

Article Category : Logistic Management

## PERANCANGAN SISTEM INFORMASI BASIS DATA INVENTARIS BARANG BERBASIS WEB MENGGUNAKAN MODEL WATERFALL

### WEB-BASED DATABASE INFORMATION SYSTEM DESIGN USING WATERFALL METHOD

**Nuraida Wahyuni<sup>\*1)</sup>, Rizki Akmal<sup>2)</sup>, Akbar Gunawan<sup>3)</sup>**

<sup>1,2,3</sup> Jurusan Teknik Industri, Universitas Sultan Ageng Tirtayasa, Jl. Jend. Sudirman KM 3, Cilegon, 42435, Indonesia

#### ARTICLE INFORMATION

Article history:

Received: May 01, 2020  
Revised: July 13, 2020  
Accepted: Agustus 24, 2020

Keywords :  
Inventory  
Database  
Design  
Waterfall model

Kata Kunci:  
Barang inventaris  
Basis data  
Desain  
Model Waterfall

#### A B S T R A C T

Good inventory management requires support from a database information system. The problem that occurs in the Untirta Industrial Engineering laboratory is that the inventory management system is still traditional method, that is data recording is done individually so that the existing data overlaps and results in invalid information. The purpose of this research is to design a database information system. The Waterfall model is used for design. This method consists of the analysis phase, the design phase, the implementation phase, the test phase and the maintenance phase. The results obtained from the analysis phase are the functional requirements for logging in users and data engineering requirements such as add, view, delete. The design phase uses DFD and ERD. Based on the context diagram, the external agents involved are the admin (head of the laboratory) and user (laboratory assistant). The design results that have been implemented through a web-based interface are then tested using a blackbox test and usability test. The test results state a value of 86% which means the results are in very good classification.

#### A B S T R A K

Pengelolaan barang inventaris yang baik membutuhkan dukungan dari sistem informasi basis data. Masalah yang terjadi di laboratorium jurusan Teknik Industri Untirta adalah sistem pengelolaan barang inventaris masih bersifat tradisional, yaitu pencatatan yang dilakukan sendiri-sendiri sehingga data yang ada tumpang tindih dan menghasilkan informasi yang kurang valid. Tujuan dari penelitian ini adalah merancang sistem informasi basis data. Model Waterfall digunakan untuk perancangan. Metode ini terdiri dari tahap analisis, tahap desain, tahap implementasi, tahap uji dan tahap pemeliharaan. Hasil yang didapatkan dari tahap analisis adalah adanya kebutuhan fungsional untuk *log in user* dan kebutuhan rekayasa data seperti tambah, lihat, hapus. Tahap desain menggunakan DFD dan ERD. Berdasarkan diagram konteks, agen eksternal yang terlibat adalah admin (kepala laboratorium) dan user (asisten laboratorium). Hasil desain yang telah diimplementasikan melalui tampilan antar-muka berbasis web kemudian diuji menggunakan uji blackbox dan uji usability. Hasil uji menyatakan nilai sebesar 86% yang berarti hasil berada dalam klasifikasi sangat baik.

\*Corresponding Author

Name : Nuraida Wahyuni  
Email : nrdwahyuni@gmail.com

This is an open access article under the [CC-BY](https://creativecommons.org/licenses/by/4.0/) license.



© 2020 Some rights reserved

## 1. PENDAHULUAN

Data merupakan representasi dari fakta, gambar, dan ide, bentuknya bisa berupa angka, kata, atau gambar. Dengan demikian, data mengacu pada atribut kualitatif atau kuantitatif dari suatu variabel atau sekumpulan variabel. Setelah data dikumpulkan, ada pertanyaan tentang apa yang harus dilakukan dengan kelimpahan data ini; maka dari itu, terdapat kebutuhan untuk mengatur data, menyimpan data, dan membuat data tersedia saat dibutuhkan. Oleh karena itu, basis data dapat diartikan sebagai kumpulan data [1].

Pada era revolusi industri 4.0 saat ini, berbagai aktivitas yang dilakukan manusia tersimpan dalam sebagai catatan digital. Aktivitas digital saat ini sudah melekat hampir pada semua lingkungan dan bidang. Walaupun belum semua lingkungan dan bidang menerapkannya karena adanya keterbatasan komponen sistem informasi. Contoh aktivitas yang lazim dilakukan secara digital sehari-hari adalah media sosial, belanja online, internet banking, akses ke sistem informasi akademik, dan sebagainya.

Pada lingkungan pendidikan dari tingkat dasar, tingkat menengah, tingkat atas, dan tingkat tinggi sudah banyak yang menerapkan teknologi dan sistem informasi dalam kegiatannya. Khususnya pada perguruan tinggi, sistem informasi dimanfaatkan dalam hampir pada setiap kegiatan. Mulai dari penerimaan mahasiswa, pengelolaan kegiatan akademik, dan sebagainya.

Pemanfaatan teknologi dan sistem informasi di Universitas Sultan Ageng Tirtayasa (Untirta) sudah dilaksanakan dalam sepuluh tahun terakhir. Dibutuhkan beberapa tahapan dan adaptasi dalam penerapannya hingga domain [untirta.ac.id](http://untirta.ac.id) reliabel seperti saat ini. Hampir seluruh aktivitas di lingkungan Untirta dapat dilakukan melalui domain tersebut, contohnya, kegiatan akademik mahasiswa dan dosen, tugas akhir, pendaftaran wisuda dan lain-lain. Namun, masih terdapat aktivitas yang belum diakomodir secara online, biasanya aktivitas tersebut bersifat lokal pada jurusan tertentu.

Civitas akademik khususnya pada jurusan Teknik Industri Untirta dapat mengakses aktivitas-aktivitas yang telah tersedia pada domain Untirta. Akan tetapi, kegiatan inventaris barang laboratorium (lab) di jurusan Teknik Industri masih dilakukan secara tradisional dan terpisah. Terdapat empat laboratorium, dan keempatnya mempunyai catatan inventaris masing-masing

sehingga pengelolaan dan pencarian masih terbatas ruang dan waktu. Sedangkan inventaris barang laboratorium merupakan tanggung jawab kepala laboratorium yang secara periodik harus melakukan pelaporan barang dan alat yang terdapat pada setiap lab.

Barang yang masuk ke lab atau studio dapat berupa mesin, alat, komputer, maupun furnitur. Laboran masing-masing studio mencatat barang masuk. Barang rusak atau tidak dapat digunakan lagi juga masuk catatan. Pencatatan juga dilakukan saat ada alat yang dipinjam oleh mahasiswa untuk penelitian seperti stopwatch, alat ukur kelembapan, alat antropometri, dan sebagainya. Permasalahan kadang timbul adalah pencatatan dilakukan oleh masing-masing lab atau studio, namun, kepala lab harus melaporkan untuk semua studio, pencatatan jadi tumpang tindih dan tidak jarang jumlahnya menjadi doble atau kurang.

Berdasarkan uraian di atas, jurusan Teknik Industri Untirta membutuhkan sistem informasi basis data untuk mendukung kegiatan inventaris barang laboratorium. Banyak model perancangan sistem informasi khususnya model software development, seperti: model waterfall, model iterative waterfall, prototype, model v-shaped, dan model Spiral [2]. Model yang cocok digunakan untuk perancangan sistem basis data dalam penelitian ini adalah model Waterfall karena kebutuhan jelas, definisi produk jelas, teknologi dipahami, sumberdaya mudah didapat, dan merupakan proyek kecil [3].

