

# Pengembangan *Dashboard* Alat Ukur Kinerja Tim (Studi Kasus: PT Javan Cipta Solusi)

Romzi Qutbi  
Program Studi Informatika  
Fakultas Teknologi Industri  
Universitas Islam Indonesia  
Yogyakarta, Indonesia  
18523229@students.uii.ac.id

Septia Rani  
Program Studi Informatika  
Fakultas Teknologi Industri  
Universitas Islam Indonesia  
Yogyakarta, Indonesia  
septia.rani@uui.ac.id

**Abstrak**—PT Javan Cipta Solusi adalah perusahaan yang bergerak di bidang teknologi informasi. Dengan banyaknya data dari proyek yang dikerjakan, maka perlu ada sebuah *dashboard* yang bisa memberikan *insight* untuk membuat keputusan yang baik ke depannya pada perusahaan. Oleh karena itu, PT Javan Cipta Solusi menerapkan sebuah metode bernama *Balanced Scorecard* (Alat Ukur) untuk mengukur dan menilai kinerja dalam perusahaan. Timload adalah salah satu *tool* yang dikembangkan oleh tim internal di PT Javan Cipta Solusi untuk mencapai tujuan tersebut. Timload adalah aplikasi berbasis web yang menggunakan *framework* Django dengan mengambil data dari *database* Taiga dan ActiveCollab untuk diproses dan kemudian disajikan dalam bentuk *dashboard* yang akan mempermudah para *stakeholder* di perusahaan untuk membuat keputusan. Tujuan dari makalah ini adalah memaparkan bagaimana *dashboard* ini dikembangkan, khususnya terkait pengembangan dua fitur yaitu halaman Detail Performa dan *Project Scoreboard*.

**Kata Kunci**—*Balanced Scorecard*, *Timload*.

## I. PENDAHULUAN

Dewasa ini perkembangan teknologi informasi dan komunikasi berkembang sangat pesat. Dengan hal ini banyak sekali kehidupan manusia yang dipermudah. Banyak perusahaan teknologi informasi yang sudah berdiri dan ingin tetap bertahan pada bidang ini. Untuk bertahan perusahaan-perusahaan harus menerapkan strategi yang baik untuk mencapai tujuannya. Ada begitu banyak aspek yang harus dipertimbangkan dalam merancang strategi tersebut. Mulai dari aspek internal (seperti visi misi perusahaan dan sumber daya manusia perusahaan) hingga aspek eksternal (seperti perilaku konsumen dan tren pasar).

Perusahaan memerlukan sebuah sistem untuk mengelola setiap strateginya sebaik mungkin yang mana aspek-aspek tersebut bisa saling terintegrasi. *Balanced Scorecard* adalah salah satu jawabannya. *Balanced Scorecard* adalah sistem manajemen yang bertujuan untuk menerjemahkan tujuan strategis perusahaan ke dalam serangkaian tujuan kinerja yang pada gilirannya, diukur, dipantau, dan diubah jika perlu untuk memastikan bahwa tujuan strategis perusahaan terpenuhi.

Pada penelitian sebelumnya sudah membahas tentang *Balanced Scoreboard* untuk mengevaluasi kinerja transportasi perkotaan dengan empat indikator yaitu perspektif sosial, perspektif ekonomi, perspektif proses, dan perspektif bisnis dan inovasi [1].

*Dashboard* Timload adalah bentuk dari penerapan *Balanced Scorecard* di PT Javan Cipta Solusi. *Dashboard* ini dibuat karena ditemukan beberapa masalah di *project* seperti ada tim yang kadang tidak mendapat *task*, ada tim yang mendapatkan terlalu banyak *task*, dan kedua tim tim tersebut diberi *reward* yang sama tanpa memperdulikan performa

masing-masing, dan juga ini menyebabkan *deliver* produk ke *client* menjadi terlambat, dan jika ini terjadi maka pengembangan produk menjadi lama dan terjadi pembengkakan biaya yang tidak baik untuk perusahaan. Dengan adanya *dashboard* ini maka setiap tim akan termotivasi dengan baik sehingga nantinya tim dengan performa yang bagus akan mendapatkan *reward* berupa bonus. Dan produk bisa di-*deliver* dengan tepat waktu.

PT Javan Cipta Solusi adalah perusahaan teknologi informasi (IT) yang berdiri sejak 2008 di Bandung [2]. Perusahaan ini telah mengembangkan produk dari berbagai klien dengan berbagai instansi seperti swasta, pemerintah kabupaten maupun kota, kementerian dan lembaga pusat, hingga Perguruan Tinggi Negeri. PT Javan Cipta Solusi mengembangkan sistem *Balancing Scorecard*-nya sendiri yang bernama Timload. Timload adalah *dashboard* alat ukur untuk memantau performa dari setiap pegawai hingga performa dari proyek yang dikerjakan oleh PT Javan Cipta Solusi. Timload dibangun menggunakan *framework* Django. Django adalah *framework full-stack* untuk membangun web dengan menggunakan bahasa pemrograman Python. Dengan kata lain Django meliputi sisi *front-end* dan juga *back-end*. *Front-end* adalah sisi depan yang akan dilihat oleh pengguna, sedangkan *back-end* adalah sisi belakang yang berhubungan dengan *database* dan logika bisnis [3].

