**PENGUJIAN TEST PLAN PADA**

**APLIKASI JAGO SHOLAT**

****

DISUSUN OLEH :

**KELOMPOK DUGLEGIR**

**MUHAMMAD FAISAL AMIR 6706160014**

**MUHAMAD IKHSAN RAMADHAN 6706162062**

**BRYAN RAFSANZANI 6706160065**

**D3IF-40-02**

1. **Test Plan Identifier**

Testing akan dilakukan pada aplikasi Mortech Mechanica untuk penguji penggunaan semua fitur oleh user.

|  |  |
| --- | --- |
| Subjek testing | User & admin |
| Penanggung Jawab | Muhammad Faisal Amir |
| Penyusun test plan | 1. Muhamad Ikhsan Ramadhan 2. Bryan Rafsanzani |
| Tanggal test plan dibuat | 30 Oktober 2018 |
| Tanggal revisi |  |

1. **Introduction**

Test plan adalah dasar yang digunakan untuk pengujian apakah hasil rancang bangun dari suatu proyek sudah sesuai dengan dengan tujuan-tujuan dari pembuatan aplikasi atau belum. Pada test plan ini kami akan melakukan pengujian apakah hasil rancang bangun aplikasi Jago Solat android sudah sesuai dengan tujuan-tujuan awal dari pembuatan aplikasi atau belum.

1. Purpose

Dokumen Test Plan ini dibuat untuk mendukung pengembangan aplikasi Jago Sholat diantaranya adalah :

* Mengidentifikasi masalah pada aplikasi.
* Memastikan aplikasi berjalan dengan semestinya.

1. Background

Sholat merupakan kewajiban bagi umat islam. Menurut survei Lembaga Survei Indonesia (LSI) mengungkapkan bahwa hanya sedikit dari kaum muda Islam Indonesia yang melaksanakan salat lima waktu..

LSI bekerjasama Goethe Institute Friedrich Naumann Stiftung dan Fur Die Freiheit, menyebutkan hasil surveinya, bahwa kaum muda Islam Indonesia yang selalu menunaikan salat 5 waktu (28,7 persen), yang sering salat 5 waktu (30,2 persen), yang kadang-kadang salat 5 waktu (39,7 persen), yang tidak pernah salat 5 waktu (1,2 persen). Polling ini melibatkan lebih dari 1.056 orang tua yang memiliki anak berusia di bawah 18 tahun. Sebanyak 59 persen dari para orang tua itu juga mengaku menggunakan perangkat sebagai bentuk 'hukuman atau penghargaan'. (Arrahmah.com)

Menurut riset Roy Morgan, kepemilikan smartphone di Indonesia dua kali lipat antara tahun 2012 dan 2013 menjadi 24%. Banyak anak-anak menggunakan telepon seluler dan smartphone sebanyak (52%), dengan anak-anak dan remaja menjadi pendorong utama pergeseran ini. Hampir sembilan dari sepuluh anak-anak (89 %) berkomunikasi secara online dengan teman-teman sementara kelompok-kelompok yang lebih kecil juga berinteraksi dengan keluarga mereka (56 %) atau guru mereka (35%) melalui internet, dan topik yang paling dibicarakan harus dilakukan dengan kegiatan sekolah. (UNICEF.org)

Oleh karena itu, tujuan dibuatnya aplikasi jago sholat ialah untuk mencatat ibadah dan melihat grafik peningkatan ibadah umat muslim agar umat muslim tau dari hasil pencatatan ibadahnya apakah sholat nya semakin mendekati waktu awal sholat atau semakin di akhirkan sholatnya.

1. **Test Items**

* 1a. Sanity check
* 1b. Core apps quality
* 1c. Tablet apps quality
* 1d. Automated testing
* 1e. Unconditional testing
* 1f. Code quality testing
* 2a. Unit test
* 2b. Data diven unit test
* 2c. Performance
* 2d. Accesbility
* 2e. Functionality
* 2f. Security
* 2g. Usability
* 2h. Automated ui test
* 2i. Espreso
* 2j. Apk analize
* 2k. Google play

1. **Features to be Tested**

|  |  |
| --- | --- |
| **Fitur** | **Deskripsi** |
| Tombol catat ibadah | Untuk memasukan data sholat kedalam database |
| Validasi waktu sholat | Tampil waktu sholat sesuai dengan sholat yang seharusnya ada pada jam tersebut |
| Validasi tanggal | Menampilkan tanggal sesuai tanggal pada hari tersebut |
| Validasi jadwal sholat | Tampil jadwal sholat sesuai dengan waktu lokasi sholat |
| Validasi waktu sholat yang akan datang | Muncul waktu sholat yang akan datang, misal setelah sholat dzuhur berarti sholat ashar dan yang akan tampil adalah sholat ashar. |
| Hitung mundur waktu sholat selanjutnya | Menampilkan waktu hitung mundur sampai sholat selanjutnya |
| Statistik ibadah | Tampil grafik peningkatan sholat |
| Data ibadah | Hasil dari pencatatan ibadah dan tampil di menu statistik |
| Update data ibadah | Data yang masuk kedalam statistik dapat di update |
| Lokasi kita berada | Tampil pada menu kompas lokasi tempat kita berada |
| Arah kompas | Arah kompas yang dituju oleh gambar sesuai dengan arah kiblat |
| Menu tata cara wudhu | Menampilkan gambar dari mulai niat wudhu hingga niat selesai wudhu |
| Geser gambar | Gambar dapat di geser ke kiri dan ke kanan |
| Menu niat sholat | Menampilkan niat berupa teks dari niat sholat shubuh hingga sholat isya |
| Menu tata cara sholat | Menampilkan gambar dari mulai niat sholat hingga selesai salam. |
| Tampil do’a setelah sholat | Menampilkan beberapa doa – doa setelah selesai sholat |

