

Ketentuan

Dokumen ini merupakan hasil kesepakatan dari pihak-pihak sebagai berikut:

- Disaster Management Innovation (DMInnovation)
- Humanitarian OpenStreetMap Team (HOT)

Seluruh isi dan materi dalam dokumen ini kemungkinan masih dapat mengalami perubahan tanpa pemberitahuan publik.

Lisensi



Seluruh materi dan isi kandungan dalam modul ini adalah milik domain publik. Anda dapat menyalin, memodifikasi, menyebarluaskan, dan menampilkan modul ini, secara keseluruhan tanpa meminta izin.

Keterangan lebih lanjut mengenai lisensi domain publik:

http://wiki.creativecommons.org/Publicdomain/zero/1.0/LegalText_(Indonesian)

Tentang Kami

Disaster Management Innovation (Dahulunya AIFDR)



Disaster Management Innovation adalah investasi bantuan Pemerintah Australia yang terbaru melalui Geoscience Australia di Indonesia. DMInnovation dibangun untuk melanjutkan keberhasilan Australia-Indonesia Facility for Disaster Reduction (AIFDR) sebelumnya yang merupakan inisiatif bersama antara pemerintah Australia dan Indonesia. AIFDR diluncurkan oleh Pemerintah Australia dan Indonesia pada upacara khusus pada tanggal 15 Juli 2010 dan secara resmi ditutup pada bulan Agustus 2015. DMInnovation akan mempertahankan program bantuan teknis ilmiah yang akan digunakan dalam kegiatan kemitraan antara lembaga ilmu pengetahuan Indonesia dan Geoscience Australia.

Humanitarian OpenStreetMap Team (HOT)



Peta gratis dan kolaboratif mempunyai keunikan yang berharga untuk pekerjaan kemanusiaan, terutama di tempat-tempat dimana peta dasar sangat jarang ditemukan, datanya sudah lama, atau seringkali berubah. Dibangun dalam dua tahun dari kolaborasi secara informal, Tim Kemanusiaan OpenStreetMap merupakan sebuah inisiatif yang baru untuk mengaplikasikan prinsip dan aktifitas dari sumber terbuka dan berbagi data terbuka untuk menuju respon kemanusiaan dan pembangunan ekonomi.

http://hot.openstreetmap.org

http://openstreetmap.id

Pengantar

Peta merupakan sarana yang bagus untuk penyampaian informasi. Peta merupakan representasi visual dunia dimana kita tinggal dan informasi merupakan hal penting dalam pengambilan keputusan. Dengan informasi yang baik serta pemahaman yang benar, tiap-tiap individu dan masyarakat dapat menggunakannya untuk membuat keputusan yang lebih baik.

OpenStreetMap (OSM) adalah alat untuk membuat serta berbagi informasi dalam peta. Siapapun dapat berkontribusi di OSM dan ribuan orang akan menambahkan data baru di OSM setiap harinya. Hal yang paling penting dari OSM adalah peta yang anda buat akan disimpan di internet dan siapapun dapat mengaksesnya kapanpun dan dimanapun secara bebas dan terbuka.

Akan tetapi, dengan sifatnya yang terbuka maka OSM dapat menerima data apa saja oleh siapa saja tanpa harus melalui persetujuan dari pihak manapun. Hal ini tentu saja bisa mengakibatkan berkurangnya kualitas data yang ada di OSM.

Oleh karena itu demi menjaga kualitas data OSM khususnya di Indonesia, Humanitarian OpenStreetMap Team (HOT) Indonesia membuat sebuah panduan untuk menjaga kualitas serta validasi data untuk para pengguna OSM. Hal ini bertujuan agar data OSM di Indonesia bisa semakin lebih baik kualitasnya sehingga bisa dengan aman dan 'nyaman' digunakan.

Dalam panduan ini nantinya akan dijelaskan berbagai macam cara untuk bisa menjaga kualitas data yang ada di OSM menggunakan perangkat lunak ataupun website yang menyediakan kemampuan tersebut. Selain itu akan dijelaskan juga bagaimana agar para peserta bisa memaksimalkan pemanfaatan OSM dalam berbagai macam kebutuhan mereka.

Panduan ini nantinya diharapkan dapat menjadi salah satu acuan dalam melakukan validasi serta menjaga kualitas data OSM oleh para pengguna OSM di seluruh Indonesia.

Semoga bermanfaat.

DAFTAR ISI

Ketentuan	1
Lisensi	i
Tentang Kami	ii
Pengantar	iv
Modul 1 Penjaminan Kualitas Data	1
1.1 Penjaminan Kualitas Data	1
Penjaminan Partisipatif	2
Penjaminan Sosial	3
Penjaminan Geografi (Ilmiah)	3
1.2 Penjaminan Kualitas Data OpenStreetMap	4
Modul 2 Kualitas Data OpenStreetMap	6
2.1 Panduan Mendijitasi Objek OpenStreetMap	6
2.1.1 Menggambar Objek	7
AMenampilkan Simbol Objek	7
B. Objek Bangunan dengan lebih dari 1 Jenis Amenitas	11
C. Objek yang Terdiri dari Beberapa Bangunan	12
2.1.2 Pemberian Nama Objek (Toponimi)	15
A. Objek Jalan	17
B. Objek Bangunan	17
2.1.3 Objek dan Penggunaan Lahan yang Saling Berdekatan	18
2.1 4 Menggambar Jalan	22
2.1 5 Menggambar Jembatan	27
2.1.6 Penggambaran Batas Administrasi	29
A. Relation	30
B. Membuat Batas Administrasi	32
2.2 Contoh Kesalahan Data di OpenStreetMap	50
2.2.1 Kesalahan Tag atau Informasi	50
2.2.2 Kesalahan Geometri	55
2.3 Menjaga Kualitas Data di OpenStreetMap	59
2.3.1 Acuan Tag / Informasi di OpenStreetMap	60
2.3.2 Alat Pemantau Data OpenStreetMap	62
2.3.3 Alat Pendeteksi Kesalahan Data OpenStreetMap	73
Modul 3 Validasi Data OpenStreetMap	79
3.1 Validasi Data Menggunakan JOSM	79
3.2 Validasi Data Menggunakan Tasking Manager	83
3.3 Validasi Data Menggunakan Overpass Turbo	86

Modul 1

Penjaminan Kualitas Data

Tujuan Pembelajaran

- Mengetahui pengertian penjaminan kualitas data
- Mengetahui dan memahami kualitas data di OSM
- Memahami pentingnya menjaga kualitas data di OSM

OpenStreetMap (OSM) merupakan salah satu peta dunia yang bersifat bebas dan terbuka. OSM memiliki begitu banyak data dan informasi mengenai berbagai macam objek yang ada di permukaan bumi. Semua itu dikumpulkan dan dimasukkan oleh pengguna OSM yang tersebar di seluruh dunia.

Sejak diluncurkan tahun 2004 oleh Steve Coast, OSM telah menjadi alternatif bahkan pilihan utama bagi masyarakat untuk melengkapi data spasial di wilayah yang mereka inginkan. Kemudahan, kelengkapan serta fleksibilitas dari OSM memungkinkan data-data yang ada di OSM seringkali digunakan kembali untuk berbagai macam keperluan seperti bisnis, teknologi, sosial bahkan kemanusiaan dan kebencanaan.

Hal tersebut menyebabkan meningkatnya jumlah data yang ada di OSM dan pastinya akan mempengaruhi kualitas data itu sendiri. Dengan sifatnya yang gratis dan terbuka, menjaga kualitas data OSM merupakan hal yang sangat penting agar nantinya pemanfaatan data OSM untuk berbagai kebutuhan khususnya yang menyangkut kepentingan masyarakat seperti kemanusiaan dan kebencanaan dapat dilakukan secara baik dan berkualitas.

1.1 Penjaminan Kualitas Data

Menurut buku "Assuring the quality of volunteered geographic information" yang ditulis oleh Goodchild dan Li, penjaminan kualitas data dalam kegiatan pemetaan partisipatif secara garis besar dapat dilakukan dengan 3 mekanisme, yaitu:

- 1. Partisipatif
- 2. Sosial
- 3. Geografi (Ilmiah)

• Penjaminan Partisipatif

Ditambahkan oleh Surowiecki dalam bukunya yang berjudul "*The Wisdom of Crowds*" mekanisme **penjaminan kualitas data** secara **partisipatif** memiliki beberapa ciri dan kelebihan seperti:

- Beberapa pengguna dapat menghasilkan suatu kesepakatan bersama atas suatu kesalahan data yang ditemukan. Hal ini biasanya terjadi seperti kesalahan informasi ataupun jenis-jenis dari objek yang dipetakan di suatu wilayah. Para pengguna yang memetakan wilayah tersebut dapat menghasilkan kesepakatan bersama sehingga tidak ada lagi kesalahan informasi terhadap objek-objek yang akan dipetakan.
- Beberapa pengamatan dan pengalaman dari suatu individu dapat memperkuat validitas pengamatan dan pengalaman individu yang lain sehingga dapat mengurangi risiko kesalahan pemahaman terhadap suatu objek tertentu.
- Secara bersama-sama dapat melakukan validasi dan pengecekan kualitas data dan kesalahan di wilayah tertentu sehingga dapat menghemat waktu, tenaga dan biaya dalam melakukan kegiatan penjaminan kualitas data.

Contoh penjaminan kualitas secara partisipatif ini pernah dilakukan oleh Humanitarian OpenStreetMap Team bekerja sama dengan Resilience Network Initiative (RNI) memetakan Kelurahan Purwodinatan, Kecamatan Semarang Tengah, Kota Semarang. Kegiatan ini melibatkan elemen masyarakat dan dibantu oleh mahasiswa lokal serta panitia sekitar dalam melakukan pemetaan hingga menginput data-data yang dipetakan ke dalam OpenStreetMap. Berikut contoh hasil pemetaan tersebut di OpenStreetMap.



• Penjaminan Sosial

Sedikit berbeda namun tetap berkaitan dengan mekanisme penjaminan kualitas secara partisipatif, penjaminan kualitas secara sosial lebih mengedepankan kualitas individu untuk melakukannya. Semakin sering seseorang melakukan validasi dan memperbaiki kesalahan-kesalahan yang ada pada data maka dia akan semakin dipercaya untuk bisa melakukan penjaminan kualitas data. Dengan demikian, seseorang yang memiliki reputasi yang baik akan dipercaya untuk memimpin yang lain untuk melakukan kegiatan penjaminan kualitas di dalam suatu kegiatan atau proyek. Selain itu, penjaminan kualitas sosial dapat dilakukan dengan membentuk suatu working group yang menjadi sebuah media untuk memberitahukan serta memperbaharui kegiatan-kegiatan yang terkait penjaminan kualitas data. Hal ini dapat mempercepat kegiatan penjaminan kualitas data dan mengefektifkan komunikasi sesama anggota.

Salah satu contoh kegiatan penjaminan kualitas sosial ini seperti yang telah dilakukan oleh *OpenStreetMap Foundation* dimana mereka membuat beberapa kelompok diskusi (*working group*) yang tiap kelompok memiliki tema diskusi yang berbeda-beda. Untuk kualitas data OpenStreetMap dibahas dalam kelompok diskusi data (*Data Working Group*) dimana dalam kelompok tersebut dibahas berbagai hal terkait data-data yang ada di OpenStreetMap seperti lisensi data, vandalisme data, perselisihan tentang data serta membantu dalam menentukan kebijakan pemakaian terhadap data-data OpenStreetMap. Kelompok diskusi data ini terdiri dari anggota OpenStreetMap foundation yang merupakan donor OpenStreetMap dan juga beberapa pengguna yang direkomendasikan oleh anggota yang berada di OpenStreetMap foundation itu sendiri. Untuk melihat tugas dan kegiatan *Data Working Group* silahkan mengunjungi https://wiki.osmfoundation.org/wiki/Data Working Group dan anda dapat menghubungi anggota mereka di data@osmfoundation.org

• Penjaminan Geografi (Ilmiah)

Mekanisme **penjaminan kualitas data** yang terakhir adalah secara **geografi**. Mekanisme ini menggunakan pendekatan teori geografi. Tidak semua pengguna dapat dan boleh untuk melakukan penjaminan kualitas menggunakan mekanisme geografi. Hanya mereka yang sudah benar-benar memahami teori geografi yang terkait dengan analisa data spasial seperti *Spatial neighbors and auto-correlation (Moran Statistics)*, *Inferential Statistics and Analysis of Variance (ANOVA)*, dan lain-lain. Oleh karena itu mekanisme ini masih jarang digunakan dalam kegiatan pemetaan partisipatif khususnya di OpenStreetMap.

Untuk contoh dokumentasi kualitas data OpenStreetMap berdasarkan pendekatan geografi di Indonesia telah dibuat oleh Departemen Teknik Geodesi dan Geomatik, Fakultas Teknik, Univesitas Gadjah Mada

Pada tahun 2012. Dokumentasi ini mengambil sampel data OpenStreetMap di beberapa kota seperti

Jakarta, Surabaya, Bandung, Yogyakarta, Padang dan Dompu dimana pernah menyelenggarakan kegiatan pemetaan OpenStreeMap di Indonesia dari tahun 2011 hingga 2012 mulai dari kegiatan pemetaan sosial di Provinsi Nusa Tenggara Barat khususnya Dompu yang diselenggarakan oleh ACCESS dan Humanitarian OpenStreetMap Team (HOT) hingga *Scenario Development for Contingency Planning* (SD4CP) yang dilakukan oleh Humanitarian OpenStreetMap Team (HOT) dengan Australia Indonesia for Disaster Reduction (AIFDR) dan Badan Nasional Penanggulangan Bencana (BNPB). Dokumentasi kualitas data tersebut dapat anda unduh di:

http://openstreetmap.id/docs/Final_Report-OSM_Evaluation_in_Indonesia_2012.pdf

Final Report

EVALUATION OF OPENSTREETMAP DATA IN INDONESIA

(Case Study: Yogyakarta, Surabaya, Jakarta, Bandung, Padang, and Dompu)



Department of Geodetic and Geomatics Engineering, Faculty of Engineering, UGM

Humanitarian OpenStreetMap Team

2012

1.2 Penjaminan Kualitas Data OpenStreetMap

Dari ketiga mekanisme penjaminan kualitas data yang telah dijelaskan, semuanya digunakan dalam penjaminan kualitas data di OpenStreetMap. Perlu diketahui bahwa kegiatan penjaminan kualitas di OpenStreetMap menjadi tanggung jawab seluruh pengguna. Oleh karena itu, OpenStreetMap telah memiliki beberapa petunjuk teknis dalam melakukan kegiatan tersebut dengan menggunakan ketiga

mekanisme yang telah dijelaskan. Terdapat 2 jenis alat yang digunakan dalam melakukan penjaminan kualitas data di OpenStreetMap, yaitu:

- 1. Alat Pemantau Data OpenStreetMap
- 2. Alat Pendeteksi Kesalahan OpenStreetMap

Alat pemantau data adalah alat yang dapat kita gunakan untuk melihat kondisi data OpenStreetMap yang telah kita buat ataupun data-data OpenStreetMap yang ada di sekitar kita serta di wilayah kegiatan pemetaan yang kita lakukan. Alat pendeteksi kesalahan akan membantu kita menunjukkan kesalahan yang terdapat pada data-data OpenStreetMap sehingga akan mempermudah kita untuk menemukan kesalahan data dibandingkan harus mencari satu persatu secara manual. Beberapa alat tersebut dibuat oleh para pengguna yang mengembangkan OpenStreetMap sesuai dengan kebutuhan mereka. Pada modul 2, kita akan mempelajari beberapa kesalahan pada data OpenStreetMap dan alat-alat terkait dengan penjaminan kualitas data di OpenStreetMap.

Modul 2

Kualitas Data OpenStreetMap

Tujuan Pembelajaran

- Mengetahui cara mendijitasi objek di OSM dengan baik dan benar
- Mengetahui dan memahami contoh-contoh kesalahan data di OSM
- Mengetahui acuan informasi/tag objek di OSM
- Mengetahui alat untuk memantau kualitas data di OSM
- Mengetahui dan memahami cara menggunakan alat-alat untuk memantau kualitas data di OSM
- Mengetahui alat untuk mendeteksi kesalahan data di OSM
- Mengetahui dan memahami cara menggunakan alat untuk mendeteksi kesalahan di OSM

Menjaga kualitas data spasial merupakan hal yang sangat penting terutama jika data-data tersebut merupakan hasil dari kegiatan pemetaan partisipatif seperti OpenStreetMap. Sebagai peta yang data-datanya berasal dari masyarakat, kualitas data OpenStreetMap seringkali dipertanyakan. Oleh karena itu, untuk membuktikan bahwa kualitas data OpenStreetMap dapat dipertanggungjawabkan, partisipasi dari para pengguna untuk tidak hanya memasukkan data ke dalam OpenStreetMap akan tetapi ikut menjaga kualitas data OpenStreetMap juga sangat dibutuhkan. Berikut adalah beberapa hal yang perlu diketahui dimana akan membantu kita untuk ikut menjaga kualitas data di OpenStreetMap.

2.1 Panduan Mendijitasi Objek OpenStreetMap

Sebagai pengguna yang telah cukup sering menggunakan OpenStreetMap baik itu sekedar memasukkan data ataupun juga ikut berpartisipasi dalam kegiatan yang menggunakan OpenStreetMap, kita seringkali menemukan atau bahkan melakukan kesalahan saat melakukan editing baik itu dikarenakan ketidaksengajaan ataupun akibat tidak mengetahui langkah yang benar saat melakukan edit di OpenStreetMap. Berikut adalah beberapa langkah keliru dan kesalahan data yang seringkali terjadi ketika kita melakukan edit di OpenStreetMap:

2.1.1 Menggambar Objek

A. Menampilkan Simbol Objek

Kegiatan menggambar objek di OpenStreetMap seringkali membuat kita bingung. Beberapa pertanyaan yang sering dialami oleh pengguna OpenStreetMap seperti "*Tag* apa yang harus digunakan?" dan "apakah objek tersebut sebaiknya digambarkan sebagai titik atau sebagai bangunan?". Apalagi jika objek tersebut terlihat jelas bentuk bangunannya di citra satelit.

Sebetulnya tidak ada yang benar-benar salah ketika kita hanya menggambarkan suatu objek sebagai bangunan atau hanya sebagai titik saja. Akan tetapi, ada baiknya jika kita mempertimbangkan estetika, keindahan serta kemudahan informasi yang ditampilkan oleh objek-objek yang kita petakan. Untuk lebih jelasnya silahkan bandingkan 2 peta di bawah ini :

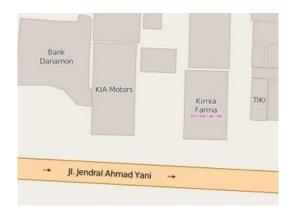




Dari kedua peta tersebut manakah yang menurut anda lebih informatif serta menampilkan informasi secara cepat? Jika anda berpikir peta kedua lebih mudah untuk dilihat dan dipahami maka anda tepat. Salah satu kekuatan dari OpenStreetMap dibandingkan peta yang lain adalah selain sifatnya yang terbuka, OpenStreetMap juga memiliki berbagai macam simbol yang merepresentasikan informasi suatu objek. Hal ini membuat orang yang melihat OpenStreetMap dapat lebih mudah mengenali objek-objek yang ada di area yang dia cari dengan cepat dan tepat.

Kenapa dua peta tersebut terlihat berbeda satu dengan yang lain padahal berada di area yang sama?. Hal ini disebabkan oleh informasi atau *tag* yang dimasukkan ke dalam objek-objek pada masingmasing peta berbeda. Mari kita lihat perbedaannya.