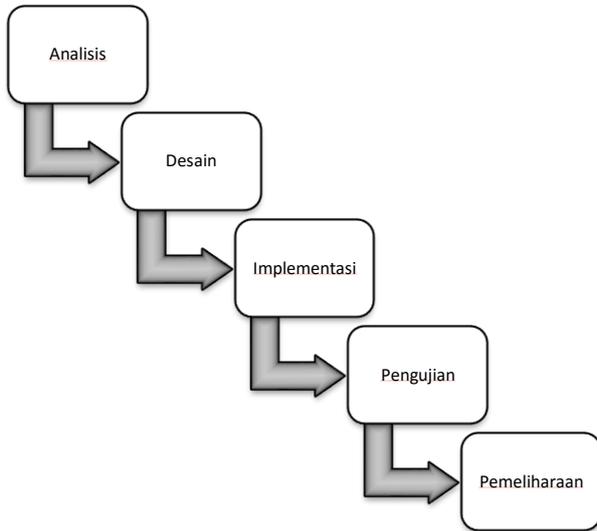
Oleh karena itu, penelitian ini bertujuan untuk merancang sebuah sistem informasi basis data inventaris barang pada laboratorium di jurusan Teknik Industri Untirta. Perancangan sistem informasi basis data menggunakan model Waterfall. Perancangan yang dilakukan berbasis web dan masih menggunakan server lokal.

## 2. METODE PENELITIAN

Metode penelitian yang dipakai diadaptasi dari metode dari model Waterfall. Model Waterfall yaitu salah satu model dalam *Software Development Life Cycle* (SDLC) yang dikemukakan oleh Winston W. Royce pada tahun 1970 untuk menggambarkan kemungkinan praktik rekayasa perangkat lunak [4]. Model Waterfall merupakan proses pengembangan perangkat lunak berurutan di mana progres pengembangan dianggap sebagai air terjun (mengalir ke bawah)

melalui fase yang harus dikerjakan agar berhasil membangun perangkat lunak [5].

Pada dasarnya, model Waterfall terdiri dari lima fase: Analisis, desain, implementasi, pengujian, dan pemeliharaan, seperti yang dapat dilihat pada [gambar 1](#) di bawah ini:



**Gambar 1.** Model Waterfall

Tahap analisis [5]: deskripsi tentang perilaku perangkat lunak yang akan dikembangkan. Melibatkan analisis sistem dan bisnis untuk menentukan persyaratan fungsional dan non-fungsional. Biasanya, persyaratan fungsional ditentukan dengan menggunakan case yang menggambarkan interaksi pengguna dengan perangkat lunak. Persyaratan non-fungsional merujuk pada berbagai kriteria, kendala, batasan, dan persyaratan yang dikenakan pada desain dan pengoperasian perangkat lunak daripada pada perilaku tertentu.

Tahap desain [6]: Tahap desain merupakan tahapan lanjut dari tahap analisis dimana dalam tahap ini disajikan desain desain dari aplikasi seperti desain antar muka, dan desain data base yang akan diterapkan ke dalam sistem Informasi.

Tahap implementasi [5]: Mengacu pada realisasi persyaratan bisnis dan spesifikasi desain ke dalam program, database, situs web, atau komponen perangkat lunak yang dapat dieksekusi melalui *programming* dan *deployment*.

Tahap pengujian [7]: Dalam pengujian perangkat lunak, kualitas didefinisikan sebagai "kesesuaian dengan persyaratan." *Use case*, persyaratan fungsional, dan persyaratan perangkat lunak lainnya menentukan *behavior* (perilaku) suatu

perangkat lunak. Tujuan pengujian perangkat lunak adalah untuk memastikan bahwa produk sesuai dengan yang diperlukan pengguna dan *stakeholder*.

Tahap pemeliharaan [8]: merupakan proses pemeliharaan sistem selama penggunaan agar tetap mampu beroperasi secara benar.

### 3. HASIL DAN PEMBAHASAN

Seperti yang sudah diuraikan di atas, bahwa tahap pertama adalah tahap analisis. Kerangka Pieces (*Performance, Information, Economics, Control, Efficiency, Service*) dipakai pada tahap ini. Kerangka Pieces adalah kerangka yang dipakai untuk mengklasifikasikan suatu masalah, kesempatan, dan petunjuk yang terdapat pada tahap analisa [9]. Melalui kerangka Pieces ini diharapkan akan mendapatkan apa saja kebutuhan fungsional dan kebutuhan non-fungsional dari sistem informasi yang akan dirancang.

Untuk mendapatkan kebutuhan tersebut, kerangka ini dimulai dengan mencari permasalahan. [Tabel 1](#) memperlihatkan permasalahan yang ada pada sistem inventaris lab pada jurusan Teknik Industri. Secara umum, persoalannya mengenai informasi barang apa saja yang terdapat pada setiap lab, karena pendataan masih bersifat tradisional yaitu dicatat pada kertas, lalu didokumentasikan di map, ada kalanya tidak terlalu *update* atau data bisa tumpang tindih. Perkara siapa yang terakhir *update* data juga tidak bisa terdeteksi karena terkadang asisten lab lupa membubuhi tanda tangan form inventaris.

**Tabel 1.** *Problem statement*

PIECES	<i>Problem statement</i>	Kode
<i>Performance</i>	Pendataan alat dan barang inventaris memerlukan waktu, tenaga, dan ruang penyimpanan.	1
<i>Information</i>	Informasi hasil pendataan disimpan terlalu banyak dan kurang terserap baik.	2
<i>Economy</i>	Penggunaan kertas yang konvensional dan memerlukan biaya	3
<i>Control</i>	Kehilangan buku catatan inventaris	4

PIECES	Problem statement	Kode
Efficiency	karena bisa diakses siapa saja	5
	Perekapan data memerlukan waktu yang lama.	
Service	Pendataan atau pencatatan alat dan barang inventaris tidak fleksibel.	6

Tabel 2 memperlihatkan hubungan sebab-akibat dari setiap permasalahan yang nanti akan menjadi landasan perencanaan kebutuhan fungsional dan kebutuhan non-fungsional. Secara umum, persoalan-persoalan tersebut berhubungan dengan waktu yang dibutuhkan untuk mencari data, data tumpang tindih, penelusuran user, karena sistem inventaris barang masih bersifat konvensional.

Tabel 2. Sebab akibat

Kode	Sebab-akibat
1	Sistem pendataan masih menggunakan cara tradisional menyebabkan waktu sinkronisasi data yang lama dan tenaga dalam pendataan alat dan barang inventaris laboratorium. Selain itu, diperlukan ruang penyimpanan yang aman (dalam binder). Pengumpulan data alat dan barang inventaris laboratorium yang dilakukan dengan cara sederhana, menyebabkan informasi data yang didapat tidak lengkap dan rentan hilang.
2	Sistem pendataan masih menggunakan cara konvensional mengharuskan pendataan menggunakan kertas atau buku catatan, hal ini menyebabkan timbulnya biaya.
3	Pendataan yang telah dikumpulkan kemudian disimpan di ruang penyimpanan, hal ini memerlukan kontrol yang baik karena menyebabkan risiko kehilangan.
4	

