

BAB 1 IMPLEMENTASI DAN UJI COBA

1.1 Struktur Direktori dan Deskripsi File

Struktur direktori dan deskripsi file merupakan bagian penting dari dokumentasi yang diperlukan dalam pengembangan *Digital Monitoring System*. Dokumentasi ini diatur berdasarkan fungsinya dan disusun sesuai dengan kerangka kerja Laravel. Daftar direktori dan dokumen dapat dilihat pada Tabel 4.1. Berikut ini adalah daftar direktori *Digital Monitoring System* beserta keterangannya:

1. *App*: Direktori ini berisi file kode inti dari sistem.
2. *Public*: Direktori ini berisi file layout serta komponennya (CSS, Gambar, *file* JavaScript).
3. *Resources*: Direktori ini berisi file tampilan halaman dan *form* sistem.
4. *Routes*: Direktori ini berisi semua file rute yang ada pada sistem. Secara *default*, Laravel menyediakan beberapa file rute: *web.php*, *api.php*, *console.php*, dan *channels.php*.
5. *Vendor*: Direktori ini berisi dependensi dari pihak ketiga yang digunakan dalam sistem informasi.

Tabel 1.1 Tabel Struktur Direktori dan Deskripsi File

Nama Direktori	Nama File	Keterangan
demo-fastware/app/Http/Controller/	AuthController.php	Untuk menangani otentikasi login dan pengelolaan data pengguna.
	CustomerController.php	Untuk mengatur data pelanggan.
	DeptManController.php	Untuk mengelola data tindak lanjut
	DetailPreventiveController.php	Untuk melakukan perubahan terhadap detail dari <i>Preventive Maintenance</i> .
	FormFPPController.php	Untuk melakukan perubahan terhadap proses <i>Corrective Maintenance</i> .
	<i>HandlingController.php</i>	Untuk mengatur data <i>handling claim</i> dan

Nama Direktori	Nama File	Keterangan
		<i>complain</i> , serta <i>close</i> proses.
	MaintenanceController.php	Untuk menampilkan dashboard, berupa grafik terhadap proses <i>Corrective Maintenance</i> .
	MesinController.php	Untuk mengatur data mesin
	PreventiveController.php	Untuk mengatur perubahan terhadap <i>Preventive Maintenance</i>
	SparepartController.php	Untuk mengatur perubahan terhadap data sparepart berdasarkan mesin.
	UserController.php	Untuk mengatur data pengguna.
demo-fastware /resources/views/dashboard/	dsHandling.blade.php	Untuk menampilkan data <i>handling</i> dalam bentuk <i>chart</i> dan <i>card</i>
	dashboardMaintenance.blade.php	Untuk menampilkan data korektiv maintenance dalam bentuk <i>chart</i> dan <i>card</i>
demo-fastware /resources/views/admin/	create.blade.php	Membuat data pengguna
	edit.blade.php	Mengubah data pengguna
	index.blade.php	Menampilkan data pengguna
	show.blade.php	Melihat data pengguna
demo-fastware /resources/views/customer/	create.blade.php	Membuat data pelanggan
	edit.blade.php	Mengubah data pelanggan

Nama Direktori	Nama File	Keterangan
	index.blade.php	Menampilkan data pelanggan
	show.blade.php	Melihat data pelanggan
demo-fastware /resources/views/deptman/	submission.blade.php	Menampilkan data data tindak lanjut.
	confirm.blade.php	Melakukan konfirmasi data <i>handling claim</i> dan <i>complain</i>
	followup.blade.php	Melakukan tindak lanjut data <i>handling claim</i> dan <i>complain</i>
	scheduleVisit.blade.php	Melihat jadwal kunjungan
	historyProgres.blade.php	Menampilkan data history porgres <i>handling</i> .
	showCloseProgres.blade.php	Menampilkan data history porgres <i>handling</i> yang sudah <i>close</i> .
	historyClaimComplain.blade.php	Menampilkan data data <i>handling claim</i> dan <i>complain</i> dengan status close.
demo-fastware /resources/views/deptmtce/	createpreventive.blade.php	Membuat Form <i>Preventive Maintenance</i>
	index.blade.php	Melihat data Form Permintaan Perbaikan
	lihatfpp.blade.php	Melihat detail Form Permintaan Perbaikan
	lihatpreventive.blade.php	Melihat detail Form <i>Preventive Maintenance</i>