Peta Pertama:



Berikut adalah tag dari beberapa objek yang diinput di peta pertama :

Key	Value	Key	Value
building	yes	building	yes
building:levels	9	building:levels	2
building:roof	concrete	building:roof	tin
building:structure	reinforced_masonry	building:structure	reinforced_masonry
building:use	Bank	building:use	shop
building:walls	brick	building:walls	brick
name	Bank Danamon	name	KIA Motors

building	yes	building	yes
building:levels	2	building:levels	3
building:roof	concrete	building:roof	concrete
building:structure	reinforced_masonry	building:structure	reinforced_masonry
building:use	medical	building:use	commercial
building:walls	brick	building:walls	brick
name	Kimia Farma	name	DALL

Peta Kedua:



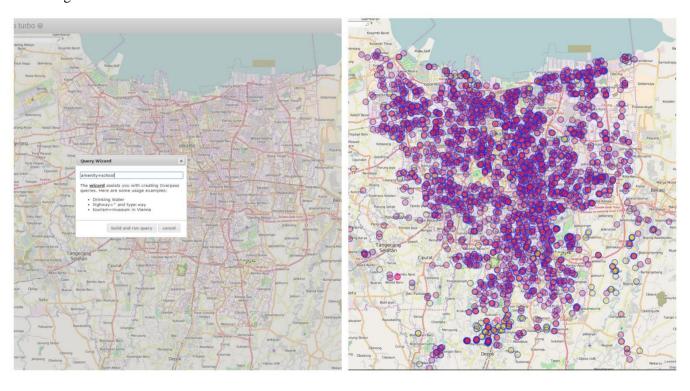
Berikut adalah tag dari objek yang diinput di peta kedua:

Key	Value	Key	Value
amenity	bank	building	yes
building	yes	building:levels	2
building:levels	9	building:roof	tin
building:roof	concrete	building:structure	reinforced masonry
building:structure	reinforced_masonry	building:use	shop
building:use	Bank	building:walls	brick
building:walls	brick	name	KIA Motors
name	Bank Danamon	shop	car
amenity	pharmacy	amenity	post_office
building	yes	building	yes
building:levels	2	building:levels	3
building:roof	concrete	building:roof	concrete
building:structure	reinforced_masonry	building:structure	reinforced_masonry
building:use	medical	building:use	commercial
building:walls	brick	building:walls	brick
name	Kimia Farma	name	TIKI

Apakah anda memperhatikan perbedaannya??

Yup, Betul sekali. Perbedaan *tag* dari kedua peta di atas adalah pada **Peta Pertama** *tag* pada objekobjek yang ada tidak diberikan keterangan spesifik jenis objek yang dipetakan dan hanya diberikan informasi yang terkait dengan *building*_*= saja. Sedangkan di **Peta Kedua**, objek-objek yang dipetakan diberikan keterangan *amenity* = * ataupun *shop*=* dimana semua objek secara spesifik diberikan keterangan bahwa objek tersebut adalah bank atau apotik dan lain-lain. Oleh karena itu, pada Peta Pertama objek-objek yang dipetakan tidak diberikan simbol atau di*render* oleh OpenStreetMap sehingga yang muncul hanya bentuk bangunannya saja (*building footprint*).

Peta yang telah diberikan keterangan tipe objek akan terlihat informatif dan relatif lebih mudah untuk dilihat oleh pengguna. Lebih lanjut, ada beberapa fungsi lain yang sebetulnya sangat bermanfaat jika kita memasukkan informasi *tag* objek dengan sesuai di OpenStreetMap. Fungsi pertama adalah memudahkan pencarian terhadap objek yang ingin kita temukan di OpenStreetMap. Khususnya jika kita ingin men*download* data dari Overpass turbo. Dengan memberikan *tag* yang sesuai terhadap objek maka kita dapat menemukan objek-objek yang memiliki tag yang sama. Sebagai contoh, kita ingin mendownload seluruh objek sekolah pada suatu area di OpenStreetMap maka jika semua sekolah di area tersebut sudah memiliki *tag* **amenity** = **school** kita akan mendapatkan jumlah berapa sekolah yang ada serta dapat langsung mendownload seluruh objek tersebut. Tetapi, jika ada beberapa sekolah belum diberikan tag **amenity**=**school** dan hanya diberikan keterangan seperti **building**=**yes** maka objek tersebut tidak akan terhitung dan tidak dapat kita download. Berikut tampilan contoh untuk fungsi di atas:



Fungsi kedua yaitu untuk pemrosesan data lebih lanjut dengan menggunakan perangkat lunak Sistem Informasi Geografi (SIG) seperti QGIS. Ketika kita sudah selesai mendownload data yang ada di OpenStreetMap, kita semestinya melakukan proses data yang kita miliki di QGIS baik itu melakukan filter data untuk klasifikasi, simbologi data hingga melakukan *layout* data. Untuk itu supaya proses pengolahan data tersebut dapat dilakukan dengan baik dan benar, data-data OpenStreetMap yang kita miliki sebaiknya sudah memiliki *tag* informasi yang sesuai dengan jenis-jenis objeknya. Hal ini agar ketika proses filter dan klasifikasi data dilakukan tidak ada objek yang terlewatkan. Sebagai contoh, kita ingin memilih data-data rumah sakit yang ada di Jakarta maka kita dapat melihat tabel atribut dari data OpenStreetMap yang kita miliki tersebut kemudian kita pilih berdasarkan jenis **amenity** = **hospital**. Berikut adalah contoh tabel atribut data OpenStreetMap rumah sakit.

amenity	name
hospital	RS HERMINA
hospital	RS Harapan Jaya
hospital	RS HAJI JAKARTA
hospital	RS Grha Kedoya
hospital	RS Gading Pluit
hospital	RS Fatmawati
hospital	RS Familly
NULL	RS Duren Sawit
NULL	RS Budi Kemuliaan
NULL	RS Budi Kemuliaan
NULL	RS Binawaluya

Dari contoh di atas, kita dapat melihat jika kita ingin memilih seluruh rumah sakit yang ada pada tabel di atas maka tidak semua dari daftar rumah sakit tersebut yang akan terpilih. Rumah sakit yang tidak memiliki *tag* amenity=hospital seperti RS Duren Sawit, RS Budi Kemuliaan dan RS Binawaluya tidak akan terpilih. Oleh karena itu, pemberian *tag* terhadap objek-objek tersebut ketika dipetakan ke OpenStreetMap menjadi hal yang penting.

Untuk keterangan lebih lanjut mengenai *tag* yang seharusnya diberikan pada objek-objek yang ingin anda petakan, anda dapat melihat informasinya di halaman wiki OSM kami di: http://wiki.openstreetmap.org/wiki/Id:Referensi Atribut Objek OSM Indonesia

B. Objek Bangunan dengan lebih dari 1 Jenis Amenitas

Beberapa kali juga kita menemukan satu bangunan yang di dalamnya terdapat beberapa toko ataupun kantor. Hal seperti ini seringkali membingungkan pengguna untuk menggambarkannya. Jika anda menemukan kasus seperti ini silahkan:

• Dijitasi bangunan utamanya terlebih dahulu dalam bentuk **poligon** (building footprint)

kemudian silahkan masukkan preset building dan isi informasi terkait building tersebut.

- Setelah selesai, dijitasi masing-masing toko/amenitas yang ada di bangunan tersebut dengan titik (point) dan berikan tag sesuai dengan informasi yang sesuai. Misalnya amenity = restaurant, shop = hairdresser, amenity = fast_food
- Setelah selesai, objek-objek yang anda petakan kurang lebih akan terlihat seperti berikut ini :



Catatan:

Untuk beberapa objek tertentu seperti **Sekolah, Mall, Kantor Pemerintah** (**Dinas / Kementrian**) walaupun sudah memasukkan *tag* sesuai dengan objek tersebut, simbol objek tidak akan muncul. Hal ini dikarenakan dari OpenStreetMap belum menyediakan simbol untuk beberapa objek tersebut. Hal ini bukan merupakan suatu kesalahan dan silahkan tetap memasukkan *tag amenity=** atau *shop=** atau *office=** yang sesuai dan objek anda akan terdijitasi dengan benar di OpenStreetMap.

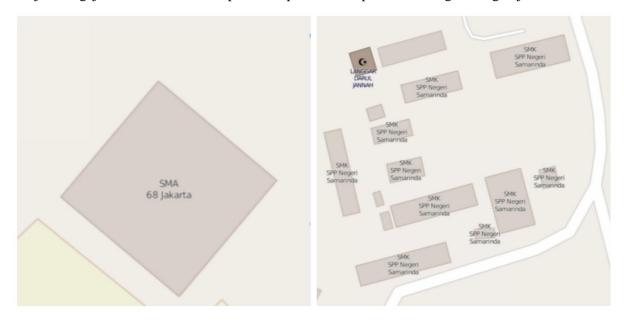
C. Objek yang Terdiri dari Beberapa Bangunan

Seringkali kita menemukan objek yang di dalamnya terdapat beberapa bangunan tetapi masih merupakan bagian dari satu jenis objek. Contoh objek dengan karakteristik seperti itu yang paling

sering kita temukan adalah sekolah ataupun rumah sakit. Jika kita menemukan objek seperti ini, sebaiknya kita menggambar bangunannya terpisah dan sesuai fungsinya masing-masing. Berikut adalah contoh bangunan dengan bentuk kompleks.



Dapat kita lihat dari contoh di atas dalam satu sekolah biasanya terdapat beberapa jenis bangunan seperti misalnya ada masjid, tempat parkir ataupun lapangan di dalamnya. Dalam contoh yang lain terlihat juga bahwa dalam 1 sekolah memiliki lebih dari 2 gedung yang digunakan untuk kegiatan belajar mengajar. Berikut adalah tampilan di OpenStreetMap untuk masing-masing objek tersebut.

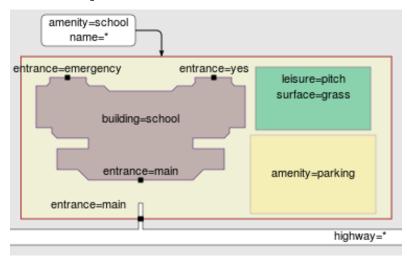


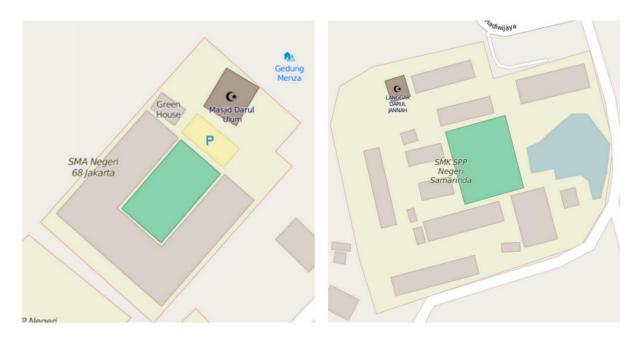
Untuk gambar di sebelah kiri (SMA 68 Jakarta) memang sekilas tidak terdapat kesalahan akan tetapi bentuk gedung dari sekolah tersebut bukanlah segi empat melainkan berbentuk seperti huruf U dengan lapangan di tengah-tengah gedung tersebut. Adapun untuk gambar sebelah kanan (SMK SPP Samarinda) terlihat digambarkan 10 gedung yang digunakan untuk kegiatan belajar mengajar kemudian diberikan nama yang sama untuk masing-masing gedung dan juga

ditandai sebagai sekolah. Hal ini tentu saja akan memberikan informasi yang salah kepada yang melihat terutama jika ingin mengetahui jumlah sekolah yang ada. Sekolah SMK SPP Samarinda yang harusnya dihitung hanya 1 sekolah oleh OpenStreetMap akan dianggap sebagai 10 sekolah dikarenakan diberikan tag yang sama yaitu amenity = school.

Untuk menghindari kesalahan data seperti di atas silahkan lakukan langkah-langkah berikut ini:

- Gambar semua bangunan yang ada sesuai dengan yang terlihat di citra satelit menggunakan **JOSM** anda
- Untuk bangunan yang digunakan untuk kegiatan belajar mengajar berikan tag building = school beserta dengan informasi gedung lain seperti tinggi gedung, jenis atap, dll (jika diperlukan).
- Untuk bangunan yang lain seperti tempat parkir ataupun masjid silahkan berikan *tag* sesuai dengan jenisnya masing-masing **tanpa diisi nama bangunan**. Untuk *tag* selengkapnya silahkan lihat di sini (https://wiki.openstreetmap.org/wiki/Id:Map_Features)
- Kemudian untuk pemberian nama dan jenis/amenitas objek, silahkan gambarkan sisi-sisi paling luar dari sekolah sehingga membentuk suatu area baru dan berikan tag amenity = school dan name = * sesuai dengan nama sekolah tersebut. Ingat! Pemberian nama sekolah hanya dilakukan sekali saja.





Ketika anda melihat hasilnya maka tampilan objek anda sekarang akan sangat berbeda dari sebelumnya. Dalam OpenStreetMap penggambaran **objek** yang **terdiri dari beberapa bangunan ataupun amenitas** seharusnya digambar sesuai dengan fungsi masing-masing bangunannya. Penggambaran area luar yang mengelilingi bangunan tersebut dengan nama objeknya juga perlu digambarkan agar terlihat sejauh mana area komplek bangunan tersebut berada sehingga jika ada satu objek seperti sekolah atau rumah sakit yang memiliki beberapa gedung akan tetap dihitung sebagai satu kesatuan/bagian dari objek utama oleh OpenStreetMap dan tidak dianggap sebagai objek yang berbeda. Sehingga saat pendataan jumlah objek untuk keperluan lebih lanjut seperti kebencanaan, informasi yang diberikan tepat dan tidak menyesatkan.

2.1.2 Pemberian Nama Objek (Toponimi)

Pemberian nama pada objek merupakan sesuatu yang sangat penting untuk diperhatikan. Dengan penamaan yang tepat dan benar maka peta kita akan memberikan informasi yang mudah dipahami dan tidak menyesatkan bagi orang yang melihat peta kita. Dengan begitu banyaknya pengguna OpenStreetMap saat ini, pemberian nama untuk satu jenis objek bisa bervariasi dan berbeda-beda baik itu cara penulisannya maupun namanya. Silahkan perhatian beberapa contoh di bawah ini :





Gambar-gambar di atas adalah beberapa contoh dari kesalahan penamaan di OpenStreetMap. Menyesuaikan dengan ketentuan yang berlaku di Indonesia dimana untuk masalah penamaan objek pada peta telah diatur oleh Badan Informasi Geospasial (BIG) selaku institusi pemerintah yang berwenang terhadap data spasial di Indonesia dan juga menyesuaikan dengan Ejaan Yang Disempurnakan (EYD) yang sesuai dengan Kamus Besar Bahasa Indonesia (KBBI), pemberian nama untuk objek-objek di OpenStreetMap khususnya di Indonesia sebisa mungkin mengacu kepada ketentuan tersebut. Beberapa hal yang harus diperhatikan ketika memberikan nama objek di OpenStreetMap adalah sebagai berikut:

A. Objek Jalan

Penulisan untuk nama jalan dan gang **WAJIB** diawali dengan huruf besar dan ditulis lengkap. Berikut beberapa contohnya:

Nama Jalan		Nama Jalan	
Jalan Matraman Raya		JI Matraman Raya / jalan matraman raya	×
Jalan TB. Simatupang		JLN. TB. Simatupang / jln tb simatupang	×
Jalan Gunung Merapi		JL. Gn. Merapi / jl gunung merapi	X
Jalan Tol Pelabuhan	/	Tol Pelabuhan / jalan tol pelabuhan	×
Gang Mesjid	/	Gg. Masjid / gang masjid	×

B. Objek Bangunan

Untuk penulisan nama objek bangunan sama seperti dengan objek jalan yaitu **WAJIB** diawali dengan huruf besar dan ditulis lengkap. Berikut adalah beberapa contoh untuk penamaan objek bangunan :

Nama Objek	Nama Objek
 SD Negeri D1 Kebon Manggis SD Kanisius SMP Negeri D2 Utan Kayu SMP Karya Guna SMA Negeri D3 Kampung Rambutan Universtias Indonesia Sekolah Tinggi Ilmu Perpajakan TK Mandiri 	 SDN 01 Kb. Manggis sd kanisius SMPN 02 utan kayu SMP KG SMA negri 03 Kp. Rambutan Univ. Indonesia STIP / SEKOLAH TINGGI ILMU PERPAJAKAN Tk Mandiri / Taman Kanak-Kanak Mandiri
 Rumah Sakit Hermina RSUD Kota Semarang Puskesmas Matraman Klinik Fakhira Posyandu Sehat 	 RS. Hermina Rumah sakit umum daerah kota semarang Pusk. Matraman Klinik fakhira POSYANDU SEHAT
 Masjid An Nur Musala Al Munnawaroh Gereja Kanisius Pura Besakih Vihara Satya Darma 	 Mesjid an nur musollah Al Munnawroh gereja kanisius Pura besakih / Pr. Besakih Wihara Satya Darma / vihara satya darma
 Terminal Pulogadung Toko Sepatu Toni Museum Nasional Stasiun Manggarai Taman Lawang Rumah Makan Bumbu Desa 	 terminal pulogadung Tk. Sepatu toni Musium nasional St. Manggarai / Setasiun Kereta Api Manggarai Tm. Lawang RM. Bumbu Desa

Contoh-contoh di atas merupakan sebagian dari penamaan yang selama ini ditemukan di OpenStreetMap khususnya di wilayah Indonesia. Jika anda menemukan objek yang tidak ditemukan pada contoh-contoh di atas silahkan beberapa panduannya di:

- http://kbbi.web.id/
- http://www.big.go.id/dokumen-standar/
- http://www.big.go.id/assets/download/sni/SNI/SNI%2019-6502.2-2000_image.pdf

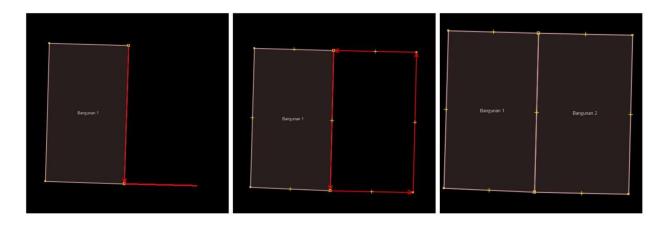
2.1.3 Objek dan Penggunaan Lahan yang Saling Berdekatan

Seringkali kita menemukan objek yang jaraknya saling berdekatan satu dengan yang lain. Hal ini seringkali membingungkan dan membuat data kita menjadi salah. Biasanya objek-objek yang sangat berdekatan berpotensi untuk membuat data kita menjadi salah. Untuk jenis kesalahan data OpenStreetMap yang lebih lengkap akan kita bahas di subbab selanjutnya.

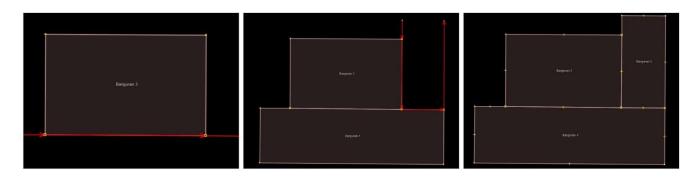
Berikut adalah contoh bangunan yang sangat berdekatan satu dengan yang lain.



Bagaimana cara yang tepat untuk mendijitasi objek seperti bangunan yang sangat berdekatan tersebut? Jika kita ingin mendijitasi bangunan yang sangat berdekatan maka *node* (titik) yang ada pada bangunan harus saling menempel dengan *node* (titik) pada bangunan di sebelahnya. Berikut adalah contoh gambaran di JOSM.



