

## BAB III

### ANALISA DAN PERANCANGAN SISTEM

#### 3.1 Analisis Sistem

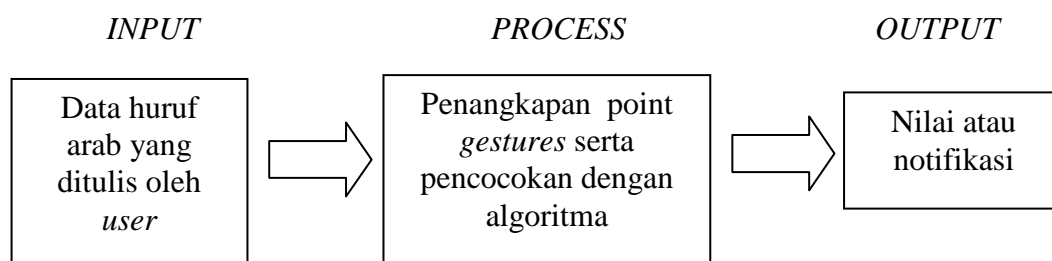
Untuk pengumpulan data yang diperlukan dalam pengerjaan tugas akhir membuat aplikasi menulis huruf arab berbasis Android, ada beberapa langkah yang dilakukan, yaitu pengamatan/observasi. Observasi yang dilakukan meliputi observasi terhadap media yang digunakan dalam pembelajaran menulis huruf arab saat ini, tata cara interaksi media terhadap *user* dalam menulis huruf arab.

Sebagian besar media pembelajaran menulis huruf arab saat ini berupa Lembar Kegiatan Belajar (LKB), tidak berwarna dan didalamnya hanya berupa titik-titik yang membentuk huruf arab. Pola tersebut akan menuntun *user* dalam menulis huruf arab dengan cara menebalkan dan mengikuti titik-titik yang membentuk pola huruf arab. Pada saat menulis huruf arab tersebut, terkadang *user* tidak mengetahui huruf yang ditulisnya dan juga tidak mengetahui arah yang benar untuk menulis huruf arab serta benar atau salahnya tulisan huruf arab. Pembelajaran menulis huruf arab biasanya diajarkan pada sekolah Taman Pendidikan Anak-anak (TPA) dan banyak jumlah anak yang belajar di TPA. Dari sekian anak yang belajar tersebut, pastilah ada beberapa anak yang tidak memperhatikan atau lupa apa yang diajarkan oleh gurunya, sehingga anak tersebut tidak memahami dalam menulis huruf hijaiyah.

Dari permasalahan tersebut, akan dibuat aplikasi menulis huruf arab berbasis Android yang digunakan dalam belajar menulis huruf arab. Aplikasi yang akan dibuat diharapkan dapat meningkatkan media pembelajaran yang

sebelumnya yaitu teks atau tulisan hijaiyah tanpa warna (hitam putih) dan petunjuknya kurang jelas. Meskipun dibimbing oleh guru TPA terkadang ada beberapa murid yang tidak memperhatikan karena banyaknya murid belajar sehingga anak tersebut tidak memahami tata cara penulisan huruf hijaiyah. Diharapkan sistem dapat membantu dalam belajar menulis huruf arab, yaitu dengan cara sistem memberikan pengenalan tata cara penulisan huruf hijaiyah sesuai dengan kaidah dengan menampilkan gambar gerak atau petunjuk gambar dan cara lainnya hasil *input user* akan dilakukan pengecekan oleh sistem salah atau benar tulisan tersebut.

Pada aplikasi terdapat fitur melihat histori nilai, sehingga ketika menggunakan aplikasi, *user* harus mempunyai *user* untuk *login* agar *user* dapat melihat histori masing-masing. Selain itu, sistem juga terdapat interaksi dengan *user*, aplikasi akan menampilkan hasil prosentase benar atau salah dari *input-an gestures* dari *user*. Proses validasi tulisan huruf arab awalnya dilakukan dengan cara penangkapan titik koordinat data kemudian melakukan proses pencocokan dengan *gestures*. Setelah proses pencocokan, sistem akan mengeluarkan *output* hasil validasi sistem terhadap tulisan *user* yaitu berupa nilai atau notifikasi benar atau salah.



Gambar 3.1 Blok Diagram Menulis Huruf Arab dengan *gestures*

Informasi validasi yang dihasilkan oleh sistem dapat membantu *user* dalam memahami penulisan huruf arab yang sesuai dengan kaidah. Bila *user* salah dalam menulis, sistem akan memberikan pembenaran sehingga *user* akan lebih memahami dalam menulis huruf arab.

### 3.1.1 Analisis Kebutuhan Sistem

Aplikasi menulis huruf arab yang akan dibangun membutuhkan data *input gestures* yang sesuai dengan kaidah sebagai acuan dalam pencocokan *input gestures* dari *user*. Hasil pencocokan tersebut nantinya akan dijadikan histori nilai.

Berikut adalah kebutuhan *user* pada tabel 3.1.

Tabel 3.1 Kebutuhan *User*

| No | Pengguna   | Kebutuhan  | Info yang dihasilkan   |
|----|--|--|--|
| 1  | <i>User</i><br>(Khususnya orang awam yang belum bisa menulis huruf hijaiyah) | a. Pengenalan tata cara huruf hijaiyah sesuai dengan kaidah<br>b. Adanya interaksi yang memberikan info salah atau benar tulisan <i>user</i> dalam menulis huruf hijaiyah<br>c. Dapat melihat hasil tulisan sebelumnya untuk melihat perkembangan <i>user</i> dalam latihan menulis huruf hijaiyah | a. Dapat memberikan info tata cara penulisan huruf hijaiyah sesuai dengan kaidah berupa gambar petunjuk atau gambar gerak.<br>b. Info prosentase benar tulisan <i>user</i> dari jumlah <i>input</i> tulisan ( <i>gestures</i> ) <i>user</i><br>c. Menampilkan nilai dan histori tulisan <i>user</i> selama dalam proses latihan. |

Aplikasi menulis huruf arab ini akan membantu *user* dalam proses latihan menulis huruf arab. Pada Tabel 3.2 merupakan kebutuhan *functional* dan *non-functional* dari Aplikasi Menulis Huruf Arab dengan Teknologi *Gestures* Berbasis Android.

Tabel 3.2 *Functional* dan *Non-functional* Aplikasi Menulis Huruf Arab

| Fuctional   | Non Functional  |
|---|---|
| <p>A. Fungsi <i>login</i> dan <i>list user</i>, digunakan untuk membuktikan bahwa <i>user</i> tidak dapat masuk ke menu utama aplikasi bila tidak mempunyai <i>user login</i>. Kemudian ketika <i>user</i> telah mempunyai <i>user login</i>, barulah dapat masuk ke menu utama aplikasi. <i>User login</i> ini berguna untuk melihat perkembangan dalam latihan menulis huruf hijaiyah karena sistem dapat menyimpan hasil nilai dari latihan menulis <i>user</i></p> <ol style="list-style-type: none"> <li>1. <i>User</i> dapat membuat <i>user login</i> baru dengan memasukkan nama <i>user</i>.</li> <li>2. <i>User</i> dapat melihat nama <i>user</i> yang telah dibuat</li> <li>3. <i>User</i> dapat masuk ke menu utama aplikasi dengan <i>user login</i></li> </ol> | <ol style="list-style-type: none"> <li>1.1. Sistem melakukan simpan data <i>user</i>.</li> <li>2.1. Sistem menampilkan data <i>user</i> yang ada pada database <i>user login</i>.</li> <li>3.1. Sistem memvalidasi <i>user login</i> dan kemudian masuk ke menu utama aplikasi</li> </ol>   |
| <p>B. Fungsi Pengenalan Huruf Hijaiyah, dibutuhkan <i>user</i> sebagai pengetahuan awal yaitu mengenal tata cara menulis huruf hijaiyah sesuai dengan kaidah beserta pelafalannya. Hal ini sangat bermanfaat untuk <i>user</i> awam atau pemula yang belum bisa menulis huruf hijaiyah, sehingga <i>user</i> dapat melihat tata cara menulis huruf hijaiyah sesuai dengan kaidah dan diharapkan <i>user</i> bisa melakukan latihan menulis huruf hijaiyah.</p> <ol style="list-style-type: none"> <li>1. <i>User</i> dapat memilih salah satu dari 28 huruf hijaiyah</li> <li>2. <i>User</i> dapat melihat tata cara berupa petunjuk gambar, gambar gerak dan mendengarkan pelafalan huruf hijaiyah yang dipilih.</li> </ol>  | <ol style="list-style-type: none"> <li>1.1. Sistem menampilkan gambar 28 huruf hijaiyah</li> <li>1.2. Sistem memproses huruf hijaiyah yang dipilih oleh <i>user</i> dan menampilkan halaman pengenalan huruf hijaiyah.</li> <li>2.1. Sistem menampilkan fitur pada pengenalan huruf hijaiyah</li> <li>2.2. Sistem dapat menampilkan gambar tata cara penulisan huruf hijaiyah, gambar gerak tata cara penulisan huruf hijaiyah dan memainkan suara pelafalan huruf hijaiyah.</li> </ol> |
| <p>C. Fungsi latihan menulis huruf hijaiyah sangat membantu <i>user</i> dalam melakukan latihan menulis huruf hijaiyah. Maka dari itu, diperlukan pembuktian apakah pencocokan <i>gestures</i> sudah sesuai dengan yang diharapkan, yaitu tulisan hasil <i>user</i> apakah sesuai dengan tata cara penulisan sesuai dengan kaidah. Pada fitur ini juga terdapat prosentase benar dari jumlah <i>user</i> melakukan latihan menulis</p> <ol style="list-style-type: none"> <li>1. <i>User</i> dapat memilih salah satu dari 28</li> </ol>  | <ol style="list-style-type: none"> <li>1.1. Sistem menampilkan gambar 28 huruf hijaiyah</li> <li>1.2. Sistem memproses huruf hijaiyah yang dipilih oleh <i>user</i> dan menampilkan halaman latihan menulis huruf hijaiyah</li> <li>2.1. Sistem menyediakan kanvas sebagai media menulis <i>user</i></li> <li>3.1. Sistem mendeteksi tulisan</li> </ol>   |



| Fuctional   | Non Functional   |
|---|--|
| <p>huruf hijaiyah</p> <ol style="list-style-type: none"> <li>2. <i>User</i> dapat menulis huruf hijaiyah pada kanvas yang telah disediakan oleh aplikasi</li> <li>3. <i>User</i> mendapatkan info prosentase benar, total jumlah tulisan yang dilakukan oleh <i>user</i> dan berapa kali salah.</li> </ol>  | <p><i>user</i> benar atau salah</p> <ol style="list-style-type: none"> <li>3.2. Sistem menghitung total jumlah <i>user</i> menulis</li> <li>3.3. Sistem melakukan perhitungan prosentase benar</li> <li>3.4. Sistem menampilkan info jumlah tulisan <i>user</i>, prosentase benar dan jumlah tulisan yang salah.</li> </ol>  |
| <p>D. Fungsi kuis dan menampilkan hasil dari latihan mengerjakan soal kuis, berguna untuk mengetahui kemampuan <i>user</i> dalam menulis huruf hijaiyah. Maka dari itu, diperlukan pembuktian apakah nilai yang keluar sesuai dengan latihan yang dilakukan oleh <i>user</i>. Jika tulisan <i>user</i> benar maka sistem juga akan mendeteksi benar dan juga sebaliknya, jika hasil tulisa <i>user</i> salah sistem akan mendeteksi salah.</p> <ol style="list-style-type: none"> <li>1. <i>User</i> dapat menulis huruf hijaiyah sesuai pilihan pada latihan kuis tes kemampuan.</li> <li>2. <i>User</i> dapat menulis huruf hijaiyah sesuai soal yang diberikan oleh sistem pada latihan kuis soal evaluasi.</li> <li>3. <i>User</i> hanya dapat menuliskan huruf hijaiyah pada latihan kuis sebanyak lima kali.</li> <li>4. <i>User</i> dapat melihat nilai dari hasil latihan kuis yang dilakukan.</li> </ol> | <ol style="list-style-type: none"> <li>1.1. Sistem menampilkan gambar 28 huruf hijaiyah</li> <li>1.2. Sistem memproses huruf hijaiyah yang dipilih oleh <i>user</i> dan menampilkan halaman kuis tes kemampuan</li> <li>1.3. Sistem menyediakan kanvas sebagai media menulis <i>user</i> pada kuis tes kemampuan</li> <li>2.1. Sistem menyediakan kanvas sebagai media menulis <i>user</i> pada kuis soal evaluasi</li> <li>2.2. Sistem memberikan soal secara acak kepada <i>user</i> pada kuis soal evaluasi</li> <li>3.1. Sistem membatasi kuis sebanyak lima soal.</li> <li>4.1. Sistem melakukan kalkulasi nilai dari hasil tulisan <i>user</i> melakukan latihan kuis</li> </ol> |

| Fuctional   | Non Functional   |
|---|--|
| <p>E. Fungsi menampilkan histori <i>nilai</i> juga sangatlah penting untuk melihat perkembangan <i>user</i> dalam menulis huruf hijaiyah. Semua hasil latihan kuis yang dilakukan <i>user</i> tersimpan pada fitur nilai. Maka dari itu perlu pembuktian apakah hasil dari <i>user</i> melakukan latihan kuis sama dengan yang ditampilkan pada fitur nilai.</p> <ol style="list-style-type: none"> <li>1. <i>User</i> dapat memilih kategori nilai tes kemampuan dan nilai soal evaluasi</li> <li>2. <i>User</i> dapat melihat histori nilai tes kemampuan dan nilai soal evaluasi.</li> <li>3. <i>User</i> dapat melihat tulisan hijaiyah dari latihan kuis yang telah dilakukan sebelumnya.</li> </ol> | <ol style="list-style-type: none"> <li>1.1. Sistem menampilkan jenis penilaian yaitu nilai tes kemampuan dan nilai soal evaluasi</li> <li>2.1. Sistem menampilkan nilai sesuai pilihan <i>user</i> disesuaikan dengan <i>user login</i> dan <i>id_detail</i> kuis</li> <li>3.1. Sistem menampilkan hasil tulisan huruf hijaiyah oleh <i>user</i> disesuaikan dengan <i>user login</i> dan <i>id_detail</i> kuis</li> </ol> |

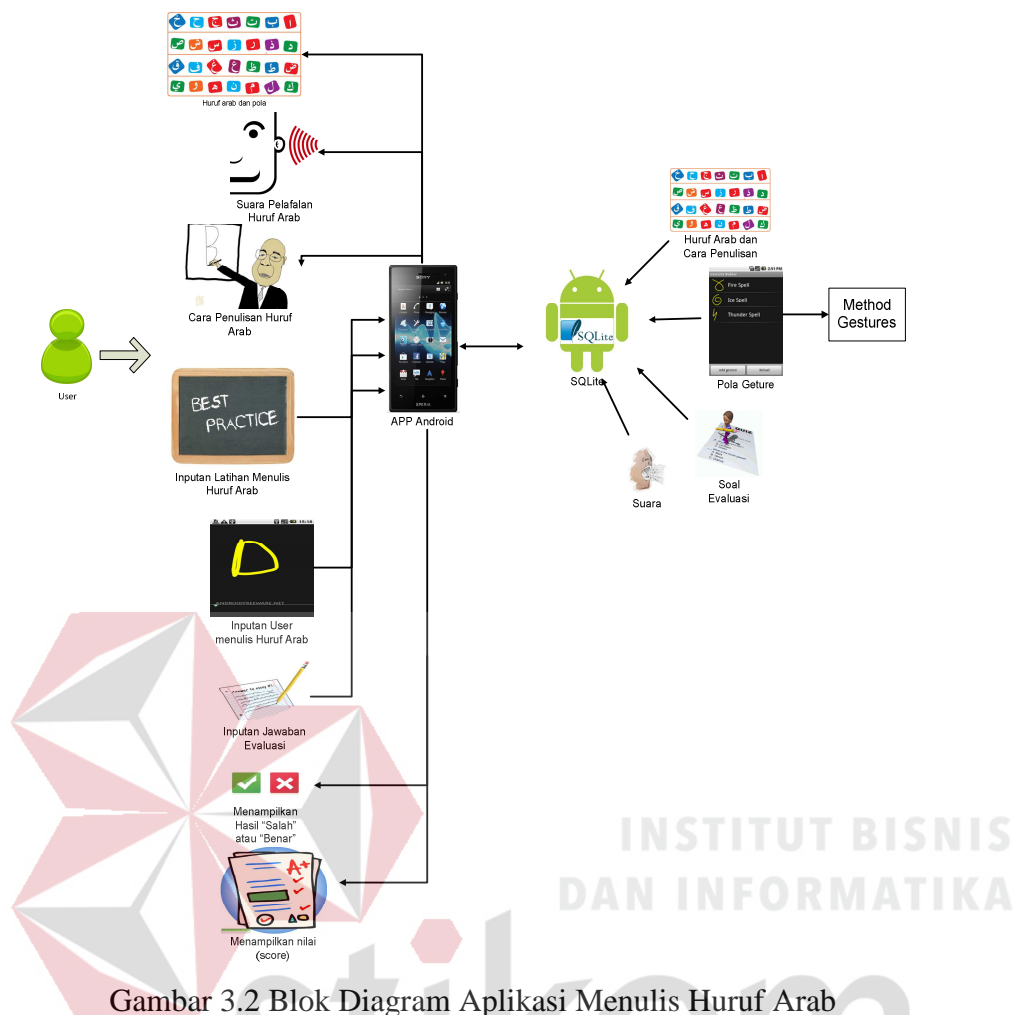
| Kebutuhan Non-Fuctional   |
|---|
| <ol style="list-style-type: none"> <li>1. Operasional <ol style="list-style-type: none"> <li>a. Lebih baik digunakan pada sistem operasi Android versi 4.1 (<i>Jelly Bean</i>)</li> <li>b. Spesifikasi CPU rekomendasi single qore</li> <li>c. Minimal RAM 512</li> <li>d. Kebutuhan <i>free spaces</i> memory internal 2 Gb</li> <li>e. <i>Handphone</i> Layar <i>touchscreen</i> sebagai media menulis.</li> <li>f. Lebih baik menggunakan layar 4 inc</li> </ol> </li> <li>2. Kinerja <ol style="list-style-type: none"> <li>a. Pencocokan <i>gestures</i> dengan menggunakan metode \$N</li> <li>b. Jumlah dalam memasukkan <i>gestures</i> sebanyak lima kali dalam latihan kuis</li> <li>c. Penyimpanan titik koordinat hasil dari <i>input</i> data <i>user</i> dan menampilkan kembali pada fitur nilai</li> </ol> </li> <li>3. Keamanan <ol style="list-style-type: none"> <li>a. Sistem aplikasi mengecek <i>userlogin</i> untuk masuk menu utama sistem</li> </ol> </li> <li>4. Info <ol style="list-style-type: none"> <li>a. Digunakan untuk menampilkan nilai dari hasil latihan kuis tes kemampuan maupun soal evaluasi</li> <li>b. Digunakan untuk menampilkan kembali histori nilai pada menu nilai dari latihan kuis yang telah dilakukan oleh <i>user</i></li> <li>c. Digunakan untuk menampilkan prosentase benar, jumlah menulis <i>user</i> dan salah dalam latihan menulis hijaiyah</li> </ol> </li> </ol> |

### 3.2 Perancangan Sistem

Dari analisis diatas, berikutnya adalah tahap perancangan sistem. Aplikasi menulis huruf arab yang akan dibangun. Berikut adalah gambaran rancangan dari aplikasi menulis huruf arab dengan menggunakan blok diagram seperti Gambar 3.2. Pada gambar tersebut menjelaskan tentang rancangan aplikasi menulis huruf arab dengan teknologi *gestures*. Pada aplikasi yang akan dibangun mempunyai beberapa fitur yaitu pengenalan huruf hijaiyah, latihan menulis hijaiyah, kuis yang terdapat dua macam yaitu kuis tes kemampuan dan kuis soal evaluasi, kemudian fitur yang terakhir adalah nilai.

Pada fitur pengenalan huruf hijiyah, *user* diberikan pengetahuan terlebih dahulu tentang huruf arab, mulai dari huruf alif sampai ya sejumlah 28 huruf arab. Pada fitur ini juga diberikan berupa bunyi pelafalan dari masing-masing 28 huruf arab. Selain itu, juga diberi pengetahuan tentang tata cara menulis huruf arab yang sesuai dengan kaidah (Mustofa, 2008), baik berupa gambar atau animasi bergerak, sehingga *user* dapat mengetahui secara jelas tata cara penulisan huruf arab yang benar.

Aplikasi yang akan dibuat juga terdapat fitur latihan menulis huruf hijaiyah yan berguna melatih *user* untuk mencoba menulis huruf hijaiyah yang sesuai dengan kaidah. Awalnya *user* harus memilih salah satu huruf dari 28 hijaiyah, kemudian *user* harus menuliskan huruf hijaiyah pada *canvas* yg telah disediakan oleh sistem. Setelah itu, sistem akan melakukan proses pengecekan *input gestures* oleh *user* dengan *gestures template*, sehingga sistem memberikan info nilai prosentase benar dan salah dan juga total dari tulisan *input gestures* oleh *user* pada latihan menulis hijaiyah.



