BAB III

OBJEK DAN METODE PENELITIAN

3.1 Objek Penelitian

Sebagian Besar objek penelitian yang di gunakan oleh penulis adalah tempat sanggar SURAGADA di bandung dimana terletak di jalan Pajajaran Gg. Babakan Ciamis 21/25 Kota Bandung . Adapun penjelasan mengenai objek penelitian adalah sebagai berikut :

3.1.1. Sejarah Singkat Sanggar

Sanggar SURAGADA ini berdiri sudah lama berdiri ,lingkung SURAGADA terletak di jalan Pajajaran Gg. Babakan Ciamis 21/25 Kota Bandung . Selain sebagai sanggar pembelajaran seni tradisional sunda di sanggar juga menyediakan apreasi seni bala-bala yaitu seni music yang terdiri atas beberapa instrument lagu yang dikolaborasikan. Bahkan barang – barang bekas dijadikan sebagai alat music di apresiasi seni bala-bala.

3.1.2. Visi Dan Misi Sanggar

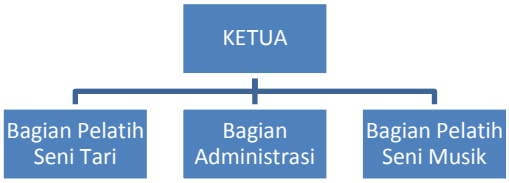
Visi

Menjadi sanggar terbaik di bidang kesenian daerah dan terus berkarya untuk generasi muda dan kemajuan bangsa.

Misi

Membudidayakan kesenian daerah dan alat musik tradisional di era sekarang ini dan untuk mengajak serta memberi pelajaran kepada generasi muda untuk kembali mengenal seni trasisional dan alat musik tradisional.

3.1.3. Struktur Organisasi Sanggar



Gambar 3.1 Struktur Organisasi Sanggar SURAGADA

(Sumber : Sanggar SURAGADA)

3.1.4. Deskirpsi Tugas

1. Bagian pelatihan seni tari bertugas untuk melatih kesenian tari.
2. Bagian administrasi bertugas untuk mengatur seluruh urusan administrasi sanggar.
3. Bagian pelatihan seni musik bertugas untuk melatih kesenian musik.

3.2 Metode Penelitian

Sugiyono (2009 :3) mendeskripsikan metode penelitian adalah sebagai berikut : Metode Penelitian diartikan sebagai cara ilmiah untuk mendapatkan data dengan tujuan dan kegunaan tertentu.

Penelitian adalah kegiatan untuk memperoleh fakta-fakta atau prinsip-prinsip (baik kegiatan untuk oenemuan, pengujian atau pengembangan) dari suatu pengetahuan secara sistematis berdasarkan ilmu pengetahuan (metode ilmiah).

3.2.1 Desain Penelitian

Untuk melakukan suatu penelitian perlu dilakukan perencanaan penelitian, agar peneltian yang dilakukan dapat berjalan dengan baik dan sistematis. Desain penelitian adalah semua proses yang diperlukan dalam perencanaan dan pelaksaan penelitian, dalam penelitian ini menggunakan jenis penelitian deskriptif dan penelitian tindakan (*action research*).

Penelitian deskriptif adalah suatu bentuk penelitian yang ditujukan untuk mendeskripsikan fenomena-fenomena yang ada, baik fenomena alamiah maupun

fenomena buatan manusia. Fenomena itu bisa berupa bentuk, aktivitas, karakteristik, perubahan, hubungan, kesamaan, dan perbedaan antara fenomena yang satu dengan fenomena lainnya (Sukmadinata, 2006:72).

Penelitian tindakan bertujuan untuk memberikan kontribusi baik kepada sisi praktis dari orang sisi praktis dari seorang yang dalam situasi bermasalah dan untuk memajukan ilmu pengetahuan secara bersamaan. Gilmore (1986:161).

3.2.2 Jenis dan Metode Pengumpulan Data

Dalam penelitian ini sumber data primer diperoleh dari pengamatan langsung (observasi) dan wawancara, sedangkan data sekunder yakni dokumen-dokumen yang ada di perpustakaan, sumber litelatur serta media elektronik yang berhubungan dengan penelitian..