Untuk objek yang bentuknya tidak sama ataupun berbeda posisi, kita tetap menghubungkan *node* (titik) yang ada pada objek tersebut dengan beberapa *node* (titik) bangunan sebelahnya kemudian melanjutkan dijitasi sesuai dengan bentuk bangunan tersebut. Berikut contohnya:



Berikut adalah salah satu objek yang sangat berdekatan yang telah didijitasi di OpenStreetMap:



Penggambaran objek yang sangat berdekatan ini berlaku pula untuk mendijitasi **penggunaan lahan** (*land use*) dimana seringkali kita menemukan objek penggunaan lahan berada langsung di sisi jalan ataupun berada di tengah-tengah perumahan. Berikut adalah contoh penggunaan lahan di OpenStreetMap:



Untuk dijitasi penggunaan lahan sebaiknya saling **menempel** dengan bagian *node* (**titik**) jalan. Sedangkan untuk penggunaan seperti sawah ataupun rerumputan yang terdapat bangunan di tengahnya, silahkan **tumpah tindih-kan** (*overlap*) bangunan di dalam penggunaan lahan tersebut. Hal ini diperbolehkan di OpenStreetMap dan tidak akan dianggap sebagai sebuah kesalahan. Berikut adalah tampilan editing di JOSM dan di OpenStreetMap untuk gambar penggunaan lahan di atas:



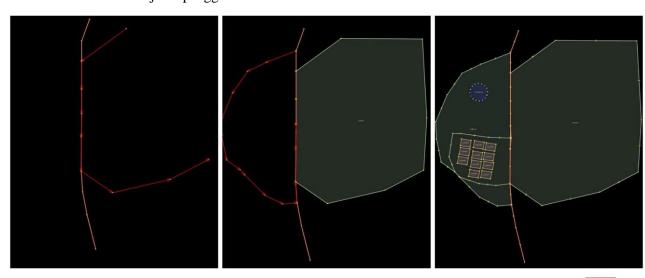
Terlihat dari contoh di atas terdapat 2 tampilan OpenStreetMap di wilayah yang berbeda. Peta diatas

merupakan contoh penggambaran penggunan lahan di OpenStreetMap yang lebih banyak terdapat penggunaan lahan alami seperti kebun, sawah, ladang maupun sungai.



Untuk peta diatas juga merupakan contoh penggambaran penggunaan lahan di OpenStreetMap akan tetapi lebih banyak penggunaan lahan non alami seperti permukiman, kawasan industri maupun tempat perbelanjaan. Jika pada gambar pertama rumah penduduk digambarkan masing-masing rumah maka peta di gambar kedua rumah-rumah penduduk digambarkan berupa area permukiman dimana untuk bentuk bangunan digambarkan hanya khusus untuk fasilitas umum dan begitu juga dengan kawasan industri. Hal ini menunjukkan bahwa kita dapat memetakan kawasan permukiman terlebih dahulu kemudian baru bisa menambahkan persil-persil rumah yang ada di kawasan tersebut yang tentunya membuat data kita akan terlihat lebih rapi dan teratur.

Berikut adalah contoh dijitasi penggunaan lahan di JOSM:



Penjaminan Kualitas Data untuk Openstreetmap

Dapat kita lihat pada contoh di atas untuk **penggunaan lahan dapat ditumpang tindih-kan** (*overlay*) dengan bangunan ataupun penggunaan lahan yang lain dan data kita tidak akan dianggap sebagai sebuah peringatan/warning. Hal ini tentu saja disebabkan karena untuk **dijitasi penggunaan lahan memang tidak boleh terdapat jarak** ataupun **ruang kosong satu dengan yang lain** sesuai dengan prinsip topologi data spasial. Oleh karena itu, sebaiknya jika ingin mendijitasi penggunaan lahan di suatu area silahkan dijitasi dulu jalan-jalan yang ada di daerah tersebut sehingga akan lebih memudahkan kita ketika ingin memulai mendijitasi penggunaan lahan dan menempelkannya dengan jalan yang sudah ada. Tentu saja keuntungannya adalah kita akan mendapatkan objek-objek kita akan terlihat lebih indah di dalam OpenStreetMap seperti contoh peta di atas.

2.1 4 Menggambar Jalan

Jalan raya merupakan salah satu objek penting yang harus dimasukkan ke dalam OpenStreetMap. Kualitas penggambaran jalan raya sangat menentukan informasi seperti rute dan arah ke suatu tempat ataupun digunakan melakukan navigasi menggunakan aplikasi seperti *OSMAnd*. Akan tetapi, seringkali kita menemukan beberapa kesalahan baik itu seperti penamaan yang kurang jelas yang sudah dibahas di subbab sebelumnya ataupun kesalahan fisik seperti arah jalan yang salah ataupun jenis jalan yang digambarkan. Berikut adalah contoh jalan raya yang kompleks :



Dalam contoh di atas ada beberapa hal yang dapat kita perhatikan dan jadikan contoh untuk penggambaran jalan raya di OpenStreetMap, yaitu :

- Arah Jalan
- Layer (Tingkatan) Jalan
- Jumlah Ruas
- Jalan yang dipisahkan oleh pemisah jalan (*separator*)
- Klasifikasi Jalur Jalan

Arah Jalan – untuk penggambaran arah jalan raya di OpenStreetMap harus digambarkan sesuai dengan arah sebenarnya di JOSM. Jika arah jalan tersebut dari utara ke selatan maka ketika mendijitasi di JOSM kita juga harus mendijitasi garisnya dari utara ke selatan. Berikut adalah bentuk jalan raya di JOSM dan hasilnya di OpenStreetMap.

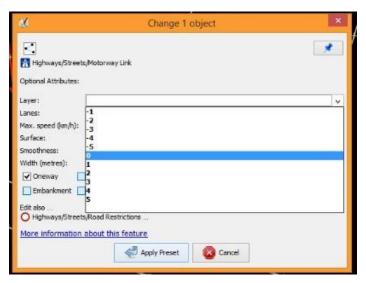


Setiap jalan raya harus diberikan **satu arah** saja, silahkan anda pilih keterangan di *tag* jalan raya yang anda pilih dan centang kotak yang bertuliskan *oneway*. Hal ini untuk menandai arah jalanan tersebut sesuai dengan yang kita gambarkan dan tidak berlaku sebaliknya.

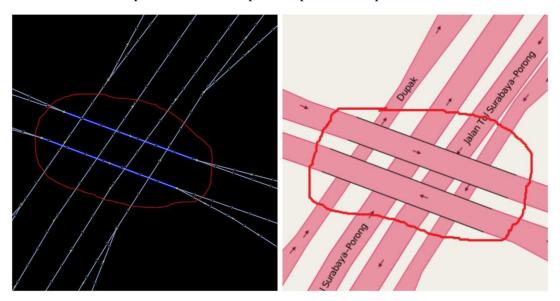


Layer (**tingkatan**) **Jalan** – Beberapa kali terutama di kota-kota besar kita menemukan jalanan yang bertingkat. Jalanan seperti ini sering kita sebut dengan Jalan Layang (*flyover*). Dalam OpenStreetMap, penggambaran jalan yang bertingkat ini kita berikan *tag* khusus agar tidak saling tumpang tindih (*overlap*) dengan jalanan lain yang sejajar. *Tag* tersebut adalah **Layer**. Informasi *layer* ini sudah

disediakan oleh preset awal JOSM dan kita tinggal mengisinya saja. Berikut tampilan isian *layer* yang ada di JOSM:



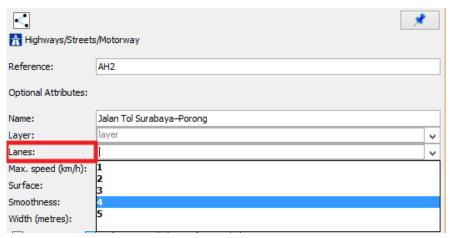
Sesuai dengan ketentuan dari OpenStreetMap, *layer* awal jika kita tidak mengisi kolom tersebut adalah 0. Maksudnya adalah jalan raya tersebut posisinya berada sejajar dan dibangun tepat di atas tanah. Jika terdapat jalan raya baru yang berada di atas tanah seperti jalan layang makan diberikan *layer* 1 dan jika berada di atasnya lagi diberikan *layer* 2 dan seterusnya. Jika menemukan jalan raya di bawah tanah seperti *subway* atau terowongan maka nilai *layer* harus diisi -1 dan seterusnya. Berikut adalah contoh tampilan di JOSM maupun di OpenStreetMap:



Jumlah Ruas Jalan – Salah satu unsur jalan yang sering dilupakan akan tetapi cukup penting untuk diketahui adalah ruas jalan. Satu jalanan bisa terdiri dari 2 atau lebih ruas jalan. Biasanya ruas jalan ini ditandai oleh garis putih putus-putus. Berikut adalah contoh jalanan dengan ruas jalannya:



Dapat kita lihat pada gambar di atas terdapat **4 ruas jalan**. Jika kita melihat jalan seperti itu maka kita bisa menambahkan informasi ruas jalan dengan menambahkan *Tag* yang disebut **Lanes** dan isi sesuai dengan jumlah ruas di jalan tersebut. Berikut tampilan menambahkan **lanes** di JOSM.



Jalan yang dipisahkan oleh pemisah jalan (*separator*) – Seringkali kita melihat ada jalanan yang dipisahkan oleh separator baik untuk memisahkan karena berbeda arah ataupun satu arah tetapi membagi untuk jalur motor dan mobil. Pemisah jalanan tersebut biasanya kita sebut dengan *separator*. Kondisi ini di OpenStreetMap sebaiknya digambarkan juga karena hal ini penting untuk menandakan apakah jalanan tersebut satu bagian atau tidak dan bisakah kita langsung melakukan putaran balik atau tidak di jalan raya tersebut. Berikut adalah contoh jalanan yang dipisahkan oleh *separator*.



Dapat kita lihat pada gambar di atas terlihat bahwa **Jalan Sudirman dan jalur koridor Transjakarta dipisahkan** oleh *separator* yang berwarna kuning walaupun kedua jalan tersebut memiliki arah yang sama. Sedangkan dapat kita lihat pula jalan di sebelahnya dipisahkan oleh *separator* yang lebih besar dengan taman dan lampu jalan juga digambarkan berbeda. Untuk kondisi **jalan yang terpisah** oleh *separator* seperti contoh di atas **harus diberikan** keterangan *oneway* pada *tag*-nya sehingga di *website* OpenStreetMap arah jalan tersebut akan terlihat yang ditunjukkan oleh arah panah.

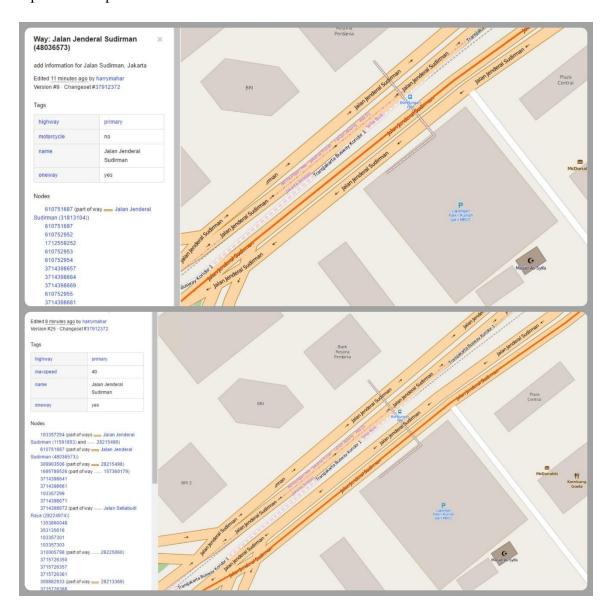
Untuk **jalan** yang memiliki **2 arah yang berlawanan** akan tetapi **tidak dipisahkan** oleh *separator* maka cukup **digambarkan satu kali** di JOSM dan **tidak diberikan** keterangan *oneway*. Tampilan di OpenStreetMap juga tidak akan diberikan arah dengan panah seperti jalan raya yang dipisahkan oleh *separator*. Berikut adalah contoh tampilan jalan raya tersebut di OpenStreetMap.



Klasifikasi Jalur Jalan – Seringkali kita melihat ada jalan dengan arah yang sama tetapi dipisahkan

oleh separator. Jalan dengan arah yang sama tetapi dipisahkan oleh separator biasanya memiliki aturan terhadap kendaraan yang diperbolehkan melewati masing-masing jalur. Aturan tersebut bisa berdasarkan jenis kendaraan ataupun batas kecepatan kendaraan dan bahkan keduanya. Jalan seperti ini seringkali kita temukan di perkotaan dimana jumlah serta volume kendaraan yang relatif padat setiap harinya. Kita dapat memasukkan klasifikasi jalur ini di OpenStreetMap. Untuk batas kecepatan maksimal kendaraan anda dapat menambahkan *tag* maxspeed=* pada masing-masing jalur jalan.

Untuk jalur cepat biasanya kendaraan roda dua seperti motor tidak diperbolehkan untuk melintas. Untuk menambahkan informasi pembatasan akses motor silahkan anda tambahkan keterangan *tag* **motorcycle** = **no** di jalur cepat pada jalan yang anda petakan. Berikut adalah contoh tampilan petanya di OpenStreetMap:

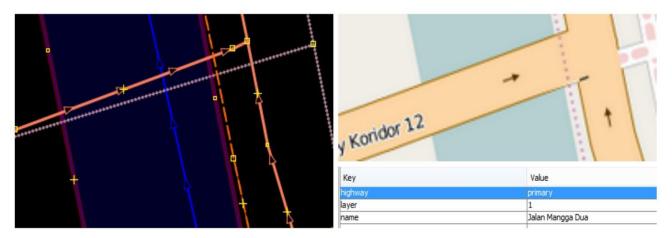


2.1 5 Menggambar Jembatan

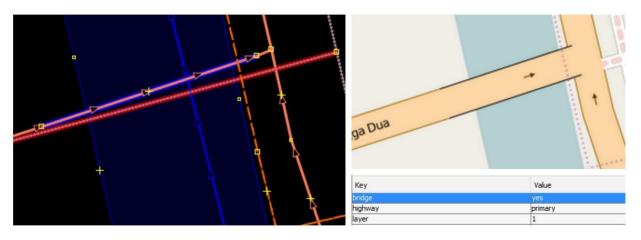
Seringkali kita menemukan jembatan di banyak tempat akan tetapi sering juga kita bingung untuk

menggambarkannya. Jika jembatannya tersambung dengan jalan raya maka biasanya kita hanya menggambarkan jalannya saja tanpa membuat jembatannya. Hal tersebut merupakan suatu kekeliruan. Oleh karena itu untuk membuat jembatan silahkan anda pisahkan ruas jalan yang seharusnya menjadi jembatan (*split*) kemudian tambahkan atribut berikut **bridge** = **yes**. Berikut adalah contoh tampilan jika sebuah jalan belum diberikan jembatan dan sesudah diberikan jembatan di JOSM dan *website* OpenStreetMap.

Sebelum diberikan tag Jembatan



Sesudah diberikan tag Jembatan



Dapat dilihat pada contoh diatas jika suatu jalan sudah diberikan *tag* **bridge** = **yes** maka dia akan memiliki garis hitam di sekitar piggir jalannya. Selain itu, kita juga harus memperhatikan bahwa selain tag **bridge** = **yes** kita **harus** memberikan **layer** = **1** sebagai tanda bahwa jembatan tersebut berada di atas sungai dan tetap harus menyimpan *tag highway* =* sesuai dengan jenis jalan yang ada pada jembatan tersebut.

2.1.6 Penggambaran Batas Administrasi

Meskipun OpenStreetMap dikhususkan untuk menggambar objek-objek fisik yang terlihat seperti sekolah, rumah sakit, terminal, stasiun, dan lain-lain tetapi kita juga dimungkinkan untuk dapat menambahkan batas-batas administrasi di dalamnya.

Beberapa kegiatan sudah menggunakan OpenStreetMap untuk memetakan batas administrasi seperti *Jakarta Mapping* pada tahun 2012 oleh BPBD DKI Jakarta. Akan tetapi, penggambaran batas administrasi ini sedikit kompleks dikarenakan harus dimasukkan relasi di dalamnya. Setiap batas satu dengan yang lain pasti saling terhubung dan oleh karena itulah kita harus berhati-hati untuk menentukan relasi dari batas tersebut. Berikut adalah contoh tahapan dalam menggambar batas administrasi di OpenStreetMap:

Sebelum kita memulai menggambar batas administrasi di OpenStreetMap khususnya di Indonesia, ada baiknya kita mengetahui dulu tingkatan administrasi di Indonesia. Tingkatan batas administrasi bisa jadi berbeda-beda antara negara yang satu dengan yang lainnya. Anda dapat melihat tingkatan batas administrasi di setiap negara di http://wiki.openstreetmap.org/wiki/Tag:boundary%3Dadministrative. Untuk tingkatan administrasi di Indonesia adalah sebagai berikut:

value	Tingkat Administrasi	Tampilan di OSM	Place
1	-	-	
2	Batas Negara		
3	E.	-	
4	Batas Provinsi		province
5	Batas Kotamadya / Kabupaten		district
6	Batas Kecamatan	M 1 MC 1 MC 2 MC 2 MC 3 MC 3 MC 4 MC 4 MC 4 MC 7 MC	municipality
7	Batas Kelurahan / Desa		village

8	Batas Dusun	
9	Batas Rukun Warga (RW)	
10	Batas Rukun Tetangga (RT)	

Seperti yang sudah disebutkan bahwa batas administrasi adalah hal yang sensitif untuk dipetakan. Oleh karena itu diperlukan beberapa persyaratan yang sebaiknya dipenuhi agar data batas administrasi kita tidak menimbulkan masalah di kemudian hari. Beberapa syarat yang perlu diperhatikan sebelum kita menggambarkan batas administrasi di OpenStreetMap yaitu:

- Data harus memiliki sumber yang jelas dari mana asalnya misalnya data bersumber dari Badan Informasi Geospasial (BIG)
- Data harus memiliki izin penggunaan (lisensi), apakah data tersebut boleh disebarluaskan ke publik atau tidak karena data ini nantinya akan dilihat dan digunakan oleh publik atau masyarakat luas
- Data harus memiliki pembagian batas-batas administrasi yang jelas dan sesuai dengan yang sudah disahkan
- Sudah memahami konsep relasi pada data OpenStreetMap terutama relasi untuk batas administrasi

Setelah mengetahui tingkatan administrasi daerah yang ingin kita petakan dan sudah memenuhi persyaratan yang sudah disebutkan di atas sekarang kita dapat memulai untuk menggambar batas administrasi di OpenStreetMap. Berikut adalah langkah-langkahnya:

A. Relation

Relasi merupakan salah satu elemen utama dalam data OSM yang terdiri dari satu atau lebih tag dan juga memuat daftar urutan objek yang tergabung dalam relasi sebagai anggota dari relasi tersebut.

Dalam membuat batas administrasi, kita harus menggunakan fitur relasi. Relasi digunakan untuk mengelompokkan batas administrasi dan memberikan tanda batas tersebut sebagai batas luar atau batas dalam.

