

Pengembangan Modul Otomatisasi Pengunduhan Citra Sentinel-1A Berbasis Web Menggunakan Metode Prototyping

Development of Web-Based Sentinel-1A Image Download Automation Module Using Prototyping Method

MUHAMMAD ASYHAR AGMALARO^{1*}, IMAS SUKAESIH SITANGGANG¹,
TAUFIK HIDAYAT¹

Abstrak

Citra Sentinel-1A dapat dimanfaatkan untuk berbagai keperluan seperti survei dan pemetaan penggunaan lahan pertanian. Sebagai contoh, citra Sentinel dapat digunakan untuk melakukan proses identifikasi lahan, validasi hasil tanam dari tanaman hortikultura seperti bawang putih. Namun, saat ini proses akuisisi dan pengunduhan data citra Sentinel masih dilakukan secara manual dengan beberapa tahapan sehingga dirasa masih belum efektif dan efisien. Oleh karena itu perlu diterapkan suatu cara alternatif untuk mempercepat akuisisi data Sentinel dengan mengoptimalkan proses otomatisasi pengunduhan data citra Sentinel. Penelitian ini bertujuan untuk membangun sebuah modul *front-end* untuk mengotomatisasi proses pengunduhan data citra Sentinel berbasis *web* menggunakan *Framework* Django. Metode *prototyping* digunakan untuk melakukan pembangunan modul *front-end* otomatisasi pengunduhan citra Sentinel. Metode ini dipilih didasarkan pada kelebihanannya dalam mendapatkan umpan balik yang cepat dari pengguna dari setiap iterasi yang dilakukan sehingga perbaikan dapat segera dilakukan sesuai dengan kebutuhan pengguna. Hasil dari penelitian ini adalah sebuah sistem otomatisasi pengunduhan citra Sentinel-1A. Sistem dapat melakukan pengunduhan data citra Sentinel melalui peta digital atau dengan memvalidasi data *geoJson* yang dimasukkan oleh pengguna. Pengembangan sistem ini dilakukan sebanyak dua iterasi. Dalam pengujian secara *black-box*, semua fungsi pada modul yang telah dikembangkan, berhasil dilakukan tanpa ada menunjukkan kesalahan.

Kata Kunci: Citra Sentinel, *Framework* Django, *Front-End*, Otomatisasi.

Abstract

Sentinel-1A imagery can be used for various purposes, such as surveys and agricultural land use mapping. For example, Sentinel-1A image can be used to carry out land processing and validate crop yields from horticultural crops such as garlic. However, the acquisition and download of Sentinel images are currently done manually with several stages, so it still needs to be more effective and efficient. Therefore, an alternative way to support the acquisition of sentinel data is necessary by optimizing the process of automating the download of Sentinel data. This study aims to build a front-end module to automate the downloading of web-based Sentinel image data using the Django Framework. The prototyping method is used to develop a front-end module for Sentinel image download automation. This method was chosen based on its advantages in getting feedback from each user from every iteration carried out so that improvements can be made quickly according to user needs. The result of this research is an automated system for downloading Sentinel-1A images that can download Sentinel image data via maps or by validating geoJson data entered by the user. The development of this system is carried out in two iterations. All functions in the developed module were successfully performed in black box testing without showing any errors.

Keywords: Automation, Django Framework, Front-End, Sentinel Image

¹ Departemen Ilmu Komputer IPB-University, Jl Meranti Wing 20 Level 5 Kampus IPB Darmaga 16680, 0251-8625584;

* Penulis Korespondensi: Tel/Faks: 0251-8625584; Surel: agmalara@ipb.ac.id

PENDAHULUAN

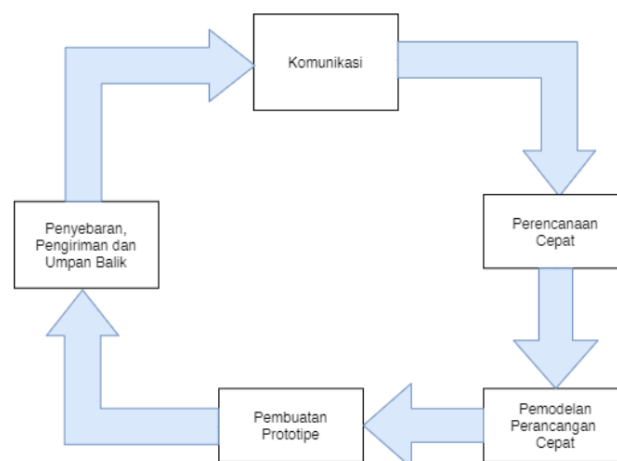
Perkembangan teknologi di bidang pemetaan bumi sudah banyak membantu manusia di berbagai bidang. Salah satunya yang berkembang sangat signifikan yaitu di bidang penginderaan jarak jauh menggunakan satelit. Perkembangan ini dapat dilihat dari semakin pentingnya penggunaan penginderaan jauh bagi penyediaan informasi sumber daya alam dan dampak lingkungan yang ditimbulkan akibat pengelolaannya (Wirandha *et al.* 2015). Salah satu satelit yang digunakan untuk penginderaan jarak jauh adalah Sentinel-1. Sentinel-1 merupakan satelit yang diluncurkan oleh The European Space Agency (ESA) dengan tujuan utama untuk memonitor daratan dan lautan. Sentinel-1 menggunakan sensor *C-band synthetic aperture radar* (C-SAR), sehingga menawarkan pencitraan resolusi menengah dan tinggi di segala cuaca (Putri *et al.* 2018). Dalam bidang pertanian, penginderaan jarak jauh perlu dilakukan karena survei dan pemetaan penggunaan lahan pertanian secara konvensional memerlukan tenaga dan biaya yang besar serta memakan waktu yang lama dalam pengambilan dan pengolahan datanya (Pötter 2014).

