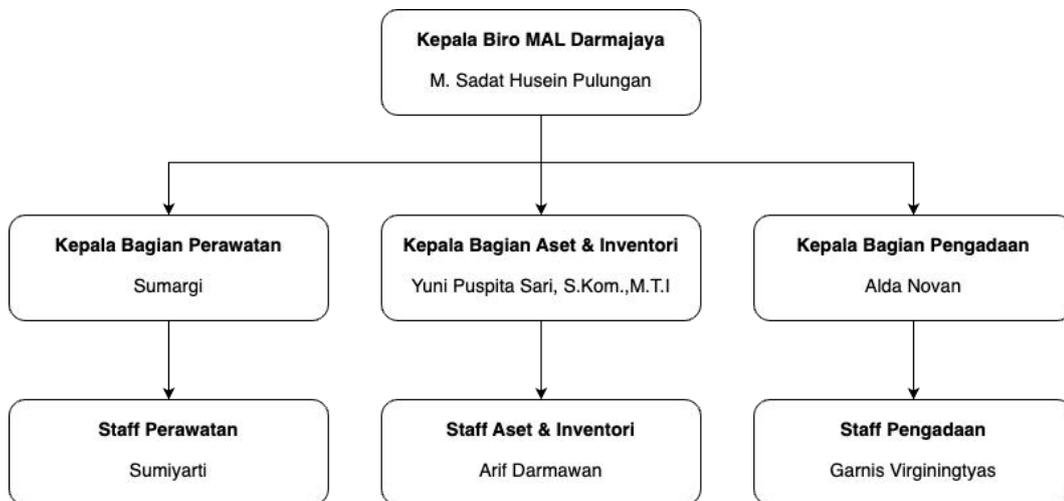


BAB II TINJAUAN PUSTAKA

2.1 Biro Manajemen Aset dan Logistik (BMAL)

2.1.1 Struktur Organisasi

Struktur organisasi pada Biro MAL Darmajaya sebagai berikut :



Gambar 2.1 Struktur Organisasi Biro MAL Darmajaya

2.1.2 Ruang Lingkup Pekerjaan

Berikut adalah ruang lingkup pekerjaan yang dapat diidentifikasi:

1. Pengelolaan Aset:
 - a. Pencatatan dan dokumentasi inventarisasi semua aset dan peralatan yang dimiliki oleh institusi.
 - b. Menentukan standar dan prosedur untuk pengadaan, penyimpanan, dan pemeliharaan aset.
 - c. Melakukan pemantauan dan pemeliharaan rutin terhadap aset untuk memastikan keandalan dan ketersediaan.

2. Peminjaman Aset:
 - a. Menerima, merekam, dan memproses permohonan peminjaman aset dari berbagai unit atau departemen di institusi.
 - b. Menentukan prioritas peminjaman aset berdasarkan kebutuhan dan ketersediaan.
 - c. Memastikan penggunaan aset sesuai dengan peraturan dan batasan yang berlaku.
3. Pengelolaan Logistik:
 - a. Mengelola persediaan dan penyimpanan barang-barang yang diperlukan oleh institusi.
 - b. Merencanakan distribusi dan pengiriman barang ke berbagai unit bagian.
 - c. Menjaga kecukupan stok barang untuk mendukung operasional institusi.
4. Manajemen Ruang:
 - a. Pengelolaan dan penjadwalan penggunaan ruang di institusi, termasuk ruang kelas, laboratorium, aula, dan fasilitas lainnya.
 - b. Memastikan ruang digunakan secara efisien dan sesuai dengan jadwal yang telah ditetapkan.
5. Pengadaan dan Pembelian:
 - a. Merencanakan dan mengkoordinasikan proses pengadaan dan pembelian aset dan barang-barang lainnya sesuai dengan kebutuhan institusi.
 - b. Menegosiasikan kontrak dengan pemasok dan vendor.
6. Penyusunan Laporan:
 - a. Membuat laporan terkait dengan pengelolaan aset, logistik, dan pengelolaan ruang.
 - b. Menyajikan informasi yang relevan kepada manajemen institusi.
7. Kebijakan dan Peraturan:
 - a. Mengembangkan dan memastikan penerapan kebijakan dan peraturan yang berkaitan dengan manajemen aset dan logistik.

