RANCANG BANGUN SISTEM INFORMASI PROYEK BERBASIS WEB

Oleh

Dimas Dwi Cahyo 1702014

Malsi Nur Adwinda Robbani 1702034

DIPLOMA TIGA

pada

PROGRAM STUDI TEKNIK ELEKTRONIKA

POLITEKNIK GAJAH TUNGGAL

Jln. Gatot Subroto KM 7, Pasir Jaya, Jatiuwung

Tangerang 15135 – Indonesia

Juli 2020

LEMBAR PENGESAHAN

Dengan ini saya menyatakan bahwa tugas akhir yang saya kumpulkan ini adalah murni hasil karya saya sendiri dan sejauh pengetahuan terbaik saya, di dalamnya tidak terdapat materi yang pernah dipublikasikan atau ditulis oleh orang lain sebelumnya, tidak juga terdapat materi yang pernah mendapatkan penghargaan atau digunakan untuk mendapat gelar akademik atau diploma di institusi pendidikan lainnya, kecuali yang dinyatakan di dalam tugas akhir ini.

Diterima dan dinyatakan memenuhi syarat kelulusan pada tanggal …….

Pembimbing TA	**Mentor Pabrik**
M. Ridwan Arif C., S.T., M.T	**Yanuar Anggit E. N., S.T**
NIP. 13-0745	**NIP.**
Penguji I	**Penguji II**
NIP.	**NIP.**

Disetujui Kaprodi Teknik Elektronika

M. Ridwan Arif C., S.T., M.T

NIP. 13-0745

Diketahui Direktur Politeknik Gajah Tunggal

Dr. Ita Mariza

NIP. 11-0143

ABSTRAK

RANCANG BANGUN SISTEM INFORMASI PROYEK BERBASIS WEB

Oleh

Dimas Dwi Cahyo
Malsi Nur Adwinda Robbani
M. Ridwan Arif C., S.T., M.T., Pembimbing Utama
Yanuar Anggit E. N., S.T., Mentor Pendamping

POLITEKNIK GAJAH TUNGGAL

Informasi proyek merupakan hal yang penting dalam sebuah perusahaan, dimana informasi proyek diperlukan untuk membantu perusahaan dalam pengambilan keputusan untuk perencanaan proyek selanjutnya. Dalam pelaksanaannya PT.UTL melakukan pendataan informasi proyek menggunakan Macro Exel yang memiliki beberapa masalah dalam hal pengelolaannya. Masalah tersebut seperti tidak dapatnya staf melakukan *input* di lapangan, terdapat beberapa dokumen yang hilang, dan ada beberapa kontraktor yang selalu menang tender dikarenakan biaya yang murah tetapi hasil pekerjaan proyek kurang baik. Penelitian ini bertujuan untuk menyelesaikan masalah tersebut.

Sistem informasi proyek berbasis web ini di rancang menggunakan metode *waterfall* dengan melakukan perancangan database dan membuat *Unified Modeling Language* (UML) seperti *Use case Diagram, Class Diagram, Activity Diagram* dan *Sequence Diagram.* MySQL digunakan untuk basis datanya dan pengembangan web menggunakan *framework* Laravel dengan bahasa program PHP dan Javascript.

Hasil penelitian berupa sistem informasi proyek berbasis web yang dapat membantu kebutuhan PT. UTL, diantaranya terdapat fitur laporan proyek berbasis web, proses dokumentasi berbasis *database cloud* baik berupa foto atau file, dan penilaian kinerja kontraktor.

Kata kunci : *Unified Modeling Language, Framework*, Laravel, PHP, Javascript.

© Copyright 2020
Oleh Dimas Dwi Cahyo, Malsi Nur Adwinda Robbani
Hak cipta dilindungi

LEMBAR PERSEMBAHAN

Tugas Akhir ini kami dedikasikan untuk kedua orang tua, adik dan kakak kami, serta semua orang yang turut mendukung dan mendoakan kami sehingga Tugas Akhir ini dapat diselesaikan tepat waktu, untuk Politeknik Gajah Tunggal dimana tempat saya menuntut ilmu, untuk kemajuan Politeknik Gajah Tunggal dan Gajah Tunggal *Group* serta untuk negara Indonesia tercinta.

PENGHARGAAN

Puji syukur penulis ucapkan kepada Allah SWT, berkat rahmat dan karunia-Nya penulis mampu menyelesaikan Tugas Akhir ini. Tidak lupa juga penulis ingin menyampaikan rasa terimakasih yang sebesar-besarnya kepada semua pihak yang membantu kelancaran penulisan Tugas Akhir ini. Penulis menyampaikan ucapan terimakasih dan penghargaan yang setinggi-tingginya kepada :

1. Kedua orang tua dan keluarga penulis, yang senantiasa memberikan doa dan dukungannya.
2. PT Gajah Tunggal Tbk yang telah memberikan fasilitas beasiswa pendidikan secara penuh kepada penulis sehingga penulis mampu melanjutkan pendidikan Diploma Tiga di Politeknik Gajah Tunggal.
3. Ibu Dr. Ita Mariza, selaku Direktur Politeknik Gajah Tunggal yang telah memperjuangkan dan memfasilitasi penulis untuk melanjutkan pendidikan di kampus Politeknik Gajah Tunggal.
4. Bapak Muhammad Ridwan Arif Cahyono selaku Kepala Jurusan Teknik Elektronika dan pembimbing penulis di kampus Politeknik Gajah Tungal.
5. Bapak Lucky Ginanjar, selaku *Departement Head Joint Maintenance Utility* yang telah mengizinkan penulis melakukan penelitian di Departemen *Joint Maintenance Utility*.
6. Bapak Yanuar Anggit Eko Nugroho, selaku pembimbing dalam penulisan Tugas Akhir ini.
7. Seluruh dosen dan staf yang membantu selama proses pendidikan di Politeknik Gajah Tunggal.
8. Seluruh staf dan karyawan Departemen *Joint Maintenance Utility* atas sambutan dan kerja sama yang baik selama penulis melaksanakan Penelitian.

Penulis berterimakasih kepada semua pihak yang telah terlibat dalam penulisan Tugas Akhir ini. Penyusunan Tugas Akhir ini tidak dapat selesai tanpa adanya bantuan dari pihak-pihak tersebut. Penulis menyadari masih banyak kekurangan di dalam diri penulis, sehingga penulis juga memohon maaf apabila terdapat kesalahan yang selama ini penulis lakukan baik disengaja maupun tidak disengaja.

DAFTAR ISI

LEMBAR PENGESAHAN 2
ABSTRAK 3
LEMBAR PERSEMBAHAN 5
PENGHARGAAN 6
DAFTAR ISI 7
DAFTAR GAMBAR 9
DAFTAR TABEL 12

BAB 1 - PENDAHULUAN 13
1.1. Latar Belakang 13
1.2. Tujuan Kajian 14
1.3. Batasan Masalah 14
1.4. Manfaat Penelitian 15

BAB 2 – DASAR TEORI KAJIAN 16
2.1. Teori Dasar Pemrograman 16
2.1.1. Bahasa Pemrograman 16
2.1.2. *Framework* 19
2.1.3. Basis data 20
2.2. Teori Dasar Sistem 23
2.3. Metode Pengembangan. 24
2.3.1. Metode *Waterfall* 24
2.3.2. *Unified Modeling Language* (UML) 26

BAB 3 – KONSEP PERANCANGAN PROGRAM 28
3.1. Perencanaan *Database* 28
3.2. *Unified Modeling Language* (UML) 44
3.2.1. *Use Case Diagram* 44
3.2.2. *Entity Relational Diagram* (ERD) 45
3.2.3. *Activity Diagram* 48
3.2.4. *Squence Diagram* 48
3.2.5. Desain *Backlog* 49

BAB 4 - HASIL KAJIAN DAN PEMBAHASAN 51
4.1 Tampilan Web 51
4.1.1 *Login* 51
4.1.2 *Dashboard* 51
4.1.3 Tambah Proyek 53
4.1.4 Detail Proyek 55
4.1.5 Riwayat Proyek 58
4.1.6 Proyek Batal 60
4.1.7 Kalender 61
4.1.8 *Rating* Kontraktor 61
4.2 Uji Verifikasi 62
4.3 Uji Validasi 63

BAB 5 - KESIMPULAN DAN SARAN 76
5.1. Kesimpulan 76
5.2. Saran 76

DAFTAR PUSTAKA 77
BIODATA PENULIS 78
LAMPIRAN 80

DAFTAR GAMBAR

Gambar 1. Arsitektur Model AJAX 19

Gambar 2. Panel Kontrol XAMPP 22

Gambar 3. Database phpMyadmin XAMPP 23

Gambar 4. Kurva S 24

Gambar 5. Model Waterfall 25

Gambar 6. Metode 28

Gambar 7. *Use Case Diagram* 44

Gambar 8. *Entity Relational Diagram* 45

Gambar 9. *Activity Diagram* 48

Gambar 10. *Sequence Diagram* 49

Gambar 11. Halaman *Login* 51

Gambar 12. Halaman *Dashboard 1* 52

Gambar 13. Halaman *Dashboard 2* 52

Gambar 14. Halaman *Dashboard 3* 52

Gambar 15. Halaman Tambah Proyek 1 53

Gambar 16. Halaman Tambah Proyek 2 53

Gambar 17. Halaman Tambah Bill of Quantity 54

Gambar 18. Halaman Desain 54

Gambar 19. Halaman *Purchase Requisiton* 55

Gambar 20. Halaman Surat Perintah Kerja 55

Gambar 21. Halaman Proyek 56

Gambar 22. Halaman Detail Proyek 56

Gambar 23. Halaman Pembaharuan Proyek 57

Gambar 24. Halaman Tambah Surat Jalan 57

Gambar 25. Halaman Tambah Perkembangan Proyek 58

Gambar 26. Halaman Riwayat Proyek 58

Gambar 27. Halaman Detail Riwayat Proyek 59

Gambar 28. Halaman Beri *Rating* 59

Gambar 29. Halaman Detail Riwayat 1 60

Gambar 30. halaman Detail Riwayat 2 60

Gambar 31. Halaman Proyek Batal 61

Gambar 32. Halaman Kalender 61

Gambar 33. Halaman Rating Kontraktor 62

Gambar 34. Halaman Detail Rating Kontraktor 62

Gambar 35. Validasi Tambah Proyek 1 64

Gambar 36. Validasi Tambah Proyek 2 64

Gambar 37. Validasi Tambah Proyek 3 65

Gambar 38. Validasi Tambah *Item BOQ 1* 65

Gambar 39. Validasi Tambah *Item BOQ 2* 66

Gambar 40. Validasi Tambah *Item BOQ 3* 66

Gambar 41. Validasi Tambah Gambar 1 66

Gambar 42. Validasi Tambah Gambar 2 67

Gambar 43. Validasi Tambah Gambar 3 67

Gambar 44. Validasi Tambah *Purchase Requisition* 1 68

Gambar 45. Validasi Tambah *Purchase Requisition* 2 68

Gambar 46. Validasi Tambah *Purchase Requisition* 2 69

Gambar 47. Validasi Tambah *Purchase Requisition* 3 69

Gambar 48. Validasi *Edit* Proyek 1 70

Gambar 49. Validasi *Edit* Proyek 2 70

Gambar 50. Validasi *Edit Item BOQ* 1 71

Gambar 51. Validasi *Edit Item BOQ* 2 71

Gambar 52, Validasi *Edit Item BOQ 3* 71

Gambar 53. Validasi Tambah Perkembangan Proyek 1 72

Gambar 54. Validasi Tambah Perkembangan Proyek 2 72

Gambar 55. Validasi Tambah *Rating* 1 73

Gambar 56. Validasi Tambah *Rating* 2 73

Gambar 57. Validasi perhitungan boq 1 73

Gambar 58. Validasi perhitungan boq 2 74

Gambar 59. Validasi perhitungan boq 3 74

DAFTAR TABEL

Tabel 1. User 29

Tabel 2. *Project* 30

Tabel 3. *Bill of Quantity* 32

Tabel 4. *Item* 33

Tabel 5. *Drawing* 35

Tabel 6. *Purchase Requisition* 36

Tabel 7. SPK 37

Tabel 8. *Contractor* 38

Tabel 9. *Rating* 39

Tabel 10. *Arrive* 40

Tabel 11. Progres 41

Tabel 12. Perkembangan 42

Tabel 13. Uji Verifikasi 63

BAB 1 - PENDAHULUAN

1.1. Latar Belakang

PT UTL merupakan salah satu perusahaan produsen ban yang memiliki hasil produksi beragam, mulai dari ban bias, ban dalam *truck*, ban sepeda motor, ban dalam sepeda motor, ban PCR (*Passanger Car Radial*), ban TBR (*Truck Bus Radial*). Dalam satu hari PT UTL dapat memproduksi puluhan hingga ratusan ribu ban per hari dengan *size* yang berbeda-beda. Untuk memproduksi ban sebanyak itu Departemen Produksi didukung departemen-departemen lain salah satunya adalah Departemen Joint Maintenance Utility yang bertugas untuk menyediakan dan menjaga ketersediaan pasokan energi listrik, hot water, steam, maupun energi lainya.

Departemen *Joint Maintenance Ut*ility yang selanjutnya akan disebut Departemen JMU memiliki divisi-divisi utama yang salah satunya adalah Divisi *Planning & Improvement* yang bertugas untuk melakukan perencanaan dan perbaikan untuk menjaga kehandalan pasokan energi. Untuk mencapat hal tersebut Divisi *Planning & Improvement* yang selanjutnya akan disebut Divisi PI sering melakukan proyek-proyek perbaikan mesin maupun prasarana lain yang menunjang kehandalan mesin. Proyek-proyek tersebut tidak dikerjakan oleh pihak Divisi PI maupun Departemen JMU, melainkan dikerjakan oleh kontraktor dari luar PT UTL.

Dalam pelaksanaan *monitoring* proyek, Divisi PI menggunakan sistem Macro Excel yang dikombinasikan dengan grafik kurva s untuk mempermudah dalam proses penilaian persentase proyek dan pemantauan pekerjaan proyek. Divisi PI juga memiliki satu orang staf yang berwenang sebagai administrator dalam peng-input-an informasi persentase dan perkembangan proyek. Proses dokumentasi arsip-arsip pendukung seperti *Bill of Quantity*, Purchase Requisition, Surat Perintah Kerja dan arsip pendukung lainya disimpan di dalam lemari arsip yang berada di kantor Departemen JMU. Proses pembaharuan informasi proyek dilakukan per minggu terhitung satu minggu setelah Surat Perintah Kerja diterbitkan.

Dalam prakteknya, pelaksanaan monitoring pada sistem berjalan masih memiliki beberapa kendala yaitu :

1. Informasi perkembangan proyek dinilai kurang *realtime* karena tidak bisa dilakukan secara langsung oleh pihak-pihak yang berada di lapangan.
2. Pembaharuan data informasi proyek kurang fleksibel karena harus dilakukan di kantor Departemen JMU.
3. Banyak dokumen dan arsip-arsip yang hilang yang menyebabkan proses audit menjadi terganggu.
4. Proses evaluasi pekerjaan kontraktor tidak dapat dilakukan dalam sistem Macro Excel sehingga Divisi PI tidak bisa memberikan umpan balik kepada Departemen *Purchase Requisition* sehingga kontraktor-kontraktor hanya dipilih berdasarkan kuantitas harga yang ditawarkan.

Berdasarkan latar belakang di atas, penulis melakukan rancang bangun sistem informasi proyek berbasis web yang dapat diakses dimana saja dan kapan saja. Serta dengan adanya fitur-fitur tambahan seperti proses dokumentasi berbasis *cloud* baik berupa foto atau file dan penilaian kinerja kontraktor sehingga diharapkan pengelolaan informasi proyek menjadi lebih baik.

1.2. Tujuan Kajian

Tujuan penelitian ini adalah merancang bangun aplikasi sistem infromasi proyek berbasis web.

1.3. Batasan Masalah

Batasan masalah dalam penelitian ini antara lain :

1. Sistem hanya dirancang sesuai kebutuhan di Departemen *Joint Maintenance Utility*.
2. Penelitian ini tidak membahas tentang sistem pada sisi gudang.
3. Penelitian ini tidak membahas kebutuhan perangkat keras *server* dan sistem.
4. Penelitian ini tidak membahas biaya perancangan dan biaya kebutuhan perangkat keras sistem.

1.4. Manfaat Penelitian

Penelitian ini memberikan manfaat sebagai berikut:

1. Proses pembaharuan informasi proyek tidak hanya dapat dilakukan di kantor Departemen JMU.
2. Mempermudah proses dokumentasi arsip-arsip proyek.
3. Dapat memberikan umpan balik kepada Departemen *Purchase Requisition* dalam penentuan kontraktor proyek.

BAB 2 – DASAR TEORI KAJIAN

2.1. Teori Dasar Pemrograman

1. HTML (*Hypertext Markup Language*)

Hypertext Markup Language (HTML) adalah sebuah bahasa *markup* yang digunakan untuk membuat sebuah halaman web, menampilkan berbagai informasi di dalam sebuah Penjelajah web Internet dan *formating hypertext* sederhana yang ditulis kedalam berkas format ASCII agar dapat menghasilkan tampilan wujudyang ter*integer*asi. Dengan kata lain, berkas yang dibuat dalam perangkat lunak pengolah kata dan disimpan kedalam format ASCII normal sehingga menjadi *home page* dengan perintah-perintah HTML. Bermula dari sebuah bahasa yang sebelumnya banyak digunakan di dunia penerbitan dan percetakan yang disebut dengan SGML(*Standard Generalized Markup Language*), HTML adalah sebuah standar yang digunakan secara luas untuk menampilkan halaman web. HTML saatini merupakan standar Internet yang didefinisikan dan dikendalikan penggunaannya oleh World Wide *Web* Consortium(W3C). HTML dibuat oleh kolaborasi Caillau TIM dengan Berners Lee Robert ketika mereka bekerja di CERN pada tahun 1989 (CERN adalah lembaga penelitian fisika energi tinggi di Jenewa). (Harison, 2016).

2. CSS (*Cascading Style Sheet*)

CSS kepanjangan dari *Cascading Style Sheet* adalah bahasa-bahasa yang merepresentasikan halaman web. Seperti warna, *layout*, dan *font*. Dengan menggunakan CSS, seorang web *developer* dapat membuat halaman web yang dapat beradaptasi dengan berbagai macam ukuran layar. Pembuatan CSS biasanya terpisah dengan halaman HTML. Meskipun CSS dapat disisipkan di dalam halaman HTML. Hal ini ditujukan untuk memudahkan pengaturan halaman HTML yang memiliki rancangan yang sama. (Henderson, 2009).