Nama Direktori	Nama File	Keterangan
	show.blade.php	Melakukan <i>Approved</i> terhadap Form Permintaan Perbaikan
	tabelpreventive.blade.php	Melihat data Form <i>Preventive Maintenance</i> dalam bentuk tabel.
demo-fastware /resources/views/fpps/	closed.blade.php	Melakukan closed terhadap Form Permintaan Perbaikan.
	create.blade.php	Membuat form permintaan perbaikan.
	history.blade.php	Menampilkan riwayat form permintaan perbaikan jika telah closed.
	index.blade.php	Menampilkan data form permintaan perbaikan.
	show.blade.php	Melihat data form permintaan perbaikan.
demo-fastware /resources/views/maintenance/	create.blade.php	membuat dan memperbarui form tindak lanjut
	edit.blade.php	Melakukan konfirmasi form permintaan perbaikan
	editpreventive.blade.php	Melakukan perubahan terhadap <i>Preventive Maintenance</i>
	index.blade.php	Menampilkan data form permintaan perbaikan

Nama Direktori	Nama File	Keterangan
	lihat.blade.php	Melihat form permintaan perbaikan, form tindak lanjut, dan riwayat progres.
	lihatpreventive.blade.php	Melihat form <i>Preventive Maintenance</i> .
	tabelpreventive.blade.php	Melihat data <i>Preventive Maintenance</i> .
demo-fastware /resources/views/mesins/	create.blade.php	Membuat data mesin
	edit.blade.php	Mengubah data mesin
	index.blade.php	Menampilkan data mesin
	show.blade.php	Melihat data mesin
demo-fastware /resources/views/mesins/	history.php	Menampilkan riwayat form permintaan perbaikan.
	index.blade.php	Menampilkan data form permintaan perbaikan.
	lihat.blade.php	Melihat form permintaan perbaikan.
demo-fastware /resources/views/sales/	<i>handling</i> .blade.php	Menampilkan data <i>handling</i> .
	create.blade.php	Membuat data <i>handling</i> .
	edit.blade.php	Mengubah data <i>handling</i> .
	showHistory.blade.php	Menampilkan data history porgres <i>handling</i> .
demo-fastware/app/models/	Customer.php	Untuk mengakses data pelanggan

Nama Direktori	Nama File	Keterangan
	DetailPreventive.php	Untuk mengakses data detail dari <i>Preventive Maintenance</i> .
	FormFPP.php	Untuk menampilkan data form permintaan perbaikan.
	<i>Handling</i> .php	Untuk mengakses data <i>handling</i>
	JadwalPreventif.php	Untuk mengakses data <i>Preventive Maintenance</i> .
	Mesin.php	Untuk mengakses data mesin.
	Role.php	Untuk mengakses data role
	ScheduleVisit.php	Untuk mengakses data tindak lanjut.
	Sparepart.php	Untuk mengakses data sparepart.
	TindakLanjut.php	Untuk mengakses data riwayat progres dari Tindak Lanjut.
	TypeMaterial.php	Untuk mengakses data <i>type material</i>
	User.php	Untuk mengakses data pengguna.

1.2 Implementasi

Implementasi sistem Digital Monitoring System (DMS) dalam aplikasi ini dilakukan dengan mengembangkan tiga modul utama: *corrective maintenance*, *preventive maintenance*, dan *handling claim and complain*. Setelah tahap *analysis* dan *design*, tim pengembang memulai penulisan kode program bersama dengan mentor industri dan pengguna dari departemen untuk memastikan kebutuhan sistem terpenuhi.

Dengan menggunakan *waterfall development* untuk ketiga modul ini, membantu tim pengembang untuk melakukan implementasi dengan pembuatan program dengan pengkodean

sesuai dengan *user interface* yang telah dibuat melalui tahap *design*, beserta dengan proses bisnis dan *user requirement* pada tahap *analysis*. Dalam proses implementasi ini dilakukan *review* mingguan untuk melakukan *update* progres yang didapat sehingga berjalan sesuai dengan target pembuatan. Pada pembuatannya, dibuat menggunakan Visual Studio Code, serta fitur dari Github untuk membantu dalam mengunggah *update* terhadap program yang dibuat oleh tim pengembang secara langsung dan proses pembuatan sistem DMS ini berlangsung selama dua bulan.

