**BAB IV**

**IMPLEMENTASI DAN PEMBAHASAN**

1. **Implementasi**

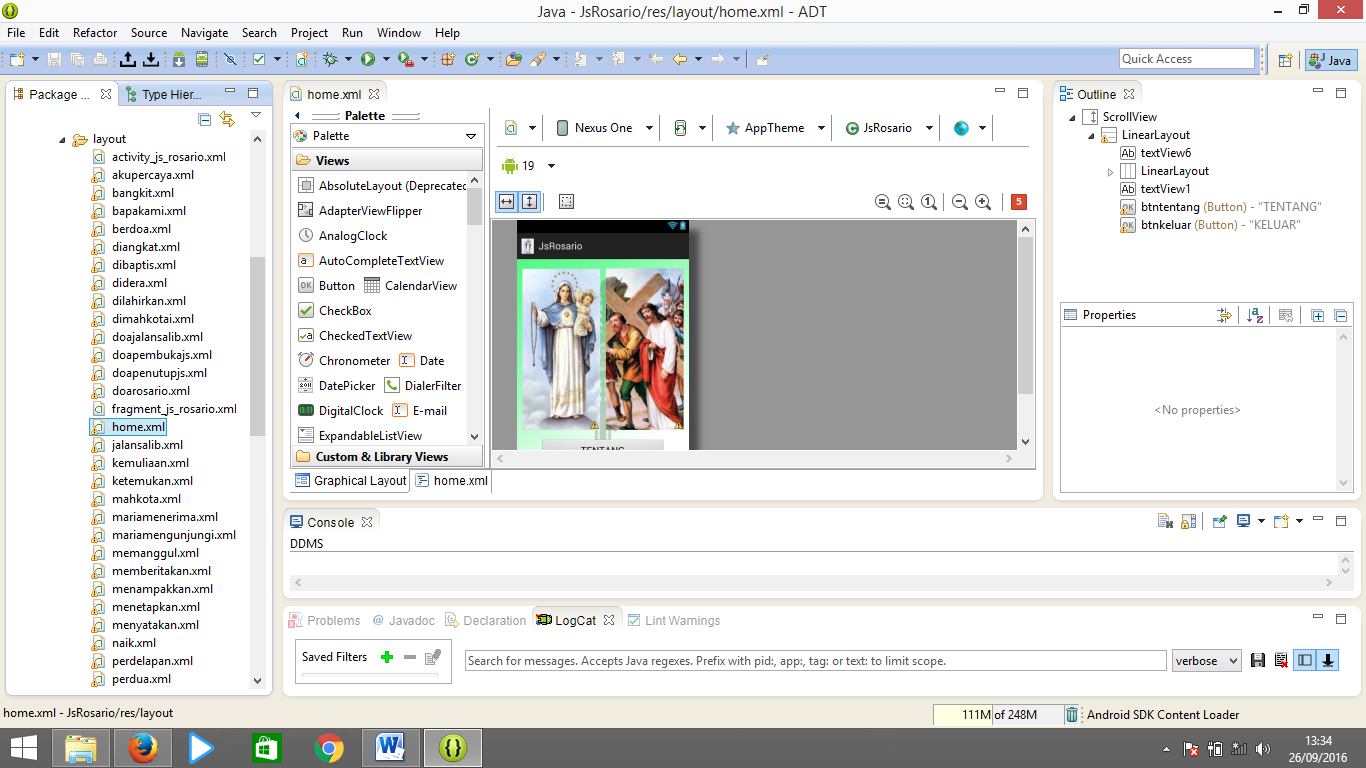
Dalam rencana penerapan sebuah sistem yang akan dibangun, agar sistem siap dioperasikan maka perlu diadakan kegiatan-kegiatan dari penerapannya. Adapun langkah-langkah yang akan dilakukan untuk penerapan sistem tersebut adalah pembuatan program, *testing* / uji coba sistem, dan sampai pada tahap pemeliharaan sistem. Berikut adalah tahap yang dilakukan penulis dalam penerapan sistem yang dibuat:

1. Proses pembuatan program

Setelah menentukan desain *interface* maka penulis mulai melakukan pengumpulan data sebagai bahan untuk pembuatan program. Penulis membuat program menggunakan bahasa pemrograman *java* dengan perangkat lunak IDE *Eclipse* versi 4.3*.* Berikut adalah tampilan *user interface* dari perangkat lunak yang digunakan:

1. *User Interface IDE Eclipse*

Gambar 30 merupakan tampilan antarmuka dari perangkat lunak *IDE Eclipse* yang digunakan untuk membuat Aplikasi Doa Rosario dan Jalan Salib.



Gambar 30. Tampilan Antarmuka Eclipse

1. Uji Coba Sistem dan Program

Proses ujicoba sistem dan program merupakan hal yang sangat penting dilakukan mengingat sangat perlunya dilakukan pengujian untuk melihat kelayakan dan kemampuan maupun stabilitas program ketika dioperasikan serta mengukur kebenaran materi yang diujikan dalam aplikasi Doa Rosario dan Jalan Salib.

Pengujian dilakukan dengan menggunakan metode *black box testing.* Pada tahap pengujian dengan metode *black box* dilakukan testing terhadap semua fungsi yang terdapat pada aplikasi Doa Rosario dan Jalan Salib. Dengan metode *black box* *testing* ini maka dapat diketahui kesalahan yang terjadi pada *Aplikasi Doa Rosario dan Jalan Salib berbasis Android* seperti ketidaksesuaian *interface*, kesalahan kinerja, dan kesalahan fungsi baik dari fungsi aplikasi maupun fungsi lainnya. Dengan memasukkan semua kemungkinan input dan menjalankan semua fitur yang tersedia pada aplikasi kemudian diperiksa apakah fungsi yang diujikan tersebut menghasilkan output sesuai dengan yang diharapkan sehingga tidak ada ditemukan kesalahan *interface*, kesalahan kinerja dan fungsi dari *aplikasi Doa Rosario dan Jalan Salib* tersebut.

Adapun *black box testing* yang dilakukan meliputi beberapa pengujian yang dapat dikelompokkan seperti pada Tabel 6.

Tabel 6. Pengujian Sistem Metode *Black Box*

|  |  |  |
| --- | --- | --- |
| Nama Pengujian | Kriteria Evaluasi Hasil | Jenis Pengujian |
| Pengujian *splash screen* | Melihat proses tampilnya *splash screen* | *Black Box Testing* |
| Pengujian terhadap menu utama | Melihat proses tampilnya menu utama | *Black Box Testing* |
| Pengujian terhadap menu Doa Rosario | Melihat proses tampilnya menu Doa Rosario | *Black Box Testing* |
| Pengujian terhadap *video* tata cara doa Rosario | Melihat proses tampilnya *video* tata cara Doa Rosario | *Black Box Testing* |
| Pengujian terhadap menu mulai Doa Rosario | Melihat proses tampilnya menu doa Rosario | *Black Box Testing* |
| Pengujian terhadap sejarah Rosario | Melihat proses tampilnya sejarah Rosario | *Black Box Testing* |
| Pengujian terhadap menu Jalan Salib | Melihat proses tampilnya menu Jalan Salib | *Black Box Testing* |
| Pengujian terhadap *video* tata cara dalam Jalan Salib | Melihat proses tampilnya *video* urutan doa dalam Jalan Salib | *Black Box Testing* |
| Pengujian terhadap menu mulai Doa Jalan Salib | Melihat proses tampilnya menu Doa Jalan Salib | *Black Box Testing* |
| Pengujian terhadap sejarah Jalan Salib | Melihat proses tampilnya sejarah Jalan Salib | *Black Box Testing* |
| Pengujian terhadap menu tentang | Melihat proses menu tentang | *Black Box Testing* |
| Pengujian terhadap menu keluar | Melihat proses menu keluar | *Black Box Testing* |

Pengujian pada sistem ini merupakan tahap-tahap apakah program yang diuji sesuai dengan hasil yang diharapkan, adapun hasil pengujian pada aplikasi ini dapat dilihat dengan beberapa pengujian.

1. Pengujian *splash screen*

Hasil pengujian ini dilakukan untuk mengetahui apakah *splash screen* dapat tampil atau tidak. Berdasarkan hasil pengujian yang dilakukan, *splash screen* dapat tampil dengan baik pada pada *handphone* *android*. Adapun hasil dari pengujian tersebut dapat dilihat pada Tabel 7.