- b. Memastikan kepatuhan terhadap regulasi yang berlaku dalam pengelolaan aset dan logistik.

Ruang lingkup pekerjaan di Biro MAL Darmajaya melibatkan banyak aspek dalam pengelolaan aset, logistik, dan pengelolaan ruang di IBI Darmajaya. Berikut data spesifikasi aset yang menjadi ruang lingkup Biro MAL Darmajaya pada table 3.1 :

Tabel 2.1 Perangkat Aset dan Ruang tidak memakai Ruang Lingkup

No	Jenis	Nama Komponen	Spesifikasi	Jumlah
1	Aset Ruang	Gedung Aula	Kapasitas 300 Orang	1 Unit
		Ruang Rapat (A)	Kapasitas 50 Orang	1 Unit
		Gedung Rapat (B2)	Kapasitas 60-70 Orang	1 Unit
		Ruang IAC Seminar	Kapasitas 20-25 Orang	1 Unit
		Laboratorium IAC	Kapasitas 40 Orang	1 Unit
		Ruang Rapat Utama (F5)	Kapasitas 20 orang	1 Unit
		Ruang Rapat Filkom (F5)	Kapasitas 20 Orang	1 Unit
		Ruang Rapat Ekonomi (F5)	Kapasitas 20 Orang	1 Unit
		Pelataran Co-Working (DSC)	Kapasitas 100 Orang	1 Unit
		Ruang Rapat Co-Working (DSC)	Kapasitas 20 Orang	1 Unit
2	Aset Barang	Microfone Wireless	Microfon Wireless Tanpa Kabel	6 Unit
		Microfone	Microfone Dengan Kabel	4 Unit
		Sound Wireless	Sound wireless tanpa kabel dengan koneksi bluetooth	4 Unit
		Sound	Sound dengan koneksi kabel jack	4 Unit

2.1.3 Prosedur Peminjaman Aset

Dalam konteks akademik, proses peminjaman aset dan pengelolaan ruang di IBI Darmajaya dapat diuraikan sebagai berikut:

1. Mahasiswa/Dosen/Staff selaku peminjam mengajukan permohonan peminjaman aset atau penggunaan ruang dengan bersurat untuk keperluan akademik atau kegiatan tertentu.
2. Permohonan dari Mahasiswa/Dosen/Staff peminjam diterima oleh Kabiro MAL Darmajaya, yang bertanggung jawab dalam mengoordinasikan peminjaman aset dan penggunaan ruang.
3. Kabiro MAL Darmajaya melakukan disposisi kepada Kepala Bagian Aset untuk menilai ketersediaan aset dan ruang yang diminta oleh mahasiswa.
4. Kepala Bagian Aset memeriksa ketersediaan aset dan ruang serta menentukan apakah permohonan dapat disetujui atau tidak.
5. Jika permohonan disetujui, Kepala Bagian Aset mengoordinasikan dengan staff aset untuk menyiapkan aset barang dan ruang sesuai dengan jadwal yang telah ditentukan.
6. Staff aset memeriksa ketersediaan barang dan ruang yang diminta, dan setelah persiapan selesai, mereka memberikan konfirmasi kepada mahasiswa/dosen/staff peminjam bahwa aset dan ruang tersebut dapat digunakan sesuai dengan jadwal yang telah disepakati dan ditulis di papan tulis dinding ruang Biro MAL sebagai pengingat.

Proses ini mencerminkan langkah-langkah dalam pengelolaan peminjaman aset dan penggunaan ruang di IBI Darmajaya, yang melibatkan kerjasama antara mahasiswa, dosen/staff peminjam, Kabiro MAL Darmajaya, Kepala Bagian Aset, dan staff aset untuk memastikan kebutuhan terpenuhi dengan efisien dan sesuai dengan aturan yang berlaku.