Relation dalam pembuatan batas administrasi terdiri dari dua jenis yaitu **Relation Tags** dan **Relation Members.**

Relation Tags

Key	Value	Keterangan
admin_level	**disesuaikan dengan batas administrasi	Tag yang berguna untuk mengidentifikasi tingkat batas administrasi pada suatu daerah, misalnya admin_level = 4 di Indonesia merupakan tingkat batas administrasi untuk batas provinsi.
boundary	administrative	Tag ini merupakan tag batas administrasi yang sebenarnya.
type	boundary	Tag ini adalah tag utama untuk mengenali bahwa objek tersebut termasuk ke dalam batas administrasi.
name	**nama administrasi	Tag yang mendeskripsikan nama dari batas administrasi tersebut, misalnya: Provinsi Jawa Barat, Kota Depok, Kecamatan Sawangan, RW 03 ataupun RW 05.
place	**contoh: suburb, island	Ini merupakan tag <i>optional</i> , tidak wajib dipakai. Tag yang berguna untuk mendeskripsikan jenis area yang ada di dalam batas administrasi tersebut apakah termasuk garis pantai dan penggunaan lahan lainnya. Info lengkap lihat di: http://wiki.openstreetmap.org/wiki/Key:place
land_area	administrative	Tag untuk menyatakan sebagai sebuah batas wilayah daratan
PROV_name	**nama provinsi	Tag yang mendeskripsikan nama Provinsi. Tag ini harus dimasukkan pada batas administrasi yang berada di tingkat bawah dari Provinsi (Kabupaten/Kotamadya, Kecamatan, Kelurahan/Desa, RW dan RT)
KAB_name	**nama kabupaten/kotamadya	Tag yang mendeskripsikan nama Kabupaten/kotamadya. Tag ini harus dimasukkan pada batas administrasi yang berada di tingkat bawah dari Kabupaten/kotamadya (Kecamatan, Kelurahan/Desa, RW dan RT)
KEC_name	**nama kecamatan	Tag yang mendeskripsikan nama Kecamatan. Tag ini harus dimasukkan pada batas administrasi yang berada di tingkat bawah dari Kecamatan (Kelurahan/Desa, RW dan RT)
KEL_name	**nama kelurahan/desa	Tag yang mendeskripsikan nama Kelurahan/Desa. Tag ini harus dimasukkan pada batas administrasi yang berada di

		tingkat bawah dari Kelurahan/Desa (RW dan RT)
RW_name	**nama RW	Tag yang mendeskripsikan nama RW. Tag ini harus dimasukkan pada batas administrasi yang berada di tingkat bawah dari RW (RT)

Relation Members

• Admin_centre

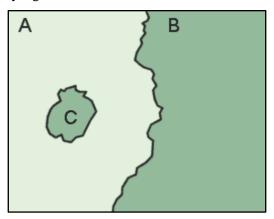
Titik yang mewakili pusat administrasi pada suatu daerah biasanya berupa ibu kota provinsi, kelurahan, dan sebagainya, tergantung dari tingkat batas administrasinya.

• Outer

Rangkaian dari garis-garis yang terhubung menjadi suatu batas tertutup seperti poligon / batas luar dari suatu wilayah administrasi.

Inner

Rangkaian dari garis-garis yang terhubung menjadi suatu batas tertutup seperti poligon dan objek tersebut berada di dalam area yang termasuk outer / batas dalam dari suatu wilayah administrasi.



Wilayah C berada di dalam wilayah A dan wilayah C diidentifikasi sebagai inner dari relation members

B. Membuat Batas Administrasi

Dalam membuat batas administrasi sangat disarankan untuk menggunakan editor Java OpenStreetMap (JOSM) karena tool-tool yang disediakan lebih lengkap, sehingga untuk melakukan editing dan modifikasi jauh lebih mudah dibandingkan editor lainnya seperti iD editor.

Catatan:

Data yang dipakai pada tutorial ini hanyalah data contoh atau data fiktif untuk mempermudah pemahaman kita dalam mempraktekan pembuatan batas administrasi.

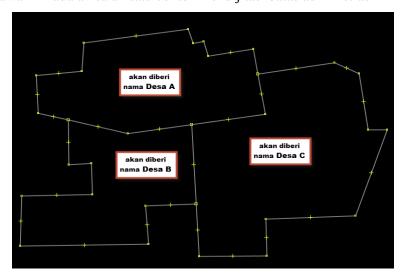
Berikut langkah-langkah yang dilakukan untuk membuat batas administrasi di OpenStreetMap yaitu:

• Mendigitasi Garis Batas Administrasi

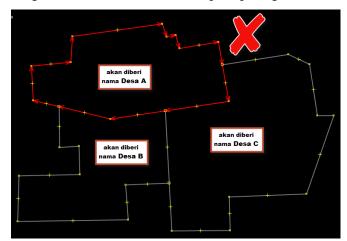
- 1. Hal pertama yang dilakukan adalah membuka editor JOSM di komputer Anda.
- Kemudian pilih wilayah yang akan dibuat batas administrasinya dengan mendownload data OSM di wilayah tersebut.
- 3. Mulailah dengan mendijitasi batas administrasi yang sudah sesuai dengan syarat dan ketentuan data yang akan dimasukkan pada penjelasan sebelumnya. Untuk mendijitasi gunakanlah tool **Draw Node (A)**.



4. Gambar di bawah ini adalah salah satu contoh mendijitasi batas administrasi

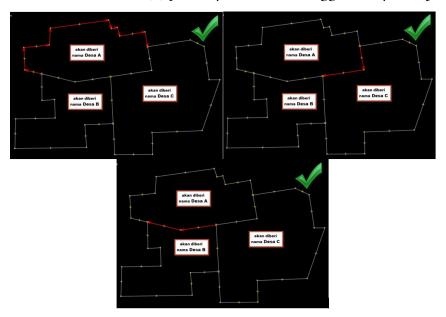


- 5. Dalam menggambar batas administrasi, harap memperhatikan hal-hal berikut:
 - Gambarlah sebagai garis yang saling tersambung satu sama lain antar batas administrasi
 - Tidak ada garis batas administrasi yang saling bertampalan atau tergambar dua kali
 - Pastikan setiap pertemuan garis merupakan garis yang terpisah (dalam artian berbeda segmen). Jika garis tersebut masih terlihat seperti pada gambar di bawah ini:



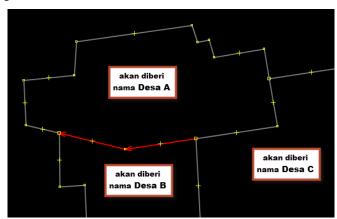
Penjaminan Kualitas Data untuk Openstreetmap

Anda harus membagi garis tersebut menggunakan Tools → Memisahkan Garis/Way (Split Way) atau menggunakan shortcut tombol (P) pada keyboard Anda hingga hasilnya menjadi seperti ini:

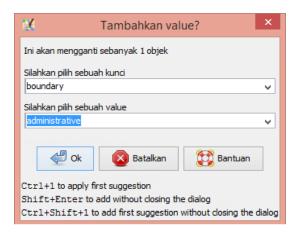


• Memberikan Tag pada Garis Batas Administrasi

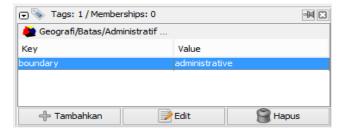
1. Langkah selanjutnya kita akan memberikan tag pada tiap garis batas yang kita buat. Pilih salah satu segmen garis dari batas administrasi.



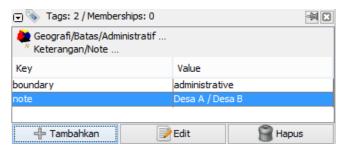
Kemudian berikan tag pada garis yang terpilih tersebut dengan cara pergi ke panel
 Tags/Membership → Tambahkan. Pada kolom "kunci" pilih/ketik "boundary" dan pada
 kolom "value" pilih/ketik "administrative". Lalu klik OK.



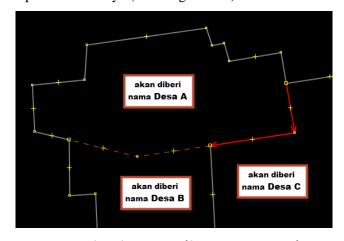
3. Setelah mengisi dialog **Tambahkan Value** tampilan hasilnya akan seperti ini



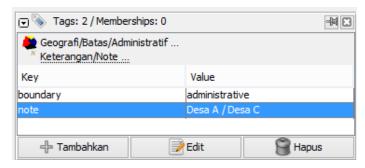
4. Selanjutnya menambahkan tag Note dengan **Key=note** dan **Value=Desa A / Desa B**, langkah yang dilakukan sama seperti langkah sebelumnya. Tag ini bertujuan untuk memberikan keterangan bahwa segmen garis tersebut merupakan batas administrasi untuk Desa A dan Desa B.



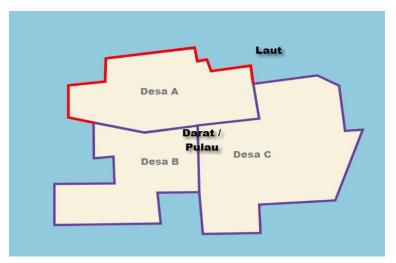
5. Mari kita lanjutkan memberi tag pada segmen-segmen garis pembentuk batas administrasi Desa A. Sekarang pilih segmen garis batas antara Desa A dan Desa C. Berikan tag dengan langkah yang sama seperti sebelumnya (dari langkah 1-4)



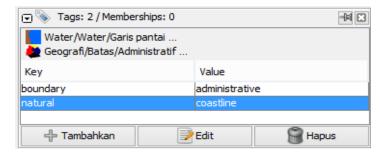
Penjaminan Kualitas Data untuk Openstreetmap



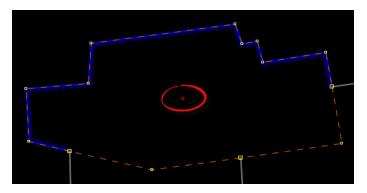
6. Tutorial kali ini akan memberikan contoh membuat batas administrasi desa dan kelurahan yang mana kelurahan tersebut wilayahnya mencakup satu daratan/pulau. Pada gambar di bawah ini, garis yang berwarna merah merupakan garis batas antara Desa A dengan wilayah laut.



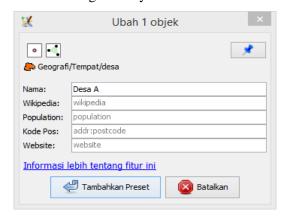
Pada kasus ini, tag yang harus diberikan sedikit berbeda, yaitu harus ditambahkan Key=natural dan Value=coastline, kemudian untuk Key=note bisa kita hilangkan.



7. Setelah segmen-segmen dari garis batas Desa A diberi tag, gambarlah sebuah titik sebagai titik pusat dari Desa A.



8. Kemudian beri tag pada titik tersebut dengan pergi ke Presets → Geografi → Tempat → Desa. Isi lah kolom Nama dengan nama desa dan kolom informasi lainnya seperti jumlah populasi dan kode pos bila Anda mengetahuinya.



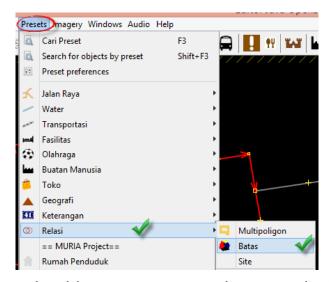
9. Setelah titik tersebut diberi tag maka hasilnya akan seperti ini:



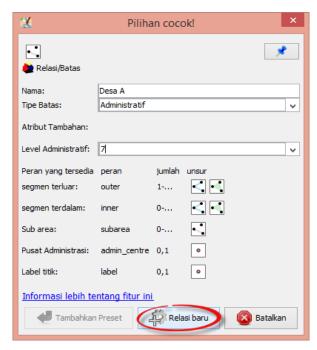
- Membuat Relasi Batas Administrasi (Contoh: Desa)
- 1. Langkah selanjutnya kita akan memulai membuat sebuah relasi untuk membuat batas administrasi Desa A. Pilih **salah satu** segmen garis pembentuk batas Desa A.



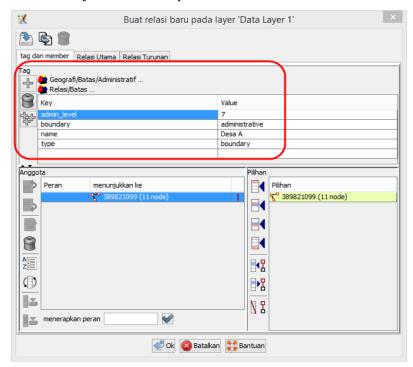
2. Kemudian pilih Presets \rightarrow Relasi \rightarrow Batas



3. Kemudian akan muncul jendela preset seperti ini. Berikan nama wilayah administrasinya (contoh: Desa A), kemudian pada kolom Tipe Batas pilih Administratif, dan pada kolom Level Administratif pilih sesuai dengan tingkat daerahnya (contoh: Desa → tingkat administrasi = 7). Setelah sudah diisi, klik **Relasi Baru**



4. Selanjutnya akan terbuka jendela relasi seperti dibawah ini:



Dari gambar di atas, perhatikan pada bagian Tag. Sudah terdapat 4 buah tag, diantaranya:

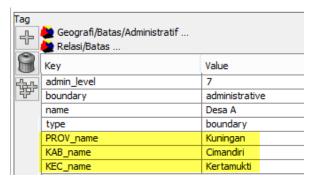
Key	Value			
admin_level	7			

boundary	adminstrative	
name	Desa A	
type	boundary	

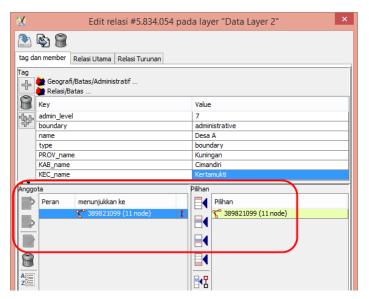
Kita butuh menambahkan beberapa tag lagi yang berisikan informasi nama tingkat administrasi yang lebih tinggi dari desa, diantaranya:

Key	Value
PROV_name	**nama provinsi
KAB_name	**name kabupaten
KEC_name	**name kecamatan

5. Untuk menambahkan tag, Anda bisa klik tombol , kemudian ketikkan pada kolom Key dan Value.



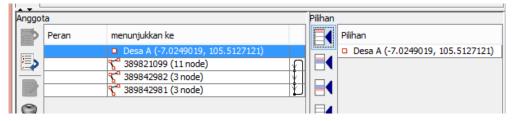
6. Selanjutnya, perhatikan pada bagian anggota relasi. Pada list anggota relasi, baru terdapat satu anggota. Selanjutnya kita akan menambahkan segmen-segmen lainnya yang membentuk batas Desa A ke dalam relasi ini. Silahkan pilih segmen tersebut satu per satu.



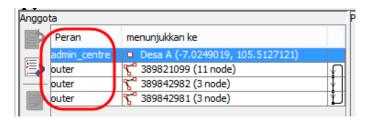
7. Biarkan jendela relasi **TETAP TERBUKA**. Pilih segmen garis lainnya yang membentuk batas Desa A. Setelah memilih, klik tombol pada jendela relasi. Maka jumlah anggota relasi akan bertambah.



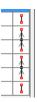
- 8. Lakukan langkah yang sama pada segmen garis pembentuk batas Desa A yang lainnya.
- 9. Jika semua segmen garis pembentuk batas Desa A sudah ditambahkan sebagai anggota, selanjutnya pilih titik pusat Desa A, klik tombol pada jendela relasi



10. Setelah anggota relasi Desa A lengkap, kita kemudian akan menentukan peran dari setiap anggota relasi. **Segmen garis pembentuk batas Desa A** kita beri peran sebagai "**outer**" atau sebagai batas luar dari wilayah administrasi. Sedangkan **Titik pusat Desa A** kita beri peran sebagai "**admin_centre**"



11. Pastikan daftar segmen pembentuk batas administrasi sudah urut. Jika daftar segmen belum urut, maka garis di sebelah kanan daftar tidak akan menutup seperti ini:



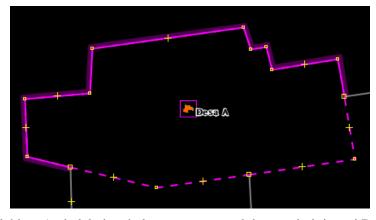
Untuk membuat daftar segmen sesuai urutan, klik tombol Sortir Relasi Member



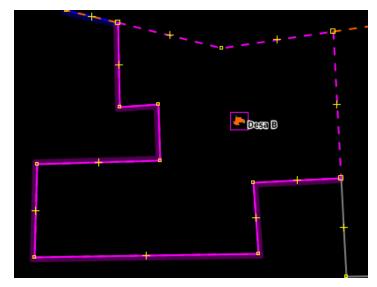
12. Jika daftar segmen sudah berurutan, maka garis di sebelah kanan daftar akan menutup seperti ini:



- 13. Setelah memberikan peran pada tiap anggota relasi dan memastikan daftar segmen pembentuk batas administrasi sudah urut, klik tombol OK
- 14. Sekarang klik dua kali di dalam area administratif Desa A. Jika relasi sudah terbentuk, maka wilayah Desa A akan berwarna ungu seperti ini:



15. Selanjutnya, silahkan Anda lakukan hal yang sama untuk batas administrasi Desa B (Ikuti langkah-langkah mulai dari bagian Pemberian Tag pada Garis Batas Administrasi hingga langkah-langkah sampai nomor 14 pada bagian Membuat Relasi Batas Administrasi Desa). Jika sudah berhasil maka hasilnya akan seperti ini:

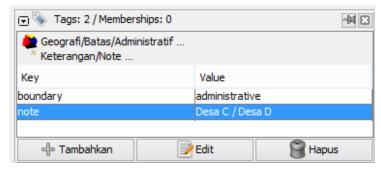


16. Sebelum kita lanjut untuk membuat batas Desa C, kita akan mencoba untuk mendigitasi batas desa baru yang ada di dalam wilayah Desa C. Mari kita sebut desa baru ini sebagai Desa D.



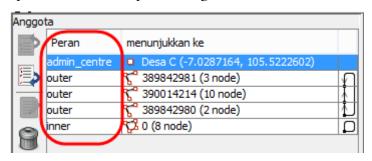
17. Pilih segmen garis batas Desa D yang baru saja Anda gambar. Berikan tag pada garis yang terpilih tersebut dengan cara pergi ke panel **Tags/Membership** → **Tambahkan**.

Tambahkan Key dan Value seperti gambar di bawah ini:



18. Untuk membuat batas administrasi Desa C, silahkan Anda lakukan langkah yang sama mulai

- dari langkah-langkah bagian Pemberian Tag pada Garis Batas Administrasi hingga langkah hingga nomor 9 pada bagian Membuat Relasi Batas Administrasi
- 19. **Pastikan segmen garis batas antara Desa C dan Desa D** sudah masuk ke dalam daftar anggota relasi Desa C.
- 20. Setelah anggota relasi Desa C lengkap, saatnya kita menentukan peran dari setiap anggota relasi.
 - Segmen garis pembentuk batas luar Desa C (garis batas antara Desa C/A, garis batas antara Desa C/B, dan garis batas antara Desa C/Laut) → berperan sebagai "outer" atau sebagai batas luar dari wilayah administrasi,
 - Segmen garis batas dalam Desa C (garis batas antara Desa C/D) → berperan sebagai "inner" atau sebagai batas dalam dari wilayah administrasi.
 - Titik pusat Desa C → berperan sebagai "admin_centre"

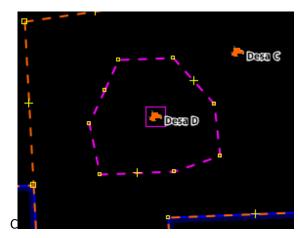


- 21. Klik tombol Sortir Relasi Member , untuk memastikan daftar anggota relasi sudah urut.
- 22. Sekarang klik dua kali di dalam area administratif Desa C. Jika relasi sudah terbentuk, maka wilayah Desa C akan berwarna ungu seperti ini:



23. Setelah membuat batas administrasi Desa C, mari kita lanjutkan untuk membuat batas Desa D. Silahkan Anda lakukan hal yang sama mulai dari langkah-langkah bagian Pemberian

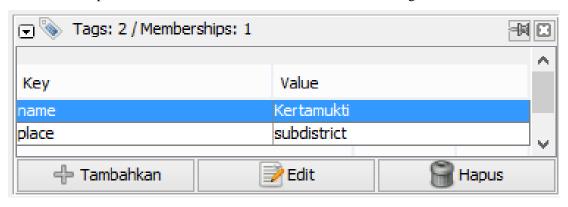
Tag pada Garis Batas Administrasi hingga langkah hingga nomor 14 pada bagian Membuat Relasi Batas Administrasi (Desa). Jika sudah berhasil maka hasilnya akan seperti ini:

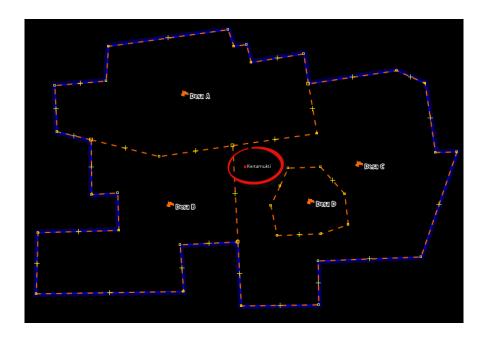


• Membuat Relasi Batas Administrasi (Contoh: Kecamatan)

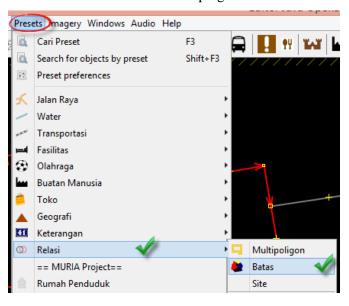
Pada langkah-langkah sebelumnya kita sudah membuat batas administrasi dalam tingkat desa, sekarang kita akan membuat batas administrasi satu tingkat di atas desa, yaitu kecamatan. Langkah yang akan kita lakukan kurang lebih sama dengan langkah-langkah sebelumnya.