Tabel 7. Hasil Pengujian *Splash Screen*

|  |  |  |  |
| --- | --- | --- | --- |
| **Hasil Pengujian** | | | |
| **Data Masukan** | **Harapan** | **Pengamatan** | **Kesimpulan** |
| Pengguna memilih icon Doa Rosario dan Jalan Salib di *handphone* | *splash screen* muncul dengan baik setelah icon Doa Rosario dan Jalan Salib di pilih | *splash screen* tampil setelah icon Doa Rosario dan Jalan Salib dipilih | Sesuai |

1. Pengujian menu utama

Hasil pengujian ini dilakukan untuk mengetahui apakah menu utama dapat tampil dengan baik atau tidak. Berdasarkan hasil pengujian yang dilakukan, menu utama bisa tampil dengan baik pada pada *smartphone* *android*. Adapun hasil dari pengujian tersebut dapat dilihat pada Tabel 8.

Tabel 8. Hasil Pengujian Menu Utama

|  |  |  |  |
| --- | --- | --- | --- |
| **Hasil Pengujian** | | | |
| **Data Masukan** | **Harapan** | **Pengamatan** | **Kesimpulan** |
| Pengguna memilih icon Doa Rosario dan Jalan Salib | Menu utama akan muncul dengan baik sesudah *splash screen.* | Menu utama tampil sesudah  *splash screen.* | Sesuai |

1. Pengujian menu Doa Rosario

Hasil pengujian ini dilakukan untuk mengetahui apakah menu Doa Rosario dapat tampil dengan baik atau tidak setelah menu utama. Berdasarkan hasil pengujian yang dilakukan, menu Doa Rosario bisa tampil dengan baik setelah menekan gambar bunda Maria pada menu utama. Adapun hasil dari pengujian tersebut dapat dilihat pada Tabel 9.

Tabel 9. Hasil Pengujian Menu Doa Rosario

|  |  |  |  |
| --- | --- | --- | --- |
| **Hasil Pengujian** | | | |
| **Data Masukan** | **Harapan** | **Pengamatan** | **Kesimpulan** |
| Pengguna memilih menu doa Rosario pada menu utama. | Menu doa Rosario dapat muncul dengan baik setelah menekan gambar Bunda Maria pada menu utama. | Menu doa Rosario tampil setelah menekan gambar Bunda Maria pada menu utama. | Sesuai |

1. Pengujian *video* tata cara Doa Rosario

Hasil pengujian ini dilakukan untuk mengetahui apakah *video* tata cara Doa Rosario dapat tampil dengan baik atau tidak setelah menu doa Rosario. Berdasarkan hasil pengujian yang dilakukan, *video* tata cara Doa Rosario dapat tampil dengan baik setelah menekan *video* dalam menu Doa Rosario. Adapun hasil dari pengujian tersebut dapat dilihat pada Tabel 10.

Tabel 10. Hasil Pengujian *video* tata cara Doa Rosario

|  |  |  |  |
| --- | --- | --- | --- |
| **Hasil Pengujian** | | | |
| **Data Masukan** | **Harapan** | **Pengamatan** | **Kesimpulan** |
| Pengguna memilih *video* tata cara Doa Rosario pada menu Doa Rosario. | *Video* tata cara doa Rosario dapat muncul dengan baik setelah menekan *video* tata cara Doa Rosario pada menu Doa Rosario. | *Video* tata cara doa Rosario tampil setelah menekan *video* tata cara Rosario pada menu Doa Rosario. | Sesuai |

1. Pengujian menu mulai Doa Rosario

Hasil pengujian ini dilakukan untuk mengetahui apakah menu mulai Doa Rosario dapat tampil dengan baik atau tidak setelah menu Doa Rosario. Berdasarkan hasil pengujian yang dilakukan, menu mulai Doa Rosario bisa tampil dengan baik setelah menekan *button* mulai doa dalam menu Doa Rosario. Adapun hasil dari pengujian tersebut dapat dilihat pada Tabel 11.

Tabel 11. Hasil Pengujian Menu Mulai Doa Rosario

|  |  |  |  |
| --- | --- | --- | --- |
| **Hasil Pengujian** | | | |
| **Data Masukan** | **Harapan** | **Pengamatan** | **Kesimpulan** |
| Pengguna memilih menu mulai doa pada menu Doa Rosario. | Menu mulai Doa Rosario dapat muncul dengan baik setelah menekan *button* mulai doa pada menu Doa Rosario. | Menu mulai doa tampil setelah menekan *button* mulai doa pada menu Doa Rosario. | Sesuai |

1. Pengujian sejarah Rosario

Hasil pengujian ini dilakukan untuk mengetahui apakah menu sejarah Rosario dapat tampil dengan baik atau tidak setelah menu Doa Rosario. Berdasarkan hasil pengujian yang dilakukan, menu sejarah Rosario bisa tampil dengan baik setelah menekan *button* sejarah Rosario dalam menu Doa Rosario. Adapun hasil dari pengujian tersebut dapat dilihat pada Tabel 12.

Tabel 12. Hasil Pengujian Sejarah Rosario

|  |  |  |  |
| --- | --- | --- | --- |
| **Hasil Pengujian** | | | |
| **Data Masukan** | **Harapan** | **Pengamatan** | **Kesimpulan** |
| Pengguna memilih menu sejarah Rosario pada menu Doa Rosario. | Menu sejarah Rosario dapat muncul dengan baik setelah menekan *button* sejarah Rosario pada menu Doa Rosario. | Menu sejarah Rosario tampil setelah menekan *button* sejarah Rosario pada menu Doa Rosario. | Sesuai |

1. Pengujian menu Jalan Salib

Hasil pengujian ini dilakukan untuk mengetahui apakah menu Jalan Salib dapat tampil dengan baik atau tidak setelah menu utama. Berdasarkan hasil pengujian yang dilakukan, menu Doa Rosario bisa tampil dengan baik setelah menekan gambar Yesus memanggul salib pada menu utama. Adapun hasil dari pengujian tersebut dapat dilihat pada Tabel 13.

Tabel 13. Hasil Pengujian Menu Jalan Salib

|  |  |  |  |
| --- | --- | --- | --- |
| **Hasil Pengujian** | | | |
| **Data Masukan** | **Harapan** | **Pengamatan** | **Kesimpulan** |
| Pengguna memilih menu Jalan Salib pada menu utama. | Menu Jalan Salib dapat muncul dengan baik setelah menekan gambar Yesus memanggul salib pada menu utama. | Menu Jalan Salib tampil setelah menekan gambar Yesus memanggul salib pada menu utama. | Sesuai |

1. Pengujian *video* tata cara Jalan Salib

Hasil pengujian ini dilakukan untuk mengetahui apakah *video* tata cara Jalan Salib dapat tampil dengan baik atau tidak setelah menu Jalan Salib. Berdasarkan hasil pengujian yang dilakukan, *video* tata cara Jalan Salib dapat tampil dengan baik setelah menekan *video* dalam menu Jalan Salib. Adapun hasil dari pengujian tersebut dapat dilihat pada Tabel 14.

Tabel 14. Hasil Pengujian *Video* Tata cara Jalan Salib

|  |  |  |  |
| --- | --- | --- | --- |
| **Hasil Pengujian** | | | |
| **Data Masukan** | **Harapan** | **Pengamatan** | **Kesimpulan** |
| Pengguna memilih *video* tata cara Jalan Salib pada menu Jalan Salib. | *Video* tata cara Jalan Salib dapat muncul dengan baik setelah menekan *video* tata cara Jalan salib pada menu Jalan Salib. | *Video* tata cara Jalan Salib tampil setelah menekan *video* tata cara Jalan Salib pada menu Jalan Salib. | Sesuai |

1. Pengujian menu mulai doa Jalan Salib

Hasil pengujian ini dilakukan untuk mengetahui apakah menu mulai doa Jalan Salib dapat tampil dengan baik atau tidak setelah menu doa Jalan Salib. Berdasarkan hasil pengujian yang dilakukan, menu mulai doa Jalan Salib dapat tampil dengan baik setelah menekan *button* mulai doa dalam menu Jalan Salib. Adapun hasil dari pengujian tersebut dapat dilihat pada Tabel 15.