1. **Features Not to be Tested**

* **Tentang kami**

1. **Test Strategy**

**1a. Sanity Check**

Sanity Testing yaitu Jenis software testing yang dilakukan setelah software sudah hamper jadi dengan fungsi-fungsi lengkap yang sudah jadi, dan bug-bug yang ditemukan pada saat smoke testing sudah berhasil diperbaiki. Tujuan sanity testing untuk memastikan bahwa bug telah diperbaiki dan tidak ada masalah lebih lanjut serta untuk menentukan bahwa fungsi yang diusulkan bekerja seperti yang diharapkan. Jika sanity test gagal maka aplikasi akan direject untuk menghemat waktu dan biaya yang terlibat dalam pengujian lebih lanjut.

Dalam testing sanity check yang harus dilakukan yaitu:

1. Mendeskripsikan Aplikasi Jago Sholat
2. Fitur-fitur apa saja yang ada pada aplikasi Jago Sholat
3. Fitur apa saja yang dapat memasukan data
4. Tata cara penggunaan fitur pada aplikasi nya
5. Deskripsi Aplikasi :
6. Nama Aplikasi : Jago Sholat
7. Platform : Android (Minimal : IceCream
8. Koneksi Internet : IYA
9. Permission : Lokasi GPS
10. Berikut fitur-fitur yang terdapat pada aplikasi :
11. Mencatat Sholat
12. Statistik Sholat
13. Jadwal Sholat 5 waktu
14. Arah Kiblat
15. Arah Kompas
16. Tata Cara
17. Fitur apa saja yang dapat memasukan data

* Fitur Catat Ibadah

1. Berikut Langkah-langkah penggunaan fitur pada aplikasi
   1. Pencatatan ibadah

|  |  |
| --- | --- |
| Nomor Test | 1.a.1 |
| Judul | Menguji fitur pencatatan ibadah menekan tombol catat ibadah |
| Teknik | 1. Membuka aplikasi 2. Slide hingga ke menu catatan 3. Klik tombol catat ibadah 4. Muncul pop-up jadwal sholat yang dapat dipilih di jam tersebut |
| Kriteria keberhasilan | Muncul feedback toast bahwa berhasil mencatat ibadah |
| Kondisi khusus | Belum masuk waktu sholat tersebut |

* 1. Update waktu sholat

|  |  |
| --- | --- |
| Nomor Test | 1.a.2 |
| Judul | Menguji fitur update catatan pada menu statistic |
| Teknik | 1. Membuka aplikasi 2. Slide hingga ke menu statistik 3. Lalu klik salah satu catatan sholat 4. Lalu ubah waktunya dan klik ubah |
| Kriteria keberhasilan | Muncul feedback toast bahwa data berhasil di ubah |
| Kondisi khusus | - |

* 1. Statistik ibadah

|  |  |
| --- | --- |
| Nomor Test | 1.a.3 |
| Judul | Menguji fitur grafik menu statistic |
| Teknik | * + - 1. Membuka aplikasi       2. Slide hingga ke menu statistik       3. Klik button grafik |
| Kriteria keberhasilan | Ketika data sholat masuk maka setiap sholatnya diberi nilai 20 untuk dimasukkan kedalam grafik |
| Kondisi khusus | Jika tidak mencatat sholat maka nilai sama dengan 0. |

* 1. Validasi jadwal sholat

|  |  |
| --- | --- |
| Nomor Test | 1.a.3 |
| Judul | Validasi waktu sholat yang akan datang |
| Teknik | 1. Membuka aplikasi 2. Slide hingga ke menu jadwal sholat 3. Cek waktu sholat yang akan datang |
| Kriteria keberhasilan | Akan muncul teks mendatang dan dibawahnya muncul waktu sholat selanjutnya |
| Kondisi khusus | - |

**1b. Core Apps Quality**

Pengguna Android mengharapkan aplikasi berkualitas baik. Kualitas aplikasi berpengaruh langsung terhadap keberhasilan aplikasi untuk jangka panjang dalam hal pemasangan, rating dan ulasan pengguna, interaksi, dan loyalitas pengguna. Dokumen ini akan membantu dalam menilai aspek-aspek dasar kualitas dalam aplikasi melalui serangkaian ringkas kriteria kualitas aplikasi inti dan pengujian terkait.

Semua aplikasi Android harus memenuhi kriteria ini. Sebelum mempublikasikan aplikasi Anda, ujilah berdasarkan kriteria ini untuk memastikannya berfungsi dengan baik pada banyak perangkat, memenuhi standar Android untuk navigasi dan desain, serta siap mendapatkan peluang promosi di Google Play Store. Pengujian Anda akan melebihi apa yang digambarkan di sini—tujuan dokumen ini adalah menetapkan karakteristik kualitas mendasar yang harus ditunjukkan oleh semua aplikasi, sehingga bisa dimasukkan dalam rencana pengujian

Dalam testing Core Apps Quality yang harus dilakukan yaitu:

1. Mendeskripsikan Aplikasi Jago Sholat
2. Fitur-fitur apa saja yang ada pada aplikasi Jago Sholat
3. Fitur apa saja yang dapat memasukan data
4. Tata cara penggunaan fitur pada aplikasi nya
   * 1. Deskripsi Aplikasi :
   1. Nama Aplikasi : Jago Sholat
   2. Platform : Android (Minimal : IceCream)
   3. Koneksi Internet : IYA
   4. Permission : Lokasi GPS
      1. Berikut fitur-fitur yang terdapat pada aplikasi :
   5. Mencatat Sholat
   6. Statistik Sholat
   7. Jadwal Sholat 5 waktu
   8. Arah Kiblat
   9. Arah Kompas
   10. Tata Cara
       1. Fitur apa saja yang dapat memasukan data

