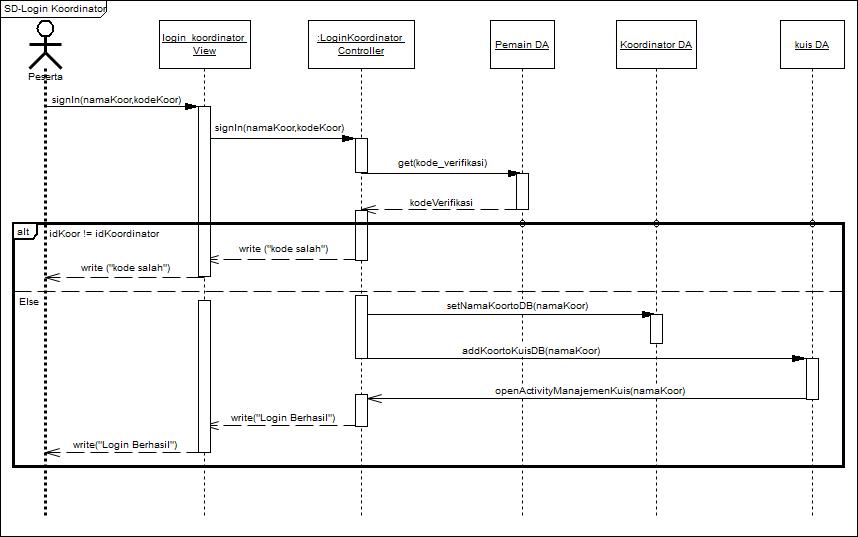
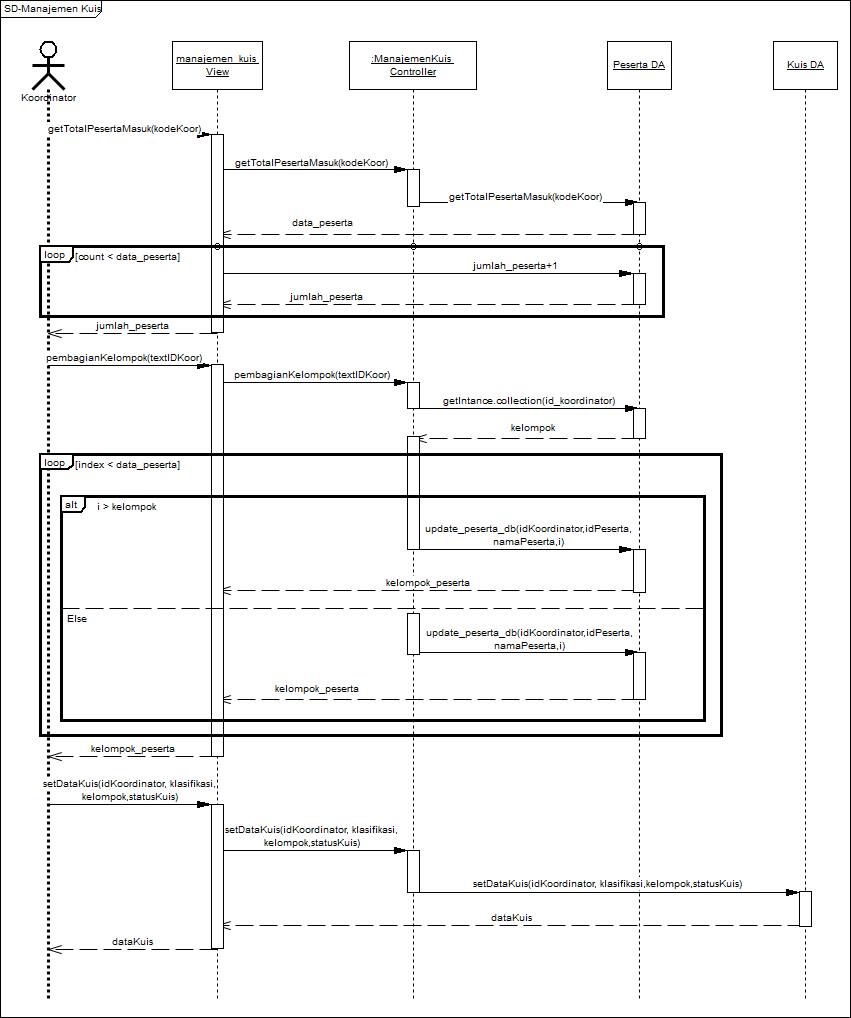
**SSD Manajemen Kuis**

**SD – Login Koordinator**

****

|  |  |
| --- | --- |
| **ID *Sequence diagram*** |  |
| **Nama *Sequence diagram*** | SD-Login Koordinator |
| **Nama *Class* Terkait** | :Login\_koordinator View, :LoginKoordinator Controller, :pemain Model,:kuis DA, :koordinator Model, :Koordinator DA |
| **Deskripsi** | Sequence ini digunakan untuk menjelaskan proses login Koordinator |
| ***Method* yang terkait** | signIn(namaKoor, IdKoor)  get(kodeVerifikasi)  getKode\_verifikasi()  generate\_id\_koor(lenght)  setNamaKoortoDB(namaKoor)  Koordinator(id\_koordinator,namaKoor)  openActivityManajemenKuis(namaKoor) |
| ***Sequence diagram* yang terkait** |  |
| **Algoritma**  Kamus data  namaKoor = variabel tunggal bertipe string  kodeKoor = variabel tunggal bertipe string  lenght\_generate = variabel tunggal bertipe integer  data\_random= variabel tunggal bertipe array of character  IdKoor = variabel tunggal bertipe string  **PROCEDURE** signIn (namaKoor, kodeKoor)  **BEGIN**  **IF** (namaKoor !=NULL) **THEN**  **IF** (kodeKoor != NULL) **THEN**  query 🡨 $this->db->query(“SELECT \* FROM koordinator WHERE id\_verifikasi = kodeKoor)  **IF** (query.result != NULL) THEN  WRITE (Layar) *“Login Berhasil”*  FUNCTION generate\_id\_koor (4)  setNamaKoortoDB (namaKoor)  **ELSE**  WRITE (Layar) *“Kode Salah”*  **END IF**  **ELSE**  WRITE (Layar) *“Kode harus diisi”*  **END IF**  **ELSE**  **IF** (kodeKoor != NULL) **THEN**  WRITE (Layar) *“Nama harus diisi”*  **ELSE**  WRITE (Layar) *“Nama dan Kode harus diisi”*  **END IF**  **END IF**  **END PROCEDURE**  **FUNCTION** generate\_id\_koor (lenght\_generate)  **BEGIN**  data\_random 🡨 “abcdefghijklmnopqrstuvwxyz1234567”  id\_koordinator 🡨 RANDOM (data\_random, lenght\_generate)  **return** id\_koordinator  **END FUNCTION**  **FUNCTION** setNamaKoortoDB (namaKoor)  **BEGIN**  Koordinator 🡨 array (  ‘Id\_koordinator’ ← $this->post (idKoor)  ‘namaKoor← $this->post (namaKoor)  )  $this->db->insert(‘koordinator’, Koordinator)  **END FUNCTION** | |

