

SENDING DRONE FLYING SPEED DATA USING ANDROID-BASED CLIENT SERVER METHOD

Daewu Gus Bintara Putra¹, Anggraini Kusumaningrum²
^{1,2)} Program Studi Teknik Informatika, Departemen Informatika,
Sekolah Tinggi Teknologi Adisutjipto
Jl. Janti, Blok R Lanud Adisutjipto Yogyakarta
Email : ¹daewu.bintara1996@gmail.com, ²anggraini@stta.ac.id

Abstract

Abstract - Drones as Unmanned Aerial Vehicle that are controlled using remote control are experiencing better development. The speed of the drone that cannot be accessed on the remote control can be determined through a client-server based application. The client side application, named Mydrone, is placed on the drone while on the server side, named Admin Drone, on earth. Mydrone and Admin Drone are applications made in this research based on android. This application on the client and server is used to monitor the speed of flying drones by using the Google Play Service library. Data sent from the client to the server in realtime by using Google Firebase. From the tests carried out it can be seen the speed of the drone when flying horizontal ly, while when flying vertically the speed is not detected. Battery capacity will also decrease when the speed of the drone gets faster, this is seen in the data when the speed is 8,741 km / h, the electricity demand taken from the battery is 31%.

Keyword: Drones, Google Play Service, Flying Speed.

1. Latar Belakang Masalah

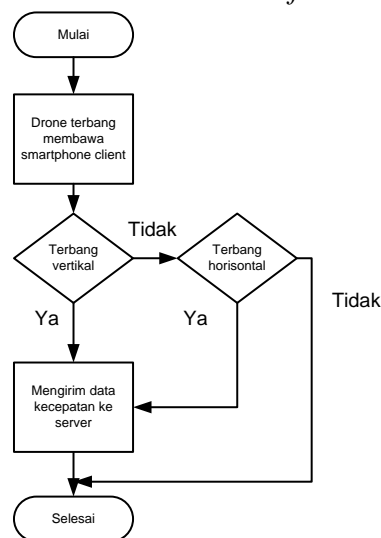
Kendaraan udara tak berawak sudah banyak dibuat oleh pabrikan dari luar negeri dan dalam negeri mengalami perkembangan yang cukup baik dengan penambahan *feature* dari produk ke produk lainnya. Kendaraan udara tak berawak juga dikenal dengan nama as *Unmanned Aerial Vehicle* (UAV), ada juga yang memberi nama sebagai *drone*, dan *quad copter* yang terbang dengan kendali terbanyak menggunakan *remote control*. *Drone* dapat membawa beban saat terbang, beban yang dibawa ini sesuai dengan kebutuhan yang diinginkan oleh pengendali *drone*, salah satunya membawa *smartphone* berbasis android untuk mengambil gambar[1] dan mendata posisi koordinat terbang[2] menggunakan metode *client server* dengan aplikasi sisi *client* menggunakan *smartphone* dan sisi *server* menggunakan komputer jinjing. *Drone* yang diterbangkan secara *remote* memiliki jarak terbang dapat dikendalikan[3] dan kecepatan terbangnya juga dapat dikendalikan[4]. Kecepatan *drone* sama halnya kecepatan kendaraan yang berjalan didarat. Kecepatan sebuah kendaraan dapat diukur dan dipantau menggunakan OpenCV[5], *real time* dengan android[6], *instant messaging* di android[7], pengolahan video[8], *sensor magnetic*[9], *motion vector*[10] dan dapat juga menggunakan perpaduan antara teknik rektifikasi citra dan tapis kalman[11].

