RANCANG BANGUN APLIKASI PENSTATUSAN SURAT PEMBERITAHUAN KEWAJIBAN PEMILIK KENDARAAN BERMOTOR (SPKPKB) BERBASIS ANDROID MENGGUNAKAN GPS (*GLOBAL POSITIONING SYSTEM*)

SKRIPSI

Untuk Memenuhi Sebagian Persyaratan Untuk Mencapai Gelar Sarjana Komputer

Disusun Oleh :

AHMAD NASHRUDDIN

10506080211005

PROGRAM STUDI INFORMATIKA

PROGRAM TEKNOLOGI INFORMASI DAN ILMU KOMPUTER

UNIVERSITAS BRAWIJAYA

2015

repository.ub.ac.id

LEMBAR PERSETUJUAN

RANCANG BANGUN APLIKASI PENSTATUSAN SURAT PEMBERITAHUAN KEWAJIBAN PEMILIK KENDARAAN BERMOTOR (SPKPKB) BERBASIS ANDROID MENGGUNAKAN GPS (*GLOBAL POSITIONING SYSTEM*)

SKRIPSI

Untuk memenuhi sebagian persyaratan mencapai gelar Sarjana Komputer

Disusun Oleh :

AHMAD NASHRUDDIN
105060802111005

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I	Dosen Pembimbing II
Aryo Pinandito, S.T., M.MT.	**Agi Putra Kharisma, S.T., M.T.**
NIP. 198305192014041001	**NIK. 2013048604301001**

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

LEMBAR PENGESAHAN

RANCANG BANGUN APLIKASI PENSTATUSAN SURAT PEMBERITAHUAN KEWAJIBAN PEMILIK KENDARAAN BERMOTOR (SPKPKB) BERBASIS ANDROID MENGGUNAKAN GPS (*GLOBAL POSITIONING SYSTEM*)

SKRIPSI

Untuk memenuhi sebagian persyaratan mencapai gelar Sarjana Komputer

Disusun oleh :

AHMAD NASHRUDDIN
105060802111005

Skripsi ini telah diuji dan dinyatakan lulus pada tanggal 8 Juli 2015

Penguji I

Dr. Eng Herman Tolle, S.T., M.T.
NIP. 197408232000121001

Penguji II

Agung Setia Budi, S.T., M.T.
NIK. 2013048704231001

Penguji III

Eriq Muhammad Adams J, S.T., M.Kom.
NIP. 198504102012121001

Mengetahui,

Ketua Program Studi Informatika / Ilmu Komputer

Drs. Marji, M.T.

NIP. 19670801 199203 1 001

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

PERNYATAAN
ORISINALITAS SKRIPSI

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah SKRIPSI ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis dikutip dalam naskah ini dan disebutkan dalam sumber kutipan dan daftar pustaka.

Apabila ternyata didalam naskah SKRIPSI ini dapat dibuktikan terdapat unsur-unsur PLAGIASI, saya bersedia SKRIPSI ini digugurkan dan gelar akademik yang telah saya peroleh (SARJANA) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku. (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 8 Juli 2015

Mahasiswa

Ahmad Nashruddin

NIM. 105060802111005

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

KATA PENGANTAR

Dengan nama Allah SWT Yang Maha Pengasih dan Penyayang. Segala puji bagi Allah SWT Tuhan semesta alam karena atas rahmat karunia dan hidayahNya-lah penulis dapat menyelesaikan Skripsi yang berjudul "Rancang Bangun Aplikasi penstatusan surat pemberitahuan kewajiban pemilik kendaraan bermotor berbasis Android". Shalawat dan salam juga tetap terlimpahkan atas junjungan besar kita Nabi Muhammad S.A.W. beserta keluarga dan para sahabat sekalian. Skripsi ini disusun untuk memenuhi sebagian persyaratan memperoleh gelar Sarjana Komputer di Program Studi Teknik Informatika Program Teknologi Informasi dan Ilmu Komputer Universitas Brawijaya Malang.

Melalui kesempatan ini, penulis ingin menyampaikan rasa hormat dan terima kasih penulis yang sebesar-besarnya kepada semua pihak yang telah memberikan bantuan – bantuan baik lahir maupun batin selama penulisan tugas akhir ini. Oleh karena itu, pada kesempatan ini penulis ingin menyampaikan rasa hormat dan terima kasih penulis kepada:

1. Bapak Aryo Pinandito, S.T., M.MT dan Agi Putra Kharisma, S.T., M.T selaku dosen pembimbing skripsi yang telah dengan sabar membimbing dan mengarahkan penulis sehingga dapat menyelesaikan skripsi
2. Bapak Satrio Gung Wicaksono, S.Kom, M.Kom. selaku dosen pembimbing akademik yang telah memberikan bimbingan, ilmu dan saran selama penulis belajar.
3. Bapak Ir. Sutrisno, M.T, Bapak Ir. Heru Nurwasito, M.Kom, Bapak Himawat Aryadita, S.T, M.Sc, dan Bapak Eddy Santoso, S.Kom selaku Ketua, Wakil Ketua 1, Wakil Ketua 2 dan Wakil Ketua 3 Program Teknologi Informasi dan Ilmu Komputer Universitas Brawijaya.
4. Bapak Drs. Marji, M.T, dan Bapak Issa Arwani, S.Kom, M.Sc selaku Ketua dan Sekretaris Program Studi Teknik Informatika Universitas Brawijaya.

5. Seluruh Dosen Teknik Informatika Universitas Brawijaya atas kesediaan membagi ilmunya kepada penulis.
6. Seluruh Civitas Akademika Teknik Informatika Universitas Brawijaya yang telah banyak memberi bantuan dan dukungan selama penulis menempuh studi di Teknik Informatika Universitas Brawijaya dan selama penyelesaian skripsi ini.
7. Ibunda Suliswati, Ayahanda Drs. Sulisto, Kakak Eny Fatmawati, S.E dan Suami Zainal Arifin, S.E, juga Keponakan tercinta Selena Nuril Putri Arifin dan Raisya Alya Putri Arifin serta seluruh keluarga besar atas segala kasih sayang, perhatian dan kesabarannya di dalam membesarkan dan mendidik penulis, serta yang senantiasa tiada henti – hentinya memberikan doa dan semangat demi terselesaikannya skripsi ini.
8. Herminah Febrianty, S.Si tercinta sekeluarga yang selalu memberikan motivasi, dukungan, semangat, dan doa selama penulis menempuh dan menyelesaikan studi di Program Teknologi Informasi dan Ilmu Komputer Universitas Brawijaya.
9. Terima kasih kepada Bagas Dimas P, S.Kom, Nehru Saputra, S.Kom, Muhammad Rizky Ramadhan, S.Kom, yang telah banyak memberikan bantuan, motivasi, dukungan, semangat, dan doa selama penulis menempuh dan menyelesaikan studi di Program Teknologi Informasi dan Ilmu Komputer Universitas Brawijaya.
10. Terima kasih juga kepada teman-teman seperjuangan Rendra Pranadipa, S.Kom, Jihad dzikri waspada, S.Kom, Dedy arief wibisono, S.Kom, Wahyu Indra, S,Kom, Wisnu tri prasetyo, S.Kom, Gozali, S.Kom, M Garisa Pradinu, S.Kom, Galuh Mazenda, S.Kom, M. Burhanuddin, S.Kom, Affan Amirullah, S.Kom, M. Faridul Qudsi, S.Kom, M. Isrofil Dwi, S.Kom, Fadhillah Aria Digdaya, S.Kom, Affan Dzaky, S.Kom, ferdy putra fajar, S.Kom, Sanches, S.Kom, Andreas yuan, S.Kom, Rio Monthe, S.Kom, M ali fahmi, S.Kom, M. Mahfud, S.Kom Teman-teman Angkatan 2009, 2010, 2011, dan 2012 Teknik Informatika atas segala bantuannya selama menempuh studi di Program Teknologi Informasi dan Ilmu Universitas Brawijaya.

11. Semua pihak yang tidak dapat penulis sebutkan satu per satu yang terlibat baik secara langsung maupun yang tidak langsung demi terselesaikannya skripsi ini.

Hanya doa yang bisa penulis berikan semoga Allah SWT memberikan pahala serta balasan kebaikan yang berlipat. Penulis menyadari bahwa skripsi ini masih banyak kekurangan dan masih jauh dari sempurna. Untuk itu, saran dan kritik yang membangun sangat penulis harapkan. Semoga skripsi ini membawa manfaat bagi penyusun maupun pihak lain yang menggunakannya.

Malang, 8 Juli 2015

Penulis

ABSTRAK

Ahmad Nashruddin. 2015. Rancang Bangun Aplikasi Penstatusan Surat Pemberitahuan Kewajiban Pemilik Kendaraan Bermotor (SPKPKB) Berbasis Android Menggunakan GPS (Global Positioning System). Program Teknologi Informasi dan Ilmu Komputer, Universitas Brawijaya, Malang. Dosen Pembimbing : Aryo Pinandito, ST., M.MT. dan Agi Putra Kharisma, ST., MT.

Proses pelaksanaan pemungutan pajak kendaraan bermotor dilaksanakan melalui rangkaian penyampaian surat pajak kendaraan bermotor yang dimulai dari penyampaian Surat Pemberitahuan Kewajiban Pajak Kendaraan Bermotor (SPKPKB). Penyampaian SPKPKB diharapkan dapat meningkatkan kepatuhan pembayaran pajak oleh wajib pajak. Namun pada kenyataannya tunggakan pajak oleh wajib pajak masih tinggi. Faktor yang mempengaruhi adalah prosedur yang panjang dalam pelaporan hasil penstatusan kendaraan. Selain itu masih terdapat ketidakpatuhan petugas lapangan dalam menyampaikan SPKPKB di lapangan. Oleh sebab itu, diperlukan sebuah solusi yang dapat meningkatkan efisiensi penyampaian SPKPKB dan menurunkan tingkat ketidak patuhan petugas. Android merupakan sistem operasi bergerak (*mobile*) yang memiliki berbagai keunggulan dan Teknologi GPS mampu melacak keberadaan suatu lokasi dengan hasil yang akurat. Dengan memanfaatkan teknologi tersebut didapatkan aplikasi baru yang berguna untuk membantu penstatusan surat dengan tingkat efisiensi lebih dari 200% dan tingkat ketidakpatuhan petugas lapangan menurun sebesar 13%.

Kata kunci : Android, GPS, Pajak kendaraan bermotor, SPKPKB

ABSTRACT

The process of collecting motor vehicle tax implemented through a series of mail delivery vehicle tax which starts from filing 'Surat Pemberitahuan Kewajiban Pajak Kendaraan Bermotor' (SPKPKB). Submission SPKPKB expected to increase compliance of tax payment by the taxpayer. But in fact tax arrears by the taxpayer is still high. Factors that influence is a lengthy procedure in reporting the results of vehicle condition. In addition there are non-compliance field officers on the delivering SPKPKB. Therefore, need a solution that can improve efficiency and reduce the disobedience of field officers. Android is a mobile operating system which has various advantages and GPS technology is able to track the whereabouts of a location with accurate results. By utilizing this technology will obtained new applications that help granting status with an efficiency rate more than 200% and non-compliance of the field officer rate decreased by 13%.

Keywords : Android, GPS, motor vehicle tax, SPKPKB

DAFTAR ISI

KATA PENGANTAR ..... **i**
ABSTRAK ..... **iv**
ABSTRACT ..... **v**
DAFTAR ISI ..... **vi**
DAFTAR GAMBAR ..... **x**
DAFTAR TABEL ..... **xii**
BAB I **1**
1.1 Latar Belakang 1
1.2 Rumusan Masalah 3
1.3 Batasan Masalah 3
1.4 Tujuan 3
1.5 Manfaat 3
1.6 Sistematika Pembahasan 4
BAB II **6**
2.1 Pajak 6
2.1.1 Pajak Kendaraan Bermotor 6
2.1.2 Rangkaian Penyampaian Surat Pajak Kendaraan Bermotor 7
1. Surat Pemberitahuan Kewajiban Pajak Kendaraan Bermotor (SPKPKB) 7
2. Surat Ketetapan Pajak Daerah (SKPD) 8
3. Surat Tagihan Pajak Daerah (STPD) 8
4. Surat Peringatan 8
5. Surat Paksa 9
BAB III **10**
3.1 Studi Literatur 10
3.2 Analisis Kebutuhan 11
3.3 Perancangan 11
3.4 Implementasi 12

3.5 Pengujian 12
3.6 Pengambilan Kesimpulan 12
BAB IV 13
4.1 Analisis Kebutuhan 14
4.1.1 Analisis Proses Bisnis 14
4.1.2 Proses Bisnis Aplikasi Android 15
4.1.3 Gambaran Umum Perangkat Lunak 17
4.1.4 Identifikasi Aktor 18
4.1.5 Analisis Data 19
4.1.6 Daftar Kebutuhan 20
4.1.7 Diagram *Use Case* 21
1. Diagram *use case* aplikasi Android 22
2. Diagram *use case* aplikasi *web* 23
4.1.8 Skenario *Use Case* 24
1. Skenario *use case* aplikasi Android 24
2. Skenario *use case* aplikasi *web* 27
4.2 Perancangan perangkat lunak 33
4.2.1 Perancangan arsitektur 34
4.2.2 Perancangan *Activity Diagram* 35
4.2.3 Perancangan *Class Diagram* 37
4.2.4 Perancangan Sequences Diagram 38
4.2.5 Perancangan Basis Data 41
1. Basis data aplikasi Android 41
2. Basis data aplikasi *web* 43
4.2.6 Perancangan Desain Antarmuka Aplikasi Android 44
BAB V IMPLEMENTASI 47
5.1 Implementasi 48
5.1.1 Spesifikasi Lingkungan Pengembangan Sistem 48
a. Spesifikasi perangkat keras 48

b. Spesifikasi perangkat lunak 48
5.1.2 Batasan-Batasan Implementasi 49
5.1.3 Implementasi Basis Data 49
5.1.4 Implementasi Class/Interface dan Layout 50
5.1.5 Implementasi Kode Pada Aplikasi Android 51
5.1.6 Implementasi Antarmuka 54
BAB VI 66
6.1 Pengujian 66
6.1.1 Pengujian Validasi 67
a. Kasus uji menambah data ekspedisi 67
b. Kasus uji menghapus data ekspedisi 69
c. Kasus uji menambah data koordinat 71
d. Kasus uji mengirim data ekspedisi ke aplikasi *web* 72
e. Kasus uji melakukan *login* 74
f. Kasus uji melakukan *logout* 75
g. Kasus uji melihat data petugas lapangan 75
h. Kasus uji membuat akun petugas lapangan 76
i. Kasus uji melihat informasi data wajib pajak 77
j. Kasus uji melihat informasi data ekspedisi 78
6.1.2 Pengujian *User Acceptance Test* 79
6.1.3 Pengujian Efisiensi Data 81
6.1.4 Pengujian Kepatuhan Petugas Lapangan 83
6.2 Analisis 86
6.2.1 Analisis Hasil Pengujian Validasi 86
6.2.2 Analisis Hasil Pengujian *User Acceptance Test* 86
6.2.3 Analisis Hasil Pengujian Efisiensi Data 90
6.2.4 Analisis Hasil Pengujian Kepatuhan Petugas Lapangan 90
BAB VI PENUTUP 92
6.1 Kesimpulan 92

6.2 Saran 93
DAFTAR PUSTAKA 94
LAMPIRAN L-1
Lampiran 1 : *Activity diagram* hapus ekspedisi aplikasi Android L-1
Lampiran 2 : kode data ekspedisi aplikasi Android L-2
Lampiran 3 : kode data detail ekspedisi aplikasi Android L-3
Lampiran 4 : kode hapus ekspedisi aplikasi Android L-4
Lampiran 5 : Implementasi Antarmuka halaman data ekspedisi aplikasi *web* .. L-5
Lampiran 6 : Implementasi Antarmuka halaman data petugas aplikasi *web* L-6
Lampiran 7 : Implementasi Antarmuka halaman buat akun aplikasi *web* L-7
Lampiran 8 : Implementasi Antarmuka halaman data wajib pajak apliksi *web* L-8
Lampiran 9 : Hasil pengisian Kuesioner L-9
Lampiran 10: Hasil pengisian Kuesioner L-10

DAFTAR GAMBAR

Gambar 3. 1 Diagram Pohon Metodologi Penelitian 10
Gambar 4. 1 Diagram Blok Perancangan 13
Gambar 4. 2 Diagram alir proses bisnis aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS 14
Gambar 4. 3 Diagram alir proses bisnis aplikasi Android 16
Gambar 4. 4 Gambaran umum perangkat lunak 18
Gambar 4. 5 *Use case* aplikasi Android 22
Gambar 4. 6 *Use case* aplikasi *web* 23
Gambar 4. 7 Arsitektur perangkat lunak 34
Gambar 4. 8 *Activity diagram* tambah ekspedisi 35
Gambar 4. 9 *Activity diagram* data ekspedisi 36
Gambar 4. 10 Pemodelan MVC kelas aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS 37
Gambar 4. 11 *Sequence diagram* tambah ekspedisi aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS 39
Gambar 4. 12 *Sequence diagram* data ekspedisi aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS 40
Gambar 4. 13 Diagram *entity relationship* aplikasi Android 41
Gambar 4. 14 Diagram *entity relationship* aplikasi *web* 43
Gambar 4. 15 Rancangan antar muka halaman utama aplikasi Android 44
Gambar 4. 16 Rancangan antar muka halaman tambah ekspedisi 45
Gambar 4. 17 Rancangan antar muka Halaman Hasil Pencarian 46
Gambar 5. 1 Diagram blok implementasi 47
Gambar 5. 3 Diagram *entity relationship* aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS 49
Gambar 5. 4 Antarmuka halaman *home* 56
Gambar 5. 5 Antarmuka halaman tambah informasi ekspedisi 59
Gambar 5. 6 Antarmuka halaman data ekspedisi 61

Gambar 5. 7 Antarmuka halaman detail data ekspedisi 62
Gambar 5. 8 Antarmuka halaman hapus data ekspedisi 64
Gambar 5. 9 Antarmuka halaman detail hapus data ekspedisi 65
Gambar 6. 1 Diagram Alir Pengujian dan Analisis 66
Gambar 6. 2 Gambar perbandingan jumlah pelaporan SPKPKB pada metode lama dan metode aplikasi 91
Gambar 6. 3 Gambar perbandingan prosentase pelaporan SPKPKB pada metode lama dan metode aplikasi 91

DAFTAR TABEL

Tabel 4. 1 Identifikasi aktor 18
Tabel 4. 2 Daftar kebutuhan fungsional aplikasi Android 20
Tabel 4. 3 Daftar kebutuhan fungsional aplikasi *web* 21
Tabel 4. 4 Skenario *use case* menambah data ekspedisi wajib pajak 24
Tabel 4. 5 Skenario *use case* melihat data ekspedisi wajib pajak 25
Tabel 4. 6 Skenario *use case* menghapus data ekspedisi wajib pajak 26
Tabel 4. 7 Skenario *use case* melakukan *login* 27
Tabel 4. 8 Skenario *use case* melakukan *logout* 28
Tabel 4. 10 Skenario *use case* membuat akun petugas lapangan 29
Tabel 4. 11 Skenario *use case* melihat data petugas lapangan 30
Tabel 4. 12 Skenario *use case* melihat data wajib pajak 31
Tabel 4. 13 Skenario *use case* melihat data ekspedisi 32
Tabel 4. 11 struktur tabel aplikasi Android entitas wajib pajak 42
Tabel 4. 12 Struktur tabel aplikasi Android entitas kendaraan 42
Tabel 4. 13 Struktur tabel petugas 43
Tabel 5. 1 Spesifikasi perangkat lunak komputer 48
Tabel 5. 2 Implementasi *class* pada kode program *.java 50
Tabel 5. 3 Implementasi *layout* 50
Tabel 6. 1 Kasus uji menambah data ekspedisi 67
Tabel 6. 2 Kasus uji menghapus data ekspedisi 69
Tabel 6. 3 Kasus uji menambah data koordinat 71
Tabel 6. 4 Kasus uji mengirim data ekspedisi ke aplikasi *web* 72
Tabel 6. 5 Kasus uji melihat data petugas lapangan 74
Tabel 6. 5 Kasus uji melihat data petugas lapangan 75
Tabel 6. 5 Kasus uji melihat data petugas lapangan 75
Tabel 6. 6 Kasus uji membuat akun petugas lapangan 76
Tabel 6. 7 Kasus uji melihat informasi data wajib pajak 77
Tabel 6. 8 Kasus uji melihat informasi data ekspedisi 78

Tabel 6. 9 Jawaban responden pengujian *user acceptance test* 79
Tabel 6. 10 Hasil pengujian *user acceptance test* 80
Tabel 6. 11 hasil pengujian efisiensi data 81
Tabel 6. 12 Data kepatuhan petugas lapangan pada metode lama 83
Tabel 6. 13 Hasil pengujian kepatuhan petugas lapangan pada metode aplikasi 85
Tabel 6. 14 Interpretasi Skor Likert 87
Tabel 6. 15 Persentase keseluruhan jawaban 87
Tabel 6. 16 Status pengujian *user acceptance test* 88
Tabel 6. 17 Status pengujian *user acceptance test* berdasarkan faktor-faktor dalam kuesioner 89

BAB I
PENDAHULUAN

1.1 Latar Belakang

Pajak merupakan kontribusi wajib pajak Negara yang terutang oleh orang pribadi atau badan yang bersifat memaksa berdasarkan Undang-Undang dengan tidak mendapatkan imbalan secara langsung dan digunakan untuk keperluan negara bagi sebesar-besarnya kemakmuran rakyat [PAN-10]. Bagi suatu Negara, pajak memegang peranan yang penting yaitu sebagai sumber penerimaan yang akan digunakan untuk membiayai kegiatan-kegiatan pemerintahan dan pembangunan serta sebagai alat regulasi. Sebagai regulasi pajak dipergunakan sebagai redistribusi pendapatan, stabilitas ekonomi, realokasi sumber- sumber ekonomi [ERL-02].

Pajak Kendaraan Bermotor (PKB) merupakan sumber penerimaan pajak daerah terbesar guna membiayai pembangunan suatu daerah [NOV-10]. Pajak Kendaraan Bermotor (PKB) yang dikelola oleh pemerintahan merupakan sumber pendapatan yang bisa dikategorikan paling besar nilainya. Hal ini dimungkinkan karena obyek pungut dari Pajak Kendaraan bermotor adalah perorangan atau instansi yang memiliki kendaraan bermotor. Dimana setiap tahunnya secara eksplisit obyek pajak selalu bertambah seiring dengan bertambahnya jumlah penduduk dan perkembangan wilayah suatu daerah [SUG-05].

Proses pelaksanaan pemungutan pajak kendaraan bermotor dilaksanakan melalui rangkaian penyampaian surat pajak kendaraan bermotor yang terdiri dari enam jenis surat pajak yang dimulai dari penyampaian Surat Pemberitahuan Kewajiban Pajak Kendaraan Bermotor (SPKPKB). Keputusan Kepala Dinas Pendapatan Daerah Provinsi Jawa Timur Nomor 843 Tahun 2012 Pasal 1, 6 menyebutkan bahwa Dinas Pendapatan Provinsi Jawa Timur berkewajiban menerbitkan SPKPKB 25 (dua puluh lima) hari sebelum berakhirnya masa pajak kendaraan bermotor baik dalam bentuk surat dan/atau media elektronik. Dalam hal penyampaian SPKPKB kepada wajib pajak kendaraan bermotor dapat disampaikan

melalui pihak ketiga yang telah ditetapkan oleh masing-masing kantor Unit Pelaksana Teknis (UPT) [YUL-13]. Dengan ini maka kantor SAMSAT memiliki petugas di lapangan yang bertugas untuk datang ke rumah wajib pajak yang terdaftar memiliki kendaraan bermotor pribadi. Hal ini bertujuan untuk menyampaikan SPKPKB guna verifikasi status dari kendaraan tersebut yaitu masih dalam status masih dimiliki, lapor jual, rusak, hilang, alamat tidak jelas atau lain-lain.

Dengan adanya penyampaian SPKPKB kepada wajib pajak diharapkan kepatuhan pembayaran pajak oleh wajib pajak. Sedangkan fenomena yang terjadi saat ini yaitu masih tingginya tunggakan pajak oleh wajib pajak meski sampai adanya surat peringatan kedua [YUL-13]. Faktor yang turut mempengaruhi meningkatnya tunggakan pajak kendaraan bermotor adalah terdapat ketidakpatuhan petugas lapangan dalam menyampaikan SPKPKB di lapangan, sehingga dapat merugikan negara. Ketidakpatuhan petugas lapangan yang antara lain tidak menyampaikan SPKPKB secara benar kepada semua wajib pajak dapat mengakibatkan data yang tersampaikan ke kantor menjadi tidak akurat dan dapat dimanipulasi. Selain itu masih terdapat prosedur yang panjang antar petugas pihak ketiga dalam melaporkan hasil penstatusan kendaraan untuk sampai ke kantor, sehingga menghabiskan waktu yang lama yang mengakibatkan masih kurangnya efisiensi dalam hal waktu.

Android merupakan sistem operasi bergerak (*mobile*) yang menggunakan versi *mobile* dari kernel linux. sistem ini memiliki berbagai keunggulan sebagai *software* berbasis kode komputer yang bisa didistribusikan secara terbuka (*open source*), sehingga *programmer* bisa membuat aplikasi baru didalamnya[COO-10]. Teknologi GPS mampu melacak keberadaan pesawat, kendaraan, kapal, laptop, ponsel, dan bahkan orang per orang. Ini semua menjadi mungkin karena teknologi GPS memiliki beberapa kemampuan memberikan data lokasi yang tepat untuk segala titik di muka bumi pada segala cuaca serta menghemat waktu dan biaya survey pemetaan dengan hasil yang tetap akurat[KUR-05]. Dengan hal ini maka diperlukan suatu aplikasi baru yang berguna untuk membantu penstatusan surat lebih efisien dan ketidakpatuhan

dalam penyampaian surat dapat dikurangi dengan memanfaatkan teknologi Android dan GPS (*Global Positioning System*).

1.2 Rumusan Masalah

Berdasarkan uraian dalam latar belakang di atas, maka dapat diambil rumusan masalah sebagai berikut:

1. Bagaimana rancangan dan implementasi aplikasi yang dapat mengefisienkan dan menurunkan tingkat ketidak patuhan petugas lapangan dalam penstatusan SPKPKB.
2. Bagaimanakah kenyamanan pengguna dalam menggunakan aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS.
3. Seberapa besar peningkatan efisiensi waktu dalam penstatusan SPKPKB menggunakan GPS berbasis Android.
4. Seberapa besar penurunan tingkat ketidakpatuhan petugas sesudah menggunakan aplikasi yang dibangun.

1.3 Batasan Masalah

Berdasarkan latar belakang dan rumusan masalah yang telah dikemukakan, penelitian ini mempunyai batasan-batasan masalah, yaitu sumber data yang digunakan adalah sumber data yang diperoleh dari kantor Dinas Pendapatan Daerah Kabupaten Malang.

1.4 Tujuan

Untuk merancang dan membuat aplikasi penstatusan SPKPKB yang dapat meningkatkan efisiensi waktu dan mengurangi ketidakpatuhan petugas lapangan dengan memanfaatkan teknologi Android menggunakan GPS.