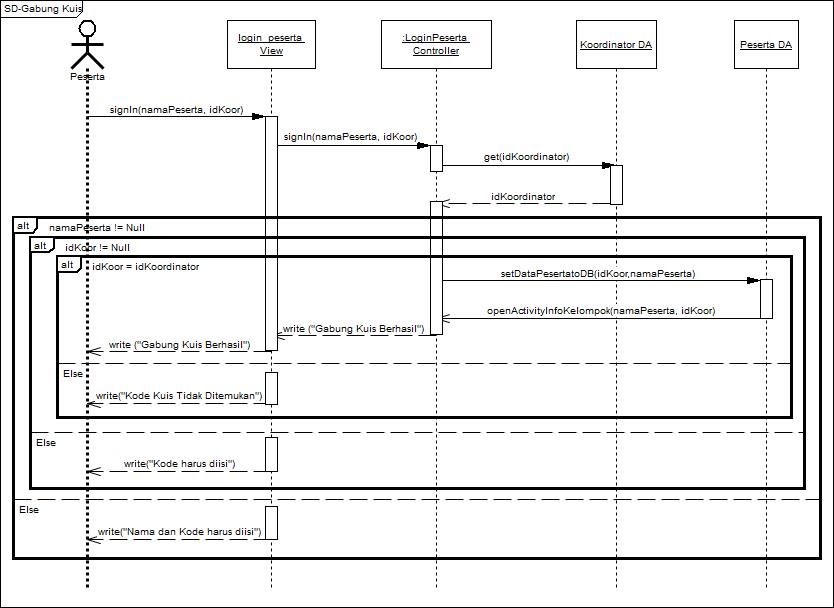
**SD – Manajemen Kuis**

****

|  |  |
| --- | --- |
| **ID *Sequence diagram*** |  |
| **Nama *Sequence diagram*** | SD-Manajemen Kuis |
| **Nama *Class* Terkait** | :Manajemen\_kuis View, :ManajemenKuis Controller, :peserta Model, :Peserta DA, :Kuis Model, :Kuis DA |
| **Deskripsi** | Sequence ini digunakan untuk menjelaskan proses pembagian kelompok dan pemilihan klasifikasi kuis |
| ***Method* yang terkait** | getTotalPesertaMasuk(kodeKoor)  getKelompok(kelompok)  pembagianKelompok(textIDKoor)  Peserta(id\_koordinator,id\_peserta,nama\_peserta)  getId\_koordinator(id\_koordinator)  update\_peserta\_db(idKoordinator,idPeserta,namaPeserta,i)  setDataKuis(idKoordinator,klasifikasi,kelompok,statusKuis)  Kuis(id\_koordinator,jumlah\_kelompok,jenis\_klasifikasi,status\_kuis) |
| ***Sequence diagram* yang terkait** |  |
| **Algoritma**  Kamus data  klasifikasi = variabel tunggal bertipe string  kelompok = variabel tunggal bertipe integer  statusKuis = variabel tunggal bertipe string  idKoordinator = variabel tunggal bertipe array of character  IdPeserta = variabel tunggal bertipe string  jumlah\_peserta = variabel tunggal bertipe integer  **READ** (Keyboard) klasifikasi  **READ** (Keyboard) kelompok  statusKuis 🡨 “Belum Mulai”  **PROCEDURE** getTotalPesertaMasuk (kodeKoor)  **BEGIN**  jumlah\_peserta 🡨 0  data\_peserta 🡨 $this->db->query(“SELECT \* FROM peserta WHERE id\_koordinator = idKoordinator)  **FOR**(count as data\_peserta) **DO**  jumlah\_peserta 🡨 jumlah\_peserta + 1  **END FOR**  **Return** jumlah\_peserta  **END PROCEDURE**  **PROCEDURE** pembagianKelompok (textIDKoor)  **BEGIN**  data\_peserta 🡨 $this->db->query(“SELECT \* FROM peserta WHERE id\_koordinator = idKoordinator)  i 🡨 1 *//variabel tunggal bertipe integer*  **FOR** (index as data\_peserta) **DO**  **IF** (i > kelompok) **THEN**  i 🡨 1  idPeserta 🡨 index.id\_peserta  kelompok\_peserta 🡨 i  $this->db->query(“UPDATE peserta SET kelompok = kelompok\_peserta WHERE id\_peserta = ‘idPeserta’”)  i++  **ELSE**  idPeserta 🡨 index.id\_peserta  kelompok\_peserta 🡨 i  $this->db->query(“UPDATE peserta SET kelompok = kelompok\_peserta WHERE id\_peserta = ‘idPeserta’”)  i++  **END IF**  **END FOR**  **END PROCEDURE**  **PROCEDURE** setDataKuistoDB()  **BEGIN**  dataKuis 🡨 array (  ‘id\_koordinator’ 🡨 idKoordinator  ‘jenis\_klasifikasi’ 🡨 $this->post(klasifikasi)  ‘jumlah\_kelompok’ 🡨 $this->post(kelompok)  ‘status\_kuis’ 🡨 statusKuis  )  $this->db->insert(‘kuis’, dataKuis)  **END PROCEDURE** | |

