

# Analisis dan Pengembangan *Reporting* untuk Manajemen Operasional Pabrik Berbasis Web *Application*

1<sup>st</sup> Fernaldi Angga Dell Rizky  
Fakultas Rekayasa Industri  
Universitas Telkom  
Bandung, Indonesia

fernaldiangga@student.telkomuniversity.ac.id

2<sup>nd</sup> Tien Febrianti Kusumasari  
Fakultas Rekayasa Industri  
Universitas Telkom  
Bandung, Indonesia

tienkusumasari@telkomuniversity.ac.id

3<sup>rd</sup> Ekky Novriza Alam  
Fakultas Rekayasa Industri  
Universitas Telkom  
Bandung, Indonesia

ekkynovrizalam@telkomuniversity.ac.id

**Abstrak**—Perkembangan teknologi di bidang industri banyak dimanfaatkan perusahaan dalam meningkatkan efektivitas dan efisiensi produksi. Peningkatan ini dapat dilakukan dengan *monitoring* dan *controlling*. Untuk itu, teknologi yang dapat dimanfaatkan dalam memudahkan proses *monitoring* dan *controlling* pada tingkat manajemen operasional adalah dengan melakukan *reporting*. Namun saat ini terdapat permasalahan mengenai pelaporan *complain* dan *feedback* dari pelanggan. Beberapa *complain* yang diakibatkan produk yang tidak sesuai dengan *standard* dapat diperhatikan oleh operasional pabrik. Solusi yang diberikan untuk membantu menyelesaikan permasalahan yang dimiliki oleh operasional pabrik adalah dengan mengembangkan sistem informasi manajemen operasional pabrik bernama Aplikasi *reporting* operasional manajemen pabrik. Oleh karena itu diperlukan pengembangan sebuah aplikasi berbasis *website* dengan nama Aplikasi *reporting* untuk manajemen operasional pabrik dengan harapan dapat memecahkan permasalahan yang sedang dihadapi oleh operasional pabrik. Aplikasi *reporting* yang dikembangkan akan memberikan fitur-fitur yang dapat dimanfaatkan oleh pihak operasional pabrik dalam mendapatkan informasi mengenai pelaporan *complain* yang terjadi ketika pelanggan melakukan pembelian produk dan *feedback* yang diberikan dari pelanggan serta pelanggan dapat lebih mudah mendapatkan informasi mengenai *complain* yang sedang terjadi di produk yang telah dibeli.

**Kata kunci**—*reporting*, *website*, *monitoring*, *controlling*, *feedback*, *complain*..

## I. PENDAHULUAN

Perkembangan teknologi di bidang industri banyak dimanfaatkan perusahaan – perusahaan dalam meningkatkan efektivitas dan efisiensi produksi. Terdapat dua aspek teknologi yaitu teknologi informasi dan teknologi komunikasi (Aziz, 2012). Teknologi informasi mencakup hal tentang proses, manipulasi, dan pengelolaan informasi. Teknologi komunikasi merupakan segala hal yang berkaitan dengan pengelolaan data (send and receive). Salah satu pemanfaatan teknologi pada bidang industri dapat digunakan pada manajemen operasi agar

menciptakan produk yang berkualitas dan kemudahan dalam mengkoordinasikan sumber daya (Mcfarlane, 2014). Masalah yang sering terjadi dan dapat diselesaikan dengan memanfaatkan teknologi pada manajemen operasi adalah masalah *monitoring* dan *controlling*. Menurut North Dakota State University, *monitor* dan *control* adalah proses pengukuran kinerja dan pengambilan tindakan korektif untuk memastikan bahwa keberjalanan bisnis benar dan lancar sesuai dengan tujuannya. *Monitoring* dan *controlling* digunakan untuk mengawasi proses bisnis yang terjadi khususnya pada level manajemen operasional dan mengontrol kualitas produksi. Perusahaan yang belum memanfaatkan teknologi memerlukan usaha dan upaya yang cukup besar dalam proses pelaporan tiap bulan dan untuk mendapatkan status terkini dari data-data yang berkaitan dengan perencanaan produksi, proses produksi, kualitas produksi, dan hasil produksi. Untuk itu, Teknologi yang dapat dimanfaatkan dalam mempermudah proses *monitoring* dan *controlling* pada level manajemen operasional adalah dengan melakukan *reporting*.