1.5 Manfaat

Dengan adanya penulisan ini diharapkan memberikan nilai dan hasil yang berguna bagi semua pihak, adapun manfaat dari penulisan ini dapat dijabarkan berikut:

a) Menyediakan aplikasi yang dapat digunakan untuk menstatus SPKPKB dengan menggunakan teknologi GPS berbasis Android pada kantor bersama SAMSAT.
b) Memberikan gambaran kepada bagian Administrasi kantor bersama SAMSAT dalam perbaikan penstatusan SPKPKB sebagai salah satu pelayanan publik yang baik dan efektif kepada masyarakat.
c) Memberikan informasi kepada bagian Administrasi kantor bersama SAMSAT tentang lokasi wajib pajak.
d) Memberikan informasi kepada bagian pencetakan surat kantor bersama SAMSAT tentang jumlah surat SPKPKB yang dicetak selama satu bulan.
e) Memberikan informasi kepada bagian pencetakan surat kantor bersama SAMSAT tentang jumlah surat SPKPKB yang di laporkan kembali dalam dalam satu bulan.
f) Untuk menekan biaya dan mempermudah tugas petugas lapangan kantor bersama SAMSAT dalam penstatusan SPKPKB.
g) Hasil penelitian dapat di jadikan sebagai bahan masukan oleh kantor bersama SAMSAT untuk meningkatkan peneriman Pajak Kendaraan Bermotor dimasa mendatang.

1.6 Sistematika Pembahasan

Sistematika pembahasan ditunjukkan untuk memberikan gambaran dan uraian dari penulisan skripsi ini secara garis besar yang meliputi beberapa bab, sebagai berikut:

Bab I : Pendahuluan

Menguraikan tentang latar belakang, rumusan masalah, tujuan, manfaat dan sistematika pembahasan.

Bab II : Kajian Pustaka dan Dasar Teori

Menguraikan tentang dasar teori dan referensi yang mendasari pengembangan aplikasi penstatusan SPKPKB berbasis Android mengunakan GPS.

Bab III : Metode Penelitian

Menguraikan dan membahas langkah kerja yang dilakukan dalam penulisan skripsi yang terdiri dari studi literatur, analisis, perancangan, implementasi, pengujian dan pengambilan kesimpulan sesuai dengan dasar teori yang ada.

Bab IV : Analisis dan Perancangan

Menguraikan hasil analisis kebutuhan dan langkah perancangan terhadap perangkat lunak yang terdiri dari analisis proses bisnis, gambaran umum perangkat lunak, identifikasi aktor, daftar kebutuhan, pemodelan *use case* dan pemodelan skenario diagram kemudian dilanjutkan dengan tahap perancangan yang terdiri perancangan arsitektur, perancangan basis data, perancangan *activity diagram* dan *class diagram*.

Bab V : Implementasi dan Pengujian

Membahas implementasi perangkat lunak sesuai dengan perancangan perangkat lunak yang telah dibuat dan memuat hasil pengujian terhadap perangkat lunak yang telah direalisasikan.

Bab VI : Kesimpulan dan Saran

Memuat kesimpulan yang diperoleh dari pembuatan dan pengujian perangkat lunak yang dikembangkan dalam skripsi ini serta saran-saran untuk pengembangan lebih lanjut.

BAB II

KAJIAN PUSTAKA DAN DASAR TEORI

2.1 Pajak

Pajak merupakan kontribusi wajib pajak Negara yang terutang oleh orang pribadi atau badan yang bersifat memaksa berdasarkan Undang-Undang dengan tidak mendapatkan imbalan secara langsung dan digunakan untuk keperluan negara bagi sebesar-besarnya kemakmuran rakyat [PAN-10]. Bagi suatu Negara, pajak memegang peranan yang penting yaitu sebagai sumber penerimaan yang akan digunakan untuk membiayai kegiatan-kegiatan pemerintahan dan pembangunan serta sebagai alat regulasi. Sebagai regulasi pajak dipergunakan sebagai redistribusi pendapatan, stabilitas ekonomi, realokasi sumber- sumber ekonomi [ERL-02].

Salah satu fungsi penerimaan pajak adalah sebagai fungsi *budgetair* [PAN-10]. Fungsi budgetair disebut pula sebagai fungsi utama pajak atau fungsi fiskal yaitu suatu fungsi dimana pajak dipergunakan sebagai alat untuk memasukkan dana secara optimal ke kas negara berdasarkan undang-undang perpajakan yang berlaku. Fungsi ini disebut fungsi utama karena fungsi inilah yang secara historis pertama kali timbul. Berdasarkan fungsi ini, pemerintah yang membutuhkan dana untuk membiayai berbagai kepentingan negara dengan memungut pajak dari penduduknya [NUR-05].

Kontribusi masyarakat melalui pembayaran pajak dimanfaatkan oleh pemerintah untuk melakukan pekerjaan-pekerjaan berupa pelayanan untuk umum, membiayai pendidikan, memperbaiki fasilitas kesehatan, perumahan, air minum, listrik, transportasi, fasilitas keamanan dan banyak lain yang ditunjukkan untuk kemakmuran dan kesejahteraan masyarakat dan negara [RIM-04].

2.1.1 Pajak Kendaraan Bermotor

Kendaraan bermotor adalah semua kendaraan beroda dua atau lebih beserta gandengannya yang digunakan di semua jenis jalan darat dan digerakkan oleh peralatan teknik berupa motor atau peralatan lainnya yang berfungsi untuk mengubah suatu

sumber energi tertentu menjadi tenaga gerak kendaraan, termasuk alat-alat berat, dan alat-alat berat bergerak.

Berdasarkan Undang-Undang 28 Tahun 2009, Pajak kendaraan bermotor adalah pajak atas kepemilikan atau penguasaan kendaraan bermotor sehingga secara otomatis yang menjadi objek dari pajak kendaraan bermotor adalah kepemilikan atau penguasaan kendaraan bermotor, tetapi atas pajak ini terdapat juga beberapa jenis objek yang dikecualikan dari pengenaan pajak, yaitu kepemilikan atau penguasaan kendaraan bermotor oleh:

1. Pemerintah pusat dan pemerintah daerah;
2. Kedutaan, konsulat, perwakilan negara asing, dan perwakilan lembaga-lembaga internasional berdasar asas timbal balik;
3. Subjek pajak lain yang ditetapkan dengan peraturan daerah.

Yang menjadi subjek pajaknya adalah orang atau badan yang memiliki ataupun menguasai, sekaligus menjadi wajib pajak. Dasar pengenaan dihitung sebagai perkalian dua unsur pokok, yakni nilai jual dan bobot.

2.1.2 Rangkaian Penyampaian Surat Pajak Kendaraan Bermotor

Proses pelaksanaan pemungutan pajak kendaraan bermotor dilaksanakan melalui rangkaian penyampaian surat pajak kendaraan bermotor yang terdiri dari enam jenis surat pajak, antara lain:

1. Surat Pemberitahuan Kewajiban Pajak Kendaraan Bermotor (SPKPKB)

Keputusan Kepala Dinas Pendapatan Daerah Provinsi Jawa Timur Nomor 843 Tahun 2012 Pasal 1, menyebutkan bahwa Dinas Pendapatan Provinsi Jawa Timur berkewajiban menerbitkan SPKPKB 25 (dua puluh lima) hari sebelum berakhirnya masa pajak kendaraan bermotor baik dalam bentuk surat dan/atau media elektronik. Dalam hal penyampaian SPKPKB kepada wajib pajak kendaraan bermotor dapat disampaikan melalui pihak ketiga yang telah ditetapkan oleh masing-masing kantor Unit Pelaksana Teknis (UPT).

2. Surat Ketetapan Pajak Daerah (SKPD)

Keputusan Kepala Dinas Pendapatan Daerah Provinsi Jawa Timur Nomor 843 Tahun 2012 Pasal 2 ayat 1, menyebutkan bahwa Dinas Pendapatan Daerah Provinsi Jawa Timur berkewajiban menerbitkan SKPD terhadap objek pajak kendaraan bermotor yang tidak didaftar ulang (penul) dengan penerbitan 1 (satu) hari setelah masa pajak kendaraan bermotor berakhir (H+1), namun belum kadaluwarsa dengan status kendaraan masih dimiliki.Peraturan Gubernur Provinsi Jawa Timur Nomor 9 Tahun 2010 Pasal 12, menyebutkan bahwa piutang pajak kendaraan bermotor mulai dikategorikan sebagai tunggakan pajak kendaraan sejak diterbitkannya Surat Ketetapan Pajak Daerah (SKPD).

3. Surat Tagihan Pajak Daerah (STPD)

Sesuai dengan Undang-Undang Nomor 28 Tahun 2009 Pasal 100 ayat 3 Perda Nomor 9 Tahun 2009 Pasal 58 ayat 1 serta Pergub Nomor 9 Tahun 2010 Pasal 12, STPD diterbitkan 30 (tiga puluh hari) setelah diterbitkan SKPD dalam hal keterlambatan pembayaran, tidak atau kurang bayar maka wajib pajak kendaraaan bermotor dikenakan sanksi administrasi sebesar 2% setiap bulan dari nominal pajak terutang untuk jangka waktu maksimal 15 (lima belas) bulan sejak Surat Tagihan Pajak Daerah diterbitkan dengan status objek pajak masih dimiliki oleh wajib pajak bersangkutan.

4. Surat Peringatan

a. Surat Peringatan Pertama (SP 1)

Perda Nomor 9 Tahun 2009 Pasal 58 ayat 2, menetapkan bahwa Surat Peringatan Pertama diterbitkan 14 (empat belas) hari setelah Surat Tagihan Pajak Kendaraan (STPD) diterbitkan.

b. Surat Peringatan Kedua (SP 2)

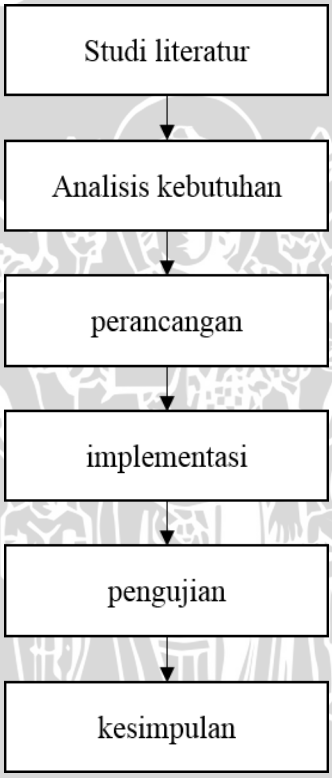
Perda Nomor 9 Tahun 2009 Pasal 58 ayat 3, menetapkan bahwa Surat Peringatan Kedua diterbitkan setelah 21 (dua puluh satu) hari setelah Surat Peringatan Pertama (SP 1) tidak dibayar.

5. Surat Paksa

Undang-Undang 28 Tahun 2009 Pasal 102 ayat 1, Perda Nomor 9 Tahun 2009 Pasal 58 ayat 4, menetapkan pajak yang terutang berdasarkan SKPD, STPD, SKPDKB, SKPDKBT, Surat Keputusan Pembetulan, Surat Keputusan Keberatan dan Putusan Banding yang tidak dibayar atau kurang bayar oleh wajib pajak kendaraan bermotor pada waktunya dapat ditagih dengan Surat Paksa [NOV-10].

BAB III
METODOLOGI PENELITIAN

Pada bab ini dijelaskan mengenai prosedur-prosedur dan kegiatan-kegiatan yang akan dilakukan dalam pengerjaan skripsi, yaitu studi literatur, analisis kebutuhan dan perancangan, implementasi, pengujian dan kesimpulan dalam pembuatan sistem aplikasi yang akan dibuat.



Gambar 3. 1 Diagram Pohon Metodologi Penelitian

3.1 Studi Literatur

Studi literatur merupakan penelusuran literatur yang bertujuan dalam menyusun dasar teori yang digunakan untuk menunjang skripsi. Penelusuran literatur

dapat bersumber dari buku, media, pakar maupun hasil penelitian orang lain. Teori-teori pendukung tersebut meliputi:

1. Pajak
   a. Pajak Kendaraan Bermotor
   b. Rangkaian Penyampaian Surat Pajak Kendaraan Bermotor

3.2 Analisis Kebutuhan

Proses analisis kebutuhan ini dilakukan dengan mengidentifikasi semua kebutuhan (*requirement*) perangkat lunak dan siapa saja yang terlibat didalamnya. Dimulai dari penjabaran analisis proses bisnis, gambaran umum perangkat lunak, identifikasi aktor, analisis data, daftar kebutuhan yang meliputi kebutuhan fungsional aplikasi Android dan aplikasi *web* kemudian dimodelkan dalam bentuk *use case diagram*. *Use case diagram* digunakan untuk mendeskripsikan kebutuhan-kebutuhan fungsional perangkat lunak. Tiap *use case* yang ada dalam diagram tersebut juga akan dijelaskan lebih rinci di dalam skenario *use case*.

3.3 Perancangan

Perancangan perangkat lunak dilakukan setelah semua kebutuhan perangkat lunak didapatkan melalui tahap analisis kebutuhan. Selama tahap ini komponen-komponen yang telah ditentukan akan dimodelkan ke dalam bentuk perancangan arsitektural untuk menggambarkan alur kerja perangkat lunak secara keseluruhan, perancangan basis data untuk menggambarkan sistem penyimpanan data, perancangan *activity diagram* untuk menggambarkan urutan proses aktivitas dari setiap *use case*, perancangan *class diagram* untuk menggambarkan perancangan struktur *class-class* yang menyusun perangkat lunak dan perancangan *sequence diagram* untuk menunjukkan aliran jalannya proses interaksi antar obyek atau *class* yang disusun berdasarkan urutan waktu.

3.4 Implementasi

Implementasi perangkat lunak mengacu kepada perancangan perangkat lunak. Implementasi perangkat lunak diawali dengan penjabaran spesifikasi lingkungan perancangan sistem, batasan-batasan implementasi, implementasi basis data menggunakan SQLite dan MySQL, implementasi dengan komponen dilakukan menggunakan bahasa pemrograman Java pada subsistem aplikasi Android dan Bahasa pemrograman HTML, CSS dan PHP untuk subsistem aplikasi *web* dan implementasi antar muka berdasarkan perancangan yang telah dilakukan.

3.5 Pengujian

Pengujian perangkat lunak dilakukan untuk mengetahui apakah kinerja sistem telah memenuhi spesifikasi kebutuhan yang melandasinya. Bagian ini diawali dengan penjelasan tentang tujuan pengujian dilanjutkan dengan pembahasan mengenai kasus uji pada pengujian validasi (*validation testing*), pengujian *user acceptance testing*, pengujian efisiensi data, serta pengujian ketidakpatuhan petugas lapangan.

Analisis juga dilakukan untuk mengetahui hasil dari pengujian perangkat lunak sehingga dapat didapatkan kesimpulan dari pengembangan perangkat lunak yang telah dilakukan.

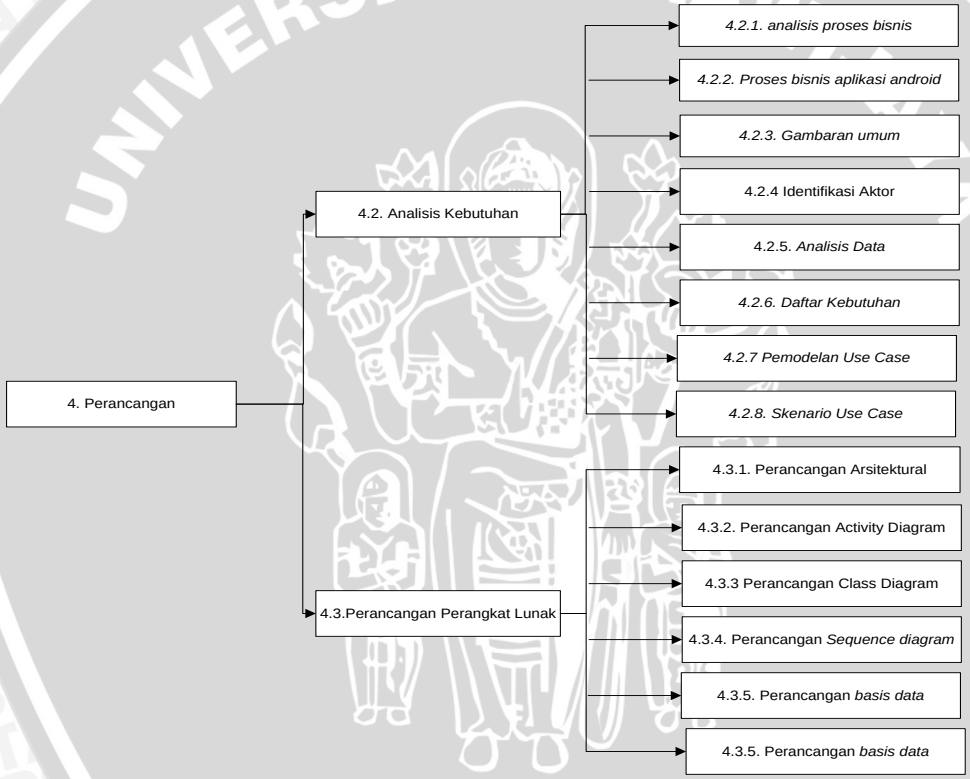
3.6 Pengambilan Kesimpulan

Pengambilan kesimpulan dilakukan setelah semua tahapan perancangan, implementasi dan pengujian sistem aplikasi telah selesai dilakukan dan didasarkan pada kesesuaian antara teori dan praktik. Kesimpulan diambil untuk menjawab rumusan masalah yang telah ditetapkan sebelumnya. Tahap terakhir dari penulisan adalah saran yang dimaksudkan untuk memperbaiki kesalahan-kesalahan yang terjadi dan menyempurnakan penulisan serta untuk memberikan pertimbangan atas pengembangan aplikasi selanjutnya.

BAB IV

ANALISIS DAN PERANCANGAN

Pada bab ini dijelaskan tentang analisis dan perancangan dari perangkat lunak. Analisis dan perancangan yang dilakukan terdiri dari dua tahap. Tahap pertama adalah proses analisis kebutuhan perangkat lunak dan dilanjutkan dengan tahap kedua berupa proses perancangan perangkat lunak. Diagram blok perancangan digambarkan pada Gambar 4. 1.



Gambar 4. 1 Diagram Blok Perancangan

Pada Gambar 4. 1 menjelaskan pada bagian perancangan terdiri dari 3 tahap utama yaitu analisi proses bisnis, analisis kebutuhan perangkat lunak terdiri dari enam tahap yaitu gambaran umum perangkat lunak, identifikasi aktor, analisis data, daftar kebutuhan, pemodelan daftar kebutuhan terhadap diagram *use case*, dan skenario

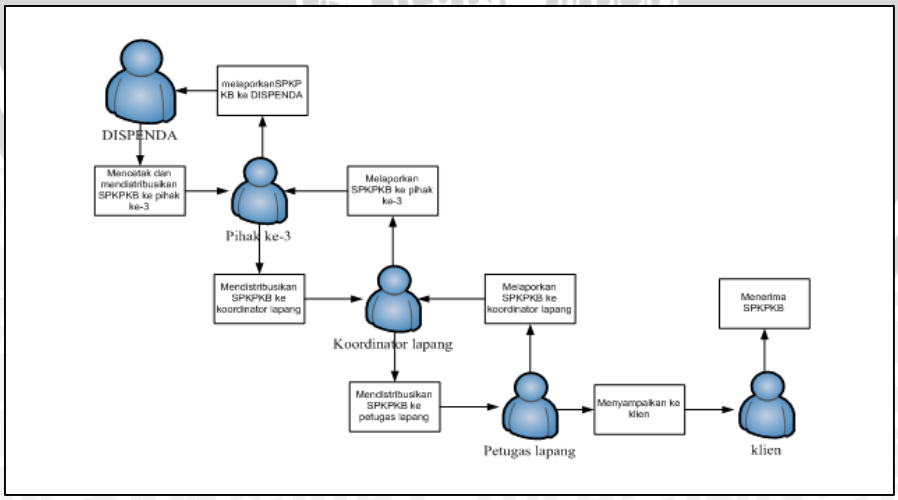
diagram *use case*. Sedangkan perancangan perangkat lunak terdiri dari lima tahap yaitu perancangan arsitektural, perancangan basis data, perancangan *activity diagram*, perancangan *class diagram* dan perancangan *sequence diagram.*

4.1 Analisis Kebutuhan

Proses analisis kebutuhan ini diawali dengan penjabaran proses bisnis, proses bisnis aplikasi Android, gambaran umum perangkat lunak, identifikasi aktor, analisis data yang akan digunakan, penjabaran tentang daftar kebutuhan. Proses analisis kebutuhan mengacu pada proses bisnis yang kemudian akan memodelkan ke dalam diagram *use case*. Analisis kebutuhan ini bertujuan untuk menggambarkan kebutuhan-kebutuhan yang harus disediakan oleh perangkat lunak agar dapat memenuhi kebutuhan pengguna.

4.1.1 Analisis Proses Bisnis

Analisis proses bisnis merupakan proses bisnis awal yang diterapkan oleh kantor Dinas Pendapatan Daerah Kabupaten Malang sebelum implementasi aplikasi. Proses ini belum melibatkan keterkaitan aplikasi/sistem khusus yang digunakan kantor Dinas Pendapatan Daerah Kabupaten Malang dalam kegiatannya. Gambar 4. 2 merupakan diagram alir yang menunjukkan proses bisnis penyampaian SPKPKB ke wajib pajak sebelum menggunakan aplikasi.

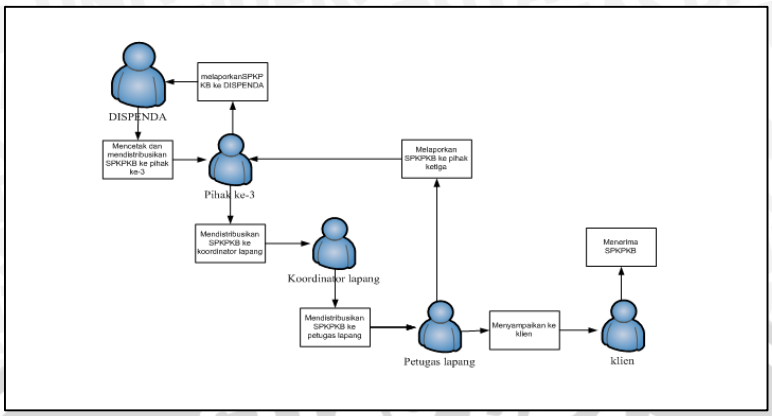


Gambar 4. 2 Diagram alir proses bisnis penstatusan SPKPKB

Diagram alir proses bisnis menjelaskan proses distribusi SPKPKB dari DISPENDA ke wajib pajak secara manual. Dinas Pendapatan Daerah Kabupaten Malang akan mencetak SPKPKB untuk di sampaikan kepada wajib pajak. DISPENDA menggunakan jasa pihak ketiga untuk mendistribusikan SPKPKB ke wajib pajak . Pihak ketiga adalah warga sipil non PNS yang mempunyai CV sendiri dan mempunyai NPWP. Proses pencetakan sampai pendistribusian ke pihak ketiga kurang lebih memakan waktu 3 hari. Pihak ketiga mempunyai 2 tingkatan petugas yang ada di bawahnya, yaitu koordinator lapangan dan petugas lapangan. Dari pihak ketiga, SPKPKB akan didistibusikan ke koordinator lapangan dan dari koordinator lapangan akan didistribusikan lagi ke petugas lapangan. Proses pendistribusian dari pihak ketiga sampai ke petugas lapang memakan waktu 2 hari. Petugas lapangan inilah yang bertugas untuk menyampaikan SPKPKB ke wajib pajak. Proses penyampaian SPKPKB ke wajib pajak dijadwalkan selama 1 minggu. Setelah SPKPKB disampaikan ke wajib pajak, petugas lapangan akan melaporkan data yang diperoleh dari wajib pajak ke koordinator lapangan. Data yang disampaikan meliputi nama wajib pajak, alamat wajib pajak, nomor polisi, status kendaraan, tanggal cetak dan tanggal terima. Kemudian koordinator lapangan akan melaporkan data tersebut ke pihak ketiga. Proses pelaporan SPKPKB kurang lebih memakan waktu 2 hari. Data akan diolah oleh pihak ketiga kemudian akan dilaporkan ke DISPENDA sebagai laporan pertanggung jawaban bulanan. Proses pengolahan data kurang lebih memakan waktu 2 hari.

4.1.2 Proses Bisnis Aplikasi Android

Proses bisnis aplikasi Android merupakan proses bisnis hasil dari pengembangan proses bisnis awal yang diterapkan oleh kantor Dinas Pendapatan Daerah Kabupaten Malang dengan memanfaatkan penggunaan teknologi Android dan GPS. Gambar 4. 3 merupakan diagram alir yang menunjukkan proses bisnis aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS.



Gambar 4. 3 Diagram alir proses bisnis aplikasi Android

Diagram alir proses bisnis menjelaskan proses distribusi SPKPKB dari DISPENDA ke wajib pajak dengan menggunakan aplikasi Android dan GPS. Dinas Pendapatan Daerah Kabupaten Malang akan mencetak SPKPKB untuk di sampaikan kepada wajib pajak.setelah SPKPKB dicetak selanjutnya akan didistribusikan ke pihak ketiga. Proses pencetakan sampai pendistribusian ke pihak ketiga kurang lebih memakan waktu 3 hari. Pihak ketiga mempunyai 2 tingkatan petugas yang ada di bawahnya, yaitu koordinator lapangan dan petugas lapangan. Dari pihak ketiga, SPKPKB akan didistibusikan ke koordinator lapangan dan dari koordinator lapangan akan didistribusikan lagi ke petugas lapangan. Proses pendistribusian dari pihak ketiga sampai ke petugas lapangan memakan waktu 2 hari. Petugas lapangan inilah yang bertugas untuk menyampaikan SPKPKB ke wajib pajak. Setelah SPKPKB disampaikan ke wajib pajak, petugas lapangan akan segera melaporkan data yang diperoleh dari wajib pajak ke bagian administrasi pihak ketiga untuk dibuat laporan. Pada bagian inilah perbedaan proses bisnis sebelum menggunakan teknologi aplikasi dan sesudah menggunakan teknologi aplikasi. Pada proses bisnis menggunakan aplikasi tidak perlu menunggu sampai 1 minggu untuk melaporkan SPKPKB yang telah disampaikan ke wajib pajak. Petugas lapangan juga tidak perlu melaporkan SPKPKB melalui koordinator lapangan melainkan dapat langsung melaporkan ke bagian administrasi pihak ketiga. Data yang disampaikan meliputi nomor polisi wajib pajak, alamat wajib pajak, status kendaraan, tanggal cetak, tanggal terima, *longitude*

dan *latitude*. Data akan diolah oleh pihak ketiga kemudian akan dilaporkan ke DISPENDA sebagai laporan pertanggung jawaban bulanan. Proses pengolahan data dan penyampaian ke DISPENDA kurang lebih memakan waktu 2 hari.

4.1.3 Gambaran Umum Perangkat Lunak

Perangkat lunak yang akan dikembangkan dalam penelitian ini adalah aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS. Aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS adalah aplikasi yang dapat digunakan untuk memberikan status kepada SPKPKB tentang status keadaan kendaraan wajib pajak dengan menggunakan *mobile phone* bersistem operasi Android sehingga akan lebih efisien dalam pelaporan SPKPKB. Status kendaraan SPKPKB disini meliputi masih dimiliki, lapor jual, rusak, hilang, alamat tidak jelas dan lainnya. Untuk mengurangi ketidakpatuhan petugas lapangan dalam menyampaikan SPKPKB ke wajib pajak maka digunakan GPS untuk mendeteksi koordinat lokasi alamat wajib pajak sehingga petugas lapangan harus menyampaikan SPKPKB kepada wajib pajak. Aplikasi ini diharapkan dapat memudahkan pengguna dalam memperoleh data tentang status kendaraan wajib pajak terbaru sehingga akan lebih efisien dan dapat mengurangi ketidakpatuhan petugas lapangan dalam penyampaian surat ke wajib pajak .

Perangkat lunak yang dikembangkan dalam skripsi ini memiliki dua bagian utama, yaitu subsistem aplikasi Android dan subsistem aplikasi *web*.

