

**LAPORAN PRAKTEK KERJA LAPANGAN PKL
PERUSAHAAN/INDUSTRI
PT. LEMBAGA PENERBANGAN DAN ANTARIKSA NASIONAL
PENGEMBANGAN APLIKASI PEMANTAUAN LOKASI KAPAL
BERBASIS DATA AIS SATELIT PADA PLATFORM WEBSITE**

Diajukan untuk memenuhi sebagian persyaratan Kurikulum Sarjana



Disusun oleh:

Raditya Rinandyaswara (175150200111047)

Muhammad Roshandi Naufal (175150200111056)

PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2020

PENGESAHAN

LAPORAN PRAKTIK KERJA LAPANGAN PKL
PERUSAHAAN/INDUSTRI
PT. LEMBAGA PENERBANGAN DAN ANTARIKSA NASIONAL
PENGEMBANGAN APLIKASI PEMANTAUAN LOKASI KAPAL BERBASIS DATA AIS
SATELIT PADA PLATFORM WEBSITE

Diajukan untuk memenuhi sebagian persyaratan Kurikulum Sarjana
Program studi Teknik Informatika
Bidang Pengembangan Website

Disusun oleh:
Raditya Rinandyaswara
NIM: 175150200111047

Muhammad Roshandi Naufal
NIM: 175150200111056


Praktik Kerja Lapangan ini dilaksanakan pada
29 Juni sampai dengan 29 Agustus 2020

Telah diperiksa dan disetujui oleh:

Mengetahui,
Ketua Jurusan Teknik Informatika

Achmad Basuki, S.T., M.MG., Ph.D.
NIP: 19741118 200312 1 002

Menyetujui,
Dosen Pembimbing PKL



Agi Putra Kharisma, S.T., M.T.
NIK: 2013048604301001

PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam laporan PKL ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain dalam kegiatan akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata didalam laporan PKL ini terbukti terdapat unsur-unsur plagiasi, saya bersedia PKL ini digugurkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No.20 Tahun 2003, Pasal 25 ayat 2 dan pasal 70).

Bekasi, 18 September 2020

Ketua Kelompok



Raditya Rinandyaswara

NIM: 175150200111047

KATA PENGANTAR

Puji syukur kehadiran Allah SWT yang telah melimpahkan rahmat, taufik dan hidayah-Nya sehingga laporan PKL yang berjudul “Pengembangan Aplikasi Pemantauan Lokasi Kapal Berbasis Data AIS Satelit Pada Platform Website” ini dapat terselesaikan.

Penulis menyadari bahwa laporan ini tidak akan berhasil tanpa bantuan dari beberapa pihak. Oleh karena itu, penulis ingin menyampaikan rasa hormat dan terima kasih kepada:

1. Bapak Agi Putra Kharisma, S.T., M.T. selaku dosen pembimbing PKL yang telah membimbing dan memandu kami sehingga dapat menyelesaikan laporan ini.
2. Bapak Adhitya Bhawiyuga, S.Kom., M.Sc. selaku ketua Program Studi Teknik Informatika.
3. Bapak Achmad Basuki, S.T., M.MG., Ph.D. selaku ketua Jurusan Teknik Informatika.
4. Ayahanda dan Ibunda dan seluruh keluarga besar atas segala nasehat, kasih sayang, perhatian dan kesabarannya di dalam membesarkan dan mendidik kami, serta yang senantiasa tiada henti-hentinya memberikan doa dan semangat demi terselesaikannya laporan ini.
5. Seluruh civitas akademika Teknik Informatika Universitas Brawijaya yang telah banyak memberi bantuan dan dukungan selama penyelesaian laporan PKL ini.

Kami menyadari bahwa dalam penyusunan laporan ini masih banyak kekurangan, sehingga saran dan kritik yang membangun sangat kami harapkan. Akhir kata, kami berharap PKL ini dapat membawa manfaat bagi semua pihak yang menggunakannya.

Bekasi, 18 September 2020
Ketua Kelompok,



Raditya Rinandiyaswara
radityarin@gmail.com

ABSTRAK

Pada era dimana perkembangan teknologi melaju dengan sangat pesat, teknologi sudah merambah ke dalam banyak bidang, salah satunya adalah perkapalan. *Automatic Identification System (AIS)* merupakan suatu teknologi yang berfungsi untuk melacak seluruh kapal yang terdaftar, sehingga dengan informasi tersebut didapatkan data yang dapat membantu mempermudah pelayaran. Aplikasi berbasis web ini dibuat dengan memanfaatkan data AIS untuk menampilkannya sedemikian rupa sebagai sumber pemantauan pada titik daerah tertentu. Dalam perancangannya aplikasi ini dibuat dengan metode Scrum sehingga mempermudah dalam pengembangannya. Dalam implementasinya, aplikasi berbasis *website ini* menggunakan MERN stack, yaitu dengan sisi *frontend* menggunakan bahasa pemrograman *ReactJS*, *ExpressJS* dan *NodeJS* digunakan pada sisi *backend*, dan menggunakan *MongoDB* sebagai *database* serta pertukaran data menggunakan protokol REST API. Untuk pengujian validasinya kami menggunakan metode *Blackbox* untuk menguji kesesuaian fitur-fitur yang sudah ditetapkan sebelumnya.

Kata kunci : pengembangan aplikasi, website, *Javascript*, *ExpressJS*, *MongoDB*, *NodeJs*, *ReactJS*, REST API, pengujian *blackbox*.

ABSTRACT

In an era where technological developments are progressing very rapidly, technology has penetrated into many fields, one of which is shipping. Automatic Identification System (AIS) is a technology that is used to track all registered vessels, so that with this information data can be obtained that can help facilitate shipping. This application was created by utilizing AIS data to display it in such a way as a monitoring source at a certain point in the area. This application is made with the Scrum method so that it makes it easier to develop. In its implementation, this web-based application uses the MERN stack, namely with the frontend using the ReactJS programming language, ExpressJS and NodeJS used on the backend side, and using MongoDB as the database also REST API protocols for data exchange. For validation testing we use the Blackbox method to test the suitability of the features that have been previously set.

Keywords : application development, website, *Javascript*, ExpressJS, MongoDB, NodeJS, ReactJS, REST API, *blackbox testing*.

DAFTAR ISI

PENGESAHAN	ii
PERNYATAAN ORISINALITAS	iii
KATA PENGANTAR	iv
ABSTRAK	v
ABSTRACT	vi
DAFTAR ISI	vii
DAFTAR TABEL	x
DAFTAR GAMBAR	xii
DAFTAR LAMPIRAN	xiv
BAB 1 PENDAHULUAN	1
1.1 Latar belakang	1
1.2 Rumusan masalah	2
1.3 Tujuan	2
1.4 Batasan masalah	2
1.5 Manfaat	3
1.6 Sistematika pembahasan	3
BAB 2 PROFIL OBYEK PKL	5
2.1 Sejarah Perusahaan	5
2.2 Alamat Perusahaan	5
2.3 Visi dan Misi Perusahaan	5
2.3.1 Visi	5
2.3.2 Misi	5
2.4 Struktur Organisasi Perusahaan	6
BAB 3 TINJAUAN PUSTAKA	7
3.1 Kajian Pustaka	7
3.2 Kajian Teori	7
3.2.1 Javascript	7

3.2.2	ReactJS	8
3.2.3	MongoDB	9
3.2.4	NodeJS	9
3.2.5	Express JS	9
3.2.6	Scrum	9
3.2.7	Use Case Diagram	11
3.2.8	Class Diagram	12
3.2.9	Sequence Diagram	14
3.2.10	Pengujian	16
3.2.11	Blackbox Testing	16
BAB 4 METODOLOGI PELAKSANAAN PKL		18
4.1	Studi Literatur	19
4.2	Analisis Kebutuhan	19
4.2.1	Pembuatan Product Backlog	19
4.2.2	Perencanaan Sprint	19
4.3	Perancangan dan Implementasi	19
4.3.1	Perancangan	20
4.3.2	Implementasi	20
4.4	Pengujian	20
4.4.1	Pengujian dan Demonstrasi Produk	20
4.4.2	Retrospektif	20
4.4.3	Perencanaan Sprint Selanjutnya	20
4.5	Pengambilan Kesimpulan dan Saran (Penutup)	21
BAB 5 ANALISIS KEBUTUHAN		22
5.1	Deskripsi Umum Sistem	22
5.2	Deskripsi Umum Perangkat Lunak	22
5.3	Pembuatan Backlog Product	22
5.4	Perencanaan Sprint	23
5.5	Analisa Kebutuhan	25
5.5.1	Use Case Diagram	25
5.5.2	Use Case Scenario	26

5.5.3 Sequence Diagram	30
5.5.4 Class Diagram	36
BAB 6 PERANCANGAN DAN IMPLEMENTASI	37
6.1 Perancangan	37
6.1.1 Perancangan Data	37
6.1.2 Perancangan Antarmuka dan algoritma	38
6.1.3 Lingkungan Pengembangan Sistem	43
6.1.3.1 Software	43
6.1.3.2 Sistem operasi	43
6.1.3.3 Hardware	43
6.1.3.4 Arsitektur Sistem	43
6.2 Implementasi	44
6.2.1 Implementasi Data	44
6.2.2 Implementasi Antarmuka dan Algoritma	45
BAB 7 PENGUJIAN	54
7.1 Pengujian Validasi	54
7.2 Analisis Hasil Pengujian	56
BAB 8 PENUTUP	57
8.1 Kesimpulan	57
8.2 Saran	57
8.3 Keberlanjutan	57
DAFTAR PUSTAKA	58
LAMPIRAN	61
Lampiran A Dokumentasi Presentasi Tugas Website yang dibuat	61

DAFTAR TABEL

Tabel 3.1 Kajian Pustaka	6
Tabel 3.2 Contoh Kode Javascript	7
Tabel 3.3 Contoh Kode ReactJs	7
Tabel 3.5 Notasi pada <i>Use Case Diagram</i>	11
Tabel 3.6 Notasi pada <i>Class Diagram</i>	12
Tabel 3.7 Notasi pada <i>Sequence Diagram</i>	14
Tabel 3.8 Kelebihan Kelemahan Blackbox Testing	15
Tabel 5.1 Kebutuhan Fungsional	21
Tabel 5.2 Kebutuhan Non Fungsional	22
Tabel 5.3 Perencanaan <i>Sprint</i>	22
Tabel 5.4 <i>Use Case Scenario melihat lokasi kapal.</i>	25
Tabel 5.5 <i>Use Case Scenario melihat detail informasi kapal.</i>	25
Tabel 5.6 <i>Use Case Scenario Membatasi data kapal yang ditampilkan.</i>	26
Tabel 5.7 <i>Use Case Scenario Menampilkan data berdasarkan MMSI</i>	26
Tabel 5.8 <i>Use Case Scenario Menampilkan data berdasarkan koordinat</i>	27
Tabel 5.9 <i>Use Case Scenario Menampilkan data berdasarkan MMSI dan koordinat</i>	27
Tabel 5.10 <i>Use Case Scenario Menampilkan tabel MMSI kapal.</i>	28
Tabel 6.1 Kode Halaman Utama menampilkan lokasi kapal	44
Tabel 6.2 Kode melihat detail AIS message	45
Tabel 6.3 Kode Menentukan Limit Data	46
Tabel 6.4 Kode pencarian dengan MMSI	47

Tabel 6.5 Kode pencarian dengan membatasi Koordinat.	48
Tabel 6.6 Kode pencarian dengan membatasi Koordinat dan MMSI tertentu.	50
Tabel 6.7 Kode melihat daftar tabel MMSI.	51
Tabel 6.8 Kode menampilkan posisi kapal.	52
Tabel 7.1 Pengujian Blackbox	53

DAFTAR GAMBAR

Gambar 2.1 Struktur Organisasi Perusahaan	5
Gambar 3.1 Alur Scrum	9
Gambar 3.2 <i>Use Case Diagram</i>	10
Gambar 3.3 Contoh <i>Class Diagram</i>	12
Gambar 3.4 Contoh <i>Sequence Diagram</i>	14
Gambar 3.5 Contoh <i>Blackbox Testing</i>	16
Gambar 4.1 Metode Penelitian	17
Gambar 5.1 <i>Use Case Diagram</i> Aplikasi Pemantauan Lokasi Kapal	24
Gambar 5.2 Sequence Melihat Lokasi Kapal	29
Gambar 5.3 Sequence Melihat Detail Informasi Kapal	29
Gambar 5.4 Sequence Membatasi Data Kapal yang Ditampilkan pada Peta	30
Gambar 5.5 Sequence Menampilkan Data Berdasarkan MMSI	31
Gambar 5.6 Sequence Menampilkan Data Berdasarkan Koordinat	32
Gambar 5.7 Sequence Menampilkan Data Berdasarkan Koordinat dan MMSI	33
Gambar 5.8 Sequence Menampilkan Tabel MMSI Kapal	34
Gambar 5.9 Class Diagram	35
Gambar 6.1 Perancangan Data	36
Gambar 6.2 Halaman Utama menampilkan lokasi kapal	37
Gambar 6.3 Komponen melihat detail informasi kapal	37
Gambar 6.4 Komponen Menentukan Limit Data	38
Gambar 6.5 Komponen pencarian dengan MMSI	38
Gambar 6.6 Komponen pencarian dengan membatasi Koordinat	39
Gambar 6.7 Komponen pencarian dengan membatasi koordinat dan MMSI tertentu	40
Gambar 6.8 Komponen melihat daftar tabel MMSI	41
Gambar 6.9 Komponen menampilkan posisi kapal	42
Gambar 6.10 Implementasi Data	43
Gambar 6.11 Halaman Utama menampilkan lokasi kapal	44
Gambar 6.12 Komponen melihat detail informasi kapal	45

Gambar 6.13 Komponen Menentukan Limit Data	46
Gambar 6.14 Komponen pencarian dengan MMSI	47
Gambar 6.15 Komponen pencarian dengan membatasi Koordinat	48
Gambar 6.16 Komponen pencarian dengan membatasi koordinat dan MMSI tertentu	49
Gambar 6.17 Komponen melihat daftar tabel MMSI	51
Gambar 6.18 Komponen menampilkan posisi kapal	52

DAFTAR LAMPIRAN

Lampiran A Dokumentasi Presentasi Tugas Website yang dibuat	58
--	-----------

BAB 1 PENDAHULUAN

1.1 Latar belakang

Perkembangan teknologi di dunia saat ini begitu pesat, salah satunya adalah bidang maritim dan perkapalan pun juga ikut berkembang. Saat ini dalam pelayanan lalu lintas kapal terdapat sebuah sistem dengan nama AIS atau merupakan singkatan dari *Automatic Identification System*. Automatic Identification System (AIS) adalah suatu sistem pelacakan otomatis yang digunakan pada kapal dan Layanan Pelacakan Kapal (Vessel Traffic Services/VTs) untuk mengidentifikasi dan menemukan kapal dengan bertukar data secara elektronik dengan kapal lain yang berdekatan dan stasiun VTS (Maulidi, 2019). Di Indonesia terdapat lembaga yang bertanggung jawab terhadap data yang didapatkan oleh sistem AIS tersebut yaitu Lembaga Penerbangan dan Antariksa Nasional.

Lembaga Penerbangan dan Antariksa Nasional atau yang biasa disingkat LAPAN adalah sebuah lembaga pemerintah non kementerian di Indonesia yang berfokus pada bidang penelitian dan pengembangan kedirgantaraan dan berbagai macam kemanfaatannya. Empat bidang fokus utama dari LAPAN yakni penginderaan jarak jauh, teknologi dirgantara, sains antariksa, dan kebijakan dirgantara (LAPAN - Lembaga Penerbangan dan Antariksa Nasional, 2020).

LAPAN memiliki satelit A2 yang berfungsi untuk menerima data lokasi kapal dari seluruh dunia menggunakan sistem AIS (*Automatic Identification System*). Data yang diterima masih mentah sehingga perlu diadakan suatu sistem untuk memvisualisasikan data tersebut dalam bentuk peta sehingga mempermudah pengguna data ais dalam memanfaatkan data tersebut.

Produk perangkat lunak adalah perangkat lunak yang digunakan oleh berbagai pengguna, bukan untuk pengguna pribadi saja. Perancangan perangkat lunak merupakan disiplin teknik yang berkaitan dengan pembuatan dan pemeliharaan produk perangkat lunak secara sistematis (Juarna, 2020). Dalam perancangan website beberapa hal yang perlu dipertimbangkan seperti bagaimana membuat desain antarmuka dan kebergunaan suatu website yaitu seberapa jauh website bisa digunakan oleh pengguna agar mencapai tujuan yang diharapkan dengan kepuasan pengguna (Tambuwun, Sengkey and Rindengan, 2020).