Pengembangan Timload berada di bawah *Internal Growth Team*, khususnya pada *Productivity Support Team*. *Productivity Support Team* bertanggung jawab untuk membantu produktivitas di PT Javan Cipta Solusi dengan cara mengembangkan *tools* dan juga menjawab permasalahan teknis tim *project*.

## II. DASAR TEORI

### A. *Balanced Scorecard*

*Balanced Scorecard* adalah suatu metode pengukuran dan penilaian kinerja suatu perusahaan atau organisasi dengan mengukur empat perspektif yaitu: perspektif keuangan, perspektif pelanggan, perspektif proses bisnis internal, serta perspektif pembelajaran dan pertumbuhan. *Balanced Scorecard* dapat dianalogikan sebagai tombol dan indikator di kokpit pesawat. Untuk tugas kompleks menavigasi dan menerbangkan pesawat, pilot memerlukan informasi rinci tentang banyak aspek penerbangan. Mereka membutuhkan informasi tentang bahan bakar, kecepatan udara, ketinggian, arah, tujuan, dan indikator lain yang merangkul lingkungan saat ini dan yang diperkirakan. Ketergantungan pada satu instrumen bisa berakibat fatal. Demikian pula, kompleksitas pengelolaan organisasi saat ini menuntut manajer untuk dapat melihat kinerja di beberapa bidang secara bersamaan [4].

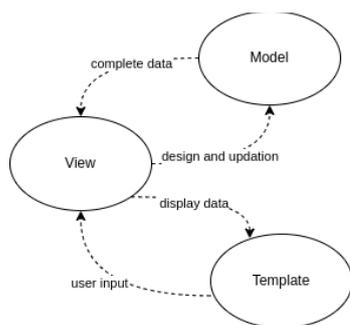
Terdapat lima fungsi dari *Balanced Scorecard*, di antaranya yaitu memberi struktur pada strategi, mempermudah komunikasi strategi dalam bisnis, menyelaraskan departemen dan divisi, membantu karyawan dalam melihat bagaimana tujuan individual mereka terkait dengan strategi organisasi, serta membuat strategi terus berada pada *track*-nya. *Balanced Scorecard* memiliki empat perspektif yang terdiri atas perspektif keuangan, perspektif proses bisnis internal, perspektif pelanggan, serta perspektif pembelajaran dan pertumbuhan. Masing-masing perspektif dapat dijelaskan sebagai berikut:

- 1) *Perspektif keuangan*. Perusahaan harus mampu mengelola keuangan dengan baik agar keuangannya terus stabil. Misalnya, biaya operasional, biaya produksi, biaya bahan baku, biaya tenaga kerja, termasuk keuntungan dari aktivitas penjualan.
- 2) *Perspektif proses bisnis internal*. Kemampuan dan keahlian yang dimiliki karyawan di dalam perusahaan akan menghasilkan proses bisnis internal yang bagus. Proses bisnis internal yang bagus akan mendatangkan pelanggan dan meraih keuntungan secara keuangan.
- 3) *Perspektif pelanggan*. Membahas tentang tujuan kinerja terkait calon pelanggan dan pasar. Pelayanan yang bagus akan membuat konsumen loyal dan bisa saja mendatangkan konsumen baru.
- 4) *Perspektif pembelajaran dan pertumbuhan*. Karyawan adalah elemen yang penting dalam perusahaan. Maka dari itu perusahaan dituntut untuk membangun sistem agar ada pengembangan pada Sumber Daya Manusia.

#### B. Model-View-Template

*Model-View-Template* adalah salah satu *design pattern* dalam pengembangan aplikasi web, khususnya digunakan pada *web framework* Django. Pada Gambar 1 Model-View-Template ditunjukkan bagaimana keterkaitan di antara ketiganya.

- 1) *Model*, berperan sebagai *interface* kepada *database*. Kelas sebagai tabel dalam *database* dan atribut kelas sebagai kolom dalam *database*.
- 2) *View*, berperan untuk mengeksekusi logika bisnis dan mengambil, membaca, atau menulis data dari *database* melalui model kemudian menampilkannya ke dalam *template*.
- 3) *Template*, berperan sebagai antarmuka yang di-*render* dan dilihat langsung oleh *user* yang mengunjungi website [5].



Gambar 1 Model-View-Template

#### C. Support Vector Machine

*Support Vector Machine* (SVM) adalah salah satu algoritma *supervised machine learning* yang mana data yang digunakan untuk *training model* adalah data yang sudah memiliki label sebelumnya. Sehingga dalam penentuan proses keputusan, mesin akan mengkategorikan data ke dalam label yang sesuai dengan karakteristik yang dimilikinya. Dalam melakukan klasifikasi, konsep SVM lebih matang dan lebih jelas secara matematis dibandingkan metode klasifikasi lainnya [6].

Dalam melakukan klasifikasi, SVM akan mencari *hyperline* terbaik dalam memaksimalkan jarak antar kelas berdasarkan N-dimensi tergantung banyak *features* sehingga nantinya akan mudah menentukan kelas atau kategori data yang baru. Dalam hal ini penulis menggunakan SVM untuk mengklasifikasi data teks ke dalam kelas-kelas tertentu [7].