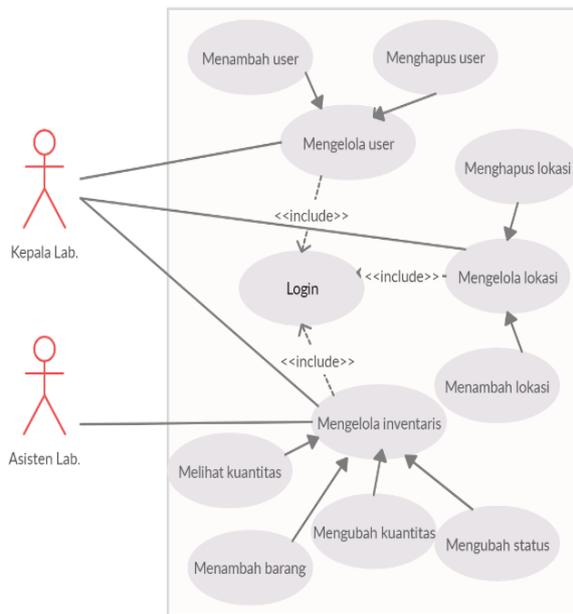
Kode	Sebab-akibat
5	Pengumpulan data alat dan barang inventaris laboratorium yang dilakukan kemudian direkap disalin ke buku catatan, hal ini membutuhkan waktu yang lama sehingga tidak efisien.
6	Sistem pendataan masih menggunakan cara lama yang tidak fleksibel dan tidak bisa dilakukan di mana saja dan kapan saja sehingga diharuskan ke laboratorium untuk pendataan alat dan barang inventaris.

Tabel 3 di bawah ini merupakan simpulan dari tabel sebelumnya. Pemakai atau user dari sistem informasi nanti harus terdeteksi, maka dibutuhkan sistem log in user. Lalu kebutuhan untuk merekayasa data seperti input, lihat, dan tambah data juga sangat dibutuhkan. Kebutuhan non-fungsional dibutuhkan agar sistem informasi mudah digunakan, desain yang menarik akan membantu memudahkan pengoperasiannya. Kebutuhan log in user didapatkan dari masalah nomor 4, berdasarkan kerangka Pieces, masalah tersebut berada pada kerangka control. Menurut [10], kerangka control berkaitan dengan keamanan sistem dan hak akses pengguna.

Tabel 3. Kebutuhan fungsional dan non-fungsional

Kebutuhan fungsional	Kebutuhan non-fungsional
Log in user (4)	Desain Sistem yang Menarik
Input data (1, 2, 3)	Mudah Digunakan
Lihat data (1, 2, 5)	
Tambah data (6)	

Gambar 2 memperlihatkan perkiraan user yang dapat menggunakan sistem informasi basis data dan otoritasnya dalam sistem. Terdapat dua user yang direncanakan dapat mengakses situs yang dirancang, yaitu Kepala Lab dan Asisten Lab menjawab kebutuhan dari log in user pada Tabel 3. Kemudian kebutuhan lainnya yaitu input data, lihat data dan tambah data, digambarkan pada Gambar 2. Kepala Lab mempunyai otoritas lebih banyak daripada Asistan Lab.



**Gambar 2.** Use case diagram

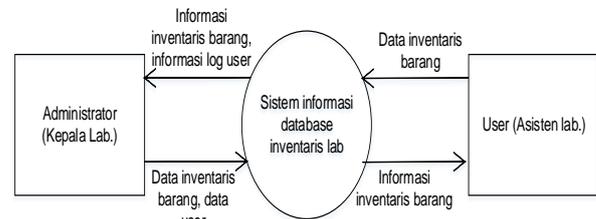
Tahap selanjutnya adalah tahap Desain. Pada tahap ini Data Flow Diagram (DFD) digunakan untuk menjelaskan aliran data yang terjadi pada sistem informasi database inventaris barang lab. DFD banyak digunakan selama fase analisis untuk menangkap persyaratan sistem apa pun. Aturan untuk DFD bersifat sederhana dan tidak rumit.

Ada empat simbol yang digunakan untuk mewakili persyaratan sistem, yaitu agen eksternal, aliran data, penyimpanan dan proses data [11]. Pada [Gambar 2](#), simbol yang digunakan untuk mewakili agen eksternal adalah persegi panjang. Agen eksternal bertindak sebagai sumber untuk aliran data. Proses ditunjukkan oleh simbol lingkaran yang merupakan transformasi data dalam sistem. Aliran data menunjukkan aliran informasi dari sumbernya ke tujuan yang diwakili oleh panah.

Data yang disimpan bertindak sebagai basis data untuk menyimpan informasi untuk sistem. Keuntungan menggunakan DFD adalah pemahaman yang kuat mengenai aliran data dan mudah untuk dikomputerisasi [12].

Sebelumnya, dijelaskan kebutuhan fungsional adalah log in user, kebutuhan rekayasa, seperti input, tambah, dan lihat. Maka, data yang mengalir akan menyesuaikan dengan kebutuhan tersebut. [Gambar 3](#) di bawah ini merupakan diagram konteks untuk rancangan sistem informasi ini. Agen eksternal dirancang ada dua, yaitu kepala lab sebagai administrator dan Asisten lab sebagai user. Perbedaannya pada aliran data,

untuk kepala lab bisa mengatur data inventaris barang dan data user, sementara untuk asisten lab hanya bisa mengatur data inventaris barang saja.



**Gambar 3 .** Diagram konteks

Penjelasan dari diagram konteks adalah diagram level 0. Diagram 0 digunakan terutama sebagai alat presentasi yang berguna untuk meringkas seluruh sistem atau subsistem secara lebih rinci daripada diagram konteks [11]. Pada level ini, sistem informasi inventaris lab dijelaskan ke dalam lima proses, yaitu proses *log in*, *manage inventory*, *manage location*, *manage user*, *system log*, dan proses *notification*. Dari proses-proses tersebut melibatkan data store seperti data *inventory*, data *location* dan data lainnya yang bisa dilihat di [Gambar 4](#).