Aplikasi Pemantauan Lokasi Kapal Berbasis Data AIS Satelit merupakan sebuah aplikasi berbasis website yang dapat menampilkan sejumlah lokasi kapal dari seluruh dunia. Tujuan aplikasi ini diharapkan dapat membantu pihak LAPAN agar dapat memvisualisasikan data lokasi kapal yang didapatkan dari Satelit LAPAN A2 sehingga berbentuk peta sehingga mempermudah dalam melakukan pemantauan.

Pengembangan aplikasi ini berbasis website karena ditargetkan untuk kemudahan mengakses pada browser bagi para pengguna, dan tidak membatasi target untuk menggunakan aplikasi pada platform mobile saja sehingga tidak perlu repot untuk mendownload aplikasi terlebih dahulu.

1.2 Rumusan masalah

Adapun permasalahan yang muncul pada pengembangan aplikasi pemantauan lokasi kapal ini adalah:

1. Bagaimana merancang dan mengimplementasi sistem pemantauan lokasi kapal dengan data AIS yang didapatkan dari LAPAN?
2. Bagaimana hasil rancangan dan hasil implementasi sistem pemantauan lokasi kapal dengan data AIS yang didapatkan dari LAPAN?
3. Bagaimana hasil pengujian dari sistem pemantauan lokasi kapal dengan data AIS yang didapatkan dari LAPAN?

1.3 Tujuan

Adapun tujuan dari dilaksanakannya Praktek Kerja Lapangan ini adalah :

1. Merancang sistem pemantauan lokasi kapal berbasis data AIS.
2. Mengimplementasi sistem pemantauan lokasi kapal berbasis data AIS.
3. Mengetahui hasil rancangan dari sistem pemantauan lokasi kapal berbasis data AIS.
4. Mengetahui hasil implementasi dari sistem pemantauan lokasi kapal berbasis data AIS
5. Mengetahui hasil pengujian dari sistem pemantauan lokasi kapal berbasis data AIS

1.4 Batasan masalah

Adapun batasan masalah dari penelitian ini adalah sebagai berikut, yaitu:

1. Aplikasi ini dikembangkan pada *platform website* menggunakan bahasa pemrograman JavaScript dengan framework ExpressJS dan NodeJS untuk backend , framework ReactJS untuk frontend dan MongoDB sebagai basis data dengan menggunakan REST API sebagai pengiriman data.
2. Data yang digunakan hanya data AIS pada tanggal 6 Desember 2018 dari pukul 00.00 hingga 23.59
3. Data yang digunakan berjumlah 338.157 data.

1.5 Manfaat

Adapun manfaat yang dihasilkan dari penelitian ini adalah:

1. Bagi LAPAN

- a. Dapat dengan mudah dalam memvisualisasikan data AIS yang didapatkan dari Satelit
- b. Dapat mempermudah melakukan pencarian berdasarkan id kapal maupun berdasarkan jangkauan area tertentu

1.6 Sistematika pembahasan

BAB I : PENDAHULUAN

Berisikan latar belakang masalah, rumusan masalah, tujuan penelitian, manfaat dari penelitian, batasan penelitian, manfaat penelitian, dan sistematika pembahasan skripsi ini.

BAB II : PROFIL OBYEK PKL

Berisikan tentang sejarah singkat perusahaan yang menjadi objek Praktek Kerja Lapangan beserta alamat perusahaan, visi dan misi perusahaan, struktur organisasi perusahaan, struktur organisasi perusahaan, tugas pokok dan fungsi perusahaan, dan rantai bisnis perusahaan.

BAB III : TINJAUAN PUSTAKA

Berisikan tentang kajian kepustakaan terkait yang digunakan sebagai referensi dalam melakukan penelitian.

BAB IV : METODOLOGI PELAKSANAAN PKL

Memuat alur kerja penelitian dalam proses penyelesaian permasalahan penelitian.

BAB V : ANALISIS KEBUTUHAN

Pada tahapan ini, segala kebutuhan dalam Sistem akan dijelaskan secara umum dan lengkap.

BAB VI : PERANCANGAN DAN IMPLEMENTASI

Pada tahap ini backlog yang telah ditentukan sudah dimulai pengerjaannya.

BAB VII : PENGUJIAN

Pada tahap ini, Setiap sprint telah dilakukan, maka akan dilakukan pengujian untuk mendapatkan hasil pada backlog apakah telah terpenuhi atau belum.

BAB VIII : PENUTUP

Pada tahap ini, jika seluruh iterasi sprint yang direncanakan telah terselesaikan, maka akan dilakukan pengambilan kesimpulan dan saran serta keberlanjutan dari pengembangan aplikasi dan pembuatan laporan yang telah dilakukan.

BAB 2 PROFIL OBYEK PKL

2.1 Sejarah Perusahaan

Sejarah Lembaga Penerbangan dan Antariksa Nasional diawali dengan dibentuknya Panitia Astronautika oleh Menteri Pertama RI, Ir. Juanda (selaku Ketua Dewan Penerbangan RI) dan R.J. Salatun (selaku Sekretaris Dewan Penerbangan RI) pada tanggal 31 Mei 1962. Lalu pada 27 November 1963, dibentuklah Lembaga Penerbangan dan Antariksa Nasional (LAPAN) dengan Keputusan Presiden (Keppres) Nomor 236 Tahun 1963 tentang LAPAN, untuk melembagakan penyelenggaraan program-program pembangunan kedirgantaraan nasional. Dalam hal penyempurnaan organisasi LAPAN, telah dikeluarkan beberapa Keppres, dengan yang terkini yakni Keppres Nomor 9 Tahun 2004 tentang Lembaga Non-Kementerian. LAPAN berfokus pada bidang penelitian dan pengembangan kedirgantaraan dan berbagai macam kemanfaatannya (LAPAN - Lembaga Penerbangan dan Antariksa Nasional, 2020).

2.2 Alamat Perusahaan

Jl. Pemuda Persil No. 1 Jakarta 13220

2.3 Visi dan Misi Perusahaan

2.3.1 Visi

Menjadi Penggerak Sektor - Sektor Pembangunan Nasional Berbasis IPTEK Penerbangan dan Antariksa Dalam Mewujudkan Visi Misi Presiden dan Wakil Presiden Indonesia Maju Yang Berdaulat, Mandiri, dan Berkepribadian Berlandaskan Gotong Royong.

2.3.2 Misi

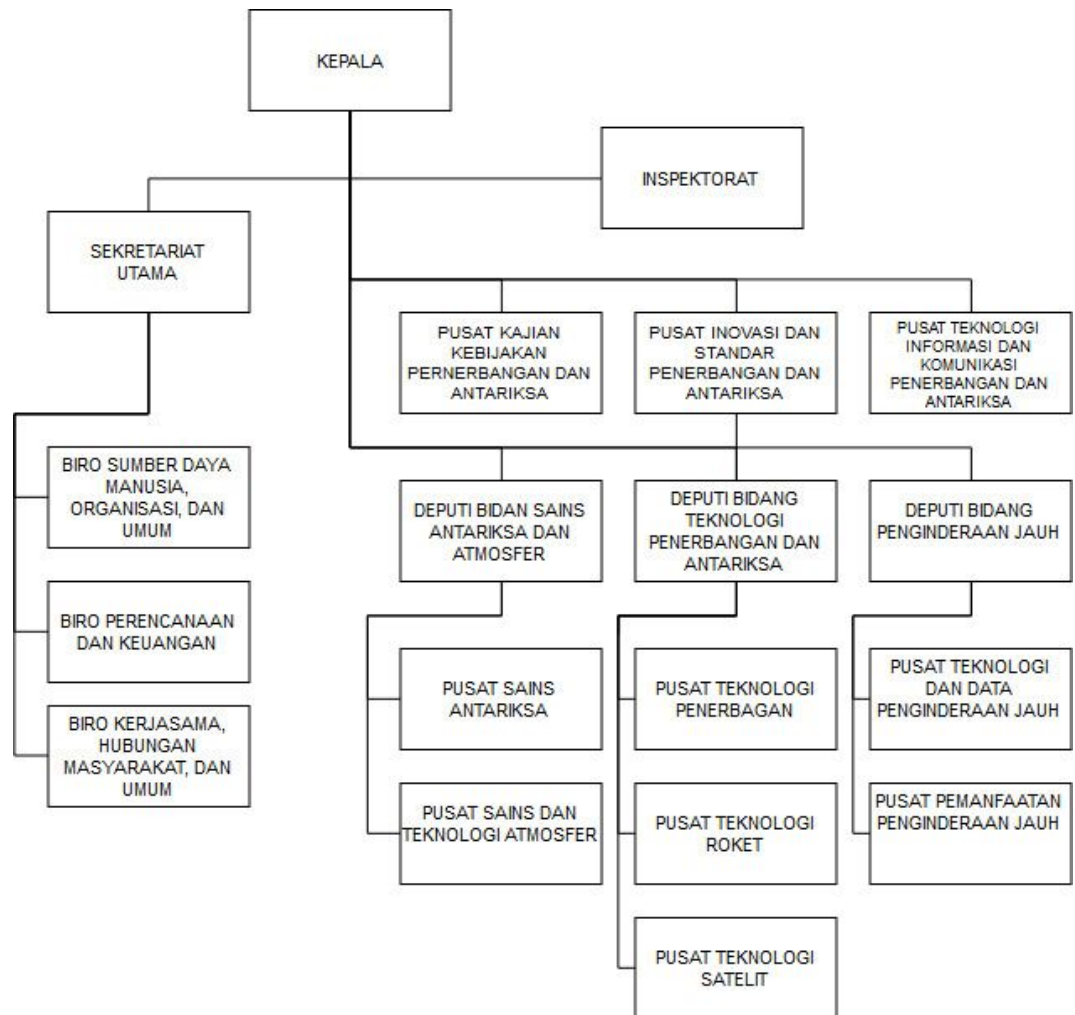
1. Memberikan dukungan teknis dan administrasi serta analisis yang cepat, akurat dan responsif kepada Presiden dan Wakil Presiden dalam pengambilan kebijakan penyelenggaraan Pemerintahan Negara berkaitan dengan sains antariksa dan atmosfer.
2. Memberikan dukungan teknis dan administrasi serta analisis yang cepat, akurat dan responsif kepada Presiden dan Wakil Presiden dalam pengambilan kebijakan penyelenggaraan Pemerintahan Negara berkaitan dengan penginderaan jauh.
3. Memberikan dukungan teknis dan administrasi serta analisis yang cepat, akurat dan responsif kepada Presiden dan Wakil Presiden dalam pengambilan kebijakan penyelenggaraan Pemerintahan Negara berkaitan dengan teknologi penerbangan dan antariksa.
4. Memberikan dukungan teknis dan administrasi serta analisis yang cepat, akurat dan responsif kepada Presiden dan Wakil Presiden dalam

pengambilan kebijakan penyelenggaraan Pemerintahan Negara berkaitan dengan kebijakan penerbangan dan antariksa.

5. Memberikan dukungan teknis dan administrasi serta analisis yang cepat, akurat dan responsif kepada Presiden dan Wakil Presiden dalam pengambilan kebijakan penyelenggaraan Pemerintahan Negara berkaitan dengan mewujudkan birokrasi LAPAN berkelas dunia.

2.4 Struktur Organisasi Perusahaan

Adapun struktur organisasi dari Lembaga Penerbangan dan Antariksa Nasional ini terlampir pada Gambar 2.1:



Gambar 2.1 Struktur Organisasi Perusahaan

Pada penelitian ini, kami berada pada posisi Staff IT di dalam Pusat Teknologi Satelit.

BAB 3 TINJAUAN PUSTAKA

3.1 Kajian Pustaka

Kajian pustaka ini digunakan sebagai pembanding antara penelitian yang sudah dilakukan dan yang akan dilakukan kelompok PKL. Penelitian tersebut di antaranya terlampir pada Tabel 3.1.

Tabel 3.1 Kajian Pustaka

No	Nama Penulis dan Judul	Persamaan	Perbedaan	
			Penelitian Terdahulu	Rencana Penelitian
1	Ahmad Rifai. Sistem Informasi Pemantauan Posisi Kendaraan Dinas Unsri Menggunakan Teknologi GPS	Aplikasi berbasis web untuk memantau lokasi.	Hanya berupa perancangan	Diterapkan hingga implementasi.
2	Lukas Tanutama. Pemantau Lokasi Benda Bergerak Berbasis Web dengan menggunakan Teknologi GPS dan 3G	Aplikasi berbasis web untuk memantau lokasi.	Menggunakan Apache sebagai Web Server dan MySQL untuk database	Menggunakan NodeJS sebagai Web Server dan MongoDB NoSQL untuk database
3	Roly Segara. Sistem Pemantauan Lokasi Anak Menggunakan Metode Geofencing pada Platform Android	Aplikasi berbasis web untuk memantau lokasi.	Aplikasi dirancang dengan bahasa PHP dan menggunakan CodeIgniter	Aplikasi dirancang dengan bahasa Javascript dengan menggunakan ReactJS

3.2 Kajian Teori

3.2.1 Javascript

Bahasa pemrograman yang banyak digunakan dalam mengembangkan suatu website. JavaScript bisa digunakan di berbagai perangkat dan browser

sehingga JavaScript merupakan salah bahasa pemrograman yang sangat didukung (Ariata, 2019).

Javascript digunakan karena merupakan salah satu bahasa pemrograman yang bersifat multiplatform dan mudah untuk dikembangkan. Berikut potongan kode javascript tertera pada Tabel 3.2.

Tabel 3.2 Contoh Kode Javascript

1	<script>
2	console.log("Saya belajar Javascript");
3	document.write("Hello World!");
4	</script>

3.2.2 ReactJS

Sebuah UI *library* yang dikembangkan oleh Facebook yang membantu developer dalam membuat UI yang interaktif. React bukanlah sebuah framework MVC. React adalah library yang bersifat composable user interface, yang artinya kita dapat membuat berbagai UI yang bisa kita bagi menjadi beberapa komponen (Musa, 2017).

ReactJS dipilih karena sifatnya yang cepat, efisien, dan bersifat *reusable*. Berikut contoh kode ReactJS tertera pada Tabel 3.3

Tabel 3.3 Contoh Kode ReactJs

1	import React from 'react';
2	import ReactDOM from 'react-dom';
3	
4	function Example() {
5	return (
6	<div className="container">
7	<div className="row justify-content-center">
8	<div className="col-md-8">
9	<div className="card">
10	<div className="card-header">Example Com
11	<div className="card-body">I'm an exampl
12	e component!</div>
13	</div>
14	</div>
15	</div>
16	</div>
17);
18	}
19	

3.2.3 MongoDB

Sebuah database yang menggunakan konsep NoSQL berbasis dokumen. Sistem penyimpanan datanya tidak menggunakan table akan tetapi menggunakan dokumen terstruktur layaknya JSON. MongoDB database ini juga memanfaatkan Javascript dalam mengoperasikan indexing, agregasi, CRUD, serta berbagai operasi database lainnya. MongoDB dipilih karena konsepnya yang NoSQL sehingga memudahkan developer dalam penggunaannya (Menenal Apa itu MongoDB : Manfaat, Keunggulan dan Cara Kerja MongoDB [LENGKAP], 2020).

3.2.4 NodeJS

Node Js merupakan open-source, cross-platform runtime environment yang digunakan pada sisi server dari aplikasi website. NodeJs ditulis dalam JavaScript dan dapat dijalankan berbagai macam sistem operasi. NodeJS dapat menggunakan database NoSQL seperti MongoDB (Ricky et al, 2019).

