

PENGEMBANGAN APLIKASI *GROUND CONTROL STATION* (GCS) UNTUK PENGAWASAN DAN PENGENDALIAN UAV

(Skripsi)

Oleh

**TAZKIA KARIMA HERLI EFISON
1515061034**



**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS LAMPUNG
2022**

ABSTRAK

PENGEMBANGAN APLIKASI *GROUND CONTROL STATION* (GCS) UNTUK PENGAWASAN DAN PENGENDALIAN UAV

Oleh

TAZKIA KARIMA HERLI EFISON

Aplikasi *Mission Planner* merupakan salah satu aplikasi *Ground Control Station* yang bersifat *open-source*. Aplikasi *Mission Planner* dibangun oleh Michael Osborne dan menjadi aplikasi GCS yang paling umum digunakan. Aplikasi *Mission Planner* digunakan untuk mengontrol serta memonitoring misi penerbangan. Aplikasi ini juga dapat menerima serta mengirim perintah data penerbangan. Dalam melakukan misi penerbangan, aplikasi *Mission Planner* dioperasikan oleh operator GCS. Dalam mengoperasikan *Mission Planner*, akan lebih baik jika tampilan aplikasi disesuaikan dengan kebutuhan operator. Oleh karena itu, dilakukan pengembangan tampilan aplikasi *Mission Planner* yang sesuai dengan kebutuhan. Sehingga diharapkan dapat mempermudah operator GCS dalam mengoperasikan penerbangan. Penelitian ini mengembangkan tampilan dari aplikasi *Mission Planner* berbasis desktop. Pengembangan dilakukan dengan menggunakan *Microsoft Visual Studio Community 2019* dengan bahasa pemrograman C#. Penelitian ini diuji coba dengan menggunakan fasilitas dari Laboratorium Teknik Digital.

Kata Kunci: *Unmanned Aerial Vehicle (UAV)*, *Ground Control Station (GCS)*, *Mission Planner*

ABSTRACT

DEVELOPMENT OF GROUND CONTROL STATION (GCS) APPLICATION FOR UAV MONITORING AND CONTROL

By

TAZKIA KARIMA HERLI EFISON

Mission Planner is an open-source software of Ground Control Station. The Mission Planner was built by Michael Osborne and is the most commonly used GCS software. This software is used to control and monitor flight missions. This software also able to receive and send flight data commands. In carrying out flight missions, the Mission Planner application is operated by the GCS operator. In operating Mission Planner, it would be better if the appearance of this software is adjusted to the needs of the operator. Therefore, the Mission Planner application display was developed according to the needs. Hopefully, this development will make it easier for GCS operators to operate flights. This study develops the appearance of the desktop-based Mission Planner software. The development is carried out using Microsoft Visual Studio Community 2019 with the C# programming language. This research was tested using facilities from the Digital Engineering Laboratory.

Keywords: Unmanned Aerial Vehicle (UAV), Ground Control Station (GCS), Mission Planner

**PENGEMBANGAN APLIKASI *GROUND CONTROL STATION (GCS)*
UNTUK PENGAWASAN DAN PENGENDALIAN *UAV***

Oleh

TAZKIA KARIMA HERLI EFISON

Skripsi

**Sebagai Salah Satu Syarat untuk Mencapai Gelar
SARJANA TEKNIK**

Pada

**Program Studi Teknik Informatika
Jurusan Teknik Elektro
Fakultas Teknik Universitas Lampung**



**JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNIK
UNIVERSITAS LAMPUNG
2022**

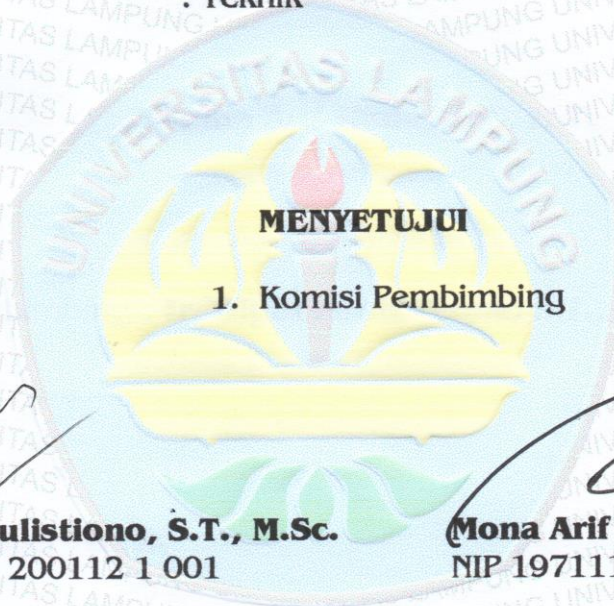
Judul Skripsi : **PENGEMBANGAN APLIKASI GROUND CONTROL STATION (GCS) UNTUK PENGAWASAN DAN PENGENDALIAN UAV**

Nama Mahasiswa : **Tazkia Karima Herli Efison**

Nomor Pokok Mahasiswa : **1515061034**

Jurusan : **Teknik Elektro**

Fakultas : **Teknik**



1. Komisi Pembimbing

Wahyu Eko Sulistiono, S.T., M.Sc.
NIP 19741201 200112 1 001

Mona Arif Muda, S.T., M.T.
NIP 19711112 200003 1 002

2. Mengetahui

**Ketua Jurusan
Teknik Elektro**

Herlinawati, S.T., M.T.
NIP 19710314 199903 2 001

**Ketua Program Studi S1
Teknik Informatika**

Mona Arif Muda, S.T., M.T.
NIP 19711112 200003 1 002

MENGESAHKAN

1. Tim Penguji

Ketua

: Wahyu Eko Sulistiono, S.T., M.Sc.

Sekretaris

: Mona Arif Muda, S.T., M.T.

Penguji

Bukan Pembimbing : Ir. Gigih Forda Nama, S.T., M.T.I., IPM.

2. Dekan Fakultas Teknik

Dr. Eng. Ir. Helmy Fitriawan, S.T., M.Sc.

NIP 19750928 200112 1 002

Tanggal Lulus Ujian Skripsi : 15 Agustus 2022

SURAT PERNYATAAN

Saya yang bertandatangan di bawah ini, menyatakan bahwa skripsi yang saya buat yang berjudul "*Pengembangan Aplikasi Ground Control Station (GCS) Untuk Pengawasan Dan Pengendalian UAV*" Merupakan hasil karya sendiri dan bukan hasil karya orang lain. Semua hasil yang tertuang dalam skripsi ini telah mengikuti kaidah penulisan karya ilmiah Universitas Lampung. Apabila di kemudian hari terbukti bahwa skripsi ini adalah merupakan salinan atau dibuat oleh orang lain, maka saya bersedia menerima sanksi sesuai dengan ketentuan akademik yang sedang berlaku.

Bandar Lampung, 22 September 2022

Yang membuat pernyataan,



Tazkia Karima Herli Efison

NPM. 1515061034

RIWAYAT HIDUP



Penulis bernama Tazkia Karima Herli Efison, dilahirkan di Denpasar pada tanggal 31 Maret 1997.

Penulis merupakan anak pertama dari 7 bersaudara dari pasangan Herli Efison dan Jamila. Penulis menempuh pendidikan formal dan lulus pada tahun

2009 di SD Muhammadiyah 1 Denpasar, kemudian lulus pada tahun 2012 di SMP PGRI 1 Denpasar, kemudian lulus pada tahun 2015 di SMA Al – Azhar 3 Bandar Lampung. Di Tahun 2015 penulis mengikuti seleksi SBMPTN dan diterima di Program Studi S1 Teknik Informatika Universitas Lampung. Selama menjadi mahasiswa penulis mengikuti organisasi tingkat jurusan, yaitu Himpunan Mahasiswa Teknik Elektro (HIMATRO) Fakultas Teknik sebagai anggota Divisi Kerohanian Periode 2015-2016 dan Divisi Media Informasi pada Departemen Kominfo Periode 2016-2017. Penulis juga mengikuti Unila Robotika dan Otomasi (URO) sebagai anggota yang aktif pada bagian Divisi *Vertical Take Off and Landing* (VTOL). Pada tahun 2018, penulis melakukan Kerja Praktik (KP) di PT Bandung Technology Zone (BTZ) Bandung dan membahas mengenai “Perancangan Interface Pada Aplikasi Faktur Generator Berbasis Web Dengan HTML dan CSS Menggunakan Framework CodeIgniter”.

