

**SISTEM INFORMASI GEOGRAFIS PEMETAAN LOKASI
PONDOK PESANTREN BERBASIS ANDROID DI
KABUPATEN PURBALINGGA**



SKRIPSI

diajukan sebagai salah satu syarat untuk memperoleh gelar
Sarjana Komputer

OLGA PUTRI PERMATASARI

1203040021

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS MUHAMMADIYAH PURWOKERTO
DESEMBER, 2016**

HALAMAN PERNYATAAN

Saya yang bertanda tangan di bawah ini :

Nama : Olga Putri Permatasari
NIM : 1203040021
Program Studi : Teknik Informatika
Fakultas : Teknik
Perguruan Tinggi : Universitas Muhammadiyah Purwokerto

Menyatakan dengan sebenar-benarnya bahwa skripsi adalah hasil karya saya dan semua sumber yang dikutip maupun dirujuk telah saya cantumkan dengan benar.

Demikian pernyataan ini saya buat dan apabila kelak di kemudian hari terbukti ada unsur penjiplakan, saya bersedia mempertanggungjawabkan sesuai dengan ketentuan yang berlaku.

Purwokerto, 01 Desember 2016

Yang membuat pernyataan



Olga Putri Permatasari

HALAMAN PERSETUJUAN

Skripsi yang diajukan oleh:

Nama : Olga Putri Permatasari
NIM : 1203040021
Program Studi : Teknik Informatika
Fakultas : Teknik
Perguruan Tinggi : Universitas Muhammadiyah Purwokerto
Judul : Sistem Informasi Geografis Pemetaan Lokasi Pondok Pesantren Berbasis *Android* di Kabupaten Purbalingga

telah diterima dan disetujui
Purwokerto, 04 Januari 2017

PEMBIMBING



Harjono, S.T., M.Eng.

NIK. 2160389

HALAMAN PENGESAHAN

Skripsi yang diajukan oleh :

Nama : Olga Putri Permatasari
NIM : 1203040021
Program Studi : Teknik Informatika
Fakultas : Teknik
Perguruan Tinggi : Universitas Muhammadiyah Purwokerto
Judul : Sistem Informasi Geografis Pemetaan
Lokasi Pondok Pesantren Berbasis *Android*
di Kabupaten Purbalingga

telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Komputer (S.Kom) pada Program Studi Teknik Informatika, Fakultas Teknik, Universitas Muhammadiyah Purwokerto.

DEWAN PENGUJI

Penguji 1 (Pembimbing) : Harjono, S.T., M.Eng.
Penguji 2 : Dimara Kusuma Hakim, S.T., M.Cs.
Penguji 3 : Tito Pinandita, S.Si., M.Kom.
Ditetapkan di : Purwokerto
Tanggal : 04 Januari 2017

Mengetahui

Dekan Fakultas Teknik



M. Taufiq Tamam, S.T., M.T.

NIK 2160223

HALAMAN PERSEMBAHAN

Tugas Akhir ini Kupersembahkan kepada :

1. Mamah, Papahku tercinta atas dukungan dan doa yang telah diberikan. Semoga ini jadi penghargaan bagi kalian semua.
2. Kedua Almarhum Kakakku tersayang yang selama hidupnya memberikan motivasi dan kasih sayang. Semoga kalian disurga bangga melihat kesuksesanku.
3. Bapak, Ibu Dosen Program Studi Teknik Informatika yang tak lelah memberikan ilmu dan motivasi.
4. Teman-teman Teknik Informatika angkatan 2012 kelas A, yang telah membantu dan mendukung sehingga saya mampu menyelesaikan Tugas Akhir ini.
5. Teman-teman Klasika Home yang selalu menghibur dan memberikan motivasi sampai akhir.

Purwokerto, 01 Desember 2016

Olga Putri Permatasari

HALAMAN MOTTO

“Jangan takut untuk mencoba sesuatu yang baru”



KATA PENGANTAR

Alhamdulillah puji syukur penulis ucapkan atas kehadiran Allah SWT atas limpahan rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan laporan tugas akhir dengan judul “Sistem Informasi Geografis Pemetaan Lokasi Pondok Pesantren Berbasis *Android* di Kabupaten Purbalingga”. Penulis menyadari tanpa bantuan dari berbagai pihak, tugas akhir ini tidak mungkin terselesaikan dengan baik. Oleh karena itu penulis menyampaikan terimakasih kepada yang terhormat:

1. Bapak Harjono, S.T., M.Eng. Sebagai dosen pembimbing.
2. Bapak Muhammad Hamka Sebagai dosen pembimbing akademik Program Studi Teknik Informatika angkatan 2012.
3. Segenap dosen Fakultas Teknik Universitas Muhammadiyah Purwokerto yang telah memberikan banyak ilmu dan bimbingannya selama penulis dalam masa perkuliahan.
4. Semua pihak yang telah membantu dalam tugas akhir ini.

Penulis menyadari masih banyak kekurangan dalam penulisan laporan maupun penelitian yang dijalankan. Oleh karena itu penulis meminta saran terhadap pembaca dalam usaha perbaikan dan pengembangan di masa mendatang.

Purwokerto, 01 Desember 2016

Olga Putri Permatasari

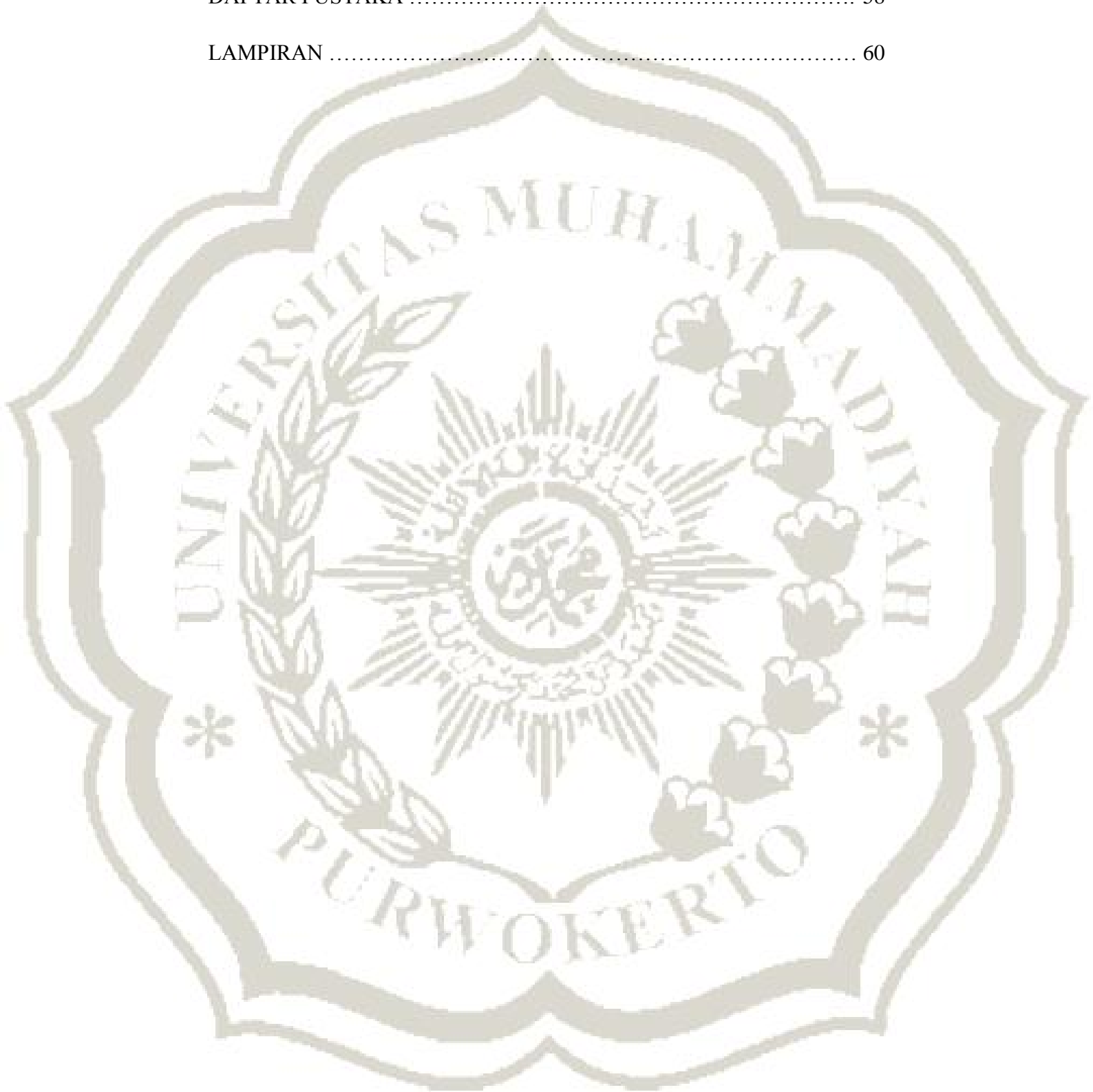
DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PERNYATAAN	ii
HALAMAN PERSETUJUAN	iii
HALAMAN PENGESAHAN	iv
HALAMAN PERSEMBAHAN	v
HALAMAN MOTTO	vi
KATA PENGANTAR	vii
DAFTAR ISI	viii
DAFTAR TABEL	xi
DAFTAR GAMBAR	xii
DAFTAR LAMPIRAN	xv
INTISARI	xvi
<i>ABSTRACT</i>	xvii
BAB I. PENDAHULUAN	1
A. Latar Belakang	1
B. Rumusan Masalah	3
C. Batasan Masalah	3
BAB II. KAJIAN PUSTAKA	4
A. Kabupaten Purbalingga	4
B. Pondok Pesantren	5

C. Sistem Informasi	6
D. Sistem Informasi Geografis	6
E. <i>Android</i>	8
F. <i>Google Maps</i>	9
G. <i>Hypertext Preprocessor</i>	10
H. <i>MySql</i>	11
I. <i>Java Script Object Notation</i>	11
J. Hasil Penelitian Sejenis	13
 BAB III. TUJUAN DAN MANFAAT	 15
A. Tujuan	15
B. Manfaat	15
 BAB IV. METODE PENELITIAN	 16
A. Jenis Penelitian	16
B. Waktu dan Tempat Penelitian	16
C. Pengumpulan Data	16
1. Dokumentasi	16
2. <i>Data Global Positioning System</i>	17
D. Alat	17
E. Pengembangan Sistem	18
 BAB V. HASIL DAN PEMBAHASAN	 43
A. Kebutuhan Data	43
B. Implementasi Sistem	43
1. Aplikasi <i>Website Admin</i>	44
2. Aplikasi <i>Android</i>	49
 BAB VI. PENUTUP	 57
A. Kesimpulan	57
B. Saran	57

DAFTAR PUSTAKA 58

LAMPIRAN 60



DAFTAR TABEL

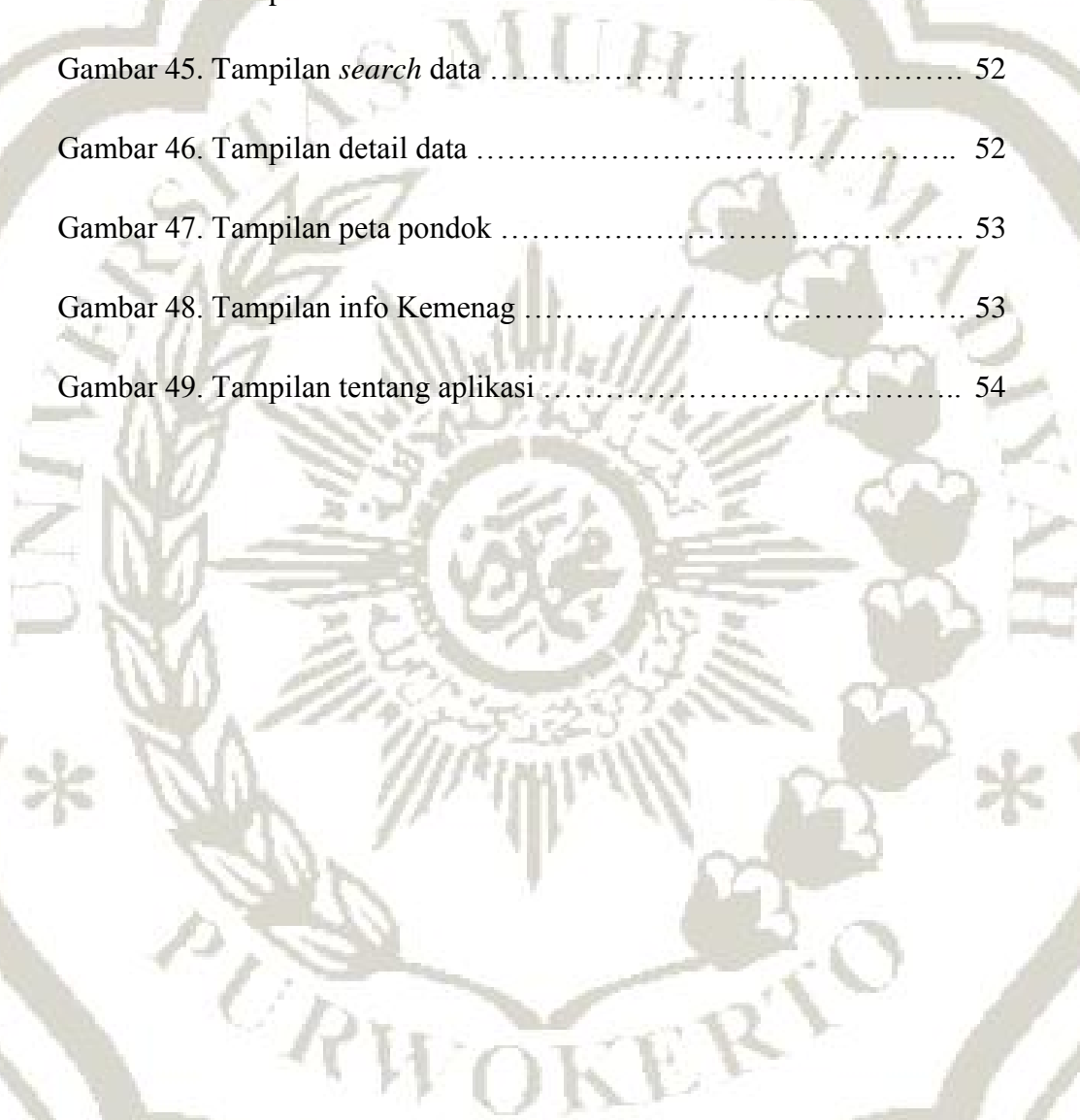
Tabel 1. Rancangan tabel data admin	26
Tabel 2. Rancangan tabel data pondok	27
Tabel 3. Rancangan tabel artikel	28
Tabel 4. Pengujian aplikasi <i>android</i>	55
Tabel 5. Pengujian aplikasi <i>website</i>	56

DAFTAR GAMBAR

Gambar 1. Komponen kunci GIS	7
Gambar 2. Model pengembangan sistem <i>waterfall</i>	19
Gambar 3. <i>Use case diagram</i> aplikasi <i>web admin</i>	21
Gambar 4. <i>Use case diagram</i> aplikasi <i>android</i>	21
Gambar 5. <i>Activity diagram</i> mengelola data pondok	22
Gambar 6. <i>Activity diagram</i> mengelola artikel	23
Gambar 7. <i>Activity diagram</i> melihat data pondok	24
Gambar 8. Relasi <i>database</i>	29
Gambar 9. Rancangan halaman <i>login</i> aplikasi <i>website admin</i>	30
Gambar 10. Rancangan halaman <i>home</i> aplikasi <i>website admin</i>	30
Gambar 11. Rancangan halaman <i>list</i> data pondok	31
Gambar 12. Rancangan halaman detail data pondok	31
Gambar 13. Rancangan halaman tambah data pondok	32
Gambar 14. Rancangan halaman ubah data pondok	32
Gambar 15. Rancangan <i>list</i> artikel	33
Gambar 16. Rancangan detail artikel	34
Gambar 17. Rancangan tambah artikel	34
Gambar 18. Rancangan ubah artikel	35
Gambar 19. Rancangan galeri pondok	35

Gambar 20. Rancangan tampilan <i>icon</i> di menu <i>smartphone</i>	36
Gambar 21. Rancangan tampilan <i>splash screen</i>	36
Gambar 22. Rancangan tampilan utama aplikasi	37
Gambar 23. Rancangan tampilan laci menu	38
Gambar 24. Rancangan tampilan <i>list data</i>	38
Gambar 25. Rancangan tampilan <i>search data</i>	39
Gambar 26. Rancangan tampilan detail data	39
Gambar 27. Rancangan tampilan peta pondok	40
Gambar 28. Rancangan tampilan info Kemenag	40
Gambar 29. Rancangan tampilan tentang aplikasi	41
Gambar 30. Halaman <i>login</i>	44
Gambar 31. Tampilan gagal <i>login</i>	45
Gambar 32. Tampilan <i>home</i>	45
Gambar 33. Tampilan <i>list pondok</i>	46
Gambar 34. Tampilan tambah data	46
Gambar 35. Tampilan detail data	47
Gambar 36. Tampilan ubah data	47
Gambar 37. Tampilan <i>list artikel</i>	48
Gambar 38. Tampilan tambah artikel	48
Gambar 39. Tampilan galeri pondok	49
Gambar 40. Tampilan <i>icon</i> di menu <i>smartphone</i>	49

Gambar 41. Tampilan <i>splash screen</i>	50
Gambar 42. Tampilan menu artikel	50
Gambar 43. Tampilan laci menu	51
Gambar 44. Tampilan <i>list data</i>	51
Gambar 45. Tampilan <i>search data</i>	52
Gambar 46. Tampilan detail data	52
Gambar 47. Tampilan peta pondok	53
Gambar 48. Tampilan info Kemenag	53
Gambar 49. Tampilan tentang aplikasi	54



DAFTAR LAMPIRAN

Lampiran 1. Kode Program	62
Lampiran 2. Data Pondok Pesantren Purbalingga	116



INTISARI

Pondok pesantren mempunyai peranan penting dalam perkembangan ilmu keislaman dan sosial budaya karena dapat merubah daerah yang memiliki budaya negatif dan menjadikan sebagai benteng dalam menghadapi jaman sekarang yang mengkhawatirkan. Maka dari itu banyak orang tua mempunyai keinginan anaknya memiliki kepribadian, ahlak dan moral beragama yang baik sehingga menempatkan anak mereka ke pondok pesantren. Terbatasnya informasi menjadikan masyarakat belum mengetahui kualitas maupun lokasi masing-masing pondok pesantren. Dengan adanya aplikasi sistem informasi geografis pondok pesantren ini diharapkan pengguna dapat mengetahui informasi tentang pesantren sehingga menjadikan sebuah pertimbangan untuk memilih pesantren. Aplikasi ini dilengkapi dengan layanan lokasi sehingga pengguna lebih mudah mengetahui arah menuju ke pondok pesantren.

Kata Kunci : Pondok Pesantren, *Android*, Sistem Informasi Geografis.

ABSTRACT

Boarding schools have an important role in the development of Islamic science and socio-cultural as it can change the areas that have a negative culture and make a fort to face today's worrisome. Therefore, parents that desire to have children with a good personality, moral and religious moral, will putting their children to boarding school. The limited information made public don't know the quality and location of each boarding. With this application of geographic information system, users are expected to find information about the boarding schools so as to make a judgment to choose boarding school. This application is equipped with location services so that users can determine the direction leading to the boarding school more easily.

Keywords: Boarding school, Android, Geographic Information System.

BAB I

PENDAHULUAN

A. Latar Belakang

Perkembangan teknologi yang cepat telah berdampak dengan kebutuhan informasi dan kemajuan manusia. Dengan berkembangnya teknologi berpengaruh besar dalam pengelolaan informasi secara lebih aktual dan optimal. Penggunaan teknologi bertujuan untuk mencapai efisiensi dalam segala aspek pengelolaan informasi, yang dibuktikan dengan kecepatan pemrosesan dan keakuratan sebuah informasi.

Kebutuhan manusia tidak hanya dilandasi dengan pendidikan umum saja tetapi pendidikan agama sangat berperan penting dalam perkembangan manusia untuk menjadikan kepribadian yang baik. Di dalam pondok pesantren mengajarkan tentang pendidikan Islam, dakwah, pengembangan kemasyarakatan dan pendidikan lainnya. Pondok pesantren didirikan untuk memahami, menghayati dan mengamalkan ajaran Islam dengan menekankan moral Islam sebagai pedoman hidup bermasyarakat dalam kesehariannya.

Pondok pesantren mempunyai peranan penting dalam perkembangan ilmu keislaman dan sosial budaya karena dapat merubah daerah yang memiliki budaya negatif dan menjadikan sebagai benteng dalam menghadapi jaman sekarang yang mengkhawatirkan. Pesantren juga salah satu lembaga pendidikan yang menjadikan generasi muda yang dapat bersaing pada zaman sekarang yang berlandaskan Islam. Maka dari itu

banyak orang tua mempunyai keinginan anaknya memiliki kepribadian, ahlak dan moral beragama yang baik sehingga menempatkan anak mereka ke pondok pesantren.

Pentingnya peranan pondok pesantren di Kabupaten Purbalingga menjadi motivasi untuk membangun dan merancang sistem dalam mencari letak lokasi pesantren yang belum tentu masyarakat mengetahui dan memberikan informasi tentang pondok pesantren tersebut. Media yang digunakan yaitu *Google Maps* yang merupakan peta *online* yang disediakan oleh *Google*. Penerapan Sistem Informasi Geografis (SIG) sangat berpengaruh penting dalam menemukan lokasi pondok pesantren yang terdapat di Kabupaten Purbalingga.

Masyarakat di Kabupaten Purbalingga khususnya para orang tua yang ingin mendaftarkan anaknya tetapi terkadang masih kesulitan dalam mencari letak lokasi pondok pesantren dan belum mengetahui informasi pondok pesantren tersebut. Dengan adanya aplikasi ini diharapkan mempermudah masyarakat Kabupaten Purbalingga untuk menemukan lokasi dan mendapatkan informasi sehingga menjadikan sebuah pertimbangan orang tua untuk mendaftarkan anaknya ke pondok pesantren.

