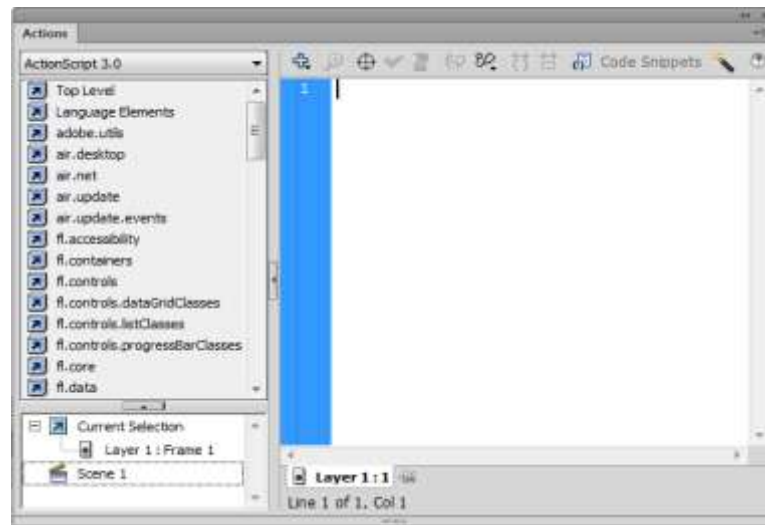
Gambar II.6. Tampilan Teks

(Sumber : Madcoms Madiun, 2012 : 74).

4. *ActionScript*

ActionScript digunakan untuk memberikan perintah *action* pada sebuah *frame*, tombol, atau *movie clip*. *Flash* memiliki dua *ActionScript* yaitu:

- *ActionScript* 3.0 menjalankan aplikasi lebih cepat.
- *ActionScript* 2.0 lebih sederhana daripada *ActionScript* 3.0. Meski *flash player* tersusun oleh *ActionScript* 2.0 yang lebih lambat dari *ActionScript* 3.0.



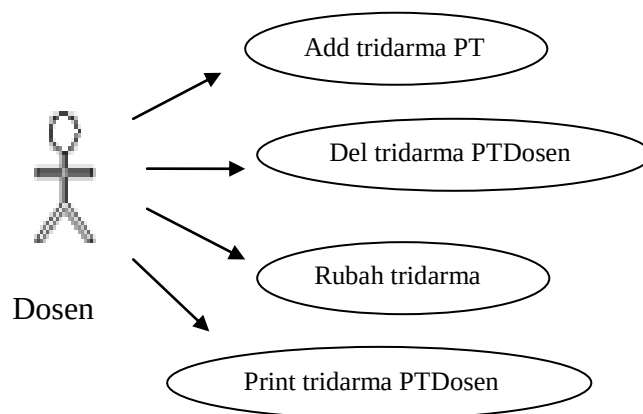
Gambar II. 7. Tampilan *ActionScript*

(Sumber : Madcoms Madiun, 2012 : 74).

II.8. Konsep UML (*Unified Modelling Language*)

1. *Use Case Diagram*

Use case adalah merupakan pemodelan untuk menggambarkan kelakuan (*behavior*) sistem yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih *actor* dengan sistem yang akan dibuat. Dengan pengertian yang cepat, diagram *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut. Terdapat beberapa simbol dalam menggambarkan *use case* yaitu *use case*, *actor* dan relasi. (Sumber : Yuni Sugiarti, 2013: 41)



Gambar II.8. Contoh *Use Case Model*

(Sumber : Yuni Sugiarti, 2013: 45)

2. Activity Diagram (Aktivitas)

Diagram aktivitas atau *Activity Diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan *actor*, jadi aktivitas yang dapat dilakukan oleh sistem. Diagram aktivitas mendukung perilaku *paralell*.

Diagram aktivitas juga banyak digunakan untuk mendefenisikan hal-hal berikut:

- Rancang proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefenisikan.
- Urutan atau pengelompokan tampilan dari sistem / *user interface* dimana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan.

- Rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujiannya.

Activity diagrams menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, decision yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* juga dapat menggambarkan proses *parallel* yang mungkin terjadi pada beberapa eksekusi.

Activity diagram merupakan state diagram khusus, di mana sebagian besar *state* adalah *action* dan sebagian besar transisi di-trigger oleh selesainya *state* sebelumnya (*Internal Processing*). Oleh karena itu *activity digram* tidak menggambarkan *behaviour internal* sebuah sistem (dan interaksi antar subsistem) secara eksak, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari *level* atas secara umum. (Sumber : Yuni Sugiarti, 2013: 75) .