2.1.1. Bahasa Pemrograman

Pemrograman digunakan untuk proses menulis, menguji, memperbaiki, dan memelihara kode yang membangun sebuah program komputer. Tujuan dari

pemrograman adalah untuk membuat suatu program yang dapat melakukan suatu pekerjaan sesuai dengan keinginan si pemrogram. Bahasa pemrograman antara lain sebagai berikut:

1. PHP

PHP *(*Hypertext Preprocessor*)*, merupakan bahasa pemrograman pada sisi *server* yang memperbolehkan *programmer* menyisipkan perintah - perintah perangkat lunak web server (Apache, IIS, atau apapun) akan dieksekusi sebelum perintah itu dikirim oleh halaman ke *browser* yang menginginkannya, contohnya adalah bagaimana memungkinkannya memasukkan tanggal sekarang pada sebuah halaman web setiap kali tampilan tanggal dibutuhkan. Sesuai dengan fungsinya yang berjalan di sisi *server* maka PHP adalah bahasa pemrograman yang digunakan untuk membangun teknologi web *application*. (Kevin Yank, 2002)

PHP telah menjadi bahasa *scripting* untuk keperluan umum yang pada awalnya hanya digunakan untuk pembangunan web yang menghasilkan halaman web dinamis. Untuk tujuan ini, kode PHP tertanam ke dalam dokumen sumber HTML dan diinterpretasikan oleh server web dengan modul PHP *prosesor*, yang menghasilkan dokumen halaman web. Sebagai bahasa pemrograman untuk tujuan umum, kode PHP diproses oleh aplikasi penerjemah dalam modus baris - baris perintah modus dan melakukan operasi yang diinginkan sesuai sistem operasi untuk menghasilkan keluaran program di *channel output* standar. Hal ini juga dapat berfungsi sebagai aplikasi grafis. PHP tersedia sebagai prosesor untuk *server* web yang paling modern dan sebagai penerjemah mandiri pada sebagian besar sistem operasi dan komputer *platform*. (Februariyanti, 2012)

2. Javascript

JavaScript adalah bahasa pemrograman web yang bersifat *Client Side Programming Language*. *Client Side Programming Language* adalah tipe bahasa pemrograman yang pemrosesannya dilakukan oleh *client*. Aplikasi *client* yang dimaksud merujuk kepada web *browser* seperti Google Chrome, Mozilla Firefox, Opera Mini dan sebagainya.

JavaScript pertama kali dikembangkan pada pertengahan dekade 90'an. Meskipun memiliki nama yang hampir serupa, JavaScript berbeda dengan bahasa pemrograman Java. Untuk penulisannya, JavaScript dapat disisipkan di dalam dokumen HTML ataupun dijadikan dokumen tersendiri yang kemudian diasosiasikan dengan dokumen lain yang dituju. JavaScript mengimplementasikan fitur yang dirancang untuk mengendalikan bagaimana sebuah halaman web berinteraksi dengan penggunanya (Henderson, 2009).

3. JQuery

JQuery merupakan sebuah JavascriptLibrary atau bisa disebut juga sebagai perpustakaan dari kumpulan kode/*listing* Javascript yang siap pakai. Dalam arti sederhana, JQuery dapat digunakan untuk meringkas sebuah *listing*. Javascript yang panjang dalam sebuah proyek pembuatan *website*. Sehingga sebagai *developer web*, akan diberikan kemudahan dalam menghadapi bagian yang mengandung Javascript. JQuery merupakan program yang berjalan pada sisi *server* dan akan ditampilkan pada *web browser*. JQuery dapat berjalan didalam HTML, atau bahasa pemrograman berbasis *web* lainnya. (Harison, 2016)

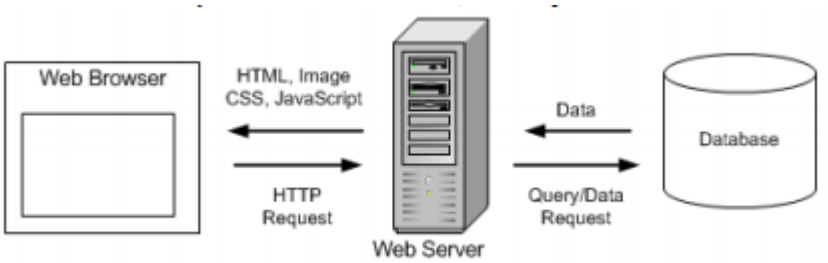
4. AJAX

AJAX diperkenalkan oleh Jesse James Garret dari Adaptive Path pada tahun 2005. Ia mendeskripsikan bagaimana mengembangkan web yang berbeda dengan metode tradisional. Ia mempublikasikan sebuah artikel yang berjudul "AJAX: *A New Approach to Web Applications*". Pada artikelnya, Garret yakin bahwa aplikasi *web* dapat menutup jurang pemisah antara web dan aplikasi desktop.

Aplikasi web yang bekerja dengan AJAX bekerja secara *asynchronously*, yang berarti mengirim dan menerima data dari *user* ke *server* tanpa perlu me-*load* kembali seluruh halaman, melainkan hanya melakukan penggantian pada bagian web yang hendak diubah. Penggunaan AJAX mulai popular ketika digunakan oleh Google pada tahun 2005. AJAX bukanlah bahasa pemrograman baru, tetapi merupakan teknik baru penggunaan standar yang telah ada. Dengan AJAX kita dapat menjadi lebih baik, cepat dan menambah unsur *user-friendly* dan interaktif pada aplikasi web kita. AJAX berbasiskan pada JavaScript dan *request* HTTP. AJAX berbasiskan standar terbuka seperti:

- JavaScript
- XML
- HTML/XHTML
- CSS

Dengan menggunakan JavaScript AJAX dapat mengirim dan menerima data antara web server dan *web browser*. Teknik yang dimiliki AJAX akan bergantian bertukar data dan mere-*load* ulang seluruh halaman. (Sunyoto, 2010)



Gambar 1. Arsitektur Model AJAX

(Sumber: Sunyoto, 2010)

Pada model ini, *server* mengirimkan respon berisi seluruh halaman termasuk *header, logo, navigasi, footer,* dll. Ketika mengklik *next* maka akan menampilkan halaman baru lagi (artinya, *header, logo, footer, navigasi* dikirim ulang) dan seterusnya akan mengirimkan data halaman baru lagi setiap diminta *request* dari *user*. Halaman seperti ini tidak masalah ketika data yang ditampilkan tidak memerlukan *response* yang cepat. Namun, akan menjadi masalah jika *user* menginginkan respon yang cepat, misalnya ketika dipilih *dropdown* tertentu maka data yang ditampilkan berubah menurut nilai dari *drop-down*. (Sunyoto, 2010)

2.1.2. *Framework*

Framework adalah sebuah kerangka program yang digunakan untuk membantu *developer* untuk mengembangkan kode secara konsisten. Dengan adanya *framework developer* bisa mengurangi jumlah *bug* pada aplikasi yang dibuat. Karena, fungsi dan variabel yang sudah tersedia di dalam komponen *framework*. Berikut merupakan contoh dari *framework* :

1) Laravel

Laravel adalah *web framework* PHP yang bersifat *open source* yang dibuat oleh Taylor Otwell dan ditujukan untuk pengembangan aplikasi web mengikuti *model-view-controller (MVC*) atau pola arsitektur. Beberapa fitur dari Laravel adalah pengembangan sistem modul-modul yang dapat dimanajemen, mengenalkan cara yang berbeda untuk mengakses *database* relasional, utilitas yang membantu dalam penyebaran aplikasi dan pemeliharaan yang mudah.

Pada laravel terdapat banyak fungsi yang digunakan untuk berinteraksi dengan database seperti mengambil semua baris, mengambil baris dengan kata kunci utama, menggunakan klausul untuk menyaring baris, melakukan sisipan, melakukan *update*, dan lain-lain. Sehingga kebutuhan dalam membangun suatu sistem berbasis web dapat dilakukan dengan mudah. Awal maret 2015, Laravel telah menjadi salah satu kerangka kerja PHP paling populer, bersama-sama dengan Symfony2, Nette, CodeIgniter dan Yii2. (Firma Sahrul, 2016)

2) Bootstrap

Menurut Spurlock dalam (Pahlevi, Mulyani, & Khoir, 2018), menyatakan bahwa "Bootstrap adalah sebuah framework untuk CSS dan berupa produk *open source* yang dibuat oleh Mark Otto dan Jacob Thornton". Pada awalnya bootstrap ini dibuat untuk membuat standarisasi *front end* untuk semua programmer di perusahaannya. Bootstrap telah berubah dari yang sebelumnya adalah CSS-Driven proyek ke sebuah *host* dari javascript *plugins* dan ikon yang dapat dengan mudah digunakan untuk formulir dan tombol"..

Bootstrap merupakan paket aplikasi siap pakai untuk membuat *front-end* sebuah *website*. Bisa dikatakan, bootstrap adalah template desain web dengan fitur plus. Bootstrap diciptakan untuk mempermudah proses desain web bagi berbagai tingkat pengguna, mulai dari level pemula hingga yang sudah berpengalaman. Cukup bermodalkan pengetahuan dasar mengenai HTML dan CSS, maka akan mudah dalam penggunaannya. (Rozi, 2015)

2.1.3. Basis data

Menurut Rosa dan Shalahuddin (2013:43) "Sistem basis data adalah sistem terkomputerisasi yang tujuan utamanya adalah memelihara data yang sudah diolah

atau informasi dan membuat informasi tersedia saat dibutuhkan". Selain itu, menurut Madcoms (2011:12) "*Database* atau sering juga disebut basis data adalah sekumpulan informasi yang disimpan dalam komputer secara sistematika dan merupakan sumber informasi yang dapat diperiksa menggunakan suatu program komputer".

Dari penjelasan di atas dapat disimpulkan bahwa *database* atau basis data ialah data yang disimpan dalam komputer yang telah diolah suatu program, dimana dengan program tersebut pengguna tidak akan sulit mencari data yang dibutuhkan.. Berikut merupakan contoh dari basis data :

1) MySQL

MySQL adalah sebuah implementasi dari sistem manajemen basisdata relasional (RDBMS) yang didistribusikan secara gratis dibawah lisensi GPL (General Public License). Setiap pengguna dapat secara bebas menggunakan MySQL, namun dengan batasan perangkat lunak tersebut tidak boleh dijadikan produk turunan yang bersifat komersial. MySQL sebenarnya merupakan turunan salah satu konsep utama dalam basisdata yang telah ada sebelumnya yaitu SQL (*Structured Query Language*). SQL adalah sebuah konsep pengoperasian basisdata, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis. Mysql berfungsi sebagai SQL yang dimiliki sendiri dan sudah diperluas oleh Mysql umumnya digunakan bersamaan dengan aplikasi yang membutuhkan data untuk membuat aplikasi server yang dinamis dan *powerfull*.

2) XAMPP

Menurut Wicaksono (2008:7) menjelaskan bahwa "XAMPP adalah sebuah *software* yang berfungsi untuk menjalankan *website* berbasis PHP dan menggunakan pengolah data MYSQL di komputer lokal". XAMPP berperan sebagai server web pada komputer lokal. XAMPP juga dapat disebut sebuah Cpanel server *virtual*, yang dapat membantu melakukan *preview* sehingga dapat dimodifikasi *website* tanpa harus online atau terakses dengan internet.

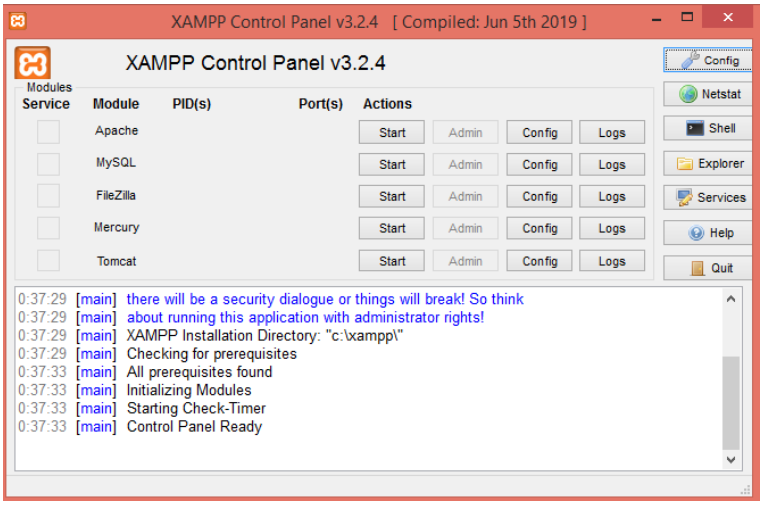
Sebagai informasi kata XAMPP merupakan singkatan dari:

X: berarti program ini dapat dijalankan diberbagai platform, misalnya Windows, Linux, mac OS, dan Solaris.

A: Apache, merupakan aplikasi *web* server, dan bertugas untuk menghasilkan halaman web yang benar kepada user berdasarkan kode PHP yang dituliskan oleh pembuat halaman web. Jika diperlukan juga berdasarkan kode PHP yang dituliskan, maka dapat saja suatu database diakses terlebih dahulu (misalnya dalam MySQL) untuk mendukung halaman web yang dihasilkan.

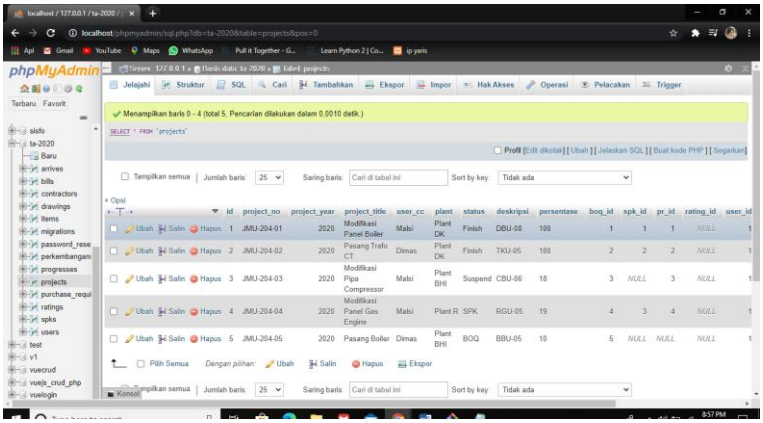
M: MySQL, merupakan aplikasi *database* server. Pengembangnya disebut *Structured Query Language* (SQL). SQL merupakan bahasa terstruktur yang digunakan untuk mengolah database beserta isinya. Pengguna dapat memanfaatkan MySQL untuk menambahkan, mengubah dan menghapus data yang berada dalam *database*.

P: PHP, bahasa pemrograman lainnya yang serupa, dan lain sebagainya.



Gambar 2. Panel Kontrol XAMPP

(Sumber: Data hasil kajian, 2020)



Gambar 3. Database phpMyadmin XAMPP

(Sumber: Data hasil kajian, 2020)

2.2. Teori Dasar Sistem

1) *Bill of Quantity* (BOQ)

Bill of Quantity merupakan suatu daftar yang berisi deskripsi, unit, jumlah dan harga material.

2) Desain

Desain adalah sebuah rencana atau gambar yang dibuat untuk menunjukan tampilan dan fungsi atau cara kerja suatu rencana sebelum dibuat.

3) *Purcase Requisition* (PR)

Puchase Rquisition atau permintaan pembelian merupakan formulir pada internal perusahaan, yang berfungsi untuk mencatat permintaan pembelian. Pencatatan aktifitas ini belum mempengaruhi posisi keuangan.

4) Surat Perintah Kerja (`SPK)

Surat perintah kerja adalah surat yang digunakan oleh suatu perusahaan dalam mengerjakan suatu proyek.

5) Kurva S

Kurva ini menunjukan hubungan antara presentase pekerjaan yang harus diselesaikan dengan waktu. Biasanya grafik ini dikenal dengan sebutan Kurva S (S-Curve) dalam satuan bobot persen. Dan terdapat dua macam bobot persen, yaitu: (Sutrisna, 2018)

1. Bobot pesen yang menyatakan perbandingan antara harga suatu jenis pekerjaan dalam waktu tertentu terhadap harga total yang tercantum dalam dokumen kontrak. Dalam hal ini grafik bobot persen menyatakan hubungan antara harga kumulatif bobot persen dengan waktu.
2. Bobot persen yang menyatakan perbandingan antara bobot suatu jenis pekerjaan dengan bobot seluruh pekerjaan. Dari bobot persen ini, dapat dibuat grafik yang menyatakan hubungan antara persentase kumulatif pekerjaan dengan waktu, dari grafik ini pula dapat diketahui persentase pekerjaan yang harus diselesaikan dalam jangka waktu tertentu.

Bobot persen yang dipakai pada proyek ini adalah sebagai berikut :

$$\text{Bobot pekerjaan} = \frac{Biaya\ Tiap\ Pekerjaan}{Biaya\ Total}\ x\ 100\%$$

Kurva S ini berfungsi untuk mengontrol pelaksanaan pekerjaan pada setiap waktu, dengan membandingkan bobot persen rencana dengan bobot persen realisasi dilapangan, sehingga perubahan yang terjadi dalam pelaksanaan tidak mengganggu atau mempengaruhi waktu pekerjaan secara keseluruhan. (Sutrisna, 2018)

Gambar 4. Kurva S

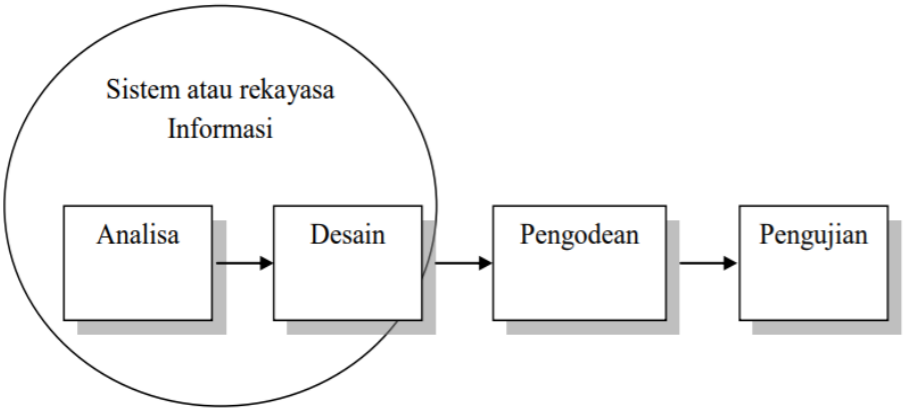
(Sumber : Data hasil kajian, 2020)

2.3. Metode Pengembangan.

2.3.1. Metode *Waterfall*

Metode *Waterfall* ialah yang menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengodean, pengujian,

dan tahap pendukung (*Support*) menurut Rosa dan Shalahuddin (2013:28). Adapun gambar model air terjun (*waterfall*) yaitu :



Gambar 5. Model Waterfall

(Sumber: Shalahuddin, 2013)

Model *Waterfall* dibagi menjadi beberapa bagaian menurut (Rosa dan shalahuddin, 2013:29) yaitu:

1. Analisis

Kebutuhan Perangkat Lunak Proses pengumpulan kebutuhan dilakukan secara intensif untuk mespesifikasikan kebutuhan perangkat lunak agar dapat dipahamai perangkat lunak seperti apa yang dibutuhkan oleh *user*. Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu untuk didokumentasikan.

2. Desain

Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program perangka lunak termaksud struktur data, arsitektur perangkat lunak, representasi antarmuka, dan proses pengodean. Tahap ini menstranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya. Desain perangkat lunak yang dihasilkan pada tahap ini juga perlu didokumentasikan.

3. Pembuatan Kode

Program Desain harus di translasikan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

4. Pengujian

Pengujian fokus pada perangkat lunak secara dari segi lojik dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

5. Pendukung atau pemeliharaan

Tidak menutup kemungkinan sebuah perangka lunak mengalami perubahan ketika sudah dikirimkan ke *user*. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru. Tahap pendukung atau pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan perangkat lunak yang sudah ada, tapi tidak untuk membuat perangkat lunak baru.

2.3.2. *Unified Modeling Language* (UML)

Unified Modeling Language (UML) merupakan sistem arsitektur yang bekerja dalam OOAD (*Object-Oriented Analysis/Design*) dengan satu bahasa yang konsisten untuk menentukan, visualisasi, mengkontruksi dan mendokumentasikan *artifact* (sepotong informasi yang digunakan atau dihasilkan dalam suatu proses rekayasa *software*, dapat berupa model, deksripsi atau *software*) yang terdapat dalam sistem *software*.