3.2.2.1 Sumber Data Primer

Sumber data Primer adalah data atau segala informasi yang diperoleh dan didapat oleh penulis langsung dari sumber-sumber, baik sumber individu maupun sekelompok bagian dari objek penelitian seperti hasil dari Wawancara dan Observasi langsung pada objek yang diteliti dalam hal ini yaitu data yang didapat dari sanggar.

1. Observasi

Pengumpulan data dengan mengadakan penelitian dan peninjauan langsung ke sanggar seni mayang sari di bandung dimana terletak di jalan Pajajaran Gg. Babakan Ciamis no. 21/25 Bandung tentang alat tradisional kacapi.Peninjauan disini dilakukan untuk mengetahui jenis-jenis dari kacapi, bentuk kacapi, bagian dan fungsi dari kacapi, jumlah senar yang digunakan dan bagaimana cara memainkan kacapi.

2. Wawancara

Pengumpulan data dengan mengadakan diskusi secara langsung kepada pengelola dan ketua sanggar yaitu Bapak Onong sebagai pengelola dan pemilik sanggar. Bapak Wawan yaitu berdiskusi alat tradisional sunda yaitu Kacapi. Bagaimana cara bermain, notasi, dan bagian-bagian dari kacapi tersebut, beserta fungsi dari masing-masing bagian tersebut.

3.2.2.2 Sumber Data Sekunder

Pengumpulan data sekunder seperti contoh- contoh penggunaam metode di dalam flash, notasi pelog , dan sebagainya dilakukan dengan menelaah data sekunder yang digunakan untuk menunjang, melengkapi dan menyempurnakan data primer, diperoleh dari kepustakaan tentang info mengenai kacapi, mulai dari sejarah kacapi, bentuk, jenis, fungsi, dan notasi. Penggunaan media Internet sebagai media pembelajaran bagaimana cara memainkan kacapi, dan serta hasil penelitian sebelumnya yang berhubungan dengan masalah dalam penelitian ini.

3.2.3 Metode Pendekatan dan Pengembangan Sistem

Dalam sub bab ini dijelaskan mengenai metode pendekatan,pengembangan sistem, dan alat bantu analisis perancangan. Berikut ini adalah uraian dari metode pendekatan dan pengembagan sistem.

3.2.3.1 Metode Pendekatan Sistem

Metode Pendekatan sistem yang digunakan oleh penulis dalam penelitian ini yaitu menggunakan metode Object Oriented Programming yaitu merupakan

pemrograman yang berorientasikan kepada objek, dimana semua data dan fungsi dibungkus dalam class-class atau object-object. Setiap object dapat menerima pesan, memproses data, mengirim, menyimpan dan memanipulasi data. Penggambaran Object Oriented Programing ini dapat digambarkan dengan tools berupa UML. Beberapa object berinteraksi dengan saling memberikan informasi satu terhadap yang lainnya. Masing-masing object harus berisikan informasi mengenai dirinya sendiri dan dapat dihubungkan dengan object yang lain. Objek yang saling berinteraksi ini dikemas dalam suatu bentuk berupa diagram. Diagram yang digunakan dalam UML ini adalah Diagram *Use-Case*, Diagram Aktivitas, Collaboration Diagram, Class Diagram, Sequence Diagram, Deployment Diagram, dan Komponen Diagram.

3.2.3.2 Metode Pengembangan Sistem

Metode pengembangan perangkat aplikasi menggunakan metode prototipe yaitu proses yang digunakan untuk membantu pengembang perangkat lunak dalam membentuk model dari perangkat lunak yang harus dibuat. Kadang-kadang pemakai atau user hanya memberikan beberapa kebutuhan umum software tanpa detail input, proses atau detail output. Di lain waktu