3.2.5 Express JS

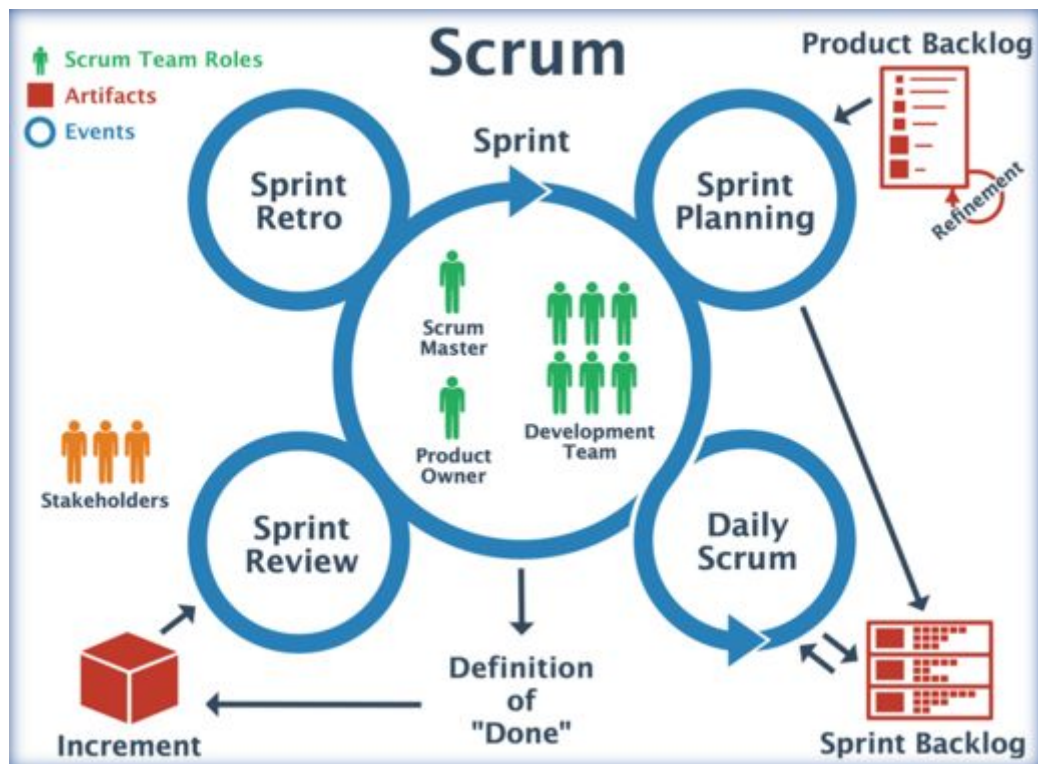
Express Js merupakan framework yang berada di atas fungsi server web Node.js untuk menyederhanakan API dan menambahkan fitur baru yang bermanfaat. Membuat lebih mudah untuk mengatur fungsionalitas aplikasi dengan middleware dan routing (Ricky et al, 2019)..

3.2.6 Scrum

Scrum merupakan sebuah proses kerangka kerja yang biasa digunakan untuk mengatur pekerjaan seperti membangun perangkat lunak pada proyek yang kompleks sehingga dapat bekerja dengan cepat karena scrum menggunakan metode iteratif dimana tiap dalam iteratifnya dilakukan beberapa fase tertentu (Scrum Methodology: Panduan Project Management, 2017).

Karena scrum bersifat iteratif maka suatu pekerjaan dapat dipecah menjadi beberapa bagian dan menjadikannya sebuah sprint yang mana disusun berdasarkan prioritas yang sudah ditentukan sebelumnya. Di tiap akhir sprint akan dilakukan pengujian dan evaluasi untuk mendapatkan feedback sehingga menjadi pelajaran untuk sprint berikutnya

Mengembangkan perangkat lunak dengan menggunakan metode Scrum terbagi menjadi beberapa alur tahapan yang digambarkan pada Gambar 3.1:



Gambar 3.1 Alur Scrum

Sumber : (Beat the drum with Scrum!! 2020,
<https://medium.com/@poojamahajan5131>)

1. Sprint
 Sprint adalah jadwal kerja yang fokus pada pembuatan produk berdasarkan dengan Product Backlog
2. Sprint Planning
 Sprint Planning adalah proses perencanaan sprint yang dimana backlog yang sudah ditentukan sebelumnya akan menjadi sprint. Sprint memiliki durasi tertentu sesuai tingkat prioritas yang ditandai dalam bentuk point. Pada umumnya sprint akan berjalan selama sekitar 14 hari atau 2 minggu.
3. Daily Scrum
 Daily Scrum bermaksud bahwa setiap tim akan melakukan rapat untuk memberitahu progress, masalah, sehingga dapat menyelesaikan masalah tersebut bersama.
4. Sprint Review
 Pada akhir satu sprint, akan dilakukan presentasi mengenai fitur yang sudah dikerjakan.
5. Sprint Retrospective

Pada tahap ini, tim scrum akan menjelaskan atau mengevaluasi bagaimana pekerjaan sprint-sprint sebelumnya sehingga dapat memperbaiki untuk sprint berikutnya.

6. Product Backlog

Product Backlog adalah suatu daftar fitur-fitur yang disusun berdasarkan prioritas yang dibutuhkan dan akan diimplementasikan nanti pada saat pengembangan.

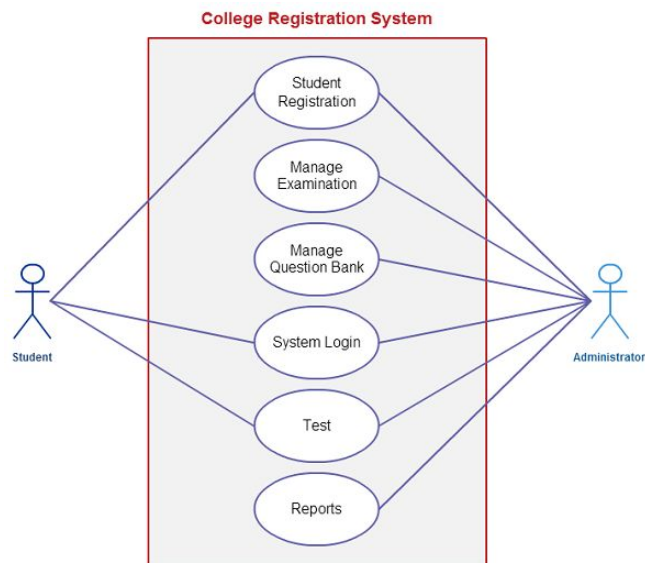
7. Sprint Backlog

Item product backlog yang terpilih untuk sprint.

Scrum dipilih karena belum matangnya ide dan desain aplikasi pada awal dalam pembuatan aplikasi ini sehingga scrum dipilih karena cocok untuk menangani perubahan di tiap sprintnya.

3.2.7 Use Case Diagram

Use Case Diagram adalah gambaran grafis dari beberapa atau semua actor, use case, dan interaksi diantaranya yang memperkenalkan suatu sistem. Use case diagram tidak menjelaskan secara detail tentang penggunaan use case, tetapi hanya memberi gambaran singkat hubungan antara use case, aktor, dan sistem. Di dalam use case ini akan diketahui fungsi-fungsi apa saja yang berada pada sistem yang dibuat (Pratama, 2019). Berikut merupakan contoh dari *use case diagram* pada gambar 3.2.

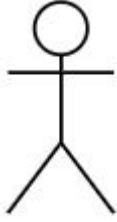
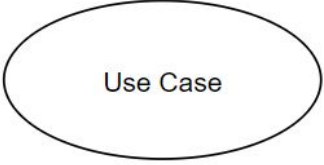

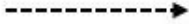
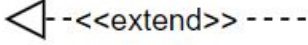



Gambar 3.2 Use Case Diagram

Sumber : (Use Case Template 2018,
<https://creately.com/blog/examples/use-case-templates-uml/>)

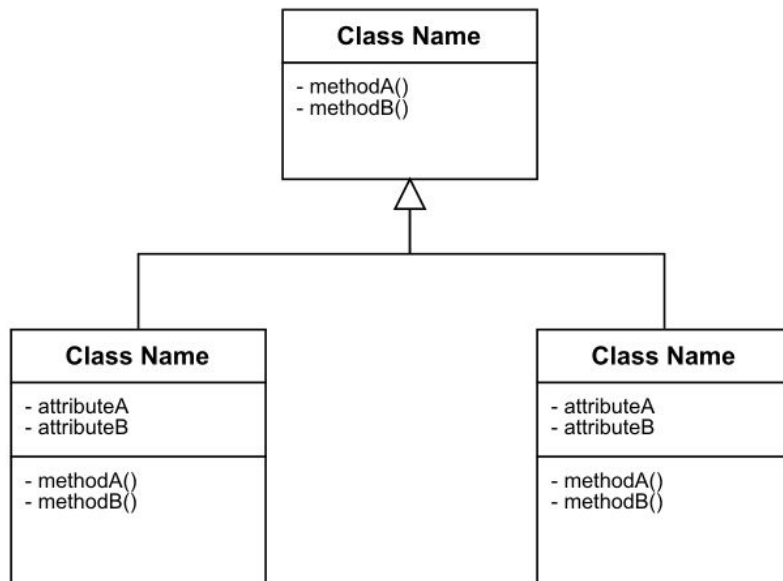
Berikut beberapa notasi yang bisa digunakan pada *use case diagram* ditunjukkan pada Tabel 3.5.

Tabel 3.5 Notasi pada *Use Case Diagram*

No	Tipe	Deskripsi	Simbol
1	<i>Actor</i>	Menggambarkan seseorang atau sesuatu yang berinteraksi dengan sistem.	
2	<i>Use case</i>	Menggambarkan fungsionalitas dari suatu sistem, yang akan mempermudah pengguna dalam memahami fungsionalitas suatu sistem	
3	<i>Asociation</i>	Penghubung antar element seperti antara aktor dan use case	
4	Generalisasi	Menggambarkan spesialisasi aktor untuk berpartisipasi dalam suatu use case	
5	<i>extend</i>	Dilakukan ketika use case menambahkan langkah ke use case lainnya.	
6	<i>include</i>	Kondisi dimana sebuah use case adalah bagian dari use case lainnya sehingga use case tersebut harus dijalankan agar suatu event dapat dilakukan	

3.2.8 Class Diagram

Class diagram merupakan penjelasan proses database dalam suatu program. Class diagram menggambarkan struktur dan deskripsi class, package dan objek serta hubungan satu sama lain seperti containment, pewarisan, asosiasi, dan lain-lain ([RPL2] Class Diagram dan Relasinya (2), 2020).

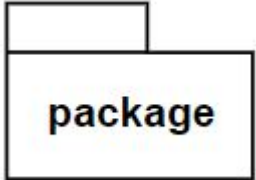
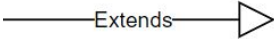
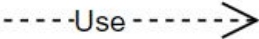



Gambar 3.3 Contoh *Class Diagram*

Berikut beberapa contoh notasi yang terdapat pada class diagram ditunjukkan pada Tabel 3.6.

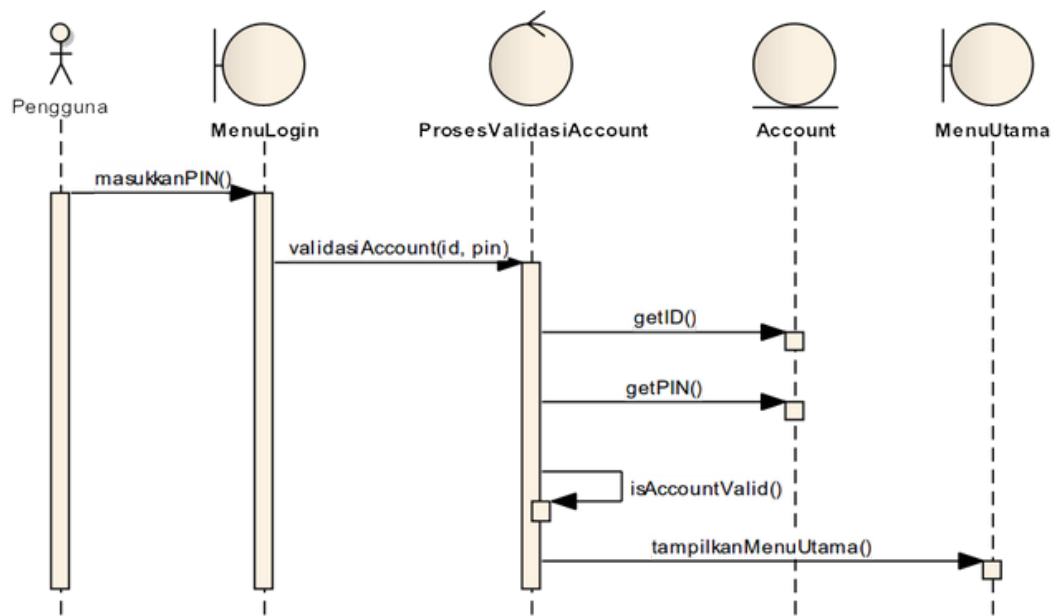
Tabel 3.6 Notasi pada *Class Diagram*

No	Tipe	Deskripsi	Simbol					
1	<i>Class</i>	Menggambarkan sebuah objek. Class dibagi menjadi nama, atribut, dan metode.	<table><tr><th>Nama Class</th></tr><tr><td>+ atribut</td></tr><tr><td>+ atribut</td></tr><tr><td>+ method</td></tr><tr><td>+ method</td></tr></table>	Nama Class	+ atribut	+ atribut	+ method	+ method
Nama Class								
+ atribut								
+ atribut								
+ method								
+ method								
2	<i>Association</i>	Menggambarkan hubungan antara 2 class, dan juga dapat menandakan hukum multiplisitas seperti <i>one-to-one</i> , <i>one-to-many</i> , dan <i>many-to-many</i> .	<table><tr><td>parent</td><td>child</td></tr></table>	parent	child			
parent	child							
3	<i>Package</i>	Kumpulan kelas yang dibungkus menjadi sesuatu bernama package						

			
4	<i>Generalization</i>	Menggambarkan hubungan pewarisan dari kelas utama ke kelas anak.	
5	<i>Dependency</i>	Menggambarkan suatu kelas yang membutuhkan kelas lain sehingga bersifat ketergantungan.	
6	<i>Agregation</i>	Menggambarkan keseluruhan bagian relationship atau relasi	

3.2.9 Sequence Diagram

Sequence diagram digunakan untuk menggambarkan perilaku sebuah skenario. Kegunaannya untuk menunjukan rangkaian pesan yang dikirim antara objek juga interaksi antara objek, sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem (Sistem Informasi, 2020).





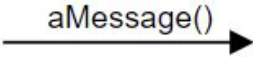
Gambar 3.4 Contoh Sequence Diagram

Sumber : (Sequence Diagram in UML 2015,
<https://rollypp.wordpress.com/2015/04/13/sequence-diagram-in-uml/>)

Berikut beberapa notasi yang ada di sequence diagram ditunjukkan pada Tabel 3.7.

Tabel 3.7 Notasi pada Sequence Diagram

No	Tipe	Deskripsi	Simbol
1	<i>Actor</i>	Notasi yang digunakan untuk menggambarkan seseorang (<i>user</i>) atau sistem lain yang menggunakan fitur atau fungsi yang berada dalam sistem.	
2	<i>Object</i>	Notasi yang digunakan untuk menggambarkan <i>object</i> yang berpartisipasi dalam sebuah <i>sequence</i> dengan mengirim pesan atau menerima pesan.	
3	<i>Lifeline</i>	Notasi yang digunakan untuk menunjukkan lama hidup suatu <i>object</i> dalam <i>sequence</i> .	

			
4	<i>Focus Control</i>	Notasi yang digunakan untuk menunjukkan bahwa sebuah <i>object</i> sedang aktif dan menjalankan tugasnya.	
5	<i>Message</i>	Notasi yang digunakan untuk menunjukkan bahwa sebuah pesan sedang dikirim dari suatu <i>object</i> ke <i>object</i> yang lain.	

3.2.10 Pengujian

Dalam perangkat lunak pengujian merupakan hal yang wajib dilakukan. Tujuan utama dari pengujian perangkat lunak sebenarnya sederhana yaitu untuk memastikan bahwa *software* yang dihasilkan sesuai dengan kebutuhan (*requirement*) yang sebelumnya ditentukan. Ketika *requirement* dari suatu sistem telah disusun maka semestinya sudah ada suatu pengujian perencanaan (*test plan*). Selain itu suatu proses testing membutuhkan tujuan akhir yang dapat dinilai sehingga pihak tester bisa berhenti melakukan suatu testing ketika tujuan-tujuan itu tercapai (Pentingnya Melakukan Pengujian Perangkat Lunak (Software Testing) – Software Engineering ITTP, 2020). Dalam proses ini kami menggunakan Blackbox Testing sebagai metode validasi dalam pengujian perangkat lunak.

3.2.11 Blackbox Testing

Blackbox Testing adalah salah satu metode validasi pengujian perangkat lunak yang dimana penguji tidak mengetahui bagaimana struktur atau desain internal kode atau program tersebut. Pengujian BlackBox adalah pengujian aspek fundamental sistem tanpa memperhatikan struktur logika internal perangkat lunak. Metode ini digunakan untuk mengetahui apakah perangkat lunak berfungsi dengan benar. Pengujian Black Box merupakan metode perancangan data uji yang didasarkan pada spesifikasi perangkat lunak (PENGUJIAN PERANGKAT LUNAK MENGGUNAKAN METODE BLACK BOX STUDI KASUS EXELSA, 2020).