Mari kita buat titik pusat **Kecamatan Kertamukti**, kemudian berikan tag:

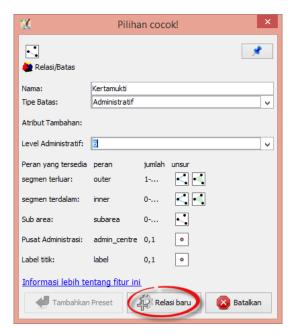




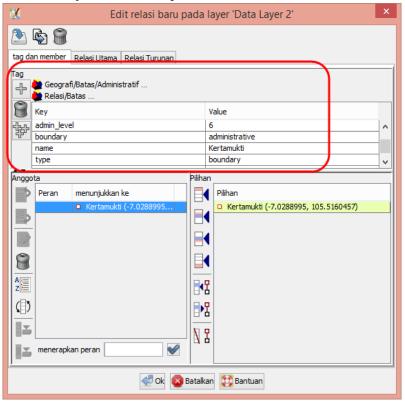
1. Pilih titik pusat Kecamatan Kertamukti. Kemudian pergi ke **Presets Relasi Batas**



Kemudian akan muncul jendela preset seperti ini. Berikan nama wilayah kecamatannya (contoh: Kertamukti), kemudian pada kolom Tipe Batas pilih Administratif, dan pada kolom Level Administratif pilih sesuai dengan tingkat daerahnya (contoh: Kecamatan → tingkat administrasi = 6). Setelah sudah diisi, klik Relasi Baru.



3. Selanjutnya akan terbuka jendela relasi seperti dibawah ini:



Dari gambar di atas, perhatikan pada bagian Tag. Sudah terdapat 4 buah tag, diantaranya:

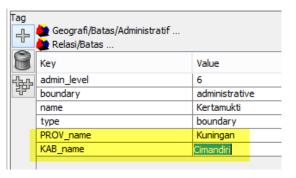


admin_level	6
boundary	administrative
name	Kuningan
type	boundary

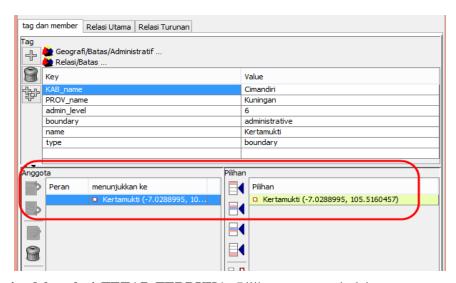
Kita butuh menambahkan beberapa tag lagi yang berisikan informasi nama tingkat administrasi yang lebih tinggi dari desa, diantaranya:

Кеу	Value
PROV_name	**nama provinsi
KAB_name	**name kabupaten

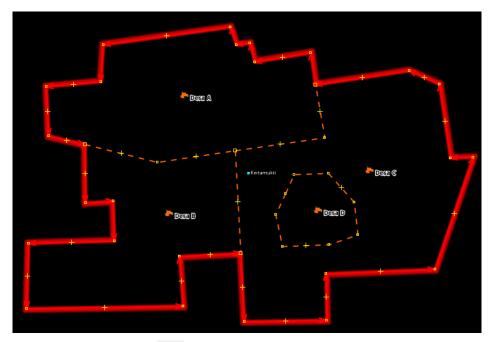
4. Untuk menambahkan tag, Anda bisa klik tombol , kemudian ketikkan pada kolom Key dan Value.



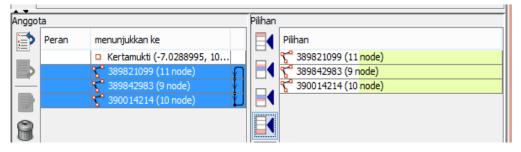
5. Selanjutnya, perhatikan pada bagian anggota relasi. Pada list anggota relasi, baru terdapat satu anggota. Selanjutnya kita akan menambahkan segmen-segmen lainnya yang membentuk batas Kecamatan Kertamukti ke dalam relasi ini.



6. **Biarkan jendela relasi TETAP TERBUKA**. Pilih segmen garis lainnya yang membentuk batas Kecamatan Kertamukti.

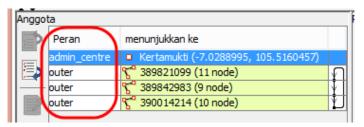


7. Setelah memilih, klik tombol pada jendela relasi. Maka jumlah anggota relasi akan bertambah.



8. Setelah anggota relasi Kecamatan Kertamukti lengkap, saatnya kita menentukan peran dari setiap anggota relasi. Segmen garis pembentuk batas Kecamatan Kertamukti kita beri peran

sebagai "outer" atau sebagai batas luar dari wilayah administrasi. Sedangkan Titik pusat Kecamatan Kertamukti kita beri peran sebagai "admin_centre"



9. Pastikan daftar segmen pembentuk batas administrasi sudah urut. Jika daftar segmen belum urut, maka garis di sebelah kanan daftar tidak akan menutup seperti ini:



Untuk membuat daftar segmen sesuai urutan, klik tombol Sortir Relasi Member



10. Jika daftar segmen sudah urut, maka garis di sebelah kanan daftar akan menutup seperti ini:



- 11. Setelah memberikan peran pada tiap anggota relasi dan memastikan daftar segmen pembentuk batas administrasi sudah berurutan, klik tombol OK
- 12. Cek hasil batas administrasi yang sudah anda buat di www.openstreetmap.org

2.2 Contoh Kesalahan Data di OpenStreetMap

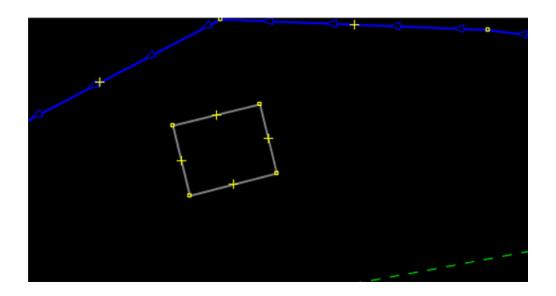
Setelah kita melihat berbagai macam panduan dalam mendijitasi objek-objek di OpenStreetMap, sekarang kita akan mempelajari beberapa kesalahan-kesalahan yang sering terjadi pada data OpenStreetMap. Dalam OpenStreetMap terdapat 2 jenis kesalahan umum yang biasa terjadi yaitu kesalahan *tag* atau informasi dan kesalahan geometri. Kita akan mempelajari 2 kesalahan tersebut lebih jauh dalam subbab ini.

2.2.1 Kesalahan *Tag* atau Informasi

Kesalahan *tag* ataupun informasi merupakan kesalahan yang sering terjadi pada data OpenStreetMap. Kesalahan ini disebabkan oleh pemberian *tag* / informasi pada objek yang dipetakan tidak tepat ataupun tidak sesuai dengan aturan di OpenStreetMap. Perlu diingat, kesalahan *tag* ini lebih

difokuskan ke teknis pemberian *tag* pada objek yang kita petakan bukan isi informasi yang ada di objek tersebut karena untuk kebenaran isi informasi pada suatu objek hanya bisa dicek kebenarannya jika kita sudah melakukan survei lapangan ke objek tersebut. Berikut adalah beberapa jenis kesalahan *tag* ataupun informasi yang ada di OpenStreetMap:

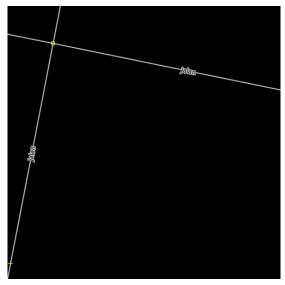
Objek Tidak Diberikan Tag



Untuk kesalahan seperti ini, kita harus segera memberikan *tag* sesuai dengan informasi yang sebenarnya pada objek tersebut. Jika hal ini tidak diperbaiki maka objek yang telah kita masukkan ke OpenStreetMap tidak akan dianggap sebagai sebuah objek oleh OpenStreetMap dan akibatnya ketika kita ingin mengambil data tersebut dari HOT Export atau yang lain objek tersebut tidak akan ada.

Kesalahan pemberian nama objek

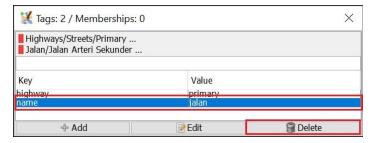
Terkadang kita memberikan sebuah nama pada objek yang kita gambar dengan nama yang terlalu general seperti di bawah ini:





Pada gambar di atas, terdapat beberapa objek jalan yang diberikan *tag* name=Jalan dan juga terdapat sebuah bangunan yang diberikan *tag* name=bangunan. Kedua jenis objek tersebut salah dalam hal pemberian nama karena objek tersebut diberikan sebuah nama yang terlalu general/umum. Setiap objek yang kita gambarkan di OpenStreetMap tidak perlu diberikan nama apabila kita tidak mengetahui nama dari objek yang kita gambarkan. Seringkali pengguna memberikan nama umum seperti jalan, rumah, bangunan, atau pemukiman. Padahal pemberian tag nama tersebut salah! Kita tidak perlu memberikan nama-nama seperti diatas karena identitas bangunan atau jalan sudah didapatkan dari tag lain seperti *highway=primary* atau *building=yes*. Cukup banyak pengguna yang menganggap perlu memberikan nama objek terutama objek jalan karena peringatan/warning "jalan belum dinamai/unnamed ways" akan muncul saat proses pengunggahan apabila ada jalan utama yang belum dinamai. Namun peringatan tersebut termasuk peringatan yang boleh diabaikan. Apabila kita tidak mengetahui nama sebuah jalan atau bangunan, kita bisa melewatkan pemberian nama tersebut dan beralih ke penggambaran objek lainnya. Jika Anda menemui objek yang sudah terlanjur diberikan nama yang terlalu general seperti gambar di atas, Anda dapat memperbaikinya dengan memilih objek

tersebut dan menghapus tag name dari objek tersebut lalu unggah perubahannya.



Tag Diberikan Pada Titik (Node) Bukan Pada Bangunan

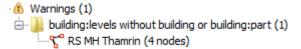
Satu masalah umum yang sering terjadi yaitu para pengguna menambahkan *tag* pada titik (*node*) yang terdapat pada bagian dari sebuah garis atau poligon. Hal ini terjadi ketika seseorang menggambar sebuah bangunan dan kemudian memilih titik (*node*) di bangunan tersebut. Pada saat ingin memberikan *tag* ke beberapa objek, pengguna kadang memilih melakukan pilihan secara keseluruhan (*select all*) sehingga baik itu titik (*node*) dan garis dari objek tersebut terpilih semua kemudian langsung menambahkan *tag* pada titik-titik (*node*) tersebut. Ingat! ini merupakan cara yang salah. Untuk menghindari kesalahan seperti ini, silahkan klik secara langsung pada bagian garis dari bangunan yang ingin kita berikan *tag* lalu masukkan informasi yang sesuai dengan objek tersebut. Berikut adalah contoh dari kesalahan ini:



Dapat kita lihat pada contoh di atas, **objek** sebelah **kanan benar** karena diberikan *tag* pada **bangunan**nya yaitu Rumah Sakit sedangkan **objek** sebelah **kiri salah** karena *tag* diberikan pada **setiap titik** (*node*) di bangunan tersebut. Hal ini dapat dilihat dari simbol Rumah Sakit di tiap sudut bangunan. Hal ini tentu akan mempengaruhi perhitungan objek yang ada di OpenStreetMap. Seharusnya rumah sakit tersebut dihitung hanya satu objek rumah sakit, akan tetapi di objek sebelah kiri akan dihitung ada 4 rumah sakit.

Bangunan yang tidak memiliki keterangan bangunan

Seringkali saat ingin meng*upload* data kita ke OpenStreetMap kita menemukan data kita diberikan peringatan seperti ini :



Ketika melihat peringatan seperti ini kita sering kali bingung dimana letak kesalahannya. Peringatan ini memberitahukan kepada kita seperti contoh di atas bahwa objek dengan nama **RS MH Thamrin** memiliki *tag* atau informasi **spesifik tentang bangunan** seperti jumlah lantai dan struktur bangunan **akan tetapi tidak ditandai** sebagai **bangunan**. Berikut adalah *tag* dari objek tersebut:

Key	Value
access:roof	yes
addr:full	Jl. Salemba Tengah
amenity	hospital
building:levels	>7
building:roof	
building:structure	reinforced_masonry
building:walls	brick
name	RS MH Thamrin
operator:type	private
rooms	131

Dapat kita lihat bahwa beberapa informasi terkait bangunan diberikan pada objek tersebut akan tetapi tag building = * tidak diberikan sehingga OpenStreetMap 'bingung' apakah objek tersebut memang sebuah bangunan ataukah hanya sebuah titik (node) atau garis (way). Oleh karena itu, silahkan kita memasukkan keterangan bangunan pada objek tersebut. Dalam contoh ini, objek yang diberikan peringatan adalah rumah sakit oleh karena itu silahkan tambahkan tag building = hospital sehingga tag objek tersebut terlihat seperti berikut:

Key	Value
access:roof	yes
addr:full	Jl. Salemba Tengah
amenity	hospital
building	hospital
building:levels	>7
building:roof	concrete
building:structure	reinforced_masonry
building:walls	brick
name	RS MH Thamrin
operator:type	private
rooms	131

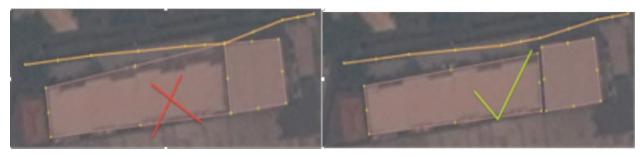
Jadi, inti dari kesalahan ini adalah jika kita menggambar sebuah **objek** berupa **bangunan** maka *tag* utama yang **harus diberikan** adalah **building** = * sesuai dengan jenis bangunan tersebut. Jika kita ragu untuk menentukan jenis bangunan pada suatu ada objek cukup berikan tag **building** = **yes** pada objek tersebut sehingga data kita tidak lagi dilihat sebagai sebuah kesalahan lagi di OpenStreetMap.

2.2.2 Kesalahan Geometri

Kesalahan geometri adalah kesalahan yang sangat banyak ditemukan di OpenStreetMap. Ini merupakan kesalahan yang terjadi baik itu pada bentuk ataupun hubungan antara objek-objek yang ada di OpenStreetMap. Secara umum kesalahan geometri yang terjadi di OpenStreetMap menggunakan aturan-aturan topologi untuk data spasial. Berikut adalah beberapa kesalahan geometri yang sering ditemukan di OpenStreetMap:

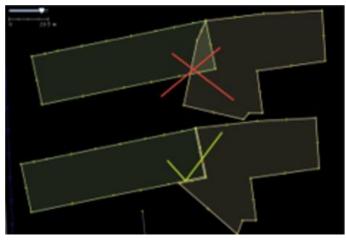
Objek Bangunan dan Jalan yang Saling Terhubung

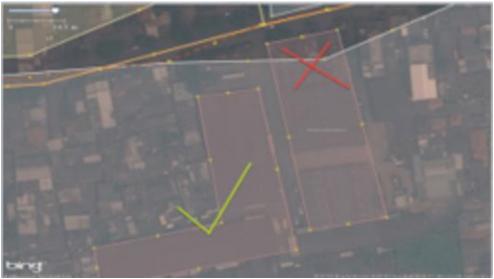
Beberapa objek sebaiknya tidak terhubung. Ketika anda menggambar sebuah objek garis yang berdekatan dengan bangunan (poligon) atau sebaliknya maka kedua objek tersebut tidak boleh terhubung. Contohnya, titik (*node*) jalan raya tidak boleh menempel ke bangunan, karena secara logika tidak mungkin sebuah jalan raya secara langsung menempel di dinding suatu bangunan! Jika anda menemukan contoh seperti itu silahkan pisahkan objek tersebut dengan menggunakan fitur *UnGlue Ways* di JOSM anda.



Objek Yang Saling Tumpang Tindih

Salah satu kesalahan umum lainnya di OpenStreetMap adalah objek yang saling tumpang tindah (*overlap*). Sebuah objek baik itu **bangunan** ataupun **jalan tidak boleh** saling **tumpang tindih** satu dengan yang lain. Sebagai contoh, poligon yang digambarkan sebagai taman di luar sebuah bangunan tidak boleh tumpang tindih dengan bangunan utama. Berikut adalah contohnya:





Dapat kita lihat pada contoh di atas, **objek** sebelah **kiri** benar karena **tidak memotong objek** yang lain **seperti** di **objek** sebelah **kanan**. Sekarang muncul pertanyaan "Mengapa objek sebelah kiri benar padahal sepertinya dia berada dalam suatu area/poligon?" Kalau diperhatikan objek sebelah kiri **berada di dalam poligon area penggunaan lahan** dan seperti yang telah dibahas di subbab sebelumnya untuk penggambaran penggunaan lahan **diperbolehkan** untuk saling **tumpang tindih** dengan objek lain.

Persimpangan Jalan Tidak Tersambung dan Tidak Tegak Lurus

Ketika anda menemukan suatu persimpangan jalan, anda tidak boleh menggambarkan pertemuan jalan raya tersebut dengan melengkung. Anda harus menggambarkan secara tegak lurus dan titik (node) pada jalan raya tersebut harus terhubung satu dengan yang lain.





Untuk jalan yang berbeda segmen sebaiknya digambarkan berbeda objek, jadi satu objek untuk satu segmen jalan. Untuk lebih jelasnya, perhatikan contoh berikut:



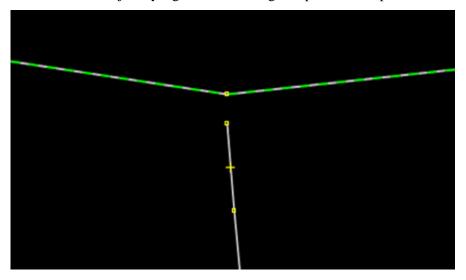
Gambar di atas merupakan contoh beberapa segmen jalan yang digambarkan sebagai satu objek jalan. Untuk memisahkannya menjadi beberapa segmen, Anda dapat menggunakan tools split way pada nodes pemisah segmen. Gambar di bawah merupakan contoh hasil pemisahan segmen dari jalan tersebut:



Jalan / Garis Tidak Terhubung

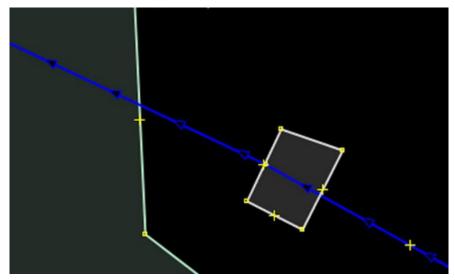
Sesuai dengan aturan topologi untuk data spasial maka segala jenis objek garis dan jalan harus

saling terhubung satu dengan yang lain. Jika ada garis yang tidak saling terhubung maka itu garis / jalan tersebut akan dianggap sebuah kesalahan oleh OpenStreetMap. Jika menemukan kesalahan seperti ini anda harus segera menghubungkan titik (node) yang saling berdekatan diantara garis/jalan tersebut. Berikut adalah contoh jalan yang tidak terhubung di OpenStreetMap:



Garis / Jalan Melewati Bangunan

Secara logika sederhana tidak akan ada jalan / sungai yang berada di tengah-tengah bangunan. Oleh karena itu jika terdapat objek garis / jalan yang melewati (*crossing*) sebuah bangunan maka OpenStreetMap akan menganggap hal itu adalah sebuah kesalahan. Silahkan perbaiki posisi salah satu objek dengan menggunakan bantuan citra satelit atau jika tidak terlihat dari citra satelit silahkan survei langsung ke lokasi objek tersebut. Berikut adalah contoh dari kesalahan garis / jalan yang melewati bangunan.