2.2 Algoritma *Bubble Sort*

Bubble Sort merupakan jenis Algoritma pengurutan yang sederhana. Algoritma ini termasuk dalam kelas Algoritma comparison Sort, yaitu melakukan perbandingan dalam operasi antar elemen. Ide dari Algoritma ini adalah melakukan perulangan, membandingkan setiap elemen array dan menukarnya dengan urutan yang salah. Proses perbandingan elemen ini diulang sampai tidak diperlukannya pertukaran lebih lanjut. [1]

Langkah-langkah *Bubble Sort* :

Data yang melakukan peminjaman diproses pengurutan berdasarkan nilai prioritas sebagai berikut:

Data Awal :

3	2	6	1
---	---	---	---

Keterangan : Nilai "3" = Mahasiswa Peminjam Pertama
 Nilai "2" = Mahasiswa Peminjam Kedua
 Nilai "6" = Dosen/Staf Peminjam Ketiga
 Nilai "1" = Mahasiswa Peminjam Keempat

Tahap 1 =

a	b
3	2

3	2	6	1
---	---	---	---

Keterangan : a = 3
 b = 2
 if (a > b) = ya

Hasil :

a	b
3	2

3	2	6	1
---	---	---	---

Tahap 2 =

a	b
3	2

3	2	6	1
---	---	---	---

Keterangan : a = 2
 b = 6
 if (a > b) = swap

Hasil :

a	b
3	6

3	6	2	1
---	---	---	---

a	b
3	6

Tahap 3 =

3	6	2	1
---	---	---	---

Keterangan : $a = 2$

$b = 1$

if ($a > b$) = ya

Hasil :

	a	b	
3	6	2	1

Tahap 4 =

	a	b	
3	6	2	1

Keterangan : $a = 6$

$b = 2$

if ($a > b$) = ya

Hasil :

	a	b	
3	6	2	1

Tahap 5 =

	a	b	
3	6	2	1

Keterangan : $a = 3$

$b = 6$

if ($a > b$) = swap

Hasil :

	a	b	
6	3	2	1

Penggunaan algoritma *Bubble Sort* untuk prioritas peminjaman aset di Biro MAL Darmajaya memiliki sejumlah keuntungan, termasuk sederhana dan mudah dimengerti, dapat mengatasi keterbatasan aset dengan cepat, tidak memerlukan keterampilan komputasi yang tinggi, dan efisien untuk jumlah data yang kecil.

2.3 Peminjaman Aset

Secara sederhana, peminjaman dapat diartikan sebagai barang atau jasa yang menjadi kewajiban pihak yang satu untuk dibayarkan kepada pihak lain sesuai dengan perjanjian tertulis ataupun lisan, yang dinyatakan atau diimplementasikan serta wajib dibayarkan kembali dalam jangka waktu tertentu. [2]

Kata “aset” dikenal pertama kali pada pertengahan abad ke-16; berasal dari bahasa Prancis “asez/asetz” yang berarti cukup. International Financial Reporting Standard (IFRS) menyebutkan bahwa aset adalah sumber daya yang dikuasai oleh perusahaan sebagai akibat dari peristiwa masa lalu; dari sanalah manfaat ekonomi masa depan diharapkan mengalir ke perusahaan. Kekayaan bersih dapat dihitung dengan mengurangkan kewajiban dari aset. [3]