Persembahan



Dengan Mengharapkan Ridho dari Allah SWT

Teriring sholawat kepada Nabi Muhammad SAW dan keluarganya

Karya Tulis ini kupersembahkan untuk:

Ayah dan Ibu

Herli Efison & Jamila

Saudara dan Saudariku

M. Aditya Barelvi H.E. & Adik - adikku

Para Sahabatku

EIE 2015 & Semua Orang Yang Telah Memberi Dukungan

Almamaterku

Universitas Lampung

MOTTO

“Tidak Ada Kata Terlambat”

***“Meski Besok Kiamat, Tetaplah Menanam
Pohon”***

SANWACANA

Bismillahirrahmanirrahim

Puji syukur penulis panjatkan ke hadirat Allah SWT karena atas segala nikmat dan rahmat-Nya. Sholawat serta salam kita agungkan ke hadirat Nabi Muhammad SAW dan keluarganya. Dengan segala karunia – Nya, penulis dapat menyelesaikan tugas akhir ini.

Tugas akhir dan skripsi dengan judul “**Pengembangan Aplikasi *Ground Control Station* (GCS) Untuk Pengawasan dan Pengendalian UAV**” ini merupakan salah satu syarat untuk memperoleh gelar sarjana Teknik di Jurusan Teknik Elektro Fakultas Teknik Universitas Lampung.

Penulis mendapatkan banyak bantuan, bimbingan, arahan, dan dukungan dalam menyelesaikan alat dan tugas akhir serta penyusunan skripsi ini. Karena itu penulis menyampaikan rasa hormat dan terima kasih kepada:

1. Allah SWT atas rahmat-Nya sehingga penulis dapat menyelesaikan tugas akhir dan skripsi ini.
2. Bapak Dr. Eng. Ir. Helmy Fitriawan, S.T., M.Sc. selaku Dekan Fakultas Teknik Universitas Lampung.

3. Ibu Herlinawati, S.T., M.T. selaku Ketua Jurusan Teknik Elektro Fakultas Teknik Universitas Lampung sekaligus pembimbing utama, terima kasih atas waktu dan kesediaannya dalam membimbing tugas akhir dan skripsi.
4. Bapak Ir. Meizano Ardhi Muhammad, S.T., M.T. selaku Sekretaris Jurusan Teknik Elektro Fakultas Teknik Universitas Lampung sekaligus Pembimbing Akademik, terima kasih atas waktu dan kesediaannya dalam membimbing sejak awal perkuliahan.
5. Mona Arif Muda Batubara, S.T., M.T. selaku Ketua Program Studi Teknik Informatika Fakultas Teknik Universitas Lampung sekaligus Dosen Pembimbing Pendamping, terima kasih atas waktu dan kesediaannya dalam membimbing tugas akhir dan skripsi, serta sebagai Pembimbing dan Pembina kegiatan URO.
6. Wahyu Eko Sulistiono, S.T., M.Sc. selaku Dosen Pembimbing Utama, terima kasih atas waktu dan kesediaannya dalam membimbing tugas akhir dan skripsi.
7. Gigih Forda Nama, S.T., M.T., selaku Dosen Penguji Utama, terima kasih atas waktu dan kesediaannya dalam membimbing tugas akhir dan skripsi.
8. Seluruh Dosen Teknik Elektro Universitas Lampung, Terima kasih atas bimbingan dan ilmu yang telah diberikan selama menuntut ilmu di Jurusan Teknik Elektro Universitas Lampung.
9. Kedua orang tua, Herli Efison dan Jamila serta saudaraku M. Aditya Barelvi H.E. dan adik-adikku yang telah membantu dan menyemangati selama perkuliahan hingga pengerjaan skripsi ini.
10. Aris Susilo, S.T. selaku Senior Supervisor URO, terima kasih atas waktu dan kesediaannya dalam membantu proses penyelesaian tugas akhir dan skripsi.

11. Nur Fadillah, Anggi Saputra, Rekhy Zakaria Pasaribu, Andrew, dan M. Fattah yang telah membantu dalam proses pengujian tampilan aplikasi sekaligus rekan satu angkatan di URO dan Lab Teknik Digital, Fakultas Teknik, Universitas Lampung.
12. Irham Riza Maulana, Heni Novita Sari, Era Desti Ramayani, Nur Fadillah, I Wayan Razpayana, Sri Lestari, Ayu Nafadila, Pariyem, yang telah membantu serta menyemangati dalam proses penyelesaian tugas akhir dan skripsi.
13. Teman – teman Asisten dan Staff Laboratorium Teknik Digital Jurusan Teknik Elektro Universitas Lampung yang telah memberikan semangat dan bantuan kepada penulis.
14. Keluargaku Electrical & Informatic Engineering (EIE) Angkatan 2015. Terima kasih atas kebersamaan dan kekeluargaan selama perkuliahan yang telah kalian berikan, semoga kita selalu bersatu semakin erat.
15. Semua pihak yang secara langsung maupun tidak langsung membantu hingga terselesaikan laporan skripsi ini.

Semoga Allah SWT membalas semua kebaikan semua pihak yang telah membantu dalam penyelesaian tugas akhir dan skripsi ini.

Bandar Lampung, 22 September 2022

Penulis,

Tazkia Karima Herli Efison

DAFTAR ISI

Halaman

HALAMAN JUDUL.....	i
DAFTAR ISI.....	xiv
DAFTAR GAMBAR	xvii
DAFTAR TABEL.....	xix
I. PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Tujuan Penelitian.....	3
1.4 Batasan Masalah.....	3
1.5 Hipotesis Penelitian.....	3
1.6 Manfaat Penelitian.....	4
1.7 Sistematika Penulisan.....	4
II. TINJAUAN PUSTAKA.....	6
2.1 Penelitian Terkait	6
2.2 Landasan Teori.....	11
2.2.1 Aplikasi Berbasis Desktop.....	11
2.2.2 Bahasa Pemrograman C#	12
2.2.3 Microsoft Visual Studio	12
2.2.4 <i>Ground Control Station (GCS)</i>	13
2.2.5 <i>Mission Planner</i>	15

2.2.6	<i>Unmanned Aerial Vehicle (UAV)</i>	16
2.2.7	<i>Flight Controller</i>	16
2.2.8	3DR Pixhawk Mini	17
2.2.9	<i>Module Telemetry</i>	18
2.2.10	<i>Global Positioning System (GPS)</i>	19
2.2.11	<i>ADAHRS (Air Data, Attitude, and Heading Reference System)</i>	19
2.2.12	GMap.NET	20
2.2.13	Use Case Diagram	20
2.2.14	Arsitektur Aplikasi <i>Mission Planner</i>	22
III. METODOLOGI PENELITIAN.....		25
3.1	Waktu dan Tempat	25
3.2	Alat dan Bahan.....	25
3.2.1	Komponen Elektronika	25
3.2.2	Komponen Pendukung.....	26
3.3	Tahapan Penelitian Metode Prototype	26
IV. HASIL DAN PEMBAHASAN		30
4.1	<i>Requirement and Analysis (Analisa Kebutuhan)</i>	30
4.1.1	Analisa Kebutuhan	31
4.1.2	Use Case Diagram	33
4.1.3	Use Case Scenario	34
4.1.4	Mengembangkan <i>Mission Planner</i> dengan <i>Visual Studio</i>	39
4.1.5	<i>System Requirements</i>	40
4.1.6	Instalasi Aplikasi	40
4.1.7	Implementasi <i>Build Mission Planner</i> pada <i>Visual Studio</i>	43
4.2	<i>Quick Design and Modelling of Quick Design</i>	46
4.2.1	<i>Icon / Logo Aplikasi</i>	46
4.2.2	Halaman Splash Screen.....	47
4.2.3	Halaman Flight Data	47
4.2.4	Halaman Flight Planner	49
4.3	<i>Construction of Prototype (Code / Debug & Integration)</i>	51

4.3.1	<i>Sourcecode</i> Menampilkan <i>IconFile</i>	51
4.3.2	<i>Sourcecode</i> Menampilkan Row Header Table Planner	52
4.3.3	<i>Sourcecode</i> Menampilkan Split Panel Status	52
4.3.4	Error .NETSDK 1	53
4.4	Deployment, Delivery, & Customer Feedback	53
4.4.1	Pengujian <i>Blackbox</i>	54
4.4.2	Pengujian <i>User Experience Questionnaire (UEQ)</i>	56
4.4.3	Responden Kuisisioner	58
4.4.4	Hasil Dari Metode Prototyping	58
4.4.5	Hasil Perhitungan UEQ	60
V.	KESIMPULAN	67
5.1	Kesimpulan	67
5.2	Saran	68
	DAFTAR PUSTAKA	69
	LAMPIRAN	73

