

**SIMULASI KENDALI TRIM PADA PELUNCURAN JACKET
DENGAN PENGATURAN BALLAST BARGE**

*Diajukan sebagai syarat untuk memperoleh gelar Sarjana Teknik pada
Department Teknik Sistem Perkapalan Fakultas Teknik
Universitas Hasanuddin*

SKRIPSI



MUHAMMAD ASSIDDIQ

D091171013

DEPARTEMEN TEKNIK SISTEM PERKAPALAN

FAKULTAS TEKNIK

UNIVERSITAS HASANUDDIN

2022

**SIMULASI KENDALI TRIM PADA PELUNCURAN JACKET
DENGAN PENGATURAN BALLAST BARGE**

*Diajukan sebagai syarat untuk memperoleh gelar Sarjana Teknik pada
Department Teknik Sistem Perkapalan Fakultas Teknik
Universitas Hasanuddin*

SKRIPSI



MUHAMMAD ASSIDDIQ

D091171013

DEPARTEMEN TEKNIK SISTEM PERKAPALAN

FAKULTAS TEKNIK

UNIVERSITAS HASANUDDIN

2022

LEMBAR PENGESAHAN

**SIMULASI KENDALI TRIM PADA PELUNCURAN JACKET DENGAN
PENGATURAN BALLAST BARGE**

Disusun dan diajukan oleh:

MUHAMMAD ASSIDDIQ

D091171013

Telah dipertahankan di hadapan Panitia Ujian yang dibentuk dalam rangka
Penyelesaian Studi Program Sarjana Program Studi Teknik Sistem perkapalan
Fakultas Teknik Universitas Hasanuddin

Pada tanggal Juli 2022

Dan dinyatakan telah memenuhi syarat kelulusan

Menyetujui,

Pembimbing Utama,



Ir. Rahimuddin, ST., MT., Ph.D
NIP.197108251999031002

Pembimbing Pendamping,



Dr. Ir. Ganding Sitepu, Dipl-Ing
NIP.196004251988111001

Ketua Departemen Teknik Sistem Perkapalan



Dr. Eng. Faisal Mahmudin, ST., M.Inf., M Eng.
NIP.198102112005011003

PERNYATAAN KEASLIAN

Yang bertanda tangan dibawah ini:

Nama Mahasiswa : Muhammad Assiddiq
NIM : D091171013
Program Studi : Departemen Teknik Sistem Perkapalan
Jenjang : S1

Menyatakan dengan ini bahwa karya tulis saya yang berjudul:

“SIMULASI KENDALI TRIM PADA PELUNCURAN JACKET DENGAN
PENGATURAN BALLAST BARGE”

Adalah karya tulisan saya sendiri dan bukan merupakan pengambilan alihan tulisan orang lain bahwa skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri.

Apabila dikemudian hari terbukti atau dapat dibuktikan bahwa sebagian atau keseluruhan skripsi ini hasil karya orang lain, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Gowa, Agustus 2022

Yang membuat pernyataan,



Muhammad Assiddiq

KATA PENGANTAR



Alhamdulillah Rabbil Alamin. Segala puji bagi-Mu Ya Rabb, puji syukur penulis panjatkan kehadiran Allah SWT yang telah memberikan rahmat, hidayah, kekuatan serta kemudahan dalam menghadapi segala kesulitan, melapangkan segala kesempitan, menenangkan segala kegundahan, serta memberi petunjuk kepada penulis sehingga dapat menyelesaikan skripsi ini dengan judul **“Simulasi Kendali Trim pada Peluncuran Jacket dengan Pengaturan Ballast Barge”** Penulisan skripsi ini dilakukan dalam rangka memenuhi salah satu syarat untuk mendapat gelar Sarjana Teknik (ST), pada Universitas Hasanuddin.

Terwujudnya skripsi ini tidak lepas dari bantuan berbagai pihak yang telah membantu dan membimbing penulis, baik itu berupa tenaga, ide-ide, semangat maupun doa. Oleh karena itu dalam kesempatan ini penulis ingin mengucapkan terima kasih yang sebesar-besarnya kepada :

1. Kedua orang tua tercinta, ibunda Sitti Aminah, S.Sos dan ayahanda Drs. H. Hafid Abdul Razak, M.Si yang telah memberikan segala tenaga dan doanya untuk memberikan yang terbaik bagi penulis, yang telah berjuang memberikan pendidikan terbaik bagi penulis ditengah kekurangan mereka. Terima kasih pula kepada saudara saya Almadyah Chaerunnisa dan Ibnu Fariq serta keluarga saya, atas semua dukungan dan semangatnya selama ini.
2. Bapak Ir. Rahimuddin ST., MT., Ph.D dan Bapak Dr. Ir. Ganding Sitepu, Dipl-Ing. yang telah berkenan membimbing, mengarahkan serta memberi kemudahan bagi penulis dalam menyelesaikan skripsi ini. Terima kasih banyak untuk waktu, tenaga, pikiran dan segenap perhatian yang kalian berikan kepada penulis.
3. Bapak Andi Haris Muhammad, ST., MT., Ph.D dan Ibu Hariyanti Rivai., ST, MT., Ph.D selaku penguji dalam tugas skripsi ini. Terima kasih untuk segala masukan dan motivasi yang telah diberikan serta bantuan bimbingannya

dalam penyelesaian skripsi ini.

4. Bapak Dr.Eng. Faisal Mahmuddin, ST, MT. Selaku Ketua Departemen dan pembimbing akademik penulis serta seluruh dosen dan staff Teknik Sistem Perkapalan yang telah membantu penulis selama menempuh pendidikan di kampus.
5. Kepada seluruh sahabat-sahabat penulis yang selalu menemani dan memberi semangat kepada penulis.
6. Kepada Teman – teman Angkatan 2017 Teknik Sistem Perkapalan, terima kasih atas segala bantuan, dukungan dan semangat kebersamaannya selama 5 tahun ini.
7. Kepada seluruh kanda-kanda senior dan dinda junior yang senantiasa memotivasi dalam penyelesaian ini khususnya member Labo Listrik dan Kendali Kapal yang telah banyak membantu penulis.
8. Dan kepada semua pihak yang telah membantu dalam penyelesaian skripsi ini yang tidak dapat penulis sebutkan satu persatu, terima kasih atas semua saran, arahan, bimbingan, semangat serta doanya.

Penulis menyadari bahwa skripsi ini masih banyak terdapat kekurangan dan kesalahan, oleh karena itu penulis memohon maaf dan senantiasa menerima kritik dan saran yang membangun dari berbagai pihak demi perbaikan-perbaikan kedepannya. Kiranya tugas akhir ini dapat bermanfaat bagi semua pihak yang berkenan untuk membaca dan mempelajarinya. Amin Yaa Rabbal' Alamiin.

Gowa, Agustus 2022

Muhammad Assiddiq

DAFTAR ISI

LEMBAR PENGESAHAN	i
PERNYATAAN KEASLIAN.....	ii
DAFTAR ISI.....	v
DAFTAR GAMBAR	vii
DAFTAR TABEL.....	x
ABSTRAK	xi
ABSTRACT.....	xii
BAB 1. PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah.....	2
1.4 Tujuan Penelitian.....	3
1.5 Manfaat Penelitian.....	3
1.6 Sistematika Penulisan.....	3
BAB 2. TINJAUAN PUSTAKA	5
2.1 <i>Jacket Platform</i>	5
2.2 Metode Instalasi Jacket.....	6
2.3 Peluncuran Jacket (<i>Jacket Launching</i>)	7
2.4 Stabilitas Kapal.....	10
2.5 Komputasi Titik CG (<i>center of gravity</i>).....	11
2.6 Komputasi Titik CB (<i>center of buoyancy</i>)	12
2.7 Trim	13
2.8 Tekanan Hidrostatik.....	14
2.9 Gaya Apung.....	15
2.10 Transformasi Geometri.....	16
2.10.1 Translasi.....	16
2.10.2 Rotasi	16
2.11 Interpolasi Linier	17

2.12 Poligon.....	18
2.13 Perkalian Silang Vektor (<i>Cross Vector Product</i>).....	19
2.14 Penerapan Hukum III Newton.....	20
2.15 Scilab.....	21
2.16 Maxsurf.....	22
BAB 3. METODOLOGI.....	24
3.1 Lokasi dan Waktu Penelitian.....	24
3.2 Studi Literatur.....	24
3.3 Pengumpulan data.....	24
3.4 Pembuatan Model.....	26
3.5 Pembuatan program dan desain <i>interface</i>	27
3.6 Flow Chart.....	28
BAB 4. PEMBAHASAN.....	29
4.1 Pembuatan Model.....	29
4.1.1. Pemodelan Barge.....	29
4.1.2 Pemodelan Jacket.....	30
4.2 Validasi Model.....	31
4.3 Fungsi Program dan GUI.....	34
4.3.1. Fungsi program.....	34
4.2. Desain GUI.....	52
4.4 Hasil Perhitungan Volume & Level Tangki.....	53
4.4.1 Hasil Perhitungan Volume Tangki.....	54
4.4.2 Hasil Perhitungan Level Tangki.....	60
4.5 Pergeseran CG memanjang Kapal.....	63
4.6 Kondisi Barge Setelah Jacket Launching.....	65
4.7 Stabilitas Memanjang Barge.....	65
BAB 5. PENUTUP.....	71
5.1 Kesimpulan.....	71
5.2 Saran.....	71
DAFTAR PUSTAKA.....	72
LAMPIRAN.....	73

DAFTAR GAMBAR

Gambar 2.1 <i>Jacket Platform</i>	5
Gambar 2.2 <i>Ballasting Stage</i>	8
Gambar 2.3 <i>Sliding Stage</i>	8
Gambar 2.4 <i>Tipping Stage</i>	9
Gambar 2.5 <i>Self-righting Stage</i>	9
Gambar 2.6 Stabilitas Kapal	11
Gambar 2.7 Lambung Kapal Dibagi Menjadi Beberapa Grid	12
Gambar 2.8 Trim Kapal	13
Gambar 2.9 Bidang Poligon.....	18
Gambar 2.10 Vektor hasil Perkalian Silang.....	19
Gambar 2.11 Pembentukan Panel	20
Gambar 2.12 Tampilan Scilab.....	22
Gambar 3.1 Jacket.....	26
Gambar 4.1 Model Barge pada Maxsurf.....	29
Gambar 4.2 Model Barge pada Scilab	30
Gambar 4.3 Model Jacket pada Scilab.....	31
Gambar 4.4 Grafik Perbedaan Hasil Displacement	32
Gambar 4.5 Grafik Error LCB Model	33
Gambar 4.6 Tampilan Scinote.....	34
Gambar 4.7 <i>Flowchart</i> genof()	35
Gambar 4.8 <i>Flowchart</i> geotransformation()	36
Gambar 4.9 <i>Flowchart</i> plotMesh()	37

Gambar 4.10 <i>Flowchart</i> MeshPipeOffset()	39
Gambar 4.11 <i>Flowchart</i> arrangeData()	40
Gambar 4.12 <i>Flowchart</i> generateMeshJacket()	41
Gambar 4.15 <i>Flowchart</i> buoyancy()	45
Gambar 4.16 <i>Flowchart</i> run()	50
Gambar 4.17 <i>Flowchart</i> postLaunch()	52
Gambar 4.18 Tampilan GUI	53
Gambar 4.19 Tangki-tangki pada Barge	54
Gambar 4.20 Grafik hubungan Volume tangki dan pergeseran jacket pada Trim 2.00°	55
Gambar 4.21 Grafik hubungan Volume tangki dan pergeseran jacket pada Trim 2.5 ⁰	56
Gambar 4.22 Grafik hubungan Volume tangki dan pergeseran jacket pada Trim 3 ⁰	57
Gambar 4.23 Grafik hubungan volume tangki dan pergeseran jacket pada Trim 3.5 ⁰	58
Gambar 4.24 Grafik hubungan Volume tangki dan pergeseran jacket pada kemiringan 4 ⁰	59
Gambar 4.25 Simulasi Perubahan Level Air pada Setiap Tangki Ballast Ketika Kapal Mengalami Trim Buritan 2 ⁰	61
Gambar 4.26 Simulasi Perubahan Level Air pada Setiap Tangki Ballast Ketika Kapal Mengalami Trim Buritan 2.5 ⁰	61
Gambar 4.27 Simulasi Perubahan Level Air pada Setiap Tangki Ballast Ketika Kapal Mengalami Trim Buritan 3 ⁰	62
Gambar 4.28 Simulasi Perubahan Level Air pada Setiap Tangki Ballast Ketika Kapal Mengalami Trim Buritan 3.5 ⁰	62

Gambar 4.29 Simulasi Perubahan Level Air pada Setiap Tangki Ballast Ketika Kapal Mengalami Trim Buritan 4°.....	63
Gambar 4.30 Grafik Perpindahan CG memanjang Barge pada setiap Trim.....	64
Gambar 4.31 Derajat Trim Haluan setelah Jacket Launching	65
Gambar 4.32 Grafik Momen Penegak saat Trim Buritan 2 derajat	66
Gambar 4.33 Grafik Momen Penegak saat Trim Buritan 2.5 derajat	67
Gambar 4.34 Grafik Momen Penegak saat Trim Buritan 3 derajat	68
Gambar 4.35 Grafik Momen Penegak saat Trim Buritan 3.5 derajat	69
Gambar 4.36 Grafik Momen Penegak saat Trim Buritan 4 derajat	70

DAFTAR TABEL

Tabel 3.1 Parameter Barge C231	24
Tabel 3.2 Parameter Jacket	26
Tabel 4.1 Displacement Barge saat even keel di setiap draft.....	30
Tabel 4.2 Perbedaan displacement model pada Scilab dan Maxsurf.....	32
Tabel 4.3 Perbedaan LCB model pada Scilab dan Maxsurf	33
Tabel 4.4 Hasil Perhitungan Volume Tangki pada Kemiringan Trim 2°.....	55
Tabel. 4.5 Hasil Perhitungan Volume Tangki pada Kemiringan Trim 2.5°.....	56
Tabel. 4.6 Hasil Perhitungan Volume Tangki pada Kemiringan Trim 3°.....	57
Tabel 4.7 Hasil Perhitungan Volume Tangki pada Kemiringan Trim 3.5°.....	58
Tabel 4.8 Hasil Perhitungan Volume Tangki pada Kemiringan Trim 4°.....	59
Tabel 4.9 Level Air Ballast pada masing – masing Tangki	60
Tabel 4.10 Posisi titik CB pada setiap Pergeseran Jacket.....	63
Tabel 4.11 Derajat Trim Haluan setelah <i>Jacket Launching</i>	65
Tabel 4.12 Momen Penegak saat Stabil pada Trim Buritan 2 derajat.....	62
Tabel 4.13 Momen Penegak saat Stabil pada Trim Buritan 2.5 derajat.....	63
Tabel 4.14 Momen Penegak saat Stabil pada Trim Buritan 3 derajat.....	64
Tabel 4.15 Momen Penegak saat Stabil pada Trim Buritan 3.5 derajat.....	65
Tabel 4.16 Momen Penegak saat Stabil pada Trim Buritan 4 derajat.....	66

SIMULASI KENDALI TRIM PADA PELUNCURAN JACKET DENGAN PENGATURAN BALLAST BARGE

Oleh Muhammad Assiddiq

Departemen Teknik Sistem Perkapalan Fakultas Teknik Universitas Hasanuddin
Pembimbing I Ir. Rahimuddin, ST., MT., Ph.D
Pembimbing II Dr. Ir. Ganding Sitepu., Dipl-Ing

ABSTRAK

Peluncuran jacket merupakan proses yang singkat tapi sangat kritis karena memiliki resiko yang besar. Jika terjadi kegagalan dalam proses Peluncuran jacket, maka dapat menyebabkan kerusakan atau tenggelamnya struktur. Peluncuran jacket melibatkan berbagai parameter, meliputi *environmental conditions, launch barge specification, ballast, trim angle*, dan lainnya. Sudut trim barge dapat diatur dengan proses *ballasting*. Sudut trim barge akan berpengaruh pada proses peluncuran jacket, salah satunya yaitu terhadap kecepatan jacket saat meluncur. Untuk itu diperlukan pengaturan ballast terhadap barge untuk mendapatkan sudut trim barge yang diinginkan. Penelitian ini dilakukan dengan membuat simulasi, pemodelan pada barge dan jacket serta fungsi program dibuat dengan menggunakan software komputasi Scilab. Penelitian ini dilakukan dengan memvariasikan derajat trim kemiringan antara 2° hingga 4° dengan pengambilan disetiap penambahan 0.5° . Penambahan air ballast dilakukan pada tangki- tangki yang berada didepan titik CG dari barge yang bertujuan untuk menyeimbangkan momen yang berasal dari jacket dengan mempertahankan kemiringan trim yang diinginkan. Hal ini juga bertujuan untuk menurunkan titik CG secara vertikal dan tidak terlalu memberatkan struktur barge. Hasil penelitian ini, didapatkan volume dan level air ballast untuk setiap tangki dengan sudut trim yang diinginkan serta perpindahan *center of gravity* (CG). Selain itu, fungsi program dihasilkan dan dapat digunakan untuk perhitungan lainnya.

Kata Kunci : Peluncuran jacket, Sudut Trim, Barge, Ballast

SIMULATION OF TRIM CONTROL ON JACKET LAUNCHING WITH BALLAST BARGE SETTING

By Muhammad Assiddiq

Department of Marine Engineering, Faculty of Engineering, Hasanuddin
University

1st Supervisor Ir. Rahimuddin, ST., MT., Ph.D

2nd Supervisor Dr. Ir. Ganding Sitepu Dipl-Ing

ABSTRACT

Jacket launching is a short process but very critical because it has a big risk. If there is a failure in the Jacket launching process, it can cause damage or sinking of the structure. Jacket launching involves various parameters, including environmental conditions, launch barge specification, ballast, trim angle, and others. The angle of the trim barge can be adjusted by ballasting process. The size of the angle of the trim barge will affect the process of launching jacket, one of which is the speed of the jacket when sliding. For this reason, it is necessary to adjust the ballast to the barge to get the desired trim barge angle. This research was conducted by making simulations, modeling on the barge and jacket as well as program functions made using Scilab computing software. This research was carried out by varying the degree of trim angle from 2° to 4° by taking each addition of 0.5°. Ballast water was added to the tanks in front of the CG point of the barge which aims to balance the moment coming from the jacket by maintaining the desired trim slope. It also aims to lower the CG point vertically and not burden the barge structure too much. The results of this study, obtained the volume and level of ballast water for each tank with the desired trim angle and center of gravity (CG) displacement. In addition, program functions are generated and can be used for other calculations.

Keyword : Jacket Launching, Ballast, Trim Angle, Barge

BAB 1. PENDAHULUAN

1.1 Latar Belakang

Perut bumi mengandung sumber minyak dan gas bumi yang berada sebagian dibawah daratan dan sebagian berada di bawah dasar laut. Sumber minyak dan gas bumi yang diperkirakan 30% berada daratan dan 70% berada dasar laut. Dalam melakukan eksploitasi sumber minyak dan gas bumi yang berada dibawah dasar laut menjadi tantangan yang tidak mudah karena gelombang air laut dan kedalaman laut sehingga memerlukan peralatan yang khusus. Sumber minyak dan gas bumi tersebut dieksploitasi menggunakan peralatan lepas pantai yaitu *offshore platform* atau anjungan lepas pantai. *Offshore Platform* adalah struktur atau bangunan yang dibangun di galangan dan dipasang di lepas pantai untuk mendukung proses eksplorasi atau eksploitasi bahan tambang (minyak dan gas bumi).

Pembuatan *offshore platform* dilakukan pada lapangan fabrikasi yang umumnya berlokasi di daerah pantai. Pada umumnya jarak antara tempat fabrikasi dan lokasi akhir (tempat beroperasi) sangatlah jauh, dapat berupa lintas negara hingga lintas benua. Sebagai contoh anjungan TLP West Seno. Struktur dibangun di perusahaan Hyundai Heavy Industry, Korea Selatan, sedangkan lokasi operasinya berada di Selat Makasar, Indonesia. Dalam memindahkan struktur dari tempat fabrikasi menuju tempat instalasi/tempat beroperasi menggunakan sebuah kapal angkut khusus atau barge (tongkang) yang memiliki daya apung besar untuk menopang struktur dan membawanya ke lokasi instalasi di lepas pantai.

Struktur yang berada pada tempat operasi akan dilakukan proses *launching*. Salah satu struktur yang menjadi perhatian dan kritis saat dilakukan proses launching ialah *jacket*. *Jacket launching* merupakan proses yang singkat tapi sangat kritis karena memiliki resiko yang besar. Jika terjadi kegagalan dalam proses *Jacket launching*, maka dapat menyebabkan kerusakan atau tenggelamnya struktur (Gerwick, 1999). Dalam proses peluncuran *jacket* ada beberapa hal yang harus

diperhatikan, diantaranya yaitu *stabilitas barge*, *rocker arm reactions*, dan *bottom clearance jacket* (Noble Denton, 2015)

Jacket launching melibatkan berbagai parameter, meliputi *environmental conditions*, *launch barge specification*, *ballast*, *trim angle*, dan lainnya. Sudut trim barge dapat diatur dengan proses *ballasting*. Sudut trim barge sangat mempengaruhi proses peluncuran jacket. jacket dapat meluncur ketika vektor gaya meluncur lebih besar dari gaya gesek. Perbedaan gaya tersebut mempengaruhi kecepatan jacket meluncur. Kecepatan jacket meningkat secara proporsional dengan penambahan sudut trim. Kecepatan jacket meluncur dapat dikendalikan dengan mengatur sudut trim barge melalui pengaturan ballast pada masing-masing tangki.

Dalam mendapatkan kecepatan meluncur jacket diperlukan rekayasa trim dan ballast yang ditentukan oleh karakteristik lambung barge dibawah air. Untuk menghitung karakteristik lambung barge dibawah air yang dapat menggunakan metode numerik yakni membagi lambung barge menjadi sel-sel kecil menggunakan grid biasa kemudian hukum fisika dapat menghitung koordinat pusat benda terapung meski berbentuk tidak teratur dengan pendekatan volume dan tekanan (Ueng, 2013).

Pada penelitian ini dibuat simulasi pengaturan ballast dan menghitung volume disetiap tangki ballast yang dibutuhkan untuk mendapatkan kondisi trim yang diinginkan. Simulasi menggunakan metode numerik untuk menghitung kondisi dan stabilitas barge. Aplikasi menggunakan software komputasi Scilab dan memanfaatkan fitur GUI.

1.2 Rumusan Masalah

Pada penelitian ini, masalah yang diteliti yaitu:

- a. Bagaimana membuat model barge dan jacket menggunakan scilab?
- b. Bagaimana program untuk menghitung volume tangki ballast saat keadaan trim?
- c. Bagaimana pengaturan ballast disetiap tangki untuk membuat kapal trim?

1.3 Batasan Masalah

Agar penelitian ini tidak terlalu melebar dari tujuan yang ingin dicapai,

maka ditentukan batasan permasalahan. Adapun batasan masalah dalam penelitian ini adalah:

- a. Simulasi menggunakan software Scilab
- b. Jacket dianggap hanya sebagai beban pada barge
- c. Gaya eksternal angin dan gelombang laut tidak diperhitungkan (kondisi air tenang)

1.4 Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah:

- a. Mendeskripsikan cara pembuatan model dengan menggunakan metode meshing pada scilab
- b. Membuat program untuk menghitung volume tangki ballast saat keadaan trim
- c. Menganalisa dan mengetahui volume tangki ballast pada barge untuk mendapatkan *trim* yang diinginkan

1.5 Manfaat Penelitian

Adapun manfaat dari penelitian ini adalah untuk memberikan pemahaman kepada para pembaca, mengenai salah satu proses instalasi bangunan lepas pantai tipe *fixed jacket platform* yaitu dengan metode launching. Selain itu dengan melakukan penelitian ini diharapkan dapat memberikan informasi mengenai proses launching yang lebih aman, efektif, efisien dan resiko yang rendah serta sebagai bahan masukan dalam rangka pengembangan keilmuan, khususnya pada *marine operation* dan *installation industry* yang semakin hari semakin berkembang.

1.6 Sistematika Penulisan

Adapun sistematika penulisan dari penyusunan penelitian ini adalah:

BAB I PENDAHULUAN

Berisikan latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, dan sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Berisikan penjelasan teori – teori yang berkaitan dan menunjang penyelesaian penelitian.

BAB III METODOLOGI PENELITIAN

Berisikan lokasi dan waktu penelitian, metode dan alur atau tahapan – tahapan dalam penyelesaian penelitian.

BAB IV HASIL DAN PEMBAHASAN

Berisikan pembahasan dari hasil penelitian.

BAB V PENUTUP

Berisikan kesimpulan yang dari hasil penelitian dan saran – saran dari penulis

DAFTAR PUSTAKA

LAMPIRAN

BAB 2. TINJAUAN PUSTAKA

2.1 Jacket Platform

Struktur *jacket* adalah salah satu jenis struktur lepas pantai terpancang yang terdiri dari struktur *deck* (*main deck, cellar deck dan heli deck*), *jacket leg* sebagai penyangga *deck*, dan tiang pancang sebagai penyalur gaya seperti Gambar 2.1. Beban struktur platform ditopang beberapa tiang yang ditancapkan pada tanah. Penggunaan struktur *jacket* hanya terbatas untuk perairan - perairan yang tidak terlalu dalam, dan konfigurasi *deck* yang sederhana.



Gambar 2.1 Jacket Platform

Jacket platform merupakan struktur terpancang (*fixed structure*) yang terdiri atas beberapa komponen utama yaitu:

1. Geladak dan bangunan atas

Geladak adalah bagian yang berada diatas permukaan air dan merupakan fasilitas utama *jacket* yang menentukan fungsi struktur. Konstruksi geladak *jacket platform* berada pada ketinggian tertentu dari permukaan air tenang (SWL) sehingga tidak terjangkau gelombang laut. Konstruksi geladak terdiri atas beberapa kelompok jenis konstruksi

(*modules*) dimana jumlah, kelengkapan dan fasilitas yang ada pada geladak tergantung pada fungsi utama yang harus dilaksanakan oleh *jacket platform* itu sendiri.

2. *Jacket (Substructure)*

Jacket merupakan badan *platform* yang sebagian besar berupa konstruksi pipa (*tubular*). Struktur *jacket* sebagian besar terendam air hingga dasar laut. Fungsi utama struktur *jacket* adalah menopang konstruksi geladak dan fasilitas produksi yang ada, menahan struktur dari beban lateral dan momen guling akibat beban lingkungan (gelombang, arus, pasang surut).

3. Tiang Pancang (*Piles*)

Tiang pancang merupakan struktur *jacket platform* yang dipancangkan ke dalam dasar laut, hingga kedalaman 30 m - 150 m. Fungsi utama tiang pancang adalah sebagai pondasi struktur *jacket* yang menahan beban lateral dan aksial yang ditransformasikan ke tanah. Untuk itu karakteristik pondasi *jacket platform* selain ditentukan oleh perancangan tiang pancang itu sendiri juga ditentukan oleh kondisi tanah yang ada (*soil mechanics*).

2.2 Metode Instalasi Jacket

Karakteristik struktur *offshore* berbeda dengan struktur *onshore*. Struktur *onshore* dibangun dan diinstal di lokasi struktur beroperasi. Struktur *offshore* arus dibangun di galangan kemudian dipindahkan, diangkut dan diinstal di lokasi struktur tersebut akan beroperasi. Ada berbagai macam metode dalam proses pemindahan, pengangkutan dan instalasi struktur *offshore*. Metode instalasi yang sering digunakan pada dunia industri adalah metode *lifting*, *self floating*, dan *launching*. (Sugiono & Rofifah, 2004) metode tersebut harus diperhitungkan sebagai bagian dari rangkaian proses *load out*, *sea transport*, *upending*, *set down*, dan *piling*.

- a. Metode *lifting* salah satu proses instalasi struktur offshore yang terdiri dari *lifting off*, *lifting in the air*, *splash zone crossing*, *deep submerging*, dan *landing of the structure* dengan menggunakan *offshore support vessel*

(OSV) crane. Pada saat ini OSV crane lebih sering digunakan untuk melakukan *lifting operation* struktur *offshore* seperti modul offshore dan struktur deck.

- b. *Self floating* adalah metode yang digunakan untuk struktur jacket yang memiliki ukuran yang besar dengan memanfaatkan buoyancy dan temporary floating tanks kemudian ditarik oleh kapal tunda ke lokasi dimana jacket akan beroperasi.
- c. *Launching* adalah metode yang digunakan jika berat struktur jacket melebihi kapasitas crane. Struktur jacket diangkut dari galangan menuju lokasi instalasi menggunakan barge. Kemudian struktur tersebut akan diluncurkan ke dalam laut dengan cara mengisi tangki ballast hingga kemiringan tertentu.

2.3 Peluncuran Jacket (*Jacket Launching*)

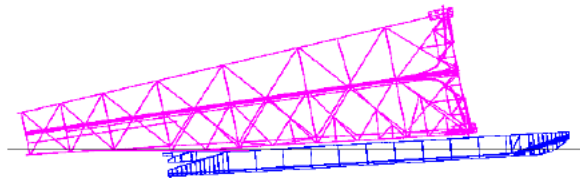
Peluncuran (*launching*) adalah salah satu proses instalasi jacket yang sangat penting dan memiliki resiko yang besar. Metode ini digunakan untuk struktur jacket yang beratnya melebihi kapasitas crane. Ketika jacket meluncur disepanjang skid-way maka draft dan trim pada launching barge berubah dan mempengaruhi respon motion pada barge. Ketika jacket mulai miring di atas rocker arm, tahap tersebut adalah tahap yang paling kritis dan beban pada rocker arm akan mencapai beban maksimum. Setelah jacket berpisah dari barge, maka jacket akan meluncur ke dalam air hingga kedalaman maksimum kemudian berosilasi beberapa kali hingga dalam keadaan stabil.

Nourpanah (2008) menyatakan bahwa analisis time -history pada metode ini diperlukan karena beberapa alasan yaitu: a) memastikan bahwa jacket tidak akan bertubrukan dengan mud-line, b) mengecek agar member jacket dapat menahan gaya reaksi yang ditimbulkan oleh rocker arm dan, c) mengecek agar member dapat menahan gaya-gaya hidrodinamika yang terjadi pada saat meluncur ke dalam laut.

Proses launching dilakukan pada kondisi laut yang tenang. Jacket diangkut menggunakan launch barge menuju lokasi instalasi. Kemudian launch barge diballast hingga mengalami trim dengan kemiringan (2° - 4°) dan sea fastening dipotong. Proses tersebut adalah proses awal pada analisa launching. Jacket

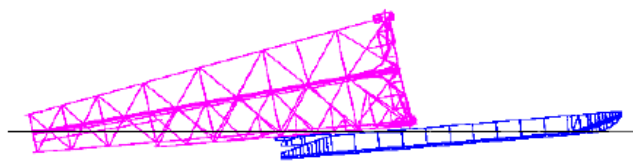
meluncur menuju rocker arm yang berada dibagian buritan launch barge. Setelah melewati rocker arm kemudian berotasi dan meluncur ke dalam laut. Setelah jacket dan barge berpisah maka jacket tersebut akan mengalami osilasi beberapa kali hingga dalam keadaan diam. Secara umum proses launching dapat dibagi ke dalam 4 tahap :

1. *Fase ballasting* yakni barge diballast hingga mencapai *trim* dan *draft* yang diinginkan. Barge di-trim berada pada sudut 2 sampai 4 derajat dimana stuktur tidak meluncur karena beratnya sendiri seperti Gambar 2.2. Hal ini dimaksudkan untuk mengamankan sudut tilting yang lebih besar selama mungkin.



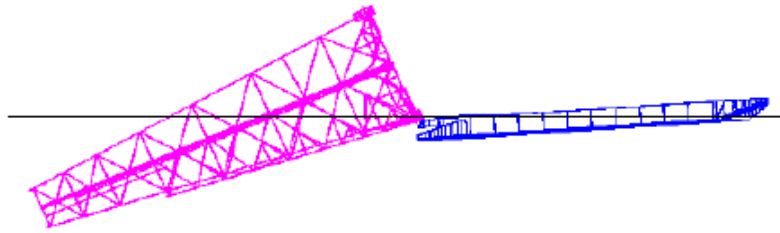
Gambar 2.2 *Ballasting Stage*

2. *Fase sliding* yakni jacket meluncur di atas barge akibat berat sendiri menuju *rocker arm* seperti Gambar 2.3.



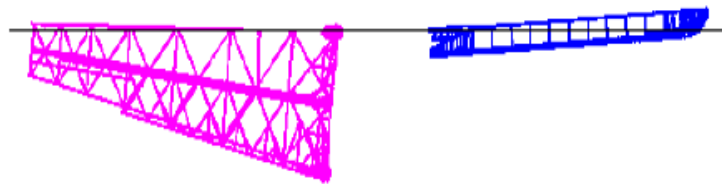
Gambar 2.3 *Sliding Stage*

3. *Tipping stage* yakni *jacket* meluncur dengan menggerakkan *rocker arm* kemudian berotasi hingga sudut maksimum yang diijinkan dan *jacket* pun meluncur tenggelam ke dalam air seperti Gambar 2.4.



Gambar 2.4 *Tipping Stage*

4. *Self-righting stage* yakni *jacket* berpisah dari barge kemudian mengalami osilasi beberapa kali hingga *jacket* dalam keadaan diam/stabil seperti Gambar 2.5.



Gambar 2.5 *Self-righting Stage*

Faktor-faktor yang mempengaruhi barge dan jacket pada saat launching sebagai berikut :

a. Barge

Kondisi pre-launch tergantung pada draft, trim dan panjang tilting beam yang berkontribusi berdasarkan parameter-parameter berikut :

- rocker tipping load
- panjang kaki jacket yang tersisa di atas rocker arm
- barge submerging depth
- kedalaman selam jacket
- kecepatan luncur brace horizontal
- stabilitas barge selama proses launching
- kekuatan longitudinal barge

b. Jacket

Parameter - parameter yang berpengaruh terhadap jacket selama proses launching ditentukan sebagai berikut :

- pusat gravitasi (center of gravity)
- pusat gaya apung (center of buoyancy)
- reserved buoyancy quantity
- panjang jacket/rasio panjang tilting beam

2.4 Stabilitas Kapal

Stabilitas adalah keseimbangan dari kapal, merupakan sifat atau kecenderungan dari sebuah kapal untuk kembali kepada kedudukan semula setelah mendapat senget (kemiringan) yang disebabkan oleh gaya - gaya dari luar (Rachman et al., 2014). Untuk mempelajari stabilitas maka harus memahami titik - titik penting pada stabilitas sebagai berikut:

a. Titik Berat (*Center of Gravity*)

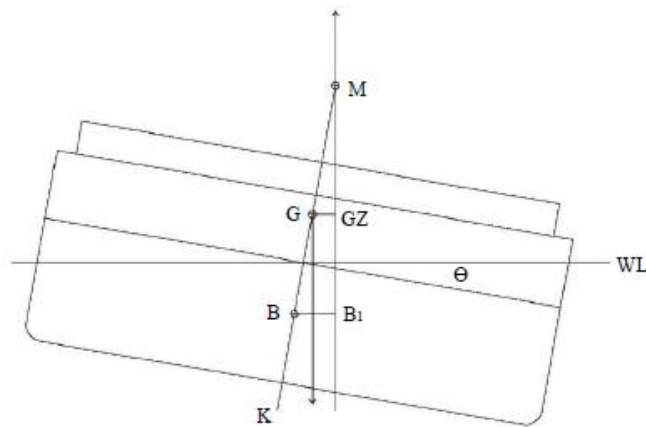
Titik berat (center of gravity) dikenal dengan titik G dari sebuah kapal, merupakan titik tangkap dari semua gaya - gaya yang menekan ke bawah terhadap kapal.

b. Titik Apung (*Center of Buoyancy*)

Titik apung (center of buoyancy) di kenal dengan titik B dari sebuah kapal, merupakan titik tangkap dari resultan gaya - gaya yang menekan tegak ke atas dari bagian kapal yang terbenam dalam air. Titik tangkap B bukanlah merupakan suatu titik yang tetap, akan tetapi akan berpindah - pindah oleh adanya perubahan sarat dari kapal. Titik B inilah yang menyebabkan kapal mampu untuk tegak kembali setelah mengalami senget.

c. Titik Metasentra (M)

Titik Metacentre (M) adalah sebuah titik yang tidak boleh dilampui oleh titik G agar stabilitas kapal positif. Titik M juga merupakan titik pusat olengan kapal. dapat dilihat pada Gambar 2.6 posisi dari setiap titik-titik penting stabilitas kapal.



Gambar 2.6 Stabilitas Kapal

Sebagaimana pada Gambar 2.6 saat kapal miring karena pengaruh gaya dari luar maka letak titik G akan tetap sedangkan letak titik B akan berpindah ke B1 karena adanya perubahan bentuk pada bagian badan kapal yang terbenam.

Dari Gambar 2.6 dapat dilihat adanya gaya sepasang sejajar yang sama dan arahnya berlawanan yaitu gaya G yang bekerja tegak lurus kebawah dan gaya B yang bekerja tegak lurus ke atas. Pasangan dua buah gaya tersebut merupakan sebuah kopel dengan lengan GZ yang akan menegakkan kapal kembali ke posisi semula setelah pengaruh gaya yang memiringkan kapal tidak bekerja lagi. Kopel tersebut disebut moment penegak (righting moment) dengan lengan GZ (righting arm). GZ merupakan jarak horizontal antara gaya yang melalui titik G dan titik B. Momen penegak (righting moment) dirumuskan pada Persamaan 1.

$$moment\ penegak = \Delta \times GZ \quad (1)$$

Dimana :

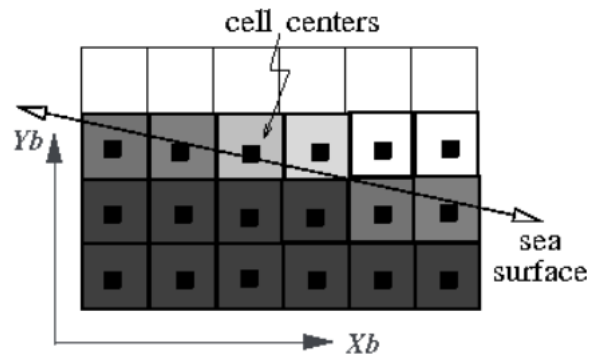
Δ = Displacement (ton)

GZ = lengan momen (m)

2.5 Komputasi Titik CG (center of gravity)

Untuk menghitung titik CG dapat menggunakan metode grid atau mengelompokkan panel. Dengan mengelompokkan grid yang signifikan dengan

koordinat X_b yang sama ke dalam irisan dan membagi badan kapal menjadi penampang L dalam arah X_b , seperti yang ditunjukkan pada Gambar 2.7. Persamaan 2 menerapkan hukum fisika untuk menghitung CG (Ueng, 2013).



Gambar 2.7 Lambung Kapal Dibagi Menjadi Beberapa Grid

$$CG_x = (\sum_{i=1}^L m_i \cdot x_i) / M_s \quad (2)$$

Dimana CG_x adalah titik CG pada sumbu x , L adalah panjang dari kapal, m_i adalah massa grid ke- i , x_i adalah koordinat x dari pusat grid ke- i , dan M_s adalah massa kapal. pendekatan yang sama digunakan untuk menghitung CG_z yaitu titik CG kapal secara vertikal.

2.6 Komputasi Titik CB (*center of buoyancy*)

Pusat daya apung B adalah pusat geometris dari bagian yang terbenam dari badan kapal. Jika bagian yang terbenam telah diketahui, algoritma untuk menghitung G dapat digunakan untuk menghitung koordinat benda dengan membagi lambung kapal menjadi beberapa grid seperti Gambar 2.7. Jika pusat grid lebih rendah dari permukaan laut lebih dari setengah lebar grid, grid benar-benar terbenam. Jika pusat grid lebih tinggi dari permukaan laut lebih dari setengah lebar grid, grid seluruhnya berada di atas permukaan laut dan volume terendamnya adalah nol (Ueng, 2013). Hal tersebut dapat diatasi dengan melakukan interpolasi pada titik grid sehingga meningkatkan akurasi perhitungan. Untuk menghitung CB dapat menggunakan Persamaan 3.

$$CB_x = (\sum_{i=1}^L \rho V_i x_i) / M_s \quad (3)$$

Dimana CB_x adalah titik CB pada sumbu x. L adalah panjang kapal. ρ adalah massa jenis air laut, V_i adalah volume grid ke-i bagian kapal yang tenggelam. M_s adalah massa kapal.

2.7 Trim

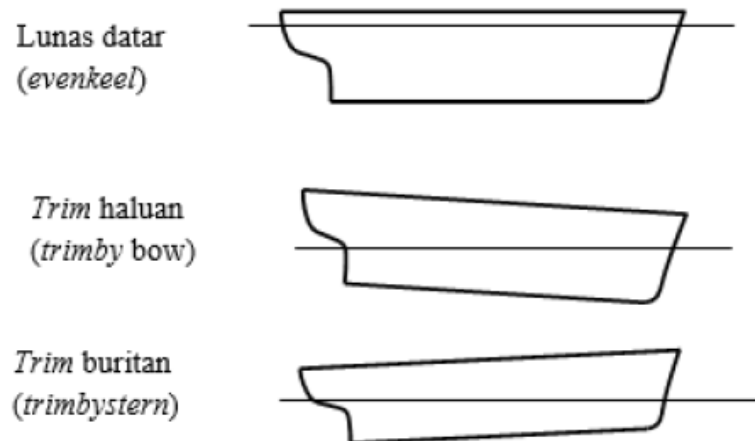
Trim didefinisikan sebagai perbedaan antara sarat di AP (T_A) dan sarat di FP (T_F).

$$Trim = T_A - T_F \quad (4)$$

Berdasarkan persamaan 2, hasil positif menunjukkan terjadi Trim Buritan. Pada saat kapal dalam kondisi trim, displasemen dan kecepatan kapal tetap konstan, tidak ada penambahan ballast dan kapal tidak mengalami penambahan kecepatan meskipun hambatan berubah ketika kondisi trim (Larsen, 2012).