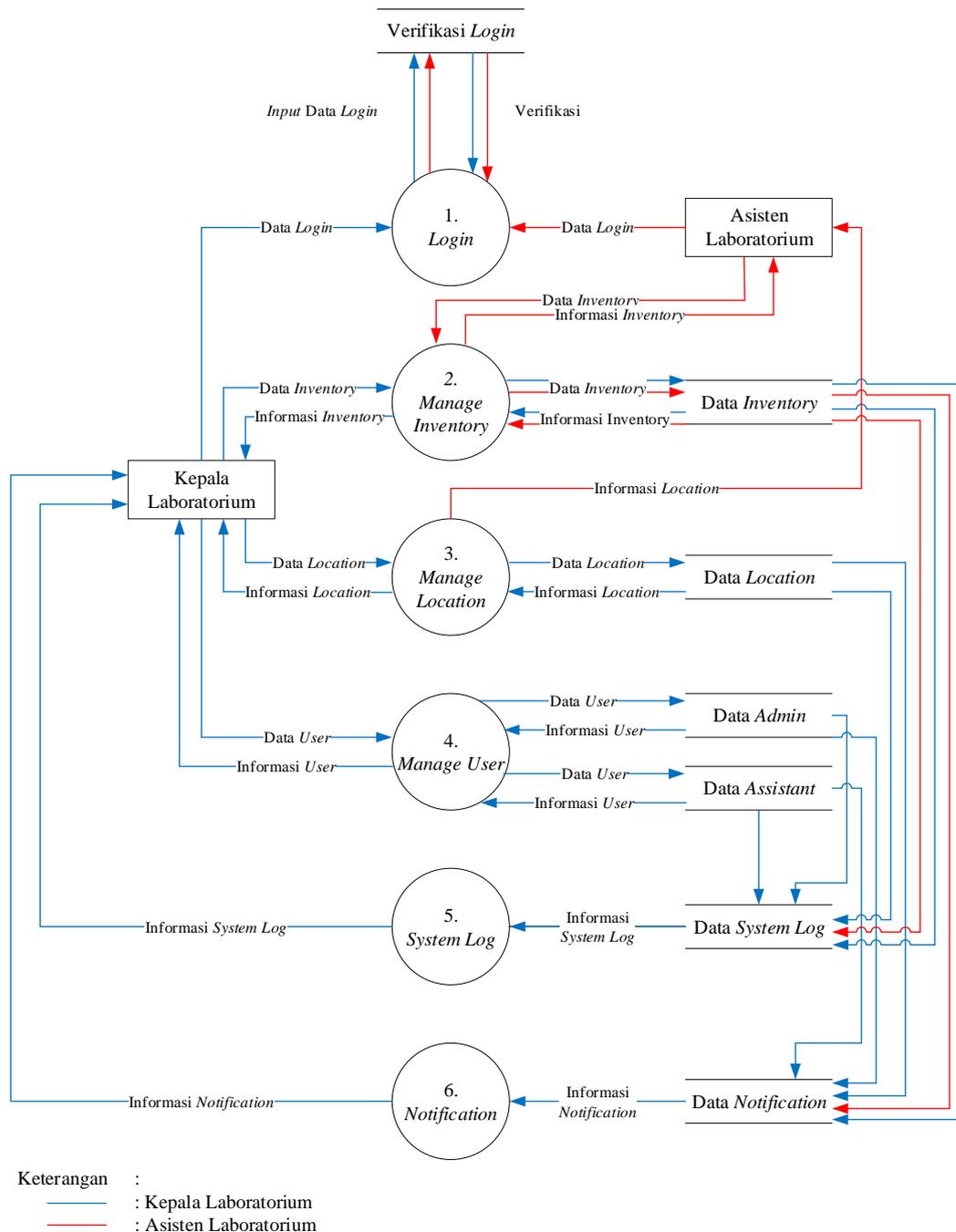
Beberapa fragmen DFD melibatkan banyak pemrosesan yang perlu dijelajahi analisis secara lebih rinci, seperti halnya langkah pemodelan, dekomposisi lebih lanjut membantu analisis mempelajari lebih lanjut tentang persyaratan sambil juga menghasilkan dokumentasi yang diperlukan [11]. [Gambar 5](#), [Gambar 6](#), [Gambar 7](#), dan [Gambar 8](#) merupakan contoh fragmen penjabaran dari [Gambar 4](#), yang merupakan DFD level 1.

Menurut [11] penyimpanan data di DFD mewakili entitas data pada *Entity Relationship Diagram* (ERD). Jika penyimpanan data tidak ditautkan ke ERD, analisis hanya mendefinisikan penyimpanan data sebagai kumpulan elemen (mungkin dengan struktur) dengan cara yang sama seperti aliran data didefinisikan [11]. Pada penelitian ini, diperlukan ERD untuk melihat relasi antar *data store* (entitas) hasil dari DFD.

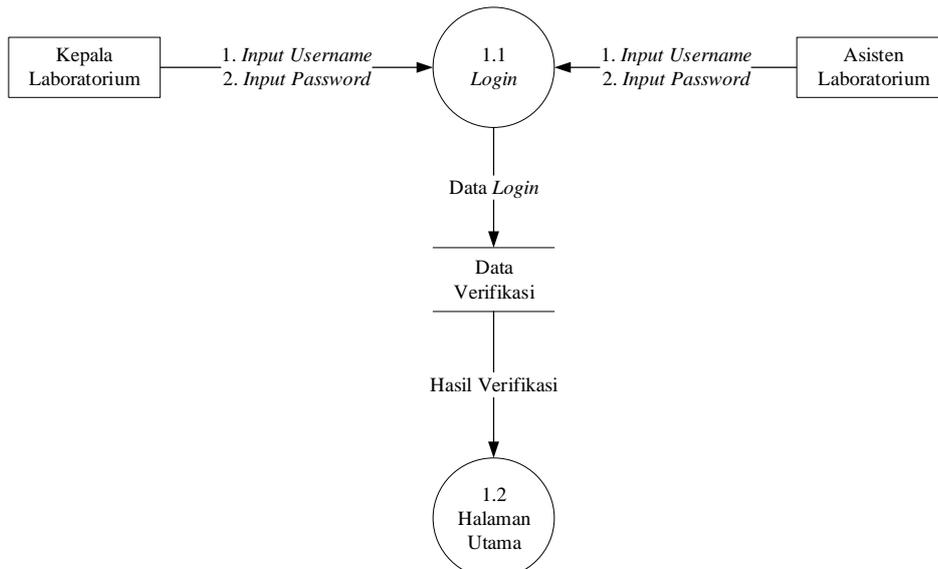
ERD menggambarkan item data (entitas) yang digunakan sistem, dan hubungan di antara mereka dan menyatakan diagram transisi keadaan atau perilaku yang tergantung waktu dari sistem, dan menunjukkan status bahwa sistem dapat berada dan koneksi yang valid antara berbagai kondisi [13]. [Gambar 9](#) di bawah ini merupakan ERD dari hasil aliran data pada DFD sebelumnya. Selain agen eksternal, entitas lain yang terlibat adalah entitas *inventory*, dan

location. Entitas *inventory location* merupakan entitas relasi antara entitas *inventory* dan entitas *location* yang akan menghasilkan informasi lokasi keberadaan barang inventaris. Skema basis data merupakan implementasi dari rancangan ERD yang telah dibuat. Gambar 10 memperlihatkan skema basis data dalam rancangan ini. Pada [Gambar 10](#) juga

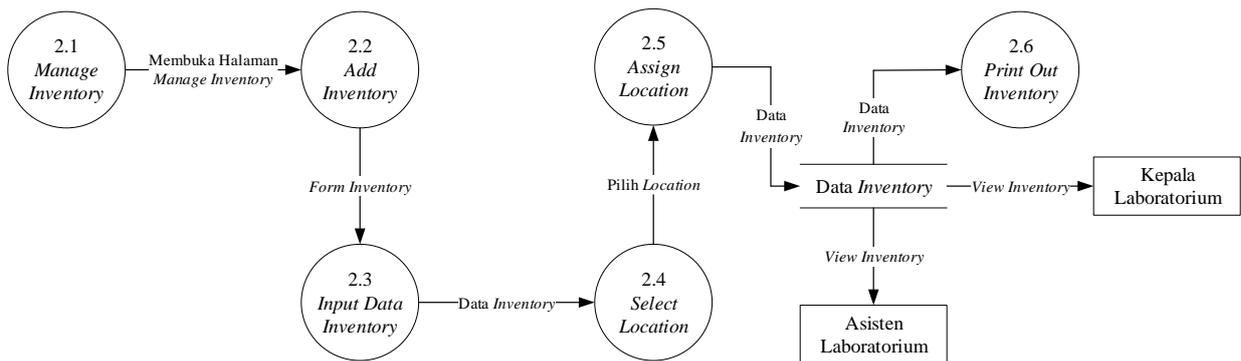
memperlihatkan tipe data pada setiap atribut. Baik Gambar 9 dan [Gambar 10](#), sama-sama memperlihatkan tingkat relasi. Hubungan banyak-ke-banyak lebih umum, tetapi biasanya diselesaikan dengan membuat entitas 'tautan' baru yang merupakan detail dari dua entitas asli [13].