DAFTAR GAMBAR

Gambar 2.1	<i>Ground Control Station</i>	15
Gambar 2.2	3DR Pixhawk Mini	17
Gambar 2.3	Bagian Use Case Diagram	21
Gambar 2.4	Basic Structure Aplikasi.....	22
Gambar 3.1	Metode Prototype	27
Gambar 4.1	<i>Usecase Diagram</i> (Halaman Flight Data).....	33
Gambar 4.2	<i>Usecase Diagram</i> (Halaman Flight Planner).....	33
Gambar 4.3	.NET Desktop Development	40
Gambar 4.4	ASP.NET and Web Development.....	41
Gambar 4.5	Universal Windows Platform Development	41
Gambar 4.6	Mobile Development with .NET.....	42
Gambar 4.7	.NET Core Cross-Platform Development	42
Gambar 4.8	Clone Git	43
Gambar 4.9	Update Submodule	43
Gambar 4.10	Project MissionPlanner.sln.....	44
Gambar 4.11	Build Solution	44
Gambar 4.12	Breakpoint.....	45
Gambar 4.13	Logo Aplikasi.....	46
Gambar 4.14	Splash Screen	46
Gambar 4.15	Halaman Flight Data	47
Gambar 4.16	Hasil Implementasi Flight Data	48
Gambar 4.17	Halaman Flight Planner	49
Gambar 4.18	Hasil Implementasi Flight Planner.....	50
Gambar 4.19	<i>Sourcecode</i> menampilkan icon file	51
Gambar 4.20	Sourcecode menampilkan row header table planner.....	51

Gambar 4.21	Sourcecode menampilkan split panel status.....	51
Gambar 4.22	<i>Error</i> NETSDK 1	52
Gambar 4.23	Hasil Implementasi Flight Planner.....	50
Gambar 4.24	UEQ Yang Digunakan	57
Gambar 4.25	Tabel Input Data Pengujian.....	60
Gambar 4.26	Tabel Tranformasi Data Pengujian UEQ.....	61
Gambar 4.27	Gambar Hasil Perhitungan Mean Data	62
Gambar 4.28	Gambar Hasil Perhitungan <i>Mean</i> Setiap Kategori.....	62
Gambar 4.29	Gambar Hasil Perhitungan Mean Dari Setiap Poin	63
Gambar 4.30	Hasil Mean Masing-masing Pertanyaan	64
Gambar 4.31	Perbandingan Dengan Benchmark Data Set	64
Gambar 4.32	Struktur Skala Penilaian UEQ.....	65

DAFTAR TABEL

Tabel 2.1	Spesifikasi 3DR Pixhawk Mini	17
Tabel 3.1	Komponen Elektronika.....	25
Tabel 3.2	Komponen Pendukung	26
Tabel 4.1	<i>Project Mission Planner</i>	30
Tabel 4.2	Kebutuhan Tampilan pada Aplikasi	32
Tabel 4.3	Connect UAV	34
Tabel 4.4	Melihat Status Data UAV.....	35
Tabel 4.5	Mengakses Seluruh Menu Pada Aplikasi	36
Tabel 4.6	Membuat Misi Penerbangan.....	36
Tabel 4.7	Melakukan Setting Awal UAV	37
Tabel 4.8	Mengatur Parameter Secara Lengkap (Sebelum terbang)	38
Tabel 4.9	Disconnect UAV	38
Tabel 4.10	Penjabaran Status Fitur pada Halaman <i>Flight Data</i>	48
Tabel 4.11	Penjabaran Fitur pada Halaman <i>Flight Planner</i>	50
Tabel 4.12	<i>Blackbox Testing</i> Akses Aplikasi	54
Tabel 4.13	<i>Blackbox Testing Connect</i> Aplikasi.....	54
Tabel 4.14	<i>Blackbox Testing</i> Akses Halaman <i>Flight Data</i>	55
Tabel 4.15	<i>Blackbox Testing</i> Akses Tiap <i>Menu</i>	55
Tabel 4.16	<i>Blackbox Testing</i> Akses Halaman <i>Flight Plan</i>	55
Tabel 4.17	<i>Blackbox Testing Disconnect</i> Aplikasi.....	56
Tabel 4.18	List Saran Responden	58
Tabel 4.19	Tabel Kategori Pengelompokan Pertanyaan	61

BAB I

PENDAHULUAN

1.1 Latar Belakang

UAV (*Unmanned Aerial Vehicle*) adalah pesawat terbang tanpa awak di udara yang memanfaatkan gaya aerodinamik sehingga menghasilkan gaya angkat pada badan pesawat. UAV memiliki berbagai macam jenis, di antaranya adalah *Fixed Wing*, *Racing Plane*, *VTOL*. UAV dapat diterbangkan secara manual oleh pilot menggunakan radio kontrol, atau secara *autonomous (auto)*. Penerbangan secara *autonomous* dapat dilakukan dengan melakukan pengolahan data sensor yang terdapat pada UAV sehingga penerbangan secara auto dapat dilakukan. Untuk melakukan penerbangan secara *autonomous* ini, dibutuhkan sebuah sistem aplikasi *Ground Control Station (GCS)* yang berfungsi untuk mengendalikan UAV. Banyak misi yang dapat dilakukan dengan menggunakan mode penerbangan secara *autonomous*. Salah satunya adalah misi penerbangan untuk melakukan pemetaan wilayah. UAV juga dapat dilengkapi dengan beberapa komponen tambahan untuk menunjang pelaksanaan misi, misalnya seperti kamera. Penggunaan perangkat tambahan ini juga dikontrol melalui aplikasi *Ground Control Station* untuk membantu berjalannya misi penerbangan.

Ground Control Station (GCS) merupakan pusat kontrol di darat yang terdiri dari beberapa fasilitas untuk mengontrol kendaraan tanpa awak di udara. GCS dapat

digunakan untuk menerima data maupun mengirim perintah. Ada beberapa aplikasi *Ground Control Station* yang biasa digunakan untuk melakukan penerbangan. Namun, salah satu aplikasi GCS yang paling umum digunakan adalah aplikasi *Mission Planner*. Aplikasi *Mission Planner* merupakan aplikasi *Ground Control Station* yang bersifat *Open Source* dan dapat disesuaikan dengan kebutuhan pengguna, baik dari segi fitur, sistem, maupun dari segi tampilannya. Aplikasi *Mission Planner* bersifat kompleks dan memiliki banyak sekali fitur sehingga terbilang sangat lengkap.

Saat ini, pengguna aplikasi *Mission Planner* dapat menggunakan aplikasi tersebut dengan mengunduhnya secara gratis pada website ardupilot.org. Aplikasi ciptaan Michael Osborne ini dapat diinstall pada desktop PC. Berawal dari adanya aplikasi tersebut dan bersifat open source, muncul lah ide untuk mengembangkan aplikasi *Mission Planner* dengan pengubahan tampilan yang sesuai dengan kebutuhan. Pengembangan tampilan yang dilakukan akan melalui pendekatan *User Experience*, dimana pengalaman user akan berperan penting dalam pengembangan aplikasi ini. Diharapkan dengan adanya pengembangan aplikasi ini dapat mempermudah pengguna dalam melakukan misi penerbangan. Dengan begitu, penelitian ini dilakukan untuk mengembangkan tampilan aplikasi *Mission Planner* berbasis *desktop*. Aplikasi ini dikembangkan dengan menggunakan aplikasi *Microsoft Visual Studio Community 2019* dengan bahasa pemrograman C#.NET. Penelitian ini akan diuji coba dengan menggunakan fasilitas milik Laboratorium Teknik Digital.

1.2 Rumusan Masalah

Bagaimana mengembangkan sebuah aplikasi *Ground Control Station* (GCS) yang memiliki tampilan sesuai dengan kebutuhan serta dapat mempermudah pengguna ketika sedang melakukan misi penerbangan.

1.3 Tujuan Penelitian

Tujuan dari penelitian ini adalah mengembangkan sebuah aplikasi *Ground Control Station* (GCS) yang mampu memudahkan pengguna dalam mengendalikan dan memantau kondisi UAV di udara secara *real time* dengan menampilkan data keadaan dan parameter penerbangan UAV sesuai dengan kebutuhan pengguna.

1.4 Batasan Masalah

Adapun batasan masalah pada penelitian ini adalah sebagai berikut :

1. GUI dikembangkan dengan menggunakan *Visual Studio Community 2019* C#.NET
2. Aplikasi *Ground Control Station* (GCS) yang digunakan adalah *Mission Planner*.
3. Parameter keberhasilan dari pengembangan aplikasi ini adalah aplikasi mampu menampilkan tampilan sesuai dengan kebutuhan dengan menggunakan dua *screen monitor*.