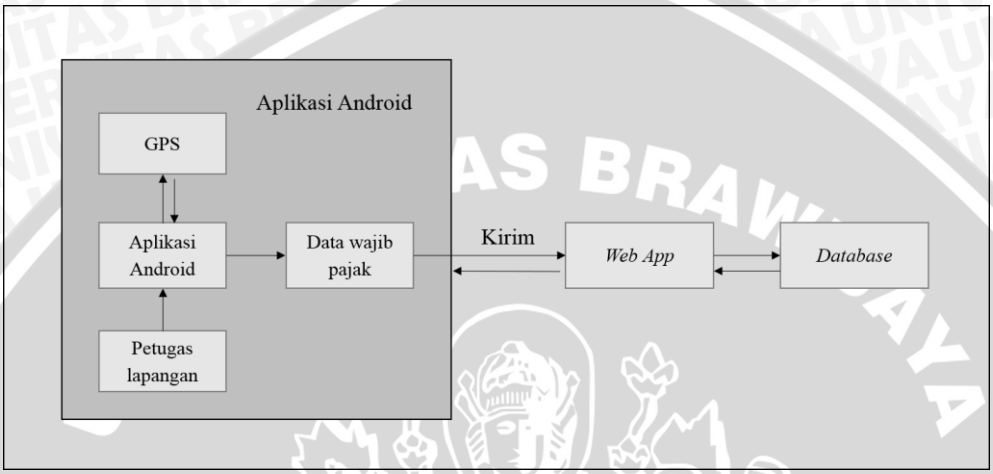
a. Aplikasi Android

Aplikasi Android adalah aplikasi berbasis Android menggunakan GPS yang dapat digunakan untuk mendata status kendaraan wajib pajakyang akan berakhir pada setiap tahun dan mencari koordinat lokasi alamat wajib pajak sesuai dengan alamat yang tertera pada SPKPKB dikirimkan ke aplikasi *web* kantor Dinas Pendapatan Daerah Kabupaten Malang.

b. Aplikasi *Web*

Aplikasi *web* dari aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS adalah aplikasi *web* yang digunakan untuk melihat dan memantau

data wajib pajak yang diperoleh dari aplikasi Android. Pada aplikasi ini, data wajib pajak dikelola oleh petugas lapangan melalui aplikasi Android. Selanjutnya data tersebut dikirimkan ke aplikasi *web* kantor Dinas Pendapatan Daerah Kabupaten Malang.



Gambar 4. 4 Gambaran umum perangkat lunak

4.1.4 Identifikasi Aktor

Tahap ini adalah tahap untuk melakukan identifikasi terhadap aktor-aktor yang berinteraksi dengan aplikasi penstatusan berbasis Android menggunakan GPS. Tabel 4. 1 memperlihatkan aktor-aktor yang terlibat beserta penjelasannya masing-masing.

Tabel 4. 1 Identifikasi aktor

Aktor	Deskripsi
Petugas lapangan	Petugas lapangan adalah orang yang bertugas di lapangan untuk menyampaikan SPKPKB ke wajib pajak. Petugas lapangan bertanggung jawab atas penyampaian SPKPKB kepada wajib pajak. Petugas lapangan berada dibawah pihak ketiga di kantor Dinas Pendapatan Daerah Kabupaten Malang. Petugas lapangan adalah orang yang pantas untuk menggunakan aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS.

Administrator	*Administrator* adalah petugas administrasi di kantor Dinas Pendapatan Daerah Kabupaten Malang yang bertugas untuk melihat atau memantau data yang masuk ke aplikasi *web* dari rangkaian aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS.

4.1.5 Analisis Data

Analisis data mengacu pada proses bisnis penyampaian SPKPKB ke wajib pajak. Analisis data bertujuan untuk mendapatkan struktur penyimpanan data yang dibutuhkan aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS. Struktur penyimpanan data pada aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS disusun berdasarkan analisis data sebagai berikut :

a. Aplikasi Android

Pada aplikasi Android data diambil sesuai dengan data yang tertulis pada SPKPKB yang meliputi identitas wajib pajak dan kendaraan wajib pajak. Pengambilan data untuk aplikasi Android berdasarkan SPKPKB bertujuan untuk memudahkan dan menyamakan data yang dibutuhkan oleh Dinas Pendapatan Daerah dalam pengolahan data lebih lanjut. Selain itu juga ditambahkan beberapa data tambahan sesuai kebutuhan aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS. Berikut data yang dibutuhkan dalam aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS:

- Data wajib pajak yang terdiri dari *id* wajib pajak, nama wajib pajak, alamat wajib pajak , *Longitude* dan *Latitude*.
- Data kendaraan yang terdiri dari *id_kendaraan*, *id* wajib pajak , nopol, status _kendaraan, jenis_kendaraan, tanggal_cetak, tanggal_terima.

b. Aplikasi *Web*

Data pada aplikasi *web* diambil dari identitas petugas lapangan yang bertujuan untuk mengetahui identitas petugas lapangan dan mendata petugas lapangan baru.

- Data petugas lapangan yang terdiri dari *id_petugas lapangan*, nama_petugas lapangan, alamat, *username* dan *password*.

4.1.6 Daftar Kebutuhan

Daftar kebutuhan mengacu pada proses bisnis penyampaian SPKPKB ke wajib pajak. Daftar kebutuhan terdiri dari kebutuhan fungsional. Daftar kebutuhan fungsional menunjukkan kebutuhan yang harus disediakan oleh aplikasi dan menunjukkan nama *use case* yang akan menyediakan fungsionalitas masing-masing kebutuhan tersebut. Kebutuhan fungsional didapatkan dengan cara memodifikasi metode penyampaian SPKPKB secara manual yang saat ini diterapkan DISPENDA menjadi metode penyampaian SPKPKB berbasis Android. Setiap kebutuhan fungsional nantinya akan diujikan terlebih dahulu sebelum dipublikasikan. Penomoran daftar kebutuhan ditulis dengan *id* KF. Pada Tabel 4. 2 menunjukkan kebutuhan fungsional pada aplikasi Android.

Tabel 4. 2 Daftar kebutuhan fungsional aplikasi Android

ID	Kebutuhan	*Use Case*
KF_1_01	Perangkat lunak harus menyediakan fasilitas untuk menambah informasi dari data wajib pajak	Menambah ekspedisi data wajib pajak
KF_1_02	Perangkat lunak harus menyediakan fasilitas untuk menghapus informasi dari data wajib pajak	Menghapus ekspedisi data wajib pajak
KF_1_03	Perangkat lunak harus menyediakan fasilitas untuk mencari letak koordinat alamat wajib pajak	Mencari koordinat alamat wajib pajak
KF_1_04	Perangkat lunak harus menyediakan fasilitas untuk mengirim informasi data wajib pajak	Mengirim ekspedisi data wajib pajak

Pada Tabel 4. 3 kebutuhan fungsional pada aplikasi *web* ditunjukkan dengan penomoran KF.

Tabel 4. 3 Daftar kebutuhan fungsional aplikasi *web*

ID	Kebutuhan	*Use Case*
KF_2_01	Aplikasi *web* harus menyediakan fasilitas untuk *login administrator*	Melakukan login
KF_2_02	Aplikasi *web* harus menyediakan fasilitas untuk *logout administrator*	Melakukan logout
KF_2_03	Aplikasi *web* harus menyediakan fasilitas untuk membuat akun petugas lapangan	Membuat akun petugas lapangan
KF_2_04	Aplikasi *web* harus menyediakan fasilitas untuk melihat semua informasi data petugas	Melihat informasi data petugas lapangan
KF_2_05	Aplikasi *web* harus menyediakan fasilitas untuk melihat semua informasi data wajib pajak	Melihat informasi data wajib pajak
KF_2_06	Aplikasi *web* harus menyediakan fasilitas untuk menampilkan semua data ekspedisi aplikasi Android	Melihat data ekspedisi

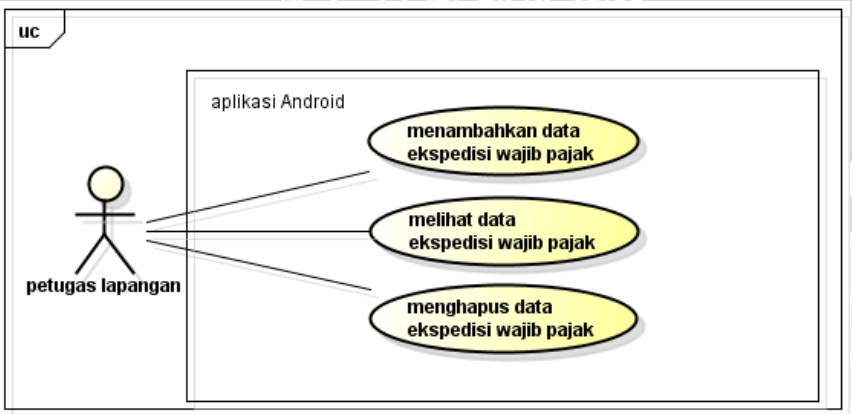
4.1.7 Diagram *Use Case*

Diagram *use case* merupakan salah satu diagram untuk memodelkan aspek perilaku perangkat lunak. Masing-masing diagram *use case* menunjukkan sekumpulan *use case*, aktor, dan hubungannya. Diagram *use case* sangat penting untuk memvisualisasikan, menspesifikasikan, dan mendokumentasikan kebutuhan perilaku perangkat lunak. Diagram *use case* juga dapat digunakan untuk mendeskripsikan interaksi antara pengguna perangkat lunak dengan perangkat lunak itu sendiri. Pemodelan diagram *use case* yang menggambarkan fungsionalitas aplikasi penstatusan

SPKPKB berbasis Android menggunakan GPS dibagi menjadi dua, yaitu diagram *use case* aplikasi Android dan diagram *use case web sevice.*

1. Diagram *use case* aplikasi Android

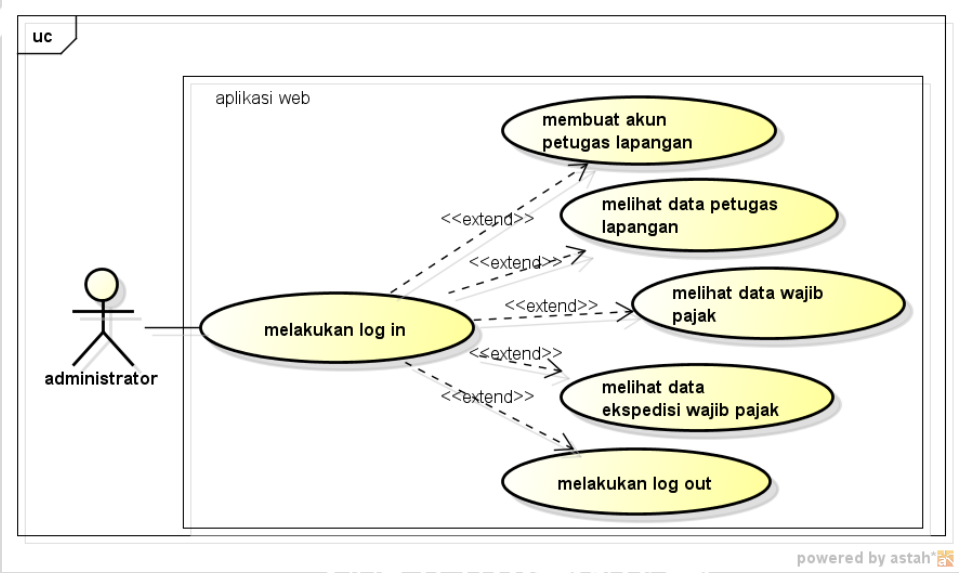
Diagram *use case* ini melibatkan petugas lapangan sebagai aktor dan 3 buah *use case* yang termasuk dalam aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS. *use case* ini juga disertai dengan skenario *use case* untuk menjelaskan rangkaian aktivitas yang terjadi di *use case* tersebut. Gambar 4. 5 merupakan *use case* dari aplikasi Android yang terdiri dari 1 aktor dan 3 buah *use case*. Pada aplikasi Android terdapat 3 fitur utama yaitu menambah data ekspedisi wajib pajak, melihat data ekspedisi wajib pajak dan menghapus data ekspedisi wajib pajak. Menambah data ekspedisi wajib pajak berfungsi untuk menambah data wajib pajak dengan cara mengetikkan informasi tentang wajib pajak pada *form* yang telah tersedia pada fitur menambah data ekspedisi wajib pajak. Setelah data selesai di masukkan, petugas lapangan dapat menambahkan koordinat *longitude* dan *latitude* dengan cara menekan tombol cari koordinat dan menekan tombol kirim ekspedisi untuk mengirimkan data ke aplikasi *web*. Melihat data ekspedisi wajib pajak berfungsi untuk melihat data yang tersimpan di *database* aplikasi Android dan menghapus data ekspedisi wajib pajak berfungsi untuk menghapus data ekspedisi yang tersimpan pada *database* aplikasi Android. Setiap *use case* yang ada pada Gambar 4. 5 akan dijelaskan oleh skenario *use case.*



Gambar 4. 5 *Use case* aplikasi Android

2. Diagram *use case* aplikasi *web*

Diagram *use case* ini melibatkan *administrator* sebagai aktor dan 4 buah *use case*. Gambar 4. 6 merupakan *use case* dari aplikasi *web* dengan *administrator* sebagai aktor dan 6 buah *use case* yaitu melakukan *login* untuk *administrator*, melakukan *logout* untuk *administrator*, membuat akun petugas lapangan, melihat data petugas lapangan, melihat data wajib pajak dan melihat data ekspedisi. *Administrator* mengakses aplikasi *web* untuk dapat melihat dan memantau data yang didapat dari aplikasi Android.



Gambar 4. 6 *Use case* aplikasi *web*

Pada aplikasi *web* terdapat 6 fitur utama yaitu melakukan *login*, membuat akun petugas lapangan untuk membuat akun petugas lapangan, melihat data petugas lapangan untuk melihat informasi tentang petugas lapangan yang terdaftar dalam aplikasi *web*, melihat data wajib pajak untuk melihat informasi tentang data wajib pajak dan melihat data ekspedisi untuk melihat semua data ekspedisi yang dikirimkan oleh aplikasi Android dan melakukan *logout* untuk *administrator*. Setiap *use case* yang ada pada Gambar 4. 6 akan dijelaskan oleh skenario *use case*.

4.1.8 Skenario *Use Case*

Skenario *use case* akan menjelaskan rangkaian aktivitas yang terjadi didalam *use case*. Skenario *use case* yang akan dijelaskan terdiri dari skenario *use case* aplikasi Android dan skenario *use case* aplikasi *web*. Pada aplikasi Android memiliki 3 skenario *use case* sedangkan aplikasi *web* memiliki 6 skenario *use case*.

1. Skenario *use case* aplikasi Android

Tabel 4. 4 Skenario *use case* menambah data ekspedisi wajib pajak

Nomor *Use Case*	KF_1_01
Nama	Tambah ekspedisi
Tujuan	Untuk menambahkan data ekspedisi wajib pajak
Deskripsi	*Use case* ini menjelaskan proses petugas lapangan menambahkan informasi data wajib pajak
Aktor	Petugas lapangan
Skenario Utama	
Kondisi awal	Petugas lapangan masuk halaman utama aplikasi Android
Aksi Aktor	**Reaksi perangkat lunak**
Petugas lapangan menekan tombol "tambah ekspedisi" dan berhasil masuk ke halaman tambah ekspedisi	1. Perangkat lunak akan menampilkan *form-form* untuk mengetikkan data lalu simpan 2. Perangkat lunak menerima informasi penambahan data ekspedisi lalu akan mengirim ke aplikasi *web* dan menyimpan di *database* aplikasi Android
Skenario Alternatif 1 : Jika petugas lapangan menekan tombol *"back"*	
	3. Perangkat lunak akan kembali ke halaman utama

Kondisi Akhir Sukses	Informasi ekspedisi berhasil dikirim ke aplikasi *web* dan ditambahkan pada aplikasi Android

Skenario *use case* menambah data ekspedisi wajib pajak menjelaskan proses petugas lapangan menambahkan data ekspedisi wajib pajak. Proses di awali dengan petugas lapangan masuk halaman utama aplikasi Android kemudian petugas lapangan menekan tombol tambah ekspedisi dan berhasil masuk ke halaman tambah ekspedisi. Selanjutnya Perangkat lunak akan menampilkan *form-form* untuk mengetikkan data kemudian akan mengirimkan ke aplikasi *web* dan menyimpan di *database* aplikasi Android. Apabila petugas lapangan menekan tombol *back* maka perangkat lunak akan kembali ke halaman utama dan untuk kondisi sukses perangkat lunak akan menampilkan Informasi ekspedisi berhasil dikirim.

Tabel 4. 5 Skenario *use case* melihat data ekspedisi wajib pajak

Nomor *Use Case*	KF_1_02
Nama	Melihat data ekspedisi wajib pajak
Tujuan	Untuk melihat data ekspedisi wajib pajak
Deskripsi	*Use case* ini menjelaskan proses petugas lapangan melihat informasi data ekspedisi wajib paja
Aktor	Petugas lapangan
Skenario Utama	
Kondisi awal	Petugas lapangan masuk halaman utama aplikasi Android
Aksi Aktor	**Reaksi perangkat lunak**
Petugas lapangan menekan tombol “data ekspedisi”	1. Perangkat lunak akan menampilkan *list* data ekspedisi yang tersimpan pada *database* aplikasi Android
Skenario Alternatif 1 : Jika petugas lapangan menekan tombol *“back”*	

	2. Perangkat lunak akan kembali ke halaman utama
Kondisi Akhir Sukses	Informasi data ekspedisi berhasil ditampilkan

Skenario *use case* melihat data ekspedisi wajib pajak menjelaskan proses petugas lapangan melihat data ekspedisi wajib pajak yang tersimpan di *database* aplikasi Android. Proses di awali dengan petugas lapangan masuk halaman utama aplikasi Android kemudian petugas lapangan menekan tombol data ekspedisi dan berhasil masuk ke halaman data ekspedisi. Selanjutnya Perangkat lunak akan menampilkan *list view* data ekspedisi wajib pajak yang tersimpan di *database* aplikasi Android. Apabila petugas lapangan menekan tombol *back* maka perangkat lunak akan kembali ke halaman utama dan untuk kondisi sukses perangkat lunak akan menampilkan *list view* data ekspedisi wajib pajak.

Tabel 4. 6 Skenario *use case* menghapus data ekspedisi wajib pajak

Nomor *Use Case*	KF_1_03
Nama	Menghapus data ekspedisi wajib pajak
Tujuan	Untuk menghapuskan data ekspedisi wajib pajak
Deskripsi	*Use case* ini menjelaskan proses petugas lapangan menghapus informasi data ekspedisi wajib pajak
Aktor	Petugas lapangan
Skenario Utama	
Kondisi awal	Petugas lapangan masuk halaman utama aplikasi Android
Aksi Aktor	**Reaksi perangkat lunak**
Petugas lapangan menekan tombol “hapus ekspedisi”	1. Perangkat lunak akan menampilkan *list* data ekspedisi yang tersimpan kemudian

	memberi tanda centang pada ekspedisi yang ingin dihapus lalu tekan hapus. 2. Perangkat lunak menerima perintah penghapusan ekspedisi kemudian melakukan proses penghapusan ekspedisi
Skenario Alternatif 1 : Jika petugas lapangan menekan tombol *"cancel"*	
	3. Perangkat lunak akan berhenti mengirimkan *request* ke *GPS* dan kembali ke fitur tambah data
Skenario Alternatif 1 : Jika petugas lapangan menekan tombol *"back"*	
	4. Perangkat lunak akan kembali ke halaman utama
Kondisi Akhir Sukses	Data ekspedisi berhasil dihapus

Skenario *use case* menghapus data ekspedisi wajib pajak menjelaskan proses petugas lapangan menghapus data ekspedisi wajib pajak. Proses di awali dengan petugas lapangan masuk halaman utama aplikasi Android kemudian petugas lapangan menekan tombol hapus ekspedisi dan berhasil masuk ke halaman hapus ekspedisi. Selanjutnya Perangkat lunak akan menampilkan *list view* data ekspedisi wajib pajak yang tersimpan di *database* aplikasi Android lalu petugas lapangan akan memilih data yang akan dihapus. Apabila petugas lapangan menekan tombol *back* maka perangkat lunak akan kembali ke halaman utama dan untuk kondisi sukses perangkat lunak akan menampilkan notifikasi data berhasil dihapus.

2. Skenario *use case* aplikasi *web*

Tabel 4. 7 Skenario *use case* melakukan *login*

Nomor *Use Case*	KF_2_01
Nama	Melakukan *login*
Tujuan	Untuk masuk ke halaman aplikasi *web*

Deskripsi	*Use case* ini menjelaskan proses *admin* melakukan *login* untuk masuk ke halaman aplikasi *web*
Aktor	*Administrator*
Skenario Utama	
Kondisi awal	*Administrator* membuka halaman aplikasi *web*
Aksi Aktor	**Reaksi Aplikasi *web***
Administrator membuka halaman aplikasi *web*	1. Aplikasi *web* akan menampilkan *form username* dan *password*
Administrator memasukkan *username* dan *password*	2. Aplikasi *web* akan menampilkan halamna utama aplikasi *web*
Skenario Alternatif 1 : Jika *usernam* dan *password* salah	
	3. Aplikasi *web* akan menampilkan kembali *form username* dan *password*
Kondisi Akhir Sukses	Aplikasi *web* akan menampilkan halaman utama aplikasi *web*

Skenario *use case* melakukan *login* menjelaskan proses *administrator* melakukan *login* untuk masuk ke halamna utama aplikasi *web*. Proses di awali dengan *administrator* masuk halaman aplikasi *web* kemudian *administrator* mengisikan *username* dan *password*. Apabila *administrator* mengisikan *username* dan *password* yang salah maka aplikasi *web* akan menampilkan kembali halaman *login* dan untuk kondisi sukses aplikasi *web* akan menampilkan halaman utama dari aplikasi *web*.

Tabel 4. 8 Skenario *use case* melakukan *logout*

Nomor *Use Case*	KF_2_02
Nama	Melakukan *logout*
Tujuan	Untuk keluar dari halaman aplikasi *web*

Deskripsi	*Use case* ini menjelaskan proses *admin* melakukan *logout* untuk keluar dari halaman aplikasi *web*
Aktor	*Administrator*
Skenario Utama	
Kondisi awal	*Administrator* membuka halaman aplikasi *web*
Aksi Aktor	**Reaksi Aplikasi *web***
Administrator menekan tombol *logout*	1. Aplikasi *web* akan keluar dari halaman utama dan menampilkan *form username* dan *password*
Kondisi Akhir Sukses	Aplikasi *web* akan keluar dari halaman utama aplikasi *web*

Skenario *use case* melakukan *logout* menjelaskan proses *administrator* melakukan *logout* untuk keluar dari halamna utama aplikasi *web*. Proses di awali dengan *administrator* masuk halaman utama aplikasi *web* kemudian *administrator* menekan tombol *logout.* Untuk kondisi sukses aplikasi *web* akan keluar dari halaman utama aplikasi *web*.

Tabel 4. 9 Skenario *use case* membuat akun petugas lapangan

Nomor *Use Case*	KF_2_03
Nama	Membuat akun petugas lapangan
Tujuan	Untuk membuat akun petugas lapangan
Deskripsi	*Use case* ini menjelaskan proses aplikasi *web* membuat akun untuk petugas lapangan
Aktor	*Administrator*
Skenario Utama	
Kondisi awal	*Administrator* masuk ke halaman utama aplikasi *web*

Aksi Aktor	Reaksi Aplikasi *web*
Administrator mengklik tombol "buat akun"	1. Aplikasi *web* akan menampilkan *form* buat akun petugas lapangan
Skenario Alternatif 1 : Jika *administrator* menekan tombol *"back"*	
	2. Aplikasi *web* akan kembali ke halaman utama
Kondisi Akhir Sukses	Aplikasi *web* menyimpan akun yang dibuat ke *database*

Skenario *use case* membuat akun petugas lapangan menjelaskan proses *administrator* membuat akun petugas lapangan. Proses di awali dengan *administrator* masuk halaman utama aplikasi *web* kemudian *administrator* menekan tombol buat akun dan berhasil masuk ke halaman buat akun. Selanjutnya Aplikasi *web* akan menampilkan *form-form* untuk mengetikkan data petugas lapangan. Apabila *administrator* menekan tombol *back* maka Aplikasi *web* akan kembali ke halaman utama dan untuk kondisi sukses Aplikasi *web* akan menyimpan akun yang dibuat ke *database*.

Tabel 4. 10 Skenario *use case* melihat data petugas lapangan

Nomor *Use Case*	KF_2_04
Nama	melihat data petugas lapangan
Tujuan	Untuk melihat informasi data petugas lapangan
Deskripsi	*Use case* ini menjelaskan proses *administrator* melihat semua informasi tentang data petugas lapangan
Aktor	*Administrator*
Skenario Utama	
Kondisi awal	*Administrator* masuk ke halaman utama aplikasi *web*

Aksi Aktor	Reaksi Aplikasi *web*
Administrator mengklik tombol "lihat data petugas"	1. Aplikasi *web* akan menampilkan halaman data petugas lapangan 2. Aplikasi *web* akan menampilkan data petugas lapangan
Skenario Alternatif 1 : Jika *administrator* menekan tombol *"back"*	
	3. Aplikasi *web* akan kembali ke halaman utama
Kondisi Akhir Sukses	Aplikasi *web* akan menampilkan *list* informasi data petugas lapangan

Skenario *use case* melihat data petugas lapangan menjelaskan proses *administrator* melihat data petugas lapangan yang tersimpan di *database* aplikasi *web*. Proses di awali dengan *administrator* masuk halaman utama aplikasi *web* kemudian *administrator* menekan tombol lihat data petugas lapangan dan berhasil masuk ke halaman lihat data petugas lapangan. Selanjutnya Aplikasi *web* akan menampilkan informasi data petugas lapangan yang tersimpan di *database* aplikasi *web*. Apabila *administrator* menekan tombol *back* maka Aplikasi *web* akan kembali ke halaman utama dan untuk kondisi sukses Aplikasi *web* akan menampilkan *list* informasi data petugas lapangan.

Tabel 4. 11 Skenario *use case* melihat data wajib pajak

Nomor *Use Case*	KF_2_05
Nama	Melihat data wajib pajak
Tujuan	Untuk melihat informasi data wajib pajak
Deskripsi	*Use case* ini menjelaskan proses *administrator* melihat data induk wajib pajakyang tersimpan di aplikasi *web*
Aktor	*Administrator*

Skenario Utama	
Kondisi awal	*Administrator* masuk ke halaman utama aplikasi *web*
Aksi Aktor	**Reaksi Aplikasi *web***
Administrator mengklik tombol "lihat data wajib pajak"	1. Aplikasi *web* akan menampilkan halaman data wajib pajak 2. Aplikasi *web* akan menampilkan data wajib pajak
Skenario Alternatif 1 : Jika *administrator* menekan tombol *"back"*	
	3. Aplikasi *web* akan kembali ke halaman utama
Kondisi Akhir Sukses	Aplikasi *web* akan menampilkan *list* data wajib pajak

Skenario *use case* melihat data wajib pajak menjelaskan proses *administrator* melihat data wajib pajak yang tersimpan di *database* aplikasi *web*. Proses di awali dengan *administrator* masuk halaman utama aplikasi *web* kemudian *administrator* menekan tombol lihat data wajib pajak dan berhasil masuk ke halaman lihat data wajib pajak. Selanjutnya Aplikasi *web* akan menampilkan informasi data wajib pajak yang tersimpan di *database* aplikasi *web*. Apabila *administrator* menekan tombol *back* maka Aplikasi *web* akan kembali ke halaman utama dan untuk kondisi sukses Aplikasi *web* akan menampilkan *list* informasi data wajib pajak.