Secara umum terdapat tiga kondisi yang dapat dialami kapal seperti Gambar 2.8, yaitu:

- Evenkeel yaitu draft depan kapal sama dengan draft belakang kapal
- Trim haluan yaitu draft belakang lebih kecil dari draft depan
- Trim buritan yaitu draft belakang lebih besar dari draft depan



Gambar 2.8 Trim Kapal

Trim pada kapal terpenuhi apabila selisih antara Longitudinal Center of Bouyancy (LCB) dan Longitudinal Center of Grafity (LCG) nilainya tidak lebih

dari 1 % Longitudinal Between Perpendicular (LPP) dari kapal tersebut. Dan untuk proses *jacket launching* barge ditrim 2 hingga 4 derajat.

2.8 Tekanan Hidrostatik

Tekanan P, didefinisikan sebagai gaya normal per satuan luas permukaan. Suatu fluida yang mengalami tekanan akan mengerahkan sebuah gaya pada setiap permukaan yang bersentuhan dengan fluida tersebut, hal ini sesuai dengan prinsip Archimedes. Untuk suatu cairan dengan sebuah permukaan bebas (*free surface*), maka h adalah selisih jarak dari permukaan bebas tersebut sampai titik tertentu dibawahnya (kedalaman) dimana tekanan akan diukur. Dengan tekanan total P, tekanan lingkungan (atmosfer bumi) P_0 , gravitasi g, dan kedalaman h, maka persamaan tekanan dapat dilihat pada Persamaan 3

$$P = P_0 + \rho \cdot g \cdot h \quad (5)$$

Keterangan :

P_0 = Tekanan Atmosfer (N/m^2)

P = tekanan hidrostatik (N/m^2)

ρ = massa jenis zat cair (kg/m^3)

g = percepatan gravitasi (m/s^2)

h = kedalaman (m)

Dari persamaan tersebut dapat diketahui pula hubungan antara tekanan, massa jenis, percepatan gravitasi dan kedalaman h yang semuanya menunjukkan hubungan yang berbanding lurus. Jadi, semakin besar nilai massa jenis maka semakin besar tekanannya, begitu juga jika percepatan gravitasi maupun kedalamannya semakin besar, maka nilai tekanan akan semakin besar. Sehingga jika suatu benda dicelupkan ke dalam wadah berisi air maka semakin dalam benda tersebut tercelup atau masuk ke dalam air, tekanan yang dialami benda pun semakin besar. Sementara jika suatu benda diubah ukurannya tidak akan menyebabkan tekanan / gaya yang dialami benda berubah selama massa jenisnya tetap. Contohnya jika mula-mula sebuah plastisin sebesar bola tenis dimasukkan ke dalam wadah

berisi air tenggelam, maka jika ukurannya diubah menjadi sebesar bola pingpong dan dimasukkan ke dalam wadah berisi air yang sama juga akan tetap tenggelam, karena yang terjadi hanya perubahan ukuran, massa jenis sama sekali tidak berubah.

2.9 Gaya Apung

Berdasarkan prinsip Archimedes, bila sebuah benda seluruhnya atau sebagian dicelupkan di dalam suatu fluida yang diam, maka fluida tersebut akan mengerahkan tekanan pada setiap bagian permukaan benda yang bersentuhan dengan fluida tersebut. Tekanan tersebut adalah lebih besar pada bagian benda yang tercelup lebih dalam. Resultan semua gaya adalah sebuah gaya yang mengarah ke atas yang dinamakan kakas apung (buoyancy) dari benda yang tercelup. Gaya resultan, yang mengarah ke atas pada benda tersebut akan menyamai beratnya dan akan beraksi secara vertikal yang arahnya ke atas melalui pusat gravitasinya (Ongga et al., 2009). Persamaan 4 merupakan persamaan umum dari rumus tekanan. Berdasarkan persamaan tersebut dapat dihitung gaya apung (F_A) pada Persamaan 5, karena gaya apung memiliki arah vertikal, maka luas penampang (A) merupakan luas bidang pada sumbu x dan y dimana tegak lurus terhadap gaya apung.

$$P = \frac{F}{A} \quad (6)$$

$$F_A = \rho \cdot g \cdot h \cdot A \quad (7)$$

Keterangan :

P = Tekanan (N/m^2)

g = Percepatan gravitasi (m/s^2)

A = Luas penampang (m^2)

h = Kedalaman (m)

F_A = Gaya apung (N)

ρ = Massa jenis (kg/m^3)

Kemampuan daya apung ialah gaya yang bekerja tegak lurus keatas melalui pusat titik apung ,dengan kekuatan yang sama dengan berat air yang dipindahka. Dengan hukum Archimedes,bahwa sebuah benda mengapung memindahkan air sama dengan beratnya sendiri, tekanan ini sama dengan berat kapal sehingga dapat mencari gaya apung berdasarkan Persamaan 6.

$$F_A = \gamma \cdot \nabla \quad (8)$$

Keterangan :

F_A = Gaya apung (N)

γ = Berat Jenis (Ton/m³) = $\rho \cdot g$

∇ = Volume displacement (m³)

2.10 Transformasi Geometri

2.10.1 Translasi

Translasi atau pergeseran merupakan jenis dari transformasi geometri di mana terjadi perpindahan atau pergeseran dari suatu titik ke arah tertentu di dalam sebuah garis lurus bidang datar. Akibatnya, setiap bidang yang ada di garis lurus tersebut juga akan digeser dengan arah dan jarak tertentu (Astutuik, 2014). Translasi pada dasarnya hanya mengubah posisi, bukan bentuk dan ukuran dari bidangnya. Persamaan 4 merupakan persamaan dari translasi

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} a \\ b \\ c \end{bmatrix} + \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (9)$$

Keterangan :

x', y', z' = titik bayangan

x, y, z = titik asal

a, b, c = vektor translasi

2.10.2 Rotasi

Rotasi atau juga dikenal dengan perputaran dalam transformasi geometri sesuai dengan namanya berarti sebuah perputaran yang ditentukan oleh titik pusat rotasi, arah rotasi, dan juga besar dari sudut rotasi. Prinsipnya adalah memutar terhadap sudut dan titik pusat yang memiliki jarak yang sama dengan titik yang diputar. Karena hanya berputar, maka transformasi ini tidak mengubah bentuk atau ukuran dari sebuah bidang (Astutuik, 2014).

$$R_x = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{bmatrix} \quad (10)$$

$$R_y = \begin{bmatrix} \cos(\beta) & 0 & \sin(\beta) \\ 0 & 1 & 0 \\ -\sin(\beta) & 0 & \cos(\beta) \end{bmatrix}. \quad (11)$$

$$R_z = \begin{bmatrix} \cos(\gamma) & -\sin(\gamma) & 0 \\ \sin(\gamma) & \cos(\gamma) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (12)$$

$$R = R_x \cdot R_y \cdot R_z. \quad (13)$$

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = R \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (14)$$

Keterangan :

x, y, z = titik awal

x', y', z' = titik bayangan

R = Koefisien matriks rotasi

α = sudut rotasi sumbu x

β = sudut rotasi sumbu y

γ = sudut rotasi sumbu z

2.11 Interpolasi Linier

Interpolasi linier adalah cara mendapatkan nilai di antara dua data berdasarkan persamaan linier. Untuk dapat melakukannya maka minimal harus diketahui dua buah data. Jika diketahui nilai $P(x_1, y_1)$ dan $T(x_2, y_2)$ maka dapat dihitung nilai y untuk nilai x yang telah diketahui dan berada diantara kedua data tersebut (nilai $P(x_1, y_1)$ dan $T(x_2, y_2)$). Sebaliknya apabila diketahui nilai y maka dapat dihitung nilai x -nya. Interpolasi linier merupakan metoda untuk penentuan nilai fungsi persamaan linier berdasarkan kesebandingan perubahan antara nilai x

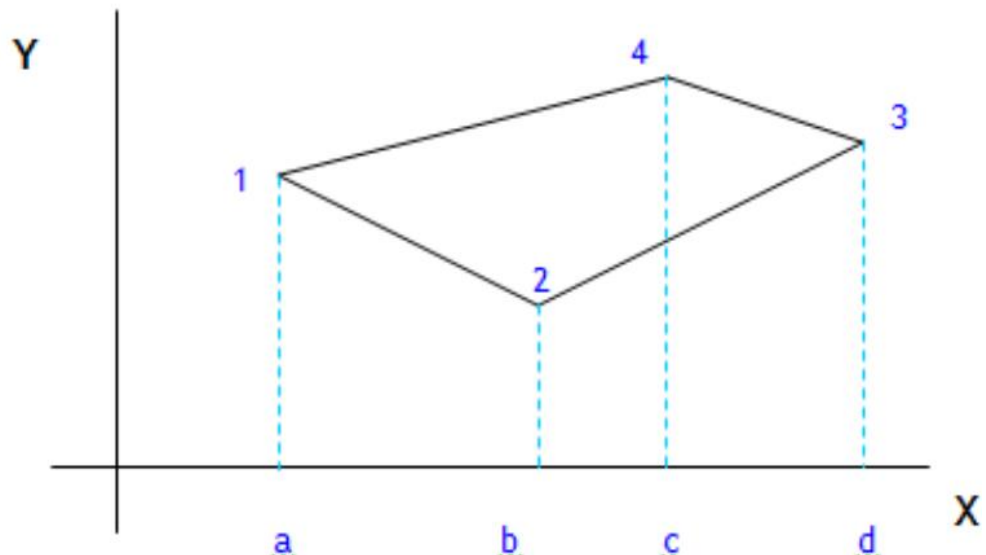
dan y (Amin et al., 2018). Persamaan 10 digunakan untuk menghitung nilai y dan x .

$$\frac{(y - y_2)}{(y_1 - y_2)} = \frac{(x - x_1)}{(x_2 - x_1)} \quad (15)$$

2.12 Poligon

Luas suatu wilayah merupakan luas pada bidang datar (X dan Y) tanpa ada unsur ketinggian (Z). Kondisi di lapangan yang tidak datar (tidak beraturan) tersebut diproyeksikan ke bidang datar sehingga yang tampak dalam gambar adalah bidang X dan Y. Akibat proyeksi tersebut, luas di gambar tampaknya lebih kecil dari pada luas di lapangan. Dalam menghitung suatu area dapat digunakan metode poligon.

Metode ini menggunakan titik ordinat, seperti poligon tertutup dengan koordinat masing-masing titik poligon diketahui seperti Gambar 2.9. Luas poligon tersebut adalah jumlah luas trapesium $14ca$ dan $43dc$ dikurangi jumlah trapesium $12ba$ dan $23db$ (ITS, n.d.).



Gambar 2.9 Bidang Poligon

Sehingga secara umum dapat disimpulkan bahwa untuk menghitung luas poligon sebanyak n titik dapat digunakan Persamaan 11 (tanda mutlak untuk menghindari luas yang negatif)

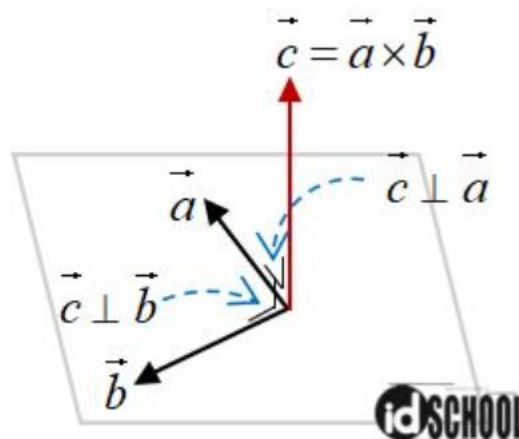
$$A = \left| \sum_{n=1}^{i=1} \left(\frac{x_i \cdot y_{i+1} - x_{i+1} \cdot y_i}{2} \right) + \frac{x_i \cdot y_i - x_i \cdot y_i}{2} \right| \quad (16)$$

2.13 Perkalian Silang Vektor (*Cross Vector Product*)

Perkalian vektor terdiri dari dua macam yaitu perkalian titik dan perkalian silang vektor. Perbedaan dari 2 jenis perkalian vektor terletak pada cara mengalikan dan hasilnya. Perkalian titik vektor (dot product) menghasilkan skalar berupa suatu nilai saja. Sementara perkalian silang vektor (cross product) menghasilkan suatu vektor berupa persamaan yang memiliki nilai bilangan dan arah. Kesimpulannya, perkalian vektor dan vektor dapat menghasilkan sebuah skalar atau sebuah vektor baru, bergantung dari jenis perkalian yang dilakukan (IdSchool, n.d.). Untuk rumus umum perkalian vektor seperti Persamaan 12.

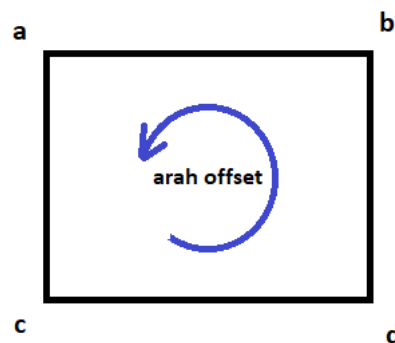
$$|\vec{a} \times \vec{b}| = |\vec{a}| \cdot |\vec{b}| \cdot \sin \theta \quad (17)$$

Pada aturan perkalian silang vektor (vector cross product) menghasilkan sebuah vektor baru yang tegak lurus dengan vektor yang dioperasikan. Misalnya pada perkalian silang vektor \vec{a} dan vektor \vec{b} menghasilkan vektor $\vec{c} = \vec{a} \times \vec{b}$. Vektor \vec{c} dari hasil perkalian tersebut adalah vektor yang tegak lurus dengan vektor \vec{a} , selain itu vektor \vec{c} juga tegak lurus dengan vektor \vec{b} seperti Gambar 2.10.



Gambar 2.10 Vektor hasil Perkalian Silang

Dalam menggabungkan titik-titik offset menjadi sebuah panel, arah putaran offset sangatlah berpengaruh pada hasil arah gaya normal yang terbentuk. Untuk mendapatkan arah gaya normal yang benar yakni gaya normal mengarah keluar dari sebuah benda maka digunakan kaidah tangan kanan seperti pada Gambar 2.11 sehingga untuk panel pada Gambar 2.11 memiliki offset a,c,d,b,a dimana a adalah (x_a, y_a, z_a) , c adalah (x_c, y_c, z_c) , d adalah (x_d, y_d, z_d) , dan b adalah (x_b, y_b, z_b)



Gambar 2.11 Pembentukan Panel

2.14 Penerapan Hukum III Newton

Hukum 3 Newton berbunyi jika suatu benda diberi gaya kepada benda lain maka benda yang dikenai gaya tersebut akan mengembalikan gaya kepada gaya yang memberi gaya tersebut dengan ukuran yang sama dan hal ini berlaku juga pada kapal. Ketika kapal memberikan gaya berat (W_{kapal}) sebagai gaya aksi ke air maka air tersebut akan mengembalikan gaya yg diberikan oleh kapal yang disebut gaya apung (F_b) sebagai gaya reaksi seperti pada Persamaan 18.

$$W_{kapal} = F_b \quad (18)$$

Pada penelitian ini, untuk mendapatkan keadaan stabil pada trim yang diinginkan (2° - 4°) maka jumlah momen total adalah 0. Untuk mendapatkan kondisi tersebut digunakan Persamaan 19. Persamaan tersebut memanfaatkan tekanan (Persamaan 7) untuk menghitung gaya apung yang ditimbulkan akibat lambung kapal yang terbenam.

$$\left(\frac{\int x \cdot db}{\int db} \right) \cdot \int \rho \cdot g \cdot h \cdot dA = a \cdot W_j - \left(\int \frac{L \cdot W_b}{W_b} \right) \cdot W_b \quad (19)$$

Dimana :

x	= titik apung (m)	a	= pergeseran jacket (m)
db	= gaya apung (N)	W_j	= berat jacket (N)
ρ	= massa jenis (kg/m^3)	l	= titik berat tangki (m)
g	= percepatan gravitasi (m/s^2)	W_b	= berat tangki (m)
dA	= luas penampang (m^2)	h	= kedalaman (m)

Untuk mendapatkan keadaan yang stabil pada trim yang diinginkan ($2^\circ - 4^\circ$) maka dilakukan transformasi rotasi pada offset barge menggunakan Persamaan 14. Ketika pergeseran jacket berubah (a) maka, komputer akan melakukan literasi hingga keadaan stabil dengan menghitung variasi luas penampang (dA) sebagai akibat perubahan offset dari transformasi geometri.

2.15 Scilab

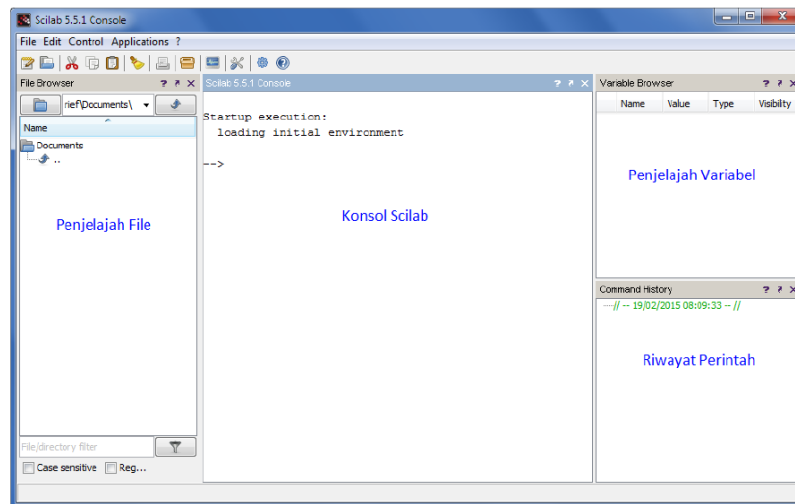
Scilab adalah sebuah perangkat lunak yang dirancang dan dikembangkan untuk komputasi numerik serta untuk visualisasi data secara dua dimensi maupun tiga dimensi. Scilab juga merupakan sebuah bahasa pemrograman tingkat tinggi yang berorientasi numerik. Scilab adalah suatu interpreter sehingga suatu kode program yang dibuat dapat dieksekusi secara langsung dan dilihat hasilnya tanpa harus melalui tahapan kompilasi (Arief, 2015).

Scilab adalah sebuah freeware yang dapat digunakan secara gratis untuk keperluan pribadi maupun komersial. Scilab tersedia dalam berbagai macam sistem operasi utama, seperti Windows (XP, Vista, 7, 8), Linux, serta MacOS X.

Alamat website Scilab adalah <http://www.scilab.org>. Pada website tersebut kita dapat memperoleh file instalasi Scilab, source code, dokumentasi tentang Scilab, modul-modul tambahan serta berbagai informasi lainnya yang berkaitan dengan Scilab.

Konsol Scilab (*Scilab Console*) pada Gambar 2.12 merupakan bagian utama dari Jendela Scilab. Sebagian besar interaksi antara kita dengan Scilab dilakukan

melalui jendela konsol Scilab. Konsol Scilab adalah tempat untuk memasukkan semua perintah diberikan kepada Scilab.



Gambar 2.12 Tampilan Scilab

2.16 Maxsurf

Bentley maxsurf adalah program computer yang dikembangkan oleh FormSys (Formating Design System), dibawah naungan Bentley System sejak tahun 2011, yang merupakan pengembangan dari beberapa program computer yaitu SACS, Maxsurf dan Moses (HMSP, n.d.).

Maxsurf adalah program aplikasi spesialis dalam bidang perkapalan dan galangan kapal, teknik lepas pantai dan rekayasa struktur. Program ini dapat memvisualisasikan, dan mengoptimalkan desain kapal dengan pengaturan lengkap yang telah di integrasikan.

Maxsurf terdiri dari beberapa sub-program aplikasi, yaitu:

1. Maxsurf Modeler
2. Maxsurf Motion
3. Maxsurf Resistance
4. Maxsurf Stability
5. Maxsurf Structure

6. Maxsurf Fitting

7. Maxsurf Link

8. Maxsurf Vpp

Beberapa fungsi pada program aplikasi ini seperti membuat bentuk lambung yang seimbang, sesuai dengan persyaratan stabilitas, tahanan kapal, seakeeping, dan kekuatan kapal.

BAB 3. METODOLOGI

Metode penelitian menggunakan metode numerik yaitu metode menyelesaikan masalah dengan operasi hitungan (aritmatika) yakni kurang, tambah, bagi dan kali. Pada penelitian ini dibuat model dengan proses meshing lambung *barge* yakni lambung *barge* dibagi menjadi elemen – elemen yang kecil atau diskrit.

3.1 Lokasi dan Waktu Penelitian

Penelitian ini dilakukan pada Laboratorium Listrik dan Kendali Departemen Teknik Sistem Perkapalan. Dalam hal waktu penelitian tugas akhir ini dilaksanakan selama 6 bulan.

3.2 Studi Literatur

Untuk mendukung proses penyelesaian penelitian ini, literatur-literatur yang dipelajari adalah jurnal-jurnal dan tesis yang berkaitan langsung dengan penelitian ini dan buku-buku sebagai tambahan referensi

3.3 Pengumpulan data

Tabel 3.1 dan Gambar merupakan data dari barge C231 milik PT. Comarco. Setelah didapatkan gambar tersebut maka dilakukan pembuatan model dengan menggunakan aplikasi Maxsurf untuk didapatkan data offset dari barge tersebut. Data offset tersebut digunakan untuk membuat meshing di Scilab.

- a. Data Barge C231 milik PT. Comarco

Adapun ukuran utama Barge C231 sebagai berikut:

Tabel 3.1 Parameter Barge C231

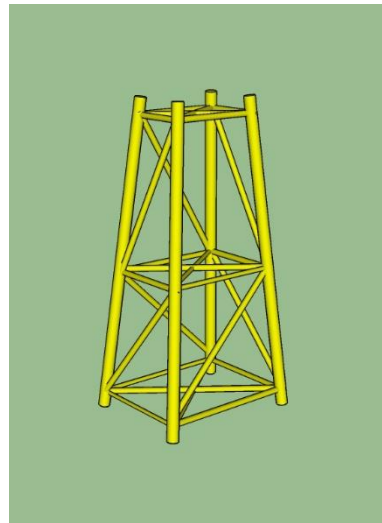
Parameter	Nilai
<i>Length Overall ((LOA)</i>	230'(70.105 m)
<i>Breadth Moulded</i>	60' (18.288)
<i>Depth Moulded</i>	14'(4.267 m)
<i>Summer Draft</i>	11'1" (3.378 m)
<i>Deck Loading</i>	10 MT/m ²
<i>Light Weight Tonnage (LWT)</i>	511 Ton
<i>LCG</i>	35.87 m

b. Data jacket fix platform

Adapun data dari *jacket* yang digunakan pada penelitian ini dapat dilihat pada Tabel 3.2, serta Gambar 3.2 merupakan gambar model dari *jacket*.

Tabel 3.2 Parameter Jacket

Parameter	Nilai
<i>Geometry</i>	<i>4 legged</i>
<i>Leg diameter</i>	1 m
<i>Bracing diameter</i>	0.5 m
<i>Elevation</i>	57 m



Gambar 3.2 Jacket

3.4 Pembuatan Model

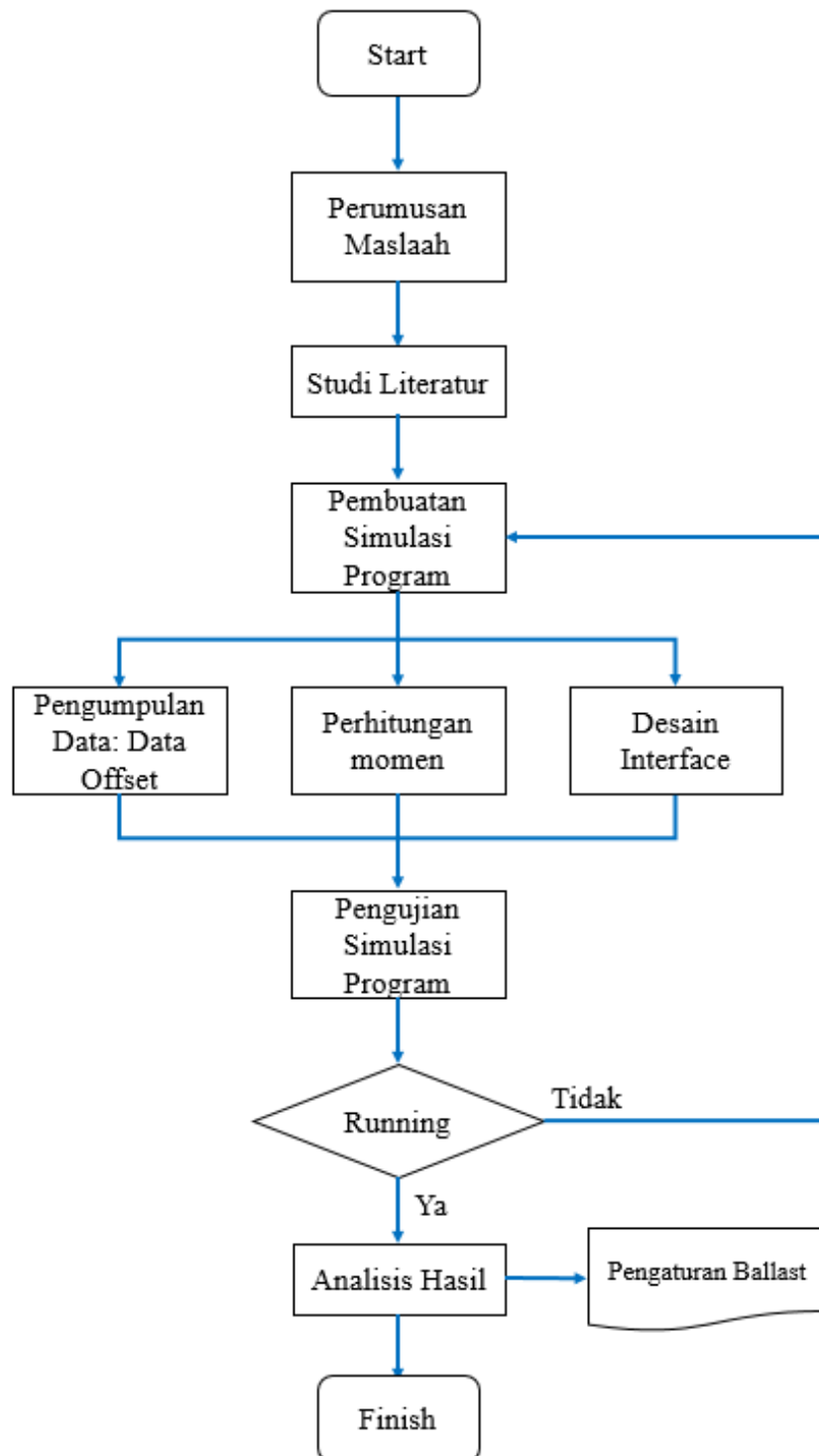
Dalam menyelesaikan suatu permasalahan *engineering* yang menggunakan perhitungan numerik dapat dilakukan dengan metode elemen/panel atau *meshing*. Pada penelitian dibuat dengan panel-panel hingga membentuk sebuah lambung barge yang sesuai dengan barge pada gambar *general arrangement*.

Pada penelitian ini model dibuat dengan menggunakan Scilab. Pada tahap ini, setiap panel dibuat dengan memerhatikan arah plot agar gaya normal disetiap panel tidak terbalik. Model *barge* dibuat dengan menggunakan fungsi yang telah dibuat yaitu `genof()`. Masukan dari fungsi tersebut ialah data offset dari barge. Sedangkan untuk keluaran yang dihasilkan ialah offset dan panel. Untuk pembuatan model jacket menggunakan fungsi `generateMeshJacket()`. Didalam fungsi tersebut terdapat lagi fungsi lainnya. Keluaran dari fungsi tersebut ialah offset dan panel.

3.5 Pembuatan program dan desain *interface*

Penelitian ini menggunakan Scilab untuk membuat program serta interface dari aplikasi yang dibuat. Untuk mendapatkan hasil penelitian yang diinginkan, program dibuat dengan menggunakan Persamaan 19 sebagai dasar berpikir. Persamaan tersebut memanfaatkan tekanan (Persamaan 7) untuk menghitung gaya apung yang ditimbulkan akibat lambung kapal yang terbenam. Untuk mendapatkan keadaan yang stabil pada trim yang diinginkan ($2^\circ - 4^\circ$) maka dilakukan transformasi rotasi pada offset barge menggunakan Persamaan 14. Ketika pergeseran jacket berubah (a) maka, komputer akan melakukan literasi hingga keadaan stabil dengan menghitung variasi luas penampang (dA) sebagai akibat perubahan offset dari transformasi geometri. Untuk menunjang jalannya simulasi maka dibuatlah beberapa program seperti plot gambar, transformasi geometri, plot kurva, generate offset, transformasi geometri dan lainnya. Selain scilab, aplikasi Maxsurf juga digunakan untuk memvalidasi model dan program yang telah dibuat.

3.6 Flow Chart

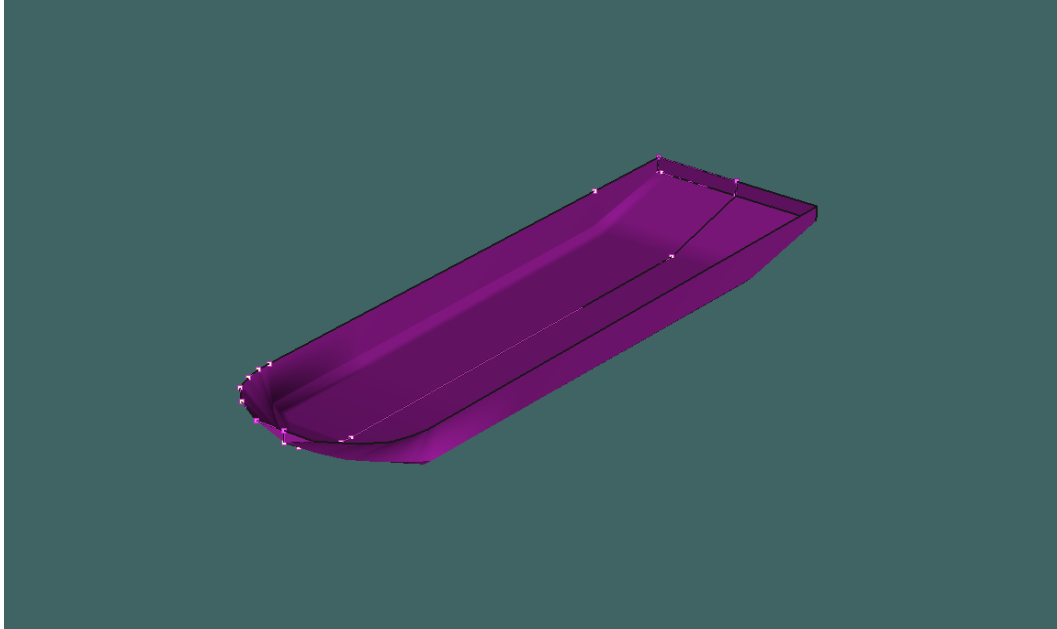


BAB 4. PEMBAHASAN

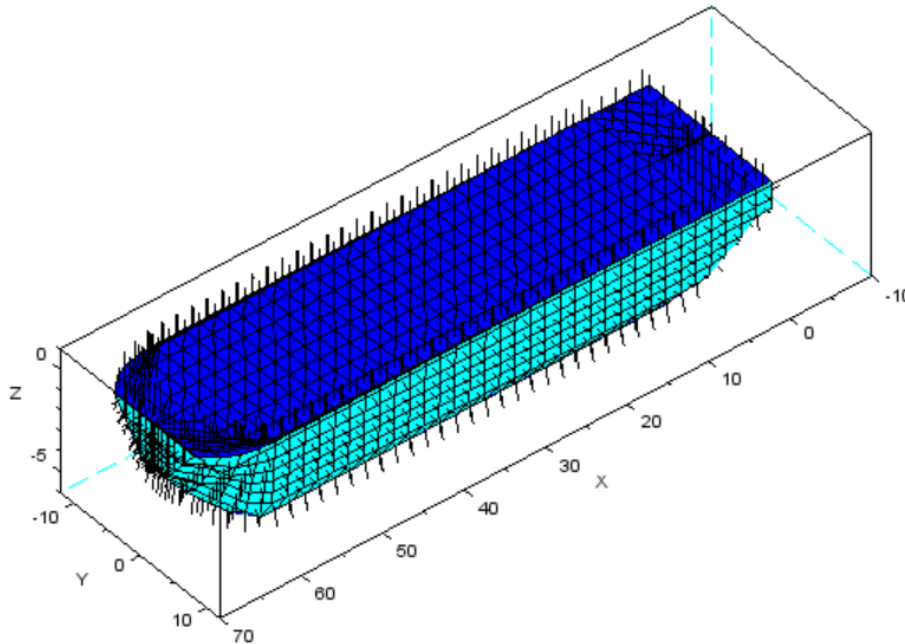
4.1 Pembuatan Model

4.1.1. Pemodelan Barge

Untuk membuat model 3 dimensi didalam Scilab diperlukan data offset yang berasal dari tabel offset aplikasi Maxsurf. Data offset tersebut memerlukan program dalam membuat panel-panel yang jika digabungkan akan membentuk lambung barge. Dalam membuat panel, digunakan kaidah tangan kanan saat menghubungkan setiap titik untuk memastikan gaya normal tidak terbalik. Pada Gambar 4.1 merupakan model dari aplikasi Maxsurf sedangkan Gambar 4.2 menunjukkan model pada aplikasi Scilab dengan garis gaya yang tegak terhadap setiap bidang panelnya. Dari hasil pemodelan barge didapatkan displacement barge pada draft 3.378 m adalah 3621.19 Ton dan Center of Buoyancy (CB) yaitu $X = 33.17$ m, $Y = 0$ m, $Z = -2.92$ m. Nilai displacement dan CB didapatkan dengan menggunakan fungsi program yang telah dibuat yaitu `bouyancy()`.



Gambar 4.1 Model Barge pada Maxsurf



Gambar 4.2 Model Barge pada Scilab

Pada Tabel 4.1 merupakan hasil perhitungan displacement dan CB dari barge dengan menggunakan fungsi program buoyancy(). Masukan dari fungsi program tersebut yaitu offset, panel, CB, displacement.

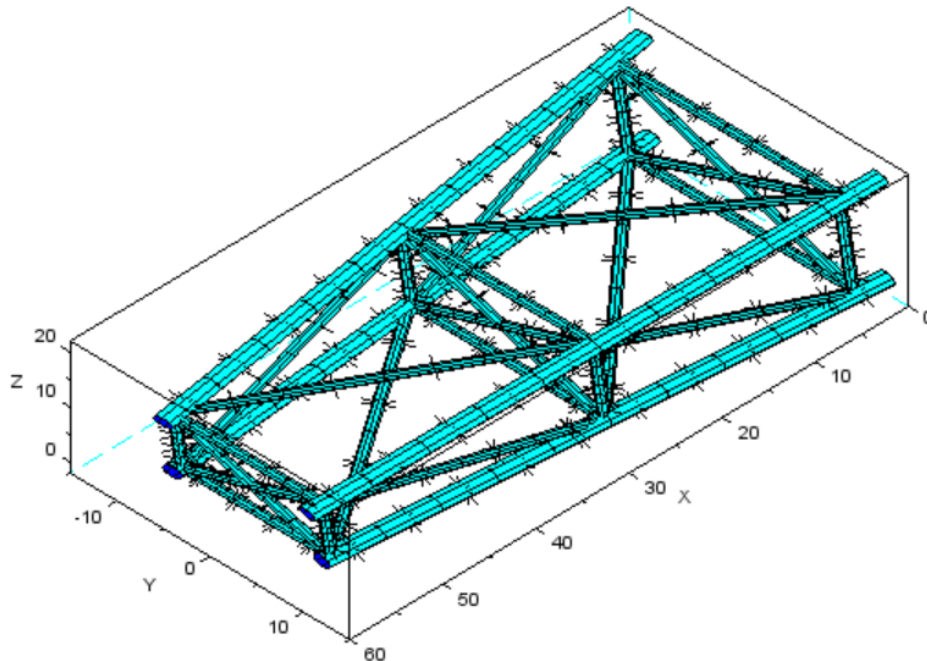
Tabel 4.1 Displacement Barge saat even keel di setiap draft

Draft (m)	Displacement (Ton)	CB (m) pada sumbu		
		X	Y	Z
1	868.34	34.94	0	-0.82
2	1961.68	33.72	0	-1.70
3	3155.15	32.91	0	-2.57
3.378	3621.19	33.17	0	-2.92
4	4394.93	33.63	0	-3.48

4.1.2 Pemodelan Jacket

Pemodelan jacket pada penelitian ini menggunakan Scilab. *Jacket* dibuat dengan struktur sederhana untuk mempermudah penelitian ini mendapatkan *Center of Gravity* (CG). Fungsi program dibuat untuk membentuk model dari jacket yaitu generateMeshJacket(). Didalam fungsi tersebut terdapat fungsi-fungsi lain yaitu meshPipeOffset(), geotransformation(), plotMesh(), dan arrangeData(). Hasil

pemodelan jacket didapatkan berat yaitu 255.03 Ton dan CoG yaitu $X = 27.55$ m, $Y = 0.09$ m dan $Z = 8.61$ m. Pada Gambar 4.3 merupakan model dari jacket yang telah dibuat.



Gambar 4.3 Model Jacket pada Scilab

4.2 Validasi Model

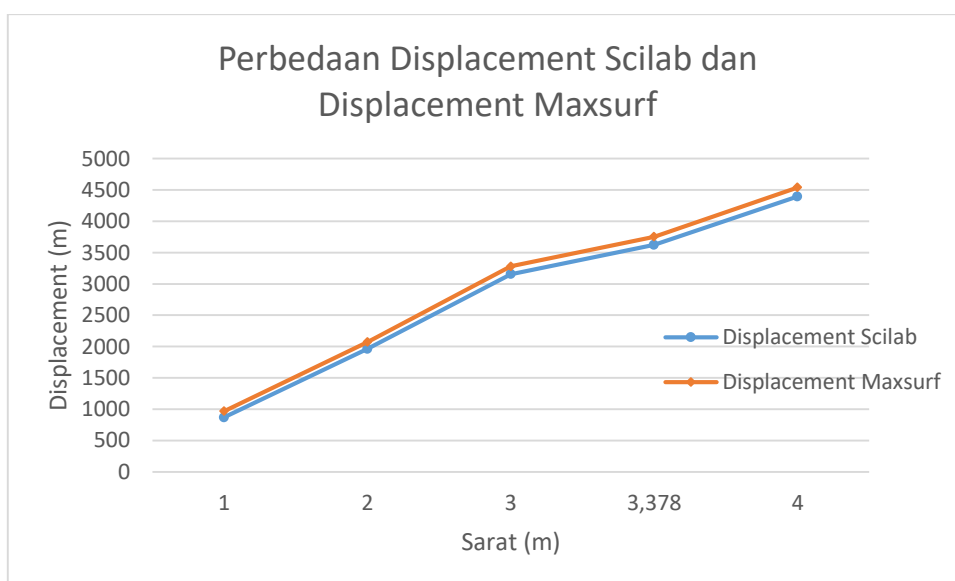
Validasi model dilakukan untuk mengetahui tingkat keakuratan dari sebuah geometri yang telah dibuat dalam hal ini adalah lambung barge. Metode yang dilakukan ialah membandingkan data displacement barge dan data posisi CB pada Maxsurf dan Scilab. Adapun rumus yang digunakan ialah.

$$\text{Perbedaan} = \frac{|data\ dari\ scilab - data\ dari\ maxsurf|}{data\ dari\ maxsurf} \times 100 \%$$

Tabel 4.2 Perbedaan displacement model pada Scilab dan Maxsurf

Draft (m)	Displacement (Ton) Scilab	Displacement (Ton) Maxsurf	Perbedaan (%)
1	868.34	966.5	10.15
2	1961.68	2072	5.32
3	3155.15	3280	3.80
3.378	3621,19	3753	3.51
4	4394.93	4540	3.19

Dari Tabel 4.2 dapat dilihat perbedaan hasil kurang dari 10% untuk draft lebih dari 1 m. Perbedaan hasil tersebut disebabkan oleh perbedaan metode perhitungan, yang mana scilab yang perhitungannya menggunakan Persamaan 2 dan perhitungan maxsurf menggunakan Simpson. Dari semua hasil displacement dapat dilihat perhitungan dari fungsi program scilab selalu lebih kecil dari maxsurf. Hal ini dapat terjadi karena meshing yang tidak cukup kecil. Gambar 4.4 merupakan grafik perbedaan hasil dari Scilab dan Maxsurf, terlihat bahwa semakin besar sarat maka perbedaan hasil semakin kecil. Hal ini dikarenakan pada draft yang kecil, panel yang terendam air juga akan sedikit, sehingga panel yang menghitung displacement menjadi sedikit yang membuat perbedaan hasil menjadi besar.