*Reporting* merupakan sebuah metode pelaporan yang dapat membantu pengguna dalam melaporkan sebuah informasi yang berguna bagi para manager atau pemegang kepentingan dari sebuah perusahaan. Semakin cepat kinerja sebuah tim dalam melakukan agenda *Reporting* dapat mempermudah dalam mengambil keputusan bagi seorang manager atau pemegang kepentingan/stakeholder untuk mengembangkan kinerja pada sebuah perusahaan agar lebih baik. Pengembangan sistem informasi penting karena membungkus kompleksitas proses bisnis yang mudah digunakan. Metrik, prediksi, serta nilai aktual atau keputusan yang dihasilkan oleh proses bisnis perusahaan diperlukan untuk pengembangan kinerja untuk mengetahui kondisi proses bisnis pada setiap unit. Dengan ditunjang berbagai jenis data yang ditampilkan menjadi informasi yang

berguna, penggunaan metode reporting diharapkan dapat membantu setiap pengambilan keputusan menjadi lebih tajam dan mampu mengoptimalkan kinerja dan tanggungjawab perusahaan.

II. KAJIAN TEORI

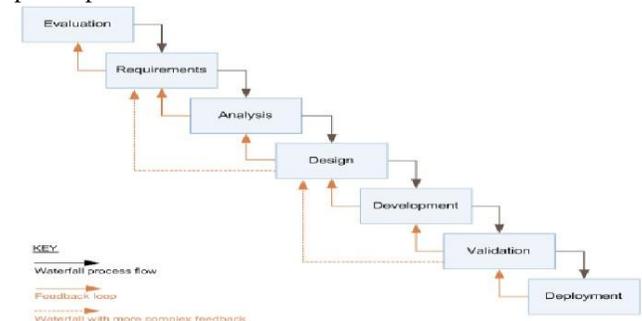
A. Laporan (*Reporting*)

Menurut Cambridge Dictionary, “Report” (laporan) merupakan deskripsi formal secara tertulis atas sekumpulan informasi-informasi yang dimaksudkan untuk disampaikan kepada seseorang yang bersangkutan. Dapat pula berupa hasil penelitian tentang sesuatu yang memerlukan informasi yang pasti yang dipublikasikan oleh sebuah institusi yang membuatnya. Reporting (pelaporan) dalam buku Papers on the Science of Administration merupakan salah satu fungsi manajemen berupa penyampaian perkembangan atau hasil kegiatan atau pemberian keterangan mengenai segala hal yang bertalian dengan tugas dan fungsi-fungsi kepada pejabat yang lebih tinggi baik secara lisan maupun tertulis sehingga dalam penerimaan laporan dapat memperoleh gambaran bagaimana pelaksanaan tugas orang yang memberi laporan (Luther M. Gullick, 2012). Selain itu, pelaporan merupakan catatan yang memberikan informasi tentang kegiatan tertentu dan hasilnya disampaikan ke pihak yang berwenang atau berkaitan dengan kegiatan tertentu (Siagina, 2013).

B. Software Development Life Cycle

Software Development Life Cycle (SDLC) adalah akronim yang digunakan untuk menggambarkan life-cycle pengembangan perangkat lunak (Ruparelia, 2010). System Development Life Cycle (SDLC) adalah metode pengembangan sistem tradisional yang banyak digunakan saat ini. SDLC merupakan framework (kerangka kerja) terstruktur yang berisi proses – proses sekuensial dimana sistem informasi dikembangkan. Terdapat beberapa model SDLC yang dapat digunakan pada pengembangan perangkat lunak diantaranya adalah Waterfall Model dan Scrum Model. The waterfall model, juga dikenal sebagai cascade model, pertama kali didokumentasikan oleh Benington pada tahun 1956 dan dimodifikasi oleh Winston Royce pada tahun 1970. Model ini telah menopang semua model lainnya sejak ia menciptakan landasan yang kokoh untuk persyaratan yang akan didefinisikan dan dianalisis sebelum mendesain atau mengembangkan apa pun. Cascade model asli dari Benington merekomendasikan bahwa perangkat lunak dikembangkan secara bertahap: analisis operasional > spesifikasi operasional > spesifikasi desain dan

pengkodean > pengembangan > pengujian > penerapan > evaluasi.