Kecepatan terbang *drone* saat terbang horisontal atau vertikal dibaca menggunakan layanan yang disediakan pada *Google Plays Service*. *Tools getSpeed* adalah sebuah fungsi yang disediakan oleh Android dan dibuat oleh perusahaan Google yang mana berfungsi untuk mendapatkan kecepatan *device* yang menggunakan fungsi tersebut. *Tools location* merupakan fungsi yang disediakan oleh Google untuk mendapatkan letak akurasi geografis *longitude* dan *latitude* (Google Developers). Kecepatan yang didapatkan dengan cara seperti itu, dikirimkan dari sisi *client* yang dibawa terbang *drone* ke sisi *server* yang ada di bumi. Metode *client server* antara dua peralatan dengan memanfaatkan perangkat *smartphone* berbasis

android dengan memanfaatkan jaringan *wifi*[12], pada penelitian ini hanya untuk *one to one* antara *drone* dengan kendali di bumi menggunakan jaringan internet, dan data yang dikirimkan memanfaatkan fasilitas yang ada pada *firebase*[13][14]. Pada penelitian ini, kecepatan terbang *drone* digunakan untuk mengetahui seberapa besar pengaruh kecepatan terbang *drone* terhadap kapasitas baterai, sehingga proses pembuatan *drone* rakitan bisa lebih maksimal.

2. Metodologi Penelitian

Metode penelitian yang digunakan dan dilakukan dalam penelitian ini, seperti terlihat pada Gambar 1, dengan merancang perangkat lunak terlebih dahulu. Perangkat lunak ini ada dua jenis, sisi *client* dan sisi *server*. Perangkat lunak sisi *client* ditempatkan ada *smartphone* berbasis android yang dibawa terbang oleh *drone*, sedangkan perangkat lunak sisi *server* ditempatkan pada *smartphone* yang ada di bumi, Setelah perangkat lunak selesai dirancang dan dibuat, selanjutnya dilakukan pengujian untuk mendapatkan data kecepatan dan mengirimkannya dari *client* ke *server* memanfaatkan *firebase*.

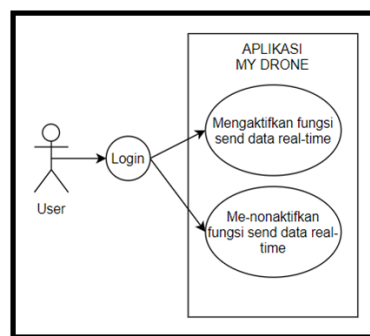


Gambar 1. Metode penelitian yang dilakukan

A. Use case diagram

Pada perancangan *use case diagram* aplikasi terdapat dua desain *use case diagram*, yaitu perancangan untuk aplikasi Sisi Client dan aplikasi Sisi Server.

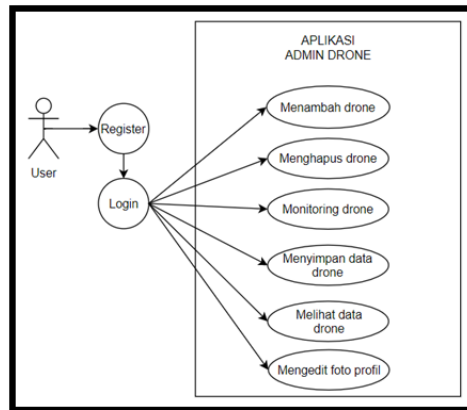
1. *Use case diagram* aplikasi Sisi Client. lihat Gambar 2, dijelaskan bahwa user bisa mengaktifkan fungsi *send data real-time* dan menonaktifkannya setelah user berhasil *login* pada aplikasi.



Gambar 2. Use Case Diagram aplikasi Sisi Client

2. *Use case diagram* aplikasi Sisi Server, lihat Gambar 3, user bisa membuat data *drone* baru, menghapus data *drone*, monitoring *drone* yang telah diaktifkan fungsi *send data*

real-time nya, menyimpan data *monitoring*, melihat data *drone* dan mengubah foto profil. Semua fungsi tersebut bisa dilakukan apabila *user* telah terdaftar dan berhasil *login* aplikasi.