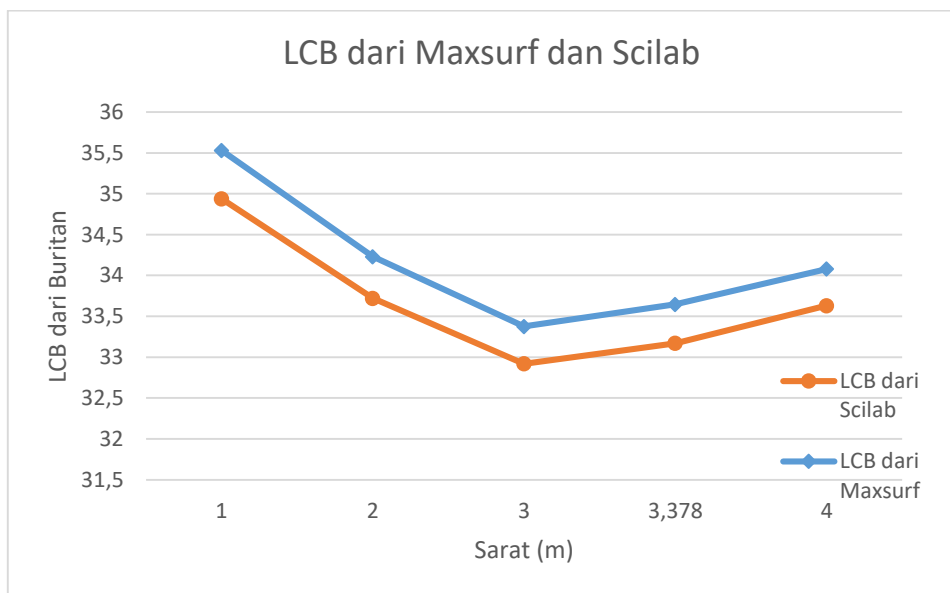


Gambar 4.4 Grafik Perbedaan Hasil Displacement

Tabel 4.3 Perbedaan LCB model pada Scilab dan Maxsurf

Draft (m)	LCB dari Scilab (m)	LCB dari Maxsurf (m)	Error (%)
1	34.94	35.53	1.66
2	33.72	34.23	1.48
3	32.92	33.37	1.36
3.378	33.17	33.647	1.41
4	33.63	34.08	1.32

Pada model yang dibuat, LCB diukur dari ujung buritan. Pada Tabel 4.3 terdapat perbedaan 1.32% sampai 1.66% dan pada Gambar 4.5 terlihat kurva dari kedua LCB memiliki bentuk yang sama, hal ini dikarenakan bentuk model yang sama. Meskipun memiliki bentuk kurva yang sama, nilai LCB Maxsurf dan Scilab tetap berbeda yang mana nilai LCB Scilab lebih dekat dengan buritan dibandingkan dengan nilai LCB Maxsurf dan perbedaan inilah disebut error. Error tersebut terjadi dikarenakan bentuk haluan yang lebih melengkung dibandingkan buritan untuk membentuk panel sehingga perhitungan pada panel kurang maksimal.

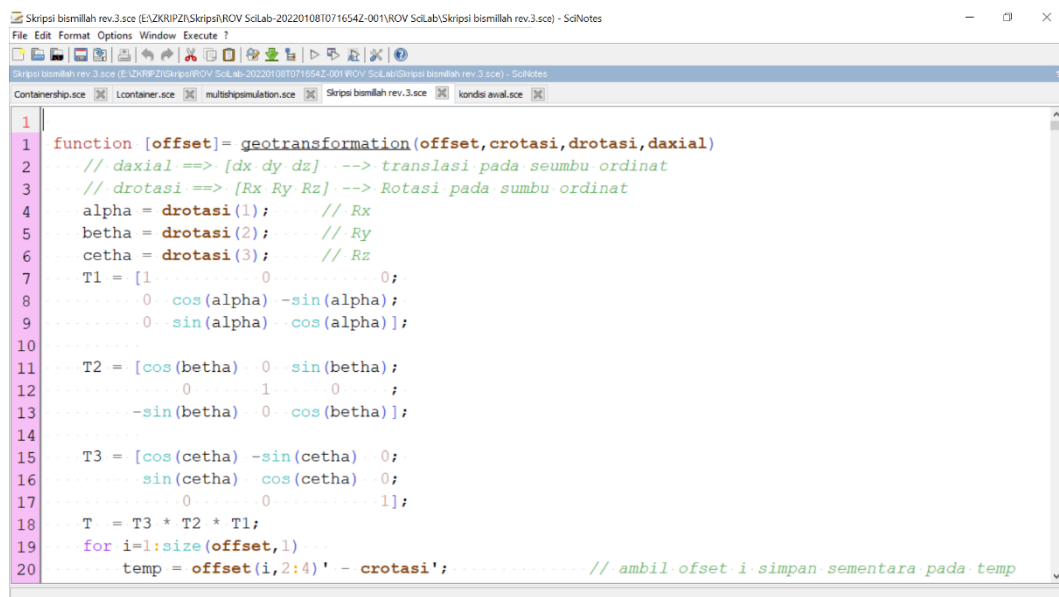


Gambar 4.5 Grafik Error LCB Model

4. 3 Fungsi Program dan GUI

4. 3. 1. Fungsi program

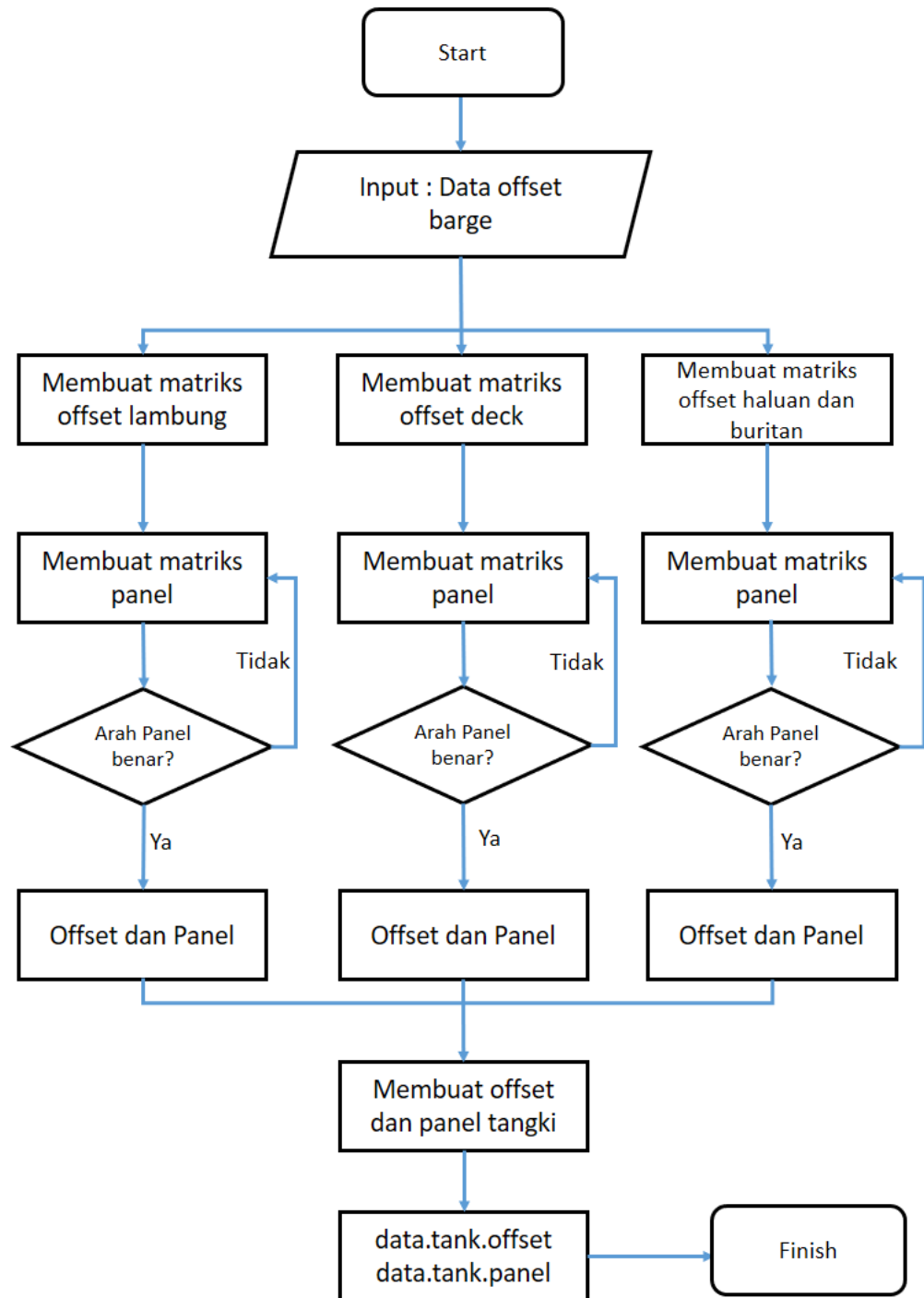
Program yang dibuat pada penelitian ini menggunakan Scilab. Scilab digunakan karena merupakan *software* komputasi *opensource* sehingga gratis untuk digunakan. *Software opensource* adalah jenis perangkat lunak yang kode sumbernya terbuka untuk dipelajari, diubah, ditingkatkan dan disebarluaskan. Untuk membuat suatu fungsi program, dilakukan pada Scinote seperti pada Gambar 4.6 . Fitur tersebut digunakan untuk menjalankan beberapa perintah sesuai dengan algoritma yang diinginkan tanpa menampilkannya di console jika tidak diperlukan. Didalam fungsi program terdapat fungsi program lain yang telah dibuat untuk mendapatkan hasil yang diinginkan. Adapun beberapa fungsi program yang dibuat pada penelitian ini yaitu `genof()`, `geotransformation()`, `plotMesh()`, `arrangeData()`, `generateMeshJacket()`, `buoyancy()`, `interpolasi()`, `plotLevel()`, `meshPipeOffset()`, `run()`, dan `postLaunch()`.



```
1
2 function [offset]= geotransformation(offset,crotasi,drotasi,daxial)
3 // daxial ==> [dx dy dz] --> translasi pada sumbu ordinat
4 // drotasi ==> [Rx Ry Rz] --> Rotasi pada sumbu ordinat
5 alpha = drotasi(1); // Rx
6 betha = drotasi(2); // Ry
7 cetha = drotasi(3); // Rz
8 T1 = [1 0 0;
9 0 cos(alpha) -sin(alpha);
10 0 sin(alpha) cos(alpha)];
11 T2 = [cos(betha) 0 sin(betha);
12 0 1 0;
13 -sin(betha) 0 cos(betha)];
14 T3 = [cos(cetha) -sin(cetha) 0;
15 sin(cetha) cos(cetha) 0;
16 0 0 1];
17 T = T3 * T2 * T1;
18 for i=1:size(offset,1)
19 temp = offset(i,2:4)' - crotasi'; //ambil offset i simpan sementara pada temp
```

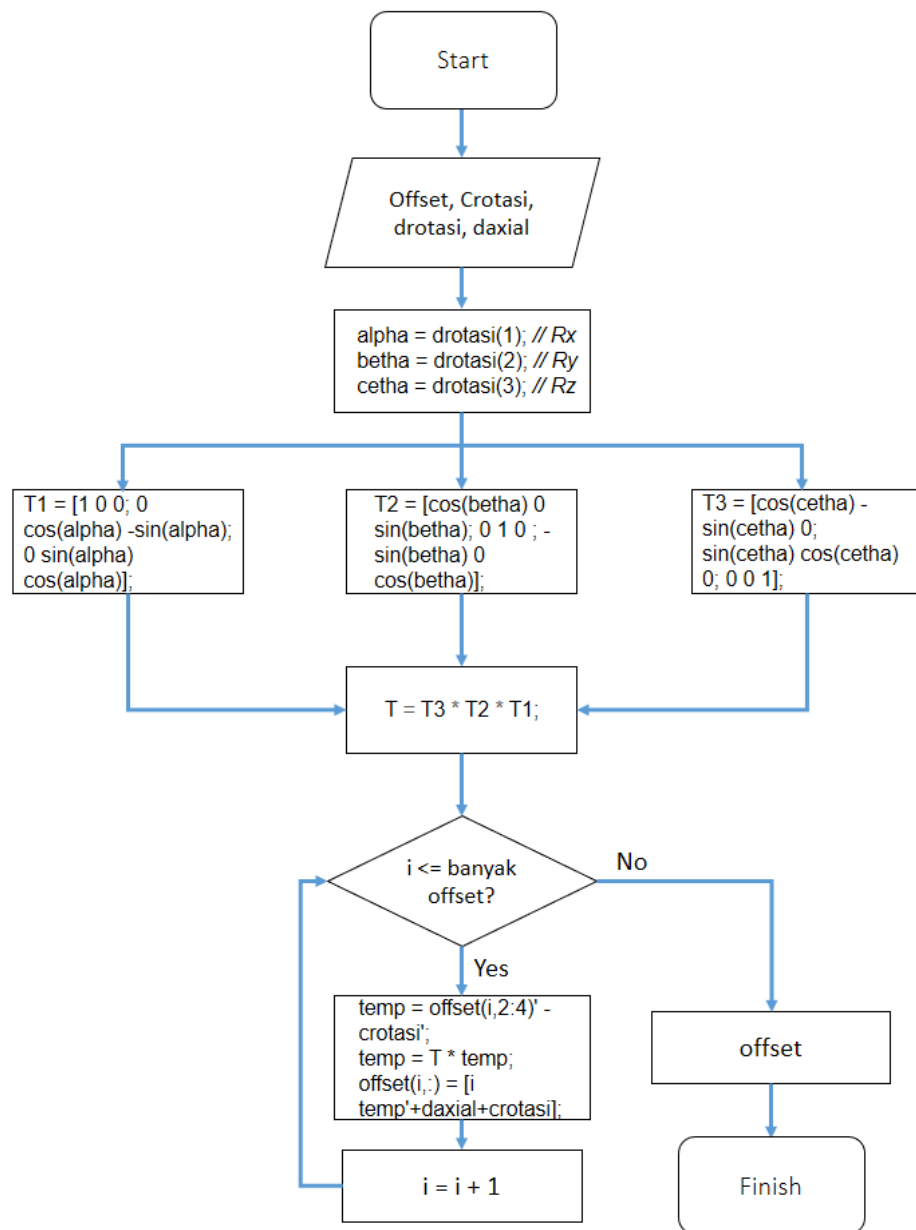
Gambar 4.6 Tampilan Scinote

- `genof()` digunakan untuk membuat offset dan panel dari barge. Masukan dari fungsi program tersebut adalah data offset dari maxsurf. Gambar 4.6 merupakan flowchart dari `genof()`



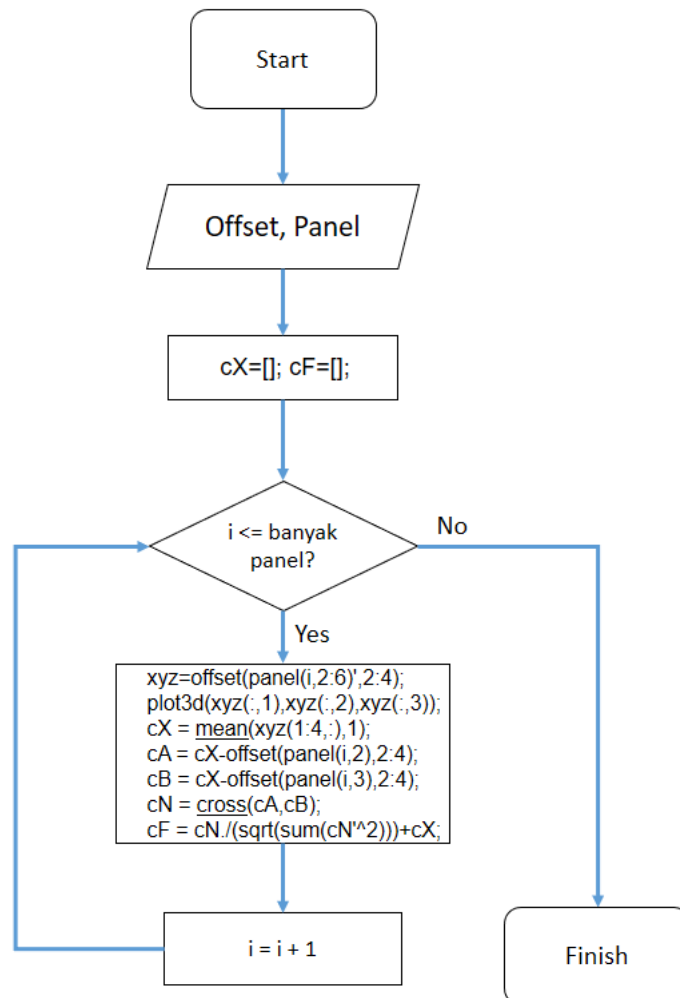
Gambar 4.7 Flowchart `genof()`

- `geotransformation()` digunakan untuk merotasi atau memindahkan objek. Masukan dari fungsi program tersebut adalah crotasi untuk titik pusat rotasi, drotasi untuk besaran rotasi dalam radian, dan daxial untuk besaran perpindahan.



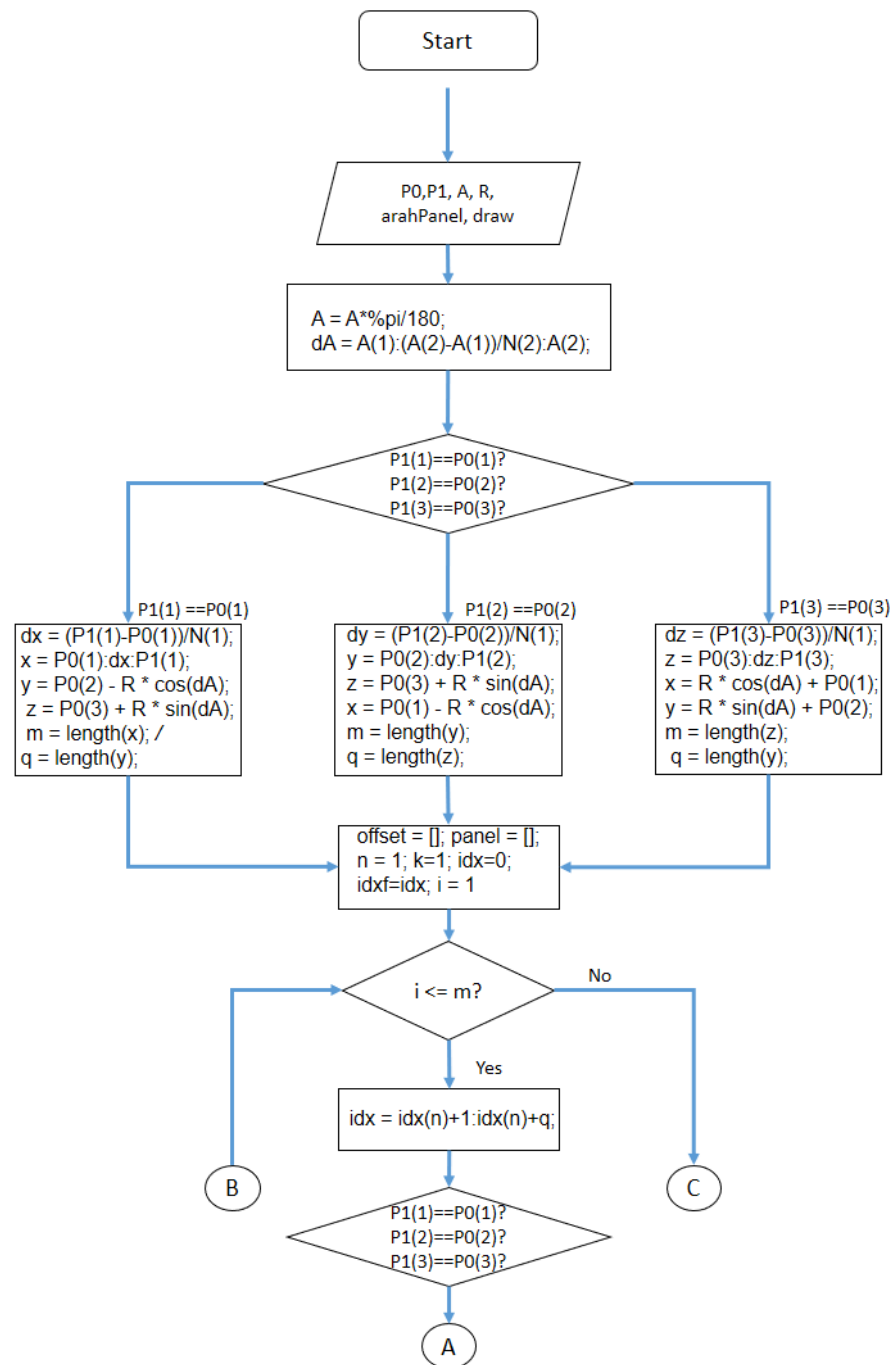
Gambar 4.8 Flowchart `geotransformation()`

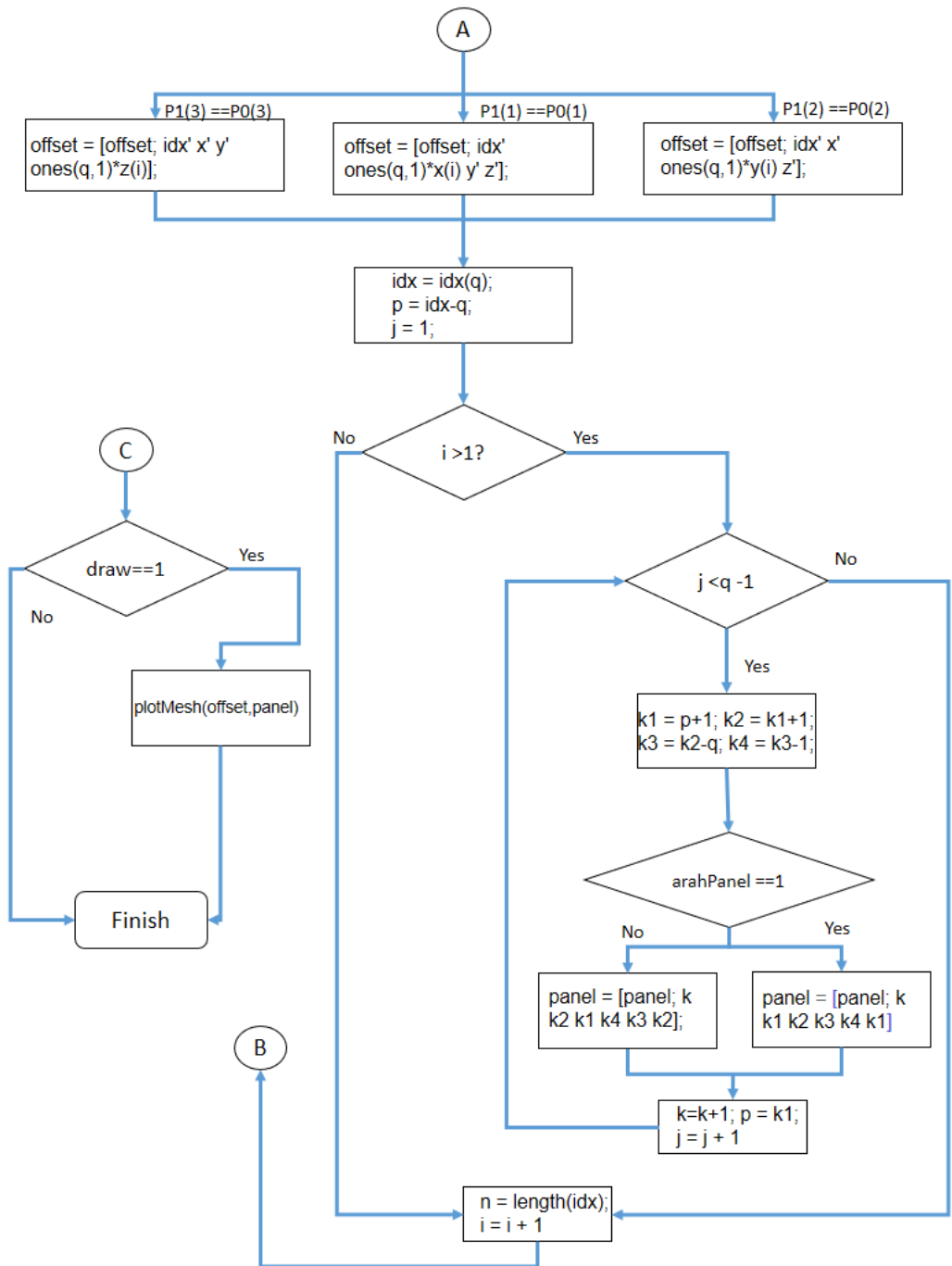
- `plotMesh()` digunakan untuk membuat visual 3 dimensi dari model. Masukan dari fungsi program tersebut adalah data offset dan data panel yang sebelumnya telah diperoleh dari hasil pemodelan menggunakan fungsi program `genof()`.



Gambar 4.9 Flowchart `plotMesh()`

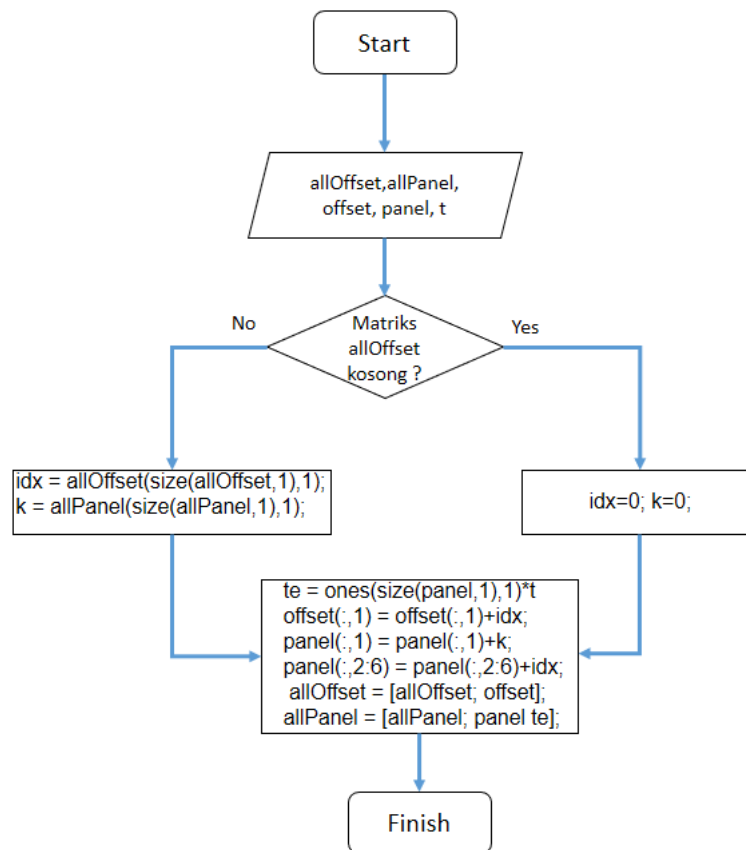
- `meshPipeOffset()` digunakan untuk membuat offset dari *leg jacket*. Masukan dari fungsi program ini yaitu P0 untuk titik awal, P1 titik akhir, A untuk range lingkaran yang akan dibuat, R untuk jari-jari dari leg yang akan dibuat, N digunakan untuk pemotongan *leg jacket* secara memanjang dan melintang.





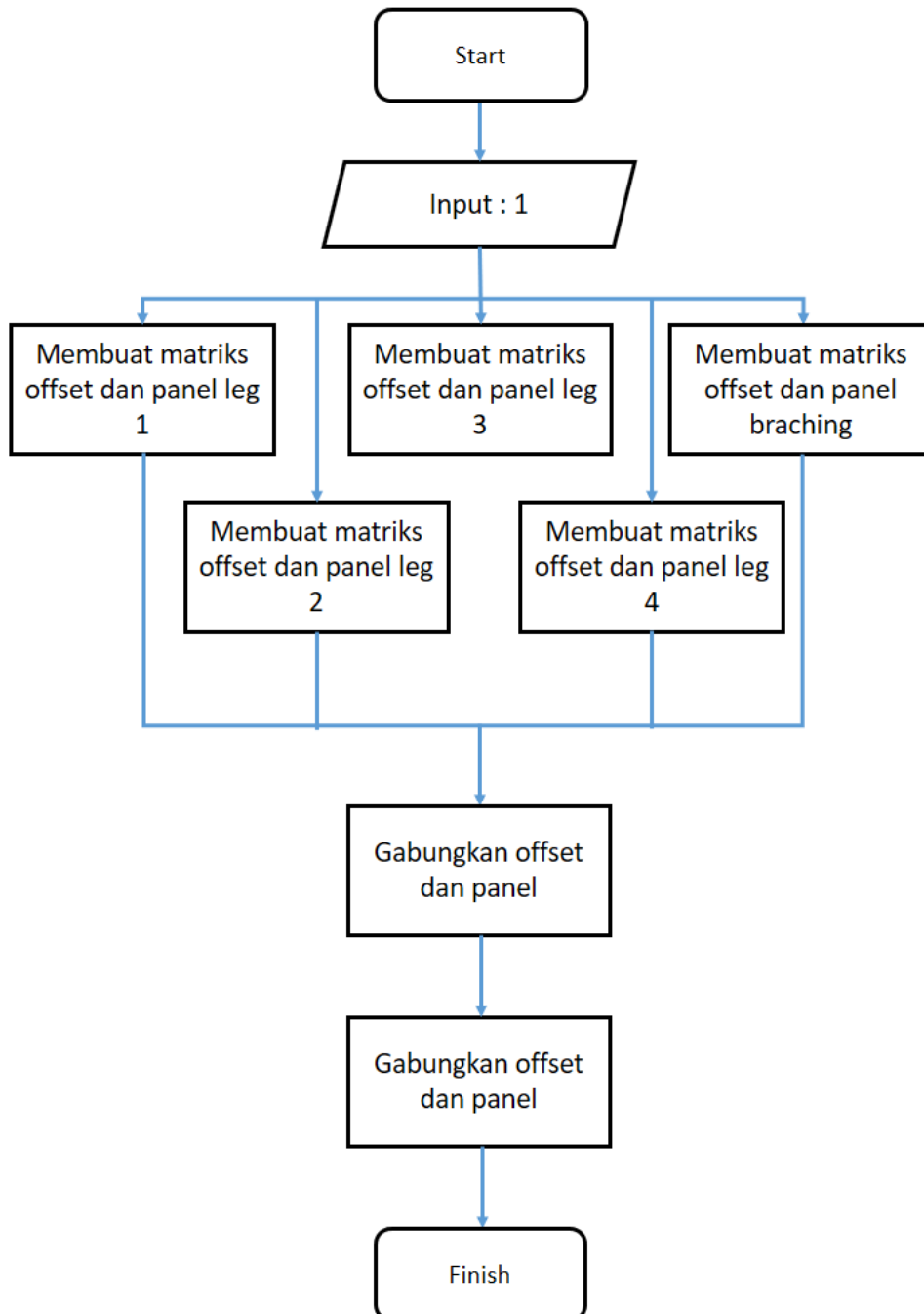
Gambar 4.10 *Flowchart MeshPipeOffset()*

- `arrangeData()` digunakan untuk menggabungkan beberapa data offset dan data panel menjadi masing-masing satu matriks yang disebut `allOffset` dan `allPanel`. Masukan dari fungsi program tersebut adalah `allOffset` untuk matriks gabungan offset, `allPanel` untuk matriks gabungan panel, `offset` untuk matriks offset yang akan digabungkan, `panel` untuk matriks panel yang akan digabungkan, `t` untuk ketebalan plat.



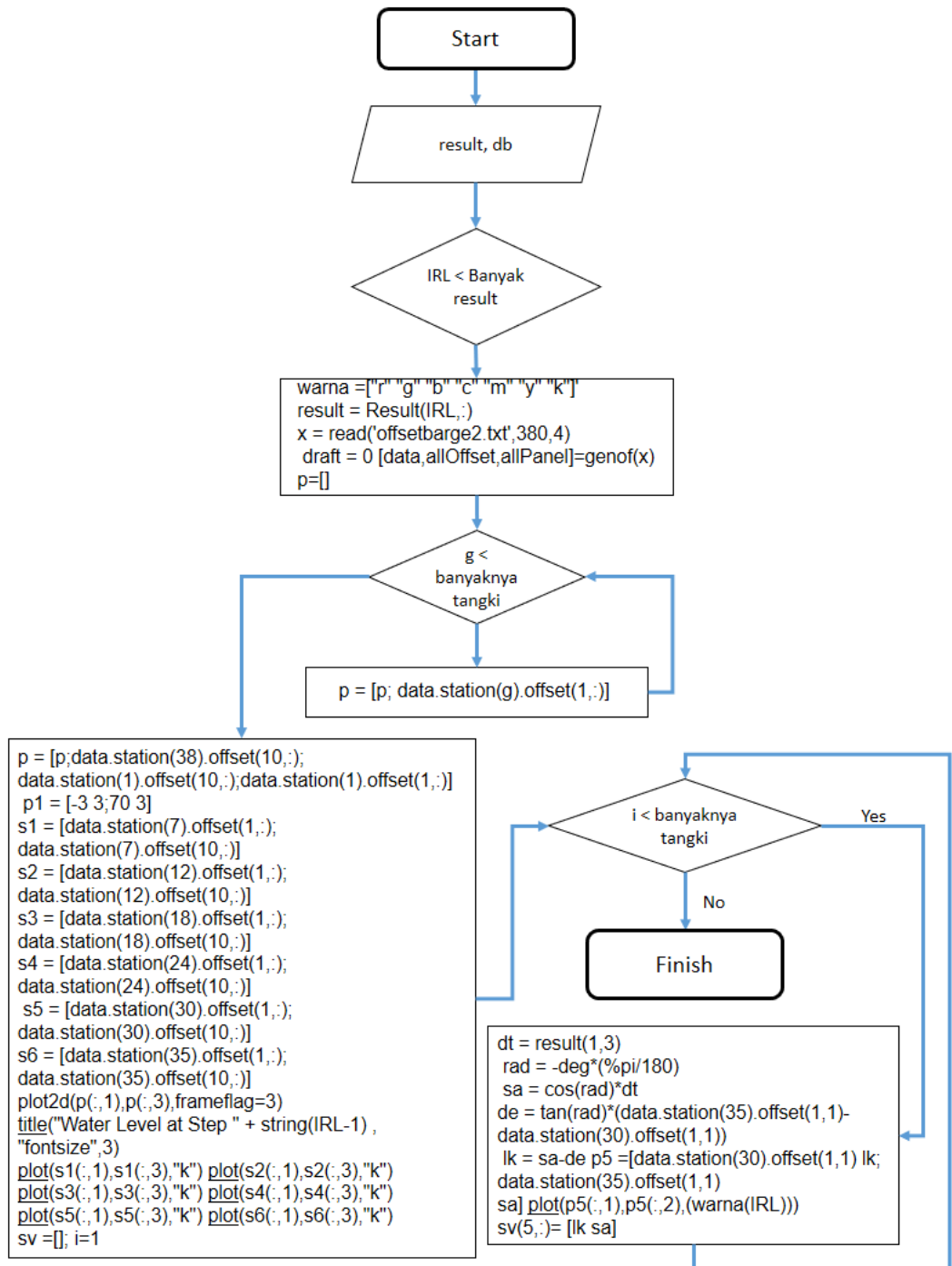
Gambar 4.11 *Flowchart* `arrangeData()`

- `generateMeshJacket()` digunakan untuk membuat offset dari jacket. Fungsi program tersebut terdapat gabungan-gabungan dari fungsi program seperti `meshPipeOffset()`, `geotransformation()`, `arrangeData()`.



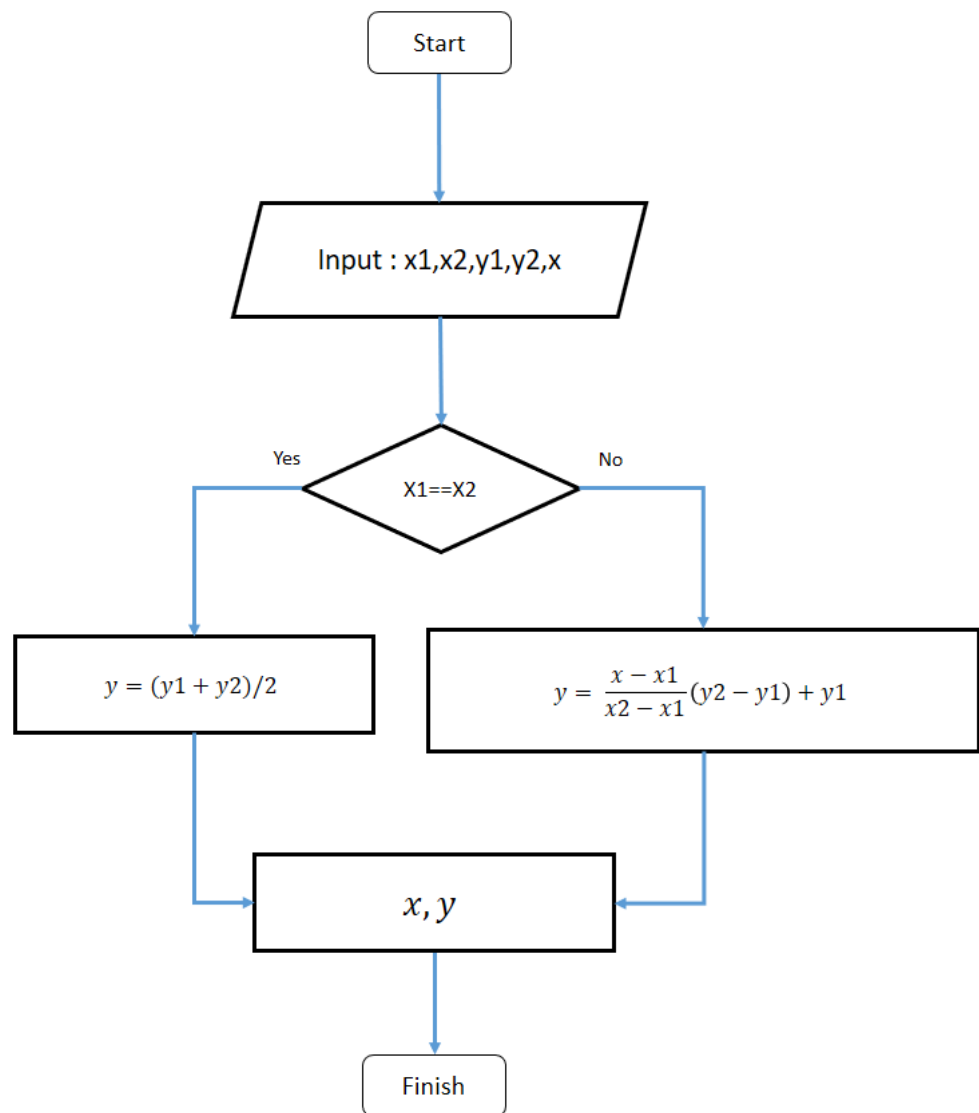
Gambar 4.12 *Flowchart* generateMeshJacket()

- `plotLevel()` digunakan untuk memvisualisasikan level air secara 2 dimensi dari hasil yang didapatkan.