* Fitur Catat Ibadah
  + 1. Berikut Langkah-langkah penggunaan fitur pada aplikasi

1. Unit testing
   * + 1. Pencatatan ibadah

|  |  |
| --- | --- |
| Nomor Test | 1.b.1 |
| Judul | Menguji fitur pencatatan ibadah menekan tombol catat ibadah |
| Teknik | * + 1. Membuka aplikasi     2. Slide hingga ke menu catatan     3. Klik tombol catat ibadah |
| Kriteria keberhasilan | Muncul toast bahwa berhasil mencatat ibadah |
| Kondisi khusus | Belum masuk waktu sholat tersebut |

1. Statistik Sholat

|  |  |
| --- | --- |
| Nomor Test | 1.b.2 |
| Judul | Statistik ibadah berupa grafik |
| Teknik | 1. Membuka aplikasi 2. Slide hingga ke menu statistik 3. Cek grafik |
| Kriteria keberhasilan | Ketika data sholat masuk maka setiap sholatnya diberi nilai 20 untuk dimasukkan kedalam grafik |
| Kondisi khusus | Jika tidak mencatat sholat maka nilai sama dengan 0. |

1. Jadwal Sholat 5 Waktu

|  |  |
| --- | --- |
| Nomor Test | 1.b.3 |
| Judul | Validasi waktu sholat yang akan datang |
| Teknik | 1. Membuka aplikasi 2. Slide hingga ke menu jadwal sholat 3. Cek waktu sholat yang akan datang |
| Kriteria keberhasilan | Akan muncul teks mendatang dan dibawahnya muncul waktu sholat selanjutnya |
| Kondisi khusus | - |

1. Arah Kiblat

|  |  |
| --- | --- |
| Nomor Test | 1.b.4 |
| Judul | Arah kiblat |
| Teknik | 1. Buka aplikasi 2. Buka menu kiblat 3. Cek arah panah icon ka’bah dari gambar kiblat |
| Kriteria keberhasilan | Arah menunjukan lokasi qiblat |
| Kondisi khusus | Tanpa hak akses lokasi, arah kiblat tidak akan ditemukan |

1. Arah Kompas

|  |  |
| --- | --- |
| Nomor Test | 1.b.5 |
| Judul | Arah kompas |
| Teknik | 1. Buka aplikasi 2. Buka menu kiblat 3. Cek gambar bulat berbentuk kompas, apakah berputar saat diputar device nya. |
| Kriteria keberhasilan | Arah menunjukan lokasi kompas yang benar |
| Kondisi khusus | Tanpa hak akses lokasi, arah kompas tidak akan ditemukan |

1. Tata Cara

* Tata Cara Wudhu

|  |  |
| --- | --- |
| Nomor Test | 1.b.6 |
| Judul | Menu tata cara wudhu |
| Teknik | 1. Buka aplikasi 2. Klik menu tata cara 3. Klik tata cara wudhu |
| Kriteria keberhasilan | Munculnya gambar dari tata cara wudhu yang bisa digeser ke kanan dan ke kiri |
| Kondisi khusus | - |

* Niat Sholat

|  |  |
| --- | --- |
| Nomor Test | 1.b.7 |
| Judul | Menu niat sholat |
| Teknik | 1. Buka aplikasi 2. Klik menu tata cara 3. Klik niat wudhu |
| Kriteria keberhasilan | Munculnya teks dari niat – niat sholat mulai dari sholat shubuh hingga sholat isya |
| Kondisi khusus | - |

* Tata Cara Shalat

|  |  |
| --- | --- |
| Nomor Test | 1.b.8 |
| Judul | Menu tata cara sholat |
| Teknik | 1. Buka aplikasi 2. Klik menu tata cara 3. Klik tata cara sholat |
| Kriteria keberhasilan | Munculnya gambar dari tata cara sholat yang bisa digeser ke kanan dan ke kiri dari mulai niat hingga salam |
| Kondisi khusus | - |

* Doa Setelah Shalat

|  |  |
| --- | --- |
| Nomor Test | 1.b.9 |
| Judul | Menu Doa Setelah sholat |
| Teknik | 1. Buka aplikasi 2. Klik menu tata cara 3. Klik doa sholat |
| Kriteria keberhasilan | Munculnya beberapa teks isi doa sholat berupa latin, terjemah, dan tulisan arabnya |
| Kondisi khusus | - |

1. Performance

|  |  |
| --- | --- |
| Nomor Test | 1b.10 |
| Judul | Menguji kecepatan membuka aplikasi |
| Teknik | 1. Membuka aplikasi JagoSholat dan menghitung waktu kecepatan buka aplikasi dengan stopwatch 2. Menghentikan stopwatch ketika aplikasi terbuka 3. Ulang sebanyak 3 kali |
| Kriteria keberhasilan | Membuka aplikasi tidak lebih dari 5 detik |
| Kondisi khusus | - |
| Tools | Smartphone Android dan Stopwatch |
|  |  |

|  |  |
| --- | --- |
| Nomor Test | 1b.11 |
| Judul | Menguji kecepatan menampilkan grafik |
| Teknik | * + 1. Membuka aplikasi JagoSholat     2. Memuka menu statistik     3. Pilih menu grafik     4. Hitung berapa detik waktu yang dibutuhkan dengan stopwatch     5. Menghentikan stopwatch ketika aplikasi terbuka     6. Ulang sebanyak 3 kali |
| Kriteria keberhasilan | Membuka grafik tidak lebih dari 5 detik |
| Kondisi khusus | - |
| Tools | Smartphone Android dan Stopwatch |
|  |  |