Beberapa contoh kesalahan-kesalahan yang disebutkan di atas merupakan kesalahan data yang paling sering ditemui di OpenStreetMap. Sebetulnya masih banyak lagi kesalahan data yang mungkin akan

kita temui di OpenStreetMap. Tapi anda tidak perlu khawatir kalau tidak mengetahui semua kesalahan data yang ada karena kita akan menemukannya dengan menggunakan alat-alat validasi di JOSM yang akan kita bahas di bab selanjutnya.

Mengedit di Luar Batas Area Download

Seperti yang kita ketahui untuk melakukan edit data OpenStreetMap di JOSM kita harus men-download data-data yang ada di area yang ingin kita edit terlebih dahulu. Hal ini bertujuan agar nantinya kita dapat mengetahui objek apa saja yang sudah dan belum terpetakan di area yang kita download tersebut. Selain itu, hal ini juga dilakukan untuk menghindari konflik data di OpenStreetMap. Walaupun begitu, seringkali pengguna ketika melakukan edit sering lupa untuk melihat batas daerah yang mereka download akibatnya mereka melakukan edit di luar batas area dan ketika ingin meng-upload terjadi konflik yang membuat data OpenStreetMap menjadi kacau.



Pada gambar di atas dapat terlihat bahwa batas area kita adalah kotak polos yang terlihat jelas sedangkan area dengan garis-garis diagonal merupakan area yang tidak dapat diedit. Oleh karena itu jangan mengedit objek yang berada di luar batas area terdownload. Jika objek tersebut berada di dalam dan di luar batas maka sebaiknya jangan melakukan editing terhadap objek tersebut. Untuk mengatasinya, anda dapat mempeluas area yang ingin anda edit dengan cara men-download kembali di JOSM.

2.3 Menjaga Kualitas Data di OpenStreetMap

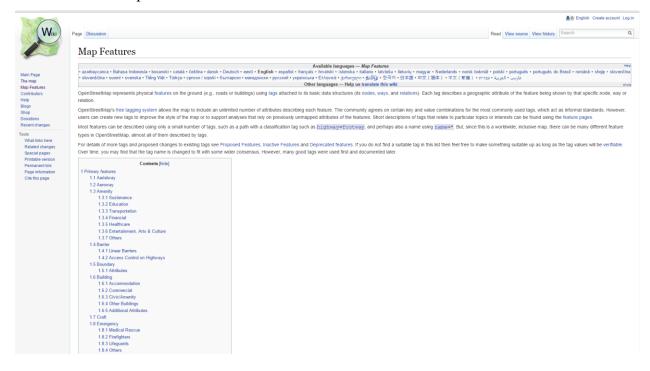
Ketika kita ingin menambahkan data atau melakukan edit serta memantau data-data yang telah kita

petakan tentu kita harus mengetahui acuan tempat yang menyediakan informasi tersebut. Hal ini kita lakukan agar nantinya objek-objek yang kita petakan tidak salah dan sesuai dengan ketentuan aturan OpenStreetMap. Selain itu, kita bisa selalu menjaga agar kualitas data di OpenStreetMap selalu baik. Ada beberapa tautan (*link*) yang akan kita pelajari dalam subbab ini dimana terbagi dalam tiga fungsi yaitu tautan untuk mencari informasi di OpenStreetMap, memantau data dan mendeteksi kesalahan di OpenStreetMap.

2.3.1 Acuan Tag / Informasi di OpenStreetMap

Dalam memberikan informasi terhadap objek yang dipetakan kita perlu memastikan bahwa informasi tersebut sudah benar dan sesuai dengan ketentuan dari OpenStreetMap. Untuk itu kita perlu memastikan acuan yang benar dari objek tersebut. OpenStreetMap sudah menyediakan halaman wikipedia khusus yang dapat dijadikan acuan secara umum. Halaman tersebut adalah *wiki map features*. Anda dapat mengunjunginya di http://wiki.openstreetmap.org/wiki/Map_Features.

Berikut adalah tampilan utama dari halaman tersebut



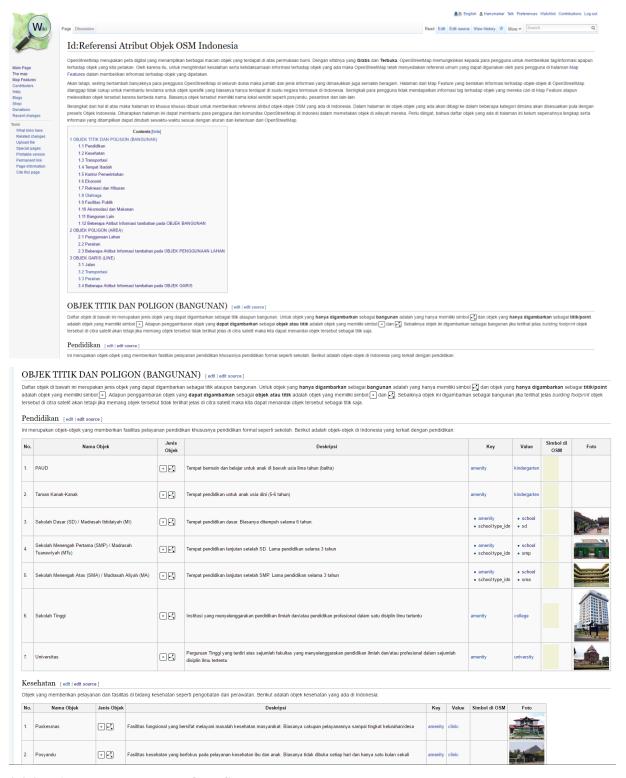
Amenity

Used to map facilities used by visitors and residents. For example: toilets, telephones, banks, pharmacies, cafes, parking and schools. See the page Amerities for an introduction on its usage

Key	Value	Element	Comment	Rendering	Photo
	Sustenance				
amenity	bar	•••	Bar is a purpose-built commercial establishment that sells alcoholic drinks to be consumed on the premises. They are characterised by a noisy and vibrant atmosphere, similar to a party and usually don't sell food. See also the description of the tags amenity=pub;bar;restaurant for a distinction between these	Y	
amenity	bbq	•	BBQ or Barbecue is a permanently built grill for cooking food, which is most typically used outdoors by the public. For example these may be found in city parks or at beaches. Use the tag fuel=* to specify the source of heating, such as fuel=wood;electric;charcoal. For mapping nearby table and chairs, see also the tag tourism=picnic_site. For mapping campfires and firepits, instead use the tag leisure=firepit.		
amenity	biergarten	•••	Biergarten or beer garden is an open-air area where alcoholic beverages along with food is prepared and served. See also the description of the tags amenity=pub;bar;restaurant. A biergarten can commonly be found attached to a beer hall, pub, bar, or restaurant. In this case, you can use biergarten=yes additional to amenity=pub;bar;restaurant.		4
amenity	cafe	•••	Cafe is generally an informal place that offers casual meals and beverages, typically, the focus is on coffee or tea. Also known as a coffeehouse/shop, bistro or sidewalk cafe. The kind of food served may be mapped with the tags outsine=* and diet=*. See also the tags amenity=restaurant; bar; fast_food.	₽	- Flaton
amenity	drinking_water	0	Drinking water is a place where humans can obtain potable water for consumption. Typically, the water is used for only drinking. Also known as a drinking fountain or bubbler.	Ť	
amenity	fast_food	•••	Fast food restaurant (see also amenity=restaurant). The kind of food served can be tagged with quisine=* and diet=*.	=	The state of the s
amenity	food_court	•••	An area with several different restaurant food counters and a shared eating area. Commonly found in malls, airports, etc.	• Ψ	
amenity	ice_cream	•••	Ice cream shop or ice cream parlour. A place that sells ice cream and frozen yoghurt over the counter	*	
amenity	pub	•••	A place selling beer and other alcoholic drinks; may also provide food or accommodation (UK). See description of amenity=bar and amenity=pub for distinction between bar and pub	₹	FELT CALLER
amenity	restaurant	••4	Restaurant (not fast food, see amenity=fast_food). The kind of food served can be tagged with quisine=* and dist=*.	₽ Ψ	200 m 00

Di *map feature* ini kita akan melihat berbagai jenis objek yang dapat dipetakan di OpenStreetMap beserta dengan keterangan dan gambarnya. Perlu diingat bahwa objek-objek yang ditampilkan di *map feature* ini berlaku global artinya objek-objek yang terdapat di luar negeri juga akan ditampilkan seperti jalur kereta salju, penitipan bayi, keran air minum. Selain itu penamaan objek-objek tersebut mungkin kurang dikenal bagi orang Indonesia. Akibatnya, banyak dari objek-objek tersebut kurang dipahami dan diketahui oleh kebanyakan pengguna di Indonesia karena sangat jarang ditemui oleh mereka. Oleh karena itu, jika anda hanya ingin mencari objek-objek khusus yang ada di Indonesia dengan nama yang sudah disesuaikan dan sudah dipahami bersama, silahkan kunjungi halaman wikipedia yang telah HOT Indonesia buat di sini:

http://wiki.openstreetmap.org/wiki/Id:Referensi Atribut Objek OSM Indonesia



2.3.2 Alat Pemantau Data OpenStreetMap

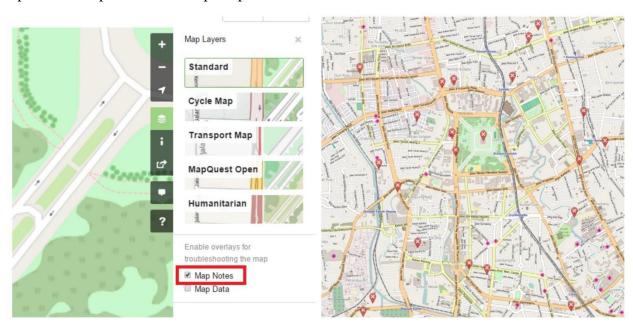
Konsep dasar OpenStreetMap adalah peta terbuka dimana semua pengguna bisa memasukkan data-data mereka ke dalamnya. Begitu banyak data yang dimasukkan sehingga seringkali kita harus selalu memastikan apakah data-data yang dimasukkan sudah benar atau belum. Untuk itu, kita dapat selalu memantau data OpenStreetMap di area yang kita inginkan. Dalam

memantau data di OpenStreetMap, kita dapat membagi menjadi 2 jenis cara pemantauan data yaitu memantau berdasarkan lokasi/wilayah dan memantau berdasarkan pengguna (*user*) OpenStreetMap. Mari kita bahas satu persatu masing-masing alat pemantau data yang dapat kita gunakan:

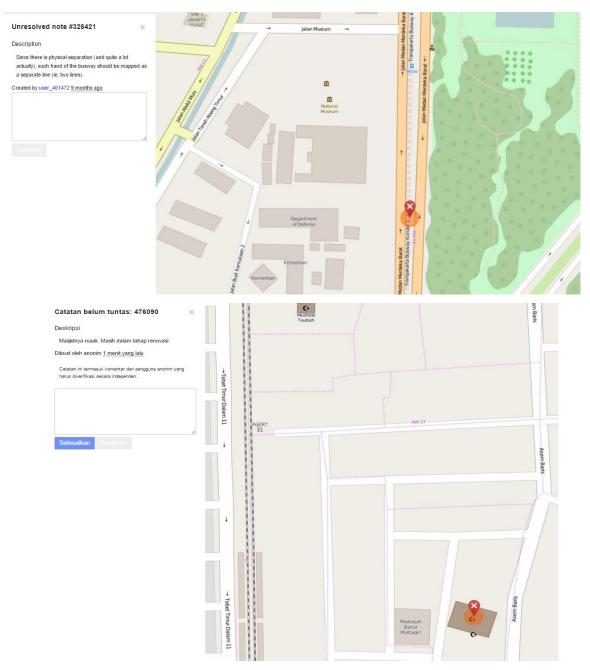
Alat Pemantau Data berdasarkan Lokasi

Catatan (*Notes*) di OpenStreetMap – Catatan (*notes*) merupakan salah satu fasilitas yang diberikan oleh OpenStreetMap kepada para pengguna agar bisa saling memberikan catatan terhadap objek-objek tertentu yang ada di OpenStreetMap. Catatan pada objek diberikan untuk memberitahukan kepada para pengguna lain jika objek tersebut masih memiliki kekurangan misalnya pada objek rumah sakit pengguna memberikan catatan bahwa rumah sakit tersebut sedang dibongkar ataupun dalam tahap pembangunan.

Catatan (*notes*) dapat kita munculkan pada halaman *website* OpenStreetMap. Ketika membuka halaman *website* OpenStreetMap kita silahkan klik **Map Layers** di sebelah kanan dari jendela komputer/laptop anda dan silahkan centang **Map Notes** sehingga kurang lebih OpenStreetMap anda akan tampil seperti berikut:



Tanda pin berwarna merah tersebut adalah catatan-catatan yang ada pada objek di area tersebut. Jika anda klik salah satu tanda merah maka akan ditampilkan informasi detail catatan tersebut berupa catatan objek, pembuat catatan, waktu catatan dibuat dan kotak komentar. Berikut adalah contohnya:



Jika anda menemukan catatan seperti contoh-contoh di atas, maka ada 2 pilihan untuk menyikapinya yaitu jika catatan tersebut bersifat hanya untuk menginformasikan kita seperti sebuah masjid masih dalam tahap renovasi maka kita bisa membalas komentar di catatan tersebut tapi tidak bisa mengedit ataupun memperbaiki catatan tersebut karena itu hanya berupa pengumuman / pemberitahuan saja. Akan tetapi, jika kita menemukan catatan yang sifatnya menyuruh kita memperbaiki objek tersebut maka kita harus segera memperbaikinya tetapi tolong diperhatikan bahwa untuk menambahkan catatan ataupun melihat catatan kita tidak harus masuk ke akun OpenStreetMap akan tetapi jika ingin memperbaiki / mengedit objek sesuai dengan catatan dan menandai catatan tersebut sudah diperbaiki

maka kita harus masuk ke akun OpenStreetMap kita terlebih dahulu.

Mengapa jika sekedar menambahkan catatan kita tidak harus masuk atau memiliki akun OpenStreetMap? Ini karena catatan yang ada di OpenStreetMap bersifat praktis dan memberikan informasi kepada para pengguna lain terutama mereka yang sudah memiliki akun OpenStreetMap secara cepat. Misalnya, ada orang yang melihat OpenStreetMap tetapi dia belum memiliki akun dan tiba-tiba melihat sebuah kesalahan pada suatu objek yang dia ketahui maka dia tidak harus membuat akun terlebih dahulu untuk menyampaikannya ke pengguna yang lain tapi cukup untuk menyampaikan catatan ke dalamnya kemudian biarkan pengguna yang sudah memiliki akun yang akan menyelesaikan masalah di catatan tersebut. Untuk bisa menambahkan catatan di OpenStreetMap silahkan klik tombol di sebelah kanan yang bertuliskan add new notes / tambahkan catatan pada peta hingga muncul kotak catatan dan pin berwarna biru yang bisa kita geser ke objek yang ingin kita berikan catatan.



Ingat! Kita tidak bisa memaksakan orang lain untuk mengenal OpenStreetMap dan membuat akun di dalamnya akan tetapi, kita bisa mengajak mereka untuk terus berkontribusi di OpenStreetMap dan salah satunya dengan menggunakan fasilitas catatan (notes) di OpenStreetMap ini.

Who Did It – Ini merupakan sebuah tautan yang membantu kita untuk bisa lebih spesifik memantau data-data OpenStreetMap di suatu wilayah. Ini merupakan tautan pengembangan OpenStreetMap yang dibuat oleh **Ilya Zverev** dengan lisensi WTFPL dan dimodifikasi dan

dikembangkan oleh **Simon Legner**. Tujuan utama dari tautan ini adalah untuk mengetahui **aktifitas memasukkan, meng-***edit* **dan menghapus data** OpenStreetMap **di suatu area** dengan memasukkan beberapa kata kunci seperti **nama pengguna, waktu aktifitas, bentuk objek** dan **area yang diinginkan**. Untuk membukanya silahkan ke http://simon04.dev.openstreetmap.org/whodidit/ . Berikut adalah tampilan awal dari tautan ini:



Sekarang kita bisa memasukkan beberapa kata kunci yang telah disebutkan pada kotak yang berada di sebelah kiri atas layar. Berikut adalah penjelasan untuk masing-masing kolom isian tersebut:

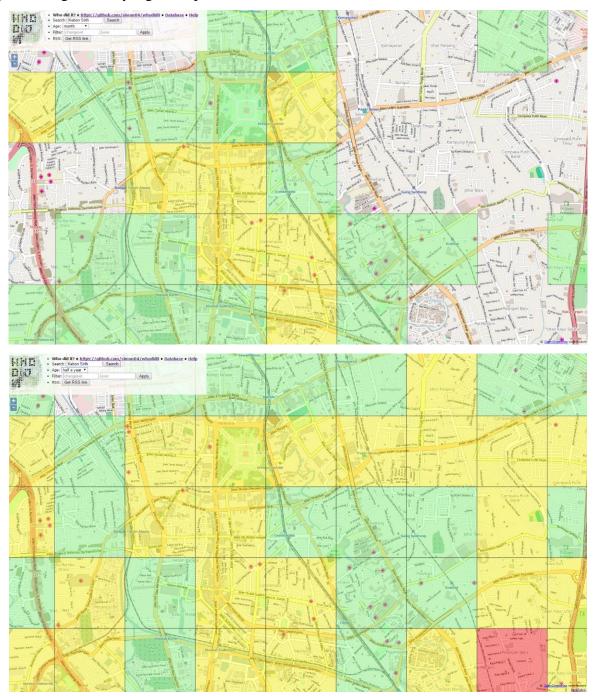


- 1: Kita bisa mencari area yang kita inginkan dengan menggunakan kotak Search ini
- 2: Kita bisa menentukan waktu aktifitas data di daerah tersebut. Pilihannya biasa dari 6 jam terakhir, 1 hari terakhir, 3 hari terakhir, 1 minggu terakhir, 1 bulan terakhir, 6 bulan terakhir

dan dari awal aktifitas.

- 3: Kita bisa mencari nama pengguna maupun jenis objek yang dimodifikasi.
- **4:** Mendapatkan tautan RSS dimana kita dapat memilih area sendiri dan ditaruh dalam blog/tautan kita.