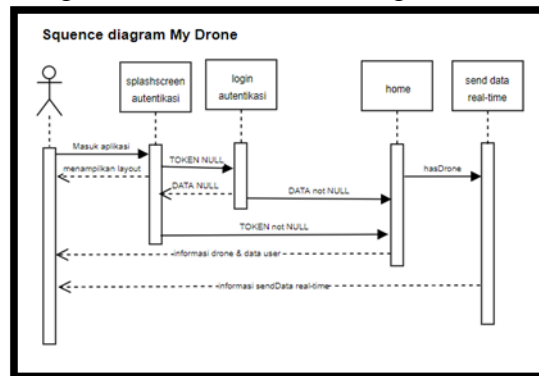


Gambar 3. Use Case Aplikasi Sisi Server

B. Sequence diagram

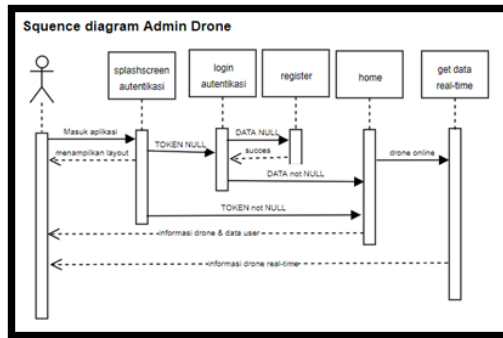
Pada *sequence diagram* aplikasi terdapat dua *sequence diagram*, yaitu *sequence diagram* untuk aplikasi Sisi Client dan aplikasi Sisi Server.

1. *Sequence diagram* sisi Client, lihat Gambar 4, *user* saat pertama kali menjalankannya akan muncul tampilan splash dan dilanjutkan ke tampilan *login* untuk memasukan dengan memasukan *email*, kemudian akan masuk ke bagian utama dan pengiriman data kecepatan, sehingga saat dibawa terbang akan membaca dan mengirmkan data ke sisi *server*.



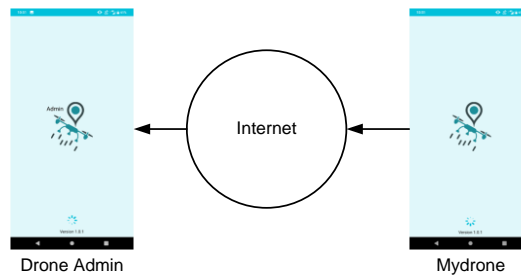
Gambar 4. Sequence Diagram Sisi Client

2. *Sequence diagram* sisi Server, lihat Gambar 5, *user* yang mengaktifkan aplikasi sisi *server* yang ada pada *smartphone* yang tidak dibawa terbang *drone* akan masuk pada tampilan *splash* dan *login*, hamper sama dengan sisi *client*, jika belum *register* diwajibkan untuk *register* terlebih dahulu. Jika sudah melewati *login* maka tampilan selanjutnya adalah data yang ditampilkan dari proses pengiriman oleh sisi *client*.



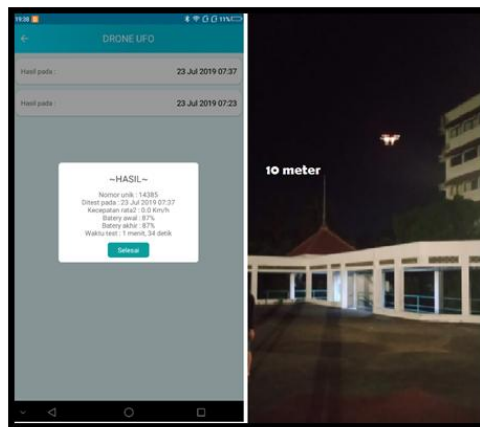
Gambar 5. *Sequence Diagram Sisi Server*

3. Hasil dan Pembahasan



Gambar 6. Hubungan *Client Server* antar aplikasi

Tidak seperti penelitian sebelumnya, dimana aplikasi sisi *client* berupa smartphone dan sisi server berupa komputer[1][2], pada penelitian ini dihasilkan perangkat lunak *client-server* yang dipasangkan pada dua buah *smartphone*. Setelah perancangan dilakukan terhadap aplikasi sisi *client* atau sisi *server* seperti terlihat pada Gambar 6, selanjutnya pengujian untuk *drone* terbang naik turun atau terbang vertikal, *drone* terbang maju atau terbang horisontal dan perbandingan terbang menggunakan *drone*, dan dibawa menggunakan sepeda motor.