Tabel 3.8 Kelebihan Kelemahan Blackbox Testing

Kelebihan	Kelemahan
-----------	-----------

Akses kode tidak diperlukan sehingga efisien untuk segmen kode besar	Cakupannya terbatas karena hanya sebagian kecil dari skenario pengujian yang dilakukan
Sebagai pemisah antara perspektif pengguna dan pengembang	Ketergantungan terhadap pengetahuan perangkat lunak user sehingga kurang efisien

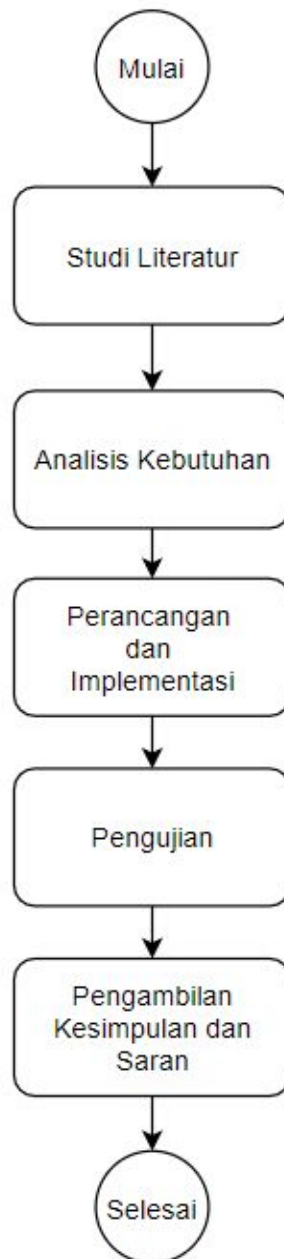
Berikut merupakan contoh dari blackbox testing ditunjukkan pada Gambar 3.5

TC ID	TC Name	Description	Shape	Expected Result
TC1	TC1_FileUpdate Scenario1	Validate that system updates the file when first character is A and second character is a digit.	1. Open the application 2. Enter the first character as "A" 3. Enter the second character as a digit	File is updated.
TC2	TC2_FileUpdateScenario2	Validate that system updates the file when first character is B and second character is a digit.	1. Open the application 2. Enter the first character as "B" 3. Enter second character as a digit	File is updated.

Gambar 3.5 Contoh *Blackbox Testing*

BAB 4 METODOLOGI PELAKSANAAN PKL

Dalam mengembangkan sebuah perangkat lunak, metode pasti sangat dibutuhkan agar proses pengembangan dapat terstruktur dengan baik. Dalam pengembangannya, penelitian ini menggunakan *Scrum*. Alur diagram metodologi dari penelitian ini digambarkan pada Gambar 4.1.



Gambar 4.1 Metode Penelitian

4.1 Studi Literatur

Studi Literatur merupakan daftar dasar teori pendukung dalam penelitian. Dasar teori tersebut berasal dari buku, artikel, jurnal, website dan dokumentasi framework yang terkait atau digunakan dalam penelitian ini. Teori pendukung tersebut diantaranya:

1. Javascript
2. ReactJS
3. MongoDB
4. NodeJS
5. ExpressJS
6. Scrum
7. Use Case Diagram
8. Class Diagram
9. Sequence Diagram
10. Pengujian
11. Blackbox Testing

4.2 Analisis Kebutuhan

Pada tahapan ini, segala kebutuhan dalam Sistem akan dijelaskan secara umum dan lengkap. Tahapan ini terdiri dari pembuatan Product Backlog dan Perencanaan Sprint.

4.2.1 Pembuatan Product Backlog

Dalam fase ini, product owner akan membuat suatu daftar yang harus dikerjakan tim dan apa-apa saja kebutuhannya yang berbentuk list fitur yang harus dikerjakan yang diurutkan secara prioritas dan yang mana harus dikerjakan terlebih dahulu.

4.2.2 Perencanaan Sprint

Dalam fase ini, dilakukannya pendefinisian terhadap apa yang bisa dikerjakan saat sprint tersebut dan bagaimana cara menyelesaikannya. Perencanaan sprint dapat diselesaikan dengan kolaborasi dari seluruh tim scrum.

4.3 Perancangan dan Implementasi

Pada tahap ini backlog yang sudah dibahas dan ditetapkan sebelumnya akan dimulai pengerjaannya. Setelah sprint selesai dilakukan maka akan dilanjutkan untuk mempresentasikan apa saja backlog yang sudah dibuat pada meeting selanjutnya.

4.3.1 Perancangan

Pada tahap ini akan dilakukannya perancangan sesuai daftar fitur yang sudah ditentukan. Tiap fiturnya akan dibuat dalam bentuk use case diagram yang mana berfungsi bagaimana suatu pengguna atau aktor berinteraksi dengan sistem. Setelah use case diagram selesai maka akan dilanjutkan pembuatan use case scenario untuk merincikan bagaimana tahapan-tahapan dari suatu use case diagram. Setelah use case scenario selesai lalu dilanjutkan dengan pembuatan sequence diagram untuk perancangan algoritma dari use case scenario.

Setelah itu perancangan dilanjutkan merancang antarmuka agar dapat membantu dalam mengimplementasikan antarmuka yang nantinya akan dibuat dalam bentuk aplikasi website.

4.3.2 Implementasi

Pada tahap ini akan dilakukannya implementasi dari perancangan yang sudah dibuat sebelumnya. Proses implementasi dikerjakan berdasarkan diagram-diagram sebelumnya seperti use case dan sequence. Setelah itu implementasi akan dimulai dan untuk aplikasi website ini memilih menggunakan Javascript sebagai bahasa pemrograman dan menggunakan ExpressJS dan NodeJS dalam pembuatan backendnya yang pengiriman datanya akan menggunakan REST API. Untuk frontendnya akan menggunakan ReactJS dan untuk penyimpanannya datanya akan menggunakan MongoDB.

4.4 Pengujian

Pada tahap ini akan dilakukan pengujian di setiap selesainya satu sprint untuk mengecek dan menguji suatu kualitas hasil backlog yang dikerjakan apakah sudah memenuhi kebutuhan atau belum. Proses pengujian ini menjadi penentu apakah backlog yang dikerjakan sudah selesai atau harus dimasukkan ke backlog kembali untuk dikerjakan di sprint selanjutnya.

4.4.1 Pengujian dan Demonstrasi Produk

Pada tahap ini akan dilakukan pengujian apakah sudah memenuhi kebutuhan atau belum. Pengujiannya dapat dilakukan dengan metode blackbox testing.

4.4.2 Retrospektif

Pada tahap ini akan dilakukan pengevaluasian dari sprint yang telah dilakukan. Setelah dilakukannya evaluasi maka akan menghasilkan suatu kesimpulan dari sprint yang telah dilakukan.

4.4.3 Perencanaan Sprint Selanjutnya

Pada tahapan ini akan dilakukan perencanaan sprint selanjutnya yang mengacu kepada kesimpulan sprint sebelumnya untuk dijadikan pertimbangan.

4.5 Pengambilan Kesimpulan dan Saran (Penutup)

Pada tahapan ini akan dilakukan pengambilan kesimpulan setelah seluruh iterasi sprint selesai yang diambil dari evaluasi dan analisa hasil pengujian dari tiap iterasi. Dengan itu maka akan terbentuknya saran agar tidak mengulangi kesalahan atau kekurangan yang sama di sprint sebelumnya.

BAB 5 ANALISIS KEBUTUHAN

5.1 Deskripsi Umum Sistem

Aplikasi website pemantauan lokasi kapal yang dibuat tim merupakan website yang bertugas untuk memvisualisasikan data koordinat kapal di database kedalam sebuah peta dunia yang dapat diakses melalui laman web. Beberapa fitur untuk pencarian data kapal bisa didapat dengan pencarian berdasarkan MMSI yang merupakan singkatan dari *Maritime Mobile Service Identity*. Selain dari pencarian berdasarkan MMSI, pengguna dapat mencari kapal pada daerah koordinat tertentu menggunakan nilai *longitude* dan *latitude*.

5.2 Deskripsi Umum Perangkat Lunak

Perangkat lunak pada sistem pemantauan lokasi kapal ini dikembangkan berbasis website dengan Javascript sebagai bahasa pemrograman utamanya dan menggunakan framework ExpressJS dan NodeJS sebagai backend, ReactJS sebagai frontend dan MongoDB sebagai database atau dikenal dengan metode pengembangan MERN Stack. Dalam sistem ini hanya ada 1 aktor yaitu pengguna yang dapat melihat posisi kapal serta informasi-informasinya. Dalam penggunaannya tidak ada fungsi login ataupun logout sehingga pengguna bisa langsung menggunakan sistem.

5.3 Pembuatan Backlog Product

Dalam pembuatan backlog product terjadi diskusi antara tim mahasiswa dengan pembimbing PKL dari pihak LAPAN mengenai sistem dan fitur apa saja yang diperlukan dalam website pemantauan lokasi kapal dengan data AIS ini. Dalam pembuatannya fungsional diberikan kode SRS_PLK_F_X. SRS merupakan singkatan dari *System Requirement Specification*, PLK singkatan dari Pemantauan Lokasi Kapal, dan F merupakan singkatan dari Fungsional. Lalu untuk *product backlog* diberikan kode PB_X. Kebutuhan fungsional adalah kebutuhan yang berisi proses-proses apa saja yang nantinya dilakukan oleh sistem. Berikut merupakan hasil diskusi kami yang menghasilkan suatu kebutuhan fungsional berupa *backlog product* yang ditunjukkan pada Tabel 5.1.

Tabel 5.1 Kebutuhan Fungsional

No	Kode Fungsional	Aktor	Definisi Kebutuhan	Kode Backlog Product
1	SRS_PLK_F_01	User	Pengguna dapat melihat lokasi kapal	PB_01

2	SRS_PLK_F_02	User	Pengguna dapat melihat detail informasi kapal	PB_02
3	SRS_PLK_F_03	User	Pengguna dapat membatasi jumlah data kapal yang ditampilkan pada peta	PB_03
4	SRS_PLK_F_04	User	Pengguna dapat menampilkan data berdasarkan MMSI	PB_04
5	SRS_PLK_F_05	User	Pengguna dapat menampilkan data berdasarkan Koordinat	PB_05
6	SRS_PLK_F_06	User	Pengguna dapat menampilkan data berdasarkan Koordinat dan MMSI	PB_06
7	SRS_PLK_F_07	User	Pengguna dapat menampilkan tabel MMSI	PB_07

Kebutuhan non fungsional adalah kebutuhan yang menitikberatkan pada properti perilaku yang dimiliki oleh sistem, berikut merupakan kebutuhan non fungsional dari sistem pemantauan lokasi kapal yang ditunjukkan pada Tabel 5.2.

Tabel 5.2 Kebutuhan Non Fungsional

No	Kebutuhan	Deskripsi
1	Usability	Sistem hanya menyediakan dalam satu Bahasa saja.
2	Availability	Sistem dapat diakses selama 24 jam
3	Portability	Sistem hanya dapat diakses di browser Google Chrome, Safari, Mozilla Firefox dan Internet Explorer.

5.4 Perencanaan Sprint

Setelah dibuat daftar *backlog product*, langkah selanjutnya akan dibuat perencanaan *sprint* untuk menentukan *task* atau tugas yang akan dikerjakan setiap harinya dalam bentuk Tabel. Isi dari Tabel ini meliputi, job yang akan dikerjakan ditulis dengan kode *backlog product* dan estimasi waktu pengerjaan setiap harinya. Berikut merupakan susunan sprint yang kami buat ditunjukkan pada Tabel 5.3.

Tabel 5.3 Perencanaan *Sprint*

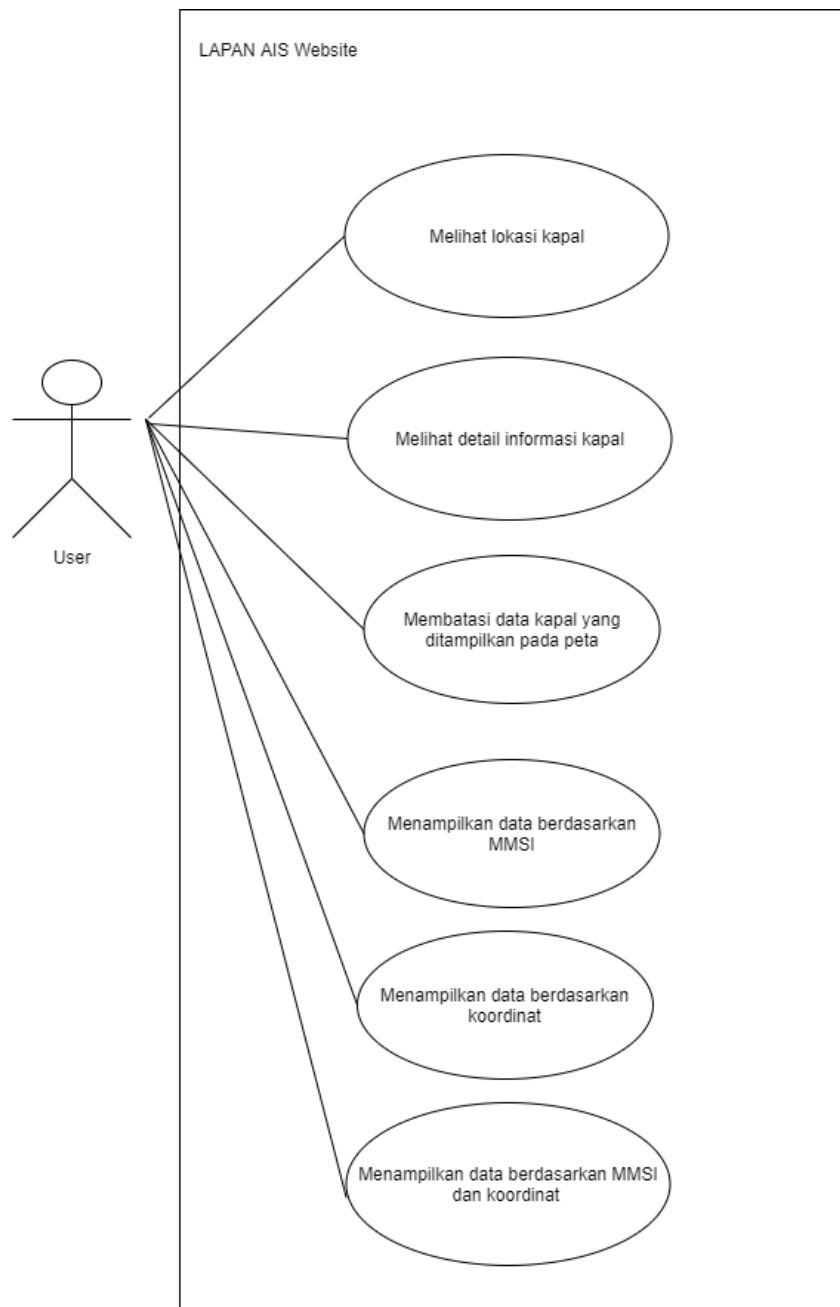
No	Job (Kode Backlog Product)	Minggu -1	Minggu-2	Minggu -3	Minggu-4	Minggu -5
1	PB_01, PB_02	7				

2	PB_03		6			
3	PB_04			6		
4	PB_05, PB_06				7	
5	PB_07					6

5.5 Analisa Kebutuhan

Analisa kebutuhan meliputi Use Case Diagram, Use Case Scenario, Sequence Diagram, dan Class Diagram.

5.5.1 Use Case Diagram



Gambar 5.1 Use Case Diagram Aplikasi Pemantauan Lokasi Kapal

Gambar 5.1 diatas memperjelas interaksi dari pengguna kepada aplikasi website yang digunakan. Interaksi pada pengguna yang dapat dilakukan pada aplikasi dideskripsikan pada gambar tersebut.

5.5.2 Use Case Scenario

Use Case Scenario digunakan untuk menjelaskan diagram use case secara lebih detail. Use Case Scenario ini berfungsi untuk mendeskripsikan suatu fungsi secara menyeluruh mengenai kondisi awal, use case dan kondisi akhir yang harus dijalani. Setelah fungsi dijalankan akan menjelaskan respon sistem terhadap aktor. Berikut merupakan use case scenario dari aplikasi pemantauan kapal yang ditunjukkan pada Tabel 5.4 - 5.9.

Tabel 5.4 Use Case Scenario melihat lokasi kapal.

<i>Use case melihat lokasi kapal.</i>	
Tujuan	Melihat lokasi kapal
Aktor	<i>User.</i>
Kondisi awal	Halaman index.
Skenario Utama	1. Sistem akan menampilkan lokasi kapal dalam bentuk peta
Skenario Alternatif	-
Kondisi akhir	Sistem akan menampilkan lokasi kapal dalam bentuk peta

Tabel 5.5 Use Case Scenario melihat detail informasi kapal.

