**INSTITUT TEKNOLOGI BANDUNG**

**PROGRAM STUDI TEKNIK ELEKTRO**

Jalan Ganesha No. 10 Gedung Labtek V Lantai 2 🕿 (022)2508135-36, 🖷 (022)2500940

Bandung 40132

**Dokumentasi Produk Tugas Akhir**

Lembar Sampul Dokumen

|  |  |  |
| --- | --- | --- |
| Judul Dokumen | TUGAS AKHIR TEKNIK ELEKTRO:  Sistem Detektor Gempa dan Tsunami Decision Support System | |
|  |  | |
| Jenis Dokumen | PERANCANGAN | |
|  | Catatan: Dokumen ini dikendalikan penyebarannya oleh Prodi Teknik Elektro ITB | |
| Nomor Dokumen | B300-01-TA1617.01.069 | |
|  |  | |
| Nomor Revisi | 01 | |
|  |  | |
| Nama File |  | |
|  |  | |
| Tanggal Penerbitan | 2 May 2017 | |
|  |  | |
| Unit Penerbit | Prodi Teknik Elektro – ITB | |
|  |  | |
| Jumlah Halaman | 35 | (termasuk lembar sampul ini) |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Data Pemeriksaan dan Persetujuan | | | | |
| Ditulis | Nama | **Christoporus Deo Putratama** | Jabatan | Anggota |
| Oleh | Tanggal | Desember 2016 | Tanda Tangan |  |
|  | Nama | **Kevin Shidqi** | Jabatan | Anggota |
|  | Tanggal | Desember 2016 | Tanda Tangan |  |
|  | Nama | **Bramantio Yuwono** | Jabatan | Anggota |
|  | Tanggal | Desember 2016 | Tanda Tangan |  |
|  |  |  |  |  |
| Diperiksa | Nama | **Dr. techn. Ary Setijadi**  **Prihatmanto** | Jabatan | Dosen Pembimbing |
| Oleh | Tanggal | Desember 2016 | Tanda Tangan |  |
|  |  |  |  |  |
| Disetujui | Nama | **Dr. techn. Ary Setijadi**  **Prihatmanto** | Jabatan | Dosen Pembimbing |
| Oleh | Tanggal | Desember 2016 | Tanda Tangan |  |
|  |  |  |  |  |

DAFTAR ISI

[DAFTAR ISI 3](#_Toc462040353)

[Catatan Sejarah Perbaikan Dokumen 4](#_Toc462040354)

[Proposal Proyek Pengembangan SISTEM DETEKTOR GEMPA DAN TSUNAMI DECISION SUPPORT SYSTEM 5](#_Toc462040355)

[1 Pengantar 5](#_Toc462040356)

[1.1 RINGKASAN ISI DOKUMEN 5](#_Toc462040357)

[1.2 Tujuan Penulisan dan Aplikasi/Kegunaan Dokumen 5](#_Toc462040358)

[1.3 REFERENSI 5](#_Toc462040359)

[1.4 DAFTAR SINGKATAN 6](#_Toc462040360)

[2 pERANCANGAN 6](#_Toc462040361)

[2.1 Desain Sistem 6](#_Toc462040362)

[2.2 Desain Komunikasi 7](#_Toc462040363)

[2.3 Desain Hardware 8](#_Toc462040364)

[2.4 Desain Software 12](#_Toc462040365)

[2.5 Desain Pemodelan 15](#_Toc462040366)

[3 Lampiran 17](#_Toc462040367)

Catatan Sejarah Perbaikan Dokumen

|  |  |
| --- | --- |
| Versi, Tgl, Oleh | Perbaikan |
|  |  |
|  |  |
|  |  |
|  |  |

Proposal Proyek Pengembangan Rancang Bangun *Flapping Wing Microaerial Vehicle* untuk Misi Monitoring dengan Kamera

# Pengantar

## RINGKASAN ISI DOKUMEN

Dokumen ini merupakan kelanjutan dari dokumen B200. Dokumen ini berisi penjabaran perancangan dari proyek “Sistem Detektor Gempa dan Tsunami Decision Support System”. Kemudian dokumen ini terdiri dari 2 bab yaitu, pengantar dan perancangan.

Pada bab pengantar, terdiri dari ringkasan dari dokumen ini, tujuan penulisan dan kegunaan dokumen, referensi, dan daftar singkatan. Sedangkan pada bab perancangan berisi tentang desain sistem, desain komunikasi, desain hardware, desain software, desain pemodelan, dan desain basis data

## Tujuan Penulisan dan Aplikasi/Kegunaan Dokumen

Tujuan penulisan dokumen ini adalah sebagai berikut :

* Sebagai penjabaran perancangan dalam proyek “Sistem Detektor Gempa dan Tsunami Decision Support System”.
* Sebagai dokumentasi lanjutan pembuatan proyek “Sistem Detektor Gempa dan Tsunami Decision Support System”.

Dokumen ini ditujukan kepada dosen pembimbing tugas akhir dan tim tugas akhir Program Studi Teknik Elektro ITB sebagai bahan penilaian tugas akhir.

## REFERENSI

[1] Croon, G.C.H.E de. The DelFly :*Design, Aerodynamics, and Artificial Intelligence of a Flapping Wing Robot*. Springer: (2016)

[2] DiLeo, Christopher dan Xinyan Deng. *Design and Experiments of a Dragonfly-Inspired Robot*, 2009. <https://engineering.purdue.edu/-xdeng/AR09.pdf> ,6 September 2016, 18.58 WIB.

[3] Diardano R., Oki N. dan Dwicky F.S., Rancang Bangun *Flapping Wings Microaerial Vehicle* : Sistem Kendali, Sensor dan Telemetri, Proposal Tugas Akhir, Institut Teknologi Bandung, 2016.

## DAFTAR SINGKATAN

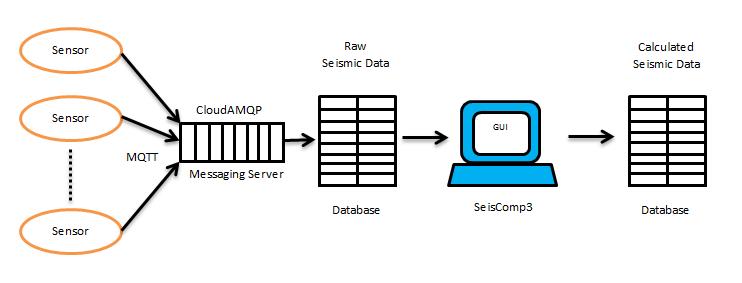
| Singkatan | Arti |
| --- | --- |
| MAV | *Micro Aerial Vehicle* |
| GCS | *Ground Control Station* |
| GPS | *Global Positioning System* |
| IMU | *Inertial Measurement Unit* |
| UAV | *Unmanned Aerial Vehicle* |
| CPU | *Central Processing Unit* |
| GUI | *Graphical User Interface* |
| 3D | 3 Dimensi |

# PERANCANGAN

## Desain Sistem

*Earthquake Cathcer Network* adalah sistem pendeteksi gempa yang mengintegrasikan 3 sumber data berupa gelombang seismik, memroses data tersebut sehingga dihasilkan suatu keluaran yang berupa keputusan harus mengeluarkan suatu peringatan tsunami atau tidak. Sistem ini bekerja dengan cara mendeteksi suatu kejadian gempa, menentukan lokasi dan kedalaman pusat gempa, menentukan magnituda dari kejadian gempa, lalu dengan sebuah model, memperkirakan bahaya tsunami yang mungkin ditimbulkan gempa tersebut. Selain itu, sistem juga memprediksi inundasi tsunami, atau seberapa jauh tsunami akan mencapai daratan.

Desain arsitektur dari sistem jaringan detektor gempa decision support system yang dirancang dapat dilihat pada gambar berikut ini.



RabbitMQ

Dapat dilihat dari beberapa sensor yang telah diletakkan, sensor tersebut akan mengirimkan beberapa data seperti sampling data percepatan tanah yang dibaca oleh sensor, lokasi sensor, dan timestamp ke messaging server (RabbitMQ) melalui protokol MQTT. Setelah itu GUI Server akan mengambil data mentah yang berada pada messaging server tersebut untuk disimpan pada database. Setelah data mentah tersebut tersimpan pada database, GUI Processing akan mengambil data tersebut dalam suatu interval waktu tertentu untuk diproses dan ditentukan apakah terjadi gempa dengan pembacaan beberapa sensor tersebut menggunakan aplikasi SeisComp3. Setelah proses dilakukan, data hasil pemrosesan tersebut akan disimpan. Jika data hasil pemrosesan menunjukkan bahwa terjadi gempa maka GUI Processing tersebut akan memberikan peringatan pada GUI tersebut dan akan memberikan peringatan juga melalui twitter. Dalam membuat desain sistem yang akan dirancang, perlu diketahui cakupan dari sistem melalui skema sistem yang akan dijelaskan pada gambar di bawah ini.

Penjelasan diagram blok:

Seismometer BMKG digunakan untuk *monitoring* gelombang seismik di seluruh Indonesia, dan disediakan aksesnya oleh BMKG. Seismometer ini merupakan metode yang paling konvensional dalam pengawasan gempa, dan telah digunakan dalam sistem yang telah berjalan sekarang ini yaitu InaTEWS. Seismometer ini merupakan salah satu sensor yang digunakan dalam sistem ini untuk mendeteksi gempa dan memprediksi tsunami. Seismometer juga bisa memberi data lokasi melalui GPS, sehingga bisa diprediksi bahaya tsunami atau tidak berdasarkan lokasi dan kedalaman gempa. Data tersebut dikirim melalui sebuah API *messaging,* melalui sebuah messaging server. Data tersebut akan dikirim ke server yang akan berfungsi sebagai pusat data dan tempat analisis data.