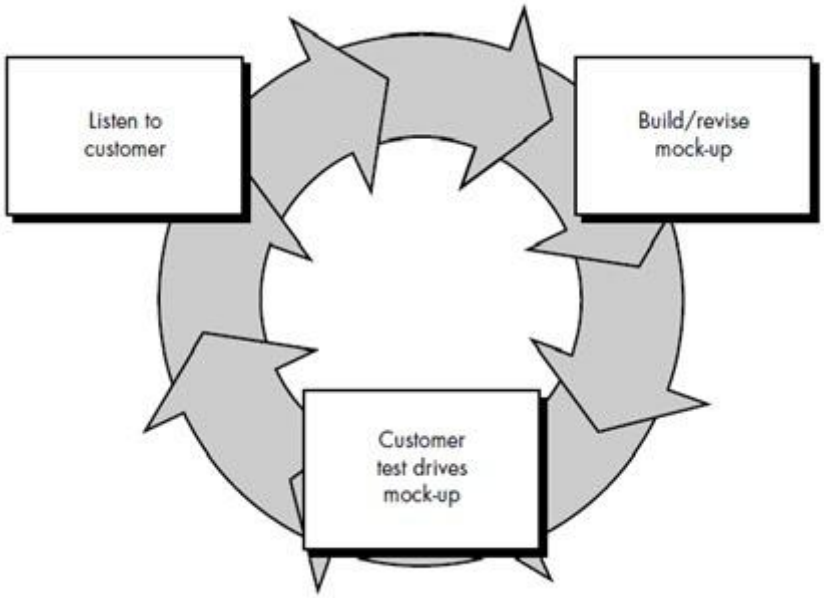
mungkin dimana tim pembangun (developer) tidak yakin terhadap efisiensi dari algoritma yang digunakan, tingkat adaptasi terhadap sistem operasi atau rancangan form user interface. Ketika situasi seperti ini terjadi model prototyping sangat membantu proses pembangunan software.

Seperti pada semua metode, prototyping dimulai dari pengumpulan kebutuhan. Dengan perencanaan yang cepat akan dibentuk konstruksi dari prototipenya. Prototipe ini dievaluasi oleh pelanggan dan digunakan untuk mengelola kembali kebutuhan dari perangkat lunak yang dikembangkan. Suatu

proses iterasi terjadi, setelah prototipe disesuaikan dengan kebutuhan pelanggan, sementara pihak pengembang makin mengerti keinginan pemakai. Tahapan proses model prototyping, bisa dijelaskan sebagai berikut:

1. Pengumpulan kebutuhan : developer dan pelanggan bertemu dan menentukan tujuan umum, kebutuhan yang diketahui dan gambaran bagian-bagian yang akan dibutuhkan berikutnya. Detail kebutuhan mungkin tidak dibicarakan disini, pada awal pengumpulan kebutuhan.
2. Perancangan : perancangan dilakukan cepat dan rancangan mewakili semua aspek software yang diketahui, dan rancangan ini menjadi dasar pembuatan prototipe.

3. Evaluasi prototipe: pelanggan mengevaluasi prototipe yang dibuat dan digunakan untuk memperjelas kebutuhan software. Perulangan ketiga proses ini terus berlangsung hingga semua kebutuhan terpenuhi. Prototype-prototype dibuat untuk memuaskan kebutuhan pelanggan dan untuk memahami kebutuhan pelanggan lebih baik.



Gambar 3.2 Metode pengembangan Prototype
(Sumber : Mahil Carr. Prototyping and Software Development Approaches)

3.2.3.3 Alat Bantu Analisis dan Perancangan

1. UML

Unified Modelling Language (UML) adalah bahasa grafis untuk mendokumentasi menspesifikasi dan membangun sistem perangkat lunak. UML berorientasi objek menerapkan banyak level abstraksi tidak bergantung proses pengembangan, tidak bergantung bahasa dan teknologi, pemaduan beberapa notasi di beragam metologi, usaha bersamadan banyak pihak, didukung oleh kakas-kasa yang diintegerasikan lewat XML(XMI). Standar UML dikelola oleh OMG (*Object Management Group*).

Tujuan UML :

1. Menyediakan bahasa pemodelan visual yang ekspresif dan siap pakai untuk mengembangkan dan pertukaran model-model yang berarti.
2. Menyediakan mekanisme perluasan dan spesialisasi untuk memperluas konsep-konsep inti.
3. Mendukung spesifikasi independen bahasa pemograman dan proses pengembangan tertentu.
4. Menyediakan basis formal untuk pemahaman bahasa pemodelan.