### III. METODOLOGI

Metodologi yang digunakan pada pengembangan *dashboard* alat ukur Timload ini yaitu metode *agile*. Metode *agile* dipilih karena bisa membantu developer mengembangkan perangkat lunak lebih efisien dan sesuai kebutuhan konsumen, dilakukan secara bertahap dan berulang-ulang [8]. Alur tahap pengembangan perangkat lunak dengan metode *agile* dapat dilihat pada Gambar 2. Terdapat 5 tahap, mulai dari tahap *analysis*, *design*, *implementation*, *evaluation*, dan *deployment*. Jika tahap pengembangan sudah mencapai tahap *evaluation* maka nantinya akan terdapat 2 pilihan yaitu jika *implementation* sudah sesuai maka akan berlanjut ke tahap *deployment*, dan jika tidak maka akan kembali ke tahap *design* lagi.

#### A. Tahap Analysis

Yang pertama yaitu tahap analisis. Pada tahap ini tim akan mengidentifikasi kebutuhan apa saja yang diperlukan dalam mencapai tujuan untuk menyelesaikan masalah yang ingin dipecahkan. Tahap analisis bisa dilakukan dengan cara melakukan *meeting* dengan klien dan mencatat apa masalahnya dan apa tujuannya.

#### B. Tahap Design

Yang kedua yaitu tahap desain. Pada tahap ini tim akan melakukan perancangan aplikasi mulai dari desain arsitektur, desain *User Interface*, hingga menentukan teknologi apa saja yang akan digunakan. *Output* dari tahap ini adalah berupa *Software Design Document* (SDD). SDD adalah rencana yang sudah didetailkan untuk mengembangkan perangkat lunak [9].

#### C. Tahap Implementation

Selanjutnya yaitu tahap implementasi. Di tahap ini tim *engineer* mulai mengimplementasikan semuanya sesuai dengan dokumen SDD. Pengerjaan meliputi *back-end* dan *front-end* dari perangkat lunak.

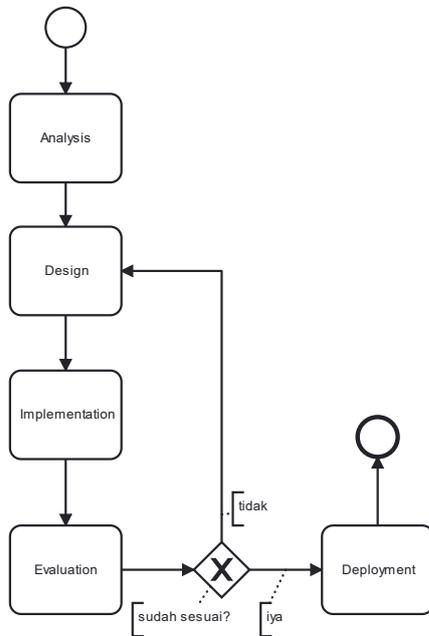
#### D. Tahap Evaluation

Tahap selanjutnya adalah tahap evaluasi, yang mana pada tahap ini fitur yang sudah dibuat oleh *engineer* akan diuji dan diberikan penilaian. Pengujian ini dilakukan pada *server staging*. *Staging server* adalah replika lingkungan produksi atau *production environment* untuk menguji kode baru dan pembaruan sebelum mempublikasikannya [10]. Jika lolos tahap ini maka akan berlanjut ke tahap *deployment*. Namun

apabila tidak lolos maka akan kembali ke tahap implementasi sesuai dengan *feedback* yang diberikan.

#### E. Tahap Deployment

Tahap yang terakhir adalah tahap *deployment*. Tahap *deployment* adalah tahap untuk melakukan publikasi aplikasi atau fitur yang sudah dikerjakan dan sudah melalui tahap *testing* sehingga bisa diakses dan digunakan oleh *user*. Tahap ini biasanya dikerjakan oleh *DevOps Engineer*.



Gambar 2 Tahap Pengembangan *Software* dengan metode *agile*

### IV. HASIL DAN PEMBAHASAN

Hasil akhir dari sistem yang dikembangkan adalah *dashboard* yang disajikan dalam bentuk *website*. *Timload* sendiri menggunakan *web framework* Django dengan *design pattern* perangkat lunak *Model-View-Template*. Pada *dashboard* *Timload* terdapat beberapa fitur atau halaman, di antaranya yaitu:

- 1) Halaman Performance yang menyajikan performa setiap tim di Javan berdasarkan banyak *task* yang diselesaikan per bulannya.
- 2) Halaman Detail Performa, menyajikan detail *tasks* yang dikerjakan setiap orang, dan dipisahkan berdasarkan yang sudah selesai dan yang belum.

Sekaligus pada halaman ini terdapat *performance analysis*.

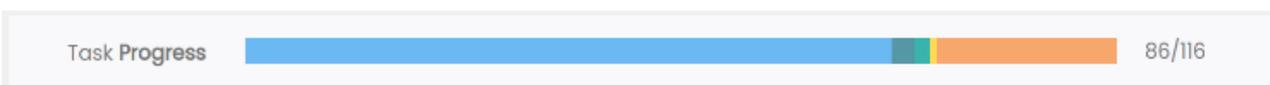
- 3) Halaman Balancing Table. Pada halaman ini menunjukkan pembagian *task* yang di-*grouping* berdasarkan *role* dan *deadline* dari *task* yang di-*assign* kepada setiap tim di Javan.
- 4) Halaman Sprint Progress. Halaman ini menyajikan *progress sprint* dari setiap *project* yang dikerjakan oleh Javan.
- 5) Halaman Project Scoreboard. Tujuan dari halaman ini dibuat adalah untuk memonitor setiap *project* yang dikerjakan oleh Javan dan menentukan sebuah *project* apakah masuk dalam kategori *excellent*, *good*, atau *poor* berdasarkan parameter yang sudah ditentukan.
- 6) Halaman Stale Task. Halaman ini menyajikan *task-task* yang tidak direspon dalam 1 minggu. Semakin banyak *task* yang tidak direspon maka semakin tidak baik.
- 7) Halaman Tim Progress. Halaman ini menyajikan *progress* setiap tim di Javan berdasarkan *task* yang sudah dikerjakan setiap harinya dan akumulasinya dalam sebulan, yang kemudian divisualisasikan dalam bentuk *line chart*.