Kelebihan dalam pemanfaatan Citra Sentinel-1A sebagai alat survei dan pemetaan lahan salah satunya karena berupa sistem radar *Synthetic Aperture Radar* (SAR), merupakan penginderaan jauh menggunakan gelombang mikro, sehingga dapat mengambil informasi spasial di bumi dalam berbagai kondisi. Citra Sentinel-1A dapat dimanfaatkan untuk survei dan pemetaan penggunaan lahan pertanian seperti tanaman bawang putih, dimana dalam mengidentifikasi bawang putih memerlukan pengunduhan citra Sentinel-1A yang lebih cepat agar identifikasi dapat berjalan lebih maksimal. Citra yang digunakan untuk survei dan pemetaan penggunaan lahan pertanian adalah citra dengan mode operasional *Interferometric Wide* (IW) yang memiliki *dual polarization* (HH+HV, VV+VH) dan *single* (HH, VV), karena di Indonesia hanya citra dengan *dual polarization* VV+VH saja yang tersedia (Komaraasih *et al.* 2020).

OpenHub (scihub.copernicus.eu) menyediakan *Graphical User Interface* untuk proses unduh data Sentinel dimana pengguna bisa menentukan parameter pencarian dan juga menggambar area yang dikehendaki secara manual melalui enam tahapan (Gambar 1). Tahapan yang dilakukan yaitu (i) mengakses website <https://scihub.copernicus.eu/> (ii) melakukan login akun yang telah terdaftar, jika belum terdaftar maka perlu melakukan register untuk mendapatkan akun (iii) mendapatkan konfirmasi email untuk akun yang telah terdaftar (minimal tiga sampai empat hari kerja) (iv) Seleksi area yang akan diunduh, dengan membuat area poligon pada peta digital (v) filtrasi parameter data Sentinel yang akan diunduh, seperti tanggal satelit mengorbit, jenis platform satelit yang dibutuhkan, tipe produk yang dibutuhkan, sensor mode yang digunakan, dan polarisasi yang tersedia (vi) Tahapan selanjutnya adalah pemilihan data yang sesuai dengan parameter yang ditetapkan kemudian mengunduh data yang sudah dipilih. Dari Tahapan tersebut ada kendala bahwa pengguna harus melakukan *self registration* dan ada beberapa tahapan manual yang harus dilakukan. Selain itu tersedia pula *Application Programming Interface* (API) Hub (<https://apihub.copernicus.eu/apihub/>) yang dikhususkan bagi pengguna yang sudah terbiasa dengan *scripting*, sehingga dapat melakukan pengunduhan tanpa menggunakan antarmuka. Bagi pengguna yang terbiasa menulis program komputer dapat menjalankan *Application Programming Interface* (API) Hub menggunakan kode *Python* dari <https://github.com/sentinelsat/sentinelsat>. Terdapat juga jasa pengadaan data komersial dengan akses *map server* bagi pengguna yang tidak ingin menunggu lama untuk mengunduh data dan melakukan pra proses data, seperti dari <https://creodias.eu/>, <https://mundiwebservices.com> dan <https://www.onda-dias.eu>. Tetapi karena sifatnya komersial maka ada biaya yang harus dikeluarkan untuk menggunakan layanan tersebut. Jika proses Otomatisasi dapat dilakukan dengan memadukan fasilitas API dan kode

METODE

Pengembangan Modul *Front-End* pada penelitian ini menggunakan metode *prototyping*. Alasan pemilihan metode ini dikarenakan kebutuhan menyelesaikan task dengan waktu yang relatif singkat sehingga dibutuhkan komunikasi dengan frekuensi yang cukup banyak antara pengembang dengan pengguna atau stakeholder. Metode ini memungkinkan pengembang dapat bekerja lebih baik dalam menentukan kebutuhan pengguna dan sebaliknya pengguna dapat berperan aktif dalam memberikan masukan untuk pengembangan sistem. Dengan metode *prototyping* ini akan dihasilkan *prototype* sistem sebagai perantara pengembang dan pengguna agar dapat berinteraksi dalam proses kegiatan pengembangan sistem informasi. Metode *prototyping* digunakan karena metode ini memiliki manfaat antara lain yaitu mewujudkan sistem sesungguhnya dalam sebuah replika sistem yang akan berjalan, menampung masukan dari pengguna untuk kesempurnaan sistem, pengguna akan lebih siap menerima setiap perubahan sistem yang berkembang sesuai dengan berjalannya *prototype* sampai dengan hasil akhir pengembangan yang akan berjalan nantinya, *prototype* dapat ditambah maupun dikurangi sesuai berjalannya proses pengembangan. Kemajuan tahap demi tahap dapat diikuti langsung oleh pengguna, penghematan sumber daya dan waktu dalam menghasilkan produk yang lebih baik dan tepat guna bagi pengguna. Menurut Pressman dan Maxim (2014), ada terdapat lima tahapan dalam metode *prototyping*, yaitu komunikasi, perencanaan cepat, pemodelan perancangan cepat, konstruksi *prototype*, dan pengiriman sekaligus umpan balik dari *client* (Gambar 2).