Tabel 4. 12 Skenario *use case* melihat data ekspedisi

Nomor *Use Case*	KF_2_06
Nama	Melihat data ekspedisi
Tujuan	Untuk melihat informasi data ekspedisi

Deskripsi	*Use case* ini menjelaskan proses *administrator* melihat semua data ekspedisi yang telah dikirimkan aplikasi Android
Aktor	*Administrator*
Skenario Utama	
Kondisi awal	*Administrator* masuk ke halaman utama aplikasi *web*
Aksi Aktor	**Reaksi Aplikasi *web***
Administrator menekan tombol "lihat ekspedisi"	1. Aplikasi *web* akan masuk ke halaman data ekspedisi 2. Aplikasi *web* akan menampilkan data ekspedisi
Skenario Alternatif 1 : Jika *administrator* menekan tombol *"back"*	
	3. Aplikasi *web* akan kembali ke halaman utama
Kondisi Akhir Sukses	Aplikasi *web* akan menampilkan *list* informasi data ekspedisi

Skenario *use case* melihat data ekspedisi menjelaskan proses *administrator* melihat data ekspedisi. Proses di awali dengan *administrator* masuk halaman utama aplikasi *web* kemudian *administrator* menekan tombol lihat data ekspedisi dan berhasil masuk ke halaman lihat data ekspedisi. Selanjutnya Aplikasi *web* akan menampilkan informasi data ekspedisi yang tersimpan di *database* aplikasi *web*. Apabila *administrator* menekan tombol *back* maka Aplikasi *web* akan kembali ke halaman utama dan untuk kondisi sukses Aplikasi *web* akan menampilkan *list* informasi data ekspedisi.

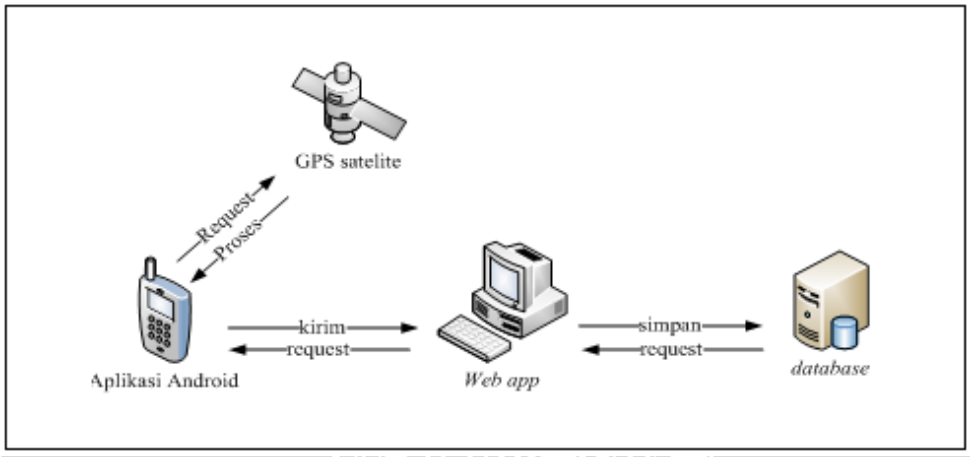
4.2 Perancangan perangkat lunak

Perancangan perangkat lunak dilakukan dalam empat tahap, yaitu perancangan arsitektural, perancangan basis data, perancangan *activity diagram* untuk

menggambarkan urutan aktivitas dari proses pada setiap *use case* yang ada pada perancangan arsitektural, perancangan *class diagram* untuk menggambarkan perancangan struktur *class-class* yang menyusun perangkat lunak, perancangan *sequence diagram* untuk menunjukkan aliran jalannya proses interaksi antar objek atau *class* yang disusun berdasarkan urutan waktu dan perancangan desain antarmuka aplikasi Android.

4.2.1 Perancangan arsitektur

Gambar 4. 7 merupakan gambaran arsitektur dari aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS. Gambaran arsitektur berfungsi untuk menggambarkan alur kerja perangkat lunak secara keseluruhan.



Gambar 4. 7 Arsitektur perangkat lunak

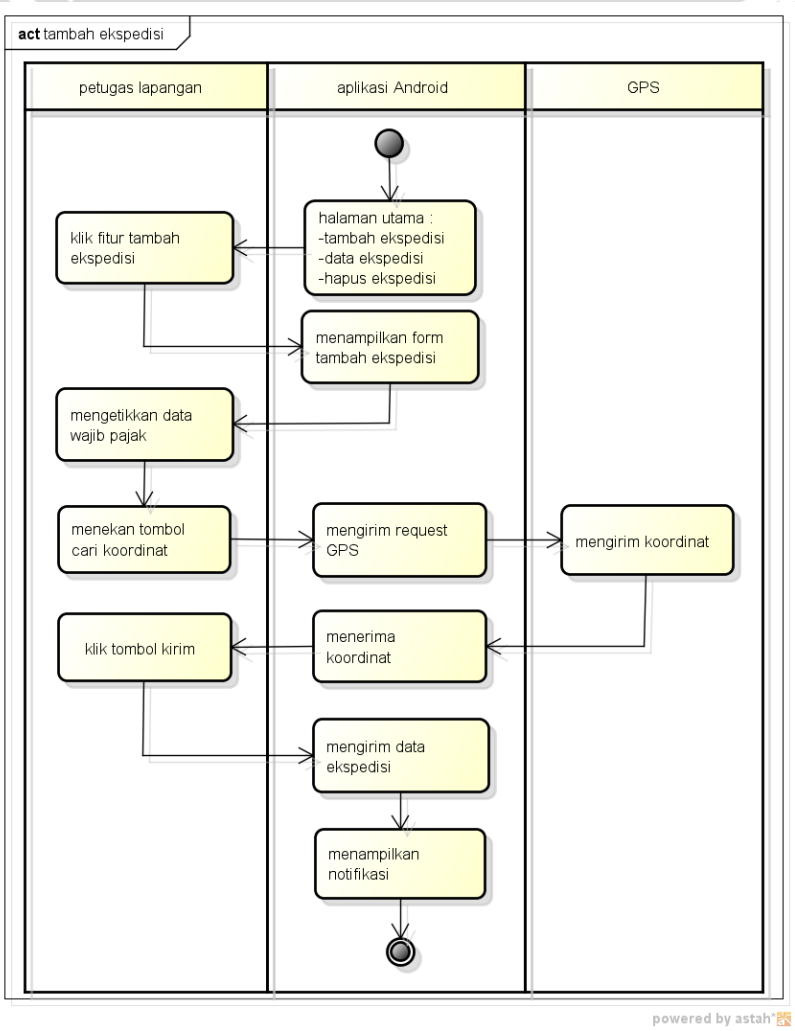
Proses kerja perangkat lunak dimulai dari petugas lapangan memasukkan data wajib pajak ke aplikasi Android kemudian melakukan *request* koordinat *longitude* dan *latitude* dengan menggunakan GPS. Setelah data didapatkan maka data tersebut akan dikirimkan ke aplikasi *web*. Data yang dikirimkan ke aplikasi *web* adalah data berupa informasi wajib pajak dan koordinat *longitude* dan *latidute*. Selanjutnya data akan disimpan di *database* aplikasi *web*.

4.2.2 Perancangan *Activity Diagram*

Perancangan *activity diagram* bertujuan untuk menggambarkan urutan aktivitas dari proses pada setiap *use case* yang ada. *Activity diagram* dibuat berdasarkan *use case* aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS yang telah dibuat.

a. *Activity diagram* tambah ekspedisi

Activity diagram tambah ekspedisi digunakan petugas lapangan untuk membuat ekspedisi data baru pada wajib pajak. Aplikasi Android memiliki 3 fitur utama yaitu tambah ekspedisi, data ekspedisi dan hapus ekspedisi.

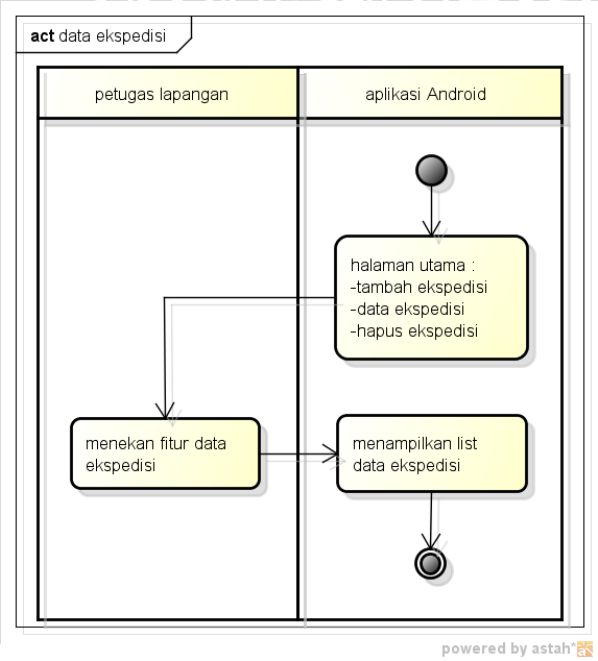


Gambar 4. 8 *Activity diagram* tambah ekspedisi

Pada Gambar 4. 8 setelah perangkat lunak berjalan akan menampilkan halaman utama yang berisi 3 fitur utama yaitu tambah ekspedisi, data ekspedisi dan hapus ekspedisi. kemudian petugas lapangan akan memilih fitur tambah ekspedisi. Setelah masuk halaman tambah ekspedisi petugas lapangan akan di sediakan beberapa *form* kosong sebagai tempat untuk mengetikkan data yang di peroleh. Setelah data selesai dimasukkan maka petugas lapangan akan menekan tombol cari koordinat untuk mendapatkan koordinat lokasi wajib pajak lalu tombol kirim untuk mengirim data ke aplikasi *web*. Bersamaan dengan aplikasi Android mengirim data maka data akan otomatis tersimpan dalam *database* aplikasi Android.

b. *Activity diagram* data ekspedisi

Activity diagram data ekspedisi berfungsi untuk melihat data ekspedisi yang tersimpan dalam *database* aplikasi Android.

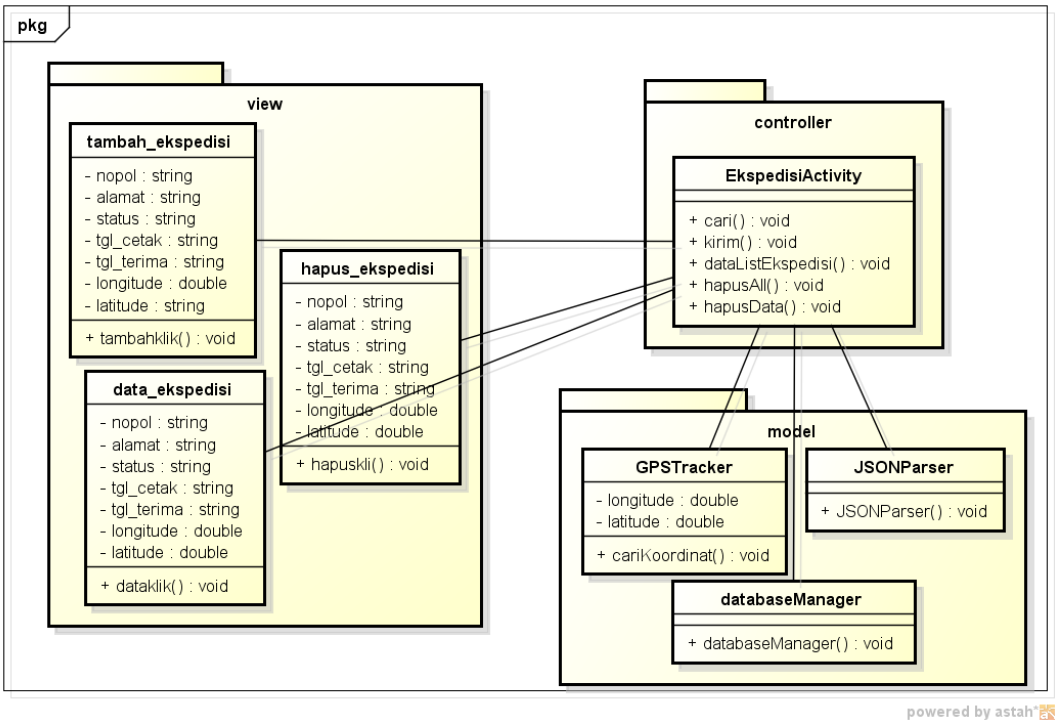


Gambar 4. 9 *Activity diagram* data ekspedisi

Pada Gambar 4. 9 menunjukkan *activity diagram* proses petugas lapangan melihat data ekspedisi. Awalnya perangkat lunak akan menampilkan halaman utama aplikasi Android lalu petugas lapangan menekan fitur data ekspedisi. Aplikasi akan

merespon dengan menampilkan *list* data ekspedisi yang tersimpan pada *database* aplikasi Android.

4.2.3 Perancangan *Class Diagram*



Gambar 4. 10 Pemodelan MVC kelas aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS

Pemodelan kelas memberikan gambaran pemodelan elemen-elemen kelas yang membentuk sebuah perangkat lunak. Kelas bisa didapatkan dengan menganalisis secara detail terhadap *use case* yang dimodelkan. Aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS juga disusun menggunakan struktur *model-view-controller* (MVC). Gambar 4. 10 menunjukkan pemodelan MVC diagram kelas dari aplikasi penstatusan SPKPKB berbasis Android menggunaka GPS yang akan dibuat.

Pada Gambar 4. 10 pemodelan kelas digambarkan dengan *interface* dan beberapa kelas yang menyusun aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS. Kelas `EkspedisiActivity` bertindak sebagai *controller* yang

memiliki relasi dengan kelas `home` untuk menampilkan halam utama aplikasi, kelas `tambah_ekspedisi` untuk menampilkan *form* pengisian data ekspedisi, kelas `data_ekspedisi` untuk menampilkan *list* data ekspedisi yang tersimpan di *database* aplikasi Android dan mempunyai relasi dengan kelas `hapus_ekspedisi` untuk menampilkan halaman hapus data. Kelas `EkspedisiActivity` juga memiliki relasi dengan kelas `databaseManager` yang berfungsi untuk menyimpan data ekspedisi, kelas `GPSTracker` berfungsi untuk mendapatkan koordinat *longitude* dan *latitude* lokasi wajib pajak dan kelas `JSONParser` berfungsi untuk mengirimkan data ekspedisi dari aplikasi Android ke aplikasi *web*.

4.2.4 Perancangan Sequences Diagram

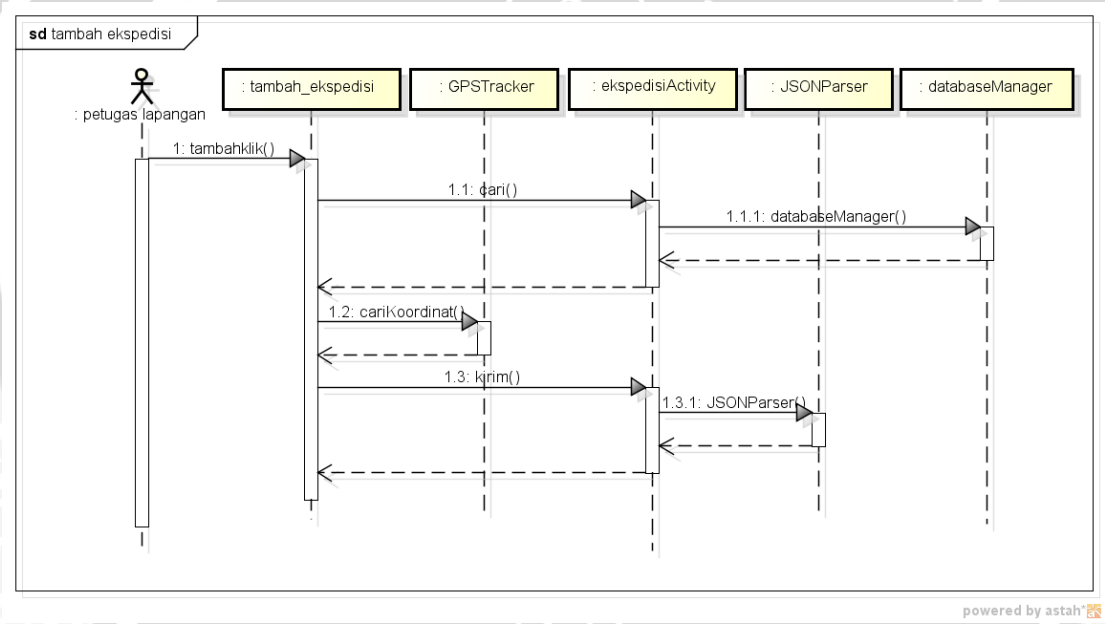
Sequences diagram menunjukkan aliran jalannya proses interaksi antar obyek atau *class* yang disusun berdasarkan urutan waktu pada aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS. *Sequence diagram* digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai *respons* dari sebuah *event* untuk menghasislkan *output* tertentu. *Sequences diagram* disusun dengan mengambil acuan pada *use case* dan *class-class* yang membentuk fungsionalitas yang digambarkan pada *use case* tersebut.

a. Sequence diagram tambah ekspedisi

Sequence diagram tambah ekspedisi menggambarkan proses ketika petugas lapangan akan menambah data ekspedisi baru. Gambar 4. 11 menjelaskan *sequence diagram* tambah ekspedisi.

Proses *sequence diagram* tambah ekspedisi menjelaskan aliran jalannya proses interaksi antar obyek dan *class* dalam fitur tambah ekspedisi. Setelah petugas lapangan masuk ke halaman utama aplikasi Android kemudian petugas lapangan akan masuk ke fitur tambah ekspedisi. Kemudian kelas `tambah_ekspedisi` akan menjalankan *method* `cari`() ke kelas `ekspedisiActivity` dan kelas `ekspedisiAvtivity` akan menjalankan *method* `databaseManager`() untuk mencari data sesuai dengan

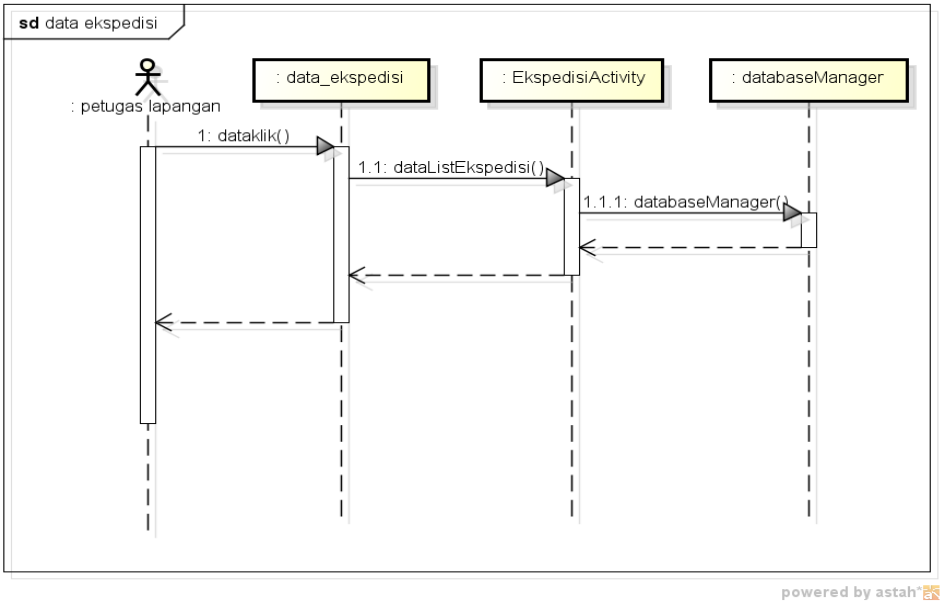
nomor polisi yang dicari. Setelah data didapat kemudian kelas `databaseManager` akan mengirimkan *return* ke kelas `EkspedisiActivity` lalu kelas `EkspedisiActivity` akan menampilkan data ke petugas lapangan. Kemudian kelas `tambah_ekspedisi` akan menjalankan *method* `cariKoordinat`() untuk meminta koordinat *longitude* dan *latitude* ke kelas `GPSTracker`. Setelah data koordinat didapat kelas `GPSTracker` akan mengirimkan *return* ke kelas `tambah_ekspedisi`. Selanjutnya petugas lapangan akan mengirim data ke aplikasi *web* dengan menjalankan *method* `kirim()` ke kelas `JSONParser`. Apabila data berhasil dikirim maka kelas `JSONParser` akan menampilkan *return* ke kelas `tambah_ekspedisi`.



Gambar 4. 11 *Sequence diagram* tambah ekspedisi aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS

b. Sequence diagram data ekspedisi

Sequence diagram data ekspedisi menggambarkan proses ketika petugas lapangan akan melihat data yang tersimpan pada *database* aplikasi Android. Gambar 4. 12 menjelaskan *sequence diagram* data ekspedisi.



Gambar 4. 12 *Sequence diagram* data ekspedisi aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS

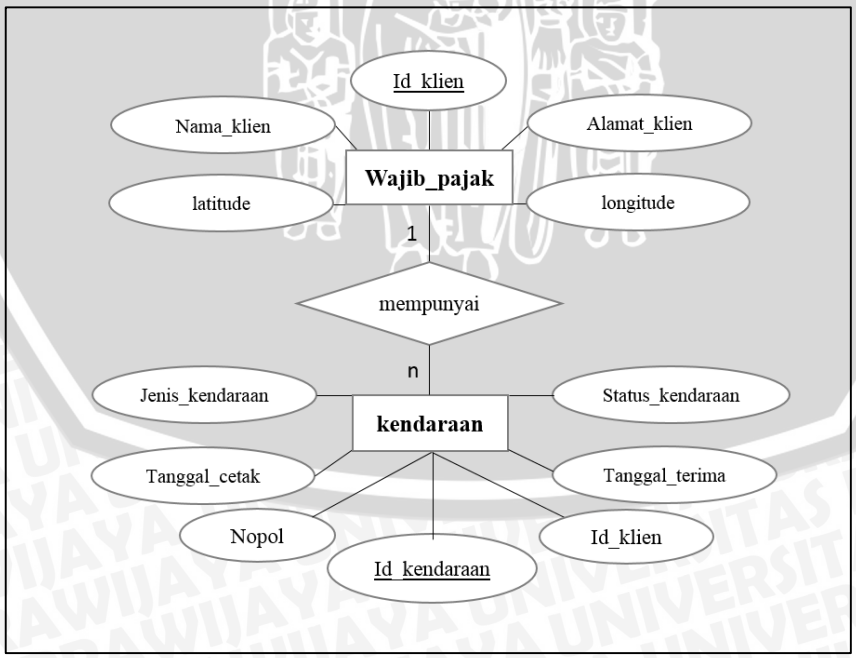
Proses *sequence diagram* data ekspedisi menjelaskan aliran jalannya proses interaksi antar obyek dan *class* dalam fitur data ekspedisi. Setelah petugas lapangan masuk ke halaman utama aplikasi Android kemudian petugas lapangan menekan tombol data ekspedisi untuk masuk ke fitur data ekspedisi. Kemudian petugas lapangan akan menjalankan *method* `dataklik()` untuk memanggil kelas `data_ekspedisi`, kelas `data_ekspedisi` berfungsi untuk menampilkan halaman data ekspedisi kemudian kelas `data_ekspedisi` akan menjalankan *method* `dataEkspedisi()` ke kelas `EkspedisiActivity` sebagai *controller*. Selanjutnya kelas `EkspedisiActivity` akan menjalankan *method* `getDataCount()` ke kelas `databaseManager` untuk mengambil data yang tersimpan di *database* aplikasi Android. Kelas `databaseManager` akan memberikan *return* ke kelas `EkspedisiActivity`. Kemudian kelas `EkspedisiActivity` akan mengirimkan *return* kembali ke kelas `data_ekspedisi` ke petugas lapangan untuk menampilkan *list* data ekspedisi yang tersimpan pada *database* aplikasi Android.

4.2.5 Perancangan Basis Data

Perancangan basis data pada aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS digunakan untuk menyimpan data-data yang didapat dari lapangan. Perancangan basis data mengambil acuan dari analisis data yang dilakukan pada tahap analisis kebutuhan. Dalam aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS terdapat 2 basis data yang digunakan, yaitu basis data SQLite untuk aplikasi Android dan basis data MySQL untuk aplikasi *web*. Basis data SQLite berfungsi untuk menyimpan data yang telah dimasukkan pada aplikasi Android dan juga sebagai *back up* data yang akan dikirimkan ke aplikasi *web* untuk menghindari kehilangan data saat proses pengiriman ataupun saat kondisi jaringan sedang *offline* sehingga tidak dimungkinkan untuk melakukan pengiriman data. Basis data MySQL berfungsi sebagai tempat penyimpanan data utama yang berisi data petugas lapangan, data wajib pajak dan data ekspedisi dari aplikasi Android.

1. Basis data aplikasi Android

Gambar 4. 13 merupakan diagram *entity relationship* aplikasi Android berdasarkan hasil analisis dari perancangan data.



Gambar 4. 13 Diagram *entity relationship* aplikasi Android

Diagram *entity relationship* akan menggambarkan rancangan hubungan antara entitas dengan atribut yang akan ditampilkan pada aplikasi Android. Pada rancangan ini data terdiri dari 2 entitas dan 12 atribut. Entitas wajib pajak memiliki atribut id wajib pajak, nama wajib pajak, alamat wajib pajak, *longitude* dan *latitude*. Sedangkan pada entitas kendaraan memiliki 7 atribut yaitu id kendaraan, id wajib pajak, nopol, status kendaraan, jenis kendaraan, tanggal cetak dan tanggal terima. Pada aplikasi Android, 1 wajib pajak dapat mempunyai banyak kendaraan.

Tabel 4. 13 struktur tabel aplikasi Android entitas wajib pajak

Field	Tipe Data	Keterangan	contoh
id_ wajib_pajak	Text	Id wajib pajak	01 / 02
nama_ wajib_pajak	Text	Nama lengkap wajib pajak	ZAINAL ARIFIN / SYAMSUL BAHRI
alamat_ wajib_pajak	Text	Alamat wajib pajak	BUKIT CEMARA 7 F/5 MULYOAGUNG DAU / DS BUNUT WETAN 02/03 KRAJAN PAKIS
longitude	Real	*Longitude*	106.918945 / 112.701671
latitude	Real	*Latitude*	-6.500899 / -7.954420

Entitas wajib pajak berfungsi untuk menyimpan data pribadi yang berhubungan dengan wajib pajak seperti nama dan alamat. Sedangkan atribut *longitude* dan *latitude* digunakan untuk mencari koordinat alamat wajib pajak di Google Maps .

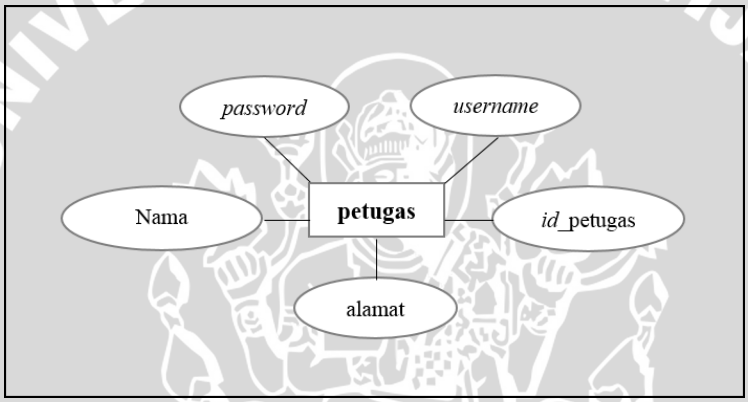
Tabel 4. 14 Struktur tabel aplikasi Android entitas kendaraan

Field	Tipe Data	Keterangan	Contoh
Id_kendaraan	Text	Id kendaraan	0001 / 0112
id_ wajib_pajak	Text	Id wajib pajak	001 / 082
nopol	Text	Nomor polisi kendaraan	N 2425 GC / N 2855 JG
status_kendaraan	Text	Status kendaraan wajib pajak	MASIH DIMILIKI / LAPOR JUAL

tanggal_cetak	Time	Tanggal surat dicetak	03/01/2014 / 17/01/2014
tanggal_terima	Time	Tanggal surat diterima wajib pajak	10/01/2014 / 24/01/2014
jenis_kendaraan	Text	Roda dua atau empat	RODA 2

Entitas kendaraan digunakan untuk menyimpan data yang berhubungan dengan kendaraan seperti nopol, status kendaraan, tanggal cetak tanggal terimadan jenis kendaraan.