1.5 Hipotesis Penelitian

Aplikasi GCS menampilkan data aplikasi menggunakan dua layar; layar pertama menampilkan halaman *Flight Data* dan layar kedua menampilkan halaman *Flight*

Plan. Pada halaman *Flight Data* akan menampilkan *Quick Data 1*, HUD (*Heads Up Display*), *Quick Data 2*, Map, Data Pesawat secara *Real Time*, dan *Camera Overlay*. Pada halaman *Flight Plan* akan menampilkan *Map*, *Menu*, serta *Setting Mission*.

1.6 Manfaat Penelitian

Adapun manfaat yang didapatkan dari penelitian ini adalah dapat mempermudah pengguna dalam menggunakan aplikasi *Ground Control Station* dalam melaksanakan misi penerbangan.

1.7 Sistematika Penulisan

Untuk memudahkan penulisan dan pemahaman mengenai penelitian, maka sistematika penulisan tugas akhir ini tersusun dari lima bab sebagai berikut :

BAB I PENDAHULUAN

Bab ini memuat latar belakang, tujuan, manfaat, perumusan masalah, batasan masalah, hipotesis, dan sistematika penulisan.

BAB II DASAR TEORI

Bab ini berisi tentang teori – teori yang mendukung mengenai perencanaan dan pembuatan aplikasi GCS.

BAB III METODOLOGI PENELITIAN

Bab ini berisi waktu dan tempat penelitian, alat dan bahan yang

digunakan, garis besar metode yang diusulkan, serta diagram alir metode yang diusulkan.

BAB IV HASIL DAN PEMBAHASAN

Bab ini menjelaskan hasil penelitian, pembahasan, dan perhitungan kinerja metode yang diusulkan.

BAB V KESIMPULAN DAN SARAN

Bab ini memuat kesimpulan yang diperoleh dari hasil penelitian, dan saran – saran untuk pengembangan lebih lanjut.

DAFTAR PUSTAKA

LAMPIRAN

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terkait

Pada bab ini terdapat beberapa penelitian terkait yang digunakan sebagai perbandingan dalam penelitian Rancang Bangun Aplikasi *Ground Control Station* untuk Pengawasan dan Pengendalian UAV.

Pengembangan Sistem Autopilot Pada Wahana Udara Tanpa Awak Fixed Wing Proto-03

Skripsi Tugas Akhir – Universitas Lampung (2015)

Penelitian yang pernah dilakukan oleh Aris Susilo adalah Pengembangan Sistem *Autopilot* Pada Wahana Udara Tanpa Awak *Fixed Wing* Proto-03. Penelitian ini mengembangkan sistem *autopilot* dengan melakukan konfigurasi aplikasi dengan perangkat keras berupa sebuah pesawat *aeromodelling*. Sehingga pesawat tersebut dapat melakukan penerbangan secara otomatis dengan mengikuti *waypoint* yang telah ditentukan.

Sistem *Ground Control Station* Berbasis *Mobile* Untuk Pengamatan Dan Pengendalian UAV

Seminar Nasional Teknologi Terapan vol. 2 (2018)

Penelitian yang pernah dilakukan oleh Ariesta Martiningtyas Handayani dan Isnan Nur Rifa'i adalah Sistem *Ground Control Station* berbasis *Mobile* untuk Pengamatan dan Pengendalian UAV. Penelitian ini dirancang dapat menampilkan parameter penerbangan UAV melalui *Android*. Data parameter penerbangan yang ditampilkan yakni data kecepatan udara (*airspeed*), *heading* (*yaw*), sudut guling dan angguk (*pitch* dan *roll*), ketinggian barometrik, *turn* dan *bank rate*, serta *climb speed*. Data tersebut telah disesuaikan dengan data yang didapat dari modul ADAHRS. Dalam implementasinya, *GCS mobile* dan *quadcopter* akan saling terhubung dengan menggunakan *transmitter* dan *receiver*. Kemudian data status *quadcopter* akan ditampilkan pada layar *Android*. Aplikasi ini dibangun untuk memantau dan mengendalikan UAV dari jarak jauh ketika sudah berada di luar jangkauan pandangan [1].

Ground Control Station Development for Autonomous UAV

Lecture Notes in Computer Science (LNAI, vol 5315 2008)

Penelitian yang pernah dilakukan mengenai *Ground Control Station Development for Autonomous UAV* oleh Ye Hong, Jiancheng Fang, dan Ye Tao ini dikembangkan menggunakan bahasa pemrograman C++. Pada GUI utama menampilkan *toolbar*, *flight path*, *equipment monitoring*, *instrument board*, *information of UAV*, dan *serially transmitted data*. Pada penelitian ini dibangun aplikasi GCS yang terdiri dari tiga hal utama, yakni GCS yang berfungsi dengan

baik, perangkat keras, dan panel instrumen untuk menampilkan informasi, jalur penerbangan, dan lain-lain [2].

A Ground Control Station for a Multi-UAV Surveillance System: Design and Validation in Field Experiments

Journal of Intelligent & Robotic Systems vol 69 (2013)

Penelitian *A Ground Control Station for a Multi-UAV Surveillance System: Design and Validation in Field Experiments* dilakukan oleh Daniel Perez-Rodriguez, Ivan Maza, Fernando Caballero, David Scarlatti, Enrique Casado, dan Anibal Ollero. Aplikasi ini bersifat *multiplatform* dan berbasis C++. GUI dibangun menggunakan Qt Library, *video image processing* menggunakan OpenCV Library, *realistic 3D Environment* dikembangkan menggunakan *OpenSceneGraph* dan *OSG Earth library*. Aplikasi *Ground Control Station* ini dapat mendeteksi penyusupan UAV, menampilkan peringatan seperti *low battery*, masalah pada transmisi video, menampilkan informasi yang dibutuhkan pada UAV, menampilkan lokasi UAV pada peta sehingga tetap terjangkau, dan menampilkan lokasi pemantauan secara 3D dengan menggunakan kamera yang terpasang pada wahana. Aplikasi ini dirancang untuk melakukan misi pemantauan suatu lingkungan [3].

Improving Design of Ground Control Station for Unmanned Aerial Vehicle: Borrowing from Design Patterns

36th EUROMICRO Conference on Software Engineering and Advanced Applications (2010)

Mladan Jovanovic, Dusan Starcevic, dan Zoran Jovanovic melakukan penelitian tentang *Improving Design of Ground Control Station for Unmanned Aerial Vehicle : Borrowing from Design Patterns*. Penelitian ini memaparkan pengalaman dalam mengembangkan arsitektur aplikasi berorientasi objek. Pengembangan ini dapat digunakan kembali untuk GCS dan UAV dengan menggunakan pola desain yang baru serta meminimalisir fitur yang tidak berguna [4].

Pengembangan Sistem Ground Control Station Berbasis Internet Webservice pada Pesawat Tanpa Awak

Indonesian Journal of Electronics and Instrumentation Systems (IJEIS 2018)

Fredy Aga Nugroho, Raden Sumiharto, dan Roghib Muhammad Hujja melakukan penelitian mengenai Pengembangan Sistem *Ground Control Station* Berbasis *Internet Webservice* pada Pesawat Tanpa Awak yang dirancang untuk melakukan pemantauan kondisi dan kendali UAV secara jarak jauh. Penelitian ini menggunakan modul ADAHRS sebagai penyedia data sensor, *webcam* untuk merekam tampilan visual dari keadaan UAV, *Raspberry Pi* sebagai pemroses data, modem sebagai penghubung antara GCS dengan internet. Sistem terdiri dari dua unit, yaitu unit terbang dan unit GCS. Unit terbang terdiri dari *Raspberry Pi*, modem, *Webcam*, modul ADHRS, serta perangkatnya dengan *controller Multi-Wii*. Kemudian GCS dapat menampilkan kondisi pesawat, perintah kendali, serta *video streaming* [5].

Analysis of Design Directions for Ground Control Station (GCS)

Journal of Computer and Communications 4 (2016)

Penelitian mengenai *Analysis of Design Directions for Ground Control Station* (GCS) oleh Yongjin (James) Kwon, Jinyeong Heo, Sekyung Jeong, Seunghee Yu, dan Suhwan Kim membahas mengenai analisa karakteristik desain serta spesifikasi dilakukan terhadap beberapa jenis GCS dan UAV di seluruh dunia yang terdiri dari tampilan halaman informasi, metode penampilan, jumlah halaman, informasi dalam situasi taktik siasat, pengoperasian peralatan peta taktik, jumlah operator, posisi operator, *layout*, dan lingkungan pengoperasian. Analisa ini menggunakan metode *Human Machine Interface* yang telah dikembangkan pada *touch screnn* dan *multi screen system* untuk penggunaan yang cepat dan tampilan yang berisi informasi yang luas. Analisa ini dilakukan sebagai persiapan dalam situasi pertempuran UAV sehingga penggunaan GCS dapat lebih optimal [6].