Gambar 3.2 Blok Diagram Aplikasi Menulis Huruf Arab

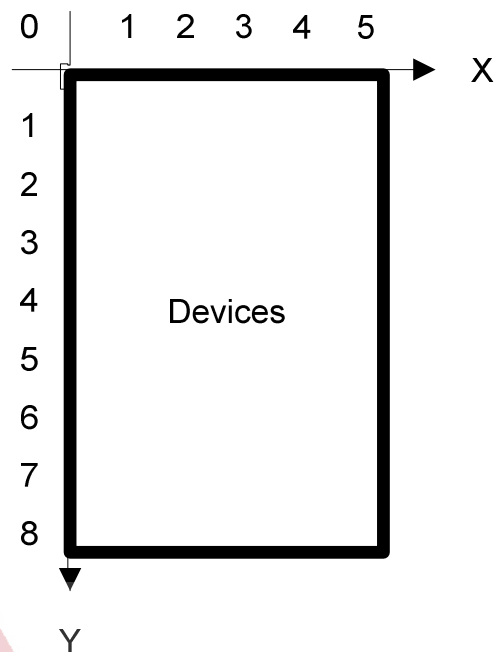
Pada fitur kuis terdapat dua macam, yaitu tes kemampuan dan soal evaluasi. Tes kemampuan memungkinkan *user* mengetahui pemahaman mengenai menulis huruf hijaiyah menurut kaidah yang benar. Awalnya *user* harus memilih salah satu huruf hijaiyah, kemudian *user* menuliskan sebanyak lima kali. Setelah itu, sistem akan mengeluarkan hasil nilai berupa prosentase keberhasilan menulis huruf hijaiyah. Pada fitur soal evaluasi, sistem akan memberikan soal dari 28 huruf hijaiyah secara acak, kemudian *user* harus menuliskan huruf hijaiyah sesuai dengan perintah sistem. Soal tersebut sebanyak lima kali, setelah itu sistem akan memberikan nilai salah dan benar dari ke lima soal tersebut kepada *user*.

Pada aplikasi terdapat fitur nilai. Fitur nilai adalah hasil histori dari *user* melakukan latihan fitur kuis tes kemampuan atau kuis soal evaluasi . Ketika user menggunakan salah satu fitur kuis, hasil akhirnya adalah nilai, dan nilai tersebut akan tersimpan pada *database* sebagai histori berdasarkan *user (iduser)*. Nilai yang disimpan adalah dua, yaitu nilai tes kemampuan dan nilai soal evaluasi. Fitur ini diharapkan dapat bermanfaat bagi *user* untuk mengetahui kemampuannya dalam menulis huruf hijaiyah, sehingga mengetahui perkembangan dalam menulis huruf arab berdasarkan nilai yang diperoleh.

Pada aplikasi menulis huruf arab, ada beberapa fitur yang menggunakan database sebagai media penyimpanan, seperti nama *user*, hasil nilai dari latihan *user* terhadap sistem, atau hasil tulisan dari *user* yang semua itu akan dijadikan *history user*. Untuk teknologi *gestures* pada Android menggunakan *event touch*, sebagai media *activity* yang dilakukan oleh *user* pada layar *mobile Android*, sedangkan dalam hal pencocokan pola huruf arab terhadap *event touch user*, menggunakan cara pencocokan terhadap titik koordinat. Berikut adalah contoh algoritma yang digunakan.

### **1. Layar Screen**

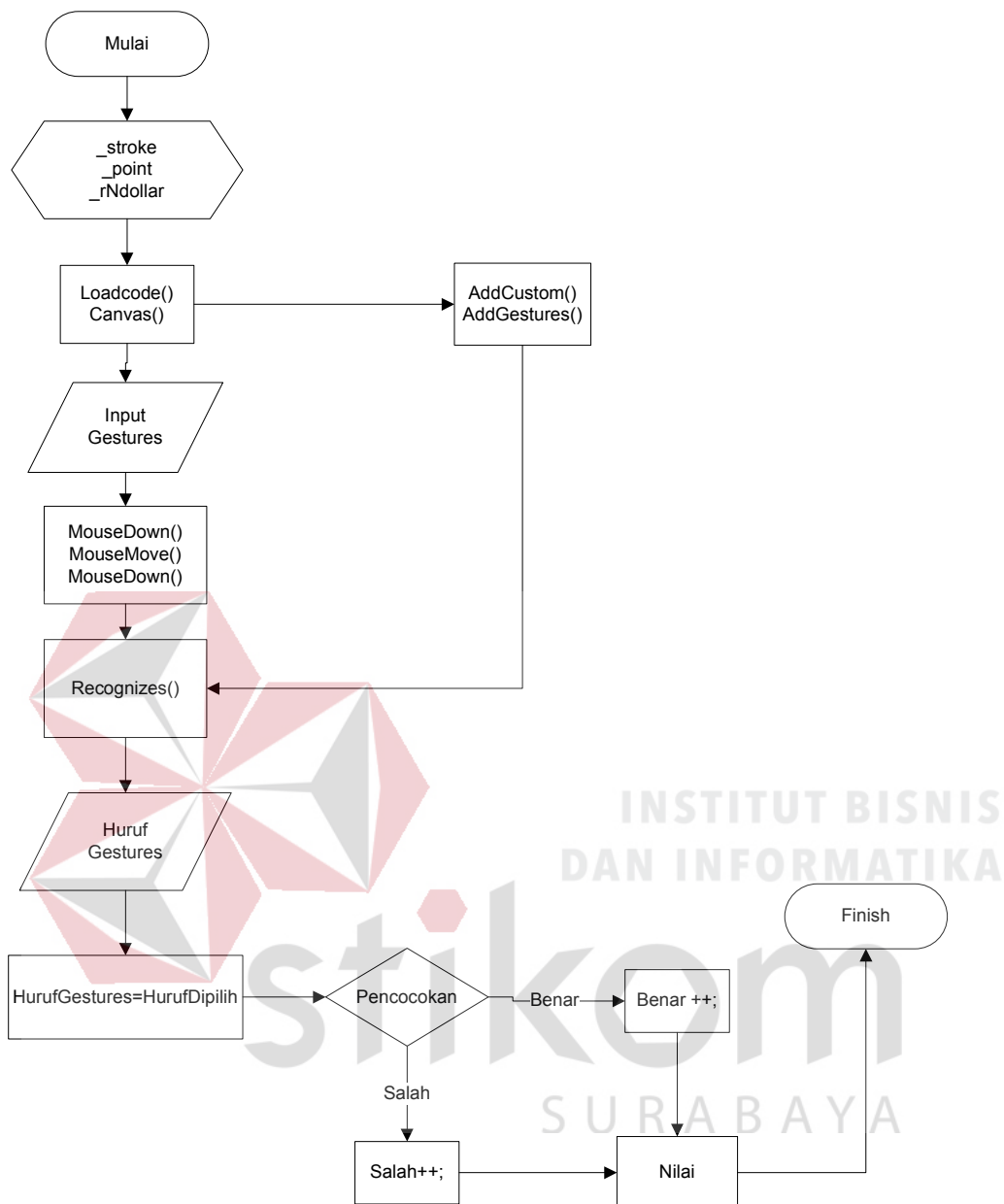
Pada *event touch*, mendeteksi titik koordinat *x* dan *y*. Penulis akan memanfaatkan titik koordinat tersebut dalam hal pencocokan huruf arab. Titik koordinat dari pola huruf arab disimpan pada sebuah *array* dan beberapa *variable* sebagai pendukung dalam pencocokan pola huruf arab yang diinginkan. Sumbu nol koordinat *x* dan *y* berada pada pojok kiri atas *devices*. Untuk lebih jelasnya seperti yang terlihat pada gambar 3.3.



Gambar 3.3 Contoh Koordinat (x,y) pada *device 4 inch*

## 2. Metode \$N

Metode \$N adalah sebuah *multistroke gesture recognizer* sederhana yang berbasis *template matching* yang berasal dari *the unistroke \$1*. *Multistroke recognizer* adalah desain *stroke recognizer 2-D* yang dirancang untuk prototipe basis *gesture* antarmuka. \$N dibuat berdasarkan \$1 *unistroke recognizer*. \$1 *unistroke* membandingkan *input gestures* dengan *gestures template* (*gestures* yang sudah ada). Hasil *gestures template* dan masukan data *gestures* ditentukan oleh nilai kedekatan rata-rata jarak *Euclidean* antara titik yang sesuai dengan dengan keduanya. *Gestures template* juga dapat memiliki penamaan atau istilah berdasarkan *index*, memungkinkan untuk meningkatkan tingkat fleksibilitas. Proses pencocokan metode \$N dapat dijelaskan pada *flow chart*, seperti pada Gambar 3.4.



Gambar 3.4 Flowchart Proses Pencocokan dengan Metode \$N

Pada proses awal Gambar 3.4, sistem membuat *variable* `_stroke`, `_point` dan `_rNdollar`. *Variable* `_stroke` digunakan untuk mengetahui jumlah `_stroke` dari *input gestures* dari *user*. Untuk *variable* `_point` adalah titik koordinat dari *input gestures* dari *user* kemudian *variable* `_rNdollar` adalah *variable* dalam mengambil fungsi dari metode \$N. Fungsi `loadcode()` adalah fungsi data koordinat yang



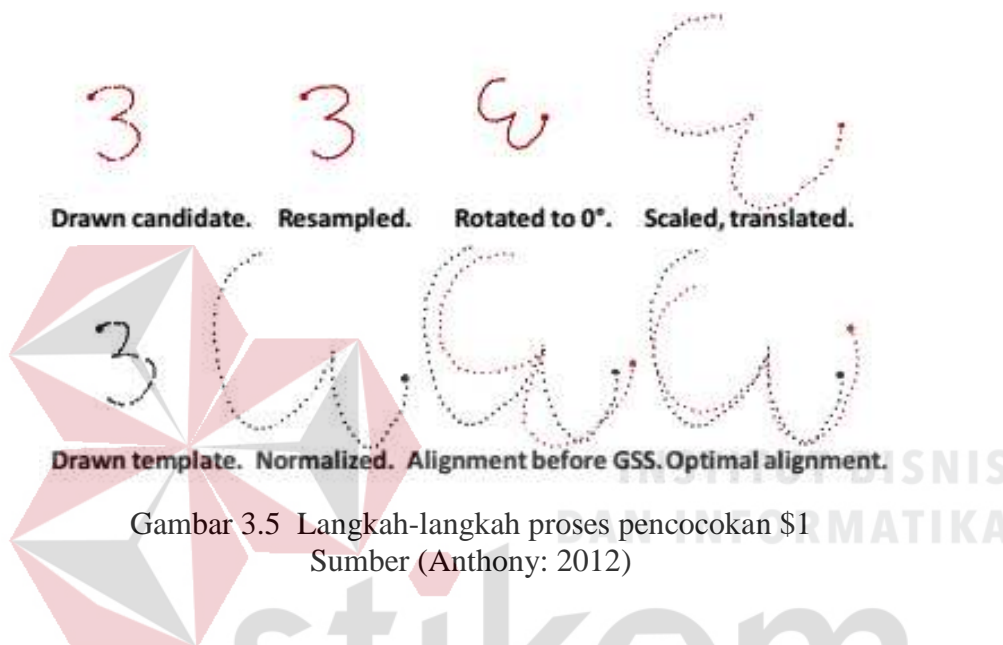
sudah ditentukan (*gestures template*) untuk menghitung *variable* yang dibutuhkan dalam pencocokan dengan *input gestures* dari *user*.

Ketika *user* memberikan *input gestures*, fungsi `MouseDown()`, `MouseMove()` dan `MouseUp` akan dijalankan. Fungsi `MouseDown` adalah fungsi mengenali titik koordinat pertama ketika *user* pertama kali menyentuh layar, kemudian dilanjutkan fungsi `MouseMove()` adalah pengenalan koordinat gerakan tangan ketika *user* bergerak atau berpindah (kondisi menyentuh layar). Fungsi terakhir `MouseUp()`, ketika gerakan tangan *user* tidak menyentuh layar, atau pengenalan titik koordinat pada kondisi terakhir tangan menyentuh layar kemudian dilepaskan.

Pada Fungsi `MouseUp()`, terdapat fungsi `GetResult()` yang dijalankan. Fungsi ini adalah fungsi dalam menampilkan nilai dengan metode pencocokan \$N, jadi pada fungsi `GetResult` terdapat fungsi `Recognizes` yang dijalankan sehingga mempunyai nilai *variable* yang nantinya akan dicocokkan dengan fungsi nilai *variable* yang ada pada fungsi `loadcode()` sehingga akan terdeteksi huruf apa yang dituliskan oleh *user*. Setelah itu terdapat fungsi `nilai()`, apabila huruf yang dipilih == huruf *gestures* maka akan bernilai benar.

Langkah penting dalam penyesuaian point pada *gestures template* dan *input gestures*, \$1 menggunakan empat langkah seperti yang digambarkan pada Gambar 3.5, pertama adalah spasial resample *strokes* yang telah ditentukan sehingga jumlah titik tersebar atau jarak tetap dengan perbandingan yang tepat (*equidistantly*) pada sepanjang *strokes*. Kedua perputaran *stroke* seperti sudut indikasi atau deteksi sudut, titik awal adalah pada  $0^{\circ}$ . Hal ini berfungsi sebagai sebuah pendekatan untuk suatu garis arah (*alignment*) kemudian digunakan untuk

mencari keselarasan sudut yang optimal atau sama. Ketiga adalah skala stroke yang tidak seragam (*non-uniformly*) untuk mencocokkan persegi referensi atau pusat titik. Keempat, menerjemahkan stroke sehingga pusatnya adalah dititik semula. Pada tahap ini menormalkan semua stroke sehingga setiap titik dalam *input gestures* sesuai spasial dengan satu titik dalam *gestures template*.



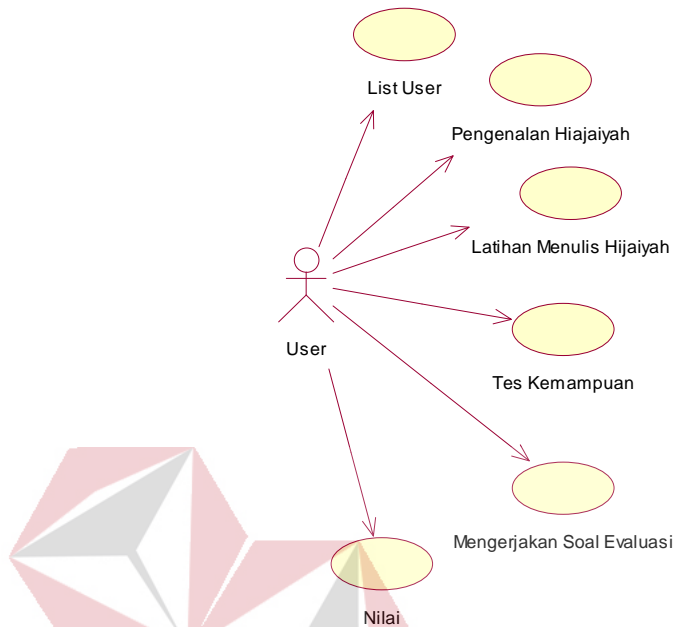
Gambar 3.5 Langkah-langkah proses pencocokan \$1  
Sumber (Anthony: 2012)

Perbedaan \$1 dengan \$N adalah \$1 hanya bisa mendeteksi satu *strokes* saja, kalau \$N bisa mendeteksi lebih dari satu *strokes*. \$N adalah basis turunan terdekat dari classifier berdasarkan dengan fungsi penilaian *Euclidean*, yaitu *geometris template matcher*. Pada Gambar 3.5 adalah gambaran model pencocokan *input gestures* dengan *gestures template*.

### 3.2.1 Use Case Diagram Aplikasi Menulis Huruf Arab

Pada Gambar 3.6 terdapat *use case diagram* Aplikasi Menulis Huruf Arab. Pada *use case diagram* aplikasi menulis huruf arab, hanya mempunyai satu aktor yaitu *user*. Aktor *user* nantinya bisa melakukan empat proses yaitu pilih *user* yang

ada atau membuat *user* baru, pengenalan huruf arab, latihan menulis huruf arab dan mengerjakan soal evaluasi.



Gambar 3.6 Use Case Diagram Menulis Huruf Arab

*User* yang belum membuat *user* pada aplikasi menulis huruf arab dapat membuat *user* baru, tetapi bagi *user* yang telah memiliki *user* dapat langsung memilih *user* tersebut. Fungsi *user* adalah sebagai acuan dalam pengembangan pembelajaran menulis huruf arab, karena di dalam *user* memiliki histori nilai. Kemudian pada aplikasi, terdapat menu pengenalan huruf arab. Bagi *user* awam atau yang belum ada pengetahuan sama sekali tentang huruf arab dan tata cara penulisan yang benar dalam menulis huruf arab, maka pada menu ini *user* diberikan tentang pengetahuan tentang huruf arab beserta tata cara penulisan huruf arab sesuai dengan kaidah, baik huruf tunggal ataupun dengan harokat.

Apabila *user* telah faham tentang tata cara penulisan huruf arab, dapat beralih ke menu latihan menulis huruf arab. Pada menu ini, sistem akan memberi

perintah, kemudian *user* menuliskan huruf arab yang sesuai dengan perintah sistem. Hasil tulisan *user* akan divalidasi oleh sistem, setelah itu sistem akan memberikan *output* kepada *user* benar atau tidaknya *input user*.

Pada aplikasi ini, juga terdapat fitur yang berfungsi untuk menguji kemampuan *user*. Model pembelajarannya hampir sama dengan fitur latihan menulis huruf arab, tetapi pada menu ini sistem dapat memberikan *output* berupa nilai pada *user*, sehingga dapat mengetahui tingkat kemampuannya dalam menulis huruf arab. *User* juga bisa tes kemampuan dalam menulis huruf arab, dengan cara *user* memilih huruf arab kemudian *user* akan menuliskan sebanyak lima kali, dari semua percobaan akan dihitung nilai dari semua *input user*.

### 3.2.2 Flow of Events Use Case List User

Berikut ini disajikan *flow of events* dari *use case* diagram aktifitas *list user*. *Flow of events* bertujuan untuk mendokumentasikan alur logika dalam *use case* yang menjelaskan secara rinci apa yang pemakai akan lakukan dan apa yang sistem itu sendiri lakukan. *Flow of events login* disajikan pada tabel 3.3 dengan kondisi akhir yang diinginkan dan kondisi akhir gagal, serta alur alternatif untuk menangani kondisi salah. Proses *list user* dilakukan untuk memulai menjalankan aplikasi menulis huruf arab.

Tabel 3.3 Flow of event list user

| Nama Use Case               | Login User   |
|-----------------------------|--|
| <b>Kebutuhan terkait</b>    | Salah satu proses untuk mengawali dalam menjalankan aplikasi menulis huruf arab adalah menu <i>list user</i> . |
| <b>Tujuan</b>               | Membuat <i>user login</i> untuk menangani <i>history</i> nilai.  |
| <b>Prasyarat</b>            | Harus mempunyai <i>user login</i> .  |
| <b>Kondisi akhir sukses</b> | <i>User</i> yang berhasil <i>login</i> dapat mengakses menu-menu utama dari aplikasi menulis huruf arab.       |
| <b>Kondisi akhir gagal</b>  | Tidak ada  |

| Nama Use Case  | Login User  |   |
|----------------|---|---|
| Aktor utama    | User.   |   |
| Aktor sekunder | Tidak ada   |   |
| Pemicu         | Aktor memilih menu “Lanjut” atau Buat Baru untuk dapat mengakses seluruh menu pada aplikasi menulis huruf arab. |   |
| Alur utama     | <b>Langkah</b>  | <b>Aksi</b>   |
|                | 1.  | Membuka halaman pembuka pada aplikasi menulis huruf arab.                         |
|                | 2.  | Sistem menampilkan halaman <i>login</i> .   |
|                | 3.  | Memilih menu Buat Baru bila <i>user</i> belum mempunyai <i>user login</i> .       |
|                | 4.  | Memilih menu Lanjut bila <i>user</i> sudah memiliki <i>user login</i> .           |
|                | 5.  | Sistem melakukan validasi <i>input login</i> kemudian mengeksekusi ke menu utama. |
|                | 6.  | Mengambil data <i>user</i> sesuai nama <i>user</i> oleh <i>user</i> .             |
| Alur Perluasan | <b>Langkah</b>  | <b>Aksi Percabangan</b>   |
|                | 3.1.  | <i>User</i> menginputkan nama pada <i>textbox</i> yang telah disediakan.          |
|                | 3.2   | Memilih Mulai bila <i>user</i> ingin melanjutkan ke menu utama.                   |
|                | 4.1   | Sistem menampilkan nama <i>user</i> yang telah tersimpan sebelumnya.              |
|                | 4.2   | <i>User</i> memilih <i>id user login</i> .  |

### 3.2.3 Flow of Events Use Case Pengenalan Hijaiyah

Berikut ini disajikan *flow of events* dari *use case* diagram aktifitas pengenalan huruf arab. *Flow of events* bertujuan untuk mendokumentasikan alur logika dalam *use case* yang menjelaskan secara rinci apa yang pemakai akan lakukan dan apa yang sistem itu sendiri lakukan. *Flow of events* pengenalan huruf arab disajikan pada tabel 3.4 dengan kondisi akhir yang diinginkan dan kondisi akhir gagal, serta alur alternatif untuk menangani kondisi salah.

Proses pengenalan huruf arab dilakukan agar *user* mendapat pengetahuan tentang tata cara menulis huruf arab.