<i>Use case melihat detail informasi kapal.</i>	
Tujuan	Melihat detail informasi kapal
Aktor	<i>User.</i>
Kondisi awal	Halaman index.
Skenario Utama	<ol style="list-style-type: none"> 1. Sistem akan menampilkan lokasi kapal dalam bentuk peta 2. Pengguna menekan salah satu icon kapal 3. Sistem akan menampilkan detail informasi kapal yang ditekan

Skenario Alternatif	-
Kondisi akhir	Sistem akan menampilkan detail informasi kapal yang ditekan

Tabel 5.6 Use Case Scenario Membatasi data kapal yang ditampilkan.

<i>Use case membatasi data kapal yang ditampilkan.</i>	
Tujuan	Menambah/mengurangi data kapal yang ditampilkan pada peta.
Aktor	<i>User.</i>
Kondisi awal	Halaman index.
Skenario Utama	<ol style="list-style-type: none"> 2. <i>User</i> menekan tombol “Number of Data Display Limit”. 3. Sistem menampilkan menu <i>drop-down</i>. 4. <i>User</i> mengisi pada kolom yang disediakan. 5. Sistem memvalidasi hasil masukan <i>User</i>. 6. Sistem menampilkan hasil yang diinginkan <i>User</i> pada peta.
Skenario Alternatif	<ol style="list-style-type: none"> 1. Jika kolom kosong atau mempunyai karakter non-numerikal, maka sistem akan mengeluarkan peringatan.
Kondisi akhir	Sistem menampilkan jumlah data lokasi kapal sesuai dengan jumlah yang dimasukkan oleh <i>User</i> .

Tabel 5.7 Use Case Scenario Menampilkan data berdasarkan MMSI

<i>Use case Menampilkan Data berdasarkan MMSI.</i>	
Tujuan	Menampilkan lokasi kapal berdasarkan MMSI.
Aktor	<i>User.</i>
Kondisi awal	Halaman index.

Skenario Utama	<ol style="list-style-type: none"> 1. <i>User</i> menekan tombol “<i>Search by MMSI</i>” 2. Sistem menampilkan menu <i>drop-down</i>. 3. <i>User</i> mengisi MMSI pada kolom yang disediakan. 4. Sistem memvalidasi isian dari <i>User</i>. 5. Sistem menampilkan hasil yang diinginkan <i>User</i> pada peta.
Skenario Alternatif	<ol style="list-style-type: none"> 1. Jika kolom kosong atau mempunyai karakter non-numerikal, maka sistem akan mengeluarkan peringatan.
Kondisi akhir	Menampilkan data lokasi kapal yang diinginkan pengguna pada peta.

Tabel 5.8 Use Case Scenario Menampilkan data berdasarkan koordinat

<i>Use case menampilkan data berdasarkan koordinat.</i>	
Tujuan	Menampilkan lokasi kapal berdasarkan koordinat.
Aktor	<i>User</i> .
Kondisi awal	Halaman index.
Skenario Utama	<ol style="list-style-type: none"> 1. <i>User</i> mengklik tombol “<i>Filter by Coordinates</i>”. 2. Sistem menampilkan menu <i>drop-down</i>. 3. <i>User</i> mengisi koordinat <i>longitude</i> dan <i>latitude</i> pada kolom yang disediakan. 4. Sistem memvalidasi isian dari <i>User</i>. 5. Sistem menampilkan hasil yang diinginkan <i>User</i> pada peta.
Skenario Alternatif	<ol style="list-style-type: none"> 1. Jika kolom kosong atau mempunyai karakter non-numerikal, maka sistem akan mengeluarkan peringatan.
Kondisi akhir	Menampilkan data lokasi kapal pada daerah tertentu yang diinginkan pengguna pada peta.

Tabel 5.9 Use Case Scenario Menampilkan data berdasarkan MMSI dan koordinat

<i>Use case menampilkan data berdasarkan MMSI dan koordinat.</i>
--

Tujuan	Menampilkan lokasi kapal berdasarkan koordinat dan MMSI
Aktor	<i>User.</i>
Kondisi awal	Halaman index.
Skenario Utama	<ol style="list-style-type: none"> 1. <i>User</i> menekan tombol “Filter By Coordinates And MMSI”. 2. Sistem menampilkan menu <i>drop-down</i>. 3. <i>User</i> mengisi pada kolom yang disediakan. 4. Sistem memvalidasi hasil masukan <i>User</i>. 5. Sistem menampilkan hasil yang diinginkan <i>User</i> pada peta.
Skenario Alternatif	<ol style="list-style-type: none"> 1. Jika kolom kosong atau mempunyai karakter non-numerikal, maka sistem akan mengeluarkan peringatan.
Kondisi akhir	Sistem menampilkan data lokasi kapal yang sesuai dengan informasi yang dimasukkan oleh <i>User</i> .

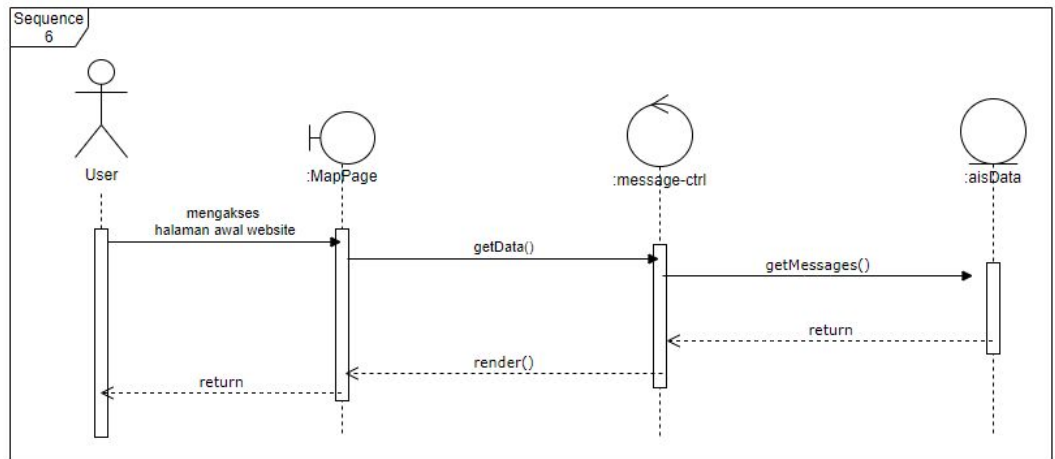
Tabel 5.10 Use Case Scenario Menampilkan tabel MMSI kapal.

<i>Use case menampilkan tabel MMSI kapal.</i>	
Tujuan	Menampilkan tabel berisi MMSI dari data kapal yang ada di dalam database.
Aktor	<i>User.</i>
Kondisi awal	Halaman index.
Skenario Utama	<ol style="list-style-type: none"> 1. <i>User</i> menekan tombol “Show MMSI Table”. 2. Sistem menampilkan menu tabel MMSI kapal. 3. <i>User</i> menekan tombol MMSI kapal yang diinginkan. 4. Sistem menampilkan data kapal tersebut.
Skenario Alternatif	-

Kondisi akhir	Sistem menampilkan data lokasi kapal sesuai dengan MMSI yang diinginkan <i>User</i> .
---------------	---

5.5.3 Sequence Diagram

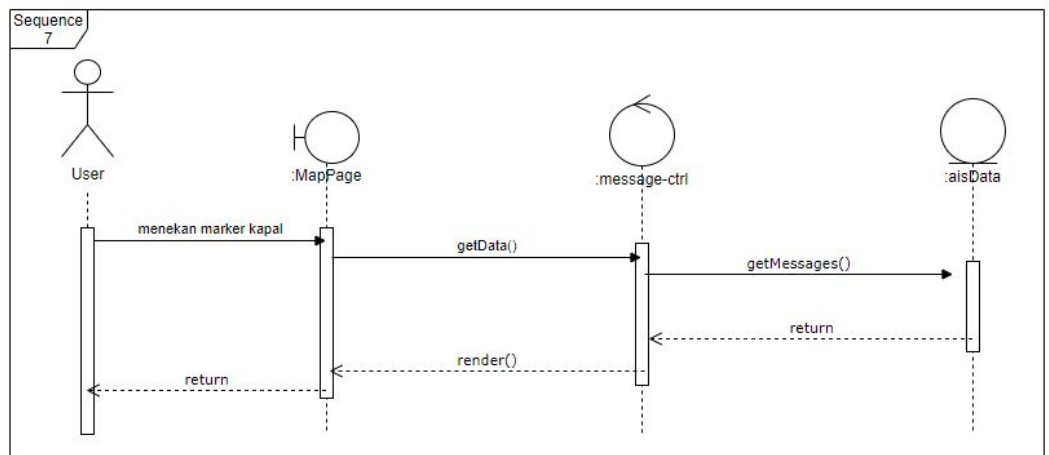
1. Sequence Melihat Lokasi Kapal



Gambar 5.2 Sequence Melihat Lokasi Kapal

Gambar 5.2 menjelaskan sequence interaksi *User* pada sistem disaat *User* mengakses halaman website. Pada saat *User* memasuki halaman website, sistem akan menampilkan data lokasi kapal sejumlah 500 buah pada peta.

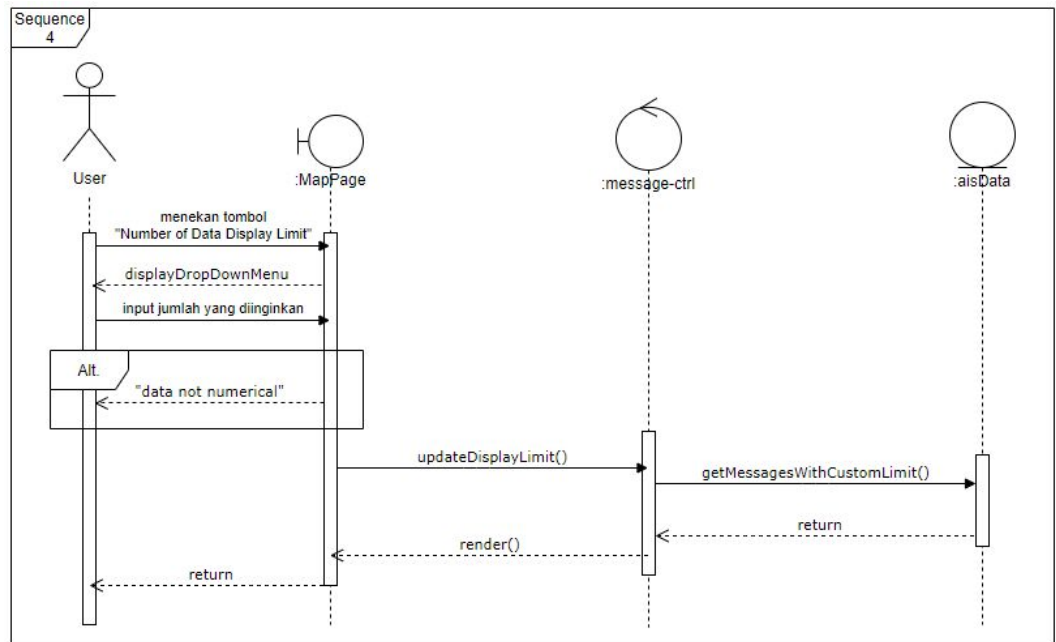
2. Sequence Melihat Detail Informasi Kapal



Gambar 5.3 Sequence Melihat Detail Informasi Kapal

Gambar 5.3 menjelaskan sequence interaksi *User* pada sistem dimana *User* ingin melihat detail informasi dari sebuah data kapal. Ketika *User* menekan salah satu marker kapal pada peta, sistem akan menampilkan sebuah *pop-up* yang berisi data dari marker kapal tersebut.

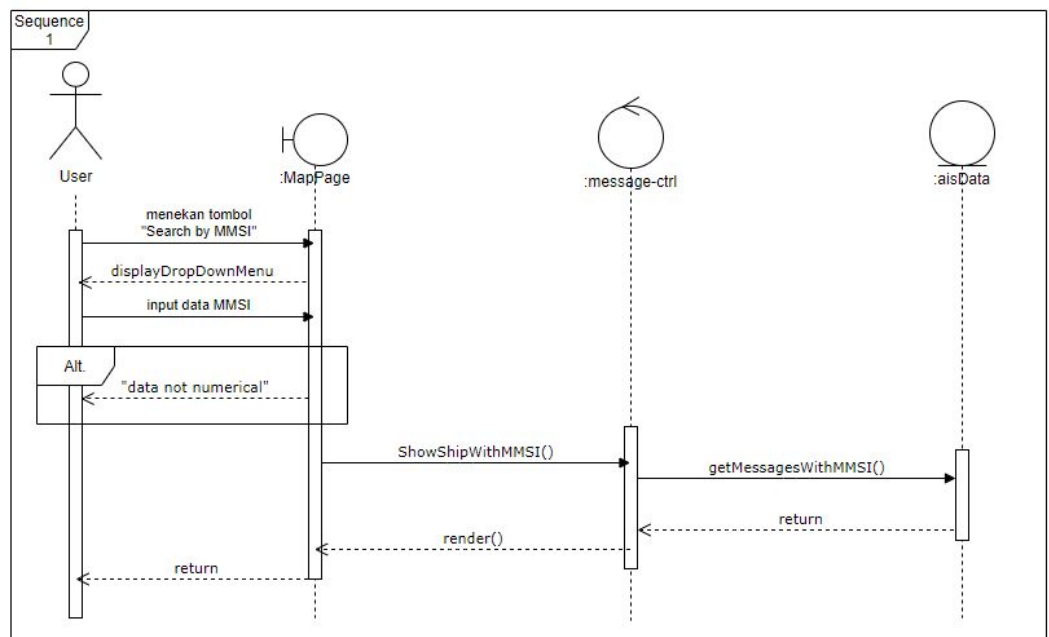
3. Sequence Membatasi Data Kapal yang Ditampilkan pada Peta



Gambar 5.4 Sequence Membatasi Data Kapal yang Ditampilkan pada Peta

Gambar 5.4 merupakan sequence diagram yang menggambarkan interaksi pengguna dan sistem ketika pengguna ingin mengatur jumlah data kapal yang ditampilkan pada website.

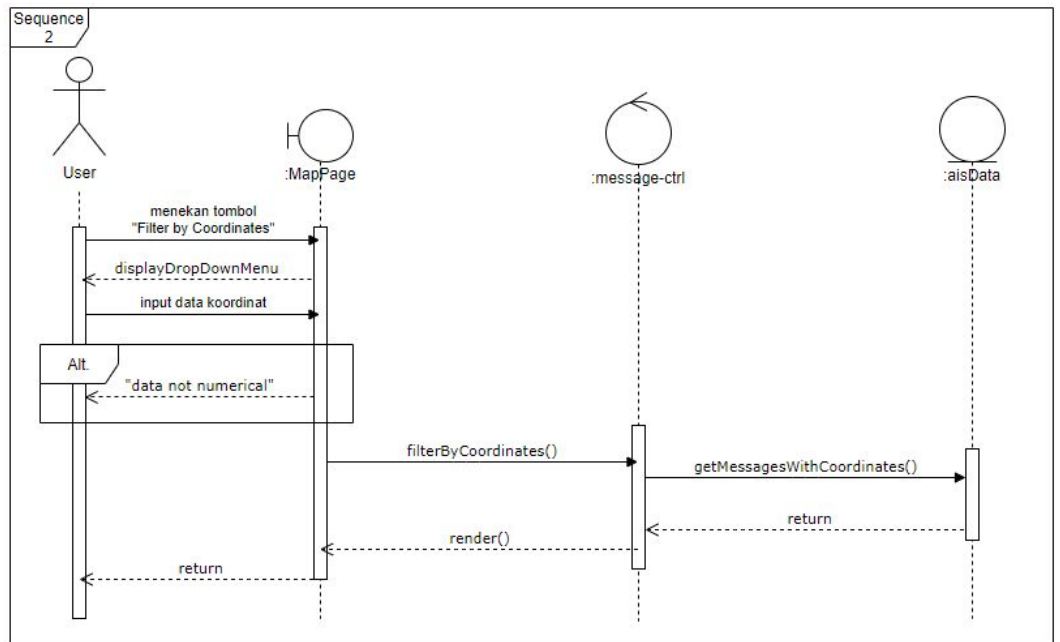
4. Sequence Menampilkan Data Berdasarkan MMSI



Gambar 5.5 Sequence Menampilkan Data Berdasarkan MMSI

Gambar 5.5 diatas merupakan sequence diagram yang menggambarkan interaksi pengguna dengan sistem ketika ingin mencari data kapal berdasarkan MMSI. Ketika pengguna menekan tombol "Search by MMSI", sistem akan mengeluarkan menu *dropdown* dimana terdapat kolom untuk pengisian MMSI yang ingin dicari oleh pengguna. Ketika pengguna telah mengisi kolom tersebut sistem akan menampilkan data kapal yang sesuai dengan MMSI yang diinginkan.

