

# **RANCANG BANGUN SISTEM APLIKASI PENGOLAH DATA HASIL PENGUJIAN *TIRE ARCHITECTURE* DAN *PHYSICAL PROPERTIES* BERBASIS WEB**

Martina Dwi Ayu Sukma Wardani<sup>1)</sup>  
Program Studi Teknik Elektronika, Politeknik Gajah Tunggal  
martinadwiayu34@gmail.com

Moni Della Putri Saksono<sup>2)</sup>  
Program Studi Teknik Elektronika, Politeknik Gajah Tunggal  
monidellaputri18@gmail.com

Kurniawan Hartanto<sup>3)</sup>  
Politeknik Gajah Tunggal  
kurniawan@poltek-gt.ac.id

## **ABSTRAK**

Laboratorium *Tire Architecture* merupakan bagian dari Departemen QA yang menguji struktur dan kualitas finish product yang di produksi. Banyaknya pengujian yang perlu dilakukan untuk menjamin kualitas produk membuat penyimpanan data yang tidak efisien dan sulit ditemukan. Saat ini, penyimpanan data menggunakan aplikasi Microsoft Excel yang diunggah di Intranet. Namun penyimpanan ini masih dinilai kurang efektif karena sulit mencari history berkas dari tahun ke tahun, penyimpanan berkas kurang rapi, pengelolaan report butuh waktu lama, file penyimpanan membutuhkan memori besar. Rancang Bangun sistem aplikasi ini bertujuan untuk membantu dalam penyimpanan berkas-berkas agar mudah ditemukan, membuat tampilan data menjadi lebih mudah dimengerti, dan mempermudah dalam pembuatan report mingguan ataupun bulanan. Selain itu juga di maksudkan untuk mempermudah pengimputan data yang akan disimpan. Aplikasi ini berbasis web dan database MySQL sehingga memungkinkan untuk penyimpanan berkas yang besar dan dapat diakses dimana saja.

Kata Kunci: *Tire Architecture, Physical properties, Database, MySQL, Web.*

## I. PENDAHULUAN

Divisi QA bertugas menjamin mutu/kualitas produk yang dihasilkan oleh suatu perusahaan dan merupakan bagian yang menjalankan fungsi monitoring, uji tes, dan memeriksa semua proses yang terlibat dalam produksi. Divisi QA memiliki 3 Laboratorium Pengujian, yaitu: Laboratorium *Inprocess control*, Laboratorium *Tire Architecture*, Laboratorium *Tire Performance*.

Laboratorium *Tire Architecture* melakukan pengecekan terhadap *Finish Product*. Pengetesan pada *Tire* disebut dengan *Cut Section*, sedangkan pengecekan pada *Tube* disebut dengan *Physical Properties*. Item-item yang di cek pada *Cut Section* berupa *Hardness*, *Tension Test*, *Peeling Test*, *Bead Pull Out* dan lain-lain. Sedangkan pada *Tube* berupa *Thickness*, *Tube Width*, *Valve Base*, *Ageing* dan lain-lain.

Banyaknya pengujian yang perlu dilakukan untuk menjamin kualitas produk membuat penyimpanan data yang tidak efisien dan sulit ditemukan. Saat ini, penyimpanan data menggunakan aplikasi Microsoft Excel yang diunggah di Intranet. Namun penyimpanan ini masih dinilai kurang efektif karena:

1. Sulit mencari *history* data pengujian dari tahun ke tahun.
2. Analisa butuh waktu lama.
3. Penyimpanan data dalam bentuk Microsoft Excel.
4. *Database spec* belum terintegrasi dengan baik.
5. Ruang penyimpanan *sofffile* membutuhkan *memory* yang besar.

Berikut merupakan *Root Cause Analysis* untuk menentukan penyebab masalah yang ada:

Tabel 6. *Root Cause Analysis* pada Masalah 1

Masalah 1	Penyimpanan Berkas Kurang Efisien
	<b>Sebab</b>
Kenapa?	Berkas sulit ditemukan
Kenapa?	Penyimpanan berdasarkan ukuran dan tipe
Kenapa?	Berkas terlalu banyak

Tabel 1. Lanjutan

Kenapa?	Tidak ada fitur pencarian
Akar Masalah	<b>Tidak ada fitur pencarian</b>

Tabel 7. *Root Cause Analysis* pada Masalah 2

Masalah 2	Sulit melihat <i>history</i> berkas per tahun
	<b>Sebab</b>
Kenapa?	Grafik yang ditampilkan berdasarkan <i>folder</i> per ukuran dan

	tipe
Kenapa?	Masih menggunakan <i>database</i> Microsoft Excel
Kenapa?	Tidak ada aplikasi pendukung
Akar Masalah	<b>Tidak ada aplikasi pendukung</b>