1. accessibility

|  |  |
| --- | --- |
| Nomor Test | 1b.12 |
| Judul | Menguji perpindahan activity |
| Teknik | 1. Buka aplikasi JagoSholat 2. Buka halaman (activity) yang berbeda secara terus menerus selama 2 menit |
| Kriteria keberhasilan | Aplikasi tidak terjadi error (force close) |
| Kondisi khusus | - |
| Tools | Smartphone Android dan Stopwatch |

**1c. Tablet Apps Quality**

Tablet adalah bagian penting dari basis yang dipasang Android dan menawarkan peluang bagus untuk keterlibatan pengguna dan monetisasi. Panduan di halaman ini akan membantu Anda memenuhi harapan pengguna tablet dengan menawarkan fitur menarik dan UI yang intuitif dan dirancang dengan baik.

* + 1. Unit testing

|  |  |
| --- | --- |
| Nomor Test | 1.c.1 |
| Judul | Pengujian Tampilan Pada Tablet |
| Teknik | 1. Menginstal aplikasi pada tablet 2. Memperhatikan tampilan pada tablet 3. Mencoba semua layout pada tablet |
| Kriteria keberhasilan | Tampilan muncul pada tablet view |
| Kondisi khusus | - |

* 1. Performance

|  |  |
| --- | --- |
| Nomor Test | 1c.2 |
| Judul | Menguji kecepatan membuka aplikasi |
| Teknik | 1. Membuka aplikasi JagoSholat dan menghitung waktu kecepatan buka aplikasi dengan stopwatch 2. Menghentikan stopwatch ketika aplikasi terbuka 3. Ulang sebanyak 3 kali |
| Kriteria keberhasilan | Membuka aplikasi tidak lebih dari 5 detik |
| Kondisi khusus | - |
| Tools | Smartphone Android dan Stopwatch |
|  |  |

|  |  |
| --- | --- |
| Nomor Test | 1c.3 |
| Judul | Menguji kecepatan menampilkan grafik |
| Teknik | * + 1. Membuka aplikasi JagoSholat     2. Memuka menu statistik     3. Pilih menu grafik     4. Hitung berapa detik waktu yang dibutuhkan dengan stopwatch     5. Menghentikan stopwatch ketika aplikasi terbuka     6. Ulang sebanyak 3 kali |
| Kriteria keberhasilan | Membuka grafik tidak lebih dari 5 detik |
| Kondisi khusus | - |
| Tools | Smartphone Android dan Stopwatch |

* 1. Accessibility

|  |  |
| --- | --- |
| Nomor Test | 1c.4 |
| Judul | Menguji perpindahan activity |
| Teknik | 1. Buka aplikasi JagoSholat 2. Buka halaman (activity) yang berbeda secara terus menerus selama 2 menit |
| Kriteria keberhasilan | Aplikasi tidak terjadi error (force close) |
| Kondisi khusus | - |
| Tools | Smartphone Android dan Stopwatch |

**1d. Automated Testing**

Automated testing yaitu yang bergantung pada pra-scripted tes yang berjalan secara otomatis, fungsinya untuk membandingkan hasil yang diharapkan dengan hasil yang sebenarnya. Sehingga dapat mengetahui apakah aplikasi berjalan sesuai dengan apa yang diharapkan, menggunakan automated testing dapat dilakukan secara berulang. Sehingga jika hasilnya tidak sama dengan yang diharapkan maka akan mendapatkan bug.

|  |  |
| --- | --- |
| Nomor Test | 1.d.1 |
| Judul | Automated Testing Menggunakan Script Kode |
| Teknik | 1. Menjalankan Project Android Studio Jago Sholat 2. Membuat file testing pada java package 3. Membuat script kode untuk automated testing 4. Menjalankan cript kode automated testing |
| Kriteria keberhasilan | Berhasil berjalan automated testing |
| Kondisi khusus | - |

**1e. Unconditional Testing**

Pengujian aplikasi menggunakan metode Unconditional Testing, yaitu pengujian tanpa syarat. Untuk Project kita hal hal yang di uji sebagai berikut :

|  |  |
| --- | --- |
| Nomor Test | 1.e.1 |
| Judul | Mencari bug secara umum |
| Teknik | 1. Menginstall aplikasi Jago Sholat pada Smartphone Xiaomi Redmi 4a 2. Membuka dan menutup aplikasi selamat 1 menit dengan total 15x buka tutup aplikasi 3. Mencoba semua tombol aplikasi secara acak 4. Mencari bug secara acak |
| Kriteria keberhasilan | Error pada aplikasi tidak di temukan |
| Kondisi khusus | - |

**1f. Code Quality Testing**

Code Quality Testing yaitu berfungsi untuk pengecekan keseluruhan source code aplikasi

|  |  |
| --- | --- |
| Nomor Test | 1.f.1 |
| Judul | Melakukan Tes terhadap code Aplikasi Jago Sholat |
| Teknik | * + - 1. Buka aplikasi Android Studio       2. Buka File project       3. Cari Tab Analyze       4. Klik Inspect Code |
| Kriteria keberhasilan | Error pada Code tidak ditemukan |
| Kondisi khusus | - |