Dari segi arsitektur sistem, *Timload* sendiri tidak memiliki *database*. *Timload* hanya memproses data dari basis data yang lain, dalam hal ini adalah Taiga dan ActiveCollab Javan untuk dianalisis.

#### A. Pengembangan Halaman Detail Performa

Halaman Detail Performa menyajikan performa masing-masing Tim Javan berdasarkan *task* yang dikerjakan di Taiga. Pada halaman ini terdapat beberapa fitur utama yaitu akumulasi status *task* yang dikerjakan pada bulan ini, Active Task, Performance Analysis, dan Finished Task.

Fitur yang pertama yaitu Task Progress. Warna biru menandakan *task* yang sudah *closed* atau *finish* sedangkan warna yang lain menandakan masih aktif seperti yang terlihat pada Gambar 3 Task Progress. Beberapa status *task* yang belum *closed* adalah *new* yaitu *task* yang baru dibuat akan tetapi belum siap dikerjakan oleh *programmer* dan harus dianalisis oleh analis, *ready* yaitu *task* yang sudah siap dikerjakan oleh *programmer*, *code review* yaitu *task* yang sudah siap untuk di-*review* oleh *code reviewer*, *ready for test* yaitu *task* yang sudah siap untuk dilakukan *testing* oleh tester, *feedback* yaitu jika *task* memiliki kekurangan dan ada masukan, *needs info* yaitu *task* yang masih memerlukan informasi lebih atau *task* yang kurang jelas maksudnya.



Gambar 3 Task Progress

Fitur yang kedua yaitu fitur Active Task yaitu daftar *tasks* yang belum *closed*, seperti terlihat pada Gambar 4. Pada *active task* terdapat detail *task* yang dikerjakan oleh Tim Javan yaitu *subject task*, *status task*, asal proyek *task*, *role* tim di *task*, dan *due date*. Pada bagian paling kiri setiap baris *task* terdapat indikator warna yang masing-masing warna menandakan hal

yang berbeda. Warna merah menandakan *task* sudah mencapai hari *due date* atau sudah melampaui hari *due date*, warna orange menandakan satu hari lagi *task* sudah *due date*, warna kuning menandakan dua hari atau lebih *task* akan mencapai *due date*.

Active task : 30 ● Due date hari H atau sudah lewat ● Due date tinggal 1 hari ● Due date tinggal lebih dari 1 hari

Task	Status	Project	Role	Due Date
[FE] Terdapat Halaman List LOA value config pada menu financial review	Testing	XL-LMS (FR)-2022	Analyst	June 15, 2022
[FE] Terdapat Halaman Update LOA value config pada menu financial review	Ready for test	XL-LMS (FR)-2022	Analyst	June 15, 2022
[INTEGRASI] pada Halaman Update LOA Value Config dapat melakukan update data yang telah di...	Ready for test	XL-LMS (FR)-2022	Analyst	June 16, 2022
[INTEGRASI] pada Halaman List LOA Value Config menampilkan list data, Bisa search , mengubah ...	Testing	XL-LMS (FR)-2022	Analyst	June 16, 2022
[INTEGRASI] pada Halaman Add New LOA Value Config dapat melakukan submit data yang telah ...	Testing	XL-LMS (FR)-2022	Analyst	June 16, 2022
Terdapat Halaman List Create FR pada menu financial review	In progress	XL-LMS (FR)-2022	Analyst	July 8, 2022
Terdapat API view detail pada menu modify FR Macro	Ready	XL-LMS (FR)-2022	Analyst	None
Terdapat API upload supporting document pada modify FR Macro	Ready	XL-LMS (FR)-2022	Analyst	None

Gambar 4 Active Task

Fitur yang ketiga yaitu Performance Analysis. Pada fitur ini terdapat 4 *chart*. *Chart* yang pertama yaitu *performance* berupa *column chart* yang merangkum performa setiap minggunya dari *engineer* sebanyak 8 minggu ke belakang. *Chart* yang kedua yaitu *donut chart* yang merangkum performa *engineer* selama 6 bulan ke belakang. *Chart* yang ketiga yaitu *donut chart* yang merangkum kategori *task* yang dikerjakan selama 6 bulan ke belakang. Yang terakhir yaitu *donut chart* yang merangkum *role* dari *engineer* dalam mengerjakan *task* dalam 6 bulan ke belakang.