Tabel 3.4 *Flow of event* Pengenalan Hijaiyah

| Nama Use Case               | Pengenalan Hijaiyah  |   |
|-----------------------------|--|---|
| <b>Kebutuhan terkait</b>    | Salah satu proses untuk memberikan pengetahuan tentang tata cara menulis huruf hijaiyah yang sesuai dengan kaidah. |   |
| <b>Tujuan</b>               | <i>User</i> dapat mengetahui tata cara menulis huruf hijaiyah yang sesuai dengan kaidah.                           |   |
| <b>Prasyarat</b>            | Tidak ada  |   |
| <b>Kondisi akhir sukses</b> | <i>User</i> dapat melihat gambaran atau tata cara menulis huruf arab yang sesuai dengan kaidah.                    |   |
| <b>Kondisi akhir gagal</b>  | Tidak ada  |   |
| <b>Aktor utama</b>          | <i>User</i> .  |   |
| <b>Aktor sekunder</b>       | Tidak ada  |   |
| <b>Pemicu</b>               | Aktor memilih menu “pengenalan huruf hijaiyah”, kemudian memilih huruf hijaiyah yang akan dipelajari.              |   |
| <b>Alur utama</b>           | <b>Langkah</b>   | <b>Aksi</b>   |
|                             | 1.   | <i>User</i> memilih menu pengenalan huruf hijaiyah.   |
|                             | 2.   | Sistem menampilkan tata cara penggunaan pada menu pengenalan huruf hijaiyah.  |
|                             | 3.   | <i>User</i> memilih salah satu huruf hijaiyah/ arab.  |
| <b>Alur Perluasan</b>       | <b>Langkah</b>   | <b>Aksi Percabangan</b>   |
|                             | 4.1.   | Tombol <i>play</i> berfungsi untuk menampilkan cara penulisan huruf hijaiyah secara bergerak.                                   |
|                             | 4.2  | Tombol <i>stop</i> berfungsi untuk kembali pada menu awal pengenalan huruf hijaiyah atau menghentikan animasi tulisan bergerak. |
|                             | 4.3  | Menu suara untuk mengetahui bunyi lafal dari masing-masing huruf arab.  |

### 3.2.4 Flow of events Use case Latihan Menulis Hijaiyah

Berikut ini disajikan *flow of events* dari *use case* diagram aktifitas latihan menulis hijaiyah. *Flow of events* bertujuan untuk mendokumentasikan alur logika dalam *use case* yang menjelaskan secara rinci apa yang pemakai akan lakukan dan apa yang sistem itu sendiri lakukan. *Flow of events* latihan menulis hijaiyah disajikan pada tabel 3.5 dengan kondisi akhir yang diinginkan dan kondisi akhir gagal, serta alur alternatif untuk menangani kondisi salah.

Proses latihan menulis huruf arab dilakukan untuk melatih *user* dalam memahami penulisan huruf arab, karena sistem juga akan memberikan *output* nilai

Tabel 3.5 Flow of event Latihan Menulis Huruf Arab

| Nama Use Case               | Latihan Menulis Hijaiyah  |  |
|-----------------------------|---|--|
| <b>Kebutuhan terkait</b>    | Salah satu proses untuk melatih <i>user</i> dalam menulis huruf hijaiyah.   |  |
| <b>Tujuan</b>               | <i>User</i> dapat mengetahui kemampuannya dalam memahami menulis hijaiyah.  |  |
| <b>Prasyarat</b>            | <i>User</i> memasukkan data tulisan huruf hijaiyah sesuai dengan pilihan huruf hijaiyah yang telah dipilih oleh <i>user</i> . |  |
| <b>Kondisi akhir sukses</b> | <i>User</i> dapat melihat prosentase nilai dari latihan menulis huruf hijaiyah.   |  |
| <b>Kondisi akhir gagal</b>  | <i>Input user</i> salah sehingga mempunyai nilai 0%   |  |
| <b>Aktor utama</b>          | <i>User</i> .   |  |
| <b>Aktor sekunder</b>       | Tidak ada   |  |
| <b>Pemicu</b>               | Aktor memilih menu “latihan menulis hijaiyah”.  |  |
| <b>Alur utama</b>           | <b>Langkah</b>  | <b>Aksi</b>  |
|                             | 1.  | <i>User</i> memilih menu latihan menulis huruf hijaiyah.                                   |
|                             | 2.  | Sistem menampilkan tata cara penggunaan aplikasi pada menu latihan menulis huruf hijaiyah. |
|                             | 3.  | Sistem menampilkan 28 huruf hijaiyah.  |
|                             | 4.  | <i>User</i> memilih salah satu huruf hijaiyah yang diinginkan.                             |
|                             | 5.  | <i>User</i> menuliskan hijaiyah sesuai dengan huruf hijaiyah yang telah dikehendaki.       |
|                             | 6.  | Sistem mengecek <i>input user</i> .  |
|                             | 7.  | Sistem memberikan <i>output</i> nilai dari hasil   |



| Nama Use Case  | Latihan Menulis Hijaiyah |                         |
|----------------|--------------------------|-------------------------|
|                |                          | latihan <i>user</i> .   |
| Alur Perluasan | <b>Langkah</b>           | <b>Aksi Percabangan</b> |
|                |                          | Tidak ada               |

### 3.2.5 Flow of events Use case Tes Kemampuan

Berikut ini disajikan *flow of events* dari *use case* diagram aktifitas tes kemampuan. *Flow of events* bertujuan untuk mendokumentasikan alur logika dalam *use case* yang menjelaskan secara rinci apa yang pemakai akan lakukan dan apa yang sistem itu sendiri lakukan. *Flow of events* tes kemampuan disajikan pada Tabel 3.6 dengan kondisi akhir yang diinginkan dan kondisi akhir gagal, serta alur alternatif untuk menangani kondisi salah.

Proses tes kemampuan dilakukan untuk menguji kemampuan *user* dalam memahami penulisan huruf arab. Awalnya *user* memilih salah satu huruf hijaiyah, kemudian *user* harus menuliskan huruf hijaiyah sesuai dengan pilihannya sebanyak lima kali. Pada akhirnya sistem akan memberikan *output* nilai berupa prosentase.

Tabel 3.6 *Flow of event* Tes Kemampuan

| Nama Use Case               | Tes Kemampuan  |             |
|-----------------------------|--|-------------|
| <b>Kebutuhan terkait</b>    | Salah satu proses untuk melatih kemampuan <i>user</i> dalam menulis huruf arab.  |             |
| <b>Tujuan</b>               | <i>User</i> dapat mengetahui kemampuannya dalam memahami menulis huruf arab.   |             |
| <b>Prasyarat</b>            | <i>User</i> memasukkan data tulisan huruf hijaiyah sesuai dengan pilihan huruf hijaiyah yang telah dipilih oleh <i>user</i> .. |             |
| <b>Kondisi akhir sukses</b> | Bila <i>user</i> memberikan jawaban benar maka akan mempunyai nilai untuk setiap <i>input</i> .                                |             |
| <b>Kondisi akhir gagal</b>  | <i>Input user</i> salah sehingga mempunyai nilai 0   |             |
| <b>Aktor utama</b>          | <i>User</i> .  |             |
| <b>Aktor sekunder</b>       | Tidak ada  |             |
| <b>Pemicu</b>               | Aktor memilih menu “tes kemampuan”.  |             |
| <b>Alur utama</b>           | <b>Langkah</b>   | <b>Aksi</b> |

| Nama <i>Use Case</i>  | Tes Kemampuan  |  |
|-----------------------|----------------|--|
|                       | 1.             | <i>User</i> memilih menu tes kemampuan.                              |
|                       | 2.             | Sistem menampilkan 28 huruf hijaiyah.                                |
|                       | 3.             | <i>User</i> memilih salah satu huruf hijaiyah.                       |
|                       | 4.             | <i>User</i> menuliskan arab sesuai pilihannya.                       |
|                       | 5.             | Sistem mengecek <i>input user</i> .                                  |
|                       | 6.             | Sistem memberikan <i>output</i> nilai dari hasil <i>input user</i> . |
| <b>Alur Perluasan</b> | <b>Langkah</b> | <b>Aksi Percabangan</b>  |
|                       |                | Tidak ada  |

### 3.2.6 *Flow of events Use case Mengerjakan Soal Evaluasi*

Berikut ini disajikan *flow of events* dari *use case* diagram aktifitas mengerjakan soal evaluasi. *Flow of events* bertujuan untuk mendokumentasikan alur logika dalam *use case* yang menjelaskan secara rinci apa yang pemakai akan lakukan dan apa yang sistem itu sendiri lakukan. *Flow of events* mengerjakan soal evaluasi disajikan pada tabel 3.7 dengan kondisi akhir yang diinginkan dan kondisi akhir gagal, serta alur alternatif untuk menangani kondisi salah.

Proses latihan mengerjakan soal evaluasi dilakukan untuk menguji kemampuan *user* dalam memahami penulisan huruf arab, karena sistem *random* soal sehingga *user* diuji ingatannya. Sistem juga akan memberikan *output* nilai.

Tabel 3.7 *Flow of event* Mengerjakan Soal Evaluasi

| Nama <i>Use Case</i>        | Mengerjakan Soal Evaluasi   |
|-----------------------------|---|
| <b>Kebutuhan terkait</b>    | Salah satu proses untuk melatih kemampuan <i>user</i> dalam mengingat dan menulis huruf hijaiyah. |
| <b>Tujuan</b>               | <i>User</i> dapat mengetahui kemampuannya dalam memahami menulis huruf hijaiyah.                  |
| <b>Prasyarat</b>            | <i>User</i> memasukkan data tulisan huruf hijaiyah sesuai dengan perintah sistem.                 |
| <b>Kondisi akhir sukses</b> | Bila <i>input user</i> benar, maka untuk setiap tulisan mempunyai nilai tidak sama dengan 0.      |
| <b>Kondisi akhir gagal</b>  | <i>Input user</i> salah sehingga mempunyai nilai 0  |
| <b>Aktor utama</b>          | <i>User</i> .   |

| Nama Use Case  | Mengerjakan Soal Evaluasi                       |  |
|----------------|---|--|
| Aktor sekunder | Tidak ada                                       |  |
| Pemicu         | Aktor memilih menu “mengerjakan soal evaluasi”. |  |
| Alur utama     | <b>Langkah</b>                                  | <b>Aksi</b>  |
|                | 1.  | User memilih menu soal evaluasi.                                       |
|                | 2.  | Sistem memberikan soal secara random                                   |
|                | 3.  | User menuliskan arab sesuai dengan soal yang diberikan oleh sistem.    |
| Alur Perluasan | <b>Langkah</b>                                  | <b>Aksi Percabangan</b>  |
|                | 3.1.  | Sistem mengecek <i>input user</i> .                                    |
|                | 3.2   | Sistem memberikan <i>output</i> nilai dari hasil latihan <i>user</i> . |

### 3.2.7 Flow of events Use case Nilai

Berikut ini disajikan *flow of events* dari *use case* diagram aktifitas nilai. *Flow of events* bertujuan untuk mendokumentasikan alur logika dalam *use case* yang menjelaskan secara rinci apa yang pemakai akan lakukan dan apa yang sistem itu sendiri lakukan. *Flow of events* nilai disajikan pada tabel 3.8 dengan kondisi akhir yang diinginkan dan kondisi akhir gagal, serta alur alternatif untuk menangani kondisi salah.

Pada fitur nilai *user* dapat mengetahui rekapan nilai yang diperolehnya selama latihan dalam menulis huruf hijaiyah.

Tabel 3.8 *Flow of event* Nilai

| Nama Use Case               | Nilai  |                                 |
|-----------------------------|--|---------------------------------|
| <b>Kebutuhan terkait</b>    | Proses untuk menyimpan rekapan nilai dari latihan dan tes kemampuan <i>user</i> .            |                                 |
| <b>Tujuan</b>               | <i>User</i> dapat menampilkan rekapan nilai.   |                                 |
| <b>Prasyarat</b>            | <i>User</i> memasukkan data tulisan huruf hijaiyah.  |                                 |
| <b>Kondisi akhir sukses</b> | Bila <i>input user</i> benar, maka untuk setiap tulisan mempunyai nilai tidak sama dengan 0. |                                 |
| <b>Kondisi akhir gagal</b>  | <i>Input user</i> salah sehingga mempunyai nilai 0   |                                 |
| <b>Aktor utama</b>          | <i>User</i> .  |                                 |
| <b>Aktor sekunder</b>       | Tidak ada  |                                 |
| <b>Pemicu</b>               | Aktor memilih menu “Nilai”.  |                                 |
| Alur utama                  | <b>Langkah</b>   | <b>Aksi</b>                     |
|                             | 1.   | <i>User</i> memilih menu nilai. |

| Nama <i>Use Case</i>  | Nilai          |   |
|-----------------------|----------------|---|
|                       | 2.             | <i>User</i> memilih nilai yang ingin ditampilkan                                  |
|                       | 3.             | Sistem memberikan <i>output</i> nilai sesuai pilihan <i>user</i> .                |
|                       | 4.             | Sistem menampilkan <i>gestures</i> dari <i>input user</i>                         |
| <b>Alur Perluasan</b> | <b>Langkah</b> | <b>Aksi Percabangan</b>   |
|                       | 2.1.           | Sistem memberikan pilihan dua menu, nilai tes kemampuan atau nilai soal evaluasi. |

### 3.2.8 *Activity Diagram* Aplikasi Menulis Huruf Arab

Dari *use case* yang ada, dibutuhkan *activity diagram* untuk menjelaskan proses/aliran proses yang terjadi pada tiap *use case*. *Activity diagram* akan dijelaskan dibawah ini.

#### A. *Activity Diagram List User*

Berikut adalah penjelasan dari *activity diagram list user* seperti yang dijelaskan pada Gambar 3.7. Pada gambar tersebut dijelaskan ketika *user* ingin melihat *list* atau daftar *user* yang akan diproses oleh sistem, jika *user* belum pernah membuat id *user* maka sistem akan meminta untuk menambah *user*, maka *user* baru bisa menggunakan aplikasi.

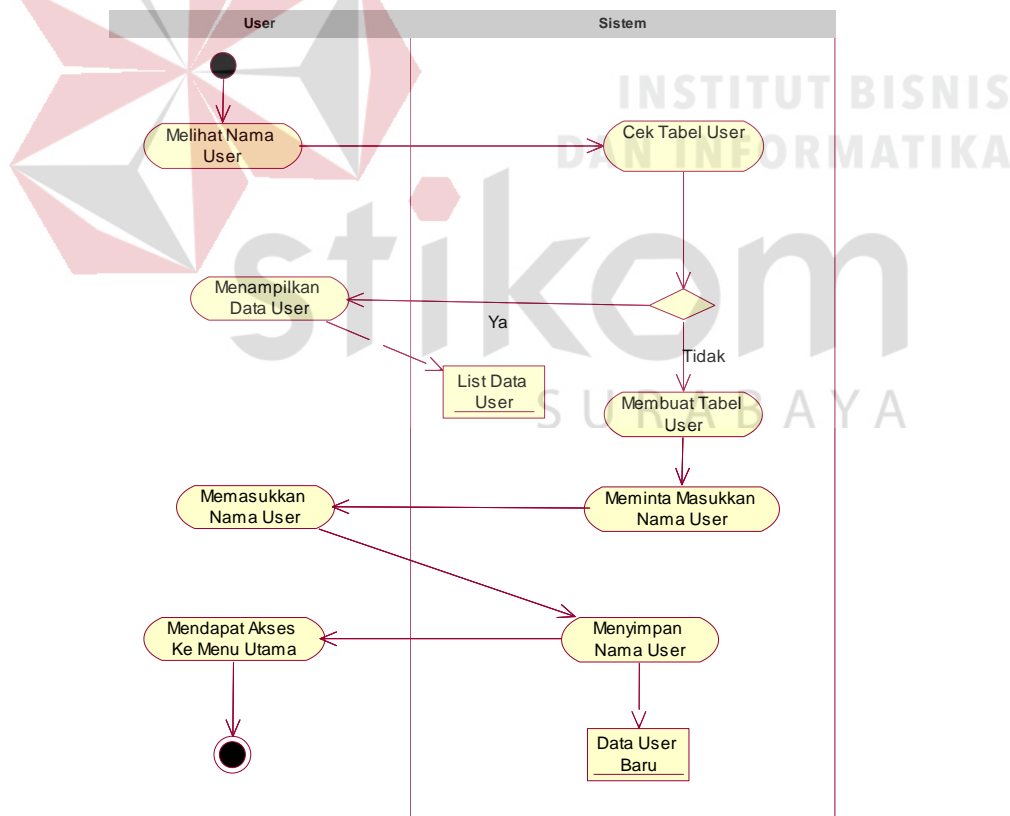
#### B. *Activity Diagram Pengenalan Hijaiyah*

Berikut adalah penjelasan dari *activity diagram* pengenalan hijaiyah seperti yang dijelaskan pada Gambar 3.8. Pada gambar tersebut dijelaskan ketika *user* ingin melihat tata cara penulisan huruf hijaiyah atau mengenal tentang huruf hijaiyah yang ada pada menu pengenalan huruf hijaiyah. Pada menu ini terdapat fitur *play* untuk menampilkan tata cara penulisan secara visual atau gambar, fitur *play* untuk melihat gambar secara bergerak. Fitur *stop* bila *user* ingin melihat

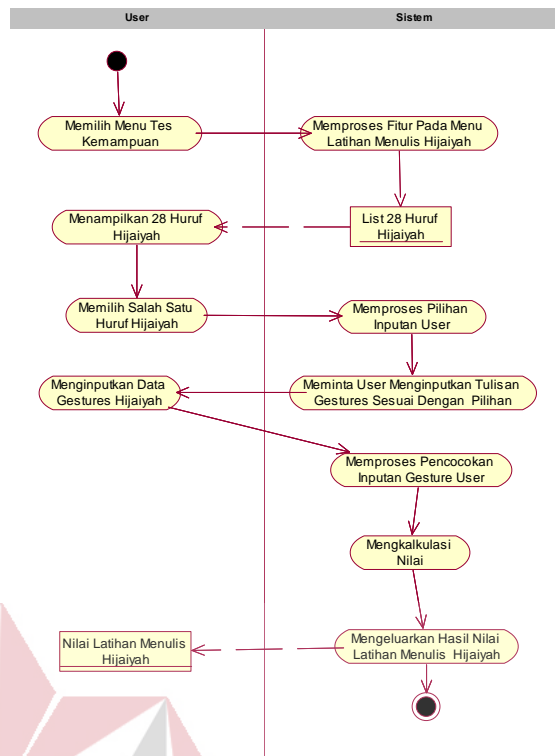
gambar awal (gambar tidak bergerak), kemudian fitur suara untuk mendengarkan pelafalan huruf hijaiyah berupa suara.

### C. *Activity Diagram Latihan Menulis Hijaiyah*

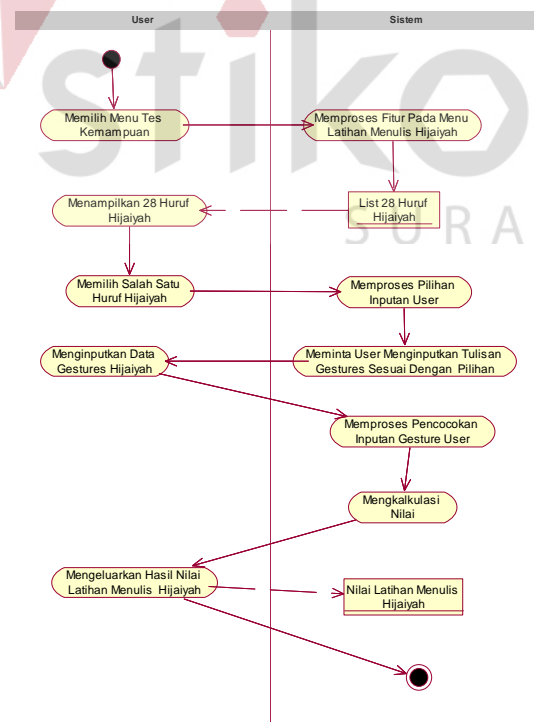
Berikut adalah penjelasan dari *activity diagram* latihan menulis hijaiyah seperti yang dijelaskan pada Gambar 3.9. Pada gambar tersebut dijelaskan proses ketika *user* melakukan tahap latihan menulis hijaiyah. Pada awalnya *user* memiliki salah satu huruf hijaiyah yang ingin dibuat latihan menulis, setelah itu sistem memberikan sebuah *canvas* yang nantinya user dapat menulis huruf hijaiyah sesuai dengan pilihan sebelumnya. Sistem mengeluarkan *output* nilai berupa prosentase benar dari jumlah tulisan.