B. Rumusan Masalah

Bagaimana merancang dan membangun aplikasi sistem informasi geografis berbasis *android* dengan pemetaan lokasi pondok pesantren di Kabupaten Purbalingga?

C. Batasan Masalah

1. Penelitian ini hanya membahas pondok pesantren yang memiliki lebih dari 10 santri di Kabupaten Purbalingga.
2. Informasi yang ditampilkan dalam aplikasi ini meliputi lokasi, gambar, serta profil pondok pesantren.
3. Jenis aplikasi yaitu aplikasi *smartphone* berbasis *android*, khususnya *android* versi 4.0 (*Jelly Bean*) atau versi setelahnya.

BAB II

KAJIAN PUSTAKA

A. Kabupaten Purbalingga

Kabupaten Purbalingga yang memiliki luas wilayah 7.777,64 kilometer persegi ini berbatasan dengan Kabupaten Pematang di utara, Kabupaten Banjarnegara di timur dan selatan, dan Kabupaten Banyumas di barat. Purbalingga yang berpenduduk 848.952 jiwa (berdasarkan Sensus Penduduk Tahun 2010, *red.*) ini dikenal sebagai kabupaten yang *pro-investasi*. Hal ini terlihat dari banyaknya industri kecil hingga besar yang tumbuh dan berkembang di Purbalingga.

Bahkan, puluhan industri penanaman modal asing (PMA) yang sebagian besar berasal dari Korea Selatan juga banyak berdiri di Purbalingga guna mengembangkan industri pembuatan rambut dan bulu mata palsu. Keberadaan industri rambut dan bulu palsu di Purbalingga juga berdampak pada berkembangnya ratusan plasma-plasma dari perusahaan tersebut, sehingga dapat meningkatkan pendapatan masyarakat dengan bekerja sampingan sebagai pembuat rambut dan bulu mata palsu.

Selain sektor industri, di Purbalingga juga banyak terdapat kerajinan yang dikembangkan masyarakat setempat, antara lain pembuatan knalpot, gula kelapa, dan sapu glagah. Kendati demikian, sektor pertanian tetap memberikan kontribusi terbesar bagi PDRB Purbalingga, yakni sebesar

31,98 persen disusul sektor perdagangan, hotel, restoran yang sebesar 18,51 persen serta sektor jasa sebesar 17,98 persen.

Kabupaten Purbalingga juga dikenal sebagai salah satu sentra penghasil sayuran terutama cabai serta buah stroberi yang berlokasi di lereng Gunung Slamet sebelah tenggara. Selain itu, keberadaan Purbalingga juga semakin dikenal dengan berbagai perkembangan objek wisata buatan di kabupaten ini, antara lain Owabong dan Sanggaluri Park.

Purbalingga yang berada di persimpangan jalan utama penghubung Purwokerto (Kabupaten Banyumas) dengan Banjarnegara maupun Purwokerto dengan Pematang, menjadikan posisi kabupaten ini menjadi sangat strategis guna mendukung perekonomian daerah sekitarnya.

Oleh karena itu, Pemerintah Kabupaten Purbalingga bersama sejumlah kabupaten lainnya, yakni Banyumas, Kebumen, Banjarnegara, dan Wonosobo, berupaya agar Pangkalan Udara (Lanud) Wirasaba dapat dikembangkan menjadi sebuah bandara komersial (<http://www.jatengprov.go.id>, 2016).

B. Pondok Pesantren

Pesantren merupakan salah satu lembaga pendidikan Islam tertua di Indonesia. Pesantren juga memiliki hubungan fungsional simbiotik dengan ajaran Islam. Yaitu, dari satu sisi keberadaan pesantren diwarnai oleh corak dan dinamika ajaran Islam yang dianut oleh para pendiri dan kiai

pesantren yang mengasuhnya, sedangkan pada sisi lain menjadi jembatan utama bagi proses internalisasi dan transmisi ajaran Islam kepada masyarakat. Melalui pesantren agama Islam menjadi membumi dan mewarnai seluruh aspek kehidupan masyarakat: sosial, keagamaan, hukum, politik, pendidikan, lingkungan, dan lain sebagainya (Nata, 2012).

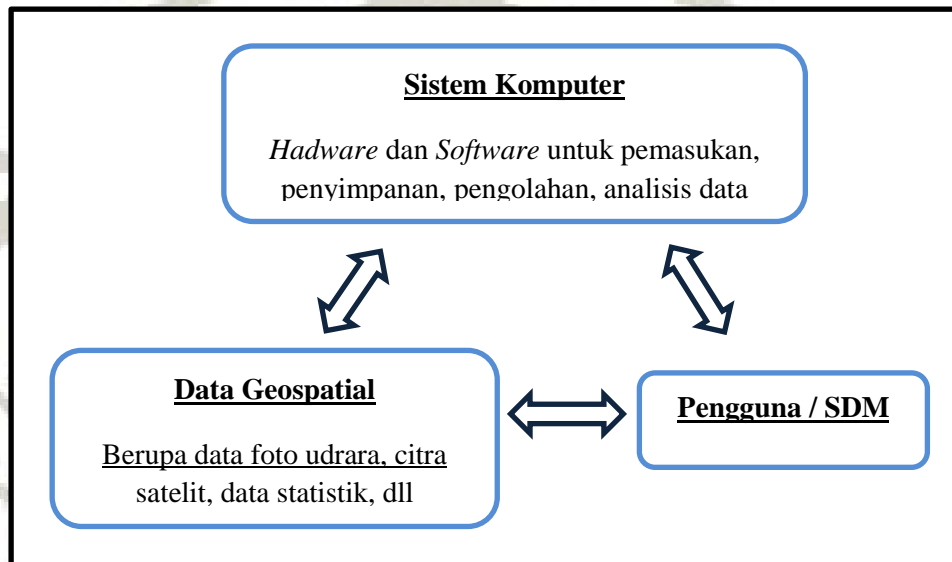
C. Sistem Informasi

Sistem informasi dapat didefinisikan sebagai suatu sistem di dalam suatu organisasi yang merupakan kombinasi dari orang – orang, fasilitas, teknologi, media, prosedur – prosedur dan pengendalian yang ditujukan untuk mendapatkan jalur komunikasi penting, memproses tipe transaksi rutin tertentu, memberi sinyal kepada manajemen dan yang lainnya terhadap kejadian – kejadian internal dan eksternal yang penting dan menyediakan suatu dasar informasi untuk pengambilan keputusan yang cerdas (Hartono, 1999).

D. Sistem Informasi Geografi

Sistem Informasi Geografi (SIG) merupakan suatu kesatuan formal yang terdiri dari berbagai sumberdaya fisik dan logika yang berkenaan dengan objek – objek yang terdapat di permukaan bumi. SIG juga merupakan sejenis perangkat lunak yang dapat digunakan untuk pemasukan, penyimpanan, manipulasi, dan keluaran informasi geografis berikut atribut – atributnya (Prahasta, 2005). Komponen kunci dalam SIG

adalah sistem komputer, data geospasial (data atribut) dan pengguna yang dapat digambarkan seperti pada Gambar 1.



Gambar 1. Komponen kunci GIS

SIG mempunyai perbedaan dengan bentuk sistem informasi lainnya, perbedaan itu adalah SIG dapat bekerja dengan data spasial. SIG mempunyai kemampuan untuk menghubungkan layer – layer data suatu titik yang sama dalam satu ruang, serta mengkombinasikan, menganalisis, dan memetakan hasilnya.

Di dalam SIG terdapat dua jenis data, yaitu data spasial dan data non-spasial. Berikut penjelasan dari dua jenis data tersebut.

a. Data Spasial

Data spasial merupakan data yang memuat tentang lokasi suatu objek dalam peta berdasarkan posisi geografis objek tersebut dalam bumi dengan menggunakan sistem koordinat.

b. Data Non-Spasial

Data ini merupakan data yang memuat karakteristik atau keterangan dari suatu objek yang terdapat dalam peta yang sama sekali tidak berkaitan dengan posisi geografi objek tertentu. Sebagai contoh data atribut dari sebuah kota adalah luas wilayah, jumlah penduduk, kepadatan penduduk, tingkat kriminalitas dan sebagainya.

E. *Android*

Android adalah sistem operasi berbasis *Linux* yang dirancang untuk perangkat seluler layar sentuh seperti telepon pintar dan komputer tablet. *Android* pada awalnya dikembangkan oleh *Android, Inc.*, dengan dukungan finansial dari *Google*, yang kemudian membelinya pada tahun 2005. Sistem operasi ini dirilis secara resmi pada tahun 2007 (Agung, 2015).

Android adalah sistem operasi *open source* dan *Google* merilis kodenya dibawah Lisensi *Apache*. Kode *open source* dan lisensi pada *Android* memungkinkan perangkat lunak untuk dimodifikasi secara bebas dan didistribusikan oleh para pengembang perangkat, operator nirkabel, dan pengembang aplikasi.

F. Google Maps

Google telah menyediakan layanan bagi para *developer* aplikasi *Android* untuk menggunakan *API Google Map* pada aplikasi mereka. *API* sendiri adalah singkatan dari *Application Programming Interface*. Selain itu, dokumentasi dan forum yang membahas penggunaan *API Google Map* pada aplikasi *android* sudah tersedia banyak di situs *developer android*. Ada dua macam penggunaan *API Google Map*, yaitu *API Google Map* standar dan *API Google Map* untuk bisnis.

Perbedaan antara *API Google Map* standar dan untuk bisnis adalah *API Google Map* standar bersifat gratis, sedangkan *API Google Map* bisnis berbayar. Kelebihan dari *API Google Map* untuk bisnis, antara lain kuota akses yang disediakan dalam permintaan layanan *API Google Map* lebih besar. Selain itu, adanya support yang memadai dari *Google Map* agar aplikasi yang menggunakan *API Google Map* bisnis memuaskan.

Perlu diketahui bahwa perkembangan penggunaan *Google Map* di *Android* yang ada pada saat ini dimulai dengan adanya *Google Map V1*, yang penggunaannya telah dihentikan pada akhir 2012. Mulai tahun 2013 aplikasi *android* yang ingin menggunakan/menampilkan *google map*, harus menggunakan layanan *google map v2*. Ada perbedaan yang mencolok dalam penerapan *source code* antara *google map v1* dengan *google map v2*. Mulai dari penggunaan SHA1 yang menggantikan MD5 untuk mendapatkan *Google API Key*, hingga penggunaan *Fragment* yang

menggantikan *MapView*. Kita juga harus menginstal *library google-play-services* terlebih dahulu pada *android SDK*.

Sebagaimana pada pengembangan *Google Map V1*, pada pengembangan aplikasi pada *Google Map V2*, kita juga mendapat akses untuk menambahkan marker (penanda lokasi), baik dengan ikon yang telah tersedia maupun menggunakan ikon kita sendiri. Marker juga dapat dilengkapi dengan keterangan, sehingga ketika marker ditekan akan muncul informasi yang disisipkan (Mufti, 2015).

G. *HyperTextPreprocessor (PHP)*

PHP Pertama kali ditemukan pada 1995 oleh seorang *Software Developer* bernama Rasmus Lerdorf. Ide awal PHP adalah ketika itu Rasmus ingin mengetahui jumlah pengunjung yang membaca resume onlinenya. *Script* yang dikembangkan baru dapat melakukan dua pekerjaan, yakni merekam informasi *visitor*, dan menampilkan jumlah pengunjung dari suatu *website*. Dan sampai sekarang kedua tugas tersebut masih tetap populer digunakan oleh dunia *web* saat ini. Kemudian, dari situ banyak orang di milis mendiskusikan *script* buatan Rasmus Lerdorf, hingga akhirnya rasmus mulai membuat sebuah *tool/script*, bernama *Personal Home Page (PHP)* (Dwiartara, 2012).

H. *MySql*

MySql merupakan salah satu DBMS (*Database Management System*) yang sangat populer di dalam pengembangan sistem. Situs ternama seperti *Facebook*, *Google*, dan *Adobe* juga menggunakan *MySql*. *MySql* memiliki dua lisensi, *open source* di bawah GPL (*GNU General Public License*) dan komersial di bawah *MySQLAB*. *MySql* umumnya menjadi satu paket dalam pembelian *hosting server*. Ketika kita akan menggunakan *MySql* di *server hosting*, maka *tool* yang digunakan adalah *PhpMyAdmin* (Mufti, 2015).

I. *Java Script Object Notation (JSON)*

JavaScript Object Notation atau (*JSON*) sendiri adalah format pertukaran data yang ringan, mudah dibaca dan ditulis oleh manusia, serta mudah diterjemahkan dan dibuat (*generate*) oleh komputer. Format ini dibuat berdasarkan bagian dari bahasa pemrograman *JavaScript*, standar *ECMA-262* edisi ke-3 Desember 1999. *JSON* merupakan format teks yang tidak bergantung pada bahasa pemrograman apapun karena menggunakan bahasa yang umum digunakan oleh programmer keluarga *C* termasuk *C*, *C++*, *C#*, *Java*, *JavaScript*, *Perl*, *Python* dan lain – lain. Oleh karena sifat – sifat tersebut, menjadikan *JSON* ideal sebagai bahasa pertukaran data.

JSON terbuat dari dua struktur, yaitu kumpulan pasangan nama / nilai dan daftar nilai terurutkan (*an ordered list of values*). Struktur – struktur data tersebut dikenal sebagai struktur data *universal*. Pada

dasarnya, semua bahasa pemrograman modern mendukung struktur data ini karena format data mudah dipertukaran dengan bahasa – bahasa pemrograman yang juga berdasarkan pada struktur data ini (Kasman, 2015).

JSON menggunakan bentuk sebagai berikut:

1. *Object*

Object adalah sepasang nama / nilai yang tidak terurutkan. *Object* dimulai dengan { (kurung kurawal buka) dan di akhiri dengan } (kurung kurawal tutup). Setiap nama di ikuti dengan : (titik dua) dan setiap pasangan nama / nilai dipisahkan oleh koma (,).

2. *Array*

Array adalah kumpulan nilai yang terurutkan. *Array* dimulai dengan [(kurung kotak buka) dan diakhiri dengan] (kurung kotak tutup). Setiap nilai dipisahkan oleh tanda koma (,).

3. *Value*

Value atau nilai dapat berupa sebuah string dalam tanda kutip ganda, atau angka, atau *true false* atau *null*, atau sebuah objek atau sebuah larik. Struktur – struktur tersebut dapat disusun bertingkat.

4. *String*

String adalah kumpulan dari nol atau lebih karakter *unicode*, yang dibungkus dengan tanda kutip ganda. Didalam string dapat

digunakan *backslash escapes* “ \ ” untuk membentuk karakter khusus. Sebuah karakter mewakili karakter tunggal pada *string*. *String* sangat mirip dengan *string C* atau *Java*.

5. *Number*

Number atau Angka sangat mirip dengan angka di *C* atau *Java*, kecuali format *oktal* dan *heksadesimal* tidak digunakan.

J. Hasil Penelitian Sejenis

Berikut ini adalah beberapa hasil penelitian sejenis tentang sistem informasi geografis:

- a. Zulfakar (2013) melakukan penelitian tentang Sistem Informasi Geografis pesantren wilayah DIYogyakarta pada *smartphone* berbasis *mobile android*. Tujuan dari penelitian ini adalah merancang dan membangun sistem informasi geografis pesantren yang berada di wilayah Daerah Istimewa Yogyakarta diatas *platform android* pada *smartphone* yang baik dan mudah untuk digunakan dalam menggali informasi letak pesantren dan informasi semua tentang pesantren. Namun aplikasi ini tidak menampilkan jarak untuk menempuh ke lokasi tujuan.
- b. Kusuma (2013) melakukan penelitian tentang Sistem Informasi Geografis Pariwisata Pulau Lombok Berbasis *Android*. Tujuan penelitian ini adalah membangun dan mengimplementasikan sistem informasi geografis pariwisata pulau Lombok berbasis *android* untuk

mengetahui lokasi wisata, jarak wisatawan dari tempat wisata, dan fasilitas yang disediakan di lokasi wisata. Namun aplikasi ini belum menggunakan fitur *search* sehingga pengguna sulit dan rumit untuk mencari lokasi yang ingin dituju.

- c. Palabiran, Cahyadi, dan Arifin (2015) melakukan penelitian tentang Sistem Informasi Geografis Kuliner, Seni dan Budaya Kota Balikpapan Berbasis *Android*. Tujuan penelitian merancang dan membangun sistem informasi geografis kuliner, seni dan budaya di kota Balikpapan untuk mempermudah wisatawan menemukan wisata kuliner dan informasi seni budaya yang ada di Balikpapan. Namun aplikasi ini pada peta tidak menampilkan gambar atau foto tempat kuliner.
- d. Hussein, Eibrahim, dan Asem (2011) melakukan penelitian tentang pemetaan lokasi Fakultas dan fasilitas di Universitas Mansoura dengan Sistem Informasi Geografis berbasis *Mobile*. Aplikasi ini dibuat dengan tujuan mempermudah orang yang pertama kali datang di Universitas Mansoura untuk menemukan lokasi Fakultas dan fasilitas yang ingin dicari
- e. Piarsa, Hadi, dan Wirdiani (2015) melakukan penelitian tentang Sistem Informasi Geografi untuk Pemetaan Jalan Pedesaan menggunakan teknologi *mobile*. Aplikasi ini memberikan informasi tentang nama jalan, panjang jalan, dan kondisi jalan. Sistem ini juga memanfaatkan GPS untuk mencari lokasi jalannya.

BAB III

TUJUAN DAN MANFAAT

A. Tujuan

Tujuan penelitian ini merancang dan membangun aplikasi yang dapat menampilkan letak pondok pesantren di Kabupaten Purbalingga dan memberikan informasi jarak, lokasi serta profil pesantren.

B. Manfaat

1. Memudahkan masyarakat dalam mengakses lokasi pondok pesantren.
2. Memberikan informasi tentang pondok pesantren agar menjadikan sebuah pertimbangan untuk menentukan pilihan pesantren.

BAB IV METODE PENELITIAN

A. Jenis Penelitian

Penelitian ini merupakan penelitian pengembangan, yaitu mengembangkan Sistem Informasi Geografis Pemetaan Lokasi Pondok Pesantren Berbasis *Android* di Kabupaten Purbalingga yang digunakan masyarakat untuk menemukan lokasi pesantren dengan mudah.

B. Waktu dan Tempat Penelitian

Waktu penelitian dimulai pada bulan April sampai November 2016 dan tempat penelitian dilakukan di Laboratorium Rekayasa Perangkat Lunak, Teknik Informatika, Universitas Muhammadiyah Purwokerto.

C. Pengumpulan Data

1. Dokumentasi

Pengumpulan data dalam penelitian ini dilakukan dengan cara dokumentasi yaitu data yang diperoleh dalam pembuatan sistem ini berasal dari dua sumber, sumber yang pertama adalah dokumen – dokumen tentang pondok pesantren di Kabupaten Purbalingga pada Kementerian Agama Purbalingga. Sumber yang kedua adalah data dari *Google Maps* yang berfungsi untuk mendapatkan lokasi *longitude* dan *latitude* dari lokasi pondok pesantren. Data bisa diperoleh dengan memanfaatkan fitur yang ada di *Google Maps*. Untuk mengetahui koordinat lokasi pondok

pesantren dilakukan dengan cara mencari titik lokasi pesantren pada *Google Maps*, kemudian pada titik lokasi klik kanan pilih *what here?/ada apa disini?*, Kemudian akan muncul angka koordinat lokasi tersebut. Selanjutnya koordinat tersebut yang berupa *latitude* dan *longitude* disimpan dan akan digunakan untuk menandai/ marker dari masing-masing lokasi pesantren.

2. Data *Global Positioning System* (GPS)

Data posisi GPS dapat digunakan sebagai data dasar koordinat bumi. Data ini berfungsi sebagai pelengkap dari data pengindraan jauh, apabila ada tempat lokasi pondok pesantren yang tidak bisa diketahui *longitude* dan *latitude* dari pengindraan jauh, maka bisa dilakukan pengambilan data dari data GPS. Data dari GPS bisa diketahui dengan kita membawa GPS ke lokasi tersebut, secara otomatis GPS akan menunjukkan lokasi *longitude* dan *latitude*.

D. Alat

Alat yang digunakan dalam penelitian ini terdiri dari:

1. Perlengkapan *Hardware*

Hardware merupakan semua peralatan fisik komputer yang terdiri dari unit *input*, proses, *output*. Spesifikasi yang digunakan dalam pembuatan sistem ini adalah:

a. *Processor Intel Core i3(1,80 GHz)*

b. RAM 2GB

2. Perlengkapan Software

Software yang digunakan dalam pembuatan sistem ini adalah:

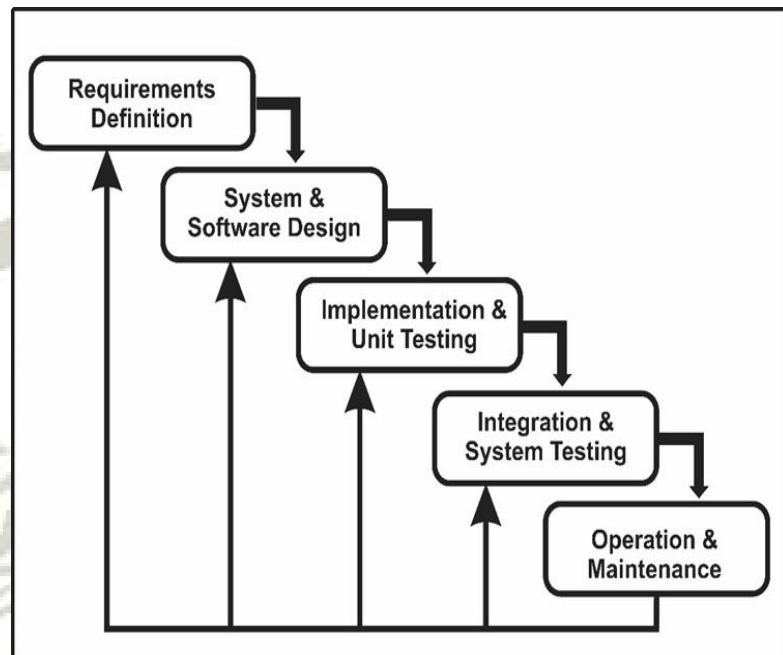
- a. Sistem operasi *Windows 7 Ultimate 32-bit*
- b. *Android Studio 1.4*
- c. *Android Software Development Kit (SDK)*
- d. *Java Development Kit (JDK)*
- e. *XAMPP 3.2.1*
- f. *SublimeText 2.0*
- g. *SQL Manager 2010 for MySql*

3. Satu unit *Smartphone Android*

E. Pengembangan Sistem

Model pengembangan sistem yang dilakukan dalam penelitian ini menggunakan model pengembangan *Waterfall Model*. *Waterfall Model* adalah sebuah contoh dari proses perencanaan, dimana semua proses kegiatan harus terlebih dulu direncanakan dan dijadwalkan sebelum dikerjakan (Sommerville, 2011).