Pengategorian *task* menggunakan metode *Support Vector Machine* dengan akurasi model sebesar 83% dari data *subject task* yang sudah diberi label seperti sampel data pada Tabel 1. Data *task* yang diambil berupa *subject task* terbaru yang *generate* dari *database* taiga. Sebanyak seribu data yang sudah dilabeli kemudian dibagi ke dalam 20% untuk *testing* dan 80% untuk *training*. Sebelum melakukan *training* model, data harus melalui tahap *pre-processing* agar data teks yang dipakai lebih terstruktur dan sesuai dengan model yang akan dipakai. Pada Tabel 2 menunjukkan bagaimana bentuk data sebelum *pre-processing* dan sesudah *pre-processing*. Pada Tabel 3 menunjukkan hasil *classification report* model yang dibuat. Pada *donut chart category* menunjukkan visualisasi dari klasifikasi setiap *task* yang dikerjakan oleh tim perindividu.

Fitur yang keempat yaitu Finished Task. Fitur ini bertujuan untuk merangkum semua *task* yang sudah ditutup pada bulan ini. Kurang lebih seperti fitur Active Task akan tetapi dengan indikator warna yang berbeda yaitu warna biru yang menandakan *task* sudah selesai.

Tabel 1 Labelling Subject Task

Subject task	Kategori
Membuat Tampilan Home Geek Academy	frontend
Data tidak update ke bulk novation dev	bugfixing
[WBS] Perbaiki tab komunikasi dan diskusi disable di role spv telaah	adjustment
Analisa Indexing Tabel untuk Tuning Performance	nonteknis
[KOPI] Perbaiki redirect after delete Data Aduan Aroma	bugfixing
web - dashboard - refactor query	querying
Restore db `sppti_kpk` di server sepatu	devops
Restore db `sppti_djp2` dari drive	devops
Create database `sppti_kpk` di server sepatu	devops

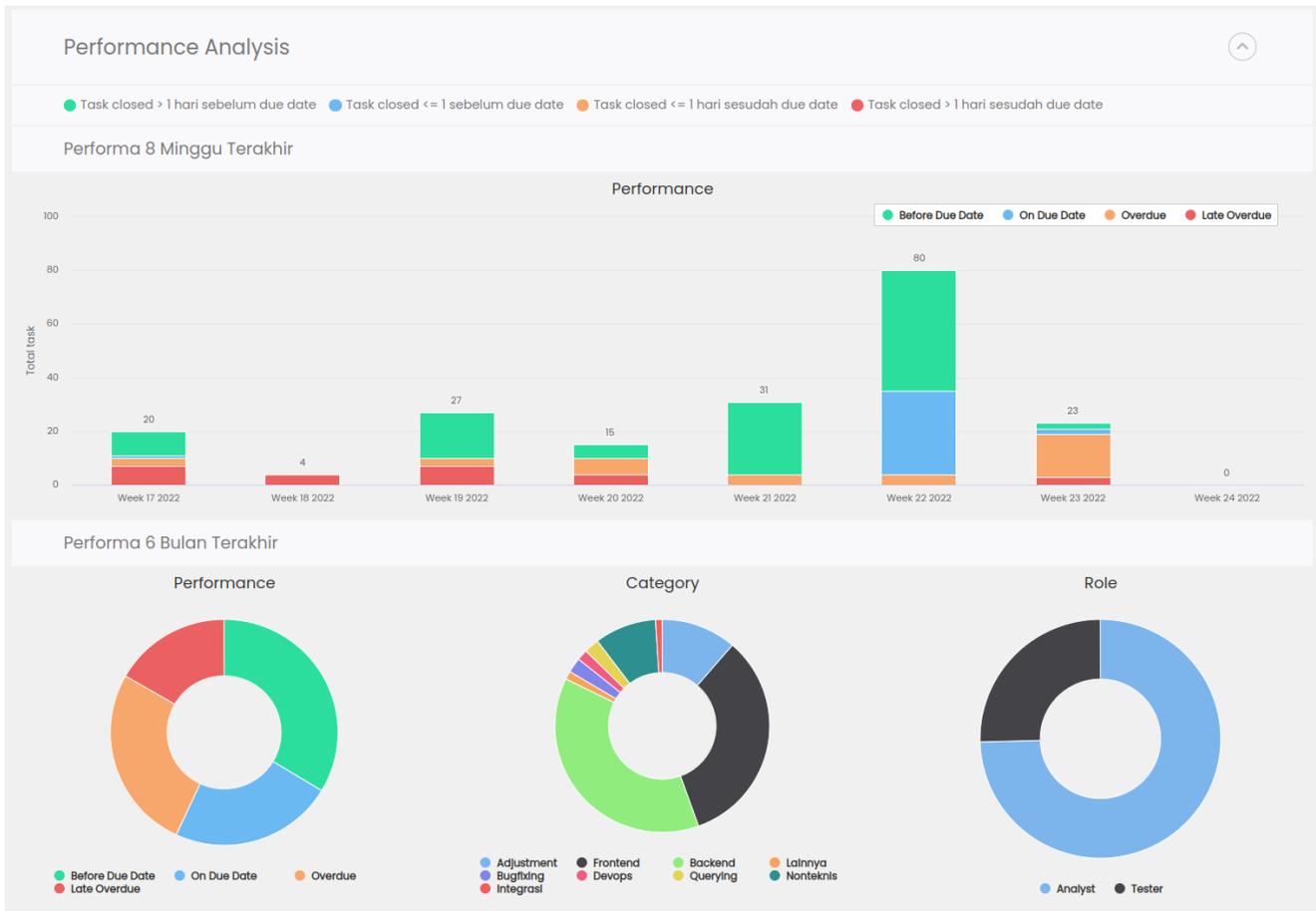
Menambahkan How to Setup project FE di Readme	frontend
Persiapan Dokumen Nego	nonteknis