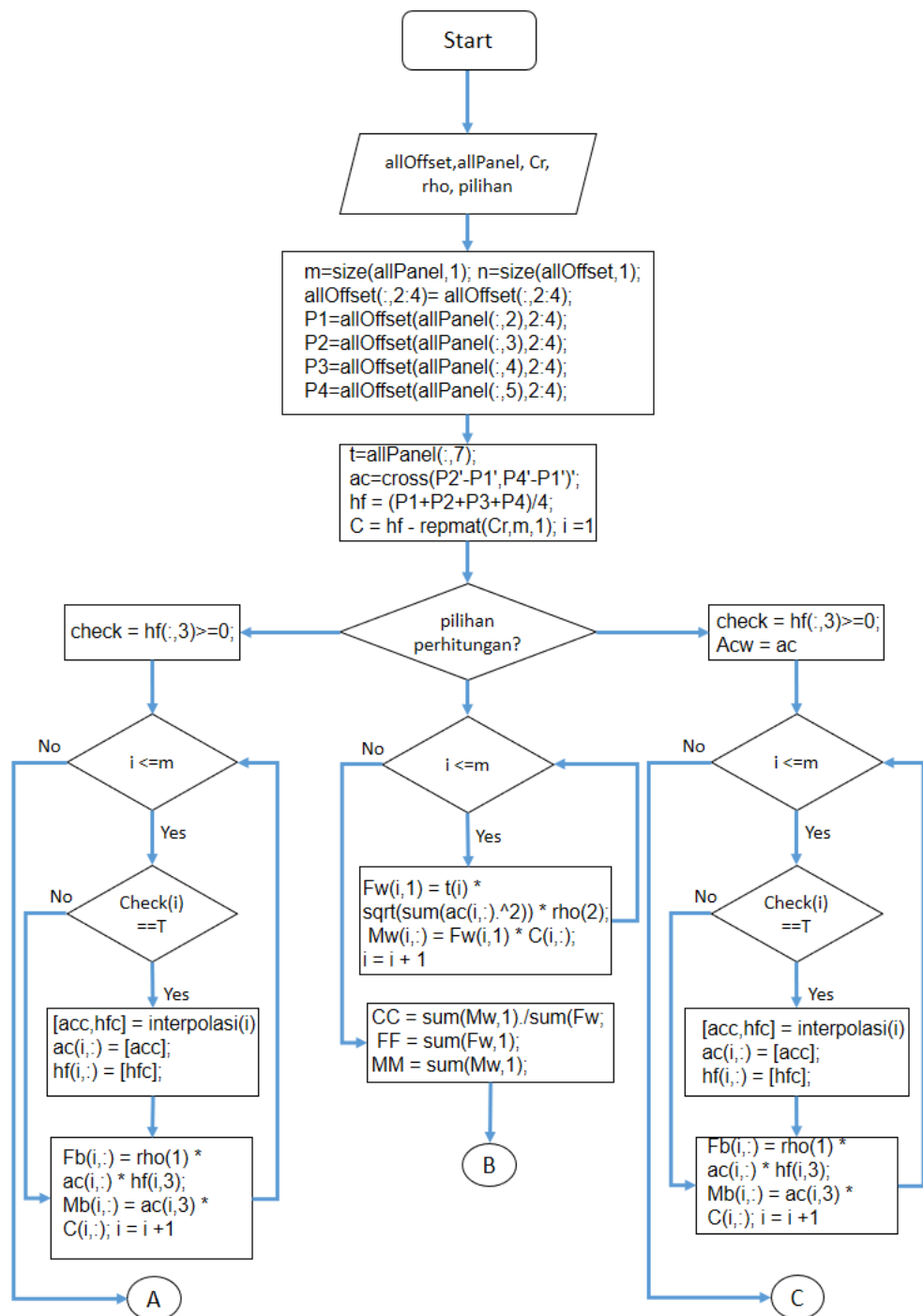
Gambar 4.13 Flowchart `plotLevel()`

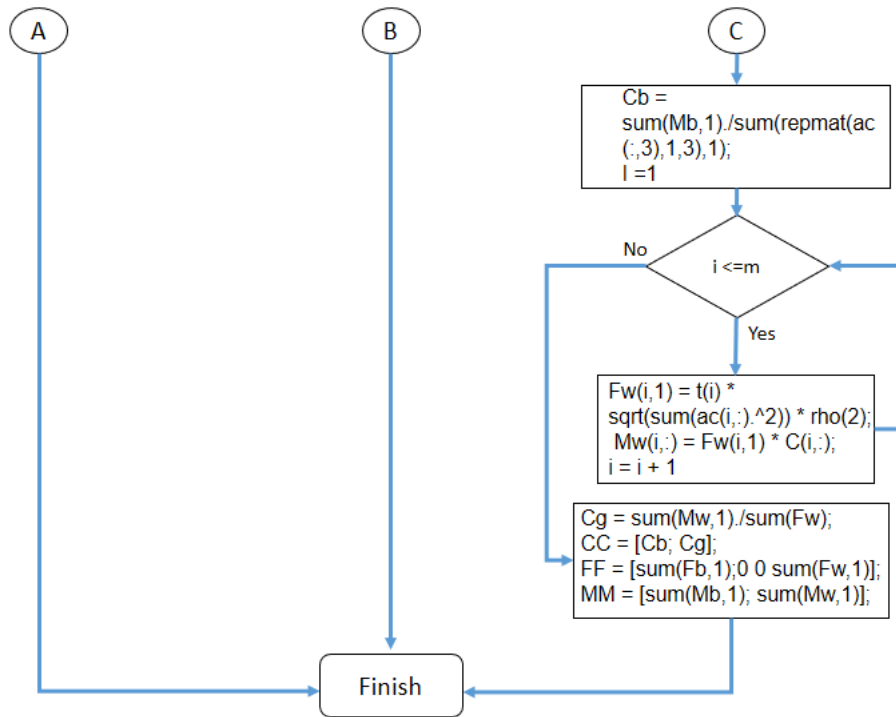
- interpolasi() digunakan untuk mencari nilai yang berada diantara dua nilai yang sudah diketahui nilainya. Fungsi program ini digunakan untuk mengurangi eror pada hasil yang didapatkan.



Gambar 4.14 *Flowchart* interpolasi()

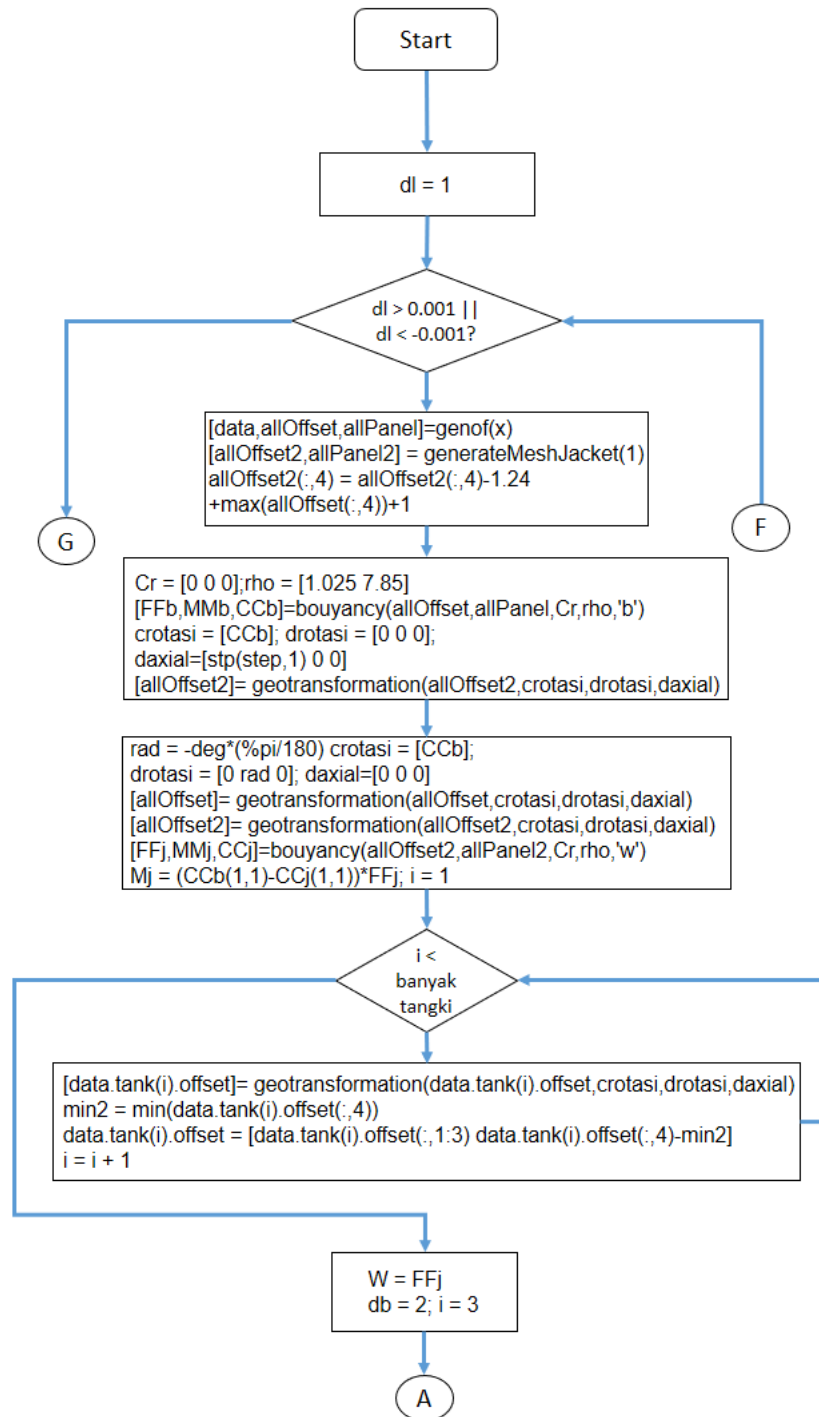
- buoyancy() digunakan untuk mencari displacement, CG, CB, dan juga berat. Masukan dari fungsi program ini yaitu allOffset, allPanel, Cr untuk titik referensi, rho yang terdiri dari massa jenis air laut (1,025 ton/m³) dan baja (8,75 ton/m³)

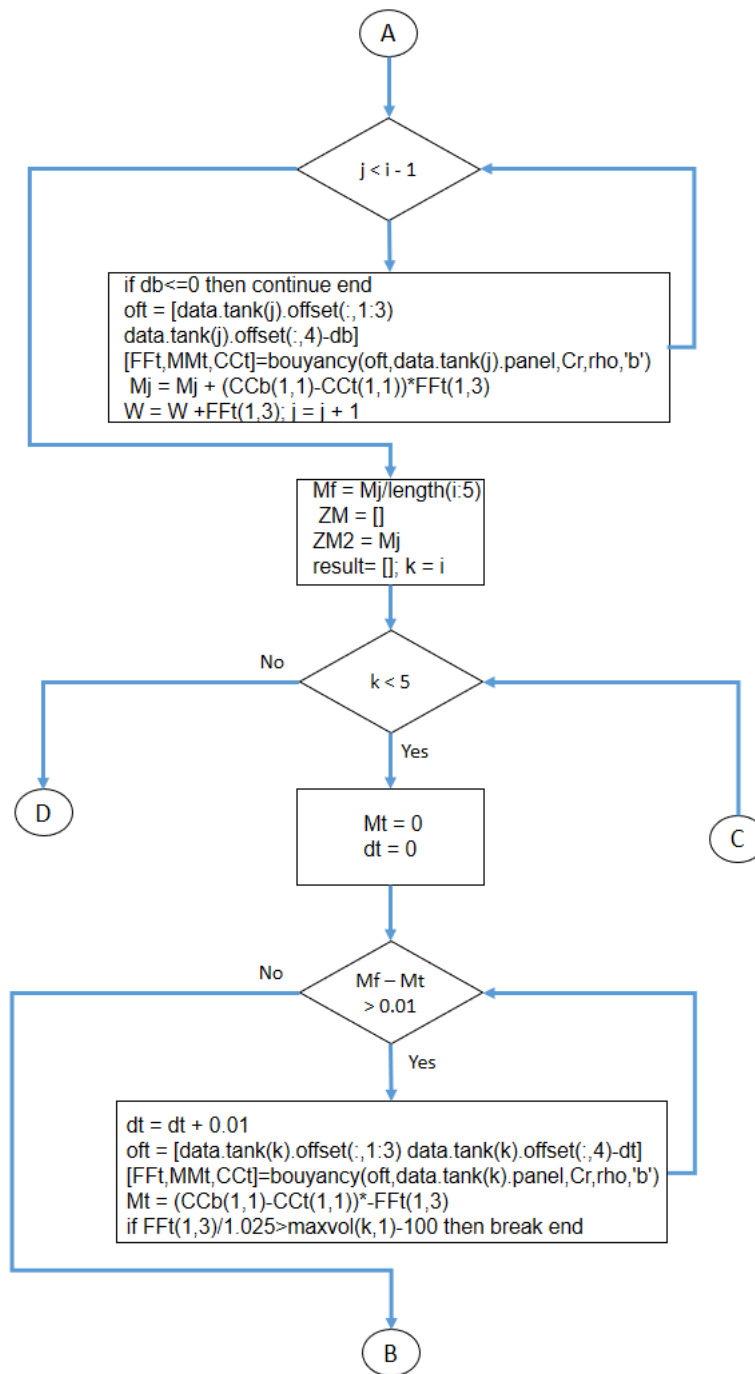


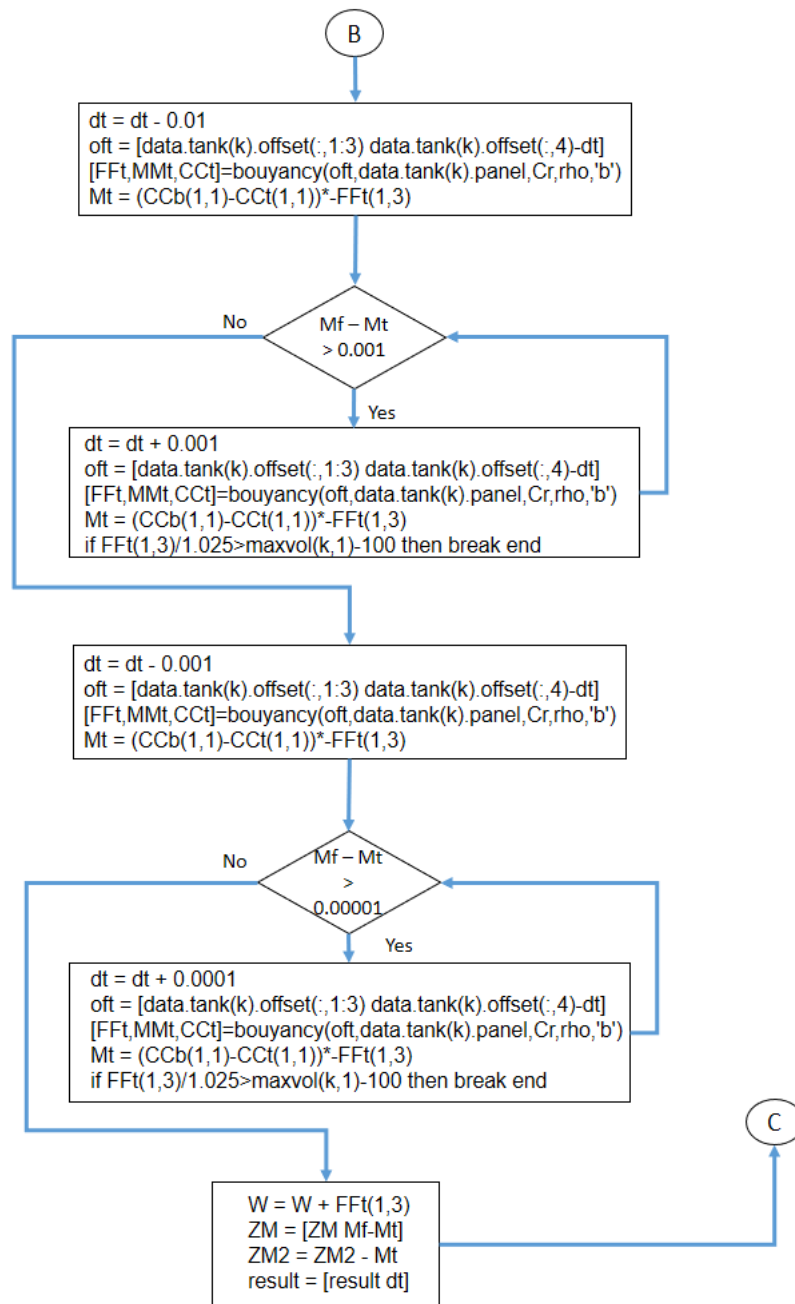


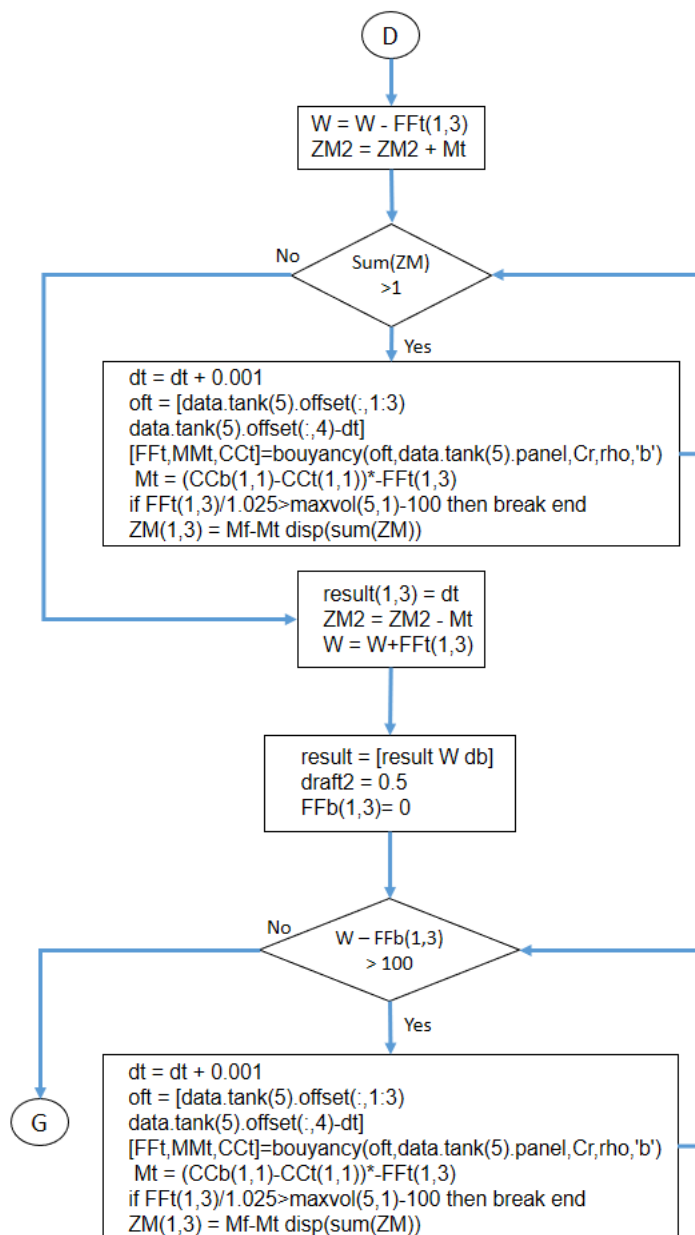
Gambar 4.15 *Flowchart* buoyancy()

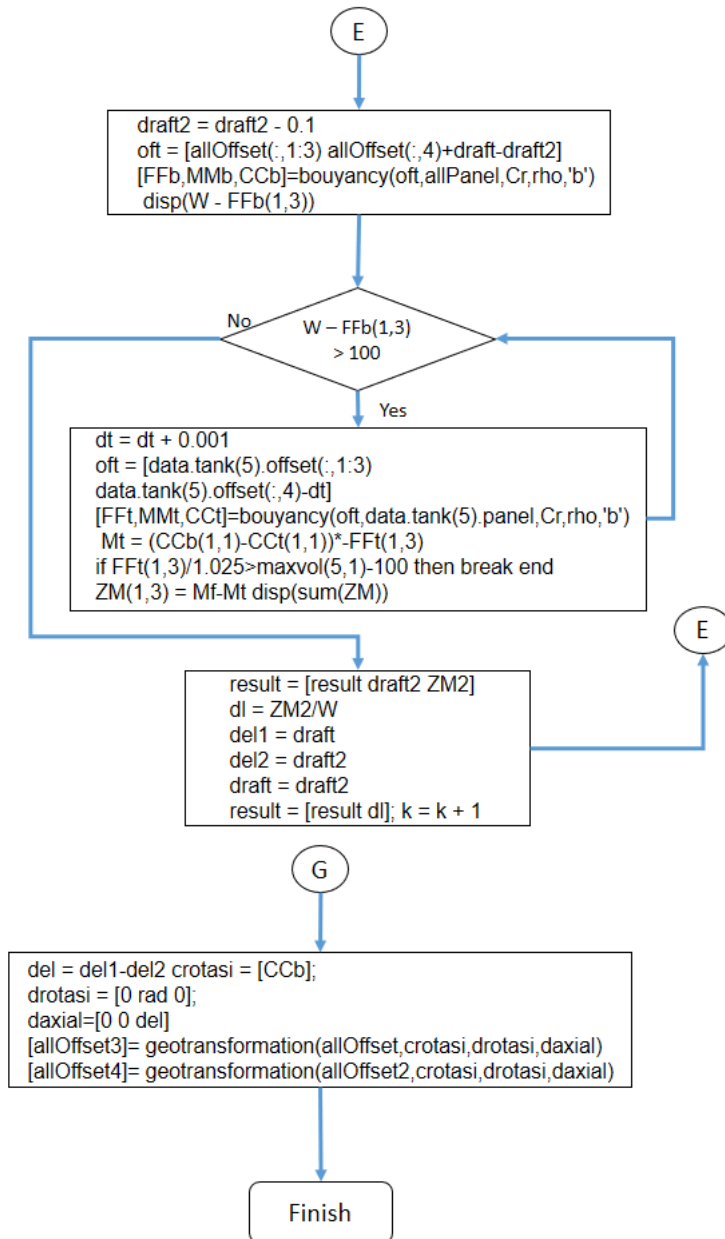
- run() digunakan untuk menjalankan simulasi sehingga mendapatkan volume tangki, level tangki, dan pergeseran titik CG dengan derajat trim yang diinginkan.





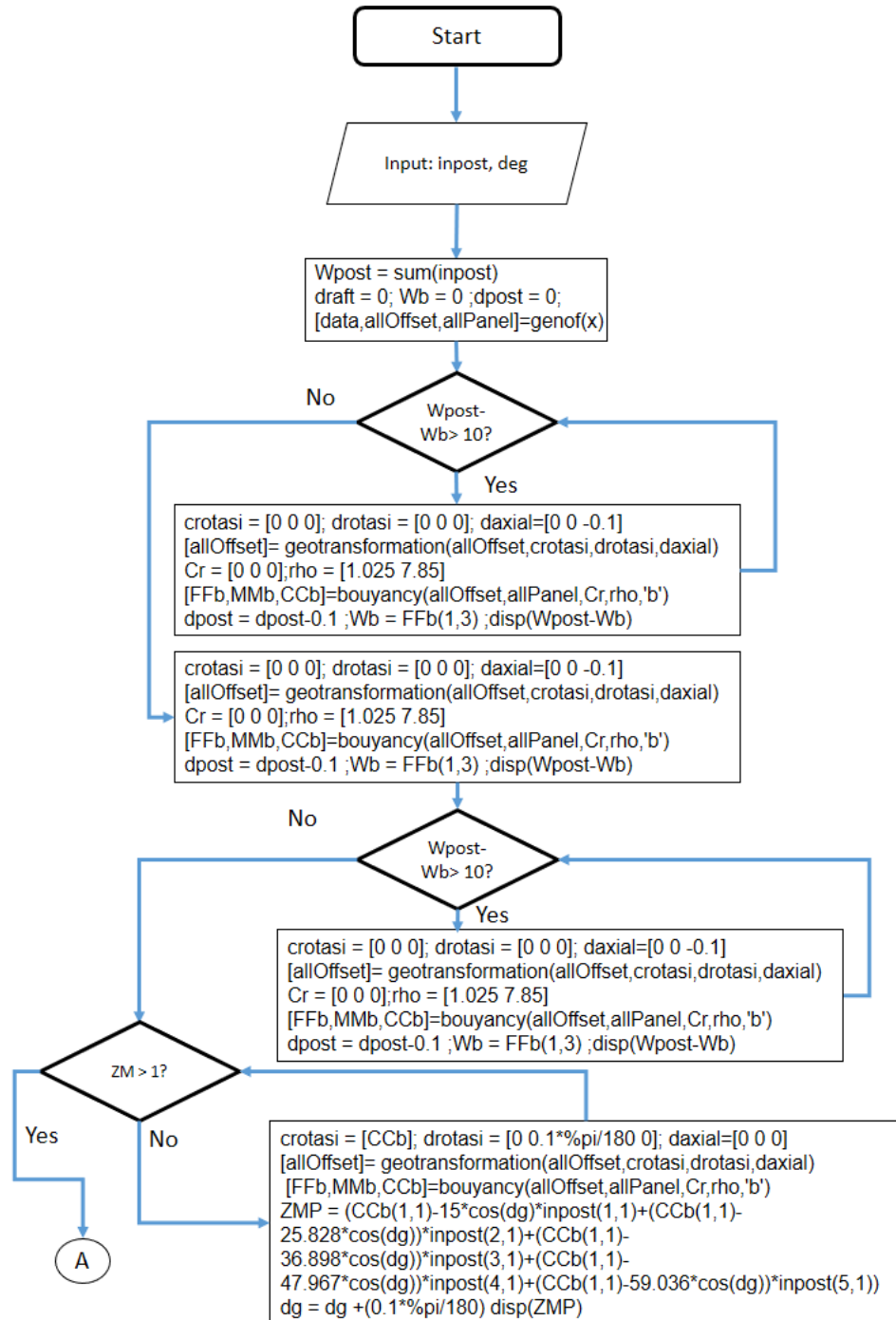


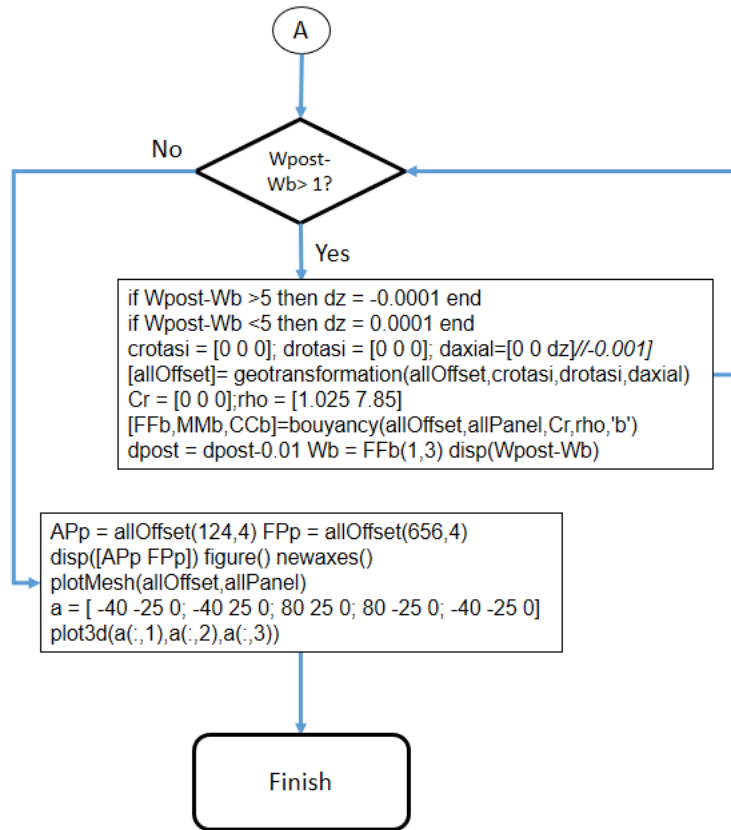




Gambar 4.16 Flowchart run()

- `postLaunch()` digunakan untuk mencari derajat trim haluan barge setelah *jacket launching*. Input program dari fungsi program tersebut ialah `inpost` yakni matriks berat air ballast dari setiap tangki.

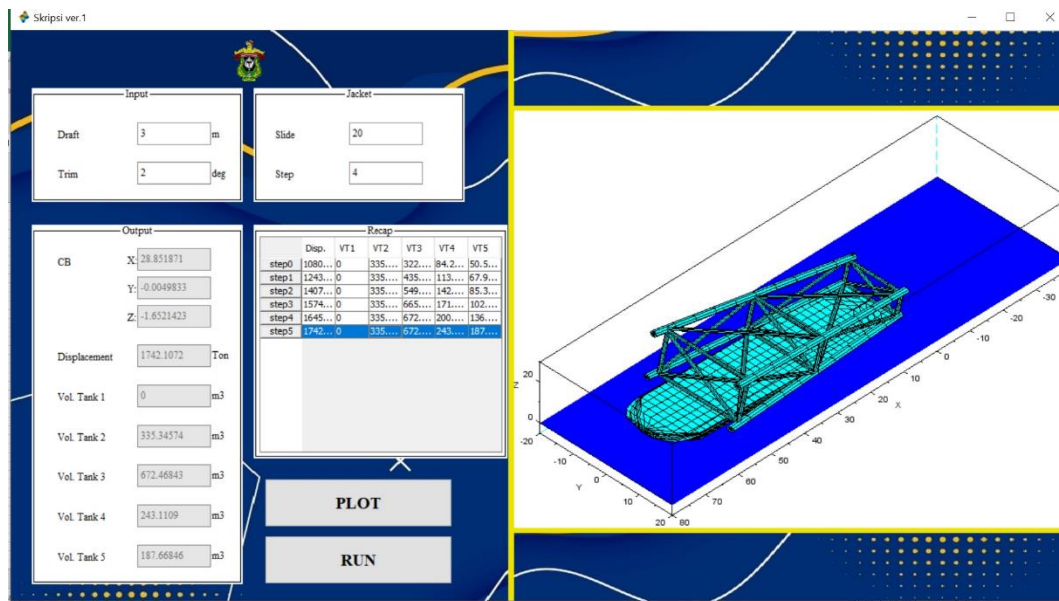




Gambar 4.17 Flowchart postLaunch()

4. 2. Desain GUI

Pada penelitian ini menggunakan scilab untuk merancang dan membuat *graphical user interface* (GUI) yang dibuat untuk memudahkan pengguna mengoperasikan program atau software yang telah dibuat. Pembuatan GUI menggunakan bantuan *toolbox* guibuilder. Namun terdapat keterbatasan pada *toolbox* tersebut sehingga diperlukan pembuatan secara manual melalui scinote.



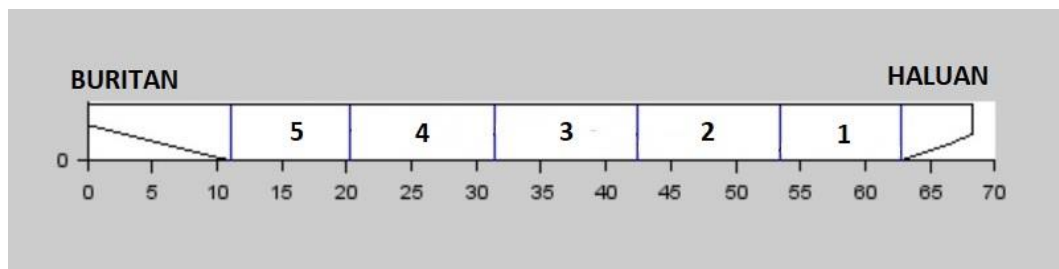
Gambar 4.18 Tampilan GUI

Pada Gambar 4.18 merupakan *user interface penelitian ini*. Yang mana terdapat frame input dan jacket untuk memasukkan variabel. Frame output digunakan untuk menampilkan hasil perhitungan dan frame recap digunakan untuk menampilkan tabel hasil perhitungan sebelumnya. Tombol plot digunakan untuk melakukan plot pada axes dan menghitung displacement dan CB berdasarkan draft dan derajat trim yang dimasukkan. Tombol run digunakan untuk menghitung volume tangki setiap step jarak slide dari jacket. Setelah setiap step telah dihitung maka akan menampilkan grafik volume tangki dan visual 2 dimensi tampak samping yang menampilkan level air. Program untuk desain GUI dapat dilihat pada lampiran.

4.4 Hasil Perhitungan Volume & Level Tangki

Pada penelitian ini menggunakan barge yang memiliki 5 tangki secara memanjang barge yang dibagi oleh sekat melintang. Tangki-tangki seperti pada Gambar 4.19 diberikan nomor secara berurutan dari haluan hingga buritan. Pada tangki-tangki tersebut akan diisi air ballast untuk melakukan peluncuran jacket dengan mempertimbangkan beban secara merata. Pada tangki 1, 2, 3, dan 4 akan

diisi air ballast yang mana pada tangki 1, 2, dan 3 akan mengalami penambahan air ballast. Hal tersebut dikarenakan tangki 1, 2, dan 3 berada didepan titik CG yang berguna untuk menyeimbangkan momen dari jacket. Pada tangki 4 tidak terjadi penambahan air ballast dikarenakan berada dibelakang titik CG yang dapat menambah momen kearah buritan dan tangki 5 dikosongkan dikarenakan beban dari jacket.



Gambar 4.19 Tangki-tangki pada Barge

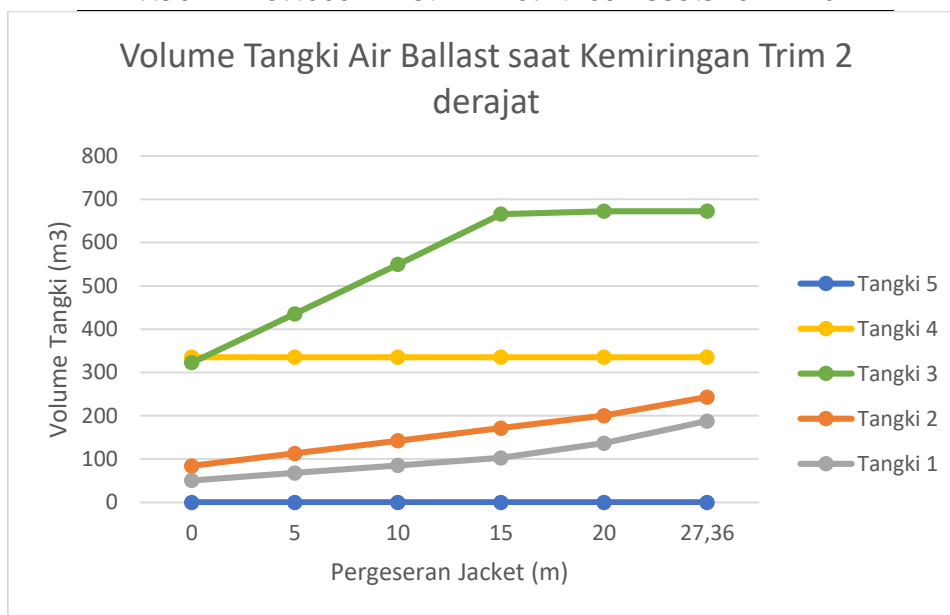
4.4.1 Hasil Perhitungan Volume Tangki

Penelitian ini dilakukan dengan memvariasikan derajat trim buritan kemiringan antara 2° hingga 4° dengan pengambilan disetiap penambahan 0.5° . Dari hasil perhitungan didapatkan volume dan level air ballast untuk setiap tangki. Pada setiap perhitungan dilakukan, jarak pergeseran jacket dari tempat awal hingga 20 meter dengan setiap penambahan 5 meter dihitung volume dan level tangki serta pada titik kritis jacket yang mana jacket sesaat akan melakukan gerak rotasi. Ketika jacket diluncurkan Kondisi kemiringan trim dipertahankan dengan cara menambahkan air ballast pada setiap tangki.

Penambahan air ballast dilakukan pada tangki- tangki yang berada didepan titik CG dari barge yang bertujuan untuk menyeimbangkan momen yang berasal dari jacket dengan mempertahankan kemiringan trim yang diinginkan. Momen trim yang ditimbulkan pada setiap tangki adalah sama sehingga setiap tangki memiliki volume yang berbeda. Hal ini juga bertujuan untuk menurunkan titik CG secara vertikal dan tidak terlalu memberatkan struktur barge .Berikut merupakan hasil perhitungan dari simulasi.

Tabel 4.4 Hasil Perhitungan Volume Tangki pada Kemiringan Trim 2°

Pergeseran Jacket (m)	Vol. tangki 1 (m ³)	Vol. tangki 2 (m ³)	Vol. tangki 3 (m ³)	Vol. tangki 4 (m ³)	Vol. tangki 5 (m ³)
0	50.534	84.2236	322.601	335.346	0
5	67.934	113.198	435.594	335.346	0
10	85.315	142.213	549.648	335.346	0
15	102.711	171.325	665.708	335.346	0
20	136.545	200.366	672.468	335.346	0
27.36	187.668	243.111	672.468	335.346	0

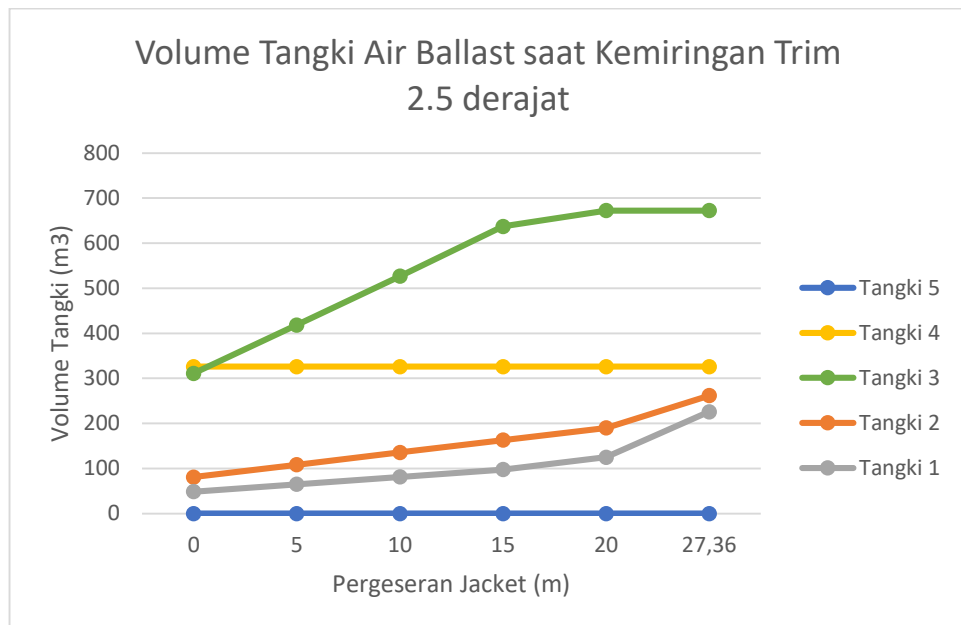


Gambar 4.20 Grafik hubungan Volume tangki dan pergeseran jacket pada Trim 2°

Pada Gambar 4.20 merupakan grafik dari volume tiap tangki dengan pergeseran jacket dengan sudut kemiringan trim 2°. Dapat terlihat pada pergeseran jacket 15 meter di tangki 3 terjadi penurunan penambahan volume tangki dikarenakan volume tangki 3 mendekati batas maksimal volume sehingga volume tangki 1 dan tangki 2 ditambahkan dengan lebih banyak.

Tabel. 4.5 Hasil Perhitungan Volume Tangki pada Kemiringan Trim 2.5°

Pergeseran Jacket (m)	Vol. tangki 1 (m ³)	Vol. tangki 2 (m ³)	Vol. tangki 3 (m ³)	Vol. tangki 4 (m ³)	Vol. tangki 5 (m ³)
0	48.5625	80.9493	311.07	325.814	0
5	64.9074	108.177	418.217	325.814	0
10	81.2428	135.442	526.59	325.814	0
15	97.5781	162.81	637.125	325.814	0
20	124.862	190.103	672.481	325.814	0
27.36	225.906	261.7	672.481	325.814	0

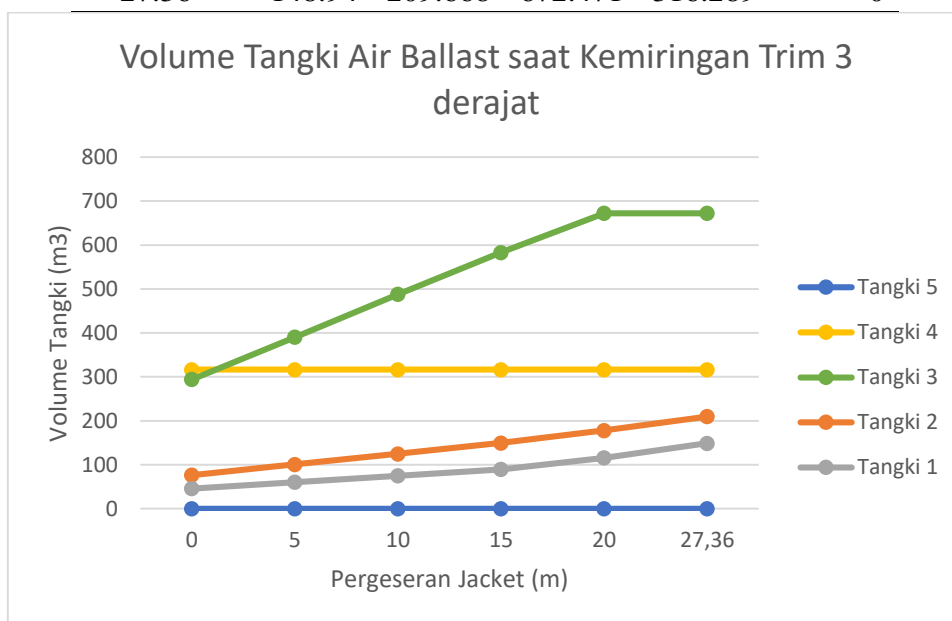


Gambar 4.21 Grafik hubungan Volume tangki dan pergeseran jacket pada Trim 2.5°

Gambar 4.21 menunjukkan volume tiap tangki pada setiap pergeseran jacket. Pada pergeseran jacket menuju 20 meter terjadi penurunan penambahan volume jika dilihat dari kemiringan grafiknya. Sehingga pada tangki 1 dan tangki 2 terjadi penambahan volume yang lebih banyak.

Tabel. 4.6 Hasil Perhitungan Volume Tangki pada Kemiringan Trim 3°

Pergeseran jacket (m)	Vol. tangki 1 (m ³)	Vol. tangki 2 (m ³)	Vol. tangki 3 (m ³)	Vol. tangki 4 (m ³)	Vol. tangki 5 (m ³)
0	45.822	76.383	294.13	316.289	0
5	60.388	100.667	390.367	316.289	0
10	74.966	124.97	487.833	316.289	0
15	89.536	149.456	583.105	316.289	0
20	115.65	178.073	672.471	316.289	0
27.36	148.94	209.668	672.471	316.289	0

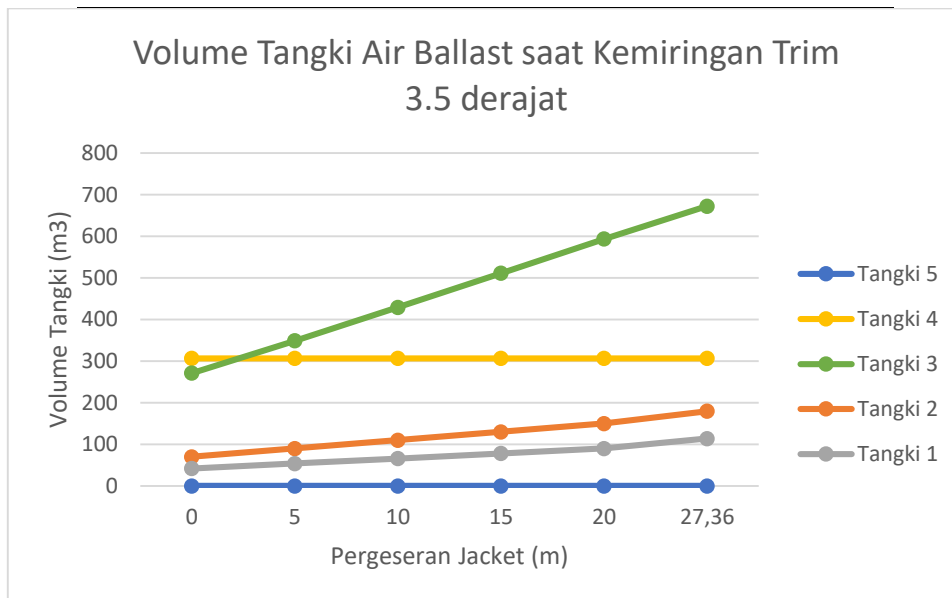


Gambar 4.22 Grafik hubungan Volume tangki dan pergeseran jacket pada Trim 3°

Pada Gambar 4.22 menunjukkan volume tiap tangki pada setiap pergeseran jacket di kemiringan trim 3°. Terlihat pada pergeseran jacket 20 meter tangki 3 akan mengalami volume maksimal sehingga tangki 1 akan mengalami penambahan yang lebih banyak untuk mengimbangi momen dari jacket.