Tabel 8. *Root Cause Analysis* pada Masalah 3

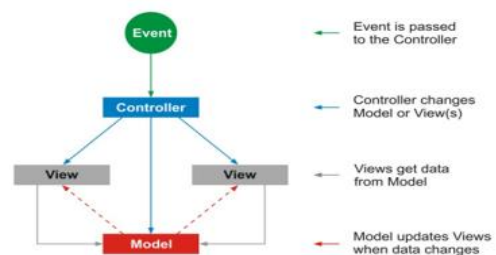
Masalah 3	Data yang sudah di <i>input</i> menjadi NG
	<b>Sebab</b>
Kenapa?	<i>Update Spec</i>
Kenapa?	<i>Spec</i> belum terintegrasi dengan baik
Kenapa?	Tidak ada aplikasi pendukung
Akar Masalah	<b>Tidak ada aplikasi pendukung</b>

Di era modern saat ini sudah memasuki tahap Revolusi Industri 4.0. Konsep penerapannya berpusat pada otomatisasi. Dibantu teknologi informasi dalam proses pengaplikasiannya, keterlibatan tenaga manusia dalam prosesnya dapat berkurang. Dengan demikian, efektivitas dan efisiensi pada suatu lingkungan kerja dengan sendirinya bertambah. Dalam Revolusi Industri 4.0, setidaknya ada lima teknologi yang menjadi pilar utama dalam mengembangkan sebuah industri siap digital, yaitu: *Internet of Things*, *Big Data*, *Artificial Intelligence*, *Cloud Computing* dan *Additive Manufacturing*. Maka dari itu merancang bangun sebuah sistem aplikasi pengolahan data hasil pengujian *Tire Architecture* dan *Physical Properties* menjadi solusi modern yang mendukung perkembangan teknologi untuk masalah yang dijabarkan pada tabel diatas.

## II. LANDASANTEORI

### *Model View Controller*

*Model View Controller* merupakan sebuah pola yang digunakan dalam pengembangan aplikasi. Tanpa adanya MVC *developer* akan cenderung mencampuradukan kode logika dengan kode tampilan serta kode untuk mengambil data dari *database* (Suprihadi, dkk. 2013).



Gambar 1. *Model View Controller*

**Laravel**

Laravel merupakan *framework* php yang memiliki teknologi terbaru diantara *framework* PHP yang lainnya. Laravel berfokus pada pengguna akhir, yang berarti berfokus pada kejelasan dan kesederhanaan, baik secara tertulis maupun penampilan. Serta menghasilkan fungsionalitas aplikasi web yang berfungsi sebagaimana mestinya. Selain itu, Laravel juga menyederhanakan proses pengembangan situs web dengan bantuan beberapa fitur unggulan, seperti *Template Engine*, *Routing*, dan *Modularity* (Saputra, 2020).

**Database**

*Database Server* adalah sebuah program komputer yang menyediakan layanan pengelolaan basis data dan melayani komputer atau program aplikasi basis data yang menggunakan model klien/server. Istilah ini juga merujuk kepada sebuah komputer (umumnya merupakan server) yang didedikasikan untuk menjalankan program yang bersangkutan. Sistem manajemen basis data (SMBD) pada umumnya menyediakan fungsi-fungsi server basis data, dan beberapa SMBD (seperti halnya MySQL atau Microsoft SQL Server) sangat bergantung kepada model klien-server untuk mengakses basis datanya (Fertian, 2016).

**Agile Software Development**

*Agile Software Development* adalah sebuah grup metodologi-metodologi pengembangan *software* yang berbasis pada pengembangan iteratif dimana keperluan dan solusi pengembangan *software* berubah terus menerus. *Agile Software Development* terdiri dari beberapa model pengembangan seperti SCRUM dan *Extreme Programming* (XP) yang cukup terkenal di kalangan para pengembang berbasis *Agile*. XP memiliki 4 nilai dasar yang menjadi inti pokok metode XP yaitu: *Communication* (Komunikasi), *Simplicity* (Kesederhanaan), *Feedback* (Umpan Balik), dan *Courage* (Keberanian). Keempat nilai dasar ini menunjukkan bahwa XP bersifat fleksibel terhadap perubahan-perubahan yang diminta oleh klien. (Nugraha, 2010).