5. Mendukung konsep-konsep pengembangan level tinggi seperti komponen, kolaborasi, *framwork* dan *pattern*.

2. Diagram *Use-Case (Use-Case Diagram)*

Bersifat statis. Diagram ini memperlihatkan himpunan *use case* dan aktor-aktor (suatu jenis khusus dari kelas). Diagram ini sangat penting untuk mengorganisasi dan memodelkan perilaku dari suatu sistem yang dibutuhkan serta diharapkan pengguna.

3. Diagram Aktivitas *(Activity Diagram)*

Bersifat dinamis. Diagram ini adalah diagram tipe khusus dari digram state yang memperlihatkan aliran dari suatu aktifitas ke aktifitas yan lainnya dalam suatu sistem. Diagram ini penting dalam pemodelan fungsi-fungsi dalam satu sistem dan memberi tekanan pada aliran kendali antar objek.

4. Collaboration Diagram

Secara fungsional digram ini hampir mirip dengan sequence diagram. Collaboration diagram memfokuskan pada interaksi dan hubungan diantara sekumpulan objek yang berkolaborasi. Hubungan-hubungan tersebut memperlihatkan objek actual dan

relasi yng terjadi diantara mereka yang digambarkan dengan sebuah garis. Diatas garis terdapat alur pesan yang dikirim objek yang berhubungan tersebut.

5. Class Diagram

Class diagram mendepkripsikan jenis-jenis objek dalam sistem dan berbagai macam hubungan statis yang terdapat diantara mereka. Class diagram juga menunjukan property dan operasi sebuah class dan batasan-batasan yang terdapat dalam hubungan-hubungan objek tersebut.

6. Sequence Diagram

Sequence diagram secara khusus menjabarkan behavior sebuah sekenario tunggal. Sequence diagram menunjukan sebuah objek contoh dan pasan-pesan yang melewati objek-objek dalam use case.

7. Deployment Diagram

Menggambarkan tata letak sebuah sistem secara fisik, menampakkan bagian-bagian software yang berjalan pada bagian-bagian hardware, menunjukkan hubungan komputer dengan perangkat (nodes) satu sama lain dan jenis hubungannya. Di dalam nodes, executeable component dan object yang dialokasikan untuk memperlihatkan unit perangkat lunak yang

dieksekusi oleh node tertentu dan ketergantungan komponen.

8. Component Diagram

Component software merupakan bagian fisik dari sebuah sistem, karena menetap di komputer tidak berada di benak para analis. Komponen merupakan implementasi software dari sebuah atau lebih class. Komponen dapat berupa source code, komponent biner, atau executable component. Sebuah komponent berisi informasi tentang logic class atau class yang diimplementasikan sehingga membuat pemetaan dari logical view ke component view. Sehingga component diagram merepresentasikan dunia riil yaitu component software yang mengandung component, interface dan relationship.

3.2.4 Pengujian Software

Software testing adalah investigasi yang dilakukan kepada software untuk mengetahui kualitas suatu produk. dilakukan untuk mengeliminasi defect yang terjadi setelah software diluncurkan, pengujian software sendiri tidak akan pernah secara penuh mengenali defect pada suatu software.black box testing dan white box testing adalah metode yang dikenal untuk menguji software.

Pengujian black-box sering disebut juga dengan pengujian tingkah laku (behavoiur testing) yang lebih terfokus kepada kebutuhan fungsional dari perangkat lunak, pengujian black-box memungkinkan pembuat perangkat lunak untuk menentukan kondisi yang terjadi untuk suatu masukan yang akan menjalankan semua kebutuhan fungsional (Pressman 2002:551).

Pengujian black box merupakan pendekatan komplementer dari teknik white box, karena pengujian black box diharapkan mampu mengungkap kelas kesalahan yang lebih luas dibandingkan teknik white box. Pengujian black box berfokus pada pengujian persyaratan fungsional perangkat lunak, untuk mendapatkan serangkaian kondisi input yang sesuai dengan persyaratan fungsional suatu program. Pengujian black box berusaha menemukan kesalahan dalam kategori :

1. fungsi-fungsi yang tidak benar atau hilang
2. kesalahan interface
3. kesalahan dalam struktur data atau akses database eksternal
4. kesalahan kinerja
5. inisialisasi dan kesalahan terminasi.

Untuk pengujian aplikasi simulasi kacapi berbasis Flash yang diterapkan di aplikasi Android ini menggunakan metode pengujian Black Box.