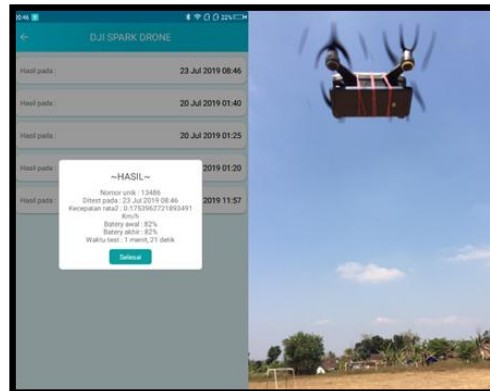


Gambar 7. Percobaan *drone* vertikal naik 10 meter

Pada Gambar 7, terlihat *drone* pada ketinggian kurang lebih 5 meter dengan acuan antenna pencakar langit gedung seperti pada gambar dimana ketinggian antenna tersebut 10 meter. Sehingga dapat ditentukan ketinggian *drone* terbang dengan menerbangkan *drone* setinggi antenna pencakar langit. Didapat hasil kecepatan terbang rata-rata 0.0 km/h dalam waktu terbang 1 menit 33 detik. Hal ini menunjukkan bahwa saat *drone* terbang vertikal, Mydrone tidak mampu membaca kecepatan dengan menggunakan fungsi *getspeed* yang

tersedia di Google Play Service. Sehingga data kecepatan yang dikirimkan dari *client* ke *server* 0.0 km/h. Sedangkan data kapasitas baterai drone berkurang dan sukses dikirimkan secara *client-server*. Sehingga untuk penelitian selanjutnya, bisa di selesaikan dengan menggunakan persamaan mencari kecepatan dimana kecepatan dihasilkan dari jarak/ waktu, dengan terlebih dahulu melakukan pengecekan apakah *drone* tersebut terbang secara vertical atau horizontal. Untuk mengetahui apakah *drone* tersebut terbang secara vertical atau horizontal dengan menggunakan data perpindahan koordinat *longitude* dan *laltitude drone*.

Pengujian selanjutnya *drone* terbang secara horisontal, dengan perbedaan kecepatan terbang dari kecepatan rendah hingga kecepatan tinggi dengan *drone* DJI SPARK seperti terlihat pada Gambar 8.



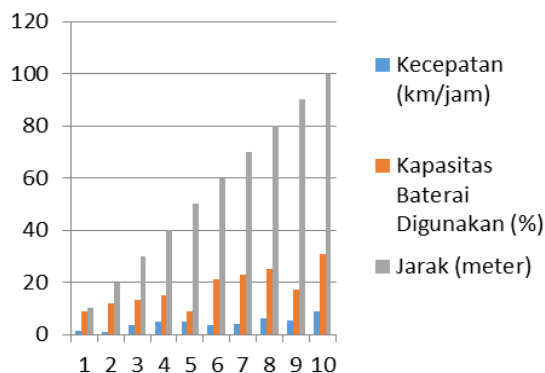
Gambar 8. Drone terbang secara horisontal.

Pada saat *drone* terbang membawa muatan *smartphone* yang didalamnya ada aplikasi Mydrone, diuji dengan jarak terbang yang berbeda-beda dan kecepatan juga berbeda, ini dilakukan untuk melihat pengaruh kecepatan terhadap kapasitas penyimpanan baterai yang ada pada *drone*. Hasil pengujian ini selengkapny dapat dilihat pada tabel 1.