1.3 Pengujian

Untuk memastikan bahwa semua kebutuhan sistem telah dipenuhi, *website* DMS diuji secara bertahap oleh tim pengembang dan mentor industri sebelum diuji coba oleh pengguna di *server*. Prosedur pengujian ini dilakukan secara bertahap, dimulai oleh tim pengembang dilakukan secara menyeluruh. Pengujian yang dilakukan adalah.

1.3.1 Unit Testing Developer

Pada proses pembuatan sistem DMS, *unit testing* digunakan untuk menguji spesifikasi program apakah telah berjalan sesuai dengan permintaan *user* dengan menggunakan *black-box testing*. Tim pengembang juga menggunakan pengujian *black-box testing* untuk memastikan bahwa tidak terjadi *bug* atau *error* dalam jalannya sistem aplikasi, seperti validasi pengisian, dan proses penyimpanan data.

Melalui unit pengujian ini, sistem DMS diuji untuk memastikan dapat memenuhi kebutuhan pengguna dan menjalankan fungsi program dengan baik.

1.3.2 Integration Testing

Proses *integration testing* digunakan untuk memastikan bahwa program bekerja secara bersamaan tanpa *error*. Tim pengembang melakukan pengujian desain antarmuka (*user interface testing*) untuk memastikan bahwa sistem DMS beroperasi dengan benar. Sebelum digabungkan menjadi satu, sistem ini dibuat secara terpisah berdasarkan prosesnya masing-masing. Tujuannya adalah untuk menyesuaikan dengan kebutuhan pengguna dan mencegah kegagalan akses menu yang digunakan dalam sistem DMS.

1.4 Net Quality Income

Net Quality Income (NQI) merupakan sebuah konsep untuk mengukur pendapatan perusahaan dengan adanya sistem yang telah diimplementasikan. NQI berguna untuk

mengetahui gambaran pendapatan yang diperoleh setelah mengurangi biaya-biaya yang dikeluarkan. Adapun perhitungan NQI pada *website* DMS ini dapat dilihat pada Gambar di bawah ini.

NO	DESKRIPSI	NILAI
1.0.0	TANGIBLE BENEFIT	
1.1.0	Penurunan biaya operasi	Rp
1.1.1	Penghematan (konsumsi) material	Rp 49.300.000
1.1.2	Biaya Pengadaan Sistem	Rp 30.000.000
# TOTAL BENEFIT		Rp 79.300.000
2.0.0	COST OF IMPLEMENTATION* (incremental cost)	
	* Biaya hanya dihitung untuk biaya baru (investasi)	q
2.1.0	Project Development	Rp
2.1.1	Manhour tim	Rp 8.400.000
# TOTAL COST OF IMPLEMENTATION		Rp 8.400.000
NET QUALITY INCOME (BENEFIT – COST)		Rp 70.900.000
BENEFIT/COST (HIGHER BETTER)		9,44

Gambar 1.1 Perhitungan *Net Quality Income*

MANHOUR TIM				
1 Tim	:	2	Orang	
Uang saku 1 orang @ hari	:	Rp35.000,00		
Durasi pengembangan	:	120	hari	
Total		:		Rp8.400.000.00

Gambar 1.2 Perhitungan *Manhour Tim*

PENGHEMATAN BIAYA PENGADAAN SISTEM				
Biaya yang mungkin dikeluarkan apabila proyek dilimpahkan kepada vendor				
Tarif pengembang @ hari	:	Rp250.000,00		
Durasi pengembangan	:	120	hari	
Total		:		Rp30.000.000,00

Gambar 1.3 Perhitungan Penghematan Biaya Pengadaan Sistem

PENGHEMATAN (KONSUMSI) MATERIAL				
Penghematan material yang digunakan seperti kertas, tinta, biaya perawatan printer untuk Dokumen Pelamar dan biaya pulsa				
Kertas				
Jumlah kertas yang dibutuhkan				
Kebutuhan kertas Form Permintaan Perbaikan	:		1	lembar
Kebutuhan kertas Form Tindak Lanjut	:		1	lembar
Kebutuhan kertas Form Persetujuan	:		1	lembar
Kebutuhan kertas Riwayat Progres Perbaikan Mesin	:		1	lembar
Kebutuhan kertas Form Klaim	:		1	lembar
Kebutuhan kertas Form Komplain	:		1	lembar
Kebutuhan kertas Form Jadwal Kunjungan	:		1	lembar
Kebutuhan kertas Riwayat Progres Klaim dan Komplain	:		1	lembar
Kebutuhan kertas dalam @sebulan 250	:		8	lembar
Total kebutuhan kertas dalam @sebulan 250	:	250	2000	lembar
Harga cetak @ lembar	:		Rp500,00	Rupiah
Biaya kertas @ bulan	:		Rp1.000.000,00	Rupiah
Biaya kertas @ Tahun	:	12	Rp12.000.000,00	Rupiah
Kebutuhan Cap Stempel @tahun 20 buah	:	Rp15.000,00	Rp300.000,00	Rupiah
Kebutuhan Perawatan Mesin printer @tahun	:		Rp1.000.000,00	Rupiah
Total Keseluruhan			Rp13.300.000,00	Rupiah

