



**RANCANG BANGUN APLIKASI PEMENUHAN BAHAN BAKU
PRODUKSI UMKM PADA *STARTUP* USAHANESIA MENGGUNAKAN
METODE *TEXT MINING* BERBASIS *WEB***



TUGAS AKHIR

**Program Studi
S1 Sistem Informasi**

Oleh:

Yudha Arista Setiawan

15.41010.0032

**FAKULTAS TEKNOLOGI DAN INFORMATIKA
INSTITUT BISNIS DAN INFORMATIKA STIKOM SURABAYA
2019**

**RANCANG BANGUN APLIKASI PEMENUHAN BAHAN BAKU
PRODUKSI UMKM PADA *STARTUP* USAHANESIA MENGGUNAKAN
METODE *TEXT MINING* BERBASIS *WEB***

TUGAS AKHIR

Diajukan sebagai syarat untuk menyelesaikan

Program Sarjana



Oleh:

Nama : Yudha Arista Setiawan

NIM : 15.41010.0032

Program : S1 (Strata Satu)

Jurusan : Sistem Informasi

**FAKULTAS TEKNOLOGI DAN INFORMATIKA
INSTITUT BISNIS DAN INFORMATIKA STIKOM SURABAYA
2019**

**RANCANG BANGUN APLIKASI PEMENUHAN BAHAN BAKU
PRODUKSI UMKM PADA *STARTUP* USAHANESIA MENGGUNAKAN
METODE *TEXT MINING* BERBASIS *WEB***

dipersiapkan dan disusun oleh

Yudha Arista Setiawan

NIM : 15.41010.0032

Telah diperiksa, diuji dan disetujui oleh Dewan Penguji

Pada Maret 2019

Susunan Dewan Penguji

Pembimbing

I. Vivine Nurcahyawati, M.Kom.
NIDN. 0723018101

II. Norma Ningsih, S.ST., M.T.
NIDN. 0729099002

Pembahas

I. Pantjawati Sudarmaningtyas, S.Kom., M.Eng.
NIDN. 0712066801

 04/03/19



 04/03/19

Tugas Akhir ini telah diterima sebagai salah satu persyaratan

untuk memperoleh gelar Sarjana



FAKULTAS TEKNOLOGI
DAN INFORMATIKA

stikom
SURABAYA

Dr. Jusak

Dekan Fakultas Teknologi dan Informatika

5/3/19

**FAKULTAS TEKNOLOGI DAN INFORMATIKA
INSTITUT BISNIS DAN INFORMATIKA STIKOM SURABAYA**

PERNYATAAN

PERSETUJUAN PUBLIKASI DAN KEASLIAN KARYA ILMIAH

Sebagai mahasiswa Institut Bisnis dan Informatika Stikom Surabaya, saya:

Nama : Yudha Arista Setiawan
NIM : 15.41010.0032
Program Studi : SI Sistem Informasi
Fakultas : Fakultas Teknologi dan Informatika
Jenis Karya : Tugas Akhir
Judul Karya : **RANCANG BANGUN APLIKASI PEMENUHAN
BAHAN BAKU PRODUKSI UMKM PADA
STARTUP USAHANESIA MENGGUNAKAN
METODE TEXT MINING BERBASIS WEB**

Menyatakan dengan sesungguhnya bahwa:

1. Demi pengembangan Ilmu Pengetahuan, Teknologi dan Seni, saya menyetujui memberikan kepada Institut Bisnis dan Informatika Stikom Surabaya Hak Bebas Royalti Non-Eksklusif (*Non-Exclusive Royalti Free Right*) atas seluruh isi/ sebagian karya ilmiah saya tersebut diatas untuk disimpan, dialihmediakan dan dikelola dalam bentuk pangkalan data (*database*) untuk selanjutnya didistribusikan atau dipublikasikan demi kepentingan akademis dengan tetap mencantumkan nama saya sebagai penulis atau pencipta dan sebagai pemilik Hak Cipta
2. Karya tersebut diatas adalah karya asli saya, bukan plagiat baik sebagian maupun keseluruhan. Kutipan, karya atau pendapat orang lain yang ada dalam karya ilmiah ini adalah semata hanya rujukan yang dicantumkan dalam Daftar Pustaka saya
3. Apabila dikemudian hari ditemukan dan terbukti terdapat tindakan plagiat pada karya ilmiah ini, maka saya bersedia untuk menerima pencabutan terhadap gelar kesarjanaan yang telah diberikan kepada saya.

Demikian surat pernyataan ini saya buat dengan sebenarnya.

Surabaya, Maret 2019

Yang menyatakan



Yudha Arista Setiawan
NIM: 15.41010.0032



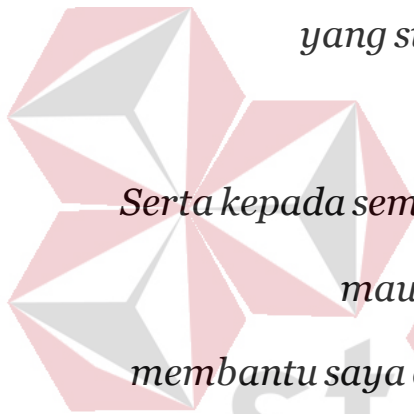
*“Believe in yourself, Do the best and Pray.
Then ALLAH will take care of the rest..”*

INSTITUT BISNIS
DAN INFORMATIKA

stikom
SURABAYA

*Ku persembahkan kepada
Ayah dan ibuku dan keluargaku tercinta,
atas segala dukungan
yang sudah mereka berikan,*

*Serta kepada semua orang yang secara langsung,
maupun tidak langsung
membantu saya dalam menyelesaikan karya ini.*



stikom
SURABAYA

ABSTRAK

Usahanesia mempunyai tujuan membantu menemukan bahan baku produksi yang sesuai dengan kebutuhan UMKM dengan memberikan akses informasi yang relevan. Permasalahan yang sering terjadi pada UMKM yaitu kesulitan untuk mendapatkan bahan baku produksi karena adanya persaingan yang ketat dalam mendapatkan bahan baku, bahan baku berkualitas rendah, dan tingginya harga bahan baku. Hal ini menyebabkan perkembangan UMKM menjadi terhambat dan tidak mendapatkan profit seperti tujuan bisnisnya.

Tugas akhir ini memberikan solusi berupa aplikasi berbasis *crowdsourcing* yang dapat mempertemukan UMKM yang sedang membutuhkan bahan baku dengan UMKM yang sedang kelebihan bahan baku untuk saling berinteraksi. Aplikasi tersebut menggunakan metode *text mining*, algoritma *naïve bayes classifier*, *boyer moore*, *jaro winkler* dan metode *graphic rating scale*. Metode *text mining* dengan *plug in Sastrawi* digunakan untuk membaca dan mengelola status portofolio yang dimasukkan oleh UMKM, sehingga menjadi sebuah kata kunci (*keyword*) yang nantinya akan digunakan pada proses berikutnya. Algoritma *naïve bayes classification* digunakan untuk dapat menentukan apakah UMKM tersebut sedang membutuhkan ataukah kelebihan bahan baku, lalu dengan menggunakan algoritma *boyer moore*, algoritma *jaro winkler* dan *graphic rating scale* untuk menghasilkan rekomendasi yang sesuai dengan kebutuhan UMKM.

Aplikasi pemenuhan bahan baku UMKM menggunakan *text mining* berbasis web ini telah diuji dengan pengujian *black box*. Hasil evaluasi aplikasi penerapan *text mining* untuk pemenuhan bahan baku UMKM berbasis web ini

memiliki tingkat keberhasilan sebesar 100% dari 34 *test case*. Sehingga Aplikasi pemenuhan bahan baku UMKM menggunakan *text mining* berbasis web ini telah sesuai dengan kebutuhan pengguna (*Story*).

Kata Kunci: *Bahan Baku UMKM, Text Mining, Web*



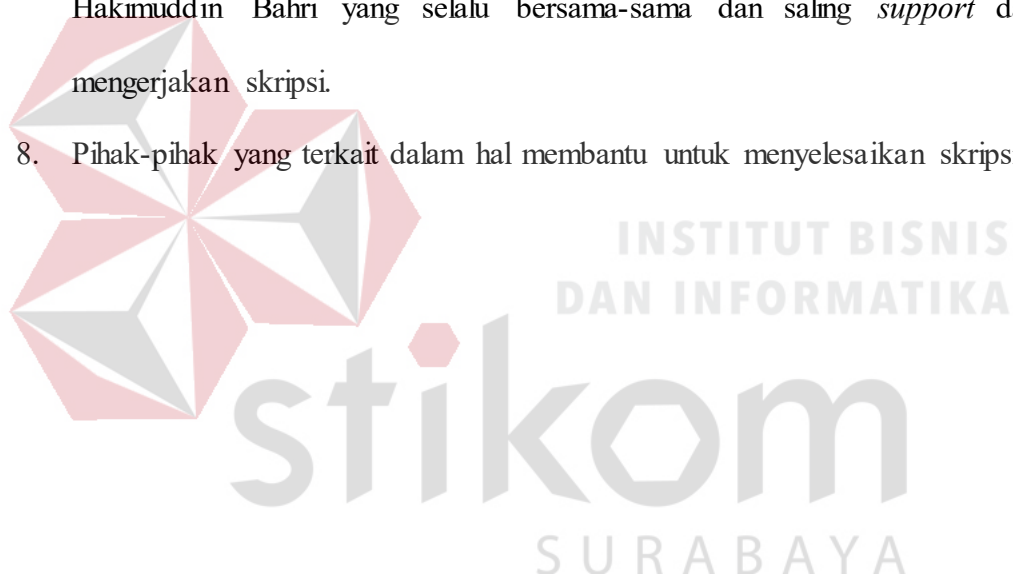
KATA PENGANTAR

Alhamdulillah puji syukur kehadiran Allah Subhanahu wata'ala, salawat serta salam penulis junjungkan kepada Nabi Besar Muhammad Shallallahu alaihi wasallam yang membawa kita keluar dari zaman jahiliyah. Atas limpahan rahmat dan karunia-Nya, tugas akhir ini dapat diselesaikan sesuai dengan rencana dengan judul **“RANCANG BANGUN APLIKASI PEMENUHAN BAHAN BAKU PRODUKSI UMKM PADA *STARTUP* USAHANESIA MENGGUNAKAN METODE *TEXT MINING* BERBASIS *WEB*”**. Tugas Akhir ini adalah salah satu syarat untuk menyelesaikan Program Studi Strata Satu di Fakultas Teknologi dan Informatika Institut Bisnis dan Informatika Stikom Surabaya.

Dengan keterbatasan dan kekurangan penulis dalam menyelesaikan tugas akhir ini, penulis menyadari bahwa ada ketidaksempurnaan, oleh sebab itu penulis sangat mengharapkan kritik dan saran yang membangun. Berkat dorongan dan motivasi dari pihak-pihak yang telah membantu penulis, maka tugas akhir ini dapat diselesaikan pada waktunya. Melalui kesempatan ini, saya mengucapkan terima kasih kepada:

1. Terima kasih kepada keluarga penulis yang selalu memberikan motivasi serta dorongan untuk menyelesaikan skripsi ini.
2. Bapak Prof. Dr. Budi Jatmiko, M.Pd selaku Rektor Institut Bisnis dan Informatika Stikom Surabaya.
3. Bapak Dr. Anjik Sukmaaji, S.Kom., M.Eng. selaku Ketua Program Studi S1 Sistem Informasi Institut Bisnis dan Informatika Stikom Surabaya

4. Ibu Vivine Nurcahyawati, M.Kom., selaku Dosen Pembimbing 1 yang selalu memberikan arahan untuk membantu menyelesaikan skripsi ini.
5. Ibu Norma Ningsih, S.ST., M.T. selaku Dosen Pembimbing 2 yang selalu memberikan arahan untuk membantu menyelesaikan skripsi ini.
6. Ibu Pantjawati Sudarmaningtyas, S.Kom., M.Eng. selaku Dosen Pembahas yang telah bersedia menjadi dosen pembahas saya dalam mengerjakan skripsi ini.
7. Terima kasih kepada Budi Prasetyo, Bhisma Dwi Wardana dan Achmad Hakimuddin Bahri yang selalu bersama-sama dan saling *support* dalam mengerjakan skripsi.
8. Pihak-pihak yang terkait dalam hal membantu untuk menyelesaikan skripsi ini.



Surabaya, Februari 2019

Penulis

DAFTAR ISI

| | Halaman |
|---|---------|
| ABSTRAK | vi |
| KATA PENGANTAR..... | viii |
| DAFTAR ISI..... | x |
| DAFTAR GAMBAR | xiii |
| DAFTAR TABEL..... | xvii |
| DAFTAR LAMPIRAN | xix |
| BAB I PENDAHULUAN | 1 |
| 1.1. Latar Belakang..... | 1 |
| 1.2. Rumusan Masalah..... | 3 |
| 1.3. Batasan Masalah | 3 |
| 1.4. Tujuan | 3 |
| 1.5. Manfaat | 3 |
| 1.6. Sistematika Penulisan | 4 |
| BAB II LANDASAN TEORI | 6 |
| 2.1. Usaha Mikro Kecil dan Menengah (UMKM)..... | 6 |
| 2.2. <i>Text Mining</i> | 7 |
| 2.3. <i>Plug In Text Mining Sastrawi</i> | 13 |
| 2.3.1. <i>Plug In Tokenizer Sastrawi</i> | 13 |
| 2.3.2. <i>Plug In Filtering & Stemming Sastrawi</i> | 14 |
| 2.4. <i>Algoritma Naïve Bayes Classifier</i> | 15 |
| 2.5. <i>API Naïve Bayes Classifier MonkeyLearn</i> | 17 |

| | |
|--|-----|
| 2.6. <i>String Matching</i> (Pencocokan <i>String</i> /Kata) | 18 |
| 2.7. Algoritma <i>Boyer-Moore</i> | 19 |
| 2.8. Algoritma <i>Jaro-Winkler</i> | 22 |
| 2.9. <i>Crowdsourcing</i> | 24 |
| 2.10. <i>Graphic Rating Scales</i> (GRS) | 26 |
| 2.11. <i>Agile Development</i> | 28 |
| 2.12. <i>Scrum</i> | 28 |
| 2.13. UML | 32 |
| 2.14. Pengujian Aplikasi | 34 |
| BAB III Metodologi Penelitian | 36 |
| 3.1. Tahap Awal | 36 |
| 3.1.1. Mengidentifikasi Masalah | 36 |
| 3.1.2. <i>Product Backlog</i> | 48 |
| 3.1.3. Model Pengembangan | 54 |
| 3.2. Tahap Pengembangan | 57 |
| 3.2.1. <i>Sprint Planning</i> | 57 |
| 3.2.2. <i>Sprint Backlog</i> | 58 |
| 3.2.3. <i>Daily Scrum</i> | 106 |
| 3.3. Tahap Akhir | 108 |
| 3.3.1. <i>Sprint Review</i> | 109 |
| 3.3.2. <i>Sprint Retrospective</i> | 126 |
| 3.3.3. <i>Product</i> | 127 |
| BAB IV IMPLEMENTASI DAN EVALUASI SISTEM | 128 |
| 4.1. Implementasi | 128 |

| | |
|--|-----|
| 4.1.1. Kebutuhan <i>Software</i> dan <i>Hardware</i> | 128 |
| 4.1.2. Aplikasi..... | 129 |
| 4.2. Evaluasi..... | 146 |
| 4.2.1. Evaluasi Kesesuaian Fitur | 146 |
| 4.2.2. Hasil <i>Testing</i> | 148 |
| BAB V KESIMPULAN DAN SARAN..... | 156 |
| 5.1. Kesimpulan | 156 |
| 5.2. Saran..... | 156 |
| DAFTAR PUSTAKA | 158 |
| BIODATA..... | 161 |
| LAMPIRAN..... | 162 |



INSTITUT BISNIS
DAN INFORMATIKA

stikom
SURABAYA

DAFTAR GAMBAR

| | Halaman |
|---|---------|
| Gambar 2.1 <i>Flowchart Plug In Tokenizer</i> | 13 |
| Gambar 2.2 <i>Flowchart Plug In Filtering</i> | 14 |
| Gambar 2.3 <i>Flowchart Plug In Stemming</i> | 15 |
| Gambar 2.4 <i>Flowchart API Naive Bayes Classifier</i> | 18 |
| Gambar 2.5 <i>Proses Crowdsourcing. (Whitla, 2009)</i> | 25 |
| Gambar 2.6 <i>Tahapan Scrum Framework. (International Scrum Institute, n.d.)</i> | 29 |
| Gambar 3.1 <i>Tahapan Penelitian Scrum</i> | 36 |
| Gambar 3.2 <i>Business Lean Canvas Usahanesia</i> | 37 |
| Gambar 3.3 <i>Business Use Case Diagram</i> | 39 |
| Gambar 3.4 <i>Activity Diagram Mendaftar</i> | 40 |
| Gambar 3.5 <i>Activity Diagram Mengelola Bahan Baku</i> | 41 |
| Gambar 3.6 <i>Activity Diagram Publikasi Aktivitas UMKM</i> | 42 |
| Gambar 3.7 <i>Activity Diagram Melakukan Penjualan</i> | 43 |
| Gambar 3.8 <i>Activity Diagram Melakukan Pembelian</i> | 44 |
| Gambar 3.9 <i>Activity Diagram Mengelola Data Kategori</i> | 45 |
| Gambar 3.10 <i>Activity Diagram Konfirmasi Pemesanan Bahan Baku</i> | 46 |
| Gambar 3.11 <i>Activity Diagram Menentukan Kebutuhan UMKM</i> | 47 |
| Gambar 3.12 <i>Activity Diagram Validasi Pembayaran</i> | 48 |
| Gambar 3.13 <i>Use Case System Diagram</i> | 50 |
| Gambar 3.14 <i>Model Pengembangan</i> | 55 |
| Gambar 3.15 <i>Sequence Diagram Login Admin</i> | 63 |

| | |
|---|-----|
| Gambar 3.16 <i>Class Diagram Login Admin</i> | 64 |
| Gambar 3.17 <i>Sequence Diagram Mengelola Bag Of Word</i> | 67 |
| Gambar 3.18 <i>Class Diagram Mengelola Bag Of Word</i> | 68 |
| Gambar 3.19 <i>Sequence Diagram Mengelola Data Kategori</i> | 71 |
| Gambar 3.20 <i>Class Diagram Mengelola Data Kategori</i> | 72 |
| Gambar 3.21 <i>Sequence Diagram Validasi Pembayaran</i> | 75 |
| Gambar 3.22 <i>Class Diagram Validasi Pembayaran</i> | 76 |
| Gambar 3.23 <i>Sequence Diagram Register UMKM</i> | 78 |
| Gambar 3.24 <i>Class Diagram Register UMKM</i> | 78 |
| Gambar 3.25 <i>Sequence Diagram Login UMKM</i> | 80 |
| Gambar 3.26 <i>Class Diagram Login UMKM</i> | 81 |
| Gambar 3.27 <i>Sequence Diagram Mengisi Timeline Portofolio</i> | 83 |
| Gambar 3.28 <i>Class Diagram Mengisi Timeline Portofolio</i> | 84 |
| Gambar 3.29 <i>Sequence Diagram Melihat Klasifikasi Status Timeline Portofolio</i> | 88 |
| Gambar 3.30 <i>Class Diagram Melihat Klasifikasi Status Timeline Portofolio</i> | 89 |
| Gambar 3.31 <i>Sequence Diagram Pembelian Bahan Baku</i> | 93 |
| Gambar 3.32 <i>Class Diagram Pembelian Bahan Baku</i> | 94 |
| Gambar 3.33 <i>Sequence Diagram Review Bahan Baku</i> | 96 |
| Gambar 3.34 <i>Class Diagram Review Bahan Baku</i> | 97 |
| Gambar 3.35 <i>Sequence Diagram Penjualan Bahan Baku</i> | 100 |
| Gambar 3.36 <i>Class Diagram Penjualan Bahan Baku</i> | 101 |
| Gambar 3.37 <i>Sequence Diagram Melihat Laporan</i> | 104 |
| Gambar 3.38 <i>Class Diagram Melihat Laporan</i> | 105 |
| Gambar 3.39 <i>Perubahan Use Case System Diagram</i> | 113 |

| | |
|---|-----|
| Gambar 3.40 Perubahan <i>Sequence Diagram</i> Melihat Klasifikasi Status <i>Timeline</i> <i>Portofolio</i> | 117 |
| Gambar 3.41 Perubahan <i>Class Diagram</i> Melihat Klasifikasi Status <i>Timeline</i> <i>Portofolio</i> | 118 |
| Gambar 3.42 Desain Halaman <i>Login Admin</i> | 119 |
| Gambar 3.43 Desain Halaman Mengelola Data <i>Bag Of Word</i> | 120 |
| Gambar 3.44 Desain Halaman Mengelola Data Kategori..... | 120 |
| Gambar 3.45 Desain Halaman <i>Login UMKM</i> | 121 |
| Gambar 3.46 Desain Halaman <i>Register Profil UMKM</i> | 122 |
| Gambar 3.47 Desain Halaman <i>Register Struktur Organisasi UMKM</i> | 122 |
| Gambar 3.48 Desain Halaman Utama UMKM..... | 123 |
| Gambar 3.49 Desain Halaman Katalog Bahan Baku..... | 123 |
| Gambar 3.50 Desain Halaman Detail Bahan Baku UMKM..... | 124 |
| Gambar 3.51 Desain Halaman Keranjang UMKM..... | 124 |
| Gambar 3.52 Desain Halaman Transaksi Pembelian UMKM | 125 |
| Gambar 3.53 Desain Transaksi Penjualan UMKM..... | 125 |
| Gambar 3.54 Desain Halaman Rekomendasi UMKM..... | 126 |
| Gambar 4.1 Halaman Utama | 130 |
| Gambar 4.2 Halaman <i>Login</i> | 131 |
| Gambar 4.3 Halaman <i>Register Profil UMKM</i> | 132 |
| Gambar 4.4 Halaman <i>Register Struktur Organisasi UMKM</i> | 132 |
| Gambar 4.5 Halaman Utama UMKM..... | 133 |
| Gambar 4.6 Form Menambahkan <i>Timeline Portofolio</i> | 133 |
| Gambar 4.7 Pemberitahuan Hasil <i>Text Mining & NBC</i> | 134 |

| | |
|---|-----|
| Gambar 4.8 Rekomendasi Bahan Baku..... | 135 |
| Gambar 4.9 Pop Up Bahan Baku Tidak Ditemukan..... | 135 |
| Gambar 4.10 Halaman Katalog Bahan Baku..... | 136 |
| Gambar 4.11 Halaman Detail Bahan Baku..... | 137 |
| Gambar 4.12 Menu Navigasi Keranjang..... | 138 |
| Gambar 4.13 Tampilan Halaman Keranjang..... | 138 |
| Gambar 4.14 Tampilan Halaman Transaksi Pembelian..... | 139 |
| Gambar 4.15 Tampilan Halaman Detail Pembelian..... | 140 |
| Gambar 4.16 Tampilan Halaman Transaksi Penjualan..... | 140 |
| Gambar 4.17 Tampilan Halaman Detail Penjualan..... | 141 |
| Gambar 4.18 Tampilan Halaman Mengelola Bahan Baku | 141 |
| Gambar 4.19 Tampilan Halaman Tambah Bahan Baku Baru..... | 142 |
| Gambar 4.20 Tampilan Halaman Ubah Bahan Baku..... | 142 |
| Gambar 4.21 Halaman <i>Review</i> Bahan Baku | 143 |
| Gambar 4.22 Halaman <i>Login Admin</i> | 143 |
| Gambar 4.23 Halaman Mengelola Data <i>Bag Of Word</i> | 144 |
| Gambar 4.24 Tambah Data <i>Bag Of Word</i> | 145 |
| Gambar 4.25 Halaman Mengelola Kategori | 145 |
| Gambar 4.26 Halaman Validasi Pembayaran | 146 |

DAFTAR TABEL

| | Halaman |
|--|---------|
| Tabel 2.1 Kriteria UMKM Berdasarkan Aset dan Omset..... | 7 |
| Tabel 2.2 Proses <i>Tokenizing</i> | 8 |
| Tabel 2.3 Proses <i>Filtering</i> | 10 |
| Tabel 2.4 Proses <i>Stemming</i> | 10 |
| Tabel 2.5 Contoh Algoritma <i>Boyer-Moore</i> | 20 |
| Tabel 2.6 Contoh Algoritma <i>Boyer-Moore</i> | 21 |
| Tabel 3.1 Masalah dan alternatif solusi..... | 38 |
| Tabel 3.2 Analisis Kebutuhan Fungsi | 49 |
| Tabel 3.3 <i>Product Backlog</i> | 51 |
| Tabel 3.4 <i>Sprint Backlog</i> | 58 |
| Tabel 3.5 <i>Flow Of Event Login Admin</i> | 63 |
| Tabel 3.6 Desain <i>Test Case Login Admin</i> | 64 |
| Tabel 3.7 <i>Flow Of Event</i> Mengelola Data <i>Bag Of Word</i> | 65 |
| Tabel 3.8 Desain <i>Test Case Login Admin</i> | 68 |
| Tabel 3.9 <i>Flow Of Event</i> Mengelola Data Kategori..... | 69 |
| Tabel 3.10 Desain <i>Test Case</i> Mengelola Data Kategori | 72 |
| Tabel 3.11 <i>Flow Of Event</i> Validasi Pembayaran | 73 |
| Tabel 3.12 Desain <i>Test Case</i> Validasi Pembayaran..... | 76 |
| Tabel 3.13 <i>Flow Of Event Register</i> UMKM | 77 |
| Tabel 3.14 Desain <i>Test Case Register</i> UMKM..... | 79 |
| Tabel 3.15 <i>Flow Of Event Login</i> UMKM..... | 79 |
| Tabel 3.16 Desain <i>Test Case Login</i> UMKM..... | 81 |

| | |
|---|-----|
| Tabel 3.17 <i>Flow Of Event</i> Mengisi <i>Timeline Portofolio</i> | 82 |
| Tabel 3.18 Desain <i>Test Case</i> Mengisi <i>Timeline Portofolio</i> | 84 |
| Tabel 3.19 <i>Flow Of Event</i> Melihat Klasifikasi Status <i>Timeline Portofolio</i> | 85 |
| Tabel 3.20 Desain <i>Test Case</i> Melihat Klasifikasi Status <i>Timeline Potofolio</i> | 90 |
| Tabel 3.21 <i>Flow Of Event</i> Pembelian Bahan Baku..... | 91 |
| Tabel 3.22 Desain <i>Test Case</i> Pembelian Bahan Baku | 94 |
| Tabel 3.23 <i>Flow Of Event Review</i> Bahan Baku | 95 |
| Tabel 3.24 Desain <i>Test Case Review</i> Bahan Baku..... | 97 |
| Tabel 3.25 <i>Flow Of Event</i> Penjualan Bahan Baku..... | 98 |
| Tabel 3.26 Desain <i>Test Case</i> Penjualan Bahan Baku..... | 102 |
| Tabel 3.27 <i>Flow Of Event</i> Melihat Laporan..... | 103 |
| Tabel 3.28 Desain <i>Test Case</i> Melihat Laporan | 105 |
| Tabel 3.29 <i>Daily scrum</i> | 106 |
| Tabel 3.30 Kesesuaian <i>Story</i> Dengan Fitur Aplikasi | 110 |
| Tabel 3.31 Perubahan <i>Flow Of Event</i> Melihat Klasifikasi Status <i>Timeline Portofolio</i> | 113 |
| Tabel 3.32 Kesesuaian Fitur Dengan <i>Story</i> | 119 |
| Tabel 4.1 Tabel Kebutuhan <i>Software</i> Dan <i>Hardware</i> | 128 |
| Tabel 4.2 Evaluasi kesesuaian fitur dengan <i>story</i> | 146 |
| Tabel 4.3 <i>Testing</i> Fungsi..... | 149 |
| Tabel 4.4 Hasil <i>Testing</i> | 155 |

DAFTAR LAMPIRAN

| | Halaman |
|--|---------|
| Lampiran 1 <i>Daily Scrum</i> | 162 |
| Lampiran 2 Gambar Hasil <i>Testing</i> | 179 |
| Lampiran 3 Data Uji Klasifikasi Status <i>Timeline Portofolio</i> (Algoritma <i>Naïve Bayes</i>)..... | 199 |



BAB I

PENDAHULUAN

1.1. Latar Belakang

Usahanesia adalah sebuah gagasan Startup dibidang sosiopreneur yang didirikan pada awal tahun 2018, yang berawal dari keikutsertaan mahasiswa Institut Bisnis dan Informatika Stikom Surabaya dalam lomba dalam lomba yang diadakan di Yogyakarta. Tujuan dari startup Usahanesia adalah membantu mengembangkan UMKM se-Jawa Timur, mulai dari pembimbingan untuk mengatasi masalah mereka, meningkatkan *branding* UMKM, untuk mendapatkan *partner* bisnis (relasi), mempermudah dalam mendapatkan bahan baku produksi mereka, serta membantu UMKM untuk lebih mudah mendapatkan SDM. Salah satu cara untuk membantu UMKM berkembang adalah dengan membantu meningkatkan kualitas produk UMKM, untuk meningkatkan kualitas produk UMKM maka diperlukan adanya bahan baku yang berkualitas.

Berdasarkan rencana strategis Dinas Koperasi tahun 2014-2019 yang bertujuan untuk peningkatan akses Koperasi dan UMKM kepada sumber daya produktif terutama berkaitan dengan jangkauan dan jenis sumber pembiayaan yang sesuai dengan kebutuhan dan perkembangan usaha Koperasi dan UMKM. Akan tetapi masih terdapat beberapa masalah yang sering terjadi pada UMKM diantaranya adalah kesulitan untuk mendapatkan sumber daya produksi karena adanya persaingan yang ketat dalam mendapatkan bahan baku, bahan baku berkualitas rendah, dan tingginya harga bahan baku. Sehingga dapat menyebabkan

perkembangan UMKM tersebut terhambat dan tidak mendapatkan profit seperti tujuan bisnisnya. Menurut Bank Indonesia pada *Website* Katadata.co.id, UMKM menyerap 97% pekerja Indonesia. Pada data tersebut jenis usaha mikro merupakan usaha yang paling banyak menyerap tenaga kerja hingga 87 persen, sementara usaha besar hanya dapat menyerap sebanyak 3,3 persen. Berdasarkan hasil wawancara yang dilakukan kepada Dinas Koperasi UMKM di Banyuwangi yang memiliki UMKM binaan sebanyak 241 UMKM setuju bahwa internet dapat memenuhi kebutuhan bahan baku tiap UMKM. Namun hal tersebut belum dimanfaatkan sepenuhnya, lebih dari 50% penggiat UMKM masih belum memanfaatkan kegunaan positif dari internet untuk meningkatkan kinerja bisnisnya.

UMKM saat ini dalam mendapatkan bahan baku adalah dengan menghubungi penjual bahan baku yang telah mereka percayai walaupun kualitasnya kurang baik dan harganya yang mahal. Selain dalam mendapatkan bahan baku, cara UMKM mendapatkan relasi hanya melalui sosial media, melalui suatu *event*/pameran maupun melalui komunitas.

Berdasarkan uraian keadaan diatas, maka diperlukan sebuah inovasi baru untuk membantu UMKM dalam bertransaksi dan berinteraksi dengan UMKM lainnya dalam memenuhi kebutuhan akan bahan baku baik dalam proses produksi. Salah satunya adalah dengan membuat sebuah aplikasi yang dapat mencari kebutuhan bahan baku secara cepat dan tepat dengan menggunakan metode *Text Mining* berbasis *Crowdsourcing* yang dapat memberikan rekomendasi *supplier* yang dibutuhkan UMKM, maka dengan adanya aplikasi ini diharapkan Usahanesia

dapat membantu UMKM menjadi lebih mudah dan cepat dalam mencari kebutuhan bahan baku serta relasi dengan UMKM lainnya.