A Ground Control Station for the UAV Flight Simulator

Acta Mechanica et Automatica vol 10 (2016)

Penelitian mengenai *A Ground Control Station for the UAV Flight Simulator* yang dilakukan oleh Slawomir Romaniuk, Zdzislaw Gosiewski, dan Leszek Ambroziak membahas aplikasi GCS bekerja sama dengan simulator penerbangan, kemudian menampilkan berbagai parameter penerbangan pesawat. Penelitian ini dibangun menggunakan Bahasa C++. Desain antarmuka aplikasi ini sederhana, sekaligus mempertahankan fungsi dan kelayakannya [7].

Sistem *Ground Control Station* untuk Pengamatan dan Pengendalian *Unmanned Aerial Vehicle*

Seminar Nasional Teknologi Terapan vol. 2 (2016)

Ariesta Martiningtyas Handayani dan Budi Sumanto membangun Sistem *Ground Control Station* untuk Pengamatan dan Pengendalian *Unmanned Aerial Vehicle*. Sistem ini dibangun untuk mendeteksi GPS, sensor IMU (*Inertial Measurement Unit*) yang terdiri dari *accelerometer*, *gyroscopes*, dan *magnetometer* yang diletakkan pada UAV. GUI dirancang menggunakan Visual Studio C#, GIS didesain menggunakan Gmap.NET, pemrosesan video menggunakan EmguCV, dan *database* menggunakan MySQL. Sistem ini dibuat untuk menampilkan lokasi UAV pada *desktop* serta data keadaan UAV seperti *altitude*, posisi (*Latitude* dan *longitude*), arah, *roll*, *pitch*, *yaw*. Selain itu, terdapat fitur *video streaming* yang dapat digunakan ketika UAV telah dipasangkan kamera [8].

2.2 Landasan Teori

2.2.1 Aplikasi Berbasis *Desktop*

Aplikasi *desktop* merupakan aplikasi yang dapat berjalan pada platform *desktop* dengan cara menginstalnya terlebih dahulu pada sistem operasi yang digunakan. Aplikasi *desktop* pada umumnya bersifat *offline* dan mandiri sehingga penggunaannya tidak banyak bergantung pada koneksi internet. Aplikasi berbasis *desktop* pada tiap sistem operasi pada umumnya memiliki tampilan yang sedikit berbeda namun memiliki fungsi yang sama [9].

2.2.2 Bahasa Pemrograman C#

C# (C Sharp) lahir seiring dengan adanya pembuatan .NET Framework. Anders Hejlsberg merupakan *Chief Architect* dari C# serta Borland Delphi dan Turbo Pascal. Bahasa C# merupakan Bahasa yang simpel, modern, berorientasi objek, dan merupakan Bahasa pemrograman bertipe aman. C# merupakan kombinasi dari Bahasa C++ dan produktivitas tinggi dari *Rapid Application Development (RAD)*. Konsep yang sulit dari C++ disederhanakan untuk mendapatkan produktivitas yang tinggi [10].

2.2.3 Microsoft Visual Studio

Microsoft Visual Studio adalah sebuah aplikasi lengkap yang dapat digunakan untuk melakukan pengembangan aplikasi. Aplikasi dapat berupa aplikasi bisnis, aplikasi personal, aplikasi console, komponen aplikasi, aplikasi Web, ataupun aplikasi Windows.

Visual Studio mencakup beberapa komponen seperti kompiler, *Integrated Development Environment (IDE)*, SDK, dan dokumentasi berupa *MSDN Library*. Selain itu, kompiler yang dimasukkan ke dalam paket *Visual Studio* yakni *Visual C#*, *Visual C++*, *Visual Basic.NET*, *Visual Basic*, *Visual InterDev*, *Visual J++*, *Visual J#*, *Visual SourceSafe*, *Visual FoxPro*.

Microsoft Visual Studio dapat berfungsi untuk mengembangkan aplikasi dalam *native code* (berupa bahasa mesin yang berjalan di atas Windows) ataupun *managed code* (berupa *Microsoft Intermediate Language* di atas *.NET Framework*). Selain

itu, *Visual Studio* juga dapat berfungsi untuk mengembangkan aplikasi *Silverlight*, aplikasi *Windows Mobile* (yang berjalan di atas *.NET Compact Framework*).

Microsoft Visual Studio merupakan salah satu aplikasi yang sering digunakan oleh para *programmer*, dan menjadi salah satu bahasa yang wajib dipelajari. *Visual Studio* juga disebut sebagai sarana untuk menghasilkan aplikasi berbasis *Windows*. Beberapa kemampuan atau manfaat dari *Visual Studio* di antaranya seperti:

1. Membuat program aplikasi berbasis *Windows*.
2. Membuat objek pembantu program seperti kontrol *ActiveX*, aplikasi Internet, *file Help*, dan lain-lain.
3. Melakukan pengujian program (*debugging*), menghasilkan program bersifat *executable* (.exe), maupun menghasilkan program yang dapat langsung dijalankan [11].

2.2.4 Ground Control Station (GCS)

Ground Control Station (GCS) merupakan stasiun kontrol darat yang berfungsi sebagai pengendali dan pemantau UAV yang sedang melaksanakan misi penerbangan secara jarak jauh dari darat. *Ground Control Station* terdiri dari tiga bagian utama, yaitu *Mission Control System* (MCS), *Payload Control System* (PCS), dan *Communication System* (CS).

Mission Control System (MCS) merupakan aplikasi *Ground Control Station* (GCS) yang terinstall pada komputer baik *Personal Computer* (PC), laptop, *Raspberry Pi*,

maupun *mini computer*. Aplikasi ini dioperasikan oleh *Ground Control Operator (GCO)* atau dalam bahasa Indonesia biasa disebut *Operator GCS*. Aplikasi GCS biasanya terdiri dari dua layar monitor, layar utama untuk aplikasi GCS dan layar tambahan untuk tampilan *First Person View (FPV)*. *Operator GCS* memantau seluruh kegiatan penerbangan dan misi yang dilaksanakan termasuk pemeriksaan sebelum penerbangan (*Pre-Flight*), pelaksanaan misi, laporan penerbangan, dan protokol darurat.

Mission Control System menampilkan semua informasi yang diperlukan pada *Operator GCS* sehingga operator tidak perlu bergantung pada bantuan eksternal. Data yang bisa dipantau oleh *operator GCS* contohnya seperti *indication airspeed*, *ground speed*, posisi pesawat, *horizon virtual*, status bahan bakar, status persentase baterai, arah, ketinggian, kecepatan dan arah angin, dan sebagainya.

Payload Control System dioperasikan oleh *Payload Operator (PO)*. Dalam kasus pemetaan wilayah, contoh *payload* adalah gimbal. PO mengkalibrasi dan mengendalikan gimbal, mengarahkan ke posisi yang diinginkan, mengunci posisi target, melacak target, beralih antara sensor IR dan EO, merekam video dan melakukan semua operasi yang diperlukan untuk mengoperasikan gimbal.

Sistem Komunikasi (*Communication System* atau CS) terdiri dari antena dengan kekuatan dan frekuensi tertentu. *Video receiver* atau penangkap sinyal video merupakan bagian dari jaringan video yang terletak di darat (*ground*). Jaringan video ini bisa berupa jaringan satu arah maupun dua arah antara wahana dengan

perangkat di darat. Dengan informasi yang terkumpul dari MCS, CS memberi perintah dan mengarahkan antena darat untuk menangkap jaringan video dan ke wahana udara setiap saat secara otomatis.

Pada penelitian ini, *Mission Control System (MCS)* diinstall pada laptop sebagai aplikasi utama GCS. *Communication System (CS)* berbentuk *antenna* yang terhubung pada laptop sebagai *transmitter* dan pada UAV sebagai *receiver*. Modul antena yang digunakan adalah *Module Telemetry 433 MHz*



Gambar 2.1 *Ground Control Station*

2.2.5 Mission Planner

Mission Planner merupakan salah satu aplikasi *Ground Control Station* yang digunakan untuk melakukan mengendalikan dan memonitoring UAV selama misi penerbangan berlangsung. *Ardupilot* berfungsi untuk mengatur kendali serta stabilisasi UAV. Sedangkan *Mission Planner* digunakan untuk mengatur *waypoint* yang nantinya harus dilewati oleh UAV serta memantau data-data yang didapatkan dari hasil uji terbang menggunakan *Ardupilot*. *Monitoring* ini perlu dilakukan guna

untuk mengawasi keadaan UAV selama melakukan penerbangan agar keadaan tetap stabil. Dengan begitu misi yang dijalankan juga dapat berjalan dengan baik. Aplikasi *Mission Planner* ini merupakan aplikasi desktop yang bersifat *Open Source*. Aplikasi ini dikembangkan oleh Michael Osborne.