**2a. Unit Testing**

Unit Testing merupakan rangkaian pengujian aplikasi secara koding, dimana setiap unit dalam pembuatan aplikasi diuji, seperti fungsi, class, prosedur, dan sejenisnya. hal ini ditujukan agar kita dapat mengetahui apakah sistem yang dibangun telah sesuai dengan rancangan awal atau tidak, serta mempermudah kita dalam melakukan debugging (pencarian kesalahan).

|  |  |
| --- | --- |
| Nomor Test | 2.a.1 |
| Judul | Menguji fungsi dari mendapatkan hasil dari GPS |
| Teknik | 1. Menjalankan project Android Studio 2. Memberikan tanda pada fungsi fetch GPS 3. Menjalankan aplikasi dalam mode debugging 4. Melihat hasil testing pada kolom run di android studio |
| Kriteria keberhasilan | Tidak terjadi eror pada fungsi |
| Kondisi khusus | - |

**2b. Data Diven Unit Test**

Data-driven testing (DDT) adalah metodologi pengujian perangkat lunak yang digunakan dalam pengujian perangkat lunak komputer untuk menggambarkan pengujian yang dilakukan menggunakan tabel kondisi secara langsung sebagai input pengujian dan output yang dapat diverifikasi serta proses di mana pengaturan dan kontrol lingkungan pengujian tidak kode keras. Dalam bentuk yang paling sederhana, penguji memasok input dari baris dalam tabel dan mengharapkan output yang terjadi di baris yang sama. Tabel biasanya berisi nilai-nilai yang sesuai dengan ruang input batas atau partisi. Dalam metodologi kontrol, konfigurasi tes "dibaca" dari database.

|  |  |
| --- | --- |
| Nomor Test | 2.b.1 |
| Judul | Kondisi ketika database kosong |
| Teknik | 1. Menjalankan project Android Studio 2. Memberikan tanda pada fungsi database kosong 3. Menjalankan aplikasi dalam mode debugging 4. Melihat hasil testing pada kolom run di android studio |
| Kriteria keberhasilan | Tidak terjadi eror maupun null data pada fungsi |
| Kondisi khusus | - |

|  |  |
| --- | --- |
| Nomor Test | 2.b.2 |
| Judul | Kondisi ketika database sudah ada data |
| Teknik | 1. Menjalankan project Android Studio 2. Memberikan tanda pada fungsi database sudah ada data 3. Menjalankan aplikasi dalam mode debugging 4. Melihat hasil testing pada kolom run di android studio |
| Kriteria keberhasilan | Tidak terjadi eror maupun null data pada fungsi |
| Kondisi khusus | - |

|  |  |
| --- | --- |
| Nomor Test | 2.b.3 |
| Judul | Kondisi ketika memasukkan data ke database |
| Teknik | 1. Menjalankan project Android Studio 2. Memberikan tanda pada fungsi memasukkan data ke database 3. Menjalankan aplikasi dalam mode debugging 4. Melihat hasil testing pada kolom run di android studio |
| Kriteria keberhasilan | Tidak terjadi eror maupun null data pada fungsi |
| Kondisi khusus | - |

|  |  |
| --- | --- |
| Nomor Test | 2.b.4 |
| Judul | Kondisi ketika menampilkan data dari database |
| Teknik | 1. Menjalankan project Android Studio 2. Memberikan tanda pada fungsi menampilkan data dari database 3. Menjalankan aplikasi dalam mode debugging 4. Melihat hasil testing pada kolom run di android studio |
| Kriteria keberhasilan | Tidak terjadi eror maupun null data pada fungsi |
| Kondisi khusus | - |

**2c. Performance**

1. Waktu Loading

|  |  |
| --- | --- |
| Nomor Test | 2.c.1 |
| Judul | Mengukur kecepatan waktu loading saat aplikasi dibuka |
| Teknik | 1. Buka aplikasi Jago Sholat bersamaan dengan stopwatch pada jam tangan 2. Menghentikan waktu stopwatch saat aplikasi sudah terbuka 3. Ulangi sebanyak beberapa kali 4. Lihat perbedaannya |
| Kriteria keberhasilan | Membuka Aplikasi kurang dari 5 detik  Dengan uji coba sebanyak 3 kali. Berhasil membuka aplikasi di waktu kurang dari 3 detik. |
| Kondisi khusus | - |
| Tools | Smartphone Android Xiaomi Redmi 5 (32/3) dan Stopwatch pada jam tangan Gshock GA-700 |

|  |  |
| --- | --- |
| Nomor Test | 2c.2 |
| Judul | Menguji kecepatan menampilkan grafik |
| Teknik | * + 1. Membuka aplikasi JagoSholat     2. Memuka menu statistik     3. Pilih menu grafik     4. Hitung berapa detik waktu yang dibutuhkan dengan stopwatch     5. Menghentikan stopwatch ketika aplikasi terbuka     6. Ulang sebanyak 3 kali |
| Kriteria keberhasilan | Membuka grafik tidak lebih dari 5 detik |
| Kondisi khusus | - |
| Tools | Smartphone Android dan Stopwatch |

1. Waktu loading grafik
2. Arah Kiblat dan kompas

|  |  |
| --- | --- |
| Nomor Test | 2.c.3 |
| Judul | Mengukur pergerakan pada arah kiblat dan kompas |
| Teknik | 1. Pastikan smartphone terkoneksi internet 2. Buka aplikasi Jago Sholat 3. Swipe dan masuk ke menu kiblat 4. Beri ijin permission lokasi |
| Kriteria keberhasilan | Arah kiblat dan kompas langsung menentukan arahnya secara otomatis dan halus pergerakannya di setiap smartphone diubah posisinya. |
| Kondisi khusus | - |
| Tools | Smartphone Android Xiaomi Redmi 5 (32/3) |