Gambar 1.4 Perhitungan Penghematan (Konsumsi) Material

PENURUNAN REWORK/SCRAP										
Penurunan waktu proses (dalam rupiah) = Selisih waktu x biaya per satuan waktu x frekuensi x pengguna yang terlibat										
	Rincian Proses	Kategori Pengguna	Pengguna		Durasi Setelah & Sebelum Ada Sistem				Penurunan Waktu Proses	
			Jumlah	Satuan	Sebelum	Satuan	Sesudah	Satuan		
1	Pengisian Form Permintaan Perbaikan	Departemen Production	3	Orang	5	Menit	1	Menit	4	Rp 6.223.505
2	Pengisian Form Tindak Lanjut	Maintenance	2	Orang	5	Menit	1	Menit	4	Rp 5.317.535
3	Pengisian Form Persetujuan	Dept. Head Maintenance	1	Orang	5	Menit	1	Menit	4	Rp 8.333.333
4	Pengisian Riwayat Progres Perbaikan Mesin	Departemen Production	3	Orang	5	Menit	1	Menit	4	Rp 6.223.505
5	Pengisian Riwayat Progres Perbaikan Mesin	Maintenance	2	Orang	5	Menit	1	Menit	4	Rp 5.317.535
6	Pengisian Riwayat Progres Perbaikan Mesin	Dept. Head Maintenance	1	Orang	5	Menit	1	Menit	4	Rp 8.333.333
7	Pengisian Form Klaim	Departemen Sales	5	Orang	3	Menit	1	Menit	2	Rp 20.833
8	Pengisian Form Komplain	Departemen Sales	5	Orang	3	Menit	1	Menit	2	Rp 4.608
9	Pengisian Riwayat Progres Klaim dan Komplain	Departemen Sales	5	Orang	3	Menit	1	Menit	2	Rp 4.608
10	Pengisian Riwayat Progres Klaim dan Komplain	Dept. Head Maintenance	1	Orang	3	Menit	1	Menit	2	Rp 4.167
11	Pengisian Form Jadwal Kunjungan	Dept. Head Maintenance	1	Orang	3	Menit	1	Menit	2	Rp 4.167
1 Kertas									Rp	78.131
500 Kertas										500
									Rp	468.785.334

Penurunan waktu proses (dalam rupiah) = Selisih waktu x biaya per satuan waktu x frekuensi x pengguna yang terlibat										
Dept. Head Maintenance					Departemen Sales					
Biaya Per Satuan Waktu					Biaya Per Satuan Waktu					
Nominal					Satuan Nominal					
Rp					Bulan Rp					
25.000.000					5.529.285					
Rp					Hari Rp					
1.250.000					276.464					
Rp					Jam Rp					
125.000					27.646					
Rp					Menit Rp					
2.083					461					
Rp					Detik Rp					
35					8					
Departemen Production					Maintenance					
Biaya Per Satuan Waktu					Biaya Per Satuan Waktu					
Nominal					Satuan Nominal					
Rp					Bulan Rp					
6.223.505					7.976.303					
Rp					Hari Rp					
311.175					398.815					
Rp					Jam Rp					
31.118					39.882					
Rp					Menit Rp					
519					665					
Rp					Detik Rp					
9					11					

Gambar 1.5 Perhitungan Penurunan Rework/Scrap

Berdasarkan Gambar 4.1 hingga 4.5, perhitungan NQI dengan DMS mencapai Rp 70.900.000,00. NQI terdiri dari man hour sebesar Rp 8.400.000,00, penghematan biaya sebesar Rp 30.000.000,00, penghematan konsumsi material sebesar Rp 13.300.000,00, dan penurunan rework/scrap sebesar Rp 468.785.334.