Gambar 3.7 Activity Diagram List User



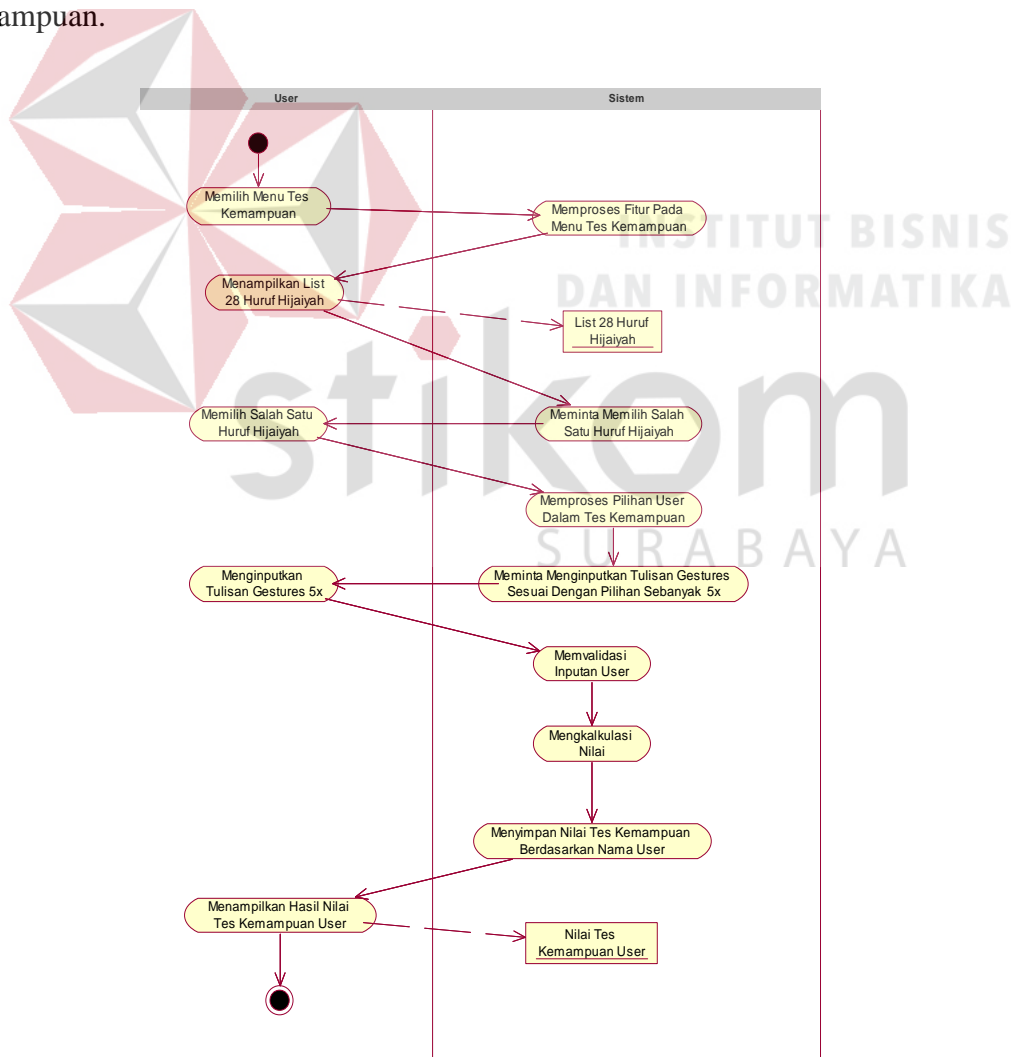
Gambar 3.8 Activity Diagram Pengenalan Hijaiyah



Gambar 3.9 Activity Diagram Latihan Menulis Hijaiyah

#### D. Activity Diagram Tes Kemampuan

Berikut adalah penjelasan dari *activity diagram* latihan menulis hijaiyah seperti yang dijelaskan pada Gambar 3.10. Pada gambar tersebut dijelaskan *user* dalam melakukan tahap tes kemampuan. Pada menu ini *user* memilih salah satu huruf hijaiyah, kemudian sistem menyediakan *canvas* yang nantinya *user* memasukkan *gestures* tulisan hijaiyah. Pada fitur ini *user* harus memasukkan tulisan *gestures* sebanyak lima kali, kemudian sistem akan memproses pencocokan pola yang ditulis oleh *user* dan sistem akan mengeluarkan nilai tes kemampuan.

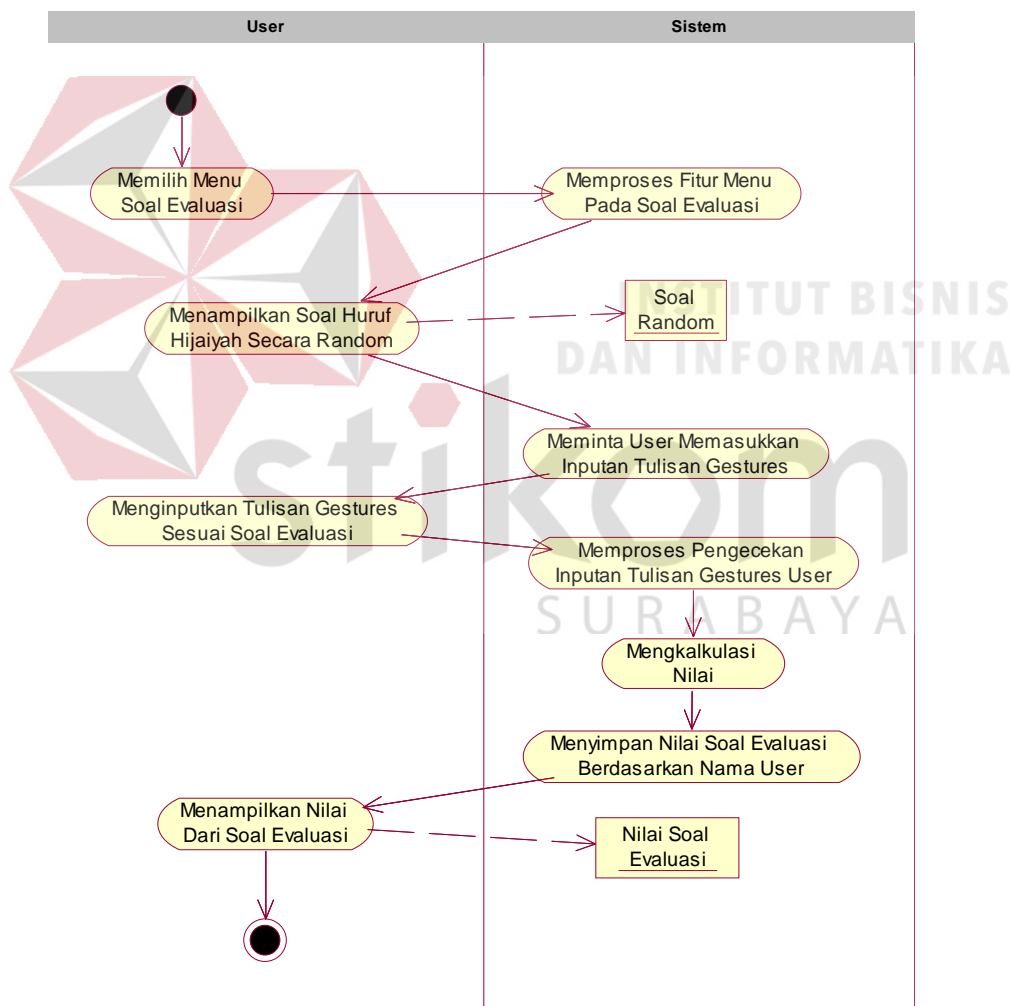


Gambar 3.10 Activity Diagram Tes Kemampuan



### E. Activity Diagram Soal Evaluasi

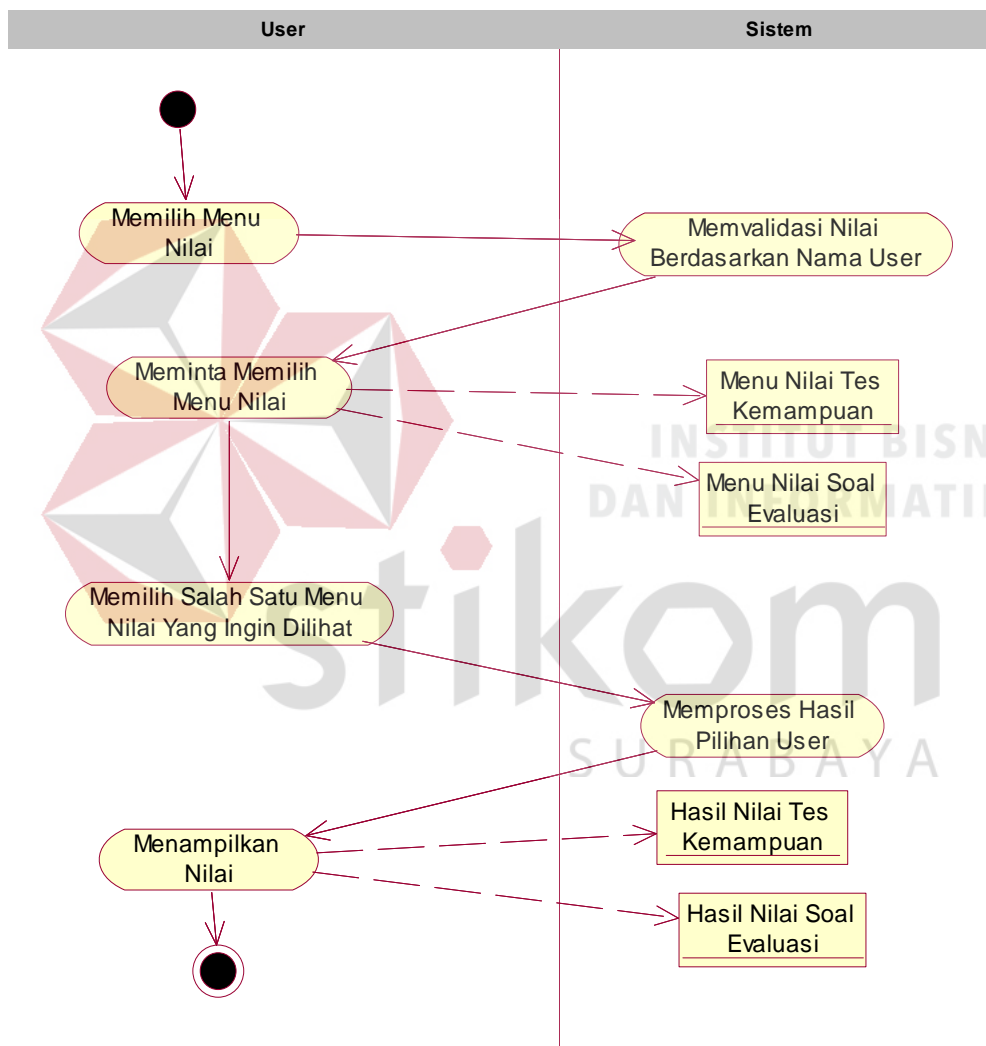
Berikut adalah penjelasan dari *activity diagram* latihan menulis hijaiyah seperti yang dijelaskan pada Gambar 3.11. Pada gambar tersebut dijelaskan fitur ini untuk menguji ingatan *user* dalam menghafal huruf hijaiyah. Pada fitur ini, sistem memberikan soal huruf hijaiyah secara acak sebanyak lima kali kepada *user*, kemudian *user* harus menuliskan huruf sesuai pertanyaan yang diberikan oleh sistem. Hasil akhirnya sistem akan memberikan sebuah nilai kepada *user*.



Gambar 3.11 Activity Diagram Soal Evaluasi

### F. Activity Diagram Nilai

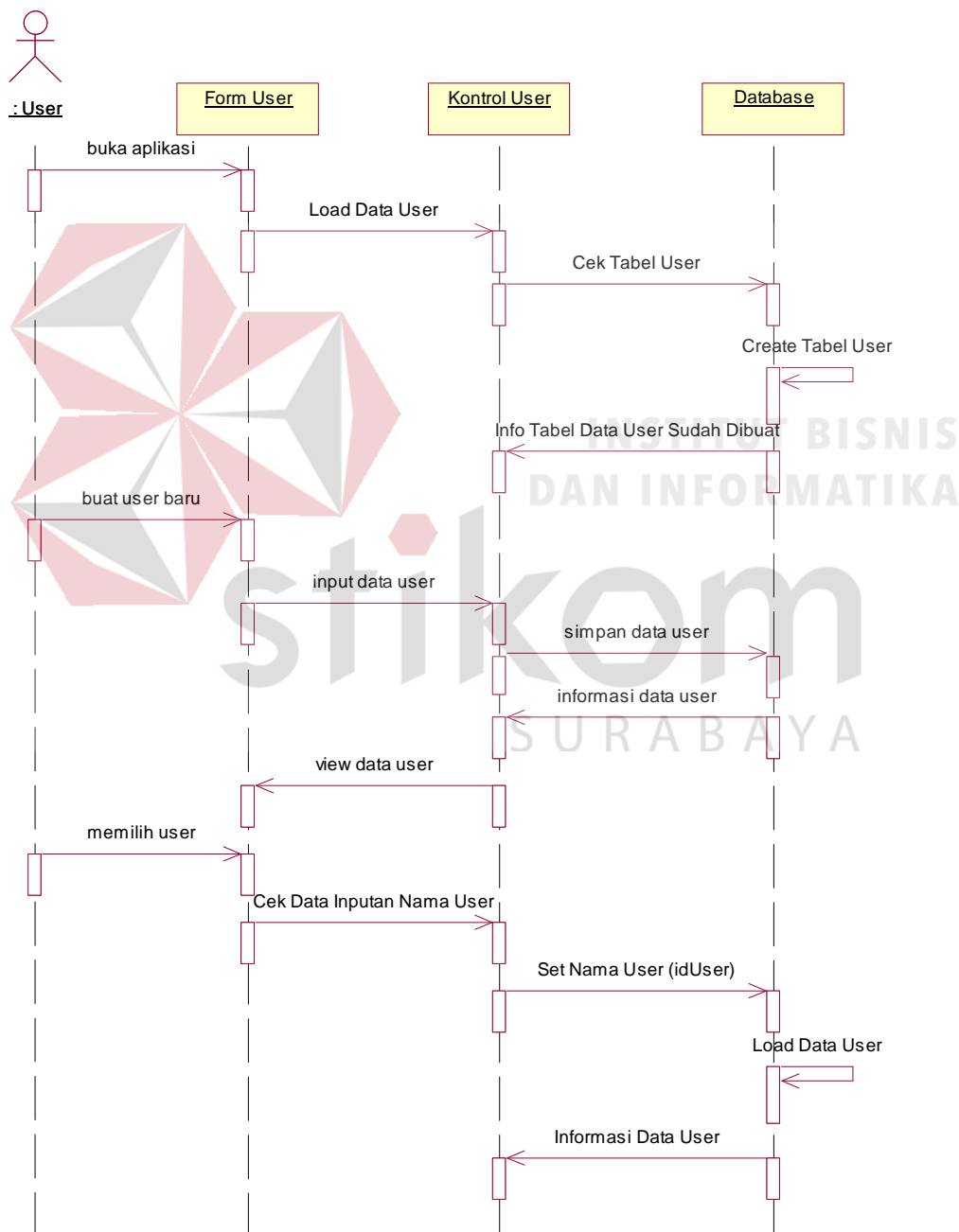
Berikut adalah penjelasan dari *activity diagram* latihan menulis hijaiyah seperti yang dijelaskan pada Gambar 3.12. Pada gambar tersebut adalah fitur melihat nilai dari hasil tes kuis *user* atau hasil histori nilai *user*. Dari nilai tersebut, *user* dapat melihat perkembangan dalam memahami menulis huruf hijaiyah.



Gambar 3.12 Activity Diagram Nilai

### 3.2.9 Sequence Diagram List User

*Sequence diagram* adalah *diagram* interaksi yang disusun berdasarkan urutan waktu. Diagram sekuensial mempresentasikan satu *flow* dari beberapa *flow* di dalam *use case*. Berikut adalah gambaran diagram sekuensial dari proses *List User*.

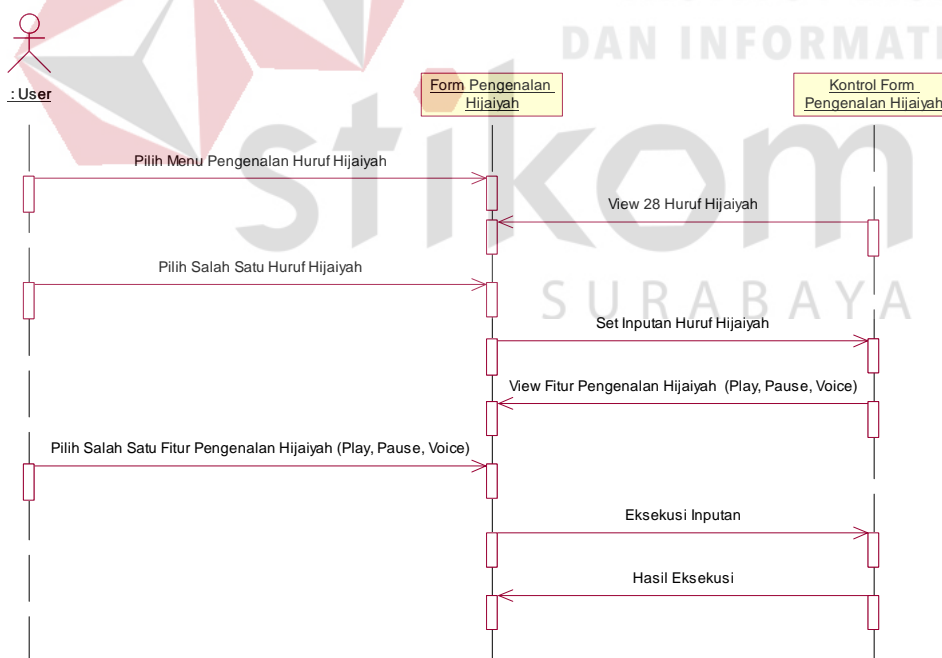


Gambar 3.13 Sequence Diagram List User

Pada Gambar 3.13 *Sequence diagram list user* menjelaskan proses-prosesnya. Ketika *user* membuka aplikasi, maka akan tampil *form user*. Bila *user* belum pernah membuat id baru, maka *user* harus membuat id baru terlebih dahulu dan sebaliknya jika sudah pernah membuat, *user* hanya memilih nama yang telah ada yang nantinya sistem akan menampilkan informasi data *user* atau *history user*.

### 3.2.10 Sequence Diagram Pengenalan Hijaiyah

Pada Gambar 3.14 adalah *Sequence diagram* pengenalan hijaiyah. pada fitur ini bisa dikatakan tutorial, karena pada fitur ini menunjukkan tata cara menulis huruf arab yang sesuai dengan kaidah, terdapat pelafalan atau bunyi dari suatu huruf arab. Fitur ini bermanfaat bagi *user* awam atau yang belum bisa atau belum faham tentang tata cara menulis huruf arab.



Gambar 3.14 *Sequence Diagram* Pengenalan Huruf Arab

Pada *form* pengenalan hijaiyah, memiliki kontrol petunjuk, yaitu tutorial tata cara menulis huruf arab berupa gambar, animasi bergerak dan juga bunyi

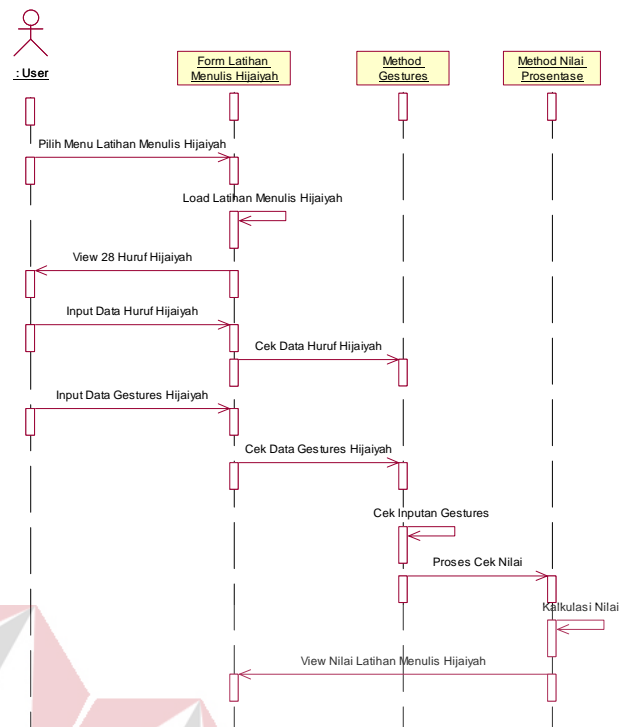
pelafalan dari huruf arab. Bila ingin melihat animasi bergerak tekan tombol *play*, tombol suara untuk mendengarkan pelafalan huruf hijaiyah yang dikehendaki.

### 3.2.11 *Sequence Diagram* Latihan Menulis Hijaiyah

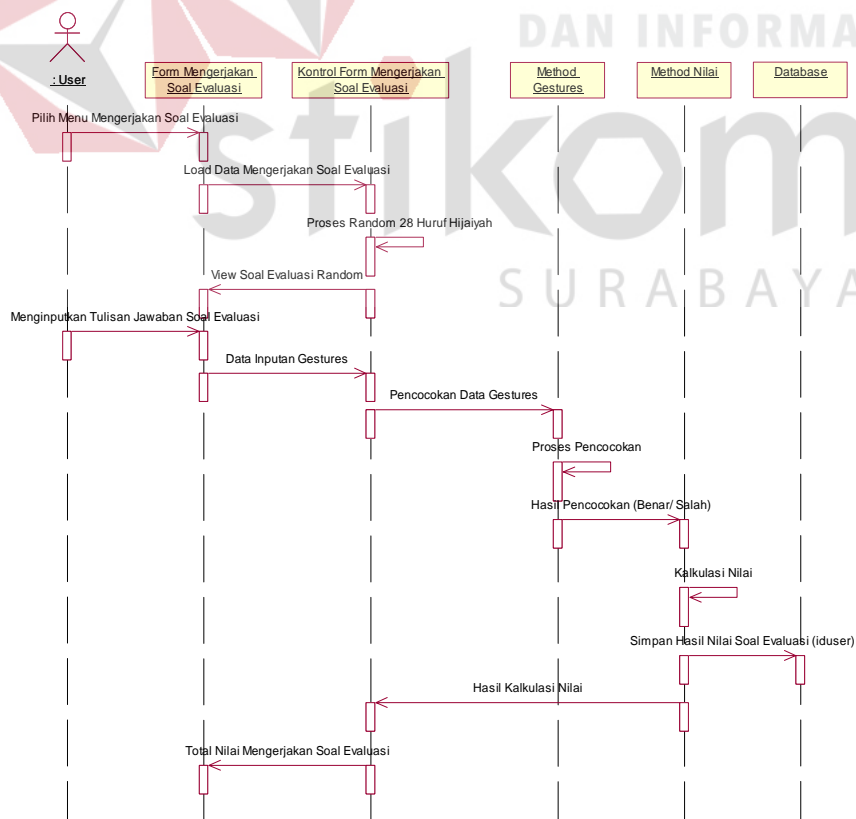
Gambar 3.15 adalah gambaran dari proses latihan menulis hijaiyah. Pada fitur ini, berguna membantu dalam memahami dan menghafal tata cara penulisan masing-masing huruf hijaiyah. Awalnya *user* harus memilih salah satu dari 28 huruf hijaiyah kemudian *user* menuliskan huruf hijaiyah yang telah dipilihnya pada layar kosong yang telah disediakan oleh sistem. Pada fitur ini sistem akan langsung memberikan penilaian berupa prosentase benar dari banyak penulisan yang telah dilakukan oleh *user*, karena ketika *user* menuliskan pada layar kosong sistem akan langsung melakukan pencocokan berdasarkan titik koordinat yang disimpan ke dalam *array* dengan algoritma yang dibuat.