Tabel 1. Hasil pengujian terbang horisontal

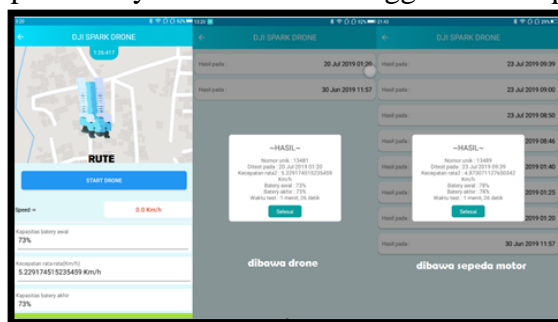
No	Jarak Terbang Horisontal (meter)	Kecepatan (km/jam)	Kapasitas Baterai <i>Drone</i> yang Digunakan (%)
1.	10	1.424	9
2.	20	0.873	12
3.	30	3.669	13
4.	40	5.013	15
5.	50	4.818	9
6.	60	3.670	21
7.	70	4.183	23
8.	80	6.098	25
9.	90	5.082	17
10.	100	8.741	31

Kecepatan terbang drone pada jarak terbang yang telah ditentukan tidak dilakukan secara konstan, tetapi dilakukan secara acak. Pada jarak terjauh, 100 meter, kecepatan paling tinggi, 8,741 km/jam, membutuhkan daya baterai untuk terbang sebesar 31%. Hasil pengujian terbang horisontal dalam bentuk grafik dapat dilihat pada gambar 9.



Gambar 9. Grafik hubungan jarak, kecepatan, dan kapasitas baterai dalam pengujian terbang horisontal

Pengujian selanjutnya membandingkan aplikasi Mydrone yang ada pada *smartphone* saat dibawa oleh *drone* dan menggunakan sepeda motor. Hal ini dilakukan untuk menguji pengiriman data yang dilakukan oleh Mydrone ke Admin *drone* melalui jaringan internet seperti pada Gambar 9. Dengan rute yang sama saat terbang menggunakan *drone*, *smartphone* yang berisi aplikasi Mydrone dibawa menggunakan sepeda motor.



Gambar 10. Perbandingan aplikasi dibawa motor dengan dibawa *drone*

Hasil pengujian yang dilakukan seperti pada Gambar 10 dapat dilihat pada Tabel 2. Pada Tabel 3 dapat dilihat bahwa data mampu dikirimkan dari Mydrone ke Admin *drone* dengan metode *client-server* dengan cara dibawa oleh *drone* atau menggunakan sepeda motor. Ini terjadi karena adanya koneksi internet yang menghubungkan antara dua perangkat *smartphone* yang terhubung secara *client-server*.

Tabel 2. Percobaan Pengiriman Data ke Server

No	Cara membawa aplikasi	Data Berhasil Dikirim dari Client Ke Server
1.	Sepeda/ Sepeda Motor	✓
2.	<i>Drone</i>	✓

Keterangan :

1. ✓ : Berhasil
2. X : Gagal

Hasil pengujian dengan menggunakan sepeda motor sebagai kendaraan yang membawa aplikasi Mydrone memiliki kesamaan data dengan penelitian sebelumnya tentang pengukuran kecepatan kendaraan[6]. Perbedaannya terletak pada cara melihat datanya, pada penelitian ini data yang didapat pada sisi *client* dikirimkan ke sisi *server* menggunakan jaringan internet

sehingga kinerjanya sangat tergantung dengan keberadaan sinyal yang disediakan oleh provider telekomunikasi yang digunakan pada sisi *client* dan sisi *server*.

4. Kesimpulan

Berdasarkan hasil pengujian dan pembahasan yang dilakukan dapat diambil kesimpulan bahwa layanan yang ada pada *Google Play Service* yaitu *getspeed()* dapat digunakan untuk mengukur kecepatan baik diudara menggunakan *drone* atau didarat menggunakan sepeda motor yang bergerak secara horisontal dengan menggunakan metode *client server* memanfaatkan *firebase* data dapat dikirimkan. Selain itu kecepatan terbang *drone* memiliki pengaruh terhadap kapasitas baterai dari *drone*, sehingga perlu kajian lebih mendalam lagi terkait kapasitas baterai jika ingin sebuah *drone* terbang dengan kecepatan tinggi.