1.2. Rumusan Masalah

Berdasarkan latar belakang di atas, maka dapat dirumuskan kendala yang terdapat dihadapi saat ini adalah bagaimana merancang bangun aplikasi pemenuhan bahan baku produksi UMKM pada *Startup Usahanesia* dengan metode *Text Mining* berbasis *web*?

1.3. Batasan Masalah

Berdasarkan perumusan masalah di atas, adapun batasan masalah pada penelitian ini adalah.

1. Cakupan UMKM hanya untuk se-Jawa Timur.
2. Tidak terbatas pada bidang UMKM.
3. Bahasa yang digunakan adalah Bahasa Indonesia.
4. Metode pembayaran hanya melalui *transfer* antar bank
5. Tidak membahas metode pengiriman bahan baku

1.4. Tujuan

Tujuan dalam tugas akhir ini yaitu membangun aplikasi yang dapat memberikan rekomendasi bahan baku dan *supplier* yang sesuai dengan kebutuhan UMKM dengan menggunakan metode *Text Mining* dan *Crowdsourcing*.

1.5. Manfaat

Adapun manfaat yang didapatkan oleh para UMKM di Surabaya adalah sebagai berikut.

1. Dapat saling bertukar sumber daya material (bahan baku) yang dimiliki UMKM-nya dengan UMKN lain sehingga kebutuhan-kebutuhan dapat terpenuhi untuk meningkatkan kualitas UMKM-nya.

1.6. Sistematika Penulisan

Pembuatan laporan penelitian ini ditulis dengan sistematika penulisan sebagai berikut :

BAB I : PENDAHULUAN

Bab ini berisi tentang latar belakang, rumusan masalah, batasan masalah, tujuan dan manfaat dari penelitian yang akan dilakukan.

BAB II : LANDASAN TEORI

Bab ini menjelaskan tentang metode dan teori-teori yang digunakan untuk mengerjakan penelitian ini yaitu metode *text mining*, algoritma *naïve bayes classifier*, algoritma *boyer moore*, algoritma *jaro winkler* dan metode *graphc rating scale*.

BAB III : METODOLOGI PENELITIAN

Bab ini berisi tentang penjelasan tahap-tahap yang dikerjakan dalam menyelesaikan Tugas Akhir yang terdiri dari identifikasi masalah, *product backlog*, *sprint planning*, *sprint backlog*, *daily scrum* dan *sprint review*. Serta *usecase business diagram*, *activity diagram*, *usecase system diagram*, *flow of event*, *sequence diagram* dan *class diagram*

BAB IV : IMPLEMENTASI DAN EVALUASI SISTEM

Bab ini berisi implementasi dari aplikasi yang dibuat dan dilakukannya evaluasi atau pengujian aplikasi menggunakan pengujian *black box*.

BAB V : KESIMPULAN DAN SARAN

Bab ini berisi tentang kesimpulan dan saran. Kesimpulan menjelaskan tentang kesimpulan dari aplikasi yang telah dibuat dan hasil evaluasi yang telah dilakukan. Saran menjelaskan tentang saran dari kekurangan aplikasi yang dibuat.



BAB II

LANDASAN TEORI

2.1. Usaha Mikro Kecil dan Menengah (UMKM)

Dalam Undang-Undang No 20 Tahun 2008 tentang Usaha Mikro, Kecil dan Menengah dapat diartikan sebagai berikut. Usaha mikro adalah usaha produktif milik orang perorangan dan/atau badan usaha perseorangan yang memenuhi kriteria Usaha Mikro.

Usaha kecil adalah usaha ekonomi produktif yang berdiri sendiri, yang dilakukan oleh orang perorangan atau badan usaha yang bukan merupakan anak perusahaan atau bukan cabang perusahaan yang dimiliki, dikuasai, atau menjadi bagian baik langsung maupun tidak langsung dari usaha menengah atau usaha besar yang memenuhi kriteria usaha kecil.

Usaha menengah adalah usaha ekonomi produktif yang berdiri sendiri, yang dilakukan oleh orang perorangan atau badan usaha yang bukan merupakan anak perusahaan atau cabang perusahaan yang dimiliki, dikuasai, atau menjadi bagian baik langsung maupun tidak langsung dengan usaha kecil atau usaha besar dengan jumlah kekayaan bersih atau hasil penjualan tahunan.

Dikutip pada buku profil bisnis usaha mikro, kecil, dan menengah Lembaga Pengembangan Perbankan Indonesia & Bank Indonesia (2015) dapat dikelompokkan dalam tiga jenis, yaitu:

1. Usaha Mikro (jumlah karyawan 10 orang)
2. Usaha Kecil (jumlah karyawan 30 orang)
3. Usaha Menengah (jumlah karyawan 300 orang)

Tabel 2.1 Kriteria UMKM Berdasarkan Aset dan Omset

| Ukuran Usaha | Kriteria | |
|----------------|---------------------------|-------------------------|
| | Aset | Omset |
| Usaha Mikro | Maksiman Rp 50 juta | Maksimal Rp 300 Juta |
| Usaha Kecil | >Rp 50 juta – Rp 500 juta | >Rp 300 juta – Rp 2,5 M |
| Usaha Menengah | >Rp 500 juta – Rp 10 M | >Rp 2,5 M – Rp 50 M |

Berdasarkan ketentuan *Startup* Usahensia jangkauan target UMKM adalah UMKM se-Jawa Timur dan dengan tidak membatasi bidang usaha UMKM. Semua UMKM tersebut dapat mencari, membeli serta menjual bahan baku mereka dengan leluasa.

2.2. *Text Mining*

Menurut Kumar & Bhatia (2013) *Text Mining* didefinisikan sebagai ekstraksi non-trivial dari informasi yang tersembunyi, sebelumnya tidak diketahui, dan berpotensi berguna dari (sejumlah besar) data tekstual. *Text Mining* adalah bidang baru yang mencoba mengekstrak informasi yang bermakna dari teks bahasa alami. Ini dapat didefinisikan sebagai teks untuk mengekstrak informasi yang berguna untuk tujuan tertentu.

Text Mining mirip dengan *data mining*, kecuali bahwa alat *data mining* dirancang untuk menangani data terstruktur dari *database*, tetapi *text mining* juga dapat bekerja dengan kumpulan data terstruktur atau semi-terstruktur seperti email, dokumen teks dan *file* HTML, dll. Akibatnya, *text mining* adalah solusi yang jauh lebih baik. Tahapan-tahapan yang harus dilakukan pada proses *text mining* adalah:

1. *Text Preprocessing*

Tahap ini melakukan analisis semantik (kebenaran arti) dan sintaktik (kebenaran susunan) terhadap teks. Tujuan dari pemrosesan awal adalah untuk

mempersiapkan teks menjadi data yang akan mengalami pengelolaan lebih lanjut. Beberapa tahapan pada proses ini yaitu *tokenizing*, *filtering*, *stemming*, *tagging*, dan *analyzing*. Tahapan tersebut dijabarkan oleh Ratniasih, Sudarma, & Gunantara (2017) sebagai berikut:

a. *Tokenizing*

Tokenizing merupakan proses penguraian deskripsi yang semula berupa kalimat menjadi kata. Contoh proses *tokenizing* pada sebuah kalimat yang dapat dilihat pada Tabel 2.2.

Tabel 2.2 Proses *Tokenizing*

| Teks Input | Teks Output |
|--|---|
| pada saat ini umkm saya sedang membutuhkan bahan baku kulit sebanyak 50 meter. | pada saat ini umkm saya sedang membutuhkan bahan baku kulit sebanyak 50 meter |

Semua kata yang menyusun kalimat pada kolom “*Text Input*” dipotong berdasarkan kata yang menyusunnya seperti yang terlihat di kolom 2 “*Hasil Token*” pada Tabel 2.2.

b. *Case Folding*

Merubah sebuah semua karakter huruf pada sebuah kalimat menjadi huruf kecil dan menghilangkan karakter yang dianggap tidak *valid* seperti angka, tanda baca, dan *Uniform Resources Locator* (URL).

c. *Tagging*

Tagging merupakan tahap untuk mencari bentuk awal/*root* dari tiap kata lampau atau hasil dari proses *stemming*.

d. *Analysing*

Analyzing merupakan tahap penentuan seberapa jauh keterhubungan antar suatu kata atau *term* terhadap suatu dokumen atau kalimat dengan menghitung nilai/bobot keterhubungan.

2. ***Text Transformation (Attribute Generation)***

Transformasi teks atau pembentukan atribut mengacu pada proses untuk mendapatkan representasi dokumen yang diharapkan. Pendekatan representasi dokumen yang lazim digunakan adalah model —*bag of words* dan model ruang vektor (*vector space model*). Transformasi teks sekaligus juga melakukan pengubahan kata-kata ke bentuk dasarnya dan pengurangan dimensi kata di dalam dokumen. Tindakan ini diwujudkan dengan menerapkan *stemming* dan menghapus *stopwords*.

a. *Filtering*

Filtering adalah tahap mengambil kata dari hasil proses token. Bisa menggunakan algoritma *stop list* atau *word list* yang didapatkan pada jurnal tentang *stop word* bahasa Indonesia dan penambahan istilah yang terdapat pada UMKM. *Filtering* dapat juga diartikan sebagai proses mengambil kata-kata penting dari hasil proses *token* atau penghapusan *stopwords*. *Stopwords* merupakan kosa kata yang bukan merupakan ciri (kata unik) dari suatu dokumen berbasis Indonesia

secara umum menurut Tala dan beberapa istilah UMKM yang terdapat pada sumber menurut Merina (2016). Beberapa kata yang dianggap tidak penting disimpan pada *database* yang nantinya dicocokkan kepada kata yang ada. Untuk contoh tahap *filtering* terlihat pada Tabel 2.3.

Tabel 2.3 Proses *Filtering*

| Hasil <i>Tokenizing</i> | Hasil <i>Filtering</i> |
|---|---|
| Pada saat ini umkm saya sedang membutuhkan bahan baku kulit sebanyak 50 meter | Umkm membutuhkan kulit sebanyak 50 meter |

b. *Stemming*

Stemming merupakan tahap untuk mencari *root* kata dari hasil *filtering*. *Stemming* adalah proses pemetaan dan penguraian berbagai bentuk (*variants*) dari suatu kata menjadi bentuk kata dasarnya. Seperti pada Tabel 2.4 berikut.

Tabel 2.4 Proses *Stemming*

| Hasil <i>Filtering</i> | Hasil <i>Stemming</i> |
|---|---|
| umkm membutuhkan kulit sebanyak 50 meter | umkm butuh kulit banyak 50 meter |

Menurut Kurniawan, Effendi, & Sitompul (2012) pada proses ini dapat menggunakan algoritma *confix-stripping stemmer* yaitu algoritma yang digunakan

untuk melakukan proses *stemming* terhadap kata-kata berimbuhan. Algoritma *confix-stripping stemmer* mempunyai aturan imbuhan sendiri dengan model sebagai berikut:

$$[[[AW+]AW+]AW+] \text{ Kata Dasar } [[+AK][+KK][+P]] \quad [1]$$

Keterangan:

AW : Awalan

KK : Kata ganti kepunyaan

AK : Akhiran

P : Partikel

Langkah-langkah algoritma *confix-stripping stemmer* adalah sebagai berikut:

- a) Kata yang belum di-*stemming* dibandingkan ke dalam *database* kamus kata dasar. Jika bertemu, maka kata tersebut diasumsikan sebagai kata dasar dan algoritma berhenti. Jika tidak sesuai dengan kata dalam kamus, lanjut ke langkah a.
- b) Jika kata di masukkan memiliki pasangan awalan akhiran “be-lah”, “be-an”, “me-i”, “di-i”, “pe-i”, atau “te-i” maka langkah *stemming* selanjutnya adalah e,c,d,e,f tetapi jika kata yang di masukkan tidak memiliki pasangan awalan-akhiran tersebut, langkah *stemming* berjalan normal c,d,e,f
- c) Hilangkan partikel dan kata ganti kepunyaan. Pertama hilangkan partikel (“-lah”, “-kah”, “-tah”, “-pun”). Setelah itu hilangkan juga kata ganti kepunyaan (“-ku”, “-mu”, atau “-nya”). Contoh : kata “bajumlah”, proses *stemming* pertama menjadi “bajumu” dan proses *stemming* kedua menjadi “baju”. Jika

kata “baju” ada di dalam kamus maka algoritma berhenti. Sesuai dengan model imbuhan, menjadi :

$$[[[AW+]AW+]AW+] \text{ Kata Dasar } [+AK] \quad [2]$$

- d) Hilangkan juga akhiran (“-i”, “-an”, dan “-kan”), sesuai dengan model imbuhan, maka terjadi

$$[[[AW+]AW+]AW+] \text{ Kata Dasar} \quad [3]$$

Contoh kata “membelian” di *stemming* menjadi ”membeli”, jika tidak ada dalam *database* kata dasar maka dilakukan proses penghilangan awalan.

- e) Penghilangan awalan (“be-”, “di-”, “ke-”, “me-”, “pe-”, “se-”, dan “te-”).
f) Jika semua langkah gagal, maka kata yang diuji pada algoritma ini dianggap sebagai kata dasar

3. ***Feature Selection (Attribute Selection)***

Feature Selection dikenal sebagai *Attribute Selection* adalah proses pemilihan subset fitur penting untuk digunakan dalam pembuatan model. Asumsi utama ketika menggunakan teknik pemilihan fitur adalah bahwa data mengandung banyak kata yang tidak relevan.

4. ***Pattern Discovery***

Pattern discovery merupakan tahap penting untuk menemukan pola atau pengetahuan (*knowledge*) dari keseluruhan teks. Tindakan yang lazim dilakukan pada tahap ini adalah operasi teks *mining*, dan biasanya menggunakan teknik-teknik *data mining*. Dalam penemuan pola ini, proses teks *mining* dikombinasikan dengan proses-proses *data mining*.

Pada tugas akhir ini metode *text mining* yang akan digunakan pada status yang terdapat *timeline portofolio* untuk dianalisa, sehingga mendapatkan bahan

baku yang dibutuhkan atau dimiliki oleh UMKM. Hasil dari proses *text mining* ini nantinya akan digunakan kata kunci (*keyword*) pada proses *string matching*. Pada pembuatan proses *text mining* pada aplikasi menggunakan *plug in sastrawi*.

2.3. Plug In Text Mining Sastrawi

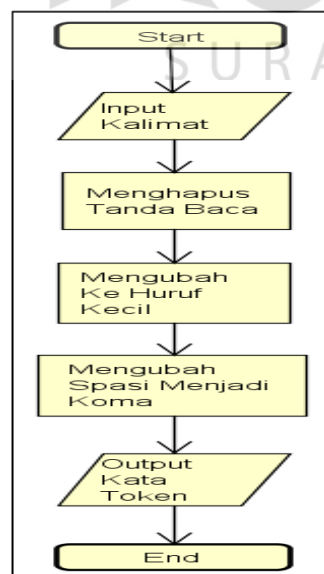
2.3.1. Plug In Tokenizer Sastrawi

Sastrawi Tokenizer adalah *library PHP* untuk melakukan *tokenization* pada Bahasa Indonesia.

Cara kerja dari *Plug In Tokenizer Sastrawi* adalah seperti berikut.

1. Teks yang akan di-*tokenizing* "Saya sedang belajar NLP Bahasa Indonesia."
2. Dilakukan *tokenizing* dengan *Plug In Tokenizer Sastrawi*.
3. Hasil *tokenizing* ["Saya", "sedang", "belajar", "NLP", "Bahasa", "Indonesia", "."].

Untuk proses *tokenizing* menggunakan *plug in tokenizer sastrawi* lebih jelas dapat dilihat pada Gambar 2.1.



Gambar 2.1 Flowchart Plug In Tokenizer

2.3.2. Plug In Filtering & Stemming Sastrawi

Sastrawi Stemming adalah *library PHP* sederhana yang digunakan untuk dapat mengubah kata yang telah berimbuhan dalam Bahasa Indonesia menjadi kata asalnya (*stem*). Meskipun sederhana, *library* ini dirancang agar berkualitas tinggi dan didokumentasikan dengan baik.

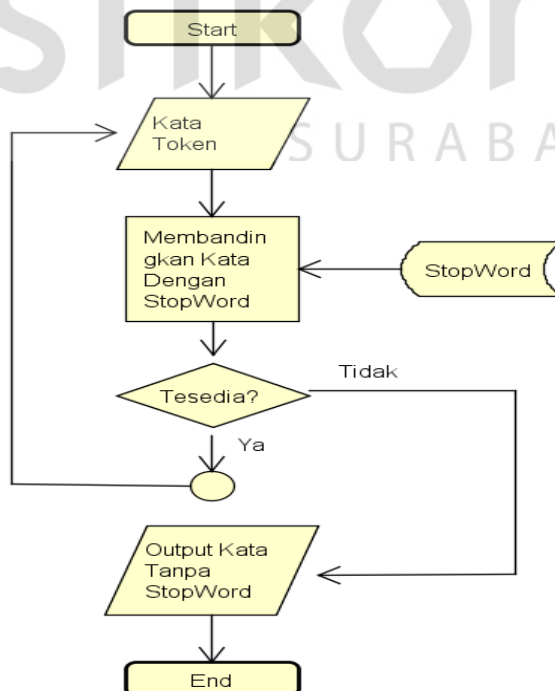
Stemming adalah proses mengubah kata berimbuhan menjadi kata dasar.

Contohnya:

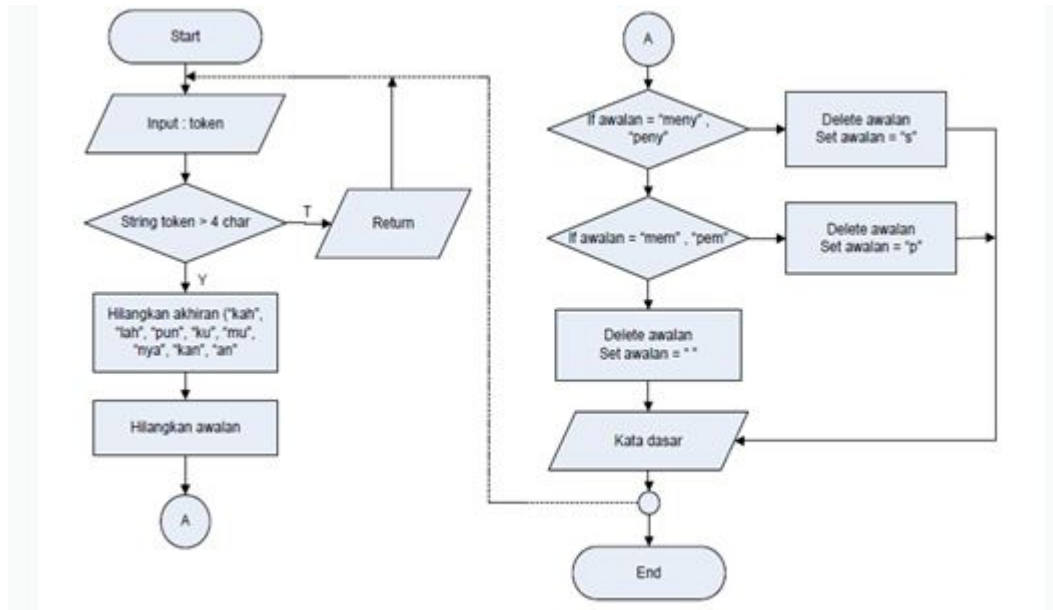
- menahan => tahan
- berbalas-balasan => balas

Proses *stemming* oleh *Sastrawi* sangat bergantung pada kamus kata dasar. *Sastrawi* menggunakan kamus kata dasar dari *kateglo.com* dengan sedikit perubahan.

Untuk proses *tokenizing* menggunakan *plug in filtering & stemming sastrawi* lebih jelas dapat dilihat pada Gambar 2.2 dan Gambar 2.3.



Gambar 2.2 Flowchart Plug In Filtering



Gambar 2.3 Flowchart Plug In Stemming

2.4. Algoritma Naïve Bayes Classifier

Menurut Kurniawan, Effendi, & Sitompul (2012) algoritma *naïve bayes classifier* merupakan algoritma yang digunakan untuk mencari nilai probabilitas tertinggi untuk mengklasifikasi data uji pada kategori yang paling tepat. Prediksi *bayes* didasarkan pada teorema *Bayes* dengan formula umum sebagai berikut:

Dalam algoritma *naïve bayes classifier* setiap dokumen direpresentasikan dengan pasangan atribut “ $x_1, x_2, x_3, \dots, x_n$ ” dimana x_1 adalah kata pertama, x_2 adalah kata kedua dan seterusnya. Sedangkan V adalah himpunan kategori berita. Pada saat klasifikasi algoritma akan mencari probabilitas tertinggi dari semua kategori dokumen yang diujikan (V_{MAP}), dimana persamaannya adalah sebagai berikut :

$$V_{MAP} = \underset{V_j \in V}{\arg \max} \left(\frac{P(x_1, x_2, x_3, \dots, x_n | V_j) P(V_j)}{P(x_1, x_2, x_3, \dots, x_n)} \right) \quad [4]$$

Untuk $P(x_1, x_2, x_3, \dots, x_n)$ nilainya konstan untuk semua kategori (V_j) sehingga persamaan dapat ditulis sebagai berikut :

$$V_{MAP} = \arg \max_{V_j \in V} (P(x_1, x_2, x_3, \dots, x_n | V_j) P(V_j)) \quad [5]$$

Persamaan diatas dapat disederhanakan menjadi sebagai berikut :

$$V_{MAP} = \arg \max_{V_j \in V} \prod_{i=1}^n (P(x_i | V_j) P(V_j)) \quad [6]$$

Dimana penjelasan formula tersebut adalah sebagai berikut:

V_j = Kategori berita $j=1, 2, 3, \dots, n$. Dimana dalam penelitian ini j_1 = kategori butuh, j_2 = kategori punya, dan j_3 = kategori tidak sama sekali.

$P(x_i | V_j)$ = Probabilitas x_i pada kategori V_j .

$P(V_j)$ = Probabilitas dari V_j .

Untuk $P(V_j)$ dan $P(x_i | V_j)$ dihitung pada saat pelatihan dimana persamaannya adalah sebagai berikut :

$$P(V_j) = \frac{|docs\ j|}{|contoh|} \quad [7]$$

$$P(X_i | V_j) = \frac{n_k + 1}{n + |kosakata|} \quad [8]$$

Keterangan :

$|doc\ j|$: Jumlah dokumen setiap kategori

$|contoh|$: Jumlah dokumen dari semua kategori

n_k : Jumlah frekuensi kemunculan setiap kata

n : Jumlah frekuensi kemunculan kata dari setiap kategori

$|kosakata|$: jumlah semua kata dari semua kategori

Pada penelitian ini penggunaan algoritma *naïve bayes classifier* digunakan untuk mengklasifikasi apakah UMKM saat ini sedang membutuhkan bahan baku ataupun sedang kelebihan bahan baku. Pada pembuatan proses *naïve bayes classifier* pada aplikasi menggunakan *API naïve bayes classifier MonkeyLearn*.

2.5. *API Naïve Bayes Classifier MonkeyLearn*

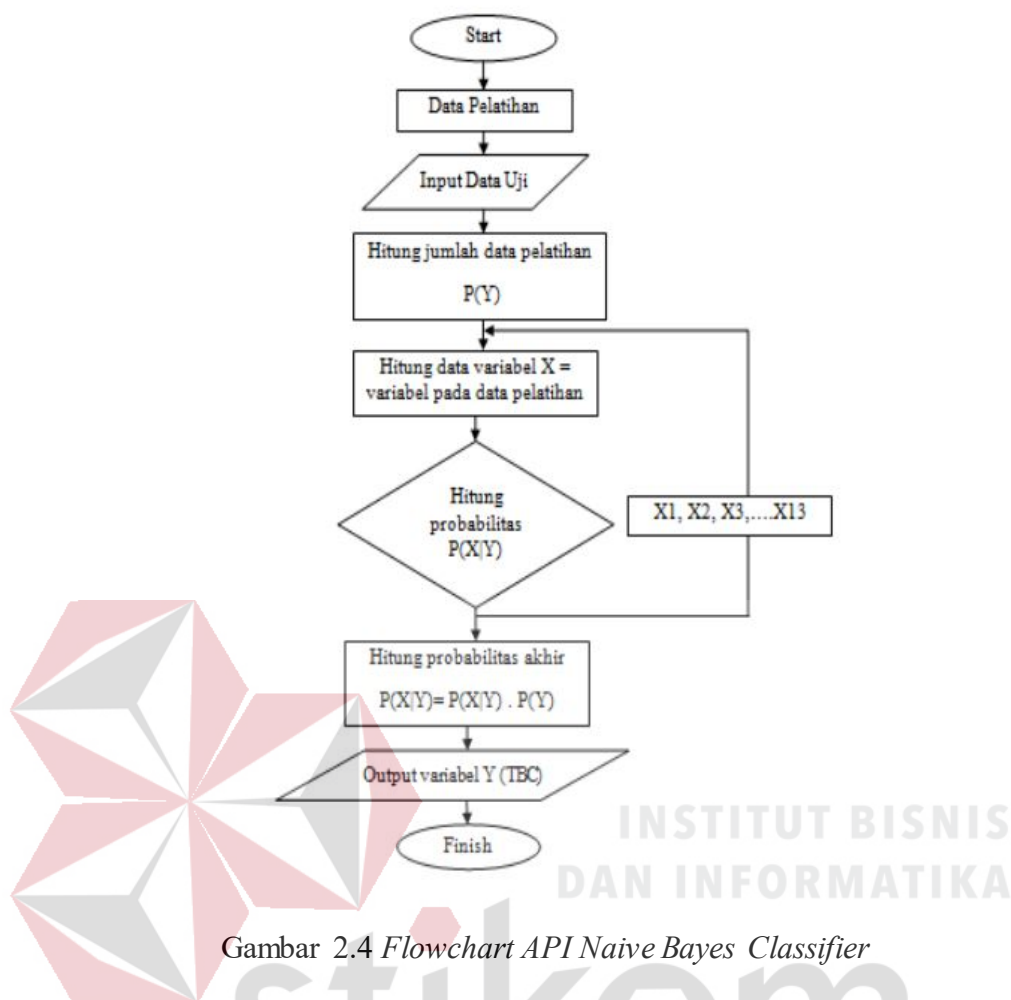
MonkeyLearn adalah *platform Machine Learning* untuk *Text Analysis*. Ini memungkinkan pengguna untuk dengan mudah mendapatkan data yang dapat ditindaklanjuti dari teks mentah. Misalnya, Anda dapat mendeteksi topik atau sentimen yang diungkapkan dalam teks seperti *tweet*, obrolan, ulasan, artikel, dan lainnya.

Model klasifikasi teks digunakan untuk mengelompokkan teks ke dalam grup yang terorganisir. Teks dianalisis oleh model dan kemudian tag yang sesuai diterapkan berdasarkan konten. Model pembelajaran mesin yang dapat secara otomatis menerapkan *tag* untuk klasifikasi dikenal sebagai *Classifiers*.

Classifiers tidak bisa hanya bekerja secara otomatis, mereka perlu dilatih untuk dapat membuat prediksi spesifik untuk teks. Pelatihan classifier dilakukan dengan:

- Mendefinisikan serangkaian *tag* yang akan digunakan oleh model.
- Membuat asosiasi antara potongan teks dan *tag* yang sesuai.

Untuk proses *tokenizing* menggunakan *API naïve bayes classifier Monkey Learn* lebih jelas dapat dilihat pada Gambar 2.4.



Gambar 2.4 Flowchart API Naive Bayes Classifier

2.6. String Matching (Pencocokan String/Kata)

Menurut Syarif (2017) *String* dalam ilmu komputer dapat diartikan dengan sekuens dari karakter. Walaupun sering juga dianggap sebagai data abstrak yang menyimpan sekuens nilai data, atau biasanya berupa *bytes* yang mana merupakan elemen yang digunakan sebagai pembentuk karakter sesuai dengan *encoding* karakter yang disepakati seperti *ASCII*, ataupun *EBCDIC*. Algoritma string matching adalah suatu metode yang digunakan untuk menemukan suatu keakuratan atau hasil dari satu atau beberapa pola teks yang diberikan. *String matching* merupakan pokok bahasan yang penting dalam ilmu komputer karena teks

merupakan bentuk utama dari pertukaran informasi antar manusia, misalnya pada literatur, karya ilmiah, halaman web dan sebagainya. Pencocokan *string* juga dapat digunakan untuk mencari pola *bit* dalam sejumlah besar *file binary*. dalam algoritma *string matching*, teks diasumsikan berada di dalam *memory*, sehingga bila kita mencari string di dalam sebuah teks, maka semua isi teks perlu dibaca terlebih dahulu kemudian disimpan didalam *memory*. *String matching* fokus pada pencarian satu, atau lebih umum, semua kehadiran sebuah kata (lebih umum disebut *pattern*) dalam sebuah teks. Semua algoritma yang akan dibahas mengeluarkan semua kehadiran pola dalam teks. Pola dinotasikan sebagai $x = x[0..m-1]$; m adalah panjangnya.

Pada tugas akhir ini metode *string matching* yang digunakan ialah dengan menggunakan algoritma *Boyer-Moore* dikarenakan Menurut Agung & Yogyakarta (2016) Algoritma *Boyer-Moore* adalah algoritma *string matching* yang paling efisien dibandingkan algoritma *string matching* lainnya.

2.7. Algoritma *Boyer-Moore*

Menurut Agung & Yogyakarta (2016) Algoritma *Boyer Moore* dianggap sebagai algoritma pencocokan *string* yang paling efisien digunakan dalam berbagai aplikasi sehari-hari. Algoritma tersebut dikembangkan oleh Bob Boyer dan J. Stroher Moore pada tahun 1977.

Pada proses pencarian *string* algoritma *BoyerMoore* membaca karakter-karakter dari pola (*pattern*) dari kanan ke kiri. Dalam kasus dimana jumlah karakter pada *pattern* lebih sedikit daripada jumlah karakter pada teks maka algoritma tersebut menggunakan 2 buah fungsi *precomputed*. Dua buah fungsi pengubah ini

disebut *good-suffix shift*. Aturan pada *good-suffix shift* bertujuan untuk menangani kasus dimana terdapat pengulangan karakter pada *pattern*.

Algoritma *Boyer-Moore* mempunyai empat konsep dasar di dalam proses pencarian *string*, yaitu :

1. *Preprocessing*
2. *Right-to-left scan*
3. *Bad-character Rule*
4. *Good-suffix Rule*

Precomputation dari algoritma *Boyer-Moore* terdiri dari *bad-character preprocessing* dan *goodsuffix preprocessing*. Prinsip dasar yang pertama dari algoritma *Boyer-Moore* adalah melakukan perbandingan antara *pattern* yang dicari dengan teks. Perbandingan *pattern* dengan teks dilakukan dari arah kanan ke kiri.

Perbandingan dimulai dengan membandingkan antara karakter paling kanan dari *pattern* dengan teks. Jika terjadi kecocokkan, maka perbandingan akan dilanjutkan dengan karakter yang disebelah kiri dari yang dibandingkan sampai ke karakter pertama dari *pattern*. Jika terjadi ketidakcocokkan maka akan dilakukan pergeseran yang ditentukan oleh 2 fungsi pergeseran yaitu *bad-character shift* dan *good-suffix shift*. Aturan dari *bad-character shift* dibutuhkan untuk menghindari pengulangan perbandingan yang gagal dari suatu karakter dalam teks dengan *pattern*. Aturan dari *good-suffix shift* dibutuhkan untuk menangani kasus yang di dalamnya terdapat pengulangan karakter pada *pattern*.

Tabel 2.5 Contoh Algoritma *Boyer-Moore*

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| K | A | I | N | | K | U | L | I | T |
| K | U | L | I | T | | | | | |

Pada contoh di atas, dengan melakukan perbandingan dari posisi paling akhir *string* dapat dilihat bahwa karakter ‘N’ pada *string* “kain” tidak cocok dengan karakter ‘I’ pada *string* “KULIT” yang dicari, dan karakter ‘N’ tidak pernah ada dalam *string* “KULIT” yang dicari sehingga *string* “KULIT” dapat digeser melewati *string* “KAIN”, sehingga posisinya seperti berikut.