*Unified Modeling Langua*ge (UML) adalah bahasa standar untuk menulis denah perangkat lunak. UML dapat digunakan untuk memvisualisasikan, menentukan, membangun, dan mendokumentasikan artefak dari sistem perangkat lunak. Dengan kata lain, seperti arsitek bangunan membuat denah yang akan digunakan oleh sebuah perusahaan konstruksi, arsitek software membuat diagram UML untuk membantu pengembang perangkat lunak membangun perangkat lunak (Pressman, 2010:841). Diagram UML antara lain sebagai berikut:

1. *Use Case Diagram*

Use case diagram merupakan permodelan untuk menggambarkan kelakuan sistem yang akan dibuat. *Use case diagram* digunakan untuk mengetahui fungsi apa saja yang ada didalam sebuah sistem dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut.

2. *Entity Relationship Diagram* (ERD)

ERD adalah model teknik pendekatan yang menyatakan atau menggambarkan hubungan suatu model. Didalam hubungan ini tersebut dinyatakan yang utama dari ERD adalah menunjukan objek data (Entity) dan hubungan (Relationship), yang ada pada Entity berikutnya. (Fridayanthie, 2016)

Proses memungkinkan analis menghasilkan struktur basis data dapat disimpan dan diambil secara efisien. Simbol-simbol dalam ERD adalah sebagai berikut:

a. Entitas: suatu yang nyata atau abstrak yang mempunyai karakteristik dimana kita akan menyimpan data.
b. Atribut: ciri umum semua atau sebagian besar instansi pada entitas tertentu.
c. Relasi: hubungan alamiah yang terjadi antara satu atau lebih entitas.
d. *Link*: garis penghubung atribut dengan kumpulan entitas dan kumpulan entitas dengan relasi

3. Diagram Aktivitas (*Activity Diagram*)

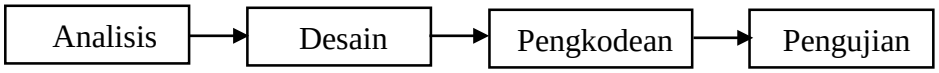
Activity diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses.

4. Squence Diagram

Sequence diagram merupakan diagram yang menjelaskan alur proses dari setiap *use case* yang sudah dibuat.

BAB 3 – KONSEP PERANCANGAN PROGRAM

Perancangan atau pengembangan suatu produk memerlukan suatu model atau metode dalam proses pengembangannya, sedangkan dalam tahapan pengembangan perangkat lunak sering disebut dengan *Software Development Life Cycle Model*. Metode yang digunakan dalam perancangan sistem informasi proyek berbasis web ini menggunakan metode *waterfall*. Metode ini memiliki model pengembangan sebagai berikut:



Gambar 6. Metode

(Sumber: Data hasil kajian, 2020)

Sesuai dengan model *waterfall*, pengembangan sistem informasi proyek berbasis web dimulai dengan melakukan analisis kebutuhan. Analisis kebutuhan mencakup informasi sistem berjalan yang saat ini berlangsung, apakah masih terdapat beberapa masalah yang terjadi. Kemudian dari hasil analisa tersebut mendapatkan solusi dengan merancang sistem baru. Tahap berikutnya adalah tahap desain, analisis kebutuhan yang telah selesai kemudian dibuatlah desain perencanaan *database* dan *Unified Modeling Language* (UML) meliputi *use case diagram*, *class diagram*, *activity diagram*, dan *sequence diagram.* Tahap implementasi dilakukan dengan menulis kode program sesuai dengan desain yang telah dirancang. Kemudian pada tahap selanjutnya melakukan uji pada sistem yang telah di buat.

3.1. Perencanaan *Database*

Tabel-tabel yang digunakan pada perancangan sistem informasi proyek ini adalah sebagai berikut:

1. *User*

Tabel *user* berfungsi untuk menyimpan data pengguna sistem. Tabel *user* memiliki tujuh kolom antara lain:

Tabel 1. User

No	Kolom	Tipe Data	Lebar
1	id	INT	11
2	departement	VARCHAR	20
3	kode	VARCHAR	5
4	email	VARCHAR	20
5	password	VARCHAR	20
6	created_at	TIMESTAMP	
7	up*date*d_at	TIMESTAMP	

(Sumber: Data hasil kajian, 2020)

a. Kolom *id*

Kolom *id* merupakan kolom yang bersifat unik yang digunakan sebagai *primary key* untuk membedakan tiap-tiap *record* pada basis data. Kolom ini memiliki tipe data *integer* dengan lebar data sebelas.

b. Kolom departemen

Kolom departemen merupakan kolom yang digunakan sebagai nama departemen sekaligus nama pengguna untuk masuk ke dalam sistem. Kolom ini memiliki tipe data *varchar* dengan lebar data dua puluh.

c. Kolom kode

Kolom kode merupakan kolom yang digunakan untuk kode departemen. Kolom ini memiliki tipe data varchar dengan lebar data lima.

d. Kolom email

Kolom email merupakan kolom yang digunakan untuk email departemen. Kolom ini memiliki tipe data *varchar* dengan lebar data dua puluh.

e. Kolom *password*

Kolom *password* merupakan kolom yang digunakan untuk *password* akun untuk masuk ke dalam sistem. Kolom ini memiliki tipe data *varchar* dengan lebar data dua puluh.

f. Kolom *created_at*

Kolom *created_at* merupakan kolom yang digunakan untuk menyimpan data tanggal dan waktu akun terbuat. Kolom ini memiliki tipe data timestamp dengan panjang data bawaan.

g. Kolom *updated_at*

Kolom *updated*_at merupakan kolom yang digunakan untuk menyimpan data tanggal dan waktu terjadinya perubahan data pada *record*.

2. *Project*

Tabel *project* berfungsi untuk menyimpan data detail seluruh proyek pada sistem. Tabel *project* memiliki empat belas kolom antara lain:

Tabel 2. *Project*

No	Kolom	Tipe Data	Lebar
1	id	INT	11
2	project_year	INT	4
3	project_no	VARCHAR	11
4	project_title	VARCHAR	100
5	user_cc	VARCHAR	30
6	plant	VARCHAR	20
7	status	VARCHAR	20
8	deskripsi	VARCHAR	255
9	persentase	VARCHAR	3
10	boq_id	INT	11
11	pr_id	INT	11
12	spk_id	INT	11
13	user_id	INT	11
14	rating_id	INT	11

(Sumber: Data hasil kajian, 2020)

a. Kolom *id*

Kolom *id* merupakan kolom yang bersifat unik yang digunakan sebagai *primary key* untuk membedakan tiap-tiap *record* pada basis data. Kolom ini memiliki tipe data *integer* dengan lebar data sebelas.

b. Kolom *project_year*

Kolom *project_year* merupakan kolom yang digunakan untuk menyimpan data tahun proyek. Kolom ini memiliki tipe data *integer* dengan lebar data empat.

c. Kolom *project_no*

Kolom *project_no* merupakan kolom yang digunakan untuk menyimpan data nomor proyek. Kolom ini memiliki tipe data *varchar* dengan lebar data sebelas.

d. Kolom *project_title*

Kolom *project_title* merupakan kolom yang digunakan untuk menyimpan data judul proyek. Kolom ini memiliki tipe data *varchar* dengan lebar data seratus.

e. Kolom *user_cc*

Kolom *user_cc* merupakan kolom yang digunakan untuk menyimpan data *user call center*. Kolom ini meiliki tipe data *varchar* dengan lebar data tiga puluh.

f. Kolom *plant*

Kolom *plant* merupakan kolom yang digunakan untuk meyimpan data *plant* tempat proyek dilaksanakan. Kolom ini memiliki tipe data *varchar* dengan lebar data dua puluh.

g. Kolom status

Kolom status merupakan kolom yang diguankan untuk menyimpan data status proyek. Kolom ini memiliki lebar data dua puluh.

h. Kolom deskripsi

Kolom deskripsi merupakan kolom yang digunakan untuk menyimpan data detail-detail proyek yang tidak bisa ditulis di judul proyek. Kolom ini memiliki tipe data *varchar* dengan lebar data 255.

i. Kolom persentase

Kolom persentase merupakan kolom yang digunakan untuk menyimpan persentase proyek. Kolom ini memiliki tipe data *varchar* dengan lebar data tiga.

j. Kolom *boq_id*

Kolom *boq_id* merupakan kolom yang digunakan untuk menyimpan *id* dari *bill of quantity*. Kolom ini memiliki tipe data *integer* dengan lebar data sebelas.

k. Kolom *pr_id*

Kolom *pr_id* merupakan kolom yang digunakan untuk menyimpan *id* dari *purchase requisition*. Kolom ini memiliki tipe data *integer* dengan lebar data sebelas.

l. Kolom spk_*id*

Kolom spk_*id* merupakan kolom yang digunakan untuk menyimpan *id* dari surat perintah kerja. Kolom ini memiliki tipe data *integer* dengan lebar data sebelas.

m. Kolom *user_id*

Kolom *user_id* merupakan kolom yang digunakan untuk menyimpan *id* dari pengguna. Kolom ini memiliki tipe data *integer* dengan lebar data sebelas.

n. Kolom *rating_id*

Kolom *rating_id* merupakan kolom yang digunakan untuk menyimpan *id* dari hasil penilaian kinerja kontraktor. Kolom ini memiliki tipe data *integer* dengan lebar data sebelas.

3. *Bill of Quantity*

Tabel *Bill of Quantity* berfungsi untuk menyimpan data biaya rencana dan biaya aktual proyek. Tabel ini memiliki tiga kolom antara lain

Tabel 3. *Bill of Quantity*

No	Kolom	Tipe Data	Lebar
1	id	INT	11
2	planned_budged	VARCHAR	20
3	actual_budget	VARCHAR	20

(Sumber: Data hasil kajian, 2020)

a. Kolom i*d*

Kolom id merupakan kolom yang bersifat unik yang digunakan sebagai *primary key* untuk membedakan tiap-tiap *record* pada basis data. Kolom ini memiliki tipe data *integer* dengan lebar data sebelas,

b. Kolom *planned_budged*

Kolom *planned_budged* merupakan kolom yang digunakan untuk menyimpan data rencana biaya dari proyek berdasarkan *bill of quantity.* Kolom ini memiliki tipe data *varchar* dengan lebar data dua puluh.

c. Kolom *actual_budged*

Kolom *actual_budged* merupakan kolom yang diguankan untuk menyimpan data biaya aktual proyek. Kolom ini memiliki tipe data *varchar* dengan lebar data dua puluh.

4. *Item*

Tabel *item* berfungsi untuk menyimpan data item *Bill of Quantity* berupa material atau jasa pada sistem. Tabel *item* memiliki sebelas kolom antara lain:

Tabel 4. *Item*

No	Kolom	Tipe Data	Lebar
1	id	INT	11
2	boq_id	INT	11
3	item_name	VARCHAR	50
4	tipe	VARCHAR	10
5	spesification	VARCHAR	255
6	quantity	INT	3
7	price_unit	VARCHAR	20
8	unit	VARCHAR	10
9	total_price	VARCHAR	20
10	bobot	VARCHAR	20
11	persentase	VARCHAR	3

(Sumber: Data hasil kajian, 2020)

a. Kolom *id*

Kolom *id* merupakan kolom yang bersifat unik yang digunakan sebagai *primary key* untuk membedakan tiap-tiap record pada basis data. Kolom ini memiliki tipe data *integer* dengan lebar data sebelas.

b. Kolom *boq_id*

Kolom *boq_id* merupakan kolom yang digunakan untuk menyimpan data *id* dari *bill of quantity* sebagai penanda bahwa *item* tertentu menjadi bagian dari *bill of quantity* tertentu. Kolom ini memiliki tipe data *integer* dengan lebar data sebelas.

c. Kolom *item_name*

Kolom *item_name* merupakan kolom yang digunakan untuk menyimpan data nama *item*. Kolom ini memiliki tipe data *varchar* dengan lebar data lima puluh.

d. Kolom tipe

Kolom tipe merupakan kolom yang digunakan untuk menyimpan data tipe *item* yang digunakan. Kolom ini memiliki tipe data *varchar* dengan lebar data 10.

e. Kolom *specification*

Kolom *specification* merupakan kolom yang digunakan untuk menyimpan data spesifikasi *item* yang diguankan. Kolom ini memiliki tipe data *varchar* dengan lebar data 255.

f. Kolom *quantity*

Kolom *quantity* merupakan kolom yang digunakan untuk menyimpan data jumlah *item* yang diguanakan. Kolom ini memiliki tipe data *integer* dengan lebar data tiga.

g. Kolom *price*_unit

Kolom *price*_unit merupakan kolom yang digunakan untuk menyimpann data harga *item* per unit. Kolom ini memiliki tiep data *varchar* dengan lebar data dua puluh.

h. Kolom unit

Kolom unit merupakan kolom yang digunakan untuk menyimpan data satuan *item* yang digunakan. Kolom ini memilikit tipe data *varchar* dengan lebar data sepuluh.

i. Kolom *total_price*

Kolom *total_price* merupakan kolom yang digunakan untuk menyimpan data total harga yang berasal dari harga per unit dikali jumlah unit. Kolom ini memiliki tipe data *varchar* dengan lebar data dua puluh.

j. Kolom bobot

Kolom bobot merupakan kolom yang digunakan untuk menyimpan data bobot *item* dibandingkan dengan total biaya proyek. Kolom ini memilikit tipe data *varchar* dengan lebar data dua puluh.

k. Kolom persentase

Kolom persentase merupakan kolom yang digunakan untuk menyimpan data persentase kedatangan *item.* Kolom ini memiliki tipe data varchar dengan lebar data tiga

5. *Drawing*

Tabel *drawing* berfungsi untuk menyimpan data desain gambar proyek. Tabel *drawing* memiliki empat kolom antara lain:

Tabel 5. *Drawing*

No	Kolom	Tipe Data	Lebar
1	id	INT	11
2	project_id	INT	11
3	name	VARCHAR	50
4	Path	VARCHAR	255

(Sumber: Data hasil kajian, 2020)

a. Kolom *id*

Kolom *id* merupakan kolom yang bersifat unik yang digunakan sebagai *primary key* untuk membedakan tiap-tiap *record* pada basis data. Kolom ini memiliki tipe data *integer* dengan lebar data sebelas.

b. Kolom *project_id*

Kolom merupakan kolom yang digunakan untuk menyimpan data *id* proyek. Kolom ini memiliki tipe data *integer* dengan lebar data sebelas.

c. Kolom *name*

Kolom *name* merupakan kolom yang digunakan untuk menyimpan data nama desain. Kolom ini memiliki tipe data *varchar* dengan lebar data lima puluh.

d. Kolom *path*

Kolom *path* merupakan kolom yang digunakan untuk menyimpan data direktori penyimpanan gambar. Kolom ini memiliki tipe data *varchar* dengan lebar data 255.

6. *Purchase Requisition*

Tabel purchase requisition berfungsi untuk menyimpan data daftar permintaan pembelian pada proyek yang akan dilakukan. Tabel purchase requisition memiliki lima kolom antara lain:

Tabel 6. *Purchase Requisition*

No	Kolom	Tipe Data	Lebar
1	id	INT	11
2	pr_no	VARCHAR	20
3	aanwijzing_*date*	DATE	
4	bid_subbmission_*date*	DATE	
5	path	VARCHAR	255

(Sumber: Data hasil kajian, 2020)

a. Kolom *id*

Kolom *id* merupakan kolom yang bersifat unik yang digunakan sebagai *primary key* untuk membedakan tiap-tiap *record* pada basis data. Kolom ini memiliki tipe data *integer* dengan lebar data sebelas.

b. Kolom *pr_no*

Kolom *pr_no* merupakan kolom yang digunakan untuk menyimpan data nomor *purchase requisition*. Kolom ini memiliki tipe data *varchar* dengan lebar data dua puluh.

c. Kolom *aanwijzing_date*

Kolom *aanwijzing_date* merupakan kolom yang digunakan untuk menyimpan data tanggal *aanwijzing*. Kolom ini memiliki tipe data *date*

d. Kolom *bid_submission_date*

Kolom *bid_submission_date* merupakan kolom yang digunakan untuk menyimpan data tanggal *bid submission*. Kolom ini memiliki tipe data *date*

e. Kolom *path*

Kolom *path* merupakan kolom yang digunakan untuk menyimpan data direktori penyimpanan gambar. Kolom ini memiliki tipe data *varchar* dengan lebar data 255.

7. SPK

Tabel spk berfungsi untuk menyimpan data surat perintah kerja pada proyek yang akan dilaksanakan. Tabel spk memiliki enam kolom antara lain:

Tabel 7. SPK

No	Kolom	Tipe Data	Lebar
1	id	INT	11
2	spk_no	VARCHAR	20
3	start_execution_*date*	DATE	
4	estimate_finish_*date*	DATE	
5	path	VARCHAR	255
6	contractor_id	INT	11

(Sumber: Data hasil kajian, 2020)

a. Kolom *id*

Kolom *id* merupakan kolom yang bersifat unik yang digunakan sebagai *primary key* untuk membedakan tiap-tiap *record* pada basis data. Kolom ini memiliki tipe data *integer* dengan lebar data sebelas.

b. Kolom spk_no

Kolom spk_no merupakan kolom yang digunakan untuk menyimpan data nomor surat perintah kerja. Kolom ini memiliki tipe data *varchar* dengan lebar data dua puluh.

c. Kolom *start_execution_date*

Kolom *start_execution_date* merupakan kolom yang digunakan untuk menyimpan tanggal mulainya pekerjaan proyek berdasarkan surat perintah kerja. Kolom ini memiliki tipe data *date*.

d. Kolom *estimate_finish_date*

Kolom *estimate_finish_date* merupakan kolom yang digunakan untuk menyimpan tanggal berakhirnya pekerjaan proyek berdasarkan surat perintah kerja. Kolom ini memiliki tipe data *date.*

e. Kolom *path*

Kolom *path* merupakan kolom yang digunakan untuk menyimpan data direktori penyimpanan gambar. Kolom ini memiliki tipe data *varchar* dengan lebar data 255.

f. Kolom *contractor_id*

Kolom *contractor_id* merupakan kolom yang digunakan untuk menyimpan data *id* kontraktor. Kolom ini memiliki tipe data *integer* dengan lebar data sebelas.

8. Kontraktor

Tabel kontaktor berfungsi untuk menyimpan data kontaktor yang bertanggung jawab pada proyek terkait. Tabel kontraktor memiliki empat kolom antara lain:

Tabel 8. *Contractor*

No	Kolom	Tipe Data	Lebar
1	id	INT	11
2	nama	VARCHAR	30
3	alamat	VARCHAR	255
4	rating	VARCHAR	10

(Sumber: Data hasil kajian, 2020)

a. Kolom *id*

Kolom *id* merupakan kolom yang bersifat unik yang digunakan sebagai *primary key* untuk membedakan tiap-tiap *record* pada basis data. Kolom ini memiliki tipe data *integer* dengan lebar data sebelas.

b. Kolom nama

Kolom nama merupakan kolom yang digunakan untuk menyimpan data nama kontraktor. Kolom ini memiliki tiep data *varchar* dengan lebar data tiga puluh.

c. Kolom alamat

Kolom alamat merupakan kolom yang digunakan untuk menyimpan data alamat kontraktor. Kolom ini memiliki tipe data *varchar* dengan lebar data 255.

d. Kolom *rating*

Kolom *rating* merupakan kolom yang digunakan untuk menyimpan data penilaian rata-rata kinerja kontraktor. Kolom ini memiliki tipe data *varchar* dengan lebar data sepuluh.