Berikut merupakan beberapa tahapan perencanaan *waterfall model*:



Gambar 2. Model Pengembangan Sistem *Waterfall* (Sommerville, 2011)

1. *Requirements Definition*

Merupakan tahapan penetapan fitur, kendala, tujuan dan data yang diperlukan sistem melalui konsultasi dengan pengguna sistem. Semua hal tersebut akan ditetapkan secara rinci dan berfungsi sebagai spesifikasi sistem. Dalam penelitian ini pihak pengembanglah yang memposisikan dirinya sebagai pengguna.

Informasinya ini di dapat dengan beberapa cara seperti pengumpulan data pondok pesantren yang ada di Kabupaten Purbalingga. Kegiatan pertama sebelum melakukan pengolahan peta adalah mempersiapkan bahan-bahan yaitu peta administrasi Kabupaten Purbalingga melalui *Google Maps* sebagai layanan peta yang digunakan dalam penelitian ini serta

data pondok pesantren yang didapat dari kegiatan pengumpulan data, atribut-atribut dari masing-masing pondok pesantren serta *Longitude/Latitude* dari pondok pesantren dengan tepat. Setelah terkumpul kemudian melakukan pengolahan dan penyesuaian data ke *map* untuk memasukan lokasi koordinat pondok pesantren.

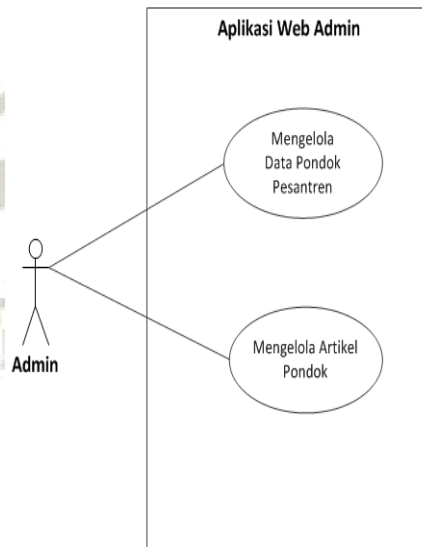
2. *System & Software Design*

Pada tahapan ini, dilakukan proses perancangan sistem secara keseluruhan. Rancangan tersebut terdiri dari rancangan sistem yang akan dibangun berdasar dari data yang diperoleh hingga rancangan spesifikasi *software* dan *hardware* yang diperlukan untuk bisa mengoperasikan sistem yang akan dibangun. Dari data yang telah terkumpul, maka dapat digambarkan rancangan aplikasi yang akan dibangun seperti berikut ini :

a. *Use Case*

1. *Use case diagram* aplikasi web admin

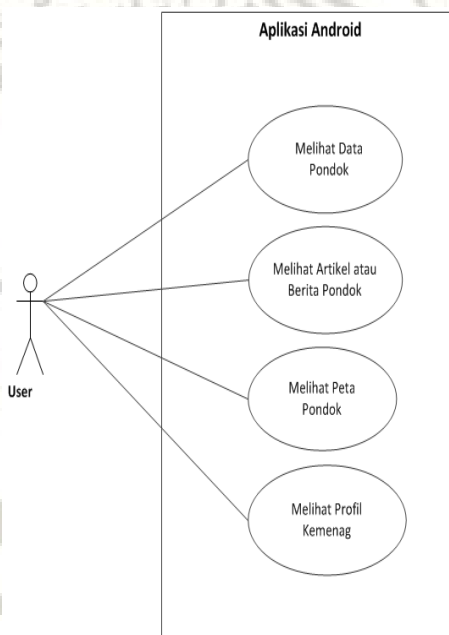
Admin dapat mengelola data pondok, artikel pondok, dan data admin seperti pada Gambar 3.



Gambar 3. *Use case diagram* aplikasi web admin

2. *Use case diagram* aplikasi android

User dapat melihat data pondok, artikel pondok, peta pondok, dan profil Kemenag di Kabupaten Purbalingga seperti pada Gambar 4.



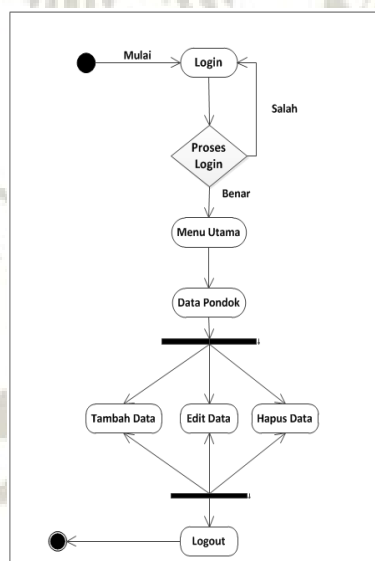
Gambar 4. *Use case diagram* aplikasi android

b. Activity Diagram

Activity Diagram ini menggambarkan berbagai alur aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing proses berawal, dan bagaimana mereka berakhir. Berikut gambaran rancangan aplikasi seperti dibawah ini:

1) *Activity diagram* admin mengelola data pondok

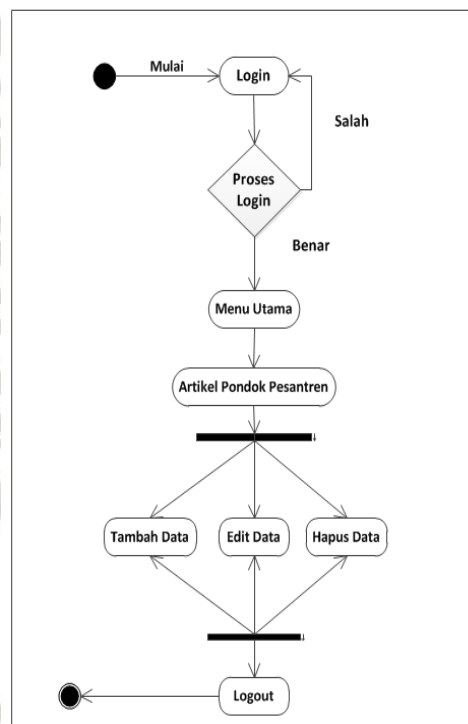
Activity diagram admin mengelola data pondok, admin harus *login* terlebih dahulu dan mengisi *username* dan *password*. Jika gagal *username* dan *password* akan salah dan jika berhasil masuk, selanjutnya dapat melakukan tambah data pondok, rubah data pondok, hapus data pondok dan *logout* untuk keluar dari program seperti pada Gambar 5 dibawah ini.



Gambar 5. *Activity diagram* mengelola data pondok

2) Activity diagram admin mengelola artikel

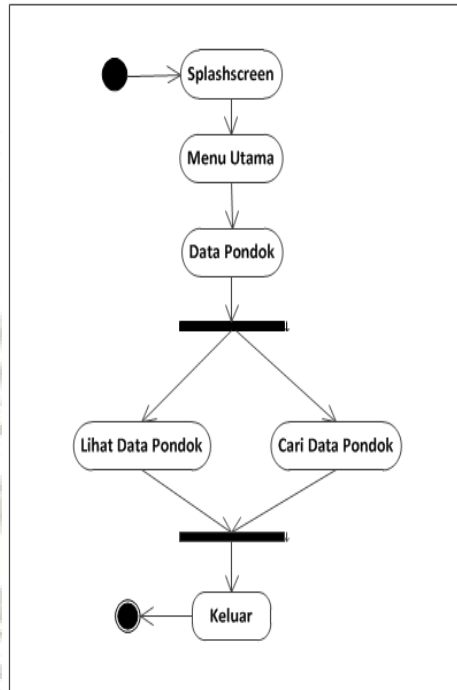
Activity diagram admin mengelola artikel, admin harus *login* terlebih dahulu dan mengisi *username* dan *password*. Jika gagal *username* dan *password* akan salah dan jika berhasil masuk, selanjutnya dapat melakukan tambah artikel, rubah artikel, hapus artikel dan *logout* untuk keluar dari program seperti pada Gambar 6 dibawah ini.



Gambar 6. Activity diagram mengelola artikel

3) *Activity diagram* user melihat data pondok

Activity diagram melihat data pondok, user dapat melihat proses detail dan cari data pondok pada aplikasi seperti Gambar 7.



Gambar 7. *Activity diagram* melihat data pondok

c. Spesifikasi *hardware* dan *software*

Spesifikasi minimal *hardware* dan *software* yang diperlukan untuk menjalankan sistem yang dibangun adalah :

- 1) Aplikasi *Web Admin*
 - a) *Personal Computer (PC) / Laptop RAM 2GB.*
 - b) *Sistem Operasi Windows 7 32bit.*
 - c) *Web Browser.*

2) Aplikasi *Android Client*

- a) *Smartphone* dengan sistem operasi *Android*
- b) *Android* versi 4.0 (*Jelly Bean*).

d. Perancangan *Database*

Perancangan *database* dilakukan dengan mengacu hasil dari pengumpulan data yang telah dianalisis pada tahap sebelumnya. Kemudian dilakukan pembuatan gambaran umum dari relasi antar tabel yang akan dibuat (*diagram* relasi). Berikut adalah tabel rancangan *database* :

1) Rancangan tabel data admin

Tabel data admin berfungsi untuk menyimpan informasi mengenai admin yang sebagai pengelola aplikasi *website*. Didalam *database* diberi nama tabel *m_admin*, berikut rancangan tabel data admin dapat dilihat pada Tabel 1.

Tabel 1. Rancangan tabel data admin

No	Nama Kolom	Tipe Data	Keterangan
1.	admin_id	varchar	Merupakan kolom kode admin pada tabel m_admin
2.	nama_admin	varchar	Merupakan kolom untuk mengisi nama admin.
3.	no_telepon	varchar	Merupakan kolom untuk mengisi nomer telepon admin.
4.	username	varchar	Merupakan kolom untuk mengisi username admin untuk masuk ke sistem.
5.	password	varchar	Merupakan kolom untuk mengisi password admin untuk masuk ke sistem.
6.	level	varchar	Merupakan kolom untuk mengetahui level admin.

2) Rancangan tabel data pondok

Tabel data pondok berfungsi untuk menyimpan informasi tentang pondok seperti nama pondok, alamat, jumlah santri, pengasuh, deskripsi, *latitude*, *longitude*, dan gambar. Didalam *database* diberi nama tabel m_pondok, berikut rancangan tabel data pondok dapat dilihat pada Tabel 2.

Tabel 2. Rancangan tabel data pondok

No.	Nama Kolom	Tipe Data	Keterangan
1.	pondok_id	varchar	Mengisi kolom kode pondok pada tabel m_pondok.
2.	nama_pondok	varchar	Merupakan kolom untuk mengisi nama pondok.
3.	alamat	varchar	Merupakan kolom untuk mengisi alamat pondok.
4.	no_telepon	varchar	Merupakan kolom untuk mengisi nomer telepon.
5.	nama_pengasuh	varchar	Merupakan kolom untuk mengisi nama pengasuh pondok.
6.	santri_putri	varchar	Merupakan kolom untuk mengisi jumlah santri putri yang terdapat di pondok.
7.	santri_putra	varchar	Merupakan kolom untuk mengisi jumlah santri putra yang terdapat di pondok.
8.	jumlah	varchar	Merupakan kolom untuk menjumlah santri putri dan putra.
9.	deskripsi	varchar	Merupakan kolom untuk mengisi deskripsi pondok.
10.	website	varchar	Merupakan kolom untuk mengisi alamat website pondok jika ada.
11.	Blog	varchar	Merupakan kolom untuk mengisi alamat blog pondok jika ada.
12.	latitude	varchar	Merupakan kolom untuk mengisi lokasi latitude pondok.
13.	longitude	varchar	Merupakan kolom untuk mengisi lokasi longitude pondok.
14.	gambar	varchar	Merupakan kolom untuk mengisi gambar pondok.

3) Rancangan tabel artikel

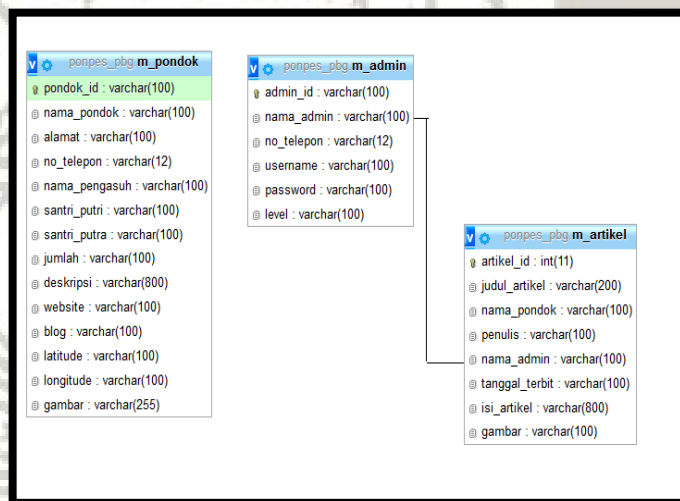
Rancangan tabel artikel berfungsi untuk menyimpan informasi kegiatan pondok seperti judul artikel, nama pondok, isi artikel, penulis, tanggal terbit, dan gambar. Didalam *database* diberi nama *m_artikel*, berikut rancangan tabel artikel dapat dilihat pada Tabel 3 dibawah ini.

Tabel3. Rancangan tabel artikel

No.	Nama Kolom	Type Data	Keterangan
1.	artikel_id	integer	Merupakan kolom kode artikel pada tabel <i>m_artikel</i> .
2.	judul_artikel	varchar	Merupakan kolom untuk mengisi judul artikel pondok.
3.	isi_artikel	varchar	Merupakan kolom untuk mengisi artikel pondok.
4.	nama_pondok	varchar	Merupakan kolom untuk mengisi nama pondok.
5.	penulis	varchar	Merupakan kolom untuk mengisi nama penulis artikel.
6.	nama_admin	varchar	Merupakan kolom untuk menampilkan nama admin yang mengolah artikel.
7.	tanggal_terbit	varchar	Merupakan kolom untuk menampilkan tanggal terbit artikel.

4) Relasi *database*

Relasi tabel digunakan untuk menunjukkan hubungan diantara tabel-tabel yang telah terbentuk berdasarkan rancangan *database*. Relasi dapat dilihat pada Gambar 8.

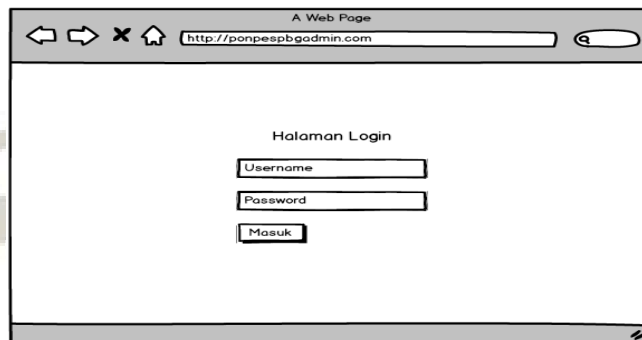


Gambar 8. Relasi *database*

e. Perancangan tampilan aplikasi

1) Rancangan halaman *login*

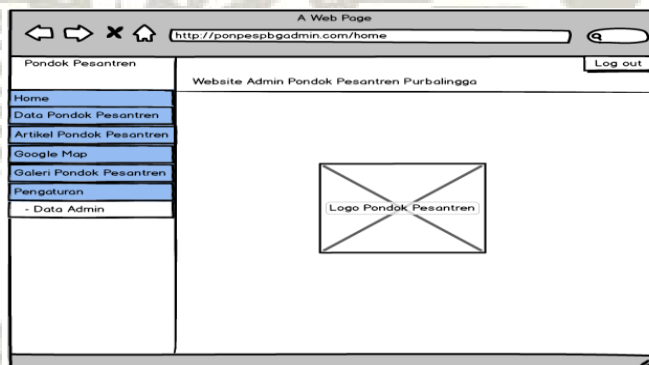
Halaman *login* merupakan halaman yang pertama diakses oleh admin ketika memuat aplikasi *website*, rancangan halaman *login* dapat dilihat pada Gambar 9.



Gambar 9. Rancangan halaman *login* aplikasi *website* admin

2) Rancangan halaman *home*

Halaman *home* adalah halaman yang diakses ketika admin berhasil dalam proses *login*, rancangan halaman *home* dapat dilihat pada Gambar 10 dibawah ini.

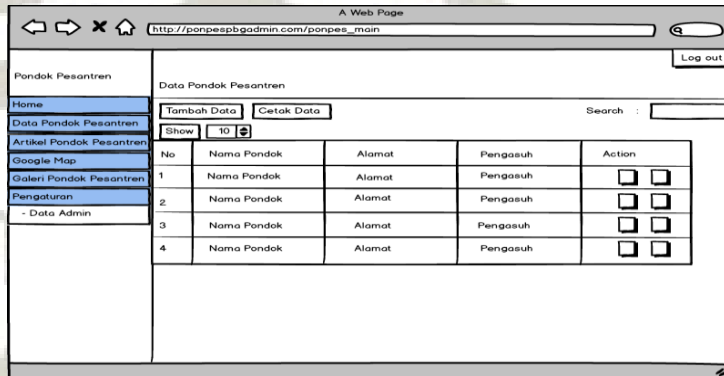


Gambar 10. Rancangan halaman *home* aplikasi *website* admin

3) Rancangan halaman *list* data pondok

Halaman *list* data merupakan halaman yang berfungsi untuk melihat data yang sudah dimasukan oleh admin. Halaman *list* berisi sebuah tabel data

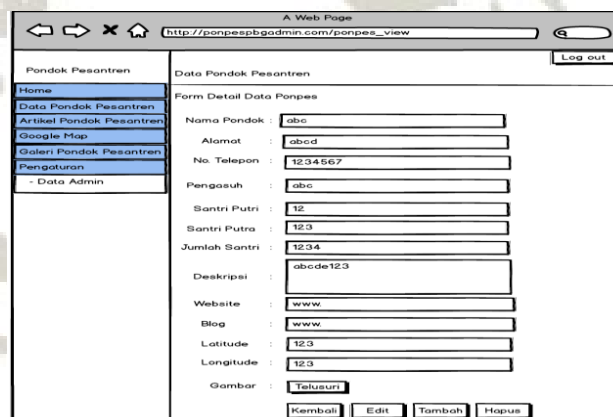
dan kolom aksi untuk setiap baris data, rancangan *list* data dapat dilihat pada Gambar 11 dibawah ini.



Gambar 11. Rancangan halaman *list* data

4) Rancangan halaman detail data pondok

Halaman detail data berfungsi untuk melihat detail data yang pada halaman *list* data hanya sebagian yang ditampilkan. Untuk mengakses halaman detail data, pengguna harus mengklik nama dari *list* data yang terdapat pada halaman *list* data, rancangan halaman detail data dapat dilihat pada Gambar 12 dibawah ini.



Gambar 12. Rancangan halaman detail data pondok

5) Rancangan halaman tambah data

Halaman tambah data berfungsi untuk menambahkan data, rancangan halaman tambah data dapat dilihat pada Gambar 13.

Pondok Pesantren	Data Pondok Pesantren
Home	Form Tambah Data Pondok
Data Pondok Pesantren	Nama Pondok : <input type="text"/>
Artikel Pondok Pesantren	Alamat : <input type="text"/>
Google Map	No. Telepon : <input type="text"/>
Galeri Pondok Pesantren	Pengasuh : <input type="text"/>
Pengaturan	Santri Putri : <input type="text"/>
- Data Admin	Santri Putra : <input type="text"/>
	Deskripsi : <input type="text"/>
	Website : <input type="text"/>
	Blog : <input type="text"/>
	Latitude : <input type="text"/>
	Longitude : <input type="text"/>
	Gambar : <input type="text"/> <input type="button" value="Telusuri"/>
	<input type="button" value="Batal"/> <input type="button" value="Simpan"/>

Gambar 13. Rancangan halaman tambah data pondok

6) Rancangan halaman ubah/edit data pondok

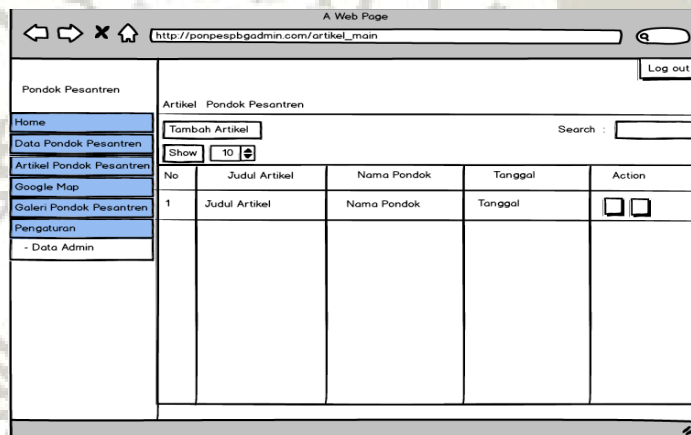
Halaman ubah data berfungsi untuk mengubah data, rancangan halaman ubah data dapat dilihat pada Gambar 14 dibawah ini.