Gambar 2 Metode *prototyping* (Pressman dan Maxim 2014)

Penjelasan tahapan *prototyping* untuk pengembangan modul *front-end* otomatisasi pengunduhan citra Sentinel adalah sebagai berikut:

Tahap Komunikasi

Tahap pertama yang dilakukan pada metode *prototyping* adalah komunikasi. Pada tahap ini dilakukan identifikasi masalah dan kebutuhan sistem melalui komunikasi yang intensif dengan *stakeholder* dan pengguna. Komunikasi bisa dilakukan secara langsung dengan cara wawancara tatap muka atau menggunakan komunikasi tidak langsung menggunakan media sosial. Komunikasi yang intens antara pengguna dan pengembang dapat membantu pengembang dalam menentukan kebutuhan pengguna yang sebenarnya dan meminimalkan kesalahan persepsi sehingga didapatkan kebutuhan sistem yang maksimal. Pada tahap ini pengembang mewawancarai pengguna terkait fitur-fitur yang akan ditambahkan pada modul *front-end* untuk otomatisasi pengunduhan citra Sentinel.

Perencanaan Cepat

Tahap kedua di dalam metode *prototyping* adalah perencanaan cepat. Pada tahap perencanaan cepat dilakukan analisis kebutuhan pengguna dan identifikasi spesifikasi kebutuhan perangkat lunak. Hasil analisis digambarkan kedalam bentuk *usecase*, selanjutnya dilakukan pembuatan prototipe antarmuka. Pembuatan prototipe ini dilakukan untuk membantu pengembangan sistem agar lebih terarah sesuai dengan kebutuhan pengguna. Pada tahap ini pembuatan prototipe antarmuka direncanakan dan dilakukan untuk mendapat gambaran bentuk modul *front-end* yang dirancang. Perencanaan secara cepat merupakan tahapan yang dilakukan oleh pengembang dalam menentukan beberapa fitur yang nantinya akan diimplementasikan dan hasil akhir dari perencanaan cepat adalah gambaran umum dari aplikasi.

Pemodelan Perancangan Cepat

Tahapan ini dilakukan setelah pengembang mengetahui berbagai rencana fitur yang akan ditanamkan ke dalam modul *front-end* yang akan dikembangkan. Tahap pemodelan perancangan secara cepat menggunakan *Unified Modeling Language* (UML) diagram seperti *Usecase Diagram*, *Activity Diagram* sesuai dengan arah pengembangan aplikasi. Tahap ini menghasilkan model yang memberikan gambaran yang dapat memetakan dan mempresentasikan semua aspek-aspek kebutuhan perangkat lunak modul *front end* pengunduhan citra Sentinel yang dapat dilihat oleh pengguna. UML bukanlah merupakan bahasa pemrograman tetapi model-model yang dihasilkan dalam bentuk diagram visual sebagai cetak biru dan sarana dalam perancangan sistem berorientasi objek.

Pembuatan Prototipe

Pada tahap ini dibuat prototipe dengan fungsionalitas yang mewakili sistem yang dirancang dan dikembangkan. Prototipe yang dibuat merupakan aplikasi berbasis *web* menggunakan bahasa pemrograman Python, dengan Django sebagai *framework*-nya. Arsitektur yang digunakan Django adalah pendekatan perancangan *web* dengan bentuk *Model-View-Template* (MTV). *Model* adalah lapisan akses data untuk mengatur bagaimana cara akses dan alur sebuah *request* yang diminta pengguna. *Template* adalah representasi tampilan web yang akan dibuat, sedangkan *View* merupakan penghubung antara *Model* dan juga *Template*. Prototipe yang dihasilkan berupa tampilan yang akan digunakan pengguna untuk melakukan pengunduhan citra Sentinel secara otomatis dengan langkah yang lebih efisien.

Penyebaran Pengiriman dan Umpan Balik

Tahap terakhir pada metode *Prototyping* yaitu *deployment*, *delivery*, dan *feedback*. Pada tahap ini prototipe akan diserahkan kepada pengguna untuk mendapatkan umpan balik mengenai modul *front-end* yang dihasilkan terkait dengan kebutuhan pengguna pada tahap komunikasi. Iterasi akan terjadi saat prototipe diperbaiki untuk memenuhi dan melengkapi fitur-fitur yang ada sesuai dengan kebutuhan pengguna.