Tabel 15. Hasil Pengujian Menu Mulai Doa Jalan Salib

|  |  |  |  |
| --- | --- | --- | --- |
| **Hasil Pengujian** | | | |
| **Data Masukan** | **Harapan** | **Pengamatan** | **Kesimpulan** |
| Pengguna memilih menu mulai doa pada menu Jalan Salib. | Menu mulai doa Jalan Salib dapat muncul dengan baik setelah menekan *button* mulai doa pada menu Jalan Salib. | Menu mulai doa tampil setelah menekan *button* mulai doa pada menu Jalan Salib. | Sesuai |

1. Pengujian sejarah jalan salib

Hasil pengujian ini dilakukan untuk mengetahui apakah menu sejarah Jalan Salib dapat tampil dengan baik atau tidak setelah menu Doa Jalan Salib. Berdasarkan hasil pengujian yang dilakukan, menu sejarah Jalan Salib dapat tampil dengan baik setelah menekan *button* sejarah Jalan Salib dalam menu doa Jalan Salib. Adapun hasil dari pengujian tersebut dapat dilihat pada Tabel 16.

Tabel 16. Hasil Pengujian sejarah jalan salib

|  |  |  |  |
| --- | --- | --- | --- |
| **Hasil Pengujian** | | | |
| **Data Masukan** | **Harapan** | **Pengamatan** | **Kesimpulan** |
| Pengguna memilih menu sejarah Jalan Salib pada menu Doa Jalan Salib. | Menu sejarah Jalan Salib dapat muncul dengan baik setelah menekan *button* sejarah Jalan Salib pada menu jalan salib. | Menu sejarah Jalan Salib tampil setelah menekan *button* sejarah Jalan Salib. | Sesuai |

1. Pengujian menu tentang

Hasil pengujian ini dilakukan untuk mengetahui apakah menu tentang dapat tampil dengan baik atau tidak. Berdasarkan hasil pengujian yang dilakukan, menu tentang dapat tampil dengan baik setelah menekan menu tentang pada menu utama. Adapun hasil dari pengujian tersebut dapat dilihat pada Tabel 17.

Tabel 17. Hasil Pengujian Menu Tentang

|  |  |  |  |
| --- | --- | --- | --- |
| **Hasil Pengujian** | | | |
| **Data Masukan** | **Harapan** | **Pengamatan** | **Kesimpulan** |
| Pengguna memilih menu tentang pada menu utama. | Menu tentang dapat muncul dengan baik setelah menekan *button* tentang pada menu utama. | Menu tentang tampil setelah menekan *button* tentang pada menu utama. | Sesuai |

1. Pengujian menu keluar

Hasil pengujian ini dilakukan untuk mengetahui apakah menu keluar dapat berfungsi dengan baik atau tidak. Berdasarkan hasil pengujian yang dilakukan, menu keluar dapat berfungsi dengan baik dengan menutup aplikasi Doa Rosario dan Jalan Salib yang berjalan. Ada pun hasil dari pengujian tersebut dapat dilihat pada Tabel 18.

Tabel 18. Hasil Pengujian Menu Keluar

|  |  |  |  |
| --- | --- | --- | --- |
| **Hasil Pengujian** | | | |
| **Data Masukan** | **Harapan** | **Pengamatan** | **Kesimpulan** |
| Pengguna memilih menu keluar. | Menu keluar menutup aplikasi Doa Rosario dan Jalan Salib dengan memberikan peringatan untuk menutup aplikasi. | Menu keluar menutup aplikasi Doa Rosario dan Jalan Salib dengan peringatan terlebih dahulu | Sesuai |

1. Manual program

Manual program merupakan konsep untuk menguraikan penggunaan program aplikasi Doa rosario dan Jalan Salib secara keseluruhan, mulai dari cara menjalankan program untuk pertama kali, mengoperasikan tombol-tombol dan proses lainnya di dalam program. Sehingga dapat memudahkan pengguna untuk menggunakan aplikasi, berikut uraian penggunaan program aplikasi Doa Rosario dan Jalan salib.

Aplikasi Doa Rosario dan Jalan Salib ini terdiri *file* *setup/packge* berektensi *.apk* berupa file master yang dapat diinstal disetiap *handphone* berbasis *android*. Untuk dapat menggunakan program aplikasi *Doa Rosario dan Jalan Salib* ini maka perlu dilakukan instalasi program. Setelah terinstal aplikasi *Doa Rosario dan Jalan Salib* pada *handphone* yang digunakan, maka akan terdapat ikon Doa Rosario dan Jalan Salib sebagai jalan pintas yang dapat digunakan untuk menjalankan aplikasi tersebut. Berikut langkah-langkah dalam menggunakan aplikasi *Doa Rosario dan Jalan Salib* berbasis *android*:

1. Tap / sentuh *icon* *aplikasi Doa Rosario dan Jalan Salib* yang ada pada *smartphone android* pengguna.
2. Layar *splash screen* akan muncul dengan gambar dan teks dengan khas Doa Rosario dan Jalan Salib.
3. Kemudian akan muncul tampilan menu utama. Tampilan menu utama memuat *image button* Doa Rosario dan Jalan Salib, *button* menu tentang dan menu keluar.
4. Jika memilih image button Doa Rosario maka pengguna akan mendapat tiga pilihan menu yaitu *video* tentang tata cara Doa Rosario, *button* mulai doa dan sejarah Rosario. Jika memilih *video* maka akan memutar *video* tentang tata cara Doa Rosario. Jika *button* mulai doa dipilih maka akan menampilkan *button* lagi yang di dalamnya merupakan doa-doa yang ada dalam Doa Rosario. Jika memilih button sejarah Rosario maka akan menampilkan sejarah singkat Doa Rosario.
5. Jika memilih *image button* Jalan Salib maka pengguna akan mendapat tiga pilihan menu yaitu *video* tentang tata cara Jalan Salib, *button* mulai doa dan sejarah Jalan Salib. Jika memilih *video* maka akan memutar *video* tentang tata cara Jalan Salib. Jika *button* mulai doa dipilih maka akan menampilkan *button* lagi yang di dalamnya merupakan doa-doa dan perhentian yang ada dalam Jalan Salib. Jika memilih button sejarah Jalan Salib maka akan menampilkan sejarah singkat Jalan Salib.
6. Apabila pengguna memilih *button* menu tentang maka akan menampilkan foto dan biodata biodata tentang pembuat aplikasi dan sumber doa yang digunakan dalam aplikasi.
7. Apabila pengguna memilih *button* menu keluar maka aplikasi doa rosario dan Jalan Salib akan ditutup dengan peringatan berupa pertanyaan untuk meyakinkan pengguna menutup aplikasi.
8. Manual instalasi

Pada manual instalasi ini akan dijelaskan tentang bagaimana cara menjalankan *aplikasi Doa Rosario dan Jalan Salib Berbasis Android* pada sistem operasi *android*:

1. *Copy file* “*JsRosario.apk*” ke *SD card* atau *internal smartphone android* melalui kabel data, *bluetooth* atau melalui media lainnya.
2. Tap *file* “*JsRosario.apk*” yang sudah dipindahkan, maka instalasi aplikasi akan berjalan. Tunggu hingga proses instalasi selesai.
3. Lihat apakah *icon* doa Rosario dan Jalan Salib sudah muncul pada layar *smartphone android* pengguna. Apabila sudah ada, *aplikasi Doa Rosario dan Jalan Salib berbasis android* siap dijalankan.
4. Pemeliharaan sistem

Pemeliharaan aplikasi perlu dilakukan agar sistem dapat berjalan lebih baik dan bertahan lebih lama, dilakukan penjagaan dari *asset* aplikasi itu sendiri maupun komponennya. Berikut kiat dan pemeliharaan aplikasi:

1. Nama *file* aplikasi asli sebaiknya tidak diganti karena akan mengalami masalah dalam kompresi *file* aslinya.
2. Penyimpanan *asset* aplikasi sangat penting, karena akan memudahkan *programmer* dalam menyusun kembali aplikasi apabila terjadi perubahan, *error,* maupun pengembangan ke tahap selanjutnya.
3. Melakukan *backup* untuk menghindari kehilangan aplikasi.
4. **Pembahasan**
5. Pembahasan *interface* / antarmuka program
6. Halaman *Splash Screen*

Halaman *Splash screen* merupakanproses yang menampilkan gambar ciri khas *aplikasi Doa Rosario dan Jalan Salib* yang akan tampil sebelum menu utama. Adapun implementasi dari halaman splash screen dapat dilihat pada Gambar 31.