Pondok Pesantren	Data Pondok Pesantren
Home	Form Edit Data Pondok
Data Pondok Pesantren	Nama Pondok : <input type="text" value="abc"/>
Artikel Pondok Pesantren	Alamat : <input type="text" value="abcd"/>
Google Map	No. Telepon : <input type="text" value="1234567"/>
Galeri Pondok	Pengasuh : <input type="text" value="abcd"/>
Pengaturan	Santri Putri : <input type="text" value="12"/>
- Data Admin	Santri Putra : <input type="text" value="123"/>
	Deskripsi : <input type="text" value="abade123"/>
	Website : <input type="text" value="www"/>
	Blog : <input type="text" value="www"/>
	Latitude : <input type="text" value="123"/>
	Longitude : <input type="text" value="123"/>
	Gambar : <input type="text"/> <input type="button" value="Telusuri"/>
	<input type="button" value="Batal"/> <input type="button" value="Simpan"/>

Gambar 14. Rancangan halaman ubah/edit data pondok

7) Rancangan halaman *list* artikel

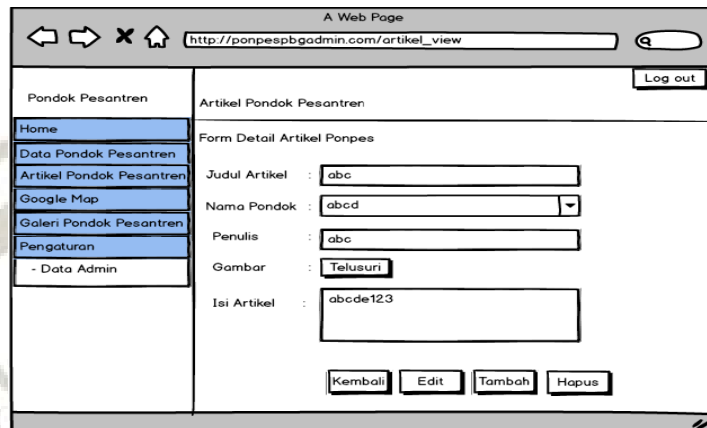
Halaman *list* artikel merupakan halaman yang berfungsi untuk melihat artikel yang sudah dimasukan oleh admin, rancangan *list* artikel dapat dilihat pada Gambar 15 dibawah ini.



Gambar 15. Rancangan halaman *list* artikel

8) Rancangan halaman detail artikel

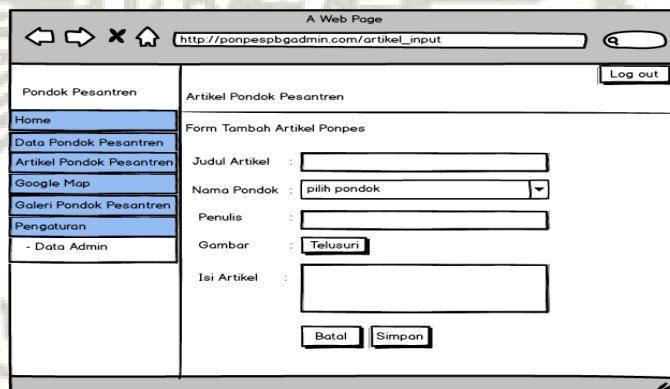
Halaman detail artikel berfungsi untuk melihat detail artikel yang pada halaman *list* artikel hanya sebagian yang ditampilkan. Untuk mengakses halaman detail artikel, pengguna harus mengklik judul dari *list* artikel yang terdapat pada halaman *list* artikel, rancangan halaman detail artikel dapat dilihat pada Gambar 16.



Gambar 16. Rancangan halaman detail artikel

9) Rancangan halaman tambah artikel

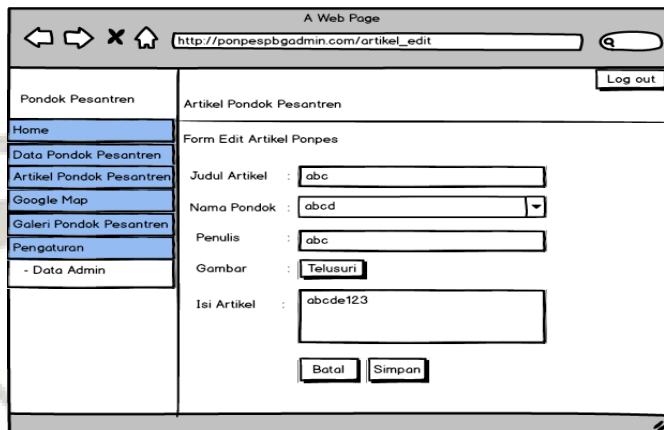
Halaman tambah artikel berfungsi untuk menambahkan artikel, rancangan halaman tambah artikel dapat dilihat pada Gambar 17 dibawah ini.



Gambar 17. Rancangan halaman tambah artikel

10) Rancangan halaman ubah/edit artikel

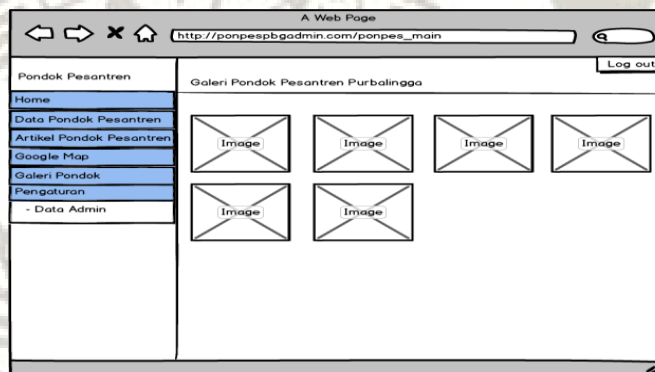
Halaman ubah artikel berfungsi untuk mengubah artikel, rancangan halaman artikel dapat dilihat pada Gambar 18.



Gambar 18. Rancangan halaman ubah/edit artikel

11) Rancangan halaman galeri pondok

Halaman galeri pondok untuk melihat galeri dari gambar-gambar data pondok pesantren, rancangan halaman galeri pondok dapat dilihat pada Gambar 19.



Gambar 19. Rancangan halaman galeri pondok

12) Tampilan *icon* aplikasi di menu *smartphone*

Tampilan *icon* aplikasi berfungsi untuk membedakan antara aplikasi sistem informasi

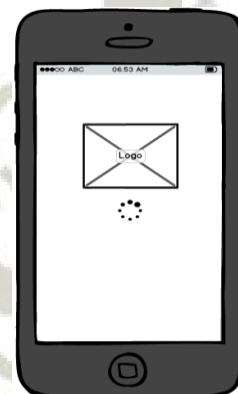
geografis pondok pesantren dengan aplikasi lain yang terdapat pada menu *home* pada *smartphone*, rancangannya dapat dilihat pada Gambar 20 dibawah ini.



Gambar 20. Tampilan *icon* aplikasi di menu *smartphone*

13) Rancangan tampilan *splash screen*

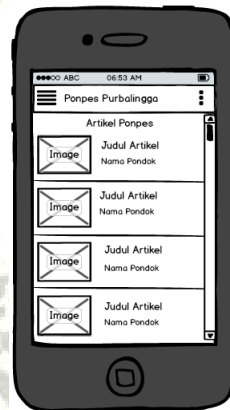
Tampilan *splash screen* merupakan tampilan yang pertama dilihat pada saat pengguna membuka aplikasi, rancangannya dapat dilihat pada Gambar 21 di bawah ini.



Gambar 21. Rancangan tampilan *splash screen*

14) Rancangan tampilan utama aplikasi

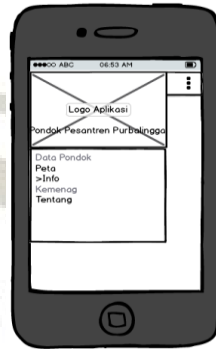
Tampilan utama aplikasi merupakan artikel pondok pesantren yang menjelaskan tentang kegiatan pondok, rancangan dapat dilihat pada Gambar 22 dibawah ini.



Gambar 22. Rancangan tampilan utama aplikasi

15) Rancangan tampilan laci menu aplikasi

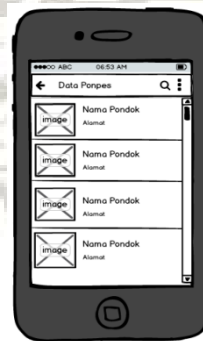
Tampilan laci menu aplikasi merupakan tampilan yang berfungsi sebagai *navigasi* pengguna dalam menggunakan aplikasi. Dengan menggunakan laci menu aplikasi, pengguna bisa memilih menu yang tersedia pada aplikasi, rancangan tampilan dapat dilihat pada Gambar 23.



Gambar 23. Rancangan tampilan laci menu

16) Rancangan tampilan *list* data

Tampilan *list* data berisi daftar data yang sudah ditambahkan oleh admin di aplikasi *website*, rancangan tampilan dapat dilihat pada Gambar 24 dibawah ini.



Gambar 24. Rancangan tampilan *list* data

17) Rancangan tampilan *search*/pencarian data

Fitur *search*/pencarian data berfungsi untuk memudahkan pengguna dalam mencari data yang diinginkan pada halaman *list* data, rancangannya dapat dilihat pada Gambar 25.



Gambar 25. Rancangan tampilan *search*/pencarian data

18) Rancangan tampilan detail data

Halaman detail data berisi tentang informasi detail dari suatu data seperti lokasi di dalam peta, alamat dan informasi lainnya, rancangan dapat dilihat pada Gambar 26 dibawah ini.



Gambar 26. Rancangan tampilan detail data

19) Rancangan tampilan peta pondok

Tampilan peta pondok berisi *marker-marker* lokasi pondok dari data yang ada, rancangan dapat dilihat pada Gambar 27 dibawah ini.



Gambar 27. Rancangan tampilan peta pondok

20) Rancangan tampilan info Kemenag

Tampilan info Kemenag berisi sedikit penjelasan tentang Kemenag Kabupaten Purbalingga, rancangan dapat dilihat pada Gambar 28 dibawah ini.



Gambar 28. Rancangan tampilan info Kemenag

21) Rancangan tampilan tentang aplikasi

Tampilan berisi tentang aplikasi sistem informasi geografi ponpes serta profil pengembang, rancangan dapat dilihat pada Gambar 29 dibawah ini.



Gambar 29. Rancangan tampilan tentang aplikasi

3. *Implementation and Unit Testing*

Pada tahapan ini, desain dari sistem direalisasikan dalam sejumlah *set* program atau unit program. Pengujian dilakukan untuk mengetahui setiap unit dapat berfungsi dengan baik. Dalam pembentukan sistem informasi geografis lokasi ponpes Kabupaten Purbalingga, sistem yang akan dibangun terlebih dahulu adalah sistem *web* untuk admin. Setelah *web* selesai dibangun, selanjutnya aplikasi *android* untuk *client* dibangun.

4. *Integration and System Testing*

Setelah sistem aplikasi dibangun, kemudian dilakukan tahap pengujian aplikasi. Tahap ini dimaksudkan untuk

menguji semua elemen – elemen aplikasi yang telah dibuat, apakah sudah sesuai dengan yang dibutuhkan

5. *Operation and Maintenance*

Tahapan ini biasanya merupakan fase yang paling lama, dimana sistem sudah berjalan dan sudah dapat digunakan. Pemeliharaan mencakup adanya koreksi pada kesalahan yang tidak ditemukan pada tahap awal. Bisa juga dilakukan peningkatan kemampuan sistem dengan menambahkan beberapa kriteria yang baru ditemukan.

BAB V

HASIL DAN PEMBAHASAN

A. Kebutuhan Data

Data yang dibutuhkan untuk bisa melakukan proses *input* kedalam sistem ini terdiri dari data *Spasial* dan data *Non-Spasial*. Data *Spasial* merupakan data yang memuat tentang lokasi suatu objek dalam peta berdasarkan posisi geografis objek tersebut dalam bumi dengan menggunakan sistem koordinat (*Latitude/Garis Lintang, Longitude/Garis Bujur*). Untuk memperoleh data ini bisa menggunakan fitur *Google Maps* atau bisa langsung menggunakan *Global Positioning System (GPS)*.

Sedangkan data *Non-Spasial* merupakan data yang memuat karakteristik atau keterangan dari suatu objek yang terdapat dalam peta yang sama sekali tidak berkaitan dengan posisi geografi objek tertentu. Untuk memperoleh data ini bisa di Kementerian Agama Purbalingga atau langsung datang ke pondok pesantren.

B. Implementasi Sistem

Sistem informasi geografis pondok pesantren ini terdiri dari dua aplikasi, yaitu aplikasi *website* untuk admin dan aplikasi android untuk *user*. Aplikasi *website* digunakan admin untuk mengelola data pondok pesantren, dan selanjutnya data pondok pesantren dapat diakses ke aplikasi *android* yang nantinya dapat digunakan oleh *user*.

Dalam proses pertukaran data antara aplikasi *website* dan aplikasi *android*, data dari *database* aplikasi *website* diubah menggunakan format *java script object notation (JSON)*. Setelah data diubah, aplikasi *android* bisa mengakses *URL* dari *JSON* tersebut.

Berikut ini ditampilkan hasil dari sistem yang dibuat :

1. Aplikasi *website* admin

a) Halaman *login*

Halaman ini merupakan halaman pertama yang diakses apabila ingin menggunakan aplikasi *website* admin ponpes Purbalingga. Login menggunakan *username* dan *password* yang telah terdaftar, tampilan halaman *login* dapat dilihat pada Gambar 30 dibawah ini.

Halaman Login

Username

Password

Masuk

Website
Ponpes Purbalingga

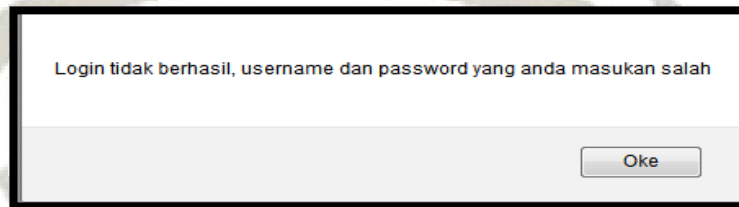
©2016. Teknik Informatika | UMP 2012.

Gambar 30. Halaman *login*

Alur dalam melakukan login aplikasi:

- 1) Admin harus mengisi *username* pada kolom *username* dan mengisi *password* pada kolom *password*
- 2) Admin mengklik pada tombol masuk untuk masuk ke aplikasi

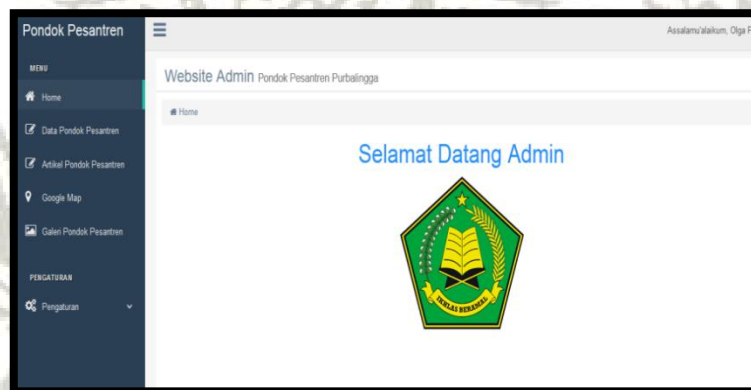
Jika admin melakukan *login* dengan kata sandi yang salah, maka muncul pemberitahuan *username* dan *password* yang anda masukan salah, seperti pada Gambar 31 dibawah ini.



Gambar 31. Tampilan gagal *login*

b) Halaman *home*

Jika admin berhasil *login* maka admin dapat masuk ke aplikasi, halaman pertama kali yang dilihat adalah halaman *home*, tampilan dapat dilihat pada Gambar 32 dibawah ini.

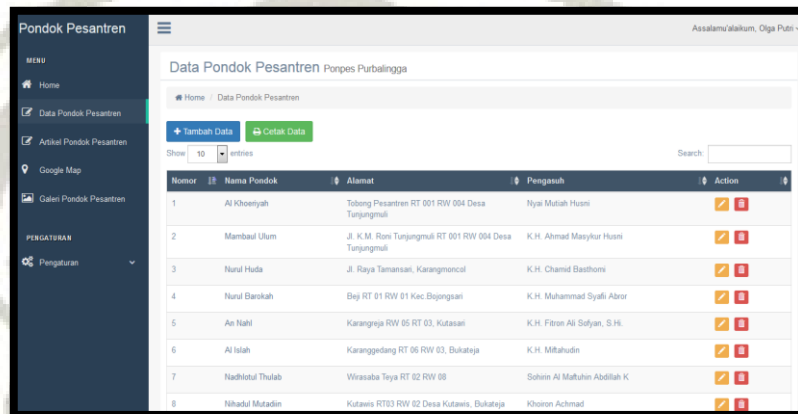


Gambar 32. Tampilan *home*

c) Halaman *list* data pondok

Halaman ini berisi daftar data pondok yang telah dimasukkan oleh admin. Dalam bagian *action* admin bisa memilih untuk mengubah

dan menghapus data, tampilan dapat dilihat pada Gambar 33 dibawah ini.

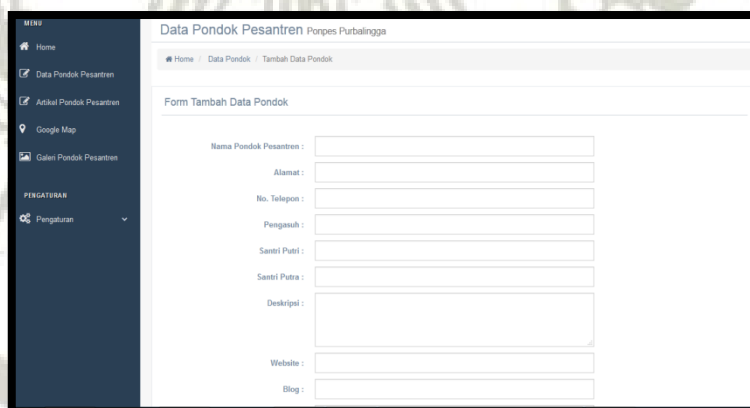


Nomor	Nama Pondok	Alamat	Pengasuh	Action
1	Al Khoiriyah	Tobong Pesantren RT 001 RW 004 Desa Tunggunmuli	Nyia Mutiah Husni	[✓] [✖]
2	Mambal Ulum	Jl. K.M. Rani Tunggunmuli RT 001 RW 004 Desa Tunggunmuli	K.H. Ahmad Masykur Husni	[✓] [✖]
3	Nurul Huda	Jl. Raya Tamansari, Karangmoncol	K.H. Chamid Bashom	[✓] [✖]
4	Nurul Barokah	Beji RT 01 RW 01 Kec Bujangari	K.H. Muhammad Syafi Abror	[✓] [✖]
5	An Nahi	Karangreja RW 05 RT 03, Kutasan	K.H. Firon Ai Sofyan, S.H.	[✓] [✖]
6	Al Islah	Karanggendang RT 06 RW 03, Bukateja	K.H. Mubahidin	[✓] [✖]
7	Nadhrotul Tholab	Wirasaba Teya RT 02 RW 08	Sohim Ai Mafuhin Abdillah K	[✓] [✖]
8	Mubadil Mutadiri	Kutasari RT03 RW 02 Desa Kutasari, Bukateja	Khoren Achmad	[✓] [✖]

Gambar 33. Tampilan *list* data pondok

d) Halaman tambah data

Halaman berisi *form* untuk mengisi/menambah data. Data yang ditambahkan adalah nama pondok, alamat, gambar, *latitude*, *longitude*, dll, tampilan dapat dilihat pada Gambar 34 dibawah ini.



Form Tambah Data Pondok

Nama Pondok Pesantren :

Alamat :

No. Telepon :

Pengasuh :

Santi Putri :

Santi Putra :

Deskripsi :

Website :

Blog :

Gambar 34. Tampilan tambah data

e) Halaman detail data

Halaman ini berisi *form* detail data, tampilan dapat dilihat pada Gambar 35 dibawah ini.

Nama Pondok Pesantren :	Al Khoeriyah
Alamat :	Tobong Pesantren RT 001 RW 004 Desa Tunjungmul
No. Telepon :	081391370026
Pengasah :	Nyai Mutiah Husni
Santri Putri :	63
Santri Putra :	0
Jumlah Santri :	63
Deskripsi :	Al Khoeriyah adalah pesantren yang satu yayasan dengan pesantren Mambaul Ulum. Pondok pesantren Al Khoeriyah merupakan pesantren khusus putri

Gambar 35. Tampilan detail data

f) Halaman ubah/edit data

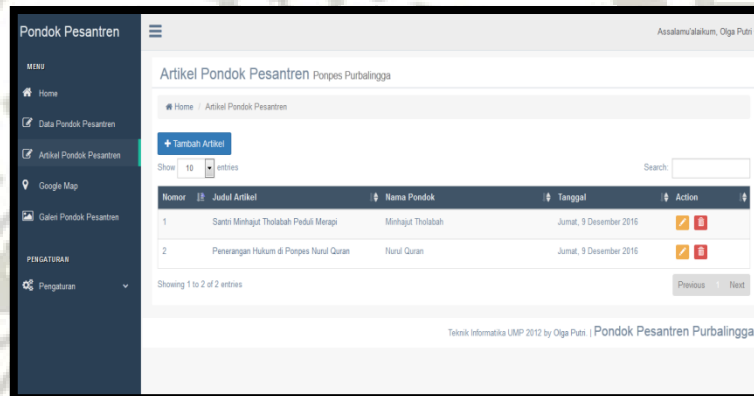
Halaman ini berisi *form* untuk mengubah data, tampilan dapat dilihat pada Gambar 36 dibawah ini.