**2d. Accesbility**

Accessibility / Aksesibilitas adalah istilah umum yang digunakan untuk menggambarkan sejauh mana produk, perangkat, layanan, atau lingkungan yang tersedia untuk orang sebanyak mungkin. Aksesibilitas dapat dilihat sebagai "kemampuan untuk mengakses" dan manfaat yang mungkin dari beberapa sistem atau entitas. Aksesibilitas sering digunakan untuk fokus pada penyandang cacat atau kebutuhan khusus dan hak mereka akses kepada badan usaha, sering melalui penggunaan teknologi bantu

1. Database Insert data

|  |  |
| --- | --- |
| Nomor Test | 2.d.1 |
| Judul | Melakukan pencatatan sholat |
| Teknik | 1. Membuka aplikasi 2. Slide hingga ke menu catatan 3. Klik tombol catat ibadah |
| Kriteria keberhasilan | Muncul toast bahwa berhasil mencatat/insert data ibadah |
| Kondisi khusus | - |

1. Database Update data

|  |  |
| --- | --- |
| Nomor Test | 2.d.2 |
| Judul | Melakukan pengubahan waktu catatan sholat |
| Teknik | 1. Membuka aplikasi 2. Slide hingga ke menu statistik 3. Pilih salah satu catatan shalat lalu tekan 4. Muncul popup, lalu ubah waktu jam ibadah |
| Kriteria keberhasilan | Muncul toast bahwa waktu telah diubah. |
| Kondisi khusus | - |

|  |  |
| --- | --- |
| Nomor Test | 2d.3 |
| Judul | Menguji perpindahan activity |
| Teknik | 1. Buka aplikasi JagoSholat 2. Buka halaman (activity) yang berbeda secara terus menerus selama 2 menit |
| Kriteria keberhasilan | Aplikasi tidak terjadi error (force close) |
| Kondisi khusus | - |
| Tools | smartphone android dan Stopwatch |

**2e. Functionality**

Functionality Testing merupakan pengujian sistem terkait dengan kesesuaian fungsional dengan ekspektasi dari pengguna akhir (client), pengujian ini dimaksudkan agar pengguna telah memakai aplikasi yang telah sesuai dengan ekspektasi awalnya saat ingin menggunakan aplikasi.

|  |  |
| --- | --- |
| Nomor Test | 2.e.1 |
| Judul | Menguji fitur pencatatan ibadah menekan tombol catat ibadah |
| Teknik | 1. Membuka aplikasi 2. Slide hingga ke menu catatan 3. Klik tombol catat ibadah 4. Muncul pop-up jadwal sholat yang dapat dipilih di jam tersebut |
| Kriteria keberhasilan | Muncul feedback toast bahwa berhasil mencatat ibadah |
| Kondisi khusus | Belum masuk waktu sholat tersebut |

|  |  |
| --- | --- |
| Nomor Test | 2.e.2 |
| Judul | menampilkan waktu sholat jadwal sholat |
| Teknik | 1. Membuka aplikasi 2. Slide hingga ke menu jadwal sholat 3. Tampil jadwal sholat |
| Kriteria keberhasilan | Muncul jadwal sholat dari shubuh hingga isya |
| Kondisi khusus | - |

|  |  |
| --- | --- |
| Nomor Test | 2.e.3 |
| Judul | Menampilkan data catatan ibadah |
| Teknik | 1. Membuka aplikasi 2. Slide hingga ke menu statistik 3. Pilih menu data ibadah |
| Kriteria keberhasilan | Muncul data catatan ibadah |
| Kondisi khusus | Jika belum mencatat data tidak ada data |

;[

|  |  |
| --- | --- |
| Nomor Test | 2.e.4 |
| Judul | Arah kiblat |
| Teknik | 1. Membuka aplikasi 2. Slide hingga ke menu kiblat 3. Cek apakah arah kiblat sudah sesuai atau belum |
| Kriteria keberhasilan | Kiblat menunjukan arah yang seharusnya |
| Kondisi khusus | Jika gps tidak nyala |

|  |  |
| --- | --- |
| Nomor Test | 2.e.5 |
| Judul | Slide bar menu |
| Teknik | 1. Membuka aplikasi 2. Slide menu ke kiri dan ke kanan |
| Kriteria keberhasilan | Muncul seluruh tampilan menu |
| Kondisi khusus | - |

|  |  |
| --- | --- |
| Nomor Test | 2.e.6 |
| Judul | Menguji slide bar image |
| Teknik | 1. Membuka aplikasi 2. Slide hingga ke menu tatacara 3. Pilih tatacara sholat dan tata cara wudhu |
| Kriteria keberhasilan | Gambar berhasil di slide ke kiri dan ke kanan |
| Kondisi khusus |  |

**2f. Security**

Security Testing adalah teknik pengujian untuk menentukan jika sistem informasi melindungi data dan mempertahankan fungsi sebagaimana dimaksud. Dengan melakukan pengujian keamanan, Security Testing tidak menjamin bahwa sistem aman tetapi penting untuk menyertakan keamanan pengujian sebagai bagian dari proses pengujian.