Tabel 2.6 Contoh Algoritma *Boyer-Moore*

| | | | | | | | | | |
|---|---|---|---|--|---|---|---|---|---|
| K | A | I | N | | K | U | L | I | T |
| | | | | | K | U | L | I | T |

Dalam contoh terlihat bahwa algoritma *Boyer-Moore* memiliki loncatan karakter yang besar sehingga mempercepat pencarian *string* karena dengan hanya memeriksa sedikit karakter, dapat langsung diketahui bahwa *string* yang dicari tidak ditemukan dan dapat digeser ke posisi berikutnya.

Secara sistematis, langkah-langkah yang dilakukan algoritma *Boyer-Moore* pada saat mencocokkan *string* adalah :

1. Buat tabel pergeseran *string* yang dicari (S) dengan pendekatan *Match Heuristic* (MH) dan *Occurence Heuristic* (OH), untuk menentukan jumlah pergeseran yang akan dilakukan jika mendapat karakter tidak cocok pada proses pencocokkan dengan *string* (T).
2. Jika dalam proses perbandingan terjadi ketidakcocokkan antara pasangan karakter pada S dan karakter pada T, pergeseran dilakukan dengan memilih salah satu nilai pergeseran dari dua tabel analisa *string* yang memiliki nilai pergeseran paling besar.
3. Dua kemungkinan penyelesaian dalam melakukan pergeseran S, jika sebelumnya belum ada karakter yang cocok adalah dengan melihat nilai pergeseran hanya pada tabel *Occurence Heuristic*, jika karakter yang tidak

cocok tiak ada pada S, maka pergeseran adalah sebanyak jumlah karakter pada S, dan jika karakter yang tidak cocok ada pada S, maka banyaknya pergeseran bergantung pada nilai tabel.

4. Jika karakter pada teks yang sedang dibandingkan cocok dengan karakter pada S, maka posisi karakter pada S dan T diturunkan sebanyak 1 posisi, kemudian dilanjutkan dengan pencocokkan pada posisi tersebut dan seterusnya. Jika kemudian terjadi ketidakcocokkan karakter S dan T, maka dipilih nilai pergeseran terbesar dari dua tabel analisis *pattern*, yaitu nilai dari tabel *Match Heuristic* dan tabel *Occurence Heuristic* dikurangi dengan jumlah karakter yang telah cocok.
5. Jika semua karakter telah cocok, artinya S telah ditemukan di dalam T, selanjutnya geser *pattern* sebanyak 1 karakter.
6. Lanjutkan sampai akhir *string* T.

Kelebihan dari algoritma *Boyer-Moore* ini semakin panjang pola yang dicari maka waktu pencarian semakin singkat. Sedangkan kekurangan algoritma *Boyer-Moore* adalah lebih lambat untuk *pattern* yang pendek dan tidak bagus untuk pencarian *binary string*.

Pada tugas akhir ini Algoritma *Boyer-Moore* digunakan untuk mencari kesamaan *keyword* yang didapat dari hasil *text mining* dengan produk bahan baku yang dijual oleh UMKM lain.

2.8. Algoritma *Jaro-Winkler*

Menurut Okta'mal, Saptono, & Sulistyo (2015), *Jaro-Winkler distance* adalah merupakan varian dari *Jaro distance metrik* yaitu sebuah algoritma untuk

mengukur kesamaan antara dua *string*, biasanya algoritma ini digunakan di dalam pendeteksian duplikat. Semakin tinggi *Jaro-Winkler distance* untuk dua *string*, semakin mirip dengan *string* tersebut. *Jaro-Winkler distance* terbaik dan cocok untuk digunakan dalam perbandingan *string* singkat seperti nama orang. Skor normalnya seperti 0 menandakan tidak ada kesamaan, dan 1 adalah sama persis.

Algoritma *Jaro-Winkler distance* memiliki kompleksitas waktu quadratic runtime complexity yang sangat efektif pada *string* pendek dan dapat bekerja lebih cepat dari algoritma edit distance. Dasar dari algoritma ini memiliki tiga bagian:

1. Menghitung panjang *string*.
2. Menemukan jumlah karakter yang sama di dalam dua *string*
3. Menemukan jumlah transposisi

Pada algoritma ini menggunakan rumus untuk menghitung jarak (d_j) antara dua *string* yaitu s_1 dan s_2 adalah:

$$d_j = \frac{1}{3} \times \left(\frac{m}{|s_1|} + \frac{m}{|s_2|} + \frac{m - t}{m} \right) \quad [9]$$

Dimana :

m = jumlah karakter yang sama persis

$|s_1|$ = panjang *string* 1

$|s_2|$ = panjang *string* 2

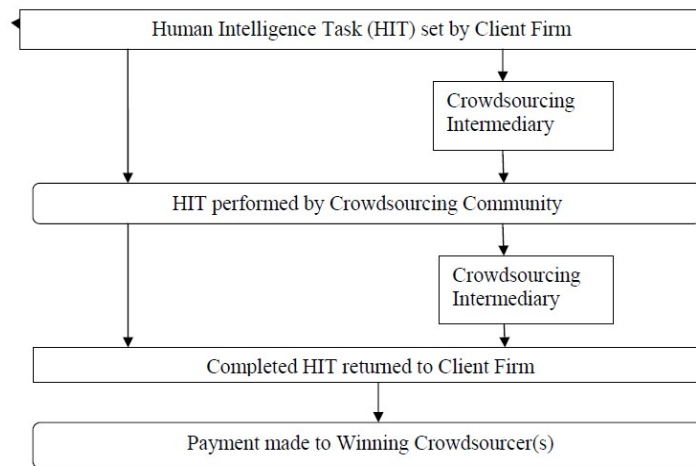
t = jumlah transposisi

Fungsi dari algoritma ini adalah memaksimalkan kinerja sistem apabila algoritma *Boyer-Moore* tidak dapat melakukan proses *string matching* apabila terjadi kesalahan pengetikan terhadap pengguna yaitu menutupi kekurangan fungsi algoritma *Boyer-Moore* dengan adanya *similarity* yang terdapat pada algoritma

Jaro-Winkler sehingga sistem dapat memberikan pembenaran apabila terdapat kesalahan pengetikan.

2.9. Crowdsourcing

Menurut Paul Whitla (2009), *Crowdsourcing* adalah proses menyelesaikan pekerjaan atau masalah dengan bantuan sekelompok orang, yaitu orang-orang yang menggunakan internet. Ide dasar di balik *Crowdsourcing* adalah untuk mengambil pekerjaan dan memberikannya kepada sekelompok pekerja di mana setiap anggota kelompok menyelesaikan pekerjaan dan mendapatkan bayaran dari usahanya. Keuntungan utama menggunakan *crowdsourcing* adalah perusahaan memberikan tugasnya kepada berbagai orang terampil yang memiliki kemampuan untuk menyelesaikan tugas dalam rentang waktu yang singkat dan dengan biaya yang jauh lebih rendah dibandingkan dengan pekerjaan yang dilakukan oleh karyawannya sendiri. Perusahaan-perusahaan kecil banyak yang menggunakan metode *crowdsourcing* karena perusahaan-perusahaan tersebut sering bekerja dengan sumber daya terbatas di dalam perusahaan dan memiliki anggaran terbatas untuk dibelanjakan pada biro iklan, perusahaan hubungan masyarakat, desainer grafis, fotografer atau apa pun. Perusahaan-perusahaan kecil ini dapat memperoleh banyak manfaat dari akses ke sejumlah besar keterampilan yang tersedia dengan harga terjangkau melalui *crowdsourcing*.



Gambar 2.5 Proses *Crowdsourcing*. (Whitla, 2009)

Berdasarkan pada Gambar 2.5 proses *crowdsourcing* memiliki 4 tahap inti yaitu :

1. *Human Intelligence Task (HIT) set by Client Firm*

Pada tahap ini proses *Crowdsourcing* berawal dari suatu perusahaan yang memberikan beberapa tugas kepada suatu komunitas. Hal itu bertujuan agar tugas-tugas perusahaan tersebut dapat terselesaikan dengan cepat daripada dikerjakan oleh perusahaan itu sendiri.

2. *HIT performed by Crowdsourcing Community*

Pemberian tugas kepada suatu komunitas jika tertarik untuk mengerjakan tugas tersebut maka, tugas tersebut secara langsung diberikan kepada komunitas yang mempunyai keahlian.

3. *Completed HIT returned to Client Firm*

Setelah pengerjaan tugas dari komunitas terselesaikan, tugas tersebut akan dikirimkan kepada perusahaan dan akan dilihat apakah telah sesuai dengan yang diinginkan perusahaan.

4. *Payment made to Winning Crowdsourcer(s)*

Jika tugas tersebut telah sesuai dengan apa yang diinginkan perusahaan, maka perusahaan akan memberikan pembayaran atau hadiah kepada komunitas tersebut.

Sedangkan kekurangan dari *Crowdsourcing* adalah terkadang *crowd* dapat mengembalikan sejumlah besar solusi yang mungkin memiliki sedikit/kurang relevansi. *Proses crowdsourcing* yang ada pada *startup* Usahanesia terdapat pada banyaknya UMKM yang akan membagikan informasi mengenai UMKMnya melalui *Timeline Portofolio*. Pengambilan informasi tersebut dianalisis dan dikelola, yang pada akhirnya terdapat keluaran informasi berupa bahan baku apa yang saat ini dibutuhkan oleh UMKM. Informasi kebutuhan tersebut diteruskan untuk membentuk menginisiasi urundaya terhadap UMKM lain yang memiliki sumber daya bahan baku lebih terhadap UMKM yang memiliki kebutuhan sumber daya bahan baku.

2.10. *Graphic Rating Scales (GRS)*

Menurut Sulistyawan, Wurijanto, & Subiantoro (2013) Metode GRS merupakan salah satu metode yang digunakan dalam melakukan penilaian kinerja karyawan. Metode ini banyak digunakan oleh perusahaan karena metode ini mudah dikembangkan dan mudah untuk dimodifikasi jika diperlukan adanya perubahan terhadap kriteria-kriteria yang menjadi bahan penilaian.

Skala penilaian grafik memberikan penilaian yang khas. Disitu didaftarkan ciri-ciri (seperti mutu dan kehandalan) serta kisaran nilai kinerja (dari yang tidak memuaskan sampai yang luar biasa memuaskan) untuk masing-masing bawahan

dengan melingkari atau memeriksa skor yang paling baik menggambarkan kinerjanya untuk masing-masing ciri.

Terdapat beberapa alasan mengapa metode ini banyak dipakai secara luas, yaitu:

1. Skala penilaian grafik mudah digunakan. Penyelia dapat menilai banyak individu dalam waktu singkat. Skala-skala ini juga mudah dipahami dan dijelaskan kepada orang-orang yang dinilai.
2. Metode ini juga mudah dibuat dan dimodifikasi jika dibutuhkan. Skala penilaian grafik membandingkan kinerja individu terhadap sebuah standar absolut. Penilaian mengevaluasi kinerja berbagai dimensi atau kriteria berbagai dimensi atau kriteria, seperti kualitas kerja, penerimaan kritik, kemauan memikul tanggung jawab dan hal-hal yang serupa lainnya. Penilai menggunakan skala berupa angka-angka mulai dari rendah sampai tinggi, dari yang dinilai jelek sampai ke nilai yang baik sekali. Atau dari kriteria yang tidak memuaskan sampai ke kinerja yang sangat bagus.

Penggunaan metode *Graphic Rating Scales* tersebut digunakan untuk melakukan pemeringkatan terhadap beberapa UMKM penyedia (*supplier*) berdasarkan hasil dari *matching* yang telah dilakukan sebelumnya yang nantinya menjadi rekomendasi yang dapat dipilih oleh pihak UMKM pencari. Beberapa penentuan kriteria yang ditetapkan ialah harga bahan baku, jumlah stok, dan kualitas (rating) bahan baku.

2.11. *Agile Development*

Menurut Appelo (2011) pengembangan perangkat lunak *agile* adalah pendekatan pengembangan perangkat lunak yang berasal dari tahun 1990-an. Pendekatan ini merupakan respon terhadap metode pengembangan birokrasi dan *ad-hoc* yang tidak dapat menyampaikan produk *software* secara konsisten. Pengembangan perangkat lunak *agile*, dengan nilai dan prinsip yang tersampai di *Agile Manifesto*, memiliki fokus terhadap orang dan tim, frekuensi penyampaian rilis yang tinggi, kolaborasi pelanggan yang tinggi dan merespon perubahan dengan minimalnya rencana awal. Nilai dan prinsip *agile* telah diimplementasikan melalui beberapa metode variasi, seperti *Scrum* dan *Extreme Programming*.

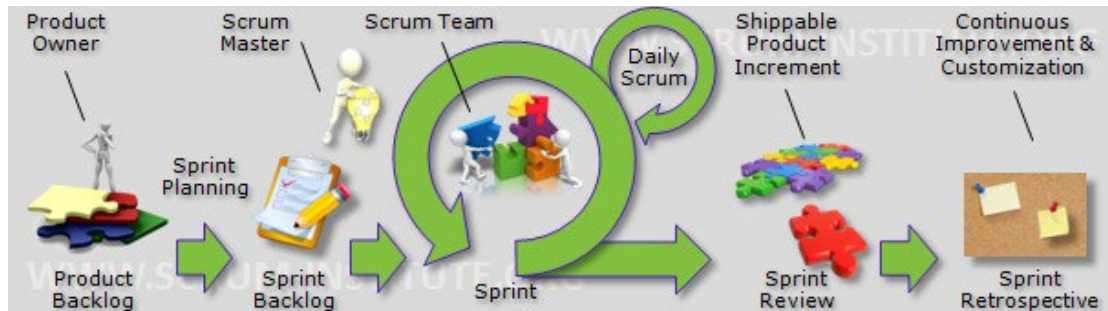
Metode pengembangan perangkat lunak *Agile* sangat tepat untuk diterapkan dalam pengembangan aplikasi *startup* Usahanesia tepatnya untuk aplikasi pemenuhan bahan baku produksi UMKM, karena dalam tahap pengembangannya adalah secara tim, dan berfokus pada UMKM sebagai pengguna sehingga akan sering terjadi perubahan secara terus menerus.

2.12. *Scrum*

Menurut Schwaber & Sutherland (2014) *scrum* adalah sebuah kerangka kerja dimana orang-orang dapat menyelesaikan permasalahan kompleks yang senantiasa berubah, dimana pada saat bersamaan menghasilkan produk dengan nilai setinggi mungkin secara kreatif dan produktif.

Scrum didasari oleh teori kontrol proses empiris, atau dengan kata lain, empirisme. Empirisme menekankan bahwa pengetahuan berasal dari pengalaman dan pembuatan keputusan didasari oleh pengetahuan yang telah dimiliki hingga saat

ini. *Scrum* menggunakan pendekatan berkala (*iterative*) dan bertahap (*incremental*) untuk meningkatkan prediktabilitas dan mengendalikan resiko. Gambar 2.6 menunjukkan tahapan-tahapan *scrum*.



Gambar 2.6 Tahapan *Scrum Framework*. (International Scrum Institute, n.d.)

Menurut Schwaber & Sutherland (2014) *scrum* memiliki 4 acara yang harus dijalankan diantaranya:

1. *Sprint planning*

Sebelum melakukan *sprint planning*, ada yang namanya *product backlog*, yang dimana *product backlog* adalah membuat *list* prioritas pekerjaan. Pekerjaan yang akan dilaksanakan di dalam *sprint* direncanakan pada saat *sprint planning*. Perencanaan ini dibuat secara kolaboratif oleh seluruh anggota tim *scrum*. Pada saat melakukan *sprint planning* harus dapat menjawab pertanyaan-pertanyaan diantaranya apa *goal* dari *sprint*, apa yang dapat dihantarkan di dalam *inkremen* sebagai hasil dari *sprint* yang sedang berjalan, apa yang perlu dilakukan untuk dapat menghantarkan *inkremen* tersebut. Setelah melakukan *sprint planning* ada yang namanya *sprint backlog*, *sprint backlog* sendiri adalah sekumpulan *item product backlog* yang telah dipilih untuk dikerjakan di *sprint*, juga di dalamnya rencana untuk mengembangkan potongan tambahan produk dan merealisasikan *sprint goal*. *Sprint backlog* adalah perkiraan mengenai fungsionalitas apa yang akan tersedia di

inkremen selanjutnya dan pekerjaan yang perlu dikerjakan untuk menghantarkan fungsionalitas tersebut menjadi potongan tambahan produk yang “Selesai”.

2. *Daily scrum*

Daily scrum adalah kegiatan dengan batasan waktu maksimal selama 15 menit, dengan posisi *standing meeting*. *Daily scrum* dilakukan untuk mensinkronisasikan pekerjaan dan membuat perencanaan untuk 24 jam ke depan. Hal ini dilakukan dengan meninjau pekerjaan semenjak acara *daily scrum* terakhir dan memperkirakan pekerjaan yang dapat dilakukan sebelum melakukan *daily scrum* berikutnya. *Daily scrum* dilaksanakan pada waktu dan tempat yang sama setiap hari untuk mengurangi kompleksitas. Pada saat pertemuan ada beberapa pertanyaan yang perlu dijawab diantaranya apa yang sudah dilakukan kemarin, apa yang akan dilakukan hari ini, apa yang menghambat untuk menyelesaikan *goal* yang sudah ditentukan.

3. *Sprint review*

Sprint Review diadakan di akhir *sprint* untuk meninjau *inkremen* dan merubah *product backlog* bila diperlukan. Pada saat *Sprint Review*, Tim *Scrum* dan *stakeholder* berkolaborasi untuk membahas apa yang telah dikerjakan dalam *sprint* yang baru usai. Berdasarkan hasil tersebut tersebut dan semua perubahan *product backlog* pada saat *sprint*, para hadirin berkolaborasi menentukan apa yang dapat dikerjakan di *sprint* berikutnya, untuk mengoptimisasi nilai produk. Pertemuan ini bersifat *informal*, bukan merupakan status *meeting*, dan presentasi dari *inkremen* diharapkan dapat mengumpulkan masukan dan menumbuhkan semangat kolaborasi.

4. *Sprint retrospective*

Sprint retrospective adalah sebuah kesempatan bagi tim *scrum* untuk meninjau dirinya sendiri dan membuat perencanaan mengenai peningkatan yang akan dilakukan di Sprint berikutnya. Tujuan dari *sprint retrospective* adalah meninjau bagaimana *sprint* yang telah selesai berlangsung, termasuk hal-hal yang berkaitan dengan orang-orangnya, hubungan antara orang-orang, proses, dan perangkat kerja, mengidentifikasi dan mengurutkan hal-hal utama yang berjalan baik, dan hal-hal yang berpotensi untuk ditingkatkan, membuat rencana implementasi, dengan tujuan peningkatan cara-cara kerja tim *scrum*.

Berdasarkan *International Scrum Institute* (n.d.) *Scrum Product Backlog* memerlukan estimasi waktu agar *Scrum Product Owner* dapat memprioritaskan daftar *Product Backlog* dan memperkirakan jadwal peluncuran aplikasinya. Dalam penulisan estimasi waktu pada *Product Backlog* tidak terdapat aturan yang pasti. Penulisan estimasi waktu yang sering digunakan adalah *numeric sizing* (1 sampai 10), ukuran *t-shirt* (XS, S, M, L, XL, XXL, XXXL) atau menggunakan urutan Fibonacci (1, 2, 3, 5, 8, 13, 21, 34).

Menurut *International Scrum Institute* (n.d.) jumlah iterasi dalam pengembangan aplikasi akan dibuat selama proses perencanaan rilis aplikasi. Perencanaan rilis aplikasi harus mempertimbangkan beberapa faktor, yaitu:

1. Prioritas dan estimasi *Scrum Product Backlog*.
2. Perkiraan kecepatan pengerjaan *Scrum Team*.
3. Kondisi kepuasan (tujuan untuk jadwal, ruang lingkup, sumber daya)

Penentuan jumlah iterasi juga dilihat dari tipe pengerjaan proyeknya, terdapat dua tipe pengerjaan proyek, yaitu:

1. *Release Plan for a feature-driven project*

Rumus penentuan iterasi *Release Plan for a feature-driven project* adalah dengan membagi jumlah estimasi tingkat kesulitan dengan kecepatan mengerjakan *Scrum Team* (poin estimasi).

2. *Release Plan for a date-driven project*

Rumus penentuan iterasi *Release Plan for a date-driven project* adalah dengan mengkalikan kecepatan mengerjakan *Scrum Team* (poin estimasi) dengan durasi maksimal pengerjaan tiap *sprint*.

Alasan metode *scrum* digunakan oleh penulis dalam pengembangan sistem pemeringkat kinerja UMKM adalah karena fungsi-fungsi pada sistem yang berfokus pada permintaan *user* sehingga akan sering terjadi perubahan secara berkala pada fungsi-fungsi sistem. Berbeda dengan metode *waterfall* yang terbagi dalam beberapa fase proyek yaitu *Requirement Analysis*, *Design*, *Development*, *Test*, dan *Release* yang membuat pengembang aplikasi untuk tidak secara fleksibel merubah fungsi fungsinya. Fungsi yang terdapat pada sistem harus dirumuskan seluruhnya pada fase proyek *Requirement Analysis* terlebih dahulu baru dapat melanjutkan ke fase selanjutnya.

2.13. UML

Menurut Hendini (2016) *Unified Modeling Language* (UML) adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak. UML merupakan metodologi dalam mengembangkan sistem berorientasi objek dan juga merupakan alat untuk mendukung pengembangan sistem. Alat bantu yang digunakan dalam perancangan berorientasi objek berbasis UML adalah sebagai berikut:

a. *Use Case Diagram*

Use case diagram merupakan pemodelan untuk kelakuan (behavior) sistem informasi yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut.

b. *Activity Diagram*

Activity Diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Simbol-simbol yang digunakan dalam *activity Diagram* yaitu:

c. *Sequence Diagram*

Sequence Diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek.

d. *Class Diagram*

Merupakan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain dari suatu sistem, juga memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem. *Class Diagram* juga menunjukkan atribut-atribut dan operasi-operasi dari sebuah kelas dan constraint yang berhubungan dengan objek yang dikoneksikan.

2.14. Pengujian Aplikasi

Menurut Shi (2010) Pengujian aplikasi sangatlah diperlukan karena untuk memastikan apakah aplikasi yang sudah atau sedang dibuat dapat berjalan sesuai dengan fungsionalitas yang diharapkan atau tidak. Pengembang atau penguji aplikasi haruslah menyiapkan sesi khusus untuk menguji aplikasi yang sudah dibuat agar kesalahan ataupun kekurangan dapat dideteksi sejak awal dan dapat diperbaiki secepatnya. Pengujian atau *testing* sendiri merupakan elemen kritis dari jaminan kualitas perangkat lunak dan merupakan bagian yang tidak terpisah dari siklus hidup pengembangan software seperti halnya analisis, desain, dan pengkodean.

Menurut Khan (2011) dalam pengujian aplikasi terdapat dua jenis pengujian salah satunya, yaitu:

1. ***Black-Box Testing***

Black-box testing merupakan pengujian yang berfokus pada spesifikasi fungsional dari perangkat lunak, tester dapat mendefinisikan kumpulan kondisi input dan melakukan pengetesan pada spesifikasi fungsional program.

Black Box Testing bukan solusi alternatif dari *White Box Testing* melainkan lebih merupakan pelengkap untuk menguji hal-hal yang tidak dapat dicakup oleh *White Box Testing*. *Black Box Testing* lebih untuk menemukan hal-hal berikut:

1. Fungsi yang tidak benar atau tidak ada.
2. Kesalahan antarmuka (*interface errors*).
3. Kesalahan pada struktur data dan akses *database*.
4. Kesalahan performansi (*performance errors*).
5. Kesalahan inisialisasi dan terminasi.

Saat ini terdapat banyak metode atau teknik untuk melaksanakan Black Box Testing, antara lain:

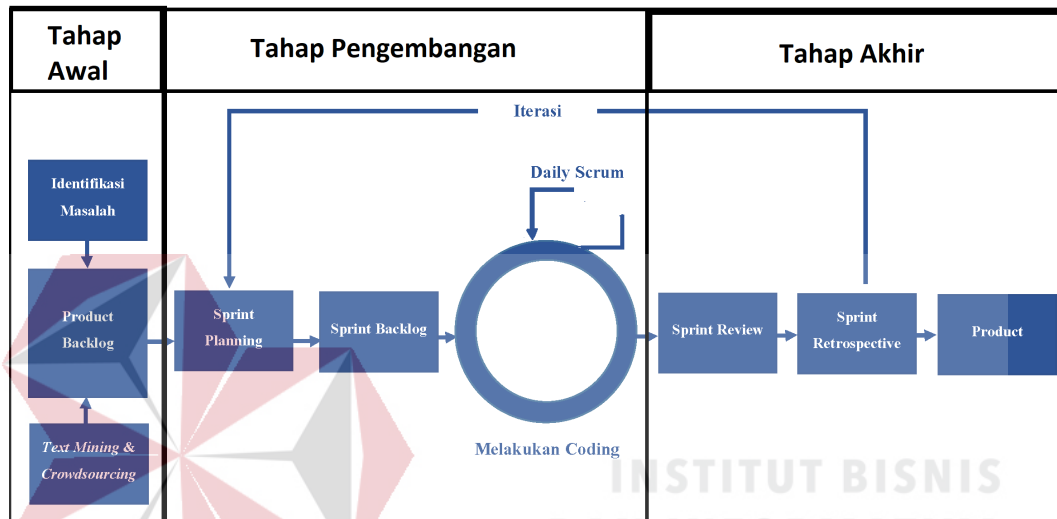
1. *Equivalence Partitioning*
2. *Boundary Value Analysis/Limit Testing*
3. *Comparison Testing*
4. *Sample Testing*
5. *Robustness Testing*
6. *Behavior Testing*
7. *Requirement Testing*
8. *Performance Testing*
9. Uji Ketahanan (*Endurance Testing*)
10. Uji Sebab-Akibat (*Cause-Effect Relationship Testing*)

Pengujian aplikasi yang akan digunakan pada pembuatan tugas akhir ini adalah *Black-Box Testing*, karena setelah menyelesaikan iterasi pertama aplikasi akan diuji apakah semua fungsionalitasnya sesuai dengan *product backlog* yang disusun ataukah tidak. Jika aplikasi sudah sesuai kemudian baru akan dilanjutkan pembuatan tampilan pada iterasi kedua.

BAB III

METODOLOGI PENELITIAN

Metode penelitian untuk menyelesaikan tugas akhir ini dengan menggunakan metode *scrum* yang dapat dilihat pada Gambar 3.1 di bawah ini.



Gambar 3.1 Tahapan Penelitian *Scrum*

3.1 Tahap Awal

Berdasarkan pada Gambar 3.1 pada tahap awal metode *scrum* ini terdapat beberapa tahapan yaitu melakukan identifikasi masalah dan membuat *product backlog*. Proses-proses tersebut dapat dijelaskan lebih lengkap sebagai berikut.

3.1.1. Mengidentifikasi Masalah

Tahap ini adalah mengidentifikasikan masalah yang ada dengan pihak-pihak yang terkait dengan cara melakukan studi literatur, wawancara, dan penyebaran kuisioner. Pihak yang dimaksud tersebut adalah *product owner*, *scrum*

master, stakeholder, dan scrum team. Pihak-pihak tersebut telah ditetapkan sebelumnya yaitu:

1. *Product Owner* adalah pihak Dewan Direksi Usahanesia.
2. *Scrum Master* adalah *Chief Technology Officer*(CTO) Usahanesia
3. *Stakeholder* adalah beberapa orang/organisasi pemegang kepentingan yaitu Usahanesia, UMKM, *Development Team*, dan para ahli.
4. *Scrum Team* adalah *IT Development* Usahanesia.

Adapun studi literatur yang dipelajari berupa jurnal yang berkaitan tentang UMKM serta *Business Lean Casvas* Usahanesia. *Business Lean Canvas* Usahanesia terdapat pada Gambar 3.2.

| <i>Problem</i> | <i>Solution</i> | <i>Unique Value Proposition</i> | <i>Unfair Advantages</i> | <i>Customer Segments</i> |
|--|---|--|---|---|
| <ul style="list-style-type: none"> - Kesulitan mendapatkan bahan baku (man, material) - Keterbatasan penguasaan teknologi yang rendah - Keterbatasan akses informasi bisnis yang relevan - Kesulitan dalam pemasaran - Masalah komunikasi | <ul style="list-style-type: none"> - Menyediakan sistem pelayanan pembinaan UMKM berdasarkan masalah yang mereka miliki - Menyediakan sistem portofolio berkelanjutan yang didukung dengan metode text mining dan gamification untuk meningkatkan daya saing - Menyediakan sistem kerjasama antar UKM dengan metode croudfounding untuk memperluas kerja sama dan meningkatkan penjualan | <ul style="list-style-type: none"> - Melakukan pendampingan pada UMKM terhadap masalah yang mereka hadapi - Menghubungkan UMKM dengan para profesional yang dibutuhkan. - Penerapan metode Gamification serta Croud sourcing untuk menarik partisipasi pengguna | <ul style="list-style-type: none"> - Penggunaan metode untuk dasar sistem agar dapat menarik partisipasi pengguna (How bussines engage user participation in crowdsourcing) - Pendampingan oleh para ahli yang dilakukan berdasarkan analisa kekurangan UKM (Data Management) | <ul style="list-style-type: none"> - UMKM yang memiliki kesulitan dalam mengembangkan produk mereka. - UMKM yang membutuhkan pembinaan dengan permasalahan yang mereka miliki |
| Key Metrics <ul style="list-style-type: none"> - Keikutsertaan UKM dan peningkatan bisnis (dapat diketahui oleh portofolio berkelanjutan) - (Indikator yang lebih jelas)* | | Channels <ul style="list-style-type: none"> - Instagram - Facebook - Twitter - Web | | |
| Cost Structure <ul style="list-style-type: none"> - Infrastruktur TI - Pembiayaan ahli* - Riset pasar | | Revenue Streams <ul style="list-style-type: none"> - Penggunaan user premium* - Iklan | | |

Gambar 3.2 *Business Lean Canvas* Usahanesia

Berdasarkan *business lean canvas* pada Gambar 3.2, Usahanesia memiliki misi untuk membantu mengatasi masalah UMKM yang mana melatarbelakangi

proyek ini yaitu adalah keterbatasan akses informasi tentang bagaimana caranya untuk mengembangkan usaha mereka. Masalah yang lainnya adalah, pihak UMKM kesulitan dalam mendapatkan bahan baku yang sesuai dengan apa yang mereka butuhkan.