9. *Rating*

Tabel *rating* berfungsi untuk menyimpan data nilai kinerja kontraktor yang melaksanakan proyek tersebut. Tabel *rating* memiliki empat kolom antara lain:

Tabel 9. *Rating*

No	Kolom	Tipe Data	Lebar
1	id	INT	11
2	contractor_id	INT	11
3	rating	VARCHAR	10
4	deskripsi	VARCHAR	255

(Sumber: Data hasil kajian, 2020)

a. Kolom *id*

Kolom *id* merupakan kolom yang bersifat unik yang digunakan sebagai *primary key* untuk membedakan tiap-tiap *record* pada basis data. Kolom ini memiliki tipe data *integer* dengan lebar data sebelas.

b. Kolom *contractor_id*

Kolom *contractor_id* merupakan kolom yang digunakan untuk menyimpan data *id* kontraktor. Kolom ini memiliki tipe data *integer* dengan lebar data sebelas.

c. Kolom *rating*

Kolom *rating* merupakan kolom yang digunakan untuk menyimpan data penilaian kinerja kontraktor terhadap suatu proyek. Kolom ini memiliki tipe data *varchar* dengan lebar data sepuluh.

d. Kolom deskripsi

Kolom deskripsi merupakan kolom yang digunakan untuk menyimpan deskripsi atau keterangan-keterangan khusus terhadap proyek. Kolom ini memiliki tipe data *varchar* dengan lebar data 255.

10. *Arrive*

Tabel *arrive* berfungsi untuk menyimpan data kedatangan barang berdasarkan surat jalan. Tabel *arrive* memiliki empat kolom antara lain:

Tabel 10. *Arrive*

No	Kolom	Tipe Data	Lebar
1	Id	INT	11
2	boq_id	INT	11
3	*date*	*DATE*	
4	Path	*VARCHAR*	255

(Sumber: Data hasil kajian, 2020)

a. Kolom *id*

Kolom id merupakan kolom yang bersifat unik yang digunakan sebagai *primary key* untuk membedakan tiap-tiap *record* pada basis data. Kolom ini memiliki tipe data *integer* dengan lebar data sebelas.

b. Kolom *boq_id*

Kolom *boq_id* merupakan kolom yang digunakan untuk menyimpan data *id bill of quantity*. Kolom ini memiliki tipe data *integer* dengan lebar data sebelas.

c. Kolom *date*

Kolom *date* merupakan kolom yang digunakan untuk menyimpan data tanggal kedatangan barang. Kolom ini memiliki tipe data *date*.

d. Kolom deskripsi

Kolom deskripsi merupakan kolom yang digunakan untuk menyimpan deskripsi atau keterangan-keterangan khusus terhadap proyek. Kolom ini memiliki tipe data *varchar* dengan lebar data 255.

11. Progres

Tabel progres berfungsi untuk menyimpan data perkembangan kedatangan barang. Tabel progres memiliki tujuh kolom antara lain:

Tabel 11. Progres

No	Kolom	Tipe Data	Lebar
1	id	INT	11
2	boq_id	INT	11
3	arrive_id	INT	11
4	item_id	INT	11
5	quantity	*VARCHAR*	10
6	*date*	*DATE*	
7	bobot	*VARCHAR*	10

(Sumber: Data hasil kajian, 2020)

a. Kolom *id*

Kolom id merupakan kolom yang bersifat unik yang digunakan sebagai *primary key* untuk membedakan tiap-tiap record pada basis data. Kolom ini memiliki tipe data *integer* dengan lebar data sebelas.

b. Kolom *boq_id*

Kolom *boq_id* merupakan kolom yang digunakan untuk menyimpan data *id* bill of quantity. Kolom ini memiliki tipe data *integer* dengan lebar data sebelas

c. Kolom arrive_id

Kolom *arrive_id* merupakan kolom yang digunakan untuk menyimpan data *id* kedatangan barang. Kolom ini memiliki tipe data *integer* dengan lebar data sebelas.

d. Kolom *item_id*

Kolom *item_id* merupakan kolom yang digunakan untuk menyimpan data *id* dari *item* berdasarkan surat jalan. Kolom ini memiliki tipe data *integer* dengan lebar data sebelas.

e. Kolom *quantity*

Kolom *quantity* merupakan kolom yang digunakan untuk menyimpan data jumlah barang yang datang. Kolom ini memiliki tipe data *varchar* dengan lebar data sebelas.

f. Kolom *date*

Kolom *date* merupakan kolom yang digunakan untuk menyimpan data tanggal kedatangan barang. Kolom ini memiliki tipe data *date.*

g. Kolom bobot

Kolom bobot merupakan kolom yang digunakan untuk menyimpan data bobot barang yang datang berdasarkan surat jalan. Kolom in memiliki tipe data *varchar* dengan lebar data sepuluh.

12. Perkembangan

Tabel perkembangan berfungsi untuk menyimpan data perkembangan pemasangan barang. Tabel perkembangan memiliki delapan kolom antara lain:

Tabel 12. Perkembangan

No	Kolom	Tipe Data	Lebar
1	id	INT	11
2	boq_id	INT	11
3	barang	*VARCHAR*	20
4	pemasangan	*VARCHAR*	20
5	total	*VARCHAR*	20
6	path	*VARCHAR*	225
7	*date*	*DATE*	
8	status	*VARCHAR*	20

(Sumber: Data hasil kajian, 2020)

a. Kolom *id*

Kolom *id* merupakan kolom yang bersifat unik yang digunakan sebagai *primary key* untuk membedakan tiap-tiap *record* pada basis data. Kolom ini memiliki tipe data *integer* dengan lebar data sebelas.

b. Kolom *boq_id*

Kolom *boq_id* merupakan kolom yang digunakan untuk menyimpan data *id* dari *bill of quantity*. Kolom ini memiliki tipe data *integer* dengan lebar data sebelas.

c. Kolom barang

Kolom barang merupakan kolom yang digunakan untuk menyimpan data kedatangan barang. Kolom ini memiliki tipe data *varchar* dengan lebar data dua puluh.

d. Kolom pemasangan

Kolom pemasangan merupakan kolom yang digunakan untuk menyimpan data persentase pemasangan. Kolom ini memiliki tipe data *varchar* dengan lebar data dua puluh.

e. Kolom total

Kolom total merupakan kolom yang digunakan untuk menyimpan data persentase total dengan perbandingan delapan puluh persen dari kedatangan barang dan dua puluh persen dari persentase pemasangan. Kolom ini memiliki tipe data *varchar* dengan lebar data dua puluh.

f. Kolom *path*

Kolom *path* merupakan kolom yang digunakan untuk menyimpan data direktori penyimpanan gambar. Kolom ini memiliki tipe data *varchar* dengan lebar data 255.

g. Kolom *date*

Kolom *date* merupakan kolom yang digunakan untuk menyimpan data tanggal perkembangan pemasasngan diambil. Kolom ini memiliki tipe data *date*.

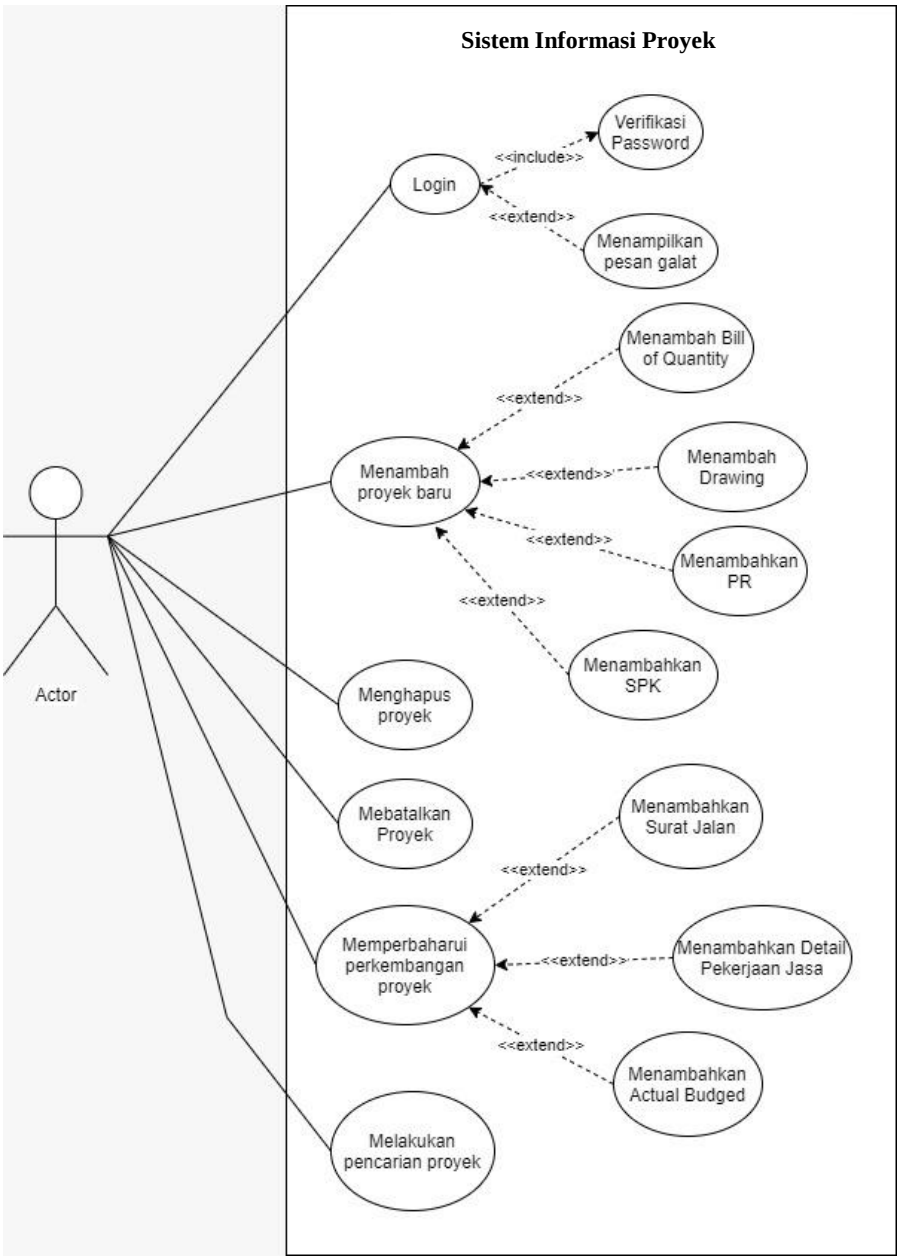
h. Kolom status

Kolom status merupakan kolom yang digunakan untuk menyimpan status perkembangan proyek. Kolom ini memiliki tipe data *varchar* dengan lebar data dua puluh.

3.2. *Unified Modeling Language* (UML)

3.2.1. Use Case Diagram

Use case diagram adalah deskripsi proses utama dari sebuah sistem dilihat dari sudut pandang pengguna.. *Use case diagram* dapat dilihat pada gambar berikut.



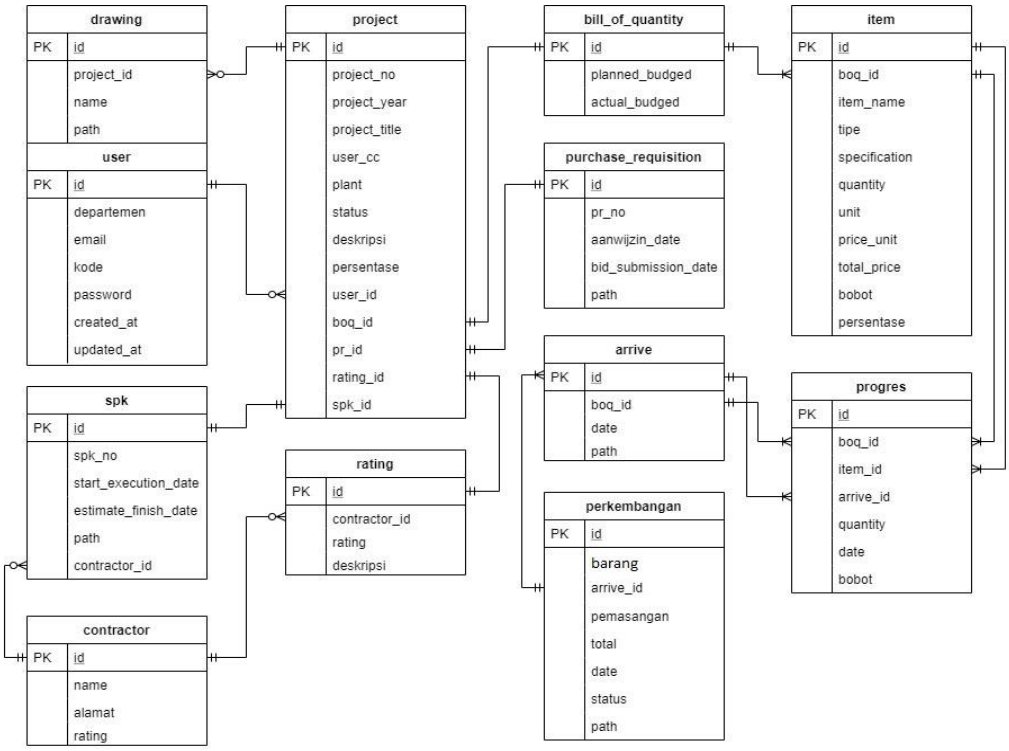
Gambar 7. *Use Case Diagram*

(Sumber: Data hasil kajian, 2020)

Berdasarkan diagram *use case* di atas terdapat *actor* yaitu *admin* dari sistem dan staf yang bekerja di lapangan. *Actor* dapat melakukan *login* sistem, menambah proyek baru, mengisi *bill of quantity,* menambahkan *purchase requisition,* menambahkan surat perintah kerja, menghapus proyek, membatalkan proyek, memperbaharui perkembangan proyek, menambahakan surat jalan, menambahkan detail pekerjaan, menambahkan *actual budged*, dan melakukan pencarian proyek.

3.2.2. *Entity Relational Diagram* (ERD)

Diagram ERD merupakan gambaran hubungan antar *entitas* atau tabel yang digunakan dalam sistem. Diagram ERD dapat dilihat pada gambar berikut.



Gambar 8. *Entity Relational Diagram*

(Sumber: Data hasil kajian, 2020)

Berdasarkan diagram ERD diatas terdapat dua belas entitias yang saling berhubungan yaitu:

a. Entitas *project*

Entitas *project* memiliki kolom *id* yang merupakan *primary key* yang berhubungan dengan kolom *project_id* pada entitas *drawing* dengan hubungan *one to many*. Kolom *user_id* merupakan *foreign key* yang berhubungan dengan kolom *id* pada entitas *user* dengan hubungan *many to one*. Kolom *boq_id* merupakan *foreign key* yang berhubungan dengan kolom *id* pada entitas *bill_of_quantity* dengan hubungan *one to one*. Kolom *pr_id* merupakan *foreign key* yang berhubungan dengan kolom *id* pada entitas pr dengan hubungan *one to one*. Kolom *spk_id* merupakan *foreign key* yang berhubungan dengan kolom *id* pada spk dengan hubungan *one to one*. Kolom *rating_id* merupakan *foreign key* yang berhubungan dengan kolom *id* pada entitas rating dengan hubungan *one to one*.

b. Entitas *bill_of_quantity*

Entitas *bill_of_quantity* memiliki kolom *id* yang merupakan *primary key* yang berhubungan dengan kolom *boq_id* pada entitas *item* dengan hubungan *one to many*. Selain itu kolom *id* juga berhubungan dengan kolom *boq_id* pada entitas *project* dengan hubungan *one to one*.

c. Entitas *purchase_requisition*

Entitaas *purchase_requisition* memiliki kolom *id* yang merupakan *primary key* yang berhubungan dengan kolom *pr_id* pada entitas *project* dengan hubungan *one to one*.

d. Entitas spk

Entitaas spk memiliki kolom *id* yang merupakan *primary key* yang berhubungan dengan kolom *spk_id* pada entitas *project* dengan hubungan *one to one*. Kolom *contractor_id* merupakan *foreign key* yang berhubungan dengan kolom *id* pada entitas *contractor* dengan hubungan *many to one*.

e. Entitas rating

Entitaas *rating* memiliki kolom *id* yang merupakan *primary key* yang berhubungan dengan kolom *rating_id* pada entitas project dengan hubungan *one to one*. Kolom *contractor_id* merupakan *foreign key* yang berhubungan dengan kolom *id* pada entitas *contractor* dengan hubungan *many to one*.

f. Entitas *user*

Entitaas *user* memiliki kolom *id* yang merupakan *primary key* yang berhubungan dengan kolom *user_id* pada entitas *project* dengan hubungan *one to many*.

g. Entitas drawing

Entitas *drawing* memiliki kolom *project_id* yang merupakan *foreign key* yang berhubungan dengan kolom *id* pada entitas *project* dengan hubungan *many to one*.

h. Entitas *contractor*

Entitaas *contractor* memiliki kolom *id* yang merupakan *primary key* yang berhubungan dengan kolom *contractor_id* pada entitas *rating* dengan hubungan *one to many* dan juga berhubungan dengan kolom *contractor_id* pada entitas spk dengan hubungan *one to many*.

i. Entitas item

Entitas *item* memiliki kolom *id* yang merupakan *primary key* yang berhubungan dengan kolom *item_id* pada entitas progres dengan hubungan *one to one*. Kolom *boq_id* merupakan *foreign key* yang berhubungan dengan kolom id pada entitas *bill_of_quantity* dengan hubungan *many to one* dan juga berhubungan dengan kolom *boq_id* pada entitas progres dengan hubungan *many to one*.

j. Entitas *arrive*

Entitas *arrive* memiliki kolom *id* yang merupakan *primary key* yang berhubungan dengan kolom *arrive_id* pada entitas progres dengan hubungan *one to many*. Kolom *boq_id* merupakan *foreign key* yang berhubungan dengan kolom *boq_id* pada entitas progres dengan hubungan *one to many* dan juga berhubungan dengan kolom *boq_id* pada entitas perkembangan dengan hubungan *many to one*.

k. Entitas perkembangan

Entitas perkembangan memiliki kolom *arrive_id* yang merupakan *foreign key* yang berhubungan dengan kolom *id* pada entitas *arrive* dengan hubungan *one to many*.