### 3.2.12 *Sequence Diagram* Mengerjakan Soal Evaluasi

Pada gambar 3.16 adalah *Sequence diagram* mengerjakan soal evaluasi yang merupakan fitur untuk menguji kemampuan *user* dalam menghafal huruf hijaiyah. Pada fitur ini sistem memberikan soal secara acak, kemudian *user* harus menjawab dengan menulis huruf hijaiyah sesuai dengan soal yang diberikan oleh sistem, sehingga *user* harus hafal bentuk dan cara penulisan dari 28 huruf arab. Setelah itu, sistem memberikan info nilai dari hasil jawaban *user* melakukan latihan kuis soal evaluasi. Nilai tersebut akan disimpan pada *database* berdasarkan idlogin atau iduser sebagai histori nilai *user* dalam melakukan latihan kuis soal evaluasi.



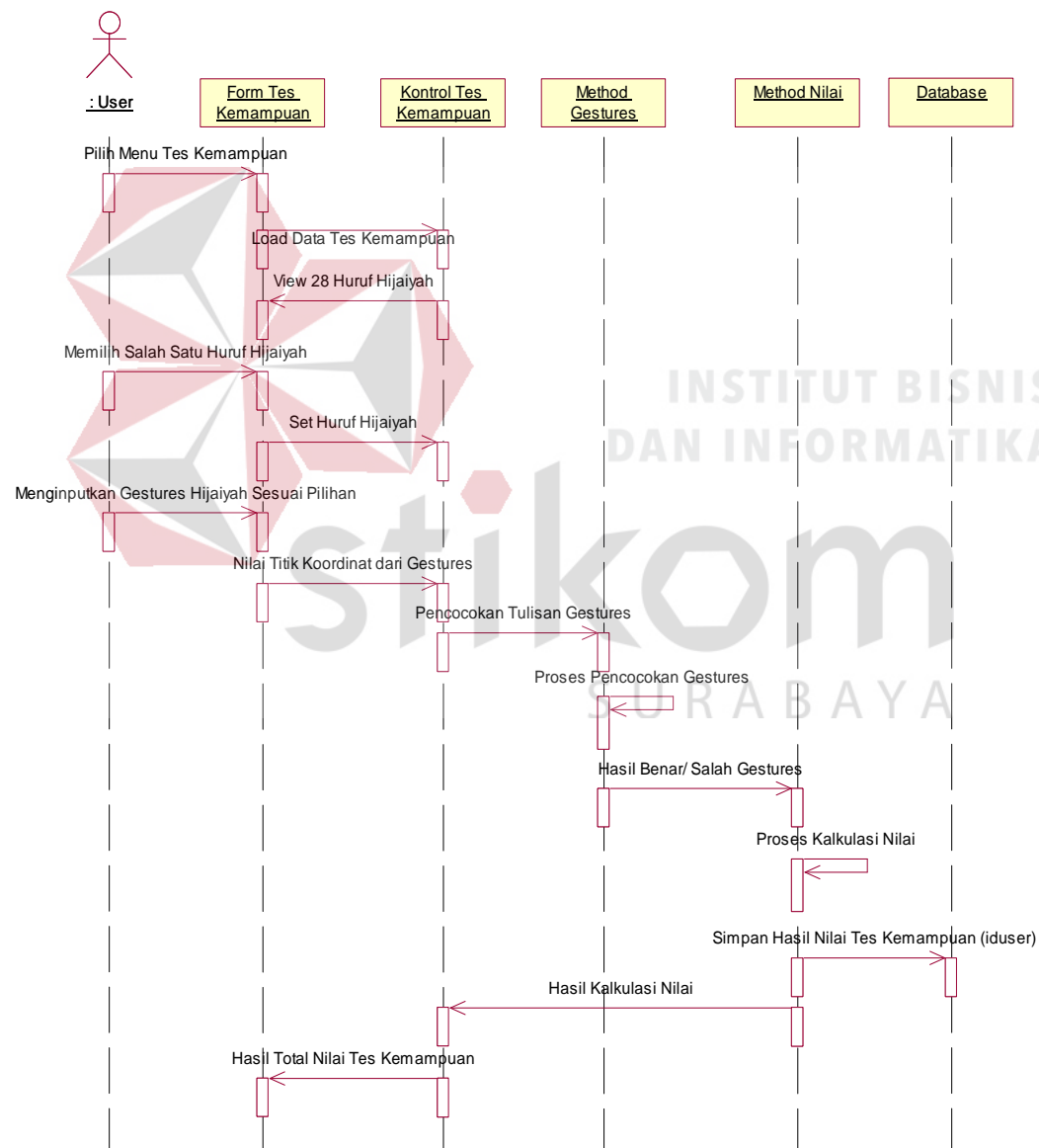
Gambar 3.15 *Sequence Diagram* Latihan Menulis Hijaiyah



Gambar 3.16 *Sequence Diagram* Mengerjakan Soal Evaluasi

### 3.2.13 Sequence Diagram Tes Kemampuan

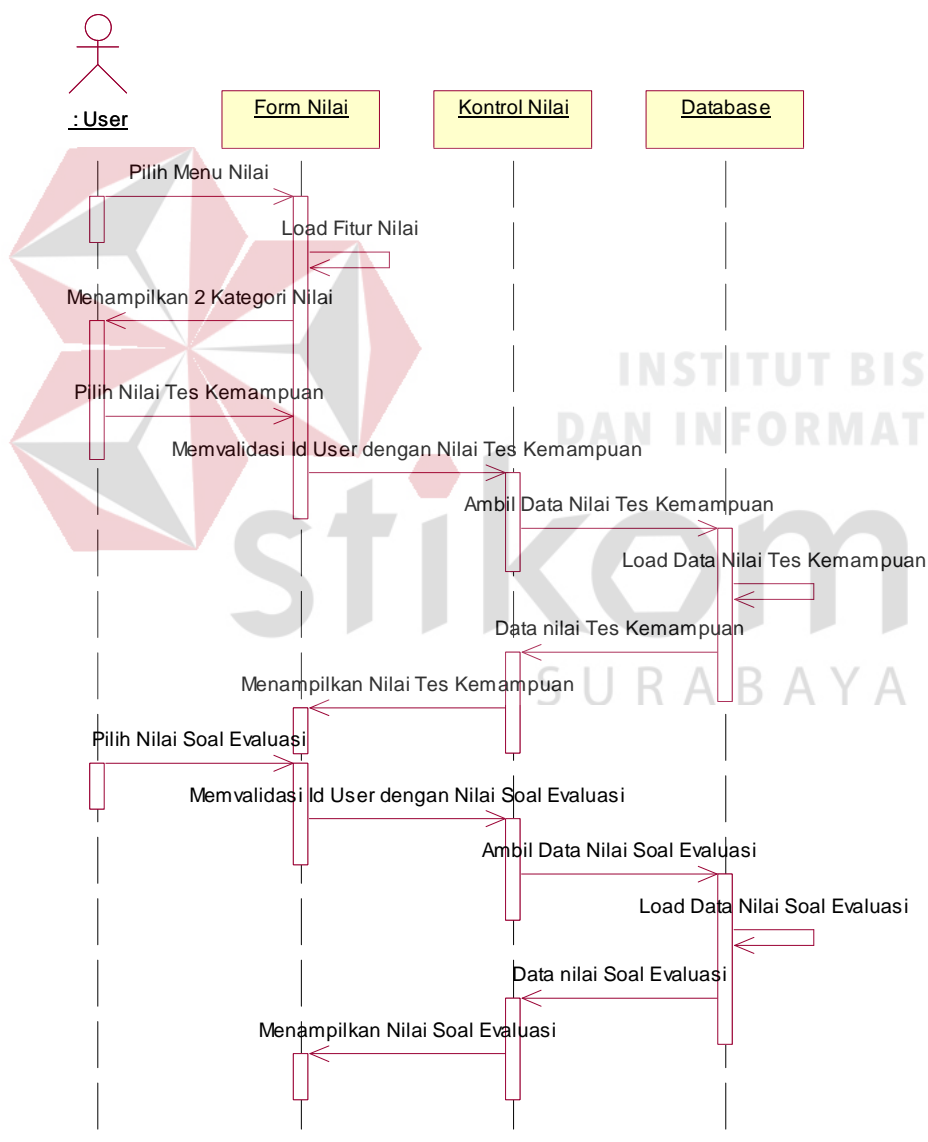
Fitur ini tidak jauh berbeda dengan *Sequence diagram* mengerjakan tes kemampuan. Perbedaannya pada fitur ini *user* akan memilih salah satu huruf arab, kemudian *user* akan mencoba menulis sebanyak lima kali. Hasil dari tulisan *user* tersebut, akan dikalkulasi oleh sistem yang nantinya akan menjadi nilai atau *score* berupa prosentase. *Sequence diagram* tersebut dapat dilihat pada Gambar 3.17.



Gambar 3.17 *Sequence Diagram* Tes Kemampuan Huruf Arab

### 3.2.14 Sequence Diagram Nilai

Fitur nilai adalah fitur untuk melihat nilai dari hasil pengerjaan *user* dalam mengerjakan tes kemampuan ataupun soal evaluasi. Semua hasil nilai latihan *user* disimpan pada fitur nilai. Pada fitur ini *user* dapat melihat dua macam penilaian yaitu nilai tes kemampuan dan nilai soal evaluasi. Berikut penjelasan seperti pada Gambar 3.18.



Gambar 3.18 Sequence Diagram Nilai



### 3.2.15 *Class Diagram* Aplikasi Menulis Huruf Arab

*Class diagram* digunakan untuk menampilkan kelas-kelas atau paket-paket dalam sistem dan relasi antar mereka. Biasanya, dibuat beberapa diagram kelas untuk satu sistem. Satu *class diagram* menampilkan subset dari kelas-kelas dan relasinya. *Class diagram* lainnya mungkin menampilkan kelas-kelas termasuk *attribut* dan operasi dari kelas-kelas pembentuk diagram.

*Class diagram* adalah alat perancangan terbaik untuk tim pengembang perangkat lunak. *Class diagram* membantu tim pengembang mendapatkan pola kelas-kelas dalam sistem, struktur sistem sebelum menuliskan kode program, dan membantu untuk memastikan bahwa sistem adalah rancangan terbaik dari beberapa alternatif rancangan. (Sholih, 2010)

Berdasarkan perencanaan sistem pada *use case diagram*, dibutuhkan *class-class* untuk membangun dan mendukung jalannya aplikasi. Hubungan antar *class* tersebut dapat digambarkan dalam sebuah *class diagram*. *Class diagram* dari sistem yang dibangun tidak ditampilkan secara keseluruhan, melainkan ditampilkan secara terpisah sesuai relasi-relasi yang ada untuk mempermudah pembacaan.

*Class* yang dibuat untuk aplikasi ini menggunakan pemodelan UML dengan konsep pemodelan *Model-View-Controller* (MVC). *Class model* merupakan *class* yang akan menangani segala sesuatu yang berhubungan dengan entitas. *Class view* merupakan *class* yang akan menangani segala sesuatu yang berhubungan dengan tampilan *user interface*. *Class controller* merupakan *class* yang menangani segala proses seperti proses pencarian data dan segala proses

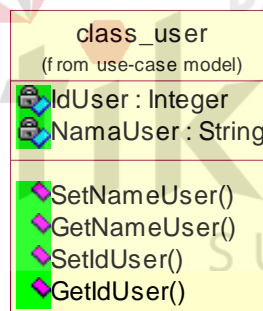
yang berhubungan dengan penyimpanan data. *Class* yang dapat digunakan dalam *class* diagram pada aplikasi menulis huruf hijaiyah yaitu:

### A. *Class Model*

*Class Model* atau *class entitas* adalah *class* yang digunakan menangani informasi yang mungkin disimpan secara permanen. Berikut ini merupakan *class* model yang ada dalam sistem informasi penjualan beserta atribut dan operasi dalam *class* tersebut.

#### A.1 *Class Diagram User*

*Class diagram user* digunakan untuk menyimpan nama *user* yang melakukan pembelajaran menulis huruf arab pada aplikasi. Kemungkinan aplikasi digunakan oleh banyak orang merupakan salah satu alasan dibuatnya database untuk menampung nama *user* dan hasil pembelajaran yang telah dikerjakan.

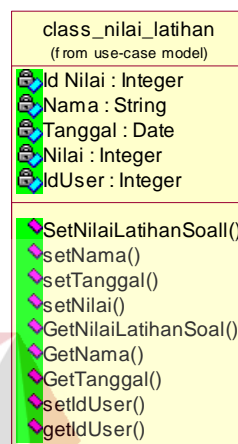


Gambar 3.19 *Class Diagram User*

Atribut yang terdapat pada *class diagram user* pada Gambar 3.19 yaitu *NamaUser* yang berfungsi untuk menyimpan nama *user* aplikasi. Operasi yang terdapat pada *class diagram user* yaitu *SetNameUser* digunakan untuk menyimpan nama *user* dan *GetNameUser* digunakan untuk mengambil nama *user* sesuai dengan daftar nama yang dipilih oleh *user* aplikasi atau yang telah dibuat.

## A. 2 Class Diagram Nilai Latihan

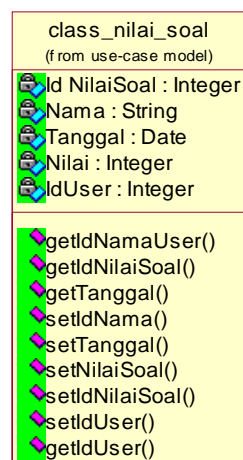
*Class diagram* nilai latihan digunakan untuk menyimpan nilai atau *score user* dalam mengerjakan soal tes kemampuan. Nilai tersebut disimpan pada database dan nantinya dijadikan histori *score* pada *user*. Gambar 3.20 merupakan *class* nilai latihan.



Gambar 3.20 *Class Diagram* Nilai Latihan

## A. 3 Class Diagram Nilai Soal

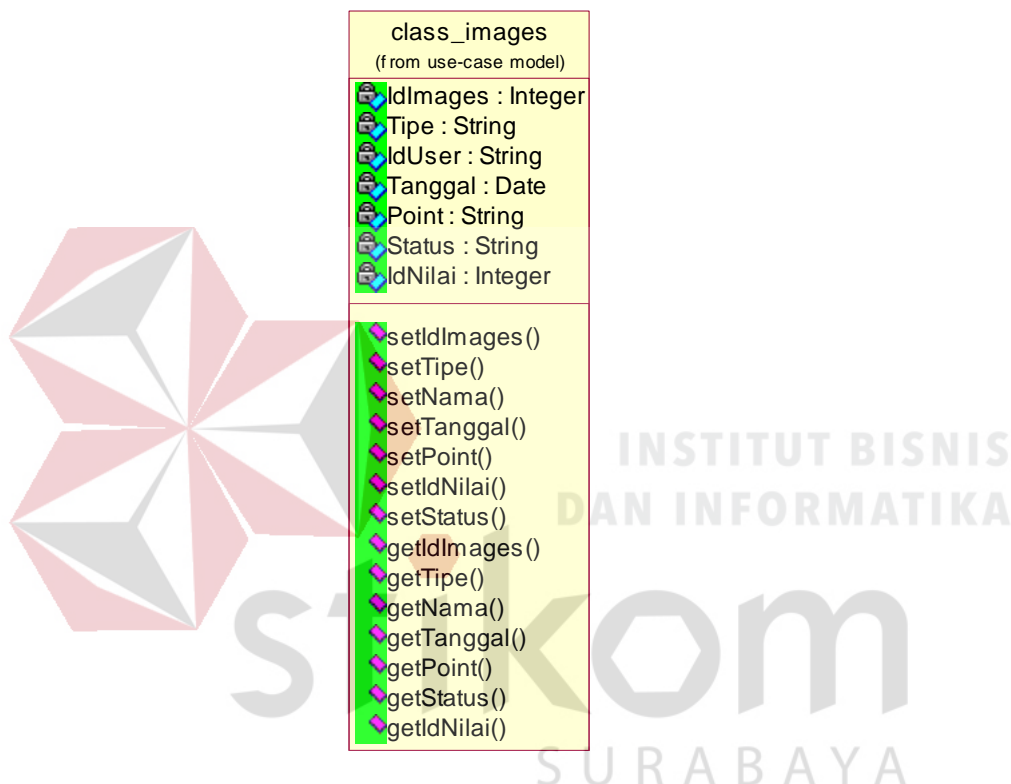
*Class diagram* nilai soal digunakan untuk menyimpan nilai atau *score user* dalam mengerjakan latihan soal. Nilai tersebut disimpan pada database dan nantinya dijadikan *history score* pada *user*. Gambar 3.21 merupakan *class* nilai.



Gambar 3.21 *Class Diagram* Nilai soal

#### A. 4 *Class Diagram Images*

*Class diagram images* digunakan untuk menyimpan gambar hasil tulisan *user* pada waktu mengerjakan latihan. Gambar tersebut disimpan pada database dan nantinya dijadikan *history gestures* pada *user*. Gambar 3.22 merupakan *class images*.



Gambar 3.22 *Class Diagram Images*

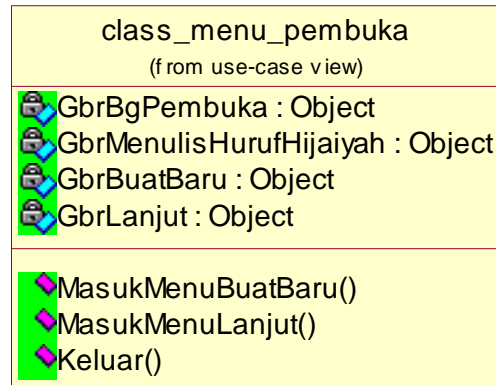
#### B. *Class Diagram View*

*Class diagram view* yaitu kelas yang terletak diantara sistem dengan sekelilingnya. Semua *form*, laporan, *user interface*, termasuk dalam kategori *Class View*. Berikut ini *Class View* pada Aplikasi Pembelajaran Menulis Arab.

##### B. 1 *Class Diagram Menu Pembuka*

*Class* pembuka berfungsi untuk menampilkan *form* awal pada saat *user* membuka aplikasi menulis huruf arab. Fitur pada *form* ini hanya sekedar

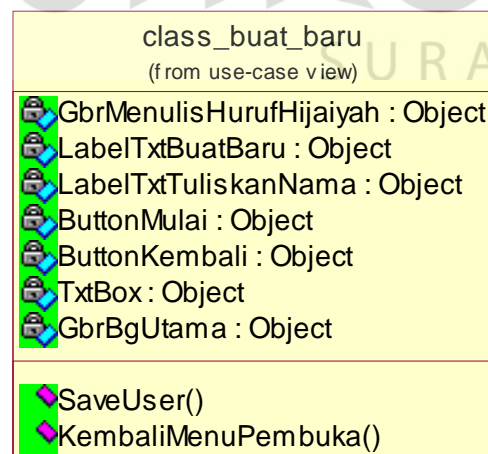
menampilkan gambar berupa sapaan dan tombol mulai. Berikut adalah *Class* Menu Pembuka pada Gambar 3.23.



Gambar 3.23 *Class Diagram* Menu Pembuka

## B. 2 *Class Diagram* Buat Baru

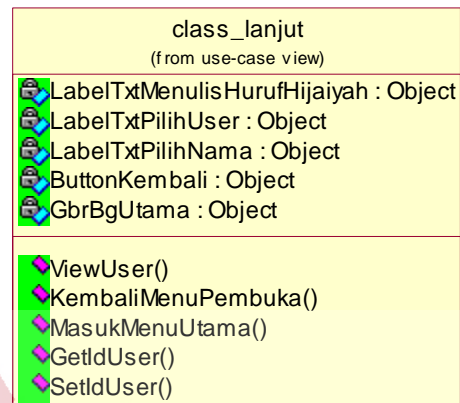
*Class diagram* buat baru adalah menu untuk membuat *user* baru bila pertama kali menjalankan aplikasi menulis huruf arab. Buat *user* ini wajib dilakukan karena bila *user* tidak mempunyai *user login*, maka tidak akan bisa masuk ke dalam menu utama aplikasi. Berikut *class* buat baru pada Gambar 3.24.