USGS (*United States Geological Survey)* adalah lembaga geologi Amerika Serikat yang menyediakan data *seismic waveform* melalui websitenya yang tersedia secara gratis. Data tersebut adalah data gempa yang diupdate secara *real-time* dan tersedia di situs <http://earthquake.usgs.gov/earthquakes/map>. Data ini digunakan sebagai sumber tambahan dalam deteksi gempa dan prediksi tsunami. Rincian data tersebut adalah sebagai berikut:

1. Kejadian gempa
2. Magnituda
3. Waktu dan tempat
4. Kedalaman gempa

Dari detail tersebut, bisa dibuat prediksi bahaya tsunami berdasarkan lokasi dan kedalaman gempa. Adanya data dari USGS yang mencakup seluruh dunia membantu kita dalam mendeteksi gempa yang agak jauh dari Indonesia sehingga tidak terdeteksi seismometer lokal, namun tetap bisa berpotensi tsunami yang membahayakan Indonesia.

QCN (Quake Catcher Network) adalah software *open-source* yang dikembangkan oleh Stanford University sebagai sistem detektor gempa berbasis sensor yang murah dan dapat dibuat oleh orang masyarakat umum. QCN diharapkan dapat menjadi sistem pendeteksi gelombang kuat terbesar di dunia yang murah dengan memanfaatkan sensor yang dihubungkan dengan komputer melalui USB, dan terhubung ke server QCN di California melalui internet. Jaringan QCN ini dapat memberitahukan suatu kejadian gempa melalui informasi yang didapat dari sensor-sensor yang dipasang ke jaringan tersebut.

Pada sistem ini, sensor yang akan dihubungkan ke QCN akan dibuat sendiri. Sensor tersebut berisi MEMS (Mini Electro-Mechanical Sensor), yaitu IMU (Inertial Measurement Unit) dan sensor piezoelectric. Sensor tersebut digunakan untuk mengukur getaran gempa dan orientasi dari getaran tersebut. Dengan cara ini dapat dibedakan antara gelombang primer dan sekunder dari gempa, sehingga kita bisa mengetahui kedalaman gempa tersebut. Melalui kerjasama banyak sensor di daerah-daerah tertentu, dapat juga ditentukan episentrum dari gempa melalui metode triangulasi, sehingga seperti data sebelumnya dapat ditentukan potensi tsunami dari gempa tersebut.

Dari gabungan ketiga data tersebut, akan dilakukan suatu proses analisis untuk menentukan bahaya atau tidaknya gempa tersebut, dan potensi tsunami. Analisis tersebut menggunakan model yang sudah dikembangkan berdasarkan kejadian gempa dan tsunami di masa lalu. Beberapa model yang akan kami gunakan adalah Tunami yang dikembangkan Tohoku University, Tunawi yang dikembangkan oleh Jerman, dan ComCod yang dikembangkan oleh USGS. Dengan adanya tiga data sumber, prediksi dapat dilakukan dengan lebih akurat. Melalui model ini, diharapkan bahwa analisis dapat dilakukan secepat mungkin, sehingga peringatan dini akan adanya tsunami dapat diberikan dalam waktu 5 menit dari sejak kejadian gempa.

Prediksi akan adanya tsunami juga termasuk prediksi inundasi, yaitu prediksi seberapa jauh tsunami akan mencapai daratan. Hal tersebut dapat diprediksi dari kenyataan bahwa tinggi dari tsunami bergantung pada panjang gelombang serta kedalaman laut. Dengan menggunakan *Bathymetry,* yaitu pengukuran kedalaman dan topografi laut, dapat diprediksi tinggi tsunami yang akan terjadi. Bila ditambah dengan pengetahuan tentang topografi daratan, kita bisa memprediksi inundasi dari tsunami tersebut. Hasil dari proses analisis dan prediksi ini sangat penting karena akan menentukan bagaimana mitigasi bencana dilakukan.

Sistem diseminasi dan evakuasi digabung menjadi sistem mitigasi bencana, yang tidak terbatas pada gempa dan tsunami saja. Diseminasi adalah proses yang identik dengan alarm jika sudah terdeteksi gempa yang berpotensi tsunami. Diseminasi dapat dilakukan dengan memanfaatkan berbagai media seperti televisi, radio, HP, *speaker,* masjid, dan sebagainya. Proses diseminasi ini juga akan diintegrasikan dengan aplikasi Semut, yaitu Smart City & Intelligent Transportation System, sebuah app berbasis android yang sedang dikembangkan oleh LSKK ITB. Melalui app tersebut, pengguna bisa mendapatkan informasi mengenai peringatan dini gempa dan tsunami, jadi pengguna bisa mengetahui apakah dia perlu melakukan evakuasi dan jika ya, seberapa jauh. Selain itu, pengguna bisa menerima semacam *travel warning* jika ada daerah yang baru saja kena gempa, atau dalam risiko terkena gempa susulan.

Sistem Jaringan Detektor Gempa dan Tsunami yang dibuat memiliki beberapa fitur, yaitu sebagai berikut.

* Pembacaan Data Seismik secara Akurat

Agar proses diseminasi dapat berjalan efektif dan efisien, pembacaan sensor harus akurat. Pembacaan sensor harus memiliki error pembacaan yang kecil sehingga error tersebut tidak mengganggu pengolahan dan penggabungan data pada data center. Pengujian fungsi ini dilakukan dengan menggunakan gempa buatan yang membuat pembacaan sensor mengindikasikan adanya getaran dan perubahan kecepatan sudut. Dengan referensi data yang diperoleh dari gempa buatan tersebut, data yang diperoleh saat testing akan dibandingkan dengan data referensi sehingga fungsi ini dapat diverifikasi peformanya.

* Pengiriman Data Seismik secara Real-Time dengan Komunikasi Nirkabel

Agar sistem ini dapat memberi peringatan ketika gempa dan tsunami terjadi, diperlukan adanya mekanisme pengiriman data seismik secara Real-Time dengan menggunakan Komunikasi Nirkabel. Untuk memverifikasi fungsi ini, testing dapat dilakukan dengan cara menghitung waktu update data seismik yang diperoleh dari Seismometer BMKG, Website USGS, dan Quake Catcher Network yang diintegrasikan dengan sensor yang dibuat pada data center. Kebutuhan peformasi dari fungsi ini terpenuhi jika waktu yang dibutuhkan untuk mengirim data kurang dari jumlah waktu yang dibutuhkan untuk melakukan analisis & penggabungan data dan golden time (5 menit).

* Safe Mode dan Backup Power

Ketika gempa dan tsunami terjadi, sangat besar kemungkinan daya listrik dari jala-jala akan mati, sehingga sensor yang dibuat harus memiliki sumber daya cadangan dan Safe Mode agar sensor dapat menggunakan baterai cadangan. Fungsi ini berjalan baik jika ketika listrik jala-jala diputus, sensor akan secara automatis menggunakan baterai cadangan sebagai pensuplai daya ke sensor.

Sistem ini dapat didemonstrasikan dengan cara menunjukkan data yang dibaca dari ketiga sumber yang telah disebutkan secara bersamaan. Lalu ketika dibuat gempa buatan untuk sensor, data yang diterima pada data center haruslah mengindikasikan bahwa gempa akan terjadi.

## Desain Komunikasi

### **Protokol Pengiriman Data dalam Jaringan**

Pada jaringan yang terdapat dalam Sistem Jaringan Detektor Gempa dan Tsunami Decision Support System, standar data yang digunakan dalam pengiriman secara aktual adalah dalam bentuk sebuah pesan singkat yang memiliki isi yang padat dan jelas.Sebagai contoh, sebuah protokol pesan singkat untuk memberitakan terjadinya sebuah gempa dan/atau tsunami adalah sebagai berikut:

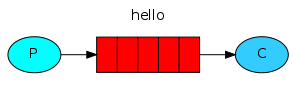
|  |
| --- |
| Tsunami Warning in BENGKULU, Eq Mag:7.0RS, 09-Dec-09 15:52:59 UTC, Loc:4.64S/101.11E,Dep:10km::BMKG |

Dapat dilihat bahwa dalam mengirimkan pesan singkat maka diperlukan susunan aturan yang dapat dimengerti oleh sistem. Pada proyek ini, protokol komunikasi yang digunakan adalah berbasis Advanced Messaging Queueing Protocol (AMQP). Protokol ini dipilih karena mempunyai beberapa fitur yang sesuai dengan sistem yang digunakan. AMQP diterapkan oleh sebuah vendor komunikasi RabbitMQ sehingga akan digunakan RabbitMQ dengan metode AMQP sebagai protokol komunikasi sistem. RabbitMQ sendiri merupakan broker pesan, dimana sebuah pesan yang dikirimkan dari pengirim menuju tujuan akan difasilitasi oleh RabbitMQ. Protokol pesan dapat dimodifikasi sehingga:

* Pengiriman pesan dapat diurutkan dalam sebuah antrian baris;
* Melakukan penerbitan pesan ke seluruh bagian sistem jaringan;
* Melakukan langganan pesan oleh pengguna pada broker;
* Melakukan penghubungan dan penyaringan pesan;
* Seleksi pesan yang akan diperoleh;
* Kendali jarak jauh pada jaringan dengan metode *remote.*

Fitur-fitur diatas dapat menguntungkan karena tidak semua data yang diperoleh merupakan informasi yang penting dan besar kemungkinan pada sistem bawa sebuah pesan dapat terkena distorsi akibat *bug* pada sistem. Selain itu mudah untuk mengendalikan sebuah sistem jaringan yang besar dengan protokol pesan AMQP ini.