Sekarang sebagai contoh, kita akan melihat aktifitas data OpenStreetMap di daerah **Kebon Sirih**, **Jakarta Pusat** dalam waktu **1 bulan terakhir** dan **6 bulan terakhir**. Berikut perbandingan hasil yang ditampilkan:



Gambar yang pertama adalah aktifitas modifikasi data dalam waktu 1 bulan terakhir dan gambar yang kedua adalah aktifitas modifikasi data dalam waktu 6 bulan terakhir. Bisa terlihat bahwa aktifitas modifikasi data di daerah Kebon Sirih dalam waktu 1 bulan terakhir lebih sedikit dibandingkan dengan dalam waktu 6 bulan terakhir. Hal ini bisa dilihat dari

jumlah kotak yang ada di gambar pertama lebih sedikit dan ada area yang tidak ada kotak sama sekali yang menandakan tidak ada modifikasi data di sana. Sedangkan di gambar kedua semua daerah terdapat kotak yang menandakan aktifitas modifikasi data di OpenStreetMap. Tiap-tiap kotak memiliki warna tertentu untuk menjelaskan perubahan data yang terjadi. Untuk perbedaan warna-warna tiap kotak berikut adalah penjelasannya:

Kotak Merah:

Menunjukkan di daerah itu hanya ada aktifitas penghapusan objek/bagian objek

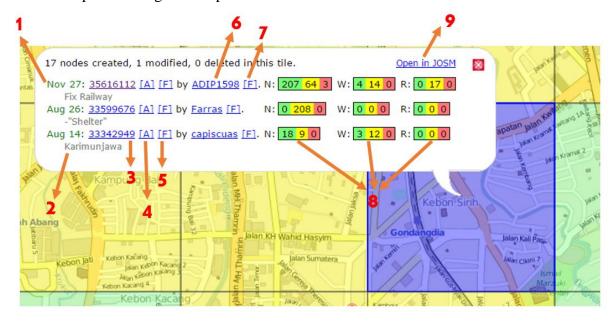
Kotak Kuning:

Menunjukkan di daerah tersebut terdapat **sedikit aktifitas penghapusan objek**/bagian objek dan/atau **banyak aktifitas edit/modifikasi objek**.

Kotak Hijau:

Menunjukkan di daerah tersebut terdapat sedikit aktifitas edit/modifikasi objek dan banyak aktifitas menambahkan objek baru.

Setelah sudah mendapatkan hasil pencarian yang dihasilkan, kita bisa memilih kotak yang ingin kita lihat aktifitas data di dalamnya. Ketika kita klik salah satu kotak maka kita akan melihat tampilan kurang lebih seperti berikut:



Berikut adalah keterangan dari beberapa tampilan di kotak tersebut :

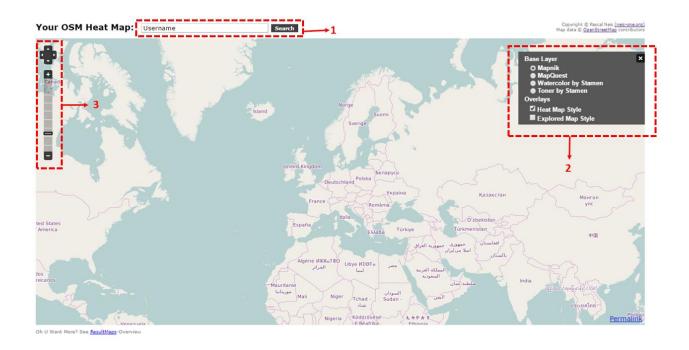
1	Tanggal Modifikasi Data (changeset) diunggah (upload) ke dalam OpenStreetMap
2	Komentar mengenai perubahan / modifikasi data (changeset) yang diunggah (upload) ke OpenStreetMap
3	Nomor id modifikasi data (changeset) di OpenStreetMap
4	Tautan (link) yang dapat menunjukkan informasi modifikasi data (changeset) di OpenStreetMap
5	Melakukan filter untuk menunjukkan modifikasi data (changeset) tertentu
6	Nama akun pengguna yang melakukan modifikasi data (changeset)
7	Melakukan filter untuk menunjukkan modifikasi data (changeset) yang dilakukan oleh akun tertentu
8	Jumlah modifikasi data (changeset) untuk tiap jenis-jenis objek dan jenis modifikasi datanya
9	Membuka kotak tersebut di JOSM

Kita bisa selalu melihat aktifitas modifikasi data di area tertentu dengan lebih spesifik dan detail. Jika kita menemukan kesalahan dalam data-data tersebut maka kita dapat memperbaikinya menggunakan alat validasi yang akan dijelaskan di bab selanjutnya. Perlu diingat, tautan (*link*) who did it ini hanya membantu kita untuk selalu bisa memantau data-data yang ada bukan untuk menampilkan kesalahan ataupun memperbaiki kesalahan yang ada di data OpenStreetMap. Untuk menampilkan kesalahan data di OpenStreetMap akan kita bahas di sub bab selanjutnya.

• Alat Pemantau Data berdasarkan Pengguna (user)

OSM Heat Map

Selain bisa memantau data berdasarkan lokasi ataupun wilayah di OpenStreetMap, kita juga bisa memantau data berdasarkan pengguna yang memasukkan data ke OpenStreetMap. Alat yang dapat kita gunakan memantau data berdasarkan fungsi ini adalah **OSM Heat Map** http://www.yosmhm.neis-one.org/ Berikut adalah tampilan dari OSM Heat Map:



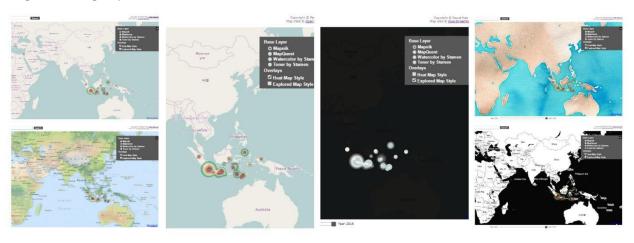
Keterangan:

1. Kotak Pencarian Username OpenStreetMap

Ini adalah alat utama di situs ini. Kita dapat memasukkan nama pengguna yang ingin kita lihat kegiatan pemetaannya di OpenStreetMap beserta dengan data-data yang dia masukkan.

2. Kotak Layer dan Overlays

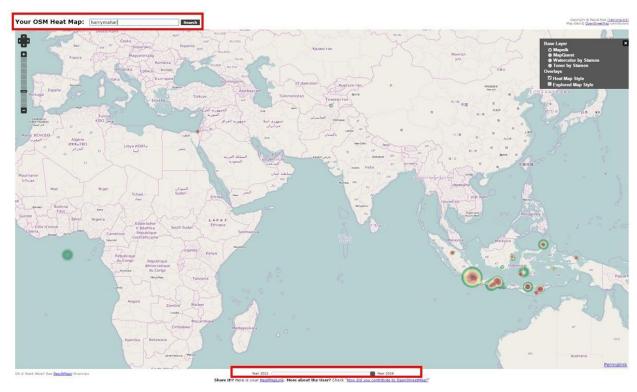
Ini adalah pilihan *layer* yang dapat kita pilih untuk kita tampilkan pada peta. Kita dapat memilih apakah tampilan seperti OpenStreetMap (*Mapnik*) ataupun hitam putih (*Toner by Stamen*) atau yang lain. Kita juga dapat memilih tampilan hasil pencarian apakah dilingkari dengan warna yang berbeda (*Heat Map Style*) atau disorot khusus (*Explored Map Style*)



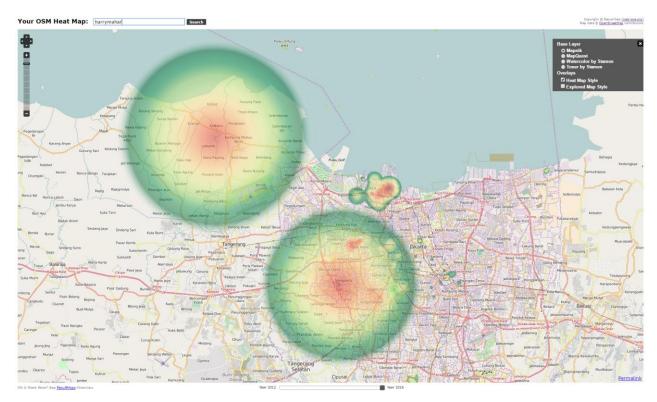
3. Alat Perbesaran dan Menggeser Peta

Alat ini mirip dengan alat-alat di tempat lain dimana fungsinya untuk memperbesar maupun memperkecil tampilan serta menggeser peta kita.

Untuk melihat data-data yang telah dimasukkan oleh pengguna tertentu, kita cukup memasukkan nama OpenStreetMap pengguna tersebut dan klik **Search.** Anda akan melihat tampilan kurang lebih seperti berikut:



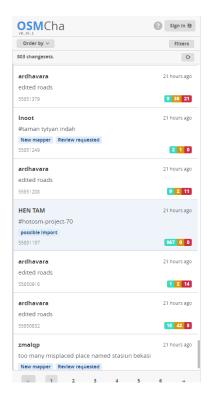
Setelah berhasil mendapatkan hasil dimana saja lokasi data-data yang diinput oleh pengguna tersebut. Anda dapat memperbesar ke daerah yang anda inginkan. Gradasi warna yang ditampilkan menunjukkan jumlah intensitas data yang diinput. Hijau menunjukkan data yang dimasukkan tidak terlalu banyak hingga merah menunjukkan data yang dimasukkan jumlahnya semakin banyak dan pengguna tersebut sering melakukan aktifitas di lokasi tersebut.



Selain itu anda juga dapat mengatur tahun saat pengguna tersebut melakukan aktifitas dimulai dari awal saat baru melakukan pemetaan pertama hingga terakhir. Anda dapat menggeser tahun aktifitas pemetaan di bagian bawah dari situs OSM Heat Map ini.

OSM Changeset Analyzer

OSMCha (beta) merupakan sebuah situs yang dikembangkan oleh Mapbox yang berfungsi untuk memantau penambahan atau perubahan data OpenStreetMap yang dilakukan oleh pengguna diseluruh dunia. Dengan menggunakan OSMCha, Anda bisa memantau setiap perubahan yang ada di sebuah area, spesifik pada username tertentu, dan bahkan spesifik pada changeset tertentu. Situs ini akan menyajikan setiap changeset yang diupload oleh semua user/user tertentu dengan memberikan gambaran lokasi yang ditambahkan/diubah di masing-masing changeset yang ditampilkan.





Validation tool for OpenStreetMap

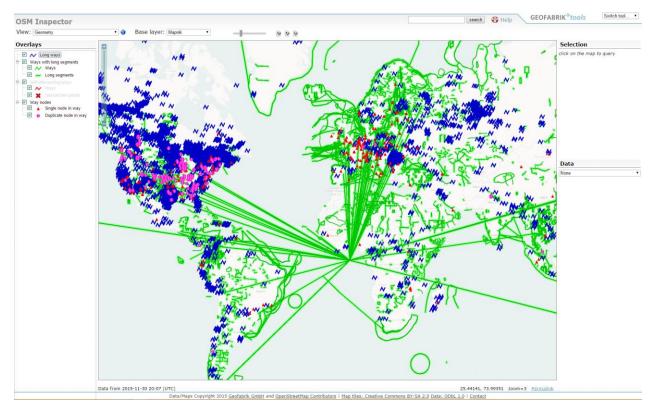


v0.36.1 •Guide |
•Github |
*File an issue

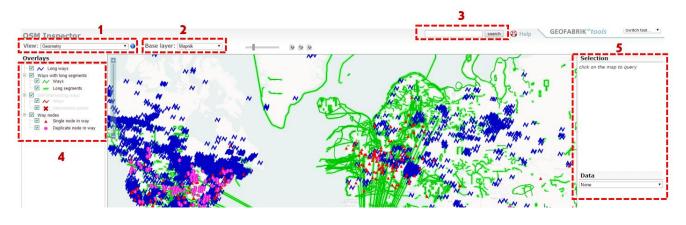
Pada setiap changeset yang muncul di sebelah kiri, Anda akan melihat tiga warna yaitu hijau (objek yang ditambahkan), kuning (objek yang dimodifikasi) dan merah (objek yang dihapus). Dengan menggunakan *OSMCha* anda bisa langsung mengecek data OSM yang ada pada setiap changeset menggunakan *iD* atau *JOSM* atau bahkan langsung memberikan komentar pada setiap changeset. Untuk mengakses situs ini Anda bisa langsung menuju ke https://osmcha.mapbox.com/.

2.3.3 Alat Pendeteksi Kesalahan Data OpenStreetMap

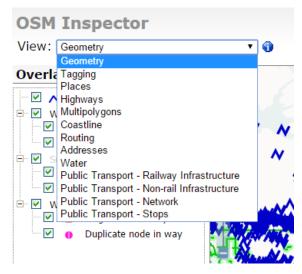
Setelah kita mengetahui tautan (*link*) untuk memantau data di OpenStreetMap, sekarang kita akan melihat bagaimana bisa melihat kesalahan data OpenStreetMap di daerah tertentu. Alat yang akan kita gunakan untuk dapat melihat kesalahan data di OpenStreetMap pada kali ini adalah *OSM Inspector*. Alat ini dikembangkan oleh *Geofabrik* dan sering digunakan untuk melihat kesalahan data OpenStreetMap. Untuk mengunjungi tautan ini silahkan klik di http://tools.geofabrik.de/osmi/. Berikut adalah tampilan dari *OSM Inspector*:



Secara garis besar *OSM Inspector* terdiri dari beberapa bagian yaitu daftar **Kesalahan Data OpenStreetMap**, **Peta OpenStreetMap** yang berisi kesalahan-kesalahan data dan **Keterangan Kesalahan Data**. Berikut adalah penjelasan untuk masing-masing bagian yang ada di *OSM Inspector*:



 View – Ini merupakan alat yang memiliki fungsi untuk melakukan filter terhadap kesalahan data OpenStreetMap yang ingin ditampilkan. Dalam view ini banyak terdapat pilihan jenis kesalahan data yang ada di OpenStreetMap seperti yang ditampilkan pada gambar di bawah ini:

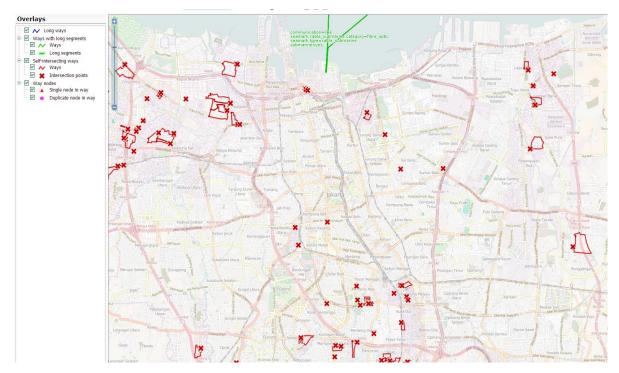


- Untuk kesalahan Geometry merupakan kesalahan yang berkaitan dengan hubungan antara objek seperti fisik objek yang telah dijelaskan pada subbab sebelumnya.
- Kesalahan *Tagging* adalah kesalahan yang terkait dengan pemberian *tag* ataupun informasi terhadap objek.
- Kesalahan *Places* merupakan kesalahan yang terkait dengan titik (*point*) seperti kota ataupun lokasi di suatu daerah.
- *Highways* merupakan kesalahan yang berkaitan dengan jalan raya seperti tipe jalan yang belum diketahui ataupun jumlah ruas jalan dan arah jalan.
- *Multipolygon* menampilkan kesalahan yang berkaitan dengan poligon yang saling berhubungan satu dengan yang lain.
- *Coastline* menampilakan kesalahan yang terjadi pada data-data garis pantai di OpenStreetMap
- Kesalahan *Routing* merupakan kesalahan yang berkaitan dengan data-data yang memiliki informasi rute di dalamnya. Sebagian besar jenis objek ini adalah garis (*ways*)
- *Addresses* merupakan kesalahan yang menunjukkan kesalahan pada penulisan alamat pada suatu objek. Biasanya objek ini merupakan bangunan / poligon
- Water merupakan kesalahan pada objek air seperti sungai, danau dan lain-lain.
- *Public Transport* merupakan kesalahan yang terjadi pada objek transportasi publik seperti rel kereta maupun rute bus umum.
- 2. Base Layer alat yang memungkinkan kita untuk memilih tampilan atau latar

belakang dari peta yang ditampilkan. Dalam *Base Layer* ini terdapat beberapa pilihan dari peta yang ingin kita tampilkan. Berikut adalah pilihan tampilan peta yang ada:

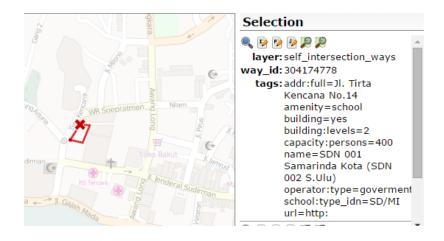


- None ketika kita memilih pilihan ini maka hanya kesalahan data saja yang akan ditampilkan tanpa memunculkan peta.
- **Geofabrik Standard** Pilihan ini akan memunculkan tampilan peta OpenStreetMap sesuai tampilan geofabrik. Peta ini mirip dengan tampilan peta asli OpenStreetMap.
- **Geofabrik German** Tampilan peta ini menggunakan *style* peta dari geofabrik Jerman. Peta yang ditampilkan terlihat lebih halus.
- **Geofabrik Topo** Pilihan ini akan memunculkan tampilan peta dengan memperlihatkan ketinggian dan bentuk permukaan bumi.
- Mapnik Pilihan ini akan memunculkan tampilan asli OpenStreetMap sesuai dengan website OpenStreetMap.
- **Public Transport** Pilihan ini memunculkan peta transport dimana hanya memunculkan jalan dan rute transportasi.
- 3. **Search** Ini merupakan menu pencarian dimana kita bisa mencari lokasi ataupun area yang ingin kita lihat kesalahannya. Fungsi ini sama dengan fungsi pencarian di *website* OpenStreetMap.
- 4. **Overlays** Ini merupakan kotak yang menampilkan daftar kesalahan sesuai dengan **view** yang telah kita pilih di area tertentu. Berikut adalah contoh daftar kesalahan dengan view **geometry**:

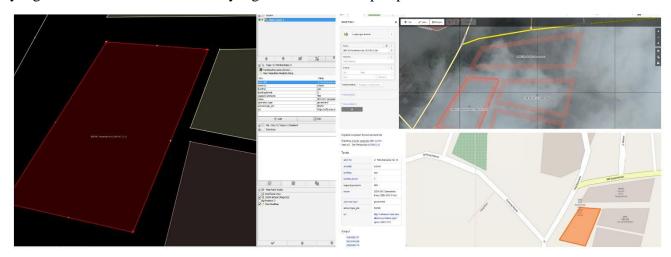


Dapat dilihat bahwa pada contoh di atas pada wilayah Jakarta untuk kesalahan jenis geometri terdapat 4 jenis kesalahan. **Daftar kesalahan** ini dapat **berubah** sesuai **dengan tingkat perbesaran** pada peta yang kita lihat. Semakin detail peta yang anda lihat maka kemungkinan daftar kesalahan di *overlays* semakin detail. Beberapa jenis kesalahan telah dijelaskan di subbab sebelumnya.

5. Selection – Ini merupakan tempat yang menunjukkan informasi kesalahan data yang kita pilih. Informasi yang ditampilkan terdiri dari informasi kesalahan, id OSM dan informasi / tag yang diberikan. Berikut adalah salah satu contoh informasi kesalahan data yang telah dipilih:



Terlihat bahwa dalam menu *Selection* kita bisa melihat informasi dari objek yang memiliki kesalahan yang dimana dalam gambar di atas kesalahannya adalah *self intersection ways* yang merupakan kesalahan geometri di bagian garis dari objek tersebut ada yang saling menyilang satu dengan yang lain. Anda dapat **memperbesar objek** tersebut dengan klik pada ikon **kaca pembesar** kemudian untuk **membuka** dan memperbaiki **objek** tersebut di **iD Editor** klik ikon kertas dengan huruf I, membuka di **Potlatch** klik ikon kertas dengan huruf P dan ikon serta hurus J untuk membuka di **JOSM**. Berikut adalah beberapa tampilan objek yang memiliki kesalahan tersebut yang dibuka di beberapa aplikasi:



Jika anda sudah mengetahui semua alat yang ada di *OSM Inspector* sekarang berlatihlah untuk selalu menggunakannya terutama ketika ingin mencari kesalahan data OpenStreetMap di daerah tertentu. Ingat bahwa *OSM Inspector* ini adalah alat hanya untuk mencari kesalahan dan ketika kita ingin memperbaiki data tersebut kita harus membukanya di *editor* seperti iD Editor ataupun JOSM.