Tabel 4.7 Hasil Perhitungan Volume Tangki pada Kemiringan Trim 3.5°

Pergeseran Jacket (m)	Vol. tangki 1 (m ³)	Vol. tangki 2 (m ³)	Vol. tangki 3 (m ³)	Vol. tangki 4 (m ³)	Vol. tangki 5 (m ³)
0	42.175	70.312	271.406	306.813	0
5	54.147	90.279	349.00	306.813	0
10	66.103	110.226	429.524	306.813	0
15	78.090	130.174	511.086	306.813	0
20	90.051	150.308	593.672	306.813	0
27.36	113.92	179.826	672.468	306.813	0

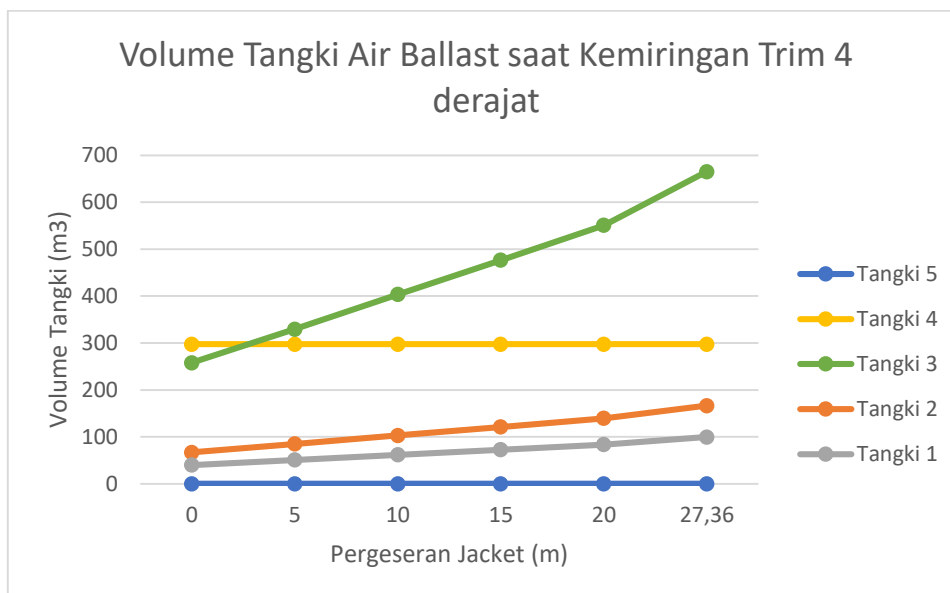


Gambar 4.23 Grafik hubungan volume tangki dan pergeseran jacket pada Trim 3.5°

Pada kemiringan trim 3.5° terlihat semakin besar pergeseran jacket maka semakin besar volume air pada setiap tangki. Terlihat pada setiap pergeseran jacket belum tangki 3 belum mengalami tangki maksimal.

Tabel 4.8 Hasil Perhitungan Volume Tangki pada Kemiringan Trim 4°

Pergeseran jacket (m)	Vol. tangki 1 (m ³)	Vol. tangki 2 (m ³)	Vol. tangki 3 (m ³)	Vol. tangki 4 (m ³)	Vol. tangki 5 (m ³)
0	42.175	70.312	271.406	306.813	0
5	54.147	90.279	349.00	306.813	0
10	66.103	110.226	429.524	306.813	0
15	78.090	130.174	511.086	306.813	0
20	90.051	150.308	593.672	306.813	0
27.36	113.92	179.826	672.468	306.813	0



Gambar 4.24 Grafik hubungan Volume tangki dan pergeseran jacket pada kemiringan 4°

Pada kemiringan trim 4° terlihat semakin besar pergeseran jacket maka semakin besar volume air pada setiap tangki. Terlihat pada pergeseran jacket 20 meter tangki 3 akan mengalami penambahan volume serta tangki 1 dan 2 akan mengalami penambahan volume juga.

Dengan semakin besar derajat kemiringan trim buritan pada barge maka semakin kecil pula volume yang diperlukan untuk membuat barge menjadi trim. Hal ini disebabkan oleh keperluan momen yang juga semakin kecil. Jika volume air

ballast semakin kecil maka displacement barge juga akan kecil dan draft akan juga akan secil.

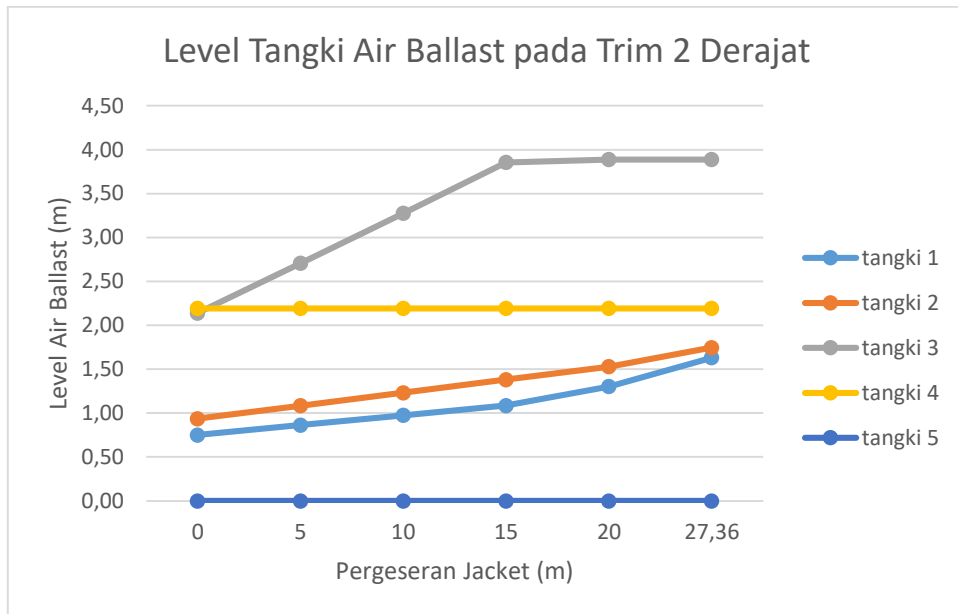
4.4.2 Hasil Perhitungan Level Tangki

Pengaturan ballast yang digunakan ialah dengan membagi momen secara merata pada setiap tangki sehingga diharapkan tidak membebani secara lebih pada struktur barge jika tangki berada jauh pada pusat titik apung dan menurunkan titik CG vertikal. Dengan menggunakan cara tersebut level air yang terdekat dengan pusat titik apung memiliki level air yang lebih tinggi. Pada penelitian ini dikarenakan kondisi barge yang miring sehingga terjadi perbedaan level diantara sekat yang lebih dekat dengan buritan dan sekat yang lebih dekat dengan haluan di tangki . Oleh karena itu, pada tabel terdapat merupakan nilai rata-rata dari level air untuk setiap tangkinya.

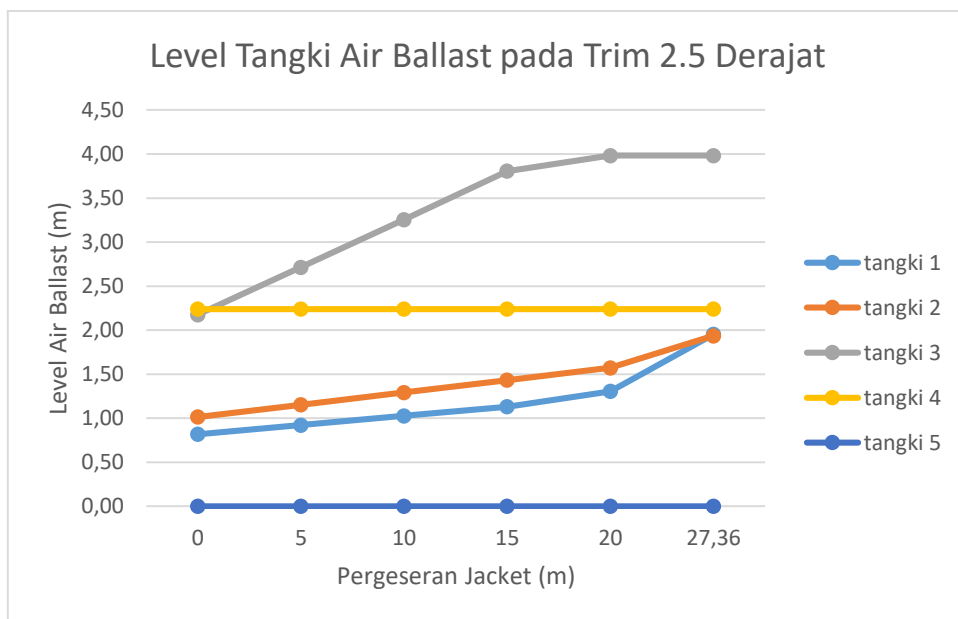
Tabel 4.9 Level Air Ballast pada masing – masing Tangki

Trim (Derajat)	Tangki	Pergeseran Jacket (m)					
		0	5	10	15	20	27.36
		Level Air Ballast (m)					
2	1	0.75	0.86	0.97	1.09	1.30	1.63
	2	0.94	1.08	1.23	1.38	1.53	1.75
	3	2.14	2.71	3.28	3.85	3.89	3.89
	4	2.19	2.19	2.19	2.19	2.19	2.19
	5	0.00	0.00	0.00	0.00	0.00	0.00
2.5	1	0.82	0.92	1.03	1.13	1.31	1.95
	2	1.01	1.15	1.29	1.43	1.57	1.94
	3	2.18	2.71	3.26	3.81	3.98	3.98
	4	2.24	2.24	2.24	2.24	2.24	2.24
	5	0.00	0.00	0.00	0.00	0.00	0.00
3	1	0.88	0.97	1.06	1.16	1.32	1.54
	2	1.09	1.21	1.33	1.46	1.58	1.77
	3	2.19	2.67	3.16	3.63	4.08	4.08
	4	2.29	2.29	2.29	2.29	2.29	2.29
	5	0.00	0.00	0.00	0.00	0.00	0.00
3.5	1	0.93	1.01	1.09	1.16	1.24	1.39
	2	1.15	1.25	1.36	1.46	1.56	1.71
	3	2.17	2.56	2.96	3.37	3.78	4.17
	4	2.33	2.33	2.33	2.33	2.33	2.33
	5	0.00	0.00	0.00	0.00	0.00	0.00
4	1	1.00	1.07	1.14	1.21	1.28	1.38
	2	1.23	1.32	1.41	1.51	1.60	1.74

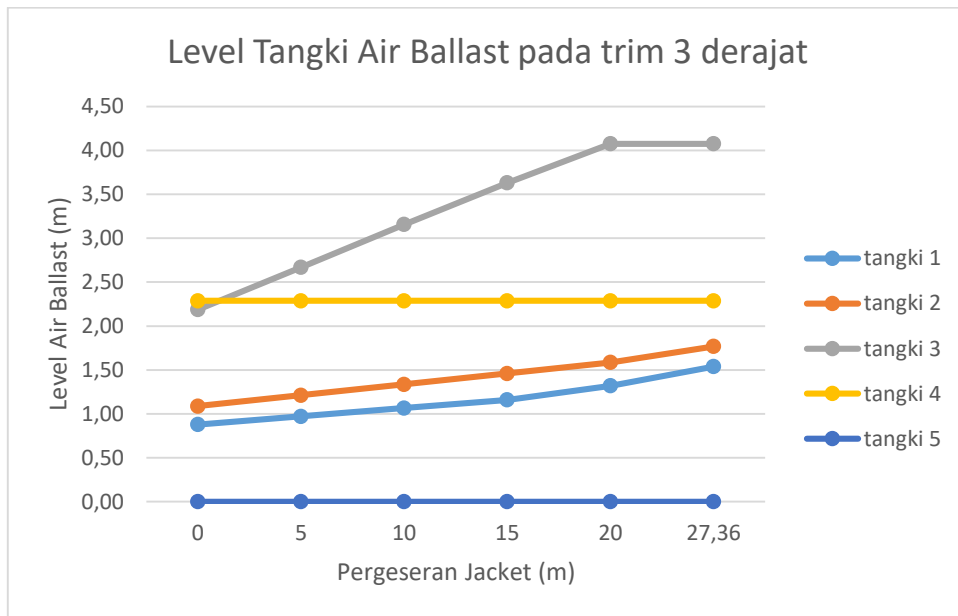
3	2.19	2.55	2.92	3.29	3.66	4.22
4	2.38	2.38	2.38	2.38	2.38	2.38
5	0.00	0.00	0.00	0.00	0.00	0.00



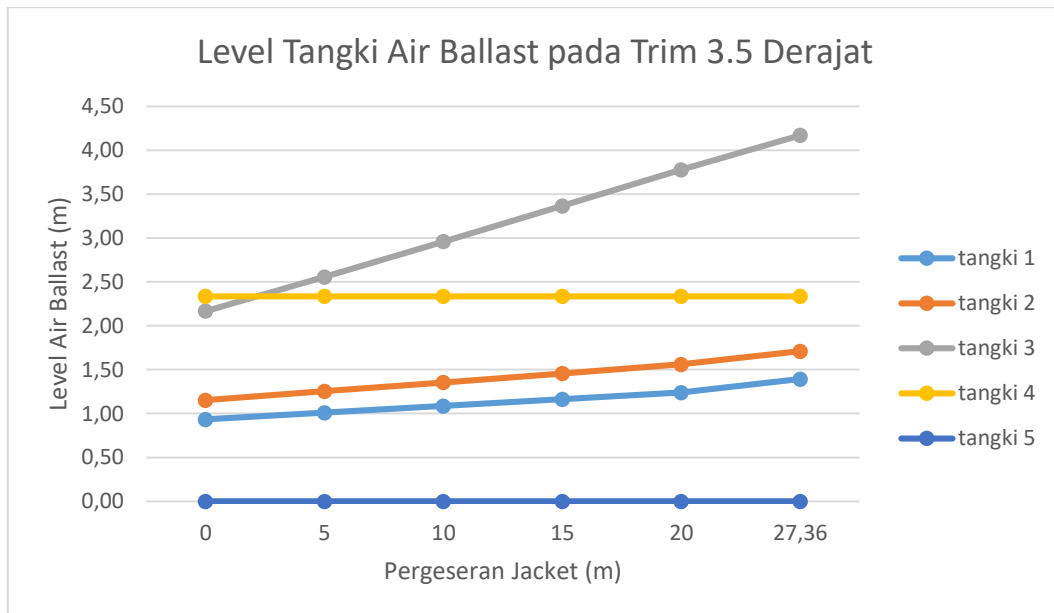
Gambar 4.25 Simulasi Perubahan Level Air pada Setiap Tangki Ballast Ketika Kapal Mengalami Trim Buritan 2°



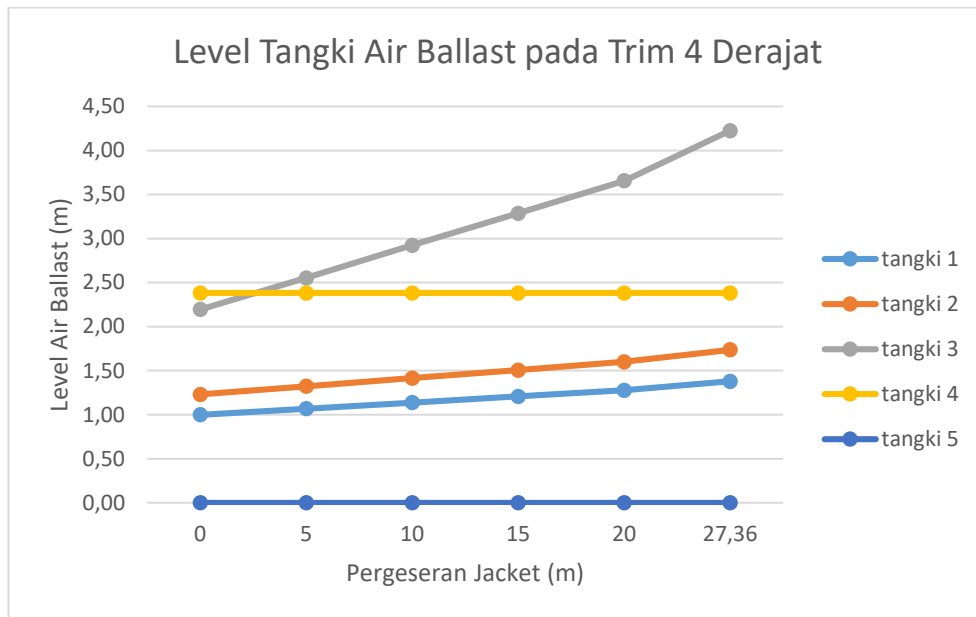
Gambar 4.26 Simulasi Perubahan Level Air pada Setiap Tangki Ballast Ketika Kapal Mengalami Trim Buritan 2.5°



Gambar 4.27 Simulasi Perubahan Level Air pada Setiap Tangki Ballast Ketika Kapal Mengalami Trim Buritan 3°



Gambar 4.28 Simulasi Perubahan Level Air pada Setiap Tangki Ballast Ketika Kapal Mengalami Trim Buritan 3.5°



Gambar 4.29 Simulasi Perubahan Level Air pada Setiap Tangki Ballast Ketika Kapal Mengalami Trim Buritan 4°

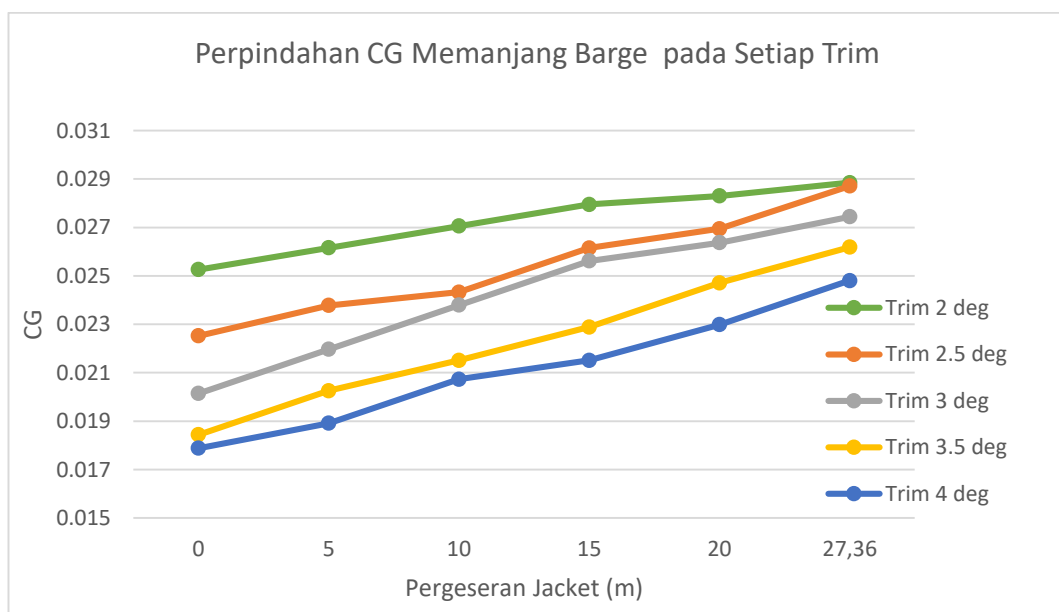
4.5 Pergeseran CG memanjang Kapal

Pada penelitian ini, pergeseran CG disebabkan oleh berpindahnya jacket dan penambahan air ballast pada tangki. Pada saat jacket mulai diluncurkan, tangki-tangki ballast akan diisi volume hasil perhitungan. Air ballast tersebut digunakan untuk menyeimbangkan barge sesuai trim buritan yang diinginkan. Dapat dilihat pada Tabel 4.10 dan Gambar 4.30 merupakan perpindahan CB dari barge. Pada saat seimbang, CG dan CB akan melewati satu garis gaya. Sehingga didapatkan posisi CB pada sumbu x juga merupakan CG pada sumbu x yang mana sumbu x sejajar dengan memanjang kapal, sumbu y sejajar dengan melintang kapal, dan sumbu z merupakan sumbu yang sejajar dengan sarat kapal.

Tabel 4.10 Posisi titik CB pada setiap Pergeseran Jacket

Trim	Pergeseran Jacket (m)	CB (m)		
		x	y	Z
2	0	25.256	-0.004	-1.300
	5	26.157	-0.004	-1.390
	10	27.058	-0.004	-1.469
	15	27.950	-0.005	-1.557
	20	28.300	-0.005	-1.601

	27.36	28.852	-0.005	-1.652
2.5	0	22.524	-0.008	-1.483
	5	23.775	-0.007	-1.602
	10	24.328	-0.007	-1.692
	15	26.149	-0.007	-1.825
	20	26.948	-0.007	-1.918
	27.36	28.711	-0.007	-2.068
3	0	20.143	-0.011	-1.559
	5	21.967	-0.011	-1.777
	10	23.792	-0.011	-1.975
	15	25.615	-0.011	-2.147
	20	26.367	-0.011	-2.413
	27.36	27.438	-0.011	-2.397
3.5	0	18.434	-0.014	-1.646
	5	20.254	-0.014	-1.823
	10	21.509	-0.015	-2.073
	15	22.888	-0.014	-2.227
	20	24.708	-0.014	-2.396
	27.36	26.184	-0.014	-2.631
4	0	17.879	-0.018	-1.791
	5	18.912	-0.018	-1.961
	10	20.727	-0.018	-2.146
	15	21.509	-0.019	-2.348
	20	22.987	-0.019	-2.584
	27.36	24.803	-0.019	-2.845



Gambar 4.30 Grafik Pergeseran CG memanjang Barge pada setiap Trim

4.6 Kondisi Barge Setelah Jacket Launching

Ketika barge telah melepaskan jacket, beban yang ditimbulkan oleh jacket akan menghilang tetapi beban pada air ballast di tangki tidak menghilang. Hal tersebut menyebabkan barge yang sebelumnya trim buritan untuk jacket launching menjadi trim haluan. Tabel 4.11 dan Gambar 4.31 menunjukkan kondisi barge setelah jacket launching.

Tabel 4.11 Derajat Trim Haluan setelah *Jacket Launching*

Derajat trim Buritan saat <i>Jacket Launching</i>	Derajat Trim Haluan setelah <i>Jacket Launching</i>
2	3.6
2.5	3.9
3	3.4
3.5	2.6
4	2



Gambar 4.31 Derajat Trim Haluan setelah Jacket Launching

4.7 Stabilitas Memanjang Barge

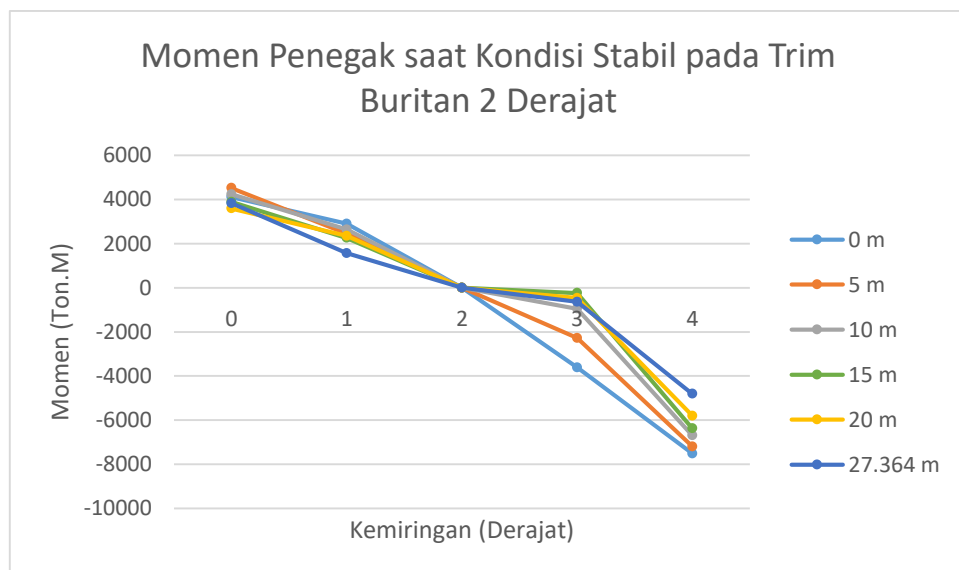
Stabilitas kapal adalah kemampuan kapal untuk menegak kembali ketika kapal diapungkan, tidak miring kekiri dan kekanan maupun trim haluan dan trim buritan, demikian pula pada saat berlayar, disebabkan oleh adanya pengaruh luar yang bekerja pada saat kapal diolengkan oleh ombak atau angin, kapal dapat tegak

kembali. Pada penelitian ini barge dimiringkan dengan variasi dua derajat lebih besar dan dua derajat lebih kecil dari kondisi stabilnya (kondisi trim yang diinginkan). Momen penegak dapat dihitung dengan menggunakan Persamaan 1.

Tabel 4.12 Momen Penegak saat Stabil pada Trim Buritan 2 derajat

Trim (Derajat)	Perpindahan Jacket (m)	Kemiringan (Derajat)				
		0	1	2	3	4
2	0	4108.518	2900.66	0	-3612.39	-7516.23
	5	4517.15	2460.74	0	-2280.78	-7196.17
	10	4233.45	2638.64	0	-948.75	-6695.04
	15	3889.432	2252.515	0	-240.633	-6379.4
	20	3587.73	2345.51	0	-463.575	-5804.31
	27.364	3831.642	1568.237	0	-639.557	-4810.84

Pada Tabel 4.12 dan Gambar 2.32 menunjukkan momen penegak pada saat kondisi stabil di trim 2 derajat. Pada derajat 0 dan 1 memiliki nilai momen penegak positif yang menandakan momen mengarah ke buritan untuk mengembalikan ke kondisi 2 derajat dan begitu pula pada derajat 3 dan 4 bernilai negatif menandakan momen mengarah ke haluan untuk mengembalikan kondisi trim ke 2 derajat. Pada derajat 2, momen penegak bernilai nol dikarenakan nilai lengan penegak juga bernilai nol.

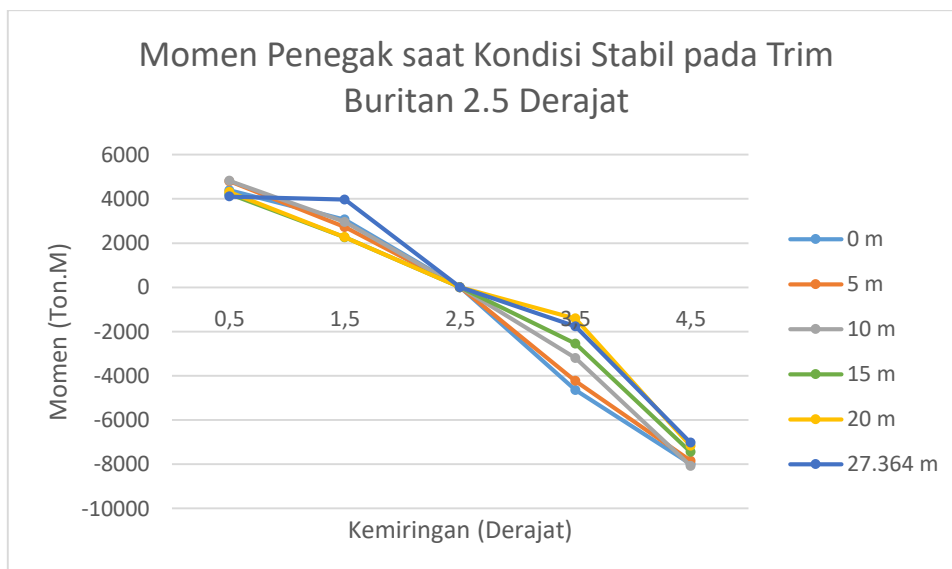


Gambar 4.32 Grafik Momen Penegak saat Trim Buritan 2 derajat

Tabel 4.13 Momen Penegak saat Stabil pada Trim Buritan 2.5 derajat

Trim (Derajat)	Perpindahan Jacket (m)	Kemiringan (Derajat)				
		0.5	1.5	2.5	3.5	4.5
2.5	0	4388.007	3060.83	0	-4646.63	-7990.13
	5	4795.108	2720.1	0	-4231.45	-7845.13
	10	4816.5	2950.3	0	-3200.12	-8079.22
	15	4211.238	2260.95	0	-2548.14	-7448.29
	20	4307.815	2264.106	0	-1410.34	-7173.4
	27.364	4106.95	3965.86	0	-1767.36	-7032.57

Pada Tabel 4.13 dan Gambar 2.33 menunjukkan momen penegak pada saat kondisi stabil di trim 2.5 derajat. Variasi derajat yang digunakan ialah 0.5, 1.5, 2.5, 3.5, dan 4.5. Pada derajat 0.5 dan 1.5 memiliki nilai momen penegak positif untuk mengembalikan ke kondisi 2 derajat dan begitu pula pada derajat 3.5 dan 4.5 bernilai negatif untuk mengembalikan kondisi trim ke 2 derajat. Momen penegak yang bernilai positif dan negatif menunjukkan arah berputar terhadap porosnya dalam hal ini CG. Positif menandakan momen berputar kearah buritan dan negatif ke arah haluan. Pada derajat 2.5, momen penegak bernilai nol dikarenakan nilai lengan penegak juga bernilai nol.

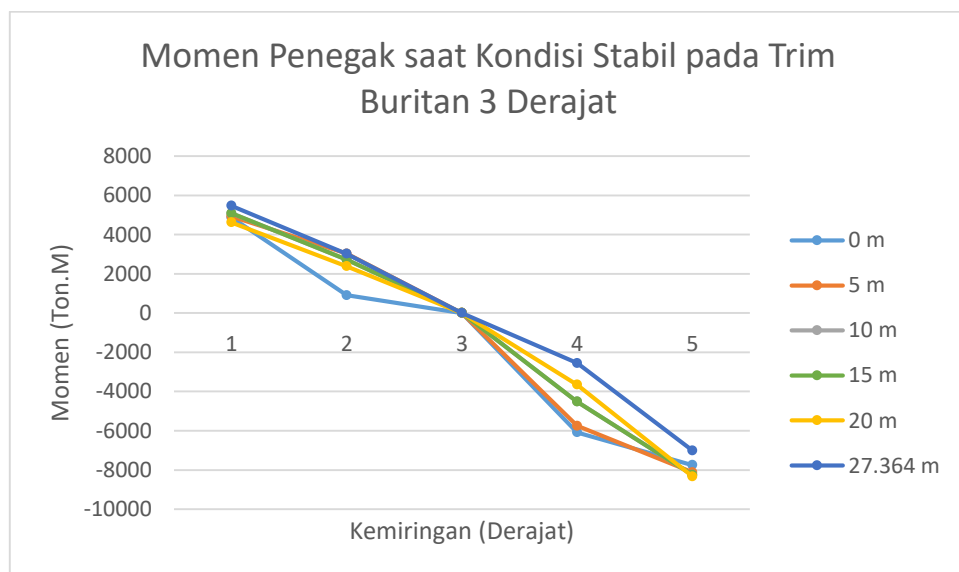


Gambar 4.33 Grafik Momen Penegak saat Trim Buritan 2.5 derajat

Tabel 4.14 Momen Penegak saat Stabil pada Trim Buritan 3 derajat

Trim (Derajat)	Perpindahan Jacket (m)	Kemiringan (Derajat)				
		1.	2	3	4	5
		Momen Penegak (Ton.m)				
3	0	4449.31	3459.02	0	-5648.36	-8256.71
	5	4929.802	3026.183	0	-5748.65	-8116.56
	10	4726.19	2997.4	0	-5010.06	-8364.74
	15	5080.501	2720.07	0	-4515.46	-8244.04
	20	4621.86	2385.256	0	-3650.32	-8326.38
	27.364	5470.383	3016.04	0	-2552.71	-7007.39

Pada Tabel 4.14 dan Gambar 2.34 menunjukkan momen penegak pada saat kondisi stabil di trim 3 derajat. Variasi derajat yang digunakan ialah 1, 2, 3, 4, dan 5. Pada derajat 1 dan 2 memiliki nilai momen penegak positif untuk mengembalikan ke kondisi 3 derajat dan begitu pula pada derajat 4 dan 5 bernilai negatif untuk mengembalikan kondisi trim ke 2 derajat. Momen penegak yang bernilai positif dan negatif menunjukkan arah berputar terhadap porosnya dalam hal ini CG, positif menunjukkan momen berputar kearah buritan dan negatif kearah haluan. Pada derajat 3, momen penegak bernilai nol dikarenakan nilai lengan penegak juga bernilai nol atau dalam keadaan stabil.

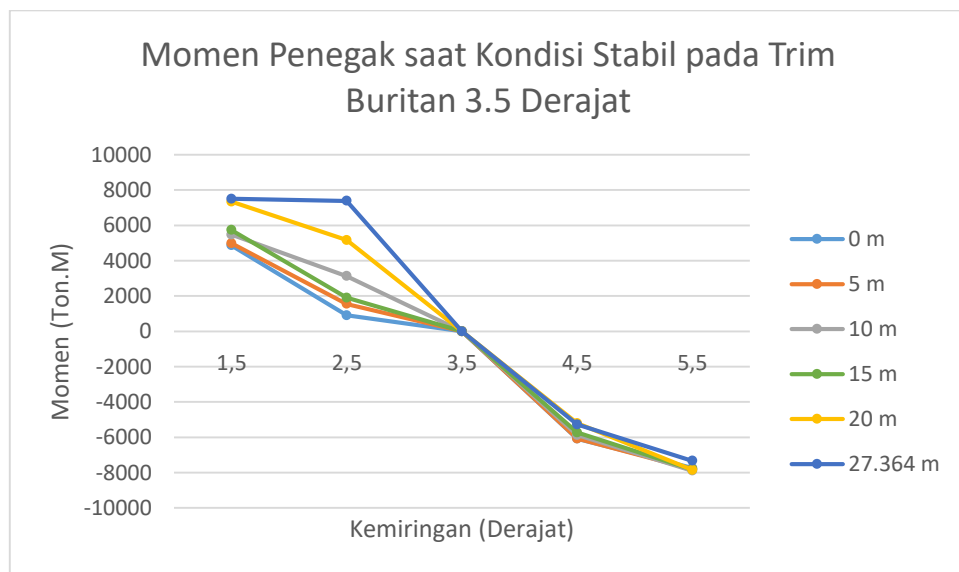


Gambar 4.34 Grafik Momen Penegak saat Trim Buritan 3 derajat

Tabel 4.15 Momen Penegak saat Stabil pada Trim Buritan 3.5 derajat

Trim (Derajat)	Perpindahan Jacket (m)	Kemiringan (Derajat)				
		1.5	2.5	3.5	4.5	5.5
3.5	0	4869.496	906.4	0	-6089.61	-7748.41
	5	4980.29	1543.64	0	-6074.12	-7805.59
	10	5468.908	3127.35	0	-5872.8	-7899.74
	15	5507.871	4184.618	0	-5479.32	-8331.94
	20	6155.267	4461.5	0	-4857.74	-8006.09
	27.364	6117.229	4898.607	0	-4012.9	-6104.51

Pada Tabel 4.15 dan Gambar 2.35 menunjukkan momen penegak pada saat kondisi stabil di trim 3.5 derajat. Variasi derajat yang digunakan ialah 1.5, 2.5, 3.5, 4.5, dan 5.5. Pada derajat 1.5 dan 2.5 memiliki nilai momen penegak positif untuk mengembalikan ke kondisi 3.5 derajat dan begitu pula pada derajat 3.5 dan 4.5 bernilai negatif untuk mengembalikan kondisi trim ke 3.4 derajat. Momen penegak yang bernilai positif dan negatif menunjukkan arah berputar terhadap porosnya dalam hal ini CG, positif menunjukkan momen berputar kearah buritan dan negatif kearah haluan. Pada derajat 3.5, momen penegak bernilai nol dikarenakan nilai lengan penegak juga bernilai nol atau dalam keadaan stabil.

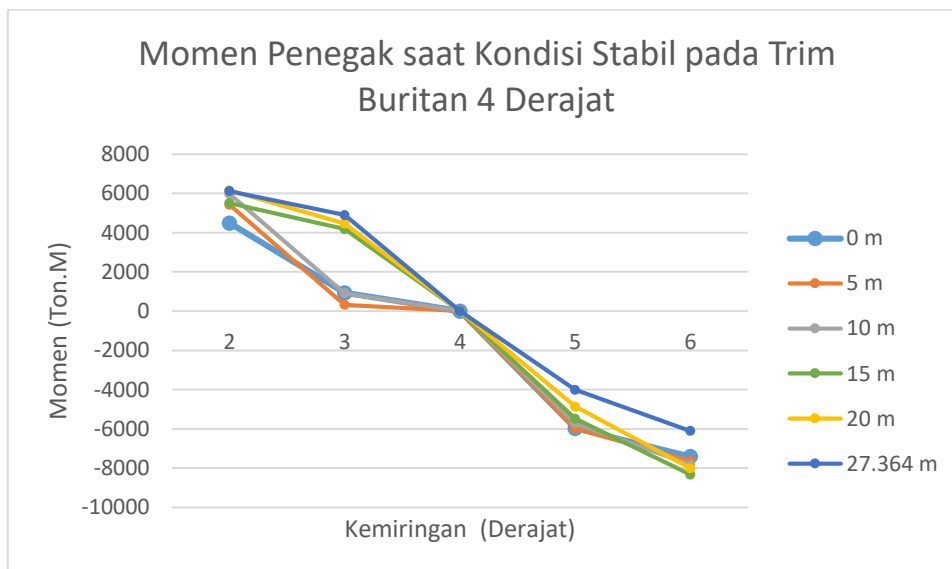


Gambar 4.35 Grafik Momen Penegak saat Trim Buritan 3.5 derajat

Tabel 4.16 Momen Penegak saat Stabil pada Trim Buritan 4 derajat

Trim (Derajat)	Perpindahan Jacket (m)	Kemiringan (Derajat)				
		2	3	4	5	6
		Momen Penegak (Ton.m)				
4	0	4488.651	936.92	0	-5965.24	-7411.79
	5	5418.55	318.8221	0	-5982.57	-7642.99
	10	5977.52	912.1785	0	-5719.6	-7827.61
	15	5738.506	1900.669	0	-5728.54	-7827.61
	20	7342.518	5161.296	0	-5202.77	-7850.82
	27.364	7501.348	7381.927	0	-5283.18	-7336.08

Pada Tabel 4.16 dan Gambar 2.36 menunjukkan momen penegak pada saat kondisi stabil di trim 4 derajat. Variasi derajat yang digunakan ialah 2, 3, 4, 5, dan 6. Pada derajat 2 dan 3 memiliki nilai momen penegak positif untuk mengembalikan ke kondisi 4 derajat dan begitu pula pada derajat 5 dan 6 bernilai negatif untuk mengembalikan kondisi trim ke 4 derajat. Momen penegak yang bernilai positif dan negatif menunjukkan arah berputar terhadap porosnya dalam hal ini CG, positif menunjukkan momen berputar kearah buritan dan negatif kearah haluan. Pada derajat 4, momen penegak bernilai nol dikarenakan nilai lengan penegak juga bernilai nol atau dalam keadaan stabil.



Gambar 4.36 Grafik Momen Penegak saat Trim Buritan 4 derajat

BAB 5. PENUTUP

5.1 Kesimpulan

1. Terdapat 2 model yang dibuat pada penelitian ini yaitu barge dan jacket. Kedua model tersebut dibuat dengan cara yang berbeda, Dimana;
 - Untuk membuat model 3 dimensi didalam Scilab diperlukan data offset yang berasal dari tabel offset aplikasi Maxsurf. Data offset tersebut memerlukan program untuk membuat panel-panel yang jika digabungkan akan membentuk lambung barge.
 - *Jacket* dibuat dengan struktur sederhana untuk mempermudah penelitian ini mendapatkan *Center of Gravity*(CG). Fungsi program dibuat untuk membuat model dari jacket yaitu `generateMeshJacket()`. Didalam fungsi tersebut terdapat fungsi-fungsi lain yaitu `meshPipeOffset()`, `geotransformation()`, `plotMesh()`, dan `arrangeData()`. Hasil pemodelan jacket didapatkan berat yaitu 255.03 Ton dan CoG yaitu $X = 27.55$ m, $Y = 0.09$ m dan $Z = 8.61$ m.
2. Program untuk menghitung volume ballast saat keadaan trim adalah menggunakan fungsi program `buoyancy()`. Fungsi program tersebut menggunakan tekanan yang diterima dari setiap panel untuk menghitung volume tangki. Untuk membuat tangki dalam keadaan trim maka offset dari tangki akan di rotasi menggunakan fungsi program `geotransformation()`.
3. Pengaturan ballast yang dibuat dengan mempertimbangkan momen untuk menyeimbangkan dibagi secara merata sehingga struktur tidak terbebani terlalu besar. Hal tersebut juga menyebabkan titik CG secara vertikal akan mendekati bottom dari barge.