Diagram blok dari sistem protokol komunikasi RabbitMQ adalah sebagai berikut:

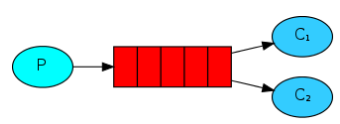


Queue

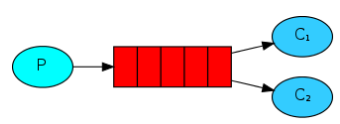
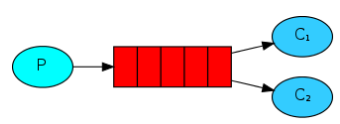
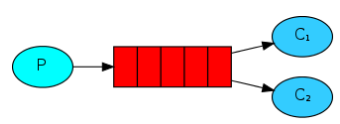
Consuming

Producing

Gambar diagram blok diatas dari sistem paling sederhana dari RabbitMQ, dimana ada sebuah komponen sistem yang menghasilkan sebuah pesan, kemudian dimasukkan ke dalam antrian dan selanjutnya dikirimkan kepada konsumen. Selanjutnya akan disertakan diagram blok dari fitur RabbitMQ berupa penerbitan pesan pada beberapa konsumen.



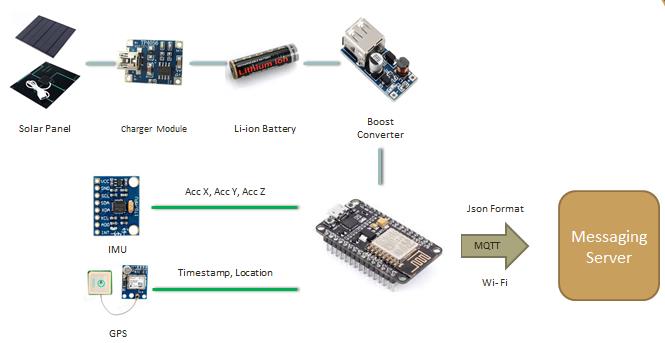
Pada Sistem Jaringan Detektor Gempa dan Tsunami Decision Support System, p



## Desain Hardware

### **Sensor**

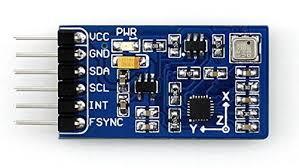
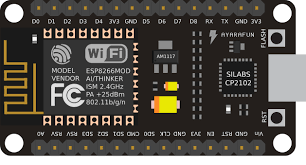
Pada sistem elektrikal sensor sendiri dibagi menjadi dua bagian, yaitu sistem daya dan sistem mikrokontroller. Untuk desain sistem elektrikal dapat dilihat pada diagram berikut ini.



Untuk menyuplai sensor yang harus bekerja selama 24 jam terus-menerus, digunakan solar panel yang akan menyuplai sensor pada siang hari. Karena solar panel tidak dapat menyuplai daya pada siang hari maka diperlukan rangkaian charging baterai, agar ketika malam sistem daya ini masih dapat menyuplai sistem mikrokontroler. Sehingga digunakan modul charge controller baterai Li-ion yang berbasis chip TP4056. Chip TP4056 ini dapat mengatur tingkat charging baterai Li-ion karena untuk mengisi baterai Li-ion dibutuhkan tegangan konstan pada 4.2 Volt dan arus yang berubah sesuai dengan kapasitas baterai yang sudah terisi. Baterai Li-ion dipilih karena sifatnya yang tidak mudah rusak jika dilakukan proses charge-discharge dibandingkan dengan baterai lain.

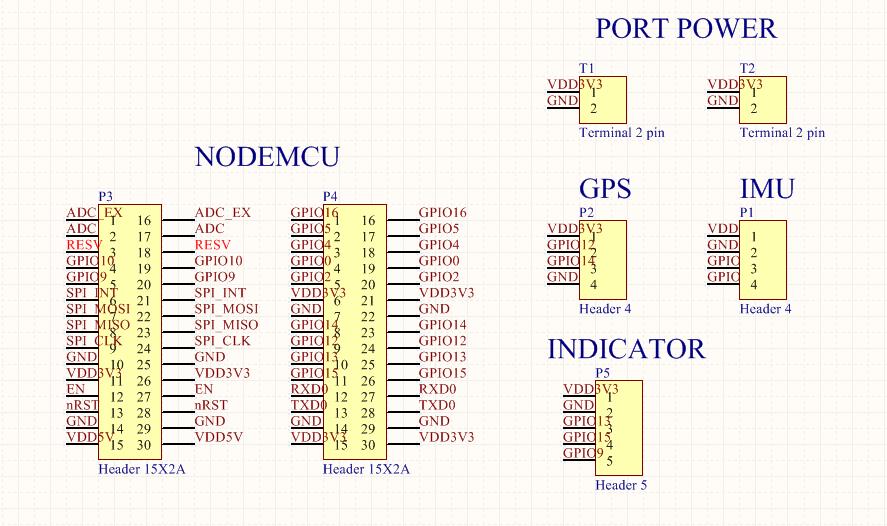
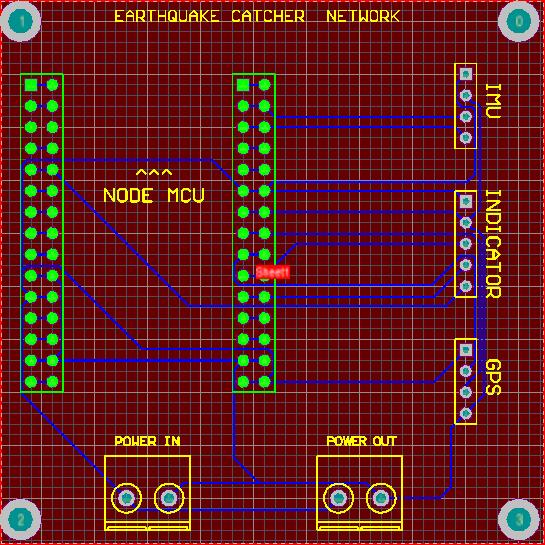
Untuk menentukan berapa daya solar panel dan kapasitas baterai yang dibutuhkan, dilakukan beberapa asumsi dalam perhitungan. Asumsi pertama adalah solar panel tidak akan bisa menyuplai daya selama 12 jam sehari yaitu dari jam 6 pagi sampai jam 6 malam. Asumsi kedua adalah solar panel hanya dapat menyuplai sensor selama 7 jam jika cuaca cerah. Dengan diketahui tegangan kerja sensor, daya maksimum yang dibutuhkan NodeMCU maka dapat diperoleh.

Diperoleh bahwa dibutuhkan baterai dengan kapasitas 12000 mAh dan solar panel yang memiliki daya sebesar 13.5 W. Perhatikan bahwa perhitungan ini tidak memperhatikan fakta bahwa solar panel tidak akan menyuplai daya sebesar 13.5 W secara konstan dan loss daya pada rangkaian charge controller. Sehingga hitungan ini hanya sebagai dasar penentuan sistem daya yang tepat untuk sensor dan diperlukan percobaan lagi untuk mengetahui berapa daya solar panel yang dibutuhkan dan kapasitas baterai yang dibutuhkan.

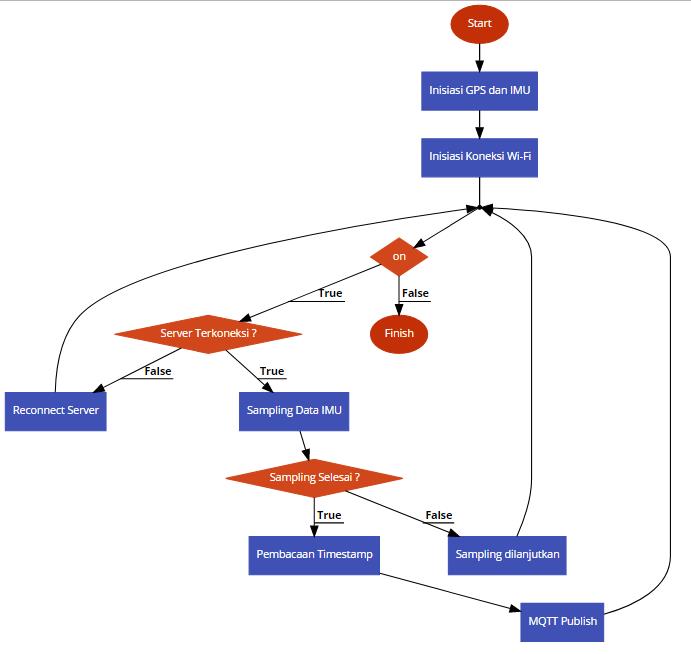


Pada sistem mikrokontroller, digunakan NodeMCU sebagai mikrokontroller dengan alasan harga yang murah dan sudah terdapat modul Wi-Fi pada NodeMCU. Untuk GPS digunakan modul GPS ublox Neo-6M dan untuk IMU digunakan MPU9255. Dipilih MPU9255 dibandingkan dengan IMU jenis lain berdasarkan pada jumlah bit register pembacaan percepatan sebanyak 16 bit, lebih banyak daripada yang lain sehingga pembacaan menjadi lebih akurat karena resolusi pembacaan akan semakin berkurang dengan digunakannya lebih banyak bit dalam merepresentasikan pembacaan percepatan.