Gambar 3.24 *Class Diagram* Buat Baru

### B. 3 Class Diagram Lanjut

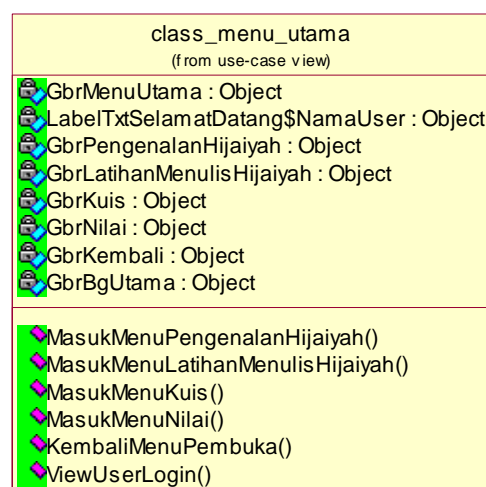
*Class diagram* lanjut adalah *class* untuk menampilkan daftar *user* yang telah terdaftar atau tersimpan. Bagi *user* yang mempunyai *user login*, maka ada di menu lanjut. Berikut rincian *class diagram* lanjut pada Gambar 3.25.



Gambar 3.25 Class Diagram Lanjut

### B. 4 Class Diagram Menu Utama

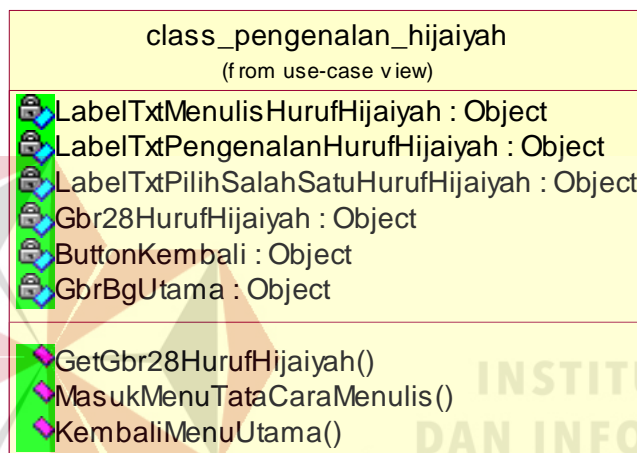
Pada *class diagram* menu utama adalah inti fitur dari aplikasi menulis huruf arab. *Atribut* yang ada pada *class* menu utama adalah menu-menu yang ada pada aplikasi, kemudian operasi yang dilakukan adalah masuk pada menu dan keluar ke menu pembuka. Berikut *class* menu utama pada Gambar 3.26.



Gambar 3.26 Class Diagram Menu Utama

### B. 5 Class Diagram Pengenalan Huruf Hijaiyah

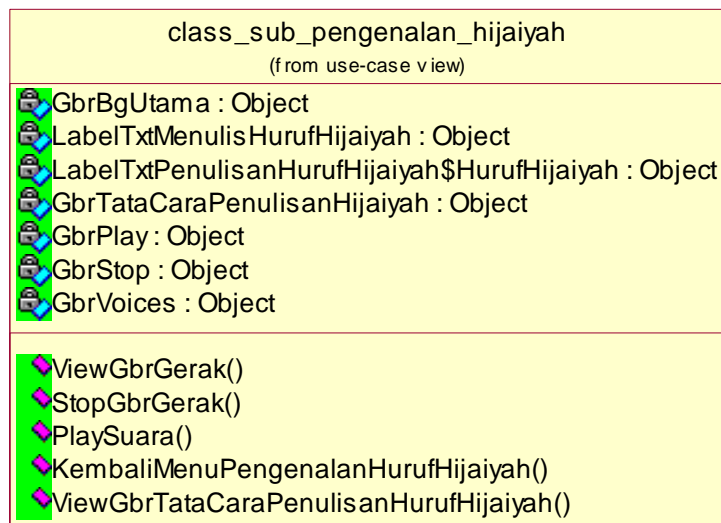
Pada *class diagram* pengenalan hijaiyah adalah *class* dalam pengenalan tata cara penulisan huruf hijaiyah/ arab sesuai dengan kaidah. Pada atribut *class* pengenalan hijaiyah terdapat gambar 28 huruf arab yang harus dipilih oleh *user* kemudian mengoperasikan ke halaman selanjutnya sesuai dengan huruf yang dipilih *user*. Berikut rincian *class* pengenalan hijaiyah pada Gambar 3.27.



Gambar 3.27 Class Diagram Pengenalan Hijaiyah

### B. 6 Class Diagram Sub Pengenalan Huruf Hijaiyah

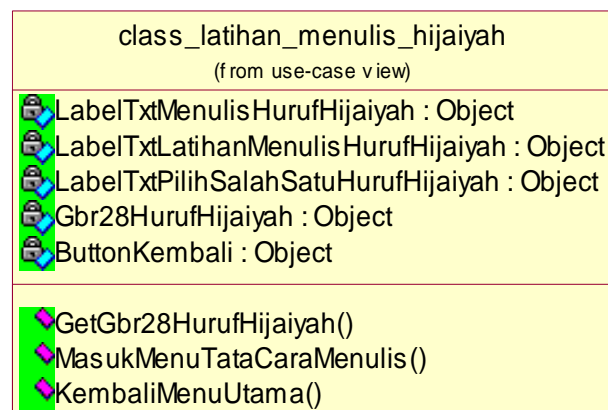
Pada *class diagram* sub pengenalan hijaiyah adalah *class* setelah *user* memilih salah satu dari 28 huruf hijaiyah kemudian pada fitur *class* ini terdapat pengenalan tata cara menulis huruf hijaiyah. Dalam sub pengenalan tata cara terdapat atribut *play*, *stop* dan *sound*. Operasi pada *class* sub pengenalan hijaiyah adalah menampilkan gambar gerak pada tombol *play*, berhenti atau kembali seperti tampilan awal pada tombol *stop* dan dapat mendengarkan suara pada tombol *sound*. Berikut rincian *class diagram* pengenalan hijaiyah pada Gambar 3.28.



Gambar 3.28 *Class Diagram* Sub Pengenalan Hijaiyah

### B. 7 *Class Diagram* Latihan Menulis Hijaiyah

Pada *class diagram* latihan menulis hijaiyah adalah *class* dalam mendeteksi *input gestures* dari *user* sesuai dengan huruf yang dipilih. Atribut pada *class* ini, *user* terlebih dahulu memilih salah satu huruf hijaiyah, kemudian *class* ini melakukan operasi mengarahkan ke halaman selanjutnya sesuai huruf hijaiyah yang dipilih *user*. Berikut rincian dari *class diagram* latihan menulis hijaiyah pada Gambar 3.29.

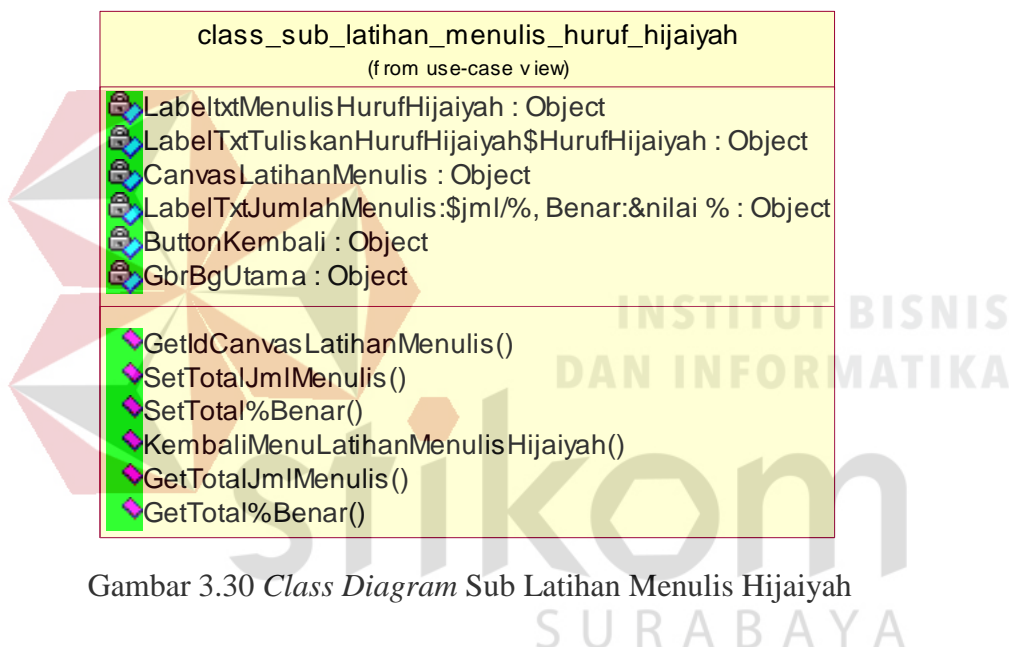


Gambar 3.29 *Class Diagram* Latihan Menulis



### B. 8 Class Diagram Sub Latihan Menulis Hijaiyah

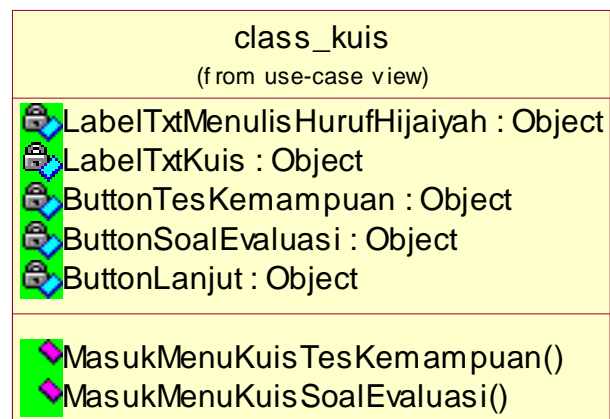
Pada *class diagram* sub latihan menulis hijaiyah adalah lanjutan dari *class* latihan menulis hijaiyah. *Class* ini adalah inti dari latihan *user* dalam menulis huruf hijaiyah sesuai huuf yang dipilih. Atribut pada *class* ini terdapat *canvas* untuk media *gestures*, kemudian melakukan operasi pencocokan *input gestures* terhadap *gestures template* (huruf hijaiyah yang dipilih *user*). Berikut rincian dari *class diagram* sub latihan menulis hijaiyah pada Gambar 3.30.



Gambar 3.30 Class Diagram Sub Latihan Menulis Hijaiyah

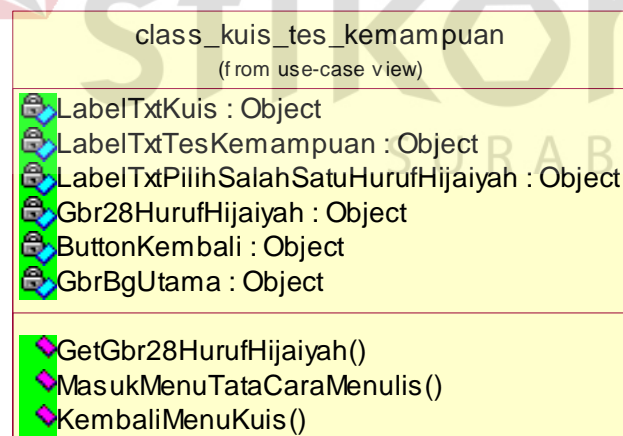
### B. 9 Class Diagram Kuis

Pada *class* kuis, terdapat fitur untuk latihan *user* dalam menulis huruf hijaiyah. *Class diagram* kuis terdapat dua macam yaitu kuis tes kemampuan dan soal evaluasi. Pada *class diagram* kuis mempunyai fungsi untuk menampilkan setiap menu kuis berdasarkan *index* yang dipilih. Berikut atribut dan operasi pada *Class* kuis pada Gambar 3.31.

Gambar 3.31 *Class Diagram* Kuis

### B. 10 *Class Diagram* Kuis Tes Kemampuan

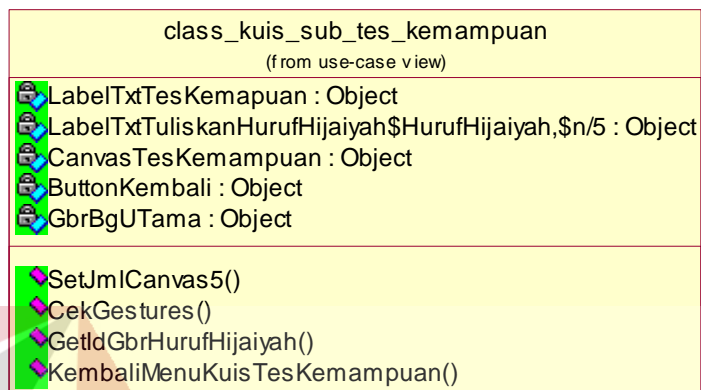
*Class diagram* kuis tes kemampuan, adalah fitur dimana sistem memberikan info prosentase benar dari hasil *input user* sesuai huruf hijaiyah yang dipilih. Pada *class* ini terdapat atribut gambar 28 huruf hijaiyah yang nantinya *user* wajib memilih salah satu huruf untuk tes kemampuan. Berikut atribut dan operasi pada *Class diagram* kuis tes kemampuan pada Gambar 3.32.

Gambar 3.32 *Class Diagram* Kuis Tes Kemampuan

### B. 11 *Class Diagram* Sub Kuis Tes Kemampuan

*Class diagram* sub kuis tes kemampuan, adalah lanjutan dari *class* kuis tes kemampuan. Pada *class* ini, terdapat atribut *canvas* sebagai media *gestures*,

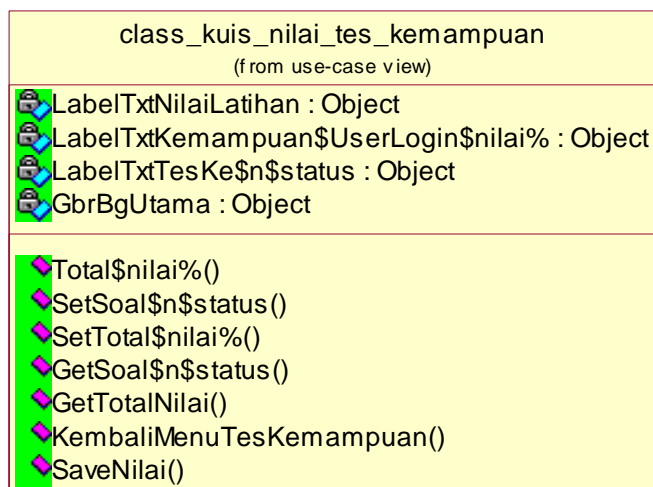
kemudian hasil *input gestures user* akan dicocokkan pada *gestures template* sehingga sistem akan melakukan operasi perhitungan prosentase benar. Berikut atribut dan operasi pada *Class diagram* sub kuis tes kemampuan pada Gambar 3.33.



Gambar 3.33 *Class Diagram* Sub Kuis Tes Kemampuan

## B.12 *Class* Kuis Nilai Tes Kemampuan

*Class* kuis nilai tes kemampuan, adalah hasil nilai dari *class* sub kuis tes kemampuan. Pada *class* ini, terdapat atribut yang berguna untuk menampilkan hasil nilai pengerjaan *user* dalam tes kemampuan. Berikut atribut dan operasi pada *Class diagram* sub kuis tes kemampuan pada Gambar 3.34.



Gambar 3.34 *Class Diagram* Kuis Nilai Tes Kemampuan

### B. 13 Class Diagram Kuis Soal Evaluasi

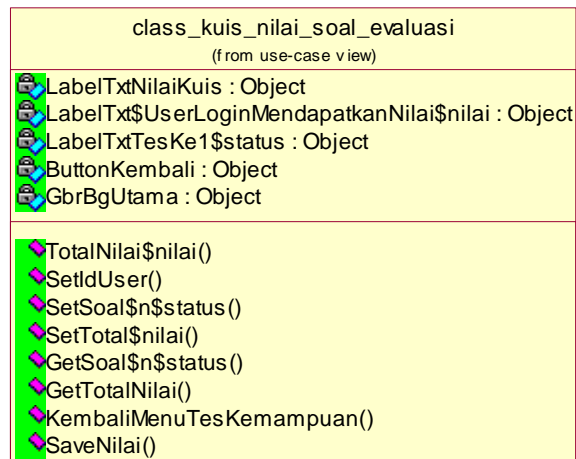
Class kuis soal evaluasi, latihan menulis yang dilakukan oleh *user* dengan sistem acak atau huruf hijaiyah yang diberikan oleh sistem adalah *random*. Pada *class* ini, terdapat atribut *canvas* sebagai media *gestures*, kemudian operasi yang dilakukan adalah pengacakan soal latihan, pencocokan *gestures* terhadap tulisan *user* dan total nilai. Berikut atribut dan operasi pada *Class diagram* kuis soal evaluasi pada Gambar 3.35.



Gambar 3.35 Class Diagram Kuis Soal Evaluasi

### B. 14 Class Diagram Kuis Nilai Soal Evaluasi

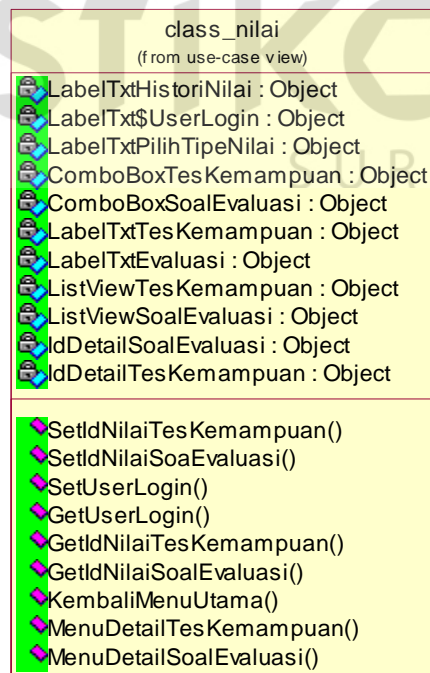
*Class diagram* kuis nilai soal evaluasi, adalah hasil nilai dari *class* kuis soal evaluasi. Pada *class* ini, terdapat atribut yang berguna untuk menampilkan hasil nilai pengerjaan *user* dalam kuis soal evaluasi. Terdapat beberapa atribut yang berguna untuk menampilkan hasil nilai dari pengerjaan *user* berdasarkan *input gestures*. Berikut atribut dan operasi pada *Class diagram* kuis nilai soal evaluasi pada Gambar 3.36.



Gambar 3.36 *Class Diagram* Kuis Nilai Soal Evaluasi

### B. 15 *Class Diagram* Nilai

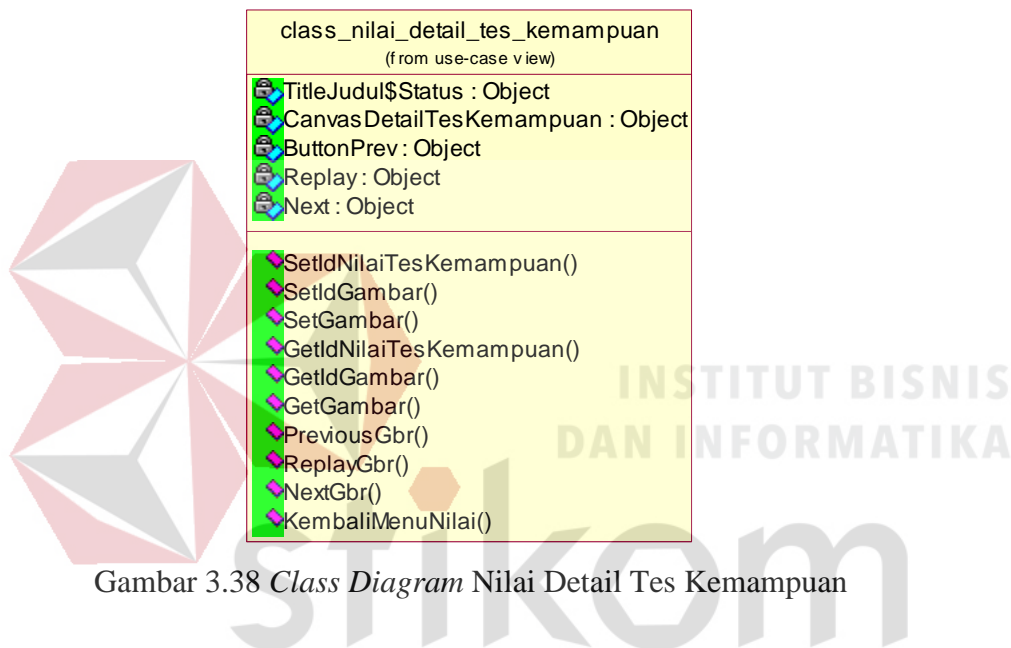
*Class diagram* nilai adalah *class* yang berisi histori nilai yang telah dilakukan oleh *user*. Hasil nilai latihan *user* dapat dilihat pada fitur nilai. Pada *class* ini terdapat dua nilai yaitu nilai tes kemampuan dan nilai soal evaluasi. Berikut *Class diagram* Nilai pada Gambar 3.37.