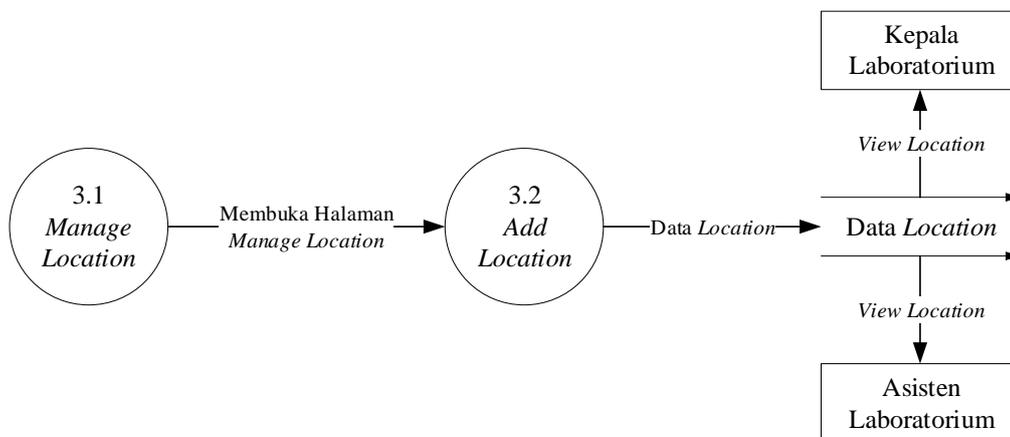
Gambar 4. DFD level 0



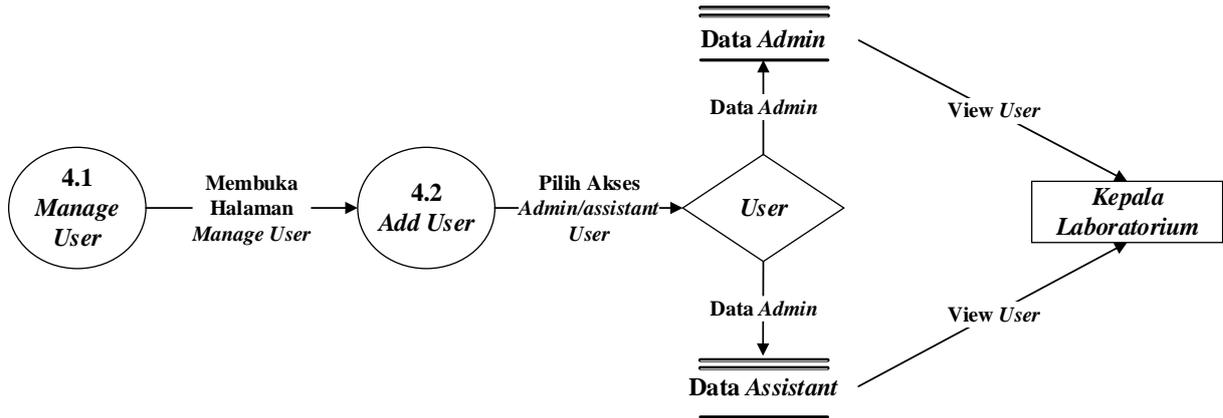
Gambar 5. DFD level 1 proses log in



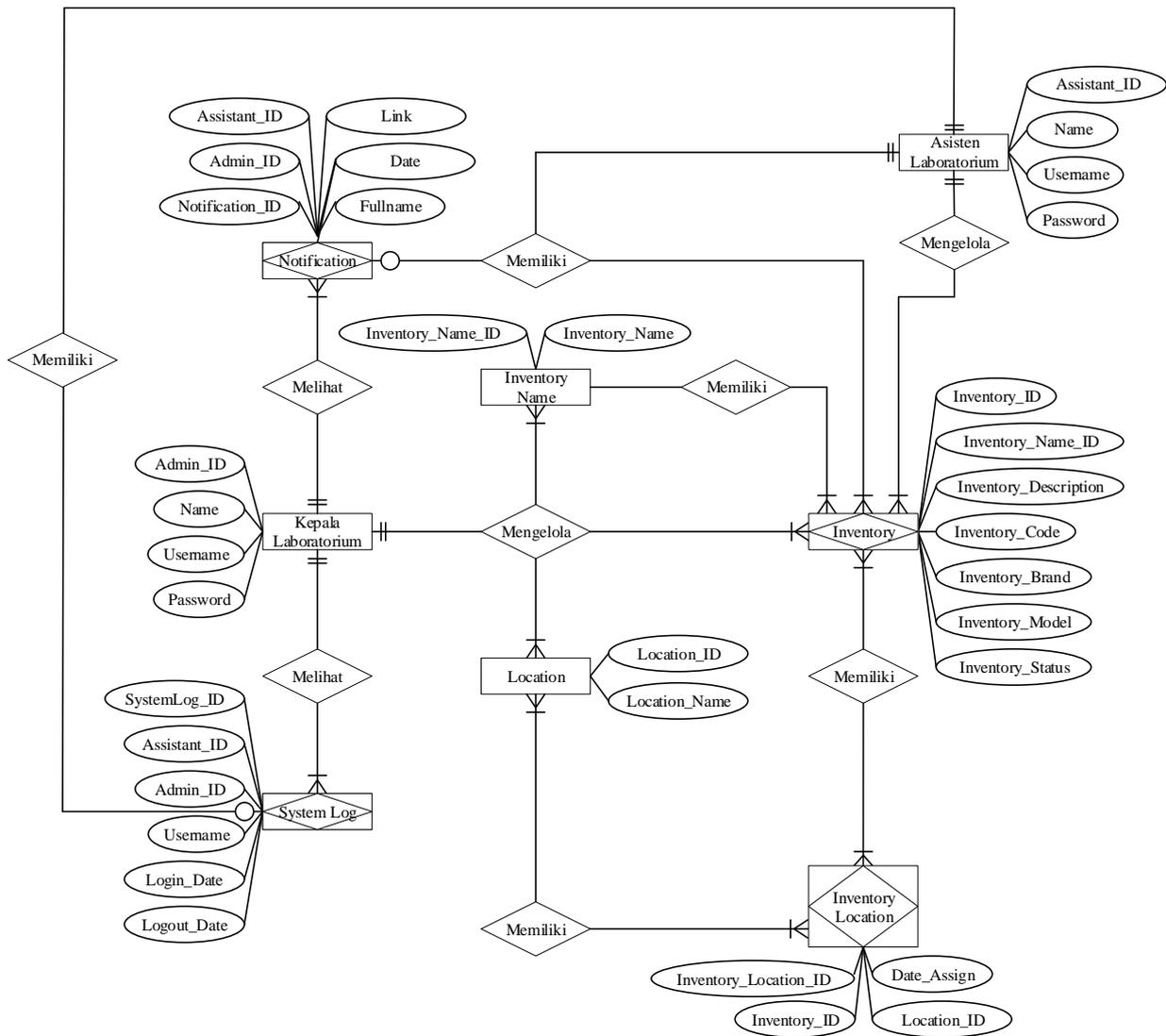
Gambar 6. DFD level 1 proses manage inventory