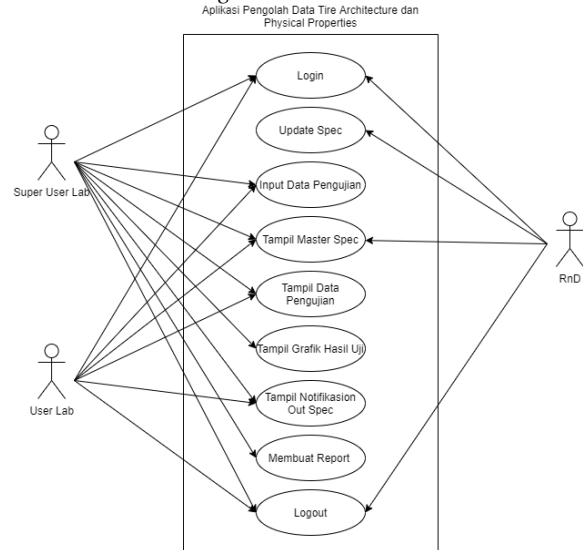
**III. METODE PENELITIAN**

Dalam penelitian ini penulis menggunakan metode Agile dalam pembuatan aplikasi. Metode ini dipilih karena sifatnya yang fleksibel terhadap perubahan-perubahan yang diminta oleh klien. Pendekatan yang dipilih penulis dalam metode Agile adalah *Extreme Programming* (XP). Pendekatan metodologi XP pada tahapan perancangan dengan menggunakan pendekatan metodologi XP, mengambil dua tahapan awal, yaitu tahap eksplorasi dan tahap perencanaan.

*Unified Modeling Language* adalah bahasa standar yang digunakan untuk menjelaskan dan

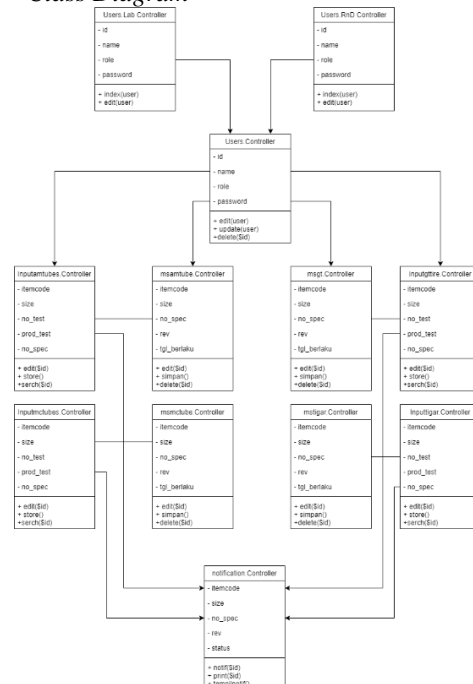
memvisualisaikan artefak dari proses analisis dan disain berorientasi objek. UML menyediakan standar pada notasi dan diagram yang bisa digunakan untuk memodelkan suatu system. Berikut merupakan UML dari sistem aplikasi pengolah data hasil pengujian *Tire Architecture* dan *Physical Properties*.