2. Basis data aplikasi *web*



Gambar 4. 14 Diagram *entity relationship* aplikasi *web*

Gambar 4. 14 merupakan diagram *entity relationship* aplikasi *web* berdasarkan hasil analisis data dari perancangan data. Diagram *entity relationship* akan menggambarkan rancangan hubungan antara entitas dengan atribut yang akan ditampilkan pada aplikasi *web*. Pada basis data aplikasi *web* hanya menggunakan 1 entitas karena data yang dibutuhkan hanya data petugas lapangan. Pada rancangan ini data terdiri dari 1 entitas dan 5 atribut. Entitas petugas memiliki atribut id petugas, nama, alamat, *username* dan *password*.

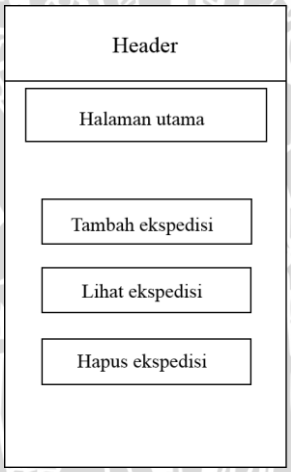
Tabel 4. 15 Struktur tabel petugas

Field	Tipe Data	Keterangan	Contoh
id_petugas	Text	*id* petugas lapangan	01 / 102

nama_	Text	Nama lengkap petugas lapangan	Ema Fauziah / Mian
alamat	Text	Alamat petugas lapangan	PURI LOSARI N/7 LOSARI SINGOSARI / JL RAYA BLIMBING NO.51
username	Text	*Username* petugas lapangan	EmaFauziah / MIAN
password	Text	*Password* petugas lapangan	********** / ****

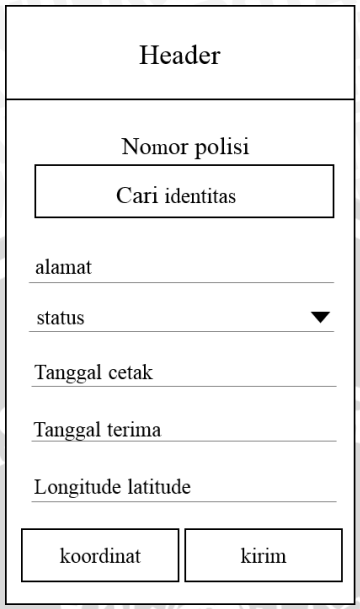
4.2.6 Perancangan Desain Antarmuka Aplikasi Android

Perancangan antarmuka merupakan rancangan untuk implementasi antarmuka aplikasi Android yang akan digunakan petugas lapangan. Berikut merupakan tampilan antarmuka halaman utama aplikasi Android.



Gambar 4. 15 Rancangan antar muka halaman utama aplikasi Android

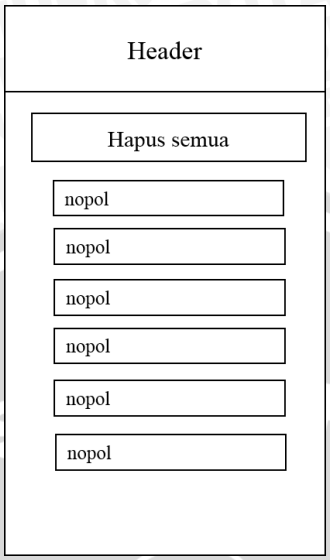
Gambar 4. 15 adalah perancangan halaman utama aplikasi android yang merupakan tampilan awal ketika petugas lapangan membuka aplikasi. Halaman utama aplikasi Android memuat header, logo halaman utama dan logo 3 fitur aplikasi Android yang terdiri dari tambah ekspedisi, lihat ekspedisi dan hapus ekspedisi. Ketika petugas lapangan menekan fitur tambah ekspedisi, aplikasi Android akan menampilkan halaman tambah ekspedisi.



Gambar 4. 16 Rancangan antar muka halaman tambah ekspedisi

Gambar 4. 16 adalah rancangan antarmuka halaman tambah ekspedisi yang berguna untuk menuliskan data wajib pajak dan status kendaraan. Petugas lapangan harus mengetikkan nomor polisi wajib pajak kemudian petugas lapangan akan menekan tombol cari identitas. Setelah data wajib pajak didapatkan selanjutnya petugas lapangan akan memilih status kendaraan sesuai dengan kondisi kendaraan wajib pajak. Status kendaraan meliputi masih dimiliki, lapor jual, rusak, hilang, alamat tidak jelas dan lainnya. Petugas lapangan tidak perlu mengetikkan tanggal cetak dan tanggal terima karena aplikasi Android akan secara otomatis mengisi tanggal tersebut. Selanjutnya petugas lapangan akan menekan tombol cari koordinat untuk mendapatkan koordinat longitude dan latitude. Kemudian petugas lapangan akan menekan tombol kirim untuk mengirimkan data yang telah dituliskan ke aplikasi web.

Dihalaman utama aplikasi Android terdapat fitur hapus ekspedisi yang apabila ditekan akan menampilkan halaman hapus ekspedisi.

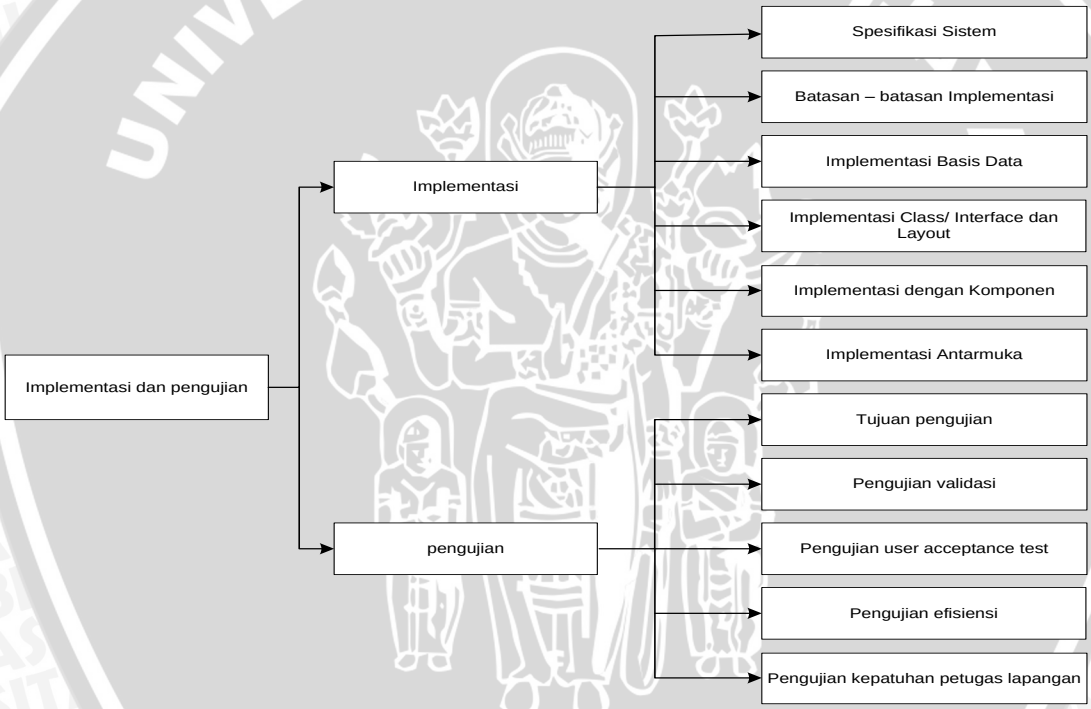


Gambar 4. 17 Rancangan antar muka Halaman Hasil Pencarian

Gambar 4. 17 adalah rancangan antarmuka halaman hapus ekspedisi yang menyimpan data-data hasil ekspedisi yang telah dikirimkan ke aplikasi web. Petugas lapangan akan memilih salah satu nomor polisi yang ingin dihapus kemudian petugas lapangan akan menekan hapus data. Petugas lapangan juga bisa menghapus data ekspedisi secara bersamaan dengan cara menekan tombol hapus semua yang ada di atas list nomor polisi. Ketika tombol hapus semua ditekan maka data ekspedisi yang tersimpan akan secara otomatis terhapus dari database aplikasi Android.

BAB V
IMPLEMENTASI

Bab ini membahas tahapan implementasi dan pengujian dari aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS berdasarkan hasil yang telah didapatkan dari tahapan analisis kebutuhan dan perancangan perangkat lunak. Tahap-tahap pembahasan implementasi yang dikerjakan digambarkan pada Gambar 5. 1 berikut ini.



Gambar 5. 1 Diagram blok implementasi

Pembahasan terdiri dari tiga tahap, Tahap pertama adalah pembahasan implementasi yang terdiri atas Pembahasan terdiri atas penjelasan tentang spesifikasi sistem, batasan-batasan dalam implementasi, implementasi tiap *class* pada *file* program, implementasi kode pada aplikasi Android, dan implementasi antarmuka. Lalu dilanjutkan dengan tahap kedua mengenai pengujian perangkat lunak yang terdiri dari

tujuan pengujian, pengujian validasi, pengujian *user acceptance testing*, pengujian efisiensi data dan pengujian kepatuhan petugas lapangan serta dilanjutkan dengan tahap ketiga berupa analisis hasil pengujian untuk mendapatkan kesimpulan berdasarkan hasil pengujian yang telah dilakukan. Hasil analisis kebutuhan yang telah diuraikan pada Bab 4 menjadi acuan untuk melakukan implementasi menjadi sebuah sistem yang dapat berfungsi sesuai dengan kebutuhan. Spesifikasi sistem diimplementasikan pada spesifikasi perangkat keras dan perangkat lunak.

5.1 Implementasi

Bagian ini diawali dengan penjelasan tentang spesifikasi lingkungan pengembangan perangkat lunak, dilanjutkan dengan implementasi basis data, implementasi aplikasi Android dan implementasi pada aplikasi *web*.

5.1.1 Spesifikasi Lingkungan Pengembangan Sistem

Aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS dikerjakan dalam lingkungan implementasi yang terdiri dari perangkat keras dan perangkat lunak.

a. Spesifikasi perangkat keras

Spesifikasi perangkat keras yang dipakai dalam proses pembuatan aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS yaitu menggunakan komputer *notebook* dengan processor Intel Core i5. Perangkat keras lain yang digunakan untuk proses implementasi dan pengujian pada aplikasi Android adalah *smartphone* yang menggunakan sistem operasi Android 4.4 (Kitkat).

b. Spesifikasi perangkat lunak

Spesifikasi perangkat lunak yang dipakai dalam proses pengembangan aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS dijelaskan pada Tabel 5. 1.

Tabel 5. 1 Spesifikasi perangkat lunak komputer

Operating System	Microsoft Windows 7
Programming Language	- Java (untuk aplikasi Android) - PHP (untuk aplikasi *web*)

Text Editor / IDE	- Eclipse LUNA (untuk Java) - Notepad++ (untuk PHP)
Server	- Apache HTTP Server (Localhost)
Database	- SQLite *Database* (untuk aplikasi Android) - MYSQL *Database* (untuk aplikasi *web*)

5.1.2 Batasan-Batasan Implementasi

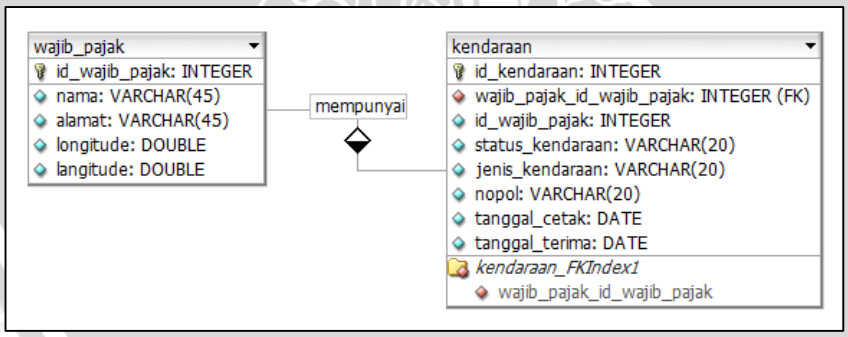
Beberapa batasan-batasan dalam mengimplementasikan sistem adalah sebagai berikut:

a. Pembuatan aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS ini dikerjakan dengan bahasa Pemrograman Java di lingkungan sistem operasi Android.

b. Sistem operasi Android yang digunakan minimal Android OS, v2.3.6 (Gingerbread).

c. Tingkat akurasi pengiriman data ke *server* tergantung kecepatan akses internet *smartphone* pengguna.

5.1.3 Implementasi Basis Data



Gambar 5. 2 Diagram *entity relationship* aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS

Implementasi penyimpanan data dilakukan dengan *database management system* MySQL pada aplikasi *administrator* dan *database management system*. Hasil implementasi penyimpanan data ini berupa *script – script* SQL.

5.1.4 Implementasi Class/Interface dan Layout

Implementasi *class*/*interface* merupakan proses implementasi dari hasil perancangan pada bab perancangan yang direalisasikan pada sebuah *file* program dengan ekstensi *.java. Tabel 5. 2 menjelaskan mengenai pasangan antara *class* dengan *file* program yang digunakan untuk mengimplementasikannya.

Tabel 5. 2 Implementasi *class* pada kode program *.java

No.	*Package*	Nama *Class*	Nama *File* Program
1	ekspedisi	*home*	home.java
2	ekspedisi	Tambah ekspedisi	tambah_ekspedisi.java
3	ekspedisi	Data ekspedisi	Data_ekspedisi.java
4	ekspedisi	List data ekspedisi	DataListEkspedisi.java
5	ekspedisi	Detail ekspedisi	DetailEkspedisi.java
6	ekspedisi	Hapus ekspedisi	Data_ekspedisi.java
7	ekspedisi	Cari koordinat	Cari_koordinat.java
8	ekspedisi	GPS Tracker	GPSTracker.java
9	ekspedisi	Database Manager	databaseManager.java
10	ekspedisi	JSON Parser	JASONParser.java

Tabel 5. 3 merupakan implementasi *layout* yang akan digunakan di dalam aplikasi. *Layout* ini berfungsi untuk mengatur tampilan aplikasi di dalam sistem operasi android yang akan digunakan pengguna untuk berinteraksi dengan sistem. Pada operasi sistem android, *layout* dibuat menggunakan *tag-tag* XML dengan ekstensi *.xml.

Tabel 5. 3 Implementasi *layout*

No.	Folder	Nama *Layout*	Nama File Program
1	/res/layout	*Home*	*Home*.xml
2	/res/layout	Tambah_ekspedisi	Activity_tambah_ekspedisi.xml
3	/res/layout	Data_ekspedisi	Activity_data_ekspedisi.xml
4	/res/layout	Hapus_ekspedisi	Activity_hapus_ekspedisi.xml

5.1.5 Implementasi Kode Pada Aplikasi Android

Aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS memiliki beberapa fitur yang terbagi dalam beberapa fungsi. Pada implementasi kode program Aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS terdapat beberapa bagian penting yang akan dijelaskan. Kode 5. 1 merupakan kode pada *class* tambah ekspedisi yang digunakan untuk mencari koordinat lokasi wajib pajak pada aplikasi Android. Kode tersebut berfungsi untuk mengambil nilai *longitude* dan *latitude* di GPS.

```
public Location getLocation() {
        try {
                locationManager = (LocationManager) mContext
                                        .getSystemService(LOCATION_SERVICE);

                // getting GPS status
                isGPSEnabled = locationManager
                .isProviderEnabled(LocationManager.GPS_PROVIDER);

                // getting network status
                isNetworkEnabled = locationManager
                .isProviderEnabled(LocationManager.NETWORK_PROVIDER);

        if (!isGPSEnabled && !isNetworkEnabled) {
                        // no network provider is enabled
                } else {
                        this.canGetLocation = true;
        if (isNetworkEnabled) {
        locationManager.requestLocationUpdates(
                LocationManager.NETWORK_PROVIDER, MIN_TIME_BW_UPDATES,
                IN_DISTANCE_CHANGE_FOR_UPDATES, this);
                Log.d("Network", "Network");

        if (locationManager != null) {
                location = locationManager
                .getLastKnownLocation(LocationManager.NETWORK_PROVIDER);

        if (location != null) {
                        latitude = location.getLatitude();
                        longitude = location.getLongitude();
                        }
                }
        }
// if GPS Enabled get lat/long using GPS Services
        if (isGPSEnabled) {
            if (location == null) {
                locationManager.requestLocationUpdates(
```

```
                LocationManager.GPS_PROVIDER,MIN_TIME_BW_UPDATES,
                MIN_DISTANCE_CHANGE_FOR_UPDATES, this);
                Log.d("GPS Enabled", "GPS Enabled");

        if (locationManager != null) {
                location = locationManager
                .getLastKnownLocation(LocationManager.GPS_PROVIDER);
         if (location != null) {
                latitude = location.getLatitude();
                longitude = location.getLongitude();
                }
            }
    }
 }
```

Kode 5. 1 *Script* Java pada *class* tambah ekspedisi untuk mengambil koordinat *longitude dan latitude*

Pada Kode 5. 1 merupakan kode pada *class* tambah ekspedisi untuk mengambil koordinat *longitude* dan *latitude* pada suatu posisi

```
JSONParser jParser = new JSONParser();

JSONObject json =
jParser.makeHttpRequest("http://192.168.43.64/ci/get_data.php?kategori=2&
&nopol="+ednopol.getText().toString().toUpperCase().replace(" ", "-
")+"&&sk="+status_kendaraan+"&&tgl="+tgl_terima.getText().toString().repl
ace(" ",
"")+"&&long="+String.valueOf(longitude)+"&&lat="+String.valueOf(latitude)
, "GET");

Log.d("url tambah: ",
"http://10.0.2.2/ci/get_data.php?kategori=2&&nopol="+ednopol.getText().to
String().toUpperCase().replace(" ", "-
")+"&&sk="+status_kendaraan+"&&tgl="+tgl_terima.getText().toString().repl
ace(" ",
"")+"&&long="+String.valueOf(longitude)+"&&lat="+String.valueOf(latitude)
);

if(json==null){
       Toast.makeText(getApplicationContext(), "Gagal",
Toast.LENGTH_LONG).show();
    }else{
        try {
             success = json.getInt("success");
             databaseManager db = new
databaseManager(tambah_ekspedisi.this);
```

```
if (success == 1) {
        JSONArray barang_array = null;

        db.insertTambahData(nopol_simpan.toUpperCase(),
sk[Integer.valueOf(status_kendaraan)], alamat.getText().toString(),
tgl_cetak.getText().toString(), tgl_terima.getText().toString(),
longitude, latitude);

 }
```

Kode 5. 2 *Script* Java pada *class* tambah data untuk mengirim data

Pada Kode 5. 2 merupakan *script* Java pada *class* tambah data yang digunakan untuk mengirim data ke aplikasi *web*. `JSONObject json = jParser.makeHttpRequest ("http://192.168.43.64/ci/get_data.php”)` *Script* ini akan mengirim data ke alamat atau *link* sesuai dengan alamat tersebut.

```
public void insertTambahData(String nopol, String status_kendaraan,
String alamat, String tanggal_cetak, String tanggal_terima, double
longitude, double latitude) {
            ContentValues contentValues = new ContentValues();
            SQLiteDatabase db = getWritableDatabase();
            contentValues.put("NOPOL", nopol);
            contentValues.put("ALAMAT", alamat);
            contentValues.put("STATUS_KENDARAAN", status_kendaraan);

            contentValues.put("TGL_CETAK", tanggal_cetak);
            contentValues.put("TANGGAL_TERIMA", tanggal_terima);
            contentValues.put("LATITUDE", latitude);
            contentValues.put("LONGITUDE", longitude);
            db.insert("kirim", null, contentValues);

      }
```

Kode 5. 3 *Script* Java pada *class* databaseManager untuk menyimpan data

Pada Kode 5. 3 merupakan *script* Java pada *class* *databaseManager* yang digunakan untuk menyimpan data yang telah *diinputkan* pada *class* tambah data. Data yang disimpan seperti nomor polisi, alamat, status kendaraan, tanggal cetak, tanggal terima, *longitude* dan *latitude*.

5.1.6 Implementasi Antarmuka

Implementasi antarmuka berdasarkan perancangan antarmuka pada bab perancangan. Antarmuka aplikasi Android merupakan sebuah aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS. Antarmuka untuk aplikasi Android terdiri atas halaman *home*, halaman tambah ekspedisi, halaman data ekspedisi dan halaman hapus ekspedisi.

Antarmuka halaman *home* merupakan tampilan awal ketika petugas lapangan menjalankan aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS. Pada halaman ini akan menampilkan logo aplikasi pada bagian header, gambar rumah sebagai penanda halaman home dan 3 buah tombol yang merupakan tombol tambah ekspedisi, data ekspedisi dan hapus ekspedisi untuk mengakses fitur selanjutnya. Berikut adalah script pada halaman utama ditunjukkan pada Kode 5. 4.

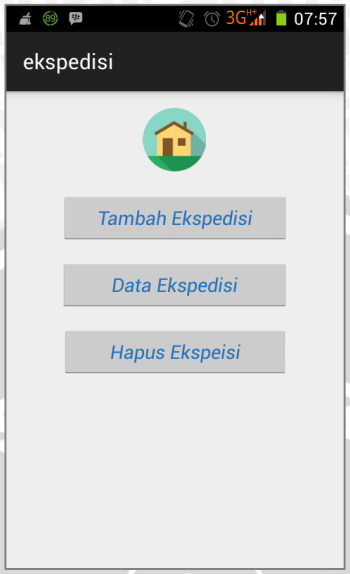
```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@drawable/background"
    android:gravity="center_horizontal"
    android:orientation="vertical"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context="com.example.ekspedisispkpkb.MainActivity" >

    <ImageView
        android:id="@+id/imageView2"
        android:layout_width="282dp"
        android:layout_height="60dp"
        android:layout_marginTop="0dp"
        android:scaleType="fitCenter"
        android:src="@drawable/home1" />

    <Button
        android:id="@+id/button1"
        android:layout_width="217dp"
        android:layout_height="wrap_content"
        android:layout_marginLeft="35dp"
        android:layout_marginRight="35dp"
        android:layout_marginTop="20dp"
        android:onClick="tambahklik"
        android:text="Tambah Ekspedisi" />
```

```
    <Button
        android:id="@+id/button2"
        android:layout_width="218dp"
        android:layout_height="wrap_content"
        android:layout_marginBottom="15dp"
        android:layout_marginLeft="35dp"
        android:layout_marginRight="35dp"
        android:layout_marginTop="15dp"
        android:onClick="dataklik"
        android:text="Data Ekspedisi" />

    <Button
        android:id="@+id/button3"
        android:layout_width="216dp"
        android:layout_height="wrap_content"
        android:layout_marginLeft="35dp"
        android:layout_marginRight="35dp"
        android:onClick="hapusklik"
        android:text="Hapus Ekspeisi" />

</LinearLayout>
```

Kode 5. 4 Script untuk menampilkan halaman *home*

Halaman *home* aplikasi Android akan menampilkan logo aplikasi, gambar rumah dan 3 tombol utama. Halaman *home* aplikasi akan selalu muncul saat pertama kali aplikasi dijalankan. Gambar 5. 3 merupakan antarmuka halaman utama aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS.

Ketika tombol tambah ekspedisi ditekan maka selanjutnya akan masuk ke halaman tambah ekspedisi. Antarmuka halaman tambah ekspedisi merupakan halaman yang digunakan untuk menambah informasi wajib pajak baru. Antarmuka halaman tambah ekspedisi berisi logo aplikasi, *form* nomor polisi, tombol cari identitas, alamat wajib pajak, status kendaraan, tanggal cetak, tanggal terima, *latitude* dan *longitude* . Antarmuka halaman tambah ekspedisi juga terdapat tombol cari koordinat dan tombol kirim di bawah *form entry* data. Berikut adalah script pada halaman tambah ekspedisi ditunjukkan pada Kode 5. 5.



Gambar 5. 3 Antarmuka halaman *home*

```
<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"

    tools:context="com.example.ekspedisispkpkb.MainActivity"
    android:background="@drawable/background" >

    <ScrollView
        android:id="@+id/scrollView1"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content">

        <RelativeLayout
            android:layout_width="match_parent"
            android:layout_height="511dp"
            android:orientation="vertical"
            android:paddingBottom="@dimen/activity_vertical_margin"
            android:paddingLeft="@dimen/activity_horizontal_margin"
            android:paddingRight="@dimen/activity_horizontal_margin"
            android:paddingTop="@dimen/activity_vertical_margin" >

            <EditText
                android:id="@+id/ednopol"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_centerHorizontal="true"
                android:hint="Nopol" />
```

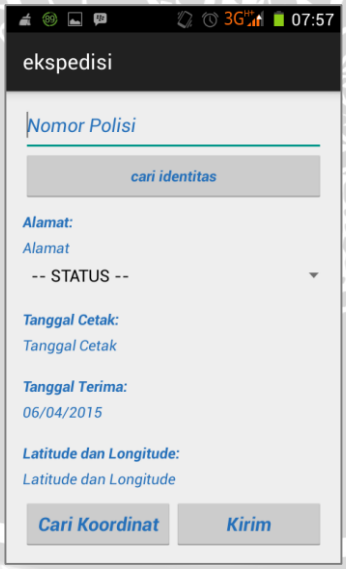
```
<Button
        android:id="@+id/btcari"
        style="?android:attr/buttonStyleSmall"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_below="@+id/ednopol"
        android:onClick="btcari"
        android:text="cari identitas" />

<TextView
        android:id="@+id/tx_alamat"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:textStyle="bold"
        android:layout_marginTop="10dp"
        android:text="Alamat:"
        android:layout_below="@+id/btcari"/>

<TextView
        android:id="@+id/tx_isi_alamat"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:layout_marginTop="5dp"
        android:text="Alamat"
        android:layout_below="@+id/tx_alamat"/>

<Spinner
        android:id="@+id/sp_status"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_below="@+id/tx_isi_alamat"
        android:layout_marginTop="5dp" />

<TextView
        android:id="@+id/tx_tgl_cetak"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:textStyle="bold"
        android:layout_marginTop="20dp"
        android:text="Tanggal Cetak:"
        android:layout_below="@+id/sp_status"/>

<TextView
        android:id="@+id/tx_isi_tgl_cetak"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:layout_marginTop="5dp"
```

```
            android:text="Tanggal Cetak"
            android:layout_below="@+id/tx_tgl_cetak"/>

    <TextView
            android:id="@+id/tx_tgl_terima"
            android:layout_width="fill_parent"
            android:layout_height="wrap_content"
            android:ems="10"
            android:textStyle="bold"
            android:layout_marginTop="20dp"
            android:text="Tanggal Terima:"
            android:layout_below="@+id/tx_isi_tgl_cetak"/>

    <TextView
            android:id="@+id/tx_isi_tgl_terima"
            android:layout_width="fill_parent"
            android:layout_height="wrap_content"
            android:ems="10"
            android:layout_marginTop="5dp"
            android:text="Tanggal Terima"
            android:layout_below="@+id/tx_tgl_terima"/>

    <TextView
            android:id="@+id/tx_lat_long"
            android:layout_width="fill_parent"
            android:layout_height="wrap_content"
            android:ems="10"
            android:textStyle="bold"
            android:layout_marginTop="20dp"
            android:text="Latitude dan Longitude:"
            android:layout_below="@+id/tx_isi_tgl_terima"/>

    <TextView
            android:id="@+id/tx_isi_lat_long"
            android:layout_width="fill_parent"
            android:layout_height="wrap_content"
            android:ems="10"
            android:layout_marginTop="5dp"
            android:text="Latitude dan Longitude"
            android:layout_below="@+id/tx_lat_long"/>

    <LinearLayout android:id="@+id/linearLayout1"
          android:layout_height="wrap_content"
          android:layout_width="fill_parent"
          android:layout_below="@+id/tx_isi_lat_long"
          android:layout_marginTop="10dp">

            <Button
                android:id="@+id/bt_cari_koordinat"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
```

```
                    android:layout_weight="1"
                    android:onClick="btcari_koordinat"
                    android:text="Cari Koordinat" >
</Button>
                </LinearLayout>
```

Kode 5. 5 *Script* untuk menampilkan halaman tambah data

Halaman tambah ekspedisi akan menampilkan *form* untuk *entry* nomor polisi wajib pajak. Petugas lapangan hanya perlu mengetikkan nomor polisi wajib pajak yang akan ditambahkan datanya kemudian petugas lapangan menekan tombol cari identitas dan aplikasi Android akan meminta data berdasarkan nomor polisi tersebut dan akan dikirimkan kembali ke aplikasi Android. Petugas lapangan kemudian mengisi status kendaraan wajib pajak sesuai dengan kondisi terkini kendaraan. Setelah itu petugas lapangan menekan tombol cari koordinat untuk mendapatkan koordinat alamat wajib pajak dan kemudian akan menekan tombol kirim untuk mengirim data ke aplikasi *web*. Proses mengirim data pada aplikasi ini menggunakan JSON. Gambar 5. 4 menunjukkan antarmuka halaman tambah ekspedisi.