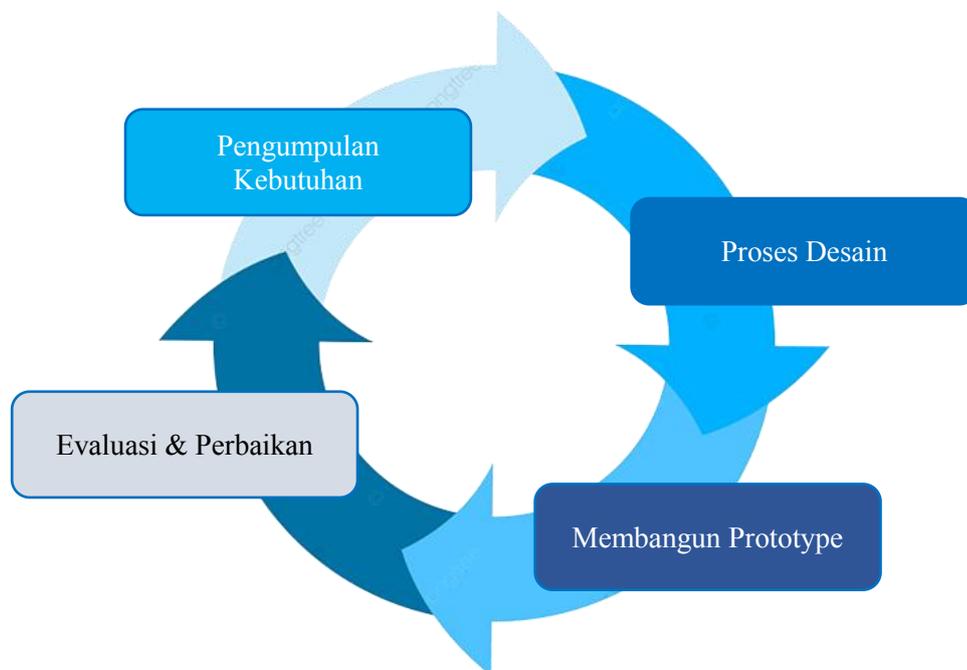
Pada dasarnya, aset adalah segala sesuatu yang dimiliki. Dalam konteks yang lebih luas, aset dapat berupa banyak hal, mulai dari sumber daya alam, sumber daya manusia, jaringan relasi, pendidikan, pengetahuan, teknologi, seni budaya, nama baik, dan kepercayaan publik. Bahkan ide, informasi, dan imajinasi pun dapat menjadi aset yang berharga. Semua perusahaan memiliki aset berwujud yang merupakan pendukung operasional dari perusahaan tersebut. menurut pendapat Hery. [3]

“Aset tetap (fixed aset) adalah aset yang secara fisik dapat dilihat keberadaannya dan sifatnya relatif permanen serta memiliki masa kegunaan (useful life) yang panjang. Aset tetap merupakan aset yang berwujud (tangible aset), berbeda dengan aset tidak berwujud (intangible assets) yang dimana tidak memiliki wujud fisik dan dihasilkan sebagai akibat dari sebuah kontrak hukum, ekonomi maupun kontrak sosial”. [3]

Dari definisi tersebut dapat disimpulkan bahwa aset tetap merupakan sumber daya yang penting untuk mendukung operasional bisnis agar tercapai tujuan perusahaan tersebut. Suatu persetujuan, dengan mana pihak yang satu mengikatkan diri untuk memberikan kenikmatan suatu barang kepada pihak yang lain selama waktu tertentu. [3]

2.4 Metode Pengembangan *Prototype*

Metode pengembangan yang digunakan dalam pembuatan aplikasi ini yaitu dengan menggunakan metode *prototype*.



Gambar 2.2 Langkah-Langkah *Prototype*

Menurut (Purnomo 2017), prototyping dimulai dengan pengumpulan kebutuhan, melibatkan pengembang dan pengguna sistem untuk menentukan tujuan, fungsi dan kebutuhan operasional sistem [4]. Langkah- langkah dalam prototyping adalah sebagai berikut :

1. Pengumpulan Kebutuhan.
2. Proses desain yang cepat.

3. Membangun prototipe.
4. Evaluasi dan perbaikan.

Mengumpulkan persyaratan melibatkan pertemuan antara pengembang dan pelanggan untuk menentukan tujuan keseluruhan perangkat lunak, mengenali kebutuhan berupa garis besar kebutuhan dasar dari sistem yang akan dibuat. Desain berfokus pada representasi aspek perangkat lunak dari sudut pandang pengguna ini meliputi format input, proses, dan output.

Desain yang cepat mengarah pada pengembangan prototipe, prototipe dievaluasi oleh pengguna dan analisis desain dan digunakan untuk menyesuaikan dengan kebutuhan perangkat lunak yang akan dikembangkan. Prototipe diatur untuk memenuhi kebutuhan pengguna, dan pada saat yang sama developer lebih mengerti dan dengan apa yang perlu dilakukan. Setelah keempat langkah prototyping dijalankan, langkah selanjutnya adalah pembuatan atau desain produk yang sebenarnya.