Nama Pondok Pesantren :	Al Khoeriyah
Alamat :	Tobong Pesantren RT 001 RW 004 Desa Tunjungmul
No. Telepon :	081391370026
Pengasah :	Nyai Mutiah Husni
Santri Putri :	63
Santri Putra :	0
Deskripsi :	Al Khoeriyah adalah pesantren yang satu yayasan dengan pesantren Mambaul Ulum. Pondok pesantren Al Khoeriyah merupakan pesantren khusus putri
Website :	-
Blog :	-

Gambar 36. Tampilan ubah data

g) Halaman *list* artikel

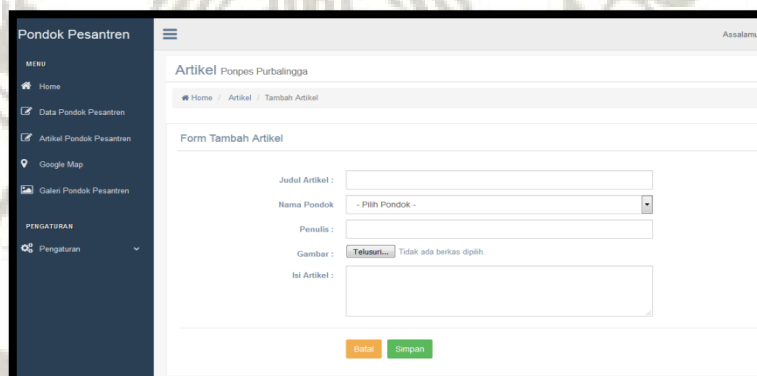
Halaman ini berisi daftar artikel yang telah dimasukkan oleh admin. Dalam *action* bisa memilih untuk ubah dan hapus data, tampilan dapat dilihat pada Gambar 37 dibawah ini.



Gambar 37. Tampilan *list* artikel

h) Halaman tambah artikel

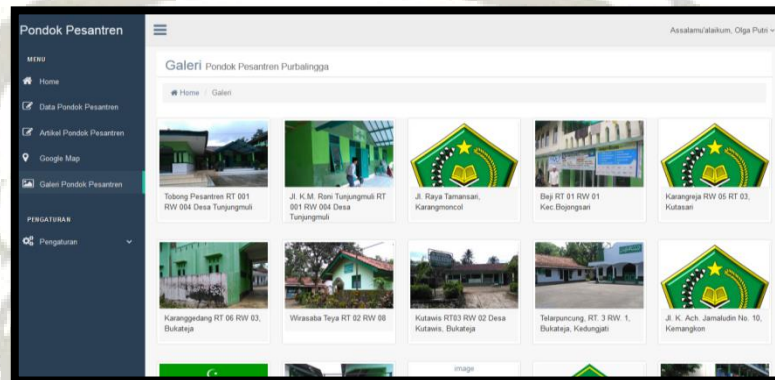
Halaman ini berisi *form* untuk menambah artikel, tampilan dapat dilihat pada Gambar 38 dibawah ini.



Gambar 38. Tampilan tambah artikel

i) Halaman galeri pondok

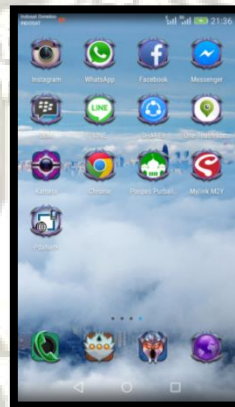
Halaman ini berisi gambar-gambar dari data pondok pesantren, tampilan dapat dilihat pada Gambar 39 dibawah ini.



Gambar 39. Tampilan galeri pondok pesantren

2. Aplikasi *android*

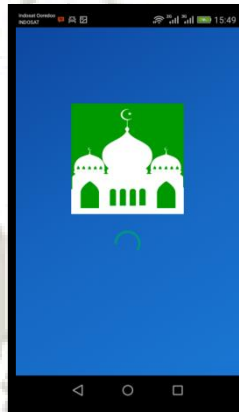
a) Tampilan *icon* aplikasi di menu *smartphone*



Gambar 40. Tampilan *icon* di menu *smartphone*

b) Tampilan *splash screen*

Tampilan *splash screen* merupakan tampilan awal ketika menjalankan aplikasi, tampilan dapat dilihat pada Gambar 41 dibawah ini.



Gambar 41. Tampilan *splash screen*

c) Tampilan menu artikel

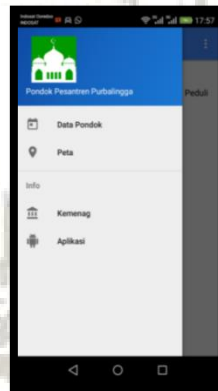
Tampilan menu artikel adalah tampilan menu utama di aplikasi, menu artikel menampilkan informasi seputar kegiatan pondok pesantren, tampilan dapat dilihat pada Gambar 42 dibawah ini.



Gambar 42. Tampilan menu artikel

d) Tampilan laci menu aplikasi

Tampilan laci menu menampilkan menu-menu aplikasi yang terdiri dari data pondok pesantren, peta, info Kemenag, dan tentang aplikasi, tampilan laci menu dapat dilihat pada Gambar 43 dibawah ini.



Gambar 43. Tampilan laci menu aplikasi

e) Tampilan *list* data

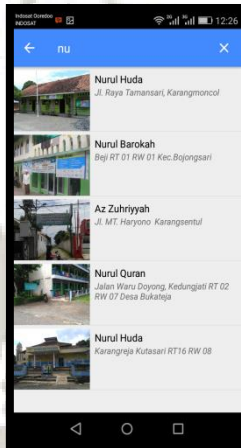
Tampilan *list* data berisi daftar data yang telah dimasukkan melalui aplikasi *website* admin. Pengguna perlu meng-klik data jika ingin melihat detail data, tampilan dapat dilihat pada Gambar 44 dibawah ini.



Gambar 44. Tampilan *list* data

f) Tampilan *search data*

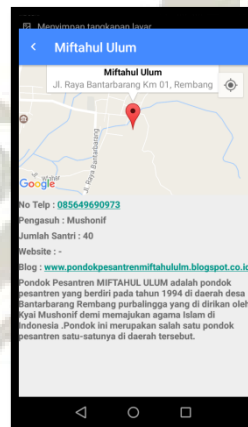
Fitur *search* berfungsi untuk pencarian data yang berdasarkan nama pondok pesantren, tampilan dapat dilihat pada Gambar 45 dibawah ini.



Gambar 45. Tampilan *search data*

g) Tampilan detail data

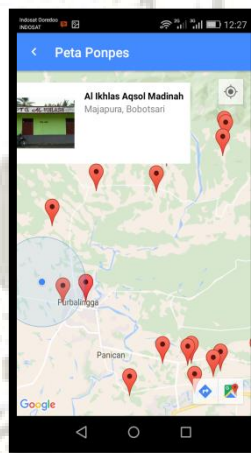
Tampilan ini berisi detail data seperti lokasi, pengasuh, jumlah santri, deskripsi, dll, tampilan dapat dilihat pada Gambar 46 dibawah ini.



Gambar 46. Tampilan detail data

h) Tampilan peta pondok

Tampilan ini berisi sebagai petunjuk lokasi pondok pesantren di Kabupaten Purbalingga, juga terdapat fitur untuk mengetahui lokasi pengguna dan *navigasi*, tampilan dapat dilihat pada Gambar 47 dibawah ini.



Gambar 47. Tampilan peta pondok

i) Tampilan info Kemenag

Tampilan ini berisi informasi tentang Kementerian Agama Purbalingga, tampilan dapat dilihat pada Gambar 47 dibawah ini.



Gambar 48. Tampilan info Kemenag

j) Tampilan tentang aplikasi

Tampilan ini berisi penjelasan aplikasi serta profil pengembang, tampilan dapat dilihat pada Gambar 49 dibawah ini.



Gambar 49. Tampilan tentang aplikasi

C. Pengujian Aplikasi

Pengujian aplikasi ini dilakukan untuk memeriksa jalannya sistem apakah sesuai atau tidak dengan yang di inginkan. Dilakukan dua pengujian, yaitu pengujian aplikasi *android* dan pengujian aplikasi *website*. Berikut ini tampilan pengujian aplikasi :

1) Tabel pengujian aplikasi *android*

Tabel 4. Pengujian aplikasi *android*

No	Data Masukan	Hasil yang Diharapkan	Hasil Pengamatan	Keterangan
1	Buka aplikasi pada menu <i>Homescreen</i>	Menampilkan halaman aplikasi	Tampil halaman utama aplikasi atau halaman artikel	Benar
2	Klik detail artikel pondok	Menampilkan detail artikel	Tampil detail artikel	Benar
3	Buka menu data pondok	Menampilkan data pondok	Tampil data pondok	Benar
4	Klik detail data pondok	Menampilkan detail data pondok	Tampil detail data pondok	Benar
5	Buka menu peta pondok	Menampilkan peta pondok	Tampil peta pondok	Benar
6	Buka menu info Kemenag	Menampilkan info Kemenag	Tampil info Kemenag	Benar
7	Buka menu tentang aplikasi	Menampilkan tentang aplikasi	Tampil tentang aplikasi	Benar

2) Tabel pengujian aplikasi *website*

Tabel 5. Tabel pengujian aplikasi *website*

No	Data Masukan	Hasil yang Diharapkan	Hasil Pengamatan	Keterangan
1	<i>Username</i> dan <i>password</i> admin	Menampilkan halaman admin	Tampil halaman admin	Benar
2	<i>Username</i> dan <i>password</i> tidak sesuai	Muncul pesan kesalahan	Muncul pesan kesalahan gagal login	Benar
3	Buka menu data pondok	Menampilkan daftar data pondok	Tampil daftar data pondok	Benar
4	Klik detail data pondok	Menampilkan detail data pondok	Tampil detail data pondok	Benar
5	Menambahkan data pondok	Menambahkan data pondok	Data pondok bertambah	Benar
6	Mengubah data pondok	Perubahan data pondok	Data pondok berubah	Benar
7	Menghapus data pondok	Menghapus data pondok	Data pondok terhapus	Benar
8	Cetak data pondok	Mencetak data pondok	Data pondok tercetak	Benar
9	Buka menu artikel	Menampilkan daftar artikel	Tampil daftar artikel	Benar
10	Klik detail artikel	Menampilkan detail artikel	Tampil detail artikel	Benar
11	Menambah artikel	Artikel bertambah	Artikel bertambah	Benar
12	Menghapus artikel	Artikel terhapus	Artikel terhapus	Benar
13	<i>Logout</i> aplikasi admin	Keluar aplikasi admin	Admin <i>logout</i> aplikasi	Benar

Validator

Risma Adhedy

BAB VI PENUTUP

A. Kesimpulan

Kesimpulan yang dapat diambil dari pembangunan aplikasi Sistem Informasi Geografis Pemetaan Lokasi Pondok Pesantren di Kabupaten Purbalingga adalah :

1. Telah dibangun sistem informasi geografis pondok pesantren di Kabupaten Purbalingga yang dapat mempermudah masyarakat khususnya pengguna *smartphone android* untuk mengetahui lokasi serta informasi seputar pondok pesantren.
2. Memperkenalkan lebih detail ke masyarakat setiap pondok pesantren yang ada di Kabupaten Purbalingga untuk memberi pertimbangan bagi yang ingin memilih pesantren.

B. Saran

Sistem yang telah dibangun dapat dikembangkan lagi dengan menambahkan beberapa fitur sebagai berikut :

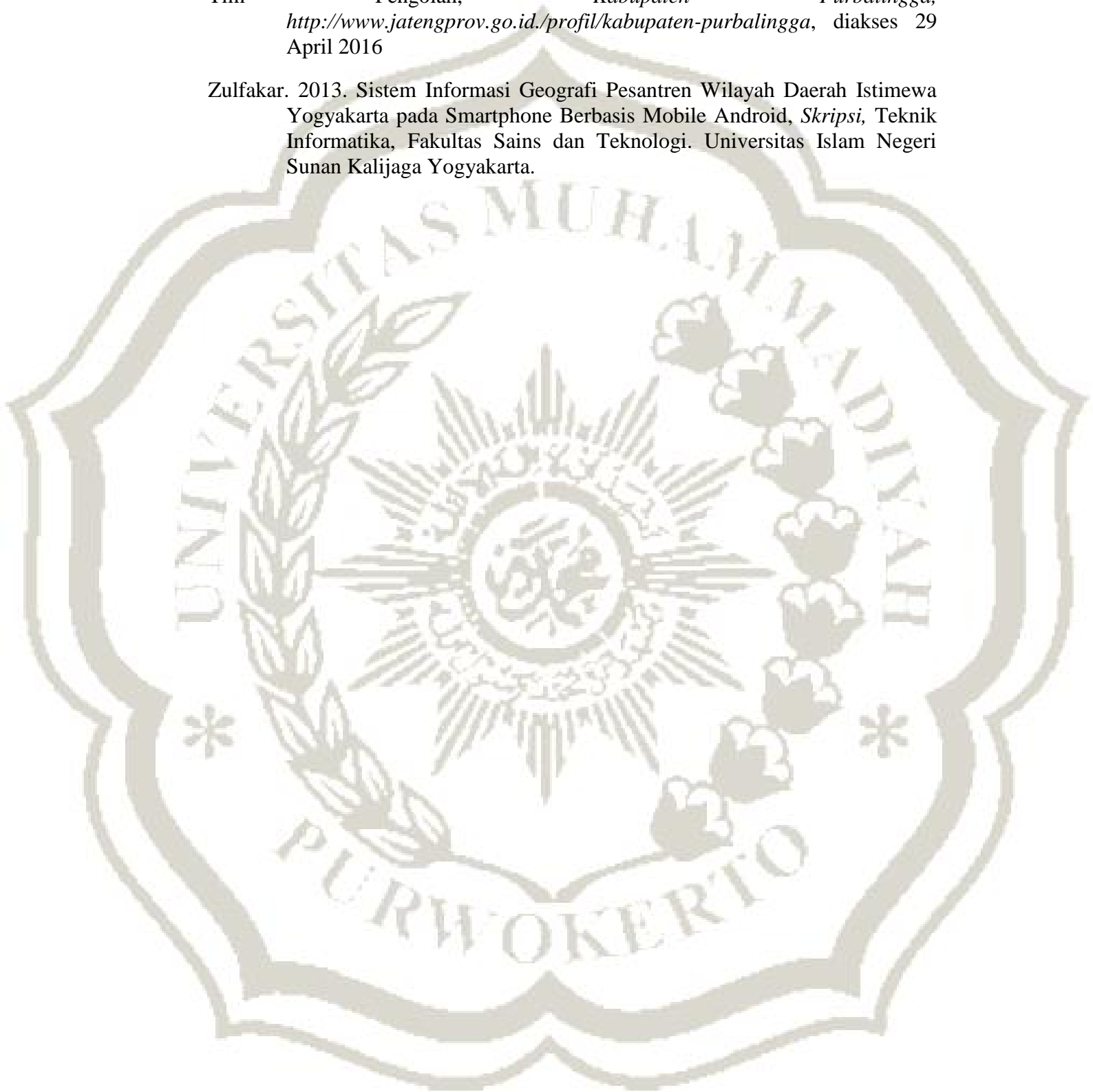
1. Menambahkan *form* pendaftaran untuk calon santri baru yang ingin masuk ke pondok pesantren.
2. Diperlukan fitur *sign up* khusus pengurus pondok pesantren supaya bisa meng-*update* artikel tanpa harus bertumpu pada satu admin.

DAFTAR PUSTAKA

- Agung, G. 2015. *Mengenal Dasar-Dasar Pemrograman Android*. PT Elex Media Komputindo. Jakarta.
- Dwiartara, L. 2012. *Menyelam dan Menaklukan Samuder PHP*. Ilmu Website. Jakarta.
- Hartono, J. 1999. *Sistem Informasi*. Wahana Komputer. Semarang.
- Hussein, A, A. Eibrahhim, E, H. and Asem, A. 2011. *Mobile Geographic Information System*. *International Journal of Computer Science & Information Tecnology (IJCSIT)*. DOI: 10.512. Vol 3. No 6. December 2011. pp 173-181.
- Jogiyanto, H, M. 2005. *Analysis dan Design Sistem Informasi Pendekatan Terstruktur dan Aplikasi Bisnis*. Andi. Yogyakarta.
- Kasman, A, D. 2015. *Trik Kolaborasi Android dengan PHP & MySql*. Lokomedia. Yogyakarta.
- Kusuma, I, J. 2013. "Sistem Informasi Geografis Pariwisata Pulau Lombok Berbasis Android", *Skripsi*, Teknik Informatika, Fakultas Sains dan Teknologi. Universitas Islam Negeri Sunan Kalijaga Yogyakarta.
- Meliono, A, M. 1990. *Kamus Besar Bahasa Indonesia*. Balai Pustaka. Jakarta.
- Mufti, Y. 2015. *Panduan Mudah Pengembangan Google Map Android*. Andi. Yogyakarta.
- Nata, A. 2012. *Kapita Selekta Pendidikan Islam*. RajaGrafindo Persada. Jakarta.
- Palabiran, M, Cahyadi, D. dan Arifin, Z. 2015. Sistem Informasi Geografis Kuliner, Seni dan Budaya Berbasis Android (Studi Kasus di Balikpapan). *Jurnal Informatika Jurusan Ilmu Komputer*. Nomor 1 Volume 10. Februari.
- Piarsa, I, Hadi, E, S. and Wirdiani, K, A. 2015. Rural Road Mapping Goegraphic Information System Using Mobile Android. *International Journal of Computer Science Issues (IJCSI)*. ISSN: 1694-0784. Vol 12. Issue 3. May 2015. pp 95-100.
- Prahasta, E. 2005. *Sistem Informasi Geografis*. CV.Informatika. Bandung.
- Sommerville, I. 2011. *Software Engineering 9th Edition*. Pearson Education. Boston.

Tim Pengolah, Kabupaten Purbalingga,
<http://www.jatengprov.go.id/profil/kabupaten-purbalingga>, diakses 29
April 2016

Zulfakar. 2013. Sistem Informasi Geografi Pesantren Wilayah Daerah Istimewa Yogyakarta pada Smartphone Berbasis Mobile Android, *Skripsi*, Teknik Informatika, Fakultas Sains dan Teknologi. Universitas Islam Negeri Sunan Kalijaga Yogyakarta.



The logo of Universitas Muhammadiyah Purwokerto is a large, light-colored watermark in the background. It features a central sunburst with rays, surrounded by a wreath of leaves and flowers. The text "UNIVERSITAS MUHAMMADIYAH" is arched across the top, and "PURWOKERTO" is written across the bottom. Two small asterisks are positioned on the left and right sides of the central emblem.

LAMPIRAN

Lampiran 1. Kode Program

1. Kode Program *Website Admin*

a. Kode program halaman *login*

```
<?php
// memulai session
session_start();
error_reporting(0);
if (isset($_SESSION['level']) & isset($_SESSION['password']))
{
header('location:pages/home.php');
}
else if (isset($_SESSION['level']) & !isset($_SESSION['password']))
{
header('location:autologout.php');
}
else if (!isset($_SESSION['level']) & !isset($_SESSION['password']))
{

}
?>
<!DOCTYPE html>
<html lang="en">

<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<!-- Meta, title, CSS, favicons, etc. -->
<meta charset="utf-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1">

<title>Login | Ponpes Purbalingga</title>

<!-- Bootstrap core CSS -->

<link href="css/bootstrap.min.css" rel="stylesheet">

<link href="fonts/css/font-awesome.min.css" rel="stylesheet">
<link href="css/animate.min.css" rel="stylesheet">

<!-- Custom styling plus plugins -->
<link href="css/custom.css" rel="stylesheet">
<link href="css/ichack/flat/green.css" rel="stylesheet">
```

```

<script src="js/jquery.min.js"></script>

<!--[if lt IE 9]>
  <script src="../../assets/js/ie8-responsive-file-warning.js"></script>
<![endif]-->

  <!-- HTML5 shim and Respond.js for IE8 support of HTML5
  elements and media queries -->
  <!--[if lt IE 9]>
    <script
    src="https://oss.maxcdn.com/html5shiv/3.7.2/html5shiv.min.js"></scrip
    t>
    <script
    src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>
    <![endif]-->

</head>

<body style="background:#F7F7F7;">

  <div class="">
    <a class="hiddenanchor" id="toregister"></a>
    <a class="hiddenanchor" id="tologin"></a>

    <div id="wrapper">
      <div id="login" class="animate form">
        <section class="login_content">
          <form action="login.php" method="post">
            <h1>Halaman Login</h1>
            <div>
              <input type="text" class="form-control"
placeholder="Username" name="username" required="" />
            </div>
            <div>
              <input type="password" class="form-control"
placeholder="Password" name="pass" required="" />
            </div>
            <div class="col-xs-3">
              <button type="submit" class="btn btn-primary btn-
block btn-flat">Masuk</button>
            </div></center>
            <div class="clearfix"></div>
            <div class="separator">
              <div class="clearfix"></div>
              <br />
            <div>

```

```

                <h1><i style="font-size:
26px;"></i>Website<br>Ponpes Purbalingga</h1>

                <p>©2016. Teknik Informatika | UMP 2012.</p>
            </div>
        </div>
    </form>
    <!-- form -->
</section>
<!-- content -->
</div>

</div>
</div>

</body>
</html>

```

b. Kode program halaman *home*

```

<?php
include "header.php";
?>

<!-- page content -->
<div class="right_col" role="main">
    <div class="row">
        <div class="col-md-12 col-sm-12 col-xs-12">
            <div class="dashboard_graph">

                <div class="row x_title">
                    <div class="col-md-6">
                        <h3>Website Admin <small>Pondok
Pesantren Purbalingga</small></h3>
                    </div>
                </div>
                <ol class="breadcrumb">
                    <li><a href="home.php"><i class="fa fa-
home"></i> Home</a></li>
                </ol>
                <center><h1><font color="#1E90FF">Selamat
Datang Admin</font></h1> </center>
                <center></center>
                <br><br><br><br>
            </div>

```



```

        </div>
    </div>
    <!-- end content -->
    <br />
<?php
include "footer.php";
?>

```

c. Kode program halaman *list data*

```

<?php
include "header.php";
include ('koneksi.php');
?>
    <!-- page content -->
    <link rel="stylesheet"
href="Data_tables/css/dataTables.bootstrap.css">
    <script type="text/javascript"
src="Data_tables/js/jquery.dataTables.min.js"></script>
    <script src="Data_tables/js/dataTables.bootstrap.js"></script>
    <script type="text/javascript"
src="js/imagesshow/tinybox.js"></script>
    <!-- Datatables -->
    <script>
    $(document).ready(function(){
    $('#myTable').dataTable({
    "order": [[ 0, "asc" ]]
    });
    });
    </script>
<div class="right_col" role="main">
    <div class="row">
        <div class="col-md-12 col-sm-12 col-xs-12">
            <div class="dashboard_graph">

                <div class="row x_title">
                    <div class="col-md-6">
                        <h3>Data Pondok Pesantren <small>Ponpes
Purbalingga</small></h3>
                    </div>
                </div>
                <div class="row x_title">
                    <div class="col-md-6">
                        <ol class="breadcrumb">
                            <li><a href="home.php"><i class="fa fa-
home"></i> Home</a></li>