Gambar 31. *Splash screen*

Kode program untuk menampilkan *splash screen*:

|  |
| --- |
| **protected** **void** onCreate(Bundle savedInstanceState) {  **super**.onCreate(savedInstanceState);  setContentView(R.layout.*activity\_js\_rosario*);  Thread background = **new** Thread() {  **public** **void** run() {  **try** {  *sleep*(3\*1000);  Intent i=**new** Intent(getBaseContext(),home.**class**);  startActivity(i);    finish();  } **catch** (Exception e) {}  }  };    background.start();  }  **protected** **void** onDestroy() {  **super**.onDestroy();  }  @Override  **public** **boolean** onCreateOptionsMenu(Menu menu) {  // Inflate the menu; this adds items to the action bar if it is present.  getMenuInflater().inflate(R.menu.*js\_rosario*, menu);  **return** **true**;  }  } |

1. Halaman menu utama

Halaman menu utama merupakan halaman yang menampilkan pilihan-pilihan menu bagi pengguna untuk memasuki sub-sub menu yang akan digunakan lebih lanjut. Halaman menu utama dapat dilihat pada Gambar 32.

  
Gambar 32. Menu utama

Kode program untuk menampilkan menu utama:

|  |
| --- |
| **public** **class** home **extends** Activity {  **final** Context context = **this**;  **private** Button button;  @Override  **protected** **void** onCreate(Bundle savedInstanceState) {  **super**.onCreate(savedInstanceState);  setContentView(R.layout.*home*);    //fungsi untuk intent button rosario  **final** ImageButton btnrosario = (ImageButton)findViewById(R.id.*btnrosario*);  btnrosario.setOnClickListener(**new** OnClickListener() {    @Override  **public** **void** onClick(View arg0) {  // **TODO** Auto-generated method stub  Intent rosario = **new** Intent(arg0.getContext(),rosario.**class**);  startActivityForResult(rosario, 0);  }  });  //fungsi untuk intent button jalan salib  **final** ImageButton btnjalansalib = (ImageButton)findViewById(R.id.*btnjalansalib*);  btnjalansalib.setOnClickListener(**new** OnClickListener() {    @Override  **public** **void** onClick(View arg0) {  // **TODO** Auto-generated method stub  Intent jalansalib = **new** Intent(arg0.getContext(),jalansalib.**class**);  startActivityForResult(jalansalib, 0);  }  });    //fungsi untuk intent button tentang  **final** Button btntentang = (Button)findViewById(R.id.*btntentang*);  btntentang.setOnClickListener(**new** OnClickListener() {    @Override  **public** **void** onClick(View arg0) {  // **TODO** Auto-generated method stub  Intent tentang = **new** Intent(arg0.getContext(),tentang.**class**);  startActivityForResult(tentang, 0);  }  });    //fungsi untuk intent button exit  **final** Button btnkeluar = (Button)findViewById(R.id.*btnkeluar*);  btnkeluar.setOnClickListener(**new** OnClickListener() {    @Override  **public** **void** onClick(View arg0) {  AlertDialog.Builder alertDialogBuilder = **new** AlertDialog.Builder(context);  // set title  alertDialogBuilder.setTitle("Rosario dan Jalan Salib");    // set dialog message  alertDialogBuilder  .setMessage("Anda Yakin Ingin Keluar !!!")  .setCancelable(**false**)  .setPositiveButton("Ya",**new** DialogInterface.OnClickListener() {  **public** **void** onClick(DialogInterface dialog,**int** id) {    // if this button is clicked, close  // current activity  //Home.this.finish();  //finish();  Intent intent = **new** Intent(Intent.*ACTION\_MAIN*);  intent.addCategory(Intent.*CATEGORY\_HOME*);  intent.setFlags(Intent.*FLAG\_ACTIVITY\_NEW\_TASK*);  startActivity(intent);  finishAffinity();  System.*exit*(1);    }  })  .setNegativeButton("Tidak",**new** DialogInterface.OnClickListener() {  **public** **void** onClick(DialogInterface dialog,**int** id) {  // if this button is clicked, just close  // the dialog box and do nothing  dialog.cancel();  }  });    // create alert dialog  AlertDialog alertDialog = alertDialogBuilder.create();    // show it  alertDialog.show();  }  });  }    @Override  **public** **void** onBackPressed() {  **new** AlertDialog.Builder(**this**)  .setMessage("Anda ingin keluar ?")  .setCancelable(**false**)  .setPositiveButton("Ya", **new** DialogInterface.OnClickListener() {    @Override  **public** **void** onClick(DialogInterface arg0, **int** arg1) {  // **TODO** Auto-generated method stub  home.**this**.finish();  System.*exit*(0);  }  })  .setNegativeButton("Tidak", **null**)  .show();  }  } |

1. Halaman menu Doa Rosario

Halaman menu Doa Rosario merupakan halaman yang menampilkan *video* tata cara Doa Rosario, mulai doa, dan sejarah Rosario. Jika pada *video* di tap / tekan maka akan memetutar *video*, tap / tekan sekali lagi untuk berhenti melihat *video*. Halaman menu Doa Rosario dapat dilihat pada Gambar 33.



Gambar 33. Menu Doa Rosario

Kode program untuk menampilkan menu Doa rosario:

|  |
| --- |
| **public** **class** rosario **extends** Activity{  **private** VideoView myVideo;  **private** **boolean** bVideo = **false**;  **private** Handler mHandler = **new** Handler();    @Override  **protected** **void** onCreate(Bundle savedInstanceState) {  **super**.onCreate(savedInstanceState);  getWindow().setFormat(PixelFormat.*TRANSLUCENT*);  setContentView(R.layout.*rosario*);    Uri clip = Uri.*parse*("android.resource://" + getPackageName() + "/" + R.raw.*rosario*);    myVideo = (VideoView)findViewById(R.id.*rosario*);  myVideo.setVideoURI(clip);  myVideo.setOnTouchListener(**new** View.OnTouchListener() {  @Override  **public** **boolean** onTouch(View v, MotionEvent event) {  // **TODO** Auto-generated method stub  **if**(!bVideo) {  bVideo = **true**;  **if**(myVideo.isPlaying()) {  myVideo.pause();  } **else** {  myVideo.start();  }  mHandler.postDelayed(**new** Runnable() {  **public** **void** run() {  bVideo = **false**;  }  }, 100);  }  **return** **false**;  }  }  );    //fungsi untuk intent button sejarah  **final** Button btnsejarah = (Button)findViewById(R.id.*btnsejarah*);  btnsejarah.setOnClickListener(**new** OnClickListener() {    @Override  **public** **void** onClick(View arg0) {  // **TODO** Auto-generated method stub  Intent sejrosario = **new** Intent(arg0.getContext(),sejrosario.**class**);  startActivityForResult(sejrosario, 0);  }  });    //fungsi untuk intent button mulai doa  **final** Button btndoa = (Button)findViewById(R.id.*btndoa*);  btndoa.setOnClickListener(**new** OnClickListener() {    @Override  **public** **void** onClick(View arg0) {  // **TODO** Auto-generated method stub  Intent doarosario = **new** Intent(arg0.getContext(),doarosario.**class**);  startActivityForResult(doarosario, 0);  }  });    }  } |

1. Halaman menu mulai doa

Halaman menu mulai doa adalah kumpulan doa-doa yang ada dalam Doa Rosario dan terdapat peristiwa yang digunakan pada hari tertentu pada saat Doa Rosario diadakan. Halaman menu mulai doa dapat dilihat pada Gambar 34.