Gambar 3.37 *Class Diagram* Nilai

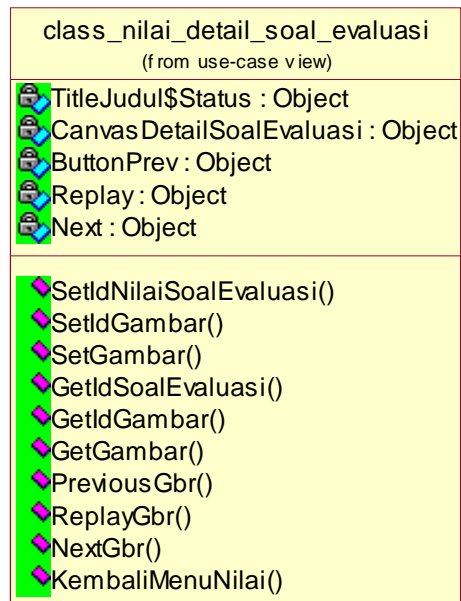
### B. 16 Class Diagram Nilai Detail Tes Kemampuan

*Class diagram* nilai detail tes kemampuan adalah *class* yang berisi histori *gestures/* tulisan yang telah dilakukan oleh *user*. Hasil *input gestures* latihan *user* dapat dilihat pada fitur nilai. Pada *class* ini terdapat beberapa fungsi seperti tombol *play, stop, dan sound*. Berikut *Class diagram* detail nilai pada tes kemampuan pada Gambar 3.38.



### B. 17 Class Diagram Nilai Detail Soal Evaluasi

*Class diagram* nilai detail soal evaluasi adalah *class* yang berisi histori *gestures/* tulisan yang telah dilakukan oleh *user*. Hasil *input gestures* latihan *user* dapat dilihat pada fitur nilai. Pada *class* ini terdapat beberapa fungsi seperti seperti melihat nilai dan juga detail nilai. Detail nilai digunakan untuk melihat hasil tulisan *user* dalam melakukan latihan soal evaluasi yang telah dilakukan sebelumnya. Berikut *Class diagram* detail nilai pada soal evaluasi pada Gambar 3.39.



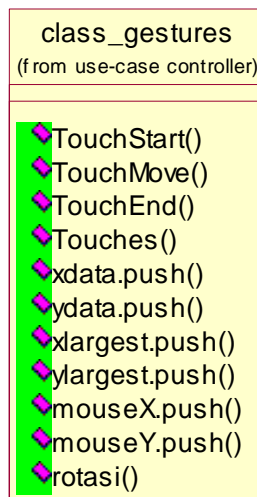
Gambar 3.39 *Class Diagram* Nilai Detail Soal Evaluasi

### C. *Class Diagram Controller*

*Class Diagram Controller* bertanggung jawab untuk mengkoordinasikan kegiatan-kegiatan terhadap *class* lainnya. *Class* ini bersifat optional, tetapi jika *class control* ini diputuskan untuk digunakan dalam sistem, maka lazimnya satu *class control* untuk satu *use case*. *Class diagram controller* digunakan untuk mengatur urutan kejadian dalam *use case* tersebut. *Class diagram controller* pada aplikasi pembelajaran menulis huruf arab yaitu:

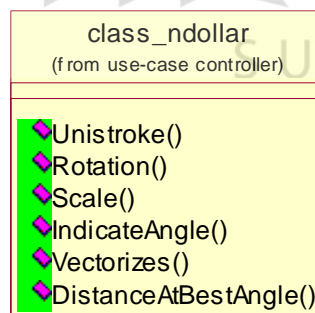
#### C.1 *Class Diagram Diagram Gestures*

*Class diagram gestures* digunakan pada aplikasi pembelajaran menulis huruf arab untuk melakukan proses pencocokan dan mendeteksi pola. Pada proses ini membutuhkan beberapa *variable* untuk proses pencocokan, seperti mousedown, mousemove dan mouse up. Untuk jelasnya seperti pada Gambar 3.40 adalah *class diagram gestures*.

Gambar 3.40 *Class Diagram Gestures*

## C.2 *Class Diagram \$N*

*Class diagram \$N* adalah metode dalam pencocokan *gestures*. Pada *class* ini terdapat beberapa *variable* yang berfungsi untuk mengoperasikan dalam pencocokan *gestures*. *Class \$N* mempunyai beberapa operasi yang nantinya masing-masing operasi menyesuaikan dengan *input gestures* dari *user*. Berikut rincian *class diagram \$N* seperti pada Gambar 3.41.

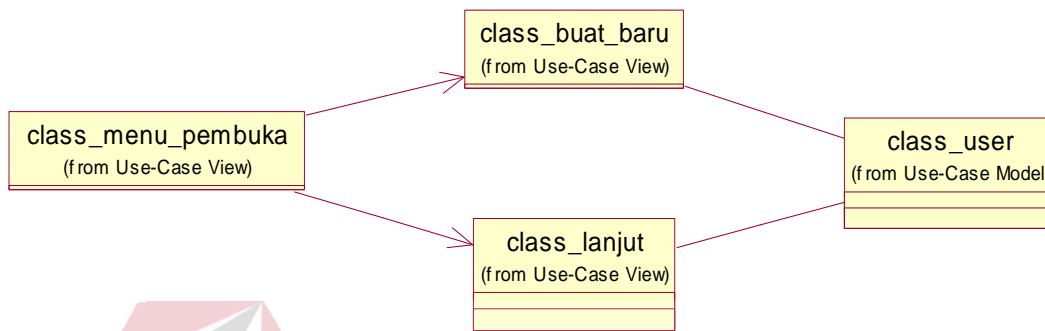
Gambar 3.41 *Class Diagram \$N*

## D. *Relasi Class Diagram*

Relasi *class diagram* adalah keterkaitan *class diagram* antara dua atau lebih yang saling berhubungan. Pada relasi *class diagram list user* tampilan

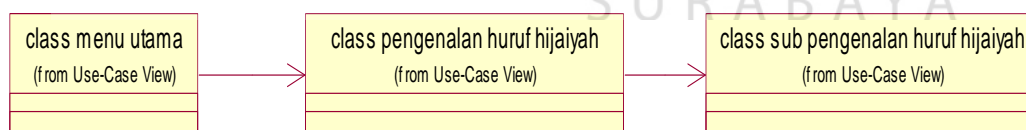


menjelaskan proses yang saling terkait antar *class login* dan *list user*. Pada relasi tersebut dijelaskan yang terkait antara *class login* yang satu dengan *class* yang lain dalam proses *login* maupun buat *user* baru di dalam aplikasi menulis huruf arab. Berikut relasi class diagram *list user* pada Gambar 3.42.



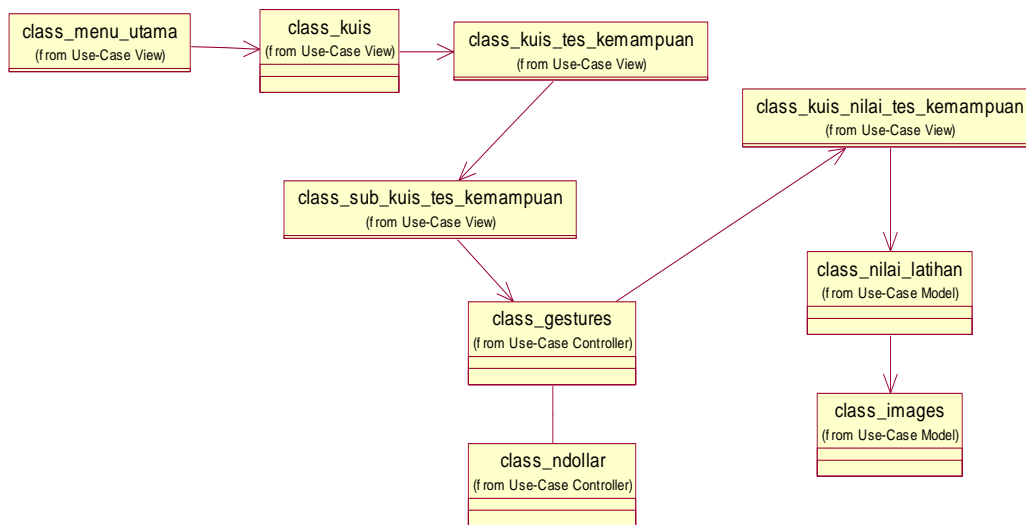
Gambar 3.42 Relasi *Class Diagram List User*

Pada Gambar 3.43 adalah relasi *class diagram* pengenalan huruf arab. Relasi *class* ini menjelaskan tentang relasi yang melakukan proses pada menu pengenalan huruf hijaiyah, serta menampilkan fitur tata cara penulisan huruf arab secara benar.



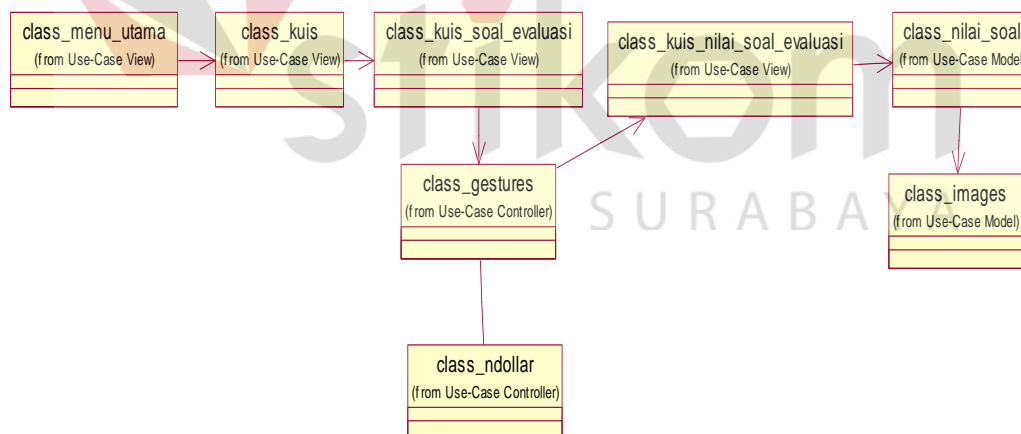
Gambar 3.43 Relasi *Class Diagram Pengenalan Huruf Arab*

Pada Gambar 3.44 adalah relasi *class diagram* kuis tes kemampuan. *Class* ini adalah relasi tentang proses dalam menjalankan kuis tes kemampuan dari *user* pada waktu latihan tes kemampuan dan juga ketika sistem dalam proses melakukan kalkulasi nilai dari *user* melakukan kuis tes kemampuan.



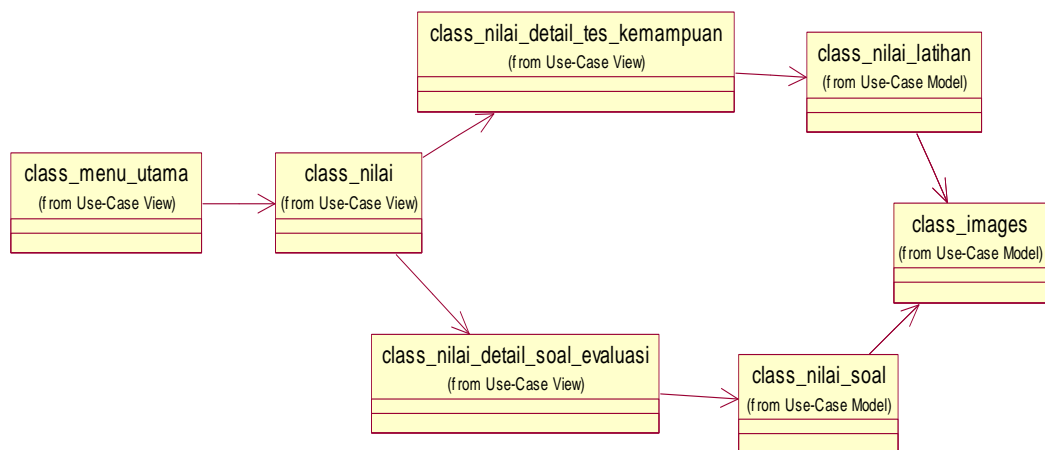
Gambar 3.44 Relasi *Class Diagram* Tes Kemampuan

Pada Gambar 3.45 adalah relasi *class diagram* mengerjakan soal evaluasi. *Class* ini adalah relasi tentang proses dalam menjalankan kuis soal evaluasi dari *user* pada waktu latihan mengerjakan soal evaluasi.



Gambar 3.45 Relasi *Class Diagram* Mengerjakan Soal Evaluasi

Pada Gambar 3.46 adalah relasi *class diagram* nilai. *Class* ini adalah relasi tentang proses dalam menampilkan histori nilai serta dapat menampilkan hasil tulisan *gestures* yang telah dilakukan oleh *user*.



Gambar 3.46 Relasi *Class Diagram* Nilai

### 3.3 Perancangan *User Interface*

Perancangan *user interface* sangat diperlukan agar *user* dapat berinteraksi dengan aplikasi, sehingga dibutuhkan perancangan secara detail mengenai desain *user interface* aplikasi berdasarkan informasi yang ditampilkan pada layar *device*. Tampilan yang akan dibuat adalah tampilan pembuka, *login*, menu utama, pengenalan huruf arab, evaluasi dan nilai.

#### 3.3.1 Tampilan *Login*

Tampilan *login* berfungsi untuk mengidentifikasi *user* dalam menggunakan aplikasi. Sistem harus mengetahui identitas *user*, karena di dalam sistem nantinya akan menyimpan *history* nilai. Bila *user* belum mempunyai *user*, maka pada tampilan *login* ada fitur untuk membuat *user* baru dan bagi *user* yang sudah memiliki *user*, maka *user* hanya tinggal memilih *user* yang dapat ditampilkan oleh sistem. Untuk lebih jelas tampilan *login* dapat dilihat pada Gambar 3.47.



Gambar 3.47 Menu Login

### 3.3.2 Menu Buat *User* Baru

Untuk membuat *user* baru, *user* memilih menu buat baru. Kemudian sistem akan menampilkan menu untuk membuat *user* baru. Pada menu tersebut terdapat *textbox* yang telah disediakan oleh sistem. *User* mengisikan nama pada *textbox*, untuk menyimpan pilih tombol mulai, kemudian sistem akan langsung menyimpan nama *user* tersebut dan langsung akan masuk ke dalam menu utama Aplikasi Menulis Huruf Arab yang disediakan oleh sistem. Untuk lebih jelas tampilan menu buat *user* baru pada Aplikasi Menulis Huruf Arab dapat dilihat pada Gambar 3.48.



Gambar 3.48 Menu Buat *User* Baru

### 3.1.1 Menu Daftar *User*

Bila *user* telah membuat idlogin, maka nama *user* akan tersimpan pada *database*. Untuk menampilkan daftar *user* yang telah dibuat, pada menu pembuka *user* memilih menu lanjut., kemudian sistem akan menampilkan nama *user* yang telah disimpan pada *database*. Pada menu daftar *user* dapat dilihat pada Gambar 3.49. *User* cukup memilih nama yang dibuat sebelumnya, kemudian pilih menu masuk. Setelah pilih menu masuk, sistem akan menampilkan menu utama pada aplikasi dan semua aktivitas yang pernah dilakukan *user* sebelumnya akan di *load* oleh sistem, sehingga histori nilai akan ditampilkan oleh sistem.



Gambar 3.49 Menu Daftar User

### 3.3.3 Tampilan Menu Utama

Setelah *user login* atau masuk, maka *user* akan masuk ke menu utama. Semua fitur yang ada pada aplikasi ada pada menu utama, yaitu pengenalan huruf hijaiyah, latihan menulis huruf hijaiyah, kuis dan nilai. Apabila *user* tidak memiliki atau belum terdaftar, maka tidak bisa masuk pada menu utama Aplikasi Menulis Huruf Arab, karena untuk mengakses menu utama pada Aplikasi Menulis Huruf Arab, *user* diwajibkan mempunyai nama *user* untuk masuk pada aplikasi. Untuk lebih jelas tampilan menu utama pada Aplikasi Menulis Huruf Arab dapat dilihat pada Gambar 3.50.



Gambar 3.50 Tampilan Menu Utama

### 3.3.4 Tampilan Pengenalan Huruf Hijaiyah

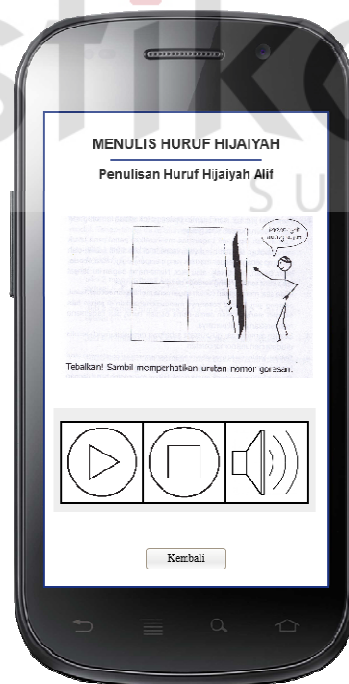
Tampilan menu pengenalan huruf hijaiyah berisi tentang tutorial atau tata cara penulisan huruf arab yang benar sesuai dengan kaidah. Bila *user* ingin mengetahui tata cara menulis atau kaidah penulisan, *user* memilih salah satu huruf arab, kemudian, masuk ke dalam menu tutorial penulisan. Untuk lebih jelas tampilan pengenalan huruf hijaiyah dapat dilihat pada Gambar 3.51.

Pada Gambar 3.52 adalah menu setelah memilih tutorial atau pengenalan huruf hijaiyah dan juga terdapat fitur mulai, maksudnya mulai untuk sistem memberikan tata cara menulis huruf arab dengan gambar bergerak, *user* juga bisa

mendengarkan lafalan huruf yang dipilihnya, dengan memilih tombol *sound* atau suara.



Gambar 3.51 Tampilan Pengenalan Huruf Arab



Gambar 3.52 Tampilan Tutorial



### 3.3.5 Menu Latihan Menulis Hijaiyah

Pada fitur latihan menulis hijaiyah, *user* dapat melakukan latihan menulis huruf hijaiyah. Awalnya memilih huruf hijaiyah yang diinginkan seperti yang terlihat pada Gambar 3.53. Setelah itu *user* menuliskan huruf hijaiyah pada *canvas* yang telah disediakan oleh sistem. Hasil tulisan *user*, akan dicek oleh sistem benar dan salahnya karena sistem memberikan info prosentase benar tulisan dari sejumlah tulisan yang dilakukan oleh *user* dan juga jumlah salah *user* dalam menulis huruf arab. Untuk jelasnya seperti pada Gambar 3.54



Gambar 3.53 Menu Latihan Menulis Huruf Hijaiyah



Gambar 3.54 *Canvas Tulisan Gestures*

### 3.3.6 Menu Kuis

Pada aplikasi ini terdapat menu kuis yang terbagi menjadi dua macam yaitu Tes Kemampuan dan Soal Evaluasi. Berikut penjelasan dari masing-masing fitur.

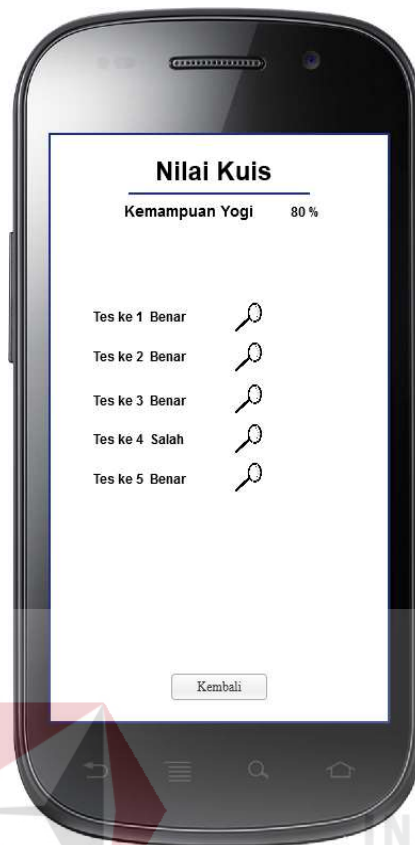
Pada menu tes kemampuan, *user* awalnya harus memilih salah satu dari 28 huruf arab seperti pada Gambar 3.55, kemudian *user* harus menuliskan jawaban sesuai huruf yang dipilih, misalkan *user* memilih huruf alif, maka *user* menuliskan huruf arab alif sebanyak lima kali pada *canvas* seperti terlihat pada Gambar 3.56. Pada akhir evaluasi sistem akan mengkalkulasi nilai dari *input user*. Seperti terlihat pada Gambar 3.57.