2.5 Unified Modeling (UML)

Menurut (Fadallah and S Rosyida 2018) UML (Unified Modeling Language) adalah metode pemodelan secara visual sebagai sarana untuk merancang dan atau membuat software berorientasi objek. UML tidak hanya merupakan sebuah Bahasa pemrograman visual saja, namun juga dapat secara langsung dihubungkan ke berbagai bahasa pemrograman, seperti JAVA, C++, Visual Basic, atau bahkan dihubungkan secara langsung ke dalam sebuah objek-oriented database [5].

Menurut (Dwinda and Henderi 2018) *Unified Modelling Language* (UML) merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung [6].

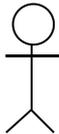
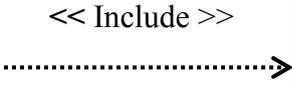
Pada perkembangan teknik pemrograman berorientasi objek, munculah sebuah standarisasi bahasa pemodelan untuk pembangunan perangkat lunak yang di bangun dengan menggunakan teknik pemrograman berorientasi objek

yaitu Unified Modeling Language (UML) dalam UML terdiri dari 13 macam diagram yang dikelompokkan dalam 3 kategori sebagai berikut :

2.4.1 Use Case Diagram

Merupakan pemodelan untuk melakukan (behavior) sistem informasi yang akan dibuat. Use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. secara kasar usecase digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu. Simbol dan keterangan use case diagram seperti pada tabel 2.1.

Tabel 2.2 Simbol dan Keterangan Use case Diagram

No	Simbol	Deskripsi
1.		<i>Use case</i> : Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal <i>frase</i> nama <i>use case</i> .
2.		Aktor: seseorang/sesuatu yang berinteraksi dengan yang akan dibuat. diluar sistem informasi. Biasanya dinyatakan menggunakan kata benda
3.		Asosiasi (<i>association</i>): merupakan komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.
4.		Include berarti <i>use case</i> yang ditambahkan akan dipanggil saat <i>use case</i> tambahan dijalankan.
		Ekstensi (<i>extend</i>) merupakan <i>use case</i> tambahan

No	Simbol	Deskripsi
5.		ke sebuah <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu.

Sumber : (Nugroho, 2010)

2.4.2 Class Diagram

Class Diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi. Atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas. Operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas. Diagram kelas dibuat agar pembuatan perogram membuat kelas-kelas sesuai rancangan didalam diagram kelas antara dokumentasi perancangan dan perangkat lunak sinkron. Berikut ini adalah simbol-simbol yang ada pada *Class* Diagram. Simbol dan keterangan class diagram seperti pada tabel 2.2.

Tabel 2.3 Simbol dan Keterangan *Class* Diagram

No	Simbol	Deskripsi
1	<i>Class</i> 	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama
2	<i>Generalization</i> 	Garis yang melambangkan konsep pewarisan dari suatu kelas ke satu atau lebih sub kelas
3	<i>Association</i> 	apa yang menghubungkan antara objek satu dengan objek lainnya

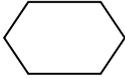
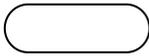
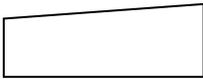
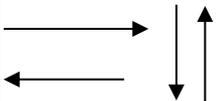
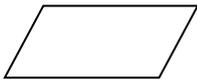
Sumber : (Nugroho, 2010)

2.4.3 Flowchart

Flowchart adalah unit simbol gambar (chart) yang menunjukkan aliran (flow) dari proses terhadap data. Flowchart adalah Suatu Bagan yang menggambarkan arus

logika dari data dalam suatu program dari awal sampai akhir. Bagan alir program merupakan alat yang berguna bagi programmer untuk mempersiapkan program yang rumit.