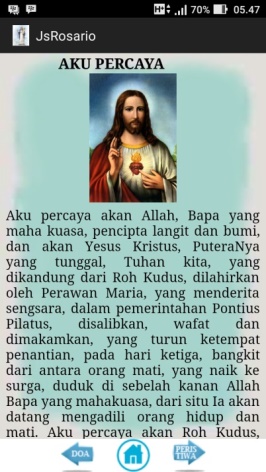
Gambar 34. Menu mulai Doa Rosario

Kode program untuk menampilkan mulai Doa Rosario:

|  |
| --- |
| **public** **class** doarosario **extends** Activity {  @Override  **protected** **void** onCreate(Bundle savedInstanceState) {  **super**.onCreate(savedInstanceState);  setContentView(R.layout.*doarosario*);    //fungsi untuk intent button Aku percaya  **final** Button btnpercaya = (Button)findViewById(R.id.*btnpercaya*);  btnpercaya.setOnClickListener(**new** OnClickListener() {    @Override  **public** **void** onClick(View arg0) {  // **TODO** Auto-generated method stub  Intent akupercaya = **new** Intent(arg0.getContext(),akupercaya.**class**);  startActivityForResult(akupercaya, 0);  }  });    //fungsi untuk intent button kemuliaan  **final** Button btnkemuliaan = (Button)findViewById(R.id.*btnkemuliaan*);  btnkemuliaan.setOnClickListener(**new** OnClickListener() {    @Override  **public** **void** onClick(View arg0) {  // **TODO** Auto-generated method stub  Intent kemuliaan = **new** Intent(arg0.getContext(),kemuliaan.**class**);  startActivityForResult(kemuliaan, 0);  }  });    //fungsi untuk intent button terpujilah  **final** Button btnterpujilah = (Button)findViewById(R.id.*btnterpujilah*);  btnterpujilah.setOnClickListener(**new** OnClickListener() {    @Override  **public** **void** onClick(View arg0) {  // **TODO** Auto-generated method stub  Intent terpujilah = **new** Intent(arg0.getContext(),terpujilah.**class**);  startActivityForResult(terpujilah, 0);  }  });    //fungsi untuk intent button Bapa kami  **final** Button btnbapa = (Button)findViewById(R.id.*btnbapa*);  btnbapa.setOnClickListener(**new** OnClickListener() {    @Override  **public** **void** onClick(View arg0) {  // **TODO** Auto-generated method stub  Intent bapakami = **new** Intent(arg0.getContext(),bapakami.**class**);  startActivityForResult(bapakami, 0);  }  });    //fungsi untuk intent button Salam Maria  **final** Button btnsalam = (Button)findViewById(R.id.*btnsalam*);  btnsalam.setOnClickListener(**new** OnClickListener() {    @Override  **public** **void** onClick(View arg0) {  // **TODO** Auto-generated method stub  Intent salammaria = **new** Intent(arg0.getContext(),salammaria.**class**);  startActivityForResult(salammaria, 0);  }  });    //fungsi untuk intent button Ya Yesus  **final** Button btnyesus = (Button)findViewById(R.id.*btnyesus*);  btnyesus.setOnClickListener(**new** OnClickListener() {    @Override  **public** **void** onClick(View arg0) {  // **TODO** Auto-generated method stub  Intent yayesus = **new** Intent(arg0.getContext(),yayesus.**class**);  startActivityForResult(yayesus, 0);  }  });    //fungsi untuk intent button Salam Ya Ratu  **final** Button btnratu = (Button)findViewById(R.id.*btnratu*);  btnratu.setOnClickListener(**new** OnClickListener() {    @Override  **public** **void** onClick(View arg0) {  // **TODO** Auto-generated method stub  Intent ratu = **new** Intent(arg0.getContext(),ratu.**class**);  startActivityForResult(ratu, 0);  }  });  }  } |

1. Halaman doa

Halaman doa merupakan doa yang ada dalam menu mulai doa, sebagai contoh Doa Aku Percaya. *Button* bacakan digunakan untuk membaca teks doa, *button* doa digunakan untuk membuka halaman doa, *button home* digunakan untuk kembali kemenu utama dan button peristiwa digunakan untuk memilih peristiwa yang digunakan saat akan berdoa Rosario. Halaman Doa Aku percaya dapat dilihat pada Gambar 35.



Gambar 35. Doa Aku Percaya

Kode program untuk menampilkan Doa Aku Percaya:

|  |
| --- |
| **protected** **void** onCreate(Bundle savedInstanceState) {  **super**.onCreate(savedInstanceState);  setContentView(R.layout.*akupercaya*);    //Memanggil file my\_sound pada folder raw  audioBackground = MediaPlayer.*create*(**this**, R.raw.*maria*);  //Set looping ke true untuk mengulang audio jika telah selesai  audioBackground.setLooping(**true**);  //Set volume audio agar berbunyi  audioBackground.setVolume(1,1);  //Memulai audio  audioBackground.start();    textjalan = (TextView) findViewById(R.id.*textView1*);  textjalan.setSelected(**true**);    String htmlText = "<p style=\"text-align: justify; font: 18px times new roman;\"> %s </p>";  String myData = "";  text = myData.replaceAll("<(.\*?)\\>"," ");    WebView webView = (WebView) findViewById(R.id.*webView1*);  webView.loadData(String.*format*(htmlText, myData), "text/html", "utf-8");  webView.setBackgroundColor(0x00000000);    tts = **new** TextToSpeech(**this**, **this**);    btnSpeak = (Button) findViewById(R.id.*btnSpeak*);    // button on click event  btnSpeak.setOnClickListener(**new** View.OnClickListener() {    **public** **void** onClick(View arg0) {  speakOut();  }    });    //fungsi untuk intent button doa  **final** ImageButton btndoa = (ImageButton)findViewById(R.id.*btndoa*);  btndoa.setOnClickListener(**new** OnClickListener() {    @Override  **public** **void** onClick(View arg0) {  // **TODO** Auto-generated method stub  tts.stop();  tts.shutdown();  audioBackground.stop();  Intent doarosario = **new** Intent(arg0.getContext(),doarosario.**class**);  startActivityForResult(doarosario, 0);  }  });    //fungsi untuk intent button home  **final** ImageButton btnhome = (ImageButton)findViewById(R.id.*btnhome*);  btnhome.setOnClickListener(**new** OnClickListener() {    @Override  **public** **void** onClick(View arg0) {  // **TODO** Auto-generated method stub  tts.stop();  tts.shutdown();  audioBackground.stop();  Intent home = **new** Intent(arg0.getContext(),home.**class**);  startActivityForResult(home, 0);  }  });    //fungsi untuk intent button pilih peristiwa  **final** ImageButton btnperistiwa = (ImageButton)findViewById(R.id.*btnperistiwa*);  btnperistiwa.setOnClickListener(**new** OnClickListener() {    @Override  **public** **void** onClick(View arg0) {  // **TODO** Auto-generated method stub  tts.stop();  tts.shutdown();  audioBackground.stop();  Intent peristiwa = **new** Intent(arg0.getContext(),peristiwa.**class**);  startActivityForResult(peristiwa, 0);  }  });  }  @Override  **public** **void** onDestroy() {  // Don't forget to shutdown!  **if** (tts != **null**) {  tts.stop();  tts.shutdown();  audioBackground.stop();  }  **super**.onDestroy();  }    **public** **void** onInit(**int** status) {  // **TODO** Auto-generated method stub    **if** (status == TextToSpeech.*SUCCESS*) {    **int** result = tts.isLanguageAvailable(**new** Locale("INDONESIA"));    //tts.setPitch(0.5f); // set pitch level    tts.setSpeechRate(0.85f); // set speech speed rate    **if** (result == TextToSpeech.*LANG\_MISSING\_DATA*  || result == TextToSpeech.*LANG\_NOT\_SUPPORTED*) {  Log.*e*("TTS", "Language is not supported");  } **else** {  btnSpeak.setEnabled(**true**);  speakOut();  }    } **else** {  Log.*e*("TTS", "Initilization Failed");  }    }    **private** **void** speakOut() {    //String text = txtText.getText().toString();    tts.speak(text, TextToSpeech.*QUEUE\_FLUSH*, **null**);  }  } |

1. Halaman pilih peristiwa

Halaman pilih peristiwa merupakan pilihan peristiwa yang dipilih sesuai dengan hari pada saat akan dilakukannya Doa Rosario. Terdapat empat peristiwa yang ada dalam Doa Rosario yaitu peristiwa gembira, sedih, mulia, dan terang. Halaman pilih peristiwa dapat dilihat pada Gambar 36.