```

```

        <li class="active">Data Pondok Pesantren</a></li>
    </ol>
    <a class='btn btn-primary'
href="ponpes_input.php"><i class='fa fa-plus'></i> Tambah Data</a>
    <a class='btn btn-success' href="cetak.php"><i
class='fa fa-print'></i> Cetak Data</a>

    <table id="myTable" class="table table-striped
responsive-utilities jambo_table">
    <thead>
    <tr class="headings">
    <th width="3%">Nomor </th>
    <th width="15%">Nama Pondok </th>
    <th width="22%">Alamat </th>
    <th width="10%" class="no-link
last"><span class="no-link">Action</span>
    </th>
    </tr>
    </thead>
    <tbody>
    <?php
    $query = mysql_query("select * from
m_pondok");
    $no = 1;
    while ($data = mysql_fetch_array($query))
    {?>
        <tr>
        <td><?php echo $no; ?></td>
        <td><a
href="ponpes_view.php?id=<?php echo $data['pondok_id']; ?>"><?php
echo $data['nama_pondok']; ?></a></td>
        <td><?php echo $data['alamat']; ?></td>
        <td>
            <a href="ponpes_edit.php?id=<?php
echo $data['pondok_id']; ?>" class="btn btn-warning btn-xs" ><span
class="glyphicon glyphicon-pencil" aria-hidden="true"></span></a>
            <a href="ponpes_delete.php?id=<?php
echo $data['pondok_id'] ?>" onclick="return confirm('Anda yakin akan
menghapus data?')" class="btn btn-danger btn-xs" ><span
class="glyphicon glyphicon-trash" aria-hidden="true"></span></a>
        </td></center>
        </tr>
    }?>
    </tbody>
    </table>

```

```

<?php
                $no++;
                }
                ?>
            </tbody>
        </table>
    </div>
</div>

<!-- end content -->
<br />
<?php
include "footer.php";
?>

```

d. Kode program halaman tambah data

```

<?php
include "header.php";
include('koneksi.php');
?>
<div class="right_col" role="main">
    <div class="row">
        <div class="col-md-12 col-sm-12 col-xs-12">
            <div class="dashboard_graph">

                <div class="row x_title">
                    <div class="col-md-6">
                        <h3>Data Pondok Pesantren <small>Ponpes
Purbalingga</small></h3>
                    </div>
                </div>
                <ol class="breadcrumb">
                    <li><a href="home.php"><i class="fa fa-
home"></i> Home</a></li>
                    <li><a href="ponpes_main.php"> Data
Pondok</a></li>
                    <li class="active">Tambah Data Pondok</a></li>
                </ol>
            </div>
        </div>
    </div>
    <div class="row">
        <div class="col-md-12 col-sm-12 col-xs-12">
            <div class="x_panel">

```

```

<div class="x_title">
    <h2>Form Tambah Data Pondok </h2>
    <ul class="nav navbar-right panel_toolbox">

    </ul>
    <div class="clearfix"></div>
</div>
<div class="x_content">
    <br />
    <form name="form_tambahpondok"
enctype="multipart/form-data" onsubmit="return validate(this)"
onClick="highlight(event)" onKeyUp="highlight(event)" data-parsley-
validate class="form-horizontal form-label-left"
action="ponpes_insert.php" method="post">
        <div class="form-group">
            <label class="control-label col-md-3 col-
sm-3 col-xs-12" for="nama">Nama Pondok Pesantren <span
class="required">: </span>
            </label>
            <div class="col-md-6 col-sm-6 col-xs-12">
                <input type="text" name="txt_nama"
required="required" class="form-control col-md-7 col-xs-12">
            </div>
        </div>

        <div class="form-group">
            <label class="control-label col-md-3 col-
sm-3 col-xs-12" for="alamat">Alamat <span class="required">:
</span>
            </label>
            <div class="col-md-6 col-sm-6 col-xs-12">
                <input type="text" name="txt_alamat"
required="required" class="form-control col-md-7 col-xs-12">
            </div>
        </div>

        <div class="form-group">
            <label class="control-label col-md-3 col-
sm-3 col-xs-12" for="no_telepon">No. Telepon <span
class="required">: </span>
            </label>
            <div class="col-md-6 col-sm-6 col-xs-12">
                <input type="text" name="txt_notlp"
required="required" class="form-control col-md-7 col-xs-12">
            </div>
        </div>
    </div>
    <div class="form-group">

```

```

        <label class="control-label col-md-3 col-sm-
3 col-xs-12" for="nama_pengasuh">Pengasuh <span class="required">:
</span>
        </label>
        <div class="col-md-6 col-sm-6 col-xs-12">
            <input type="text" name="txt_pengasuh"
required="required" class="form-control col-md-7 col-xs-12">
        </div>
    </div>
    <div class="form-group">
        <label class="control-label col-md-3 col-sm-
3 col-xs-12" for="santriputri">Santri Putri <span class="required">:
</span>
        </label>
        <div class="col-md-6 col-sm-6 col-xs-12">
            <input type="text" name="txt_santriputri"
required="required" class="form-control col-md-7 col-xs-12">
        </div>
    </div>
    <div class="form-group">
        <label class="control-label col-md-3 col-sm-
3 col-xs-12" for="santriputra">Santri Putra <span class="required">:
</span>
        </label>
        <div class="col-md-6 col-sm-6 col-xs-12">
            <input type="text" name="txt_santriputra"
required="required" class="form-control col-md-7 col-xs-12">
        </div>
    </div>
    <div class="form-group">
        <label class="control-label col-md-3 col-sm-
3 col-xs-12" for="deskripsi">Deskripsi <span class="required">:
</span>
        </label>
        <div class="col-md-6 col-sm-6 col-xs-12">
            <textarea name="txt_deskripsi"
required="required" class="form-control"></textarea>
        </div>
    </div>
    <div class="form-group">
        <label class="control-label col-md-3 col-sm-
3 col-xs-12">Website <span class="required">: </span>
        </label>

```

```

<input name="txt_website" ="" class="form-control col-md-7 col-xs-12" required="required" type="text">
    </div>
</div>
<div class="form-group">
    <label class="control-label col-md-3 col-sm-3 col-xs-12">Blog <span class="required">: </span> </label>
    <div class="col-md-6 col-sm-6 col-xs-12">
        <input name="txt_blog" ="" class="form-control col-md-7 col-xs-12" required="required" type="text">
    </div>
</div>
<div class="form-group">
    <label class="control-label col-md-3 col-sm-3 col-xs-12">Latitude <span class="required">: </span> </label>
    <div class="col-md-6 col-sm-6 col-xs-12">
        <input name="txt_latitude" ="" placeholder="Gunakan titik sebagai pemisah." class="form-control col-md-7 col-xs-12" required="required" type="text">
    </div>
</div>
<div class="form-group">
    <label class="control-label col-md-3 col-sm-3 col-xs-12">Longitude <span class="required">: </span> </label>
    <div class="col-md-6 col-sm-6 col-xs-12">
        <input name="txt_longitude" ="" placeholder="Gunakan titik sebagai pemisah." class="form-control col-md-7 col-xs-12" required="required" type="text">
    </div>
</div>
<div class="form-group">
    <label class="control-label col-md-3 col-sm-3 col-xs-12">Gambar <span class="required">: </span> </label>
    <div class="col-md-6 col-sm-6 col-xs-12">
        <input type="file" name="file" id="file">
    </div>
</div>

<div class="ln_solid"></div>

```



```

<div class="form-group">
  <div class="col-md-6 col-sm-6 col-xs-12
  col-md-offset-3">
    <a href="ponpes_main.php" class="btn
    btn-warning">Batal </a>
    <input class="btn btn-success"
    name="simpan" type="submit" value="Simpan">
  </div>
</div>
</form>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
</div>

<?php
include "footer.php";
?>

```

e. Kode program ubah data

```

<div class="dashboard_graph">
    <div class="row x_title">
        <div class="col-md-6">
            <h3>Data Pondok Pesantren <small>Ponpes
Purbalingga</small></h3>
        </div>
    </div>
    <ol class="breadcrumb">
        <li><a href="home.php"><i class="fa fa-
home"></i> Home</a></li>
        <li><a href="ponpes_main.php"> Data
Pondok</a></li>
        <li class="active">Edit Data Pondok</a></li>
    </ol>
</div>
<div class="row">
<div class="col-md-12 col-sm-12 col-xs-12">
<div class="x_panel">
<div class="x_title">
<h2>Form Edit Data Pondok </h2>
<ul class="nav navbar-right panel_toolbox">

</ul>
<div class="clearfix"></div>
</div>
<div class="x_content">
<br />
<form name="form_editpondok"
enctype="multipart/form-data" onSubmit="return validate(this)"
onClick="highlight(event)" onKeyUp="highlight(event)" data-parsley-
validate class="form-horizontal form-label-left"
action="ponpes_update.php" method="post">
    <input type="hidden" name="id" value="<?php
echo $data['pondok_id']; ?>" />
    <div class="form-group">
        <label class="control-label col-md-3 col-sm-
3 col-xs-12" for="nama">Nama Pondok Pesantren <span
class="required">: </span>
        </label>
        <div class="col-md-6 col-sm-6 col-xs-12">
            <input type="text" name="txt_nama"
required="required" class="form-control col-md-7 col-xs-12"
value="<?php echo $data['nama_pondok']; ?>">
        </div>
    </div>
</div>

```

```

<div class="form-group">
  <label class="control-label col-md-3 col-
sm-3 col-xs-12" for="alamat">Alamat <span class="required">:
</span>
  </label>
  <div class="col-md-6 col-sm-6 col-xs-12">
    <input type="text" name="txt_alamat"
required="required" class="form-control col-md-7 col-xs-12"
value="<?php echo $data['alamat']; ?>">
  </div>
</div>
<div class="form-group">
  <label class="control-label col-md-3 col-
sm-3 col-xs-12" for="no_telepon">No. Telepon <span
class="required">: </span>
  </label>
  <div class="col-md-6 col-sm-6 col-xs-12">
    <input type="text" name="txt_notlp"
required="required" class="form-control col-md-7 col-xs-12"
value="<?php echo $data['no_telepon']; ?>">
  </div>
</div>
<div class="form-group">
  <label class="control-label col-md-3 col-
sm-3 col-xs-12" for="nama_pengasuh">Pengasuh <span
class="required">: </span>
  </label>
  <div class="col-md-6 col-sm-6 col-xs-12">
    <input type="text" name="txt_pengasuh"
required="required" class="form-control col-md-7 col-xs-12"
value="<?php echo $data['nama_pengasuh']; ?>">
  </div>
</div>
<div class="form-group">
  <label class="control-label col-md-3 col-
sm-3 col-xs-12" for="santriputri">Santri Putri <span
class="required">: </span>
  </label>
  <div class="col-md-6 col-sm-6 col-xs-12">
    <input type="text"
name="txt_santriputri" required="required" class="form-control col-
md-7 col-xs-12" value="<?php echo $data['santri_putri']; ?>">
  </div>
</div>
<div class="form-group">

```

```

<label class="control-label col-md-3 col-sm-3 col-xs-12"
for="santriputra">Santri Putra <span class="required">: </span>
</label>
<div class="col-md-6 col-sm-6 col-xs-12">
<input type="text"
name="txt_santriputra" required="required" class="form-control col-
md-7 col-xs-12" value="<?php echo $data['santri_putra']; ?>">
</div>
</div>

<div class="form-group">
<label class="control-label col-md-3 col-
sm-3 col-xs-12" for="deskripsi">Deskripsi <span class="required">:
</span>
</label>
<div class="col-md-6 col-sm-6 col-xs-12">
<textarea name="txt_deskripsi"
required="required" class="form-control" ><?php echo
$data['deskripsi']; ?></textarea>
</div>
</div>
<div class="form-group">
<label class="control-label col-md-3 col-
sm-3 col-xs-12">Website <span class="required">: </span>
</label>
<div class="col-md-6 col-sm-6 col-xs-12">
<input name="txt_website" =""
class="form-control col-md-7 col-xs-12" required="required"
type="text" value="<?php echo $data['website']; ?>">
</div>
</div>
<div class="form-group">
<label class="control-label col-md-3 col-
sm-3 col-xs-12">Blog<span class="required">: </span>
</label>
<div class="col-md-6 col-sm-6 col-xs-12">
<input name="txt_blog" ="" class="form-
control col-md-7 col-xs-12" required="required" type="text"
value="<?php echo $data['blog']; ?>">
</div>
</div>

<div class="form-group">
<label class="control-label col-md-3 col-
sm-3 col-xs-12">Posisi Latitude <span class="required">: </span>

```

```

        </label>
        <div class="col-md-6 col-sm-6 col-xs-12">
            <input name="txt_latitude" =""
class="form-control col-md-7 col-xs-12" required="required"
type="text" value="<?php echo $data['latitude']; ?>">
        </div>
    </div>
    <div class="form-group">
        <label class="control-label col-md-3 col-
sm-3 col-xs-12">Posisi Longitude <span class="required">: </span>
        </label>
        <div class="col-md-6 col-sm-6 col-xs-12">
            <input name="txt_longitude" =""
class="form-control col-md-7 col-xs-12" required="required"
type="text" value="<?php echo $data['longitude']; ?>">
        </div>
    </div>
    <div class="form-group">
        <label class="control-label col-md-3 col-
sm-3 col-xs-12">Tambah Gambar <span class="required">: </span>
        </label>
        <div class="col-md-6 col-sm-6 col-xs-12">
            <img src=<?php echo
$data['gambar']; ?> width="250" height="200">
            <br>
            <br>
            <input type="file" name="file"
id="file">
        </div>
    </div>

    <div class="ln_solid"></div>
    <div class="form-group">
        <div class="col-md-6 col-sm-6 col-xs-12
col-md-offset-3">
            <a href="ponpes_main.php" class="btn
btn-warning">Batal </a>
            <input class="btn btn-success"
name="simpan" type="submit" value="Simpan">
        </div>
    </div>
</form>
</div>
</div>
</div>
</div>

```

```

</div>
</div>
<?php
include "footer.php";
?>

```

f. Kode program hapus data

```

<?php
include 'connect.php';
if (isset($_GET['id'])) {
    $dbh->exec("DELETE FROM m_pondok WHERE pondok_id =
'$_GET[id]'");
    if (isset($_GET['id'])) {
        $dbh->exec("DELETE FROM m_pondok WHERE pondok_id =
'$_GET[id]'");
        ;
    }
header("location:ponpes_main.php")
?>

```

g. Kode program detail data

```

<?php
include 'connect.php';
include "header.php";

if (isset($_GET['id'])) {
    $query = $dbh->query("SELECT * FROM m_pondok WHERE
pondok_id = '$_GET[id]'");
    $data = $query->fetch(PDO::FETCH_ASSOC);
} else {
    echo "ID tidak tersedia!<br /><a href='home.php'>Kembali</a>";
    exit();
}

if ($data === false) {
    echo "Data tidak ditemukan!<br /><a
href='home.php'>Kembali</a>";
    exit();
}
?>
<div class="right_col" role="main">
    <div class="row">

```



```

<div class="col-md-12 col-sm-12 col-xs-12">
  <div class="dashboard_graph">

    <div class="row x_title">
      <div class="col-md-6">
        <h3>Data Pondok Pesantren <small>Pondok
Pesantren Purbalingga</small></h3>
      </div>
    </div>
    <ol class="breadcrumb">
      <li><a href="home.php"><i class="fa fa-
home"></i> Home</a></li>
      <li><a href="ponpes_main.php"> Data
Ponpes</a></li>
      <li class="active">Detail Data Ponpes</a></li>
    </ol>
  </div>
  <div class="row">
    <div class="col-md-12 col-sm-12 col-xs-12">
      <div class="x_panel">
        <div class="x_title">
          <h2>Form Detail Data Ponpes </h2>
          <ul class="nav navbar-right panel_toolbox">

            </ul>
          <div class="clearfix"></div>
        </div>
        <div class="x_content">
          <br />
          <form name="form_viewpondok"
enctype="multipart/form-data" onsubmit="return validate(this)"
onClick="highlight(event)" onKeyUp="highlight(event)" data-parsley-
validate class="form-horizontal form-label-left" method="post">
            <input type="hidden" name="id" value="<?php
echo $data['pondok_id']; ?>" />
            <div class="form-group">
              <label class="control-label col-md-3 col-
sm-3 col-xs-12" for="nama">Nama Pondok Pesantren<span
class="required">: </span>
              </label>
              <div class="col-md-6 col-sm-6 col-xs-12">
                <input type="text" class="form-control"
readonly="readonly" value="<?php echo $data['nama_pondok']; ?>">
              </div>
            </div>
          </div>
        </div>
      </div>
    </div>
  </div>

```

```

<div class="form-group">
    <label class="control-label col-md-3 col-
sm-3 col-xs-12" for="alamat">Alamat <span class="required">:
</span>
        </label>
        <div class="col-md-6 col-sm-6 col-xs-12">
            <textarea type="text" class="form-
control" readonly="readonly" placeholder="-"><?php echo
$data['alamat']; ?></textarea>
        </div>
    </div>
    <div class="form-group">
        <label class="control-label col-md-3 col-
sm-3 col-xs-12" for="no_telepon">No. Telepon <span
class="required">: </span>
        </label>
        <div class="col-md-6 col-sm-6 col-xs-12">
            <input type="text" class="form-control"
readonly="readonly" placeholder="-" value="<?php echo
$data['no_telepon']; ?>">
        </div>
    </div>
    <div class="form-group">
        <label class="control-label col-md-3 col-
sm-3 col-xs-12" for="nama_pengasuh">Pengasuh <span
class="required">: </span>
        </label>
        <div class="col-md-6 col-sm-6 col-xs-12">
            <input type="text" class="form-control"
readonly="readonly" placeholder="-" value="<?php echo
$data['nama_pengasuh']; ?>">
        </div>
    </div>
    <div class="form-group">
        <label class="control-label col-md-3 col-
sm-3 col-xs-12" for="santri_putri">Santri Putri <span
class="required">: </span>
        </label>
        <div class="col-md-6 col-sm-6 col-xs-12">
            <input type="text" class="form-control"
readonly="readonly" placeholder="-" value="<?php echo
$data['santri_putri']; ?>">
        </div>
    </div>
</div>

```

```

<label class="control-label col-md-3 col-sm-3 col-xs-12"
for="santri_putra">Santri Putra <span class="required">: </span>
</label>
<div class="col-md-6 col-sm-6 col-xs-12">
<input type="text" class="form-control"
readonly="readonly" placeholder="-" value="<?php echo
$data['santri_putra']; ?>">
</div>
</div>
<div class="form-group">
<label class="control-label col-md-3 col-sm-
3 col-xs-12" for="jumlah">Jumlah Santri <span class="required">:
</span>
</label>
<div class="col-md-6 col-sm-6 col-xs-12">
<input type="text" class="form-control"
readonly="readonly" placeholder="-" value="<?php echo
$data['jumlah']; ?>">
</div>
</div>
<div class="form-group">
<label class="control-label col-md-3 col-sm-
3 col-xs-12" for="deskripsi">Deskripsi <span class="required">:
</span>
</label>
<div class="col-md-6 col-sm-6 col-xs-12">
<textarea type="text" class="form-control"
readonly="readonly" placeholder="-" ><?php echo $data['deskripsi'];
?></textarea>
</div>
</div>
<div class="form-group">
<label class="control-label col-md-3 col-sm-
3 col-xs-12">Website <span class="required">: </span>
</label>
<div class="col-md-6 col-sm-6 col-xs-12">
<input type="text" class="form-control"
readonly="readonly" placeholder="-" value="<?php echo
$data['website']; ?>">
</div>
</div>
<div class="form-group">
<label class="control-label col-md-3 col-sm-
3 col-xs-12">Blog <span class="required">: </span>
</label>

```

```

        <div class="col-md-6 col-sm-6 col-xs-12">
            <input type="text" class="form-control"
readonly="readonly" placeholder="-" value="<?php echo $data['blog'];
?>">
        </div>
    </div>

    <div class="form-group">
        <label class="control-label col-md-3 col-sm-
3 col-xs-12">Posisi Longitude <span class="required">: </span>
        </label>
        <div class="col-md-6 col-sm-6 col-xs-12">
            <input type="text" class="form-control"
readonly="readonly" placeholder="-" type="text" value="<?php echo
$data['longitude']; ?>">
        </div>
    </div>
    <div class="form-group">
        <label class="control-label col-md-3 col-sm-
3 col-xs-12">Posisi Latitude <span class="required">: </span>
        </label>
        <div class="col-md-6 col-sm-6 col-xs-12">
            <input type="text" class="form-control"
readonly="readonly" placeholder="-" value="<?php echo
$data['latitude']; ?>">
        </div>
    </div>
    <div class="form-group">
        <label class="control-label col-md-3 col-sm-
3 col-xs-12">Gambar <span class="required">: </span>
        </label>
        <div class="col-md-6 col-sm-6 col-xs-12">
            <img src=<?php echo $data['gambar']
?> width="250" height="200">
            <br>
            <br>
        </div>
    </div>

    <div class="ln_solid"></div>
    <div class="form-group">
        <div class="col-md-6 col-sm-6 col-xs-12
col-md-offset-3">
            <a href="ponpes_main.php" class="btn
btn-success">Kembali </a>