Gambar 3.55 Daftar 28 Huruf Hijaiyah Pada Menu Tes Kemampuan

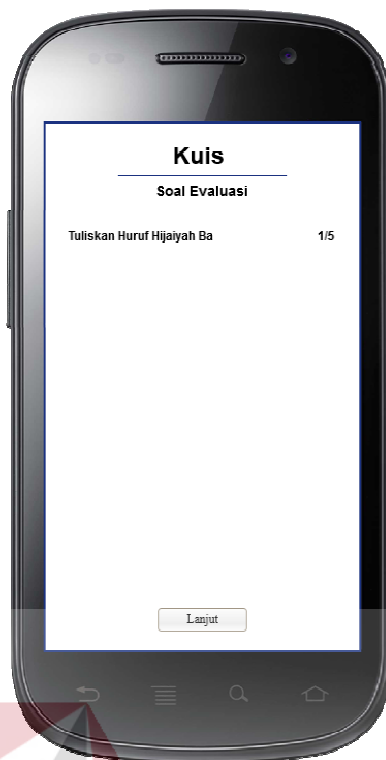


Gambar 3.56 Canvas Menulis Huruf Hijaiyah Pada Tes Kemampuan

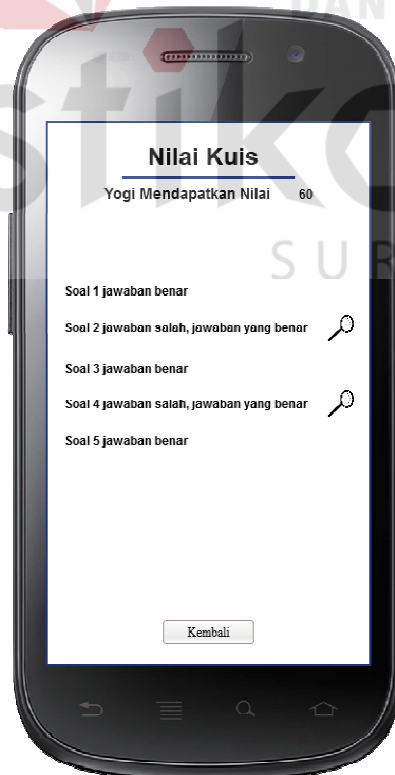


Gambar 3.57 Output Nilai Tes Uji Kemampuan

Pada kuis kedua soal evaluasi, yaitu *user* mengerjakan soal yang diberikan oleh sistem secara acak. Setelah itu, pada fitur ini disediakan sebuah *canvas* yang berguna untuk *user* dalam menuliskan jawaban atau *input gestures*. Kemudian *user* menuliskan jawaban huruf arab sesuai dengan soal yang diberikan oleh sistem sebanyak lima kali. Untuk lebih jelasnya, seperti yang terlihat pada gambar 3.58. Ketika *user* telah menuliskan jawaban sebanyak lima kali, sistem akan menampilkan hasil dari jawaban *user*, karena sistem melakukan proses kalkulasi nilai benar dan salah, kemudian memberikan *output* berupa hasil nilai pengerjaan *user* dalam menjawab soal. Seperti terlihat pada Gambar 3.59.



Gambar 3.58 Soal Evaluasi Mengerjakan Soal.



Gambar 3.59 Output Hasil Nilai Soal Evaluasi.

### 3.3.7 Tampilan Nilai

Pada aplikasi menulis huruf hijaiyah, *user* dapat melihat *history* nilai Tes Kemampuan atau soal evaluasi yang dilakukan dan telah disimpan oleh sistem, seperti terlihat pada Gambar 3.60.



Gambar 3.60 Tampilan Menu Nilai

### 3.4. Desain uji coba

Aplikasi yang dirancang dan diimplementasikan harus diuji untuk mengetahui tingkat keberhasilan dari pemakai aplikasi tersebut. Untuk melakukan uji coba pada aplikasi yang telah dibuat maka digunakan beberapa data seperti data *user*, data huruf hijaiyah dan kuis yang dimasukkan sebagai data masukan. Data yang dimasukkan tersebut bertujuan untuk menganalisa hasil *output* yang ditampilkan oleh sistem, apakah telah sesuai dengan tujuan pembuatan aplikasi menulis huruf arab atau masih memerlukan perbaikan.

Untuk aplikasi menulis huruf arab, pengujian akan dilakukan dengan pendekatan metode *black box testing*. Pengujian *black box* adalah pengujian aspek *fundamental* sistem tanpa memperhatikan struktur logika internal perangkat lunak. Metode ini digunakan karena aplikasi menulis huruf arab memiliki beberapa fungsi yang berjalan secara otomatis dibalik layar. Pengujian *black box* merupakan metode perancangan uji coba yang didasarkan pada spesifikasi perangkat lunak yang dibuat. Adapun hal-hal yang akan disajikan pada Tabel 3.9

Tabel 3.9 Rencana Pengujian Aplikasi Menulis Huruf Arab

| Requirement yang diuji            | Fungsi yang diuji  | Keterangan   |
|-----------------------------------|--|--|
| <b>Login dan Daftar User</b>      | Mengelola <i>login user</i> dan menampilkan nama <i>user</i> | Terkait dengan fungsional A, fitur ini diuji untuk membuktikan bahwa <i>user</i> tidak dapat masuk ke menu utama aplikasi bila tidak mempunyai <i>user login</i> . Kemudian ketika <i>user</i> telah mempunyai <i>user login</i> , maka <i>user</i> dapat masuk ke menu utama aplikasi. <i>User login</i> ini berguna untuk melihat perkembangan dalam latihan menulis huruf hijaiyah karena sistem dapat menyimpan hasil nilai dari latihan menulis <i>user</i> . |
| <b>Pengenalan huruf arab</b>      | Mengelola data pengenalan arab                               | Terkait dengan fungsional B, fitur ini juga dibutuhkan <i>user</i> sebagai pengetahuan awal yaitu mengenal tata cara menulis huruf hijaiyah sesuai dengan kaidah beserta pelafalannya. Hal ini sangat bermanfaat untuk <i>user</i> awam atau pemula yang belum bisa menulis huruf hijaiyah, sehingga <i>user</i> dapat melihat tata cara menulis huruf hijaiyah sesuai dengan kaidah dan diharapkan <i>user</i> bisa melakukan latihan menulis huruf hijaiyah.     |
| <b>Latihan menulis huruf arab</b> | Mengelola data pola arab                                     | Terkait dengan fungsional C, latihan menulis huruf hijaiyah sangat membantu <i>user</i> dalam melakukan latihan menulis huruf hijaiyah. Maka dari itu, diperlukan pembuktian apakah  |

| Requirement yang diuji       | Fungsi yang diuji       | Keterangan   |
|------------------------------|-------------------------|--|
|                              |                         | pencocokan <i>gestures</i> sudah sesuai dengan yang diharapkan, yaitu tulisan hasil <i>user</i> apakah sesuai dengan tata cara penulisan sesuai dengan kaidah. Pada fitur ini juga terdapat prosentase benar dari jumlah <i>user</i> melakukan latihan menulis.  |
| <b>Mengerjakan soal kuis</b> | Mengelola data evaluasi | Terkait dengan fungsional D, mengerjakan soal kuis, fitur ini berguna untuk mengetahui kemampuan <i>user</i> dalam menulis huruf hijaiyah. Maka dari itu, diperlukan pembuktian apakah nilai yang keluar sesuai dengan latihan yang dilakukan oleh <i>user</i> . Jika tulisan <i>user</i> benar maka sistem juga akan mendeteksi benar dan juga sebaliknya, jika hasil tulisan <i>user</i> salah sistem akan mendeteksi salah. |
| <b>Nilai</b>                 | Mengelola data kontrak  | Terkait dengan fungsional E, fitur nilai juga sangatlah penting untuk melihat perkembangan <i>user</i> dalam menulis huruf hijaiyah. Semua hasil latihan kuis yang dilakukan <i>user</i> tersimpan pada fitur nilai. Maka dari itu perlu pembuktian apakah hasil dari <i>user</i> melakukan latihan kuis sama dengan yang ditampilkan pada fitur nilai.  |

Pengujian aplikasi juga berdasarkan tabel fungsional yaitu pada tabel 3.2.

Di sini akan diberikan penjelasan testing yang akan diuji dengan tabel fungsional.

Berikut gambaran pada tabel 3.10.

Tabel 3.10 Gambaran Tes Case ID dengan Index Fungsional

| Index Tabel Fungsional | Test Case ID |
|------------------------|--------------|
| A1                     | 1            |
| A3                     | 2            |
| A2, A3                 | 3            |
| A3                     | 4            |
| B1                     | 5            |
| B2                     | 6            |



| Index Tabel Fungsional | <i>Test Case ID</i> |
|------------------------|---------------------|
| B2                     | 7                   |
| C1, C2, C3             | 8                   |
| C1, C2, C3             | 9                   |
| D1, D3                 | 10                  |
| D2, D3                 | 11                  |
| D4                     | 12                  |
| E1, E2                 | 13                  |
| E1, E2                 | 14                  |
| E3                     | 15                  |

### 3.4.1 Uji Coba Fungsi Aplikasi

Tahap uji coba dilakukan untuk mengetahui apakah fungsi-fungsi yang ada pada aplikasi menulis huruf arab berjalan sesuai dengan yang diharapkan. Setiap fitur yang disediakan akan diuji hasilnya sesuai dengan tabel *test case*. Desain uji coba fungsi aplikasi adalah sebagai berikut:

#### A. Desain Uji Coba *Login* dan *Daftar User*

Pada desain uji coba proses *login* dan *daftar user* bertujuan untuk mengetahui apakah proses masuk aplikasi ataupun menambah *user* tersebut dapat berfungsi dengan baik. Uji coba ini dilakukan dengan *login* tanpa *user login* untuk pembuktian bahwa *user* tidak akan dapat mengakses menu utama atau sebaliknya bila *user* mempunyai *user login* seharusnya dapat mengakses menu utama aplikasi, kemudian menambah *user login* diharapkan nama *user* baru dapat tampil di *daftar user*. Desain uji coba *login* dan *daftar user* dapat dilihat pada Tabel 3.11.

Tabel 3.11 Desain Uji Coba *Daftar User*

| <i>Test Case ID</i> | Tujuan                    | <i>Input</i>  | <i>Output yang diharapkan</i>                     | Status |
|---------------------|---------------------------|---|---|--------|
| 1.                  | Menambah Nama <i>User</i> | Pilih menu buat baru, ke kemudian masukkan nama <i>user</i> | Nama <i>user</i> tampil pada menu utama aplikasi. |        |

| <i>Test Case ID</i> | <i>Tujuan</i>  | <i>Input</i>                    | <i>Output yang diharapkan</i>                                | <i>Status</i> |
|---------------------|--|---------------------------------|--|---------------|
|                     |  | dan pilih tombol mulai          |  |               |
| 2.                  | Masuk menu lanjut dengan nama <i>user</i> yang dipilih | Pilih Nama <i>User</i>          | Tampil nama <i>User</i> yang dipilih pada halaman menu utama |               |
| 3                   | Menampilkan nama <i>user</i> yang tersimpan            | Masuk menu lanjut               | Tampil nama <i>user</i> yang sudah ada.                      |               |
| 4                   | Masuk aplikasi tanpa ada <i>user login</i>             | Pilih menu lanjut <i>user</i> . | Tidak dapat masuk ke dalam menu utama aplikasi.              |               |

### B. Desain Uji Pengenalan Huruf Hijaiyah/ Arab

Pada desain uji coba pengenalan huruf arab bertujuan untuk mengetahui apakah pengenalan huruf arab dapat berfungsi dengan baik. Uji coba ini dilakukan dengan masuk ke menu pengenalan hijaiyah kemudian memilih salah satu huruf hijaiyah. Setelah memilih salah satu huruf hijaiyah mencoba fitur *play*, *stop* dan *sound*. Diharapkan pengenalan tata cara menulis hijaiyah sesuai dengan pilihan huruf yang telah dipilih *user*, serta muncul tata cara penulisan yang sesuai dengan kaidah. Desain uji coba huruf hijaiyah bisa dilihat pada tabel 3.12.

Tabel 3.12 Desain Uji Coba Event

| <i>Test Case ID</i> | <i>Tujuan</i>   | <i>Input</i>   | <i>Output yang Diharapkan</i>  | <i>Status</i> |
|---------------------|---|--|--|---------------|
| 5                   | Mengecek kesesuaian pilihan <i>user</i> memilih huruf arab. | Sentuhan jari pada salah satu pilihan dari 28 huruf arab . | Muncul tata cara menulis huruf arab sesuai dengan pilihan <i>user</i> .      |               |
| 6                   | Cek fungsi lihat tata cara penulisan huruf hijaiyah         | Pilih tombol play  | Muncul tata cara penulisan huruf arab yang dipilih <i>user</i> dengan gambar |               |

| <i>Test Case ID</i> | <i>Tujuan</i>   | <i>Input</i>                                  | <i>Output yang Diharapkan</i>  | <i>Status</i> |
|---------------------|---|---|--|---------------|
|                     |   |   | bergerak.  |               |
| 7                   | Cek fungsi lihat tata cara penulisan tanpa gambar gerak | Pilih tombol play kemudian pilih tombol stop. | Muncul tata cara penulisan huruf arab berupa gambar tapi tidak bergerak. |               |

### C. Desain Uji Coba Latihan Menulis Huruf Hijaiyah/ Arab

Pada desain uji coba latihan menulis huruf arab bertujuan untuk mengetahui apakah menu menulis menulis huruf hijaiyah dapat berfungsi dengan baik. Uji coba ini dilakukan dengan masuk ke menu latihan menulis hijaiyah kemudian memilih salah satu huruf hijaiyah. Setelah memilih salah satu huruf hijaiyah *user* mencoba memasukkan *gestures* atau tulisan tangan. Diharapkan sistem dapat mengecek benar atau salah *gestures* dari *input user* yang sesuai dengan kaidah. Desain uji coba huruf hijaiyah bisa dilihat pada tabel 3.13.

Tabel 3.13 Desain Uji Coba Menulis Huruf Arab

| <i>Test Case ID</i> | <i>Tujuan</i>                                     | <i>Input</i>  | <i>Output yang Diharapkan</i>                         | <i>Status</i> |
|---------------------|---|---|---|---------------|
| 8                   | Cek fungsi latihan menulis huruf arab bila benar. | Masuk pada menu latihan menulis hijaiyah, lalu memasukkan tulisan huruf arab/ <i>gestures</i> dengan sengaja benar, sebanyak lima kali. | Prosentase benar 100% <i>input</i> dari <i>user</i> . |               |
| 9                   | Cek fungsi latihan menulis huruf arab bila salah. | Memilih menu latihan menulis hijaiyah, lalu memasukkan tulisan huruf arab/ <i>gestures</i> dengan sengaja salah, sebanyak lima kali.    | Prosentase benar 0% <i>input</i> dari <i>user</i> .   |               |

#### D. Desain Uji Coba Kuis Menulis Huruf Hijaiyah/ Arab

Pada desain uji coba kuis menulis huruf arab bertujuan untuk mengetahui apakah menu kuis menulis menulis huruf hijaiyah dapat berfungsi dengan baik. Uji coba ini dilakukan dengan masuk ke menu latihan menulis hijaiyah kemudian memilih salah satu huruf hijaiyah. Setelah memilih salah satu huruf hijaiyah *user* mencoba memasukkan *gestures* atau tulisan tangan. Diharapkan sistem dapat mengecek benar atau salah *gestures* dari *input user* yang sesuai dengan kaidah dan menampilkan info nilai dari latihan *user* dalam pengerjaan kuis. Desain uji coba huruf hijaiyah bisa dilihat pada tabel 3.14.

Tabel 3.14 Desain Uji Coba Kuis Menulis Huruf Arab

| <i>Test Case ID</i> | <i>Tujuan</i>   | <i>Input</i>   | <i>Output yang Diharapkan</i>  | <i>Status</i> |
|---------------------|---|--|--|---------------|
| 10                  | Cek fungsi kuis menulis huruf arab pada tes kemampuan sebanyak lima kali. | Memilih menu kuis soal evaluasi kemudian memasukkan tulisan huruf arab/ <i>gestures</i> sesuai dengan pilihan <i>user</i> , sebanyak lima kali.            | <i>User</i> dapat menuliskan huruf pada <i>canvas</i> sebanyak lima kali.  |               |
| 11                  | Cek fungsi kuis menulis huruf arab pada soal evaluasi sebanyak lima kali. | Memilih menu kuis soal evaluasi kemudian memasukkan tulisan huruf arab/ <i>gestures</i> sesuai dengan soal yang diberikan oleh sistem, sebanyak lima kali. | Prosentase benar 0% <i>input</i> dari <i>user</i> . Di contohkan huruf ba' |               |
| 12.                 | Cek hasil nilai setelah <i>user</i> melakukan kuis.                       | Memasukkan tulisan huruf arab pada soal kuis.  | Nilai dari latihan <i>user</i> dalam mengerjakan soal kuis                 |               |

#### E. Desain Uji Coba Penilaian

Pada desain uji coba penilaian bertujuan untuk mengetahui apakah menu menulis menulis huruf hijaiyah dapat berfungsi dengan baik. Uji coba ini

dilakukan dengan masuk ke menu nilai kemudian memilih salah satu kuis tes kemampuan atau soal evaluasi. Setelah memilih salah satu kuis, diharapkan sistem dapat menampilkan histori nilai sesuai dengan kuis yang telah dilakukan oleh user. Desain uji coba penilaian bisa dilihat pada tabel 3.15.

Tabel 3.15 Desain Uji Coba Nilai

| <i>Test Case ID</i> | <i>Tujuan</i>  | <i>Input</i>  | <i>Output Yang Diharapkan</i>   | <i>Status</i> |
|---------------------|--|---|---|---------------|
| 13                  | Cek fungsi penilaian pada kuis tes kemampuan               | Memasukkan tulisan huruf arab/ <i>gestures</i> huruf alif secara acak benar/salah | Muncul kalkulasi nilai yang telah dilakukan di kuis   |               |
| 14                  | Cek fungsi kesesuaian nilai yang tersimpan                 | Memilih Menu nilai  | Muncul kalkulasi nilai tes kemampuan dan hasilnya sama dengan nilai yang telah dilakukan.   |               |
| 15                  | Cek fungsi kesesuaian <i>input gestures</i> yang tersimpan | Memilih Menu nilai  | <i>Gestures</i> yang tersimpan pada fitur nilai sesuai dengan <i>input gestures</i> yang dilakukan <i>user</i> ketika melakukan kuis tes kemampuan. |               |

### 3.4.2 Uji Coba Fungsi Aplikasi Pencocokan

Pada proses uji coba ini, panelis mencoba kemampuan aplikasi dalam melakukan proses pencocokan *gestures* huruf arab. Hal yang dilakukan adalah mencoba menulis setiap huruf arab mulai dari alif sampai ya sebanyak sepuluh kali. Dari uji coba tersebut dapat dilihat salah dan benar dari total uji coba 28 huruf arab. Berikut tabel pencocokan pada tabel 3.16.

Tabel 3.16 Hasil Uji Coba Pencocokan 28 Huruf Arab

| No | Huruf Arab | Prosentase Pencocokan |       |
|----|------------|-----------------------|-------|
|    |            | Benar                 | Salah |
| 1. | 'Alif      |                       |       |
| 2. | Baa'       |                       |       |
| 3. | Taa'       |                       |       |

| No     | Huruf Arab | Prosentase Pencocokan |       |
|--------|------------|-----------------------|-------|
|        |            | Benar                 | Salah |
| 4.     | Tsaa'      |                       |       |
| 5.     | Jiim       |                       |       |
| 6.     | Haa'       |                       |       |
| 7.     | Kho        |                       |       |
| 8.     | Daal       |                       |       |
| 9.     | (Dzh)aal   |                       |       |
| 10.    | Raa'       |                       |       |
| 11.    | Zaayn      |                       |       |
| 12.    | Siin       |                       |       |
| 13.    | Shiin      |                       |       |
| 14.    | Saad       |                       |       |
| 15.    | Daad       |                       |       |
| 16.    | Taa'       |                       |       |
| 17.    | (Zh)aa'    |                       |       |
| 18.    | 'Ayn       |                       |       |
| 19.    | Ghayn      |                       |       |
| 20.    | Faa'       |                       |       |
| 21.    | Qaaf       |                       |       |
| 22.    | Kaaf       |                       |       |
| 23.    | Laam       |                       |       |
| 24.    | Miim       |                       |       |
| 25.    | Nuun       |                       |       |
| 26.    | Haa'       |                       |       |
| 27.    | Waaw'      |                       |       |
| 28.    | Yaa'       |                       |       |
| Jumlah |            |                       |       |

### 3.4.3 Uji Coba Fungsi Aplikasi Terhadap Pengguna

Pada proses uji coba ini, *user* bisa melihat tata cara penulisan huruf arab sesuai dengan kaidah, kemudian *user* dapat melakukan latihan kuis dan melihat histori nilai dari hasil latihan mengerjakan kuis. Uji coba ini dilakukan dengan cara mencoba aplikasi menulis huruf arab langsung kepada *user*.

Pada Tabel 3.17 adalah uji coba tes kemampuan untuk 10 *user*. Masing-masing akan melakukan latihan sebanyak lima kali. Berikut bentuk tabel *user* yang akan melakukan uji coba beserta status dari *user*.

Tabel 3.17 Daftar User yang akan melakukan uji coba

| Nama User     | Status |
|---------------|--------|
| <b>Arie</b>   |        |
| <b>Dika</b>   |        |
| <b>Yuli</b>   |        |
| <b>Inggit</b> |        |
| <b>Atik</b>   |        |
| <b>Ismail</b> |        |
| <b>Izzah</b>  |        |
| <b>Fadila</b> |        |
| <b>Deni</b>   |        |
| <b>Ardi</b>   |        |

Pada tabel 3.18 adalah bentuk tabel untuk hasil nilai dari uji coba 10 *user* yang melakukan latihan tes kemampuan. Berikut adalah bentuk tabel hasil nilai dari masing-masing 10 *user*.

Tabel 3.18 Hasil Nilai Uji Coba Tes Kemampuan

| User          | Huruf | Nilai |    |     |    |   |
|---------------|-------|-------|----|-----|----|---|
|               |       | I     | II | III | IV | V |
| <b>Arie</b>   |       |       |    |     |    |   |
| <b>Dika</b>   |       |       |    |     |    |   |
| <b>Yuli</b>   |       |       |    |     |    |   |
| <b>Inggit</b> |       |       |    |     |    |   |
| <b>Atik</b>   |       |       |    |     |    |   |
| <b>Ismail</b> |       |       |    |     |    |   |
| <b>Izzah</b>  |       |       |    |     |    |   |
| <b>Fadila</b> |       |       |    |     |    |   |
| <b>Deni</b>   |       |       |    |     |    |   |
| <b>Ardi</b>   |       |       |    |     |    |   |