5.2 Saran

1. Untuk melakukan penelitian selanjutnya dapat menggunakan keadaan yang dinamis dan memperhatikan keadaan dari jacket serta memperhitungkan kekuatan struktur dari barge

DAFTAR PUSTAKA

- Amin, I. H., Lusiana, V., & Hartono, B. (2018). Pencarian Lintasan pada Collision Detection Menggunakan Pendekatan Interpolasi Linier. *Prosiding SINTAK 2018*, 57–61.
- Arief, S. (2015). *Pengenalan Scilab*.
- Astutiuk, A. E. (2014). *Transformasi Geometri dan Poligonal*.
- Gerwick, B. (1999). Construction of Marine and Offshore Structures, Second Edition. In *Construction of Marine and Offshore Structures, Second Edition*. <https://doi.org/10.1201/9781420049602>
- HMSF. (n.d.). *MODUL LATIHAN SOFTWARE MAXSURF*.
- Honarvar, M. R., Pirooz, M. D., & Bahaari, M. R. (2008). A physical and numerical modeling for launching of jackets (case study on Balal PLQ Platform). *Journal of Offshore Mechanics and Arctic Engineering*, 130(3), 1–8. <https://doi.org/10.1115/1.2904714>
- IdSchool. (n.d.). *Perkalian Silang Vektor (Vector Cross Product) _ idschool*. Retrieved May 14, 2022, from <https://idschool.net/sma/perkalian-silang-vektor-cross-product-a-x-b/>
- ITS. (n.d.). *Luas dan volume*. Share ITS. [http://share.its.ac.id/pluginfile.php/14931/mod_resource/content/4/Luas dan Volume.pdf](http://share.its.ac.id/pluginfile.php/14931/mod_resource/content/4/Luas%20dan%20Volume.pdf)
- Larsen, N. L. (2012). *Understanding The Trimming a vessel. March*.
- Noble Denton, G. (2015). *TECHNICAL STANDARDS COMMITTEE 0032 / ND This document has been replaced by the standard which may be accessed through https://my.dnvgl.com/ This document may still be valid for some existing projects*. 1–39.
- Ongga, P., Sanwaty, Y., Rondonuwu Samuel, F., & Kristiyanto Hari, W. (2009). Konsepsi mahasiswa tentang tekanan hidrostatik. *Fakultas MIPA, Universitas Negeri Yogyakarta*, 181–185.
- Rachman, I., Subiyanto, L., Suhardjito, G., & Indartono, A. (2014). Identifikasi Garis Stabilitas Melintang Kapal Melalui Percobaan Kemiringan Menggunakan Delphi Berbasis Arduino. *Transmisi*, 16(3), 121-127–127. <https://doi.org/10.12777/transmisi.16.3.121-127>
- Sugiono, & Rofifah, D. (2004). *Teknologi Produksi Dan Perawatan Bangunan Laut*. 12–26.
- Ueng, S. K. (2013). Physical models for simulating ship stability and hydrostatic motions. *Journal of Marine Science and Technology (Taiwan)*, 21(6), 674–685. <https://doi.org/10.6119/JMST-012-1121-1>

LAMPIRAN

Lampiran 1. Fungsi Program genof()

```
function [data, allOffset, allPanel]=genof(x)

for i = 1:x(size(x,1),1)
    offset = []
    for j = 1: size(x,1)
        if (i == x(j,1)) then
            offset = [offset; x(j,2:4)]
            data.station (i).offset = [offset]
        end
    end
end
// offset lambung selesai

id = 0
offset =[]
for i = 1: size(data.station.offset)
    R = [flipdim(data.station(i).offset,1)]
    L = [data.station(i).offset(2:10,1) data.station(i).offset(2:10,2)*-1
data.station(i).offset(2:10,3)]
    re = [R;L]
    id4=[]
    for j = 1: size(re,1)
        id = id +1
        offset = [offset; id re(j,:)]
        id4=[id4;id]
    end
    data.station(i).re = [re id4]
end
// panel lambung selesai

idx = 0 // index untuk offset lambung
```



```

panel = []
for i = 1 : length(data.station.re)-1
    n=[]
    for j = 1 :size(data.station(i).re,1)-1
        idx = idx + 1
        t1 = data.station(i).re(j,4)
        t2 = data.station(i).re(j+1,4)
        t3 = data.station(i+1).re(j,4)
        t4 = data.station(i+1).re(j+1,4)
        panel = [panel;idx t3 t4 t2 t1 t3] //membuat panel
    end
end
offset2=[]
for i = 1:size(offset,1)
    offset2 = [offset2; offset(i,1:3) offset(i,4)-draft]
end
allOffset=[]
allPanel=[]
t = 15/1000 // ketebalan 15 mm
[allOffset,allPanel] = arrangeData(allOffset,allPanel,offset2,panel,t)

sekat = [ 1;7;12; 18; 24; 30; 35;38] // start sekat
idt = 0
for rl = 1 : 2
    for n = 1 :length(sekat)
        o = sekat(n,1)
        offset =[]
        idx = 0
        for i = 1:size(data.station(1).offset,1)

```

```

for j = 1:size(data.station(1).offset,1)
    if (j <= i) then
        idx = idx +1
        if rl ==1 then
            offset = [offset;idx data.station(o).offset(j,1)
data.station(o).offset(j,2) data.station(o).offset(i,3)-draft]
        else
            offset = [offset;idx data.station(o).offset(j,1) data.station(o).offset(j,2)*-1
data.station(o).offset(i,3)-draft]
        end
    end
end
end
end
if rl ==1 then idt = idt + 1;data.sekat(idt).offset = [offset; offset(:,1)+55
offset(:,2) offset(:,3)*-1 offset(:,4)] end
panel = [] //panel buritan dan haluan
panel2=[]
idx = 0
j = 1
for i = 1:size(offset,1)-10
    if rl ==1 then
        if offset(i,3) < offset(i+1,3) && offset(i,4) == offset(i+1,4)then
            t1 = offset(i,1)
            t2 = offset(i+1,1)
            t3 = offset(i+j,1)
            t4 = offset(i+j+1,1)
            idx = idx+1
            panel = [panel; idx t1 t2 t4 t3 t1]
        else
            idx = idx + 1

```

```

        t1 = offset(i,1)
        t2 = offset(i+j,1)
        t3 = offset(i+j+1,1)
        panel=[panel; idx t1 t2 t3 t1 t1]
        j = j + 1
    end
else
    if offset(i,3) > offset(i+1,3) && offset(i,4) == offset(i+1,4) then
        t1 = offset(i,1)
        t2 = offset(i+1,1)
        t3 = offset(i+j,1)
        t4 = offset(i+j+1,1)
        idx = idx+1
        panel = [panel; idx t1 t3 t4 t2 t1]
    else
        idx = idx + 1
        t1 = offset(i,1)
        t2 = offset(i+j,1)
        t3 = offset(i+j+1,1)
        panel=[panel; idx t3 t1 t2 t3 t3]
        j = j + 1
    end
end
end
if o == 1 || o == length(data.station.offset) then
    if o == 1 then panel = [panel(:,1) flipdim(panel(:,2:6),2)] end
    t = 15/1000 // ketebalan 15 mm
    [allOffset,allPanel] = arrangeData(allOffset,allPanel,offset,panel,t)
end
end
end

```

```

end
zz=[panel(:,2:6)]
zzz= [panel(:,1) flipdim(zz,2); panel(:,1)+45 zz+55]

offset=[]           // membuat deck
for i = 1: length(data.station.re)
    for j = 1: size(data.station(i).re,1)

        offset = [offset; data.station(i).re(j,4) data.station(i).re(j,1:2)
data.station(i).re(1,3) ]
    end
end
offsetdeck = [offset(:,1:3) offset(:,4)-draft]
idx = 0           // membuat panel
panel = []
for i = 1 : length(data.station.re)-1
    for j = 1 :size(data.station(i).re,1)-1
        idx = idx + 1
        t1 = data.station(i).re(j,4)
        t2 = data.station(i).re(j+1,4)
        t3 = data.station(i+1).re(j,4)
        t4 = data.station(i+1).re(j+1,4)
        panel = [panel;idx t1 t2 t4 t3 t1] //membuat panel
    end
end
t = 8/1000// ketebalan 8 mm
[allOffset,allPanel] = arrangeData(allOffset,allPanel,offsetdeck,panel,t)
//plotMesh(allOffset,allPanel)
t = 10/1000

```

```

// tangki tangki//
data.tank(1).offset = [allOffset(115:228,1)-114 allOffset(115:228,2:4);
data.sekat(2).offset(:,1)+114 data.sekat(2).offset(:,2:4) ;
data.sekat(3).offset(:,1)+114+110 data.sekat(3).offset(:,2:4)]

data.tank(1).panel = [allPanel(109:198,1)-108 allPanel(109:198,2:6)-114 ; zzz(:,1)+
90 flipdim(zzz(:,2:6),2)+114 ; zzz(:,1)+ 90+90 zzz(:,2:6)+114+110 ]

data.tank(2).offset = [allOffset(210:342,1)-209 allOffset(210:342,2:4) ;
data.sekat(3).offset(:,1)+133 data.sekat(3).offset(:,2:4) ;
data.sekat(4).offset(:,1)+133+110 data.sekat(4).offset(:,2:4)]

data.tank(2).panel = [allPanel(199:306,1)-198 allPanel(199:306,2:6)-209 ; zzz(:,1)+
108 flipdim(zzz(:,2:6),2)+133 ; zzz(:,1)+ 108+90 zzz(:,2:6)+133+110 ]

data.tank(3).offset = [allOffset(324:456,1)-323 allOffset(324:456,2:4);
data.sekat(4).offset(:,1)+133 data.sekat(4).offset(:,2:4) ;
data.sekat(5).offset(:,1)+133+110 data.sekat(5).offset(:,2:4)]

data.tank(3).panel = [allPanel(307:414,1)-306 allPanel(307:414,2:6)-323 ; zzz(:,1)+
108 flipdim(zzz(:,2:6),2)+133 ; zzz(:,1)+ 108+90 zzz(:,2:6)+133+110 ]

data.tank(4).offset = [allOffset(438:570,1)-437 allOffset(438:570,2:4) ;
data.sekat(5).offset(:,1)+133 data.sekat(5).offset(:,2:4) ;
data.sekat(6).offset(:,1)+133+110 data.sekat(6).offset(:,2:4)]

data.tank(4).panel = [allPanel(415:522,1)-414 allPanel(415:522,2:6)-437 ; zzz(:,1)+
108 flipdim(zzz(:,2:6),2)+133 ; zzz(:,1)+ 108+90 zzz(:,2:6)+133+110 ]

```

```

data.tank(5).offset = [allOffset(552:665,1)-551 allOffset(552:665,2:4) ;
data.sekat(6).offset(:,1)+114 data.sekat(6).offset(:,2:4) ;
data.sekat(7).offset(:,1)+114+110 data.sekat(7).offset(:,2:4)]

data.tank(5).panel = [allPanel(523:612,1)-522 allPanel(523:612,2:6)-551 ; zzz(:,1)+
90 flipdim(zzz(:,2:6),2)+114 ; zzz(:,1)+ 90+90 zzz(:,2:6)+114+110 ]
for i = 1: length(data.tank.panel)
    te = ones(size(data.tank(i).panel,1),1)
    temp = [data.tank(i).panel te]
    data.tank(i).panel = [data.tank(i).panel te]
end
endfunction

```

Lampiran 2. Fungsi Program generateMeshJacket()

```

function [allOffset2, allPanel2]=generateMeshJacket(draw)
// clf
t = 10/1000
    allOffset2=[]; allPanel2=[];
    P0=[57.5 8 11.24]; P1=[0 8 11.24]; A =[0 360]; R = 1; N=[10 8]; arahPanel=-1;
[offset,panel,idx,k] = meshPipeOffset(P0,P1,A,R,N,arahPanel,draw);crotasi = [P0];
drotasi = [ 0 0.16 6.1959];daxial = [0 0 0];[offset]=
geotransformation(offset,crotasi,drotasi,daxial);[allOffset2,allPanel2]=arrangeData(
allOffset2,allPanel2,offset,panel,t); //kaki kiri atas

    P0=[57 -8 11.24]; P1=[0 -8 11.24]; A =[0 360]; R = 1; N=[10 8]; arahPanel=-1;
[offset,panel,idx,k] = meshPipeOffset(P0,P1,A,R,N,arahPanel,draw);crotasi = [P0];
drotasi = [ 0 0.16 0.08];daxial = [0 0 0];[offset]=
geotransformation(offset,crotasi,drotasi,daxial);[allOffset2,allPanel2]=arrangeData(
allOffset2,allPanel2,offset,panel,t); //kaki kanan atas

    P0=[56 8 1.24]; P1=[0 8 1.24]; A =[0 360]; R = 1; N=[10 8]; arahPanel=-1;
[offset,panel,idx,k] = meshPipeOffset(P0,P1,A,R,N,arahPanel,draw);crotasi = [P0];
drotasi = [ 0 0 6.1959];daxial = [0 0 0];[offset]=
geotransformation(offset,crotasi,drotasi,daxial);[allOffset2,allPanel2]=arrangeData(
allOffset2,allPanel2,offset,panel,t); //kaki kiri bawah

    P0=[56 -8 1.24]; P1=[0 -8 1.24]; A =[0 360]; R = 1; N=[10 8]; arahPanel=-1;
[offset,panel,idx,k] = meshPipeOffset(P0,P1,A,R,N,arahPanel,draw);crotasi = [P0];
drotasi = [ 0 0 0.08];daxial = [0 0 0];[offset]=
geotransformation(offset,crotasi,drotasi,daxial);[allOffset2,allPanel2]=arrangeData(
allOffset2,allPanel2,offset,panel,t); //kaki kanan bawah

    P0=[55 8 1.24]; P1=[55 8 11.24]; A =[0 360]; R = 0.5; N=[4 8]; arahPanel=-1;
[offset,panel,idx,k] = meshPipeOffset(P0,P1,A,R,N,arahPanel,draw);crotasi = [P0];
drotasi = [ 0 0 0.08 0];daxial = [0 0 0];[offset]=

```

```

P0=[55 8 1.24]; P1=[55 8 11.24]; A =[0 360]; R = 0.5; N=[4 8]; arahPanel=-1;
[offset,panel,idx,k] = meshPipeOffset(P0,P1,A,R,N,arahPanel,draw);crotasi = [P0];
drotasi = [ 0 0.08 0];daxial = [0 0 0];[offset]=
geotransformation(offset,crotasi,drotasi,daxial);[allOffset2,allPanel2]=arrangeData(
allOffset2,allPanel2,offset,panel,t); // antar kaki kiri vertikal

```

```

P0=[55 -8 1.24]; P1=[55 -8 11.24]; A =[0 360]; R = 0.5; N=[4 8]; arahPanel=-1;
[offset,panel,idx,k] = meshPipeOffset(P0,P1,A,R,N,arahPanel,draw);crotasi = [P0];
drotasi = [ 0 0.08 0];daxial = [0 0 0];[offset]=
geotransformation(offset,crotasi,drotasi,daxial);[allOffset2,allPanel2]=arrangeData(
allOffset2,allPanel2,offset,panel,t); // antar kaki kanan vertikal

```

```

P0=[55 -8 1.24]; P1=[55 8 1.24]; A =[0 360]; R = 0.5; N=[4 8]; arahPanel=-1;
[offset,panel,idx,k] = meshPipeOffset(P0,P1,A,R,N,arahPanel,draw);crotasi = [P0];
drotasi = [ 0 0 0];daxial = [0 0 0];[offset]=
geotransformation(offset,crotasi,drotasi,daxial);[allOffset2,allPanel2]=arrangeData(
allOffset2,allPanel2,offset,panel,t); // antar kaki bawah horizontal

```

```

P0=[55.7 -8 11.24]; P1=[55.7 8 11.24]; A =[0 360]; R = 0.5; N=[4 8]; arahPanel=-
1;
[offset,panel,idx,k] = meshPipeOffset(P0,P1,A,R,N,arahPanel,draw);crotasi = [P0];
drotasi = [ 0 0 0];daxial = [0 0 0];[offset]=
geotransformation(offset,crotasi,drotasi,daxial);[allOffset2,allPanel2]=arrangeData(
allOffset2,allPanel2,offset,panel,t); // antar kaki atas horizontal

```

```

P0=[29 -10.36 15.84]; P1=[29 10.36 15.84]; A =[0 360]; R = 0.5; N=[4 8];
arahPanel=-1;
[offset,panel,idx,k] = meshPipeOffset(P0,P1,A,R,N,arahPanel,draw);crotasi = [P0];
drotasi = [ 0 0 0];daxial = [0 0 0];[offset]=
geotransformation(offset,crotasi,drotasi,daxial);[allOffset2,allPanel2]=arrangeData(
allOffset2,allPanel2,offset,panel,t); // antar kaki atas horizontal mid

```



```

P0=[28 -10.36 1.24]; P1=[28 10.36 1.24]; A =[0 360]; R = 0.5; N=[4 8];
arahPanel=-1;
[offset,panel,idx,k] = meshPipeOffset(P0,P1,A,R,N,arahPanel,draw);crotasi = [P0];
drotasi = [ 0 0 0];daxial = [0 0 0];[offset]=
geotransformation(offset,crotasi,drotasi,daxial);[allOffset2,allPanel2]=arrangeData(
allOffset2,allPanel2,offset,panel,t); // antar kaki bawah horizontal mid

```

```

P0=[28 10.36 1.24]; P1=[28 10.36 15.84]; A =[0 360]; R = 0.5; N=[4 8];
arahPanel=-1;
[offset,panel,idx,k] = meshPipeOffset(P0,P1,A,R,N,arahPanel,draw);crotasi = [P0];
drotasi = [ 0 0.08 0];daxial = [0 0 0];[offset]=
geotransformation(offset,crotasi,drotasi,daxial);[allOffset2,allPanel2]=arrangeData(
allOffset2,allPanel2,offset,panel,t); // antar kaki kiri vertikan mid

```

```

P0=[28 -10.36 1.24]; P1=[28 -10.36 15.84]; A =[0 360]; R = 0.5; N=[4 8];
arahPanel=-1;
[offset,panel,idx,k] = meshPipeOffset(P0,P1,A,R,N,arahPanel,draw);crotasi = [P0];
drotasi = [ 0 0.08 0];daxial = [0 0 0];[offset]=
geotransformation(offset,crotasi,drotasi,daxial);[allOffset2,allPanel2]=arrangeData(
allOffset2,allPanel2,offset,panel,t); // antar kaki kanan vertikan mid

```

```

P0=[3 -12 1.24]; P1=[3 12 1.24]; A =[0 360]; R = 0.5; N=[4 8]; arahPanel=-1;
[offset,panel,idx,k] = meshPipeOffset(P0,P1,A,R,N,arahPanel,draw);crotasi = [P0];
drotasi = [ 0 0 0];daxial = [0 0 0];[offset]=
geotransformation(offset,crotasi,drotasi,daxial);[allOffset2,allPanel2]=arrangeData(
allOffset2,allPanel2,offset,panel,t); // antar kaki bawah horizontal ap

```

```

P0=[3 -12 1.24]; P1=[3 -12 19.74]; A =[0 360]; R = 0.5; N=[4 8]; arahPanel=-1;
[offset,panel,idx,k] = meshPipeOffset(P0,P1,A,R,N,arahPanel,draw);crotasi = [P0];
drotasi = [ 0 0.08 0];daxial = [0 0 0];[offset]=
geotransformation(offset,crotasi,drotasi,daxial);[allOffset2,allPanel2]=arrangeData(
allOffset2,allPanel2,offset,panel,t); // antar kaki kanan vertikln ap

```

```

P0=[3 12 1.24]; P1=[3 12 19.74]; A =[0 360]; R = 0.5; N=[4 8]; arahPanel=-1;
[offset,panel,idx,k] = meshPipeOffset(P0,P1,A,R,N,arahPanel,draw);crotasi = [P0];
drotasi = [ 0 0.08 0];daxial = [0 0 0];[offset]=
geotransformation(offset,crotasi,drotasi,daxial);[allOffset2,allPanel2]=arrangeData(
allOffset2,allPanel2,offset,panel,t); // antar kaki kiri vertikal ap

```

```

P0=[4.5 -12 19.74]; P1=[4.5 12 19.74]; A =[0 360]; R = 0.5; N=[4 8]; arahPanel=-
1;
[offset,panel,idx,k] = meshPipeOffset(P0,P1,A,R,N,arahPanel,draw);crotasi = [P0];
drotasi = [ 0 0 0];daxial = [0 0 0];[offset]=
geotransformation(offset,crotasi,drotasi,daxial);[allOffset2,allPanel2]=arrangeData(
allOffset2,allPanel2,offset,panel,t); // antar kaki bawah horizontal ap

```

```

P0=[55 8 1.24]; P1=[55 8 20.24]; A =[0 360]; R = 0.5; N=[4 8]; arahPanel=-1;
[offset,panel,idx,k] = meshPipeOffset(P0,P1,A,R,N,arahPanel,draw);crotasi = [P0];
drotasi = [ 1 0.08 0];daxial = [0 0 0];[offset]=
geotransformation(offset,crotasi,drotasi,daxial);[allOffset2,allPanel2]=arrangeData(
allOffset2,allPanel2,offset,panel,t); // diagonal fp

```

```

P0=[55 -8 1.24]; P1=[55 -8 20.24]; A =[0 360]; R = 0.5; N=[4 8]; arahPanel=-1;
[offset,panel,idx,k] = meshPipeOffset(P0,P1,A,R,N,arahPanel,draw);crotasi = [P0];
drotasi = [ -1 0.08 0];daxial = [0 0 0];[offset]=
geotransformation(offset,crotasi,drotasi,daxial);[allOffset2,allPanel2]=arrangeData(
allOffset2,allPanel2,offset,panel,t); // diagonal fp

```

```

P0=[28 -10.36 1.24]; P1=[28 -10.36 27]; A =[0 360]; R = 0.5; N=[4 8];
arahPanel=-1;
[offset,panel,idx,k] = meshPipeOffset(P0,P1,A,R,N,arahPanel,draw);crotasi = [P0];
drotasi = [ -0.97 0.08 0];daxial = [0 0 0];[offset]=
geotransformation(offset,crotasi,drotasi,daxial);[allOffset2,allPanel2]=arrangeData(
allOffset2,allPanel2,offset,panel,t); // diagonal mid

```

```

P0=[28 10.36 1.24]; P1=[28 10.36 27]; A =[0 360]; R = 0.5; N=[4 8]; arahPanel=-
1;
[offset,panel,idx,k] = meshPipeOffset(P0,P1,A,R,N,arahPanel,draw);crotasi = [P0];
drotasi = [ 0.97 0.08 0];daxial = [0 0 0];[offset]=
geotransformation(offset,crotasi,drotasi,daxial);[allOffset2,allPanel2]=arrangeData(
allOffset2,allPanel2,offset,panel,t); // diagonal mid

```

```

P0=[3 12 1.24]; P1=[3 12 31.5]; A =[0 360]; R = 0.5; N=[4 8]; arahPanel=-1;
[offset,panel,idx,k] = meshPipeOffset(P0,P1,A,R,N,arahPanel,draw);crotasi = [P0];
drotasi = [ 0.935 0.08 0];daxial = [0 0 0];[offset]=
geotransformation(offset,crotasi,drotasi,daxial);[allOffset2,allPanel2]=arrangeData(
allOffset2,allPanel2,offset,panel,t); // diagonal ap

```

```

P0=[3 -12 1.24]; P1=[3 -12 31.5]; A =[0 360]; R = 0.5; N=[4 8]; arahPanel=-1;
[offset,panel,idx,k] = meshPipeOffset(P0,P1,A,R,N,arahPanel,draw);crotasi = [P0];
drotasi = [ -0.935 0.08 0];daxial = [0 0 0];[offset]=
geotransformation(offset,crotasi,drotasi,daxial);[allOffset2,allPanel2]=arrangeData(
allOffset2,allPanel2,offset,panel,t); // diagonal ap

```

```

P0=[55 -8 1.24]; P1=[55 -8 30]; A =[0 360]; R = 0.5; N=[4 8]; arahPanel=-1;
[offset,panel,idx,k] = meshPipeOffset(P0,P1,A,R,N,arahPanel,draw);crotasi = [P0];
drotasi = [ 0 -1.08 0.08];daxial = [0 0 0];[offset]=
geotransformation(offset,crotasi,drotasi,daxial);[allOffset2,allPanel2]=arrangeData(
allOffset2,allPanel2,offset,panel,t); // diagonal kiri fp mid

```

```

P0=[59 10.36 1.24]; P1=[28 10.26 1.24]; A =[0 360]; R = 0.5; N=[4 8];
arahPanel=-1;
[offset,panel,idx,k] = meshPipeOffset(P0,P1,A,R,N,arahPanel,draw);crotasi = [P1];
drotasi = [ 0 -0.35 -0.08];daxial = [0 0 0];[offset]=
geotransformation(offset,crotasi,drotasi,daxial);[allOffset2,allPanel2]=arrangeData(
allOffset2,allPanel2,offset,panel,t); // diagonal kanan fp mid

```

```
P0=[28 10.36 1.24]; P1=[-4.5 10.36 1.24]; A =[0 360]; R = 0.5; N=[4 8];
arahPanel=-1;
[offset,panel,idx,k] = meshPipeOffset(P0,P1,A,R,N,arahPanel,draw);crotasi = [P0];
drotasi = [ 0 0 0.73];daxial = [0 0 0];[offset]=
geotransformation(offset,crotasi,drotasi,daxial);[allOffset2,allPanel2]=arrangeData(
allOffset2,allPanel2,offset,panel,t); // diagonal bawah mid ap
endfunction
```

Lampiran 3. Fungsi Program buoyancy()

```

function [FF, MM, CC]=bouyancy(allOffset, allPanel, Cr, rho, pilihan)
//[FF,MM,CC,C,ac,hf]
//rho:[rho_air rho_pelat] // rho air = 1.025 T/m3 ; rho pelat = 8.75 T/m3
// Cr adalah titik referensi; Cr = [0 0 0]

m=size(allPanel,1);
n=size(allOffset,1);
allOffset(:,2:4)= allOffset(:,2:4);
P1=allOffset(allPanel(:,2),2:4); P2=allOffset(allPanel(:,3),2:4);
P3=allOffset(allPanel(:,4),2:4); P4=allOffset(allPanel(:,5),2:4);
t=allPanel(:,7); // t adalah ketebalan tiap panel
ac=cross(P2'-P1',P4'-P1'); // ac adalah luasan tiap panel disetiap sumbu
hf = (P1+P2+P3+P4)/4;
C = hf - repmat(Cr,m,1); // C adalah titik pusat panel

switch pilihan
case 'b'
check = hf(:,3)>=0;
for(i=1:m),
if(check(i)),
[acc,hfc] = interpolasi(i)
ac(i,:) = [acc];
hf(i,:) = [hfc];
end
Fb(i,:) = rho(1) * ac(i,:) * hf(i,3);
Mb(i,:) = ac(i,3) * C(i,:);
end
CC = sum(Mb,1)./sum(repmat(ac(:,3),1,3),1); // ac(:,3) sudah pasti minus
FF = sum(Fb,1);
MM = sum(Mb,1);
case 'w'
//perhitungan berat//

```

```

//perhitungan berat//
for(i=1:m),
    Fw(i,1) = t(i) * sqrt(sum(ac(i,:).^2)) * rho(2);
    Mw(i,:) = Fw(i,1) * C(i,:);
end
CC = sum(Mw,1)./sum(Fw);// titik berat terhadap x,y,z
FF = sum(Fw,1);
MM = sum(Mw,1);
case 'a'
//perhitungan bouyancy//
check = hf(:,3)>=0;
acw = ac
for(i=1:m),
    if(check(i)),
        ac(i,:) = [0 0 0];
    end
    Fb(i,:) = rho(1) * ac(i,:) * hf(i,3);
    Mb(i,:) = ac(i,3) * C(i,:);
end
Cb = sum(Mb,1)./sum(repmat(ac(:,3),1,3),1);
//perhitungan berat//
for(i=1:m),
    Fw(i,1) = t(i) * sqrt(sum(acw(i,:).^2)) * rho(2);
    Mw(i,:) = Fw(i,1) * C(i,:);
end
Cg = sum(Mw,1)./sum(Fw);// titik berat terhadap x,y,z
CC = [Cb; Cg];
FF = [sum(Fb,1);0 0 sum(Fw,1)];    MM = [sum(Mb,1); sum(Mw,1)];
end
endfunction

```

Lampiran 4. Fungsi Program meshPipeOffset()

```
function [offset, panel, idx, k]=meshPipeOffset(P0, P1, A, R, N, arahPanel, draw)
    A = A*pi/180;
    dA = A(1):(A(2)-A(1))/N(2):A(2);
    if(P1(1)~=P0(1))
        dx = (P1(1)-P0(1))/N(1);
        x = P0(1):dx:P1(1);
        y = P0(2) - R * cos(dA);
        z = P0(3) + R * sin(dA);
        m = length(x);           // jumlah data slice
        q = length(y);           // jumlah offset dalam slice
    elseif(P1(2)~=P0(2))
        dy = (P1(2)-P0(2))/N(1);
        y = P0(2):dy:P1(2);
        z = P0(3) + R * sin(dA);
        x = P0(1) - R * cos(dA);
        m = length(y);           // jumlah data slice
        q = length(z);           // jumlah offset dalam slice
    elseif(P1(3)~=P0(3))
        dz = (P1(3)-P0(3))/N(1);
        z = P0(3):dz:P1(3);
        x = R * cos(dA) + P0(1);
        y = R * sin(dA) + P0(2);
        m = length(z);           // jumlah data slice
        q = length(y);           // jumlah offset dalam slice
    end
    offset = [];
    panel = [];
    n = 1;
    k=1;
```

```

idx=0;
idxf=idx;
for i=1:m           // jumlah slice
    idx = idx(n)+1:idx(n)+q;
    // [offset] = offsetArrange(P0,P1,offset,idx,x,y,z,i);
    if(P1(3)~=P0(3))
        offset = [offset; idx' x' y' ones(q,1)*z(i)];
    elseif (P1(1)~=P0(1))
        offset = [offset; idx' ones(q,1)*x(i) y' z'];
    else
        offset = [offset; idx' x' ones(q,1)*y(i) z'];
    end
    idx = idx(q);
    p = idx-q;
    if(i > 1)
        for j = 1:q-1;   // jumlah panel
            k1 = p+1; k2 = k1+1; k3 = k2-q; k4 = k3-1;
            if (arahPanel== -1)
                // panel = [panel; k k1 k2 k3 k4 k1];
                panel = [panel; k k2 k1 k4 k3 k2];
            else
                panel = [panel; k k1 k2 k3 k4 k1]
            end
            k=k+1;
            p = k1;
        end
    end
    n = length(idx);   // jumlah nomor indekx
end
idx=idx+1;
if (draw==1) then
end
endfunction

```


Lampiran 5. Fungsi Program geotransformation()

```
function [offset]=geotransformation(offset, crotasi, drotasi, daxial)
// daxial ==> [dx dy dz] --> translasi pada sumbu ordinat
// drotasi ==> [Rx Ry Rz] --> Rotasi pada sumbu ordinat
alpha = drotasi(1); // Rx
betha = drotasi(2); // Ry
cetha = drotasi(3); // Rz
T1 = [1      0      0;
      0 cos(alpha) -sin(alpha);
      0 sin(alpha) cos(alpha)];

T2 = [cos(betha) 0 sin(betha);
      0      1      0 ;
      -sin(betha) 0 cos(betha)];

T3 = [cos(cetha) -sin(cetha) 0;
      sin(cetha) cos(cetha) 0;
      0      0      1];
T = T3 * T2 * T1;
for i=1:size(offset,1)
    temp = offset(i,2:4)' - crotasi'; // ambil ofset i simpan sementara pada
temp
    temp = T * temp; // transformasi temp menjadi nilai temp yang
baru
    offset(i,:) = [i temp'+daxial+crotasi]; // simpan nilai temp ke offset i
end
endfunction
```

Lampiran 6. Fungsi Program plotMesh()

```
function plotMesh(offset, panel)
    //h=figure(1);
    //xyz=[];
    cX=[]; cF=[];
    for i=1:size(panel,1)
        xyz=offset(panel(i,2:6)',2:4);
        plot3d(xyz(:,1),xyz(:,2),xyz(:,3));
        cX = mean(xyz(1:4,:),1);
        cA = cX-offset(panel(i,2),2:4);
        cB = cX-offset(panel(i,3),2:4);
        cN = cross(cA,cB);
        cF = cN./(sqrt(sum(cN'^2)))+cX;
        //xarrows([cX(1) cF(1)],[cX(2) cF(2)],[cX(3) cF(3)],-1,1);
    end
endfunction
```

Lampiran 7. Fungsi Program arrangeData()

```
function [allOffset, allPanel]=arrangeData(allOffset, allPanel, offset, panel, t)
    if (isempty(allOffset)) then
        idx=0; k=0;
    else
        idx = allOffset(size(allOffset,1),1);
        k = allPanel(size(allPanel,1),1);
    end
    te = ones(size(panel,1),1)*t
    offset(:,1) = offset(:,1)+idx;
    panel(:,1) = panel(:,1)+k;
    panel(:,2:6) = panel(:,2:6)+idx;
    allOffset = [allOffset; offset];
    allPanel = [allPanel; panel te];
endfunction
```

Lampiran 8. Fungsi Program interpolasi()

```
function [acc, hfc]=interpolasi(i)
    t1=allOffset(allPanel(i,2),2:4); t2=allOffset(allPanel(i,3),2:4);
    t3=allOffset(allPanel(i,4),2:4); t4=allOffset(allPanel(i,5),2:4);
    if t1(1,3)<0 || t2(1,3)<0 || t3(1,3)<0 ||t4(1,3)<0 then
        x = 0;  x1 = t1(1,3);  x2 = t2(1,3);  y1 = t1(1,2);  y2 = t2(1,2)
        if x2==x1 then
            y = (y1+y2)/2
        else
            y = (((x - x1)/(x2 - x1))*(y2 - y1))+y1
        end
        if t1(1,1)==t2(1,1) then
            if t1(1,3)<t2(1,3) then
                t2(1,2) = y
                t2(1,3) = 0
                t3(1,2) = y
                t3(1,3) = 0
            else
                t1(1,2) = y
                t1(1,3) = 0
                t4(1,2) = y
                t4(1,3) = 0
            end
        end
    end
    acc =cross(t2'-t1',t4'-t1')
    hfc = (t1+t2+t3+t4)/4
else
    acc = [0 0 0]
    hfc = (t1+t2+t3+t4)/4
end
endfunction
```

Lampiran 9. Fungsi Program run()

```

function [allOffset3, result, db]=run(r)
    // n = step
    // Lj = slide jacket
    dl = 1
    while dl > 0.001 || dl <-0.001
        [data,allOffset,allPanel]=genof(x)
        [allOffset2,allPanel2] = generateMeshJacket(1)
        allOffset2(:,4) = allOffset2(:,4)-1.24+max(allOffset(:,4))+1
        Cr = [0 0 0];rho = [1.025 7.85]
        [FFb,MMb,CCb]=bouyancy(allOffset,allPanel,Cr,rho,'b')

        crotasi = [CCb]; drotasi = [0 0 0]; daxial=[stp(step,1) 0 0]
        [allOffset2]= geotransformation(allOffset2,crotasi,drotasi,daxial)
        //deg = 3 // derajat
        rad = -deg*(%pi/180)
        crotasi = [CCb]; drotasi = [0 rad 0]; daxial=[0 0 0]
        [allOffset]= geotransformation(allOffset,crotasi,drotasi,daxial)
        [allOffset2]= geotransformation(allOffset2,crotasi,drotasi,daxial)
        [FFj,MMj,CCj]=bouyancy(allOffset2,allPanel2,Cr,rho,'w')
        Mj = (CCb(1,1)-CCj(1,1))*FFj
        for i = 1:5
            [data.tank(i).offset]= geotransformation(data.tank(i).offset,crotasi,drotasi,daxial)
            min2 = min(data.tank(i).offset(:,4))
            data.tank(i).offset = [data.tank(i).offset(:,1:3) data.tank(i).offset(:,4)-min2]
        end
        if CCb(1,1)<=(data.tank(1).offset(1,2)+data.tank(1).offset(334,2))/2 &&
        CCb(1,1)>data.tank(1).offset(1,2) then
            i = 1
        elseif CCb(1,1)<= (data.tank(2).offset(353,2)+data.tank(2).offset(1,2))/2 &&
        CCb(1,1)>(data.tank(1).offset(1,2)+data.tank(1).offset(334,2))/2 then
    
```

```

i = 2
elseif Ccb(1,1)<=(data.tank(3).offset(353,2)+data.tank(3).offset(1,2))/2 &&
Ccb(1,1)>(data.tank(2).offset(353,2)+data.tank(2).offset(1,2))/2 then
    i = 3
elseif Ccb(1,1)<=(data.tank(4).offset(353,2)+data.tank(4).offset(1,2))/2 &&
Ccb(1,1)>(data.tank(3).offset(353,2)+data.tank(3).offset(1,2))/2 then
    i = 4
elseif Ccb(1,1) <=(data.tank(5).offset(353,2)+data.tank(5).offset(1,2))/2 &&
Ccb(1,1)>(data.tank(4).offset(353,2)+data.tank(4).offset(1,2))/2 then
    i = 5
end

W = FFj
db = 2 // level di tangki 1 2
for j = 2:i-1
    if db<=0 then continue end
    oft = [data.tank(j).offset(:,1:3) data.tank(j).offset(:,4)-db]
    [FFt,MMt,CCt]=bouyancy(oft,data.tank(j).panel,Cr,rho,'b')
    Mj = Mj + (Ccb(1,1)-CCt(1,1))*FFt(1,3)
    W = W +FFt(1,3)
end
Mf = Mj/length(i:5)
ZM = []
ZM2 = Mj
result= []
//W = FFj ini yang kasi pusing
for k = i:5
    Mt = 0
    dt = 0
    while Mf-Mt>0.01

```

```

dt = dt + 0.01
oft = [data.tank(k).offset(:,1:3) data.tank(k).offset(:,4)-dt]
[FFt,MMt,CCt]=bouyancy(oft,data.tank(k).panel,Cr,rho,'b')
Mt = (CCb(1,1)-CCt(1,1))*-FFt(1,3)
if FFt(1,3)/1.025>maxvol(k,1)-100 then break end
disp([Mf-Mt])
end
dt = dt - 0.01
oft = [data.tank(k).offset(:,1:3) data.tank(k).offset(:,4)-dt]
[FFt,MMt,CCt]=bouyancy(oft,data.tank(k).panel,Cr,rho,'b')
Mt = (CCb(1,1)-CCt(1,1))*-FFt(1,3)
while Mf-Mt>0.001
dt = dt + 0.001
oft = [data.tank(k).offset(:,1:3) data.tank(k).offset(:,4)-dt]
[FFt,MMt,CCt]=bouyancy(oft,data.tank(k).panel,Cr,rho,'b')
Mt = (CCb(1,1)-CCt(1,1))*-FFt(1,3)
if FFt(1,3)/1.025>maxvol(k,1)-100 then break end
disp([Mf-Mt])
end
dt = dt - 0.001
oft = [data.tank(k).offset(:,1:3) data.tank(k).offset(:,4)-dt]
[FFt,MMt,CCt]=bouyancy(oft,data.tank(k).panel,Cr,rho,'b')
Mt = (CCb(1,1)-CCt(1,1))*-FFt(1,3)
while Mf-Mt>0.00001
dt = dt + 0.0001
oft = [data.tank(k).offset(:,1:3) data.tank(k).offset(:,4)-dt]
[FFt,MMt,CCt]=bouyancy(oft,data.tank(k).panel,Cr,rho,'b')
Mt = (CCb(1,1)-CCt(1,1))*-FFt(1,3)
if FFt(1,3)/1.025>maxvol(k,1)-100 then break end
disp([Mf-Mt])

```

```

end

    draft2 = draft2 - 0.1
    oft = [allOffset(:,1:3) allOffset(:,4)+draft-draft2]
    [FFb,MMb,CCb]=bouyancy(oft,allPanel,Cr,rho,'b')
    disp(W - FFb(1,3))
while W - FFb(1,3) > 1
    draft2 = draft2 + 0.01
    oft = [allOffset(:,1:3) allOffset(:,4)+draft-draft2]
    [FFb,MMb,CCb]=bouyancy(oft,allPanel,Cr,rho,'b')
    disp(W - FFb(1,3))
end

result = [result draft2 ZM2]
dl = ZM2/W
del1 = draft
del2 = draft2
draft = draft2

result = [result dl]
//messagebox(['level tank3 = '; 'level tank 2 = '; 'level tank1 = '; 'displacement = ' ;
'draft = '; 'sigma momen = '; 'koreksi jarak CG = '] + [string(result)])
end
//clf
del = del1-del2

    crotasi = [CCb]; drotasi = [0 rad 0]; daxial=[0 0 del]
    [allOffset3]= geotransformation(allOffset,crotasi,drotasi,daxial)
    [allOffset4]= geotransformation(allOffset2,crotasi,drotasi,daxial)
//plotMesh(allOffset3,allPanel)
if step == 1 || step==n+1 then
    plotMesh(allOffset4,allPanel2)
end

endfunction