```



```

m_pondok");
$query = mysql_query("select * from
while ($data = mysql_fetch_array($query))
{?>
    <div class="col-md-55">
    <div class="thumbnail">
        <div class="image view view-first">
            <img style="width: 100%; display:
block;" src=?php echo $data['gambar'] ?> alt="image" />
            <div class="mask">
                <p><?php echo
$data['nama_pondok'] ?></p></a>
                <div class="tools tools-bottom">
                    <a data-toggle="modal" data-
target="#myModal<?php echo $data['pondok_id']; ?>"><i class="fa
fa-search-plus"></i></a>
                </div>
            </div>
            </div>
            </div>
            <div class="caption">
                <p><?php echo $data['alamat']
?></p>
            </div>
        </div>
    </div>

    <div class="modal fade bs-example-modal-
lg" id="myModal<?php echo $data['pondok_id']; ?>" tabindex="-1"
role="dialog" aria-hidden="true">
        <div class="modal-dialog modal-lg">
            <div class="modal-content">

                <div class="modal-header">
                    <button type="button" class="close"
data-dismiss="modal"><span aria-hidden="true"></span>
                    </button>
                    <h2 class="modal-title"
id="myModalLabel"><?php echo $data['nama_pondok'] ?></h2>
                </div>
                <div class="modal-body">
                    <img src=?php echo $data['gambar'] ?>
width="870" height="500">
                    <h4><?php echo $data['alamat']
?></h4>

                </div>
            </div>
        </div>
    </div>

```

```

        </div>
    </div>

    <?php
        }
    ?>
    <!-- Large modal -->

    </div>

    <!-- end content -->
    <br />
    <?php
    include "footer.php";
    ?>

```

2. Kode program aplikasi *android*

a. Kode program halaman *main*

```

package com.olga.ponpesbg;

import android.content.Context;
import android.content.Intent;
import android.graphics.Bitmap;
import android.os.AsyncTask;
import android.os.Bundle;
import android.os.Handler;
import android.support.v4.widget.SwipeRefreshLayout;
import android.view.LayoutInflater;
import android.view.View;
import android.support.design.widget.NavigationView;
import android.support.v4.view.GravityCompat;
import android.support.v4.widget.DrawerLayout;
import android.support.v7.app.ActionBarDrawerToggle;
import android.support.v7.app.AppCompatActivity;
import android.support.v7.widget.Toolbar;
import android.view.Menu;
import android.view.MenuItem;
import android.view.ViewGroup;
import android.widget.AbsListView;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ArrayAdapter;
import android.widget.ImageView;

```



```

import android.widget.ListView;
import android.widget.ProgressBar;
import android.widget.TextView;
import android.widget.Toast;

import com.nostra13.universalimageloader.core.DisplayImageOptions;
import com.nostra13.universalimageloader.core.ImageLoader;
import
com.nostra13.universalimageloader.core.ImageLoaderConfiguration;
import com.nostra13.universalimageloader.core.assist.FailReason;
import
com.nostra13.universalimageloader.core.listener.ImageLoadingListener
;
import com.olga.ponpespbg.model.ArtikelListModel;

import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;

import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.net.HttpURLConnection;
import java.net.MalformedURLException;
import java.net.URL;
import java.util.ArrayList;
import java.util.List;

public class MainActivity extends AppCompatActivity
    implements NavigationView.OnNavigationItemSelectedListener,
    AdapterView.OnItemClickListener {

    private String url_list =
"http://192.168.1.100/ponpespbgadmin/jsonponpes/artikel_list.php";
    private String url_image_loader =
"http://192.168.1.100/ponpespbgadmin/pages/";

    private ListView lvartikel;//untuk memanggil ke ponpes detail

    public static final String ARTIKEL_ID = "artikel_id";
    public static final String JUDUL_ARTIKEL = "judul_artikel";
    public static final String NAMA_PONDOK = "nama_pondok";
    public static final String PENULIS = "penulis";

```

```

public static final String NAMA_ADMIN = "nama_admin";
public static final String TANGGAL_TERBIT = "tanggal_terbit";
public static final String ISI_ARTIKEL = "isi_artikel";
public static final String GAMBAR = "gambar";

private List<ArtikelListModel> artikels;
SwipeRefreshLayout swipeRefresh;
private ProgressBar mProgressbar;
String status = "ada";

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    swipeRefresh = (SwipeRefreshLayout)
    findViewById(R.id.swipe_refresh);
    mProgressbar = (ProgressBar) findViewById(R.id.progressBar2);

    lvartikel = (ListView)findViewById(R.id.lvartikel);
    lvartikel.setOnItemClickListener(this);

    lvartikel.setOnScrollListener(new AbsListView.OnScrollListener()
    {
        @Override
        public void onScrollStateChanged(AbsListView view, int
        scrollState) {

        }

        @Override
        public void onScroll(AbsListView view, int firstVisibleItem, int
        visibleItemCount, int totalItemCount) {

            boolean enable = false;
            if (lvartikel != null && lvartikel.getChildCount() > 0) {
                // check if the first item of the list is visible
                boolean firstItemVisible =
                lvartikel.getFirstVisiblePosition() == 0;
                // check if the top of the first item is visible
                boolean topOfFirstItemVisible =
                lvartikel.getChildAt(0).getTop() == 0;
                // enabling or disabling the refresh layout
                enable = firstItemVisible && topOfFirstItemVisible;
            }
            swipeRefresh.setEnabled(enable);
        }
    });
}

```

```

    }
    });

    swipeRefresh.setColorSchemeResources(R.color.orange,
R.color.green, R.color.blue);
    swipeRefresh.setOnRefreshListener(new
SwipeRefreshLayout.OnRefreshListener() {
    @Override
    public void onRefresh() {
        new Handler().postDelayed(new Runnable() {
            @Override
            public void run() {
                swipeRefresh.setRefreshing(false);
                new JSONObject().execute(url_list);
            }
        }, 2000);
    }
    });
    DisplayImageOptions defaultOptions = new
DisplayImageOptions.Builder()
        .cacheInMemory(true).cacheOnDisk(true).build();

    ImageLoaderConfiguration config = new
ImageLoaderConfiguration.Builder(getApplicationContext())
        .defaultDisplayImageOptions(defaultOptions)
        .build();
    ImageLoader.getInstance().init(config); // Do it on Application
start

    Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
    setSupportActionBar(toolbar);

    new JSONObject().execute(url_list);
    mProgressbar.setVisibility(View.VISIBLE);

    lvartikel = (ListView) findViewById(R.id.lvartikel);

    // FloatingActionButton fab = (FloatingActionButton)
findViewById(R.id.fab);
    // fab.setOnClickListener(new View.OnClickListener() {
    //     @Override
    //     public void onClick(View view) {

```

```

//          Snackbar.make(view, "Replace with your own action",
Snackbar.LENGTH_LONG)
//          .setAction("Action", null).show();
//      }
//  });

    DrawerLayout drawer = (DrawerLayout)
findViewById(R.id.drawer_layout);
    ActionBarDrawerToggle toggle = new ActionBarDrawerToggle(
        this, drawer, toolbar, R.string.navigation_drawer_open,
R.string.navigation_drawer_close);
    drawer.setDrawerListener(toggle);
    toggle.syncState();

    NavigationView navigationView = (NavigationView)
findViewById(R.id.nav_view);
    navigationView.setNavigationItemSelectedListener(this);
}

@Override
public void onItemClick(AdapterView<?> parent, View view, int
position, long id) {
    Intent intent = new Intent(this, ArtikelDetail.class);
    ArtikelListModel artikelModel = artikels.get(position);
    intent.putExtra(ARTIKEL_ID, artikelModel.getArtikel_id());
    intent.putExtra(NAMA_PONDOK,
artikelModel.getNama_pondok());
    intent.putExtra(JUDUL_ARTIKEL,
artikelModel.getJudul_artikel());
    intent.putExtra(PENULIS, artikelModel.getPenulis());
    intent.putExtra(NAMA_ADMIN,
artikelModel.getNama_admin());
    intent.putExtra(TANGGAL_TERBIT,
artikelModel.getTanggal_terbit());
    intent.putExtra(ISI_ARTIKEL, artikelModel.getIsi_artikel());
    intent.putExtra(GAMBAR, artikelModel.getGambar());
    startActivity(intent);
}

public class JSONTask extends AsyncTask<String, String,
List<ArtikelListModel>> {
    @Override
    protected List<ArtikelListModel> doInBackground(String...
params) {
        HttpURLConnection connection = null;

```

```

BufferedReader reader = null;
try {
    URL url = new URL(params[0]);
    connection = (URLConnection) url.openConnection();
    connection.connect();
    InputStream stream = connection.getInputStream();
    reader = new BufferedReader(new
InputStreamReader(stream));
    StringBuffer buffer = new StringBuffer();
    String line = "";
    while ((line = reader.readLine()) != null) {
        buffer.append(line);
    }
    String finalJson = buffer.toString();
    JSONObject parentObject = new JSONObject(finalJson);
    JSONArray parentArray =
parentObject.getJSONArray("artikellist");
    String data = parentObject.getString("data");

    List<ArtikelListModel> artikelModelList = new
ArrayList<>();
    artikels = artikelModelList;

    if (data.equals("ada")) {
        for (int i = 0; i < parentArray.length(); i++) {
            JSONObject finalObject =
parentArray.getJSONObject(i);
            ArtikelListModel artikelModel = new
ArtikelListModel();

artikelModel.setArtikel_id(finalObject.getString("artikel_id"));

artikelModel.setJudul_artikel(finalObject.getString("judul_artikel"));

artikelModel.setNama_pondok(finalObject.getString("nama_pondok"));
;

artikelModel.setPenulis(finalObject.getString("penulis"));

artikelModel.setNama_admin(finalObject.getString("nama_admin"));

artikelModel.setTanggal_terbit(finalObject.getString("tanggal_terbit"));

artikelModel.setIsi_artikel(finalObject.getString("isi_artikel"));

```

```

artikelModel.setGambar(finalObject.getString("gambar"));

        artikelModelList.add(artikelModel);
    }
    return artikelModelList;
} else if (data.equals("kosong")) {
    status = "kosong";
}

} catch (MalformedURLException e) {
    e.printStackTrace();
} catch (JSONException e) {
    e.printStackTrace();
} catch (IOException e) {
    e.printStackTrace();
}
}
return null;
}

@Override
protected void onPostExecute(List<ArtikelListModel> result) {
    super.onPostExecute(result);
    mProgressbar.setVisibility(View.GONE);
    if (status.equals("kosong")) {
        Toast.makeText(getApplicationContext(), "Data tidak ada",
        Toast.LENGTH_SHORT).show();
        lartikel.setAdapter(null);
    } else {
        ArtikelAdapter adapter = new
        ArtikelAdapter(getApplicationContext(), R.layout.artikel_list, result);
        lartikel.setAdapter(adapter);
    }
}

public class ArtikelAdapter extends ArrayAdapter {
    private List<ArtikelListModel> artikelModelList;
    private int resource;
    private LayoutInflater inflater;

    public ArtikelAdapter(Context context, int resource, List objects)
    {
        super(context, resource, objects);
        {
            artikelModelList = objects;

```

```

        this.resource = resource;
        inflater = (LayoutInflater)
getSystemService(LAYOUT_INFLATER_SERVICE);
    }
}

@Override
public View getView(int position, View convertView, ViewGroup
parent) {
    if (convertView == null) {
        convertView = inflater.inflate(resource, null);

    }

    TextView tvJudulArtikel;
    TextView tvNamaPondok;
    ImageView imgaArtikel;

    tvJudulArtikel = (TextView)
convertView.findViewById(R.id.tvjudul);
    tvNamaPondok = (TextView)
convertView.findViewById(R.id.tvnamapondok);
    imgaArtikel = (ImageView)
convertView.findViewById(R.id.imgArtikel);

    tvNamaPondok.setText(artikelModelList.get(position).getNama_pondo
k());

    tvJudulArtikel.setText(artikelModelList.get(position).getJudul_artikel(
));

    final ProgressBar progressBar = (ProgressBar)
convertView.findViewById(R.id.progressBar);

    ImageLoader.getInstance().displayImage(url_image_loader +
artikelModelList.get(position).getGambar(), imgaArtikel, new
ImageLoadingListener() {
        @Override
        public void onLoadingStarted(String imageUri, View view) {
            progressBar.setVisibility(view.VISIBLE);
        }
    }

    @Override

```



```

        public void onLoadingFailed(String imageUrl, View view,
FailReason failReason) {
            progressBar.setVisibility(view.GONE);

        }

        @Override
        public void onLoadingComplete(String imageUrl, View
view, Bitmap loadedImage) {
            progressBar.setVisibility(view.GONE);

        }

        @Override
        public void onLoadingCancelled(String imageUrl, View
view) {
            progressBar.setVisibility(view.GONE);
        }
    });
    return convertView;
}
}

@Override
public void onBackPressed() {
    DrawerLayout drawer = (DrawerLayout)
findViewById(R.id.drawer_layout);
    if (drawer.isDrawerOpen(GravityCompat.START)) {
        drawer.closeDrawer(GravityCompat.START);
    } else {
        super.onBackPressed();
    }
}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    // Inflate the menu; this adds items to the action bar if it is
present.
    getMenuInflater().inflate(R.menu.main, menu);
    return true;
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    // Handle action bar item clicks here. The action bar will
// automatically handle clicks on the Home/Up button, so long

```

```

// as you specify a parent activity in AndroidManifest.xml.
int id = item.getItemId();

//noinspection SimplifiableIfStatement
if (id == R.id.action_settings) {
    Intent i = new Intent(MainActivity.this, Tentang.class);
    startActivity(i);
    return true;
}

return super.onOptionsItemSelected(item);
}

@SuppressWarnings("StatementWithEmptyBody")
@Override
public boolean onOptionsItemSelected(MenuItem item) {
    // Handle navigation view item clicks here.
    int id = item.getItemId();

    if (id == R.id.data_pondok) {
        Intent i = new Intent(MainActivity.this, PonpesMain.class);
        startActivity(i); //manggil layout

        // Handle the camera action
    } else if (id == R.id.peta) {
        Intent i = new Intent(MainActivity.this, PetaPonpes.class);
        startActivity(i); //manggil layout

    } else if (id == R.id.kemenag) {
        Intent i = new Intent(MainActivity.this, Kemenag.class);
        startActivity(i);

    } else if (id == R.id.aplikasi) {
        Intent i = new Intent(MainActivity.this, Tentang.class);
        startActivity(i);

    }

    DrawerLayout drawer = (DrawerLayout)
findViewById(R.id.drawer_layout);
    drawer.closeDrawer(GravityCompat.START);
    return true;
}
}

```

b. Kode program halaman *list data*

```
package com.olga.ponpespbg;

import android.content.Context;
import android.content.Intent;
import android.graphics.Bitmap;
import android.os.AsyncTask;
import android.os.Bundle;
import android.os.Handler;
import android.support.design.widget.FloatingActionButton;
import android.support.design.widget.Snackbar;
import android.support.v4.view.MenuItemCompat;
import android.support.v4.widget.SwipeRefreshLayout;
import android.support.v7.app.AppCompatActivity;
import android.support.v7.widget.Toolbar;
import android.view.LayoutInflater;
import android.view.Menu;
import android.view.MenuInflater;
import android.view.MenuItem;
import android.view.View;
import android.view.ViewGroup;
import android.widget.AbsListView;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ArrayAdapter;
import android.widget.ImageView;
import android.widget.ListView;
import android.widget.ProgressBar;
import android.support.v7.widget.SearchView;
import android.widget.TextView;
import android.widget.Toast;

import
com.nostra13.universalimageloader.core.DisplayImageOptions;
import com.nostra13.universalimageloader.core.ImageLoader;
import
com.nostra13.universalimageloader.core.ImageLoaderConfiguration;
import com.nostra13.universalimageloader.core.assist.FailReason;
import
com.nostra13.universalimageloader.core.listener.ImageLoadingListe
ner;
import com.olga.ponpespbg.model.PonpesListModel;

import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;
```

```

import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.net.HttpURLConnection;
import java.net.MalformedURLException;
import java.net.URL;
import java.nio.channels.AsynchronousCloseException;
import java.util.ArrayList;
import java.util.List;

public class PonpesMain extends AppCompatActivity implements
ListView.OnItemClickListener {

    private String url_list =
"http://192.168.1.100/ponpespbgadmin/jsonponpes/ponpes_list.php";
    private String url_image_loader =
"http://192.168.1.100/ponpespbgadmin/pages/";
    private String url_search =
"http://192.168.1.100/ponpespbgadmin/jsonponpes/ponpes_search.ph
p?nama_pondok=";

    private ListView lvpondok;//untuk memanggil ke ponpes detail
    public static final String PONDOK_ID = "pondok_id";
    public static final String NAMA_PONDOK = "nama_pondok";
    public static final String ALAMAT = "alamat";
    public static final String NO_TELEPON = "no_telepon";
    public static final String NAMA_PENGASUH =
"nama_pengasuh";
    public static final String SANTRI_PUTRA = "santri_putra";
    public static final String SANTRI_PUTRI = "santri_putri";
    public static final String JUMLAH = "jumlah";
    public static final String DESKRIPSI = "deskripsi";
    public static final String WEBSITE = "website";
    public static final String BLOG = "blog";
    public static final String LATITUDE = "latitude";
    public static final String LONGITUDE = "longitude";
    public static final String GAMBAR = "gambar";

    private List<PonpesListModel> pondoks;
    SwipeRefreshLayout swipeRefresh;
    private ProgressBar mProgressbar;
    String status = "ada";

    @Override

```

```

protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_ponpes_main);

    swipeRefresh = (SwipeRefreshLayout)
findViewById(R.id.swipe_refresh);
    mProgressbar = (ProgressBar) findViewById(R.id.progressBar2);

    lvpondok = (ListView)findViewById(R.id.lvpondok);
    lvpondok.setOnItemClickListener(this);

    lvpondok.setOnScrollListener(new
AbsListView.OnScrollListener() {

        @Override
        public void onScrollStateChanged(AbsListView view, int
scrollState) {

        }

        @Override
        public void onScroll(AbsListView view, int firstVisibleItem,
int visibleItemCount, int totalItemCount) {
            boolean enable = false;
            if (lvpondok != null && lvpondok.getChildCount() > 0) {
                // check if the first item of the list is visible
                boolean firstItemVisible =
lvpondok.getFirstVisiblePosition() == 0;
                // check if the top of the first item is visible
                boolean topOfFirstItemVisible =
lvpondok.getChildAt(0).getTop() == 0;
                // enabling or disabling the refresh layout
                enable = firstItemVisible && topOfFirstItemVisible;
            }
            swipeRefresh.setEnabled(enable);
        }
    });

    swipeRefresh.setColorSchemeResources(R.color.orange,
R.color.green, R.color.blue);
    swipeRefresh.setOnRefreshListener(new
SwipeRefreshLayout.OnRefreshListener() {
        @Override
        public void onRefresh() {
            new Handler().postDelayed(new Runnable() {

```

```

        @Override
        public void run() {
            swipeRefresh.setRefreshing(false);
            new JSONTask().execute(url_list);
        }
    }, 2000);
    });

    //image loader
    DisplayImageOptions defaultOptions = new
    DisplayImageOptions.Builder()
        .cacheInMemory(true).cacheOnDisk(true).build();

    ImageLoaderConfiguration config = new
    ImageLoaderConfiguration.Builder(getApplicationContext())
        .defaultDisplayImageOptions(defaultOptions)
        .build();
    ImageLoader.getInstance().init(config); // Do it on Application
    start

    Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
    setSupportActionBar(toolbar);

    new JSONTask().execute(url_list);
    mProgressbar.setVisibility(View.VISIBLE);

    toolbar.setNavigationIcon(R.mipmap.ic_keyboard_arrow_left_white_2
    4dp);
    toolbar.setNavigationOnClickListener(new
    View.OnClickListener() {
        @Override
        public void onClick(View v) {
            onBackPressed();
        }
    });

    mProgressbar.setVisibility(View.VISIBLE);

    lvpondok = (ListView)findViewById(R.id.lvpondok);

    // FloatingActionButton fab = (FloatingActionButton)
    findViewById(R.id.fab);
    // fab.setOnClickListener(new View.OnClickListener() {
    //     @Override

```

```

//      public void onClick(View view) {
//          Snackbar.make(view, "Replace with your own action",
Snackbar.LENGTH_LONG)
//              .setAction("Action", null).show();
//      }
//  });
}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    MenuInflater inflater = getMenuInflater();
    inflater.inflate(R.menu.menu_search, menu);
    final MenuItem searchItem = menu.findItem(R.id.search);
    searchItem.setIcon(R.mipmap.ic_search_white_24dp);
    final SearchView searchView = (SearchView)
MenuItemCompat.getActionView(searchItem);
    searchView.setOnQueryTextListener(new
SearchView.OnQueryTextListener() {
        @Override
        public boolean onQueryTextSubmit(String query) {
            new JSOINTask().execute(url_search+query.toString());
            searchView.clearFocus();
            return true;
        }

        @Override
        public boolean onQueryTextChange(String newText) {
            return false;
        }
    });

    return super.onCreateOptionsMenu(menu);
}

public class JSOINTask extends AsyncTask<String, String,
List<PonpesListModel>> { //transfer data website ke android
    @Override
    protected List<PonpesListModel> doInBackground(String...
params) {
        HttpURLConnection connection = null;
        BufferedReader reader = null;
        try {
            URL url = new URL(params[0]);
            connection = (HttpURLConnection) url.openConnection();
            connection.connect();
            InputStream stream = connection.getInputStream();

```



```

        reader = new BufferedReader(new
InputStreamReader(stream));
        StringBuffer buffer = new StringBuffer();
        String line = "";
        while ((line = reader.readLine()) != null) {
            buffer.append(line);
        }
        String finalJson = buffer.toString();
        JSONObject parrentObject = new JSONObject(finalJson);
        JSONArray parrentArray =
parrentObject.getJSONArray("pondoklist");
        String data = parrentObject.getString("data");

        List<PonpesListModel> ponpesModelList = new
ArrayList<>();
        pondoks = ponpesModelList;

        if (data.equals("ada")) {
            for (int i = 0; i < parrentArray.length(); i++) {
                JSONObject finalObject =
parrentArray.getJSONObject(i);
                PonpesListModel ponpesModel = new
PonpesListModel();

                ponpesModel.setPondok_id(finalObject.getString("pondok_id"));

                ponpesModel.setNama_pondok(finalObject.getString("nama_pondok"
));

                ponpesModel.setAlamat(finalObject.getString("alamat"));

                ponpesModel.setNo_telepon(finalObject.getString("no_telepon"));

                ponpesModel.setNama_pengasuh(finalObject.getString("nama_penga
suh"));

                ponpesModel.setSantri_putri(finalObject.getString("santri_putri"));

                ponpesModel.setSantri_putra(finalObject.getString("santri_putra"));

                ponpesModel.setJumlah(finalObject.getString("jumlah"));

                ponpesModel.setDeskripsi(finalObject.getString("deskripsi"));

                ponpesModel.setWebsite(finalObject.getString("website"));
                ponpesModel.setBlog(finalObject.getString("blog"));
            }
        }