HASIL DAN PEMBAHASAN

Berikut pengembangan modul *front-end* otomatisasi pengunduhan citra Sentinel yang mengikuti tahapan pada metode *prototyping*:

Komunikasi

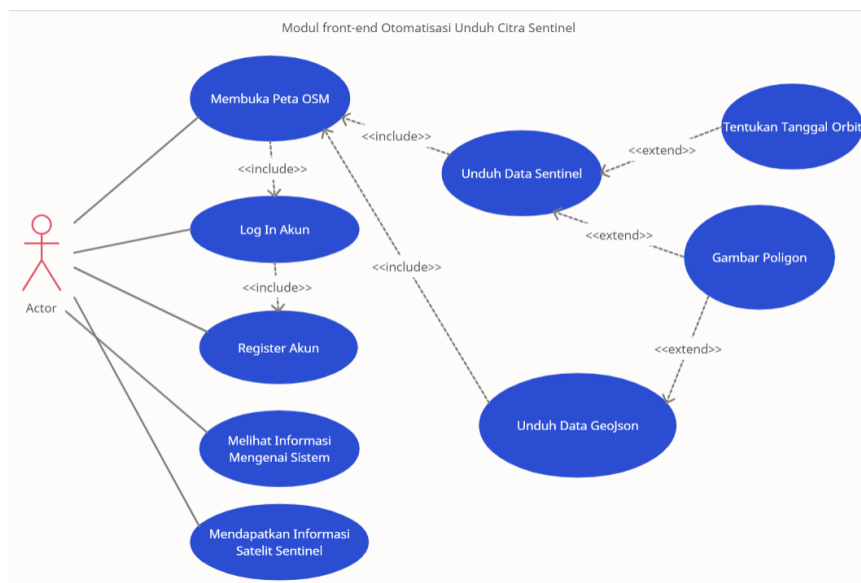
Pada tahapan komunikasi iterasi pertama dilakukan diskusi antara peneliti dan pengguna maupun *stakeholder* untuk mengumpulkan informasi umum dalam mengembangkan modul *front-end*. Pengambilan informasi dilakukan dengan cara wawancara langsung kepada *stakeholder* dan juga peneliti yang juga memanfaatkan Sentinel untuk kegiatan penelitian. Pengembang mewawancarai pengguna dan juga *stakeholder* untuk mendapat ide mengenai apa yang diminta dari sistem. Hasil pengumpulan informasi ke pengguna dan juga *stakeholder* berisi tentang permasalahan yang dirasakan selama melakukan pengambilan citra satelit Sentinel-1A diantaranya bahwa untuk mendapatkan citra Sentinel-1A secara manual harus registrasi melalui *website Copernicus Open Access Hub* (<https://scihub.copernicus.eu/>) dan membutuhkan waktu sampai beberapa hari untuk mendapatkan akun akses data. Selain itu, *stakeholder* juga meminta semua operasi pengambilan citra bisa dilakukan secara otomatis dalam sekali tahapan saja dengan cakupan data yang tersedia di wilayah Indonesia. Komunikasi dilanjutkan pada iterasi kedua dengan *Stakeholder* yang menghasilkan beberapa kebutuhan baru. Pada tahapan komunikasi iterasi kedua ini peneliti tidak melakukan komunikasi dengan *real user*, tetapi peneliti berasumsi kedua *stakeholder* dapat mewakili *real user*. Hasil dari wawancara terhadap kedua *stakeholder* tersebut, diantaranya tetap menambahkan fungsi register email dan login pada modul otomatisasi pengunduhan data citra Sentinel agar pengguna sistem nanti bisa lebih terkontrol. Namun validasi pengguna baru saat register melalui email tidak perlu menunggu waktu yang lama dan pengguna dapat segera menggunakan sistem untuk melakukan pengunduhan data Sentinel. Fungsi lain dari registrasi dan login menggunakan email yang valid adalah agar pengguna dapat mendapatkan notifikasi proses pengunduhan semua data citra Sentinel-1A telah selesai dilakukan dan pengguna mendapatkan informasi lokasi pengambilan citra. *Stakeholder* juga mengusulkan untuk membuat fungsi konversi area poligon yang dibuat oleh pengguna pada peta digital menjadi file dengan format GeoJson sehingga dapat memenuhi kebutuhan penggunaan informasi poligon dalam pra proses selanjutnya atau penelitian lainnya. Fasilitas baru ini tidak tersedia di *website Copernicus Open Access Hub*. Seluruh hasil wawancara tersebut dituangkan di dalam *User Story* (Tabel 1).

Tabel 1 *User Story* hasil komunikasi pada iterasi pertama dan kedua

Pengguna	Kebutuhan Pengguna	<i>User Story</i>
Iterasi Pertama		
Peneliti	Mengetahui informasi mengenai sistem	Sebagai pengguna, saya ingin mendapatkan informasi mengenai sistem yang dibuat, manfaat dari menggunakan sistem yang telah dikembangkan.
	Mengetahui informasi mengenai Sentinel	Sebagai pengguna, saya ingin mendapatkan informasi mengenai Sentinel. Bagaimana Sentinel dapat diakuisi dan diunduh dan bagaimana user menerima <i>file</i> yang sudah diunduh.
	Fungsi Otomatisasi Unduh Citra	Sebagai pengguna, saya ingin mendapatkan data mentah yang sudah difiltrasi sehingga saya tidak perlu memilah untuk pemercepatan proses pengolahan data.
Iterasi Kedua		
Peneliti	Fungsi Login	Sebagai Pengguna harus dapat melakukan registrasi akun dan juga melakukan login untuk mengakses sistem otomatisasi pengunduhan citra Sentinel.
	Mengunduh file geoJson	Sebagai pengguna, saya ingin bisa mendapatkan file geoJson ketika saya sudah menggambar koordinat di peta sistem yang sudah dibuat. Ini bisa diperlukan untuk kegiatan lainnya.