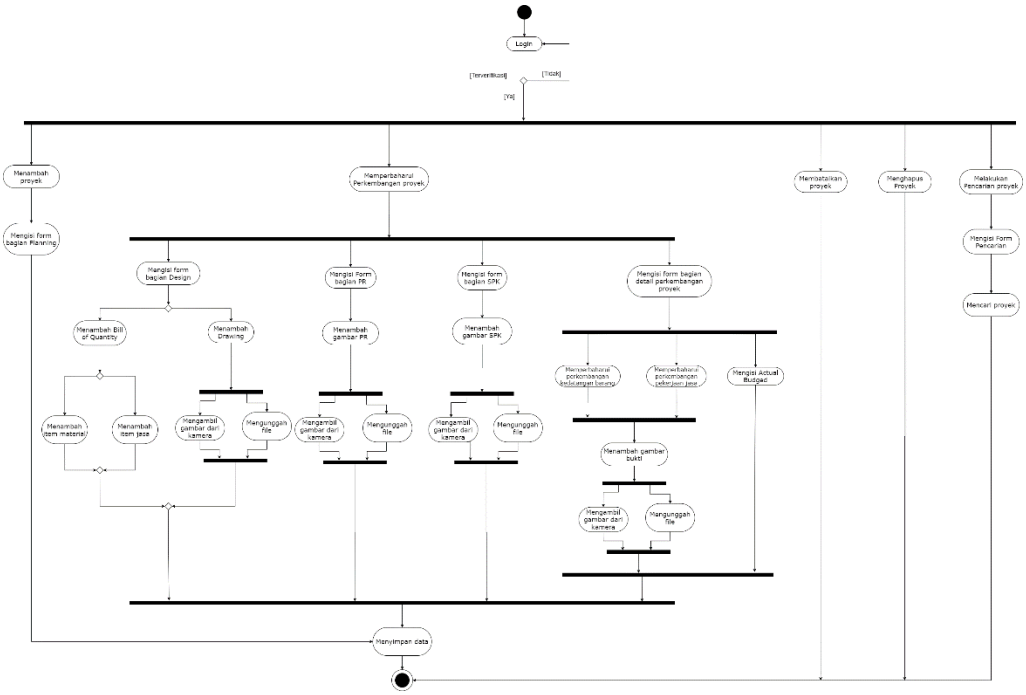
l. Entitas progres

Entitas progres memilik kolom *arrive_id* yang merupakan *foreign key* yang berhubungan dengan kolom *id* pada entitas *arrive* dengan hubungan *many to one*. Kolom *item_id* merupakan *foreign key* yang berhubungan dengan kolom i*d* pada entitas item dengan hubungan *many to one*. Kolom *boq_id* merupakan *foreign key*

yang berhubungan dengan kolom *boq_id* pada entitas *arrive* dengan hubungan *many to one*.

3.2.3. Activity Diagram

Diagram *Activity* merupakan gambar aktivitas user terhadap semua menu yang dibuat dalam sistem. Diagram *activity* dapat dilihat pada gambar berikut.



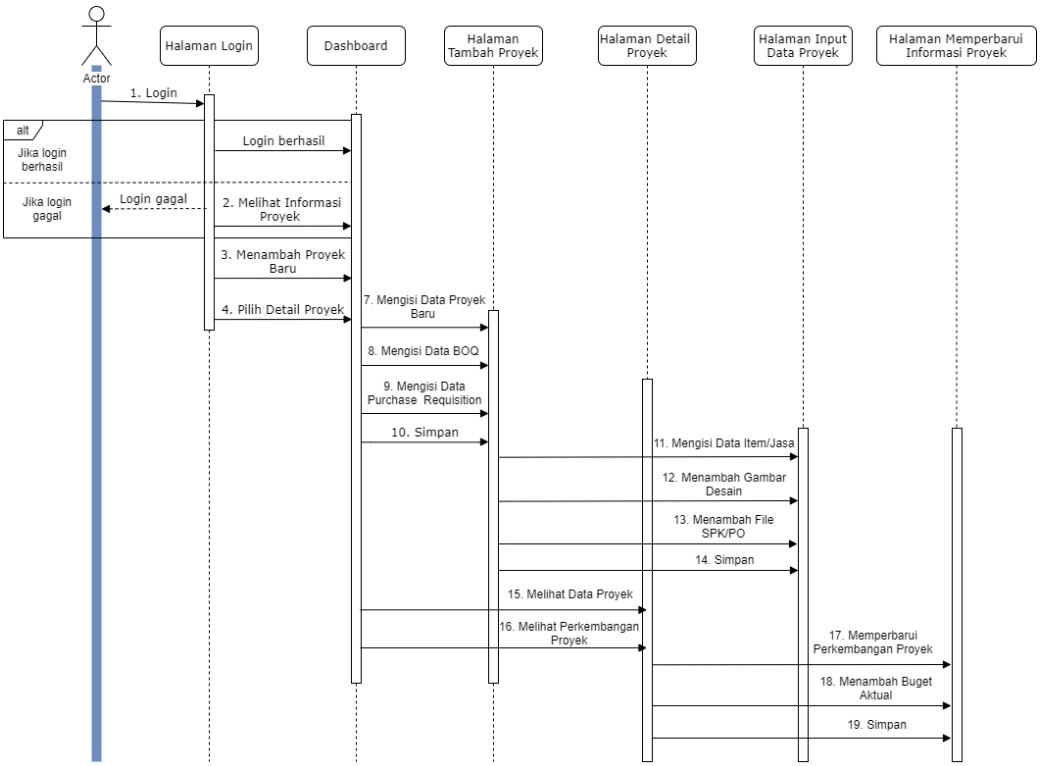
Gambar 9. *Activity Diagram*

(Sumber : Data hasil kajian, 2020)

Berdasarkan diagram *activity* di atas *user* harus *login* terlebih dahulu kemudian *user* dapat melakukan aktivitas menambah proyek, memperbaharui perkembangan proyek, membatalkan proyek, menghapus proyek, dan melakukan pencarian proyek.

3.2.4. Squence Diagram

Diagram *Sequence* merupakan gambar kegiatan *user* berdasarkan rangkaian urutan waktu penggunaan sistem. Diagram *sequence* dapat dilihat pada gambar berikut.



Gambar 10. *Sequence Diagram*

(Sumber: Data hasil kajian, 2020)

Berdasarkan diagram *activity* di atas kita dapat mengetahui urutan kegiatan *user* sesuai dengan urutan kegiatan dari kiri ke kanan.

3.2.5. Desain *Backlog*

Halaman *User* :

1. *User* dapat melakukan *login*
2. *User* dapat melihat informasi proyek
3. *User* dapat menambah proyek baru
4. *User* dapat mengisi data *form planning*
5. *User* dapat mengisi data BOQ (*Bill Of Quantity*)
6. *User* dapat mengupload gambar/file desain proyek
7. *User* dapat mengisi data *Purchase Requisition*
8. *User* dapat mengupload gambar/*file Purchase Requisition.*
9. *User* dapat mengisi data Surat Perintah Kerja
10. *User* dapat mengupload gambar/*file* Surat Perintah Kerja

11. *User* dapat memperbarui perkembangan kedatangan barang
12. *User* dapat mengupload gambar/*file* surat kedatangan barang
13. *User* dapat memperbarui perkembangan pekerjaan jasa
14. *User* dapat mengupload gambar/*file* perkembangan pekerjaan jasa
15. *User* dapat mengisi data informasi *actual budged.*
16. *User* dapat melakukan pembatalan proyek
17. *User* dapat melihat daftar Riwayat Proyek yang telah dikerjakan.
18. *User* dapat melihat perkembangan proyek berdasarkan grafik kurva s.
19. *User* dapat melakukan penilaian terhadap hasil kinerja kontraktor melalui *smartphone* atau melalui *Website*.

.

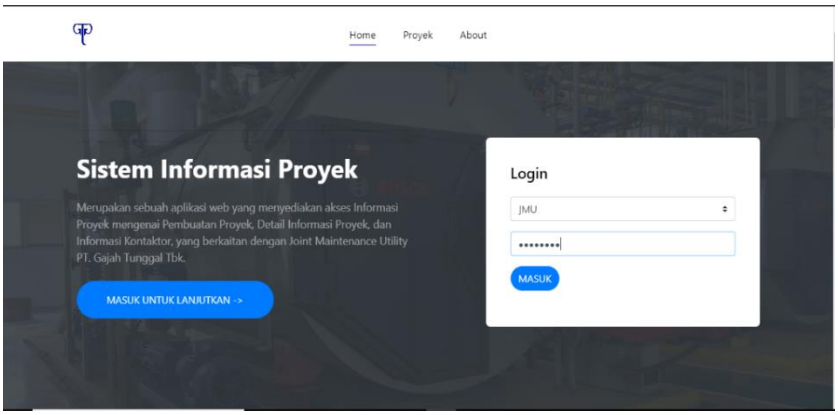
BAB 4 - HASIL KAJIAN DAN PEMBAHASAN

Bab ini membahas tentang hasil yang didapat setelah melakukan penelitian beserta pembahasan berdasarkan kajian yang dilakukan. Untuk lebih jelasnya akan dijelaskan pada sub bab berikut :

4.1 Tampilan Web

4.1.1 Login

Halaman *login* berfungsi sebagai tempat masuk bagi pengguna yang ingin mengakses halaman utama pada sistem informasi proyek. Pengguna yang ingin melakukan login harus memasukan *username* dan *password* yang sudah terdaftar di basis data.

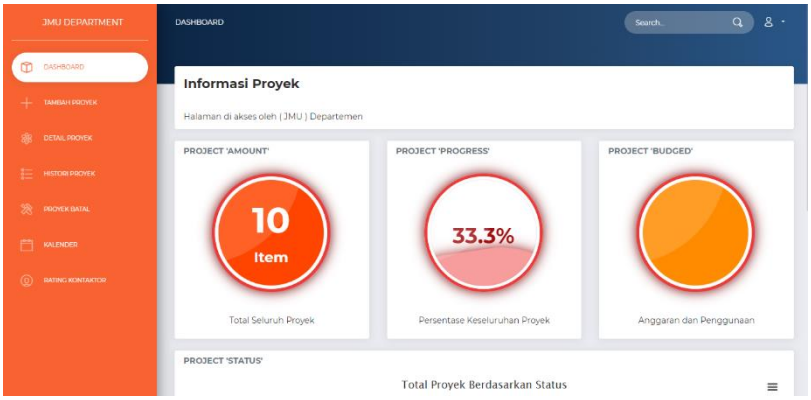


Gambar 11. Halaman *Login*

(Sumber: Data hasil kajian, 2020)

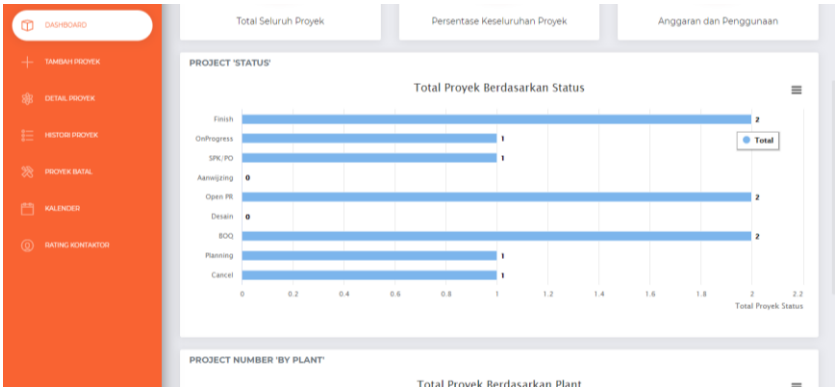
4.1.2 Dashboard

Halaman dashboard berfungsi untuk memonitor keseluruhan jumlah proyek, persentase perkembangan proyek, jumlah status proyek dan banyaknya proyek pada masing - masing *plant*.



Gambar 12. Halaman *Dashboard 1*

(Sumber: Data hasil kajian, 2020)



Gambar 13. Halaman *Dashboard 2*

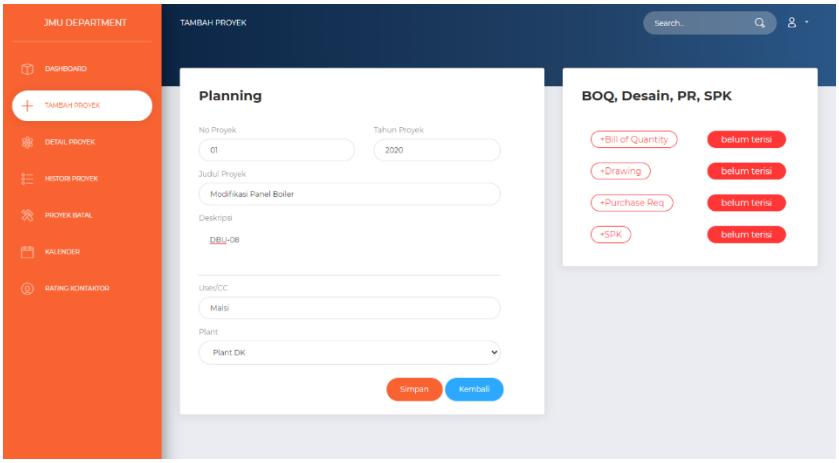
(Sumber: Data hasil kajian, 2020)

Gambar 14. Halaman *Dashboard 3*

(Sumber: Data hasil kajian, 2020)

4.1.3 Tambah Proyek

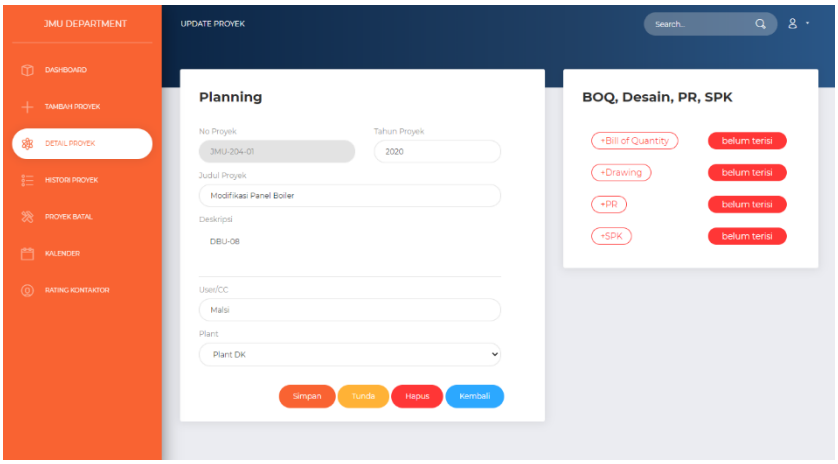
Halaman ini berfungsi untuk menambahkan proyek baru dengan cara mengisi *form* yang tersedia seperti gambar berikut.



Gambar 15. Halaman Tambah Proyek 1

(Sumber: Data hasil kajian, 2020)

Setelah data tersimpan maka halaman akan dialihkan ke halaman edit proyek seperti gambar berikut. Pada halaman ini kita dapat mengakses halaman *Biil of Quantity*, Desain, *Purcahase Requisition* dan Surat Perintah Kerja.

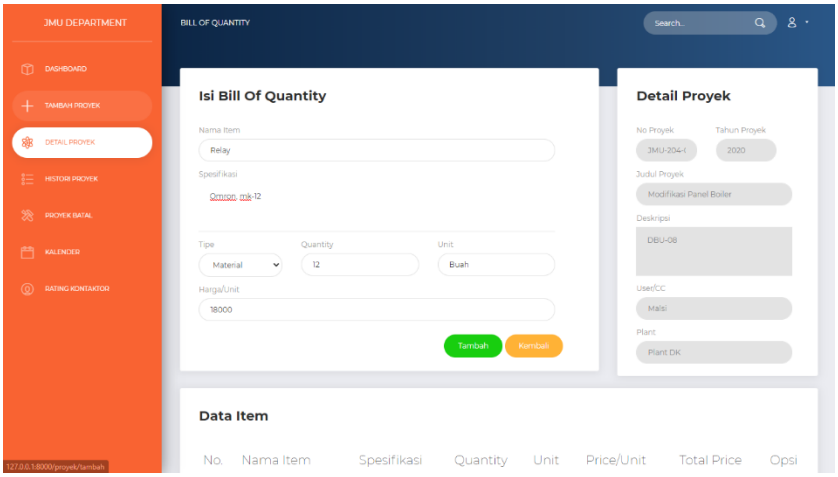


Gambar 16. Halaman Tambah Proyek 2

(Sumber : Data hasil kajian, 2020)

1. *Bill of Quantity*

Halaman ini pengguna dapat memasukan data *item* seperti unit, jumlah dan harga material.

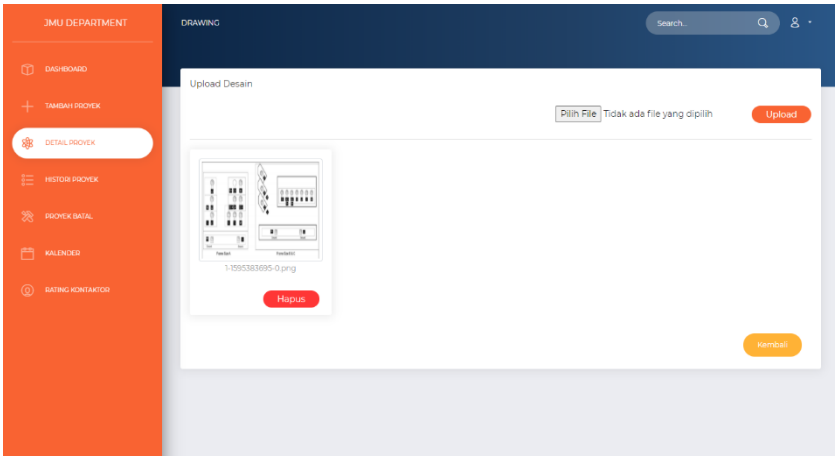


Gambar 17. Halaman Tambah Bill of Quantity

(Sumber : Data hasil kajian, 2020)

2. Desain (*Drawing*)

Halaman ini pengguna dapat mengunggah gambar desain melalui kamera smartphone atau file yang tersimpan.

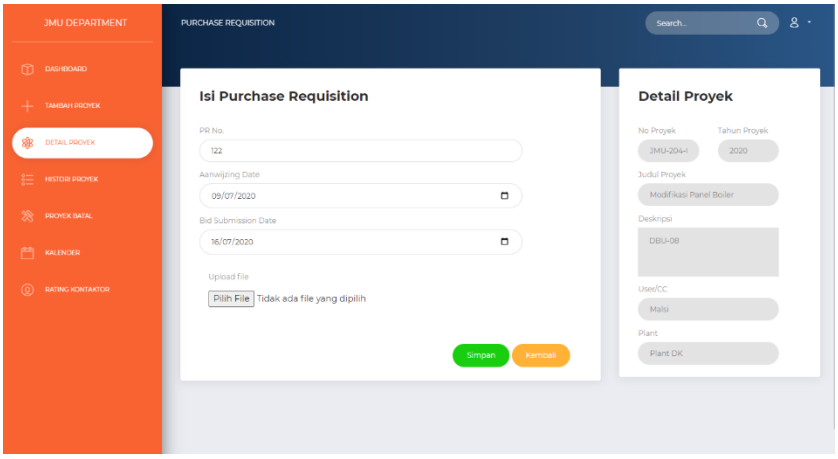


Gambar 18. Halaman Desain

(Sumber : Data hasil kajian, 2020)

3. *Puchase Requisition* (PR)

Halaman ini pengguna dapat mengisi nomor PR, tanggal pertemuan dengan para peserta tender (*Aanwijzing Date*) dan tanggal penganjuan kontraktor (*Bid Submission Date*).

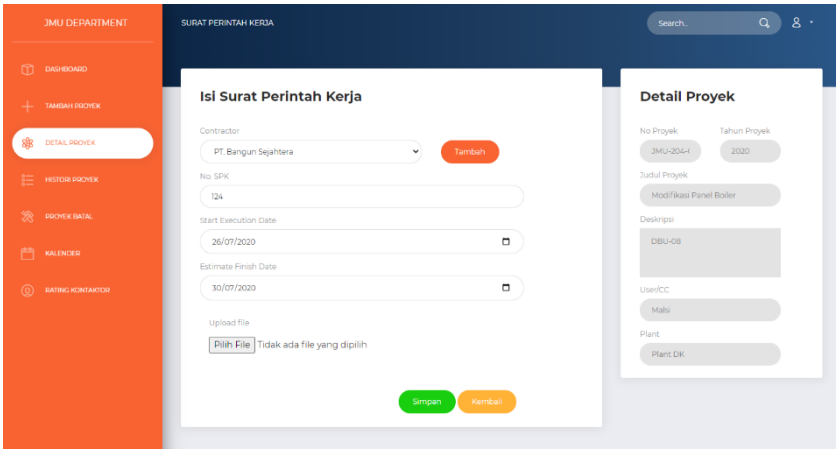


Gambar 19. Halaman *Purchase Requisiton*

(Sumber : Data hasil kajian, 2020)

4. Surat Perintah Kerja (SPK)

Halaman ini berfungsi untuk mengisi data surat perinjtah kerja yang diterima oleh pengguna seperti nomor SPK, tanggal mulai eksekusi proyek dan tanggal perkiraan proyek selesai.