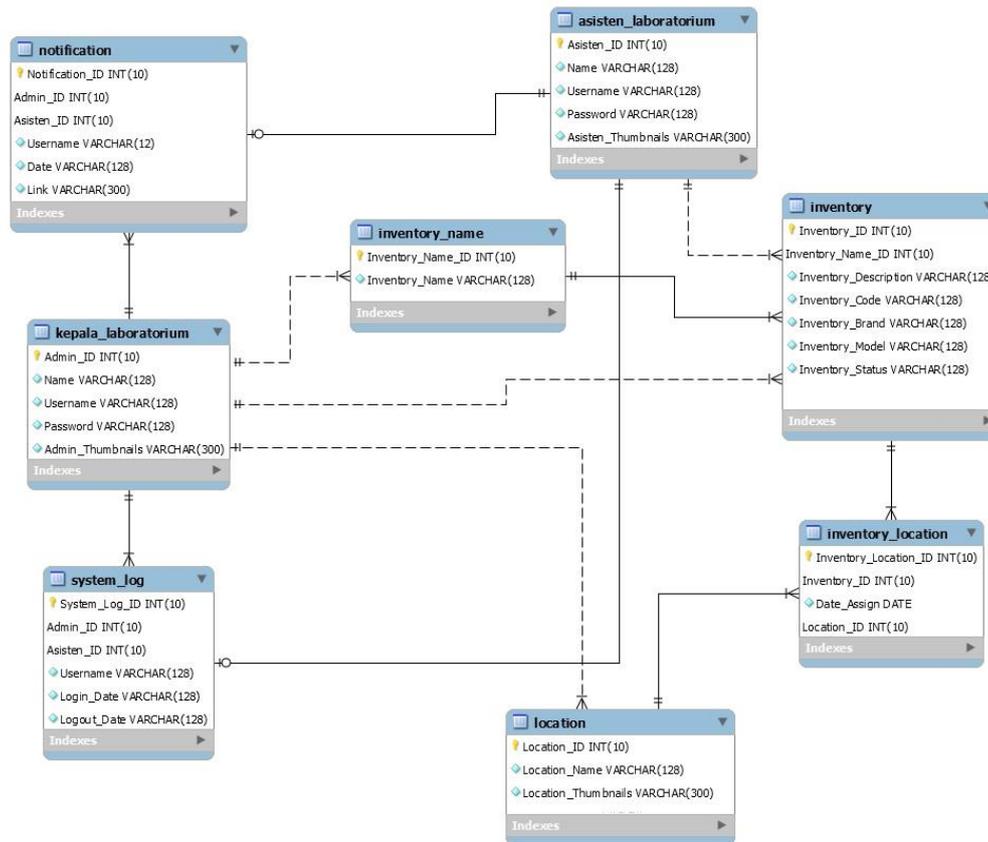
Gambar 7. DFD level 1 proses manage location



Gambar 8. DFD level 1 proses *manage user*



Gambar 9. ERD

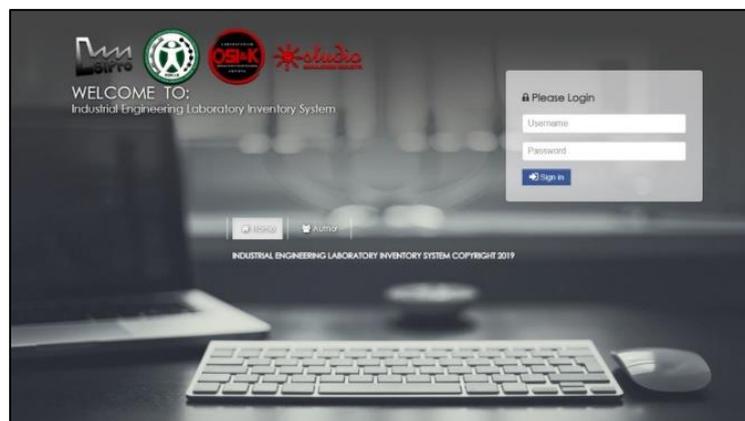


Gambar 10. Skema basis data

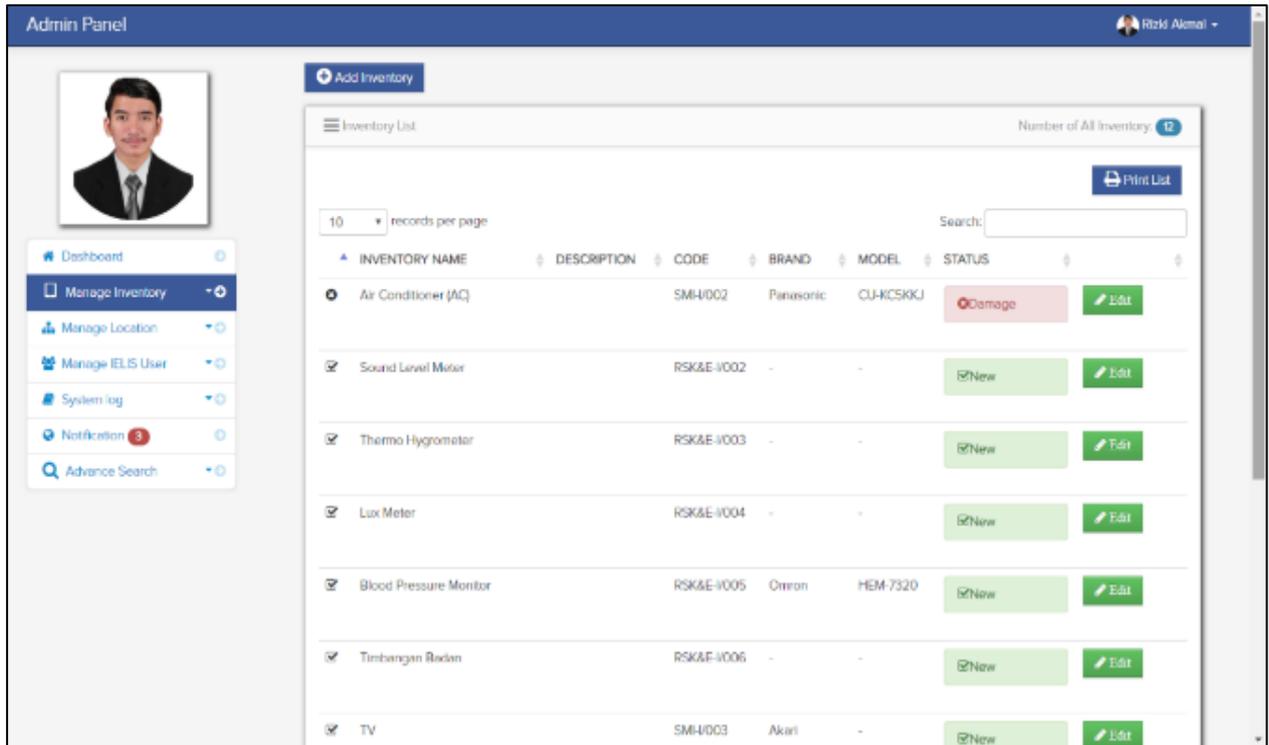
Tahap berikutnya dari model Waterfall adalah tahap implementasi. Pada tahap ini desain data yang sudah dirancang kemudian dipakai sehingga bisa ditampilkan pada desain antar-muka seperti yang terlihat pada Gambar 11, Gambar 12, Gambar 13, dan Gambar 14.