Rangkaian dan desain PCB pada sistem mikrokontroler dapat dilihat pada diagram dibawah ini

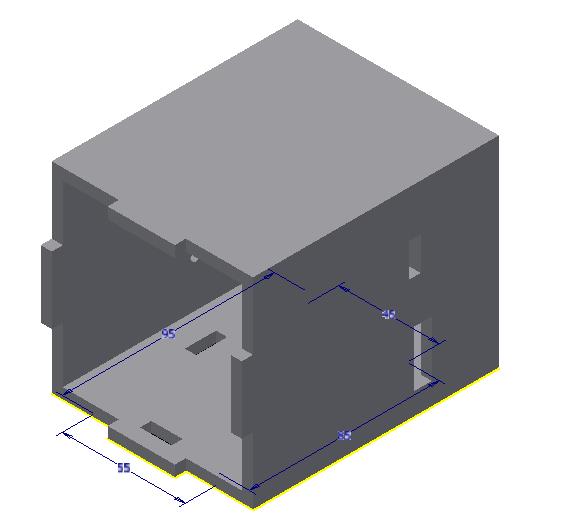
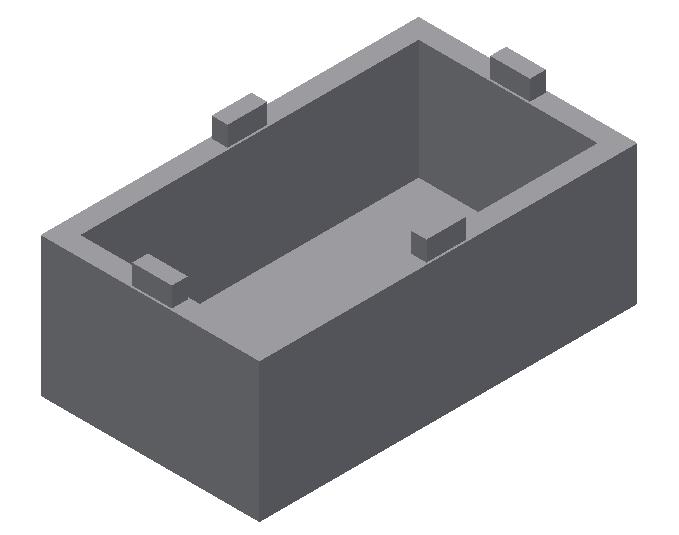


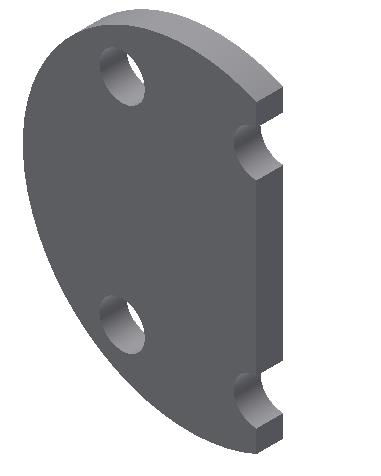
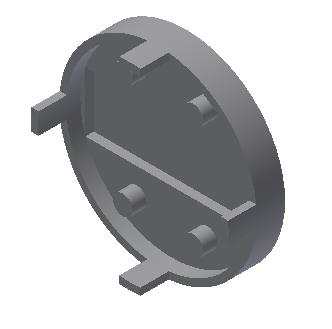
Dapat dilihat bahwa koneksi dari mikrokontroller NodeMCU ke GPS, IMU, dan modul Indikator digunakan header untuk memudah pemasangan dan instalasi. Ukuran board PCB yang didesain adalah 6.5 cm x 6.5 cm menyesuaikan dengan casing yang didesain. Untuk koneksi PCB ini ke sistem daya digunakan terminal dengan dua port sehingga lebih mudah dihubungkan antar sistem.

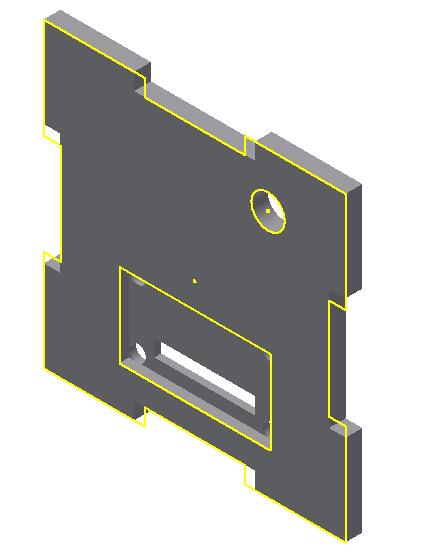
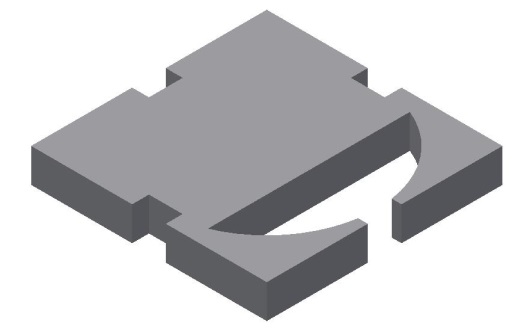


Sedangkan flowchart algoritma yang akan digunakan pada sensor dapat dilihat pada gambar diatas. Pada awal mula program dilakukan inisiasi untuk membuat modul GPS, IMU, dan koneksi Wi-Fi dapat digunakan. Pada kondisi menyala sensor akan melakukan pemeriksaan apakah sensor terhubung ke server, jika tidak sensor akan melakukan percobaan koneksi ke server ulang sampai sensor terkoneksi dengan server. Ketika sudah terkoneksi dengan server, sensor akan melakukan sampling data percepatan dengan periode 25 ms. Ketika telah diperoleh jumlah sampling yang diinginkan maka akan dilakukan proses publish message ke messaging server dengan melalui protokol MQTT. Tetapi sebelum dilakukan publish, dilakukan terlebih dahulu pembacaan Timestamp dari GPS. Setelah proses publish selesai dilakukan, sensor akan mengulangi proses yang sudah dijelaskan dari awal kembali.

Untuk menyatukan sistem daya dan sistem mikrokontroler yang sudah disebutkan tadi, diperlukan sebuah casing yang membuat sensor tidak mudah rusak atau terganggu dengan lingkungan luar. Berikut adalah desain 3D casing dengan menggunakan aplikasi Autodesk Inventor.







Desain pada gambar pertama berfungsi untuk tempat meletakkan baterai sehingga baterai mudah diganti dan dapat ditempel pada casing utama pada gambar kedua yang berfungsi sebagai tempat meletakkan PCB dan sistem daya. Pada casing utama tersebut, terdapat lubang untuk meletakkan casing baterai yang sudah dijelaskan sebelumnya dan lubang sebagai tempat meletakkan GPS dan sebagai tempat kabel IMU yang akan digunakan pada casing IMU pada gambar 3,4 dan 5. Gambar 3 merupakan tempat untuk meletakkan modul MPU9255 yang dapat ditutup oleh casing pada gambar 4 dan 5. Desain 3D pada gambar 6 berfungsi untuk menutup casing utama pada gambar 2 yang sudah dijelaskan sehingga kabel untuk solar panel dapat masuk ke modul sistem mikrokontroler dan casing utama. Casing pada gambar 6 juga berfungsi untuk meletakkan modul indikator.

### **Power System**

*.*Pada sensor yang didesain, diperlukan mekanisme *self-powering* yang dilakukan oleh sensor tersebut. Pada sensor ECN ini, digunakan solar panel sebagai sumber daya sensor. Solar panel harus memiliki spesifikasi yang tepat agar sistem daya sensor ini dapat men-*charge* sensor pada siang hari dan disimpan ke baterai sehingga ketika malam hari, baterai memiliki kapasitas yang cukup untuk mensuplai daya ke sensor pada malam hari. Oleh karena itu diperlukan suatu sistem *charge controller* yang akan mengatur proses *charging* yang dilakukan solar panel ke baterai. Diagram *power system* dapat dilihat pada gambar berikut.

Sensor

Charge Controller

Solar Panel

Battery

**Gambar Diagram Blok Power System**

Fungsi *charge controller* adalah untuk mengatur tingkat *charging* yang dilakukan oleh solar panel ke baterai agar tidak terjadi *overcharge* pada baterai yang dapat menyebabkan baterai dapat mengalami kerusakan. Agar *power system* dapat menyuplai daya sensor selama 24 jam, solar panel dan baterai harus memiliki spesifikasi yang sesuai. Dengan daya yang diperlukan sensor sebesar 5 watt dan kemampuan solar panel yang dapat menyuplai daya maksimum selama 7 jam pada kondisi cuaca cerah, dapat diperoleh spesifikasi kapasitas dan daya solar panel dan baterai yang diperlukan.

Pada malam hari :

Pada siang hari :

Sehingga diperoleh baterai dan solar panel yang sesuai. Solar panel harus memiliki kemampuan suplai daya minimum sebesar 13.5 W dan dengan kapasitas minimum 18900 mAh. Baterai harus memiliki kapasitas minimum sebesar 12000 mAh.

Agar menghindari *overcharge* diperlukan charge controller yang dapat mengatur tingkat *charging* yang dilakukan solar panel ke baterai. Diagram blok *charge controller* dapat dilihat pada gambar berikut.