Untuk Pengujian Security Testing aplikasi Jago Sholat menggunakan ADB tools, dan Mobile Security Framework.

|  |  |
| --- | --- |
| Nomor Test | 2.f.1 |
| Judul | Pengujian Keamanaan dan kerahasiaan dari database |
| Teknik | 1. Menggunakan Aplikasi khusus bawaan security testing 2. Mendapat hasil testing |
| Kriteria keberhasilan | Database tidak terlihat, serta tidak ada cela untuk SQL injection |
| Kondisi khusus | Tidak ada |

**2g. Usability**

Usability Testing adalah Jenis testing yang digunakan untuk evaluasi produk (aplikasi atau website) dengan pengujian sebagai pengguna.

|  |  |
| --- | --- |
| Nomor Test | 2.g.1 |
| Judul | Testing berdasarkan hasil kuisioner pengguna |
| Teknik | 1. Membuat kuisioner tentang aplikasi 2. Menguji pada 30 orang secara acak 3. Koresponden mencoba aplikasi 4. Koresponden mengisi kusioner sesuai hasil tes yang di coba koresponden |
| Kriteria keberhasilan | Mendapatkan informasi dari koresponden dengan laporan laporan dari setiap perangkat yang berbeda |
| Kondisi khusus | - |

**2h. Automated UI Test**

|  |  |
| --- | --- |
| Nomor Test | 2.h.1 |
| Judul | Melakukan Automated testing menggunakan Robotium Android Testing Tool. |
| Teknik | 1. Siapkan Java library dan ADT bundle 2. Buka Aplikasi Java ADT yang berisikan apk andoid yang akan ditesting 3. Buat nama android project testing 4. Lihat/cari elemen tombol |
| Kriteria keberhasilan | Akan muncul deskripsi tombol-tombol dan item beserta tipe layoutnya di bagian kanan atas aplikasi |
| Kondisi khusus | - |

**2i. Espreso**

Alat Espresso Test Recorder memungkinkan membuat pengujian UI untuk aplikasi tanpa menulis kode pengujian apa pun. Dengan merekam skenario pengujian, bisa merekam interaksi dengan sebuah perangkat dan menambahkan pernyataan untuk memverifikasi elemen UI dalam snapshot aplikasi tertentu. Espresso Test Recorder kemudian mengambil rekaman yang sudah disimpan dan secara otomatis menghasilkan pengujian UI yang sesuai yang bisa jalankan untuk menguji aplikasi.

|  |  |
| --- | --- |
| Nomor Test | 2.i.1 |
| Judul | Espresso Testing |
| Teknik | 1. Menjalankan Project Jago Sholat pada Android Studio 2. Menggunakan Library Espresso Testing 3. Membuat script testing untuk fungsi yang ada di aplikasi 4. Menjalankan mode debuggin espresso pada android studio |
| Kriteria keberhasilan | Tidak ada error ataupun testing yang failed. |
| Kondisi khusus | - |

**2j. Apk Analyze**

Android Studio menyertakan debugger yang memungkinkan Anda untuk men-debug aplikasi yang berjalan pada Android Emulator atau perangkat Android yang terhubung. Dengan debugger Android Studio :

* Memilih perangkat untuk men-debug aplikasi Anda.
* Menyetel breakpoint dalam kode Java dan C/C++ Anda.
* Memeriksa variabel dan mengevaluasi ekspresi pada saat waktu proses.
* Mengambil tangkapan layar dan video dari aplikasi Anda.

|  |  |
| --- | --- |
| Nomor Test | 2.j.1 |
| Judul | Analisa script kode |
| Teknik | 1. Buka Project Jago Sholat pada Android Studio 2. Build APK Debuging pada Android Studio 3. Melakukan Analisa pada APK yang telah di build debugging |
| Kriteria keberhasilan | Mendapatkan informasi mengenai script kode project aplikasi |
| Kondisi khusus | - |

**2k. Google Play**

Google menyediakan Fitur testing sebelum aplikasi di published ke Google Play Store, terlebih dahulu pada Google Play Console akan di lakukan testing dari aplikasi, mencari secara menyeluruh kemungkinan kemungkinan kesalahan fatal yang akan terjadi.

|  |  |
| --- | --- |
| Nomor Test | 2.k.1 |
| Judul | Google Testing Certifite |
| Teknik | 1. Buka Project Jago Sholat pada Android Studio 2. Generate Signed APK 3. Masukkan Key Store Path 4. Masukkan Key Store Password 5. Masukkan Key Alias 6. Masukkan Key Password 7. Generate file Jar dan APK 8. Unggah APK pada Google Play Console 9. Unggah APK pada App Release 10. Mengisi kuisioner pada Google Play Consol 11. Secara otomatis mendapatkan review testing dari Google Play Console |
| Kriteria keberhasilan | Mendapatkan content rating dari Google Play Console |
| Kondisi khusus | - |

1. **Item Pass / Fail Criteria**

|  |  |
| --- | --- |
| **Fitur** | **Kriteria** |
| Tombol catat ibadah | User hanya perlu menekan tombol catat ibadah, jika berhasil maka data akan masuk kedalam database dan tampil di menu statistik |
| Validasi waktu sholat | Tampil jadwal sholat sesuai dengan sholat pada jam tersebut |
| Validasi tanggal | Tampil tanggal sesuai dengan tanggal hari ini |
| Validasi jadwal sholat | Akan muncul teks jadwal sholat berupa sholat dan waktu. |
| Validasi waktu sholat yang akan datang | Akan muncul teks mendatang dan dibawahnya muncul waktu sholat selanjutnya |
| Hitung mundur waktu sholat selanjutnya | Akan muncul teks berupa waktu yang berkurang terus |
| Statistik ibadah | Ketika data sholat masuk maka setiap sholatnya diberi nilai 20 untuk dimasukkan kedalam grafik |
| Data ibadah | Tampil data ibadah dari hasil inputan user di menu pencatatan ibadah |
| Update data ibadah | Data terupdate atau terubah |
| Lokasi kita berada | Tampil pada menu kompas lokasi tempat kita berada |
| Arah kompas | Arah kompas yang dituju oleh gambar sesuai dengan arah kiblat |
| Menu tata cara wudhu | Menampilkan gambar dari mulai niat wudhu hingga niat selesai wudhu |
| Geser gambar | Gambar dapat di geser ke kiri dan ke kanan |
| Menu niat sholat | Menampilkan niat berupa teks dari niat sholat shubuh hingga sholat isya |
| Menu tata cara sholat | Menampilkan gambar dari mulai niat sholat hingga selesai salam. |
| Tampil do’a setelah sholat | Menampilkan beberapa doa – doa setelah selesai sholat |

1. **Suspension Criteria**

Proses Testing akan di suspend atau di skip ketika fungsi yang terdapat pada aplikasi tidak bisa di jalankan dan tidak memiliki fungsi.