Modul 3

Validasi Data OpenStreetMap

Tujuan Pembelajaran

- Mengetahui pengertian validasi data di OSM
- Mengetahui dan memahami cara melakukan validasi data menggunakan JOSM
- Mengetahui dan memahami cara melakukan validasi data menggunakan Tasking Manager
- Mengetahui dan memahami cara melakukan validasi data menggunakan Overpass
 Turbo

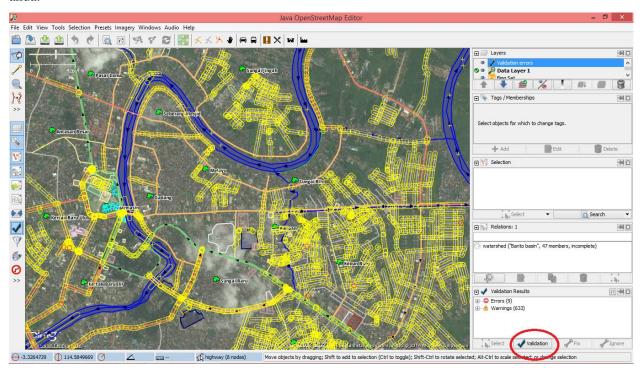
Setelah mempelajari pengertian penjaminan kualitas data spasial serta memahami berbagai macam hal terkait penjaminan kualitas data di OpenStreetMap seperti cara mendijitasi yang benar kemudian mengetahui jenis-jenis kesalahan data yang ada di OpenStreetMap. Kita juga telah mempelajari alat ataupun website yang dapat kita gunakan untuk memantau data OpenStreetMap dan memperbaiki kesalahan data yang ada di daerah tertentu. Sekarang, kita akan melihat bagaimana data-data yang sudah kita pantau dan telah kita temukan kesalahannya tersebut dapat diperbaiki sehingga memiliki kualitas data yang baik. Kegiatan memperbaiki kesalahan data di OpenStreetMap disebut dengan validasi data. Dalam panduan ini kita akan mempelajari tiga teknik melakukan validasi menggunakan tiga alat yang berbeda. Alat validasi yang akan kita pelajari dalam panduan ini yaitu JOSM, Tasking Manager dan Overpass Turbo sehingga dengan mengetahui beberapa alat validasi kita dapat melakukan validasi data OpenStreetMap sesuai dengan kebutuhan kita.

3.1 Validasi Data Menggunakan JOSM

JOSM merupakan salah satu editor yang paling sering digunakan untuk mengedit data OpenStreetMap. Salah satu alasan mengapa JOSM paling banyak digunakan adalah dapat digunakan secara *offline* serta banyaknya alat serta kedinamisan dalam mengedit data termasuk untuk melakukan validasi data. Berikut adalah cara kita bisa menggunakan JOSM langsung untuk melakukan validasi data:

- Download area yang ingin anda validasi
- Jika Anda tidak melihat toolbar "Hasil Validasi / Validation Result" pada jendela di panel

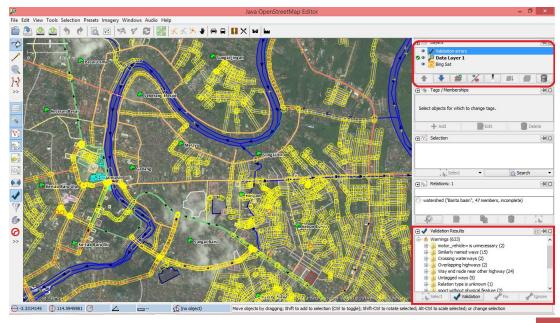
bagian kanan, Silahkan **klik** pada tanda **cheklist** berwarna **biru** pada bagian kiri dari JOSM anda.



Pastikan bahwa tidak ada bagian pada objek yang dipilih. Jika ada objek yang terpilih maka hanya objek tersebut yang akan divalidasi sedangkan kita ingin memvalidasi seluruh area yang telah kita download. Arahkan mouse anda pada jendela validasi dan klik "Validasi"



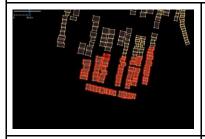
 Peta tersebut akan berubah dan segala macam Peringatan akan terlihat berwarna kuning dan untuk kesalahan berwarna merah. Pada jendela "Hasil Validasi" Anda akan melihat daftar peringatan dan kesalahan, jika ada.



- Kesalahan (*Error*) harus seluruhnya diperbaiki. Anda dapat memperbesar pada objek yang terdapat kesalahan dengan cara klik kanan pada kesalahan pada jendela tersebut dan kemudian pilihlah "Zoom to Problem". Setelah itu Anda bisa memperbaiki kesalahan tersebut secara manual.
- Beberapa kesalahan dapat diperbaiki secara otomatis, seperti "Titik yang terduplikasi" (Duplicated Nodes). Anda dapat melihat pada folder untuk melihat daftar kesalahan dan memilih tombol "Perbaiki" pada jendela tersebut. Beberapa kesalahan (error) membutuhkan untuk diperiksa secara manual.

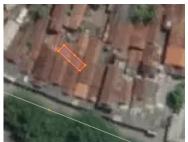
Biasanya terdapat lebih **banyak peringatan** (*warning*) daripada **kesalahan** (*error*). Dengan memberikan peringatan, JOSM memberitahukan Anda bahwa objek tersebut kemungkinan memiliki kesalahan. Anda harus mengeceknya untuk menentukan apakah hal tersebut merupakan merupakan benar kesalahan atau tidak. Beberapa kesalahan dan peringatan yang sering muncul adalah:

Error



Duplicated node (Titik terduplikasi)

Error/kesalahan ini muncul ketika sebuah objek terupload 2x atau lebih sehingga terdapat tumpang tindih titik pada posisi yang sama.



Duplicated ways (Garis terduplikasi)

Error/kesalahan ini muncul sama seperti dengan error *duplicated node*. Terdapat 2 atau lebih garis yang berada pada posisi yang sama yang disebabkan oleh objek yang terupload 2x atau lebih.



Unconnected coastline (Garis pantai tidak tertutup)

Error/kesalah ini muncul karena sebuah garis pantai tidak tertutup.

Warning



Crossing building (Bangunan bersinggungan)

Warning/peringatan ini muncul karena terdapat 2 atau lebih bangunan yang saling bersinggungan.



Crossing ways (Garis bersinggungan)

Peringatan ini muncul karena terdapat 2 jalan/garis yang tidak bersinggungan. Setiap percabangan jalan harus dibuat titik pada persimpangan tersebut. Apabila pada kondisi sebenarnya jalan tersebut terdapat terowongan/jalan layang, berikan tambahan tag *layer*=* sesuai dengan posisi segmen jalan yang bersinggungan dengan jalan/garis yang lain.



Overlapping highways (Jalan saling bertumpang tindih)

Peringatan ini muncul apabila terdapat bagian jalan yang saling bertumpang tindih dengna bagian jalan lain.



Node connect highways and building

Peringatan ini muncul apabila ada titik dari objek bangunan dan jalan yang menempel satu sama lain.



Untaged ways (Objek belum diberi tag)

Peringatan ini muncul apabila terdapat sebuah objek yang belum diberi tag



Way end node near other highways (titik tidak terhubung dengan garis terdekat)

Peringatan ini muncul apabila terdapat sebuah titik dari objek jalan yang berdekatan dengan jalan lainnya namun tidak terhubung. Apabila kondisi dilapangan jalanan tersebut tidak terhubung dengan jalan lain, abaikan peringatan ini.



Building inside building (bangunan didalam bangunan)

Peringatan ini muncul apabila terdapat sebuah objek bangunan yang berada didalam objek bangunan lainnya.



Unnamed ways (Jalan belum diberi nama)

Peringatan ini muncul apabila terdapat sebuah objek jalan yang belum diberikan nama jalan. Abaikan jika Anda memang belum mengetahui nama jalan tersebut. Jangan diberi nama general seperti misalnya name=Jalan!



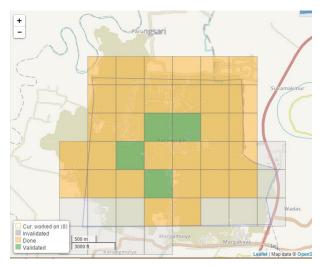
Similarly named ways (Jalan memiliki nama yang sama)

Peringatan ini muncul apabila terdapat sebuah jalan yang memiliki nama yang tidak jauh berbeda misalnya Jalan Bomber 1 dan Jalan Bomber 2. Abaikan jika pada kenyataannya jalan tersebut memang memiliki nama yang sama.

3.2 Validasi Data Menggunakan Tasking Manager

Alat lain yang dapat kita gunakan untuk melakukan validasi data OpenStreetMap adalah *Tasking Manager*. Seperti yang sudah kita pelajari pada panduan awal OpenStreetMap, tasking manager adalah alat untuk memetakan area tertentu di OpenStreetMap secara bersama-sama. Tujuan dari tasking manager ini adalah agar data-data yang diinput ke dalam OpenStreetMap tidak saling tumpang tindih satu sama lain dan dapat dipetakan secara cepat. Akan tetapi akibat banyak pengguna OpenStreetMap yang ikut memetakan dalam waktu cepat dan bersama-sama, kualitas data yang dimasukkan ke dalam OpenStreetMap berisiko memiliki banyak kesalahan. Oleh karena itu area-area yang telah selesai dipetakan di tasking manager perlu untuk divalidasi kembali. Biasanya dalam suatu

proyek yang menggunakan tasking manager, terdapat beberapa orang yang memang dikhususkan untuk melakukan validasi. Berikut adalah contoh tampilan proyek yang menggunakan tasking manager.



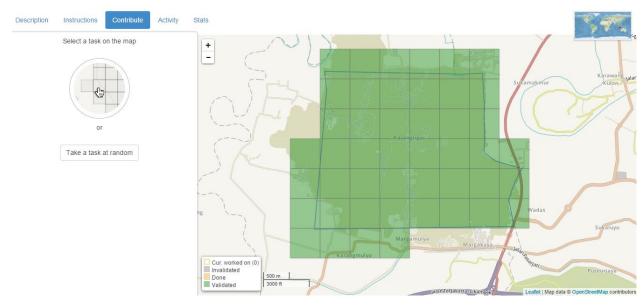
Pada gambar di atas dapat kita lihat bahwa dalam tasking terdapat kotak-kotak yang membagi wilayah dijitasi untuk tiap-tiap pengguna. Kotak-kotak tersebut memiliki warna berbeda-beda dimana untuk kotak yang semua objeknya telah dipetakan akan diberikan warna **kuning**, yang belum dipetakan akan diberikan warna **abu-abu** dan yang telah divalidasi akan diberikan warna **hijau**.

Secara teknis, suatu proyek pemetaan yang menggunakan tasking manager sebaiknya tidak hanya selesai dipetakan saja tetapi harus selesai divalidasi semua yang artinya kotak-kotak yang ada di tasking tersebut semuanya sebaiknya berwarna hijau. Hal ini dimaksudkan agar kita tidak hanya menambahkan objek saja ke dalam OpenStreetMap tetapi memastikan bahwa data-data yang dimasukkan memiliki kualitas yang baik.

Untuk melakukan validasi di *tasking manager* kita dapat mengklik kotak yang berwarna **kuning** yaitu kotak yang telah dipetakan semua objeknya kemudian klik **Review the work**, kemudian akan muncul pilihan **Edit with** dan silahkan pilih editor yang ingin anda gunakan untuk melakukan validasi, biasanya kita kembali akan menggunakan **JOSM**.



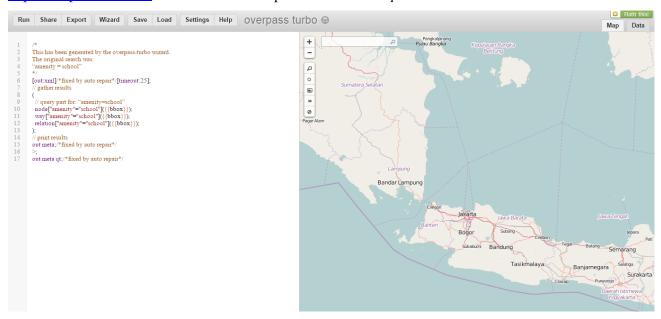
Setelah itu data-data yang ada dalam kotak tersebut akan di*download* secara otomatis ke dalam JOSM dan lakukan validasi seperti yang telah dijelaskan pada subbab sebelumnya. Setelah anda telah selesai melakukan validasi terhadap semua data di kotak tersebut, silahkan klik tulisan **Validate** sehingga kotak tersebut akan berubah menjadi warna hijau. Jika anda merasa datanya masih terlalu banyak salah dan belum semua yang dipetakan maka anda dapat memilih pilihan invalidate untuk kotak tersebut. Anda dapat memilih kotak selanjutnya untuk kembali melakukan validasi.



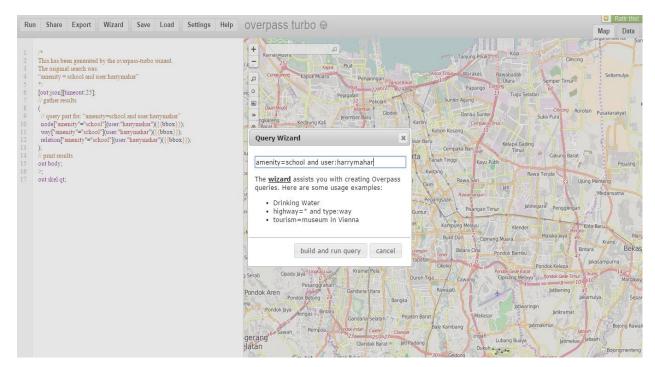
Validasi menggunakan tasking manager ini merupakan cara validasi yang dapat digunakan secara bersama-sama dalam suatu area dan validasi yang dapat dilakukan secara cepat sehingga data-data yang ada di area tertentu dapat diperbaiki dan memiliki kualitas data yang baik di OpenStreetMap.

3.3 Validasi Data Menggunakan Overpass Turbo

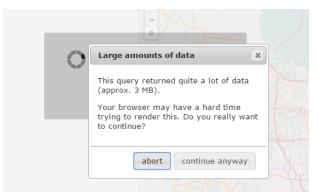
Jika pada subbab sebelumnya kita sudah mempelajari bagaimana melakukan validasi menggunakan JOSM untuk kegiatan individu dan validasi menggunakan tasking manager untuk melakukan validasi secara bersama-sama. Sekarang, kita akan mempelajari bagaimana melakukan validasi terhadap objek-objek tertentu yang ada di OpenStreetMap. Dengan begitu kita tidak perlu melakukan pengecekan terhadap semua objek di area tertentu tetapi bisa memilih objek yang ingin divalidasi. Sebagai contoh, kita ingin melakukan validasi untuk semua objek sekolah di wilayah Jakarta. Kita tidak perlu mendownload seluruh data di Jakarta dan hanya mengambil objek sekolah di Jakarta. Hal ini sangat bermanfaat karena kita dapat menghemat download data-data di OpenStreetMap dan menghemat waktu untuk melakukan validasi. Alat yang akan kita gunakan untuk bisa melakukan validasi terhadap objek-objek tertentu adalah *Overpass Turbo*. Silahkan kunjungi tautan http://overpass-turbo.eu/. Berikut adalah tampilan awal dari overpass turbo:



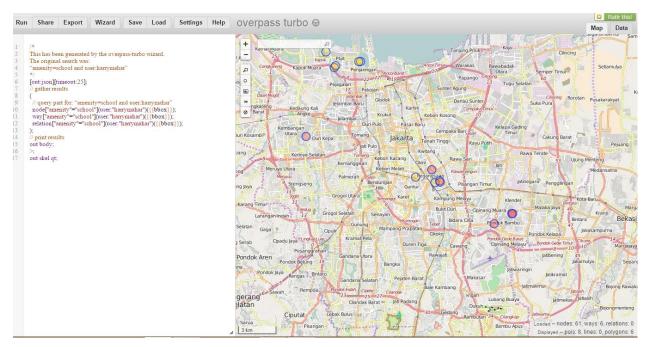
Kita dapat mencari area yang ingin kita validasi dengan mengklik kotak pencarian di sebelah kiri atas dari tampilan peta. Selain itu kita bisa memperbesar dan memperkecil sama seperti di tautan peta OpenStreetMap. Setelah mendapatkan area yang kita ingin validasi, selanjutnya adalah memfilter objek yang ingin kita validasi. Filter yang kita lakukan adalah dengan menggunakan *tag* yang terdapat pada objek tersebut. Misalnya kita ingin mencari sekolah maka kita menggunakan *tag* amenity = school dan/atau jika kita ingin mencari hasil data-data yang dimasukkan oleh pengguna OpenStreetMap tertentu kita dapat menggunakan *tag* user: (nama pengguna). Untuk melakukan filter ini silahkan klik menu Wizard kemudian ketikkan *tag* yang kita inginkan. Ketika selesai klik build and run query



Nanti mungkin akan muncul notifikasi yang mengatakan bahwa kemungkinan objek yang dipilih ukurannya lebih besar dari biasanya. Jika anda menemukan notifikasi seperti ini silahkan pilih continue anyway.



Setelah itu silahkan tunggu beberapa saat dan kemudian untuk objek **sekolah** hasilnya akan muncul kurang lebih seperti berikut:



Kalau kita lihat di sebelah kanan bawah akan muncul jumlah objek yang diambil berdasarkan *tag* yang telah kita gunakan untuk memfilter yaitu **bangunan sekolah yang diedit ataupun dimasukkan oleh pengguna OSM yang bernama harrymahar**. Kurang lebih tampilannya seperti berikut:

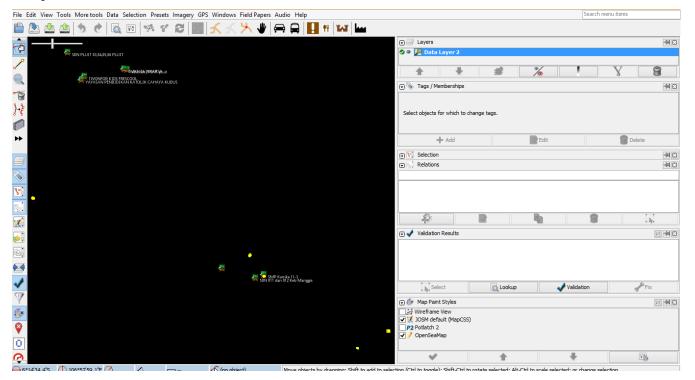
```
Loaded – nodes: 61, ways: 6, relations: 0
Displayed – pois: 8, lines: 0, polygons: 6
```

Langkah selanjutnya, kita akan men-*donwload* objek-objek yang sudah kita filter tadi dan memasukkannya ke dalam JOSM untuk divalidasi. Untuk melakukan itu silahkan klik menu **Export** kemudian pilih *load data into OSM Editor JOSM*. Perlu diingat bahwa sebelum menentukan pilihan ini, JOSM anda harus sudah dibuka terlebih dahulu.



Jika anda sudah melakukan itu, maka data-data yang diambil akan otomatis masuk ke dalam JOSM

dan setelah itu buka JOSM anda dan silahkan lakukan validasi jika diantara objek-objek tersebut terdapat kesalahan.



Selamat! Anda telah berhasil mempelajari cara validasi menggunakan *overpass turbo*. Anda dapat melakukan validasi dengan memfilter objek-objek tertentu dengan menggunakan informasi yang ada pada objek tersebut. Cara validasi ini sangat mudah dan cepat. Kita sekaligus bisa memantau data di area kita yang dimasukkan oleh para pengguna yang tergabung dalam proyek tertentu. Setelah mengetahui tiga cara validasi di OpenStreetMap, kita bisa menentukan teknik validasi mana yang paling sesuai dengan kebutuhan dan kondisi kegiatan pemetaan kita. Selamat memetakan!