Tabel 2 Pre-processing Data Teks

Sebelum	Sesudah
Membuat Tampilan Home Geek Academy	tampilan home geek academy
Data tidak update ke bulk novation dev	data update bulk novation dev
[WBS] Perbaiki tab komunikasi dan diskusi disable di role spv telaah	wbs perbaiki tab komunikasi diskusi disable role spv telaah
Analisa Indexing Tabel untuk Tuning Performance	analisa indexing tabel tuning performance
[KOPI] Perbaiki redirect after delete Data Aduan Aroma	kopi perbaiki redirect after delete data aduan aroma
web - dashboard - refactor query	web dashboard refactor query
Restore db `sppti_kpk` di server sepatu	restore db sppti_kpk server
Restore db `sppti_djp2` dari drive	restore db sppti_djp2 drive
Create database `sppti_kpk` di server sepatu	create database sppti_kpk server sepatu
Menambahkan How to Setup project FE di Readme	how to setup project fe readme
Persiapan Dokumen Nego	persiapan dokumen nego

Tabel 3 Classification Report

	precision	recall	f1-score	support
adjustment	0.65	0.76	0.70	17
backend	0.93	0.93	0.93	44
bugfixing	0.76	0.80	0.78	20
devops	1.00	0.50	0.67	6
frontend	0.81	0.93	0.87	45
integrasi	0.67	0.29	0.40	7
lainnya	0.90	0.69	0.76	13
nonteknis	0.88	0.85	0.86	52
querying	0.75	0.86	0.80	7
accuracy			0.83	211
macro avg	0.82	0.73	0.75	211
weighted avg	0.84	0.83	0.83	211



Gambar 5 Performance Analysis

Finished task : 86

Task	Status	Project	Role	Due Date
Terdapat Api view detail pada menu master P2M Purpose	Closed	XL-LMS (FR)-2022	Analyst	June 7, 2022
Terdapat API Delete pada menu master P2M Purpose	Closed	XL-LMS (FR)-2022	Analyst	June 7, 2022
simpel Production terdapat LOG debugging dan sentry	Closed	Ombudsman - SIMPEL - 2022	Tester	June 7, 2022
Terdapat API Submit pada menu master P2M Purpose	Closed	XL-LMS (FR)-2022	Analyst	June 7, 2022
Terdapat API Update pada menu master P2M Purpose	Closed	XL-LMS (FR)-2022	Analyst	June 7, 2022
terdapat table master p2m purpose pada cloud	Closed	XL-LMS (FR)-2022	Analyst	June 7, 2022
simpel Production terdapat LOG debugging dan sentry	Closed	Ombudsman - SIMPEL - 2022	Analyst	June 7, 2022
Terdapat API List pada menu master P2M Purpose	Closed	XL-LMS (FR)-2022	Analyst	June 7, 2022
Terdapat API P2M Purpose Status pada menu master P2M Purpose	Closed	XL-LMS (FR)-2022	Analyst	June 7, 2022

Gambar 6 Finished Task

### B. Pengembangan Project Scoreboard

Project Scoreboard dikembangkan untuk memonitor performa setiap *project* di Javan berdasarkan empat parameter yaitu:

- 1) Persentase *task* yang *closed* tepat waktu (*closed* sebelum atau tepat *deadline*).
- 2) Persentase *task ready for test* tepat waktu (*ready for test* sebelum atau tepat jam 8 pagi *deadline task*).

- 3) Persentase *task needs info* karena *task* yang dibuat oleh analis masih kurang jelas atau tidak bisa dimengerti oleh *programmer*.

- 4) Rata-rata umur *task*.

Kemudian dilakukan *scoring* terhadap masing-masing parameter berdasarkan nilai tertentu seperti pada Tabel 4, Tabel 5, Tabel 6, dan Tabel 7. Dari *score* parameter-parameter yang sudah dihitung akan didapatkan *score* final dengan cara mengakumulasi setiap *score* parameter. Kemudian dikategorikan ke dalam 4 kategori seperti pada Tabel 8.

Tabel 4 Scoring Task Tepat Waktu

Persentase Task Tepat Waktu	Score (1-10)
>= 95 %	10
<95 % and >= 85%	9
<85 % and >= 75%	8
<75 % and >= 65%	7
<65 % and >= 55%	6
<55 % and >= 45%	5
<45 % and >= 35%	4
<35 % and >= 25%	3
<25 % and >= 15%	2
<15 % and >= 5%	1
<5%	0

Tabel 5 Scoring Ready For Test Tepat Waktu

Persentase Ready For Test Tepat Waktu	Score (0-5)
>= 95%	5
<95 % and >= 85%	4
<85 % and >= 75%	3
<75 % and >= 65%	2
<65 % and >= 55%	1
<55%	0

Tabel 6 Scoring SLA Needs Info

Persentase SLA Needs Info	Score (0-5)
<5%	5
>=5% and < 10%	4
>=10% and < 15%	3
>= 15% and <20%	2
>= 20% and <25%	1
>= 25	0