Gambar 5. 4 Antarmuka halaman tambah informasi ekspedisi

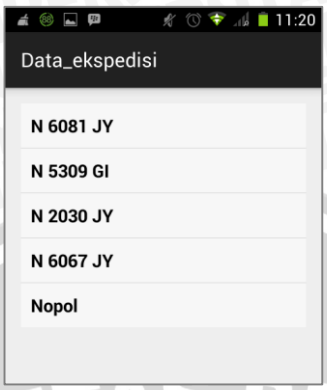
Tombol selanjutnya adalah tombol data ekspedisi. Antarmuka halaman data ekspedisi merupakan halaman yang digunakan untuk melihat informasi data wajib

pajak yang telah ditambahkan. Pada halaman data ekspedisi akan menampilkan logo aplikasi dan data wajib pajak yang tersimpan. Data yang tersimpan akan ditampilkan dalam bentuk *list* data yang berisi nomor polisi, alamat, status kendaraan, tanggal cetak, tanggal terima, masa berlaku, *longitude* dan *latitude*. Berikut adalah *script layout* pada halaman data ekspedisi ditunjukkan pada Kode 5. 6.

```
<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android"

    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context="com.example.ekspedisispkpkb.Data_ekspedisi"
    android:background="@drawable/background">

    <ListView
        android:id="@+id/lv_data_ekspedisi"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_alignParentTop="true" >

    </ListView>

</RelativeLayout>
```

Kode 5. 6 *Script layout* untuk menampilkan halaman data ekspedisi

Halaman data ekspedisi akan menampilkan data yang tersimpan dalam bentuk *list view* berdasarkan nomor polisi wajib pajak. Untuk melihat data ekspedisi lebih detail petugas lapangan bisa menekan salah satu nomor polisi dan aplikasi akan menampilkan detail data ekspedisi wajib pajak. Detail data ekspedisi wajib pajak berisi data tentang wajib pajak sesuai dengan informasi yang dimasukkan dan dikirimkan ke aplikasi *web*. Gambar 5. 5 menunjukan antarmuka halaman data ekspedisi dan Gambar 5. 6 menunjukan antarmuka halaman detail data ekspedisi.



Gambar 5. 5 Antarmuka halaman data ekspedisi

Untuk melihat data dengan lebih detail dapat menekan salah satu nomor polisi yang tersimpan dan aplikasi akan menampilkan detail data ekspedisi yang tersimpan tersebut. Kode 5.8 merupakan kode untuk menampilkan detail data ekspedisi.

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

            databaseManager db = new databaseManager(this);
            //db.DeleteData();
            Log.d("Total: ", String.valueOf(db.getDataCount()));
        List<DataListEkspedisi> data = db.getAllData();
        final ListView lv = (ListView)
findViewById(R.id.lv_data_ekspedisi);
        DataEkspedisiAdapter de = new DataEkspedisiAdapter(this,
R.layout.item_per_data, data);
        lv.setAdapter(de);
        lv.setOnItemClickListener(new OnItemClickListener() {
            @Override
                public void onItemClick(AdapterView<?> a, View v, int
position, long id) {
                        Object o = lv.getItemAtPosition(position);
                DataListEkspedisi obj_DataBarang =
(DataListEkspedisi)o;

                        // Creating an intent to open the uri
                        Intent intent = new
Intent(Data_ekspedisi.this, Detail_ekspedisi.class);
                        intent.putExtra("tipe","1");

                        intent.putExtra("nopol",
obj_DataBarang.getNopol());
```

```
                                intent.putExtra("status",
obj_DataBarang.getStatus_kendaraan());

                                intent.putExtra("alamat",
obj_DataBarang.getAlamat());
                                intent.putExtra("tgl_cetak",
obj_DataBarang.getTgl_cetak());
                                intent.putExtra("tgl_terima",
obj_DataBarang.getTgl_terima());
                                intent.putExtra("lat_long",
obj_DataBarang.getLatitude()+","+obj_DataBarang.getLongitude());

                                startActivity(intent);
                        }});
```

Kode 5. 7 *Script* untuk menampilkan halaman detail data ekspedisi

Kode 5. 7 adalah kode untuk menampilkan detail data ekspedisi yang tersimpan pada *database* aplikasi Android.

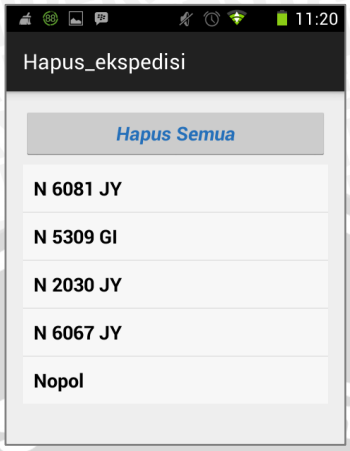
Gambar 5. 6 Antarmuka halaman detail data ekspedisi

Gambar 5. 6 merupakan antarmuka halaman detail data ekspedisi. Pada halaman detail data ekspedisi terdapat informasi wajib pajak yang tersimpan pada *database* aplikasi Android. Tombol terakhir adalah tombol hapus ekspedisi. Antarmuka halaman hapus ekspedisi merupakan halaman yang digunakan untuk

menghapus data ekspedisi yang tersimpan dalam *database* aplikasi Android. Pada halaman ini terdapat *list view* data ekspedisi berdasarkan nomor polisi yang dapat dihapus satu persatu dan juga tombol hapus semua untuk menghapus semua data ekspedisi yang tersimpan. Berikut adalah *script layout* pada halaman hapus ekspedisi ditunjukkan pada Kode 5. 8.

```
<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@drawable/background"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context="com.example.ekspedisispkpkb.Hapus_ekspedisi" >

    <Button
        android:id="@+id/bt_hapus_all"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:text="Hapus Semua"
        android:onClick="bthapusall"
        />

    <ListView
        android:id="@+id/lv_hapus_ekspedisi"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@+id/bt_hapus_all"
        android:layout_marginTop="5dp"
        android:layout_centerHorizontal="true" >

    </ListView>

</RelativeLayout>
```

Kode 5. 8 *Script layout* untuk menampilkan halaman hapus ekspedisi



Gambar 5. 7 Antarmuka halaman hapus data ekspedisi

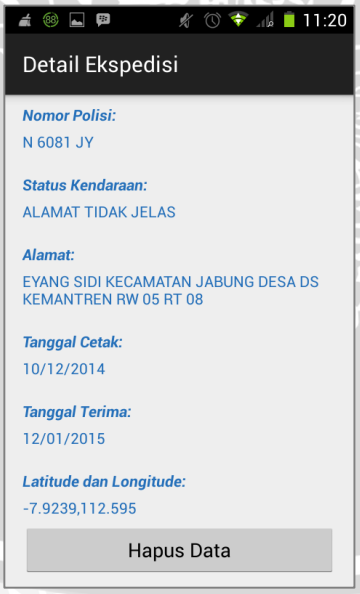
Halan hapus ekspedisi akan menampilkan data ekspedisi yang akan dihapus dalam *list view* sedangkan halaman detail hapus data ekspedisi menampilkan detail data ekspedisi yang akan dihapus. Gambar 5. 7 menunjukan antarmuka halaman hapus ekspedisi dan Gambar 5. 8 menunjukkan antarmuka halaman detail hapus data ekspedisi. Untuk melihat data yang akan dihapus dengan lebih detail dapat menekan salah satu nomor polisi yang tersimpan dan aplikasi akan menampilkan detail data ekspedisi yang tersimpan tersebut. Kode 5. 9 merupakan kode untuk menampilkan detail hapus data ekspedisi

```
public class Detail_ekspedisi extends ActionBarActivity {
        Intent data;
        TextView nopol,jenis, status, nama, alamat,
tgl_cetak,tgl_terima,lat_long;
        String snopol;
        @Override
        protected void onCreate(Bundle savedInstanceState) {
                super.onCreate(savedInstanceState);
                setContentView(R.layout.activity_detail_ekspedisi);
                data=getIntent();
                if(data.getStringExtra("tipe").equals("1")){
                        Button bt= (Button) findViewById(R.id.bt_hapus);
                        bt.setVisibility(View.GONE);
                }else{
                        Button bt= (Button) findViewById(R.id.bt_hapus);
                        bt.setVisibility(View.VISIBLE);
                }
                nopol=(TextView) findViewById(R.id.tx_isi_nopol);
```

```
            status=(TextView)
findViewById(R.id.tx_isi_status_kendaraan);
            alamat=(TextView) findViewById(R.id.tx_isi_alamat);
            tgl_cetak=(TextView) findViewById(R.id.tx_isi_tgl_cetak);
            tgl_terima=(TextView) findViewById(R.id.tx_isi_tgl_terima);
            lat_long=(TextView) findViewById(R.id.tx_isi_lat_long);