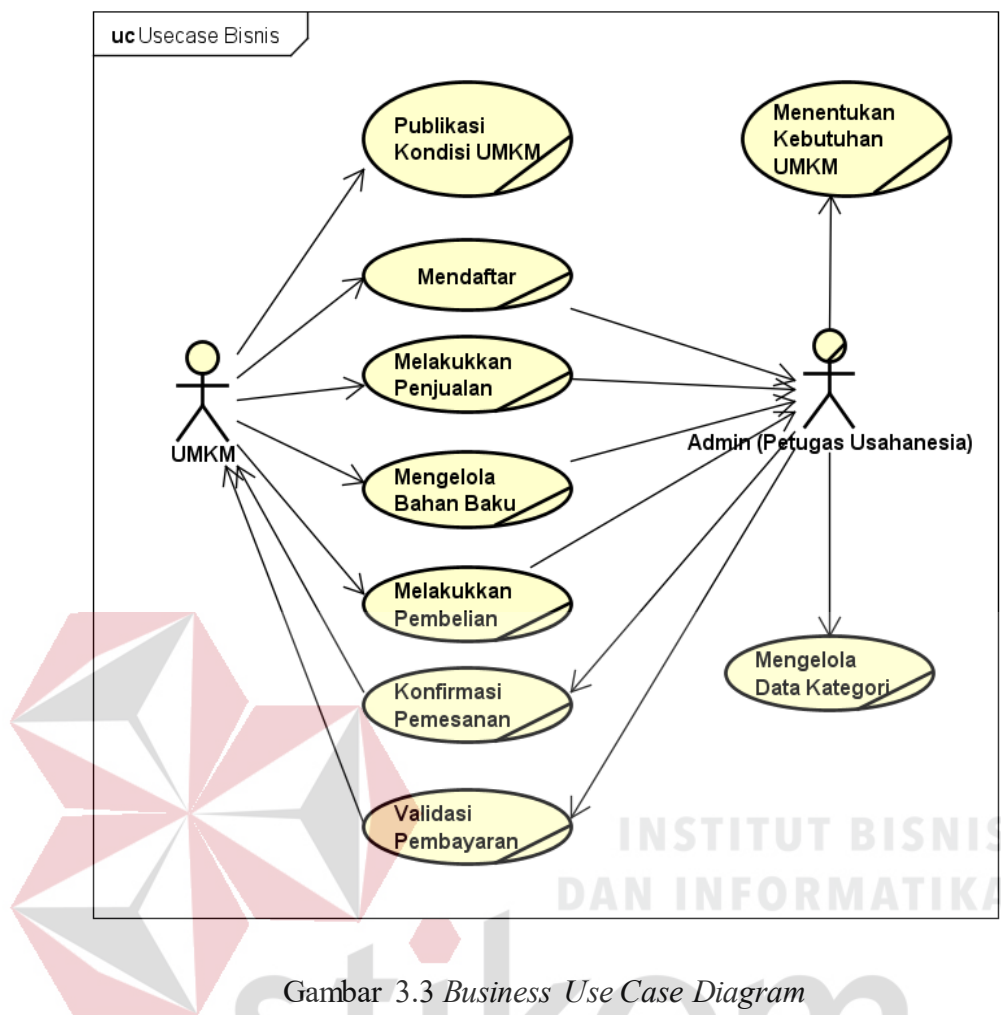
Tabel 3.1 Masalah dan alternatif solusi

| Masalah | Alternatif Solusi |
|--|--|
| Kesulitan mendapatkan bahan baku (material). | Membuat aplikasi pemenuhan bahan baku produksi UMKM dengan metode <i>Text Mining</i> . |

Berdasarkan hasil identifikasi dan *Lean Canvas Usahanesia* terdapat masalah dan alternatif solusi dapat dilihat pada Tabel 3.1 namun untuk mengidentifikasi masalah lebih jelas maka dilakukan identifikasi model bisnis saat ini dengan menggunakan *business use case diagram* dan *activity diagram*. Berikut adalah langkah-langkah yang dilakukan dalam identifikasi model bisnis.

A. *Bussiness Use Case Diagram*

Diagram ini digunakan pengembang untuk membantu memahami alur kegiatan bisnis yang dilakukan oleh UMKM dan hubungannya dengan pihak Usahanesia. *Bussiness Use Case* dapat dilihat pada Gambar 3.3.



Gambar 3.3 Business Use Case Diagram

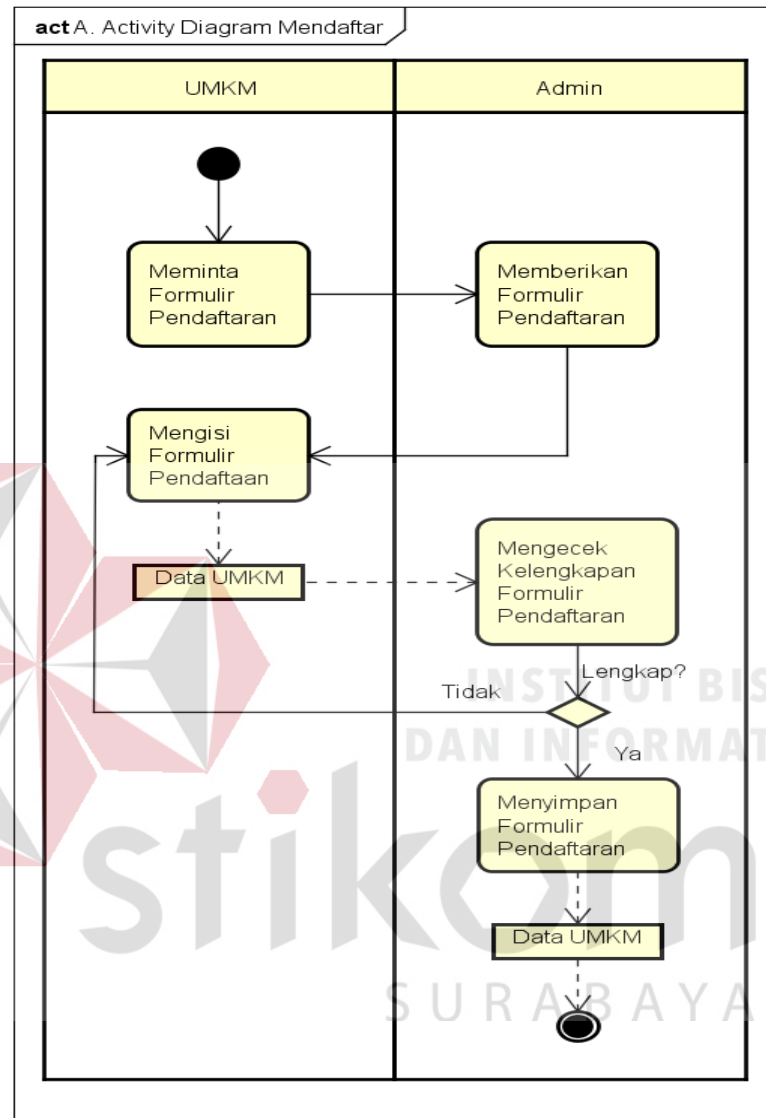
B. Activity Diagram

Setelah mengelompokkan alur kegiatan bisnis kedalam *business use case* maka selanjutnya dilakukan penggambaran lebih jelas akan langkah-langkah dalam aliran kerja, siapa yang melakukannya dan objek yang digunakan dengan menggunakan *activity diagram*.

1. Activity Diagram Mendaftar

Activity diagram mendaftar ini menggambarkan proses pendaftaran yang dilakukan UMKM dengan mengisi data-data yang diperlukan dan memberikanya

kepada *admin* (petugas usahanesia) untuk dilakukan pengecekan kelengkapan berkas. Seperti pada Gambar 3.4.

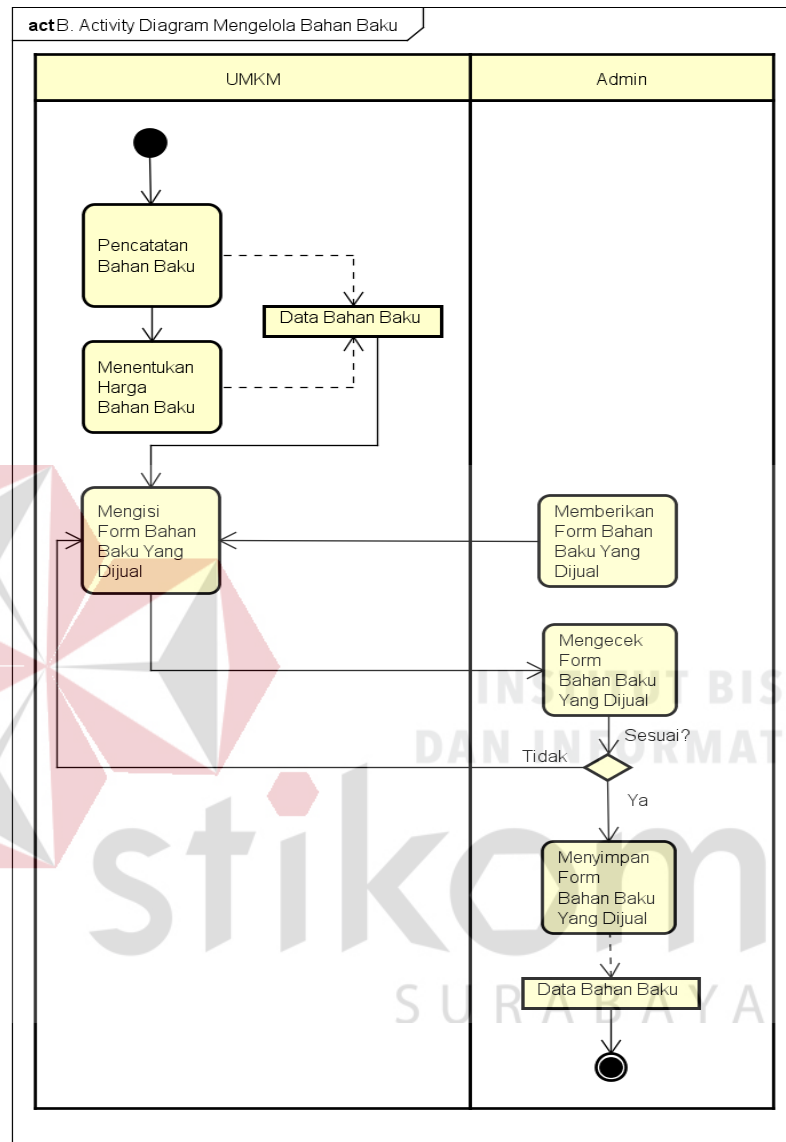


Gambar 3.4 Activity Diagram Mendaftar

2. Activity Diagram Mengelola Bahan Baku

Pada *activity diagram* mengelola bahan baku ini menggambarkan proses pengelolaan bahan baku yang dilakukan oleh UMKM mulai dari mencatat bahan

baku yang ada hingga mencatat bahan baku yang berlebih yang ingin dijual oleh UMKM. Seperti pada Gambar 3.5.

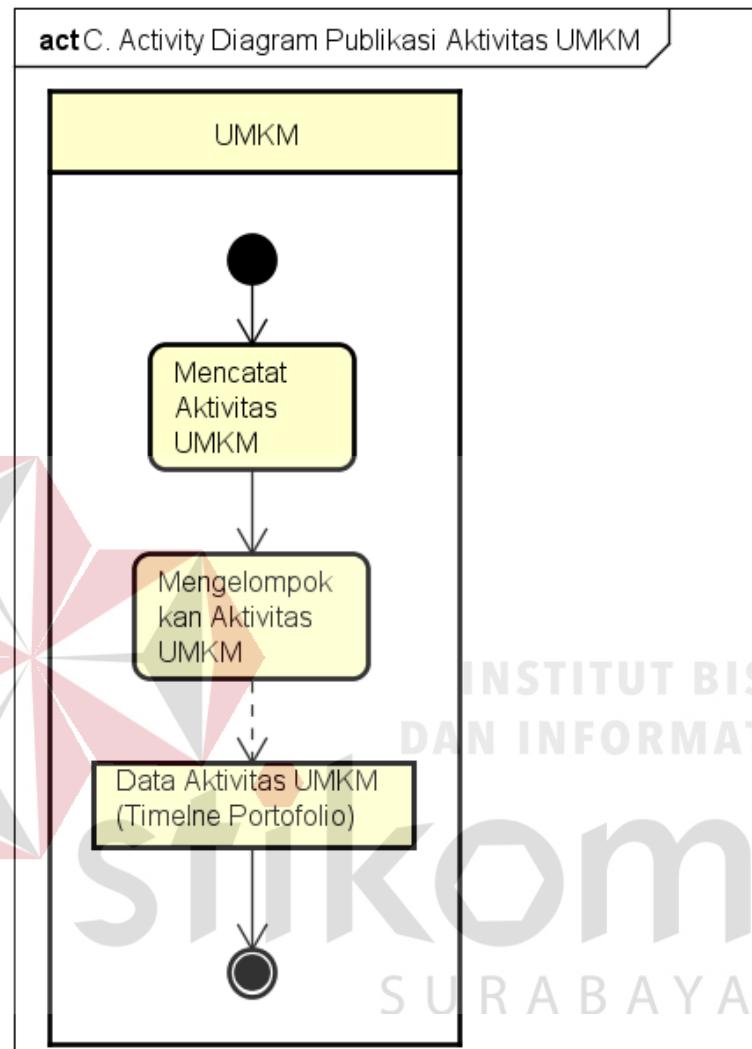


Gambar 3.5 Activity Diagram Mengelola Bahan Baku

3. Activity Diagram Publikasi Aktivitas UMKM

Activity diagram publikasi aktivitas UMKM ini menggambarkan proses UMKM dalam melakukan pencatatan aktivitas yang dilakukan UMKM yang

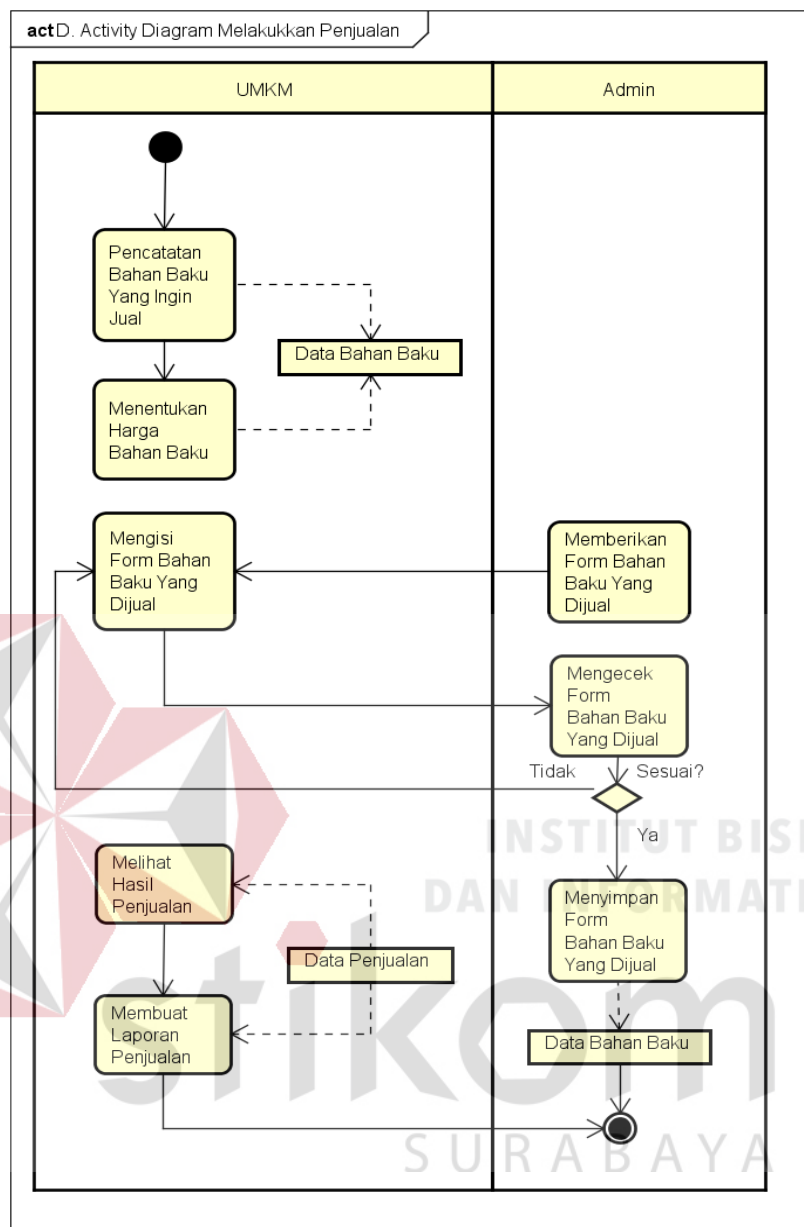
nantinya dapat digunakan *admin* untuk menentukan kebutuhan bahan baku dari UMKM tersebut. Seperti pada Gambar 3.6 berikut.



Gambar 3.6 *Activity Diagram* Publikasi Aktivitas UMKM

4. *Activity Diagram* Melakukan Penjualan

Pada *activity diagram* melakukan penjualan ini menggambarkan proses UMKM dalam melakukan penjualan mulai dari mencatat bahan baku yang ingin jual hingga membuat laporan penjualan. Seperti pada Gambar 3.7 berikut.

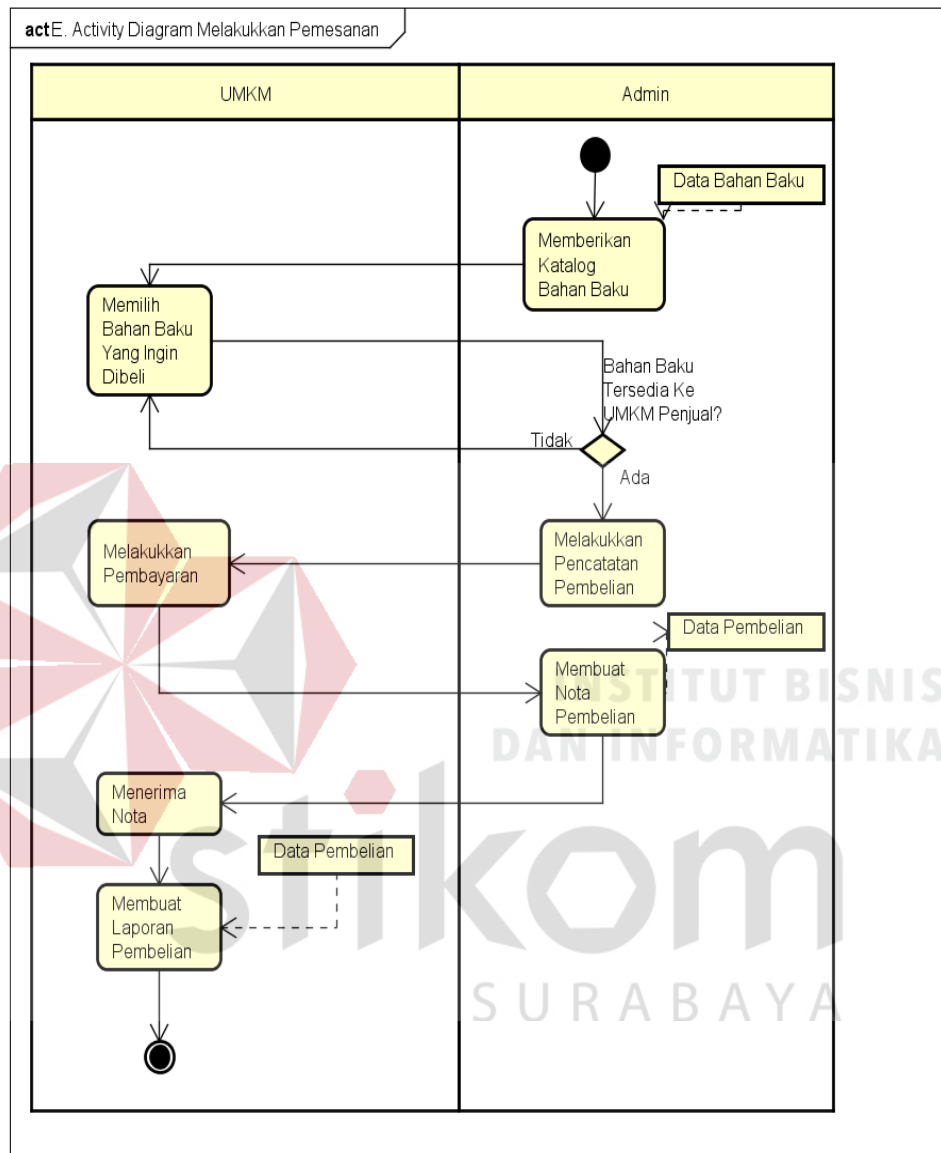


Gambar 3.7 Activity Diagram Melakukan Penjualan

5. Activity Diagram Melakukan Pembelian

Pada *activity diagram* melakukan pembelian ini menggambarkan proses pembelian yang dilakukan oleh UMKM mulai dari UMKM memilih bahan baku yang dibutuhkan berdasarkan katalog bahan baku yang diberikan oleh *admin*

(petugas Usahanesia) hingga membuat laporan pembelian bahan baku. Seperti pada Gambar 3.8 berikut.

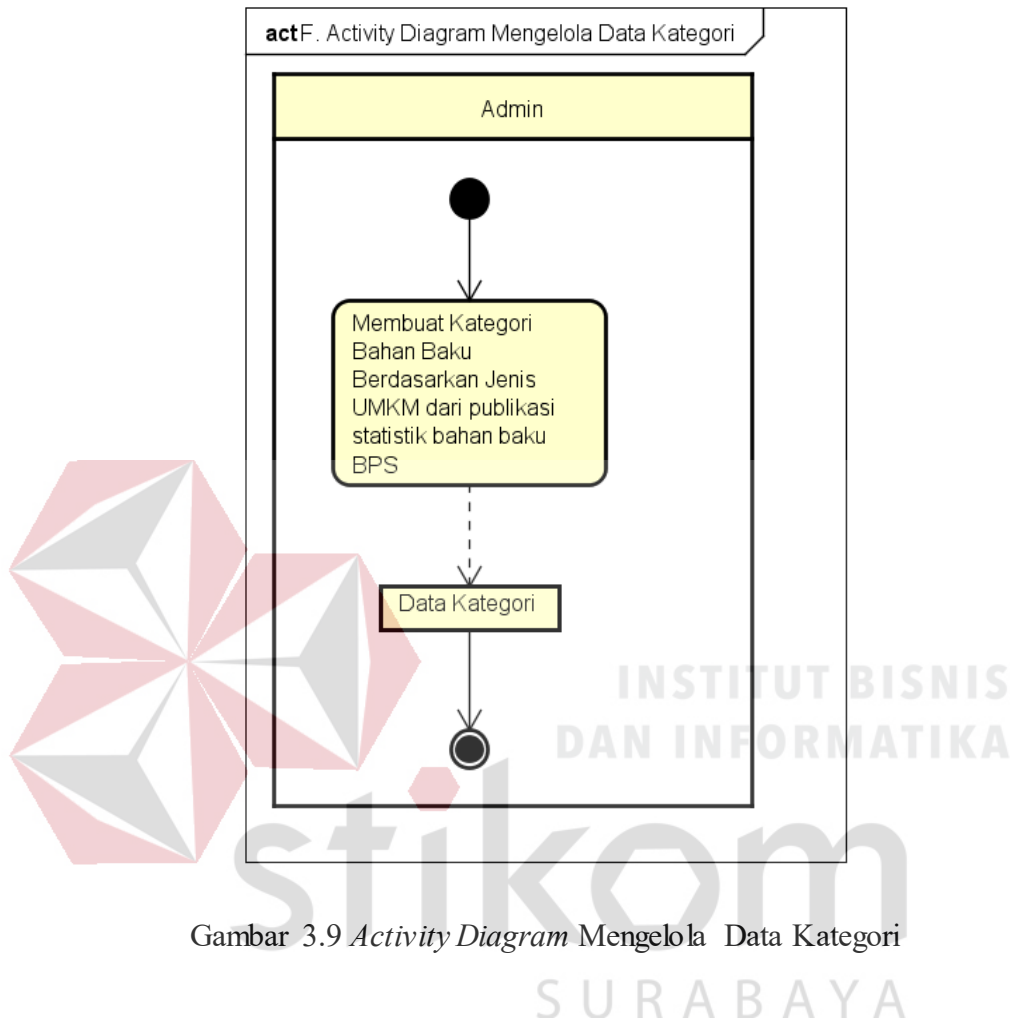


Gambar 3.8 Activity Diagram Melakukan Pembelian

6. Activity Diagram Mengelola Data Kategori

Pada *activity diagram* mengelola data kategori ini menggambarkan proses pengelolaan data kategori untuk bahan baku yang dilakukan oleh *admin* (petugas Usahanesia) berdasarkan pada publikasi Statistik Industri Manufaktur (Bahan

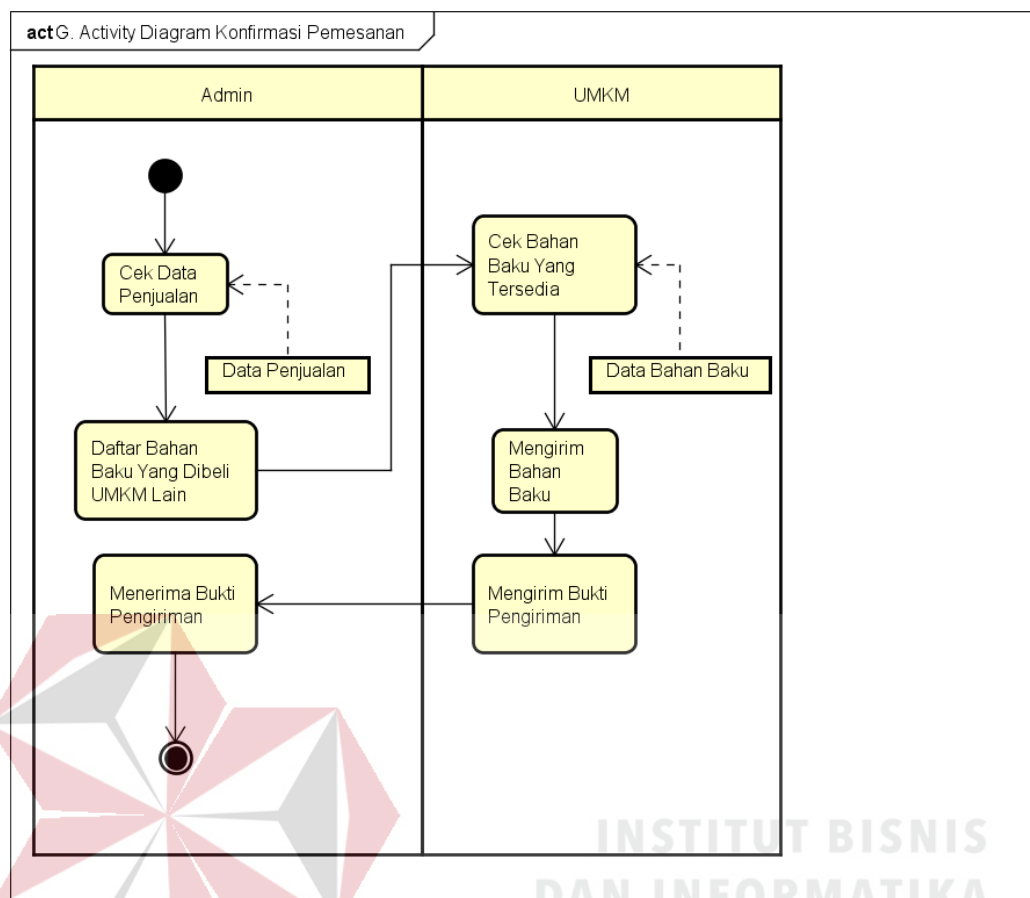
Baku) Provinsi Jawa Timur 2015 yang dibuat oleh Pramono (2015). Seperti pada Gambar 3.9 berikut.



Gambar 3.9 Activity Diagram Mengelola Data Kategori

7. Activity Diagram Konfirmasi Pemesanan

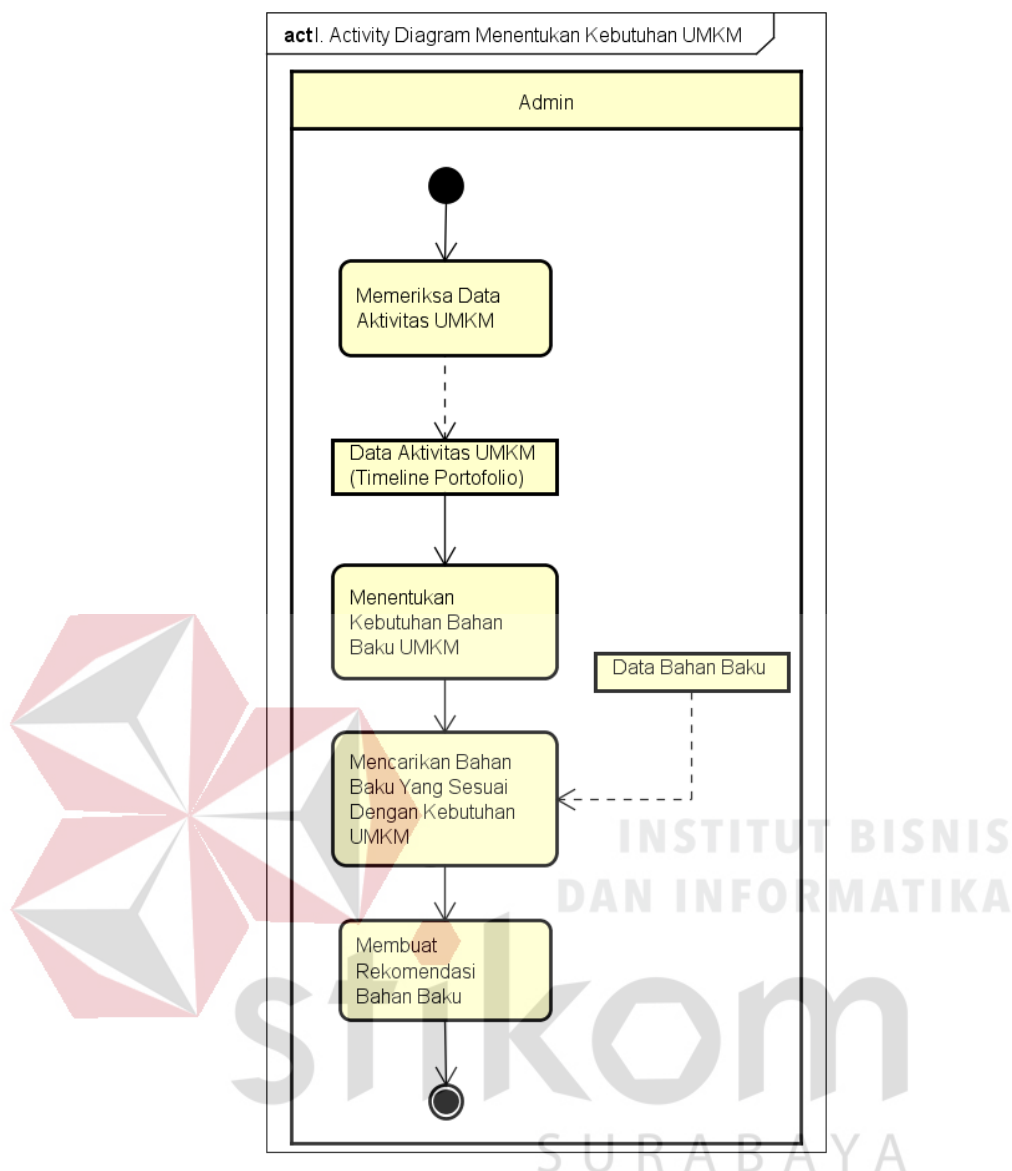
Pada *activity diagram* konfirmasi pemesanan ini menggambarkan proses pengonfirmasian pemesanan bahan baku yang dilakukan oleh *admin* dengan UMKM penjual bahan baku mulai dari pengecekan data penjualan oleh *admin* hingga pengiriman bahan baku yang dilakukan oleh UMKM penjual. Seperti pada Gambar 3.10 berikut.