Gambar 36. Pilih peristiwa

Kode program untuk menampilkan pilih peristiwa:

|  |
| --- |
| **protected** **void** onCreate(Bundle savedInstanceState) {  **super**.onCreate(savedInstanceState);  setContentView(R.layout.*peristiwa*);    //fungsi untuk intent button Peristiwa Gembira  **final** Button btngembira = (Button)findViewById(R.id.*btngembira*);  btngembira.setOnClickListener(**new** OnClickListener() {    @Override  **public** **void** onClick(View arg0) {  // **TODO** Auto-generated method stub  Intent pergembira = **new** Intent(arg0.getContext(),pergembira.**class**);  startActivityForResult(pergembira, 0);  }  });    //fungsi untuk intent button Peristiwa Sedih  **final** Button btnsedih = (Button)findViewById(R.id.*btnsedih*);  btnsedih.setOnClickListener(**new** OnClickListener() {    @Override  **public** **void** onClick(View arg0) {  // **TODO** Auto-generated method stub  Intent persedih = **new** Intent(arg0.getContext(),persedih.**class**);  startActivityForResult(persedih, 0);  }  });    //fungsi untuk intent button Peristiwa Sedih  **final** Button btnmulia = (Button)findViewById(R.id.*btnmulia*);  btnmulia.setOnClickListener(**new** OnClickListener() {    @Override  **public** **void** onClick(View arg0) {  // **TODO** Auto-generated method stub  Intent permulia = **new** Intent(arg0.getContext(),permulia.**class**);  startActivityForResult(permulia, 0);  }  });    //fungsi untuk intent button Peristiwa Terang  **final** Button btnterang = (Button)findViewById(R.id.*btnterang*);  btnterang.setOnClickListener(**new** OnClickListener() {    @Override  **public** **void** onClick(View arg0) {  // **TODO** Auto-generated method stub  Intent perterang = **new** Intent(arg0.getContext(),perterang.**class**);  startActivityForResult(perterang, 0);  }  });  }  } |

1. Halaman sejarah Rosario

Halaman sejarah Rosario merupakan sejarah singkat tentang Doa Rosario. Halaman sejarah Rosario dapat dilihat pada Gambar 37.



Gambar 37. Sejarah Doa Rosaio

Kode program untuk menampilkan sejarah Doa Rosario:

|  |
| --- |
| **protected** **void** onCreate(Bundle savedInstanceState) {  **super**.onCreate(savedInstanceState);  setContentView(R.layout.*sejrosario*);    //Memanggil file my\_sound pada folder raw  audioBackground = MediaPlayer.*create*(**this**, R.raw.*maria*);  //Set looping ke true untuk mengulang audio jika telah selesai  audioBackground.setLooping(**true**);  //Set volume audio agar berbunyi  audioBackground.setVolume(1,1);  //Memulai audio  audioBackground.start();    textjalan = (TextView) findViewById(R.id.*textView1*);  textjalan.setSelected(**true**);    String htmlText = "<p style=\"text-align: justify; font: 18px times new roman;\"> %s </p>";  String myData = "";  text = myData.replaceAll("<(.\*?)\\>"," ");    WebView webView = (WebView) findViewById(R.id.*webView1*);  webView.loadData(String.*format*(htmlText, myData), "text/html", "utf-8");  webView.setBackgroundColor(0x00000000);    tts = **new** TextToSpeech(**this**, **this**);    btnSpeak = (Button) findViewById(R.id.*btnSpeak*);    //txtText = (TextView) findViewById(R.id.textView2);    // button on click event  btnSpeak.setOnClickListener(**new** View.OnClickListener() {    **public** **void** onClick(View arg0) {  speakOut();  }    });  }  @Override  **public** **void** onDestroy() {  // Don't forget to shutdown!  **if** (tts != **null**) {  tts.stop();  tts.shutdown();  audioBackground.stop();  }  **super**.onDestroy();  }    **public** **void** onInit(**int** status) {  // **TODO** Auto-generated method stub    **if** (status == TextToSpeech.*SUCCESS*) {    **int** result = tts.isLanguageAvailable(**new** Locale("INDONESIA"));    //tts.setPitch(0.5f); // set pitch level    tts.setSpeechRate(0.85f); // set speech speed rate    **if** (result == TextToSpeech.*LANG\_MISSING\_DATA*  || result == TextToSpeech.*LANG\_NOT\_SUPPORTED*) {  Log.*e*("TTS", "Language is not supported");  } **else** {  btnSpeak.setEnabled(**true**);  speakOut();  }    } **else** {  Log.*e*("TTS", "Initilization Failed");  }    }    **private** **void** speakOut() {    //String text = txtText.getText().toString();    tts.speak(text, TextToSpeech.*QUEUE\_FLUSH*, **null**);  }  } |

1. Halaman menu Jalan Salib

Halaman menu Jalan Salib merupakan halaman yang menampilkan *video* tata cara Jalan Salib, mulai doa, dan sejarah Jalan Salib. Jika pada *video* di tap / tekan maka akan memetutar *video*, tap / tekan sekali lagi untuk berhenti melihat *video*. Halaman menu jalan salib dapat dilihat pada Gambar 38.



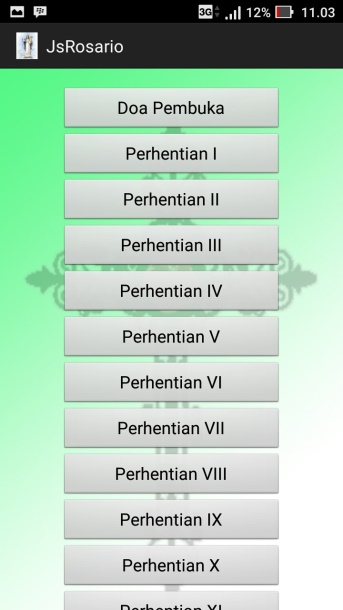
Gambar 38. Menu Jalan Salib

Kode program untuk menampilkan menu Jalan Salib:

|  |
| --- |
| **protected** **void** onCreate(Bundle savedInstanceState) {  **super**.onCreate(savedInstanceState);  getWindow().setFormat(PixelFormat.*TRANSLUCENT*);  setContentView(R.layout.*jalansalib*);    Uri clip = Uri.*parse*("android.resource://" + getPackageName() + "/" + R.raw.*jalansalibjadi*);    myVideo = (VideoView)findViewById(R.id.*jalansalibjadi*);  myVideo.setVideoURI(clip);  myVideo.setOnTouchListener(**new** View.OnTouchListener() {  @Override  **public** **boolean** onTouch(View v, MotionEvent event) {  // **TODO** Auto-generated method stub  **if**(!bVideo) {  bVideo = **true**;  **if**(myVideo.isPlaying()) {  myVideo.pause();  } **else** {  myVideo.start();  }  mHandler.postDelayed(**new** Runnable() {  **public** **void** run() {  bVideo = **false**;  }  }, 100);  }  **return** **false**;  }  }  );    //fungsi untuk intent button sejarah  **final** Button btnsejarah = (Button)findViewById(R.id.*btnsejarah*);  btnsejarah.setOnClickListener(**new** OnClickListener() {    @Override  **public** **void** onClick(View arg0) {  // **TODO** Auto-generated method stub  Intent sejjalansalib = **new** Intent(arg0.getContext(),sejjalansalib.**class**);  startActivityForResult(sejjalansalib, 0);  }  });    //fungsi untuk intent button mulai doa  **final** Button btndoa = (Button)findViewById(R.id.*btndoa*);  btndoa.setOnClickListener(**new** OnClickListener() {    @Override  **public** **void** onClick(View arg0) {  // **TODO** Auto-generated method stub  Intent doajalansalib = **new** Intent(arg0.getContext(),doajalansalib.**class**);  startActivityForResult(doajalansalib, 0);  }  });  }  } |

1. Halaman menu mulai doa

Halaman menu mulai doa adalah kumpulan doa-doa yang ada dalam Jalan Salib. Adapaun implementasi dari tampilan halaman menu mulai Doa Jalan Salib dapat dilihat pada Gambar 39.