```


Lampiran 10. Fungsi Program plotlevel()

```
function plotlevel(result, db)
    figure(); global deg
    newaxes();handles.ax_1.margins = [ 0 0 0 0];handles.ax_1.axes_bounds =
    [0.5,0.15,0.5,0.7];handles.ax_1.auto_scale = "on";handles.ax_1.data_bounds = [-40 -
    20 -5; 80 20 30];handles.ax_1.line_mode = "on";handles.ax_1.line_style =
    1;handles.ax_1.mark_size = 5
    for IRL = 1:size(Result,1)
        warna =["r" "g" "b" "c" "m" "y" "k"]
        result = Result(IRL,:)
        x = read('offsetbarga2.txt',380,4)
        draft = 0
        [data,allOffset,allPanel]=genof(x)
        p=[]
        for g = 1: length(data.station.offset)
            p = [p; data.station(g).offset(1,:)]
        end
        p =
        [p;data.station(38).offset(10,:);data.station(1).offset(10,:);data.station(1).offset(1,:)]
        p1 = [-3 3;70 3]
        s1 = [data.station(7).offset(1,:);data.station(7).offset(10,:)]
        s2 = [data.station(12).offset(1,:);data.station(12).offset(10,:)]
        s3 = [data.station(18).offset(1,:);data.station(18).offset(10,:)]
        s4 = [data.station(24).offset(1,:);data.station(24).offset(10,:)]
        s5 = [data.station(30).offset(1,:);data.station(30).offset(10,:)]
        s6 = [data.station(35).offset(1,:);data.station(35).offset(10,:)]
        plot2d(p(:,1),p(:,3),frameflag=3)
        title("Water Level at Step " + string(IRL-1) , "fontsize",3)
        plot(s1(:,1),s1(:,3),"k")
        plot(s2(:,1),s2(:,3),"k")
        plot(s3(:,1),s3(:,3),"k")
        nlot(s4(:,1) s4(:,3) "k")
```

```

plot(s4(:,1),s4(:,3),"k")
plot(s5(:,1),s5(:,3),"k")
plot(s6(:,1),s6(:,3),"k")
sv=[]
dt = result(1,3)//tank no.5
deg = 2//strtod(handles.ed_2.string)
rad = -deg*(%pi/180)
sa = cos(rad)*dt
de = tan(rad)*(data.station(35).offset(1,1)-data.station(30).offset(1,1))
lk = sa-de
p5=[data.station(30).offset(1,1) lk; data.station(35).offset(1,1) sa]
plot(p5(:,1),p5(:,2),(warna(IRL)))
sv(5,:)= [lk sa]

dt = result(1,2)//tank no.4
rad = -deg*(%pi/180)
sa = cos(rad)*dt
de = tan(rad)*(data.station(30).offset(1,1)-data.station(24).offset(1,1))
lk = sa-de
p4=[data.station(24).offset(1,1) lk; data.station(30).offset(1,1) sa]
plot(p4(:,1),p4(:,2),(warna(IRL)))
sv(4,:)= [lk sa]

dt = result(1,1)//tank no.3
rad = -deg*(%pi/180)
sa = cos(rad)*dt
de = tan(rad)*(data.station(24).offset(1,1)-data.station(18).offset(1,1))
lk = sa-de
p3=[data.station(18).offset(1,1) lk; data.station(24).offset(1,1) sa]
plot(p3(:,1),p3(:,2),(warna(IRL)))

```

```

p3 =[data.station(18).offset(1,1) lk; data.station(24).offset(1,1) sa]
plot(p3(:,1),p3(:,2),(warna(IRL)))
sv(3,:)= [lk sa]

dt = db
rad = -deg*(%pi/180)
sa = cos(rad)*dt
de = tan(rad)*(data.station(18).offset(1,1)-data.station(12).offset(1,1))
lk = sa-de
p2 =[data.station(12).offset(1,1) lk; data.station(18).offset(1,1) sa]
plot(p2(:,1),p2(:,2),(warna(IRL)))
sv(2,:)= [lk sa]

dt = db
rad = -deg*(%pi/180)
sa = cos(rad)*dt
de = tan(rad)*(data.station(12).offset(1,1)-data.station(7).offset(1,1))
lk = sa-de
p1 =[data.station(7).offset(1,1) lk; data.station(12).offset(1,1) sa]
//plot(p1(:,1),p1(:,2))
sv(1,:)= [0 0]
disp(IRL-1)
disp(sv)
end
endfunction

```

Lampiran 11.Fungsi Program postLaunching()

```

function postLaunching(inpost)
    global deg
    Wpost = sum(inpost)
    draft = 0
    Wb = 0
    dpost = 0
    [data,allOffset,allPanel]=genof(x)
    while Wpost-Wb >100
        crotasi = [0 0 0]; drotasi = [0 0 0]; daxial=[0 0 -0.1]
        [allOffset]= geotransformation(allOffset,crotasi,drotasi,daxial)
        Cr = [0 0 0];rho = [1.025 7.85]
        [FFb,MMb,CCb]=bouyancy(allOffset,allPanel,Cr,rho,'b')
        dpost = dpost-0.1
        Wb = FFb(1,3)
        disp(Wpost-Wb)
    end
    crotasi = [0 0 0]; drotasi = [0 0 0]; daxial=[0 0 0.1]
    [allOffset]= geotransformation(allOffset,crotasi,drotasi,daxial)
    Cr = [0 0 0];rho = [1.025 7.85]
    [FFb,MMb,CCb]=bouyancy(allOffset,allPanel,Cr,rho,'b')
    dpost = dpost-0.01
    Wb = FFb(1,3)
    disp(Wpost-Wb)
    while Wpost-Wb >10
        crotasi = [0 0 0]; drotasi = [0 0 0]; daxial=[0 0 -0.01]
        [allOffset]= geotransformation(allOffset,crotasi,drotasi,daxial)
        Cr = [0 0 0];rho = [1.025 7.85]
        [FFb,MMb,CCb]=bouyancy(allOffset,allPanel,Cr,rho,'b')
        dpost = dpost-0.01
        Wb = FFb(1,3)
        disp(Wpost-Wb)
    end
    crotasi = [0 0 0]; drotasi = [0 0 0]; daxial=[0 0 0.01]
    [allOffset]= geotransformation(allOffset,crotasi,drotasi,daxial)
    Cr = [0 0 0];rho = [1.025 7.85]
    [FFb,MMb,CCb]=bouyancy(allOffset,allPanel,Cr,rho,'b')
    dpost = dpost-0.01
    Wb = FFb(1,3)
    disp(Wpost-Wb)
    while Wpost-Wb >0
        crotasi = [0 0 0]; drotasi = [0 0 0]; daxial=[0 0 -0.001]
        [allOffset]= geotransformation(allOffset,crotasi,drotasi,daxial)
        Cr = [0 0 0];rho = [1.025 7.85]
        [FFb,MMb,CCb]=bouyancy(allOffset,allPanel,Cr,rho,'b')
        dpost = dpost-0.01
        Wb = FFb(1,3)
        disp(Wpost-Wb)
    end
    disp('alihalihalihalihalihalihalihalihalihalihalihalihalihalihalihalihalihalih')
    disp(dpost)
    ZMP = (CCb(1,1)-15)*inpost(1,1)+(CCb(1,1)-25.828)*inpost(2,1)+(CCb(1,1)-36.898)*inpost(3,1)+(CCb(1,1)-
    47.967)*inpost(4,1)+(CCb(1,1)-59.036)*inpost(5,1)
    dg = 0
    while ZMP<0
        crotasi = [CCb]; drotasi = [0 0.1*%pi/180 0]; daxial=[0 0 0]
        [allOffset]= geotransformation(allOffset,crotasi,drotasi,daxial)
        [FFb,MMb,CCb]=bouyancy(allOffset,allPanel,Cr,rho,'b')
        ZMP = (CCb(1,1)-15*cos(dg)*inpost(1,1)+(CCb(1,1)-25.828*cos(dg))*inpost(2,1)+(CCb(1,1)-
        36.898*cos(dg))*inpost(3,1)+(CCb(1,1)-47.967*cos(dg))*inpost(4,1)+(CCb(1,1)-59.036*cos(dg))*inpost(5,1))
        dg = dg +(0.1*%pi/180)
        disp(ZMP)
    end
    Wb = FFb(1,3)
    dz = 0

```

```

dz = 0
while Wpost-Wb >1 ||Wpost-Wb <-1
    if Wpost-Wb >5 then dz = -0.0001 end
    if Wpost-Wb <5 then dz = 0.0001 end
    crotasi = [0 0 0]; drotasi = [0 0 0]; daxial=[0 0 dz]/-0.001]
    [allOffset]= geotransformation(allOffset,crotasi,drotasi,daxial)
    Cr = [0 0 0];rho = [1.025 7.85]
    [FFb,MMb,CCb]=bouyancy(allOffset,allPanel,Cr,rho,'b')
    dpost = dpost-0.01
    Wb = FFb(1,3)
    disp(Wpost-Wb)
end
dg2 = dg*180/%pi
disp(dg2)
disp(Wpost-FFb(1,3))
APp = allOffset(124,4)
FPp = allOffset(656,4)
disp([APp FPp])
figure()
newaxes()
plotMesh(allOffset,allPanel)
a = [-40 -25 0; -40 25 0; 80 25 0; 80 -25 0; -40 -25 0]
plot3d(a(:,1),a(:,2),a(:,3))
endfunction

```

Lampiran 12. Program *Graphical User Interface* (GUI)

```
//////////
f=figure('figure_position',[0,0],'figure_size',[1280,720],'auto_resize','on','background',[-2],'figure_name','Skripsi
ver.1','dockable','off','infobar_visible','off','toolbar_visible','off','menubar_visible','off','default_axes','on','visible','off');
//////////
handles.dummy = 0;
handles.img4=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-
1],'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[12],'FontUnits','points','FontWeight','normal','Foreg
roundColor',[-1,-1,-
1],'HorizontalAlignment','left','ListboxTop',[],'Max',[1],'Min',[0],'Position',[0.45,0,0.6,0.12],'Relief','default','SliderStep',[
0.01,0.1],'String','E:\ZKRIPZI\Skripsi\ROV SciLab-20220108T071654Z-001\ROV
SciLab\backgroundtop.jpg','Style','image','Value',[1,1,0,0,0],'VerticalAlignment','middle','Visible','on','Tag','img1','Callba
ck','img1_callback(handles)')
handles.img3=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-
1],'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[12],'FontUnits','points','FontWeight','normal','Foreg
roundColor',[-1,-1,-
1],'HorizontalAlignment','left','ListboxTop',[],'Max',[1],'Min',[0],'Position',[0.45,0.86,0.6,0.14],'Relief','default','SliderSte
p',[0.01,0.1],'String','E:\ZKRIPZI\Skripsi\ROV SciLab-20220108T071654Z-001\ROV
SciLab\backgroundtop.jpg','Style','image','Value',[1,1,0,0,0],'VerticalAlignment','middle','Visible','on','Tag','img1','Callba
ck','img1_callback(handles)')
handles.img2=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-
1],'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[12],'FontUnits','points','FontWeight','normal','Foreg
roundColor',[-1,-1,-
1],'HorizontalAlignment','left','ListboxTop',[],'Max',[1],'Min',[0],'Position',[0,0,0.47,1],'Relief','default','SliderStep',[0.01,
0.1],'String','E:\ZKRIPZI\Skripsi\ROV SciLab-20220108T071654Z-001\ROV
SciLab\backgroundleft.jpg','Style','image','Value',[1,1,0,0,0],'VerticalAlignment','middle','Visible','on','Tag','img1','Callba
ck','img1_callback(handles)')
handles.imgline=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-
1],'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[12],'FontUnits','points','FontWeight','normal','Foreg
roundColor',[-1,-1,-
1],'HorizontalAlignment','left','ListboxTop',[],'Max',[1],'Min',[0],'Position',[0.47,0,0.006,1],'Relief','default','SliderStep',[
0.01,0.1],'String','E:\ZKRIPZI\Skripsi\ROV SciLab-20220108T071654Z-001\ROV
SciLab\line.jpg','Style','image','Value',[1,1,0,0,0],'VerticalAlignment','middle','Visible','on','Tag','img1','Callback','img1_c
allback(handles)')
handles.imgline1=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-
1],'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[12],'FontUnits','points','FontWeight','normal','Foreg
roundColor',[-1,-1,-
1],'HorizontalAlignment','left','ListboxTop',[],'Max',[1],'Min',[0],'Position',[0.47,0.86,1,0.01],'Relief','default','SliderStep',
[0.01,0.1],'String','E:\ZKRIPZI\Skripsi\ROV SciLab-20220108T071654Z-001\ROV
SciLab\line1.jpg','Style','image','Value',[1,1,0,0,0],'VerticalAlignment','middle','Visible','on','Tag','img1','Callback','img1
_callback(handles)')
handles.imgline2=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-
1],'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[12],'FontUnits','points','FontWeight','normal','Foreg
roundColor',[-1,-1,-
1],'HorizontalAlignment','left','ListboxTop',[],'Max',[1],'Min',[0],'Position',[0.47,0.12,1,0.01],'Relief','default','SliderStep',
[0.01,0.1],'String','E:\ZKRIPZI\Skripsi\ROV SciLab-20220108T071654Z-001\ROV
SciLab\line1.jpg','Style','image','Value',[1,1,0,0,0],'VerticalAlignment','middle','Visible','on','Tag','img1','Callback','img1
_callback(handles)')
handles.frame1_input=uicontrol(f,'unit','normalized','BackgroundColor',[1,1,1],'Enable','on','FontAngle','normal','FontNa
me','Tahoma','FontSize',[12],'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-
1],'HorizontalAlignment','left','ListboxTop',[],'Max',[1],'Min',[0],'Position',[0.02,0.7,0.2,0.2],'Relief','default','SliderStep',
[0.01,0.1],'String','UnName2','Style','frame','Value',[0],'VerticalAlignment','middle','Visible','on','Tag','frame1_input','Cal
lback','','constraints','createConstraints('gridbag',[1,2,1,1],[0,0],'both','center'),'border','createBorder('titled',
'createBorder('line','Gray',1.5),'_('Input'),'center','top','createBorderFont('Times New Roman',12,'normal'))')
handles.frame2_output=uicontrol(f,'unit','normalized','BackgroundColor',[1,1,1],'Enable','on','FontAngle','normal','FontN
ame','Tahoma','FontSize',[12],'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-
1],'HorizontalAlignment','left','ListboxTop',[],'Max',[1],'Min',[0],'Position',[0.02,0.03,0.2,0.63],'Relief','default','SliderSte
p',[0.01,0.1],'String','UnName2','Style','frame','Value',[0],'VerticalAlignment','middle','Visible','on','Tag','frame2_output',
'Callback','','constraints','createConstraints('gridbag',[1,2,1,1],[0,0],'both','center'),'border','createBorder('titled',
'createBorder('line','Gray',1.5),'_('Output'),'center','top','createBorderFont('Times New Roman',12,'normal'))')
handles.frame3_jacket=uicontrol(f,'unit','normalized','BackgroundColor',[1,1,1],'Enable','on','FontAngle','normal','FontN
ame','Tahoma','FontSize',[12],'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-
1],'HorizontalAlignment','left','ListboxTop',[],'Max',[1],'Min',[0],'Position',[0.23,0.7,0.2,0.2],'Relief','default','SliderStep',
[0.01,0.1],'String','UnName2','Style','frame','Value',[0],'VerticalAlignment','middle','Visible','on','Tag','frame1_input','Cal
lback','','constraints','createConstraints('gridbag',[1,2,1,1],[0,0],'both','center'),'border','createBorder('titled',
```

```

createBorder('line', 'Gray', 1.5),_(('Jacket'), 'center', 'top',createBorderFont ("Times New Roman",12,'normal')))
handles.frame4_table=uicontrol(f,'unit','normalized','BackgroundColor',[1,1,1],'Enable','on','FontAngle','normal','FontName',
'Tahoma','FontSize',[12],'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-
1],'HorizontalAlignment','left','ListboxTop',[0],'Max',[1],'Min',[0],'Position',[0.23,0.25,0.24,0.41],'Relief','default','SliderSt
ep',[0.01,0.1],'String','UnName2','Style','frame','Value',[0],'VerticalAlignment','middle','Visible','on','Tag','frame2_output
','Callback','', 'constraints',createConstraints('gridbag', [1, 2, 1, 1], [0, 0], 'both', 'center'),'border',createBorder('titled',
createBorder('line', 'Gray', 1.5),_(('Recap'), 'center', 'top',createBorderFont ("Times New Roman",12,'normal')))
handles.pb_plot=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-
1],'Enable','on','FontAngle','normal','FontName','Times New
Roman','FontSize',[20],'FontUnits','points','FontWeight','bold','ForegroundColor',[-1,-1,-
1],'HorizontalAlignment','center','ListboxTop',[0],'Max',[1],'Min',[0],'Position',[0.24,0.13,0.1778846,0.0863636],'Relief','d
efault','SliderStep',[0.01,0.1],'String','PLOT','Style','pushbutton','Value',[0],'VerticalAlignment','middle','Visible','on','Tag
','pb_plot','Callback','pb_plot_callback(handles)')
handles.ax_1=newaxes();handles.ax_1.margins = [ 0 0 0 0];handles.ax_1.axes_bounds =
[0.5,0.15,0.5,0.7];handles.ax_1.auto_scale = "off";handles.ax_1.data_bounds = [-40 -20 -5; 80 20
30];handles.ax_1.line_mode = "on";handles.ax_1.line_style = 1;handles.ax_1.mark_size = 5
handles.ed_1=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-
1],'Enable','on','FontAngle','normal','FontName','Times New
Roman','FontSize',[12],'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-
1],'HorizontalAlignment','left','ListboxTop',[0],'Max',[1],'Min',[0],'Position',[0.12,0.8,0.07,0.04],'Relief','default','SliderSte
p',[0.01,0.1],'String','','Style','edit','Value',[0],'VerticalAlignment','middle','Visible','on','Tag','ed_1','Callback','')
handles.ed_2=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-
1],'Enable','on','FontAngle','normal','FontName','Times New
Roman','FontSize',[12],'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-
1],'HorizontalAlignment','left','ListboxTop',[0],'Max',[1],'Min',[0],'Position',[0.12,0.73,0.07,0.04],'Relief','default','SliderSt
ep',[0.01,0.1],'String','','Style','edit','Value',[0],'VerticalAlignment','middle','Visible','on','Tag','ed_2','Callback','')
handles.ed_CBx=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-
1],'Enable','off','FontAngle','normal','FontName','Times New
Roman','FontSize',[12],'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-
1],'HorizontalAlignment','left','ListboxTop',[0],'Max',[1],'Min',[0],'Position',[0.12,0.58,0.07,0.04],'Relief','default','SliderSt
ep',[0.01,0.1],'String','','Style','edit','Value',[0],'VerticalAlignment','middle','Visible','on','Tag','ed_1','Callback','')
handles.ed_CBz=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-
1],'Enable','off','FontAngle','normal','FontName','Times New
Roman','FontSize',[12],'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-
1],'HorizontalAlignment','left','ListboxTop',[0],'Max',[1],'Min',[0],'Position',[0.12,0.48,0.07,0.04],'Relief','default','SliderSt
ep',[0.01,0.1],'String','','Style','edit','Value',[0],'VerticalAlignment','middle','Visible','on','Tag','ed_1','Callback','')
handles.ed_displacement=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-
1],'Enable','off','FontAngle','normal','FontName','Times New
Roman','FontSize',[12],'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-
1],'HorizontalAlignment','left','ListboxTop',[0],'Max',[1],'Min',[0],'Position',[0.12,0.41,0.07,0.04],'Relief','default','SliderSt
ep',[0.01,0.1],'String','','Style','edit','Value',[0],'VerticalAlignment','middle','Visible','on','Tag','ed_1','Callback','')
handles.ed_voltank1=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-
1],'Enable','off','FontAngle','normal','FontName','Times New
Roman','FontSize',[12],'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-
1],'HorizontalAlignment','left','ListboxTop',[0],'Max',[1],'Min',[0],'Position',[0.12,0.34,0.07,0.04],'Relief','default','SliderSt
ep',[0.01,0.1],'String','','Style','edit','Value',[0],'VerticalAlignment','middle','Visible','on','Tag','ed_1','Callback','')
handles.ed_voltank2=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-
1],'Enable','off','FontAngle','normal','FontName','Times New
Roman','FontSize',[12],'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-
1],'HorizontalAlignment','left','ListboxTop',[0],'Max',[1],'Min',[0],'Position',[0.12,0.27,0.07,0.04],'Relief','default','SliderSt
ep',[0.01,0.1],'String','','Style','edit','Value',[0],'VerticalAlignment','middle','Visible','on','Tag','ed_1','Callback','')
handles.ed_voltank3=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-
1],'Enable','off','FontAngle','normal','FontName','Times New
Roman','FontSize',[12],'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-
1],'HorizontalAlignment','left','ListboxTop',[0],'Max',[1],'Min',[0],'Position',[0.12,0.2,0.07,0.04],'Relief','default','SliderSte
p',[0.01,0.1],'String','','Style','edit','Value',[0],'VerticalAlignment','middle','Visible','on','Tag','ed_1','Callback','')
handles.ed_voltank4=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-
1],'Enable','off','FontAngle','normal','FontName','Times New
Roman','FontSize',[12],'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-

```

```

handles.ed_voltank4=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1],
'Enable','off','FontAngle','normal','FontName','Times New Roman','FontSize',[12],
'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1],
'HorizontalAlignment','left','ListboxTop',[],
'Max',[1],
'Min',[0],
'Position',[0.12,0.13,0.07,0.04],
'Relief','default','SliderStep',[0.01,0.1],
'String','','Style','edit','Value',[0],
'VerticalAlignment','middle','Visible','on','Tag','ed_1','Callback','')
handles.ed_voltank5=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1],
'Enable','off','FontAngle','normal','FontName','Times New Roman','FontSize',[12],
'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1],
'HorizontalAlignment','left','ListboxTop',[],
'Max',[1],
'Min',[0],
'Position',[0.12,0.06,0.07,0.04],
'Relief','default','SliderStep',[0.01,0.1],
'String','','Style','edit','Value',[0],
'VerticalAlignment','middle','Visible','on','Tag','ed_1','Callback','')
handles.ed_3=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1],
'Enable','on','FontAngle','normal','FontName','Times New Roman','FontSize',[12],
'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1],
'HorizontalAlignment','left','ListboxTop',[],
'Max',[1],
'Min',[0],
'Position',[0.32,0.8,0.07,0.04],
'Relief','default','SliderStep',[0.01,0.1],
'String','','Style','edit','Value',[0],
'VerticalAlignment','middle','Visible','on','Tag','ed_1','Callback','')
handles.ed_4=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1],
'Enable','on','FontAngle','normal','FontName','Times New Roman','FontSize',[12],
'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1],
'HorizontalAlignment','left','ListboxTop',[],
'Max',[1],
'Min',[0],
'Position',[0.32,0.73,0.07,0.04],
'Relief','default','SliderStep',[0.01,0.1],
'String','','Style','edit','Value',[0],
'VerticalAlignment','middle','Visible','on','Tag','ed_2','Callback','')
handles.pb_run=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1],
'Enable','on','FontAngle','normal','FontName','Times New Roman','FontSize',[20],
'FontUnits','points','FontWeight','bold','ForegroundColor',[-1,-1,-1],
'HorizontalAlignment','center','ListboxTop',[],
'Max',[1],
'Min',[0],
'Position',[0.24,0.03,0.1778846,0.0863636],
'Relief','default','SliderStep',[0.01,0.1],
'String','RUN','Style','pushbutton','Value',[0],
'VerticalAlignment','middle','Visible','on','Tag','pb_run','Callback','pb_run_callback(handles)')
handles.txt_draft=uicontrol(f,'unit','normalized','BackgroundColor',[1,1,1],
'Enable','on','FontAngle','normal','FontName','Times New Roman','FontSize',[12],
'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1],
'HorizontalAlignment','left','ListboxTop',[],
'Max',[1],
'Min',[0],
'Position',[0.045,0.8,0.04,0.0341477],
'Relief','default','SliderStep',[0.01,0.1],
'String','Draft','Style','text','Value',[0],
'VerticalAlignment','middle','Visible','on','Tag','txt_pidrudder','Callback','')
handles.txt_m=uicontrol(f,'unit','normalized','BackgroundColor',[1,1,1],
'Enable','on','FontAngle','normal','FontName','Times New Roman','FontSize',[12],
'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1],
'HorizontalAlignment','left','ListboxTop',[],
'Max',[1],
'Min',[0],
'Position',[0.19,0.81,0.01,0.02],
'Relief','default','SliderStep',[0.01,0.1],
'String','m','Style','text','Value',[0],
'VerticalAlignment','middle','Visible','on','Tag','txt_pidrudder','Callback','')
handles.txt_trim=uicontrol(f,'unit','normalized','BackgroundColor',[1,1,1],
'Enable','on','FontAngle','normal','FontName','Times New Roman','FontSize',[12],
'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1],
'HorizontalAlignment','left','ListboxTop',[],
'Max',[1],
'Min',[0],
'Position',[0.045,0.73,0.04,0.0341477],
'Relief','default','SliderStep',[0.01,0.1],
'String','Trim','Style','text','Value',[0],
'VerticalAlignment','middle','Visible','on','Tag','txt_pidrudder','Callback','')
handles.txt_deg=uicontrol(f,'unit','normalized','BackgroundColor',[1,1,1],
'Enable','on','FontAngle','normal','FontName','Times New Roman','FontSize',[12],
'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1],
'HorizontalAlignment','left','ListboxTop',[],
'Max',[1],
'Min',[0],
'Position',[0.19,0.74,0.016,0.021],
'Relief','default','SliderStep',[0.01,0.1],
'String','deg','Style','text','Value',[0],
'VerticalAlignment','middle','Visible','on','Tag','txt_pidrudder','Callback','')
handles.txt_CB=uicontrol(f,'unit','normalized','BackgroundColor',[1,1,1],
'Enable','on','FontAngle','normal','FontName','Times New Roman','FontSize',[12],
'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1],
'HorizontalAlignment','left','ListboxTop',[],
'Max',[1],
'Min',[0],
'Position',[0.045,0.58,0.02,0.03],
'Relief','default','SliderStep',[0.01,0.1],
'String','CB','Style','text','Value',[0],
'VerticalAlignment','middle','Visible','on','Tag','txt_pidrudder','Callback','')
handles.txt_CBx=uicontrol(f,'unit','normalized','BackgroundColor',[1,1,1],
'Enable','on','FontAngle','normal','FontName','Times New Roman','FontSize',[12],
'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1],
'HorizontalAlignment','left','ListboxTop',[],
'Max',[1],
'Min',[0],
'Position',[0.11,0.58,0.01,0.0341477],
'Relief','default','SliderStep',[0.01,0.1],
'String','X','Style','text','Value',[0],
'VerticalAlignment','middle','Visible','on','Tag','txt_pidrudder','Callback','')
handles.txt_CBy=uicontrol(f,'unit','normalized','BackgroundColor',[1,1,1],
'Enable','on','FontAngle','normal','FontName','Times New Roman','FontSize',[12],
'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1],
'HorizontalAlignment','left','ListboxTop',[],
'Max',[1],
'Min',[0],
'Position',[0.11,0.53,0.01,0.0341477],
'Relief','default','SliderStep',[0.01,0.1],
'String','Y','Style','text','Value',[0],
'VerticalAlignment','middle','Visible','on','Tag','txt_pidrudder','Callback','')
handles.txt_CBz=uicontrol(f,'unit','normalized','BackgroundColor',[1,1,1],
'Enable','on','FontAngle','normal','FontName','Times New Roman','FontSize',[12],
'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1],
'HorizontalAlignment','left','ListboxTop',[],
'Max',[1],
'Min',[0],
'Position',[0.11,0.48,0.01,0.0341477],
'Relief','default','SliderStep',[0.01,0.1],
'String','Z','Style','text','Value',[0],
'VerticalAlignment','middle','Visible','on','Tag','txt_pidrudder','Callback','')

```



```

[FFt,MMt,CCt]=bouyancy(oft,data.tank(2),panel,Cr,rho,'b')
handles.ed_voltank2.string = [string(FFt(1,3)/1.025)]
voltank2(step) = FFt(1,3)/1.025
a = [-40 -25 0; -40 25 0; 80 25 0; 80 -25 0; -40 -25 0]
plot3d(a(:,1),a(:,2),a(:,3))
Cr = [0 0 0];rho = [1.025 7.85]
[FFb,MMb,CCb]=bouyancy(allOffset3,allPanel,Cr,rho,'b')
handles.ed_CBx.string = [string(CCb(1,1))]
handles.ed_CBy.string = [string(CCb(1,2))]
handles.ed_CBz.string = [string(CCb(1,3))]
isi=[isi; "step"+string(step-1) string(displacement(step)) string(voltank1(step)) string(voltank2(step))
string(voltank3(step)) string(voltank4(step)) string(voltank5(step))]
table = [parameter;isi]
handles.recap.string = table
Result(step,:)=result
AP(step) = allOffset3(124,4)
FP(step) = allOffset3(656,4)
end
plotMesh(allOffset3,allPanel)
figure()
newaxes()
plot(-stp,voltank5,"o-",-stp,voltank4,"x-",-stp,voltank3,"d-")
xlabel("Slide Distance(m)")
ylabel("Tank Volume(m3)")
title("Tank Volume x Slide Distance", "fontsize",3)
figure()
newaxes()
disp(AP,FP)
plot(-stp,AP,"o-",-stp,FP,"x-")
xlabel("Slide Distance(m)")
ylabel("Draft(m)")
title("Draft AP&FP x Slide Distance", "fontsize",3)
end
plotlevel(Result,db)
messagebox("Calculation Complete")
endfunction

```

Hasil Run: Draft = 3, Trim = 2°, Slide Distance = 20, Step = 4

Input

Draft: 3 m
Trim: 2 deg
Slide: 20
Step: 4

Output

Displacement: 1742.1072 Ton

Vol. Tank 1: 0 m³
Vol. Tank 2: 335.34574 m³
Vol. Tank 3: 672.46843 m³
Vol. Tank 4: 243.1109 m³
Vol. Tank 5: 187.66846 m³

Recap

	Disp.	VT1	VT2	VT3	VT4	VT5
step0	1080...0	335...	322...	84.2...	50.5...	
step1	1243...0	335...	435...	113...	67.9...	
step2	1407...0	335...	549...	142...	85.3...	
step3	1574...0	335...	665...	171...	102...	
step4	1645...0	335...	672...	200...	136...	
step5	1742...0	335...	672...	243...	187...	

PLOT
RUN

Graphic window number 7

Tank Volume x Slide Distance

Tank Volume(m³) vs Slide Distance(m)

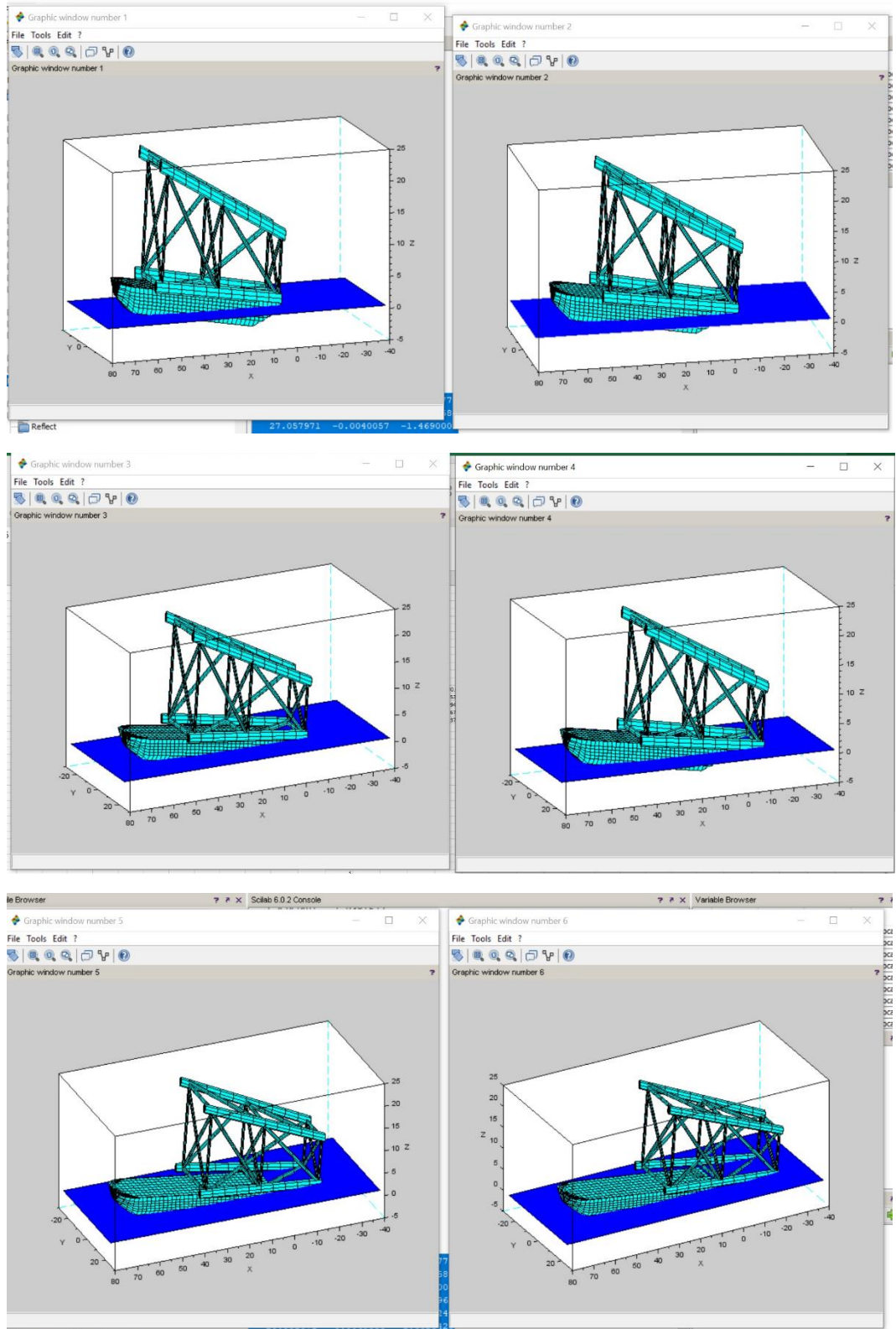
Graphic window number 9

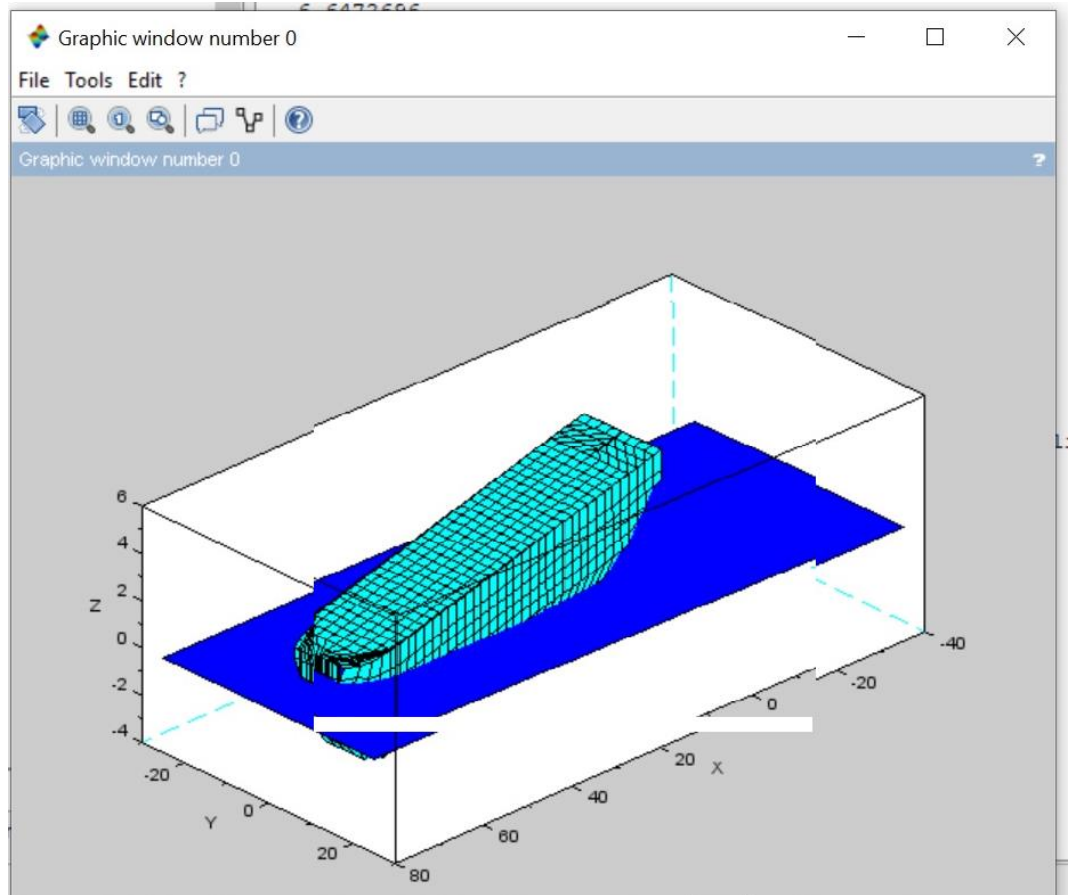
Water Level at Step 5

0 5 10 15 20 25 30 35 40 45 50 55 60 65 70

Polyspace_Workspace
Reflect
Rockstar Games

26.156734 -0.0041419 -1.390058
27.057971 -0.0040057 -1.469000
27.950214 -0.0051526 -1.5571961
28.300294 -0.005081 -1.6014244
28.851871 -0.0049833 -1.6521423





Hasil Run: Draft = 3, Trim = 2.5°, Slide Distance = 20, Step = 4

Skripsi ver.1

Input

Draft: 3 m

Trim: 2.5 deg

Output

CB X: 28.710549

Y: -0.0073962

Z: -2.0675552

Displacement: 1774.2588 Ton

Vol. Tank 1: 0 m³

Vol. Tank 2: 325.81396 m³

Vol. Tank 3: 672.48098 m³

Vol. Tank 4: 261.69978 m³

Vol. Tank 5: 225.90631 m³

Jacket

Slide: 20

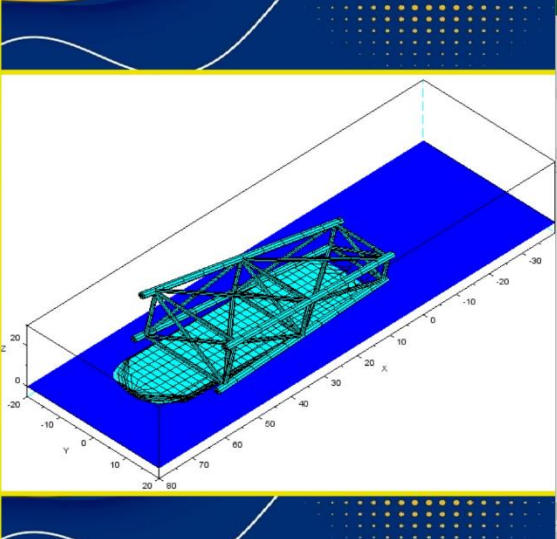
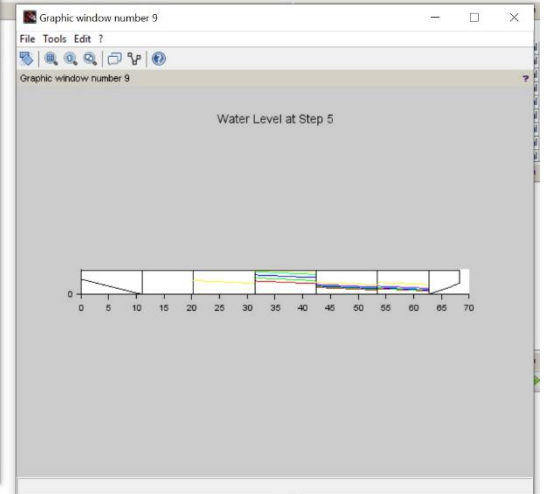
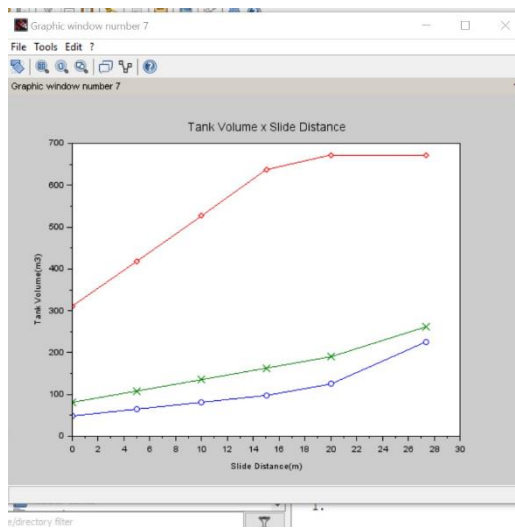
Step: 4

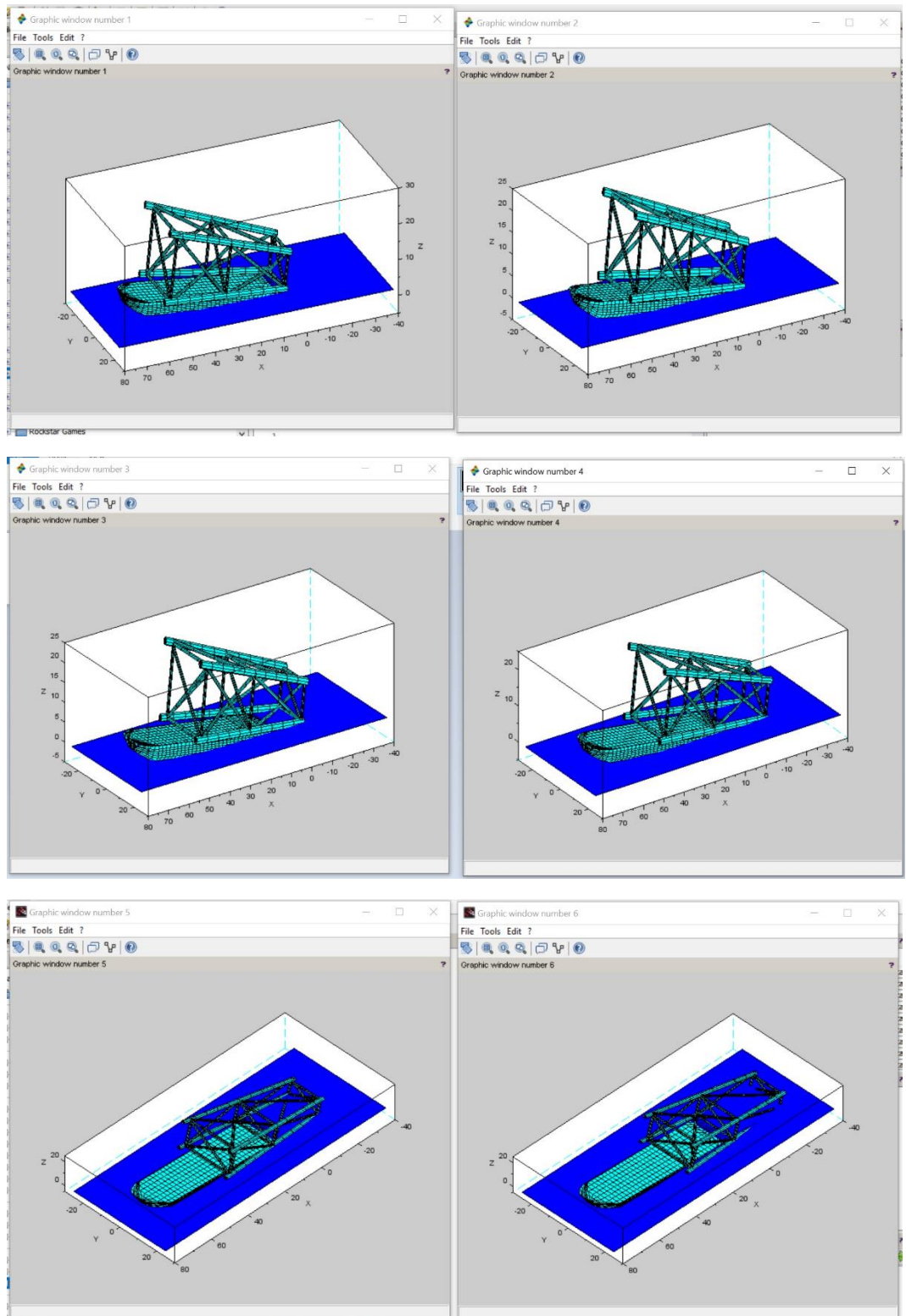
Recap

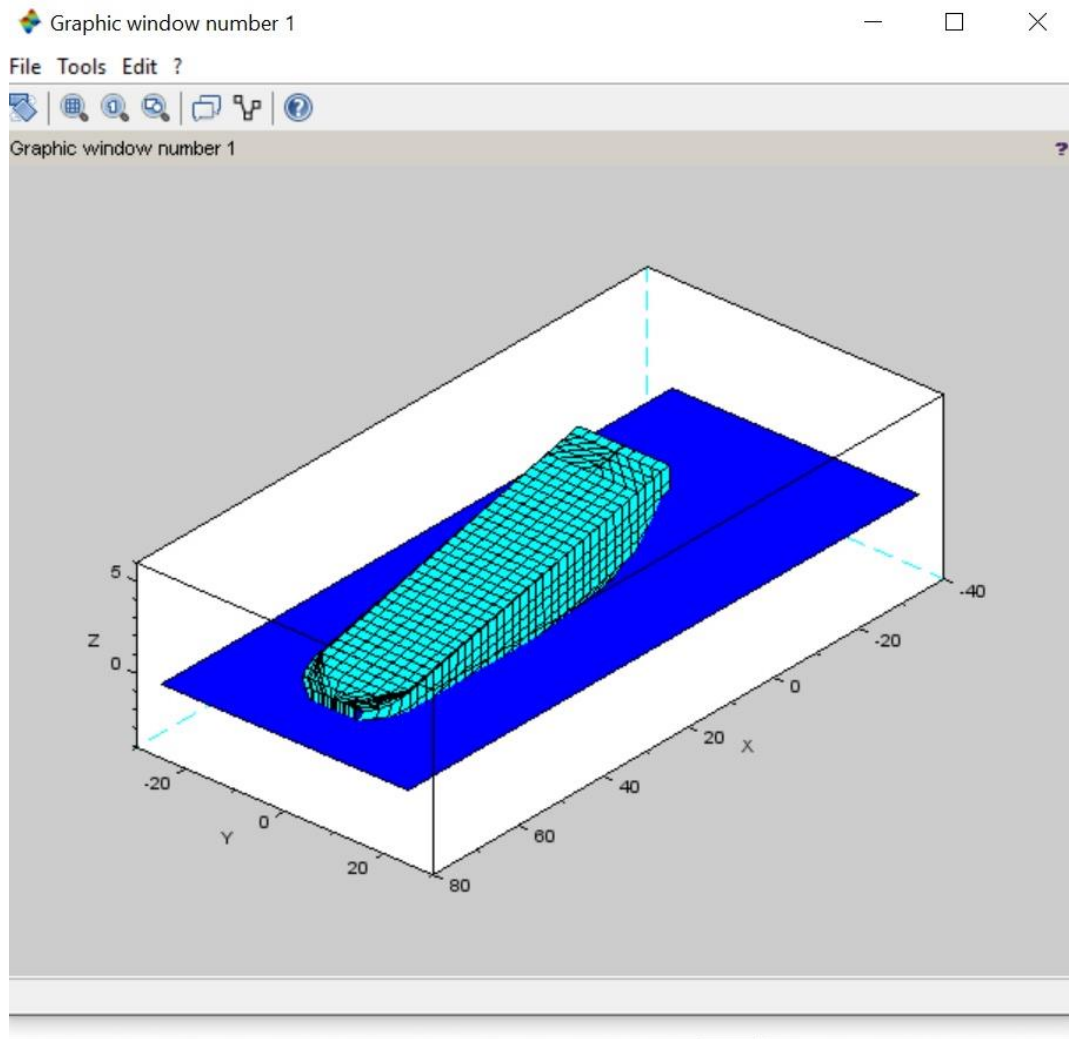
	Dep.	VT1	VT2	VT3	VT4	VT5
step0	1036...	0	325...	311...	80.9...	48.5...
step1	1191...	0	325...	418...	108...	64.9...
step2	1347...	0	325...	526...	135...	81.2...
step3	1505...	0	325...	637...	162...	97.5...
step4	1597...	0	325...	672...	190...	124...
step5	1774...	0	325...	672...	251...	225...