1. **Test Deliverables**
2. Master test plan
3. Screen Prototype
4. **Testing Task**

|  |  |
| --- | --- |
| **Task** | **Assigned To** |
| Create Acceptance Test Plan | Project Manager, Client |
| Create Integration Test Plan | Developers, Project Manager |
| Define Unit Test Rules and Procedures | Developers, Project Manager |
| Define Turnover procedures for each level | Developers |
| Verify prototypes of Screens | Clients, Project Manager |
| Verify Prototypes of Reports | Client, Project Manager |

1. **Enviromental Needs**

Spesifikasi dan Perincian segala sesuatu yang dibutuhkan dan digunakan selama proses testing berjalan, bisa berupa hardware yaitu spesifikasi handphone atau hal lain selain hardware.

1. Xiaomi Redmi 4a

|  |  |  |
| --- | --- | --- |
| Operating System | : | Android OS Nouggat(7.1) |
| Language | : | Indonesian (Regional Setting: Indonesian) |
| Processor | : | Qualcomm MSM8917 snapdragon 425 , quad-core 1.4 GHz Cortex-A53, GPU Adreno 308 |
| Memory Internal | : | 16 GB |
| RAM | : | 2GB RAM |

1. Xiaomi Redmi 5

|  |  |  |
| --- | --- | --- |
| Operating System | : | Android OS Nouggat (7.1) |
| Language | : | Indonesian (Regional Setting: Indonesian) |
| Processor | : | Qualcomm snapdragon 4250, quad-core 1.8 GHz Cortex-A53, GPU Adreno 506 |
| Memory Internal | : | 32 GB |
| RAM | : | 3 GB RAM |

1. Xiaomi Redmi Note 5

|  |  |  |
| --- | --- | --- |
| Operating System | : | Android OS Oreo (8.1) |
| Language | : | Indonesian (Regional Setting: Indonesian) |
| Processor | : | Qualcomm SDM636 snapdragon 636 , quad-core 1.8 GHz Kryo 260, GPU Adreno 509 |
| Memory Internal | : | 32 GB |
| RAM | : | 3GB RAM |

1. **Responsibilities**

Membuat Test Plan : Muhammad Faisal Amir, Muhamad Ikhsan Ramadhan, Bryan Rafsanzani

Membuat Design Test : Muhammad Faisal Amir, Muhamad Ikhsan Ramadhan, Bryan Rafsanzani

Melakukan Testing : Muhammad Faisal Amir, Muhamad Ikhsan Ramadhan, Bryan Rafsanzani

Membuat Dokumen Final : Muhammad Faisal Amir, Muhamad Ikhsan Ramadhan, Bryan Rafsanzani

1. **Staffing and Training Needs**

Tugas-Tugas yang dilakukan

1. Project Manager :  Bertanggung Jawab untuk mengatur semua penerapan dalam melakukan pengujian pada aplikasi android
2. Test Manager : Bertanggung jawab mengembangkan test plan
3. Test Engineer : Bertanggung jawab membuat rancangan desain dari testing, proses atau langkah melakukan testing,  dan menjalankan testing

|  |  |
| --- | --- |
| Project Manager | Thahir Cherid |
| Test Manager | Muhamad Ikhsan Ramadhan |
| Test Engineer | Bryan Rafsanzani |

1. **Schedule**

|  |  |  |
| --- | --- | --- |
| Test Phase | Responsible person | Time |
| Test Plan Creation | Muhamad Ikhsan Ramadhan | 2 week |
| Unit Testing | Bryan Rafsanzani | Developing Time |
| Features Testing | Muhammad Faisal Amir | 1 week |
| Integration Testing | Muhamad Ikhsan Ramadhan | 1 week |
| Use Case Validation | Muhamad Faisal Amir | 1 week |
| User Interface | Bryan Rafsanzani | 1 week |
| Load Testing | Bryan Rafsanzani | 1 week |
| Performance Testing | Muhamad Ikhsan Ramadhan | 1 week |
| Release to Production | Muhammad Faisal Amir | 1 week |

1. **Risk and Contingencies**

Tidak ada pemberitahuan membutuhkan koneksi internet, jadi ketika layanan internet pada smartphone tidak menyala, aplikasi tidak memunculkan pemberitahuan eror seperti tidak terjadi apa – apa.

Tidak memanfaatkan cache data sehingga ketika sudah pernah melakukan Masuk dan keluar dari aplikasi, sistem akan minta user untuk Masuk ulang dikarena data Masuk aplikasi tersebut tidak tersimpan pada cache data.

1. **Approvals**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nama** | **Versi yang disetujui** | **Jabatan** | **Tanggal** | **Tanda Tangan** |
| Muhamad Ikhsan Ramadhan | TP V.1 | Pembuat | 26/9/18 |  |
| Muhammad Faisal Amir |  | Pembuat |  |  |
| Bryan Rafsanzani |  | Pembuat |  |  |
| Muhamad Ikhsan Ramadhan |  | Pembuat |  |  |