GAMBAR 2.1 WATERFALL MODEL

Scrum model merupakan model Software Development Life Cycle (SDLC) yang lunak dan responsive serta berbasis metodologi agile yang mampu memberikan nilai dan manfaat yang baik terhadap software yang dikembangkan. Model Scrum dalam pelaksanaannya tidak hanya sebagai model untuk pengembangan perangkat lunak melainkan lebih ke manajemen pengembangan perangkat lunak sehingga seluruh tim yang terlibat dapat secara mudah mengontrol tugas-tugas yang ada supaya kinerja menjadi lebih efektif dan efisien. Model ini juga banyak digunakan oleh beberapa praktisi teknologi informasi karena dapat mengetahui secara cepat dan tepat dan kualitas perangkat lunak yang dikembangkan (Suharno, Gunantara, & Sudarma, 2020). Gambar 2 menunjukkan proses dari SDLC Scrum.



GAMBAR 2.2 SCRUM MODEL

Waterfall Model dan Scrum Model pasti memiliki kelebihan dan kekurangannya masing – masing. Semua model SDLC adalah baik jika digunakan pada project yang tepat. Tabel 1 menunjukkan perbandingan antara Waterfall Model dan Scrum

TABEL 1. 1  
TABEL PERBANDINGAN DAN PEMILIHAN SDLC MODEL

| Point          | Scrum Model              | Waterfall Model           | Penelitian ini  |
|----------------|--------------------------|---------------------------|---|
| Based On       | Value                    | Fixed Budget, Time, Scope | Penelitian merupakan peneliti yang hanya dilakuk sekali, jadi lebih b menggunakan f daripada sprints ya berkelanjutan |
| Flexibility    | High                     | Low                       | Penelitian merupakan tu akhir yang han dikerjakan sekali d selesai Tidak terdapat dalam pengerj: tugas akhir ini      |
| Deliverable    | Increment                | In the end of             |   |
| Risk           | Relatively low           | High for all              |   |
| Time to Market | Fast                     | Slow                      |   |
| Quality        | Using Definition of Done | On Time Priority          |   |

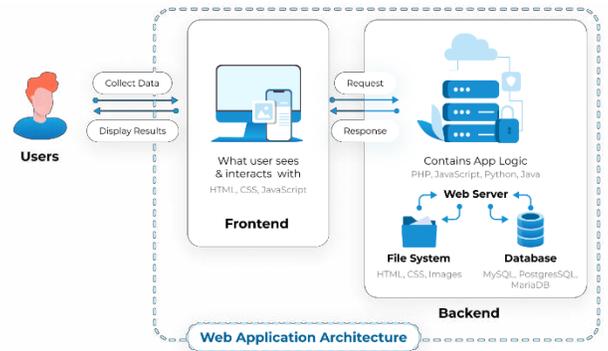
Model SDLC yang dipilih: *Scrum Model*

C. Design Science Research

Design Science Research berakar pada Teknik dan ilmu pengetahuan (Simon 1996). Pada dasarnya adalah paradigma pemecahan masalah. DSR berusaha untuk meningkatkan pengetahuan manusia dengan penciptaan artefak inovatif dan generasi dari Design Knowledge melalui solusi inovatif untuk masalah dunia nyata (Hevner, March, Park, & Ram 2004). Dengan ini paradigma penelitian telah menghasilkan peningkatan minat dalam 20 tahun terakhir, khususnya karena potensi untuk berkontribusi dalam mendorong kemampuan inovasi organisasi serta berkontribusi pada transformasi keberlangsungan yang sangat dibutuhkan masyarakat. Tujuan dari proyek penelitian DSR adalah untuk memperluas batas-batas manusia dan organisasi kemampuan dengan merancang artefak baru dan inovatif yang diwakili oleh konstruksi, model, metode dan instansi. DSR bertujuan untuk menghasilkan pengetahuan tentang bagaimana hal-hal dapat dan harus dibangun untuk diatur atau dirancang untuk mencapai serangkaian tujuan yang diinginkan atau disebut sebagai Design Knowledge.

D. Web Application Architecture

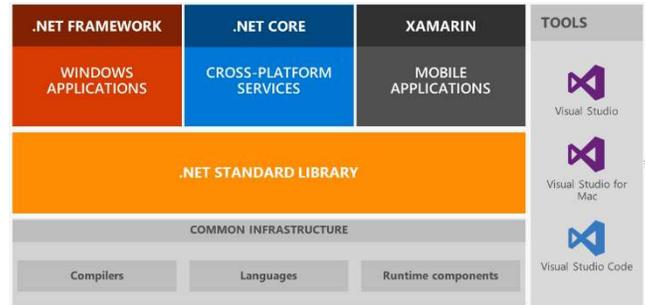
Web application architecture adalah sebuah blueprint interaksi simultan antara komponen, database, sistem middleware, user interface, dan server dalam suatu aplikasi (Dhaduk, 2021). Web application architecture juga menggambarkan tentang hubungan antara client side dengan server side. Web application architecture diperlihatkan pada Gambar 3 dimana dijelaskan tentang interaksi antara user dengan web application