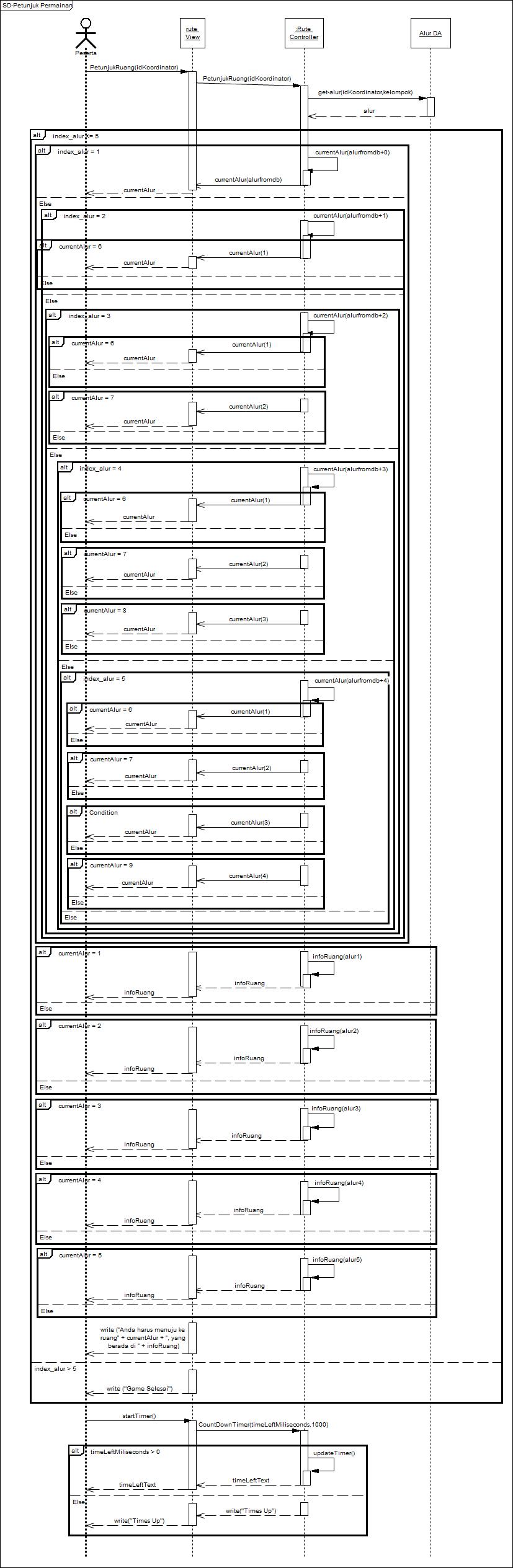
**SSD Manajemen Kuis**

**SD - Gabung Kuis**

****

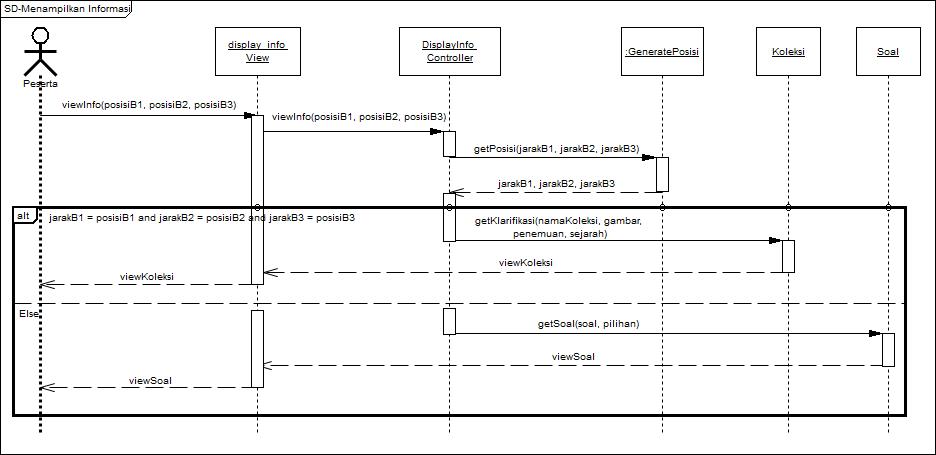
|  |  |
| --- | --- |
| **ID *Sequence diagram*** |  |
| **Nama *Sequence diagram*** | SD-Gabung Kuis |
| **Nama *Class* Terkait** | :login\_peserta View, :LoginPeserta Controller, :Koordinator Model, :Koordinator DA, :Peserta Model, :Peserta DA |
| **Deskripsi** | Sequence ini digunakan untuk menjelaskan proses Gabung Kuis bagi peserta |
| ***Method* yang terkait** | signIn(namaPeserta,idKoor)  Koordinator(id\_koordinator,namaKoor)  getId\_koordinator()  setDataPesertatoDB(idKoor,namaPeserta)  Peserta(id\_koordinator,id\_peserta,nama\_peserta)  openActivityKelompok(namaPeserta(idKoor) |
| ***Sequence diagram* yang terkait** |  |
| **Algoritma**  Kamus data  namaPeserta = variabel tunggal bertipe string  idKoor = variabel tunggal bertipe string  lenght\_generate = variabel tunggal bertipe integer  data\_random= variabel tunggal bertipe array of character  IdPeserta = variabel tunggal bertipe string  **PROCEDURE** signInPeserta (namaPeserta, idKoor)  **BEGIN**  **IF** (namaPeserta!=NULL) **THEN**  **IF** (idKoor!= NULL) **THEN**  query 🡨 $this->db->query(“SELECT \* FROM koordinator WHERE id\_koordinator = idKoor)  **IF** (query.result != NULL) THEN  WRITE (Layar) *“Gabung Kuis Berhasil”*  FUNCTION generate\_id\_peserta(4)  setDataPesertatoDB (namaPeserta, idKoor)  **ELSE**  WRITE (Layar) *“Kode Kuis Tidak Ditemukan”*  **END IF**  **ELSE**  WRITE (Layar) *“Kode harus diisi”*  **END IF**  **ELSE**  **IF** (idKoor!= NULL) **THEN**  WRITE (Layar) *“Nama harus diisi”*  **ELSE**  WRITE (Layar) *“Nama dan Kode harus diisi”*  **END IF**  **END IF**  **END PROCEDURE**  **FUNCTION** generate\_id\_peserta (lenght\_generate)  **BEGIN**  data\_random 🡨 “abcdefghijklmnopqrstuvwxyz1234567”  IdKoor 🡨 RANDOM (data\_random, lenght\_generate)  **return** idPeserta  **END FUNCTION**  **PROCEDURE** setDataPesertatoDB (namaPeserta, idKoor)  **BEGIN**  Peserta 🡨 array (  ‘id\_koordinator’ 🡨 $this->post (idKoor)  ‘id\_peserta’ 🡨 $this->post (idPeserta)  ‘namaPeserta 🡨 $this->post (namaPeserta)  )  $this->db->insert(‘peserta’, Peserta)  **END PROCEDURE** | |

**SD-Petunjuk Kuis**

****

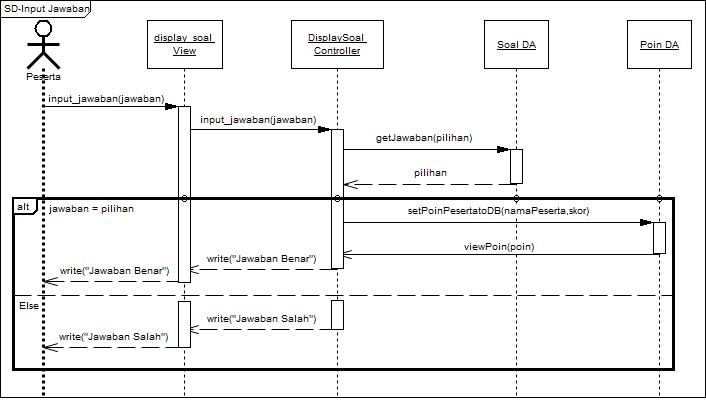
|  |  |
| --- | --- |
| **ID *Sequence diagram*** |  |
| **Nama *Sequence diagram*** | SD Petunjuk |
| **Nama *Class* Terkait** | Rute View, :Rute Controller, Alur DA |
| **Deskripsi** | Sequence ini digunakan untuk menjelaskan proses Petunjuk Kuis yang dijalankan |
| ***Method* yang terkait** | PetunjukRuang(idKoordinator)  get-alur(idKoordinator,kelompok)  currentAlur(alurfromdb)  infoRuang(alur)  startTimer()  CountDownTimer(timeLeftMilisecconds,1000)  updateTimer() |
| ***Sequence diagram* yang terkait** |  |
| **Algoritma**  Kamus data  idKoordinator = variabel tunggal bertipe string  kelompok = variabel tunggal bertipe integer  alurfromdb = variabel tunggal bertipe integer  infoRuang = variabel tunggal bertipe string  timeLeftMiliseconds = variabel tunggal bertipe integer  timeLeftText = variabel tunggal bertipe integer  countDownTimer = variabel tunggal bertipe CountDownTimer  **PROCEDURE** PetunjukRuang(kelompok, idKoordinator)  **BEGIN**  alur 🡨 $this->db->query(“SELECT \* FROM alur WHERE id\_koordinator = idKoordinator AND kelompok = kelompok)  **FOR**(get\_alur as alur) **DO**  alurfromdb 🡨 get\_alur.alur  alurKe 🡨 get\_alur.alur  alurKe 🡨 alurKe + 1  **IF**(alurKe <=5) **THEN**  alur1 🡨 “Lantai 1 sebelah kiri pintu masuk ke museum”  alur2 🡨 “Lantai 1 sebelah kanan pintu masuk ke museum”  alur3 🡨 “Lantai 2 sebelah kanan pintu masuk ke museum”  alur4 🡨 “Lantai 2 sebelah kiri pintu masuk ke museum”  alur5 🡨 “Lantai 1 sebelah kanan pintu masuk ke museum”  **IF**(alurfromdb = 1) **THEN**  infoRuang = alur1  **END IF**  **IF**(alurfromdb = 2) **THEN**  infoRuang = alur2  **END IF**  **IF**(alurfromdb = 3) **THEN**  infoRuang = alur3  **END IF**  **IF**(alurfromdb = 4) **THEN**  infoRuang = alur4  **END IF**  **IF**(alurfromdb = 5) **THEN**  infoRuang = alur5  **END IF**  **WRITE** (Layar) “Anda harus menuju ke ruang” + currentAlur + “, yang berada di ” + infoRuang  Alurfromdb 🡨 alurfromdb + 1  **IF**(alurfromdb = 6) **THEN**  Alurfromdb 🡨 1  **END IF**  Update\_alur 🡨 array (  ‘id\_koordinator’ 🡨 idKoordinator  ‘kelompok’ 🡨 kelompok  ‘alur’ 🡨 alurfromdb  ‘alur\_ke’ 🡨 alurKe  )  $this->db->insert(‘alur’, Update\_alur)  **ELSE IF**(alurKe > 5) **THEN**  WRITE (Layar) “Game Selesai”  //pindah layout ke ranking kuis  **END IF**  **END FOR**  **END PROCEDURE**  **PROCEDURE** startTimer()  **BEGIN**  countDownTimer 🡨 CountDownTimer(timeLeftMiliseconds, 1000)  **IF** (timeLeftMiliseconds > 0) **THEN**  timeLeftMiliseconds 🡨 1  **PROCEDURE** updateTimer()  **ELSE**  **WRITE** (Layar) “Times Up”  //pindah layout ke layout pengecekan ruang  **END IF**  **END FOR**  **END PROCEDURE**  **PROCEDURE** updateTimer()  **BEGIN**  Minutes = (timeLeftMiliseconds **DIV** 1000) **DIV** 60  Seconds = (timeLeftMiliseconds **DIV** 1000) **MOD** 60  **IF** (seconds < 10) **THEN**  timeLeftText += seconds  **END IF**  **return** timeLeftText  **END PROCEDURE** | |
|  | |