Gambar 3.10 Activity Diagram Konfirmasi Pemesanan Bahan Baku

8. Activity Diagram Menentukan Kebutuhan UMKM

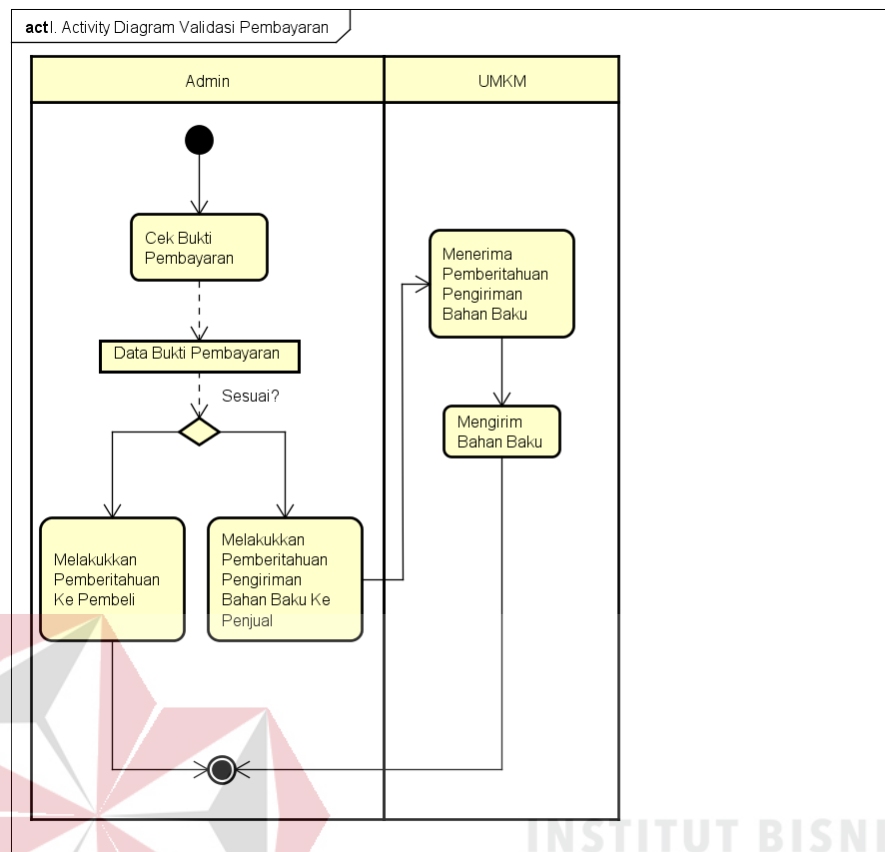
Pada *activity diagram* menentukan kebutuhan UMKM ini menggambarkan proses yang dilakukan *admin* (petugas Usahanesia) dalam menentukan kebutuhan bahan baku UMKM berdasarkan pada postingan aktivitas UMKM yang dilakukan pada proses sebelumnya. *Activity diagram* ini menggambarkan aktivitas *admin* mulai dari memeriksa data aktivitas UMKM hingga membuat rekomendasi bahan baku yang sesuai dengan kebutuhan UMKM. Seperti pada Gambar 3.11 berikut.



Gambar 3.11 *Activity Diagram* Menentukan Kebutuhan UMKM

9. *Activity Diagram* Validasi Pembayaran

Pada *activity diagram* validasi pembayaran ini menggambarkan proses validasi pembayaran yang dilakukan oleh *admin* (petugas Usahanesia) mulai dari melakukan pengecekan data bukti pembayaran hingga memberitahukan UMKM pembeli mengenai pengiriman bahan baku yang telah dilakukan oleh UMKM penjual. Seperti pada Gambar 3.12 berikut.



Gambar 3.12 Activity Diagram Validasi Pembayaran

3.1.2. Product Backlog

Tahap *product backlog* merupakan tahap pembuatan *list story* yang berisi semua fitur, fungsi, kebutuhan dan prioritas *user story* terhadap aplikasi dari produk yang akan dikerjakan. Tahapan-tahapan pembuatan *produk backlog* adalah sebagai berikut.

A. Analisis Kebutuhan Fungsi

Sebelum membuat *produk backlog* hal yang harus dilakukan terlebih dahulu adalah dilakukannya analisis kebutuhan fungsi agar pengembang dapat mengetahui

fitur dan fungsi apa saja yang harus dibuat untuk memenuhi kebutuhan pengguna.

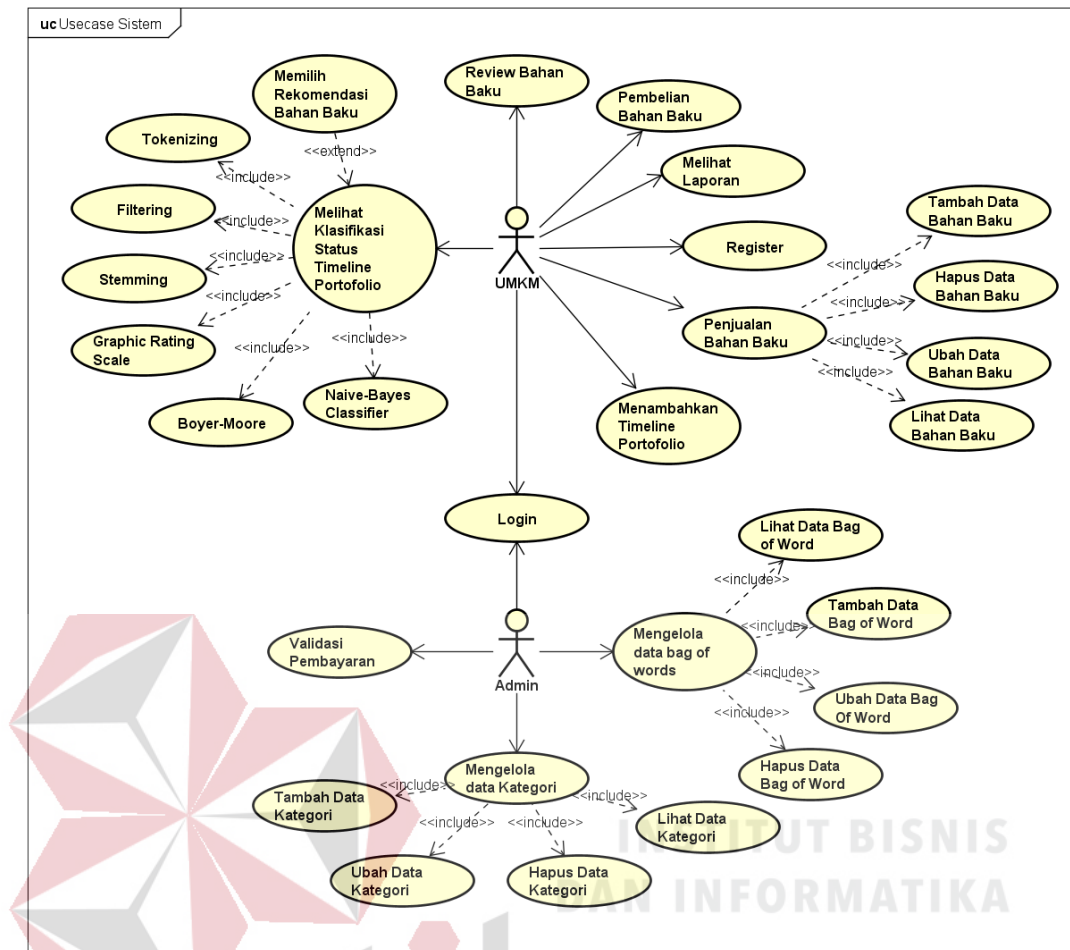
Hasil dari analisis kebutuhan fungsi dapat dilihat pada Tabel 3.2 berikut.

Tabel 3.2 Analisis Kebutuhan Fungsi

| Pengguna | Kebutuhan Fungsi |
|----------------------------|--|
| UMKM | 1. Pengelolaan Bahan Baku |
| | 2. Publikasi Aktivitas UMKM |
| | 3. Penjualan Bahan Baku |
| | 4. Pembelian Bahan Baku |
| | 5. Pendaftaran UMKM |
| | 6. Upload Bukti Pembayaran |
| | 7. Upload Bukti Pengiriman |
| | 8. Konfirmasi Pemesanan |
| | 9. Melihat Katalog Produk |
| | 10. Melihat Laporan Transaksi |
| | 11. Memilih Rekomendasi Bahan Baku |
| | 12. <i>Login</i> UMKM |
| | 13. <i>Review</i> Bahan Baku |
| | 14. Melihat Klasifikasi Status Aktivitas UMKM (<i>Timeline Portofolio</i>) |
| Admin (Petugas Usahanesia) | 1. Validasi Pembayaran |
| | 2. Pengelolaan Data Kategori |
| | 3. Menentukan Kebutuhan UMKM |
| | 4. Menambahkan Data <i>Bag Of Word</i> |
| | 5. Cek Pengguna Aktif |

B. Use Case System Diagram

Setelah melakukan analisis kebutuhan fungsi, hal yang kemudian harus dilakukan adalah memetakan hasil analisis kebutuhan fungsi kedalam *use case system diagram* seperti pada Gambar 3.13 berikut.



Gambar 3.13 Use Case System Diagram

C. Product Backlog

Setelah membuat *use case system diagram*, maka dilakukannya pemetaan fitur-fitur yang akan dikerjakan berdasarkan analisis kebutuhan fungsi yang telah dilakukan. Hal tersebut digambarkan menggunakan tabel yang memiliki kolom *ID*, *story*, estimasi, dan prioritas. *ID* adalah identitas pembeda pada setiap *story*. *Story* adalah bagian-bagian yang akan dikerjakan oleh tim pengembang atas permintaan pemilik produk. Estimasi adalah tingkat kesulitan dari setiap *story* yang ada. Penentuan estimasi berdasarkan angka bilangan fibonacci, contoh: 1, 2, 3, 5, 8, dst.

Total estimasi nantinya akan menjadi pertimbangan *product owner* untuk menentukan berapa kali akan melakukan literasi. Dan prioritas adalah tingkat pengerjaan *story*/fitur yang akan dilakukan terlebih dahulu.

Pada pembuatan aplikasi pemenuhan bahan baku pada *startup* Usahanesia ini. Setiap *story*/fitur tersebut digambarkan menggunakan Tabel 3.3 berikut ini.

Tabel 3.3 *Product Backlog*

| ID | Story | Sub Story | Estimasi (Tingkat Kesulitan) | Prioritas |
|-----|---|---|------------------------------|-----------|
| 1. | Login (UMKM) | Membuat sesi login menggunakan <i>Google Sign In</i> untuk UMKM | 3 | 8 |
| 2. | Register (UMKM) | Membuat <i>register</i> data-data kelengkapan UMKM | 1 | 9 |
| 3. | Login (Admin) | Membuat sesi login untuk <i>admin</i> | 1 | 1 |
| 4. | Mengelola Data Bag of Word (Admin) | Membuat user interface halaman pengelolaan <i>bag of word</i> | 1 | 2 |
| 5. | | Membuat proses CRUD (<i>Create, Read, Update, Delete</i>) data <i>bag of word</i> | 1 | 3 |
| 6. | Mengelola Data Kategori (Admin) | Membuat user interface halaman pengelolaan kategori bahan baku | 1 | 4 |
| 7. | | Membuat proses CRUD (<i>Create, Read, Update, Delete</i>) data kategori bahan baku | 1 | 5 |
| 8. | Validasi Pembayaran (Admin) | Membuat user interface halaman validasi pembayaran | 1 | 6 |
| 9. | | Membuat proses validasi pembayaran | 1 | 7 |
| 10. | | Melakukan analisis <i>text mining</i> terhadap portofolio berkelanjutan UMKM. | 3 | 21 |
| 11. | Melihat Klasifikasi Status Timeline Portofolio (UMKM) | Menggunakan <i>plug-in</i> dari <i>Sastrawi/tokenizing</i> untuk membantu dalam proses <i>text mining</i> pada tahap <i>tokenizing</i> | 5 | 22 |
| 12. | | Menggunakan <i>plug-in</i> dari <i>Sastrawi/sastrawi</i> untuk membantu proses <i>text mining</i> pada tahap <i>stemming</i> dan <i>filtering stopwords</i> | 5 | 23 |

| ID | Story | Sub Story | Estimasi (Tingkat Kesulitan) | Prioritas |
|-----|-----------------------------|---|------------------------------|-----------|
| 13. | | Menggunakan <i>bag of words</i> UMKM untuk membantu proses <i>text mining</i> | 5 | 24 |
| 14. | | Menggunakan <i>plug-in</i> algoritma <i>Naïve Bayes Classification</i> (NBC) untuk menentukan <i>cluster</i> kebutuhan atau kelebihan UMKM akan bahan baku. | 5 | 25 |
| 15. | | Menggunakan algoritma <i>Boyer Moore</i> untuk mencari kesamaan kata kunci hasil dari NBC berdasarkan portofolio UMKM dengan bahan baku yang dijual UMKM lain. | 3 | 26 |
| 16. | | Menggunakan algoritma <i>Jaro-Winkler</i> untuk mencari tingkat kesamaan kata kunci hasil dari NBC berdasarkan portofolio UMKM dengan bahan baku yang dijual UMKM lain. | 3 | 27 |
| 17. | | Menggunakan <i>Graphic Rating Scale</i> (GRS) untuk membuat rekomendasi bahan baku yang dibutuhkan UMKM berdasarkan hasil dari algoritma <i>Boyer-Moore</i> ataupun algoritma <i>Jaro-Winkler</i> | 3 | 28 |
| 18. | Pembelian Bahan Baku (UMKM) | Membuat <i>user interface</i> untuk katalog bahan baku | 1 | 10 |
| 19. | | Membuat <i>user interface</i> untuk keranjang bahan baku | 1 | 11 |
| 20. | | Membuat <i>user interface</i> untuk hasil pembelian dan detail pembelian bahan baku | 1 | 12 |
| 21. | | Membuat proses transaksi pembelian | 2 | 16 |
| 22. | Penjualan Bahan Baku (UMKM) | Membuat <i>user interface</i> untuk hasil penjualan bahan baku | 1 | 15 |
| 23. | | Membuat <i>user interface</i> untuk pengelolaan bahan baku | 1 | 13 |

| ID | Story | Sub Story | Estimasi (Tingkat Kesulitan) | Prioritas |
|--------|---------------------------|--|---------------------------------|-----------|
| 24. | | Membuat proses pengelolaan bahan baku | 1 | 14 |
| 25. | Melihat Laporan (UMKM) | Membuat laporan pembelian | 1 | 17 |
| 26. | | Membuat laporan penjualan | 1 | 18 |
| 27. | Review Bahan Baku | UMKM dapat memberikan <i>feedback</i> bahan baku yang telah dibeli | 2 | 19 |
| 28. | | Notifikasi terhadap UMKM apabila ada pemberitahuan terbaru mengenai proses pembelian ataupun penjualan yang dilakukan. | 2 | 20 |
| Jumlah | | | 57 | 28 |

Pada tahap *Product Backlog* ditentukan pula kesepakatan awal dengan *Product Owner* Usahanesia terkait pengembangan sistem, berikut adalah kesepakatan awal terkait pengembangan aplikasi pemenuhan bahan baku produksi UMKM:

1. Berdasarkan pandangan *Release plan for a feature- driven project* dapat ditentukan bahwa kecepatan *Scrum Team* dalam menyelesaikan 1 kali iterasi berjumlah 28 estimasi.
2. Jumlah Iterasi berjumlah 2 kali. Kegiatan pada Iterasi pertama adalah mengerjakan seluruh fungsi-fungsi yang ada pada aplikasi pemenuhan bahan baku produksi UMKM dengan metode *text mining* dan membuat tampilan awal, sedangkan kegiatan pada iterasi kedua adalah memperbaiki tampilan dan memperbaiki kekurangan-kekurangan berdasarkan hasil *review* dengan *product owner*.

3. Durasi waktu 1 kali iterasi adalah 4 minggu.

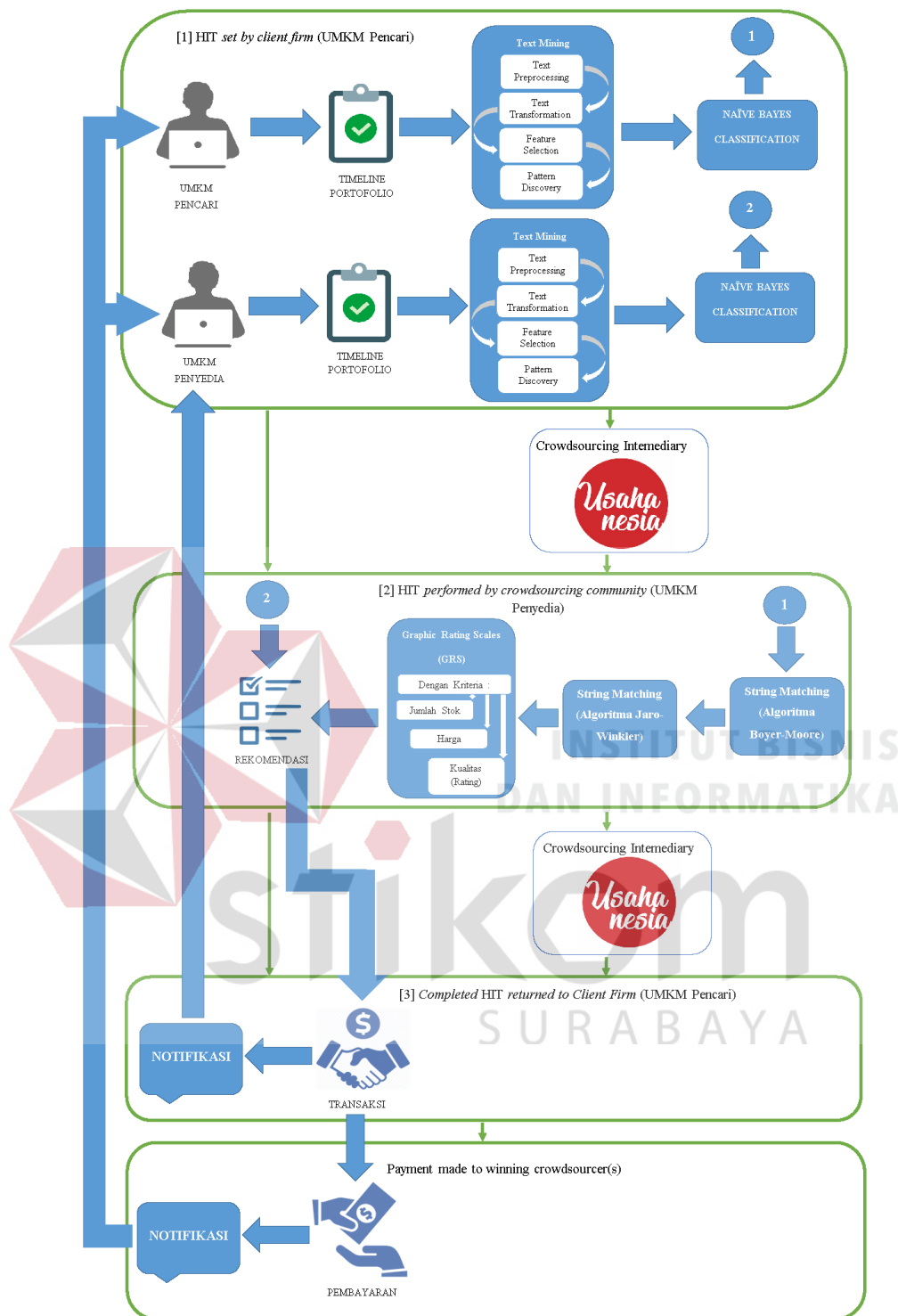
3.1.3. Model Pengembangan

Model Pengembangan digunakan untuk menjelaskan alur informasi pada pembuatan aplikasi pemenuhan bahan baku UMKM ini. Model pengembangan ini dibuat dengan tujuan untuk dapat mempermudah pengembang dalam pembuatan aplikasi pemenuhan bahan baku UMKM ini. Model pengembangan aplikasi ini dapat dilihat pada Gambar 3.14.

Model pengembangan pada Gambar 3.14 menggambarkan alur dari aplikasi pemenuhan kebutuhan bahan baku UMKM yang dibuat berdasarkan langkah-langkah pada metode *crowdsourcing*.

[1]. *Human Intelligence Task (HIT) set by Client Firm*

Pada tahap ini proses *Crowdsourcing* berawal dari UMKM mengisikan status UMKM pada *timeline portofolio*. Kemudian status tersebut akan diproses menggunakan metode *text mining*, mulai dari merubah status tersebut menjadi kata-kata (*tokenizing*), menghilangkan kata yang tidak penting (*filtering*), hingga merubah kata yang telah yang talh difilter menjadi kata dasar (*stemming*). Dari proses *text mining* tersebut akan menghasilkan beberapa kata kunci (*keyword*) yang akan digunakan pada proses selanjutnya, namun sebelum ke proses selanjutnya hasil dari *text mining* tersebut akan diklasifikasikan terlebih dahulu dengan menggunakan algoritma *naïve bayes classifier*. Hasil dari proses klasifikasi tersebut akan menunjukkan apakah UMKM tersebut dengan membutuhkan atau kelebihan bahan baku.



Gambar 3.14 Model Pengembangan

[2]. *HIT performed by Crowdsourcing Community*

Selanjutnya pada tahap ini jika hasil dari proses klasifikasi menunjukkan UMKM tersebut sedang kelebihan bahan baku, maka sistem akan mengarahkan UMKM ke halaman bahan baku untuk dapat menjual bahan baku yang nantinya bisa menjadi rekomendasi untuk UMKM yang sedang membutuhkan bahan baku. Dan jika hasil dari proses klasifikasi menunjukkan UMKM tersebut sedang membutuhkan bahan baku maka *keyword* dari hasil *text mining* yang sebelumnya akan digunakan untuk melakukan proses *string matching* menggunakan algoritma *boyer moore* dan *jaro winkler* untuk dapat menemukan bahan baku yang dibutuhkan UMKM. Fungsi dari penggunaan algoritma *boyer moore* disini untuk dapat mencari bahan baku yang dibutuhkan secara lebih cepat dan penggunaan algoritma *jaro winkler* adalah untuk dapat mengatasi kesalahan pengetikan (*typo*) bahan baku pada status UMKM sehingga sistem tetap dapat menemukan bahan baku yang dicari. Kemudian setelah bahan baku ditemukan maka *list* bahan baku tersebut akan dilakukan pemilihan tiga bahan baku yang terbaik berdasarkan harga, jumlah stok dan kualitas (*rating*) menggunakan metode *graphic rating scale*. Penggunaan metode *graphic rating scale* disini untuk dapat menghasilkan rekomendasi yang nantinya bisa dipilih oleh UMKM.

[3]. *Completed HIT returned to Client Firm*

Setelah UMKM memilih bahan baku yang diinginkan, maka UMK tersebut dapat langsung melanjutkan pada proses transaksi pembelian. Jika proses pembelian telah selesai maka sistem akan memberikan notifikasi kepada UMKM penyedia bahan baku bahwa bahan baku yang dijual ada yang membutuhkannya.

[4]. *Payment made to Winning Crowdsourcer(s)*

Kemudian setelah UMKM penyedia menyetujui penjualan bahan bakunya. Maka UMKM pencari dapat langsung melakukan pembayaran melalui transfer dan mengirimkan bukti pembayaran agar bahan baku dapat segera dikirim. Jika bukti pembayaran telah divalidasi maka UMKM penyedia bahan baku akan mendapatkan notifikasi untuk dapat segera mengirimkan bahan baku yang dijualnya ke alamat UMKM pembeli. Dan UMKM pembeli juga akan mendapatkan notifikasi bahwa bahan baku sedang dikirim.

3.2. Tahap Pengembangan

Pada tahap pengembangan ini terdapat beberapa proses-proses yang harus dilakukan yaitu *sprint planning*, *sprint backlog* dan *daily scrum*. untuk penjelasan lebih lanjut mengenai setiap proses-proses tersebut adalah sebagai berikut.

3.2.1. *Sprint Planning*

Berdasarkan *product backlog* jumlah iterasi yang akan dilakukan dalam pembuatan aplikasi pemenuhan bahan baku produksi adalah sebanyak 2 iterasi yang telah ditentukan berdasarkan kesepakatan *scrum team* dengan *product owner*. Penjelasan detail mengenai kegiatan yang dilakukan pada setiap iterasi adalah sebagai berikut :

A. *Sprint Planning* Iterasi Pertama

Sprint planning pada iterasi pertama memiliki *sprint goal* aplikasi pemenuhan bahan baku berbasis *web* dapat digunakan untuk membaca data UMKM yang terdapat pada *Timeline Portofolio* dengan metode *Text Mining* dan *Naïve-*

Bayes Classifier, membuat rekomendasi produk bahan baku menggunakan metode GRS, membuat katalog produk, membuat proses *String Matching* dengan algoritma *Boyer-Moore*, membuat proses transaksi dan pembayaran, membuat laporan hasil pembelian dan penjualan bahan baku, membuat pengelolaan bahan baku. Untuk lebih menjelaskan tahapan-tahapan pada sprint planning iterasi pertama ini maka digunakannya *use case system diagram*, *flow of event*, *sequence diagram*, dan *class diagram*. Penjelasan lebih lanjut mengenai setiap proses-proses tersebut adalah sebagai berikut.

B. *Sprint Planning* Iterasi Kedua

Pada *sprint planning* iterasi kedua ini yang akan dikerjakan adalah hal yang sama dari *sprint planning* iterasi pertama, dengan perbedaannya adalah memperbaiki tampilan dan memperbaiki sistem berdasarkan dari hasil *review* dengan *product owner*.

3.2.2. *Sprint Backlog*

Sprint backlog adalah hasil dari *sprint planning* yang terdapat fungsi dari setiap *story* yang telah didapat di *product owner* dan tim *developer*. *Product owner* dan *developer team* mendiskusikan sekiranya fungsi dan tujuan dari setiap *story* yang telah dibuat pada *product backlog*. Laporan dari *sprint backlog* dapat dilihat pada Tabel 3.4 berikut.

Tabel 3.4 *Sprint Backlog*

| No. | Story | Fungsi | Tujuan |
|-----|--|---|--|
| 1. | Membuat sesi <i>login</i> menggunakan <i>Google Sign In</i> untuk UMKM | Menggunakan API Google untuk mengambil data | Dapat membuka halaman utama <i>admin</i> yang digunakan pada |

| No. | Story | Fungsi | Tujuan |
|-----|--|--|---|
| | | UMKM yang terdapat pada Google | aplikasi pemenuhan bahan baku UMKM |
| 2. | Membuat <i>register</i> data-data kelengkapan UMKM | Menyimpan data UMKM secara umum | Dapat menyimpan data UMKM secara umum |
| 3. | Membuat sesi <i>login</i> untuk <i>admin</i> | Menyimpan data UMKM secara umum | Dapat Menyimpan data UMKM secara umum |
| 4. | Membuat <i>user interface</i> halaman pengelolaan <i>bag of word</i> | Menampilkan tampilan halaman <i>bag of word</i> meliputi list <i>bag of word</i> , tombol tambah, edit dan hapus | Mempermudah <i>admin</i> dalam menggunakan aplikasi untuk proses pengelolaan data <i>bag of word</i> |
| 5. | Membuat proses CRUD (<i>Create, Read, Update, Delete</i>) data <i>bag of word</i> | <ul style="list-style-type: none"> - Menampilkan data <i>bag of word</i> - Menambah data <i>bag of word</i> - Mengubah data <i>bag of word</i> - Menghapus data <i>bag of word</i> | Data dapat digunakan pada proses klasifikasi status <i>timeline portofolio</i> |
| 6. | Membuat <i>user interface</i> halaman pengelolaan kategori bahan baku | Menampilkan halaman kategori meliputi list kategori, tombol tambah, edit dan hapus | Mempermudah <i>admin</i> dalam menggunakan aplikasi untuk proses pengelolaan data kategori bahan baku |
| 7. | Membuat proses CRUD (<i>Create, Read, Update, Delete</i>) data kategori bahan baku | <ul style="list-style-type: none"> - Menampilkan data kategori bahan baku - Menambah data kategori bahan baku - Mengubah data kategori bahan baku - Menghapus data kategori bahan baku | Data dapat digunakan pada proses pengelolaan bahan baku |
| 8. | Membuat <i>user interface</i> halaman validasi pembayaran | Menampilkan halaman validasi pembayaran meliputi list pembayaran, tombol valid, dan tidak valid | Mempermudah <i>admin</i> dalam melakukan proses validasi pembayaran UMKM |
| 9. | Membuat proses validasi pembayaran | Melakukan validasi dengan menekan tombol valid atau tidak valid | Dapat memperbarui status pembelian yang telah dilakukan UMKM |
| 10. | Melakukan analisis <i>text mining</i> terhadap portofolio | Melakukan proses <i>tokenizing, filtering stopword</i> dan <i>stemming</i> | Dapat memperoleh kata-kata yang akan menjadi <i>keyword</i> saat |

| No. | Story | Fungsi | Tujuan |
|-----|---|--|--|
| | berkelanjutan UMKM. | pada status <i>timeline portfolio</i> | melakukan proses klasifikasi dengan NBC |
| 11. | Menggunakan <i>plug-in</i> dari <i>Sastrawi/tokenizing</i> untuk membantu dalam proses <i>text mining</i> pada tahap <i>tokenizing</i> | Mempermudah melakukan <i>tokenizing</i> berbahasa Indonesia pada status <i>timeline portfolio</i> | Memecah kalimat pada status <i>timeline portfolio</i> menjadi beberapa kata |
| 12. | Menggunakan <i>plug-in</i> dari <i>Sastrawi/sastrawi</i> untuk membantu proses <i>text mining</i> pada tahap <i>stemming</i> dan <i>filtering stopwords</i> | Mempermudah melakukan <i>filtering</i> dan <i>stemming</i> pada tahapan <i>text mining</i> berbahasa Indonesia | Membersikan kalimat dari kata-kata yang tidak penting dan mengubah beberapa kata penting hasil dari <i>tokenizing</i> menjadi kata dasar |
| 13. | Menggunakan <i>bag of words</i> UMKM untuk membantu proses <i>text mining</i> | Menilai kata yang berasal dari proses <i>stemming</i> dengan kamus <i>bag of words</i> UMKM | Untuk menilai seberapa penting kata yang dihasilkan |
| 14. | Membuat <i>user interface</i> untuk katalog bahan baku | Menampilkan <i>list</i> bahan baku yang dijual oleh UMKM lain. | Mempermudah UMKM dalam mencari bahan baku yang dibutuhkan. |
| 15. | Membuat <i>user interface</i> untuk keranjang bahan baku | Menampilkan <i>list</i> bahan baku yang ingin dibeli oleh UMKM. | Mempermudah UMKM dalam melihat bahan baku apa saja yang ingin dibeli. |
| 16. | Membuat <i>user interface</i> untuk hasil pembelian dan detail pembelian bahan baku | Menampilkan <i>list</i> pembelian yang telah dilakukan dan <i>list</i> bahan baku apa saja yang dibeli | Mempermudah UMKM dalam melihat <i>history</i> transaksi pembelian yang telah dilakukan |
| 17. | Membuat <i>user interface</i> untuk hasil penjualan bahan baku | Menampilkan <i>list</i> penjualan yang telah dilakukan. | Mempermudah UMKM dalam melihat <i>history</i> transaksi penjualan yang telah dilakukan |
| 18. | Membuat proses transaksi pembelian | Memproses pembelian yang dilakukan oleh UMKM | Dapat menyimpan data pembelian yang dilakukan UMKM |

| No. | Story | Fungsi | Tujuan |
|-----|--|--|--|
| 19. | Membuat laporan pembelian | Menampilkan laporan hasil pembelian yang dipilih UMKM | UMKM dapat melihat dan mencetak laporan pembelian yang diinginkan |
| 20. | Membuat user interface untuk pengelolaan bahan baku | Menampilkan halaman bahan baku meliputi list bahan baku yang dijual, tombol tambah, edit dan hapus bahan baku | Mempermudah UMKM dalam melakukan pengelolaan bahan baku yang ingin dijual |
| 21. | Membuat proses pengelolaan bahan baku | <ul style="list-style-type: none"> - Menampilkan data bahan baku - Menambah data bahan baku - Mengubah data bahan baku - Menghapus data bahan baku | Data dapat digunakan untuk proses pembelian |
| 22. | Membuat laporan penjualan | Menampilkan laporan hasil penjualan yang telah dilakukan | UMKM dapat melihat dan mencetak laporan hasil penjualan yang diinginkan |
| 23. | Menggunakan <i>plug-in</i> algoritma <i>Naïve Bayes Classification</i> (NBC) untuk menentukan <i>cluster</i> kebutuhan atau kelebihan UMKM akan bahan baku. | Mempermudah melakukan klasifikasi terhadap kata-kata yang diolah pada tahap <i>text mining</i> . | Menentukan apakah dari hasil dari kata-kata tersebut, UMKM membutuhkan bahan baku atau kelebihan bahan baku sesuai dengan masalah yang dihadapi UMKM |
| 24. | Menggunakan algoritma <i>Boyer Moore</i> untuk mencari kesamaan kata kunci hasil dari NBC berdasarkan portofolio UMKM dengan bahan baku yang dijual UMKM lain. | Mencari kesamaan kata hasil NBC dengan bahan baku yang dijual UMKM lain | Menemukan bahan baku yang sesuai dengan kebutuhan UMKM |
| 25. | Menggunakan algoritma <i>Jaro-Winkler</i> untuk mencari tingkat kesamaan kata kunci hasil dari NBC berdasarkan portofolio UMKM | Mencari kesamaan kata hasil NBC dengan bahan baku yang dijual UMKM lain | Menemukan bahan baku yang sesuai dengan kebutuhan UMKM |

| No. | Story | Fungsi | Tujuan |
|-----|---|--|---|
| | dengan bahan baku yang dijual UMKM lain. | | |
| 26. | Menggunakan <i>Graphic Rating Scale</i> (GRS) untuk membuat rekomendasi bahan baku yang dibutuhkan UMKM berdasarkan hasil dari algoritma <i>Boyer-Moore</i> ataupun algoritma <i>Jaro-Winkler</i> | Membuat rekomendasi bahan baku yang paling sesuai dengan apa yang dibutuhkan UMKM berdasarkan hasil dari algoritma <i>boyer-moore</i> atau algoritma <i>jaro-winkler</i> | Menampilkan rekomendasi bahan baku yang paling sesuai dengan apa yang dibutuhkan UMKM |
| 27. | UMKM dapat memberikan <i>feedback/review</i> bahan baku yang telah dibeli | Mengisi <i>review/feedback</i> oleh UMKM terhadap bahan baku yang telah dibeli | Dapat mengisi <i>review/feedback</i> terhadap bahan baku yang telah dibeli. |
| 28. | Notifikasi terhadap UMKM apabila ada pemberitahuan terbaru mengenai proses pembelian ataupun penjualan yang dilakukan. | Memunculkan pemberitahuan terbaru mengenai proses pembelian dan penjualan yang dilakukan | Pengguna selalu mendapatkan kabar yang <i>up to date</i> |

1. Fungsi Login Admin

Fungsi ini digunakan oleh *Admin* untuk dapat mengakses fungsi-fungsi yang ada pada halaman *admin* aplikasi pemenuhan bahan baku UMKM. Penjelasan mengenai perancangan fungsi *login admin* adalah sebagai berikut.

a. Flow Of Event Login Admin

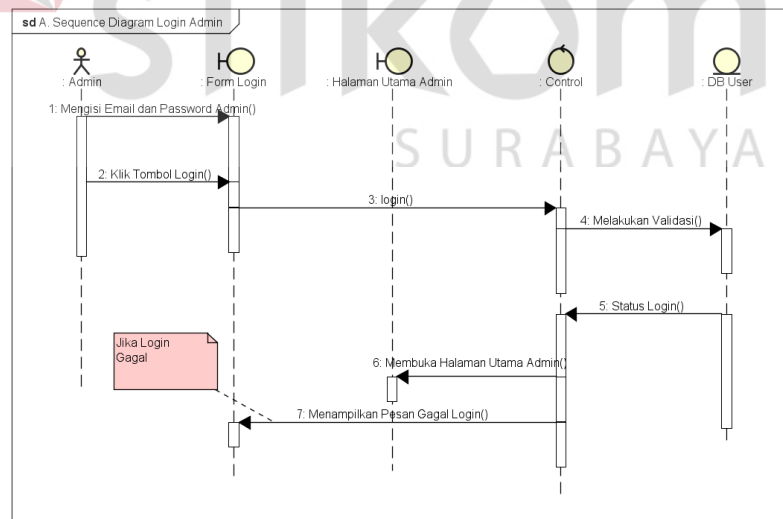
Proses ini dilakukan oleh *admin* untuk dapat masuk kedalam halaman *admin* aplikasi pemenuhan bahan baku UMKM. Seperti pada Tabel 3.5 berikut.