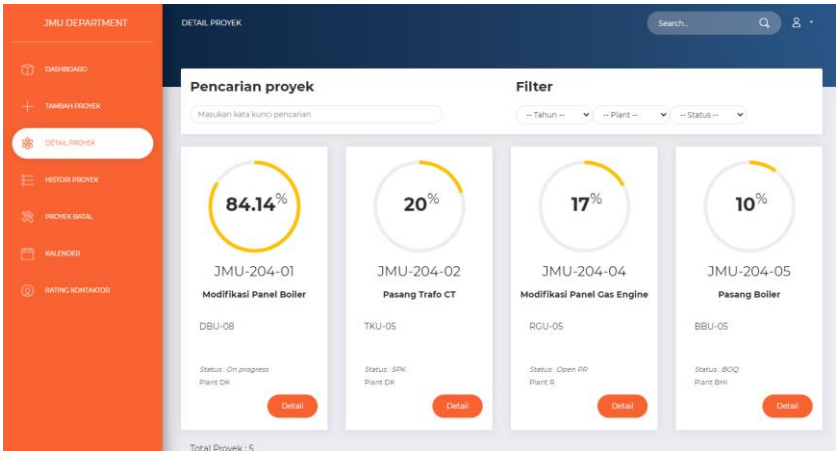


Gambar 20. Halaman Surat Perintah Kerja

(Sumber : Data hasil kajian, 2020)

4.1.4 Detail Proyek

Halaman ini menampilkan persentase perkembangan masing - masing proyek. Tombol detail pada *cardview* akan mengarahkan pengguna menuju halaman detail.



Gambar 21. Halaman Proyek

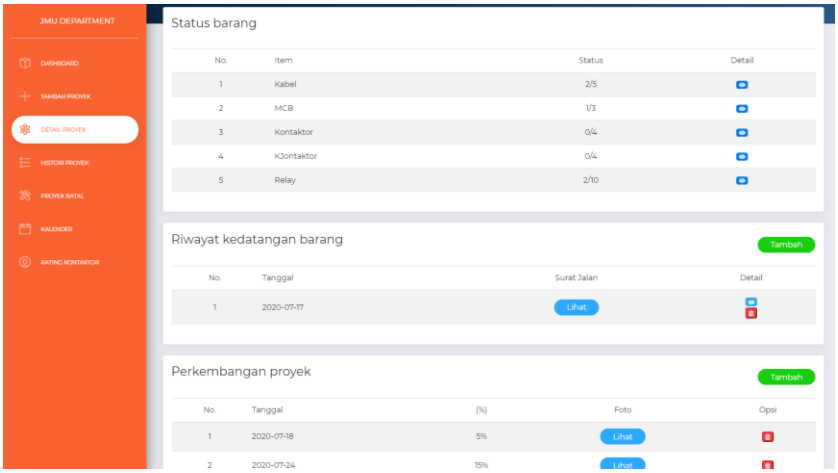
(Sumber : Data hasil kajian, 2020)

Halaman detail berisi informasi detail dan grafik kurva s yang berkaitan dengan proyek. Tombol *edit* digunakan untuk mengakses halaman *edit* yang berfungsi untuk meng-*edit* data yang berkaitan dengan proyek seperti pada Gambar 15. Tombol *update* digunakan untuk mengakses halaman *update progress.*

Gambar 22. Halaman Detail Proyek

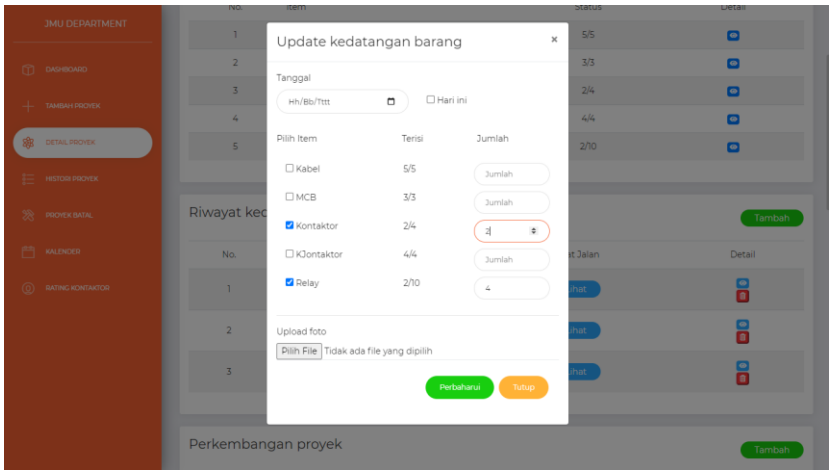
(Sumber : Data hasil kajian, 2020)

Pada halaman *update progress* pengguna dapat melihat status barang, riwayat kedatangan barang dan perkembangan proyek serta memperbarui data - data tersebut.



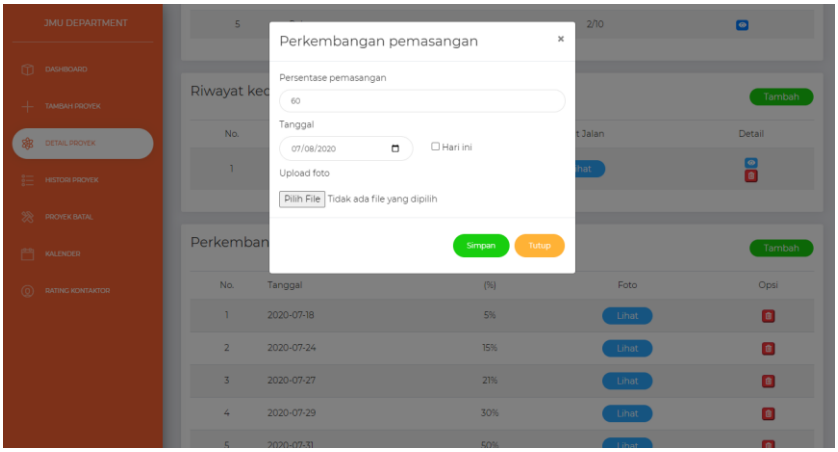
Gambar 23. Halaman Pembaharuan Proyek

(Sumber : Data hasil kajian, 2020)



Gambar 24. Halaman Tambah Surat Jalan

(Sumber : Data hasil kajian, 2020)

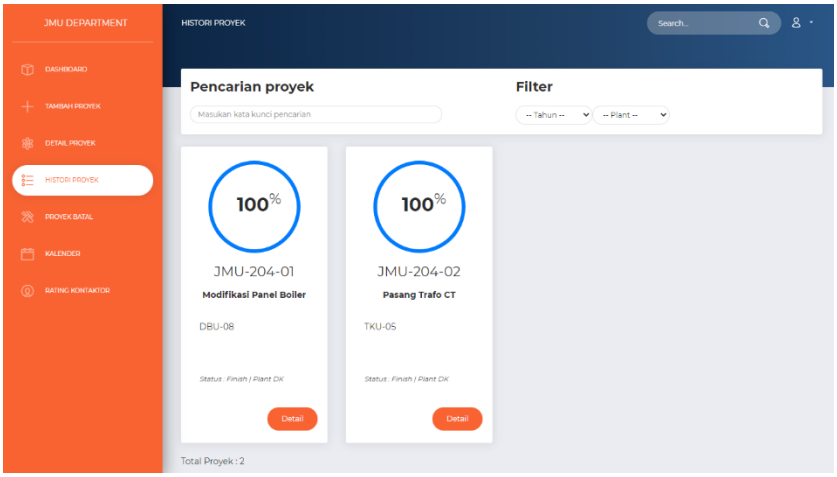


Gambar 25. Halaman Tambah Perkembangan Proyek

(Sumber : Data hasil kajian, 2020)

4.1.5 Riwayat Proyek

Halaman ini menampilkan data proyek yang sudah 100% selesai dikerjakan. Sama seperti halaman detail proyek, tombol detail pada *cardview* akan mengarahkan pengguna menuju halaman detail.



Gambar 26. Halaman Riwayat Proyek

(Sumber : Data hasil kajian, 2020)

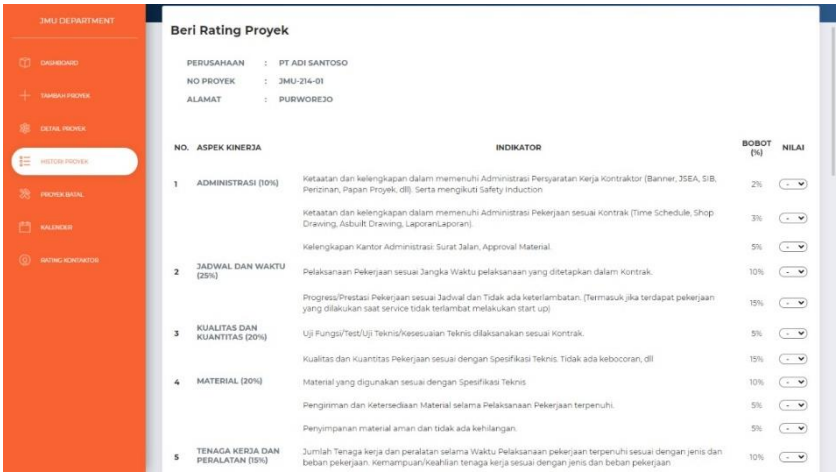
Halaman detail berisi informasi detail dan grafik kurva s yang berkaitan dengan proyek. Tombol *rating* digunakan mengakses halaman untuk menilai kinerja kontraktor, tombol lihat desain digunakan untuk mengakses halaman desain

seperti pada Gambar 17, dan tombol detail riwayat digunakan untuk mengakses halaman detail riwayat.

Gambar 27. Halaman Detail Riwayat Proyek

(Sumber : Data hasil kajian, 2020)

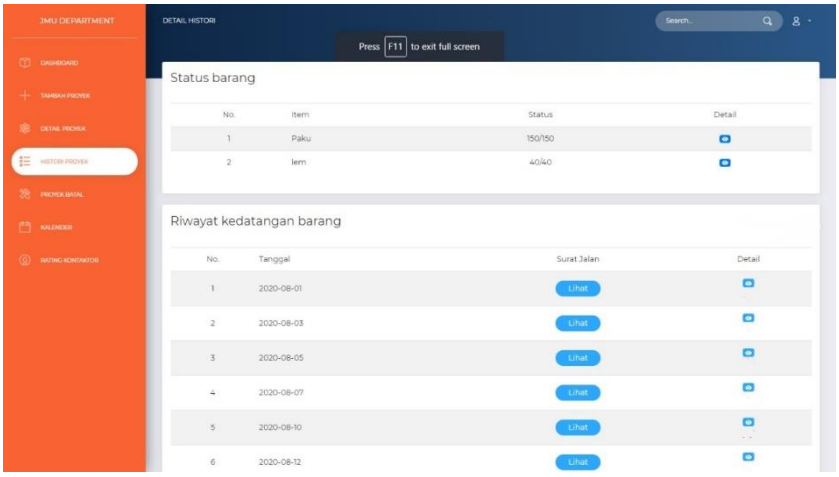
Halaman beri *rating* berfungsi untuk memberikan penilaian hasil kinerja kontraktor terhadap proyek.



Gambar 28. Halaman Beri *Rating*

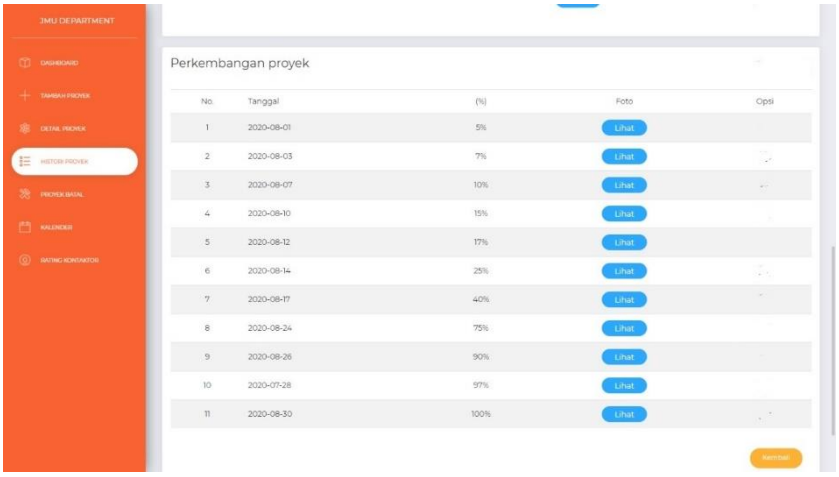
(Sumber : Data hasil kajian, 2020)

Halaman detail riwayat berfungsi untuk melihat data riwayat kedatangan barang dan perkemmbangan proyek.



Gambar 29. Halaman Detail Riwayat 1

(Sumber : Data hasil kajian, 2020)

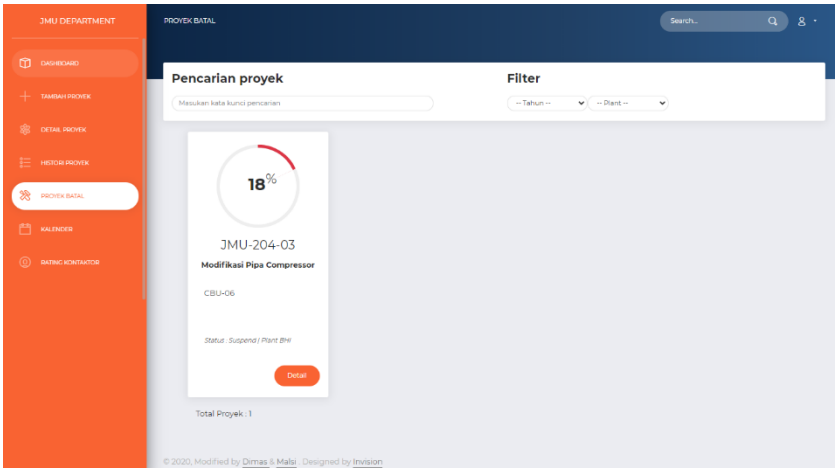


Gambar 30. halaman Detail Riwayat 2

(Sumber : Data hasil kajian, 2020)

4.1.6 Proyek Batal

Halaman ini menampilkan data proyek yang telah ditunda atau batal untuk dibuat. Sama seperti halaman detail proyek, tombol detail pada *cardview* akan mengarahkan pengguna menuju halaman detail yang berisi informasi detail proyek.

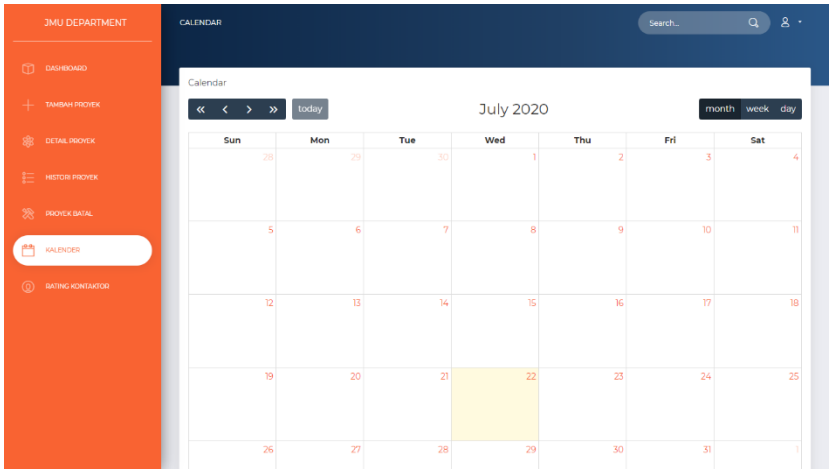


Gambar 31. Halaman Proyek Batal

(Sumber : Data hasil kajian, 2020)

4.1.7 Kalender

Halaman kalender merupakan fitur tambahan untuk melihat tanggal dan kedepannya dapat dikembangkan untuk menampilkan informasi proyek.



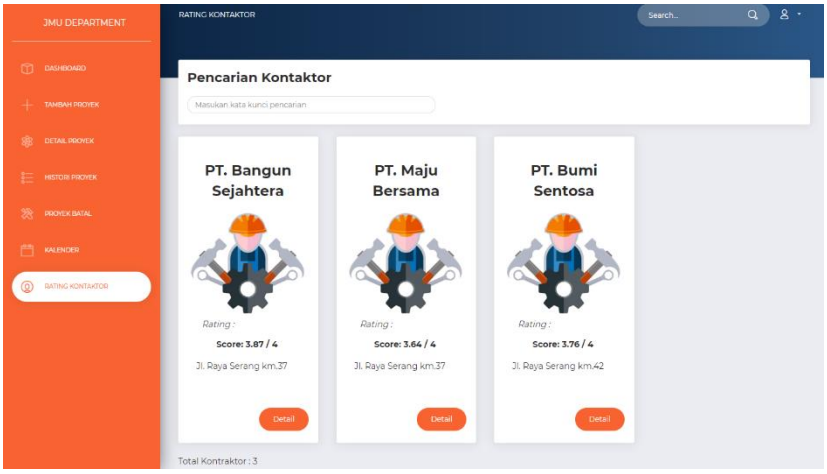
Gambar 32. Halaman Kalender

(Sumber : Data hasil kajian, 2020)

4.1.8 *Rating* Kontraktor

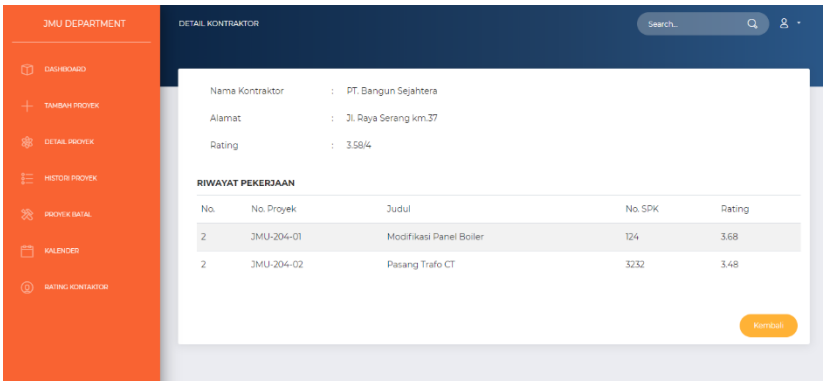
Halaman ini menampilkan hasil rata-rata seluruh penilaian kontraktor yang sebelumnya telah diisi pada masing-masing riwayat proyek. Tombol detail

pada *cardview* berfungsi untuk menampilkan detail riwayat proyek yang telah dikerjakan beserta penilaiannya.



Gambar 33. Halaman Rating Kontraktor

(Sumber : Data hasil kajian, 2020)



Gambar 34. Halaman Detail Rating Kontraktor

(Sumber : Data hasil kajian, 2020)

4.2 Uji Verifikasi

Uji verifikasi merupakan pengujian fungsi dan fitur-fitur yang tersedia pada sistem. Pengujian verifikasi ini dilakukan untuk mengecek apakah fitur pada aplikasi berfungsi sesuai dengan fungsi yang diharapkan. Jika ketika melakukan pengujian masih terdapat fitur-fitur yang tidak sesuai makaakan dilakukan perbaikan-perbaikan sampai aplikasi berfungsi sesuai denghan instruksi yang diharapkan. Berikut merupakan hasil dari uji verifikasi sistem.