**SD – Menampilkan Informasi**

****

|  |  |
| --- | --- |
| **ID *Sequence diagram*** |  |
| **Nama *Sequence diagram*** | SD-Menampilkan Informasi |
| **Nama *Class* Terkait** | display\_info View, :DisplayInfo Controller, :GeneratePosisi, Koleksi DA, Soal DA |
| **Deskripsi** | Sequence ini digunakan untuk menjelaskan proses menampilkan informasi koleksi museum |
| ***Method* yang terkait** | viewinfo (posisiB1, posisiB2, posisiB3)  getPosisi (jarakB1, jarakB2, jarakB3)  getKlasifikasi(namaKoleksi, gambar, penemuan, sejarah)  getSoal (soal,pilihan) |
| ***Sequence diagram* yang terkait** |  |
| **Algoritma**  Kamus data  posisiB1 = variabel tunggal bertipe float  posisiB2 = variabel tunggal bertipe float  posisiB3 = variabel tunggal bertipe float  jarakB1 = variabel tunggal bertipe float  jarakB2 = variabel tunggal bertipe float  jarakB3 = variabel tunggal bertipe float  timeLeftMiliseconds = variabel tunggal bertipe integer  timeLeftText = variabel tunggal bertipe integer  countDownTimer = variabel tunggal bertipe CountDownTimer  **PROCEDURE** viewInfo (posisiB1, posisiB2, posisiB3)  **BEGIN**  klasifikasi 🡨 array (  ‘posisiB1’← $this->post (jarakB1)  ‘posisiB2’← $this->post (jarakB2)  ‘posisiB3’← $this->post (jarakB3)  )  query 🡨 $this->db->query(“SELECT \* FROM koleksi WHERE jarakB1 = posisiB1 AND jarakB2 = posisiB2 AND jarakB3 = posisiB3 )  **END PROCEDURE**  **PROCEDURE** startTimer()  **BEGIN**  countDownTimer 🡨 CountDownTimer(timeLeftMiliseconds, 1000)  **IF** (timeLeftMiliseconds > 0) **THEN**  timeLeftMiliseconds 🡨 1  **PROCEDURE** updateTimer()  **ELSE**  **WRITE** (Layar) “Times Up”  //pindah layout ke layout petunjuk  **END IF**  **END FOR**  **END PROCEDURE**  **PROCEDURE** updateTimer()  **BEGIN**  Minutes = (timeLeftMiliseconds **DIV** 1000) **DIV** 60  Seconds = (timeLeftMiliseconds **DIV** 1000) **MOD** 60  **IF** (seconds < 10) **THEN**  timeLeftText += seconds  **END IF**  **return** timeLeftText  **END PROCEDURE** | |

**SD – Input Jawaban**

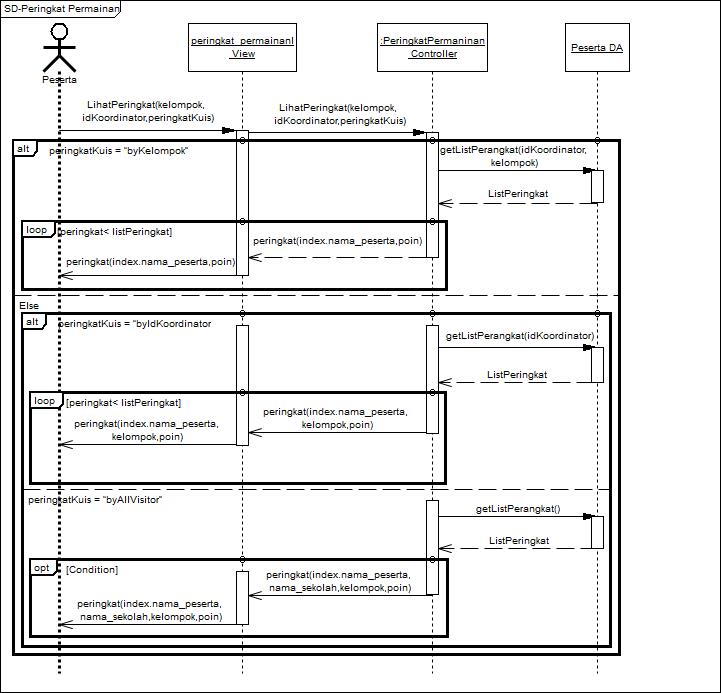
****

|  |  |
| --- | --- |
| **ID *Sequence diagram*** |  |
| **Nama *Sequence diagram*** | SD Input Jawaban |
| **Nama *Class* Terkait** | display\_soal View, DisplaySoal Controller, Soal DA, Poin DA |
| **Deskripsi** | Sequence ini digunakan untuk menjelaskan proses input jawaban soal peserta |
| ***Method* yang terkait** | input\_jawaban (jawaban)  getJawaban (pilihan)  setPoinPesertatoDB(namaPeserta,skor)  viewPoin(poin) |
| ***Sequence diagram* yang terkait** |  |
| **Algoritma**  Kamus data  jawaban = variabel tunggal bertipe string  skor = variabel tunggal bertipe integer  klasifikasi = variabel tunggal bertipe integer  jenis\_klasifikasi = variabel tunggal bertipe integer  idPeserta = variabel tunggal bertipe string  idKoor = variabel tunggal bertipe string  soal = variabel tunggal bertipe string  seconds = variabel tunggal bertipe integer  poin = variabel tunggal bertipe integer  timeLeftMiliseconds = variabel tunggal bertipe integer  timeLeftText = variabel tunggal bertipe integer  countDownTimer = variabel tunggal bertipe CountDownTimer  timeLeftMiliseconds 🡨 6000 //1 menit  **PROCEDURE** inputJawaban (klasifikasi, jenis\_klasifikasi, soal, IdPeserta, IdKoor)  **BEGIN**  **READ** (Keyboard) jawaban  **IF** (jawaban = A) **THEN**  cekJawaban(“A”, klasifikasi, jenis\_klasifikasi, soal, IdPeserta, IdKoor)  **END IF**  **IF** (jawaban = B) **THEN**  cekJawaban(“B”, klasifikasi, jenis\_klasifikasi, soal, IdPeserta, IdKoor)  **END IF**  **IF** (jawaban = C) **THEN**  cekJawaban(“C”, klasifikasi, jenis\_klasifikasi, soal, IdPeserta, IdKoor)  **END IF**  **IF** (jawaban = D) **THEN**  cekJawaban(“D”, klasifikasi, jenis\_klasifikasi, soal, IdPeserta, IdKoor)  **END IF**  **END FUNCTION**  **PROCEDURE** cekJawaban (jawaban, klasifikasi, jenis\_klasifikasi, soal, IdPeserta, IdKoor)  **BEGIN**  Cek\_soal ← $this->db->query(“SELECT \* FROM alur WHERE jenis\_klasifikasi = jenis\_klasifikasi AND id\_klasifikasi = id\_klasifikasi AND id\_soal = soal)  Var\_answer = Cek\_soal.correctAnswer  **IF** (jawaban = Var\_answer) **THEN**  WRITE (Layar) *“Jawaban Benar”*  **PROCEDURE** updatePoin(idPeserta, idKoor, seconds)  //pindah ke layout informasi koleksi  **ELSE**  WRITE (Layar) *“Jawaban Salah”*  //pindah ke layout informasi koleksi  **END IF**  **END FUNCTION**  **PROCEDURE** startTimer()  **BEGIN**  countDownTimer 🡨 CountDownTimer(timeLeftMiliseconds, 1000)  **IF** (timeLeftMiliseconds > 0) **THEN**  timeLeftMiliseconds 🡨 1  **PROCEDURE** updateTimer()  **ELSE**  **WRITE** (Layar) “Times Up”  //pindah layout ke layout petunjuk  **END IF**  **END FOR**  **END PROCEDURE**  **PROCEDURE** updateTimer()  **BEGIN**  Seconds = (timeLeftMiliseconds **DIV** 1000) **MOD** 60  **IF** (seconds < 10) **THEN**  timeLeftText += seconds  **END IF**  **return** timeLeftText  **END PROCEDURE** | |