Tabel 7 Scoring Rata-Rata Umur Task

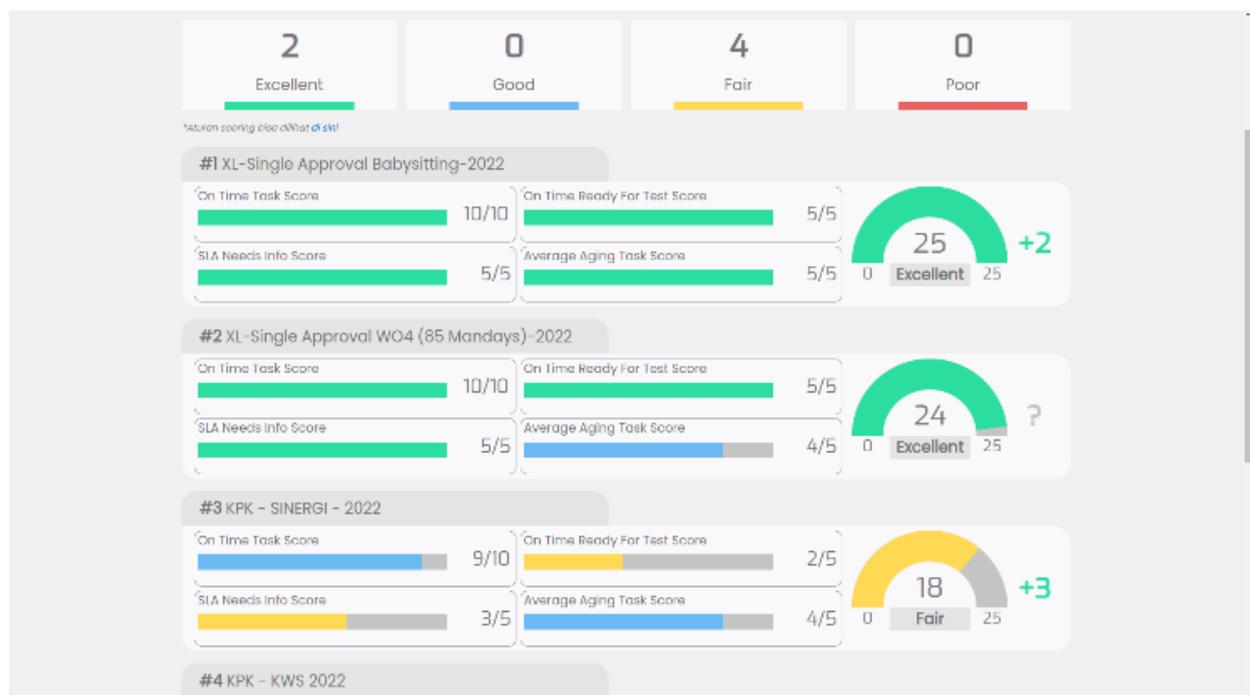
Rata-Rata Umur Task (Hari)	Score (0-5)
<= 3	5
>3 and <=4	4
>4 and <= 5	3
>5 and <= 6	2
>6 and <= 7	1
>7	0

Tabel 8 Final Score

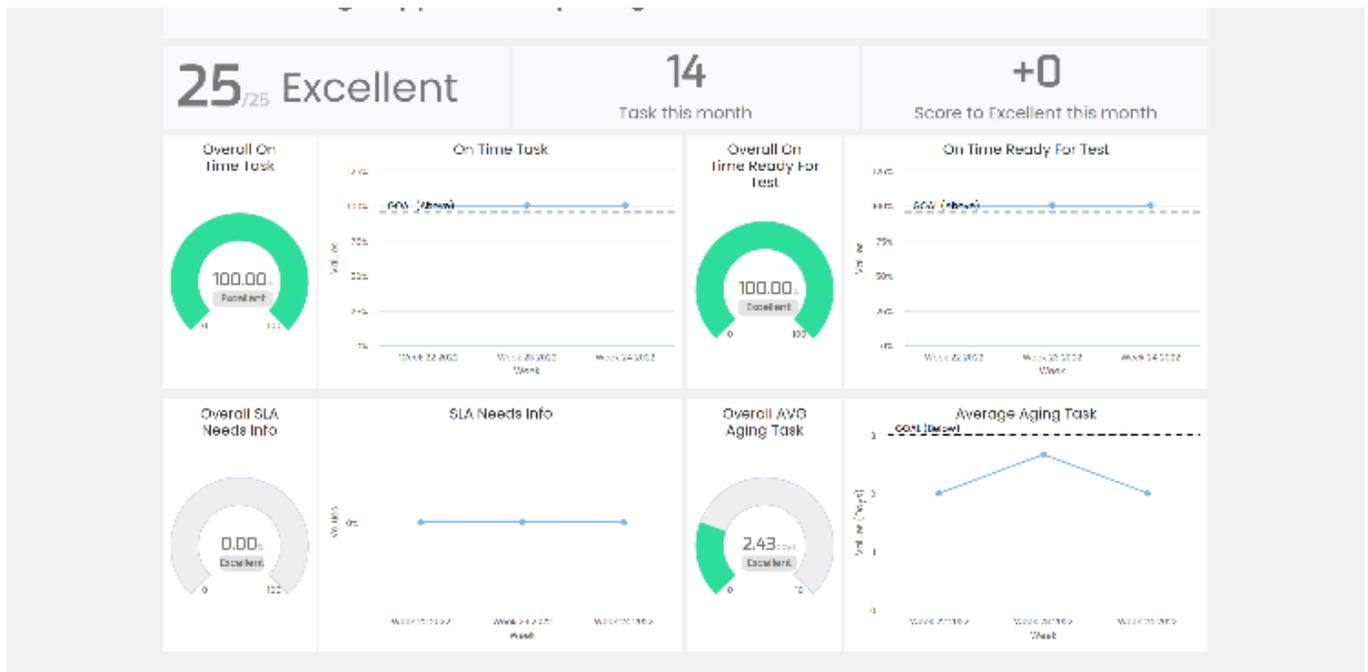
Mode	Score
Excellent	>=23
Good	<23 and >=21
Fair	<21 and >=15
Poor	<15