2.2.6 Unmanned Aerial Vehicle (UAV)

Unmanned Aerial Vehicle (UAV) adalah pesawat yang dikendalikan tanpa awak dan dapat berfungsi dengan adanya kendali jarak jauh oleh pilot. UAV menggunakan hukum aerodinamika untuk melakukan penerbangan. UAV mampu membawa muatan seperti senjata ataupun muatan lainnya. UAV memiliki ukuran, bentuk, karakter, serta konfigurasi yang bervariasi. UAV dapat dikontrol melalui alat pengendali jarak jauh serta dapat terbang berdasarkan perintah misi yang telah dimasukkan ke dalam sistem. [13].

2.2.7 Flight Controller

Flight Controller (FC) adalah sistem pengendali terbang otomatis yang terdiri dari berbagai komponen dan sensor. *Flight controller* berfungsi untuk menjaga UAV dapat terkendali dan memiliki keseimbangan yang baik. Sensor keseimbangan yang terdapat pada *flight controller* adalah *sensor gyro* dan *accelerometer*. *Flight controller* jenis baru memiliki sensor tambahan seperti tekanan udara (*barometer*), kompas (*magnetometer*), dan *GPS*. Sensor *barometer* berfungsi untuk menjaga ketinggian UAV pada ketinggian tertentu. Sensor *magnetometer* dan *GPS* digunakan untuk menjaga orientasi, *autopilot* dan fitur *failsafe*. Terdapat jenis-jenis *flight controller* yang memiliki komponen *hardware* yang sama namun setiap

	Tegangan input maksimum: <i>45V (10S LiPo)</i>
	Deteksi arus maksimum: <i>90A</i>
	Input daya USB: <i>4.1 - 5.5V</i>
	Input tegangan servo: <i>0 - 10V</i>
<i>Interface</i>	<i>1 x UART Serial Port (for GPS)</i>
	<i>Spektrum DSM/DSM2/DSM-X® Satellite Compatible RC input</i>
	<i>Futaba S BUS® Compatible RC input</i>
	<i>PPM Sum Signal RC Input</i>
	<i>I2C (untuk sensor digital)</i>
	<i>CAN (untuk kontrol motor digital)</i>
	<i>ADC (untuk sensor analog)</i>
	<i>Micro USB Port</i>
<i>Dimensi dan berat</i>	<i>Dimensi : 38x43x12mm</i>
	<i>Berat: 15.8g</i>

2.2.9 Module Telemetry

Telemetry merupakan sebuah alat yang memungkinkan dalam melakukan pengukuran jarak jauh dan pelaporan informasi kepada perancang atau operator sistem. Pada penelitian ini menggunakan *Telemetry* dengan frekuensi sebesar 433 MHz. Jika jangkauan *telemetry* terputus atau jangkauan *remote control* terputus maka sistem *failsafe* akan memerintahkan wahana untuk mengaktifkan *mode RTL* [15].

2.2.10 Global Positioning System (GPS)

Pada wahana ini menggunakan *Global Positioning System (GPS)* yaitu *Here GNSS*. GPS ini memiliki beberapa fitur di antaranya *Notification RGB LED*, *HMC5983 MAG*, and *LIS3MDL MagGPS*, dan *6CM Ground plane*. GPS merupakan sistem navigasi berbasis satelit yang terdiri dari 24 jaringan satelit yang ditempatkan di orbit. GPS ini dilengkapi dengan *compass* dan merupakan komponen yang penting pada wahana dan sebagai arah petunjuk untuk wahana menjalankan misi yang diperintahkan ketika wahana dalam *mode autonomous* atau *mode manual*. Keluaran dari GPS yaitu *Latitude* dan *Longitude*. Hingga saat ini GPS merupakan sistem satelit navigasi yang paling populer dan paling banyak di aplikasikan di dunia, baik di darat, laut, udara, maupun di angkasa [16].

2.2.11 ADAHRS (Air Data, Attitude, and Heading Reference System)

ADAHRS (*air data, attitude, and heading reference system*) merupakan gabungan dari *sensor air data (AD)* dan sistem referensi *attitude and heading (AHRS)*. Sistem ini memiliki peran penting dalam memberikan data parameter-parameter penerbangan yang akan digunakan oleh modul lain dalam UAV. Parameter penerbangan yang dapat dibaca oleh ADAHRS adalah sudut *yaw*, sudut *roll*, sudut *pitch*, serta data kecepatan, ketinggian, suhu, koordinat GPS, dan tekanan yang akan digunakan sebagai referensi dalam mengendalikan UAV. Sistem ADAHRS yang diteliti berbasis pada *Arduino Mega 2560*. Sistem mendapatkan data dari sensor *Razor 9DOF IMU*, sensor ketinggian berbasis *BMP085*, sensor kecepatan udara berbasis *MPXV5004DP* + pitot sederhana, dan penerima *GPS PMB648*. Data dari

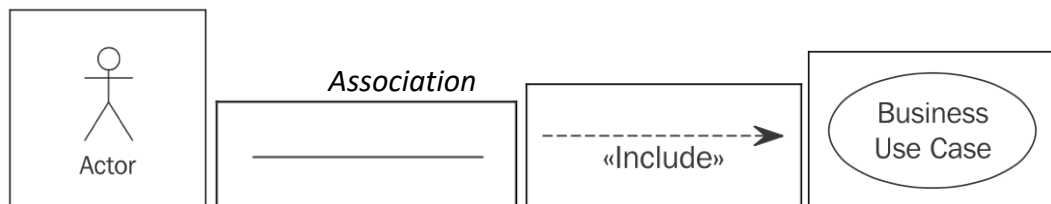
sensor-sensor tersebut nantinya akan diolah untuk dikirimkan ke modul yang lain melalui komunikasi serial [17].

2.2.12 GMap.NET

GIS (*Geographyc Information System*) didesain dengan menggunakan *library* Gmap.Net. *Google Map Service* adalah sebuah jasa peta global virtual gratis dan *online* yang disediakan oleh perusahaan *Google*. *Google Maps* menawarkan peta yang dapat diseret dan gambar satelit untuk seluruh dunia. *Google Maps* juga menawarkan pencarian suatu tempat dan rute perjalanan. *Google maps* dapat digunakan pada berbagai platform dan juga banyak digunakan sebagai petunjuk arah pada aplikasi *android* [18].

2.2.13 Use Case Diagram

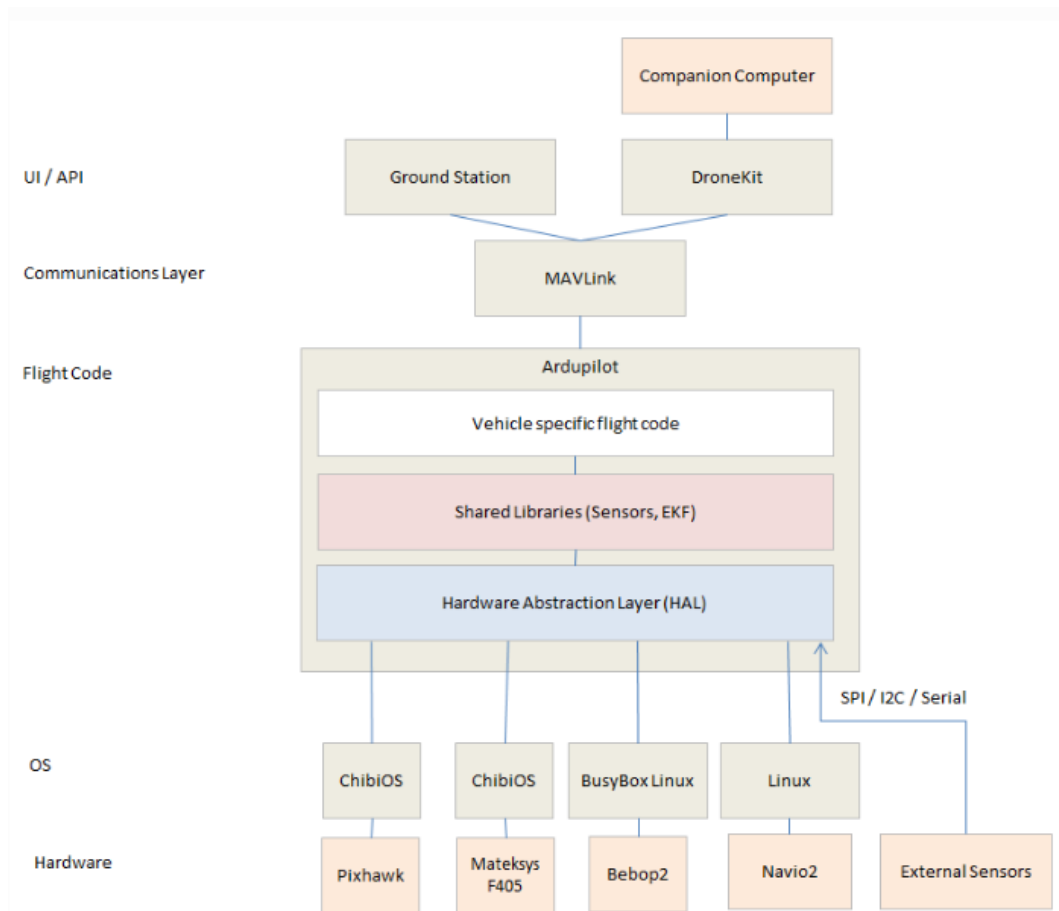
Use Case Diagram adalah representasi dari interaksi pengguna dengan sistem, hubungan antara pengguna dan sistem ini dijadikan sebuah hubungan yang saling terikat yaitu pengguna dapat menggunakan fungsionalitas dari sistem, dan sistem akan dioperasikan oleh pengguna. Berikut ini adalah komponen yang biasanya digunakan pada *Use Case*:



Gambar 2. 3 Bagian *Use Case Diagram* [19]

- Aktor, adalah representasikan peran orang luar atau pengguna saat berinteraksi dengan sistem bisnis. Misalnya, seorang aktor dapat menjadi pelanggan, mitra bisnis, pemasok, atau sistem bisnis lainnya sesuai dengan sistem atau kasus yang diteliti
- *Association*, adalah hubungan/*relationship* antara *actor* dengan *Business Use Case* , yang berarti *actor* dapat mengakses fungsionalitas dari *Use Case* yang dituju
- *Include*, adalah hubungan antara dua buah *Use Case* yang harus saling berhubungan, atau menjadi prasyarat agar bias menggunakan fungsi tertentu.
- *Business Use Case*, menggambarkan interaksi antara seorang aktor dan sistem bisnis, artinya menggambarkan fungsionalitas sistem bisnis yang digunakan oleh aktor tersebut:
- *Subject*, adalah gambaran atau representasi dari sistem atau bisnis yang di rancangkan. *Subject* berbentuk persegi Panjang yang berarti semua *Use Case* yang ada di dalamnya adalah termasuk ke dalam cakupan sistem yang sama [19].

2.2.14 Arsitektur Aplikasi *Mission Planner*



Gambar 2.4 Basic *Structure* aplikasi

Struktur dasar *ArduPilot* dibagi menjadi 5 bagian, yakni sebagai berikut:

- *Vehicle Code*
- *Shared Libraries*
- *Hardware Abstraction Layer (AP_HAL)*
- *Tools Directories*
- *External Support Code* (Contohnya *Mavlink*, *dronekit*) [20]

1. Vehicle Code

Vehicle Code atau daftar wahana adalah daftar yang berisi *firmware* tertentu untuk setiap tipe wahana. Ada 6 tipe wahana pada *Mission Planner* yaitu *Plane*, *Copter*, *Rover*, *Sub*, *Blimp*, dan *Antenna Tracker*. Terdapat elemen dan parameter yang berbeda untuk tiap tipe wahana tersebut.

2. Libraries

Library berisi *driver sensor*, EKF untuk estimasi ketinggian dan posisi, serta kode untuk kontrol seperti PID.

3. AP_HAL

AP_HAL (*Hardware Abstraction Layer*) adalah lapisan yang membuat *ArduPilot* bisa digunakan pada banyak platform yang berbeda.

Ada AP_HAL tingkat atas di perpustakaan/AP_HAL yang mendefinisikan antarmuka bahwa kode lanjutan memiliki fitur tertentu, kemudian ada sub *direktori* AP_HAL_XXX untuk setiap jenis perangkat, misalnya AP_HAL_AVR untuk perangkat berbasis AVR, AP_HAL_PX4 untuk *Pixhawk* dan AP_HAL_Linux untuk perangkat berbasis *Linux*.

4. Tools Directories

Tools directories adalah daftar pendukung untuk *tools* tambahan yang juga digunakan untuk kontrol wahana. Sebagai contoh, *tools autotest* menyediakan infrastruktur untuk tes wahana secara otomatis dan *tools Replay* memberikan *utility log Replay*.

5. External Support Code

Beberapa perangkat membutuhkan dukungan kode eksternal untuk fitur – fitur baru atau fitur khusus yang ada pada perangkat tersebut. Beberapa sumber dukungan eksternal saat ini seperti *PX4NuttX*, *PX4Firmware*, *uavcan*, dan *mavlink*

BAB III

METODOLOGI PENELITIAN

3.1 Waktu dan Tempat

Penelitian mulai dilaksanakan pada bulan April 2020 sampai dengan bulan Juni 2022. Penelitian dilakukan di Laboratorium Teknik Digital, Laboratorium Terpadu Teknik Elektro, Jurusan Teknik Elektro Universitas Lampung.

3.2 Alat dan Bahan

Alat dan bahan yang diperlukan pada penelitian skripsi ini adalah sebagai berikut :

3.2.1 Komponen Elektronika

Komponen elektronika yang diperlukan dalam penelitian ini adalah sebagai berikut:

Tabel 3.1 Komponen elektronika

No.	Nama	Kuantitas (Unit)	Kegunaan
1.	<i>Flight Controller 3DR Pixhawk Mini</i>	1	Suatu alat yang terdiri dari berbagai komponen dan sensor untuk menjaga UAV tetap terkendali dan memiliki keseimbangan yang baik

2.	<i>Module Telemetry 433 MHz</i>	1	Sebuah perangkat yang digunakan mengirim data dan perintah misi dari GCS ke wahana dan sebaliknya
3.	<i>Global Positioning System (GPS)</i>	1	Sebuah perangkat yang digunakan untuk mengetahui posisi wahana saat melakukan misi.

3.2.2 Komponen Pendukung

Komponen pendukung lainnya yang diperlukan dalam pembuatan system adalah sebagai berikut :

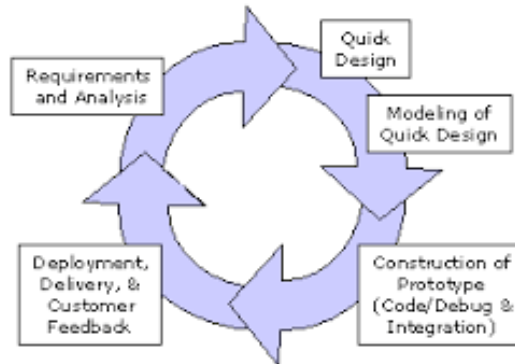
Tabel 3.2 Komponen pendukung

No.	Nama	Kuantitas(Unit)	Kegunaan
1.	Laptop	1	Digunakan untuk melakukan <i>coding</i> dan penelitian.
2.	<i>Microsoft Visual Studio</i>	1	Aplikasi untuk pembuatan pengkodean dan <i>compile</i> program
3.	Kabel <i>MicroUSB</i>	1	Sebagai penghubung antara laptop dengan <i>flight controller</i>

3.3 Tahapan Penelitian Metode *Prototype*

Prototype adalah salah satu pendekatan dalam rekayasa aplikasi yang secara langsung mendemonstrasikan bagaimana sebuah aplikasi atau komponen-

komponen aplikasi akan bekerja dalam lingkungannya sebelum tahapan konstruksi aktual dilakukan.



Gambar 3.1 Metode *Prototype*

Pengembang dan pengguna bertemu dan mendefinisikan obyektif keseluruhan dari aplikasi dan mengidentifikasi segala kebutuhan yang diketahui. Secara ideal *prototype* berfungsi untuk mengidentifikasi kebutuhan aplikasi. *Prototype* dapat menjadi paradigma yang efektif bagi rekayasa aplikasi. Kunci dari *prototype* adalah mendefinisikan aturan dan kebutuhan di awal, yakni pengguna dan pengembang sama-sama menyepakati bahwa *prototype* digunakan sebagai mekanisme pendefinisian kebutuhan [19].

Metode *Prototype* merupakan suatu hal baru dalam metode pengembangan perangkat lunak. *Prototype* tidak hanya sekedar evolusi dalam dunia pengembangan perangkat lunak, namun telah berevolusi menjadi metode pengembangan perangkat lunak yakni sistem *sekuensial*. Sistem sekuensial atau disebut dengan SDLC atau *waterfall development model*. Dalam model *prototype*, perangkat lunak yang dihasilkan nantinya akan dipresentasikan kepada pengguna, dan pengguna akan diberikan kesempatan untuk memberikan masukan sehingga

perangkat lunak yang dihasilkan nantinya akan sangat sesuai dengan keinginan dan kebutuhan pengguna.