|  |  |
| --- | --- |
| **ID *Sequence diagram*** | Update poin |
| **Nama *Sequence diagram*** |  |
| **Nama *Class* Terkait** |  |
| **Deskripsi** |  |
| ***Method* yang terkait** |  |
| ***Sequence diagram* yang terkait** |  |
| **Algoritma**  Kamus data  idPeserta = variabel tunggal bertipe string  idKoor = variabel tunggal bertipe string  poinfromKuis = variabel tunggal bertipe integer  seconds = variabel tunggal bertipe integer  **PROCEDURE** updatePoin (idPeserta, idKoor, seconds)  **BEGIN**  PoinfromDB 🡨 $this->db->query(“SELECT \* FROM peserta WHERE id\_koordinator = idKoor AND id\_peserta = idPeserta)  poinfromKuis = 1 \* 10 \* seconds  poinTotal = PoinfromDB.poin  poinTotal = poinTotal + poin  update\_poin 🡨 array (  ‘id\_koordinator’← idKoor  ‘id\_peserta’← idPeserta  ‘poin’← poinTotal  )  $this->db->insert(‘poin’, update\_poin)  **END PROCEDURE** | |

**SSD Melihat Peringkat**

**SD-Peringkat Kuis**

****

|  |  |
| --- | --- |
| **ID *Sequence diagram*** |  |
| **Nama *Sequence diagram*** | SD peringkat |
| **Nama *Class* Terkait** | Peringkat\_kuis View, PeringkatKuis Controller, Peserta DA |
| **Deskripsi** | Sequence ini digunakan untuk menjelaskan proses melihat peringkat kuis |
| ***Method* yang terkait** | LihatPeringkat(kelompok,idKoordinator,peringkatKuis)  getListPeringkat(idKoordinator,kelompok)  peringkat(index,nama\_peserta,poin)  getListPeringkat(idKoordinator)  peringkat(index,nama\_peringkat,kelompok,poin)  getListPerangkat()  peringkat(index,nama\_peserta,nama\_sekolah,poin) |
| ***Sequence diagram* yang terkait** |  |
| **Algoritma**  Kamus data  idKoordinator = variabel tunggal bertipe string  kelompok = variabel tunggal bertipe integer  peringkatKuis = variabel tunggal bertipe string  **WRITE** (Layar) “Peringkat yang ditampilkan berdasarkan :”  **READ** (Keyboard) peringkatKuis  **PROCEDURE** LihatPeringkat(kelompok,idKoordinator,peringkatKuis)  **BEGIN**  **IF**(peringkatKuis = “byKelompok”)**THEN**  listPeringkat 🡨 $this->db->query(“SELECT \* FROM peserta WHERE id\_koordinator = idKoordinator AND kelompok = kelompok ORDER BY poin ASC)  Integer i  **FOR** (peringkat as listPeringkat)  **WRITE** (Layar) i  **WRITE** (Layar) peringkat.nama\_peserta  **WRITE** (Layar) peringkat.poin  **END FOR**  **ELSE IF**(peringkatKuis = “byIdKoordinator”)**THEN**  listPeringkat 🡨 $this->db->query(“SELECT \* FROM peserta WHERE id\_koordinator = idKoordinator ORDER BY poin ASC)  Integer i  **FOR** (peringkat as listPeringkat)  **WRITE** (Layar) i  **WRITE** (Layar) peringkat.nama\_peserta  **WRITE** (Layar) peringkat.kelompok  **WRITE** (Layar) peringkat.poin  **END FOR**  **ELSE IF**(peringkatKuis = “byAllVisitor”)**THEN**  listPeringkat 🡨 $this->db->query(“SELECT \* FROM peserta ORDER BY poin ASC)  Integer i  **FOR** (peringkat as listPeringkat)  **WRITE** (Layar) i  **WRITE** (Layar) peringkat.nama\_peserta  **WRITE** (Layar) peringkat.nama\_sekolah  **WRITE** (Layar) peringkat.kelompok  **WRITE** (Layar) peringkat.poin  **END FOR**  **END IF**  **END PROCEDURE** | |

**User Interface**

**Perancangan User Interface**

Rancangan user interface yang dijelaskan pada sub bab ini dibuat untuk menggambarkan tampilan frontend yang digunakan oleh pengguna untuk berinteraksi dengan aplikasi. Rancangan ini berdasarkan system sequence diagram yang telah dibuat.

|  |  |
| --- | --- |
| **No. UI** | **GUI-U-01** |
| **Nama UI** | Tampilan halaman awal aplikasi Museum Geologi |
| **Pengguna** | Pemandu dan rombongan pengunjung Museum Geologi |
| ***Precondition*** | Pengguna telah masuk ke dalam aplikasi |
| **Gambar** | |
|  | |
| **Deskripsi** | Tampilan ini adalah tampilan saat pertama kali pertama membuka aplikasi. Pada tampilan ini menggunakan card view. User dapat memilih 2 pilihan yaitu bermain untuk memainkan kuis dan lihat peringkat untuk melihat peringkat kuis. Berikut komponen yang terdapat pada gambar dijelaskan berdasarkan angka :   1. *Card view* “Bermain” merupakan tombol yang berguna untuk berpindah ke halaman informasi lokasi kuis. 2. *Card view* “Lihat Peringkat” merupakan tombol yang berguna untuk berpindah ke halaman peringkat kuis. |