GAMBAR 2. 3  
WEB APPLICATION ARCHITECTURE

Setelah pengguna membuka halaman web, server akan mengirimkan data tertentu ke browser sebagai response dari request yang dilakukan oleh user. Lebih tepatnya, web client dapat meminta web resource atau yang lebih dikenal sebagai web document (HTML, JSON, PDF, etc) melalui web server. Setelah serangkaian proses request dilakukan, informasi yang diminta akan muncul. Setelah itu, interaksi antara pengguna dan web akan dimulai.

E. ASP.NET Core



GAMBAR 2. 4  
ASP.NET CORE

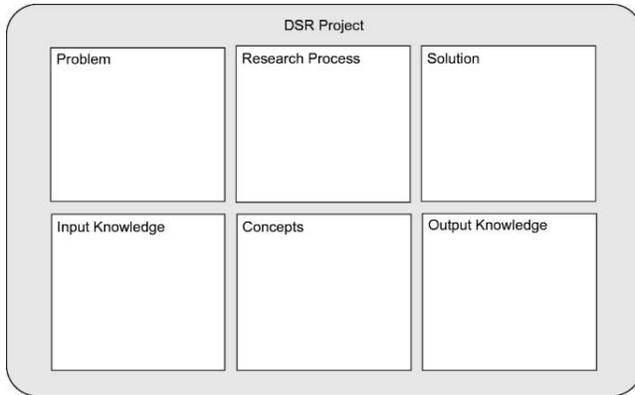
Dilansir langsung dari website milik Microsoft, ASP.NET merupakan web – development framework yang populer untuk membangun web application pada platform .NET. ASP.NET Core adalah versi open – source dari ASP.NET, yang berjalan secara cross platform di macOS, Linux, dan Windows. ASP.NET Core pertama kali dirilis pada tahun 2016 dan merupakan desain ulang dari versi ASP.NET khusus Windows sebelumnya. Versi sebelumnya dari ASP.NET Core adalah ASP.NET Framework. ASP.NET Framework sebenarnya memiliki potensi yang besar untuk portable tetapi dalam perkembangannya terlalu banyak fragmentasi, hanya khusus berjalan di Windows, dan berbayar (Equan, 2017).

III. METODE

A. Design Science Research

Design Science Research Grid (Brocke & Maedche 2019) memungkinkan peneliti untuk secara efektif merencanakan, mengkoordinasikan dan mengkomunikasikan proyek DSR mereka. Kini DSR

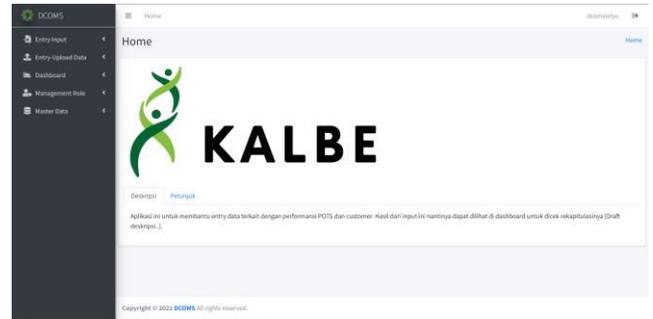
bermaksud untuk menempatkan seluruh DSR proyek pada satu halaman, menyoroti komponen penting untuk mencerminkan dan berkomunikasi sesuai ruang lingkungannya. Respresentasi proyek DSR semacam itu membantu merencanakan dan mengkomunikasikan DSR proyek dengan lebih baik serta untuk menerima umpan balik dari pemangku kepentingan yang berbeda pada tahap awal dan untuk mempertanyakan dan memperbarui ruang lingkup saat proyek berlangsung. Seperti yang ditunjukkan dibawah ini, Grid DSR terdiri dari enam dimensi terpenting dari proyek DSR.