P-MOS

P-MOS

Sensor

Battery

Solar Panel

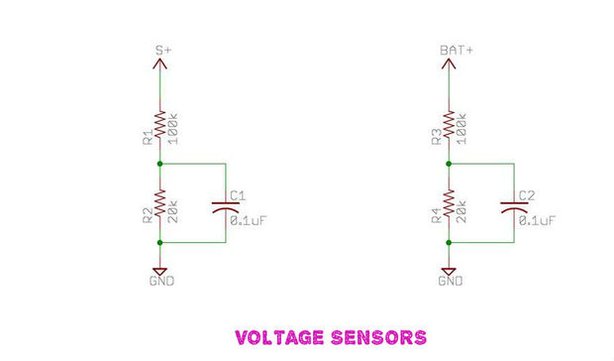
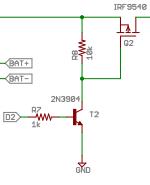
Voltage Sensor

Voltage Sensor

MCU

**Gambar Diagram Charge Controller**

Pada charge controller ini digunakan dua buah P-MOS untuk melakukan switching tegangan dari solar panel ke baterai ataupun baterai ke sensor. Selain itu, digunakan juga sensor tegangan untuk mengukur tegangan baterai dan solar panel. MCU yang digunakan adalah arduino nano. Pemilihan MCU ini berdasarkan oleh harga arduino nano yang murah serta memiliki semua fitur yang dibutuhkan dalam *charge controller* ini. Implementasi rangkaian sensor tegangan dan rangkaian switching P-MOS dapat dilihat pada gambar berikut.

****

**Gambar Implementasi Rangkaian Sensor Tegangan dan Switching P-MOS**

Dalam mengimplementasikan sensor tegangan digunakan rangkaian pembagi tegangan dan inputnya diberikan ke pin ADC dari arduino nano. Adapun perhitungan nilai ADC pada arduino sebagai berikut.

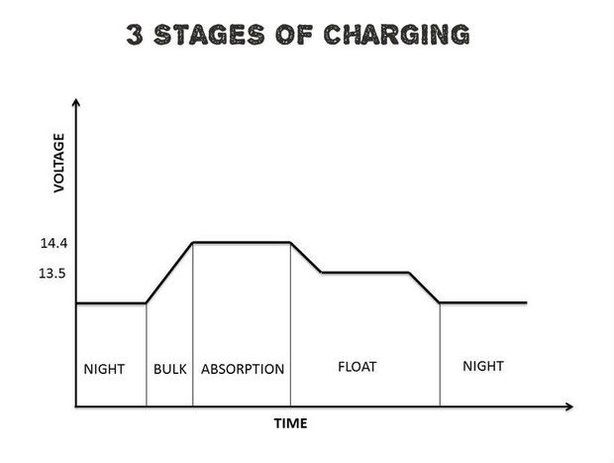
5V = ADC count 1024

1 ADC count = (5/1024)Volt= 0.0048828Volt

Vout=Vin\*R2/(R1+R2)

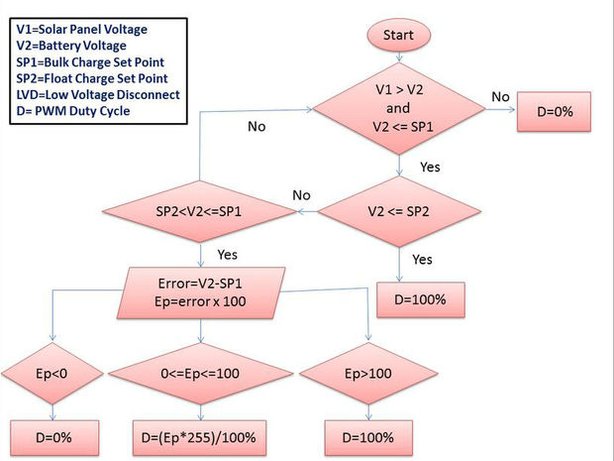
Vin = Vout\*(R1+R2)/R2 R1=100 and R2=20

Vin= ADC count\*0.00488\*(120/20) Volt



**Gambar Tahap Charging Baterai**

Pada *charge controller* ini, proses *charging* dibagi menjadi tiga tingkatan, yaitu *Bulk Charge, Absorption Charge,* dan *Float Charge.* Pada *bulk charge* arus dialirkan secara maksimum ke baterai sehingga tegangan baterai meningkat secara bertahap. Pada *Absorption Charge* tegangan baterai mencapai *bulk voltage.* Pada tahap tersebut, *charge controller* mempertahankan nilai PWM sehingga tegangan baterai konstan. Hal ini untuk mencegah baterai mengalami *over-heating.* Selanjutnya pada tahap *Float Charge,* baterai sudah dalam kondisi penuh sehingga diperlukan baterai untuk *discharge.* Dengan tahap *charging* seperti tersebut, dapat didesain software pada MCU untuk mengimplementasikan tahap *charging* tersebut*.* Flowchart software dapat dilihat pada gambar berikut.

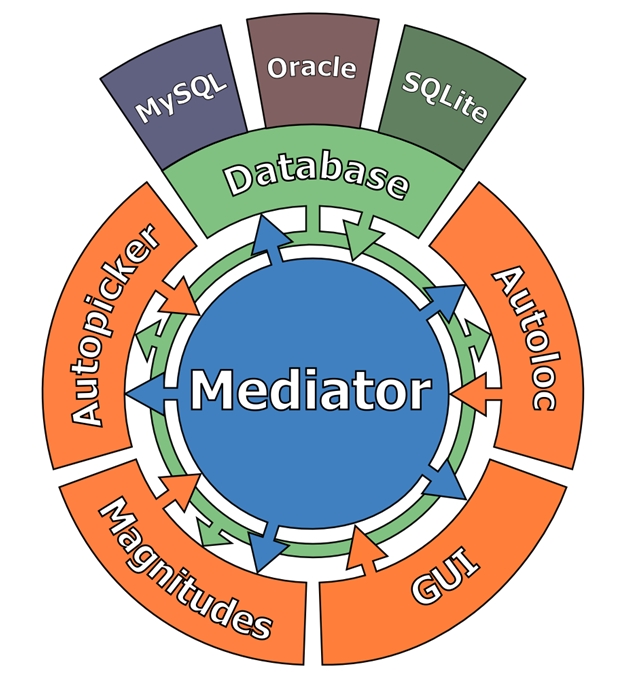


**Gambar Flowchart Charge Controller**

Hal pertama yang dilakukan adalah memeriksa apakah tegangan solar panel melebihi tegangan baterai dan tegangan baterai kurang dari tegangan *bulk-*nya. Jika tidak maka PWM akan memiliki duty cycle 0%. Jika iya, makadilakukan pemeriksaan lagi apakah tegangan solar panel kurang dari tegangan *float-*nya. Jika iya maka PWM akan menghasilkan sinyal dengan duty cycle sebesar 100%. Lalu tegangan baterai akan diperiksa apakah berada pada rentang tegangan *charging*-nya. Hasil perbedaan tegangan baterai dan tegangan *bulk* dijadikan nilai duty cycle dari nilai PWM yang dihasilkan.

## Desain Software

Sistem otomatis SeisComP3 terdiri dari satu set aplikasi independen yang masing-masing melakukan tugas diskrit. Komunikasi antara aplikasi direalisasikan oleh sistem pesan berbasis TCP / IP. Sistem pesan ini didasarkan pada toolkit open source "Spread" yang menyediakan layanan olahpesan berkinerja tinggi di jaringan area lokal dan luas. Di bagian atas "Spread" seorang mediator, disebut scmaster yang menangani persyaratan tambahan dari SeisComP3 yang tidak disediakan secara native oleh "Spread". Sistem pesan digunakan untuk pertukaran meta data (misalnya picks) dan administrasi modul program. Model data SeisComP3 didasarkan pada skema QuakeML versi 0.5. QuakeML juga digunakan sebagai database object schema. Secara default SeisComP3 menggunakan database MySQL, tapi PostgreSQL juga didukung.



Akuisisi data waveform didasarkan pada protokol SeedLink yang mapan dan protokol ArcLink baru dikembangkan di GFZ Potsdam. Aplikasi di SeisComP3 dapat dibagi dalam empat kelompok yang berbeda: akuisisi data, pemrosesan, antarmuka pengguna grafis dan utilitas. Uraian rinci tentang aplikasi ada di bab 5. Tabel 4-1 menunjukkan gambaran singkat tentang aplikasi.

|  |  |  |
| --- | --- | --- |
| **Aplikasi** | **Type** | **Description** |
| Seedlink | data acquisition | providing realtime waveform data |
| Arclink | data acquisition | providing archive waveform data |
| scmaster | processing | handling messaging |
| scqc | processing | determination of waveform quality parameter |
| scautopick | processing | automatic picking |
| scautoloc | processing | automatic event detection and localization |
| scamp | processing | amplitude calculation |
| scmag | processing | magnitude calculation |
| scevent | processing | origin association, best, magnitude selection, best origin selection |
| scrttv | graphical user interface | real-time waveform monitor |
| scmv | graphical user interface | map overview showing,actual station status and events |
| scesv | graphical user interface | summary view of most important event information |
| scolv | graphical user interface | reviewing and revising origins, manual picking tool |
| scqcv | graphical user interface | showing station quality status |
| scmm | graphical user interface | message monitoring |
| scbulletin | utility | creating bulletins of events from the database |
| scdb | utility | inserting objects from QuakeML file or messaging into the database |
| scevtlog | utility | logging the event history |
| scevtls | utility | listing events for a given time range |
| scevtstreams | utility | listing all waveform streams used for event detection |
| scimex | utility | exchange of meta data objects between SeisComP3 systems with filter functionality |
| scimport | utility | forwarding of meta data objects from one messaging system to another |
| scm | utility | performance monitor similar to UNIX top |
| scproclat | utility | logging message history |
| scvoice | utility | event alert with optional voice output |
| scxmldump | utility | event dump to QuakeML file from database |
| sczip | utility | zip implementation of SeisComP3 |