|  |  |
| --- | --- |
| **No. UI** | **GUI-U-02** |
| **Nama UI** | Tampilan halaman informasi lokasi kuis |
| **Pengguna** | Pemandu dan rombongan pengunjung Museum Geologi |
| ***Precondition*** | 1. Pengguna telah masuk ke dalam aplikasi 2. Pengguna memilih menu bermain pada tampilan halaman awal aplikasi |
| **Gambar** | |
|  | |
| **Deskripsi** | Tampilan ini adalah halaman informasi lokasi kuis aplikasi Museum Geologi. Tampilan ini menampilkan lokasi-lokasi ruangan yang ada di Museum Geologi yang akan dijadikan untuk lokasi kuis, lokasi-lokasi ruangan tersebut ditampilkan dalam bentuk cardview yang nantinya dapat dipilih untuk dilihat informasi lebih detail pada setiap ruangannya. Berikut komponen yang terdapat pada gambar dijelaskan berdasarkan angka :   1. *Card view* “Ruang Geologi Indonesia” adalah tombol yang berguna untuk berpindah ke halaman informasi detail ruang geologi indonesia. 2. *Card view* “Ruang Sumber Daya Geologi” adalah tombol yang berguna untuk berpindah ke halaman informasi detail ruang sumber daya geologi. 3. *Card view* “Ruang Manfaat dan Bencana” adalah tombol yang berguna untuk berpindah ke halaman informasi detail ruang manfaat dan bencana geologi. 4. *Card view* “Ruang Sejarah Kehidupan” adalah tombol yang berguna untuk berpindah ke halaman informasi detail ruang sejarah kehidupan. 5. Tombol “Lanjutkan” adalah tombol yang berguna untuk berpindah ke halaman pilih peran. |

|  |  |
| --- | --- |
| **No. UI** | **GUI-U-03** |
| **Nama UI** | Tampilan halaman informasi detail ruangan yang dijadikan sebagai lokasi kuis |
| **Pengguna** | Pemandu dan rombongan pengunjung Museum Geologi |
| ***Precondition*** | 1. Pengguna telah masuk ke dalam aplikasi 2. Pengguna memilih menu bermain pada tampilan halaman awal aplikasi 3. Pengguna memilih salah satu ruangan yang ada pada tampilan halaman informasi lokasi kuis |
| **Gambar** | |
|  | |
| **Deskripsi** | Tampilan ini adalah halaman informasi detail ruangan yang dijadikan sebagai lokasi kuis aplikasi Museum Geologi. Tampilan ini menampilkan semua jenis klasifikasi dan koleksi yang ada pada ruangan. Berikut komponen yang terdapat pada gambar dijelaskan berdasarkan angka :   1. Merupakan gambar lokasi ruang yang dipilih pada halaman halaman informasi lokasi kuis. 2. Merupakan isi dari informasi detail ruangan meliputi klasifikasi koleksi beserta koleksi-koleksi yang ada diruang tersebut menggunakan s*croll view*. 3. Tombol “Kembali” adalah tombol yang berguna untuk berpindah ke halaman sebelumnya yaitu halaman informasi lokasi kuis. |

|  |  |
| --- | --- |
| **No. UI** | **GUI-U-04** |
| **Nama UI** | Tampilan halaman pilih peran |
| **Pengguna** | Pemandu dan rombongan pengunjung Museum Geologi |
| ***Precondition*** | 1. Pengguna telah masuk ke dalam aplikasi 2. Pengguna memilih menu bermain pada tampilan halaman awal aplikasi 3. Pengguna menekan button lanjutkan pada tampilan halaman informasi lokasi kuis |
| **Gambar** | |
|  | |
| **Deskripsi** | Tampilan ini adalah halaman untuk user dalam memilih peran. Berikut komponen yang terdapat pada gambar dijelaskan berdasarkan angka :   1. *Card view* “Koordinator” adalah tombol yang berguna untuk memilih peran sebagai koordinator dan berpindah ke halaman verifikasi koordinator. 2. *Card view* “Peserta” adalah tombol yang berguna untuk memilih peran sebagai peserta dan berpindah ke halaman verifikasi gabung kuis. |

|  |  |
| --- | --- |
| **No. UI** | **GUI-K-01** |
| **Nama UI** | Tampilan halaman verifikasi koordinator |
| **Pengguna** | Pemandu atau pembimbing rombongan pengunjung Museum Geologi |
| ***Precondition*** | Pengguna telah memilih peran dalam kuis sebagai koordinator pada halaman pilih peran |
| **Gambar** | |
|  | |
| **Deskripsi** | Tampilan ini adalah halaman untuk koordinator dalam melakukan verifikasi untuk bermain sebagai koordinator. Berikut komponen yang terdapat pada gambar dijelaskan berdasarkan angka :   1. Form “Nama” adalah form yang digunakan untuk *input* nama peserta. 2. Form “Kode Verifikasi” adalaha form yang digunakan untuk *input* kode verifikasi untuk dapat peran sebagai koordinator. 3. Tombol “Masuk” adalah tombol yang berguna untuk melanjutkan proses verifikasi koordinator. 4. Merupakan pesan yang ditampilkan jika terjadi kesalahan pada saat melakukan verifikasi. Terdapat 4 *error,* yaitu : 5. Verifikasi gagal dikarenakan kode verifikasi yang dimasukkan salah, maka aplikasi akan memberikan pesan pemberitahuan yang berisi “Kode verifikasi yang anda masukkan salah”. 6. Verifikasi gagal dikarenakan nama koordinator dan kode verifikasi tidak diisi, maka aplikasi akan memberikan pesan pemberitahuan yang berisi “Nama dan kode verifikasi harus diisi”. 7. Verifikasi gagal dikarenakan nama koordinator tidak diisi, maka aplikasi akan memberikan pesan pemberitahuan yang berisi “Nama koordinator harus diisi”. 8. Verifikasi gagal dikarenakan kode verifikasi tidak diisi, maka aplikasi akan memberikan pesan pemberitahuan yang berisi “Kode verifikasi harus diisi”. |

|  |  |
| --- | --- |
| **No. UI** | **GUI-K-02** |
| **Nama UI** | Tampilan halaman manajemen kuis |
| **Pengguna** | User yang berperan sebagai Koordinator |
| ***Precondition*** | 1. Pengguna telah memilih peran dalam kuis sebagai koordinator pada halaman pilih peran. 2. Koordinator telah melakukan verifikasi sebagai koordinator dan berhasil. |
| **Gambar** | |
|  | |
| **Deskripsi** | Tampilan ini adalah halaman untuk koordinator dalam melakukan manajemen kuis. Pada tampilan ini koordinator dapat melakukan manajemen kuis meliputi penentuan jumlah kelompok dan penentuan jenis klasifikasi. Berikut komponen yang terdapat pada gambar dijelaskan berdasarkan angka :   1. Merupakan *text view* yang berguna untuk menampilkan nama koordinator dan id koordinator yang dijadikan kode kuis. 2. Merupakan *text view* yang berguna untuk menampilkan jumlah peserta yang masuk ke dalam kuis. 3. *Dropdown* “Jumlah Kelompok” adalah *dropdown* yang berguna untuk menentukan jumlah kelompok kuis, *dropdown* “Jumlah Kelompok” ini jika jumlah peserta yang masuk <5, maka berisi pilihan angka 1 – jumlah maksimal peserta masuk, jika >5, maka berisi pilihan angka 1-5. 4. *Dropdown* “Jenis Klasifikasi” adalah *dropdown* yang berguna untuk menentukan jumlah kelompok kuis, *dropdown* “Jenis Klasifikasi” ini berisi 4 pilihan yaitu TK, SD 1-3, SD 4-6, dan SMP. 5. Tombol “Lanjutkan” adalah tombol yang berguna untuk melanjutkan proses manajemen kuis dan berpindah ke halaman mulai kuis koordinator dan list kelompok. 6. Tombol “*Logout*” adalah tombol yang berguna untuk keluar dari peran koordinator dan berpindah kembali ke halaman pilih peran. |