Daftar Pustaka

- [1] Wintolo, H., Sudaryanto, S., & Pramudito, C. G. (2018). Remote Camera For Android Based Smartphones Installed On The Syma X8HG Drone. *Angkasa: Jurnal Ilmiah Bidang Teknologi*, 10(2), 130-140.
- [2] Wintolo, H., Kusumaningrum, A., & Aditya, R. (2019). Pengiriman Data Koordinat Global Position System (GPS) Pada Drone Dengan Memanfaatkan Jaringan Internet. *Simetris: Jurnal Teknik Mesin, Elektro dan Ilmu Komputer*, 10(1), 141-146.
- [3] Basukesti, A., & Indrianingsih, Y. (2014). Sistem Pengendalian Jarak Terbang Pada Pesawat Quadcopter Untuk Menghindari Loss of Control. *Teknoin*, 2
- [4] Hanafi, N., Miftahudin, A. M. I., Akbar, C., Ali, N., & Dewanto, R. S. (2018). Pengendalian Kecepatan Gerak Quadcopter Berbasis IMU pada Gerakan Tangan dengan Metode Template Matching. *Jurnal Rekayasa Elektrika*, 14(3).
- [5] Andrew, A., Buliali, J. L., & Wijaya, A. Y. (2017). Deteksi Kecepatan Kendaraan Berjalan di Jalan Menggunakan OpenCV. *Jurnal Teknik ITS*, 6(2), A415-A420.
- [6] Gustina, R., Setiawan, W., & ER, N. I. Pengukuran Kecepatan Kendaraan Secara Realtime Berbasis Android. *Majalah Ilmiah Teknologi Elektro*, 15(1), 91-94.
- [7] Prasetya, K. R. A., Setiawan, W., & Djuni, I. K. D. Visualisasi Kecepatan Kendaraan Menggunakan Instant Messaging Berbasis Android. *Majalah Ilmiah Teknologi Elektro*, 15(2), 1-6.
- [8] Sadewo, S. S., Sumiharto, R., & Candradewi, I. (2015). Sistem Pengukur Kecepatan Kendaraan Berbasis Pengolahan Video. *IJEIS (Indonesian Journal of Electronics and Instrumentation Systems)*, 5(2), 177-186.
- [9] Ramdhani, A. S. (2016). *Rancang Bangun Sistem Pengukur Kecepatan Kendaraan Menggunakan Sensor Magnetik* (Doctoral dissertation, Universitas Pendidikan Indonesia).
- [10] Sukabhakti, Y. M. N., & Sulistyaningrum, D. R. (2019). Estimasi Kecepatan Kendaraan Bergerak Menggunakan Motion Vector. *Jurnal Sains dan Seni ITS*, 8(1), 7-11.
- [11] Gusa, R. F. (2014). Teknik Rektifikasi Citra dan Tapis Kalman Dalam Mengestimasi Kecepatan Kendaraan. *Jurnal Nasional Teknik Elektro*, 3(1), 8-17.
- [12] Ramadan, D. N., Permana, A. G., & Hafidudin, H. (2017). Perancangan Dan Realisasi Mobil Remote Control Menggunakan Firebase. *Jurnal Elektro dan Telekomunikasi Terapan*, 4(1), 505-505.
- [13] Irawan, H. (2017). *Rancang Bangun Wireless Sensor Network Pada Pendeteksi Dini Potensi Kebakaran Lahan Gambut Menggunakan Banana Pi IoT* (Doctoral dissertation, Institut Teknologi Sepuluh Nopember).
- [14] Kartikadarma, E., Yutriatmansyah, W. W., Udayanti, E. D., & Hafidhoh, N. U.

(2019). Implementasi Firebase Cloud Messaging Pada Emergency Call Berbasis Android. *Simetris: Jurnal Teknik Mesin, Elektro dan Ilmu Komputer*, 10(1), 83-90.