Tabel 13. Uji Verifikasi

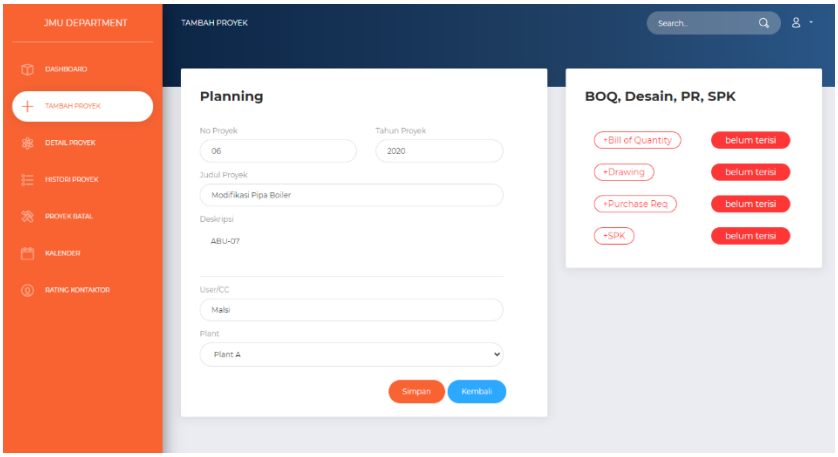
No	Menu	Jumlah Tes	Persentase Keberhasilan
1	*Login*	5	100%
2	*Dashboard*	5	100%
3	Tambah Proyek	5	100%
4	*Bill of Quantity* (BOQ)	5	100%
5	Desain	5	100%
6	*Purchase Requisition* (PR)	5	100%
7	Surat Perintah Kerja (SPK)	5	100%
8	Detail Proyek	5	100%
9	*Edit* Proyek	5	100%
10	*Update* Proyek	5	100%
11	Riwayat Proyek	5	100%
12	Beri *Rating*	5	100%
13	Lihat Desain	5	100%
14	Detail Riwayat	5	100%
15	Proyek Batal	5	100%
16	*Rating* Kontraktor	5	100%
17	Detail Riwayat Kontraktor	5	100%

(Sumber : Data hasil kajian, 2020)

4.3 Uji Validasi

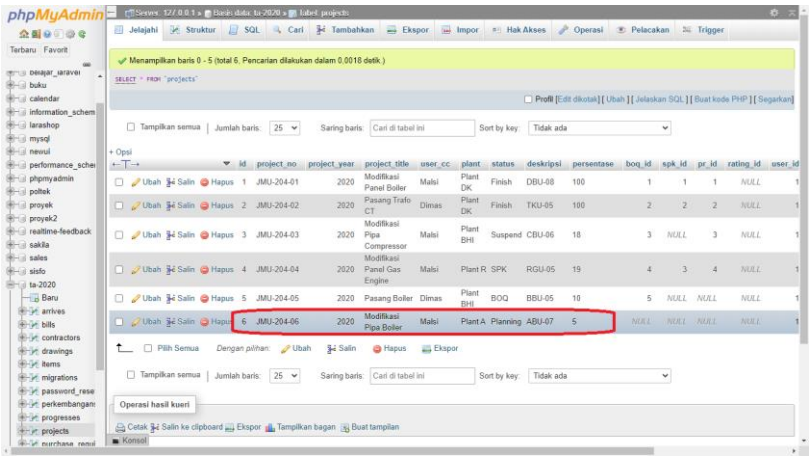
Uji validasi yang dilakukan perancangan ini yaitu membandingkan hasil keluaran dengan tabel pada *database*. Apakah data yang ditampilkan pada keluaran program sesuai dengan data pada *database.* Hal ini dikarenakan *database* digunakan sebagi acuan data sebenarnya. Uji validasi yang dilakukan pada perancangan ini dapat dilihat sebagai berikut.

1. Hasil data yang disimpan melalui halaman tambah proyek tersimpan di *database* dan ditampilkan di halaman detail proyek.



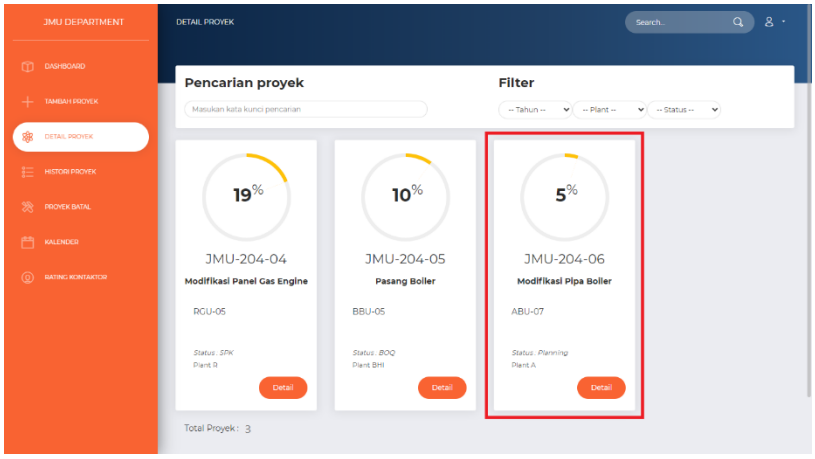
Gambar 35. Validasi Tambah Proyek 1

(Sumber : Data hasil kajian, 2020)



Gambar 36. Validasi Tambah Proyek 2

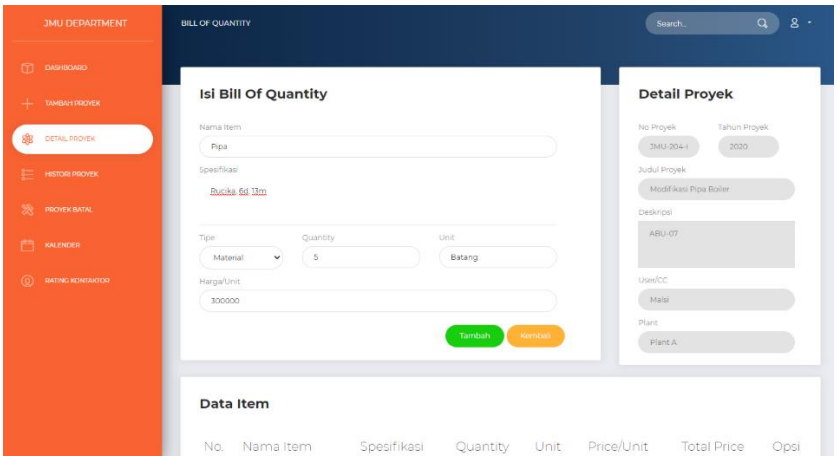
(Sumber : Data hasil kajian, 2020)



Gambar 37. Validasi Tambah Proyek 3

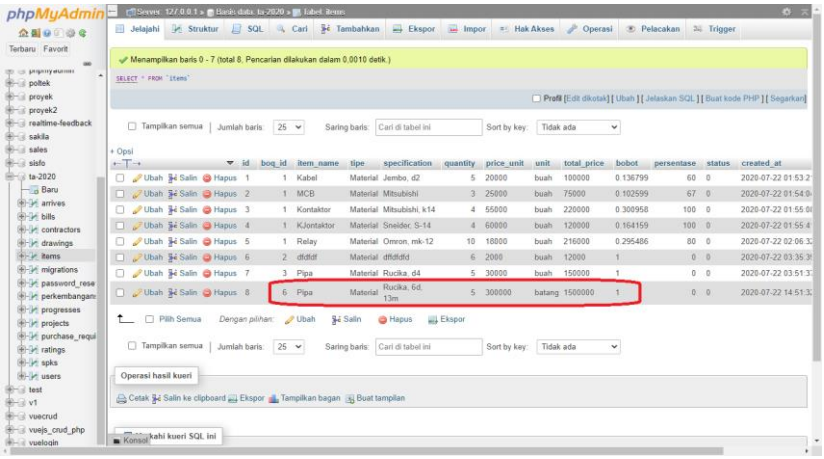
(Sumber : Data hasil kajian, 2020)

2. Hasil data yang disimpan melalui halaman *bill of quantity* tersimpan di *database*.



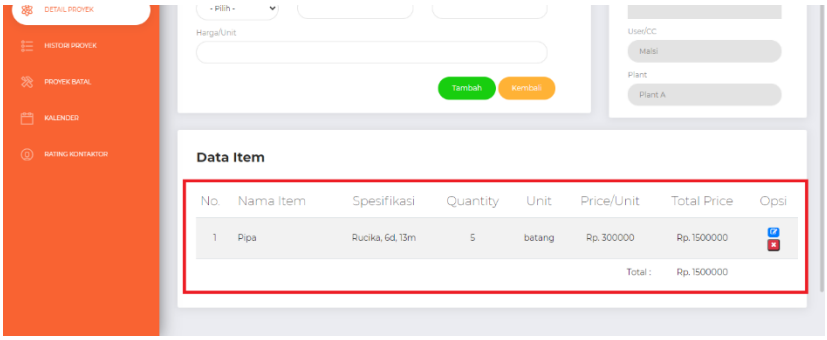
Gambar 38. Validasi Tambah *Item BOQ 1*

(Sumber : Data hasil kajian, 2020)



Gambar 39. Validasi Tambah *Item BOQ 2*

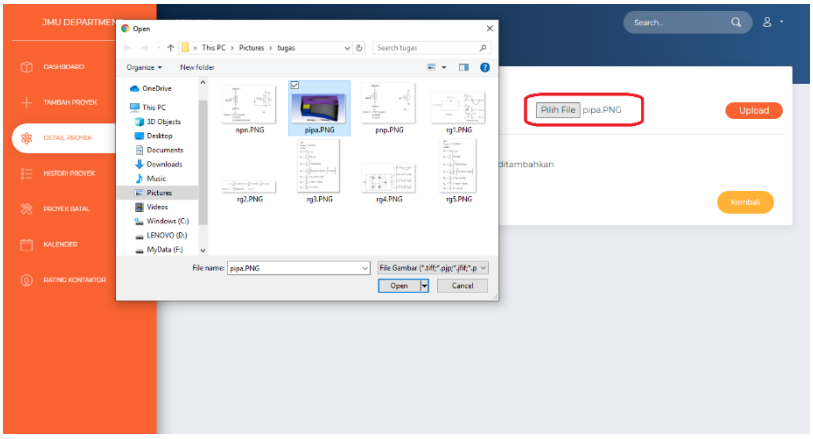
(Sumber : Data hasil kajian, 2020)



Gambar 40. Validasi Tambah *Item BOQ 3*

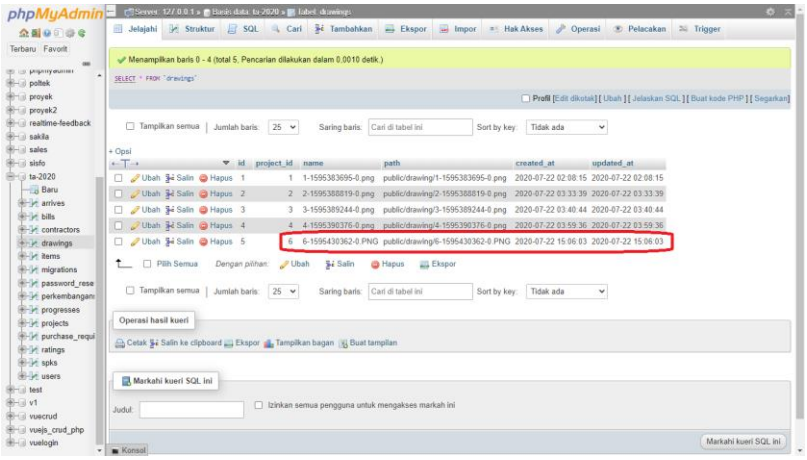
(Sumber : Data hasil kajian, 2020)

3. Hasil data yang disimpan melalui halaman desain tersimpan di *database*.



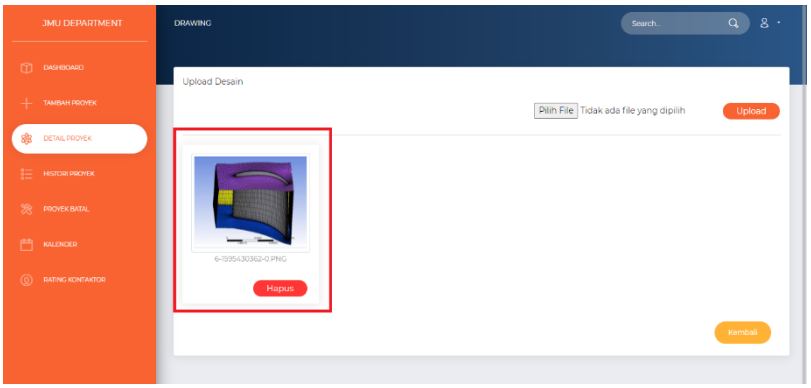
Gambar 41. Validasi Tambah Gambar 1

(Sumber : Data hasil kajian, 2020)



Gambar 42. Validasi Tambah Gambar 2

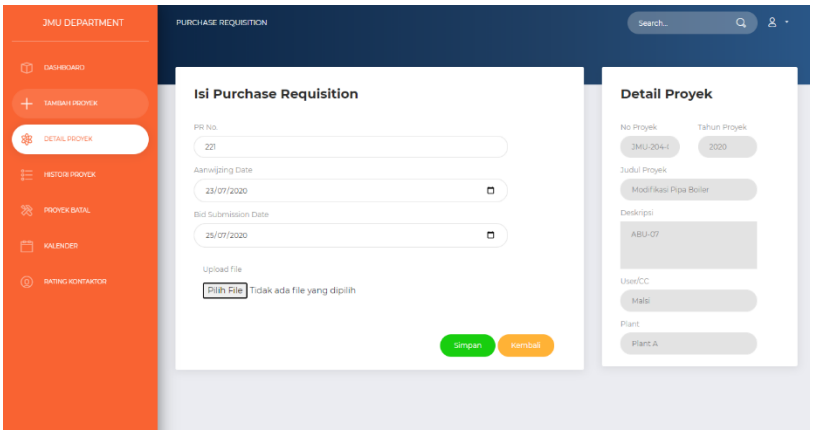
(Sumber : Data hasil kajian, 2020)



Gambar 43. Validasi Tambah Gambar 3

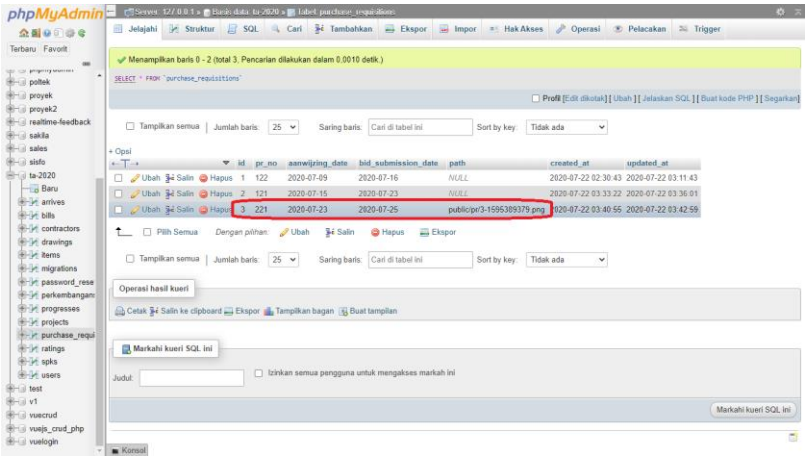
(Sumber : Data hasil kajian, 2020)

4. Hasil data yang disimpan melalui halaman *purchase requisition* tersimpan di *database*.



Gambar 44. Validasi Tambah *Purchase Requisition* 1

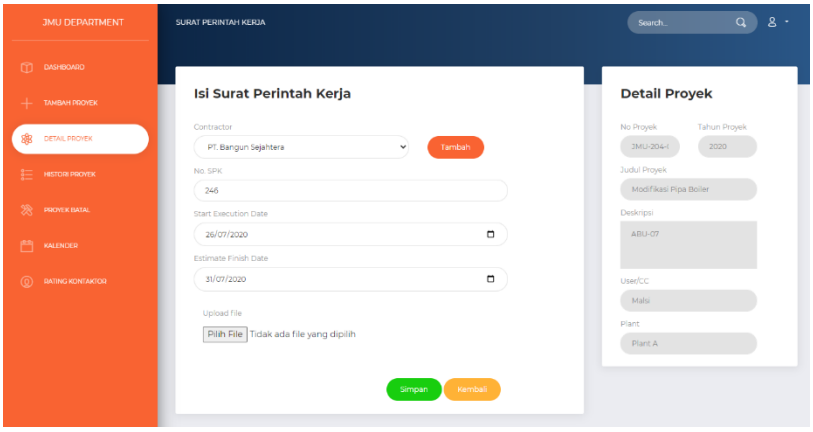
(Sumber : Data hasil kajian, 2020)



Gambar 45. Validasi Tambah *Purchase Requisition* 2

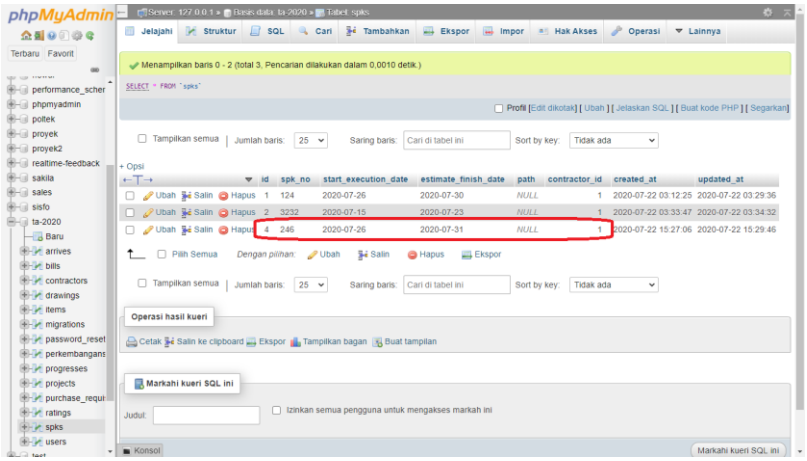
(Sumber : Data hasil kajian, 2020)

5. Hasil data yang disimpan melalui halaman surat perintah kerja tersimpan di *database*.



Gambar 46. Validasi Tambah *Purchase Requisition* 2

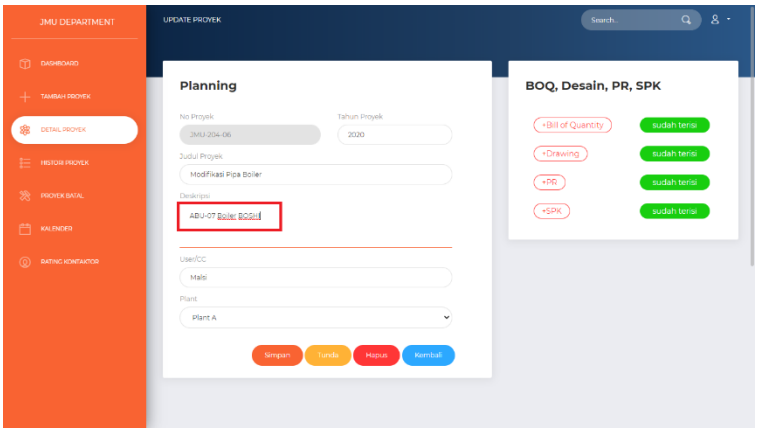
(Sumber : Data hasil kajian, 2020)