|  |  |
| --- | --- |
| **No. UI** | **GUI-K-03** |
| **Nama UI** | Tampilan halaman mulai kuis koordinator dan list kelompok |
| **Pengguna** | User yang berperan sebagai Koordinator |
| ***Precondition*** | 1. Pengguna telah memilih peran dalam kuis sebagai koordinator pada halaman pilih peran. 2. Koordinator telah melakukan verifikasi sebagai koordinator dan berhasil. 3. Koordinator telah memanajemen kuis. |
| **Gambar** | |
|  | |
| **Deskripsi** | Tampilan ini adalah halaman untuk koordinator memulai kuis atau membuka akses kuis dan melihat list kelompok. *State* awal pada tampilan ini yaitu button “mulai kuis”  *enable* dan button “hentikan kuis” *disable*, jika koordinator memulai kuis dengan menekan button “mulai kuis” maka button “mulai kuis” akan *disable* dan button “hentikan kuis” *enable*. Berikut komponen yang terdapat pada gambar dijelaskan berdasarkan angka :   1. Merupakan *text view* yang berguna untuk menampilkan nama koordinator dan id koordinator yang dijadikan kode kuis. 2. Merupakan *text view* yang berguna untuk menampilkan jenis klasifikasi kuis. 3. Merupakan *linear view* yang beguna untuk menampilkan tombol kelompok 1 sampai dengan jumlah kelompok yang telah ditentukan pada saat melakukan manajemen kuis sebelumnya. 4. Tombol “Mulai Kuis” adalah tombol yang berguna untuk membuka akses kuis agar peserta dapat memainkan kuis. 5. Tombol “Hentikan Kuis” adalah tombol yang berguna untuk menutup akses kuis dan peserta tidak dapat melanjutkan kuis. 6. Tombol “*Logout*” adalah tombol yang berguna untuk keluar dari peran koordinator dan berpindah kembali ke halaman pilih peran. |

|  |  |
| --- | --- |
| **No. UI** | **GUI-K-04** |
| **Nama UI** | Tampilan halaman detail list kelompok |
| **Pengguna** | User yang berperan sebagai Koordinator |
| ***Precondition*** | 1. Pengguna telah memilih peran dalam kuis sebagai koordinator pada halaman pilih peran. 2. Koordinator telah melakukan verifikasi sebagai koordinator dan berhasil. 3. Koordinator telah memanajemen kuis. 4. Koordinator telah memilih kelompok yang ingin dilihat. |
| **Gambar** | |
|  | |
| **Deskripsi** | Tampilan ini adalah halaman detail list kelompok. Aplikasi akan menampilkan data dari database sesuai dengan kelompok yang dipilih oleh koordinator. Berikut komponen yang terdapat pada gambar dijelaskan berdasarkan angka :   1. Merupakan *text view* yang berguna untuk menampilkan nama koordinator dan id koordinator yang dijadikan kode kuis. 2. Merupakan *text view* yang berguna untuk menampilkan jenis klasifikasi kuis. 3. Merupakan *scroll view* yang berguna untuk menampilkan nama-nama peserta yang ada pada kelompok yang dipilih pada halaman mulai kuis koordinator dan list kelompok. 4. Tombol “Kembali” adalah tombol yang berguna untuk berpindah ke halaman sebelumnya yaitu halaman mulai kuis koordinator dan list kelompok. 5. Tombol “*Logout*” adalah tombol yang berguna untuk keluar dari peran koordinator dan berpindah kembali ke halaman pilih peran. |

|  |  |
| --- | --- |
| **No. UI** | **GUI-P-01** |
| **Nama UI** | Tampilan halaman gabung kuis peserta |
| **Pengguna** | User yang berperan sebagai Peserta |
| ***Precondition*** | Pengguna telah memilih peran dalam kuis sebagai peserta pada halaman pilih peran |
| **Gambar** | |
|  | |
| **Deskripsi** | Tampilan ini adalah halaman untuk peserta dalam melakukan verifikasi untuk gabung kuis. Berikut komponen yang terdapat pada gambar dijelaskan berdasarkan angka :   1. Form “Nama Peserta” adalah form yang digunakan untuk *input* nama peserta. 2. Form “Kode Verifikasi” adalaha form yang digunakan untuk *input* kode verifikasi untuk dapat peran sebagai koordinator. 3. Tombol “Masuk” adalah tombol yang berguna untuk melanjutkan proses verifikasi koordinator. 4. Merupakan pesan yang ditampilkan jika terjadi kesalahan pada saat melakukan verifikasi. Terdapat 5 *error,* yaitu : 5. Verifikasi gagal dikarenakan kode verifikasi yang dimasukkan salah, maka aplikasi akan memberikan pesan pemberitahuan yang berisi “Kode verifikasi yang anda masukkan salah”. 6. Verifikasi gagal dikarenakan nama peserta dan kode verifikasi tidak diisi, maka aplikasi akan memberikan pesan pemberitahuan yang berisi “Nama dan kode verifikasi harus diisi”. 7. Verifikasi gagal dikarenakan nama peserta tidak diisi, maka aplikasi akan memberikan pesan pemberitahuan yang berisi “Nama peserta harus diisi”. 8. Verifikasi gagal dikarenakan kode verifikasi tidak diisi, maka aplikasi akan memberikan pesan pemberitahuan yang berisi “Kode verifikasi harus diisi”. 9. Verifikasi gagal dikarenakan kuis sudah dimulai, maka aplikasi akan memberikan pesan pemberitahuan yang berisi “Anda tidak dapat bergabung kuis dikarenakan kuis sudah dimulai”. |

|  |  |
| --- | --- |
| **No. UI** | **GUI-P-02** |
| **Nama UI** | Tampilan halaman informasi kelompok dan menunggu kuis dimulai |
| **Pengguna** | User yang berperan sebagai Peserta |
| ***Precondition*** | 1. Pengguna telah memilih peran dalam kuis sebagai peserta pada halaman pilih peran. 2. Peserta telah melakukan verifikasi untuk gabung kuis dan berhasil. |
| **Gambar** | |
|  | |
| **Deskripsi** | Tampilan ini adalah tampilan untuk peserta melihat kelompok dan menunggu kuis dimulai. Berikut komponen yang terdapat pada gambar dijelaskan berdasarkan angka :   1. Merupakan *text view* yang berguna untuk menampilkan nama peserta dan jenis klasifikasi kuis. 2. Merupakan *text view* yang berguna untuk menampilkan informasi kelompok, jika korrdinator belum melakukan manajemen kuis, maka *text view* tersebut akan berisi “Tunggu koordinator membagikan kelompok”. 3. Merupakan *text view* yang berguna untuk menampilkan *countdown* kuis yang akan segera dimulai. 4. Tombol “*Logout*” adalah tombol yang berguna untuk keluar dari peran peserta dan berpindah kembali ke halaman pilih peran. |

|  |  |
| --- | --- |
| **No. UI** | **GUI-P-03** |
| **Nama UI** | Tampilan halaman petunjuk kuis |
| **Pengguna** | User yang berperan sebagai Peserta |
| ***Precondition*** | 1. Pengguna telah memilih peran dalam kuis sebagai peserta pada halaman pilih peran. 2. Peserta telah melakukan verifikasi untuk gabung kuis dan berhasil. 3. Koordinator telah memulai kuis. |
| **Gambar** | |
|  | |
| **Deskripsi** | Tampilan ini adalah tampilan untuk peserta melihat petunjuk kuis dan alur. Berikut komponen yang terdapat pada gambar dijelaskan berdasarkan angka :   1. Merupakan *text view* yang berguna untuk menampilkan nama peserta dan jenis klasifikasi kuis. 2. Merupakan *text view* yang berguna untuk menampilkan informasi petunjuk kuis. 3. Merupakan *text view* yang berguna untuk menampilkan *countdown* kuis untuk menuju ke pentunjuk ruangan. 4. Tombol “*Logout*” adalah tombol yang berguna untuk keluar dari peran peserta dan berpindah kembali ke halaman pilih peran. |