### Scevent

Mengasosiasikan asal untuk Acara atau membentuk Acara baru jika tidak ada kecocokan yang cocok ditemukan. Memilih ukuran yang disukai.  
Sebagai konsekuensi dari sistem real-time, sistem SeisComP3 menciptakan beberapa asal (hasil proses pelokalan) untuk satu gempa karena seiring berjalannya waktu lebih banyak fase seismik tersedia. Rewel menerima asal-usul ini dan mengasosiasikan asal mula kejadian. Hal ini juga memungkinkan untuk mengimpor Origins dari agensi lain.Origin Matching

Rekan-rekan yang peduli berasal dari Acara dengan mencari kecocokan terbaik dari Asal (masuk) yang baru ke Asal Lain untuk Peristiwa yang Ada. Jika pertandingan tidak ditemukan Event baru bisa terbentuk. Asal baru dicocokkan dengan Origin yang ada dengan membandingkan perbedaan lokasi (horisontal saja), beda waktu asal, dan matching Picks. Pertandingan terbaik pertama lebih disukai dimana pilihan untuk pertandingan Asal (terendah ke tertinggi).

### scautoloc

Scautoloc adalah program SeisComP3 yang bertanggung jawab untuk secara otomatis menemukan kejadian seismik dalam waktu dekat. Biasanya berjalan sebagai daemon, terus menerus membaca picks dan amplitudo dan memprosesnya secara real time. Modus offline juga tersedia. Scautoloc membaca picks otomatis dan beberapa amplitudo yang terkait. Atas dasar itu, ia mencoba untuk mengidentifikasi kombinasi picks yang sesuai dengan peristiwa seismik yang umum. Jika lokasi yang dihasilkan memenuhi kriteria konsistensi tertentu, dilaporkan, kemudian diteruskan ke program lain yang mengambil asalnya sebagai masukan.

#### Location procedure

Prosedur scautoloc untuk mengidentifikasi dan menemukan kejadian seismik pada dasarnya terdiri dari langkah-langkah berikut:

* Pilih persiapan

Pada scautoloc setiap pick masuk harus disertai dengan seperangkat amplitudo tertentu. Karena pada data SeisComP3 model amplitudo dan picks adalah objek independen, amplitudo ditambahkan sebagai atribut pada picks yang sesuai saat penerimaan oleh scautoloc.

* Pilih penyaringan

Setiap pick masuk disaring, yaitu dicentang jika pick sudah usang dan jika set lengkap amplitudo terkait sudah ada. Jika sebuah stasiun menghasilkan picks sangat sering, ini dianggap sebagai gangguan yang lebih mungkin terjadi dan menghasilkan ambang SNR yang meningkat.

* Asosiasi

Ini pertama kali mencoba mengaitkan pick masuk dengan asal-usul yang diketahui. Khusus untuk acara besar dengan lokasi yang stabil berdasarkan banyak pilihan yang sudah ada, ini adalah cara yang lebih disukai untuk menangani pick. Jika asosiasi berhasil, proses nukleasi dapat dilewati. Dalam keadaan tertentu, picks keduanya terkait dan dimasukkan ke dalam nukleator.

* Nukleasi

Jika asosiasi langsung gagal, scautoloc mencoba membuat asal baru dari ini dan pilihan lain yang tidak terkait, yang sebelumnya diterima. Proses ini disebut "nukleasi". Scautoloc melakukan pencarian grid di luar ruang dan waktu, yang merupakan prosedur yang agak mahal karena memerlukan banyak sumber daya baik dari segi CPU dan RAM. Algoritma nukleasi tambahan akan tersedia di masa depan. Grid adalah himpunan diskrit-dalam prinsip- titik-titik acak yang menggambarkan area yang diminati cukup padat. Dalam pencarian grid, masing-masing titik grid diambil sebagai hypocenter hipotetis untuk semua picks masuk. Setiap pick masuk kembali diproyeksikan tepat waktu untuk masing-masing titik grid, dengan asumsi bahwa ini adalah onset "P" pertama yang datang. Jika memilih memang sesuai dengan kedatangan event seismik "P", dan jika acara ini dicatat pada jumlah stasiun yang cukup, proyeksi baru yang diproyeksikan akan dikelompokkan dengan pick sebelumnya dari acara yang sama. Cluster akan terpadat di sekitar waktu asal di titik grid yang paling dekat dengan hypocenter. Pada prinsipnya, grid bisa jadi padat sehingga lokasi yang didapat dari pencarian grid bisa langsung digunakan. Namun, karena memori RAM dan kecepatan CPU terbatas, ini tidak mungkin dilakukan. Oleh karena itu, jika sebuah cluster diidentifikasi sebagai asal potensial, itu tidak berarti bahwa semua picks yang berkontribusi sebenarnya sesuai dengan kedatangan "P". Ini mungkin juga merupakan pertandingan kebetulan yang disebabkan oleh kekasaran grid atau kemungkinan kontaminasi oleh kebisingan yang dipetik. Oleh karena itu, program lokasi (LocSAT) dijalankan untuk mencoba lokasi dan uji coba jika rangkaian picks memang membentuk hypocenter yang konsisten. Jika memilih RMS residual terlalu besar, perbaikan dilakukan dengan mengecualikan masing-masing picks yang berkontribusi satu kali untuk menguji apakah pengurangan RMS dapat dicapai. Jika asal baru memenuhi semua persyaratan, maka diterima sebagai lokasi acara seismik baru.  
Poin grid ditentukan dalam file teks "grid.txt". File default yang dikirim dengan scautoloc mendefinisikan grid dengan titik terdistribusi global di permukaan, dan titik kedalaman terbatas pada daerah yang dikenal dengan seismisitas. Ini mungkin dimodifikasi, namun tidak boleh terdiri dari terlalu banyak titik grid (> 3000, tergantung kecepatan CPU dan RAM). Lihat di bawah untuk rincian lebih lanjut tentang file grid.

* Penyempurnaan asal

Asal usul yang dihasilkan atau diperbaharui melalui hubungan dan / atau nukleasi mungkin masih terkontaminasi oleh fase yang salah ditafsirkan sebagai pendatang "P". Scautoloc mencoba memperbaiki asal-usul ini berdasarkan mis. Pilih SNR dan amplitudo. Dalam langkah pemrosesan ini, juga mencoba mengaitkan fase yang menyelinap selama upaya asosiasi pertama, mis. Karena lokasi awalnya salah Jika asal berisi jumlah kedatangan yang cukup untuk mengasumsikan hasil lokasi yang cukup baik, scautoloc juga mencoba mengasosiasikan picks sebagai fase sekunder seperti "pP". Fase sekunder semacam itu hanya "terkait dengan lemah", yaitu fase-fase ini tidak digunakan untuk lokasi. Bagi analis, bagaimanapun, berguna untuk memiliki kemungkinan "pP" fase yang telah ditentukan sebelumnya.

* Penyaringan asal

Proses ini melibatkan pemeriksaan konsistensi akhir dari asal-usul baru / yang diperbarui, dll. Selama prosedur ini, asal-usul tidak dimodifikasi lagi.  
Dalam perjalanan nukleasi dan asosiasi, serta penyempurnaan dan penyaringan asal, kriteria heuristik tertentu diterapkan untuk membandingkan "kualitas" asal mula yang bersamaan. Kriteria ini digabungkan dalam nilai asal internal, yang didasarkan pada sifat pengambilan sendiri dalam konteks asal (residu, RMS, azimuthal gap) masing-masing. Selain itu, amplitudo memberikan cara yang berharga untuk membandingkan kualitas asal. Jelas, sebuah pick dengan SNR tinggi kemungkinannya akan menjadi semburan suara sementara daripada pick yang melebihi ambang SNR. Sebuah pick SNR tinggi sehingga meningkatkan nilai asal. Demikian pula, pick yang terkait dengan amplitudo absolut yang besar lebih cenderung sesuai dengan onset seismik yang sebenarnya, terutama dalam kasus observasi amplitudo besar simultan di stasiun tetangga. Sebuah kasus khusus muncul, ketika beberapa stasiun di dekatnya melaporkan amplitudo di atas ambang "XXL" tertentu. Untuk rinciannya lihat bagian "Asal mula awal". Amplitudo yang digunakan oleh scautoloc adalah tipe "snr" dan "mb", sesuai dengan amplitudo SNR (relatif, unit-less) dan amplitudo "mb" (absolut). Kedua amplitudo ini disediakan oleh scautopick. Dalam kasus setup di mana scautopick diganti dengan pemetik otomatis yang berbeda, kedua amplitudo ini harus diberikan pada scautoloc. Jika tidak, picks tidak digunakan. Pada saat ini adalah persyaratan yang ketat, kedepannya bisa berubah.