GAMBAR 3.1  
DESIGN SCIENCE RESEARCH

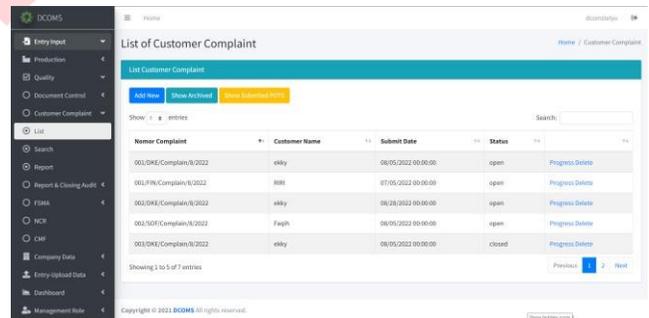
1. **Problem Description:** Pada studi kasus ini pabrik pada perusahaan ini mengalami sebuah permasalahan dimana belum adanya agenda pelaporan pada setiap kegiatan yang telah dilakukan.
2. **Input Knowledge:** Pada studi kasus ini tim pada proyek diharapkan mengerti permintaan stakeholder seperti teknologi yang sesuai dengan proyek ini, dan mengadaptasi dari aplikasi yang sudah ada.
3. **Research Process:** Pada studi kasus ini kegiatan pertama yang dilakukan adalah wawancara kemudian melakukan analisis design, dilanjut dengan design mockup menggunakan figma lalu melakukan validasi design, prototyping online kemudian yang terakhir validasi design.
4. **Key Concepts:** Pada studi kasus ini produk adalah kunci dari didirikannya aplikasi ini yaitu dengan menggabungkan beberapa divisi yang berbeda beda agar mudah melakukan pemasaran
5. **Solution Description:** Pada studi kasus ini solusi yang ditawarkan yaitu membuat reporting yang dapat menampung data produk dari berbagai divisi agar menjadi satu untuk dapat dilaporkan. Hal ini memudahkan perekapan data yang mana terpusat hanya pada aplikasi tersebut.
6. **Output Knowledge:** Pada studi kasus ini output knowledge yang dihasilkan adalah hasil dari pengembangan mockup yang telah didesign agar menjadi reporting pada manajemen pabrik di sebuah perusahaan.

IV. HASIL DAN PEMBAHASAN

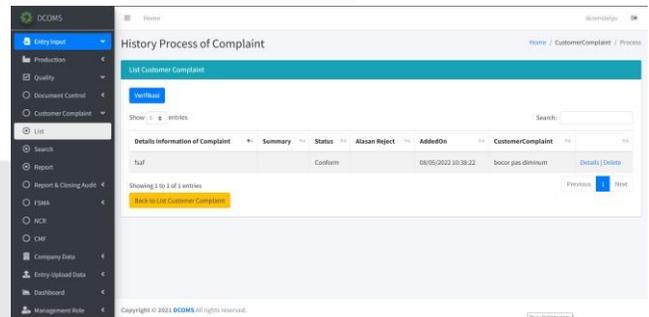


GAMBAR 4.1  
TAMPILAN DASHBOARD UTAMA

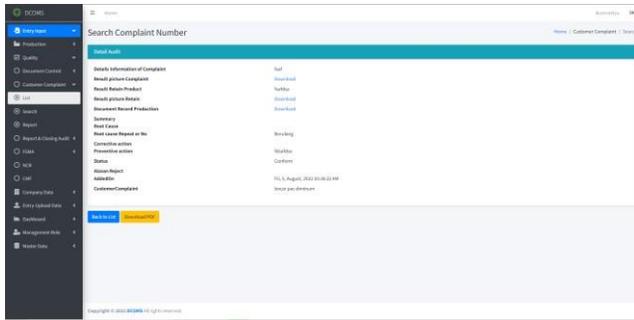
Pada halaman utama dashboard pabrik ini dapat kita lihat terdapat beberapa sub tab pada bagian kiri yang dimana kita dapat melakukan input, upload data, melihat dashboard, mengelola role dan melihat master data. Selanjutnya masuk kedalam sub tab input data. Kita dapat menginput data yang diinginkan melalui halaman tersebut. Pada gambar dibawah ini ditampilkan halaman sub tab *quality* yang memiliki beberapa tampilan lain seperti *Customer Complaint*.