            snopol=data.getStringExtra("nopol");
            nopol.setText(data.getStringExtra("nopol"));
            status.setText(data.getStringExtra("status"));
            alamat.setText(data.getStringExtra("alamat"));
            tgl_cetak.setText(data.getStringExtra("tgl_cetak"));
            tgl_terima.setText(data.getStringExtra("tgl_terima"));
            lat_long.setText(data.getStringExtra("lat_long"));
        }
```

Kode 5. 9 *Script* untuk menampilkan halaman detail hapus data ekspedisi

Kode 5. 9 adalah kode untuk menampilkan detail hapus data ekspedisi yang tersimpan pada *database* aplikasi Android. Gambar 5. 8 merupakan antarmuka halaman detail hapus data ekspedisi. Pada halaman detail hapus data ekspedisi terdapat informasi wajib pajak dan tombol hapus data untuk menghapus data tersebut.

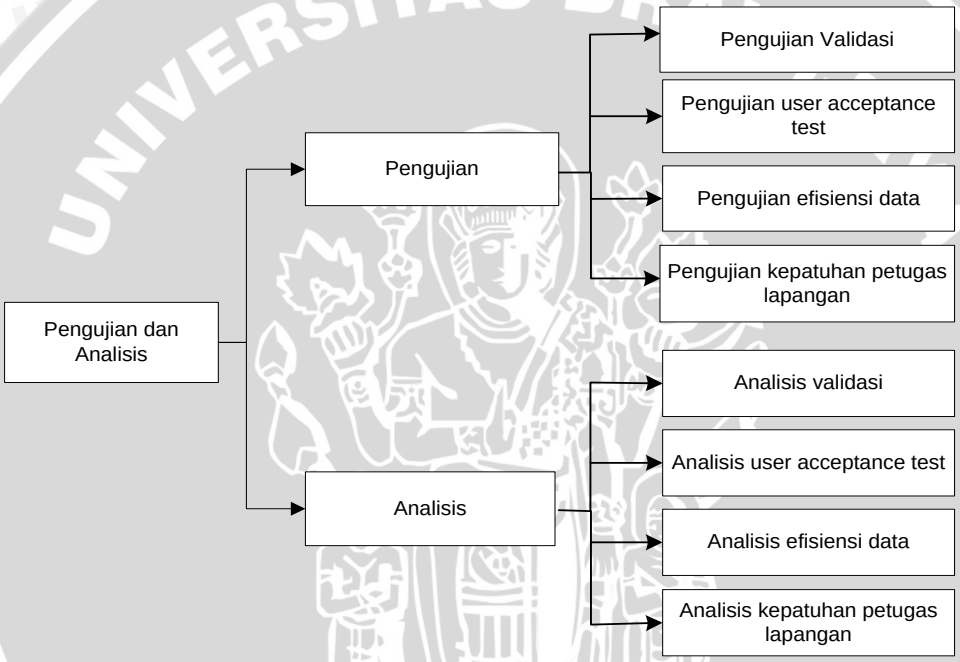


Gambar 5. 8 Antarmuka halaman detail hapus data ekspedisi

BAB VI

PENGUJIAN DAN ANALISIS

Pada bab ini menjelaskan pengujian dan analisis yang dilakukan pada aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS yang telah dibangun. Pengujian yang dilakukan yaitu pengujian validasi, pengujian *user acceptance test*, pengujian efisiensi data dan pengujian kepatuhan petugas lapangan.



Gambar 6. 1 Diagram Alir Pengujian dan Analisis

6.1 Pengujian

Pengujian dilakukan untuk membangun tingkat keyakinan akan kualitas dari aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS yang akan dikembangkan. Pengujian yang dilakukan pada sistem ini ialah pengujian validasi dengan *black-box testing*, pengujian *user acceptance test* dengan menggunakan kuisioner, pengujian efisiensi data dan pengujian kepatuhan petugas lapangan.

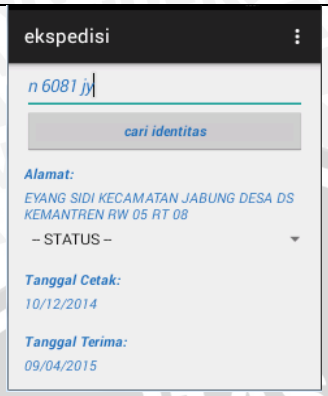
6.1.1 Pengujian Validasi

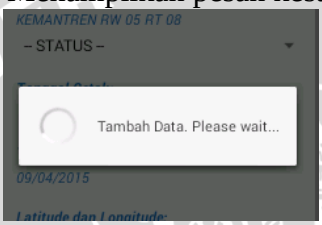
Pengujian validasi menggunakan metode Black Box dimana pengujian yang dilakukan adalah pengujian fungsionalitas dari sistem, apakah sistem berfungsi dengan hasil yang diinginkan atau tidak. Berikut tabel hasil pengujian validasi sistem.

a. Kasus uji menambah data ekspedisi

Tabel 6. 1 Kasus uji menambah data ekspedisi

Nama Kasus Uji	Kasus menambah data ekspedisi
Objek Uji	Kebutuhan Fungsional (KF_1_01)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa *aplikasi Android* yang dibangun dapat memenuhi kebutuhan untuk menambahkan data ekspedisi terbaru wajib pajak
Data Masukan	1. Nomor polisi N 6081 JY 2. Alamat eyang sidi jabung ds kemantren 05 08 3. Status kendaraan Alamat tidak jelas 4. Tanggal cetak 10-12-2014 5. Tanggal terima 18-12-2015 6. Masa berlaku 4-01-2015 7. Longitude -7,9239 dan latitude 112,595 8. Klik tombol kirim
Prosedur Uji	1. Petugas lapangan menjalankan aplikasi Android 2. Petugas lapangan memasukkan data pada aplikasi Android berupa nomor polisi 3. Petugas lapangan menekan tombol cari identitas 4. Petugas lapangan memilih status kendaraan alamat tidak jelas 5. Petugas lapangan menekan tombol cari koordinat 6. Petugas lapangan mengirim data ke aplikasi *web*
Hasil Yang Diharapkan	Data ekspedisi wajib pajak nomor polisi N 6081 JY, alamat eyang sidi jabung ds kemantren 05 08, status kendaraan Alamat tidak jelas, tanggal cetak 10-12-2014, tanggal terima 18-12-2015, masa berlaku 4-01-2015, longitude -7,9239 dan latitude 112,595 berhasil disimpan di *database* dan aplikasi Android menampilkan pesan bahwa data ekspedisi berhasil ditambahkan
Hasil	1. Data ekspedisi wajib pajak dapat dimasukkan

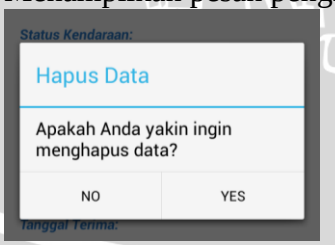
	 2. Menampilkan notifikasi data ekspedisi berhasil disimpan
Status Validasi	Valid
Alur alternatif: Jika data ekspedisi tidak berhasil disimpan	
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa *aplikasi Android* yang dibangun dapat menampilkan pesan kesalahan ketika data ekspedisi tidak dapat disimpan
Data Masukan	1. Nomor polisi N 6081 JY 2. Alamat eyang sidi jabung ds kemantren 05 08 3. Status kendaraan Alamat tidak jelas 4. Tanggal cetak 10-12-2014 5. Tanggal terima 18-12-2015 6. Masa berlaku 4-01-2015 7. Longitude -7,9239 dan latitude 112,595 8. Klik tombol kirim
Prosedur Uji	1. Petugas lapangan menjalankan aplikasi Android 2. Petugas lapangan memasukkan data pada aplikasi Android berupa nomor polisi 3. Petugas lapangan menekan tombol cari identitas 4. Petugas lapangan memilih status kendaraan alamat tidak jelas 5. Petugas lapangan menekan tombol cari koordinat 6. Petugas lapangan mengirim data ke aplikasi *web*
Hasil Yang Diharapkan	Aplikasi Android menampilkan pesan kesalahan saat data ekspedisi tidak berhasil ditambahkan
Hasil	1. Data ekspedisi wajib pajak dapat dimasukkan

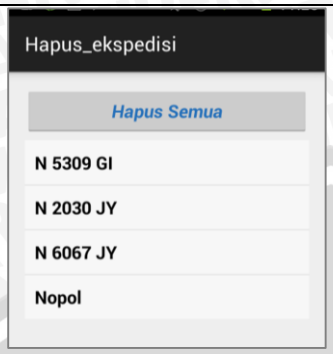
	 2. Menampilkan pesan kesalahan 
Status Validasi	Valid

b. Kasus uji menghapus data ekspedisi

Tabel 6. 2 Kasus uji menghapus data ekspedisi

Nama Kasus Uji	Kasus uji menghapus data ekspedisi
Objek Uji	Kebutuhan Fungsional (KF_1_02)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi Android yang dibangun dapat memenuhi kebutuhan untuk menghapus data ekspedisi wajib pajak
Data Masukan	1. Klik tombol hapus ekspedisi
Prosedur Uji	1. Petugas lapangan menjalankan aplikasi Android 2. Petugas lapangan masuk kehalaman hapus data ekspedisi 3. Petugas lapangan memilih data yang akan dihapus berdasarkan nomor polisi wajib pajak N 6081 JY 4. Aplikasi Android menampilkan detail hapus data ekspedisi 5. Aplikasi Android menampilkan pesan penghapusan 6. Petugas lapangan menekan tombol hapus data ekspedisi
Hasil Yang Diharapkan	Data ekspedisi wajib pajak berhasil dihapus dari *database*

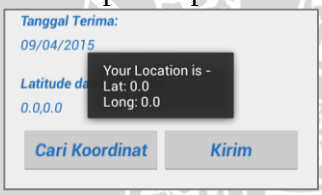
Hasil	1. Memilih data ekspedisi yang akan dihapus berdasarkan nomor polisi N 6081 JY 2. Menampilkan detail hapus data ekspedisi 3. Menampilkan pesan penghapusan  4. Data ekspedisi berhasil dihapus

	
Status Validasi	Valid

c. Kasus uji menambah data koordinat

Tabel 6. 3 Kasus uji menambah data koordinat

Nama Kasus Uji	Kasus uji menambah data koordinat
Objek Uji	Kebutuhan Fungsional (KF_1_03)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi Android yang dibangun dapat memenuhi kebutuhan untuk menambahkan koordinat lokasi wajib pajak
Data Masukan	1. Klik tombol cari koordinat
Prosedur Uji	1. Petugas lapangan menjalankan aplikasi Android 2. Petugas lapangan masuk kehalaman tambah data ekspedisi 3. Petugas lapangan menekan tombol cari koordinat
Hasil Yang Diharapkan	Aplikasi Android akan menampilkan koordinat *longitude* 112,5948787 dan *latitude* -7,9237699 lokasi wajib pajak
Hasil	1. Menampilkan pesan koordinat berhasil didapatkan 2. Koordinat *longitude* dan *latitude* berhasil ditambahkan

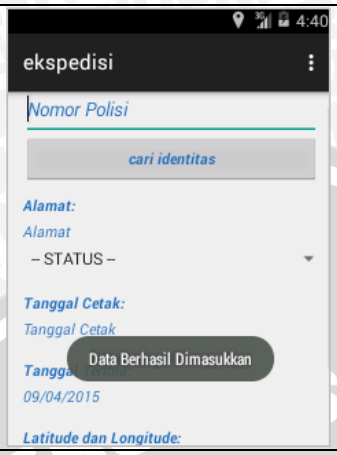
Status Validasi	Valid
Alur alternatif: Jika koordinat lokasi tidak berhasil didapatkan	
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi Android yang dibangun dapat menampilkan pesan kesalahan ketika koordinat lokasi tidak berhasil didapatkan
Data Masukan	1. Klik tombol cari koordinat
Prosedur Uji	1. Petugas lapangan menjalankan aplikasi Android 2. Petugas lapangan masuk kehalaman tambah data ekspedisi 3. Petugas lapangan menekan tombol cari koordinat
Hasil Yang Diharapkan	Aplikasi Android menampilkan pesan kesalahan koordinat *latitude* 0,0 dan *longitude* 0,0 saat koordinat lokasi wajib pajak tidak berhasil didapatkan
Hasil	1. Menampilkan pesan kesalahan 
Status Validasi	Valid

d. Kasus uji mengirim data ekspedisi ke aplikasi *web*

Tabel 6. 4 Kasus uji mengirim data ekspedisi ke aplikasi *web*

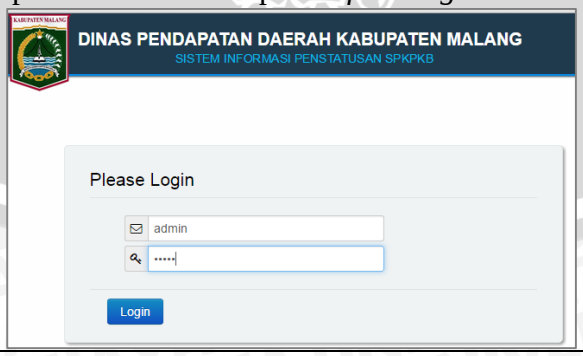
Nama Kasus Uji	Kasus uji mengirim data ekspedisi ke aplikasi *web*
Objek Uji	Kebutuhan Fungsional (KF_1_04)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi Android yang dibangun dapat memenuhi kebutuhan untuk mengirim data ekspedisi ke aplikasi *web*
Data Masukan	1. Nomor polisi N 6081 JY 2. Alamat eyang sidi jabung ds kemantren 05 08 3. Status kendaraan Alamat tidak jelas

	4. Tanggal cetak 10-12-2014 5. Tanggal terima 18-12-2015 6. Masa berlaku 4-01-2015 7. Longitude -7,9239 dan latitude 112,595 8. Klik tombol kirim
Prosedur Uji	1. Petugas lapangan menjalankan aplikasi Android 2. Petugas lapangan memasukkan data pada aplikasi Android berupa nomor polisi 3. Petugas lapangan menekan tombol cari identitas 4. Petugas lapangan memilih status kendaraan alamat tidak jelas 5. Petugas lapangan menekan tombol cari koordinat 6. Petugas lapangan mengirim data ke aplikasi *web*
Hasil Yang Diharapkan	Aplikasi Android akan menampilkan pesan data ekspedisi berhasil dikirim
Hasil	1. Data ekspedisi wajib pajak dapat dimasukkan 2. Menekan tombol kirim 3. Menampilkan pesan data ekspedisi berhasil dikirim

	
Status Validasi	Valid

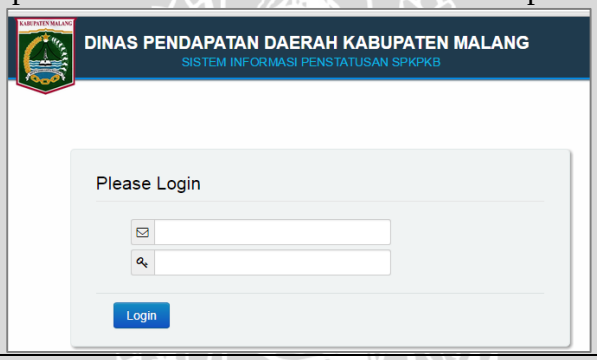
e. Kasus uji melakukan *login*

Tabel 6. 5 Kasus uji melihat data petugas lapangan

Nama Kasus Uji	Kasus melakukan *login administrator*
Objek Uji	Kebutuhan Fungsional (KF_2_01)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi *web* yang dibangun dapat memenuhi kebutuhan untuk melakukan *login administrator*
Data Masukan	1. Membuka halaman aplikasi *web*
Prosedur Uji	1. *Administrator* menuliskan *username* dan *password* 2. *Administrator* menekan tombol *login*
Hasil Yang Diharapkan	aplikasi *web* menampilkan halaman utama aplikasi *web*
Hasil	1. Aplikasi *web* menampilkan *form login* 
Status Validasi	Valid

f. Kasus uji melakukan *logout*

Tabel 6. 6 Kasus uji melihat data petugas lapangan

Nama Kasus Uji	Kasus melakukan *logout administrator*
Objek Uji	Kebutuhan Fungsional (KF_2_02)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi *web* yang dibangun dapat memenuhi kebutuhan untuk melakukan *logout administrator*
Data Masukan	1. *Administrator* menekan tombol *logout*
Prosedur Uji	1. *Administrator* menekan tombol *logout*
Hasil Yang Diharapkan	aplikasi *web* akan keluar dari halaman utama aplikasi *web*
Hasil	2. Aplikasi *web* keluar dari halaman utama aplikasi *web* 
Status Validasi	Valid

g. Kasus uji melihat data petugas lapangan

Tabel 6. 7 Kasus uji melihat data petugas lapangan

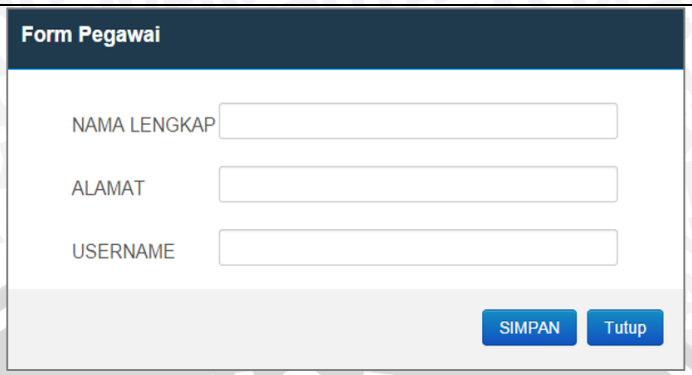
Nama Kasus Uji	Kasus melihat data petugas lapangan
Objek Uji	Kebutuhan Fungsional (KF_2_03)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi *web* yang dibangun dapat memenuhi kebutuhan untuk melihat data petugas lapangan
Data Masukan	2. Klik tombol petugas lapangan
Prosedur Uji	3. *Administrator* menjalankan aplikasi *web* 4. *Administrator* masuk ke halaman data petugas lapangan

Hasil Yang Diharapkan	aplikasi *web* dapat menampilkan halaman data petugas lapangan
Hasil	3. *Administrator* masuk ke halaman data petugas lapangan
Status Validasi	Valid

h. Kasus uji membuat akun petugas lapangan

Tabel 6. 8 Kasus uji membuat akun petugas lapangan

Nama Kasus Uji	Kasus membuat akun petugas lapangan
Objek Uji	Kebutuhan Fungsional (KF_2_04)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi *web* yang dibangun dapat memenuhi kebutuhan untuk membuat akun petugas lapangan
Data Masukan	1. Klik tombol buat akun 2. Nama petugas lapangan Mian 3. Alamat Jl. Blimbing No. 51 4. *Username* petugas lapangan **** 5. Klik simpan
Prosedur Uji	*1. Administrator* menjalankan aplikasi *web* *2. Administrator* masuk ke halaman data petugas lapangan *3. Administrator* menekan tombol buat akun *4. administrator* memasukkan data petugas lapangan berupa nama, alamat dan *username* *5. Administrator* menyimpan data petugas lapangan
Hasil Yang Diharapkan	Akun petugas lapangan berhasil disimpan di *database* aplikasi *web*
Hasil	*1. Administrator* menekan tombol buat akun

	 4. *Administrator* menyimpan akun
Status Validasi	Valid

i. Kasus uji melihat informasi data wajib pajak

Tabel 6. 9 Kasus uji melihat informasi data wajib pajak

Nama Kasus Uji	Kasus melihat informasi data wajib pajak
Objek Uji	Kebutuhan Fungsional (KF_2_05)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi *web* yang dibangun dapat memenuhi kebutuhan untuk melihat informasi data wajib pajak
Data Masukan	1. Klik tombol data wajib pajak
Prosedur Uji	1. *Administrator* menjalankan aplikasi *web* 2. *Administrator* masuk ke halaman data wajib pajak
Hasil Yang Diharapkan	aplikasi *web* dapat menampilkan halaman data wajib pajak
Hasil	1. *Administrator* masuk ke halaman data wajib pajak

	 NO | NAMA | NOPOL | ALAMAT | TANGGAL CETAK | STATUS | TANGGAL TERIMA | MASA BERLAKU 345 | N 6067 JY | KONANG | 11/12/2014 | DIMILIKI | 19/12/2014 | 05/01/2015 346 | N 6687 GH | KONANG | 11/12/2014 | DIMILIKI | 19/12/2014 | 05/01/2015 348 | N 2708 J | PAHLAWAN LURING | 11/12/2014 | DIMILIKI | 19/12/2014 | 05/01/2015 349 | N 6675 JO | IMAM BONJOL | 11/12/2014 | DIMILIKI | 19/12/2014 | 05/01/2015 350 | N 2581 HM | GAJAH MADA | 11/12/2014 | DIMILIKI | 19/12/2014 | 05/01/2015 351 | N 5352 GI | BAYANG | 11/12/2014 | DIMILIKI | 19/12/2014 | 05/01/2015 352 | N 2137 JY | DS PANDANSARI LOR RW 02/16 | 11/12/2014 | DIMILIKI | 19/12/2014 | 05/01/2015 353 | N 5061 GI | BEGAWAN | 11/12/2014 | DIMILIKI | 19/12/2014 | 05/01/2015 356 | N 6701 JO | PATEGUHAN | 11/12/2014 | DIMILIKI | 19/12/2014 | 05/01/2015 357 | N 6968 HS | DS ARGOSARI RW 02/11 | 11/12/2014 | DIMILIKI | 19/12/2014 | 05/01/2015
Status Validasi	Valid

j. Kasus uji melihat informasi data ekspedisi

Tabel 6. 10 Kasus uji melihat informasi data ekspedisi

Nama Kasus Uji	Kasus melihat informasi data ekspedisi
Objek Uji	Kebutuhan Fungsional (KF_2_06)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi *web* yang dibangun dapat memenuhi kebutuhan untuk melihat informasi data ekspedisi
Data Masukan	1. Klik tombol data ekspedisi
Prosedur Uji	3. *Administrator* menjalankan aplikasi *web* 4. *Administrator* masuk ke halaman data ekspedisi
Hasil Yang Diharapkan	aplikasi *web* dapat menampilkan halaman data ekspedisi
Hasil	1. *Administrator* masuk ke halaman data ekspedisi NO | NAMA | NOPOL | STATUS | ALAMAT | TANGGAL CETAK | TANGGAL TERIMA | MASA BERLAKU | PETA 345 | N 6067 JY | DIMILIKI | KONANG JABUNG DS SIDOREJO 04 01 | 11/12/2014 | 19/12/2014 | 05/01/2015 | MAPS 346 | N 6687 GH | DIMILIKI | KONANG JABUNG DS SIDOREJO 04 07 | 11/12/2014 | 19/12/2014 | 05/01/2015 | MAPS 348 | N 2708 J | DIMILIKI | PAHLAWAN LURING JABUNG DS SUKOPURO 02 05 | 11/12/2014 | 19/12/2014 | 05/01/2015 | MAPS 349 | N 6675 JO | DIMILIKI | IMAM BONJOL JABUNG DS SUKOPURO 09 27 | 11/12/2014 | 19/12/2014 | 05/01/2015 | MAPS 350 | N 2581 HM | DIMILIKI | GAJAH MADA JABUNG DS SUKOPURO 10 38 | 11/12/2014 | 19/12/2014 | 05/01/2015 | MAPS 351 | N 5352 GI | DIMILIKI | BAYANG JABUNG DS PANDANSARI LOR 01 11 | 11/12/2014 | 19/12/2014 | 05/01/2015 | MAPS 352 | N 2137 JY | DIMILIKI | DS PANDANSARI LOR RW 02/16 JABUNG DS PANDANSARI LOR 02 16 | 11/12/2014 | 19/12/2014 | 05/01/2015 | MAPS 353 | N 5061 GI | DIMILIKI | BEGAWAN JABUNG DS PANDANSARI LOR 03 26 | 11/12/2014 | 19/12/2014 | 05/01/2015 | MAPS 356 | N 6701 JO | DIMILIKI | PATEGUHAN JABUNG DS ARGOSARI 01 05 | 11/12/2014 | 19/12/2014 | 05/01/2015 | MAPS
Status Validasi	Valid

6.1.2 Pengujian *User Acceptance Test*

Pengujian *user acceptance test* adalah pengujian yang dilakukan untuk mengetahui seberapa besar tingkat penerimaan dan kenyamanan pengguna terhadap aplikasi baru yang akan diterapkan. Pengujian *user acceptance test* dilakukan dengan menggunakan kuisioner yang diberikan kepada pengguna sebagai responden untuk mengetahui tingkat penerimaan dan kenyamanan pengguna.

Faktor-faktor dalam kuesioner meliputi *Effective, Easy to learn, Efficient, error handling* dan *Engaging*. *Effective* yaitu menguji seberapa lengkap dan akurat data dalam mencapai tugas atau tujuan. *Easy to learn* yaitu seberapa mudah bagi pengguna ketika menggunakan aplikasi. *Efficient* yaitu menguji seberapa cepat dalam mencapai tugas atau tujuan. *Error handling* yaitu menguji bagaimana aplikasi dapat menangani kesalahan. *Engaging* yaitu menguji semenarik dan senyaman apa ketika pengguna menggunakan aplikasi. Pertanyaan yang diajukan dalam kuisioner ini meliputi tingkat kemudahan dalam penggunaan aplikasi, antarmuka aplikasi, respon aplikasi terhadap fungsi-fungsi yang dikerjakan dan kinerja aplikasi. Hasil dari jawaban responden pengujian *user acceptance test* ditunjukkan pada Tabel 6. 11.

Tabel 6. 11 Jawaban responden pengujian *user acceptance test*

No	Nama	JK	1	2	3	4	5	6	7	8
1	Nurudin Syamsuri	L	5	5	5	5	5	5	5	5
2	Mulyono	L	5	5	4	5	4	4	4	5
3	Tumiarti	P	5	5	4	4	5	4	4	5
4	Veronica yuni hartatik	P	4	4	4	5	5	3	3	5
5	Andik	L	5	4	4	5	4	5	5	5
6	Aditya	L	5	4	5	5	5	5	5	5
7	Giovani cindy fransiska	P	5	5	5	4	5	3	3	4
8	Aryo fajar	L	4	4	5	4	5	3	3	5
9	Marpuah	P	4	4	4	5	5	5	4	5

Responden pada kuisioner *user acceptance test* terdiri dari 9 orang yang terdiri dari 2 *administrator*, 1 koordinator lapangan dan 6 petugas lapangan. Responden adalah anggota pihak ketiga yang bertugas untuk menyampaikan SPKPKB ke tangan wajib pajak sehingga responden sangat cocok untuk menggunakan aplikasi ini. Responden diminta untuk mengisi kuisioner tentang tanggapan pengguna aplikasi Android dengan memberikan tanda silang pada angka 1 sampai 5 pada setiap pertanyaan yang diajukan. Tabel 6. 12 merupakan hasil dari pengujian *user acceptance test*.

Tabel 6. 12 Hasil pengujian *user acceptance test*

No	Pernyataan	Jawaban				
		STS	TS	N	S	SS
Effective						
1	Saya merasa penggunaan aplikasi *mobile* ini sesuai dengan kebutuhan dilapangan	0	0	0	3	6
Easy to learn						
2	Saya dapat mengerti penggunaan aplikasi ini dengan mudah	0	0	0	5	4
Efficient						
3	Fitur atau fungsi- fungsi dalam aplikasi ini dapat merespon dengan cepat dan sesuai	0	0	0	5	4
Engaging						
4	Tampilan aplikasi ini telah cukup menarik	0	0	0	3	6
5	Saya merasa nyaman menggunakan aplikasi ini	0	0	0	1	8
Effective						

6	Aplikasi ini menampilkan informasi yang jelas dan mudah dipahami	0	0	0	2	7
7	Aplikasi ini dapat membantu dalam memberikan status SPKPKB dengan mudah dan cepat	0	0	3	2	4
Error Handling						
8	Saya dapat mengatasi kesalahan selama menggunakan aplikasi ini	0	0	3	3	3

Keterangan :

STS : Sangat Tidak Setuju
TS : Tidak Setuju
N : Netral
S : Setuju
SS : Sangat Setuju

6.1.3 Pengujian Efisiensi Data

Pengujian efisiensi data dilakukan untuk mengetahui seberapa efisien pendistribusian SPKPKB menggunakan aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS. Pendistribusian SPKPKB dihitung sejak SPKPKB dicetak di DISPENDA Kabupaten Malang hingga dilaporkan kembali ke DISPENDA Kabupaten Malang. Pengujian efisiensi dilakukan dengan membandingkan waktu yang digunakan untuk pendistribusian SPKPKB pada metode lama (sebelum menggunakan aplikasi) dengan metode aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS. Perbandingan efisiensi waktu ditunjukkan pada Tabel 6. 13 berikut.

Tabel 6. 13 hasil pengujian efisiensi data

No	Aktifitas	Aktor	Metode Lama	Metode Aplikasi
1	Pencetakan SPKPKB	DISPENDA	2 hari	2 hari
2	Pendistribusian SPKPKB ke pihak ketiga	DISPENDA	1 hari	1 hari
3	Pendistribusian SPKPKB ke koordinator lapangan	Pihak ketiga	1 hari	1 hari

4	Pendistribusian SPKPKB ke petugas lapangan	Koordinator lapangan	1 hari	1 hari
5	Penyampaian SPKPKB ke wajib pajak	Petugas lapangan	7 hari	1 hari
6	Pelaporan SPKPKB ke koordinator lapangan	Petugas lapangan	1 hari	-
7	Pelaporan SPKPKB ke pihak ketiga	Koordinator lapangan	1 hari	-
8	Rekapitulasi laporan	Pihak ketiga	1 hari	-
9	Pelaporan SPKPKB ke DISPENDA	Pihak ketiga	1 hari	1 hari
Jumlah			16 hari	7 hari

Pada pengujian efisiensi data didapatkan data dari metode lama dan data dari metode aplikasi. Pada metode lama aktifitas pencetakan SPKPKB oleh DISPENDA dilakukan selama 2 hari kemudian aktifitas pendistribusian SPKPKB ke pihak ketiga oleh DISPENDA dilakukan selama 1 hari. Dari pihak ketiga akan didistribusikan ke koordinator lapangan selama 1 hari kemudian aktifitas pendistribusian SPKPKB ke petugas lapangan oleh koordinator lapangan selama 1 hari. Dari koordinator lapangan dilanjutkan penyampaian SPKPKB ke wajib pajak oleh petugas lapangan selama 7 hari. Setelah disampaikan ke wajib pajak, SPKPKB akan di laporkan kembali ke koordinator lapangan oleh petugas lapangan selama 1 hari kemudian koordinator lapangan akan melaporkan kembali SPKPKB ke pihak ketiga selama 1 hari. Kemudian aktifitas rekapitulasi laporan SPKPKB oleh pihak ketiga selama 1 hari dilanjutkan pelaporan SPKPKB ke DISPENDA oleh pihak ketiga selama 1 hari. Total jumlah waktu yang diperlukan untuk aktifitas pencetakan sampai dengan pelaporan kembali ke DISPENDA dengan menggunakan metode lama memakan waktu selama 16 hari.

Pada metode aplikasi aktifitas pencetakan SPKPKB oleh DISPENDA dilakukan selama 2 hari kemudian aktifitas pendistribusian SPKPKB ke pihak ketiga oleh DISPENDA dilakukan selama 1 hari. Dari pihak ketiga akan didistribusikan ke koordinator lapangan selama 1 hari kemudian aktifitas pendistribusian SPKPKB ke petugas lapangan oleh koordinator lapangan selama 1 hari. Dari koordinator lapangan dilanjutkan penyampaian SPKPKB ke wajib pajak oleh petugas lapangan selama 1

hari. Setelah SPKPKB disampaikan ke wajib pajak, selanjutnya SPKPKB akan langsung dilaporkan ke DISPENDA sehingga tidak memakan banyak waktu. Total jumlah waktu yang diperlukan untuk aktifitas pencetakan sampai dengan pelaporan kembali ke DISPENDA dengan menggunakan metode aplikasi memakan waktu selama 7 hari. Pada metode lama, pendistribusian dilakukan secara manual sejak SPKPKB dicetak sampai dilaporkan kembali ke DISPENDA sehingga memakan waktu yang lama sedangkan pada metode aplikasi proses pendistribusian manual hanya sampai pada penyampaian SPKPKB ke wajib pajak setelah itu SPKPKB akan dilaporkan ke DISPENDA dengan cara mengirimkan data melalui aplikasi Android sehingga akan menghemat banyak waktu. Perhitungan waktu diatas adalah perhitungan waktu pendistribusian SPKPKB untuk satu wilayah kecamatan sedangkan DISPENDA Kabupaten Malang membawahi 14 Kecamatan di Kabupaten Malang.

6.1.4 Pengujian Kepatuhan Petugas Lapangan

Pengujian kepatuhan petugas lapangan dilakukan untuk mengetahui tingkat penurunan ketidakpatuhan petugas lapangan dalam menyampaikan SPKPKB ke wajib pajak. Pengujian kepatuhan petugas lapangan dilakukan dengan cara membandingkan jumlah SPKPKB yang tersampaikan dan tidak tersampaikan pada metode lama (sebelum menggunakan aplikasi) dan metode aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS. Perbandingan jumlah data kepatuhan petugas lapangan pada metode lama akan ditunjukkan pada Tabel 6. 14 sedangkan perbandingan data kepatuhan petugas lapangan pada metode aplikasi ditunjukkan pada Tabel 6. 15 berikut ini.

Tabel 6. 14 Data kepatuhan petugas lapangan pada metode lama

No	Tanggal Cetak	Jumlah Surat Dicetak	Jumlah Surat Dilaporkan				Jumlah Surat Tidak Dilaporkan
			DM	LJ	ATJ	JML	
1	14/11/2014	84	62	-	10	72	12
2	15/11/2014	70	16	-	44	60	10

3	18/11/2014	17	15	-	2	17	0
4	19/11/2014	64	36	-	9	45	19
5	20/11/2014	27	18	-	9	27	0
6	21/11/2014	44	27	-	11	38	6
7	22/11/2014	60	41	-	11	52	8
Jumlah		366	311				55

Dari data diatas diketahui pengambilan data dilakukan selama 7 kali cetak atau 8 hari. Pada tanggal 14/11/2014 jumlah surat dicetak sebanyak 84 surat, jumlah surat dilaporkan sebanyak 72 surat dengan rincian 62 surat berstatus dimiliki dan 10 surat berstatus alamat tidak jelas sedangkan jumlah surat tidak dilaporkan sebanyak 12 surat. Pada tanggal 15/11/2014 dicetak surat sebanyak 70 surat dan dilaporkan sebanyak 60 surat dengan rincian 16 surat berstatus dimiliki dan 44 surat berstatus alamat tidak jelas sedangkan jumlah surat tidak dilaporkan sebanyak 10 surat. Pada tanggal 18/11/2014 dicetak surat sebanyak 17 surat dan dilaporkan sebanyak 17 surat dengan rincian 15 surat berstatus dimiliki dan 2 surat berstatus alamat tidak jelas. Pada tanggal 19/11/2014 dicetak surat sebanyak 64 surat dan dilaporkan sebanyak 45 surat dengan rincian 36 surat berstatus dimiliki dan 9 surat berstatus alamat tidak jelas sedangkan jumlah surat tidak dilaporkan sebanyak 19 surat.

Pada tanggal 20/11/2014 dicetak surat sebanyak 27 surat dan dilaporkan sebanyak 27 surat dengan rincian 18 surat berstatus dimiliki dan 9 surat berstatus alamat tidak jelas. Pada tanggal 21/11/2014 dicetak surat sebanyak 44 surat dan dilaporkan sebanyak 38 surat dengan rincian 27 surat berstatus dimiliki dan 11 surat berstatus alamat tidak jelas sedangkan jumlah surat tidak dilaporkan sebanyak 6 surat. Pada tanggal 22/11/2014 dicetak surat sebanyak 60 surat dan dilaporkan sebanyak 52 surat dengan rincian 41 surat berstatus dimiliki dan 11 surat berstatus alamat tidak jelas sedangkan jumlah surat tidak dilaporkan sebanyak 8 surat. Total jumlah surat yang dilaporkan kembali adalah 311 surat dan total jumlah surat yang tidak dilaporkan kembali adalah 55 surat.

Tabel 6. 15 Hasil pengujian kepatuhan petugas lapangan pada metode aplikasi

No	Tanggal Cetak	Jumlah Surat Dicetak	Jumlah Surat Dilaporkan				Jumlah Surat Tidak Dilaporkan
			DM	LJ	ATJ	JML	
1	01/12/2014	33	23		10	33	-
2	02/12/2014	59	47	3	4	54	5
3	05/12/2014	61	35	2	24	61	-
4	06/12/2014	54	42	3	7	52	2
5	08/12/2014	60	49		11	60	-
6	10/12/2014	53	51	2		53	-
7	11/12/2014	43	42	1		43	-
Jumlah		363	356				7

Pada Tabel 6. 15 diatas diketahui, pengambilan data dilakukan sebanyak 7 kali cetak atau 10 hari kerja. Pada tanggal 01/12/2014 dicetak surat sebanyak 33 surat dan dilaporkan semua sebanyak 33 surat dengan rincian 23 surat berstatus dimiliki dan 10 surat berstatus alamat tidak jelas. Pada tanggal 02/12/2014 dicetak surat sebanyak 59 surat dan dilaporkan sebanyak 54 surat dengan rincian 47 surat berstatus dimiliki, 3 surat berstatus lapor jual dan 4 surat berstatus alamat tidak jelas sedangkan jumlah surat tidak dilaporkan sebanyak 5 surat. Pada tanggal 05/12/2014 dicetak surat sebanyak 61 surat dan dilaporkan sebanyak 61 surat dengan rincian 35 surat berstatus dimiliki, 2 surat berstatus lapor jual dan 24 surat berstatus alamat tidak jelas. Pada tanggal 06/12/2014 dicetak surat sebanyak 54 surat dan dilaporkan sebanyak 52 surat dengan rincian 42 surat berstatus dimiliki, 3 surat berstatus lapor jual dan 7 surat berstatus alamat tidak jelas sedangkan jumlah surat tidak dilaporkan sebanyak 2 surat.

Pada tanggal 08/12/2014 dicetak surat sebanyak 60 surat dan dilaporkan sebanyak 60 surat dengan rincian 49 surat berstatus dimiliki dan 11 surat berstatus alamat tidak jelas. Pada tanggal 10/12/2014 dicetak surat sebanyak 53 surat dan dilaporkan sebanyak 53 surat dengan rincian 51 surat berstatus dimiliki dan 2 surat berstatus lapor jual. Pada tanggal 11/12/2014 dicetak surat sebanyak 43 surat dan

dilaporkan sebanyak 43 surat dengan rincian 42 surat berstatus dimiliki dan 1 surat berstatus lapor jual. Total jumlah surat yang dilaporkan kembali adalah 356 surat dan total jumlah surat yang tidak dilaporkan kembali adalah 7 surat.

6.2 Analisis

Proses analisis bertujuan untuk mendapatkan kesimpulan dari hasil pengujian aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS yang telah dilakukan. Proses analisis mengacu pada dasar teori sesuai dengan hasil pengujian yang didapatkan. Analisis dilakukan terhadap metode pengembangan perangkat lunak dan hasil pengujian di setiap tahap pengujian. Proses analisis yang dilakukan meliputi analisis metode pengembangan perangkat lunak, analisis hasil pengujian validasi, analisis hasil pengujian *user acceptance test*, analisis hasil pengujian efisiensi data dan analisis hasil pengujian kepatuhan petugas lapangan.

6.2.1 Analisis Hasil Pengujian Validasi

Proses analisis terhadap hasil pengujian validasi dilakukan dengan melihat konformitas antara hasil kinerja sistem dengan daftar kebutuhan. Berdasarkan hasil pengujian validasi dapat disimpulkan bahwa implementasi dan fungsionalitas aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS telah memenuhi kebutuhan yang telah dijabarkan pada tahap analisis kebutuhan.

6.2.2 Analisis Hasil Pengujian *User Acceptance Test*

Proses analisis terhadap hasil pengujian *user acceptance test* dilakukan dengan menggunakan skala Likert. Interpretasi skor Likert atau persentase dari setiap skor Likert ditunjukkan pada Tabel 6. 16. Hasil perhitungan persentase keseluruhan jawaban dari setiap pernyataan ditunjukkan pada Tabel 6. 17. Hasil status pengujian *user acceptance test* ditunjukkan pada Tabel 6. 18 dan Hasil Status pengujian *user acceptance test* berdasarkan faktor-faktor dalam kuesioner Tabel 6. 19.

Tabel 6. 16 Interpretasi Skor Likert

Skor Likert	Interpretasi skor dengan interval = 20	Pilihan
1	0% - 19.99%	Sangat Tidak Setuju
2	20% - 39.99%	Tidak Setuju
3	40% - 59.99%	Netral
4	60% - 79.99%	Setuju
5	80% - 100%	Sangat Setuju

Keterangan:

Interval = 20 didapatkan dari pembagian nilai 100 dengan jumlah skor Likert

Tabel 6. 17 Persentase keseluruhan jawaban

No	Pernyataan	STS	TS	N	S	SS	Total Skor	Index (%)
	Effective							
1	Saya merasa penggunaan aplikasi *mobile* ini sesuai dengan kebutuhan dilapangan	0	0	0	3	6	42	93%
	Easy to learn							
2	Saya dapat mengerti penggunaan aplikasi ini dengan mudah	0	0	0	5	4	40	88%
	Efficient							
3	Fitur atau fungsi- fungsi dalam aplikasi ini dapat merespon dengan cepat dan sesuai	0	0	0	5	4	40	88%
	Engaging							
4	Tampilan aplikasi ini telah cukup menarik	0	0	0	3	6	42	93%

5	Saya merasa nyaman menggunakan aplikasi ini	0	0	0	1	8	44	97%
Effective								
6	Aplikasi ini menampilkan informasi yang jelas dan mudah dipahami	0	0	0	2	7	43	95%
7	Aplikasi ini dapat membantu dalam memberikan status SPKPKB dengan mudah dan cepat	0	0	3	2	4	37	82%
Error Handling								
8	Saya dapat mengatasi kesalahan selama menggunakan aplikasi ini	0	0	3	3	3	36	80%

Keterangan:

STS : Sangat Tidak Setuju **S** : Setuju

TS : Tidak Setuju **SS** : Sangat Setuju

N : Netral

Total Skor = (S_{STS} x 1) + (S_{TS} x 2) + (S_N x 3) + (S_S x 4) + (S_{SS} x 5) …… 1)

Index (%) = Total Skor / Y x 100 ……2)

Y: (Skor Likert tertinggi) x (Jumlah audien)

Tabel 6. 18 Status pengujian *user acceptance test*

Aspek Penilaian	Persentase (%)	Status
Data pada aplikasi sesuai dengan kebutuhan dilapangan	93%	Sangat Setuju

Mudah dalam penggunaan aplikasi	88%	Sangat Setuju
Aplikasi dapat merespon dengan cepat	88%	Sangat Setuju
Tampilan aplikasi ini telah cukup menarik	93%	Sangat Setuju
Aplikasi menampilkan informasi yang jelas	95%	Sangat Setuju
Aplikasi dapat membantu dalam menstatus SPKPKB dengan mudah dan cepat	82%	Sangat Setuju
kemudahan dalam mengatasi kesalahan dalam aplikasi	80%	Sangat Setuju

Tabel 6. 19 Status pengujian *user acceptance test* berdasarkan faktor-faktor dalam kuesioner

Faktor-faktor dalam Kuesioner	**Rata-rata Persentase (%)**	**Status**
Effective	90%	Sangat Setuju
Easy to learn	88%	Sangat Setuju
Efficient	88%	Sangat Setuju
Engaging	95%	Sangat Setuju
Error handling	80%	Sangat Setuju

Kesimpulan :

Dari hasil pengujian *user acceptance test* yang telah dilakukan, bahwa index presentase(%) yang didapat pada setiap pernyataan menunjukkan hasil dengan status sangat setuju. Dengan adanya hasil tersebut menunjukkan bahwa aplikasi layak diterima dengan hasil memuaskan.

6.2.3 Analisis Hasil Pengujian Efisiensi Data

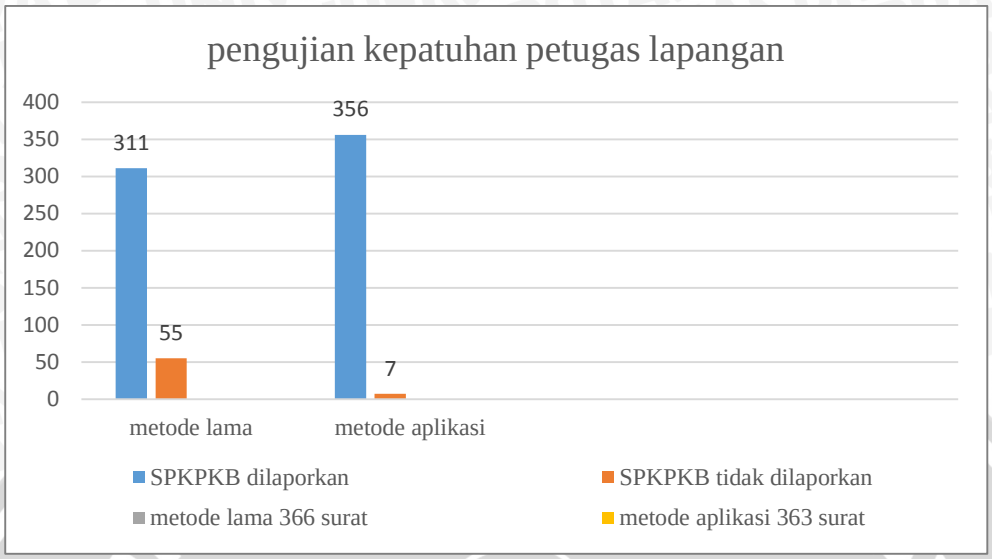
Proses analisis terhadap hasil pengujian efisiensi data dilakukan dengan melihat hasil dari pengujian yang didapatkan. Pengujian efisiensi data dihitung dari lamanya waktu pendistribusian SPKPKB mulai dari pencetakan dikantor DISPENDA kemudian disampaikan ke wajib pajak dan dilaporkan kembali ke kantor DISPENDA sebagai laporan pertanggung jawaban. Dari hasil pengujian efisiensi data, diketahui lama waktu pendistribusian SPKPKB menggunakan metode lama adalah 16 hari sedangkan pendistribusian SPKPKB menggunakan metode aplikasi adalah 7 hari. Dari data tersebut diperoleh selisih jarak antara pendistribusian SPKPKB menggunakan metode lama dan metode aplikasi selama 9 hari. Dengan menggunakan metode aplikasi akan meningkatkan efisiensi data sebesar 228% atau lebih dari 2 kali semula.

6.2.4 Analisis Hasil Pengujian Kepatuhan Petugas Lapangan

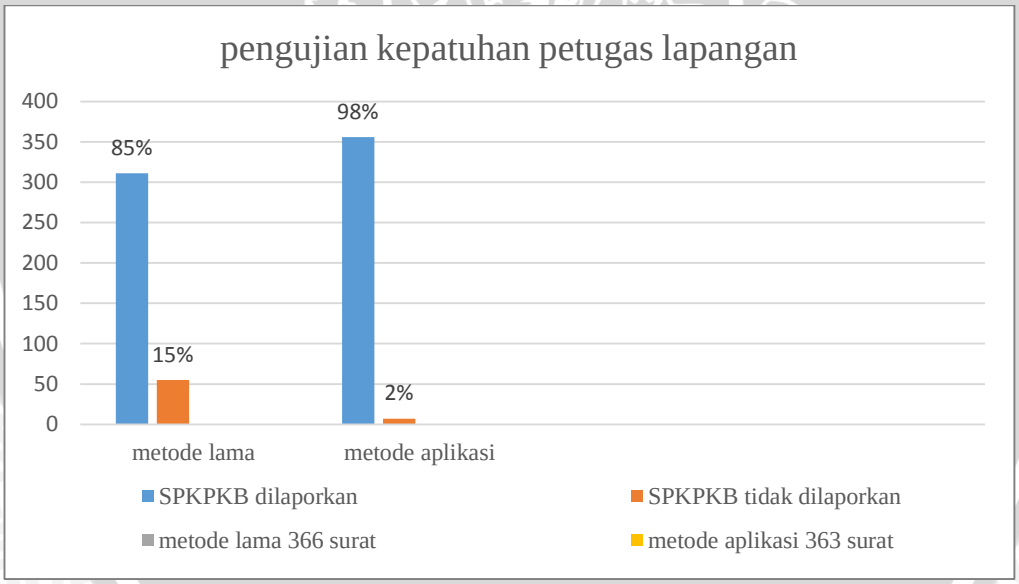
Proses analisis terhadap hasil pengujian kepatuhan petugas lapangan dilakukan dengan menghitung jumlah selisih surat yang dicetak dan dilaporkan kembali ke kantor DISPENDA Kabupaten Malang sebelum dan sesudah menggunakan aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS. Gambar perbandingan jumlah pelaporan SPKPKB pada metode lama dan metode aplikasi Gambar 6. 2 dan gambar perbandingan prosentase pelaporan SPKPKB pada metode lama dan metode aplikasi Gambar 6. 3.

Pada metode lama surat yang dicetak oleh DISPENDA sebanyak 366 surat, dilaporkan sebanyak 311 surat atau aebanyak 85% dan tidak dilaporkan sebanyak 55 surat atau sebanyak 15% sedangkan pada metode aplikasi surat yang dicetak oleh DISPENDA sebanyak 363 surat, dilaporkan sebanyak 356 surat atau sebanyak 89% dan tidak dilaporkan sebanyak 7 surat atau sebanyak 2%.

Dari data diatas dapat diketahui terjadi penurunan ketidakpatuhan terhadap pelaporan SPKPKB sebesar 48 surat atau sebanyak 13%.



Gambar 6. 2 Gambar perbandingan jumlah pelaporan SPKPKB pada metode lama dan metode aplikasi



Gambar 6. 3 Gambar perbandingan prosentase pelaporan SPKPKB pada metode lama dan metode aplikasi

BAB VI
PENUTUP

6.1 Kesimpulan

Berdasarkan hasil analisis kebutuhan, perancangan, implementasi dan pengujian yang dilakukan, maka diambil kesimpulan sebagai berikut:

1. Berdasarkan perancangan proses bisnis aplikasi Android yang dilakukan, didapatkan rancangan yang lebih efisien dalam penstatusan SPKPKB dengan cara memotong aliran tahapan pelaporan SPKPKB dari petugas lapangan ke pihak ketiga.
2. Berdasarkan hasil pengujian *user acceptance test* yang dilakukan dengan mengisi kuesioner dari pengguna, menunjukan rata-rata prosentase berdasarkan beberapa faktor yaitu faktor *effective* dengan rata-rata prosentase adalah 90%, faktor *easy to learn* dengan rata-rata prosentase adalah 88%, faktor *efficient* dengan rata-rata prosentase adalah 88%, faktor *engaging* dengan rata-rata prosentase adalah 95% dan faktor *error handling* dengan rata-rata prosentase adalah 80% sehingga dapat disimpulkan bahwa pengguna merasa nyaman menggunakan aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS.
3. Berdasarkan hasil pengujian efisiensi data yang telah dilakukan, penggunaan aplikasi penstatusan SPKPKB dapat meningkatkan efisiensi data sebesar 228% atau dapat menghemat waktu selama 9 hari daripada menggunakan metode lama.
4. Berdasarkan hasil pengujian kepatuhan petugas lapangan yang telah dilakukan menunjukkan penurunan ketidakpatuhan petugas lapangan dalam melaporkan SPKPKB sebesar 48 surat atau 13% saat menggunakan metode aplikasi Android.

6.2 Saran

Saran yang dapat diberikan untuk pengembangan aplikasi penstatusan SPKPKB berbasis Android menggunakan GPS selanjutnya adalah :

1. Untuk penelitian lebih lanjut, pengembang aplikasi ini dapat bekerja sama dengan DISPENDA Kabupaten Malang untuk mendapatkan izin penuh penggunaan data wajib pajak sehingga dapat menjangkau seluruh wilayah di Kabupaten Malang.
2. Untuk penelitian lebih lanjut, bisa menggunakan smartphone yang menyediakan layanan *block note* untuk tanda tangan wajib pajak agar data lebih akurat.
3. Untuk penelitian lebih lanjut, bisa ditambahkan fitur untuk mengolah data yang telah di inputkan pada aplikasi *web* menjadi laporan pertanggung jawaban bulanan secara otomatis.
4. Untuk penelitian lebih lanjut, bisa ditambahkan juga pada aplikasi *web* fitur untuk melihat peta sesuai dengan koordinat *longitude* dan *latitude* yang telah didapatkan dari aplikasi Android.

DAFTAR PUSTAKA

[AMR-11] Amri, Syaiful. 2011. *Membangun Sistem Navigasi Di Surabaya Menggunakan Google Maps API*. Institut Teknologi Sepuluh Nopember. Surabaya.

[COO-10] Cooper, Martin. 2010. *step by step smartphone android*. Pace University. Massachusetts.

[DAV-93] Davenport, Thomas. 1993. *Process Innovation: Reengineering work through information technology*, Harvard Business School Press. Boston.

[ERL-02] Erly, S. 2002. *Hukum Pajak.* Salemba Empat. Jakarta.

[KUR-05] Kurose, Jim. Ross, Keith. 2005. Computer Networking 3rd Ed. Addison Wesley. USA.

[NOV-10] Y, Novita. 2010. *Analisis Faktor-Faktor yang Mempengaruhi Penerimaan Pajak Kendaraan Bermotor di Jambi*. Universitas Andalas. Padang. Skripsi.

[NUR-05] Nurmantu, S. 2005. *Pengantar Perpajakan*. Edisi ke-3. Granit. Jakarta.

[OFF-13] Official U.S. Government information. 2013. *Global Positioning System (*GPS*) and related topics.*

[PAN-10] Pandiangan, L. 2010. *Hindari Kesalahani Pajak Rakyat Senang jika Anda Patuhi 37 Larangan Perpajakan*. PT Elex Media Komputindo. Jakarta.

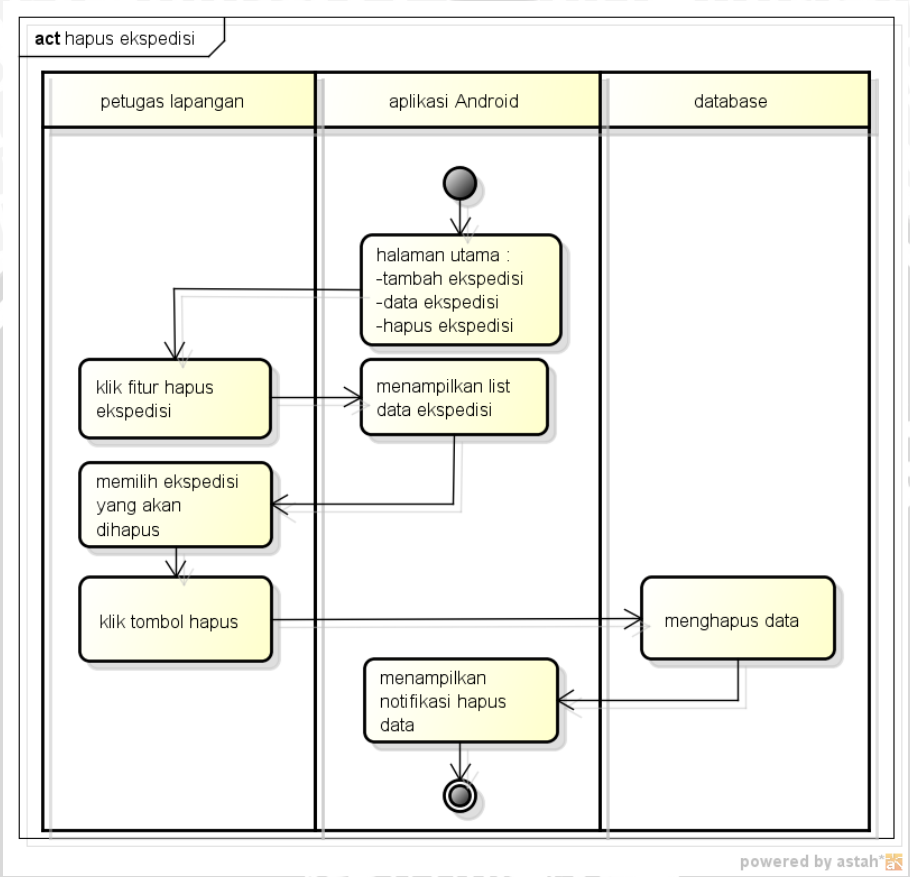
[RIM-04] Rimsyi, K. 2004. *Perpajakan*. PT Gramedia Pustaka Utama. Jakarta.

[SUG-05] Sugianto. 2005. *Pajak dan Retribusi Daerah (Pengelolaan Pemerintah Daerah dalam Aspek Keuangan, Pajak dan Retribusi Daerah)*. PT Gramedia Widiasarana Indonesia. Jakarta.

[YUL-13] Yularsa, N. A. 2013. *Analisis Perilaku Wajib Pajak Terkait dengan Rangkaian Penyampaian Surat Tunggakan Pajak Kendaraan Bermotor*. Universitas Brawijaya. Malang. Skripsi.

LAMPIRAN

Lampiran 1 : *Activity diagram* hapus ekspedisi aplikasi Android



Lampiran 2 : kode data ekspedisi aplikasi Android

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

            databaseManager db = new databaseManager(this);
            //db.DeleteData();
            Log.d("Total: ", String.valueOf(db.getDataCount()));
        List<DataListEkspedisi> data = db.getAllData();
        final ListView lv = (ListView)
findViewById(R.id.lv_data_ekspedisi);
        DataEkspedisiAdapter de = new DataEkspedisiAdapter(this,
R.layout.item_per_data, data);
        lv.setAdapter(de);
        lv.setOnItemClickListener(new OnItemClickListener() {
            @Override
                    public void onItemClick(AdapterView<?> a, View v, int
position, long id) {
                            Object o = lv.getItemAtPosition(position);
                    DataListEkspedisi obj_DataBarang =
(DataListEkspedisi)o;

                                // Creating an intent to open the uri
                                Intent intent = new
Intent(Data_ekspedisi.this, Detail_ekspedisi.class);
                                intent.putExtra("tipe","1");
                                //intent.putExtra("id",
String.valueOf(obj_DataBarang.getID()));
                                intent.putExtra("nopol",
obj_DataBarang.getNopol());
                                //intent.putExtra("jenis",
obj_DataBarang.getJenis_kendaraan());
                                intent.putExtra("status",
obj_DataBarang.getStatus_kendaraan());
                                //intent.putExtra("nama",
obj_DataBarang.getNama());
                                intent.putExtra("alamat",
obj_DataBarang.getAlamat());
                                intent.putExtra("tgl_cetak",
obj_DataBarang.getTgl_cetak());
                                intent.putExtra("tgl_terima",
obj_DataBarang.getTgl_terima());
                                intent.putExtra("lat_long",
obj_DataBarang.getLatitude()+","+obj_DataBarang.getLongitude());
                                // Opening the uri in an activity
                                startActivity(intent);
                    );
```