Project Scoreboard mempunyai dua halaman, yang pertama adalah halaman *project scoreboard*-nya sendiri yang menyajikan *score* setiap parameter *score* dan *final score* masing-masing *project* (seperti terlihat pada Gambar 7). *Score* parameter disajikan dalam bentuk *progress bar* dan *final score* disajikan dalam bentuk *gauge chart*. Kemudian diurutkan berdasarkan *score* yang paling besar hingga yang paling kecil. Selain itu, pada halaman ini ditampilkan *trend score* dari setiap *project*. Tanda “+” menunjukkan tren yang positif diikuti dengan naik berapa tingkat dibandingkan dengan bulan sebelumnya. Begitupun sebaliknya dengan tanda “-” jika mengalami penurunan tingkat. Serta tanda “?” jika tidak ada data bulan sebelumnya.

Yang kedua adalah halaman detail performa yang menyajikan tren setiap parameter setiap minggunya dalam satu bulan dan divisualisasikan dalam bentuk *line chart*. Pada Gambar 8 menunjukkan halaman detail performa *project*.



Gambar 7 Halaman Project Scoreboard



Gambar 8 Halaman Detail *Performa Project Scoreboard*

## V. KESIMPULAN

Berdasarkan pengembangan *dashboard* alat ukur kinerja tim yang bernama “Timload”, khususnya pada halaman detail performa dan *project scoreboard*, dapat disimpulkan bahwa *web framework* Django terbukti dapat mempermudah pengembangan sistem informasi berbasis web, dan karena Django adalah *framework open source* maka dapat dikustomisasi sesuai dengan keinginan. Selain itu, penerapan *Balanced Scorecard* pada Timload terbukti berhasil dengan digunakannya Timload untuk mengukur kinerja Tim dan performa *project* di PT Javan Cipta Solusi. Pemanfaatan metode *Support Vector Machine* dalam melakukan pengklasifikasian data teks juga terbukti berhasil dengan didapatkannya akurasi model sebesar 83%.

## DAFTAR PUSTAKA

- [1] I. Moufad and F. Jawab, "Designing a dashboard for evaluating the Performance of urban transport: Case of Balanced Scorecard," *International Journal of Scientific & Engineering Research*, vol. 8, pp. 182-185, 2017.
- [2] PT Javan Cipta Solusi, *Javan's Hanbook*, 2021.
- [3] A. Muhardian, "Belajar Django #1: Pengenalan Dasar Django untuk Pemula," 24 Agustus 2016. [Online]. Available: <https://www.petanikode.com/django-untuk-pemula/>. [Diakses 15 Juni 2022].
- [4] R. S. Kaplan dan D. P. Norton, "The Balanced Scorecard—Measures that Drive Performance," *Harvard Business Review*, Februari 1992. [Online]. Available: <https://hbr.org/1992/01/the-balanced-scorecard-measures-that-drive-performance-2>. [Diakses 15 juni 2022].
- [5] S. "Django Tutorial - MVT Pattern on Django (Model View Template)," TutorTech, 7 Juni 2021. [Online]. Available: <https://rrtutors.com/tutorials/django-mvt-design-pattern>. [Diakses 15 Juni 2022].
- [6] samsudiney, "Penjelasan Sederhana tentang Apa Itu SVM?," *medium.com*, 25 Juli 2019. [Online]. Available: <https://medium.com/@samsudiney/penjelasan-sederhana-tentang-apa-itu-svm-149fec72bd02>. [Diakses 18 Juni 2022].
- [7] G. Delyani, "Machine Learning Python : Kenali Tentang Algoritma Support Vector Machine, Yuk!," *DQLab*, 4 April 2021. [Online]. Available: <https://www.dqlab.id/kenali-tentang-algoritma-support-vector-machine>. [Diakses 15 Juni 2022].
- [8] M. M. Haekal, "Apa Itu Agile? Pengertian, Prinsip, Metode, dan Kelebihan [Terlengkap]," 30 Juli 2022. [Online]. Available: <https://www.niagahoster.co.id/blog/agile-adalah/>. [Diakses 18 Juni 2022].
- [9] Lucid Content Team, "How to create software design documents," *Lucidchart*, [Online]. Available: <https://www.lucidchart.com/blog/how-to-create-software-design-documents>. [Diakses 15 Juni 2022].
- [10] S. Priharto, "Staging Environment: Pengertian, Manfaat dan Batasannya," *Aksaragama*, 5 April 2021. [Online]. Available: <https://aksaragama.com/teknologi/staging-environment/>. [Diakses 15 Juni 2022].