Tahapan model *prototype* dapat dijabarkan sebagai berikut :

A. *Requirement and Analysis*

Pada tahap ini, pengembang dan masyarakat bertemu dan mendefinisikan keseluruhan perangkat lunak yang diinginkan masyarakat, mengidentifikasi segala kebutuhan, mengidentifikasi masalah dari masyarakat serta mendefinisikan tujuan ke depan dari pengembangan *software* yang akan dibuat dan sasaran yang akan dicapai.

B. *Quick Design and Modelling Quick Design*

Tahapan ini melanjutkan proses dari tahap *communication* yaitu membuat perencanaan dan permodelan secara cepat lebih difokuskan dalam mempresentasikan aspek-aspek apa saja yang akan ditampilkan atau dilihat oleh masyarakat.

C. *Construction of Prototype*

Tahapan ini merupakan proses pembuatan program yang berdasarkan spesifikasi sistem dan desain perancangan yang didapatkan dari hasil tahapan sebelumnya.

D. *Deployment Delivery and Feedback*

Hasil dari pembuatan program akan diberikan ke masyarakat dan kemudian dievaluasi oleh masyarakat sendiri. *Feedback* yang didapatkan dari keluhan

ataupun saran yang diberikan oleh masyarakat setelah mengevaluasi *prototype* yang telah dicoba.

Kelebihan Model *Prototype*

- Pengguna berpartisipasi aktif dalam proses pengembangan sistem. Sehingga hasil dari produk yang dikembangkan akan dapat sesuai dengan kebutuhan pengguna.
- Penentuan kebutuhan lebih mudah untuk diimplementasikan
- Dapat mempersingkat waktu dalam pengembangan perangkat lunak.
- Adanya komunikasi yang baik antara pengembang dan pengguna.
- Pengembang dapat bekerja dengan lebih baik dalam menentukan kebutuhan pengguna.
- Dapat menghemat waktu dalam pengembangan perangkat.
- Dapat mempermudah penerapan karena pengguna mendefinisikan hal yang diharapkan.

Kekurangan Model *Prototype*

- Proses analisis serta perancangan yang terlalu singkat.
- Bersifat kurang fleksibel ketika terjadi perubahan.
- Meskipun pengguna dapat mengikuti alur proses pengembangan seperti revisi *prototype*, tetapi pengguna biasanya tidak memperhatikan dari segi kualitas serta pemeliharaan jangka panjang.
- Pengembang terkadang bernegosiasi mengenai implementasi sistem dengan menggunakan sistem dan algoritma yang tidak efisien dan relevan.

DAFTAR PUSTAKA

- [1] A. M. Handayani and I. N. Rifa'i, "Sistem Ground Control Station Berbasis Mobile Untuk," *Semin. Nas. Teknol. Terap.*, vol. 2, no. 16, pp. 23–30, 2018.
- [2] Y. Hong, J. Fang, and Y. Tao, "Ground control station development for autonomous UAV," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 5315 LNAI, no. PART 2, pp. 36–44, 2008, doi: 10.1007/978-3-540-88518-4_5.
- [3] D. Perez, I. Maza, F. Caballero, D. Scarlatti, E. Casado, and A. Ollero, "A ground control station for a Multi-UAV surveillance system: Design and validation in field experiments," *J. Intell. Robot. Syst. Theory Appl.*, vol. 69, no. 1–4, pp. 119–130, 2013, doi: 10.1007/s10846-012-9759-5.
- [4] M. Jovanović, D. Starčević, and Z. Jovanović, "Improving design of ground control station for unmanned aerial vehicle: Borrowing from design patterns," *Proc. - 36th EUROMICRO Conf. Softw. Eng. Adv. Appl. SEAA 2010*, no. May 2014, pp. 65–73, 2010, doi: 10.1109/SEAA.2010.31.
- [5] F. A. Nugroho, R. Sumiharto, and R. M. Hujja, "Pengembangan Sistem Ground Control Station Berbasis Internet Webserver pada Pesawat Tanpa Awak," *IJEIS (Indonesian J. Electron. Instrum. Syst.)*, vol. 8, no. 1, p. 1, 2018, doi: 10.22146/ijeis.30126.
- [6] Y. (James) Kwon, J. Heo, S. Jeong, S. Yu, and S. Kim, "Analysis of Design

- Directions for Ground Control Station (GCS),” *J. Comput. Commun.*, vol. 04, no. 15, pp. 1–7, 2016, doi: 10.4236/jcc.2016.415001.
- [7] S. Romaniuk, Z. Gosiewski, and L. Ambroziak, “A Ground Control Station for the UAV Flight Simulator,” *Acta Mech. Autom.*, vol. 10, no. 1, pp. 28–32, 2016, doi: 10.1515/ama-2016-0005.
- [8] A. M. Handayani and B. Sumanto, “Sistem Ground Control Station untuk Pengamatan dan Pengendalian Unmanned Aerial Vehicle,” *Semin. Nas. Teknol. Terap.*, vol. 2, pp. 1000–1003, 2016.
- [9] Hilman, “Perbedaan Aplikasi Berbasis Web, Aplikasi Berbasis Desktop, dan Aplikasi Berbasis Mobile”, *Plimbi*, 9 November 2016, [Online]. Tersedia: <https://www.plimbi.com/article/166177/aplikasi-berbasis-web-desktop-dan-mobile> [Diakses: 12 Januari 2021].
- [10] Agro Rachmatullah, “Mempelajari C# : Bahasa Pemrograman Modern”, *Academia*, 2002, [Online]. Tersedia: https://www.academia.edu/32837430/Mempelajari_C_Bahasa_Pemrograman_Modern [Diakses: 12 Januari 2021]
- [11] Hera, “Fungsi Microsoft Visual Studio” , *Hemera Academy*, 23 April 2018, [Online]. Tersedia : <https://itlearningcenter.id/fungsi-microsoft-visual-studio/> [Diakses : 12 Januari 2021]
- [12] A-techSYN, “Payload Control System”, *A-techSYN*, [Online]. Tersedia: <https://www.a-techsyn.com/> [Diakses : 04 Maret 2021]
- [13] R. Hidayat and R. Mardiyanto, “Pengembangan Sistem Navigasi Otomatis Pada UAV (Unmanned Aerial Vehicle) dengan GPS (Global Positioning

- System) Waypoint,” *J. Tek. ITS*, vol. 5, no. 2, 2017, doi: 10.12962/j23373539.v5i2.16342.
- [14] H. S. Saroinsong *et al.*, “Rancang Bangun Wahana Pesawat Tanpa Awak (Fixed Wing) Berbasis Ardupilot,” *E-Journal Tek. Elektro Dan Komput.*, vol. 7, no. 1, pp. 73–84, 2018.
- [15] R. Kalnay, R. Grootes, T. I. Chronicles, R. Dyurgerov, and I. O. Paper, “an Introduction To Telemetry,” 2015.
- [16] Winardi, “Penentuan Posisi dengan GPS untuk Survei Terumbu Karang,” *Puslit Oceanogr.*, vol. 53, no. 9, pp. 1689–1699, 2006, doi: 10.1017/CBO9781107415324.004.
- [17] A. P. Manggala, R. Sumiharto, and S. B. Wibowo, “Purwarupa Air Data, Attitude, dan Heading Reference System untuk Unmanned Aerial Vehicle,” *IJEIS (Indonesian J. Electron. Instrum. Syst.)*, vol. 2, no. 1, pp. 21–32, 2012, doi: 10.22146/ijeis.2337.
- [18] Independent Software, “Gmap.NET Beginners Tutorial Maps Markers, Polygon, and Routes”, *Independent Software*, 29 Agustus 2016, [Online]. Tersedia: <http://www.independent-software.com/gmap-net-beginners-tutorial-maps-markers-polygons-routes-updated-for-vs2015-and-gmap1-7.html> [Diakses: 10 Februari 2021]
- [19] Ganney, P. S., Pisharody, S., & Claridge, E. (2013). Software Engineering. In *Clinical Engineering: A Handbook for Clinical and Biomedical Engineers*. <https://doi.org/10.1016/B978-0-12-396961-3.00009-3>.

- [20] Ardupilot Dev Team, “Learning Ardupilot Introduction” *Ardupilot*, 2021, [Online]. Tersedia: <https://ardupilot.org/dev/docs/learning-ardupilot-introduction.html> [Diakses: 21 Maret 2022]
- [21] M. Schrepp, “User Experience Questionnaire Handbook Version 8,” in *URL: https://www.researchgate.net/publication/303880829_User_Experience_Questionnaire_Handbook_Version_2*. (Accessed: 02.02. 2017), 2019, pp. 1–15.