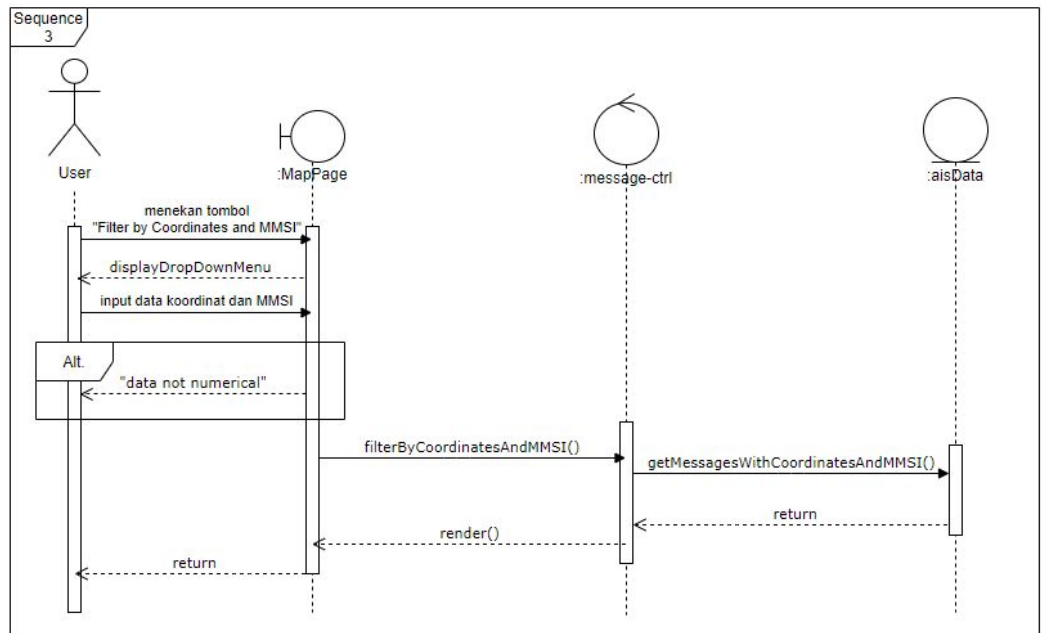
5. Sequence Menampilkan Data Berdasarkan Koordinat



Gambar 5.6 Sequence Menampilkan Data Berdasarkan Koordinat

Gambar 5.6 adalah gambar sequence interaksi pengguna dengan sistem dimana pengguna ingin menggunakan fitur “Filter by Coordinate”. Pada interaksi tersebut pengguna akan mendapatkan hasil akhir tampilan data lokasi kapal yang ditampilkan sesuai dengan daerah koordinat yang dimasukkan oleh pengguna.

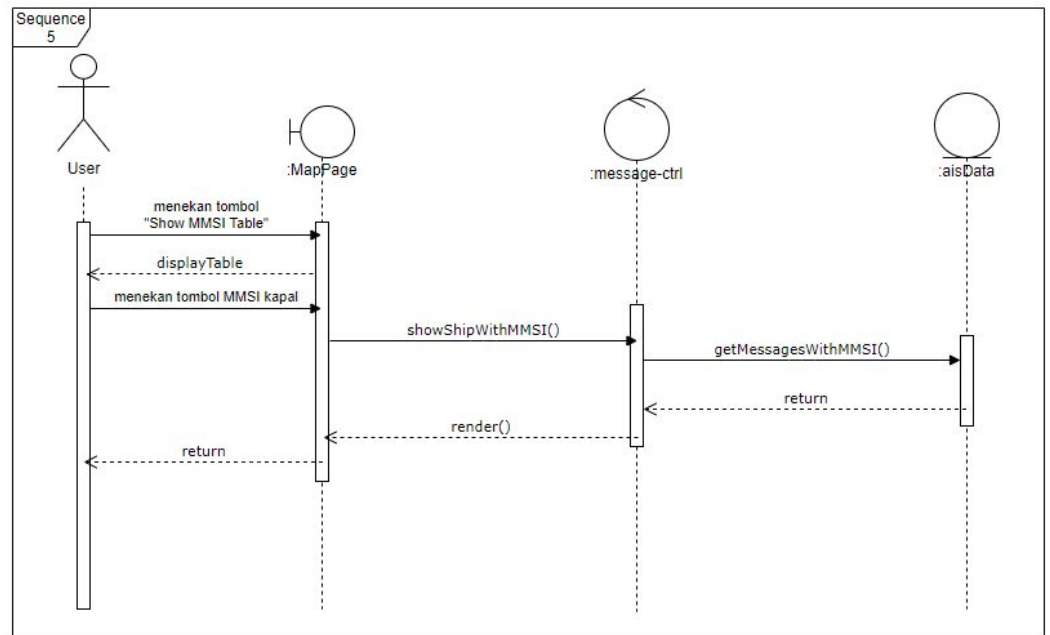
6. Sequence Menampilkan Data Berdasarkan Koordinat dan MMSI



Gambar 5.7 Sequence Menampilkan Data Berdasarkan Koordinat dan MMSI

Gambar 5.7 menunjukkan sequence interaksi pengguna dengan sistem dimana pengguna menggunakan fitur “Filter by Coordinate and MMSI”. Pada fitur tersebut pengguna memasukkan informasi berupa koordinat dan MMSI kapal yang diinginkan, yang kemudian sistem akan memvalidasi dan menampilkan data kapal yang berada pada koordinat yang diberikan pengguna dengan MMSI yang sesuai.

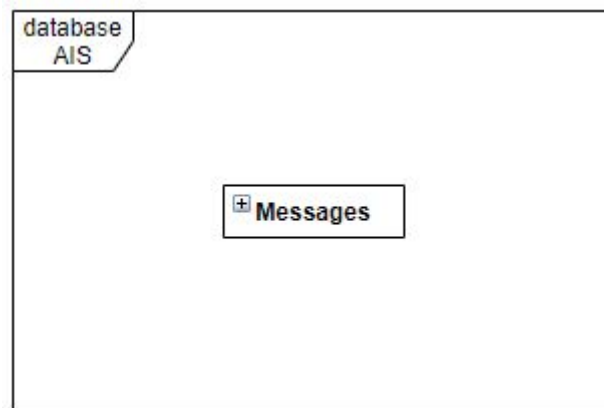
7. Sequence Menampilkan Tabel MMSI Kapal



Gambar 5.8 Sequence Menampilkan Tabel MMSI Kapal

Gambar 5.8 menunjukkan sequence interaksi pengguna dengan sistem dimana pengguna menggunakan fitur "Show MMSI Table". Dimana pada fitur tersebut pengguna akan menggunakan tabel MMSI kapal yang ditampilkan oleh sistem untuk mencari data kapal yang diinginkan pengguna.

5.5.4 Class Diagram



Gambar 5.9 Class Diagram

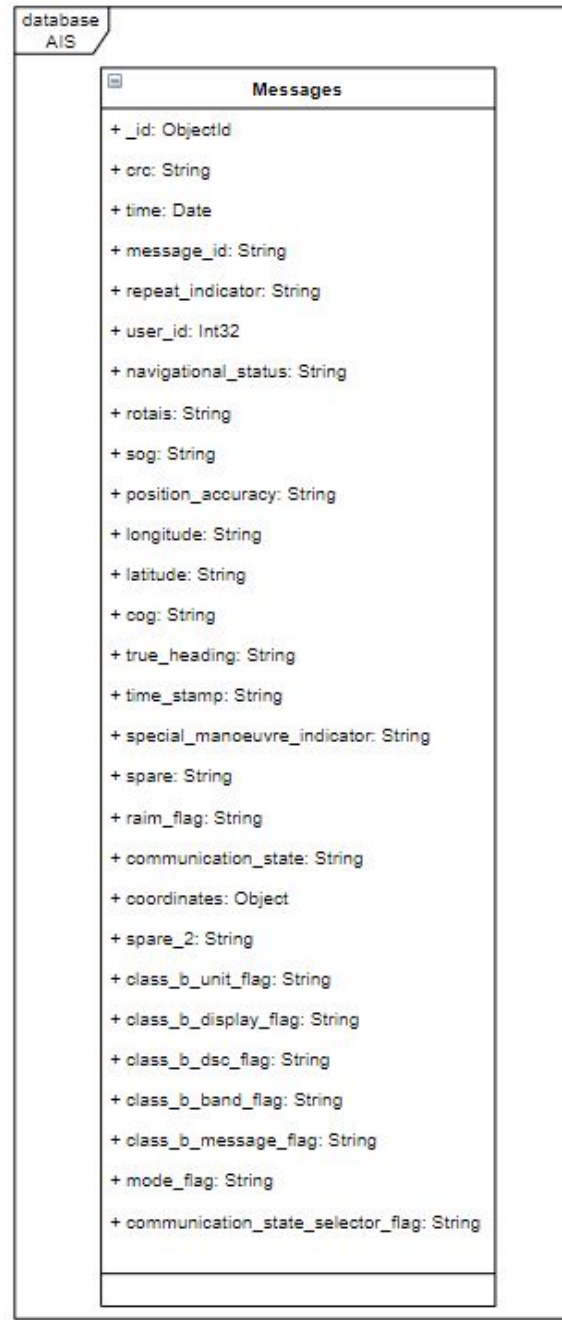
Gambar 5.22 merupakan sebuah gambar class diagram yang mendeskripsikan class yang berada pada aplikasi website. Dalam website ini hanya menggunakan 1 kelas yaitu kelas Message.

BAB 6 PERANCANGAN DAN IMPLEMENTASI

6.1 Perancangan

Pada tahap perancangan ini akan dibagi menjadi perancangan data, perancangan antarmuka, serta lingkungan pengembangan.

6.1.1 Perancangan Data

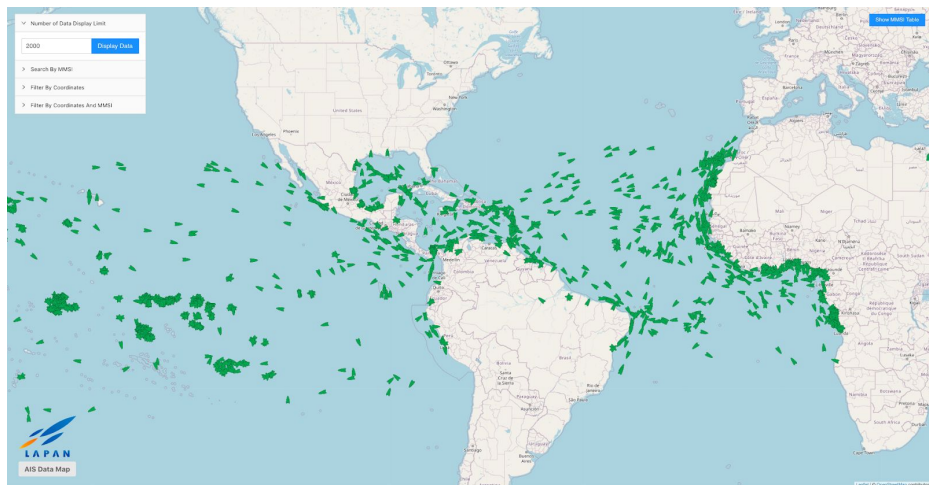


Gambar 6.1 Perancangan Data

Gambar 6.1 merupakan sebuah gambar class diagram untuk menggambarkan data beserta komponen yang ada di dalam kelas tersebut. Pada aplikasi website ini mempunyai kelas Messages dimana berisi komponen data dari lokasi kapal yang didapatkan dengan AIS.

6.1.2 Perancangan Antarmuka dan algoritma

1. Halaman Utama menampilkan lokasi kapal



Gambar 6.2 Halaman Utama menampilkan lokasi kapal

Gambar 6.2 merupakan desain dari halaman utama yang akan dibuat. Dalam halaman ini pengguna dapat melihat posisi kapal yang tersebar di seluruh dunia yang setiap data kapal hanya ditampilkan 1 data pertama saja.

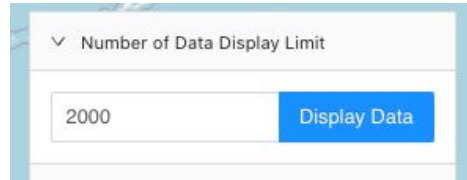
2. Komponen melihat detail informasi kapal



Gambar 6.3 Komponen melihat detail informasi kapal

Gambar 6.3 merupakan desain dari komponen yang berfungsi untuk melihat detail kapal.

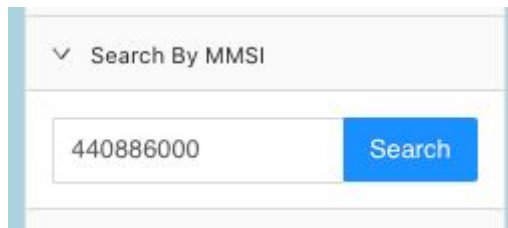
3. Komponen Menentukan Limit Data

A screenshot of a web form titled "Number of Data Display Limit". It features a text input field containing the number "2000" and a blue button labeled "Display Data".

Gambar 6.4 Komponen Menentukan Limit Data

Gambar 6.4 merupakan desain dari komponen yang berfungsi untuk menginput limit data. Pengguna dapat memasukan jumlah data yang ingin ditampilkan melalui komponen ini. Pengguna diharuskan menginput terlebih dahulu sebelum menekan display data, jika belum maka akan ada peringatan untuk mengisi data terlebih dahulu.

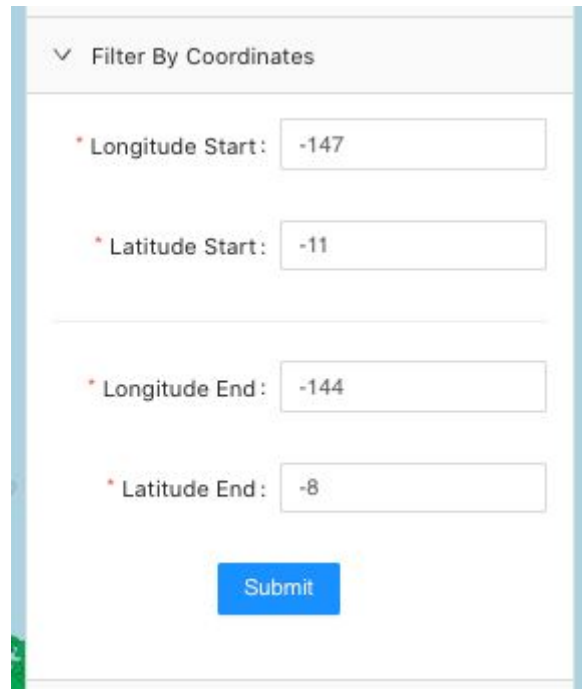
4. Komponen pencarian dengan MMSI

A screenshot of a web form titled "Search By MMSI". It features a text input field containing the MMSI number "440886000" and a blue button labeled "Search".

Gambar 6.5 Komponen pencarian dengan MMSI

Gambar 6.5 merupakan desain dari komponen yang berfungsi untuk melakukan pencarian berdasarkan dengan MMSI. Pengguna dapat memasukan MMSI kapal yang ingin ditampilkan melalui komponen ini. Pengguna diharuskan menginput terlebih dahulu sebelum menekan search, jika belum maka akan ada peringatan untuk mengisi data terlebih dahulu.

5. Komponen pencarian dengan membatasi Koordinat



The image shows a web form titled "Filter By Coordinates" with a dropdown arrow. It contains four input fields, each preceded by a red asterisk indicating a required field. The first two fields are "Longitude Start" with the value "-147" and "Latitude Start" with the value "-11". The next two fields are "Longitude End" with the value "-144" and "Latitude End" with the value "-8". Below these fields is a blue "Submit" button.

Gambar 6.6 Komponen pencarian dengan membatasi Koordinat.

Gambar 6.6 merupakan desain dari komponen yang berfungsi untuk melakukan pencarian dengan membatasi suatu area koordinat. Pengguna dapat memasukan koordinat awal dan koordinat akhir yang ingin ditampilkan melalui komponen ini. Pengguna diharuskan menginput terlebih dahulu sebelum menekan submit, jika belum maka akan ada peringatan untuk mengisi data terlebih dahulu.