Tabel 2.4 Simbol dan Keterangan *Flowchart*

Simbol	Nama Simbol	Keterangan
	<i>Predefined Process</i>	Simbol untuk mempersiapkan penyimpanan yang akan digunakan sebagai tempat pengolahan didalam storage
	<i>Terminal</i>	Simbol untuk permulaan atau akhir dari suatu program
	<i>Manual Input</i>	Simbol untuk pemasukan data secara manual on-line keyboard
	<i>Arus / Flow</i>	Penghubung antara prosedur / proses
	<i>Connector</i>	Simbol keluar / masuk prosedur atau proses dalam lembar / halaman yang sama
	<i>Off-line Connector</i>	Simbol keluar / masuk prosedur atau proses dalam lembar / halaman yang lain
	<i>Input-Output</i>	Simbol yang menyatakan proses input dan output tanpa tergantung dengan jenis peralatannya
	<i>Document</i>	Simbol yang menyatakan input berasal dari dokumen dalam bentuk kertas atau output di cetak dikertas

--	--	--

2.6 Penelitian Terkait/Sebelumnya

Dalam penyusunan skripsi ini, peneliti terinspirasi dan mereferensi dari penelitian penelitian sebelumnya yang berkaitan dengan skripsi ini. Daftar penelitian terkait ialah sebagai berikut :

Tabel 2.5 Penelitian Terkait

No	Judul, Penulis, Tahun	Jumlah dan Atribut	Algoritma/Metode
1.	Penerapan Algoritma <i>Bubble Sort</i> pada Sistem Perpustakaan Menggunakan Framework Codeigniter [1]	Ada 2 atribut : Buku Sering Dipinjam, Prioritas Peminjaman	<i>Bubble Sort</i> <i>Waterfall</i>
2.	Rancang Bangun Aplikasi Peminjaman Ruangan Berbasis Web [7]	Ada 3 atribut : Aplikasi Peminjaman Ruangan, CodeIgniter, Jadwal Peminjaman	<i>Bubble Sort</i> <i>Prototype</i>
3.	Ubiquitous Roomie Sistem Booking Ruangan Kampus Berbasis Website [8]	Ada 3 Atribut : Peminjaman, <i>Booking Room</i> , <i>Web Application</i> .	<i>Waterfall</i>
4.	Model Aplikasi Peminjaman Ruangan Berbasis Web Pada	Ada 2 Atribut : <i>Software</i>	<i>Software Development Life</i>

No	Judul, Penulis, Tahun	Jumlah dan Atribut	Algoritma/Metode
	Tingkat Fakultas di Perguruan Tinggi [9]	<i>Development Life Cycle,</i> <i>Black box testing</i>	<i>Cycle,</i> <i>Waterfall</i>
5.	Sistem Informasi Manajemen Ruang Program Studi Teknik Informatika Menggunakan Framework Laravel [5]	Ada 3 atribut : Ruang, <i>Black Box Testing,</i> <i>Mean Opinion Score</i>	<i>Waterfall</i>

2.7 Pengujian Perangkat Lunak (*Black Box Testing*)

Black Box testing berfokus pada pengujian dari masing-masing spesifikasi fungsional perangkat lunak. Seorang tester dapat mendefinisikan kumpulan kondisi input dan melakukan pengetesan pada fungsionalitas perangkat lunak (Mustaqbal, 2015). Metode *Black Box testing* terdiri atas beberapa metode, antara lain *Equivalence Partitioning*, *Boundary Value Analysis*, *State Transition Testing*, dan *Decision Table Testing*. *Black Box Testing* merupakan metode pengujian perangkat lunak yang digunakan untuk menguji sebuah perangkat lunak tanpa mengetahui struktur internal kode atau program.

Dalam pengujiannya, penguji menyadari apa yang harus dilakukan oleh program, tapi tidak memiliki pengetahuan tentang bagaimana melakukannya. Kelebihan *black box testing* yaitu :

1. Efisien untuk segmen kode besar.
2. Akses kode tidak diperlukan
3. Pemisahan antara perspektif pengguna dan pengembang

Selain memiliki kelebihan, *black box testing* juga memiliki kelemahan, yaitu :

1. Cakupan terbatas karena hanya sebagian kecil dari skenario pengujian yang dilakukan.

2. Pengujian tidak efisien karena keberuntungan tester dari pengetahuan tentang perangkat lunak internal