* File grid

File konfigurasi grid terdiri dari satu baris per titik grid, masing-masing titik grid yang ditentukan oleh 6 kolom, mis .:  
-10.00 105.00 20.0 5.0 180.0 8  
Kolom tersebut adalah koordinat titik titik (lintang, bujur, kedalaman), diameter, jarak stasiun maksimum dan jumlah tangkapan minimum. Garis di atas menetapkan titik grid yang berpusat pada 10 ° S / 105 ° BT pada kedalaman 20 km. Hal ini sensitif terhadap kejadian dalam 5 ° dari pusat. Stasiun dalam jarak hingga 180 ° dapat digunakan untuk melakukan nukleasi suatu peristiwa. Setidaknya 8 pilihan harus berkontribusi pada asal di lokasi ini. Diameternya harus dipilih cukup besar agar sel grid bisa tumpang tindih, tapi tidak terlalu besar. Ukuran juga menentukan jendela waktu untuk mengelompokkan pick dalam pencarian grid. Jika jendela waktu terlalu lama maka risiko kontaminasi dengan picks salah meningkat. Jarak stasiun maksimum memungkinkan untuk membatasi ke stasiun tertentu untuk titik grid yang sesuai. Misalnya. Stasiun dari Australia biasanya tidak diharuskan menciptakan sebuah acara di Eropa. Jika ada keraguan, atur nilainya menjadi 180. Jumlah minimum menentukan berapa jumlah pengambilan yang diperlukan untuk titik grid tertentu agar memungkinkan pembuatan asas baru. File grid default berisi grid global dengan jarak tanam ~ 5 ° dengan titik tambahan pada kedalaman yang lebih tinggi di mana peristiwa fokus dalam diketahui terjadi.

* File konfigurasi stasiun

File konfigurasi stasiun berisi baris yang terdiri dari kode jaringan, kode stasiun, flag penggunaan (0 atau 1) dan jarak nukleasi maksimum. Tanda pemakaian 1 menunjukkan stasiun harus digunakan oleh scautoloc. Jika tidak digunakan, 0 harus ditentukan di sini. Jarak nukleasi maksimum adalah jarak (dalam derajat) dari stasiun sampai stasiun ini dapat berkontribusi pada asal baru. Jika jarak ini 180 °, stasiun ini dapat berkontribusi pada asal-usul baru di seluruh dunia. Namun, jika jaraknya hanya 10 °, jangkauan stasiun ini terbatas. Ini adalah pengaturan yang membantu dalam kasus stasiun biasa-biasa saja di wilayah di mana terdapat banyak stasiun yang bagus dan terpercaya di dekatnya. Stasiun ini kemudian tidak menimbulkan risiko terhadap lokasi yang dihasilkan di luar jarak nukleasi maksimum. Kode jaringan dan stasiun mungkin wildcard (\*) untuk kenyamanan. Misalnya.:  
\* \* 1 90  
GE \* 1 180  
GE HLG 1 10  
TE RGN 0 10  
Contoh di atas berarti bahwa semua stasiun dari semua jaringan secara default dapat membuat peristiwa baru dalam 90 °. Stasiun GE dapat membuat acara dari jarak jauh, kecuali stasiun HLG yang agak bising di jaringan GE, yang dibatasi hingga 10 °. Dengan menetapkan kolom ke-3 menjadi 0, TE RGN diabaikan oleh scautoloc.

* Preliminary origins

Biasanya, scautoloc tidak akan melaporkan asal-usulnya dengan jumlah fase pendefinisian tertentu (ditentukan oleh autoloc.minPhaseCount), biasanya 6-8 fase. Dalam kasus peristiwa yang menghasilkan amplitudo yang sangat besar pada sejumlah stasiun yang cukup (selanjutnya disebut "peristiwa XXL"), adalah mungkin untuk menghasilkan asal mula awal yang didasarkan pada jumlah yang kurang.  
Prasyaratnya adalah bahwa semua picks ini memiliki amplitudo dan SNR luar biasa yang luar biasa dan berada dalam wilayah yang relatif kecil. Picks tersebut selanjutnya disebut "XXL picks". Pick dipilih secara internal sebagai "XXL pick" jika amplitudonya melebihi ambang batas tertentu (ditentukan oleh autoloc.thresholdXXL) dan memiliki SNR> 8. Untuk pengambilan SNR yang lebih besar dengan amplitudo yang lebih kecil dapat mencapai tag XXL, karena dibenarkan untuk diobati. Sebuah pick SNR besar sebagai XXL pick bahkan jika amplitudonya agak di bawah ambang amplitudo XXL. Kriteria XXL harus dinilai sebagai solusi untuk mengidentifikasi picks yang membenarkan nukleasi asal mula.

* Pembalakan

Scautoloc menghasilkan dua jenis file log: file log aplikasi biasa yang berisi pemrosesan dan riwayat lokasi dan log pilihan opsional. Log pengambilan berisi semua picks yang diterima dengan amplitudo yang terkait dalam file teks sederhana, satu entri per baris. Log pilihan ini harus selalu aktif karena memungkinkan pemutaran untuk pemecahan masalah dan optimalisasi scautoloc. Jika sesuatu tidak berjalan seperti yang diharapkan, memutar ulang log pilihan akan memberi cara yang berguna untuk menemukan sumber masalahnya tanpa perlu mengolah bentuk gelombang mentah lagi. File log aplikasi berisi informasi aneka dalam format variabel. Format entri bisa berubah kapan saja, jadi tidak ada aplikasi hilir yang harus bergantung padanya. Ada beberapa baris khusus, namun. Ini berisi kata kunci tertentu yang memungkinkan penyaringan informasi penting dengan menggunakan grep. Kata kunci ini adalah NEW, UPD dan OUT, untuk yang baru, diperbarui dan keluaran asal. Mereka dapat digunakan mis. seperti  
Grep '\ (NEW \ --- UPD \ --- OUT \)' ~ / .seiscomp3 / log / scautoloc.log  
Ini akan mengekstrak semua baris yang berisi kata kunci di atas, memberikan sejarah asal yang sangat sederhana (dan primitif).

### scmag

Tujuan dari scmag adalah menghitung besaran. Dibutuhkan amplitudo dan asal mula sebagai masukan dan menghasilkan StationMagnitudes dan NetworkMagnencies sebagai output. Besaran yang dihasilkan dikirim ke kelompok "MAGNITUDE". Scmag tidak mengakses bentuk gelombang apapun. Ini hanya menggunakan amplitudo yang telah dihitung sebelumnya, mis. Dengan scamp. Tujuan dari scmag adalah decoupling perhitungan besarnya dari pengukuran amplitudo. Hal ini memungkinkan beberapa modul untuk menghasilkan amplitudo secara bersamaan, seperti scautopick dan scamp. Begitu asal datang, amplitudo yang terkait dengan pengambilan diambil dari buffer memori atau database untuk menghitung besarannya. Saat ini jenis skala berikut diterapkan:

MLv  
Skala lokal dihitung pada komponen vertikal dengan menggunakan istilah koreksi agar sesuai dengan standar ML

MLh  
Skala lokal dihitung pada komponen horisontal untuk spesifikasi SED. Lihat mlh

Mb  
Ukuran gelombang bodi band sempit menggunakan filter Butterworth urutan ketiga dengan frekuensi sudut 0,7 dan 2,0 Hz.

MB  
Badan lebar pita melintang.

Mwp  
Ukuran gelombang tubuh Tsuboi dkk. (1995)  
Selain itu, scmag menghitung besaran turunan berikut ini:

Mw (mB)  
Estimasi momen Mw berdasarkan mB menggunakan regresi Mw vs mB Bormann dan Saul (2008)

Mw (Mwp)  
Estimasi momen Mw berdasarkan Mwp menggunakan Mw vs Mwp regresi Whitmore dkk. (2002).

M (ringkasan)  
Ringkasan besarnya, yang terdiri dari rata-rata tertimbang dari besaran individu dan mencoba menjadi kompromi terbaik antara semua besaran. Lihat di bawah untuk konfigurasi dan juga mengingatkan bagaimana menambahkan besarnya ringkasan ke daftar besaran pilihan yang mungkin atau cara membuatnya selalu disukai.

Mw (avg)  
Estimasi momen Mw berdasarkan rata-rata tertimbang besaran lainnya, saat ini MLv, mb dan Mw (mB), di masa depan mungkin besaran lainnya juga, terutama yang cocok untuk acara yang sangat besar. Tujuan Mw (avg) adalah memiliki, pada tahap apapun selama pemrosesan, perkiraan "terbaik" besarnya dengan menggabungkan semua besaran yang ada menjadi rata-rata tertimbang tunggal. Awalnya rata-rata hanya terdiri dari pengukuran MLv dan / atau mb, namun begitu pengukuran Mw (mB) tersedia, ini (dan di masa depan, magnitude besar lainnya) menjadi semakin berat rata-rata.