1. **Use Case Diagram**



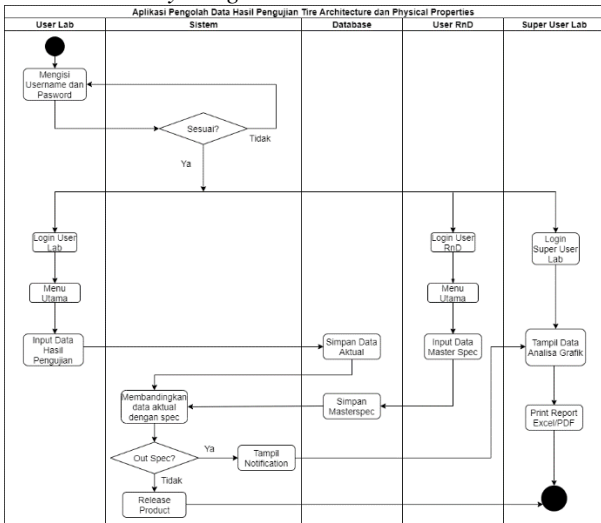
Gambar 2. Use Case Diagram

2. **Class Diagram**



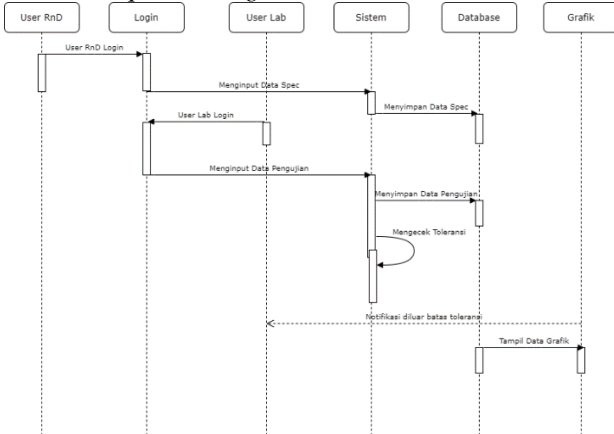
Gambar 3. Class Diagram

3. Activity Diagram



Gambar 4. Activity Diagram

4. Sequence Diagram



Gambar 5. Sequence Diagram

IV. HASIL DAN PEMBAHASAN Iterasi Pengembangan Sistem

Pada tahap iterasi pengembangan sistem, akan dilakukan dua kali iterasi, dimana pada setiap iterasi dilakukan lebih dari satu kali pertemuan untuk membahas pengembangan sistem. Setiap iterasi akan melalui tiga tahapan yaitu analisis sistem, dan pembuatan dan pengujian sistem.

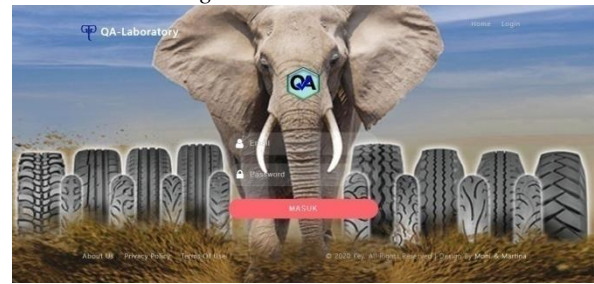
Gambaran Umum Aplikasi

6. Halaman Utama



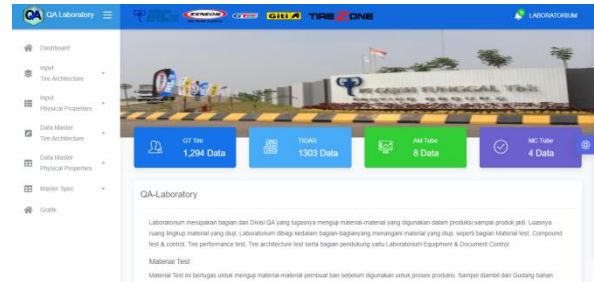
Gambar 6. Interface Halaman Utama

7. Halaman Login



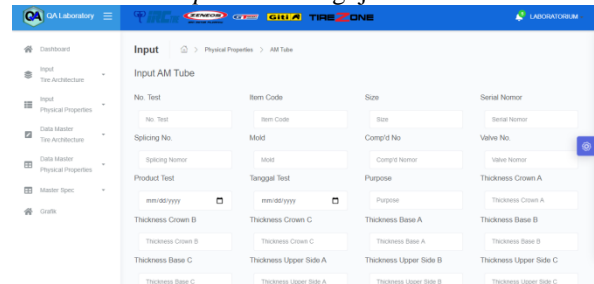
Gambar 7. Interface Login

8. Halaman Dashboard



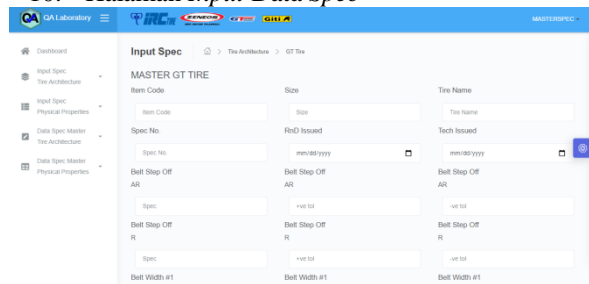
Gambar 8. Interface Dashboard

9. Halaman Input Data Pengujian



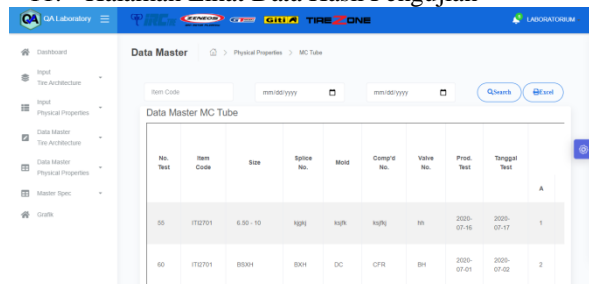
Gambar 9. Interface Input Data Pengujian