Tabel 3.5 Flow Of Event Login Admin

| Nama Flow Of Event | Login Admin | | |
|----------------------------|--|--|---|
| Aktor | Admin. | | |
| Deskripsi | Pihak <i>admin</i> Usahanesia dapat melakukan <i>login</i> untuk dapat mengakses aplikasi. | | |
| Kondisi Awal | Admin mempunyai <i>email</i> dan <i>password</i> . | | |
| Kondisi Akhir | Sistem menampilkan halaman <i>admin</i> . | | |
| Aliran Kejadian Utama | No. | Aktor | Sistem |
| | 1. | Admin mengisi Email dan password yang dimiliki | Sistem melakukan validasi email dan password, jika benar sistem akan menampilkan halaman utama admin. |
| Aliran Kejadian Alternatif | 1. | Admin salah mengisi Email dan Password yang dimiliki | Sistem akan menampilkan pesan error dan akan kembali ke halaman login. |

b. Sequence Diagram Login Admin

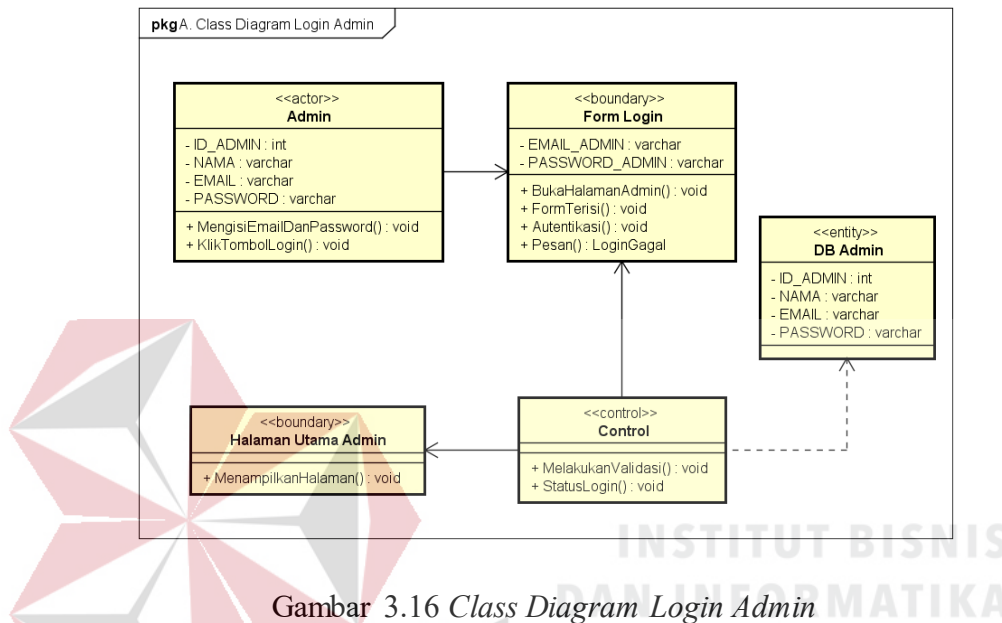
Pada diagram ini menggambarkan alur proses *login admin* lebih detail. Seperti pada Gambar 3.15 berikut.



Gambar 3.15 Sequence Diagram Login Admin

c. **Class Diagram Login Admin**

Pada diagram ini menggambarkan *attribute*, *operation* dan *relation* yang ada pada setiap *actor*, *boundary*, *control* dan *entity* pada fungsi *login admin* berdasarkan alur di *sequence diagram*. Seperti pada Gambar 3.16 berikut.



Gambar 3.16 Class Diagram Login Admin

d. **Desain Test Case Login Admin**

Desain uji coba pada fungsi *login admin* apakah sudah berjalan sesuai keinginan atau tidak. Desain uji coba fungsi *login admin* dapat dilihat pada Tabel 3.6 berikut.

Tabel 3.6 Desain Test Case Login Admin

| Test Case ID | Tujuan | Input | Output Yang Diharapkan | Status |
|--------------|--|---------------------------------|---|--------|
| 1. | Melakukan <i>login admin</i> | Email dan password <i>admin</i> | <i>Login</i> berhasil dan sistem menampilkan halaman <i>admin</i> | |
| 2. | Melakukan <i>login</i> dengan format salah | Tidak ada input | Muncul pesan : email dan password tidak boleh kosong | |
| 3. | Melakukan <i>login</i> dengan format salah | Email dan password yang salah | Muncul pesan : email atau password salah | |

2. Fungsi Mengelola Data *Bag Of Word*

Fungsi ini digunakan oleh *Admin* untuk dapat mengelola data *bag of word* yang nantinya akan digunakan saat melakukan klasifikasi *timeline portofolio*. Fungsi ini juga mencakup proses tambah data, ubah data dan hapus data *bag of word*. Penjelasan mengenai perancangan fungsi cek pengguna aktif adalah sebagai berikut.

a. *Flow Of Event* Mengelola Data *Bag Of Word*

Proses ini dilakukan oleh *admin* untuk dapat mengelola data *bag of word* yang nantinya akan digunakan saat melakukan klasifikasi *timeline portofolio*. Seperti pada Tabel 3.7 berikut.

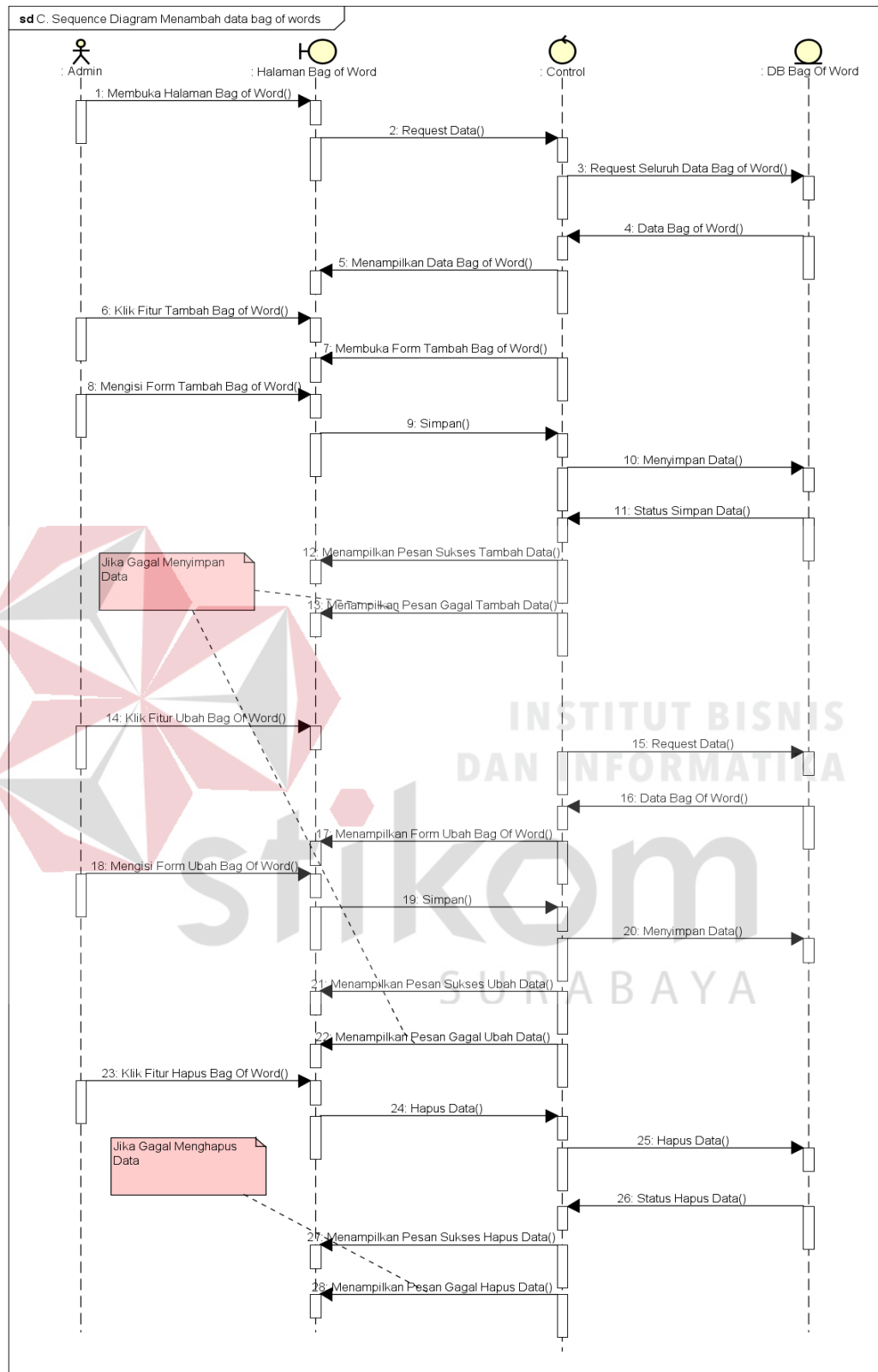
Tabel 3.7 *Flow Of Event* Mengelola Data *Bag Of Word*

| Nama <i>Flow Of Event</i> | | Mengelola Data <i>Bag Of Word</i> | |
|---------------------------|-----|---|--|
| Aktor | | <i>Admin.</i> | |
| Deskripsi | | Pihak aktor dapat mengelola data <i>bag of word</i> mulai dari tambah data, ubah data, hingga hapus data <i>bag of word</i> . | |
| Kondisi Awal | | Aktor membuka halaman <i>bag of word</i> . | |
| Kondisi Akhir | | Sistem menampilkan halaman <i>bag of word</i> . | |
| Aliran Kejadian Utama 1 | No. | Aktor | Sistem |
| | 1. | <i>Admin</i> memilih menu <i>bag of word</i> | Sistem menampilkan menu data <i>bag of word</i> dan menampilkan seluruh data <i>bag of word</i> |
| | 2. | <i>Admin</i> memilih fitur tambah data <i>bag of word</i> | Sistem menampilkan form tambah data <i>bag of word</i> |
| | 3. | <i>Admin</i> mengisi data <i>bag of word</i> | Sistem melakukan validasi isi form jika terisi sistem akan menyimpan data <i>bag of word</i> dan menampilkan pesan data berhasil ditambahkan |
| Aliran Kejadian Utama 2 | 1. | <i>Admin</i> memilih menu <i>bag of word</i> | Sistem menampilkan menu data <i>bag of word</i> dan menampilkan seluruh data <i>bag of word</i> |

| Nama <i>Flow Of Event</i> | Mengelola Data <i>Bag Of Word</i> | | |
|------------------------------|-----------------------------------|--|---|
| | 2. | <i>Admin</i> memilih fitur ubah data <i>bag of word</i> | Sistem menampilkan form data <i>bag of word</i> yang ingin diubah |
| | 3. | <i>Admin</i> merubah data <i>bag of word</i> | Sistem melakukan validasi isi form jika terisi sistem akan menyimpan data <i>bag of word</i> dan menampilkan pesan data berhasil diubah |
| Aliran Kejadian Utama 3 | 1. | <i>Admin</i> memilih menu <i>bag of word</i> | Sistem menampilkan menu data <i>bag of word</i> dan menampilkan seluruh data <i>bag of word</i> |
| | 2. | <i>Admin</i> memilih fitur hapus data <i>bag of word</i> | Sistem menampilkan pesan konfirmasi hapus data |
| | 3. | <i>Admin</i> menyetujui konfirmasi hapus data | Sistem menghapus data <i>bag of word</i> yang dipilih dan menampilkan pesan data berhasil diubah |
| Aliran Kejadian Alternatif 1 | 1. | <i>Admin</i> tidak mengisi form tambah data <i>bag of word</i> | Sistem akan menampilkan pesan data harus di isi dan akan kembali ke halaman <i>bag of word</i> . |
| Aliran Kejadian Alternatif 2 | 1. | <i>Admin</i> tidak menyetujui konfirmasi hapus data | Sistem akan kembali ke halaman <i>bag of word</i> . |

b. Sequence Diagram Mengelola Data *Bag Of Word*

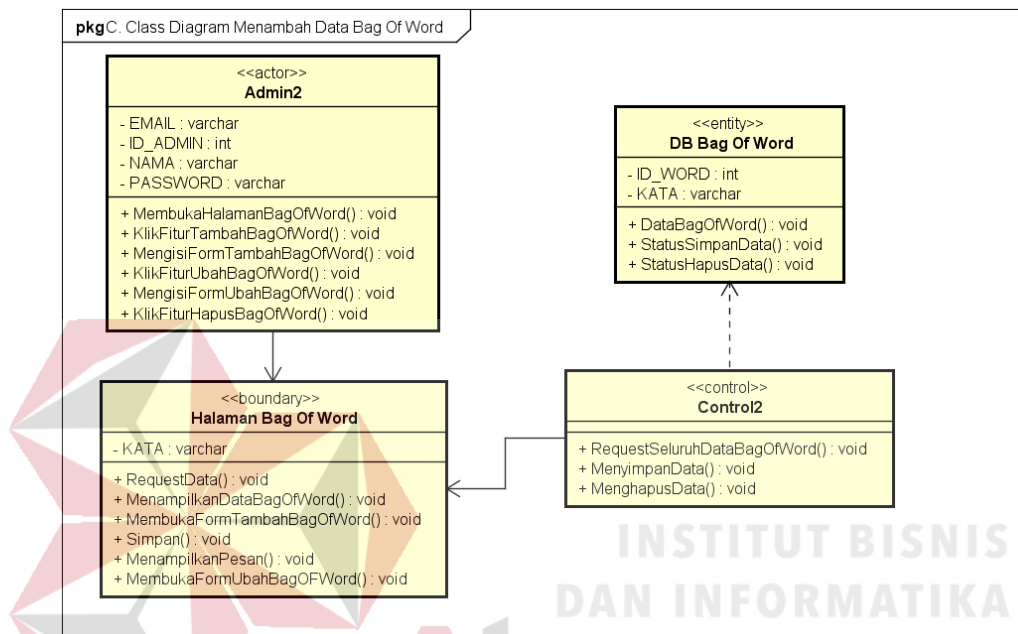
Pada diagram ini menggambarkan alur proses mengelola data *bag of word* lebih detail. Seperti pada Gambar 3.17 berikut.



Gambar 3.17 Sequence Diagram Mengelola Bag Of Word

c. **Class Diagram Mengelola Data Bag Of Word**

Pada diagram ini menggambarkan *attribute*, *operation* dan *relation* yang ada pada setiap *actor*, *boundary*, *control* dan *entity* pada fungsi mengelola data *bag of word* berdasarkan alur di *sequence diagram*. Seperti pada Gambar 3.18 berikut.



Gambar 3.18 Class Diagram Mengelola Bag Of Word

d. **Desain Test Case Mengelola Data Bag of Word**

Desain uji coba pada fungsi mengelola data *bag of word* apakah sudah berjalan sesuai keinginan atau tidak. Desain uji coba fungsi *login admin* dapat dilihat pada Tabel 3.8 berikut.

Tabel 3.8 Desain Test Case Login Admin

| Test Case ID | Tujuan | Input | Output Yang Diharapkan | Status |
|--------------|--|-------------------------|--|--------|
| 1. | Melakukan tambah data <i>bag of word</i> | Kata <i>bag of word</i> | Data berhasil ditambahkan | |
| 2. | Melakukan tambah data <i>bag</i> | Tidak ada input | Muncul pesan : data tidak boleh kosong | |

| Test Case ID | Tujuan | Input | Output Yang Diharapkan | Status |
|--------------|--|-------------------------|--|--------|
| | <i>of word</i> dengan format salah | | | |
| 3. | Melakukan ubah data <i>bag of word</i> | Kata <i>bag of word</i> | Data berhasil diubah | |
| 4. | Melakukan ubah data <i>bag of word</i> dengan format salah | Tidak ada input | Muncul pesan : Data tidak boleh kosong | |

3. Fungsi Mengelola Data Kategori

Fungsi ini digunakan oleh *Admin* untuk dapat mengakses fungsi-fungsi yang ada pada halaman *admin* aplikasi pemenuhan bahan baku UMKM. Penjelasan mengenai perancangan fungsi *login admin* adalah sebagai berikut.

a. Flow Of Event Mengelola Data Kategori

Proses ini dilakukan oleh *admin* untuk dapat masuk kedalam halaman *admin* aplikasi pemenuhan bahan baku UMKM. Seperti pada Tabel 3.9 berikut.

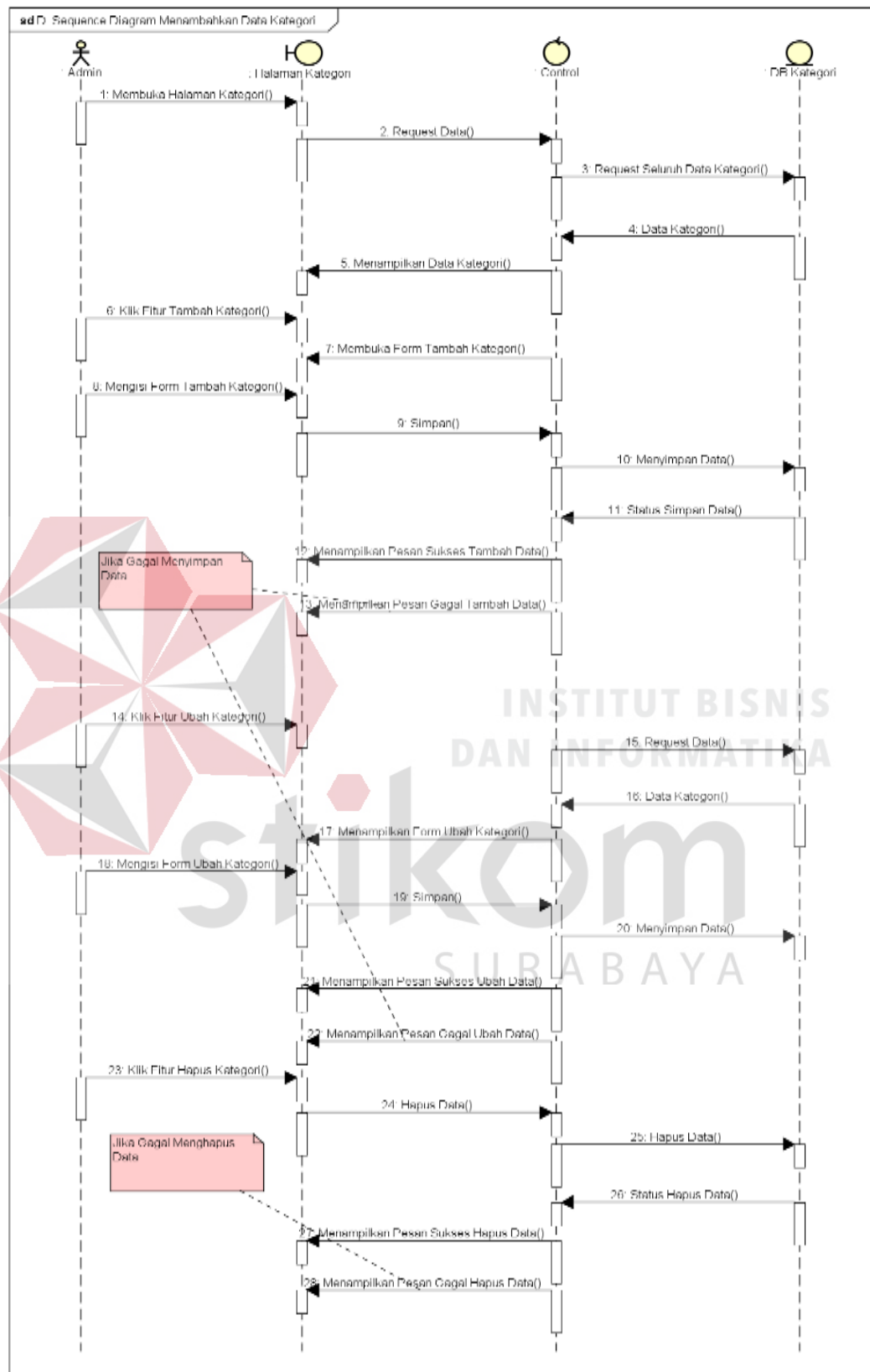
Tabel 3.9 Flow Of Event Mengelola Data Kategori

| Nama Flow Of Event | Mengelola Data Kategori | | |
|-------------------------|--|--|---|
| Aktor | <i>Admin.</i> | | |
| Deskripsi | Pihak aktor dapat mengelola data kategori mulai dari tambah data, ubah data, hingga hapus data kategori. | | |
| Kondisi Awal | Aktor membuka halaman kategori. | | |
| Kondisi Akhir | Sistem menampilkan halaman kategori. | | |
| Aliran Kejadian Utama 1 | No. | Aktor | Sistem |
| | 1. | <i>Admin</i> memilih menu kategori bahan baku | Sistem menampilkan menu data kategori bahan baku dan menampilkan seluruh data kategori bahan baku |
| | 2. | <i>Admin</i> memilih fitur tambah data kategori bahan baku | Sistem menampilkan form tambah data kategori bahan baku |
| | 3. | <i>Admin</i> mengisi data kategori bahan baku | Sistem melakukan validasi isi form jika terisi sistem akan menyimpan data |

| Nama <i>Flow Of Event</i> | Mengelola Data Kategori | | |
|------------------------------|-------------------------|---|--|
| | | | kategori bahan baku dan menampilkan pesan data berhasil ditambahkan |
| Aliran Kejadian Utama 2 | 1. | <i>Admin</i> memilih menu kategori bahan baku | Sistem menampilkan menu data kategori bahan baku dan menampilkan seluruh data kategori bahan baku |
| | 2. | <i>Admin</i> memilih fitur ubah data kategori bahan baku | Sistem menampilkan form data kategori bahan baku yang ingin diubah |
| | 3. | <i>Admin</i> merubah data kategori bahan baku | Sistem melakukan validasi isi form jika terisi sistem akan menyimpan data kategori bahan baku dan menampilkan pesan data berhasil diubah |
| Aliran Kejadian Utama 3 | 1. | <i>Admin</i> memilih menu kategori bahan baku | Sistem menampilkan menu data kategori bahan baku dan menampilkan seluruh data kategori bahan baku |
| | 2. | <i>Admin</i> memilih fitur hapus data kategori bahan baku | Sistem menampilkan pesan konfirmasi hapus data |
| | 3. | <i>Admin</i> menyetujui konfirmasi hapus data | Sistem menghapus data kategori bahan baku yang dipilih dan menampilkan pesan data berhasil diubah |
| Aliran Kejadian Alternatif 1 | 1. | <i>Admin</i> tidak mengisi form tambah data kategori bahan baku | Sistem akan menampilkan pesan data harus di isi dan akan kembali ke halaman kategori bahan baku. |
| Aliran Kejadian Alternatif 2 | 1. | <i>Admin</i> tidak menyetujui konfirmasi hapus data | Sistem akan kembali ke halaman kategori bahan baku. |

b. Sequence Diagram Mengelola Data Kategori

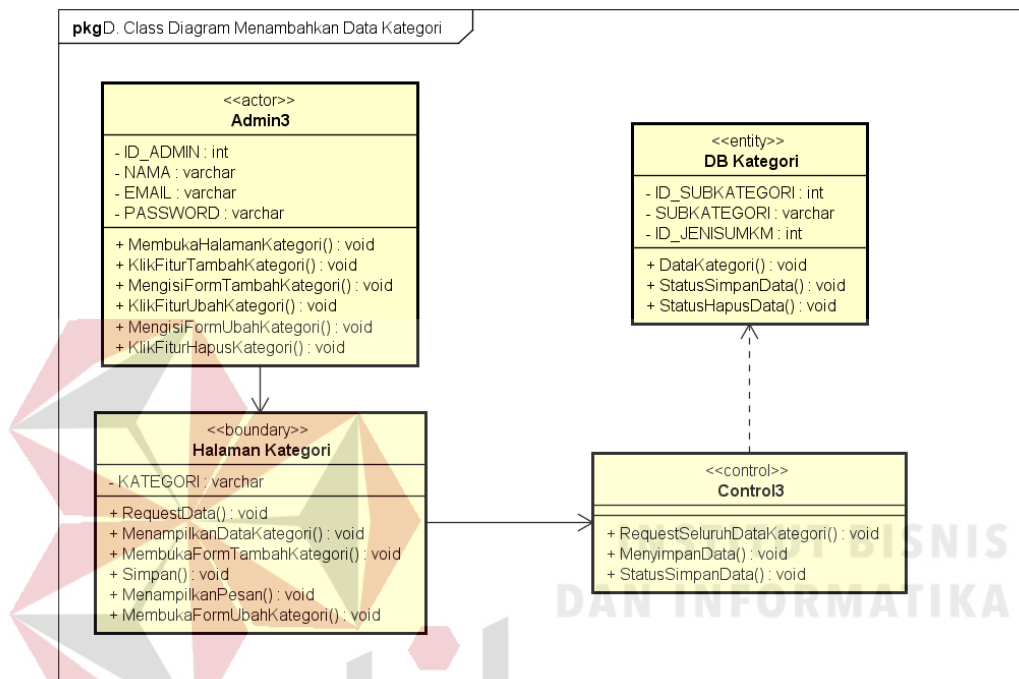
Pada diagram ini menggambarkan alur proses mengelola data kategori lebih detail. Seperti pada Gambar 3.19 berikut.



Gambar 3.19 Sequence Diagram Mengelola Data Kategori

c. Class Diagram Mengelola Data Kategori

Pada diagram ini menggambarkan *attribute*, *operation* dan *relation* yang ada pada setiap *actor*, *boundary*, *control* dan *entity* pada fungsi mengelola data kategori berdasarkan alur di *sequence diagram*. Seperti pada Gambar 3.20 berikut.



Gambar 3.20 Class Diagram Mengelola Data Kategori

d. Desain Test Case Mengelola Data Kategori

Desain uji coba pada fungsi mengelola data kategori apakah sudah berjalan sesuai keinginan atau tidak. Desain uji coba fungsi mengelola kategori dapat dilihat pada Tabel 3.10 berikut.

Tabel 3.10 Desain Test Case Mengelola Data Kategori

| Test Case ID | Tujuan | Input | Output Yang Diharapkan | Status |
|--------------|--------------------------------|---------------------------------|---------------------------|--------|
| 1. | Melakukan tambah data kategori | Jenis UMKM, Kategori Bahan Baku | Data berhasil ditambahkan | |

| Test Case ID | Tujuan | Input | Output Yang Diharapkan | Status |
|--------------|--|---------------------------------|--|--------|
| 2. | Melakukan tambah data kategori dengan format salah | Tidak ada input | Muncul pesan : data tidak boleh kosong | |
| 3. | Melakukan ubah data kategori | Jenis UMKM, Kategori Bahan Baku | Data berhasil diubah | |
| 4. | Melakukan ubah data kategori dengan format salah | Tidak ada input | Muncul pesan : Data tidak boleh kosong | |

4. Fungsi Validasi Pembayaran

Fungsi ini digunakan oleh *admin* untuk dapat melakukan validasi pembelian yang telah dilakukan UMKM dengan melakukan pengecekan pada bukti transfer yang telah *diupload* oleh UMKM pada aplikasi pemenuhan bahan baku UMKM. Penjelasan mengenai perancangan fungsi cek pengguna aktif adalah sebagai berikut.

a. Flow Of Event Validasi Pembayaran

Proses ini dilakukan oleh *admin* untuk dapat melakukan validasi pembelian yang telah dilakukan UMKM dengan melakukan pengecekan pada bukti transfer yang telah *diupload* oleh UMKM. Seperti pada Tabel 3.11 berikut.