Desain antar-muka menggunakan pemrograman PHP dan Mysql. Seperti yang telah diuraikan sebelumnya, kebutuhan fungsional yang telah ditetapkan kemudian diimplementasikan, seperti

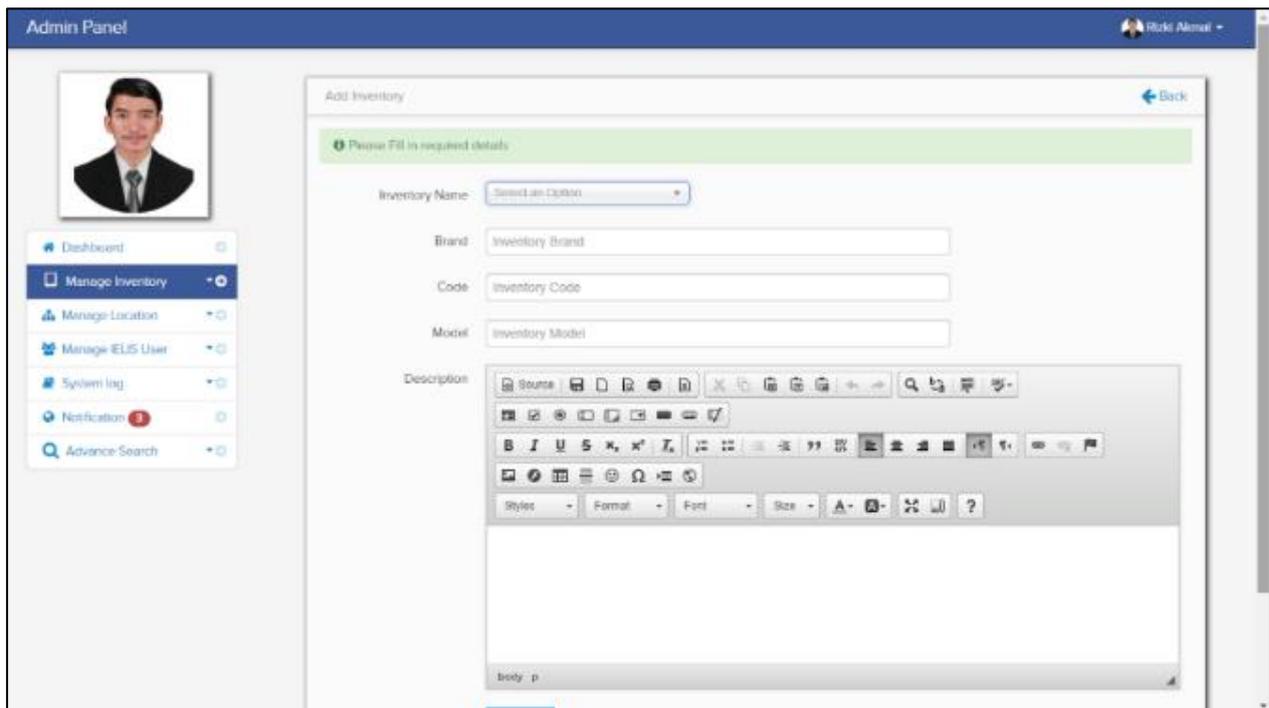
pada Gambar 11, yang merupakan hasil implementasi kebutuhan control pada kerangka Pieces. Dengan adanya fungsi ini, diharapkan aktivitas dapat tercatat dalam system log. Kebutuhan lain adalah merekayasa inventory dapat ditampilkan pada Gambar 12 dan 13. Pada halaman ini, pengguna dapat melakukan proses tambah data inventory maupun update jumlah inventory yang sudah terdaftar. Gambar 14 merupakan implementasi kebutuhan performance pada kerangka Pieces.



Gambar 11. Tampilan halaman log in



Gambar 12. Tampilan halaman *manage inventory*



Gambar 13. *Add inventory*

6/14/2019		Industrial Engineering Laboratory Inventory System			
					
<b>INDUSTRIAL ENGINEERING LABORATORY INVENTORY SYSTEM</b>					
LOCATION: Studio Manajemen Industri					
DATE: Thursday, June 13th, 2019					
INVENTORY CODE	TYPE	MODEL	BRAND	DATE ASSIGNED	STATUS
SMI-I/003	TV	-	Akari	2019-05-20	New
SMI-I/004	Printer	L120	Epson	2019-05-20	New
SMI-I/002	Air Conditioner (AC)	CU-KC5KKJ	Panasonic	2019-05-19	Damage
SMI-I/001	Air Conditioner (AC)	CU-KC5KKJ	Panasonic	2019-05-17	Used
Kepala Laboratorium			Ketua Jurusan		
<b>Ade Irman Saeful M. S., S.T., M.T.</b> NIP. 198206152012121002			<b>Putro Ferro Ferdinant, S.T., M.T.</b> NIP. 198103042008121001		

**Gambar 14.** Laporan daftar *inventory*

Tahap selanjutnya adalah pengujian. Pengujian dilakukan dengan uji *blackbox* dan uji *usability*. Uji *blackbox* adalah metode pengujian perangkat lunak yang menganalisis fungsionalitas perangkat lunak dengan membandingkan nilai input dengan nilai output [14]. Hasil uji *blackbox* memperlihatkan perancangan sistem informasi ini dapat diterima. Uji *usability* dilakukan untuk mengetahui apakah sistem informasi yang sudah dirancang dapat digunakan oleh *user*. Uji dilakukan dengan cara memberikan kesempatan pada *user* untuk menggunakan sistem informasi yang telah dirancang dan menuliskan pengalaman menggunakan sistem informasi tersebut berdasarkan pertanyaan yang telah disediakan, pertanyaan disusun berdasarkan dimensi *learnability*, *effectiveness*, *memorability*, *error*, dan *satisfaction*. *User* akan menilai pengalamannya menggunakan skala likert 1

(sangat tidak setuju), 2 (tidak setuju), 3 (agak setuju), 4 (setuju) dan 5 (sangat setuju).

Uji dilakukan pada 10 responden dengan 11 pertanyaan. [Tabel 4](#) di bawah ini merupakan daftar pertanyaan pada uji *usability*. Rekapitulasi jawaban dari responden dapat dilihat pada [Tabel 5](#). Dari tabel tersebut, didapatkan skor total adalah 473. Untuk mengetahui skor *usability*, kemudian hasil tersebut dibandingkan dengan skor tertinggi yang diharapkan dengan cara mengalikan antara nilai tertinggi kuesioner dikali jumlah pertanyaan dikali jumlah responden. Nilai tertinggi kuesioner adalah 5, maka skor yang diharapkan adalah 550. Oleh karena itu, nilai yang didapat yaitu 473 dibagi 550 dihasilkan nilai 86%. Hasil dari uji *usability* sebesar 86% yang berarti nilai kualifikasi menunjukkan sangat baik (*excellent*) seperti yang dapat dilihat pada [Gambar 15](#).

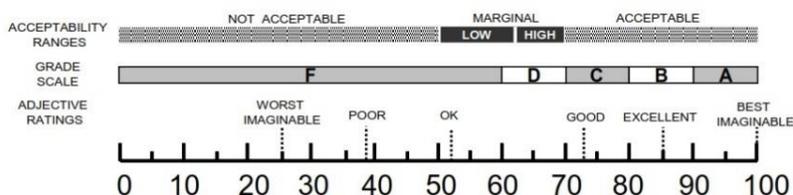
**Tabel 4.** Daftar pertanyaan uji *usability*

No	Pertanyaan
1	Apakah tampilan <i>web</i> ini menarik?
2	Apakah <i>web</i> ini mudah dipahami?
3	Apakah <i>web</i> ini mudah diakses?
4	Apakah <i>web</i> ini mudah digunakan?