10. Halaman Input Data Spec



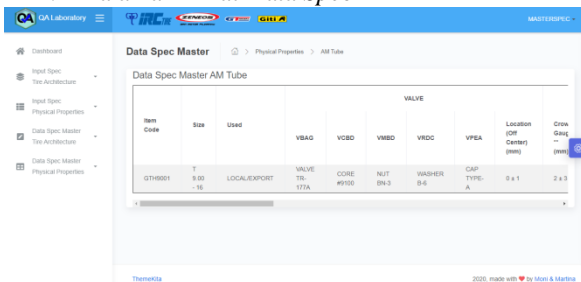
Gambar 10. Interface Input Data Spec

11. Halaman Lihat Data Hasil Pengujian



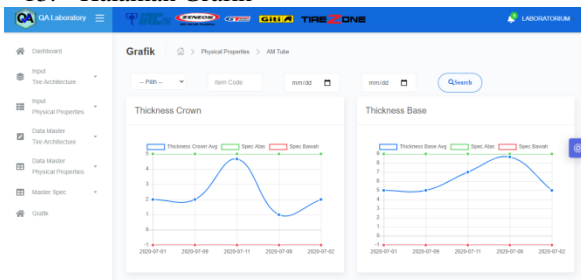
Gambar 11. Interface Lihat Hasil Pengujian

### 12. Halaman Lihat Data Spec



Gambar 12. Interface Lihat Spec

### 13. Halaman Grafik



Gambar 13. Interface Lihat Grafik

### 14. Halaman Notifikasi

Gambar 14. Interface Notifikasi

### Uji Verifikasi

Tabel 4. Uji Verifikasi

Pengujian	Hasil
Halaman Login	Berhasil
Halaman Home	Berhasil
Halaman Input Pengujian	Berhasil
Halaman Input Spec	Berhasil
Halaman Tampil Data Pengujian	Berhasil
Halaman Tampil Spec	Berhasil
Halaman Grafik	Berhasil
Halaman Notifikasi	Berhasil

### Keuntungan Penggunaan Aplikasi

Berdasarkan perbandingan dengan sistem sebelumnya, didapatkan hasil perubahan yang mampu memberikan keuntungan bagi departemen/perusahaan. Beberapa keuntungan yang didapat dari rancang bangun aplikasi tersebut, yaitu:

1. Sistem membutuhkan tidak membutuhkan *memory* besar karena berbasis web sehingga

tidak membuat *computer* laboratorium lamban.

2. Sistem dapat mengurangi *human error* dalam menganalisa hasil pengujian.
3. Sistem dapat mempersingkat waktu dalam proses pembuatan analisa dan *report* data hasil pengujian.
4. Sistem dapat mempersingkat waktu dalam pembuatan grafik.
5. Sistem dapat mempersingkat waktu pencarian *historical* data pengujian dan data *spec*.

## V. KESIMPULAN DAN SARAN

### Kesimpulan

Dari penelitian ini, penulis dapat menyimpulkan bahwa Aplikasi Pengolah Data Hasil Pengujian *Tire Architecture* dan *Physical Properties* dapat:

1. Meminimalisir waktu pencarian data,
2. Mempermudah pembuatan report, serta
3. Mengurangi *human error* pada sistem sebelumnya ketika mengintegrasikan database *spec*.

### Saran

Untuk penelitian selanjutnya, penulis menyarankan untuk:

1. Sistem Aplikasi Pengolah Data Pengujian *Tire Architecture* dan *Physical Properties* dapat terintegrasi dengan *Oracle* dan *Barcode System*.
2. Meningkatkan fungsi I/O *Spec* agar sistem menjadi lebih baik.

## DAFTAR PUSTAKA

Fertian, W. 2016. Pengertian Database Server. Binus University. (<https://student-activity.binus.ac.id/himsisfo/2016/07/pengertian-database-server/> diakses pada 19 Juli 2020).

Nugraha, F. (2010, January 20). Metode Pengembangan Aplikasi Secara Cepat dan Adaptif Menggunakan Agile Software Development. TEKNOJURNAL: (<https://teknajournal.com/metode-pengembangan-aplikasi-secara-cepat-dan-adaptif-menggunakan-agile-software-development/> diakses pada 5 Agustus 2020).

Saputra, Y. M. 2020. Pengertian Framework Laravel. Ruang Guru. (<https://ruangguru.co/pengertian-framework-laravel/> diakses pada 19 Juli 2020).

Supriyadi., Hudiono, R. K., Wijaya, L. S. (2013). Rancang Bangun Jejaring Klaster Berbasis Web Menggunakan Metode Model View Controller. Semarang: Seminar Nasional Unimus, 2019.