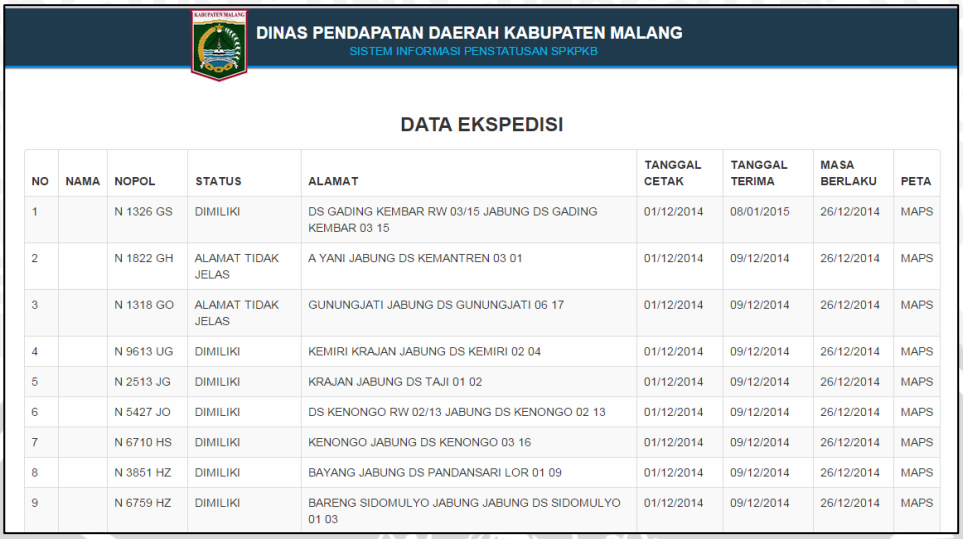
Lampiran 3 : kode data detail ekspedisi aplikasi Android

```
public class Detail_ekspedisi extends ActionBarActivity {
        Intent data;
        TextView nopol,jenis, status, nama, alamat,
tgl_cetak,tgl_terima,lat_long;
        String snopol;
        @Override
        protected void onCreate(Bundle savedInstanceState) {
                super.onCreate(savedInstanceState);
                setContentView(R.layout.activity_detail_ekspedisi);
                data=getIntent();
                if(data.getStringExtra("tipe").equals("1")){
                        Button bt= (Button) findViewById(R.id.bt_hapus);
                        bt.setVisibility(View.GONE);
                }else{
                        Button bt= (Button) findViewById(R.id.bt_hapus);
                        bt.setVisibility(View.VISIBLE);
                }
                nopol=(TextView) findViewById(R.id.tx_isi_nopol);
                status=(TextView)
findViewById(R.id.tx_isi_status_kendaraan);
                alamat=(TextView) findViewById(R.id.tx_isi_alamat);
                tgl_cetak=(TextView) findViewById(R.id.tx_isi_tgl_cetak);
                tgl_terima=(TextView) findViewById(R.id.tx_isi_tgl_terima);
                lat_long=(TextView) findViewById(R.id.tx_isi_lat_long);

                snopol=data.getStringExtra("nopol");
                nopol.setText(data.getStringExtra("nopol"));
                status.setText(data.getStringExtra("status"));
                alamat.setText(data.getStringExtra("alamat"));
                tgl_cetak.setText(data.getStringExtra("tgl_cetak"));
                tgl_terima.setText(data.getStringExtra("tgl_terima"));
                lat_long.setText(data.getStringExtra("lat_long"));
                //databaseManager db = new databaseManager(this);
                //db.DeleteData();
                //Log.d("Total: ", String.valueOf(db.getDataCount()));

        }
```

Lampiran 4 : kode hapus ekspedisi aplikasi Android

```
protected void onCreate(Bundle savedInstanceState) {
            super.onCreate(savedInstanceState);
            setContentView(R.layout.activity_hapus_ekspedisi);
            databaseManager db = new databaseManager(this);
            //db.DeleteData();
            Log.d("Total: ", String.valueOf(db.getDataCount()));
        List<DataListEkspedisi> data = db.getAllData();
        final ListView lv = (ListView)
findViewById(R.id.lv_hapus_ekspedisi);
        de = new DataEkspedisiAdapter(this, R.layout.item_per_data,
data);
        lv.setAdapter(de);
        lv.setOnItemClickListener(new OnItemClickListener() {
            @Override
                    public void onItemClick(AdapterView<?> a, View v, int
position, long id) {
                            Object o = lv.getItemAtPosition(position);
                    DataListEkspedisi obj_DataBarang =
(DataListEkspedisi)o;

                                // Creating an intent to open the uri
                                Intent intent = new
Intent(Hapus_ekspedisi.this, Detail_ekspedisi.class);
                                intent.putExtra("tipe","2");
                                //intent.putExtra("id",
String.valueOf(obj_DataBarang.getID()));
                                intent.putExtra("nopol",
obj_DataBarang.getNopol());
                                //intent.putExtra("jenis",
obj_DataBarang.getJenis_kendaraan());
                                intent.putExtra("status",
obj_DataBarang.getStatus_kendaraan());
                                //intent.putExtra("nama",
obj_DataBarang.getNama());
                                intent.putExtra("alamat",
obj_DataBarang.getAlamat());
                                intent.putExtra("tgl_cetak",
obj_DataBarang.getTgl_cetak());
                                intent.putExtra("tgl_terima",
obj_DataBarang.getTgl_terima());
                                intent.putExtra("lat_long",
obj_DataBarang.getLatitude()+","+obj_DataBarang.getLongitude());
                                // Opening the uri in an activity
                                startActivity(intent);
                                finish();
                    }  });
```

Lampiran 5 : Implementasi Antarmuka halaman data ekspedisi aplikasi *web*

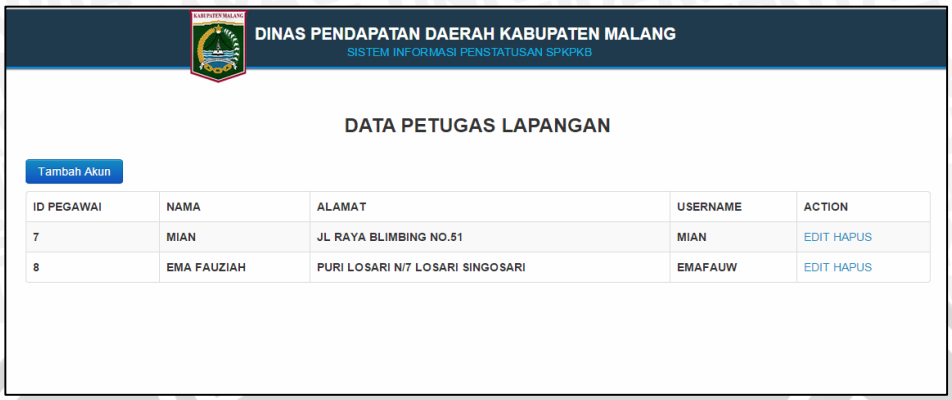


DINAS PENDAPATAN DAERAH KABUPATEN MALANG
SISTEM INFORMASI PENSTATUSAN SPKPKB

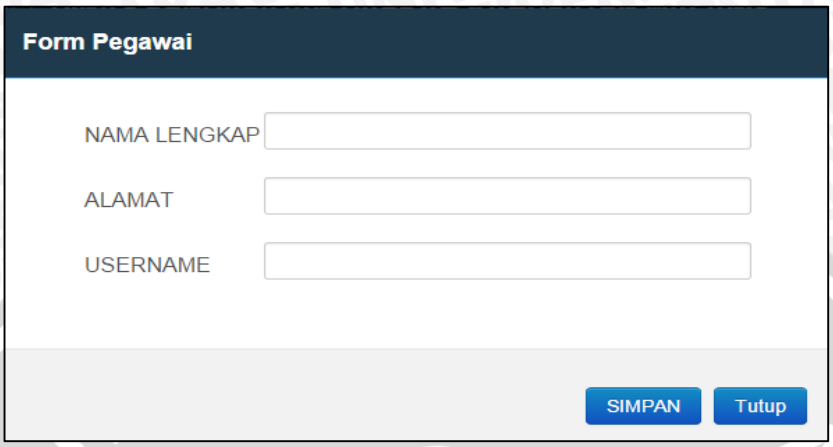
DATA EKSPEDISI

NO	NAMA	NOPOL	STATUS	ALAMAT	TANGGAL CETAK	TANGGAL TERIMA	MASA BERLAKU	PETA
1		N 1326 GS	DIMILIKI	DS GADING KEMBAR RW 03/15 JABUNG DS GADING KEMBAR 03 15	01/12/2014	08/01/2015	26/12/2014	MAPS
2		N 1822 GH	ALAMAT TIDAK JELAS	A YANI JABUNG DS KEMANTREN 03 01	01/12/2014	09/12/2014	26/12/2014	MAPS
3		N 1318 GO	ALAMAT TIDAK JELAS	GUNUNGJATI JABUNG DS GUNUNGJATI 06 17	01/12/2014	09/12/2014	26/12/2014	MAPS
4		N 9613 UG	DIMILIKI	KEMIRI KRAJAN JABUNG DS KEMIRI 02 04	01/12/2014	09/12/2014	26/12/2014	MAPS
5		N 2513 JG	DIMILIKI	KRAJAN JABUNG DS TAJI 01 02	01/12/2014	09/12/2014	26/12/2014	MAPS
6		N 5427 JO	DIMILIKI	DS KENONGO RW 02/13 JABUNG DS KENONGO 02 13	01/12/2014	09/12/2014	26/12/2014	MAPS
7		N 6710 HS	DIMILIKI	KENONGO JABUNG DS KENONGO 03 16	01/12/2014	09/12/2014	26/12/2014	MAPS
8		N 3851 HZ	DIMILIKI	BAYANG JABUNG DS PANDANSARI LOR 01 09	01/12/2014	09/12/2014	26/12/2014	MAPS
9		N 6759 HZ	DIMILIKI	BARENG SIDOMULYO JABUNG JABUNG DS SIDOMULYO 01 03	01/12/2014	09/12/2014	26/12/2014	MAPS

Lampiran 6 : Implementasi Antarmuka halaman data petugas aplikasi *web*



Lampiran 7 : Implementasi Antarmuka halaman buat akun aplikasi *web*



repository.ub.ac
UNIVERSITAS BRAWIJAYA

Lampiran 8 : Implementasi Antarmuka halaman data wajib pajak aplikasi *web*

DINAS PENDAPATAN DAERAH KABUPATEN MALANG
SISTEM INFORMASI PENSTATUSAN SPKPKB

DATA INDUK NOMOR POLISI

NO	NAMA	NOPOL	ALAMAT	TANGGAL CETAK	STATUS	TANGGAL TERIMA	MASA BERLAKU
1		N 1326 GS	DS GADING KEMBAR RW 03/15	01/12/2014	DIMILIKI	08/01/2015	26/12/2014
2		N 1822 GH	A YANI	01/12/2014	ALAMAT TIDAK JELAS	09/12/2014	26/12/2014
3		N 1318 GO	GUNUNGJATI	01/12/2014	ALAMAT TIDAK JELAS	09/12/2014	26/12/2014
4		N 9613 UG	KEMIRI KRAJAN	01/12/2014	DIMILIKI	09/12/2014	26/12/2014
5		N 2513 JG	KRAJAN	01/12/2014	DIMILIKI	09/12/2014	26/12/2014
6		N 5427 JO	DS KENONGO RW 02/13	01/12/2014	DIMILIKI	09/12/2014	26/12/2014
7		N 6710 HS	KENONGO	01/12/2014	DIMILIKI	09/12/2014	26/12/2014
8		N 3851 HZ	BAYANG	01/12/2014	DIMILIKI	09/12/2014	26/12/2014
9		N 6759 HZ	BARENG SIDOMULYO JABUNG	01/12/2014	DIMILIKI	09/12/2014	26/12/2014
10		N 5573 JO	SUMBERKRECO	01/12/2014	DIMILIKI	08/01/2015	26/12/2014

Lampiran 9 : Hasil pengisian Kuesioner

APLIKASI PENSTATUSAN SPKPKB
BERBASIS ADNDROID MENGGUNAKAN GPS

Nama : NURUDIN SYAMSURI
Jenis Kelamin* : Laki-laki / Perempuan (*coret salah satu)
No. Identitas (KTP/SIM/No. Induk) : .....
Alamat : Kemantren Jabung
No. HP : 081334362365

-kuesioner-

Berilah tanda silang (X) pada jawaban yang menurut Anda paling sesuai!

1 = Sangat Tidak Setuju **2 = Tidak Setuju**
3 = Netral **4 = Setuju**
5 = Sangat Setuju

No	Pernyataan	Jawaban				
1	Saya merasa penggunaan aplikasi *mobile* ini sesuai dengan kebutuhan dilapangan	1	2	3	4	X
2	Saya dapat mengerti penggunaan aplikasi ini dengan mudah	1	2	3	4	X
3	Fitur atau fungsi- fungsi dalam aplikasi ini dapat merespon dengan cepat dan sesuai	1	2	3	4	X
4	Tampilan aplikasi ini telah cukup menarik	1	2	3	4	X
5	Aplikasi ini menampilkan informasi yang jelas dan mudah dipahami	1	2	3	4	X
6	Aplikasi ini dapat membantu dalam memberikan status SPKPKB dengan mudah dan cepat	1	2	3	4	X
7	Saya dapat mengatasi kesalahan selama menggunakan aplikasi ini	1	2	3	4	X
8	Saya merasa nyaman menggunakan aplikasi ini	1	2	3	4	X

Lampiran 10: Hasil pengisian Kuesioner

APLIKASI PENSTATUSAN SPKPKB
BERBASIS ADNDROID MENGGUNAKAN GPS

Nama : MULYONO
Jenis Kelamin* : Laki-laki / ~~Perempuan~~ (*coret salah satu)
No. Identitas (KTP/SIM/No. Induk) :
Alamat : Jl. FLAMBOYAN RT. 07 RW 03 DS. SUMBERNOLAGA
No. HP : 089665592055

-kuesioner-

Berilah tanda silang (X) pada jawaban yang menurut Anda paling sesuai!

1 = Sangat Tidak Setuju **2 = Tidak Setuju**
3 = Netral **4 = Setuju**
5 = Sangat Setuju

No	Pernyataan	Jawaban				
1	Saya merasa penggunaan aplikasi *mobile* ini sesuai dengan kebutuhan dilapangan	1	2	3	4	X
2	Saya dapat mengerti penggunaan aplikasi ini dengan mudah	1	2	3	4	X
3	Fitur atau fungsi- fungsi dalam aplikasi ini dapat merespon dengan cepat dan sesuai	1	2	3	X	5
4	Tampilan aplikasi ini telah cukup menarik	1	2	3	4	X
5	Aplikasi ini menampilkan informasi yang jelas dan mudah dipahami	1	2	3	X	5
6	Aplikasi ini dapat membantu dalam memberikan status SPKPKB dengan mudah dan cepat	1	2	3	X	5
7	Saya dapat mengatasi kesalahan selama menggunakan aplikasi ini	1	2	3	X	5
8	Saya merasa nyaman menggunakan aplikasi ini	1	2	3	4	X