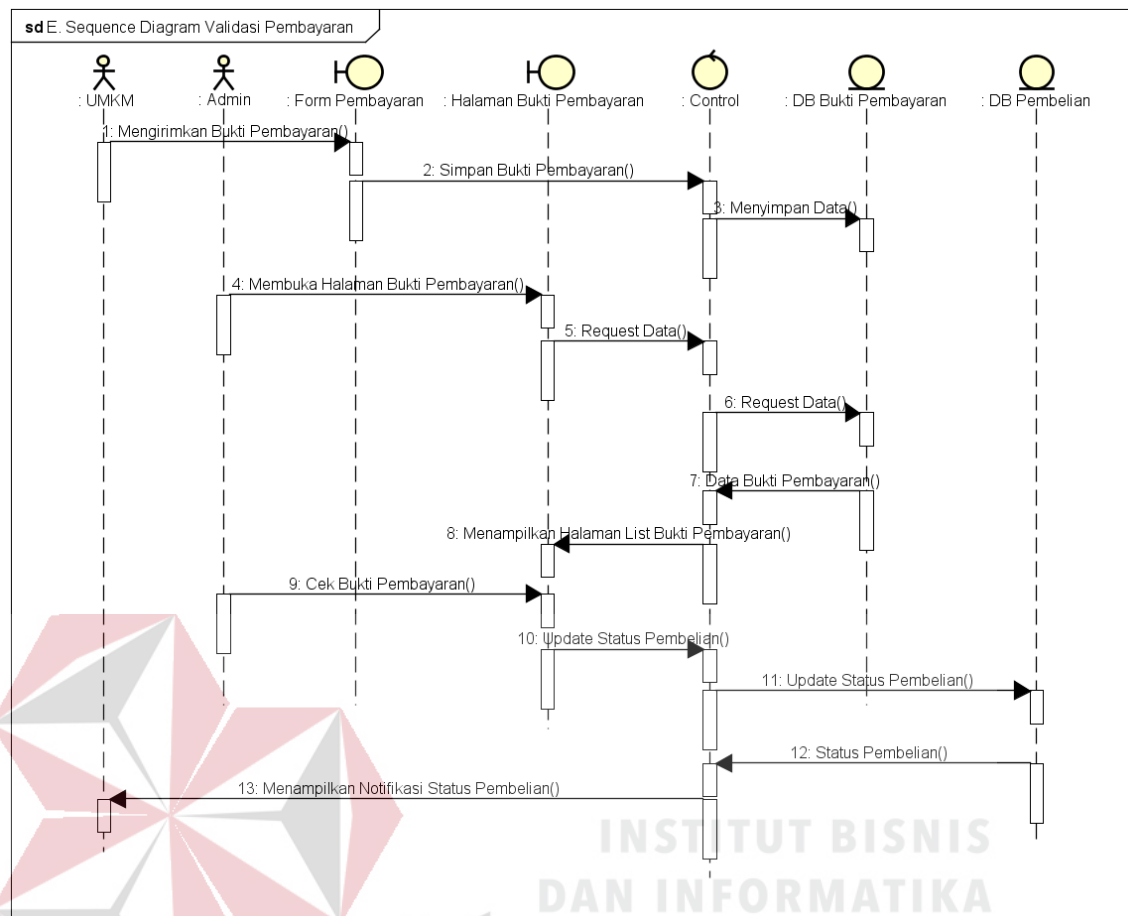
Tabel 3.11 *Flow Of Event* Validasi Pembayaran

| Nama Flow Of Event | Validasi Pembayaran |
|--------------------|--|
| Aktor | <i>Admin.</i> |
| Deskripsi | Pihak aktor dapat melakukan validasi pembelian yang telah dilakukan UMKM dengan melakukan pengecekan pada bukti transfer yang telah <i>diupload</i> oleh UMKM. |
| Kondisi Awal | Aktor membuka halaman bukti pembayaran. |
| Kondisi Akhir | UMKM menerima notifikasi pembayaran berhasil. |

| Nama <i>Flow Of Event</i> | Validasi Pembayaran | | |
|---------------------------|---------------------|---|--|
| | No. | Aktor | Sistem |
| Aliran Kejadian Utama 1 | 1. | UMKM mengirimkan bukti pembayaran | Sistem menyimpan bukti pembayaran dan menampilkan pesan bukti pembayaran berhasil disimpan |
| | | | |
| Aliran Kejadian Utama 2 | 1. | <i>Admin</i> membuka menu validasi pembayaran | Sistem menampilkan menu halaman validasi pembayaran dan menampilkan list bukti pembayaran |
| | 2. | <i>Admin</i> mengecek bukti pembayaran | Sistem menampilkan bukti pembayaran |
| | 3. | <i>Admin</i> menekan tombol valid pada bukti pembayaran | Sistem merubah status pembelian pada data pembelian dan menampilkan notifikasi pembayaran berhasil kepada UMKM |
| Aliran Kejadian Utama 3 | 1. | <i>Admin</i> menekan tombol tidak valid pada bukti pembayaran | Sistem menampilkan notifikasi bukti pembayaran tidak valid kepada UMKM |

b. *Sequence Diagram* Validasi Pembayaran

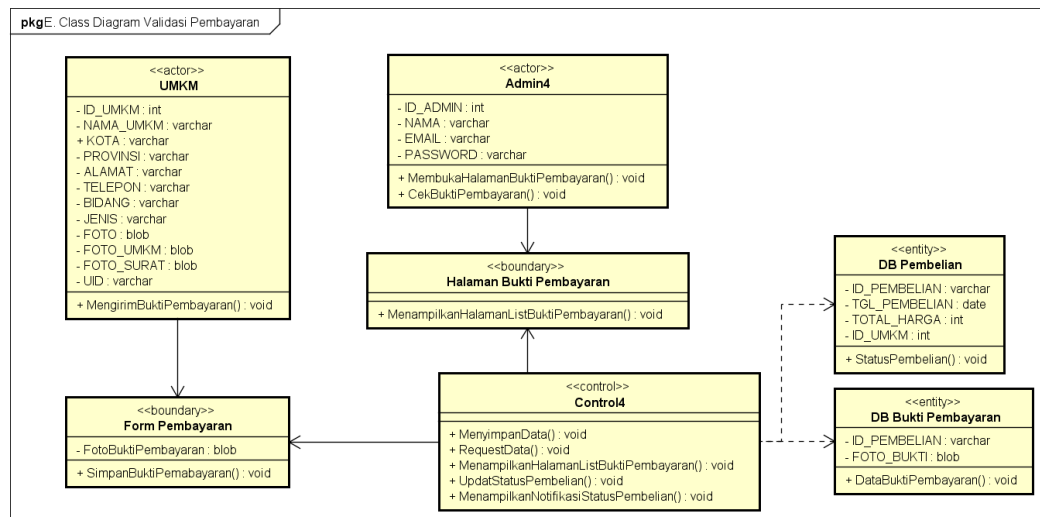
Pada diagram ini menggambarkan alur proses validasi pembayaran lebih detail. Seperti pada Gambar 3.21 berikut.



Gambar 3.21 Sequence Diagram Validasi Pembayaran

c. **Class Diagram Validasi Pembayaran**

Pada diagram ini menggambarkan *attribute*, *operation* dan *relation* yang ada pada setiap *actor*, *boundary*, *control* dan *entity* pada fungsi validasi pembayaran berdasarkan alur di *sequence diagram*. Seperti pada Gambar 3.22 berikut.



Gambar 3.22 Class Diagram Validasi Pembayaran

d. Desain Test Case Validasi Pembayaran

Desain uji coba pada fungsi mengelola data kategori apakah sudah berjalan sesuai keinginan atau tidak. Desain uji coba fungsi mengelola kategori dapat dilihat pada Tabel 3.12 berikut.

Tabel 3.12 Desain Test Case Validasi Pembayaran

| Test Case ID | Tujuan | Input | Output Yang Diharapkan | Status |
|--------------|---|-------------------------|---|--------|
| 1. | Melakukan validasi pembayaran | Klik tombol valid | Sistem menampilkan pesan : validasi pembayaran berhasil | |
| 2. | Melakukan penolakan validasi pembayaran | Klik tombol tidak valid | Sistem menampilkan pesan : validasi pembayaran berhasil | |

5. Fungsi Register UMKM

Fungsi ini digunakan oleh UMKM untuk dapat mengisi data data kelengkap UMKM agar dapat mengakses aplikasi pemenuhan bahan baku UMKM. Penjelasan mengenai perancangan fungsi cek pengguna aktif adalah sebagai berikut.

a. Flow Of Event Register UMKM

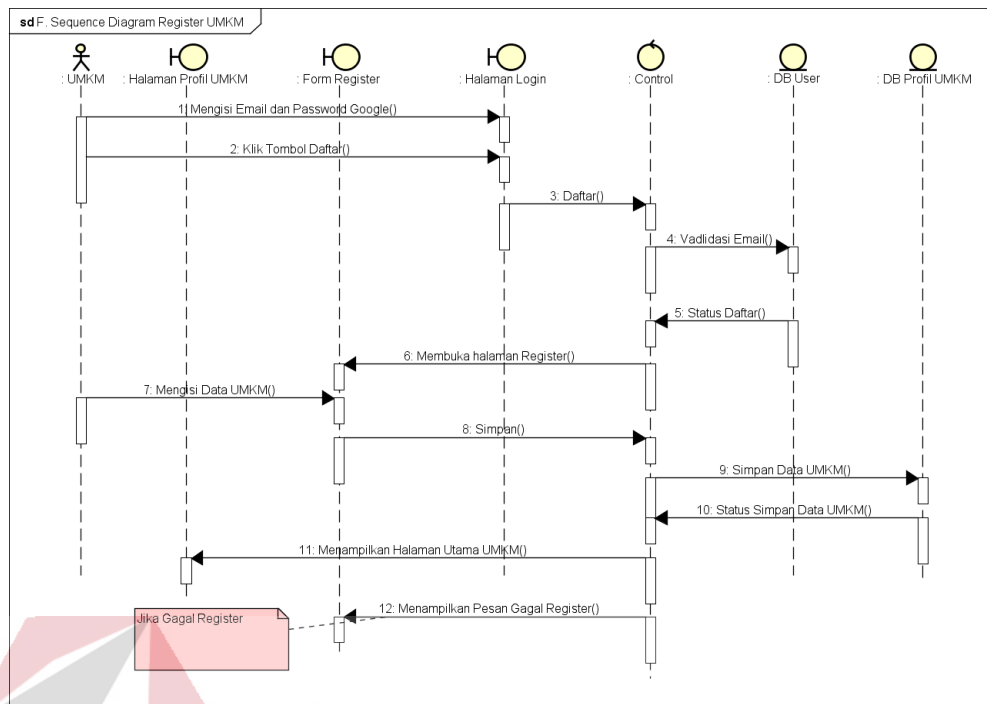
Proses ini dilakukan oleh UMKM untuk dapat mengakses aplikasi pemenuhan bahan baku UMKM, UMKM diharuskan mengisi form biodata UMKM untuk dapat melanjutkan akses pada aplikasi ini. Seperti pada Tabel 3.13 berikut.

Tabel 3.13 *Flow Of Event Register UMKM*

| Nama Flow Of Event | Register UMKM | | |
|-------------------------|---|-------------------------------|---|
| Aktor | UMKM. | | |
| Deskripsi | Pihak aktor dapat melakukan <i>register</i> untuk menggunakan aplikasi pemenuhan bahan baku UMKM dengan menggunakan akun Google mereka. | | |
| Kondisi Awal | Aktor telah melakukan <i>login</i> dengan akun Google. | | |
| Kondisi Akhir | Sistem menampilkan halaman utama UMKM. | | |
| Aliran Kejadian Utama 1 | No. | Aktor | Sistem |
| | 1. | UMKM melakukan <i>login</i> . | Sistem melakukan validasi email apakah sudah terdaftar pada aplikasi. Jika tidak maka sistem akan menampilkan halaman form biodata UMKM |
| | 2. | UMKM mengisi form biodata | Sistem melakukan pengecekan kelengkapan data. Jika ya maka sistem akan menyimpan biodata UMKM dan menampilkan |
| Aliran Kejadian Utama 2 | 1. | UMKM melakukan <i>login</i> . | Sistem melakukan validasi email apakah sudah pada aplikasi. Jika iya maka sistem akan menampilkan halaman utama UMKM |

b. Sequence Diagram Register UMKM

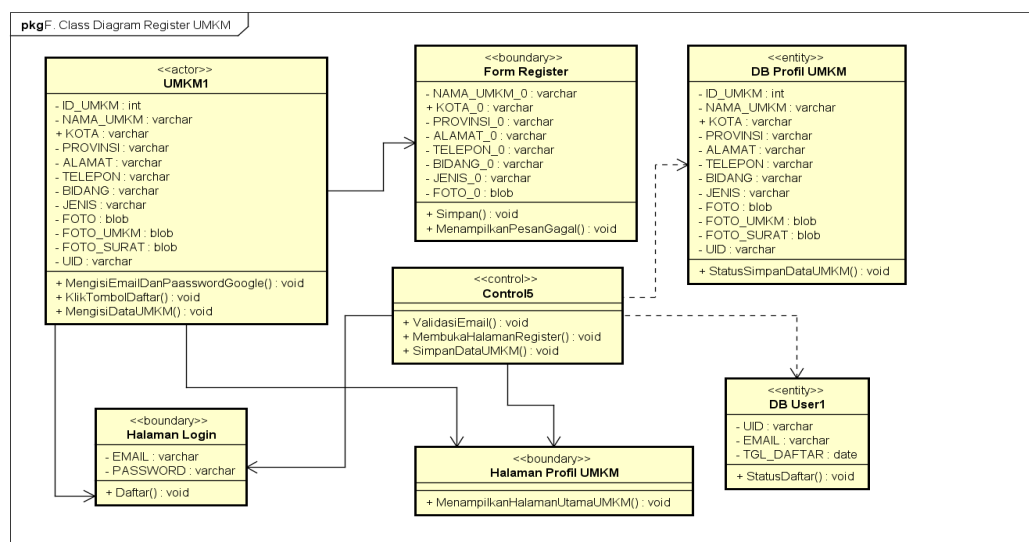
Pada diagram ini menggambarkan alur proses *register* UMKM lebih detail. Seperti pada Gambar 3.23 berikut.



Gambar 3.23 Sequence Diagram Register UMKM

c. Class Diagram Register UMKM

Pada diagram ini menggambarkan *attribute*, *operation* dan *relation* yang ada pada setiap *actor*, *boundary*, *control* dan *entity* pada fungsi *register* UMKM berdasarkan alur di *sequence diagram*. Seperti pada Gambar 3.24 berikut.



Gambar 3.24 Class Diagram Register UMKM

d. **Desain *Test Case Register* UMKM**

Desain uji coba pada fungsi registrer UMKM apakah sudah berjalan sesuai keinginan atau tidak. Desain uji coba fungsi *register* UMKM dapat dilihat pada Tabel 3.14 berikut.

Tabel 3.14 Desain *Test Case Register* UMKM

| Test Case ID | Tujuan | Input | Output Yang Diharapkan | Status |
|--------------|---|--|--|--------|
| 1. | Melakukan <i>register</i> | Nama_Ukm, Provinsi, Alamat, Kota, Telepon, Bidang, Jenis Ukm, Foto, Foto_Ukm, Foto_Surat, | Registrasi UMKM berhasil dan sistem menampilkan halaman utama UMKM | |
| 2. | Melakukan <i>register</i> dengan format salah | Tidak ada input | Muncul pesan : data tidak boleh kosong | |

6. **Fungsi *Login* UMKM**

Fungsi ini digunakan oleh UMKM untuk dapat mengakses fungsi-fungsi yang ada pada halaman UMKM aplikasi pemenuhan bahan baku UMKM. Penjelasan mengenai perancangan fungsi *login* UMKM adalah sebagai berikut.

a. ***Flow Of Event Login* UMKM**

Proses ini dilakukan oleh UMKM untuk dapat masuk kedalam halaman UMKM aplikasi pemenuhan bahan baku UMKM. Seperti pada Tabel 3.15 berikut.

Tabel 3.15 *Flow Of Event Login* UMKM

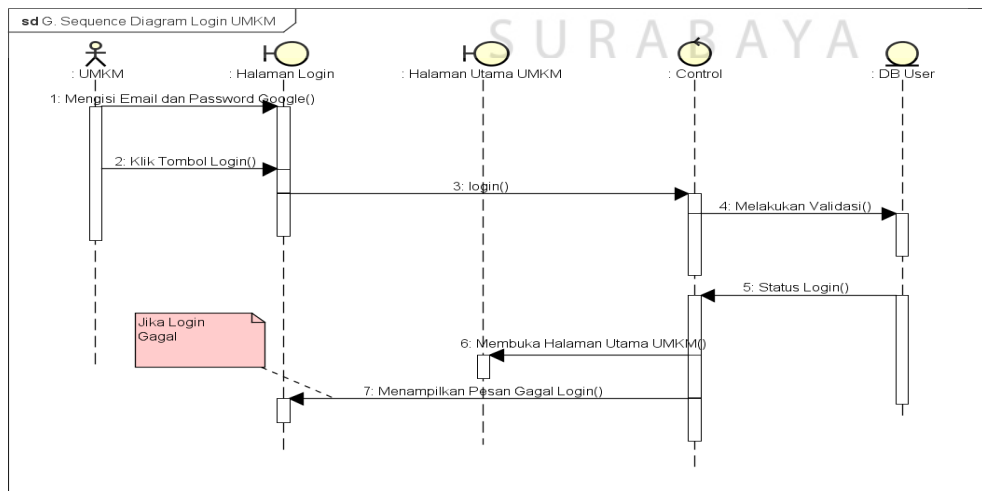
| Nama Flow Of Event | <i>Login</i> UMKM |
|--------------------|--|
| Aktor | UMKM. |
| Deskripsi | Pihak aktor dapat melakukan proses <i>login</i> ini untuk dapat mengakses fungsi-fungsi yang ada pada halaman UMKM aplikasi pemenuhan bahan baku UMKM. |

| Nama Flow Of Event | | Login UMKM | |
|----------------------------|-----|---|---|
| Kondisi Awal | | aktor mempunyai <i>email</i> dan <i>password</i> Akun Google. | |
| Kondisi Akhir | | Sistem menampilkan halaman utama UMKM. | |
| Aliran Kejadian Utama 1 | No. | Aktor | Sistem |
| | 1. | UMKM memasukkan email dan password google yang dimiliki | Sistem melakukan validasi email dan password google, jika benar sistem akan menampilkan halaman utama UMKM. |
| Aliran Kejadian Alternatif | 1. | UMKM salah memasukkan email dan password google yang dimiliki | Sistem akan menampilkan pesan error dan akan kembali ke halaman <i>login</i> . |

b. Sequence Diagram Login UMKM

Pada diagram ini menggambarkan alur proses *login* UMKM lebih detail.

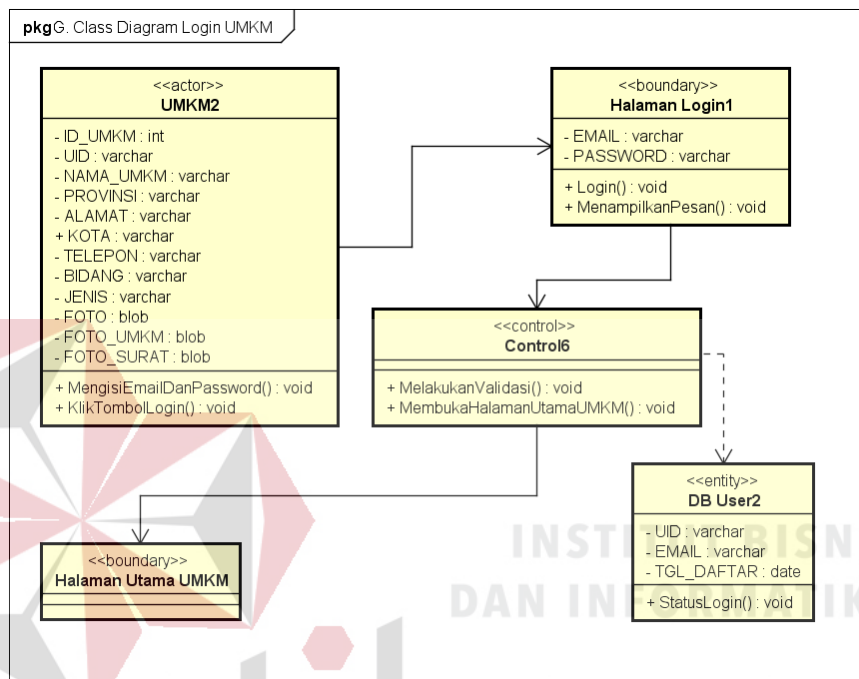
Seperti pada Gambar 3.25 berikut.



Gambar 3.25 Sequence Diagram Login UMKM

c. **Class Diagram Login UMKM**

Pada diagram ini menggambarkan *attribute*, *operation* dan *relation* yang ada pada setiap *actor*, *boundary*, *control* dan *entity* pada fungsi *login* UMKM berdasarkan alur di *sequence diagram*. Seperti pada Gambar 3.26 berikut.



Gambar 3.26 Class Diagram Login UMKM

d. **Desain Test Case Login UMKM**

Desain uji coba pada fungsi *login* UMKM apakah sudah berjalan sesuai keinginan atau tidak. Desain uji coba fungsi *login* UMKM dapat dilihat pada Tabel 3.16 berikut.

Tabel 3.16 Desain Test Case Login UMKM

| Test Case ID | Tujuan | Input | Output Yang Diharapkan | Status |
|--------------|-----------------------------|---------------------------------|---|--------|
| 1. | Melakukan <i>login</i> UMKM | Email dan password <i>admin</i> | <i>Login</i> berhasil dan sistem menampilkan halaman utama UMKM | |

| Test Case ID | Tujuan | Input | Output Yang Diharapkan | Status |
|--------------|--|-------------------------------|--|--------|
| 2. | Melakukan <i>login</i> dengan format salah | Tidak ada input | Muncul pesan : email dan password tidak boleh kosong | |
| 3. | Melakukan <i>login</i> dengan format salah | Email dan password yang salah | Muncul pesan : email atau password salah | |

7. Fungsi Menambahkan *Timeline Portofolio*

Fungsi ini digunakan oleh UMKM untuk dapat melakukan posting aktivitas-aktivitasnya. Penjelasan mengenai perancangan fungsi cek pengguna aktif adalah sebagai berikut.

a. *Flow Of Event Mengisi Timeline Portofolio*

Proses ini dilakukan oleh UMKM untuk dapat melakukan posting aktivitas-aktivitasnya. Seperti pada Tabel 3.17 berikut.

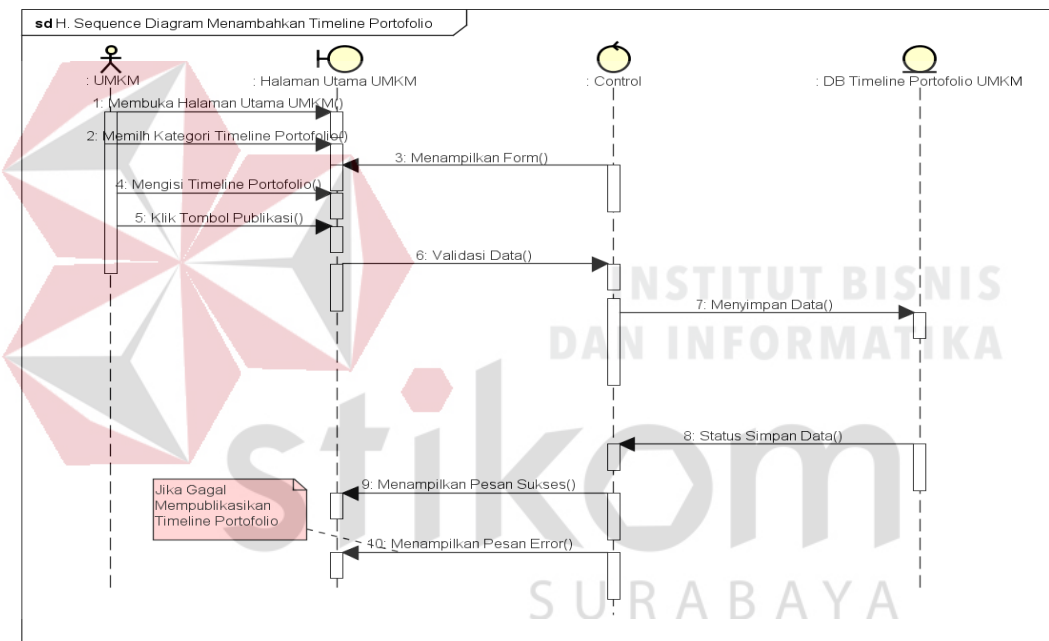
Tabel 3.17 *Flow Of Event Mengisi Timeline Portofolio*

| Nama Flow Of Event | Mengisi <i>Timeline Portofolio</i> | | |
|-----------------------|---|---|---|
| Aktor | UMKM. | | |
| Deskripsi | Pihak aktor dapat melakukan posting aktivitas-aktivitasnya. | | |
| Kondisi Awal | Aktor membuka halaman utama UMKM. | | |
| Kondisi Akhir | Sistem akan menampilkan postingan aktor pada halaman utama. | | |
| Aliran Kejadian Utama | No | User | Sistem |
| | 1. | UMKM memilih menu dashboard | System menampilkan menu dashboard |
| | 2. | UMKM memilih kategori <i>timeline portofolio</i> bebas yang akan ditambahkan | Sistem menampilkan <i>form timeline portofolio</i> kategori bebas |
| | 3. | UMKM memasukkan <i>timeline portofolio</i> pada <i>form timeline portofolio</i> . | Sistem memvalidasi data apakah <i>timeline portofolio</i> tersebut telah ada di database atau belum. Jika berhasil data akan disimpan dan UMKM akan |

| Nama <i>Flow Of Event</i> | Mengisi <i>Timeline Portofolio</i> | | |
|---------------------------|------------------------------------|--|---|
| | | | mendapatkan poin dari kegiatan menambahkan <i>timeline portofolio</i> . Sistem juga akan menampilkan pesan sukses |

b. Sequence Diagram Mengisi *Timeline Portofolio*

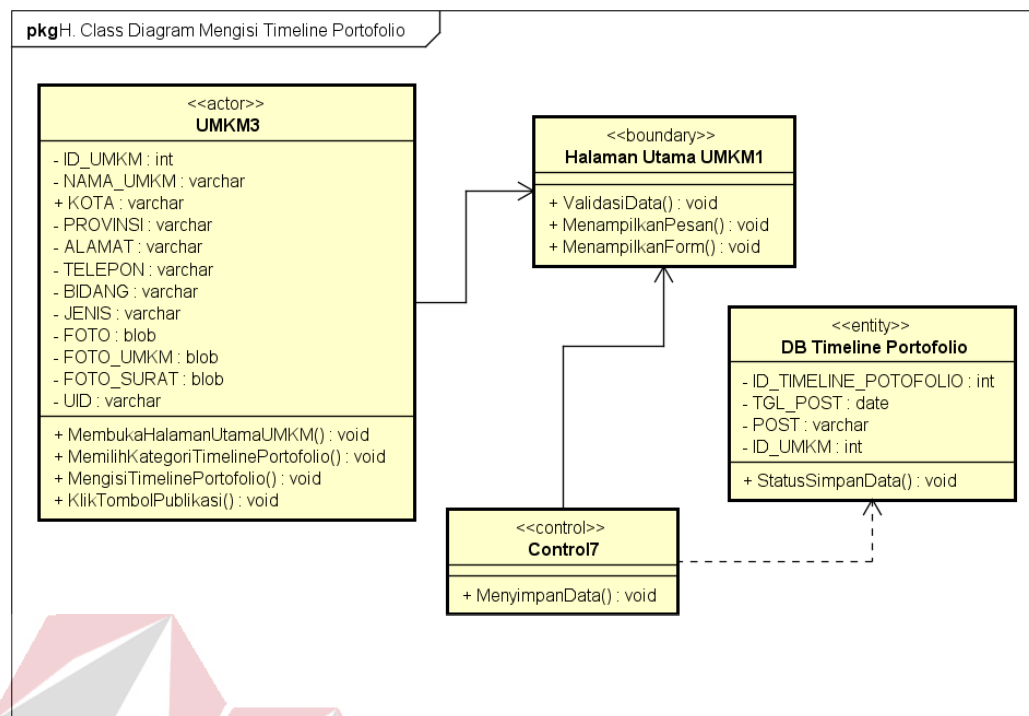
Pada diagram ini menggambarkan alur proses mengisi *timeline portofolio* lebih detail. Seperti pada Gambar 3.27 berikut.



Gambar 3.27 Sequence Diagram Mengisi *Timeline Portofolio*

c. Class Diagram Mengisi *Timeline Portofolio*

Pada diagram ini menggambarkan *attribute*, *operation* dan *relation* yang ada pada setiap *actor*, *boundary*, *control* dan *entity* pada fungsi mengisi *timeline portofolio* berdasarkan alur di *sequence diagram*. Seperti pada Gambar 3.28 berikut.



Gambar 3.28 Class Diagram Mengisi Timeline Portofolio

d. Desain Test Case Mengisi Timeline Portofolio

Desain uji coba pada fungsi mengisi *timeline portofolio* apakah sudah berjalan sesuai keinginan atau tidak. Desain uji coba mengisi *timeline portofolio* dapat dilihat pada Tabel 3.18 berikut.

Tabel 3.18 Desain Test Case Mengisi Timeline Portofolio

| Test Case ID | Tujuan | Input | Output Yang Diharapkan | Status |
|--------------|--|-----------------|--|--------|
| 1. | Mengisi <i>timeline portofolio</i> | Status | Pengisian berhasil dan sistem menampilkan status yang telah dibuat | |
| 2. | Mengisi <i>timeline portofolio</i> dengan format salah | Tidak ada input | Muncul pesan : status tidak boleh kosong | |

8. Fungsi Melihat Klasifikasi Status *Timeline Portofolio*

Fungsi ini digunakan oleh secara otomatis untuk menentukan kekurangan atau kelebihan bahan baku yang dimiliki oleh UMKM berdasarkan *timeline portofolio* UMKM dengan penggunaan beberapa algoritma yaitu, *text mining*, *Naïve bayes classifier*, dan *Boyer Moore* pada aplikasi pemenuhan bahan baku UMKM. Penjelasan mengenai perancangan fungsi cek pengguna aktif adalah sebagai berikut.

a. *Flow Of Event* Melihat Klasifikasi Status *Timeline Portofolio*

Proses ini dilakukan oleh sistem secara otomatis untuk menentukan kekurangan atau kelebihan bahan baku yang dimiliki oleh UMKM berdasarkan *timeline portofolio* UMKM dengan penggunaan beberapa algoritma yaitu, *text mining*, *Naïve bayes classifier*, dan *Boyer Moore* pada aplikasi pemenuhan bahan baku UMKM. Seperti pada Tabel 3.19 berikut.