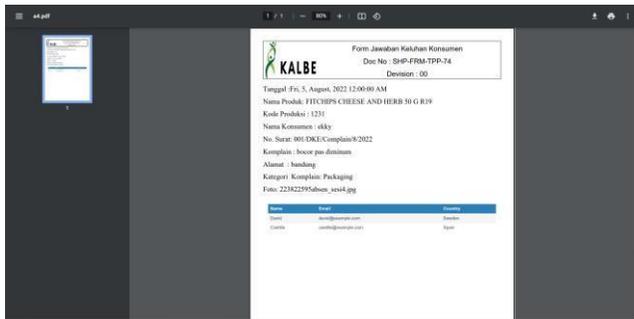
GAMBAR 4.2  
TAMPILAN DASHBOARD CUSTOMER COMPLAINT



GAMBAR 4.3  
TAMPILAN DASHBOARD PROCESS CC



GAMBAR 4. 4  
TAMPILAN DASHBOARD DETAIL CC



GAMBAR 4. 5  
TAMPILAN PDF COSTUMER COMPLAINT

V. KESIMPULAN

Penelitian ini memiliki hasil sebuah aplikasi reporting pabrik yang memberikan informasi tentang komplain terhadap produk yang telah dibeli oleh pelanggan. dan memberikan kemudahan bagi pekerja untuk melakukan generate report untuk mengambil keputusan terhadap komplain yang telah diajukan oleh konsumen. Tampilan pada aplikasi reporting pabrik juga dibuat atas penelitian hasil wawancara dengan metode user story yang dilakukan begitu juga dengan fitur yang disediakan. Dalam proses implementasi pembangunan front-end aplikasi reporting pabrik menggunakan metode DSR(Design Science Research) sangat membantu pengembangan dalam aplikasi.

Penelitian ini melakukan pengujian front-end pada aplikasi reporting pabrik menggunakan Blackbox Testing dimana pengujian ini sangat membuktikan aplikasi reporting pabrik berjalan dengan baik dan sesuai dengan alur proses bisnis yang berjalan. Dapat dilihat hasil tersebut penelitian ini dapat dinilai sudah berjalan dengan baik sesuai dengan harapan yang telah dibangun di awal penelitian ini dimulai

REFERENSI

[1] Adani, M. (2020). Penjelasan Lengkap Metode Agile dalam Pengembangan Perangkat Lunak. Sekawan Media.  
 [2] Ambler, T., & Clark, B. H. (2009). Dashboards as a Service. May 2014.

[3] Aziz, A. (2012). Pemanfaatan Teknologi Informasi dalam Pengembangan Bisnis Pos information technology utilization in business post. 10(1), 35–50.  
 [4] Bugwandeen, K., & Ungerer, M. (2019). EXPLORING THE DESIGN OF PERFORMANCE DASHBOARDS IN RELATION TO ACHIEVING ORGANISATIONAL STRATEGIC GOALS. 30(August), 161–175.  
 [5] Daremuk, I. (2021). Modern Web Application Architecture Explained: Components, Best Practices and More. Litslink.  
 [6] Dhaduk, H. (2021). An Ultimate Guide of Web Application Architecture. Simform.  
 [7] Equan. (2017). Perlukah belajar .NET Core? Medium.  
 [8] Febtriko, A., & Puspitasari, I. (2018). Mengukur Kreatifitas Dan Kualitas Pemograman Pada Siswa Smk Kota Pekanbaru Jurusan Teknik Komputer Jaringan Dengan Simulasi Robot. Rabit : Jurnal Teknologi Dan Sistem Informasi Univrab, 3(1), 1–9.  
 [9] Hanselman, S. (2016). An update on ASP.NET Core 1.0 RC2. Hanselman.  
 [10] Hasanah, H. (2017). TEKNIK-TEKNIK OBSERVASI (Sebuah Alternatif Metode Pengumpulan Data Kualitatif Ilmu-ilmu Sosial). At-Taqaddum, 8(1), 21.  
 [11] Kurniawan, A. (2021). Kuesioner adalah Instrumen untuk Mengumpulkan Data, Ketahui Jenis dan Kelebihannya. Merdeka.Com.  
 [12] Mcfarlane, D. a. (2014). The challenges of operations management for business managers. International Journal of Operations and Logistics Management, 29(March), 16–29.  
 [13] Peterheria, N. (2018). Why Agile SDLC Model Is The Best for Your Startup. Product Tribe.  
 [14] Rahmayudha, S. (2017). Perancangan Model Dashboard Untuk Monitoring Evaluasi Mahasiswa. Jurnal Informatika:Jurnal Pengembangan IT (JPIT), 2(1), 13–17.  
 [15] Ruparelia, N. B. (2010). Software development lifecycle models. ACM SIGSOFT Software Engineering Notes, 35(3), 8–13.  
 [16] Setiawan, T. (2021). Uji Reliabilitas Cronbach’s Alpha Kuesioner Penelitian. Malaskuliah.  
 Velcu-laitinen, O., & Yigitbasioglu, O. M. (2012). The Use of Dashboards i Performance Management : Evidence from

Sales Managers. 12(November 2011), 39–58.