PLOT

RUN



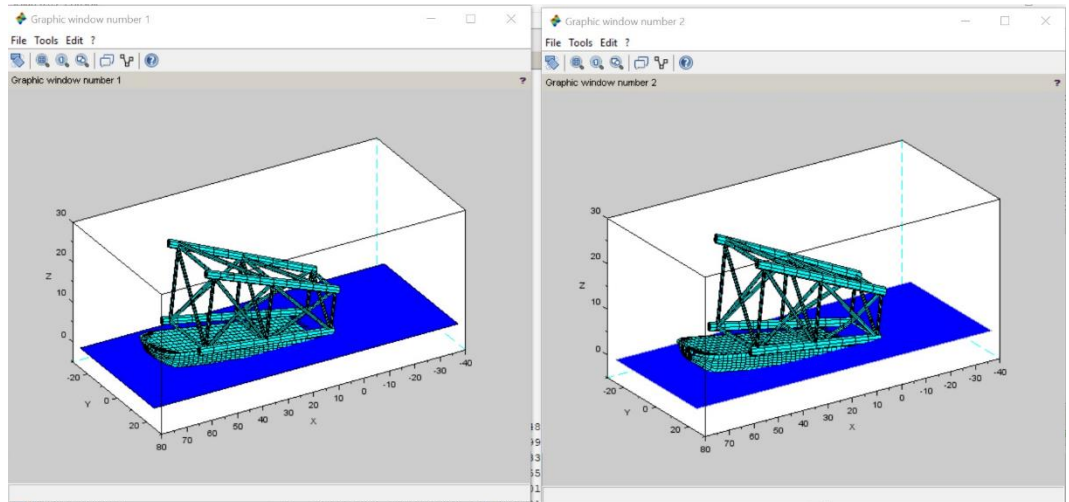
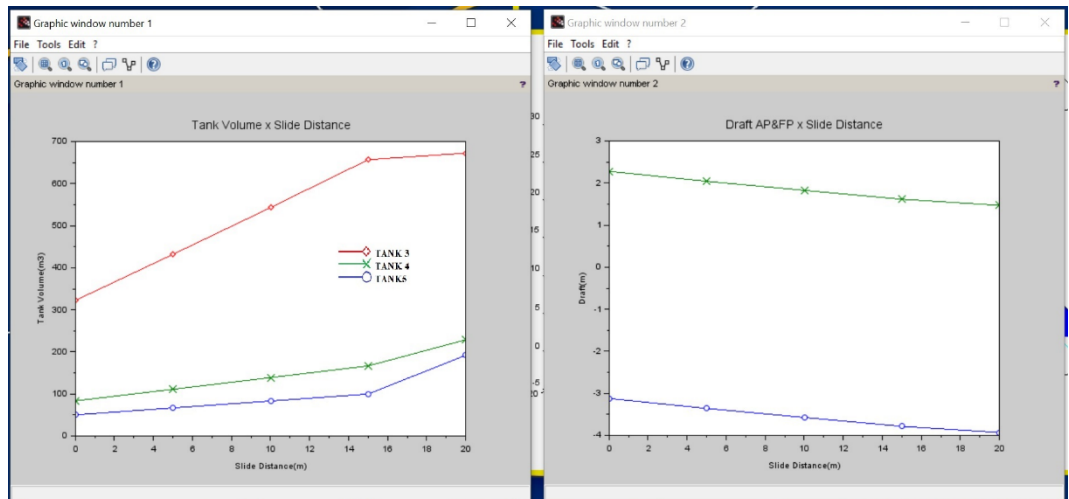


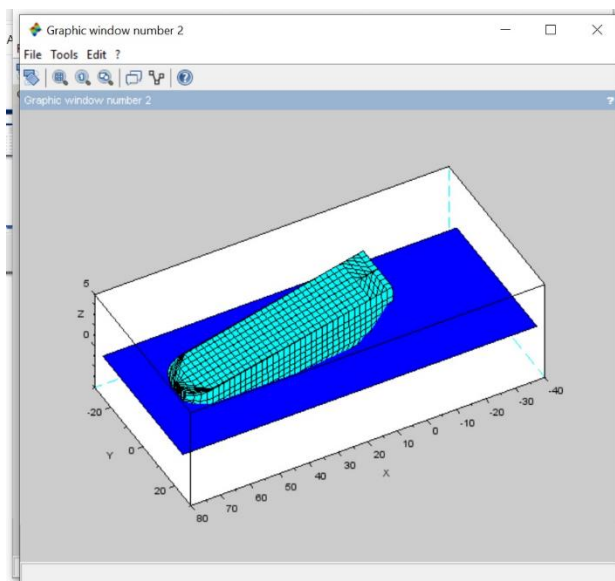
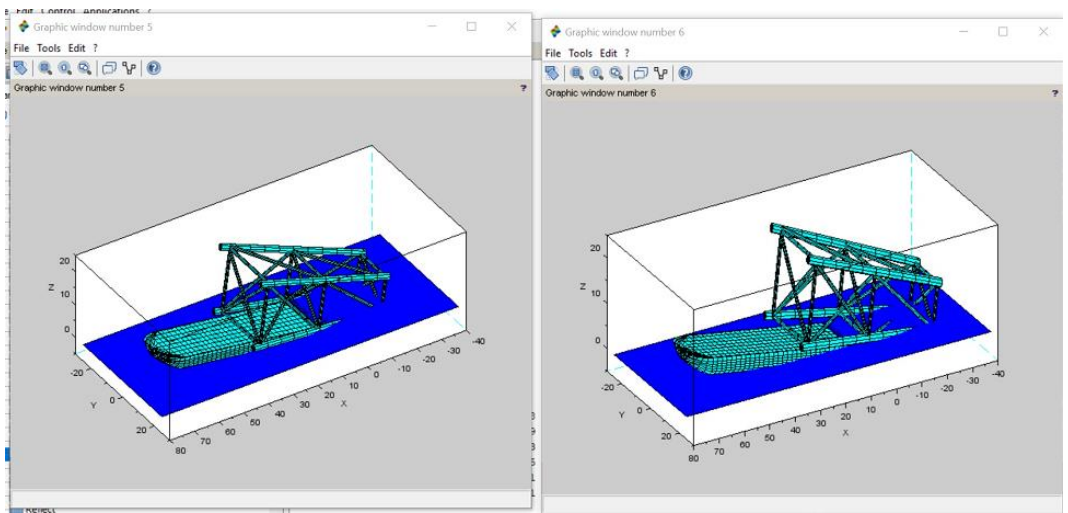
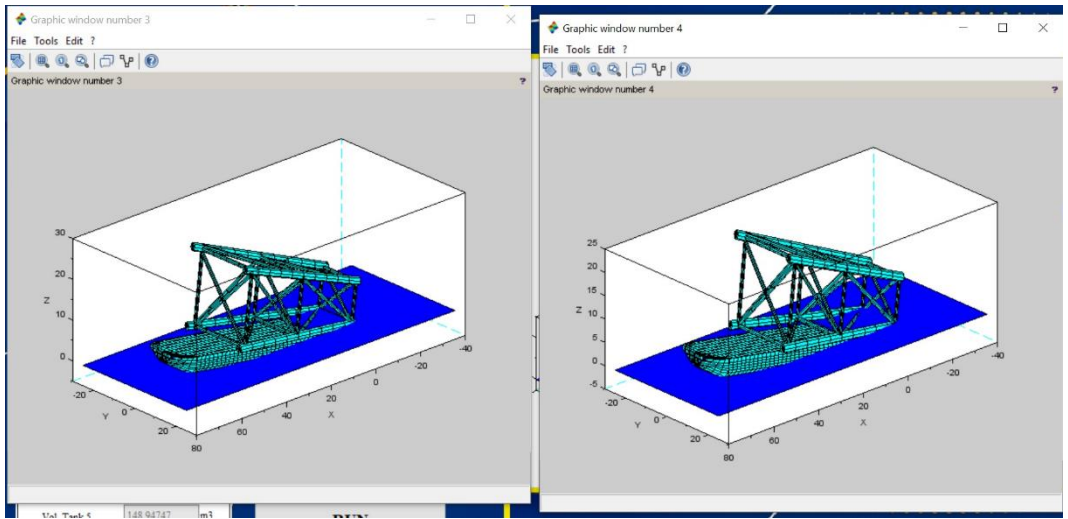
Hasil Run: Draft = 3, Trim = 3°, Slide Distance = 20, Step = 4

The software interface is divided into several sections:

- Input:** Draft = 3 m, Trim = 3 deg.
- Jacket:** Slide = 20, Step = 4.
- Output:**
 - CB: X: 27.438322, Y: -0.0105769, Z: -2.3965101
 - Displacement: 1604.9927 Ton
 - Vol. Tank 1: 0 m³
 - Vol. Tank 2: 316.28942 m³
 - Vol. Tank 3: 672.4711 m³
 - Vol. Tank 4: 209.66778 m³
 - Vol. Tank 5: 148.94747 m³
- Recap Table:**

	Dep.	VT1	VT2	VT3	VT4	VT5
step0	974...	0	316...	294...	76.3...	45.8...
step1	1113...	0	316...	390...	100...	60.3...
step2	1253...	0	316...	487...	124...	74.9...
step3	1390...	0	316...	583...	149...	89.5...
step4	1638...	0	316...	672...	213...	1378...
step5	1604...	0	316...	672...	209...	148...
- Buttons:** PLOT, RUN
- 3D Model:** A wireframe model of a ship hull is shown in a 3D coordinate system with axes X, Y, and Z.

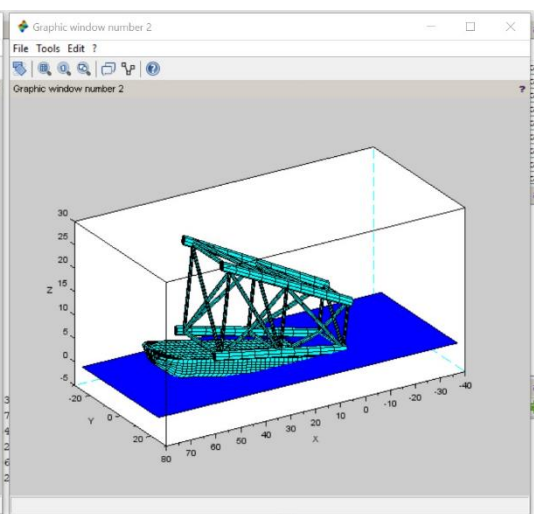
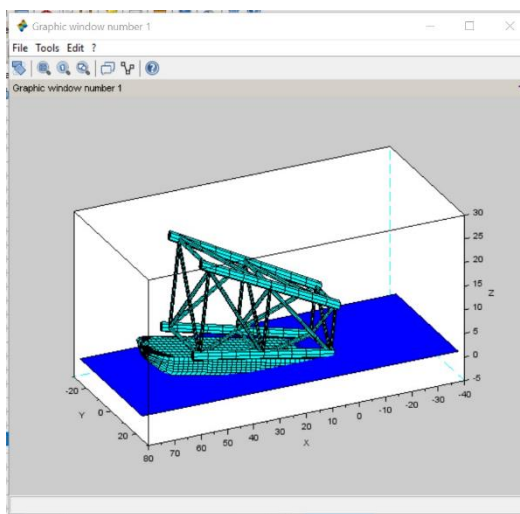
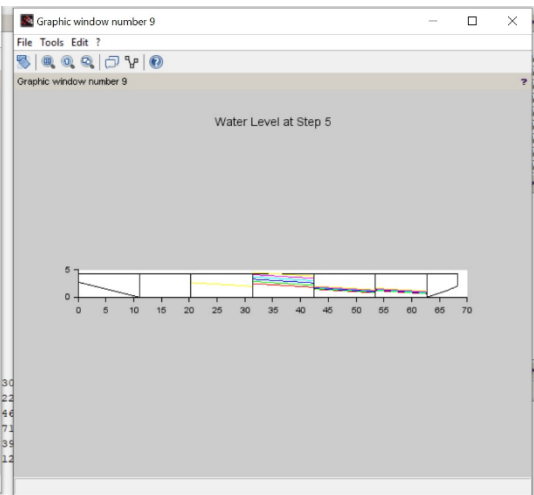
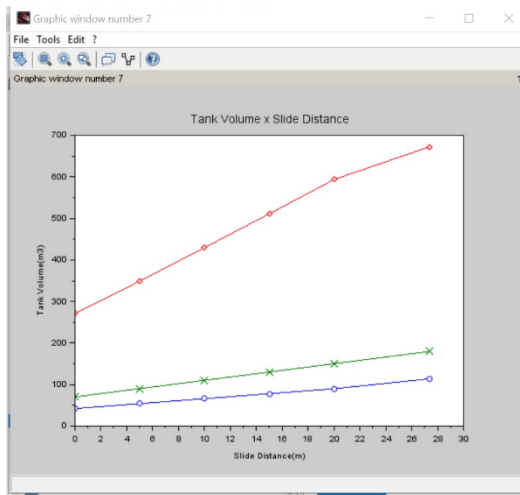


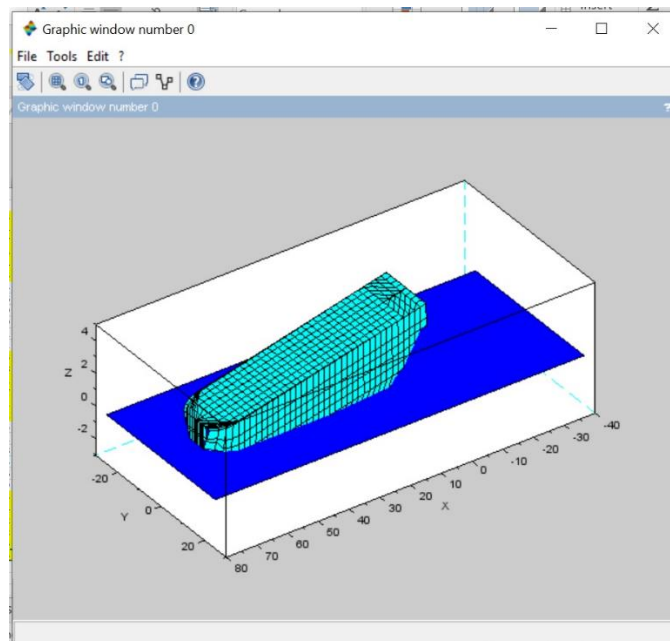
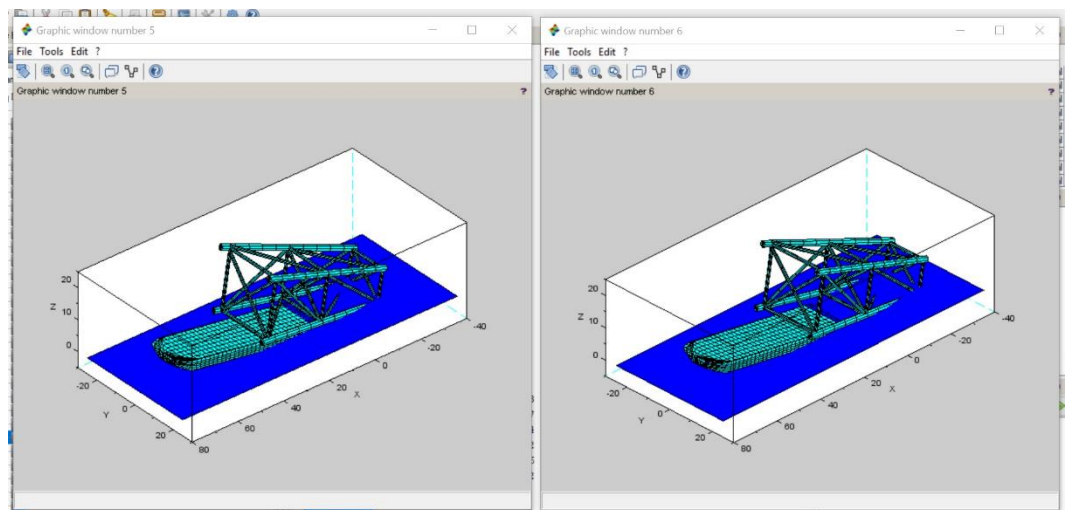
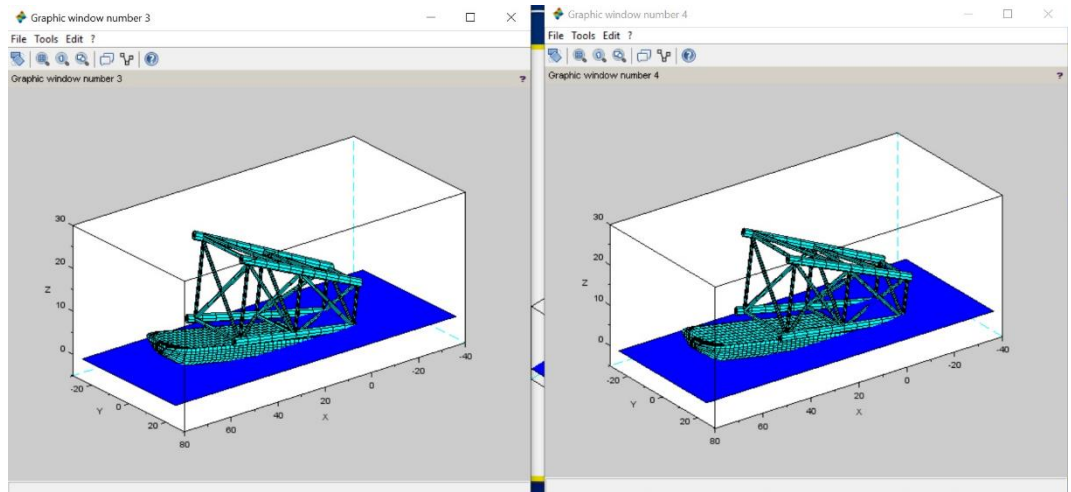


Hasil Run: Draft = 3, Trim = 3.5°, Slide Distance = 20, Step = 4

The screenshot shows the Skripoti ver.1 software interface. On the left, there are input fields for Draft (3 m) and Trim (3.5 deg) under the 'Input' section, and Slide (20) and Step (4) under the 'Jacket' section. Below these are output fields for CB (X: 26.184019, Y: -0.0144968, Z: -2.6312122), Displacement (1488.7202 Ton), and five tank volumes (Vol. Tank 1 to 5). A 'Recap' table is also present, and at the bottom are 'PLOT' and 'RUN' buttons. On the right, a 3D model of a ship's hull is shown in a blue water environment within a coordinate system.

	Disp.	VT1	VT2	VT3	VT4	VT5
step0	891...	0	306...	271...	70.3...	42.1...
step1	1004...	0	306...	349...	90.2...	54.1...
step2	1119...	0	306...	429...	110...	66.1...
step3	1235...	0	306...	511...	130...	78.0...
step4	1353...	0	306...	593...	150...	90.0...
step5	1488...	0	306...	672...	175...	113...





Hasil Run: Draft = 3, Trim = 4°, Slide Distance = 20, Step = 4

Input

Draft: 3 m

Trim: 4 deg

Jacket

Slide: 20

Step: 4

Output

CB X: 24.803182

Y: -0.0189612

Z: -3.8450591

Displacement: 1425.893 Ton

Vol. Tank 1: 0 m³

Vol. Tank 2: 297.33762 m³

Vol. Tank 3: 665.1835 m³

Vol. Tank 4: 166.35469 m³

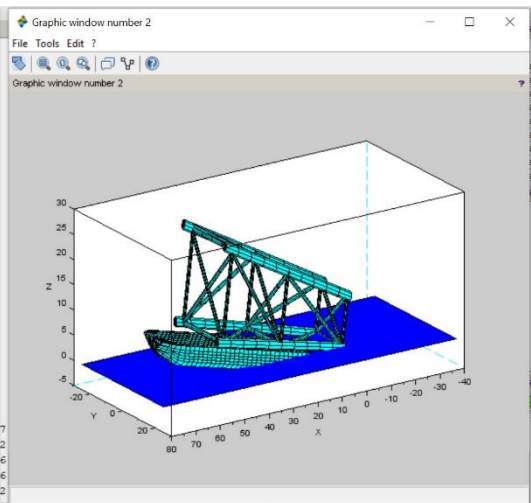
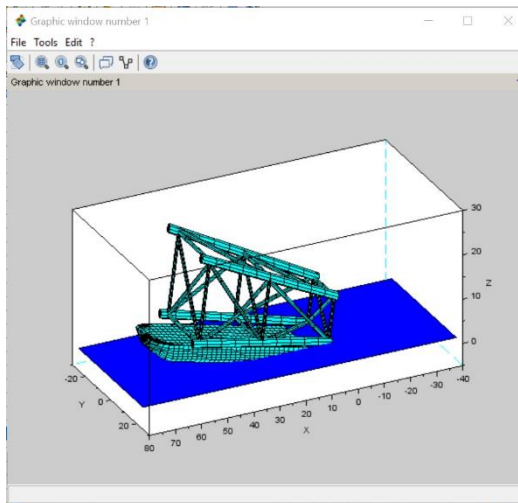
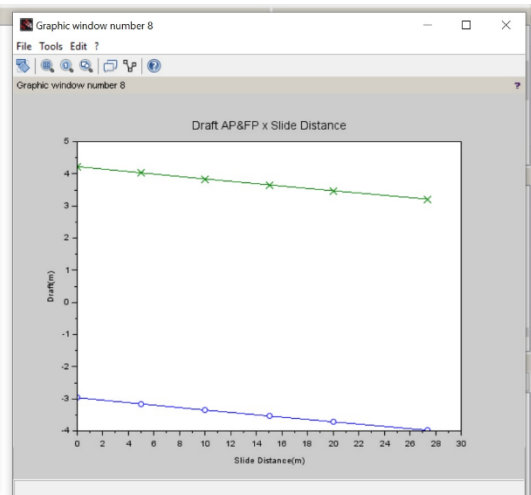
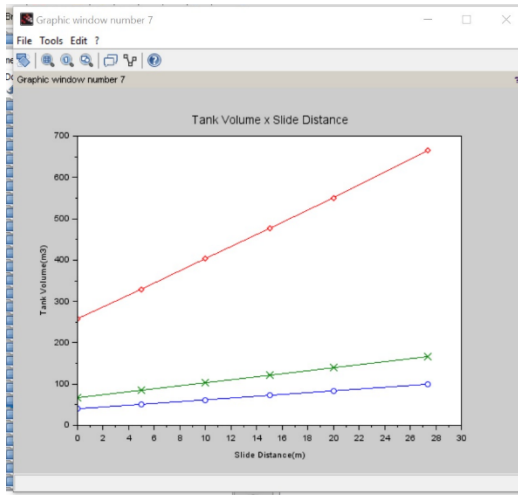
Vol. Tank 5: 99.573301 m³

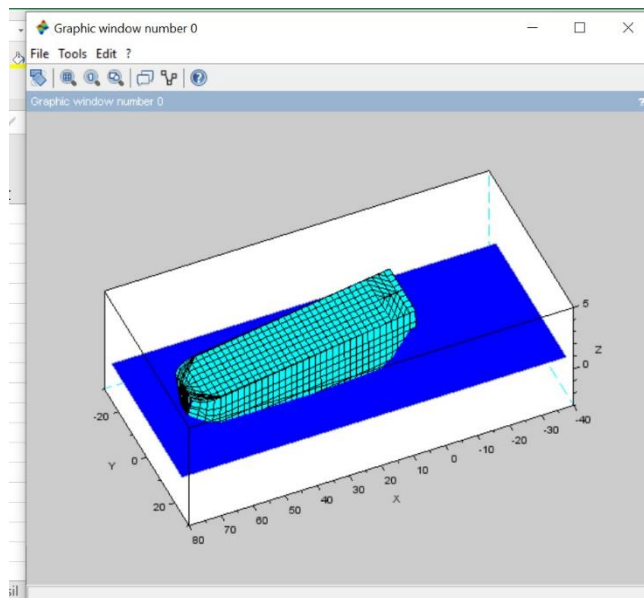
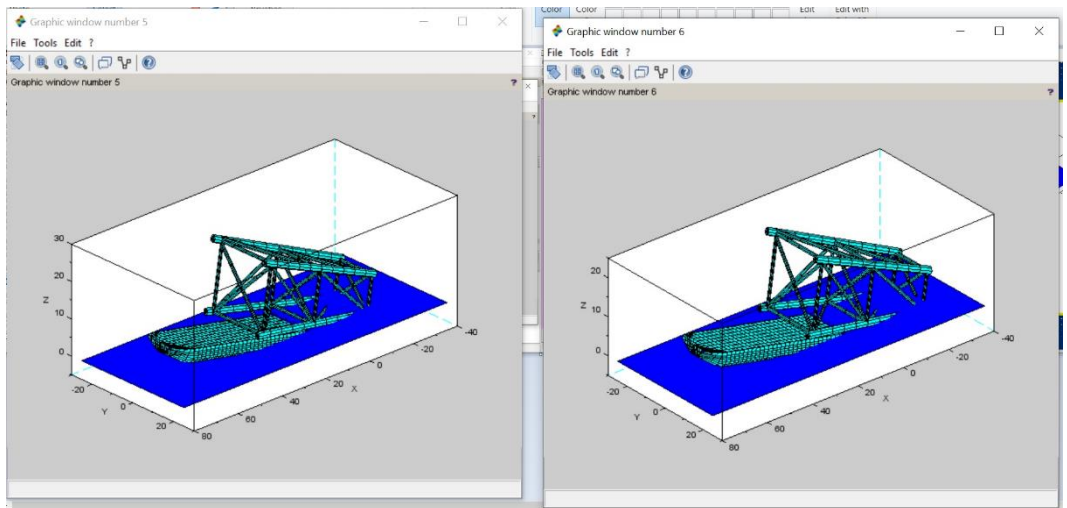
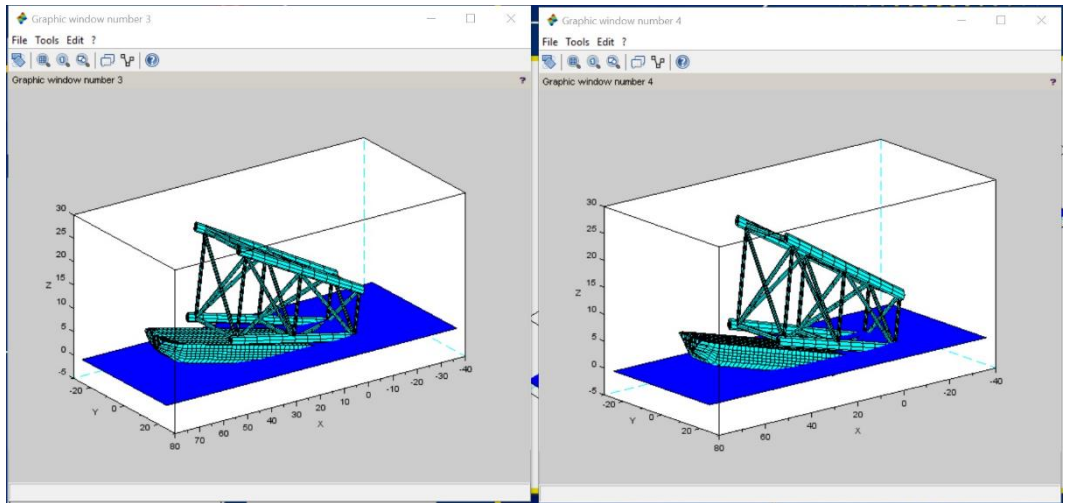
Recap

	Disp.	VT1	VT2	VT3	VT4	VT5
step0	845...	0	297...	257...	66.9...	40.1...
step1	948...	0	297...	329...	85.0...	50.9...
step2	1054...	0	297...	403...	103...	61.8...
step3	1150...	0	297...	476...	121...	72.7...
step4	1264...	0	297...	550...	139...	83.5...
step5	1425...	0	297...	665...	166...	99.5...

PLOT

RUN







SURAT PENUGASAN

No.6371/UN4.7.1/TD.06/2022

Dari : Dekan Fakultas Teknik Universitas Hasanuddin

Kepada : 1. **Rahimuddin, S.T., M.T., Ph.D.** Pemb. I
2. **Dr. Ir. Ganding Sitepu, Dipl.-Ing.** Pemb. II

Isi : 1. Berdasarkan Surat Ketua Departemen Teknik Sistem Perkapalan Fakultas Teknik Nomor : 6370/UN4.7.7/TD.06/2022 tanggal 31 Maret 2022 tentang Dosen PEMBIMBING MAHASISWA, maka dengan ini kami menugaskan Saudara untuk membimbing penulisan Skripsi/Tugas Akhir mahasiswa Teknik Sistem Perkapalan Fakultas Teknik Universitas Hasanuddin di bawah ini :

Nama : **Muhammad Assiddiq** No. Stambuk : **D091171013**

Judul Skripsi/Tugas Akhir :

Simulasi Pengaturan Ballast untuk Kendali Trim pada Barge Jacket Launching

2. Surat penugasan pembimbing ini mulai berlaku sejak tanggal ditetapkannya dan berakhir sampai selesainya penulisan Skripsi/Tugas Akhir Mahasiswa tersebut.
3. Agar surat penugasan ini dilaksanakan sebaik - baiknya dengan penuh rasa tanggung jawab.

Ditetapkan di Gowa,
Pada tanggal, 31 Maret 2022

a.n Dekan,
Wakil Dekan I Bidang Akademik, Riset dan
Inovasi Fakultas Teknik UH

Prof. Baharuddin Hamzah, ST.,M.Arch.,Ph.D
Nip. 19690308 199512 1 001

Tembusan :

1. Dekan FT-UH.
2. Ketua Departemen Teknik Sistem Perkapalan FT-UH.
3. Mahasiswa yang bersangkutan





SURAT PENUGASAN

No.16400/UN4.7.1/TD.06/2022

Dari : Dekan Fakultas Teknik Universitas Hasanuddin

Kepada : Mereka yang tercantum namanya dibawah ini.

- Isi : 1. Bahwa berdasarkan peraturan Akademik Universitas Hasanuddin Tahun 2018 pasal 19 (SK. Rektor Unhas nomor : 2781/UN4.1/KEP/2018), dengan ini menugaskan Saudara sebagai PANITIA UJIAN SARJANA Program Strata Satu (S1) Teknik Sistem Perkapalan Fakultas Teknik Universitas Hasanuddin dengan susunan sebagai berikut :
- Ketua : Rahimuddin, S.T., M.T., Ph.D.
Sekretaris : Dr. Ir. Ganding Sitepu, Dipl.-Ing.
Anggota : 1. Andi Haris Muhammad, S.T., M.T., Ph.D
2. Haryanti Rivai, S.T., M.T..Ph.D.

Untuk menguji bagi mahasiswa tersebut dibawah ini :

Nama/Nim : Muhammad Assiddiq / D091171013

Departemen : Teknik Sistem Perkapalan

Judul Thesis/Skripsi :

Simulasi Pengaturan Ballast untuk Kendali Trim pada Barge Jacket Launching

2. Waktu ujian ditetapkan oleh Panitia Ujian Akhir Program Strata Satu (S1).
3. Agar surat penugasan ini dilaksanakan sebaik-baiknya dengan penuh rasa tanggung jawab.
4. Surat penugasan ini berlaku sejak tanggal ditetapkan sampai dengan berakhirnya Ujian Sarjana tersebut, dengan ketentuan bahwa segala sesuatunya akan ditinjau dan diperbaiki sebagaimana mestinya apabila dikemudian hari ternyata terdapat kekeliruan dalam keputusan ini.

Ditetapkan di Gowa,

Pada tanggal , 9 Agustus 2022

a.n Dekan,

Wakil Dekan Bidang Akademik, Riset dan Inovasi Fakultas Teknik UH

Dr. Amil Ahmad Ilham, S.T., M.IT.

Nip.19731010 199802 1 001

Tembusan :

1. Dekan FT-UH.
2. Ketua Departemen Teknik Sistem Perkapalan FT-UH.
3. Kasubag Umum dan Perlengkapan FT-UH





**KEMENTERIAN PENDIDIKAN, KEBUDAYAAN,
RISET, DAN TEKNOLOGI
UNIVERSITAS HASANUDDIN**

KAMPUS TAMALANREA

JALAN PERINTIS KEMERDEKAAN KM.10 MAKASSAR 90245

TELEPON : 0411-586200 (6 SALURAN), 584002, FAX. 585188

SURAT PERSETUJUAN

Nomor : 193300/UN4.1.1.2.1.1/PK.02.03/2022

Berdasarkan Peraturan Rektor Universitas Hasanuddin tentang Penyelenggaraan Program Sarjana Nomor : 2781/UN4.1/KEP/2018 tanggal 16 Juli 2018, dengan ini menerangkan bahwa :

NIK : 7311030105000002
N a m a : MUHAMMAD ASSIDDIQ
Tempat/Tanggal Lahir : JAKARTA, 1 MEI 2000
NIM : D091171013
Fakultas : TEKNIK
Program Studi : TEK. SISTEM PERKAPALAN

Telah memenuhi syarat untuk Ujian Skripsi Strata I (S1) **PERIODE JULI 2022**. Demikian Surat Persetujuan ini dibuat untuk digunakan dalam proses pelaksanaan ujian skripsi, dengan ketentuan dapat mengikuti wisuda **PERIODE JULI 2022**, jika persyaratan kelulusan/wisuda telah dipenuhi. Terima Kasih.

Makassar, 14 JULI 2022



Kepala Biro Administrasi Akademik
u.b Kepala Sub Bagian Pendidikan dan Evaluasi
Universitas Hasanuddin,

MURSALIM, S.Sos.

NIP. 19730216 199601 1001

Keterangan :

Nomor User : D091171013

Nomor password/pin : 2169976

Alamat Website : <http://unhas.ac.id/akad/wisuda/>

Catatan

1. Bagi Mahasiswa yang telah melaksanakan ujian Sarjana dan dinyatakan lulus, segera menyerahkan lembar pengesahan Skripsi dan Berita Acara Ujian Sarjana ke Sub Bagian Akademik Fakultas, untuk memperoleh nomor Alumni dan didaftar sebagai Wisudawan pada periode berjalan.
2. Jika terjadi perubahan Judul Skripsi agar melaporkan ke Kasubag. Pendidikan Fakultas sebelum didaftar sebagai Wisudawan pada Periode berjalan
3. Pada saat ON-LINE Mahasiswa diharapkan mengisi identitas diri sesuai surat izin ujian ini
4. Surat izin ini hanya berlaku untuk Wisuda periode berjalan (WISUDA PERIODE JULI 2022)





BERITA ACARA UJIAN SARJANA

Terhadap Mahasiswa

Nama : Muhammad Assiddiq
Stambuk : D091171013
Judul : *Simulasi Pengaturan Ballast untuk Kendali Trim pada Barge Jacket Launching*
Hari/Tanggal : Selasa, 16 Agustus 2022
Waktu : 09.00-10.00 Wita
Tempat : Ruang Sidang Teknik Sistem Perkapalan (Daring/Lur
Keputusan Sidang / Catatan : *Lulus, A (86)*

PANITIA UJIAN

No.	Susunan Panitia	Nama	Tanda Tangan
1	Ketua/Anggota	Rahimuddin, S.T., M.T., Ph.D.	1.....
2	Sekretaris/Anggota	Dr. Ir. Ganding Sitepu, Dipl.-Ing.	2.....
3	Anggota	Andi Haris Muhammad, S.T., M.T., Ph.D	3.....
4	Anggota	Haryanti Rivai, S.T., M.T..Ph.D.	4.....

Ketua Sidang,

Rahimuddin, S.T., M.T., Ph.D.
Nip. 19710825 199903 1 002

Gowa ,

2022

Sekretaris Sidang,

Dr. Ir. Ganding Sitepu, Dipl.-Ing.
Nip. 19600425 198811 1 001