Gambar 47. Validasi Tambah *Purchase Requisition* 3

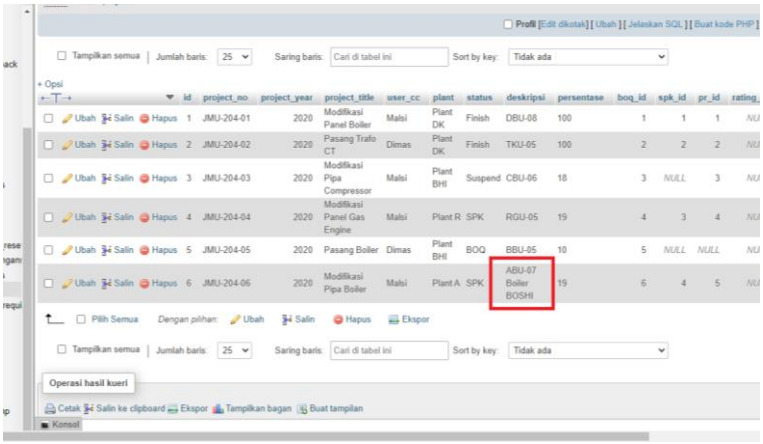
(Sumber : Data hasil kajian, 2020)

6. Hasil data yang di-*edit* melalui halaman *edit* proyek tersimpan di *database*.



Gambar 48. Validasi *Edit* Proyek 1

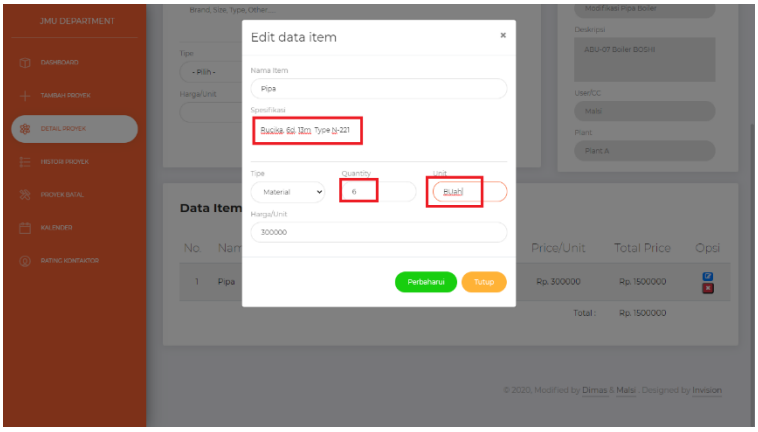
(Sumber : Data hasil kajian, 2020)



Gambar 49. Validasi *Edit* Proyek 2

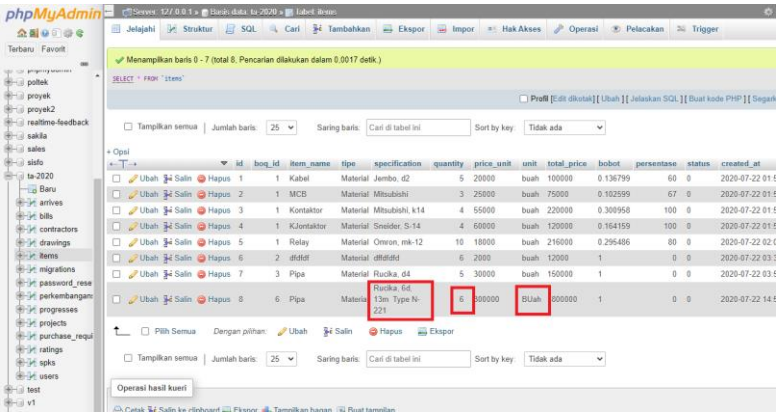
(Sumber : Data hasil kajian, 2020)

7. Hasil data yang di-*edit* melalui halaman *edit bill of quantity* tersimpan di *database*.



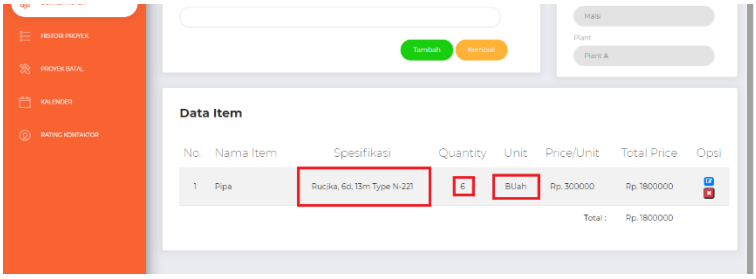
Gambar 50. Validasi *Edit Item BOQ* 1

(Sumber : Data hasil kajian, 2020)



Gambar 51. Validasi *Edit Item BOQ* 2

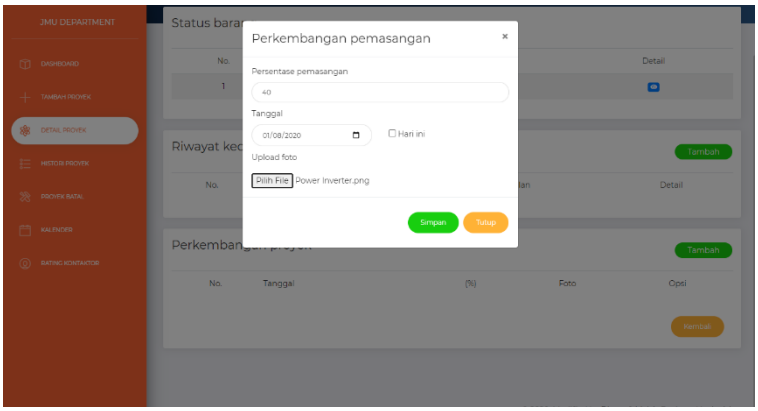
(Sumber : Data hasil kajian, 2020)



Gambar 52, Validasi *Edit Item BOQ 3*

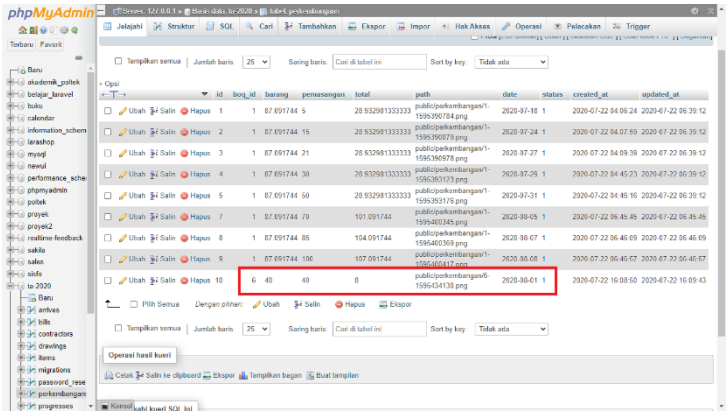
(Sumber : Data hasil kajian, 2020)

8. Hasil data yang disimpan melalui halaman up*date* tersimpan di *database*.



Gambar 53. Validasi Tambah Perkembangan Proyek 1

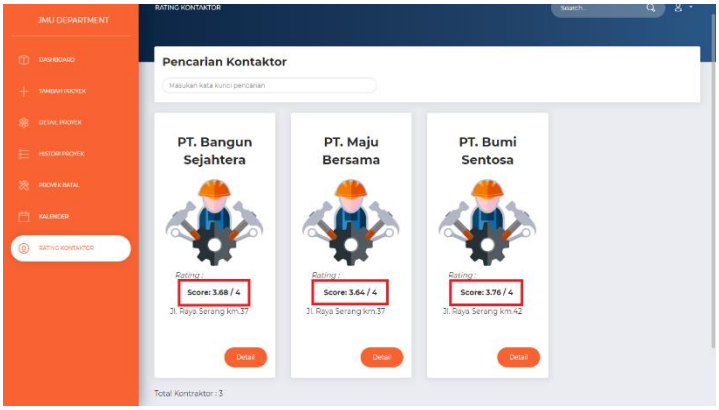
(Sumber : Data hasil kajian, 2020)



Gambar 54. Validasi Tambah Perkembangan Proyek 2

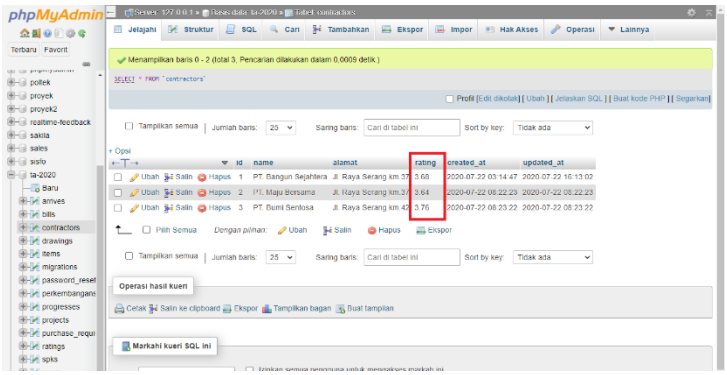
(Sumber : Data hasil kajian, 2020)

9. Hasil data yang disimpan melalui halaman *rating* tersimpan di *database* dan ditampilkan di halaman *rating* kontraktor



Gambar 55. Validasi Tambah *Rating* 1

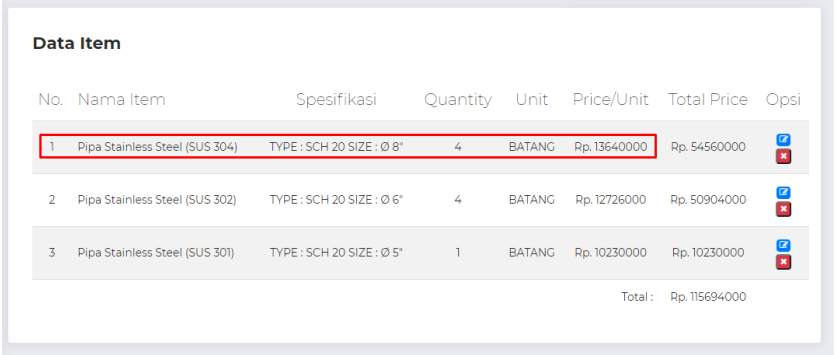
(Sumber : Data hasil kajian, 2020)



Gambar 56. Validasi Tambah *Rating* 2

(Sumber : Data hasil kajian, 2020)

10. Validasi perhitungan boq:

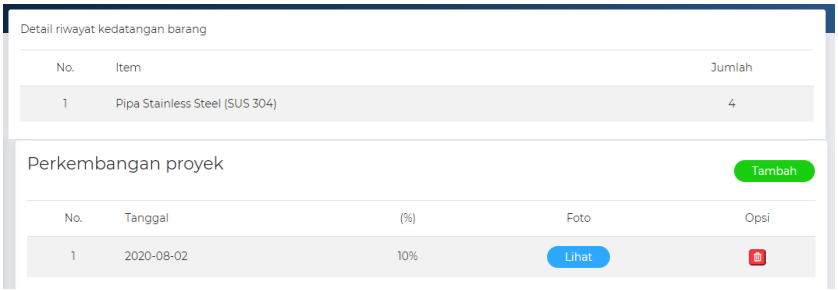


Data Item

No.	Nama Item	Spesifikasi	Quantity	Unit	Price/Unit	Total Price	Opsi
1	Pipa Stainless Steel (SUS 304)	TYPE : SCH 20 SIZE : Ø 8"	4	BATANG	Rp. 13640000	Rp. 54560000	
2	Pipa Stainless Steel (SUS 302)	TYPE : SCH 20 SIZE : Ø 6"	4	BATANG	Rp. 12726000	Rp. 50904000	
3	Pipa Stainless Steel (SUS 301)	TYPE : SCH 20 SIZE : Ø 5"	1	BATANG	Rp. 10230000	Rp. 10230000	
					Total :	Rp. 115694000	

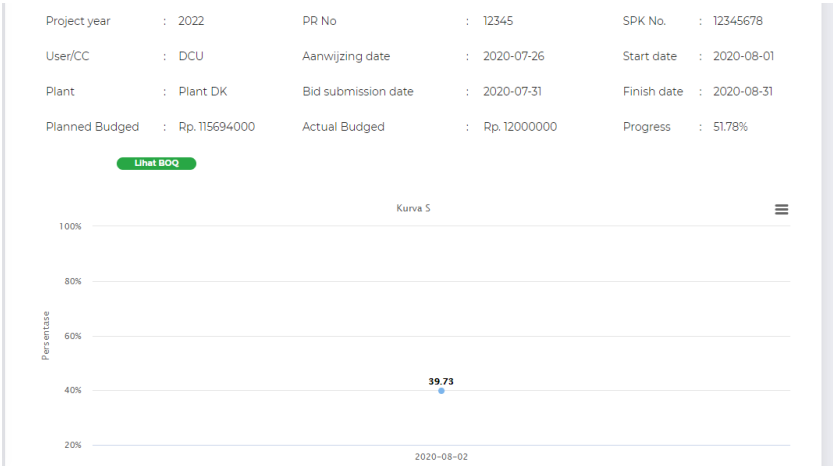
Gambar 57. Validasi perhitungan boq 1

(Sumber : Data hasil kajian, 2020)



Gambar 58. Validasi perhitungan boq 2

(Sumber : Data hasil kajian, 2020)



Gambar 59. Validasi perhitungan boq 3

(Sumber : Data hasil kajian, 2020)

Pada Gambar 57 merupakan riwayat kedatangan barang berupa Pipa *Stainless Steel* (*SUS* 304) yang berjumlah empat dengan harga per batang adalah Rp.13.640.000,- Total biaya proyek adalah Rp. 115.694.000,-. Persentase perkembangan proyek pada tanggal 2 Agustus 2020 adalah 10% seperti pada Gambar 58. Berdasarkan data di atas dapat dapatkan perhitungan sebagai berikut:

Diketahui:

Harga barang per batang	= Rp.13.640.000,-
Jumlah barang	= 4
Total biaya proyek	= Rp.115.694.000,-
Persentase proyek	= 10%

Ditanya:

Persentase kurva s = ?

Jawaban:

Persentase kurva s didapatkan dari:

$$\%kurva\ s = (Total\ bobot\ barang\ \times 80\%) + (Persentase\ perkembangan\ proyek\ \times 20\%)$$

$$Bobot\ barang = \frac{Harga\ barang}{Total\ harga\ barang} \times 100\%$$

$$Bobot\ barang = \frac{13.640.000}{115.694.000} \times 100\%$$

$$Bobot\ barang = 11{,}789\%$$

$$Total\ bobot\ barang = Bobot\ barang\ \times Jumlah\ barang$$

$$Total\ bobot\ barang = 11{,}789\% \times 4$$

$$Total\ bobot\ barang = 47{,}159\%$$

$$\%kurva\ s = (47{,}159\% \times 80\%) + (10\% \times 20\%)$$

$$\%kurva\ s = 37{,}727 + 2$$

$$kurva\ s = 39{,}727\%$$

Berdasarkan perbandingan persentase perhitungan manual diatas dapat disimplukan bahwa perhitugan manual sesuai dengan penggunaan aplikasi seperti pada Gambar 59.

BAB 5 - KESIMPULAN DAN SARAN

5.1. Kesimpulan

Kesimpulan yang dihasilkan dalam penulisan tugas akhir yang berjudul “Rancang Bangun Sistem Informasi Berbasis Web” adalah sistem telah dapat berjalan, sistem dapat menyimpan berkas-berkas kedalam *database*, dan sistem dapat memberikan penilaian kontraktor pada tiap-tiap proyek.

5.2. Saran

Untuk penelitian selanjutnya diharapkan dapat membahas kebutuhan perangkat keras sistem dan *server*.

DAFTAR PUSTAKA

Elmayati, E. (2016). Aplikasi Unit Kegiatan Mahasiswa (UKM) Pada Stmik Mura Lubuklinggau Berbasis Web. Jurnal Teknologi Informasi MURA.

Februariyanti, H., & Zuliarso, E. (2012). Rancang bangun sistem perpustakaan untuk jurnal elektronik. Dinamik.

Firma Sahrul, B., Safi'ie, M. A., & WA, O. D. (2016). Implementasi Sistem Informasi Akademik Berbasis Web Menggunakan Framework Laravel. Jurnal Transformasi.

Fridayanthie, E. W., & Mahdiati, T. (2016). Rancang Bangun Sistem Informasi Permintaan Atk Berbasis Intranet (Studi Kasus: Kejaksaan Negeri Rangkasbitung). Jurnal khatulistiwa informatika.

Harison, H., & Syarif, A. (2016). Sistem Informasi Geografis Sarana Pada Kabupaten Pasaman Barat. Jurnal TeknoIf.

Henderson, H. (2009). Encyclopedia of computer science and technology. Infobase Publishing.

Kevin, Y. (2002). Build Your Own Database Driven Website Using PHP & MySQL, SitePoint Pty. Ltd, Australia.

Madcoms, M. (2011). Aplikasi Web Database dengan Dreamweaver dan PHP-MySQL. Andi, Yogyakarta.

Pahlevi, O., Mulyani, A., & Khoir, M. (2018). Sistem Informasi Inventori Barang Menggunakan Metode Object Oriented Di Pt. Livaza Teknologi Indonesia Jakarta. PROSISKO: Jurnal Pengembangan Riset dan Observasi Sistem Komputer.

Pressman, R. S. (2010). Software Engineering-A Practitioner's Approach.

Rozi, Z. A. (2015). Bootstrap Design Framework. Elex Media Komputindo.

Shalahuddin, M., & Rosa, A. S. (2013). Rekayasa perangkat lunak terstruktur dan berorientasi objek. Bandung: Informatika.

Sunyoto, A. (2010). AJAX (Asynchronus JavaScript and XML). Data Manajemen dan Teknologi Informasi (DASI)

Sutrisna, E. (2018). Analisis Time Schedule Proyek Pembangunan Gedung Vip Rsud Cideres Kabupaten Majalengka. Proceeding Stima.

BIODATA PENULIS

Data Pribadi Penulis

Nama	: Dimas Dwi Cahyo
Tempat Tanggal Lahir	: Klaten, 29 Januari 1998
Agama	: Islam
Alamat	: Dukuhan RT 1 RW 1, Barukan, Manisrenggo, Klaten, Jawa Tengah
Email / No.Hp	: dimas.stmp@gmail.com/085925293385

Pendidikan

2004 - 2010	: SD Negeri 2 Barukan
2010 - 2013	: SMP Negeri 1 Manisrenggo
2013 - 2017	: SMK Negeri 2 Depok

Organisasi

2007 - 2010	: Pramuka
2011 - 2013	: OSIS
2013 - 2016	: Jurnalistik

Data Pribadi Penulis

Nama	: Malsi Nur Adwinda Robbani
Tempat Tanggal Lahir	: Ngawi, 23 September 1999
Agama	: Islam
Alamat	: Perum Bukit Tiara Blok M2. No.20, RT.37/06, Cikupa Tangerang
Email / No.Hp	: malsinur.a.r@gmail.com/087778533652

Pendidikan

2005 - 2011	: SD Negeri Pasir Jaya
2011 - 2014	: SMP Negeri 3 Cikupa
2014 - 2017	: SMK Negeri 1 Kab. Tangerang

Organisasi

2008 - 2011	: Pramuka
2011 - 2014	: Paskibra
2013 – Sekarang	: Karang Taruna 37

LAMPIRAN