### scamp

Langkah-langkah scamp beberapa jenis amplitudo dari data waveform. Ini mendengarkan asal usul dan mengukur amplitudo dalam jendela waktu yang ditentukan dari titik asal. Amplitudo objek yang dihasilkan dikirim ke grup pesan "AMPLITUDE". Scamp adalah rekan scmag. Biasanya, semua amplitudo dihitung sekaligus oleh scamp dan kemudian dipublikasikan. Hanya sangat jarang amplitudo perlu dikompilasi ulang jika lokasi asal berubah secara signifikan. Amplitudo dapat digunakan kembali oleh scmag, membuat perhitungan besarnya dan update efisien. Saat ini, pemetik otomatis di SeisComP 3, scautopick, juga mengukur seperangkat kecil amplitudo (yaitu "snr" dan "mb", rasio signal-to-noise dan amplitudo yang digunakan dalam perhitungan mb magnitude, masing-masing) untuk setiap pick otomatis. Di jendela waktu tetap. Jika sudah ada amplitudo, mis. Yang sebelumnya ditentukan oleh scautopick, scamp tidak akan mengukurnya lagi untuk arus masing-masing. Amplitudo juga dibutuhkan, bagaimanapun, untuk pick manual. Scamp melakukan hal ini juga. Pilihan dengan berat lebih kecil dari 0,5 pada Origin yang sesuai dibuang.  
Amplitudo untuk besaran berikut saat ini dihitung:  
MLv  
Skala lokal dihitung pada komponen vertikal dengan menggunakan istilah koreksi agar sesuai dengan standar ML.  
MLsed  
Amplitudo lokal dihitung pada horizontals. Lihat MLsed  
Mb  
Ukuran gelombang bodi band sempit menggunakan filter Butterworth urutan ketiga dengan frekuensi sudut 0,7 dan 2,0 Hz. Perhatikan bahwa amplitudo ini juga dihitung dengan scautopick untuk semua picks otomatis.  
MB  
Badan lebar pita melintang.  
Mw (mB)  
Estimasi momen Mw berdasarkan mB.

### scqc

scqc menentukan parameter kualitas arus data seismik. Parameter output adalah parameter kontrol kualitas rata-rata (QC) dalam hal pesan kualitas gelombang. Dalam interval reguler, laporan pesan dikirim yang berisi representasi rata-rata jangka pendek dari parameter QC spesifik untuk jangka waktu tertentu. Pesan alarm dihasilkan jika rata-rata jangka pendek (misalnya 90an) dari parameter QC berbeda dari rata-rata jangka panjang (misalnya 3600) lebih tinggi dari ambang batas yang ditentukan.  
Untuk menghindari beban yang berlebihan, pesan QC dikirim didistribusikan dari waktu ke waktu. Berikut parameter yang ditentukan:  
Keterlambatan [s]  
Perbedaan waktu antara waktu kedatangan dan catatan akhir akhir waktu ditambah setengah panjang rekaman (mean data latency, berlaku untuk semua sampel dalam catatan)  
Latency [s]  
Perbedaan waktu antara waktu sekarang dan catatan waktu kedatangan (feed latency)  
Offset [counts]  
Nilai rata-rata semua sampel dari sebuah catatan  
RMS [diperhitungkan]  
Nilai offset rata-rata akar offset (RMS) dikoreksi  
Spike (interval [s], amplitudo [counts])  
Dalam kasus terjadinya lonjakan dalam rekaman parameter ini memberikan interval waktu antara lonjakan yang berdekatan dan amplitudo rata-rata lonjakan; Secara internal daftar lonjakan disimpan (spike time, spike amplitudo); Algoritma finder spike masih pendahuluan  
Gap (interval [s], panjang [s])  
Dalam kasus kesenjangan data antara dua catatan berturut-turut, parameter ini memberikan interval waktu jeda dan panjang rata-rata celah  
Waktu [%]  
Miniseed record timing quality (0 - 100%)

### scmaster

Scmaster dirancang sebagai sejenis mikrokernel atau mediator yang mendelegasikan permintaan klien. Oleh karena itu aplikasi kunci yang bertanggung jawab atas orkestrasi sistem terdistribusi. Agar dapat berpartisipasi dalam sistem terdistribusi, klien perlu mengirimkan permintaan koneksi ke master scmaster. Pada gilirannya master mengembalikan pesan pengakuan yang memberi tahu klien tentang penerimaan atau penolakannya. Jika permintaan sambung berhasil, pesan pengakuan akan memberi klien pesan kelompok yang tersedia kepadanya. Selain itu, semua klien yang terhubung saat ini akan diberi tahu tentang anggota yang baru bergabung. Jika master dikonfigurasi dengan database klien juga akan menerima pesan tindak lanjut langsung yang menyimpan alamat database ini. Alamat tersebut dapat digunakan untuk mengambil data arsip nanti. Setelah koneksi terbentuk setiap pesan akan melewati master terlebih dahulu dimana diproses sesuai dan kemudian diteruskan ke kelompok sasaran. Setelah klien selesai memproses pesan putuskan akan dikirim ke master yang pada gilirannya memberitahukan semua klien yang tersisa tentang kepergian tersebut.  
Scmaster dapat dikonfigurasi dengan database untuk memastikan integritas sistem. Sebelum pesan didistribusikan oleh scmaster pesan tersebut ditulis ke database yang ditentukan. Dengan cara ini setiap pesan disimpan sebelum memasuki sistem. Jika terjadi kerusakan, semua informasi yang diperlukan dapat dipulihkan dari database. Saat ini, driver ada untuk MySQL, PostgreSQL dan sqlite. Perhatikan bahwa scmaster bisa berjalan tanpa database tapi kehilangan integritas data dalam melakukannya.

### SeedLink

SeedLink? Adalah protokol akuisisi data real-time dan perangkat lunak client-server yang mengimplementasikan protokol ini. SeedLink? Protokol didasarkan pada TCP. Semua koneksi diprakarsai oleh klien. Selama fase handshaking, klien dapat berlangganan ke stasiun dan aliran tertentu dengan menggunakan perintah sederhana dalam pengkodean ASCII. Saat handshaking selesai, aliran SeedLink? "Paket" yang terdiri dari SeedLink 8 byte? Header (berisi nomor urut) diikuti dengan rekaman Mini-SEED 512 byte, dikirim ke klien. Paket masing-masing stasiun selalu ditransfer dalam urutan tepat waktu (FIFO). SeedLink? Implementasi yang digunakan di SeisComP adalah yang tertua dan paling banyak digunakan, namun, ada implementasi lain. Implementasi terkenal lainnya dikerahkan di IRIS DMC dan beberapa produsen telah menerapkan SeedLink? Di firmware digitizer mereka Semua implementasi umumnya kompatibel, namun tidak semuanya mendukung SeedLink penuh protokol. Di sisi lain IRIS DMC mengimplementasikan beberapa ekstensi yang tidak didukung oleh server lain. Berikut ini kami menggunakan "SeedLink" Untuk menunjukkan SeedLink. Implementasi yang digunakan di SeisComP. Sumber data SeedLink Server bisa apa saja yang didukung oleh SeedLink? Plug-in - sebuah program kecil yang mengirimkan data ke SeedLink. Server Plug-in dikendalikan oleh SeedLink Server, mis., Plug-in secara otomatis dinyalakan ulang jika macet atau terjadi timeout. Data yang disertakan oleh plug-in dapat berupa paket Mini-SEED atau hanya sampel bilangan bulat mentah dengan informasi waktu yang menyertainya. Dalam kasus terakhir, SeedLink? Server menggunakan "Stream Processor" terintegrasi untuk membuat aliran data yang diinginkan dan merakit paket Mini-SEED.

#### Handshaking

Ketika koneksi TCP / IP telah ditetapkan, server akan menunggu klien memulai handshaking tanpa mengirimkan data ke klien terlebih dahulu. Selama handshaking klien mengirim SeedLink? Perintah ke server Perintah digunakan untuk mengatur koneksi ke mode tertentu, selector aliran setup, meminta nomor urut paket untuk memulai dan akhirnya memulai transmisi data. SeedLink? Perintah terdiri dari senar ASCII diikuti oleh nol atau beberapa argumen yang dipisahkan oleh spasi dan diakhiri dengan carriage return (<cr>, kode ASCII 13) diikuti oleh optional linefeed (<lf>, kode ASCII 10). Perintah dapat dibagi menjadi dua kategori: "perintah tindakan" dan "perintah pengubah". Perintah aksi melakukan fungsi seperti memulai transfer data. Perintah modifier digunakan untuk mengambil spesialisasi atau memodifikasi fungsi yang dilakukan oleh perintah aksi yang diikuti. Ketika sebuah server menerima perintah pengubah, ia merespons dengan string ASCII "OK" diikuti oleh carriage return dan feed line untuk mengetahui bahwa perintah tersebut telah diterima. Jika perintah tidak dikenali oleh server atau memiliki parameter yang tidak valid, maka string ASCII "ERROR" dikirim sebagai tanggapan terhadap klien diikuti oleh carriage return dan feed line. Klien tidak boleh mengirim perintah lebih lanjut sebelum menerima respons terhadap perintah pengubah sebelumnya. Jika terjadi kesalahan jaringan atau timeout, klien harus menutup koneksi dan memulai sesi baru ..

## Desain Basis Data

Pada server terdapat media penyimpan data berupa database MySQL. Database MySQL dibuat untuk menyimpan data GeoJSON atau parameter-parameter yang dibutuhkan untuk analisa gempa. Penggunaan MySQL umum digunakan dalam mengimplementasikan database sehingga desain server pada proyek ini menggunakan MySQL. Oleh karena itu, akan dibuat kolom-kolom yang sesuai dengan data GeoJSON.

Data-data tersebut meliputi:

1. Nama sensor + *Timestamp*
2. *Time Zone*
3. Interval waktu pengiriman
4. Garis Lintang
5. Garis Bujur
6. Percepatan sumbu x
7. Percepatan sumbu y
8. Percepatan sumbu z

Nama sensor beserta *timestamp* akan menjadi data yang unik sehingga bisa dijadikan *primary key* untuk pengisian *query* pada *database* MySQL. Desain program untuk dapat melakukan penyimpanan pesan menuju Database adalah sebagai berikut:



**Gambar Flowchart dari program konversi pesan dari RabbitMQ menuju MySQL**