Perencanaan Cepat

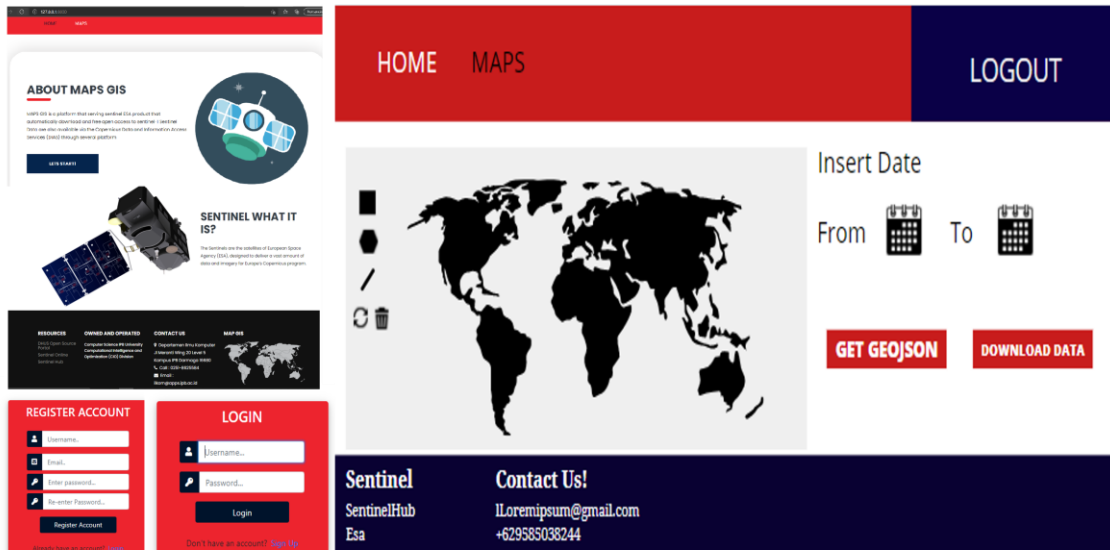
Pada tahapan perencanaan cepat dilakukan perumusan kebutuhan fungsional yang nantinya akan ditanamkan ke dalam modul *Front-end*. Tahapan perencanaan cepat diawali dengan membuat *use case diagram*. *Use case diagram* dibuat berdasarkan *user story* yang telah dibuat pada tahap komunikasi. Aktor yang terdapat pada sistem tersebut hanya satu yaitu pengguna pengunduhan citra. *Use case* pada iterasi pertama dan kedua memfasilitasi semua kebutuhan pengguna yang harus dikembangkan di dalam modul. Berdasarkan hasil *user story*, peneliti membuat rancangan prototipe antarmuka untuk mempermudah pendekatan masalah yang dimiliki. Tampilan prototipe antarmuka tersebut merepresentasikan fungsi dan fitur-fitur yang ada di dalam modul otomatisasi pengunduhan citra Sentinel. Detail kebutuhan calon pengguna yang digambarkan dalam bentuk hasil akhir *use case diagram* bisa dilihat pada Gambar 3.



Gambar 3 Use case Diagram modul front-end pengunduhan citra Sentinel

Tampilan *homepage* terdiri dari komponen navigasi, *footer* dan isi. Komponen tampilan didapatkan dari *Bootstrap* yang disesuaikan dengan kebutuhan yang dilakukan pada tahap komunikasi. Tampilan isi *homepage* berisi informasi mengenai sistem yang dibuat peneliti dan manfaat yang didapatkan pengguna. Hal ini bertujuan agar pengguna lebih siap dalam menggunakan sistem yang nanti akan digunakan.

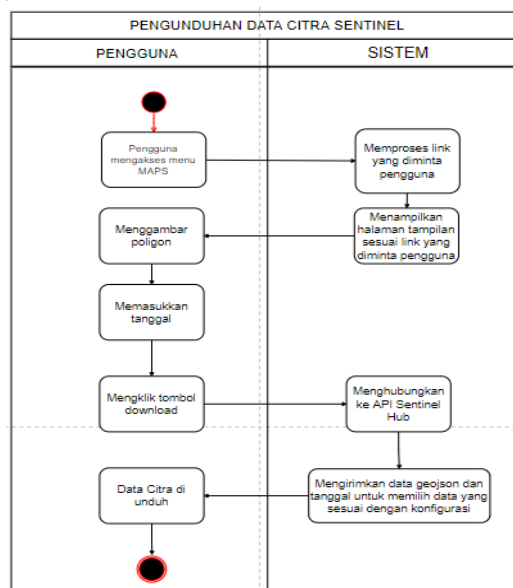
Dari *user story* juga terdapat fitur yang ditambahkan dimana pengguna sistem melakukan registrasi akun serta login. Registrasi dilakukan otomatis oleh sistem sehingga tidak membutuhkan waktu yang lama untuk proses validasi email untuk *username* dan *password* dan pengguna dapat langsung mengakses sistem otomatisasi pengunduhan citra Sentinel setelah melakukan login. Sistem otomatisasi pengunduhan citra tersebut dapat melakukan pengaturan tanggal untuk menentukan data citra di tanggal berapa yang akan diambil. Sistem juga dapat menggambar poligon untuk mendapatkan data geoJson serta sistem dapat melakukan pengunduhan data geoJson dan juga data citra Sentinel. Tampilan Prototipe antarmuka yang berfungsi membantu pengembang dalam mengembangkan sistem dapat dilihat pada Gambar 4.