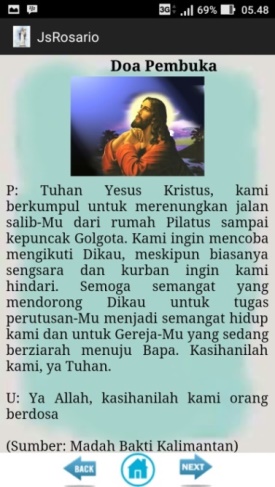
Gambar 39. Menu mulai Doa Jalan Salib

Kode program untuk menampilkan Doa Jalan Salib:

|  |
| --- |
| **protected** **void** onCreate(Bundle savedInstanceState) {  **super**.onCreate(savedInstanceState);  setContentView(R.layout.*doajalansalib*);    //fungsi untuk intent button doa pembuka  **final** Button btnpembuka = (Button)findViewById(R.id.*btnpembuka*);  btnpembuka.setOnClickListener(**new** OnClickListener() {    @Override  **public** **void** onClick(View arg0) {  // **TODO** Auto-generated method stub  Intent doapembukajs = **new** Intent(arg0.getContext(),doapembukajs.**class**);  startActivityForResult(doapembukajs, 0);  }  });  //fungsi untuk intent button perhentian I  **final** Button btn1 = (Button)findViewById(R.id.*btn1*);  btn1.setOnClickListener(**new** OnClickListener() {    @Override  **public** **void** onClick(View arg0) {  // **TODO** Auto-generated method stub  Intent perhentiansatu = **new** Intent(arg0.getContext(),perhentiansatu.**class**);  startActivityForResult(perhentiansatu, 0);  }  });    //fungsi untuk intent button perhentian II  **final** Button btn2 = (Button)findViewById(R.id.*btn2*);  btn2.setOnClickListener(**new** OnClickListener() {    @Override  **public** **void** onClick(View arg0) {  // **TODO** Auto-generated method stub  Intent perdua = **new** Intent(arg0.getContext(),perdua.**class**);  startActivityForResult(perdua, 0);  }  });    //fungsi untuk intent button perhentian XIII  **final** Button btn13 = (Button)findViewById(R.id.*btn13*);  btn13.setOnClickListener(**new** OnClickListener() {    @Override  **public** **void** onClick(View arg0) {  // **TODO** Auto-generated method stub  Intent pertigabelas = **new** Intent(arg0.getContext(),pertigabelas.**class**);  startActivityForResult(pertigabelas, 0);  }  });    //fungsi untuk intent button perhentian XIV  **final** Button btn14 = (Button)findViewById(R.id.*btn14*);  btn14.setOnClickListener(**new** OnClickListener() {    @Override  **public** **void** onClick(View arg0) {  // **TODO** Auto-generated method stub  Intent perempatbelas = **new** Intent(arg0.getContext(),perempatbelas.**class**);  startActivityForResult(perempatbelas, 0);  }  });    //fungsi untuk intent button perhentian XIV  **final** Button btnpenutup = (Button)findViewById(R.id.*btnpenutup*);  btnpenutup.setOnClickListener(**new** OnClickListener() {    @Override  **public** **void** onClick(View arg0) {  // **TODO** Auto-generated method stub  Intent doapenutupjs = **new** Intent(arg0.getContext(),doapenutupjs.**class**);  startActivityForResult(doapenutupjs, 0);  }  });  }  } |

1. Halaman doa

Halaman doa merupakan doa yang ada dalam menu mulai doa. Jalan Salib di awali dengan doa pembuka. *Button* bacakan digunakan untuk membaca teks doa, *button* *back* digunakan untuk membuka halaman sebelumnya, *button next* digunakan untuk membuka halaman selanjutnya dan *button home* digunakan untuk kembali ke menu utama. Halaman doa pembuka dapat dilihat pada Gambar 40.



Gambar 40. Doa pembuka

Kode program untuk menampilkan Doa Pembuka:

|  |
| --- |
| **protected** **void** onCreate(Bundle savedInstanceState) {  **super**.onCreate(savedInstanceState);  setContentView(R.layout.*doapembukajs*);    //Memanggil file my\_sound pada folder raw  audioBackground = MediaPlayer.*create*(**this**, R.raw.*jalansalib*);  //Set looping ke true untuk mengulang audio jika telah selesai  audioBackground.setLooping(**true**);  //Set volume audio agar berbunyi  audioBackground.setVolume(1,1);  //Memulai audio  audioBackground.start();    textjalan = (TextView) findViewById(R.id.*textView1*);  textjalan.setSelected(**true**);    String htmlText = "<p style=\"text-align: justify; font: 18px times new roman;\"> %s </p>";  String myData = "";  text = myData.replaceAll("<(.\*?)\\>"," ");    WebView webView = (WebView) findViewById(R.id.*webView1*);  webView.loadData(String.*format*(htmlText, myData), "text/html", "utf-8");  webView.setBackgroundColor(0x00000000);    tts = **new** TextToSpeech(**this**, **this**);    btnSpeak = (Button) findViewById(R.id.*btnSpeak*);    //txtText = (TextView) findViewById(R.id.textView2);    // button on click event  btnSpeak.setOnClickListener(**new** View.OnClickListener() {    **public** **void** onClick(View arg0) {  speakOut();  }  });    //fungsi untuk intent button back  **final** ImageButton btnback = (ImageButton)findViewById(R.id.*btnback*);  btnback.setOnClickListener(**new** OnClickListener() {    @Override  **public** **void** onClick(View arg0) {  // **TODO** Auto-generated method stub  tts.stop();  tts.shutdown();  audioBackground.stop();  Intent jalansalib = **new** Intent(arg0.getContext(),jalansalib.**class**);  startActivityForResult(jalansalib, 0);  }  });    //fungsi untuk intent button home  **final** ImageButton btnhome = (ImageButton)findViewById(R.id.*btnhome*);  btnhome.setOnClickListener(**new** OnClickListener() {    @Override  **public** **void** onClick(View arg0) {  // **TODO** Auto-generated method stub  tts.stop();  tts.shutdown();  audioBackground.stop();  Intent home = **new** Intent(arg0.getContext(),home.**class**);  startActivityForResult(home, 0);  }  });    //fungsi untuk intent button next  **final** ImageButton btnnext = (ImageButton)findViewById(R.id.*btnnext*);  btnnext.setOnClickListener(**new** OnClickListener() {    @Override  **public** **void** onClick(View arg0) {  // **TODO** Auto-generated method stub  tts.stop();  tts.shutdown();  audioBackground.stop();  Intent perhentiansatu = **new** Intent(arg0.getContext(),perhentiansatu.**class**);  startActivityForResult(perhentiansatu, 0);  }  });  }  @Override  **public** **void** onDestroy() {  // Don't forget to shutdown!  **if** (tts != **null**) {  tts.stop();  tts.shutdown();  audioBackground.stop();  }  **super**.onDestroy();  }    **public** **void** onInit(**int** status) {  // **TODO** Auto-generated method stub    **if** (status == TextToSpeech.*SUCCESS*) {    **int** result = tts.isLanguageAvailable(**new** Locale("INDONESIA"));    //tts.setPitch(0.5f); // set pitch level    tts.setSpeechRate(0.85f); // set speech speed rate    **if** (result == TextToSpeech.*LANG\_MISSING\_DATA*  || result == TextToSpeech.*LANG\_NOT\_SUPPORTED*) {  Log.*e*("TTS", "Language is not supported");  } **else** {  btnSpeak.setEnabled(**true**);  speakOut();  }    } **else** {  Log.*e*("TTS", "Initilization Failed");  }    }    **private** **void** speakOut() {    //String text = txtText.getText().toString();    tts.speak(text, TextToSpeech.*QUEUE\_FLUSH*, **null**);  }  } |

1. Halaman sejarah Jalan Salib

Halaman sejarah Jalan Salib merupakan halaman yang akan menampilkan sejarah singkat tentang Jalan Salib. Halaman sejarah Jalan Salib dapat dilihat pada Gambar 41.