No	Pertanyaan
5	Apakah <i>web</i> menyediakan informasi yang detail?
6	Apakah <i>web</i> sesuai fungsinya?
7	Apakah <i>web</i> ini membantu meringankan pekerjaan?
8	Apakah anda merasakan bekerja dengan program <i>web</i> bisa lebih cepat?
9	Apakah <i>web</i> yang telah dibuat dapat mendukung pengolahan <i>database</i> inventaris laboratorium?
10	Apakah penggunaan warna <i>web</i> sudah sesuai?
11	Apakah penggunaan <i>font</i> atau tulisan <i>web</i> sudah sesuai?

Tabel 5. Rekapitulasi jawaban responden

Responden	Pertanyaan											Total
	1	2	3	4	5	6	7	8	9	10	11	
1	4	4	4	4	3	4	5	5	5	4	4	46
2	4	5	5	5	4	5	5	5	4	5	5	52
3	5	4	4	4	4	4	5	5	4	3	4	46
4	4	4	5	4	5	4	4	4	5	4	4	47
5	3	4	4	4	5	4	4	4	5	4	4	45
6	5	5	4	4	3	5	5	5	4	4	4	48
7	4	5	4	4	4	4	5	5	4	4	4	47
8	3	4	4	5	4	5	5	4	5	4	4	47
9	3	4	4	5	5	5	4	4	5	4	4	47
10	4	5	4	4	5	4	5	4	5	4	4	48
Total	39	44	42	43	42	44	47	45	46	40	41	473



Gambar 15. Skor penilaian *usability* [15]

#### 4. KESIMPULAN

Permasalahan mengenai inventaris barang pada lab di jurusan Teknik Industri adalah pendataan masih menggunakan cara yang bersifat tradisional, yaitu pencatatan yang dilakukan sendiri-sendiri pada tiap lab sehingga menyebabkan ketersediaan informasi tumpang tindih dan tidak sinkron, sehingga membutuhkan sistem informasi *database* untuk memudahkan aktivitas tersebut. Desain sistem informasi yang diusulkan menggunakan model Waterfall. Di mana pada tahap Analisa, didapat kebutuhan fungsional dan kebutuhan non-fungsional untuk

desain ini berupa *log in user* dan kebutuhan rekayasa data *inventory*. Desain data dilakukan menggunakan DFD terlihat pada diagram konteks. Agen eksternal yang terlibat adalah kepala lab dan asisten lab. Kedua agen eksternal ini dapat merekayasa data, namun hanya kepala lab yang dapat merekayasa *user*. Tampilan desain sistem informasi dapat dilihat pada Gambar 11, Gambar 12, Gambar 13 dan Gambar 14. Hasil pengujian *usability* didapatkan 86% yang berarti nilai yang sangat baik dan dapat digunakan untuk dipakai di jurusan Teknik Industri. Arah penelitian selanjutnya adalah evaluasi penggunaan sistem informasi yang telah

dirancang sebagai bagian dari penerapan siklus PDCA (*Plan Do Check Action*).

## 5. DAFTAR PUSTAKA

- [1] K. L. Berg, T. Seymour, and R. Goel, "History Of Databases," *Int. J. Manag. Inf. Syst.*, vol. 17, no. 1, pp. 29–36, 2012, doi: 10.19030/ijmis.v17i1.7587.
- [2] M. A. Rather and V. Bhatnagar, "A comprative study of sdlc model," no. October 2015, 2016.
- [3] M. STOICA, M. MIRCEA, and B. GHILIC-MICU, "Software Development: Agile vs. Traditional," *Inform. Econ.*, vol. 17, no. 4/2013, pp. 64–76, 2013, doi: 10.12948/issn14531305/17.4.2013.06.
- [4] D. W. W. Royce, "Managing the Development of large Software Systems," *IEEE Wescon*, no. August, pp. 1–9, 1970.
- [5] S. T. ind, Karambir, "A Simulation Model for the Spiral Software Development Life Cycle," *Int. J. Innov. Res. Comput. Commun. Eng.*, vol. 03, no. 05, pp. 3823–3830, 2015, doi: 10.15680/ijirccce.2015.0305013.
- [6] Y. Firmansyah and U. Udi, "Penerapan Metode SDLC Waterfall Dalam Pembuatan Sistem Informasi Akademik Berbasis Web Studi Kasus Pondok Pesantren Al-Habib Sholeh Kabupaten Kubu Raya, Kalimantan Barat," *J. Teknol. dan Manaj. Inform.*, vol. 4, no. 1, 2017, doi: 10.26905/jtmi.v4i1.1605.
- [7] A. Stellman and J. Greene, "Applied Software Project Management: Estimation," pp. 1–14, 2014.
- [8] R. Inggi, B. Sugiantoro, and Y. Prayudi, "Penerapan System Development Life Cycle ( Sdlc ) Dalam ( Sdlc ) Dalam Mengembangkan," no. December, 2018, doi: 10.5281/zenodo.2528444.
- [9] R. Tullah and M. I. Hanafri, "Evaluasi Penerapan Sistem Informasi Pada Politeknik LP3I Jakarta Dengan Metode Pieces," *J. Sisfotek Glob.*, vol. 4, no. 1, pp. 22–28, 2014.
- [10] J. Anfa and D. Chalidyanto, "Evaluasi Kinerja Billingsystem Rawat Inap Menggunakan Kerangka Pieces," *J. Adm. Kesehat. Indones.*, vol. 4, no. 1, p. 18, 2016, doi: 10.20473/jaki.v4i1.2016.18-27.
- [11] J. Satzinger, *Systems Analysis and Design in a Changing Word Fifth Edition*. 2010.
- [12] M. A. B. M. Hussain, A. S. Baharudin, and K. Karkonasasi, "USM internship and career portal," *Int. J. Appl. Eng. Res.*, vol. 11, no. 20, pp. 10247–10251, 2016.
- [13] D. Yeates and T. Wakefield, *Systems Analysis and Design 2nd Edition*. 2004.
- [14] T. S. Jaya, "Pengujian Aplikasi dengan Metode Blackbox Testing Boundary Value Analysis (Studi Kasus: Kantor Digital Politeknik Negeri Lampung)," *J. Inform. Pengemb. IT*, vol. 3, no. 2, pp. 45–46, 2018, doi: 10.30591/jpit.v3i1.647.
- [15] A. Bangor, T. Staff, P. Kortum, J. Miller, and T. Staff, "Determining what individual SUS scores mean: adding an adjective rating scale," *Determ. what Individ. SUS scores mean adding an adjective Rat. scale*, vol. 4, no. 3, pp. 114–123, 2009.

## Biografi Penulis

### Nuraida Wahyuni



Penulis merupakan dosen pada jurusan Teknik Industri Universitas Sultan Ageng Tirtayasa, bidang keilmuan manajemen industri. Bekerja sebagai dosen dari Desember 2008.

### Rizki Akmal



Penulis merupakan alumni jurusan Teknik Industri Universitas Sultan Ageng Tirtayasa, bidang keilmuan manajemen industri. Saat ini penulis bekerja sebagai cost control pada PT. Wilmar International.

### Akbar Gunawan



Penulis merupakan dosen pada jurusan Teknik Industri Universitas Sultan Ageng Tirtayasa, Bidang keilmuan manajemen industri. Bekerja sebagai dosen dari Desember 2014.