Gambar 4 *Prototipe antarmuka* modul front-end otomatisasi pengunduhan Citra Sentinel

Pemodelan dan Perancangan Cepat

Pada tahapan pemodelan dan perancangan cepat digambarkan berbagai fungsi yang ada di dalam modul *front-end* otomatisasi ke dalam bentuk pemodelan *activity diagram*. *Activity diagram* bertujuan untuk memodelkan berbagai proses yang terjadi pada sistem. Dimulai ketika pengguna meminta alamat *website* yang akan diakses, lalu *website* akan menampilkan halaman sesuai dengan alamat yang diminta oleh pengguna. Data akan dikirimkan ke dalam sistem selanjutnya akan diproses divalidasi di dalam *framework* Django. Aktivitas selanjutnya adalah mengakses menu MAPS pada *navigation bar*. Sistem kemudian meneruskan *request* halaman yang diminta pengguna dan selanjutnya pengguna dapat melakukan seleksi area pada peta *Openstreet Map* OSM (dengan menggunakan *library* leaflet) terlebih dahulu untuk mendapatkan informasi poligon berformat geoJson. Berikutnya pengguna memberikan informasi tanggal unduh citra Sentinel. Untuk memulai proses pengunduhan, pengguna perlu menekan tombol *download*. Aktivitas pengguna dalam menggunakan fitur dan fungsi dalam modul, dapat dilihat pada representasi *activity diagram* pada Gambar 5.



Gambar 5 *Activity diagram* modul front-end otomatisasi pengunduhan citra Sentinel

Pembuatan *Prototype*

Berdasarkan prototipe antarmuka, pada tahap ini dilakukan pembuatan prototipe dengan fungsionalitas untuk modul *front-end* yang diawali dengan membangun halaman antar muka untuk fungsi otomatisasi pengunduhan citra Sentinel. Halaman pengunduhan citra Sentinel dibangun menggunakan bahasa HTML dan juga CSS. Halaman tersebut memiliki kolom peta OSM yang digunakan untuk melakukan seleksi area. Halaman pengunduhan citra juga memiliki tombol pengunduhan data geoJson dan data citra Sentinel-1A. Selain itu terdapat tombol untuk memasukkan tanggal yang akan digunakan untuk memilih data tanggal berapa yang dibutuhkan. Pada peta digital tersebut juga ditambahkan sebuah *layer* yang digunakan untuk memproses data geoJson menggunakan *leaflet* dan juga tombol kontrol peta yang digunakan untuk melakukan seleksi area. Halaman pengunduhan juga ditambahkan *sub-menu* yang berisi navigasi halaman utama dan juga tombol logout. Selain itu, pada tahapan ini dibangun sebuah tampilan registrasi dan login akun yang mengakomodir kebutuhan pengguna.

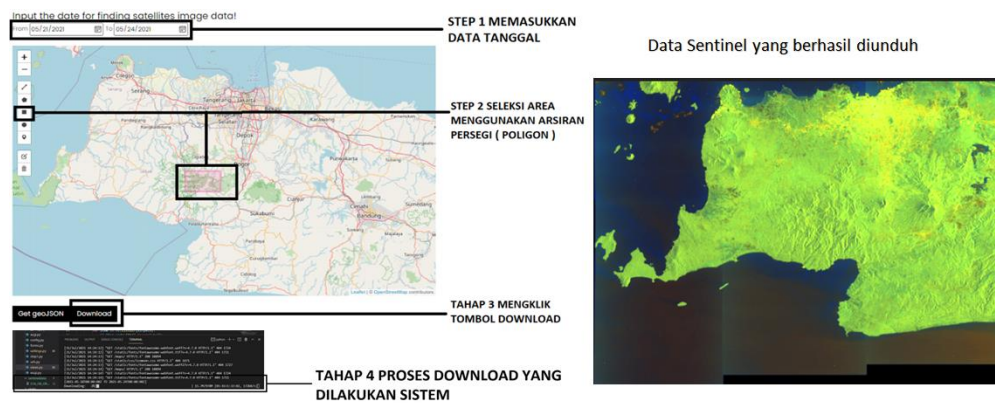
Fungsi otomatisasi pengunduhan di dalam modul dikembangkan di dalam file `views.py` seperti yang ditunjukkan pada Gambar 6. Sistem mengirimkan data seleksi area dan juga tanggal yang sudah ditetapkan pengguna ke dalam API `sentinelhub` dan diteruskan ke dalam sistem otomatisasi pengunduhan data citra Sentinel. Sistem ini menggunakan *library* `sentinelsat` untuk membuat filtrasi data yang dibutuhkan. Tabel 2 mendeskripsikan filtrasi data yang dilakukan untuk memilah data mana yang dibutuhkan pengguna dengan atribut yang dikonfigurasi untuk melakukan mekanisme pengunduhan data citra Sentinel. Proses pengunduhan data Sentinel dapat dilihat pada Gambar 7. Data citra Sentinel tersebut akan masuk ke dalam *folder/sentineldata* jika berhasil diunduh, apabila data gagal untuk diunduh maka otomatis terhapus oleh sistem.

```
#alur download
def download(request):
    if request.method == 'POST':
        data = json.loads(request.body)
        footprint = geojson_to_wkt(geojson.loads(data['geoJson']))
        username = config.username # ask ITC for the username and password
        password = config.password
        api = SentinelAPI(username, password, 'https://apihub.copernicus.eu/apihub') # fill with SMARTSeeds user and password
        tanggal = '{0} TO {1}'.format(data['dateFrom'].replace('.000Z', 'Z'), data['dateTo'].replace('.000Z', 'Z'))
        print (tanggal)
        products = api.query(footprint,
                             producttype =config.producttype,
                             orbitdirection =config.orbitdirection,
                             platformname='Sentinel-1',
                             date=tanggal
                             )
```