|  |  |
| --- | --- |
| **No. UI** | **GUI-P-04** |
| **Nama UI** | Tampilan halaman petunjuk ruang |
| **Pengguna** | User yang berperan sebagai Peserta |
| ***Precondition*** | 1. Pengguna telah memilih peran dalam kuis sebagai peserta pada halaman pilih peran. 2. Peserta telah melakukan verifikasi untuk gabung kuis dan berhasil. 3. Koordinator telah memulai kuis. |
| **Gambar** | |
|  | |
| **Deskripsi** | Tampilan ini adalah tampilan untuk peserta dalam mengetahui apakah sudah berada di lokasi yang sesuai dengan yang sudah ditentukan atau belum. Jika belum, aplikasi akan memberikan informasi berupa tulisan “Anda belum berada di lokasi yang sesuai” dan button “Cek ulang lokasi”, jika sudah berada di lokasi, aplikasi akan berpindah ke halaman informasi koleksi. Berikut komponen yang terdapat pada gambar dijelaskan berdasarkan angka :   1. Merupakan *text view* yang berguna untuk menampilkan nama peserta dan jenis klasifikasi kuis. 2. Merupakan *text view* yang berguna untuk menampilkan nama ruang untuk dikunjungi. 3. Merupakan gambar petunjuk lokasi ruang yang harus dikunjungi untuk mendapatkan informasi koleksi dan soal kuis. 4. Merupakan *text view* yang berguna untuk menampilkan informasi petunjuk ruang berupa *text.* 5. Merupakan *text view* yang berguna untuk memberikan informasi bahwa peserta belum berada dilokasi yang sesua dengan petunjuk. 6. Tombol “Cek Ulang Lokasi” merupakan tombol untuk melakukan proses pengecekan ulang lokasi. 7. Tombol “*Logout*” adalah tombol yang berguna untuk keluar dari peran peserta dan berpindah kembali ke halaman pilih peran. |

|  |  |
| --- | --- |
| **No. UI** | GUI-P-05 |
| **Nama UI** | Tampilan halaman informasi koleksi |
| **Pengguna** | User yang berperan sebagai Peserta |
| **Precondition** | 1. Pengguna telah memilih peran dalam kuis sebagai peserta pada halaman pilih peran. 2. Peserta telah melakukan verifikasi untuk gabung kuis dan berhasil. 3. Koordinator telah memulai kuis. |
| **Gambar** | |
|  | |
| **Deskripsi** | Tampilan ini adalah menampilkan informasi koleksi. Infromasi yang akan ditampilkan adalah informasi koleksi-koleksi dari klasifikasi koleksi yang paling terdekat dengan android. Berikut komponen yang terdapat pada gambar dijelaskan berdasarkan angka :   1. Merupakan *text view* yang berguna untuk menampilkan nama peserta dan jenis klasifikasi kuis. 2. Merupakan *scroll view* yang berguna untuk menampilkan informasi koleksi-koleksi yang ada pada suatu klasifikasi. 3. Tombol “Cari Klasifikasi Terdekat” adalah tombol yang berguna untuk melakukan proses pencarian klasifikasi koleksi terdekat. 4. Tombol “*Logout*” adalah tombol yang berguna untuk keluar dari peran peserta dan berpindah kembali ke halaman pilih peran. |

|  |  |
| --- | --- |
| **No. UI** | **GUI-P-06** |
| **Nama UI** | Tampilan halaman kuis |
| **Pengguna** | User yang berperan sebagai Peserta |
| ***Precondition*** | 1. Pengguna telah memilih peran dalam kuis sebagai peserta pada halaman pilih peran. 2. Peserta telah melakukan verifikasi untuk gabung kuis dan berhasil. 3. Koordinator telah memulai kuis. 4. Peserta telah berada dilokasi yang sesuai dengan petunjuk ruangan. |
| **Gambar** | |
|  | |
| **Deskripsi** | Tampilan ini adalah tampilan untuk peserta dalam menjawab soal kuis. Berikut komponen yang terdapat pada gambar dijelaskan berdasarkan angka :   1. Merupakan *text view* yang berguna untuk menampilkan nama peserta dan jenis klasifikasi kuis. 2. Merupakan *text view* dan gambar untuk menampilkan soal kuis. 3. Merupakan *card view* untuk tombol pilihan a, b, c, dan d yang berguna untuk melanjutkan proses pengecekan jawaban dan perhitungan poin. 4. Merupakan *text view* yang berguna untuk menampilkan *countdown* sisa waktu untuk menjawab soal kuis. 5. Tombol “*Logout*” adalah tombol yang berguna untuk keluar dari peran peserta dan berpindah kembali ke halaman pilih peran. |

|  |  |
| --- | --- |
| **No. UI** | **GUI-P-07** |
| **Nama UI** | Tampilan halaman informasi benar atau salahnya jawaban |
| **Pengguna** | User yang berperan sebagai Peserta |
| ***Precondition*** | 1. Pengguna telah memilih peran dalam kuis sebagai peserta pada halaman pilih peran. 2. Peserta telah melakukan verifikasi untuk gabung kuis dan berhasil. 3. Koordinator telah memulai kuis. 4. Peserta telah berada dilokasi yang sesuai dengan petunjuk ruangan. |
| **Gambar** | |
|  | |
| **Deskripsi** | Tampilan ini adalah tampilan untuk informasi benar atau salahnya jawaban. Berikut komponen yang terdapat pada gambar dijelaskan berdasarkan angka :   1. Merupakan *text view* yang berguna untuk menampilkan nama peserta dan jenis klasifikasi kuis. 2. Merupakan *text view* dan gambar untuk menampilkan informasi benar atau salahnya jawaban dan menampilkan poin yang didapatkan setelah menjawab soal kuis. 3. Merupakan *text view* yang berguna untuk menampilkan *countdown* sisa waktu untuk melihat informasi benar atau salahnya jawaban setelah itu akan berpindah ke halaman informasi koleksi. 4. Tombol “*Logout*” adalah tombol yang berguna untuk keluar dari peran peserta dan berpindah kembali ke halaman pilih peran. |

|  |  |
| --- | --- |
| **No. UI** | **GUI-PK-01** |
| **Nama UI** | Tampilan halaman untuk menampilkan peringkat |
| **Pengguna** | Pemandu dan rombongan pengunjung Museum Geologi |
| ***Precondition*** | 1. Pengguna telah masuk kedalam aplikasi 2. Pengguna telah memilih lihat peringkat pada tampilan halaman awal aplikasi atau pada tampilan halaman kuis selesai |
| **Gambar** | |
|  | |
| **Deskripsi** | Tampilan ini adalah tampilan untuk pengguna dalam melihat list peringkat. Berikut komponen yang terdapat pada gambar dijelaskan berdasarkan angka :   1. Merupakan *text view* untuk menampilkan informasi peringkat peserta yang telah menyelesaikan kuis, *text view* ini ada jika pengguna telah bermain sebagai peserta dan menyelesaikan kuis. 2. Merupakan form untuk melakukan filterisasi untu peringkat yang ditampilkan berdasarkan id koordinator atau kelompok. 3. Tombol “Filter” adalah tombol yang berguna untuk melakukan proses filterisasi. 4. Merupakan *list view* untuk menampilkan list peringkat kuis. 5. Tombol “Kembali” adalah tombol yang berguna untuk berpindah ke tampilan halaman awal aplikasi. |