6. Komponen pencarian dengan membatasi koordinat dan MMSI tertentu

Filter By Coordinates And MMSI

* MMSI: 98336492

* Longitude Start: -147

* Latitude Start: -11

* Longitude End: -144

* Latitude End: -8

Submit

Gambar 6.7 Komponen pencarian dengan membatasi koordinat dan MMSI tertentu

Gambar 6.7 merupakan desain dari komponen yang berfungsi untuk melakukan pencarian dengan membatasi suatu area koordinat dan MMSI tertentu. Pengguna dapat memasukkan koordinat awal dan koordinat akhir dan MMSI yang ingin ditampilkan melalui komponen ini. Pengguna diharuskan menginput terlebih dahulu sebelum menekan submit, jika belum maka akan ada peringatan untuk mengisi data terlebih dahulu.

7. Komponen melihat daftar tabel MMSI

MMSI Table

No.	MMSI	Action
31	2241152	See Ships with 2241152 MMSI
32	2242155	See Ships with 2242155 MMSI
33	2242156	See Ships with 2242156 MMSI
34	2242175	See Ships with 2242175 MMSI
35	2579991	See Ships with 2579991 MMSI
36	2579994	See Ships with 2579994 MMSI
37	2579995	See Ships with 2579995 MMSI
38	2579997	See Ships with 2579997 MMSI
39	2579999	See Ships with 2579999 MMSI
40	2972005	See Ships with 2972005 MMSI

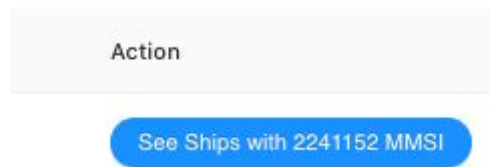
[<](#)[1](#)[2](#)[3](#)[4](#)[5](#)[6](#)[...](#)[734](#)[>](#)

10 / page ▾

Gambar 6.8 Komponen melihat daftar tabel MMSI

Gambar 6.8 merupakan desain dari komponen untuk melihat daftar tabel MMSI kapal.

8. Komponen menampilkan posisi Kapal



Gambar 6.9 Komponen menampilkan posisi kapal

Gambar 6.9 merupakan design dari komponen untuk menampilkan posisi kapal berdasarkan MMSI tertentu.

6.1.3 Lingkungan Pengembangan Sistem

6.1.3.1 Software

DBMS	MongoDB
Framework	ExpressJS dan ReactJS
Bahasa Pemrograman	Javascript
Browser	Chrome dan Safari
IDE	Visual Studio Code

6.1.3.2 Sistem operasi

Sistem Operasi	Windows 10 dan MacOS Catalina
----------------	-------------------------------

6.1.3.3 Hardware

Storage	512 GB
RAM	16 GB
Processor	Intel i7-8750G

6.1.3.4 Arsitektur Sistem

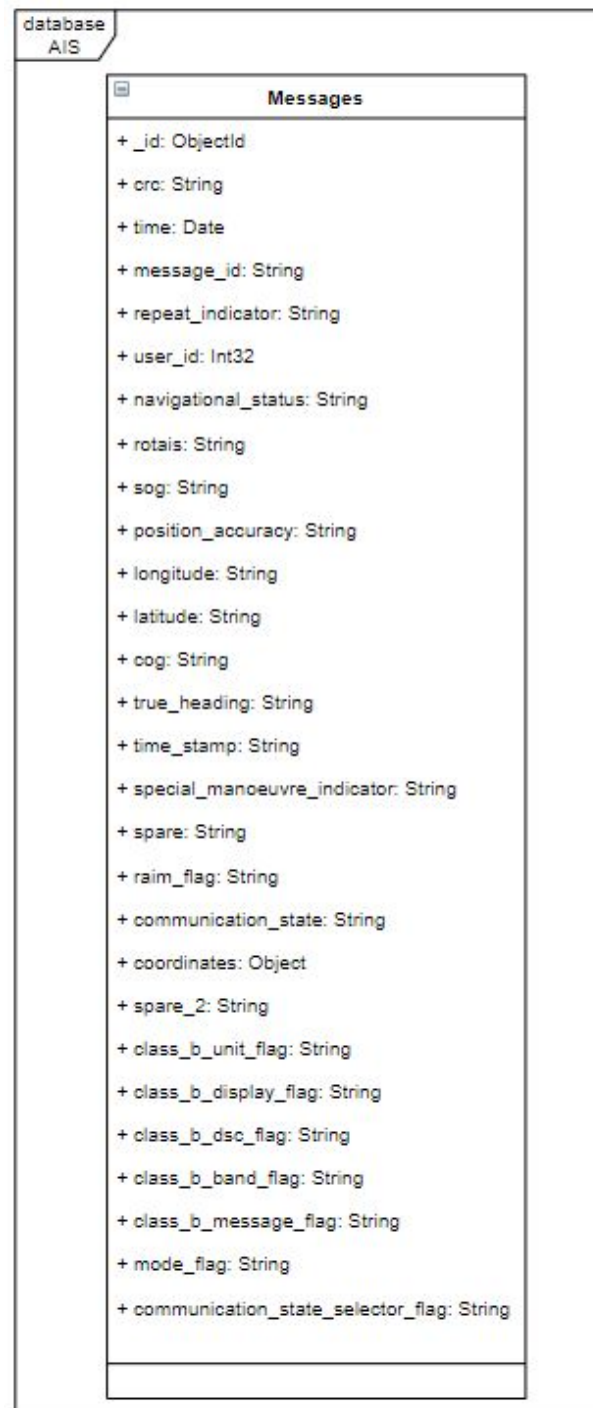
Arsitektur Sistem yang digunakan adalah three tier (3 tier) dengan konsep client server programming yang memiliki 3 fungsionalitas sistem yang independent yaitu komponen client aplikasi server dan database.

Client	Web Browser
Server	NodeJS + ExpressJS
Database	MongoDB

6.2 Implementasi

Pada tahap implementasi ini akan dibagi menjadi implementasi data dan implementasi antarmuka.

6.2.1 Implementasi Data



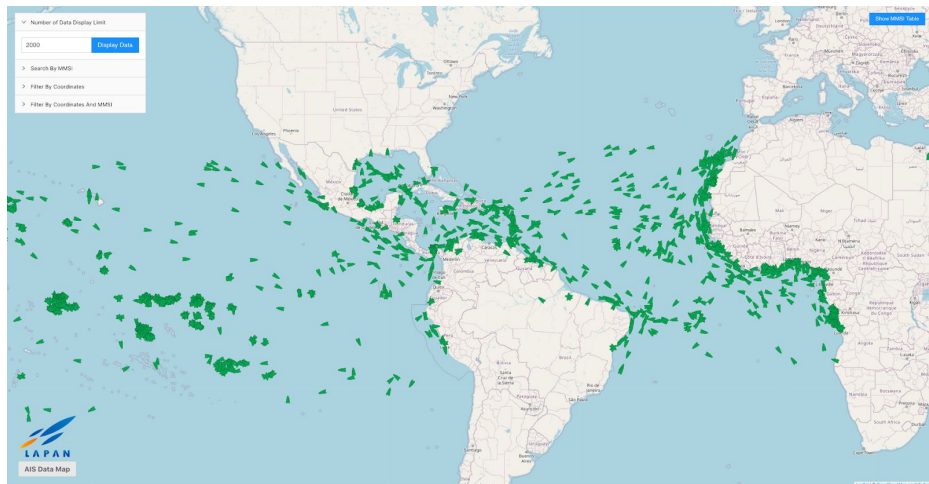
Gambar 6.10 Implementasi Data

Pada gambar 6.10, tabel Message berisikan variabel `_id` bertipe ObjectId yang tergenerate otomatis oleh MongoDB, `crc` bertipe String, `time` bertipe String, `message_id` bertipe String, `repeat_indicator` bertipe String, `user_id` bertipe Int, `navigational_status` bertipe String, `rotais` bertipe String, `sog` bertipe String, `rotation_accuracy` bertipe String, `longitude` bertipe String, `latitude` bertipe String, `cog` bertipe String, `true_heading` bertipe String, `time_stamp` bertipe String, `special_manuevre_indicator` bertipe String, `spare` bertipe String, `raim_flag` bertipe String, `communication_state` bertipe String, `coordinates` bertipe Object String yang dimana didalamnya berisi array string yang berisikan coordinate bertipe String, `spare_2` bertipe String, `class_b_unit_flag` bertipe String, `class_b_display_flag` bertipe String, `class_b_dsc_flag` bertipe String, `class_b_band_flag` bertipe String, `class_b_message_flag` bertipe String, `mode_flag` bertipe String, dan `communication_state_selector_flag` bertipe String.

6.2.2 Implementasi Antarmuka dan Algoritma

Setelah dilakukan analisa dan perancangan pada tahap sebelumnya, pada bagian ini merupakan hasil dari implementasi berdasarkan analisa dan perancangan yang telah dilakukan.

1. Halaman Utama menampilkan lokasi kapal



Gambar 6.11 Halaman Utama menampilkan lokasi kapal

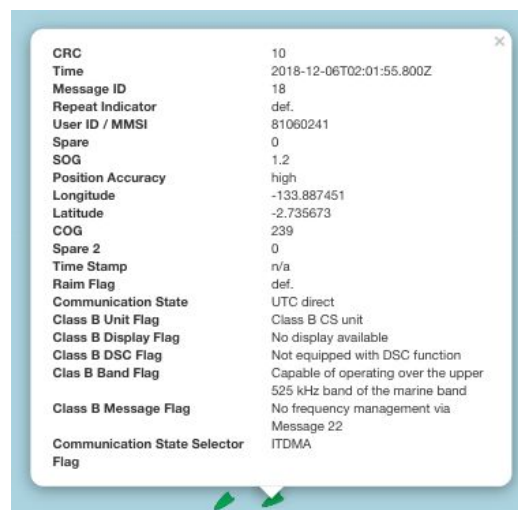
Gambar 6.11 merupakan desain dari halaman utama yang akan dibuat. Dalam halaman ini pengguna dapat melihat posisi kapal yang tersebar di seluruh dunia yang setiap data kapal hanya ditampilkan 1 data pertama saja. Berikut merupakan potongan kode program ditunjukkan pada Tabel 6.1.

Tabel 6.1 Kode Halaman Utama menampilkan lokasi kapal

1	<code>async function getData(data, limit) {</code>
2	<code> let result = [];</code>
3	<code> for await (const item of data) {</code>
4	<code> let data = await Message.findOne({</code>

5	user_id: item
6	}).exec();
7	result.push(data);
8	if (result.length > limit) {
9	break;
10	}
11	}
12	return result;
13	}
14	
15	getMessages = async(req, res) => {
16	try {
17	let data = await Message.find().distinct('user_id');
18	let limit = 500
19	let result = await getData(data, limit);
20	return res.json({
21	success: true,
22	data: result
23	})
24	} catch (e) {
25	return res.status(400).json({
26	success: false,
27	error: e
28	})
29	}
30	}

2. Komponen melihat detail informasi kapal



Gambar 6.12 Komponen melihat detail informasi kapal

Gambar 6.12 merupakan desain dari komponen yang berfungsi untuk melihat detail kapal. Berikut merupakan potongan kode program ditunjukkan pada Tabel 6.2.

Tabel 6.2 Kode melihat detail AIS message

1	async function getData(data, limit) {
2	let result = [];

3	for await (const item of data) {
4	let data = await Message.findOne({
5	user_id: item
6	}).exec();
7	result.push(data);
8	if (result.length > limit) {
9	break;
10	}
11	}
12	return result;
13	}
14	
15	getMessages = async(req, res) => {
16	try {
17	let data = await Message.find().distinct('user_id');
18	let limit = 500
19	let result = await getData(data, limit);
20	return res.json({
21	success: true,
22	data: result
23	})
24	} catch (e) {
25	return res.status(400).json({
26	success: false,
27	error: e
28	})
29	}
30	}

3. Komponen Menentukan Limit Data

Gambar 6.13 Komponen Menentukan Limit Data

Gambar 6.13 merupakan desain dari komponen yang berfungsi untuk menginput limit data. Pengguna dapat memasukkan jumlah data yang ingin ditampilkan melalui komponen ini. Pengguna diharuskan menginput terlebih dahulu sebelum menekan display data, jika belum maka akan ada peringatan untuk mengisi data terlebih dahulu. Berikut merupakan potongan kode program ditunjukkan pada Tabel 6.3.

Tabel 6.3 Kode Menentukan Limit Data

1	getMessagesWithCustomLimit = async(req, res) => {
2	try {
3	var limitReq = parseInt(req.params.limit_number)
4	let data = await Message.find().distinct('user_id');
5	let limit = limitReq
6	let result = await getData(data, limit);
7	return res.json({
8	success: true,

9	data: result
10	})
11	} catch (e) {
12	return res.status(400).json({
13	success: false,
14	error: e
15	})
16	}
17	}

4. Komponen pencarian dengan MMSI

Gambar 6.14 Komponen pencarian dengan MMSI

Gambar 6.14 merupakan desain dari komponen yang berfungsi untuk melakukan pencarian berdasarkan dengan MMSI. Pengguna dapat memasukan MMSI kapal yang ingin ditampilkan melalui komponen ini. Pengguna diharuskan menginput terlebih dahulu sebelum menekan search, jika belum maka akan ada peringatan untuk mengisi data terlebih dahulu. Berikut merupakan potongan kode program ditunjukkan pada Tabel 6.4.

Tabel 6.4 Kode pencarian dengan MMSI

1	getMessagesWithMMSI = async(req, res) => {
2	try {
3	var mmsi = req.params.user_id
4	var query = {
5	user_id: mmsi
6	}
7	var m = Message.find(query).limit(50)
8	m.exec(function(err, message) {
9	return res.status(200).json({
10	success: true,
11	data: message
12	});
13	})
14	} catch (e) {
15	return res.status(400).json({
16	success: false,
17	error: e
18	})
19	}
20	}

5. Komponen pencarian dengan membatasi Koordinat

Filter By Coordinates

* Longitude Start: -147

* Latitude Start: -11

* Longitude End: -144

* Latitude End: -8

Submit

Gambar 6.15 Komponen pencarian dengan membatasi Koordinat.

Gambar 6.15 merupakan desain dari komponen yang berfungsi untuk melakukan pencarian dengan membatasi suatu area koordinat. Pengguna dapat memasukkan koordinat awal dan koordinat akhir yang ingin ditampilkan melalui komponen ini. Pengguna diharuskan menginput terlebih dahulu sebelum menekan submit, jika belum maka akan ada peringatan untuk mengisi data terlebih dahulu. Berikut merupakan potongan kode program ditunjukkan pada Tabel 6.5.

Tabel 6.5 Kode pencarian dengan membatasi Koordinat.

1	getMessagesWithCoordinates = async(req, res) => {
2	try {
3	var longStart = parseFloat(req.params.long_start)
4	var latStart = parseFloat(req.params.lat_start)
5	var longEnd = parseFloat(req.params.long_end)
6	var latEnd = parseFloat(req.params.lat_end)
7	var query = {
8	\$and: [
9	{ \$and: [
10	{ 'coordinates.coordinates.0': { \$gte: longStart } },
11	{ 'coordinates.coordinates.0': { \$lte: longEnd } }] },
12	{ \$and: [
13	{ 'coordinates.coordinates.1': { \$gte: latStart } },
14	{ 'coordinates.coordinates.1': { \$lte: latEnd } }] }
15]
16	}
17	var m = Message.find(query).limit(100)
18	m.exec(function(err, message) {
19	return res.status(200).json({
20	success: true,

21	data: message
22	});
23	}}
24	} catch (e) {
25	return res.status(400).json({
26	success: false,
27	error: e
28	})
29	}
30	}

6. Komponen pencarian dengan membatasi koordinat dan MMSI tertentu

Filter By Coordinates And MMSI

* MMSI: 98336492

* Longitude Start: -147

* Latitude Start: -11

* Longitude End: -144

* Latitude End: -8

Submit

Gambar 6.16 Komponen pencarian dengan membatasi koordinat dan MMSI tertentu

Gambar 6.16 merupakan desain dari komponen yang berfungsi untuk melakukan pencarian dengan membatasi suatu area koordinat dan MMSI tertentu. Pengguna dapat memasukan koordinat awal dan koordinat akhir dan MMSI yang ingin ditampilkan melalui komponen ini. Pengguna diharuskan menginput terlebih dahulu sebelum menekan submit, jika belum maka akan ada peringatan untuk mengisi data terlebih dahulu. Berikut merupakan potongan kode program ditunjukkan pada Tabel 6.6.

Tabel 6.6 Kode pencarian dengan membatasi Koordinat dan MMSI tertentu.