Gambar 6 Potongan kode penyambungan API Sentinel dengan sistem

Tabel 2 *Query product* yang diotomatisasi

Atribut	Deskripsi
<i>Footprint</i>	geoJson yang telah didapatkan dari seleksi area menggunakan <i>leaflet</i> .
Producttype	Tipe produk data yang telah terdeteksi.
Orbitdirection	Arah orbit
Platformname	Tipe satelit Sentinel yang digunakan.
Date	Tanggal data yang diperlukan.



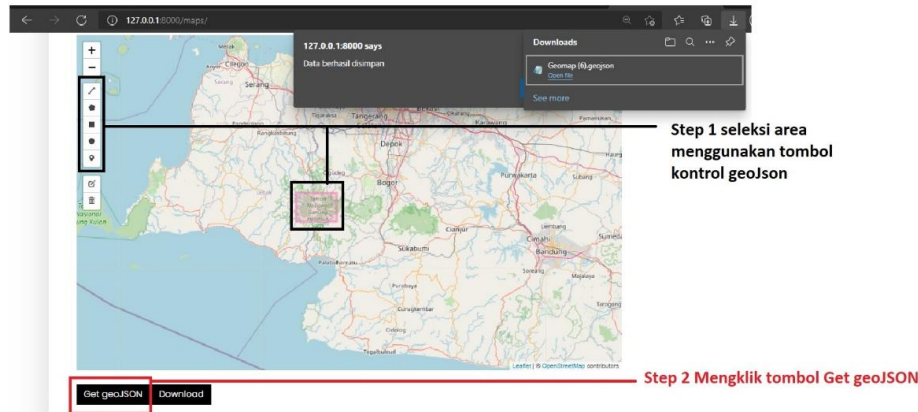
Gambar 7 Proses Pengunduhan Data Sentinel

Sistem yang dikembangkan menggunakan modul *backend* untuk mengatur alur data di dalam sistem. Modul *backend* yang dikembangkan adalah konfigurasi data pengguna dan juga konfigurasi pengunduhan citra Sentinel. Data pengguna yang sudah diregistrasi di dalam sistem akan disimpan di dalam *database Sqlite3* yang merupakan database default yang tersedia di dalam *framework Django*. Pemberitahuan akan diberikan kepada pengguna ketika data citra Sentinel berhasil diunduh (Gambar 8).



Gambar 8 Email notifikasi data citra berhasil diunduh

Fungsi notifikasi tersebut dikembangkan dengan menggunakan *library send_mail* yang disediakan oleh *framework Django*. Fungsi notifikasi ini menggunakan *Simple mail transfer protocol (SMTP)* gmail, sebuah standard elektronik untuk pengiriman email di dalam pengembangannya. Pada fungsi notifikasi dan fungsi *send_mail* yang dibuat, memerlukan subject email berupa isi pesan email dengan nama, alamat email yang didapatkan melalui data registrasi akun. Untuk pengaturan alamat email pengirim dibuat konfigurasinya di dalam modul *backend*. Selain fungsi otomatisasi tahap ini juga mengembangkan fungsi pengunduhan file geoJson. GeoJson adalah sebuah format json yang dirancang mewakili fitur geografis sederhana. Fungsi pengunduhan geoJson dibangun dengan memanfaatkan seleksi area yang sudah dilakukan pengguna menggunakan tombol kontrol leaflet dan layer peta OSM. Otomatisasi pengunduhan file geoJson yang dikembangkan menggunakan fungsi *force download of data text* yang dibangun dengan menggunakan Javascript. File geoJson didapatkan pengguna setelah melakukan seleksi area dan menekan tombol "Get geoJson" (Gambar 9). Data yang diunduh akan memiliki format nama Geomap.geojson.



Gambar 9 Proses yang dilakukan untuk mengunduh file geoJson

Penyebaran Pengiriman dan Umpan Balik

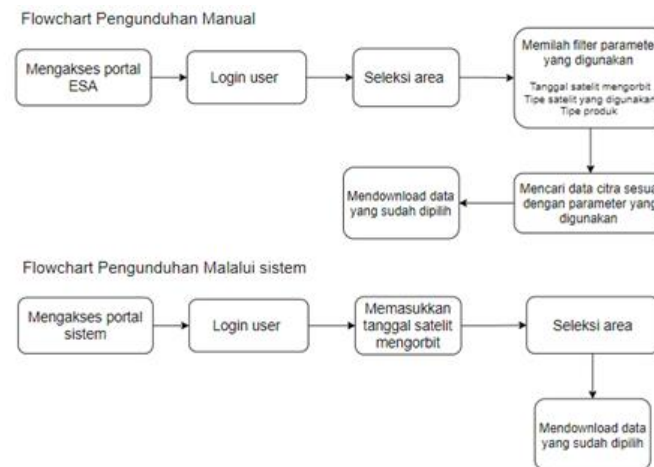
Setelah setiap fungsi yang dibangun sudah selesai dibuat, fungsi tersebut dilakukan pengujian. Pengujian dilakukan pengembang dan juga *Stakeholder* untuk memastikan masukan dan keluaran sistem telah sesuai. Pengujian menggunakan metode *black-box*. Pengujian pada iterasi pertama dan kedua dilakukan dengan memvalidasi kondisi pengujian dan hasil yang diharapkan. Hasil pengujian menunjukkan bahwa semua kondisi pengujian berhasil sesuai dengan yang diharapkan (Tabel 3).