Gambar 41. Sejarah Jalan Salib

Kode program untuk menampilkan Sejarah Jalan Salib:

|  |
| --- |
| **protected** **void** onCreate(Bundle savedInstanceState) {  **super**.onCreate(savedInstanceState);  setContentView(R.layout.*sejjalansalib*);    //Memanggil file my\_sound pada folder raw  audioBackground = MediaPlayer.*create*(**this**, R.raw.*jalansalib*);  //Set looping ke true untuk mengulang audio jika telah selesai  audioBackground.setLooping(**true**);  //Set volume audio agar berbunyi  audioBackground.setVolume(1,1);  //Memulai audio  audioBackground.start();    textjalan = (TextView) findViewById(R.id.*textView1*);  textjalan.setSelected(**true**);    String htmlText = "<p style=\"text-align: justify; font: 18px times new roman;\"> %s </p>";  String myData = "";  text = myData.replaceAll("<(.\*?)\\>"," ");    WebView webView = (WebView) findViewById(R.id.*webView1*);  webView.loadData(String.*format*(htmlText, myData), "text/html", "utf-8");  webView.setBackgroundColor(0x00000000);    tts = **new** TextToSpeech(**this**, **this**);    btnSpeak = (Button) findViewById(R.id.*btnSpeak*);    //txtText = (TextView) findViewById(R.id.textView3);    // button on click event  btnSpeak.setOnClickListener(**new** View.OnClickListener() {    **public** **void** onClick(View arg0) {  speakOut();  audioBackground.stop();  }    });  }  @Override  **public** **void** onDestroy() {  // Don't forget to shutdown!  **if** (tts != **null**) {  tts.stop();  tts.shutdown();  audioBackground.stop();  }  **super**.onDestroy();  }    **public** **void** onInit(**int** status) {  // **TODO** Auto-generated method stub    **if** (status == TextToSpeech.*SUCCESS*) {    **int** result = tts.isLanguageAvailable(**new** Locale("INDONESIA"));    //tts.setPitch(0.5f); // set pitch level    tts.setSpeechRate(0.85f); // set speech speed rate    **if** (result == TextToSpeech.*LANG\_MISSING\_DATA*  || result == TextToSpeech.*LANG\_NOT\_SUPPORTED*) {  Log.*e*("TTS", "Language is not supported");  } **else** {  btnSpeak.setEnabled(**true**);  speakOut();  }    } **else** {  Log.*e*("TTS", "Initilization Failed");  }    }    **private** **void** speakOut() {    //String text = txtText.getText().toString();    tts.speak(text, TextToSpeech.*QUEUE\_FLUSH*, **null**);  }  } |

1. Halaman menu tentang

Halaman menu tentang merupakan halaman yang menampilkan isi tentang biodata pembuat aplikasi Doa Rosario dan Jalan Salib. Halaman menu tentang dapat dilihat pada Gambar 42.



Gambar 42. Menu tentang

Kode program untuk menampilkan menu tentang:

|  |
| --- |
| **protected** **void** onCreate(Bundle savedInstanceState) {  **super**.onCreate(savedInstanceState);  setContentView(R.layout.*tentang*);  }  } |

1. Halaman menu keluar

Halaman menu keluar merupakan halaman yang berfungsi menutup aplikasi Doa Rosario dan Jalan Salib yang digunakan pengguna dengan memberi peringatan untuk memastikan bahwa pengguna yakin ingin keluar dari aplikasi. Halaman menu keluar dapat dilihat pada Gambar 43.



Gambar 43. Menu keluar

1. Pembahasan Hasil Responden Pengguna

Penilain terhadap Aplikasi Doa Rosario dan Jalan Salib Berbasis Android, penulis melakukan penyebaran kuisioner yang diberikan kepada 10 orang. Selanjutnya mereka menggunakan aplikasi Doa Rosario dan Jalan Salib kemudian dilakukan pengumpulan data dengan mengisi kuesioner yang telah dibagikan sebelumnya oleh penulis.

1. Pernyataan kuesioner

Pernyataan kuesioner merupakan tahap untuk mengetahui pernyataan apa saja yang diajukan kepada responden terhadap aplikasi Doa Rosario dan Jalan Salib ini. Berikut pernyataan kuesioner seperti pada Tabel 19.

Tabel 19. Pernyataan Kuesioner

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Pernyataan** | **Alternatif Jawaban** | | | | |
| **5** | **4** | **3** | **2** | **1** |
| **SB** | **B** | **CB** | **KB** | **TB** |
| 1. | Tampilan aplikasi sudah bersifat baik / *user friendly* |  |  |  |  |  |
| 2. | Fungsional aplikasi sudah berjalan sesuai dengan fungsinya |  |  |  |  |  |
| 3. | Aplikasi mampu menampilkan *video* |  |  |  |  |  |
| 4. | *Backsound* yang ada dalam aplikasi sesuai dengan Doa Rosario dan Jalan Salib |  |  |  |  |  |
| 5. | Aplikasi terlihat menarik, baik dari segi kecerahan warna, tampilan dan teks |  |  |  |  |  |
| 6. | Aplikasi ini dapat membantu ketika melakukan Doa Rosario dan Jalan Salib |  |  |  |  |  |
| 7. | Tentang keseluruhan aplikasi. |  |  |  |  |  |

Keterangan bobot yang digunakan dalam penilaian alternatif jawaban dapat dilihat pada Tabel 20.

Tabel 20. Keterangan Bobot

|  |  |
| --- | --- |
| **Keterangan** | **Nilai** |
| Sangat Baik | 5 |
| Baik | 4 |
| Cukup Baik | 3 |
| Kurang Baik | 2 |
| Tidak Baik | 1 |

1. Hasil Kuesioner

Hasil kuesioner merupakan hasil perhitungan dari penilaian responden terhadap Aplikasi Doa Rosario dan Jalan Salib Berbasis Android. Hasil kuesioner dapat dilihat pada Tabel 21.

Tabel 21. Hasil kuesioner

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Responden** | **Jawaban Pertanyaan** | | | | | | | **Jumlah** |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** |
| 1 | 4 | 5 | 3 | 3 | 5 | 4 | 4 | 28 |
| 2 | 5 | 5 | 4 | 4 | 5 | 5 | 4 | 32 |
| 3 | 4 | 4 | 4 | 5 | 5 | 5 | 4 | 31 |
| 4 | 4 | 5 | 5 | 4 | 5 | 4 | 4 | 31 |
| 5 | 5 | 5 | 5 | 5 | 4 | 4 | 4 | 32 |
| 6 | 5 | 5 | 5 | 5 | 5 | 4 | 4 | 33 |
| 7 | 5 | 4 | 4 | 4 | 4 | 5 | 4 | 30 |
| 8 | 5 | 4 | 4 | 4 | 5 | 5 | 5 | 32 |
| 9 | 5 | 4 | 4 | 5 | 5 | 5 | 5 | 33 |
| 10 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 35 |
| **Jumlah Skor Hasil Pengumpulan Data** | | | | | | | | **317** |

Sedangkan untuk bobot daftar pertanyaan dapat dilihat pada Tabel 22.

Tabel 22. Bobot daftar pertanyaan

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Keterangan** | **Daftar Pertanyaan** | | | | | | |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** |
| Sangat Baik | 7 | 6 | 4 | 5 | 8 | 6 | 7 |
| Baik | 3 | 4 | 5 | 4 | 2 | 4 | 3 |
| Cukup Baik | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| Kurang Baik | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Tidak Baik | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Berdasarkan hasil kuesioner pada tabel dapat dilihat menggunakan skala pengukuran, disini penulis menggunakan likert sebagai skala pengukuran yaitu sebagai berikut:

Jumlah skor kriteria = (skor tertinggi tiap item) x (jumlah pertanyaan) x (jumlah responden)

Jumlah skor kriteria = 5 x 7 x 10

= 350

Berdasarkan tabel untuk total jawaban responden adalah 317 dan jumlah skor kriterium adalah 350. Dengan demikian Aplikasi Doa Rosario dan Jalan Salib berbasis android menurut pendapat 10 orang responden yaitu:

Total hasil responden = 317 / 350 x 100

= 0,91 x 100

= 91 %

Jadi, sikap responden terhadap aplikasi yang dibangun sebesar 91 % dan termasuk dalam interval sangat baik.

0 20% 40% 60% 80%  **91%** 100%

TB KB CB B SB

Gambar 44. Persentase *range* kelompok responden

Keterangan :

TB = Tidak Baik

KB = Kurang Baik

CB = Cukup Baik

B = Baik

SB = Sangat Baik

Berdasarkan hasil perhitungan kuesioner dengan skala *likert* yang didapat dari 10 respon dan 7 pertanyaan maka didapat hasil interpretasi sebesar 91% atau dengan nilai 317 dari nilai maksimal 350, maka dapat disimpulkan bahwa aplikasi ini sangat baik dan layak untuk digunakan.