1	getMessagesWithCoordinatesAndMMSI = async(req, res) => {
2	try {
3	var mmsi = parseInt(req.params.mmsi)
4	var longStart = parseFloat(req.params.long_start)
5	var latStart = parseFloat(req.params.lat_start)
6	var longEnd = parseFloat(req.params.long_end)
7	var latEnd = parseFloat(req.params.lat_end)
8	var query = {
9	\$and: [
10	{ \$and: [
11	{ 'coordinates.coordinates.0': { \$gte: longStart } },
12	{ 'coordinates.coordinates.0': { \$lte: longEnd } }] },
13	{ \$and: [
14	{ 'coordinates.coordinates.1': { \$gte: latStart } },
15	{ 'coordinates.coordinates.1': { \$lte: latEnd } }] }
16]
17	}
18	var m = Message.find(query).limit(100)
19	m.exec(function(err, message) {
20	return res.status(200).json({
21	success: true,
22	data: message
23	});
24	})
25	} catch (e) {
26	return res.status(400).json({
27	success: false,
28	error: e
29	})
30	}
31	}

7. Komponen melihat daftar tabel MMSI

MMSI Table

No.	MMSI	Action
31	2241152	See Ships with 2241152 MMSI
32	2242155	See Ships with 2242155 MMSI
33	2242156	See Ships with 2242156 MMSI
34	2242175	See Ships with 2242175 MMSI
35	2579991	See Ships with 2579991 MMSI
36	2579994	See Ships with 2579994 MMSI
37	2579995	See Ships with 2579995 MMSI
38	2579997	See Ships with 2579997 MMSI
39	2579999	See Ships with 2579999 MMSI
40	2972005	See Ships with 2972005 MMSI

< 1 2 3 4 5 6 ... 734 >

10 / page ▾

Gambar 6.17 Komponen melihat daftar tabel MMSI

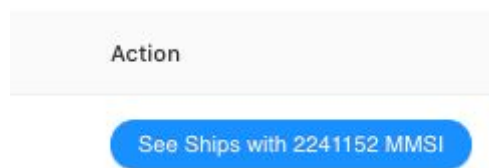
Gambar 6.17 merupakan desain dari komponen untuk melihat daftar tabel MMSI kapal. Berikut merupakan potongan kode program ditunjukkan pada Tabel 6.7.

Tabel 6.7 Kode melihat daftar tabel MMSI.

1	<code>getMMSI = async(req, res) => {</code>
2	<code> try {</code>

3	let data = await Message.find().distinct('user_id');
4	return res.json({
5	success: true,
6	data: data
7	});
8	} catch (e) {
9	return res.status(400).json({
10	success: false,
11	error: e
12	});
13	}
14	}

8. Komponen menampilkan posisi Kapal dengan MMSI dari tabel



Gambar 6.18 Komponen menampilkan posisi kapal

Gambar 6.18 merupakan desain dari komponen untuk menampilkan posisi kapal berdasarkan MMSI tertentu. Berikut merupakan potongan kode program ditunjukkan pada Tabel 6.7.

Tabel 6.8 Kode menampilkan posisi kapal.

1	getMessagesWithMMSI = async(req, res) => {
2	try {
3	var mmsi = req.params.user_id
4	var query = {
5	user_id: mmsi
6	}
7	var m = Message.find(query).limit(50)
8	m.exec(function(err, message) {
9	return res.status(200).json({
10	success: true,
11	data: message
12	});
13	});
14	} catch (e) {
15	return res.status(400).json({
16	success: false,
17	error: e
18	});
19	}
20	}

BAB 7 PENGUJIAN

7.1 Pengujian Validasi

Pada tahap ini kami akan melakukan pengujian dengan menggunakan metode *Blackbox Testing* yang merupakan metode untuk melakukan pengujian validasi. Berikut hasil pengujian yang kami lakukan ditunjukkan pada Tabel 7.1

Tabel 7.1 Pengujian Blackbox

No	Test Name	Test Case	Expected Result	Result	Status
1	Pengujian melihat lokasi kapal	-	Sistem akan menampilkan lokasi kapal dalam bentuk peta	Sistem akan menampilkan lokasi kapal dalam bentuk peta	Valid
2	Pengujian melihat detail informasi kapal	-	Sistem akan menampilkan detail informasi kapal	Sistem akan menampilkan detail informasi kapal	Valid
3	Pengujian menentukan limit data	Memasukkan jumlah data yang ingin ditampilkan	Sistem akan menampilkan data sejumlah data yang dimasukkan	Sistem akan menampilkan data sejumlah data yang dimasukkan	Valid
		Tidak memasukkan jumlah data yang diinginkan	Sistem akan menampilkan peringatan untuk memasukkan jumlah data	Sistem akan menampilkan peringatan untuk memasukkan jumlah data	Valid
4	Pengujian pencarian dengan MMSI	Memasukkan MMSI yang ingin dicari	Sistem akan menampilkan kapal sesuai dengan MMSI yang dimasukkan	Sistem akan menampilkan kapal sesuai dengan MMSI yang dimasukkan	Valid
		Tidak memasukkan MMSI yang ingin dicari	Sistem akan menampilkan peringatan untuk	Sistem akan menampilkan peringatan untuk	Valid

			memasukan MMSI	memasukan MMSI	
5	Pengujian pencarian dengan membatasi Koordinat	Memasukkan koordinat yang ingin dicari	Sistem akan menampilkan kapal sesuai dengan koordinat yang dimasukkan	Sistem akan menampilkan kapal sesuai dengan koordinat yang dimasukkan	Valid
		Tidak memasukkan koordinat yang ingin dicari	Sistem akan menampilkan peringatan untuk memasukan koordinat	Sistem akan menampilkan peringatan untuk memasukan koordinat	Valid
6	Pengujian pencarian dengan membatasi koordinat dan MMSI	Memasukkan koordinat dan MMSI yang ingin dicari	Sistem akan menampilkan kapal sesuai dengan koordinat dan MMSI yang dimasukkan	Sistem akan menampilkan kapal sesuai dengan koordinat dan MMSI yang dimasukkan	Valid
		Tidak memasukkan koordinat dan MMSI yang ingin dicari	Sistem akan menampilkan peringatan untuk memasukan koordinat dan MMSI	Sistem akan menampilkan peringatan untuk memasukan koordinat dan MMSI	Valid
7	Pengujian melihat daftar tabel MMSI	Menekan tombol Show MMSI Table	Sistem akan menampilkan tabel MMSI	Sistem akan menampilkan tabel MMSI	Valid
8	Pengujian melihat kapal melalui tabel MMSI	Menekan tombol See Ships with MMSI	Sistem akan menampilkan kapal sesuai dengan MMSI yang dipilih	Sistem akan menampilkan kapal sesuai dengan MMSI yang dipilih	Valid

7.2 Analisis Hasil Pengujian

Analisa dari hasil pengujian dengan menggunakan metode Blackbox menunjukan bahwa sistem yang dibangun memiliki validitas 8 dari 8 yaitu 100%. Dengan ini menandakan sistem yang dibangun sudah sesuai dengan kebutuhan sebelumnya. Dengan demikian Sistem Pemantauan Lokasi Kapal berbasis website ini dapat dinyatakan sesuai dengan daftar kebutuhan yang telah dijelaskan di bagian-bagian sebelumnya yaitu dalam dokumentasi dari tahap analisis, perancangan, hingga implementasi.

BAB 8 PENUTUP

8.1 Kesimpulan

Berdasarkan hasil analisis, perancangan, implementasi, dan pengujian yang dilakukan, maka dapat diperoleh kesimpulan sebagai berikut:

1. Tahapan perancangan sistem pemantauan lokasi kapal diawali dengan tahap analisis yaitu dengan diskusi tim bersama pembimbing PKL dari pihak LAPAN untuk membuat backlog product, perencanaan sprint. Setelah hal itu selesai lalu dilanjutkan dengan pembuatan *use case diagram*, *use case scenario*, *sequence diagram*, hingga *class diagram*. Setelah itu dilanjutkan tahap perancangan dan implementasi, yang mana didalamnya dilakukan perancangan data dan perancangan antarmuka serta algoritmanya serta implementasi data dan implementasi antarmuka serta algoritmanya. Lalu dilanjutkan dengan pengujian *Blackbox Testing* sebagai metode validasi bahwa sistem yang dibuat telah sesuai dengan kebutuhan yang ditetapkan sebelumnya. Pertukaran data dalam sistem ini dirancang dengan menggunakan protokol REST API.
2. Hasil dari perancangan sistem pemantauan lokasi kapal berhasil diimplementasikan dengan *tech stack* MERN Stack yaitu MongoDB, ExpressJS, ReactJS, dan NodeJS.
3. Hasil rancangan dan implementasi dari aplikasi sistem pemantauan lokasi kapal telah berhasil memenuhi kriteria kepuasan yang diberikan dari pihak penguji dari pihak LAPAN. Sesuai pengujian validasi didapatkan bahwa sistem berhasil memenuhi kebutuhan yang telah ditetapkan oleh pihak LAPAN dengan validitas setingkat 100%.

8.2 Saran

Sistem yang dibuat masih banyak kekurangan, oleh karena itu ada beberapa saran yang bisa diterapkan pada sistem kedepannya, antara lain:

1. Ditambahkan fitur waktu dan tanggal sehingga bisa melakukan pencarian berdasarkan waktu atau tanggal tertentu.
2. Ditambahkan fitur untuk memasukkan data sehingga data bisa update setiap saat sesuai dengan input data terakhir.
3. Menggunakan server yang lebih cepat agar meminimalisir waktu tunggu transmisi data.

8.3 Keberlanjutan

Sebagaimana pengembangan aplikasi ini sebagai aplikasi dasar dari visualisasi data kapal, masih banyak fitur yang dapat dikembangkan pada aplikasi ini. Salah satu ide dan saran yang diberikan oleh pihak LAPAN saat presentasi tugas akhir adalah dengan membuat fitur yang dapat memperlihatkan jalur

berupa garis dari selama perjalanan kapal tersebut sehingga perjalanan kapal bisa terlihat dengan jelas.

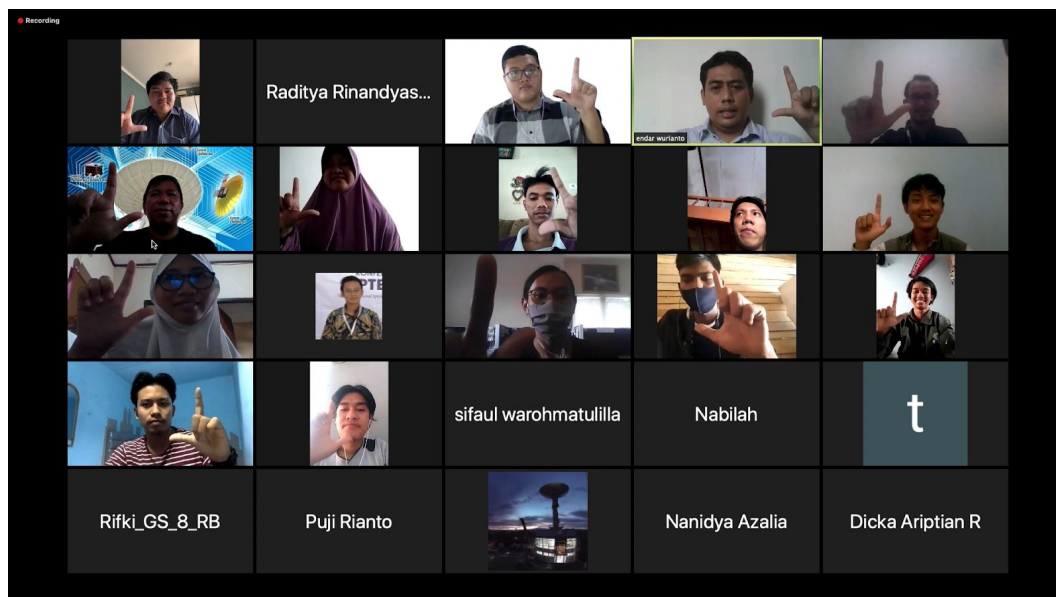
DAFTAR PUSTAKA

- Gurendo, D., 2015. *Software Development Life Cycle (SDLC). Scrum Model Step by Step*. [Online]
Available at:
<https://xbsoftware.com/blog/software-development-life-cycle-sdlc-scrum-step-step/>
[Accessed 09 03 2020].
- Hutauruk, M. K., 2019. *UML Diagram : Use Case Diagram*. [Online]
Available at:
<https://socs.binus.ac.id/2019/11/26/uml-diagram-use-case-diagram/>
[Accessed 15 07 2020].
- Tirta, R., n.d. APLIKASI PENGOLAHAN DATA PEGAWAI. *APLIKASI PENGOLAHAN DATA PEGAWAI*.
- Maulidi, A., 2019. *Disain Sistem Navigasi Automatic Identification System (AIS) Transceiver Berbasis Mini Computer Pada Kapal Nelayan Tradisional Di Madura*. *INOVTEK POLBENG*, 9(1), p.12.
- Lapan.go.id. 2020. LAPAN - Lembaga Penerbangan Dan Antariksa Nasional. [online] Available at: <<https://www.lapan.go.id/page/tugas-dan-fungsi>> [Accessed 15 September 2020].
- Lapan.go.id. 2020. LAPAN - Lembaga Penerbangan Dan Antariksa Nasional. [online] Available at: <<https://www.lapan.go.id/page/sejarah-lapan>> [Accessed 15 September 2020].
- Ariata, C., 2019. *Apa Itu Javascript? Pemahaman Dasar Mengenai Javascript Bagi Para Pemula*. [online] Hostinger Tutorial. Available at: <<https://www.hostinger.co.id/tutorial/apa-itu-javascript/>> [Accessed 15 September 2020].
- Musa, F., 2017. *React JS, Babun, Dan Cara Installasinya..* [online] Koding Indonesia. Available at: <<https://www.kodingindonesia.com/react-js-dan-installasi/>> [Accessed 15 September 2020].
- IDCloudHost. 2020. *Mengenal Apa Itu MongoDB : Manfaat, Keunggulan Dan Cara Kerja MongoDB [LENGKAP]*. [online] Available at: <<https://idcloudhost.com/mengenal-apa-itu-mongodb-manfaat-keunggulan-dan-cara-kerja-mongodb-lengkap/>> [Accessed 15 September 2020].
- Ricky, H., Leo Willyanto, S. and Alexander, S., 2019. REAL-TIME BPMN WEBSITE MENGGUNAKAN TEKNOLOGI MERN STACK. *Jurnal Infra*, [online] 7(2). Available at: <<http://publication.petra.ac.id/index.php/teknik-informatika/article/view/8758>> [Accessed 15 September 2020].

- Blog Dewaweb. 2017. *Scrum Methodology: Panduan Project Management*. [online] Available at: <<https://www.dewaweb.com/blog/scrum-methodology-panduan-project-management/>> [Accessed 15 September 2020].
- Pratama, A., 2019. *Belajar UML - Use Case Diagram - Codepolitan.Com*. [online] CodePolitan.com. Available at: <<https://www.codepolitan.com/mengenal-uml-diagram-use-case>> [Accessed 15 September 2020].
- Slideshare.net. 2020. *[RPL2] Class Diagram Dan Relasinya (2)*. [online] Available at: <<https://www.slideshare.net/rizkiadamkurniawan/rpl2-class-diagram-dan-relasinya-2>> [Accessed 15 September 2020].
- Repository.widyatama.ac.id. 2020. [online] Available at: <<https://repository.widyatama.ac.id/xmlui/bitstream/handle/123456789/4700/Bab%202.pdf?sequence=11&isAllowed=y>> [Accessed 15 September 2020].
- Se.ittelkom-pwt.ac.id. 2020. *Pentingnya Melakukan Pengujian Perangkat Lunak (Software Testing) – Software Engineering ITTP*. [online] Available at: <<http://se.ittelkom-pwt.ac.id/pentingnya-melakukan-pengujian-perangkat-lunak-software-testing/>> [Accessed 15 September 2020].
- Docplayer.info. 2020. PENGUJIAN PERANGKAT LUNAK MENGGUNAKAN METODE BLACK BOX STUDI KASUS EXELSA - PDF Free Download. [online] Available at: <<https://docplayer.info/144751690-Pengujian-perangkat-lunak-menggunakan-metode-black-box-studi-kasus-exelsa.html>> [Accessed 15 September 2020].
- Juarna, A., 2020. Perancangan Perangkat Lunak. [online] Available at: <<http://ajuarna.staff.gunadarma.ac.id/Downloads/files/4927/PerancanganPerangkatLunak.pdf>> [Accessed 8 October 2020].
- Tambuwun, T., Sengkey, R. and Rindengan, Y., 2020. Perancangan Aplikasi Web Berbasis Usability.

LAMPIRAN

Lampiran A Dokumentasi Presentasi Tugas Website yang dibuat



Keterangan:

Dokumentasi ketika kelompok PKL melakukan presentasi mengenai tugas yang dibuat kepada pihak LAPAN dan peserta magang lainnya.