Tabel 3 Hasil pengujian *black-box testing*

Fungsi	Kondisi Pengujian	Hasil yang diharapkan	Hasil Pengujian
Mengakses halaman <i>homepage</i>	Menuliskan alamat <i>homepage</i> dari sistem	Pengguna dapat melihat halaman <i>homepage</i> sistem yang sudah dibuat oleh pengembang	Berhasil
Mengakses informasi Sentinel	Menekan tab link sentinelhub yang terdapat pada <i>footer</i>	Pengguna dialihkan ke website sentinelhub	Berhasil
Melakukan Login	Pengguna dapat melakukan login pada login <i>page</i>	Pengguna berhasil login dan dialihkan ke <i>homepage</i> .	Berhasil
Melakukan pengunduhan file geoJson	Menekan tombol "Get GeoJson"	Pengguna berhasil mendapatkan file geojson.	Berhasil
Melakukan pengunduhan data citra Sentinel	Menekan tombol "Download"	Pengguna berhasil mengunduh file citra Sentinel yang akan masuk kedalam folder /sentineldata	Berhasil

Perbandingan pengunduhan citra yang dilakukan secara manual dengan menggunakan sistem yang telah dikembangkan bisa dilihat pada Gambar 10. *Flowchart* pada pengunduhan citra secara manual terlihat lebih panjang dibandingkan dengan menggunakan sistem yang telah dikembangkan. Perbedaan proses yang dilakukan oleh pengguna terletak pada proses penetapan parameter yang digunakan. Parameter yang sebelumnya ada pada proses pengunduhan manual dilakukan secara otomatis di dalam sistem yang dikembangkan. Pengguna tidak perlu lagi memilih data citra yang akan diunduh karena sudah dibangun sistem otomatisasi untuk melakukan proses tersebut sehingga waktu yang digunakan pengguna akan lebih efisien. Hasil pengujian dan perbaikan sistem otomatisasi pengunduhan citra Sentinel telah berhasil dilaksanakan. Pada

tahapan ini, tidak ada umpan balik dari stakeholder sehingga sistem pengunduhan citra Sentinel selesai dikembangkan.



Gambar 10 Perbandingan proses yang dilakukan pengguna untuk mengunduh data citra Sentinel

SIMPULAN

Sistem otomatisasi pengunduhan citra Sentinel berhasil dikembangkan. Pengembangan sistem ini menggunakan metode prototyping dengan iterasi sebanyak dua kali. Pengguna pada sistem ini adalah peneliti pengguna data citra Sentinel. Sistem ini dapat melakukan pengunduhan data citra Sentinel sesuai dengan data tanggal yang dibutuhkan pengguna. Sistem juga dapat mengunduh data geoJson yang bisa digunakan untuk kebutuhan yang lainnya. Berdasarkan hasil pengujian yang dilakukan dengan metode *black box*, semua fungsional sistem dapat berjalan dengan baik.

DAFTAR PUSTAKA

- Janani R, Vijayani S. 2016. *An Effective Text Pattern Matching Algorithm for Retrieving Information from Desktop*. Indian Journal of Science and Technology. 9(43): 1-11.
- Komaraasih RI, Sitanggang SI, Agmalaro MA. 2020. Sentinel-1A Image Classification for Identification of Garlic Plants using a Decision Tree Algorithm. *International Conference on Computer Science and Its Application in Agriculture (ICOSICA)*
- Pötter L. 2014. Sentinel 1-A Menatap Bumi. Tersedia dari: <http://www.dw.com/id/-1-amenatap-bumi/g-17629624>
- Pressman RS, Maxim. 2014. *Software Engineering: A Practitioner's Approach*, Seventh Edition. New York (US): McGraw-Hill.
- Putri DR, Sukmono A, Sudarsono B. 2018. Analisis Kombinasi Citra Sentinel -1A dan Citra Sentinel -2A untuk Klasifikasi Tutupan Lahan (Studi Kasus: Kabupaten Demak, Jawa Tengah). *Jurnal Geodesi UNDIP*, 7(2): 85-96.
- Shyam A, Mukesh N. 2020. A Django Based Educational Resource Sharing Website: *Shreic. Journal of Scientific Research*. 64(1): 138-152.
- Tamrakar A, Daga M, Lunia A. 2019. *Illustration on the Superiority of Django Over The Other Python Web Frameworks*. *Journal of The Gujarat Research Society*. 21(17): 660-668
- Wirandha FS, Marwan, Nizamuddin. 2015. Klasifikasi Penggunaan Lahan Menggunakan Citra Satelit SPOT-6 di Kabupaten Aceh Barat Daya dan Aceh Besar. Seminar Nasional dan Expo Teknik Elektro.