```

```

ponpesModel.setLatitude(finalObject.getString("latitude"));

ponpesModel.setLongitude(finalObject.getString("longitude"));

ponpesModel.setGambar(finalObject.getString("gambar"));

        ponpesModelList.add(ponpesModel);
    }
    return ponpesModelList;
} else if (data.equals("kosong")) {
    status = "kosong";
}
} catch (MalformedURLException e) {
    e.printStackTrace();
} catch (IOException e) {
    e.printStackTrace();
} catch (JSONException e) {
    e.printStackTrace();
} finally {
    if (connection != null) {
        connection.disconnect();
    }
    try {
        if (reader != null) {
            reader.close();
        }
    } catch (IOException e) {
        e.printStackTrace();
    }
}
}
return null;
}

@Override
protected void onPostExecute(List<PonpesListModel> result) {
    super.onPostExecute(result);
    mProgressbar.setVisibility(View.GONE);
    if (status.equals("kosong")) {
        Toast.makeText(getApplicationContext(), "Data tidak ada",
Toast.LENGTH_SHORT).show();
        lvpondok.setAdapter(null);
    } else {
        PondokAdapter adapter = new
PondokAdapter(getApplicationContext(), R.layout.pondok_list, result);

```

```

    }
    }
}

public class PondokAdapter extends ArrayAdapter {
    private List<PonpesListModel> ponpesModelList;
    private int resource;
    private LayoutInflater inflater;

    public PondokAdapter(Context context, int resource, List
objects) {
        super(context, resource, objects);
        {
            ponpesModelList = objects;
            this.resource = resource;
            inflater = (LayoutInflater)
getSystemService(LAYOUT_INFLATER_SERVICE);
        }
    }

    @Override
    public View getView(int position, View convertView,
ViewGroup parent) {
        if (convertView == null) {
            convertView = inflater.inflate(resource, null);

        }
        TextView tvNamaPondok;
        TextView tvAlamat;
        ImageView imgPondok;

        tvNamaPondok = (TextView)
convertView.findViewById(R.id.tvNamaPondok);
        tvAlamat = (TextView)
convertView.findViewById(R.id.tvAlamat);
        imgPondok = (ImageView)
convertView.findViewById(R.id.imgPondok);

        tvNamaPondok.setText(ponpesModelList.get(position).getNama_pon
dok());

        tvAlamat.setText(ponpesModelList.get(position).getAlamat());

```

```

        final ProgressBar progressBar = (ProgressBar)
convertView.findViewById(R.id.progressBar);

        ImageLoader.getInstance().displayImage(url_image_loader
+ ponpesModelList.get(position).getGambar(), imgPondok, new
ImageLoadingListener() {
            @Override
            public void onLoadingStarted(String imageUri, View
view) {
                progressBar.setVisibility(view.VISIBLE);
            }

            @Override
            public void onLoadingFailed(String imageUri, View
view, FailReason failReason) {
                progressBar.setVisibility(view.GONE);
            }

            @Override
            public void onLoadingComplete(String imageUri, View
view, Bitmap loadedImage) {
                progressBar.setVisibility(view.GONE);
            }

            @Override
            public void onLoadingCancelled(String imageUri, View
view) {
                progressBar.setVisibility(view.GONE);
            }
        });
        return convertView;
    }

    @Override
    public void onItemClick(AdapterView<?> parent, View view, int
position, long id) {
        Intent intent = new Intent(this, PonpesDetail.class);
        PonpesListModel ponpesModel = pondoks.get(position);
        intent.putExtra(PONDOK_ID, ponpesModel.getPondok_id());
        intent.putExtra(NAMA_PONDOK,
ponpesModel.getNama_pondok());
        intent.putExtra(ALAMAT, ponpesModel.getAlamat());
    }

```

```

        intent.putExtra(NO_TELEPON,
ponpesModel.getNo_telepon());
        intent.putExtra(NAMA_PENGASUH,
ponpesModel.getNama_pengasuh());
        intent.putExtra(SANTRI_PUTRA,
ponpesModel.getSantri_putra());
        intent.putExtra(SANTRI_PUTRI,
ponpesModel.getSantri_putri());
        intent.putExtra(JUMLAH, ponpesModel.getJumlah());
        intent.putExtra(DESKRIPSI, ponpesModel.getDeskripsi());
        intent.putExtra(WEBBSITE, ponpesModel.getWebsite());
        intent.putExtra(BLOG, ponpesModel.getBlog());
        intent.putExtra(LATITUDE, ponpesModel.getLatitude());
        intent.putExtra(LONGITUDE, ponpesModel.getLongitude());
        intent.putExtra(GAMBAR, ponpesModel.getGambar());
        startActivity(intent);

    }

}

```

c. Kode program halaman detail data

```

package com.olga.ponpespbg;

import android.os.Bundle;
import android.support.design.widget.FloatingActionButton;
import android.support.design.widget.Snackbar;
import android.support.v7.app.AppCompatActivity;
import android.support.v7.widget.Toolbar;
import android.text.method.ScrollingMovementMethod;
import android.text.util.Linkify;
import android.view.View;
import android.widget.TextView;

import com.google.android.gms.maps.CameraUpdate;
import com.google.android.gms.maps.CameraUpdateFactory;
import com.google.android.gms.maps.GoogleMap;
import com.google.android.gms.maps.MapFragment;
import com.google.android.gms.maps.OnMapReadyCallback;
import
com.google.android.gms.maps.model.BitmapDescriptorFactory;
import com.google.android.gms.maps.model.LatLng;

```

```

import com.google.android.gms.maps.model.MarkerOptions;

import org.w3c.dom.Text;

public class PonpesDetail extends AppCompatActivity implements
OnMapReadyCallback {

    GoogleMap mMap;
    private String ponpes_id, nama_pondok, alamat, no_telepon,
nama_pengasuh, jumlah_santri, website, blog, deskripsi, gambar;
    private double latitude, longitude;
    private TextView tvNo_Telepon, tvpengasuh, tvJumlah_Santri,
tvWebsite, tvblog, tvDeskripsi;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_ponpes_detail);
        initMap();

        ponpes_id =
getIntent().getStringExtra(PonpesMain.PONDOK_ID);
        nama_pondok=
getIntent().getStringExtra(PonpesMain.NAMA_PONDOK);
        alamat = getIntent().getStringExtra(PonpesMain.ALAMAT);
        no_telepon =
getIntent().getStringExtra(PonpesMain.NO_TELEPON);
        nama_pengasuh =
getIntent().getStringExtra(PonpesMain.NAMA_PENGASUH);
        jumlah_santri =
getIntent().getStringExtra(PonpesMain.JUMLAH);
        website = getIntent().getStringExtra(PonpesMain.WEBSITE);
        blog = getIntent().getStringExtra(PonpesMain.BLOG);
        deskripsi =
getIntent().getStringExtra(PonpesMain.DESKRIPSI);
        gambar = getIntent().getStringExtra(PonpesMain.GAMBAR);

        latitude =
Double.parseDouble(getIntent().getStringExtra(PonpesMain.LATIT
UDE));
        longitude =
Double.parseDouble(getIntent().getStringExtra(PonpesMain.LONGI
TUDE));

        tvNo_Telepon = (TextView)findViewById(R.id.tvno_telepon);
        tvNo_Telepon.setText("No Telp : " + no_telepon);

```

```

Linkify.addLinks(tvNo_Telepon, Linkify.PHONE_NUMBERS);

    tvpengasuh = (TextView)findViewById(R.id.tvpengasuh);
    tvpengasuh.setText("Pengasuh : " + nama_pengasuh);

    tvJumlah_Santri =
(TextView)findViewById(R.id.tvjumlah_santri);
    tvJumlah_Santri.setText("Jumlah Santri : " + jumlah_santri);

    tvWebsite = (TextView)findViewById(R.id.tvWebsite);
    tvWebsite.setText("Website : " + website);
    Linkify.addLinks(tvWebsite, Linkify.WEB_URLS);

    tvblog = (TextView)findViewById(R.id.tvblog);
    tvblog.setText("Blog : " + blog);
    Linkify.addLinks(tvblog, Linkify.WEB_URLS);

    tvDeskripsi =(TextView)findViewById(R.id.tvDeskripsi);
    tvDeskripsi.setText(deskripsi);
    tvDeskripsi.setMovementMethod(new
ScrollingMovementMethod());

    Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
    setSupportActionBar(toolbar);

    toolbar.setTitle(nama_pondok);//membuat nama toolbar sesuai
nama pondok
    setSupportActionBar(toolbar);

    toolbar.setNavigationIcon(R.mipmap.ic_keyboard_arrow_left_white_
24dp);
    toolbar.setNavigationOnClickListener(new
View.OnClickListener() {//membuat back
        @Override
        public void onClick(View v) {
            onBackPressed();
        }
    });

    FloatingActionButton fab = (FloatingActionButton)
findViewById(R.id.fab);
    // fab.setOnClickListener(new View.OnClickListener() {
    //     @Override
    //     public void onClick(View view) {

```



```

//          Snackbar.make(view, "Replace with your own action",
Snackbar.LENGTH_LONG)

//          .setAction("Action", null).show();
//      }
//  });
}

private void initMap() {
    MapFragment mapFragment = (MapFragment)
getFragmentManager().findFragmentById(R.id.mapDetailFragment);
    mapFragment.getMapAsync(this);

}

@Override
public void onMapReady(GoogleMap googleMap) {
    mMap = googleMap;
    LatLng marker = new LatLng(latitude, longitude);
    goTOLocateZoom(latitude, longitude, 10);
    mMap.addMarker(new
MarkerOptions().position(marker).icon(BitmapDescriptorFactory.defa
ultMarker(BitmapDescriptorFactory.HUE_ORANGE)).title(nama_po
ndok).snippet(alamat)).showInfoWindow();

    googleMap.setMyLocationEnabled(true);
}

```

d. Kode program detail artikel

```

package com.olga.ponpespbg;

import android.graphics.Bitmap;
import android.net.Uri;
import android.os.Bundle;
import android.support.design.widget.FloatingActionButton;
import android.support.design.widget.Snackbar;
import android.support.v7.app.AppCompatActivity;
import android.support.v7.widget.Toolbar;
import android.view.View;
import android.widget.ImageView;
import android.widget.ProgressBar;
import android.widget.TextView;

import com.nostra13.universalimageloader.core.DisplayImageOptions;
import com.nostra13.universalimageloader.core.ImageLoader;

```

```

import
com.nostra13.universalimageloader.core.ImageLoaderConfiguration;
import com.nostra13.universalimageloader.core.assist.FailReason;
import
com.nostra13.universalimageloader.core.listener.ImageLoadingListen
er;

public class ArtikelDetail extends AppCompatActivity {

    private      String      url_image_loader      =
"http://192.168.1.100/ponpespbgadmin/pages/";

    private String nama_pondok, judul_artikel, tanggal_terbit, penulis,
isi_artikel, gambar;
    private      TextView      tvNama_pondok,      tvJudulArtikel,
tvTanggalTerbit, tvPenulis, tvArtikel;
    private ImageView imgArtikel;

    private ProgressBar progressBar;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_artikel_detail);
        Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
        setSupportActionBar(toolbar);

        toolbar.setNavigationIcon(R.mipmap.ic_keyboard_arrow_left_white_
24dp);
        toolbar.setNavigationOnClickListener(new
View.OnClickListener() { //membuat back
            @Override
            public void onClick(View v) {
                onBackPressed();
            }
        });
        gambar = getIntent().getStringExtra(PonpesMain.GAMBAR);
        imgArtikel = (ImageView)findViewById(R.id.imgDetail);

        DisplayImageOptions      defaultOptions      =      new
DisplayImageOptions.Builder()
            .cacheInMemory(true).cacheOnDisk(true).build();
        ImageLoaderConfiguration      config      =      new
ImageLoaderConfiguration.Builder(getApplicationContext())

```

```

.build();

    ImageLoader.getInstance().init(config); // Do it on Application
start

    progressBar = (ProgressBar)findViewById(R.id.progressBar3);

    ImageLoader.getInstance().displayImage(url_image_loader +
gambar, imgArtikel, new ImageLoadingListener() {
    @Override
    public void onLoadingStarted(String imageUri, View view) {
        progressBar.setVisibility(view.VISIBLE);
    }

    @Override
    public void onLoadingFailed(String imageUri, View view,
FailReason failReason) {
        progressBar.setVisibility(view.GONE);

    }

    @Override
    public void onLoadingComplete(String imageUri, View view,
Bitmap loadedImage) {
        progressBar.setVisibility(view.GONE);

    }

    @Override
    public void onLoadingCancelled(String imageUri, View
view) {
        progressBar.setVisibility(view.GONE);
    }
});

    nama_pondok =
getIntent().getStringExtra(MainActivity.NAMA_PONDOK);
    judul_artikel=
getIntent().getStringExtra(MainActivity.JUDUL_ARTIKEL);
    tanggal_terbit =
getIntent().getStringExtra(MainActivity.TANGGAL_TERBIT);
    penulis = getIntent().getStringExtra(MainActivity.PENULIS);
    isi_artikel =
getIntent().getStringExtra(MainActivity.ISI_ARTIKEL);

```

```

tvNama_pondok =(TextView)findViewById(R.id.tvNamaPondok);

    tvNama_pondok.setText(nama_pondok);

    tvJudulArtikel =(TextView)findViewById(R.id.tvJudulArtikel);
    tvJudulArtikel.setText(judul_artikel);

    tvTanggalTerbit
=(TextView)findViewById(R.id.tvTanggalTerbit);
    tvTanggalTerbit.setText(tanggal_terbit);

    tvPenulis =(TextView)findViewById(R.id.tvPenulis);
    tvPenulis.setText(penulis);

    tvArtikel =(TextView)findViewById(R.id.tvArtikelIsi);
    tvArtikel.setText(isi_artikel);

    //    FloatingActionButton fab = (FloatingActionButton)
    findViewById(R.id.fab);
    //    fab.setOnClickListener(new View.OnClickListener() {
    //        @Override
    //        public void onClick(View view) {
    //            Snackbar.make(view, "Replace with your own action",
    Snackbar.LENGTH_LONG)
    //                .setAction("Action", null).show();
    //        }
    //    });
    }

}

```

e. Kode program peta pondok

```

package com.olga.ponpespbg;

import android.content.Context;
import android.graphics.Color;
import android.graphics.Typeface;
import android.os.Bundle;
import android.support.design.widget.FloatingActionButton;
import android.support.design.widget.Snackbar;
import android.support.v7.app.AppCompatActivity;
import android.support.v7.widget.Toolbar;
import android.util.Log;
import android.view.Gravity;

```

```

import android.view.View;
import android.widget.LinearLayout;
import android.widget.TextView;

import com.google.android.gms.maps.CameraUpdate;
import com.google.android.gms.maps.CameraUpdateFactory;
import com.google.android.gms.maps.GoogleMap;
import com.google.android.gms.maps.SupportMapFragment;
import
com.google.android.gms.maps.model.BitmapDescriptorFactory;
import com.google.android.gms.maps.model.LatLng;
import com.google.android.gms.maps.model.Marker;
import com.google.android.gms.maps.model.MarkerOptions;

import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;

import java.io.IOException;
import java.io.InputStreamReader;
import java.net.HttpURLConnection;
import java.net.URL;

public class PetaPondok extends AppCompatActivity {

    private static final String LOG_TAG = "ExampleApp";

    private static final String SERVICE_URL =
"http://192.168.1.100/ponpespbgadmin/jsonponpes/peta_ponpes.php"
;

    protected GoogleMap map;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_peta_pondok);

        setUpMapIfNeeded();

        Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
        setSupportActionBar(toolbar);

        toolbar.setNavigationIcon(R.mipmap.ic_keyboard_arrow_left_white_
24dp);
        toolbar.setNavigationOnClickListener(new
View.OnClickListener() {

```

@Override

```
        public void onClick(View v) {
            onBackPressed();
        }
    });

}

    public void setUpMapIfNeeded() {
        if (map == null) {
            map = ((SupportMapFragment)
getSupportFragmentManager().findFragmentById(R.id.FragPetaPond
ok)).getMap();
            if (map != null) {
                setUpMap();
                goToLocationZoom(-7.389609, 109.363130, 9);
                map.setMyLocationEnabled(true);
            }
        }
    }

    public void setUpMap() {
        // Retrieve the city data from the web service
        // In a worker thread since it's a network operation.
        new Thread(new Runnable() {
            public void run() {
                try {
                    retrieveAndAddCities();
                } catch (IOException e) {
                    Log.e(LOG_TAG, "Cannot retrieve cities", e);
                    return;
                }
            }
        }).start();
    }

    public void retrieveAndAddCities() throws IOException {
        HttpURLConnection conn = null;
        final StringBuilder json = new StringBuilder();
        try {
            // Connect to the web service
            URL url = new URL(SERVICE_URL);
            conn = (HttpURLConnection) url.openConnection();
```

```

InputStreamReader in = new InputStreamReader(conn.getInputStream());

    // Read the JSON data into the StringBuilder
    int read;
    char[] buff = new char[1024];
    while ((read = in.read(buff)) != -1) {
        json.append(buff, 0, read);
    }
} catch (IOException e) {
    Log.e(LOG_TAG, "Error connecting to service", e);
    throw new IOException("Error connecting to service", e);
} finally {
    if (conn != null) {
        conn.disconnect();
    }
}

// Create markers for the city data.
// Must run this on the UI thread since it's a UI operation.
runOnUiThread(new Runnable() {
    public void run() {
        try {
            createMarkersFromJson(json.toString());
        } catch (JSONException e) {
            Log.e(LOG_TAG, "Error processing JSON", e);
        }
    }
});
}

void createMarkersFromJson(String json) throws JSONException {
    // De-serialize the JSON string into an array of city objects
    JSONArray jsonArray = new JSONArray(json);
    for (int i = 0; i < jsonArray.length(); i++) {
        // Create a marker for each city in the JSON data.
        JSONObject jsonObj = jsonArray.getJSONObject(i);

        map.addMarker(new MarkerOptions()
            .title(jsonObj.getString("nama_pondok"))
            .snippet(jsonObj.getString("alamat"))
            .position(new LatLng(
                jsonObj.getJSONArray("latlng").getDouble(0),
                jsonObj.getJSONArray("latlng").getDouble(1)
            ))
        );
    }
}

```



```

    ).icon(BitmapDescriptorFactory.defaultMarker(BitmapDescriptorFactory.HUE_
    ORANGE))

        );

        LatLng kemenag = new LatLng(-7.385901, 109.366011);

        map.addMarker(new
        MarkerOptions().position(kemenag).icon(BitmapDescriptorFactory.de
        faultMarker(BitmapDescriptorFactory.HUE_RED)).title("Kementrian
        Agama Purbalingga").snippet("Mayjend DI. Panjaitan").flat(true));

        map.setInfoWindowAdapter(new
        GoogleMap.InfoWindowAdapter() {
            @Override
            public View getInfoWindow(Marker marker) {
                return null;
            }

            @Override
            public View getInfoContents(Marker marker) {

                Context mContext = getApplicationContext();
                LinearLayout info = new LinearLayout(mContext);
                info.setOrientation(LinearLayout.VERTICAL);

                TextView title = new TextView(mContext);
                title.setTextColor(Color.BLACK);
                title.setGravity(Gravity.CENTER);
                title.setTypeface(null, Typeface.BOLD);
                title.setText(marker.getTitle());

                TextView snippet = new TextView(mContext);
                snippet.setTextColor(Color.GRAY);
                snippet.setText(marker.getSnippet());

                info.addView(title);
                info.addView(snippet);

                return info;
            }
        });
    }
}

```

```

private void goToLocationZoom(double lat, double lng, int zoom) {

    LatLng ll = new LatLng(lat, lng);
    CameraUpdate update =
    CameraUpdateFactory.newLatLngZoom(ll, zoom);
    map.moveCamera(update);
}

}

```

f. Kode program info Kemenag

```

package com.olga.ponpespbg;

import android.content.Intent;
import android.os.Bundle;
import android.support.design.widget.FloatingActionButton;
import android.support.v7.app.AppCompatActivity;
import android.support.v7.widget.Toolbar;
import android.text.util.Linkify;
import android.view.View;
import android.widget.TextView;

public class Kemenag extends AppCompatActivity {

    private TextView tvtelepon, tvwebsite;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_kemenag);
        Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
        setSupportActionBar(toolbar);

        toolbar.setNavigationIcon(R.mipmap.ic_keyboard_arrow_left_white_24dp);
        toolbar.setNavigationOnClickListener(new
        View.OnClickListener() { //membuat back
            @Override
            public void onClick(View v) {
                onBackPressed();
            }
        });
    }
}

```

```

    }
    });

    FloatingActionButton fab = (FloatingActionButton)
findViewById(R.id.fab);
    fab.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            Intent i = new Intent(Kemenag.this, PetaPondok.class);
            startActivity(i);//manggil peta
        }
    });

    tvtelepon = (TextView)findViewById(R.id.tvtelepon);
    Linkify.addLinks(tvtelepon, Linkify.PHONE_NUMBERS);

    tvwebsite = (TextView)findViewById(R.id.tvwebsite);
    Linkify.addLinks(tvwebsite, Linkify.WEB_URLS);

}
}

```

g. Kode program tentang aplikasi

```

package com.olga.ponpespbg;

import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.support.v7.widget.Toolbar;
import android.text.util.Linkify;
import android.view.View;
import android.widget.TextView;

public class Tentang extends AppCompatActivity {

    private TextView tvemail;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_tentang);
        Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
        setSupportActionBar(toolbar);
    }
}

```

```
toolbar.setNavigationIcon(R.mipmap.ic_keyboard_arrow_left_white_24dp);
    toolbar.setNavigationOnClickListener(new
View.OnClickListener() { //membuat back
    @Override
    public void onClick(View v) {
        onBackPressed();
    }
});

tvemail = (TextView)findViewById(R.id.tvemail);
Linkify.addLinks(tvemail, Linkify.EMAIL_ADDRESSES);
}
}
```