Tabel 3.19 *Flow Of Event* Melihat Klasifikasi Status *Timeline Portofolio*

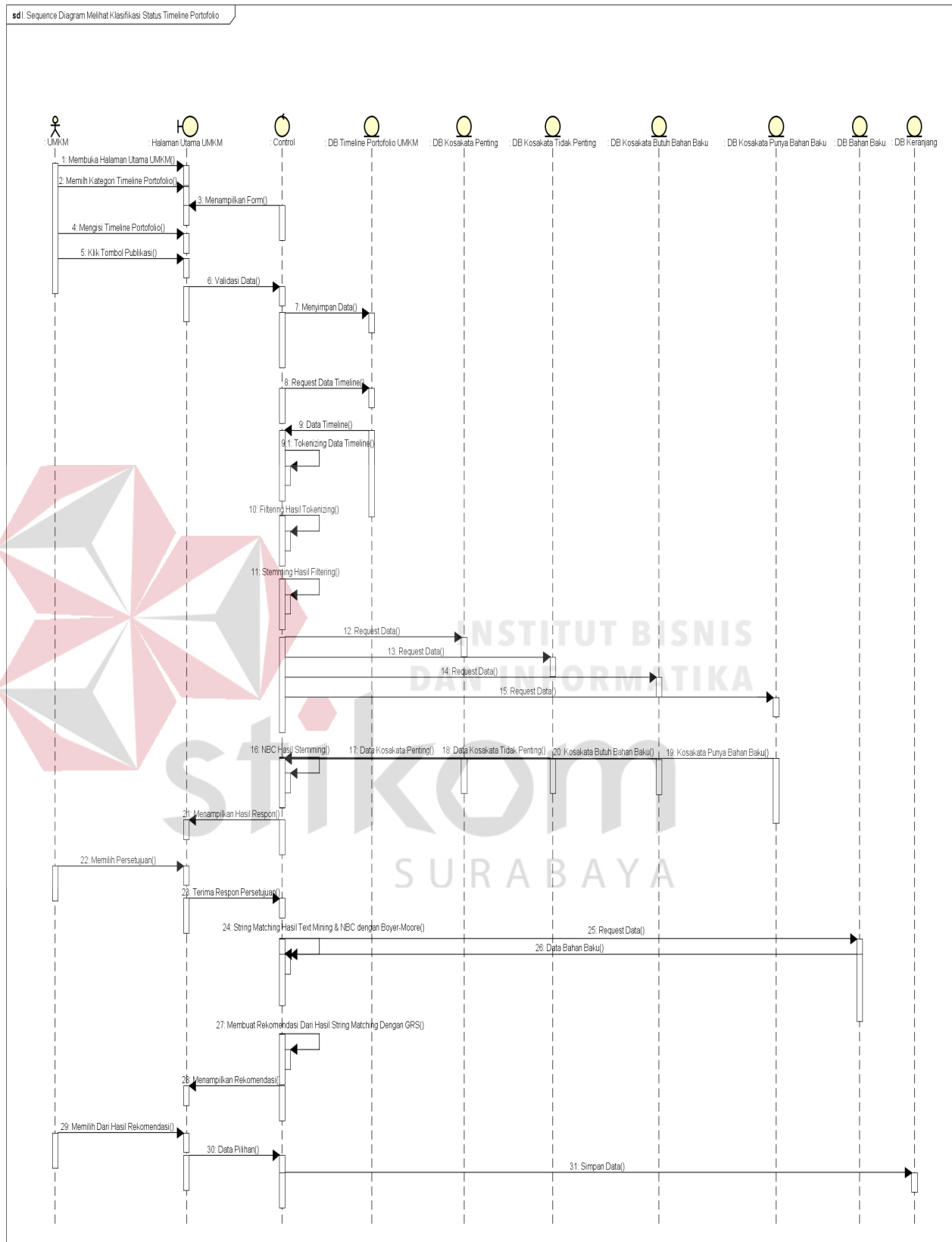
| Nama <i>Flow Of Event</i> | | Melihat Klasifikasi Status <i>Timeline Portofolio</i> | |
|--------------------------------|-----------|--|--|
| Aktor | | UMKM. | |
| Deskripsi | | Sistem dapat secara otomatis untuk menentukan kekurangan atau kelebihan bahan baku yang dimiliki oleh aktor berdasarkan <i>timeline portofolio</i> UMKM dengan penggunaan beberapa algoritma yaitu, <i>text mining</i> , <i>Naïve bayes classifier</i> , dan <i>Boyer Moore</i> pada aplikasi pemenuhan bahan baku UMKM. | |
| Kondisi Awal | | Aktor melakukan posting aktivitas UMKMnya. | |
| Kondisi Akhir | | <ol style="list-style-type: none"> 1. Sistem menampilkan hasil rekomendasi bahan baku yang sesuai dengan kebutuhan UMKM. 2. Sistem menampilkan halaman form untuk UMKM menjual bahan bakunya. | |
| Aliran Kejadian Utama 1 | No | User | Sistem |
| | 1. | UMKM melakukan posting <i>timeline portofolio</i> | Sistem melakukan validasi data posting <i>timeline portofolio</i> . Jika ya maka |

| Nama <i>Flow Of Event</i> | Melihat Klasifikasi Status <i>Timeline Portofolio</i> | | |
|---------------------------|---|---|--|
| | | | sistem akan menyimpan data <i>timeline portofolio</i> |
| | 2. | | Sistem melakukan proses <i>tokenizing, filtering</i> dan <i>stemming</i> pada data posting <i>timeline portofolio</i> |
| | 3. | | Sistem melakukan proses <i>Naïve-Bayes Classifier</i> pada hasil <i>stemming</i> . Jika hasilnya adalah butuh maka sistem akan melanjutkan ke proses <i>String Matching</i> |
| | 4. | | Sistem melakukan proses <i>string matching</i> dengan bahan baku yang dijual UMKM lain menggunakan <i>algoritma Boyer-Moore</i> |
| | 5. | | Jika bahan baku ditemukan maka sistem akan melakukan proses <i>Graphic Rating Scale</i> untuk menentukan rekomendasi. Dan sistem akan menampilkan rekomendasi bahan baku kepada UMKM |
| | 6. | UMKM memilih bahan baku dari hasil rekomendasi | Sistem akan menyimpan data bahan baku kedalam keranjang untuk dilakukan proses pembelian |
| Alur Kegiatan Utama 2 | 1. | UMKM melakukan posting <i>timeline portofolio</i> | Sistem melakukan validasi data posting <i>timeline portofolio</i> . Jika ya maka sistem akan menyimpan data <i>timeline portofolio</i> |
| | 2. | | Sistem melakukan proses <i>tokenizing, filtering</i> dan <i>stemming</i> pada data posting <i>timeline portofolio</i> |
| | 3. | | Sistem melakukan proses <i>Naïve-Bayes Classifier</i> pada hasil <i>stemming</i> . |
| | 4. | | Jika hasilnya adalah kelebihan maka sistem akan menampilkan form tambah bahan baku yang ingin dijual |

| Nama <i>Flow Of Event</i> | Melihat Klasifikasi Status <i>Timeline Portofolio</i> | | |
|---------------------------|---|---|---|
| Alur Kegiatan Utama 3 | 1. | UMKM melakukan posting <i>timeline portofolio</i> | Sistem melakukan validasi data posting <i>timeline portofolio</i> . Jika ya maka sistem akan menyimpan data <i>timeline portofolio</i> |
| | 2. | | Sistem melakukan proses <i>tokenizing, filtering</i> dan <i>stemming</i> pada data posting <i>timeline portofolio</i> |
| | 3. | | Sistem melakukan proses <i>Naïve-Bayes Classifier</i> pada hasil <i>stemming</i> . Jika hasilnya adalah butuh maka sistem akan melanjutkan ke proses <i>String Matching</i> |
| | 4. | | Sistem melakukan proses <i>string matching</i> dengan bahan baku yang dijual UMKM lain menggunakan <i>algoritma Boyer-Moore</i> |
| | 5. | | Jika bahan baku tidak ditemukan maka sistem akan menampilkan pesan bahan baku tidak ditemukan kepada UMKM |

b. Sequence Diagram Melihat Klasifikasi Status *Timeline Portofolio*

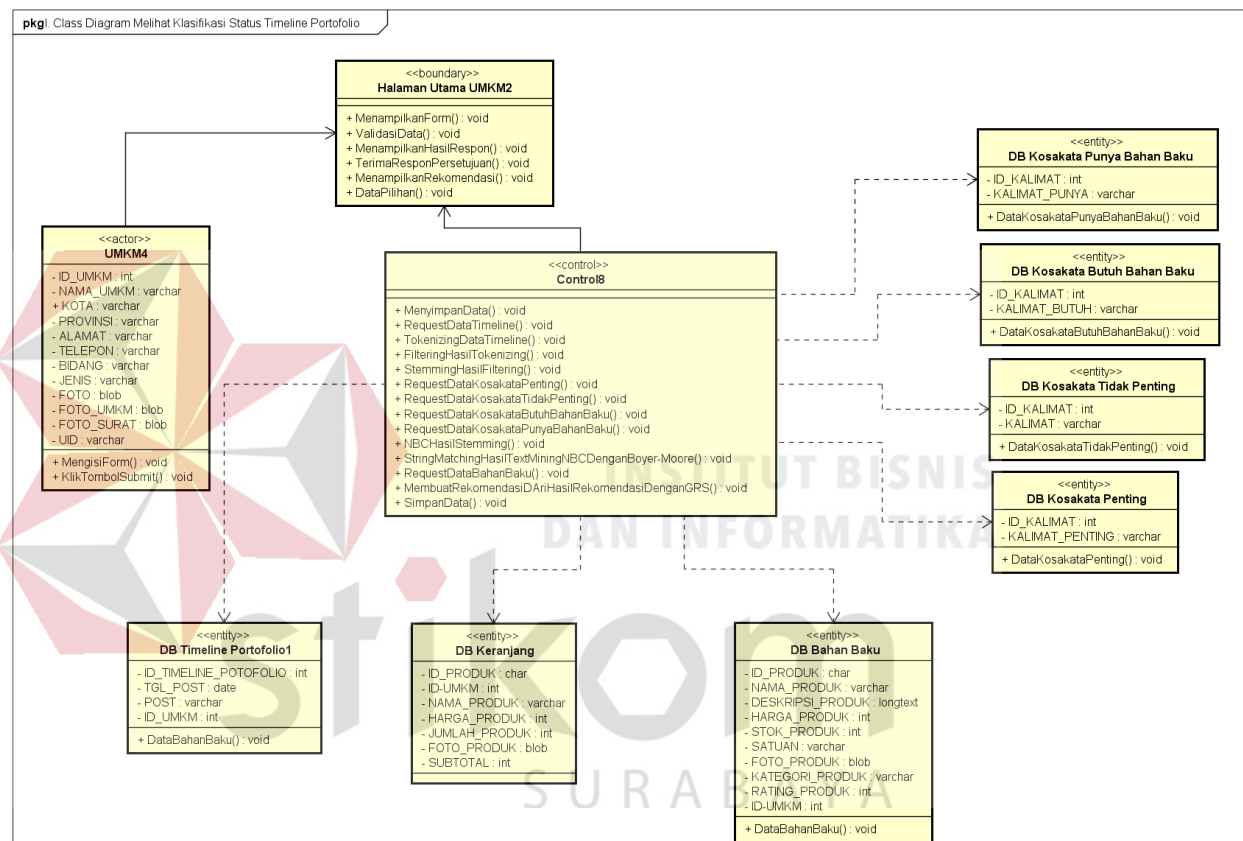
Pada diagram ini menggambarkan alur proses melihat klasifikasi status *timeline portofolio* lebih detail. Seperti pada Gambar 3.29 berikut.



Gambar 3.29 Sequence Diagram Melihat Klasifikasi Status Timeline Portofolio

c. **Class Diagram Melihat Klasifikasi Status *Timeline Portfolio***

Pada diagram ini menggambarkan *attribute*, *operation* dan *relation* yang ada pada setiap *actor*, *boundary*, *control* dan *entity* pada fungsi secara otomatis untuk menentukan kekurangan atau kelebihan bahan baku yang dimiliki berdasarkan alur di *sequence diagram*. Seperti pada Gambar 3.30 berikut.



Gambar 3.30 Class Diagram Melihat Klasifikasi Status *Timeline Portfolio*

d. **Desain Test Case Melihat Klasifikasi Status *Timeline Portfolio***

Desain uji coba pada fungsi klasifikasi status *timeline portfolio* apakah sudah berjalan sesuai keinginan atau tidak. Pengujian pada algoritma naïve bayes classifier hanya menggunakan dataset 30 dokumen yang dapat dilihat pada halaman

lampiran 3. Desain uji coba fungsi melihat klasifikasi status *timeline portofolio* dapat dilihat pada Tabel 3.20 berikut.

Tabel 3.20 Desain *Test Case* Melihat Klasifikasi Status *Timeline Potofolio*

| Test Case ID | Tujuan | Input | Output Yang Diharapkan | Status |
|--------------|---|-------------------|---|--------|
| 1. | Melakukan klasifikasi status <i>timeline portofolio</i> | Status | Klasifikasi berhasil dan sistem menampilkan rekomendasi bahan baku | |
| 2. | Melakukan klasifikasi status <i>timeline portofolio</i> dengan format salah | Tidak ada input | Menampilkan pesan : status tidak boleh kosong | |
| 3. | Melakukan klasifikasi status <i>timeline portofolio</i> dengan format salah | Status yang salah | Klasifikasi gagal dan sistem tidak menampilkan rekomendasi bahan baku | |

9. Fungsi Pembelian Bahan Baku

Fungsi ini digunakan oleh UMKM untuk dapat melihat katalog bahan baku dan membeli bahan baku mereka inginkan pada aplikasi pemenuhan bahan baku UMKM. Penjelasan mengenai perancangan fungsi pembelian bahan baku adalah sebagai berikut.

a. *Flow Of Event* Pembelian Bahan Baku

Proses ini dilakukan oleh UMKM untuk dapat melihat katalog bahan baku dan membeli bahan baku mereka inginkan pada aplikasi pemenuhan bahan baku UMKM. Seperti pada Tabel 3.21 berikut.

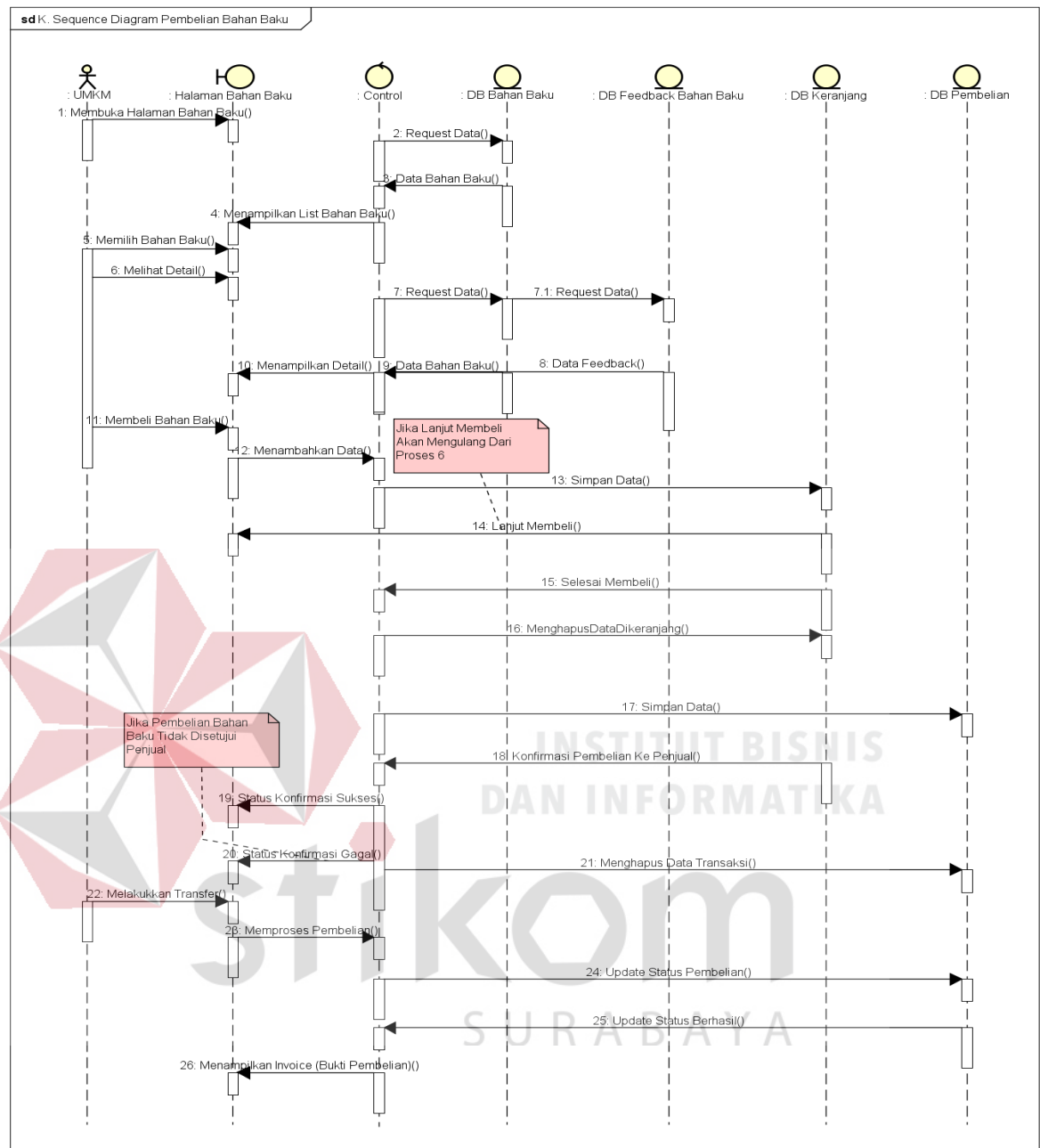
Tabel 3.21 *Flow Of Event* Pembelian Bahan Baku

| Nama Flow Of Event | | Pembelian Bahan Baku | |
|-------------------------|-----|--|---|
| Aktor | | UMKM. | |
| Deskripsi | | Pihak aktor melihat katalog bahan baku dan membeli bahan baku mereka inginkan pada aplikasi pemenuhan bahan baku UMKM. | |
| Kondisi Awal | | Aktor membuka halaman katalog bahan baku. | |
| Kondisi Akhir | | Sistem menampilkan <i>invoice</i> pembelian bahan baku. | |
| Aliran Kejadian Utama | No. | Aktor | Sistem |
| | 1. | | Sistem menampilkan katalog bahan baku |
| | 2. | UMKM memilih bahan baku yang ingin dibeli dengan menekan tombol beli | Sistem akan menambahkan bahan baku kedalam keranjang |
| | 3. | UMKM membuka halaman keranjang | Sistem menampilkan halaman keranjang |
| | 4. | UMKM menekan tombol selesai belanja | Sistem melakukan proses pembelian dan menampilkan notifikasi kepada UMKM penjual konfirmasi penjualan bahan baku. |
| | 5. | | Jika iya maka sistem akan menampilkan notifikasi kepada UMKM pembeli untuk melakukan transfer |
| Aliran Kejadian Utama 2 | 6. | UMKM melakukan transfer dan melakukan <i>upload</i> bukti transfer | Jika bukti transfer telah di validasi maka sistem akan menampilkan <i>invoice</i> (Bukti Pembelian). |
| | 1. | | Sistem menampilkan katalog bahan baku |
| | 2. | UMKM memilih bahan baku yang ingin dibeli dengan menekan tombol detail bahan baku | Sistem akan menampilkan halaman detail bahan baku |
| | 3. | UMKM memilih bahan baku yang ingin dibeli dengan menekan tombol beli | Sistem akan menambahkan bahan baku kedalam keranjang |
| | 4. | UMKM membuka halaman keranjang | Sistem menampilkan halaman keranjang |
| | 5. | UMKM menekan tombol selesai belanja | Sistem melakukan proses pembelian dan menampilkan notifikasi |

| Nama <i>Flow Of Event</i> | Pembelian Bahan Baku | | |
|---------------------------|----------------------|--|--|
| | | | kepada UMKM penjual konfirmasi penjualan bahan baku. |
| | 6. | | Jika iya maka sistem akan menampilkan notifikasi kepada UMKM pembeli untuk melakukan transfer |
| | 7. | UMKM melakukan transfer dan melakukan <i>upload</i> bukti transfer | Jika bukti transfer telah di validasi maka sistem akan menampilkan <i>invoice</i> (Bukti Pembelian). |
| Aliran Kejadian Utama 3 | 1. | | Sistem menampilkan katalog bahan baku |
| | 2. | UMKM memilih bahan baku yang ingin dibeli dengan menekan tombol beli | Sistem akan menambahkan bahan baku kedalam keranjang |
| | 3. | UMKM membuka halaman keranjang | Sistem menampilkan halaman keranjang |
| | 4. | UMKM menekan tombol lanjutan belanja | Sistem menampilkan halaman katalog bahan baku. |

b. **Sequence Diagram Pembelian Bahan Baku**

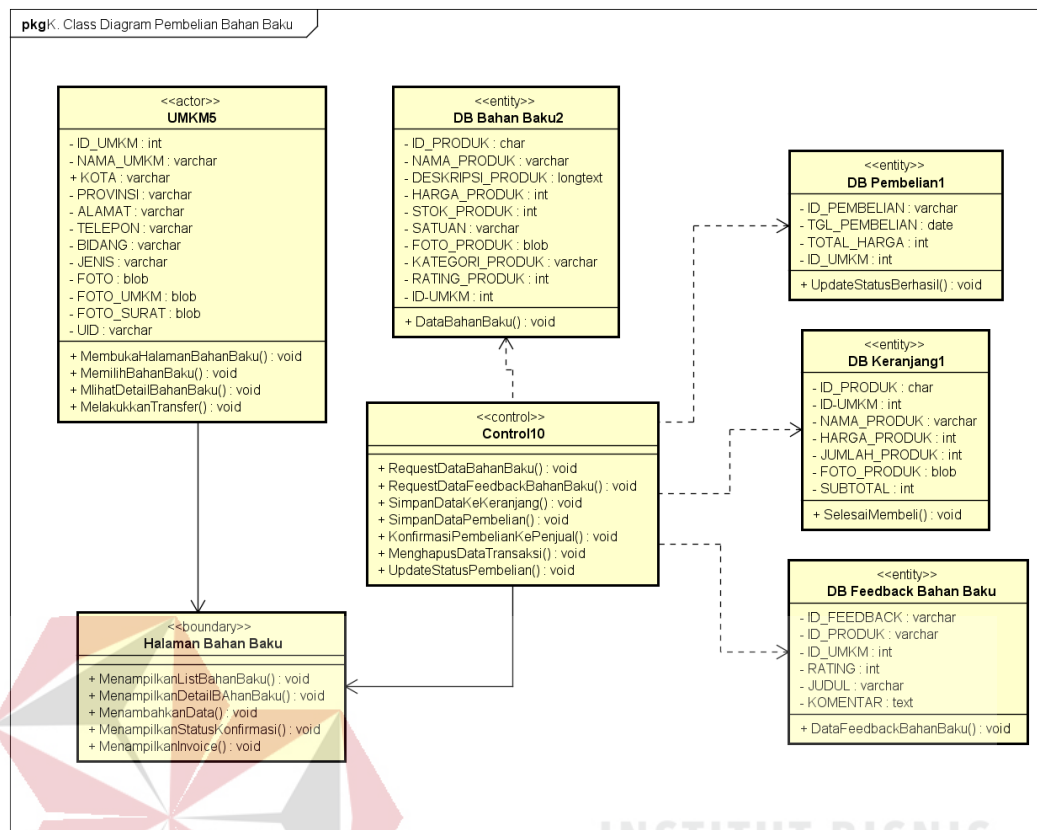
Pada diagram ini menggambarkan alur proses pembelian bahan baku lebih detail. Seperti pada Gambar 3.31.



Gambar 3.31 Sequence Diagram Pembelian Bahan Baku

c. Class Diagram Pembelian Bahan Baku

Pada diagram ini menggambarkan *attribute*, *operation* dan *relation* yang ada pada setiap *actor*, *boundary*, *control* dan *entity* pada fungsi pembelian bahan baku berdasarkan alur di *sequence diagram*. Seperti pada Gambar 3.32 berikut.



Gambar 3.32 Class Diagram Pembelian Bahan Baku

d. Desain Test Case Pembelian Bahan Baku

Desain uji coba pada fungsi pembelian bahan baku apakah sudah berjalan sesuai keinginan atau tidak. Desain uji coba fungsi pembelian bahan baku dapat dilihat pada Tabel 3.22 berikut.

Tabel 3.22 Desain Test Case Pembelian Bahan Baku

| Test Case ID | Tujuan | Input | Output Yang Diharapkan | Status |
|--------------|--|----------------------------------|---|--------|
| 1. | Melihat detail pembelian | Menekan tombol detail bahan baku | Sistem menampilkan halaman detail bahan baku | |
| 2. | Melanjutkan belanja setelah memasukkan bahan baku ke keranjang | Menekan tombol lanjutkan belanja | Sistem menampilkan halaman katalog bahan baku | |

| Test Case ID | Tujuan | Input | Output Yang Diharapkan | Status |
|--------------|------------------------------------|--------------------------------|--|--------|
| 3. | Menyelesaikan pembelian bahan baku | Menekan tombol selesai belanja | Sistem menampilkan <i>pop up</i> pembelian sedang diproses | |

10. Fungsi Review Bahan Baku

Fungsi ini digunakan oleh UMKM untuk dapat memberikan *review/feedback* terhadap bahan baku yang telah mereka beli sebelumnya. Penjelasan mengenai perancangan fungsi cek pengguna aktif adalah sebagai berikut.

a. Flow Of Event Review Bahan Baku

Proses ini dilakukan oleh UMKM untuk dapat memberikan *review/feedback* terhadap bahan baku yang telah mereka beli sebelumnya. Seperti pada Tabel 3.23 berikut.

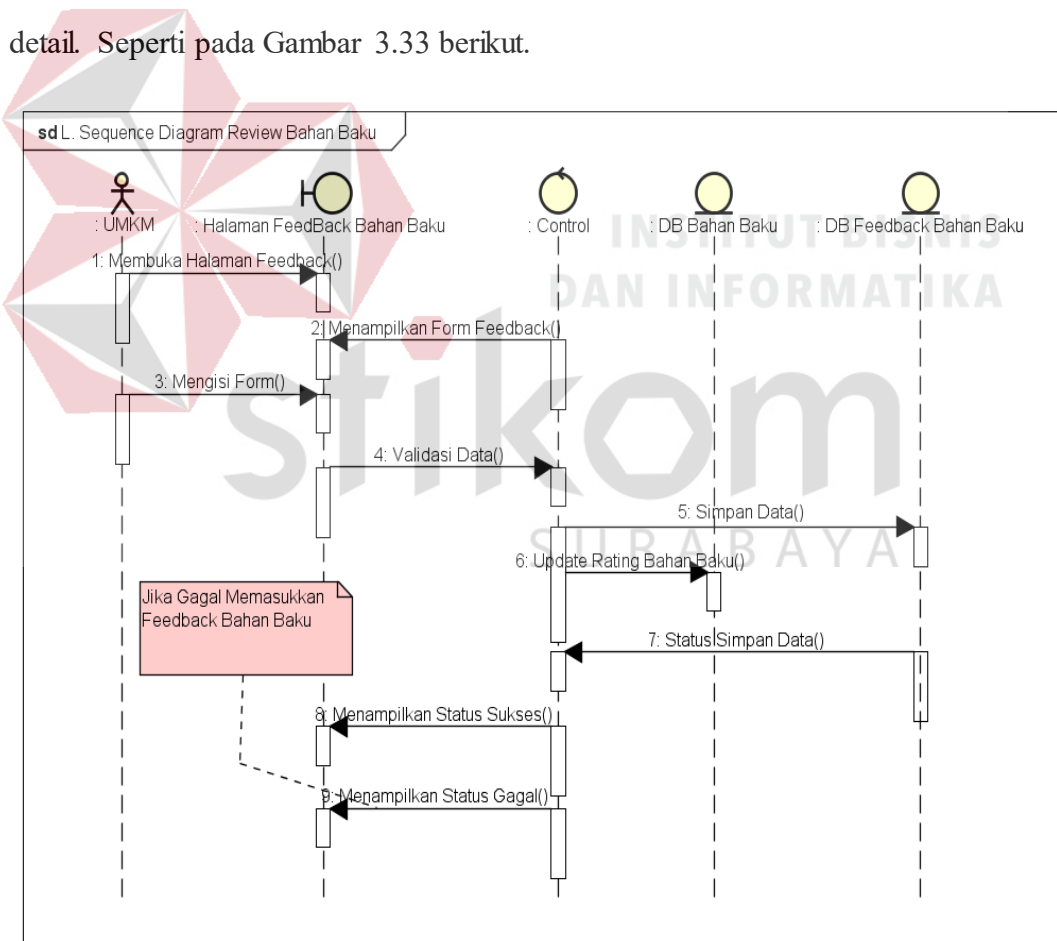
Tabel 3.23 *Flow Of Event Review Bahan Baku*

| Nama Flow Of Event | Review Bahan Baku | | |
|-----------------------|--|---|---|
| Aktor | UMKM. | | |
| Deskripsi | Pihak aktor dapat memberikan <i>review/feedback</i> terhadap bahan baku yang telah mereka beli sebelumnya. | | |
| Kondisi Awal | Aktor membuka halaman <i>review</i> bahan baku. | | |
| Kondisi Akhir | Sistem menyimpan data <i>review</i> bahan baku dan menampilkan halaman utama UMKM. | | |
| Aliran Kejadian Utama | No. | Aktor | Sistem |
| | 1. | UMKM membuka halaman <i>review</i> bahan baku | Sistem menampilkan halaman <i>review</i> bahan baku dan menampilkan form <i>review</i> bahan baku |
| | 2. | UMKM mengisi form <i>review</i> bahan baku | Sistem melakukan validasi data <i>review</i> . Jika benar maka sistem akan menyimpan data <i>review</i> dan |

| | | | |
|-----------------------------------|----|--|---|
| | | | menampilkan halaman utama UMKM |
| Aliran Kejadian Alternatif | 1. | UMKM membuka halaman <i>review</i> bahan baku | Sistem menampilkan halaman <i>review</i> bahan baku dan menampilkan form <i>review</i> bahan baku |
| | 2. | UMKM salah mengisi form <i>review</i> bahan baku | Sistem melakukan validasi data <i>review</i> . Jika salah sistem akan menampilkan pesan error |

b. Sequence Diagram Review Bahan Baku

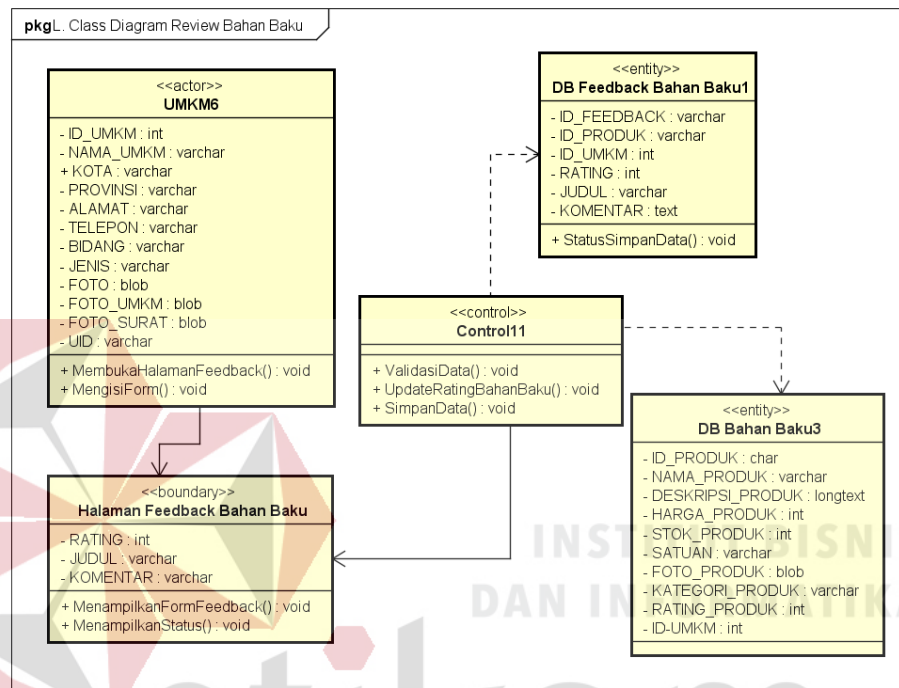
Pada diagram ini menggambarkan alur proses *review* bahan baku lebih detail. Seperti pada Gambar 3.33 berikut.



Gambar 3.33 Sequence Diagram Review Bahan Baku

c. Class Diagram Review Bahan Baku

Pada diagram ini menggambarkan *attribute*, *operation* dan *relation* yang ada pada setiap *actor*, *boundary*, *control* dan *entity* pada fungsi *review* bahan baku berdasarkan alur di *sequence diagram*. Seperti pada Gambar 3.34 berikut.



Gambar 3.34 Class Diagram Review Bahan Baku

d. Desain Test Case Review Bahan Baku

Desain uji coba pada fungsi *review* bahan baku apakah sudah berjalan sesuai keinginan atau tidak. Desain uji coba fungsi *review* bahan baku dapat dilihat pada Tabel 3.24 berikut.

Tabel 3.24 Desain Test Case Review Bahan Baku

| Test Case ID | Tujuan | Input | Output Yang Diharapkan | Status |
|--------------|---------------------------------------|-------------------------|--|--------|
| 1. | Melakukan pemberian rating bahan baku | Rating, Judul, Komentar | Review berhasil di simpan dan sistem menampilkan review bahan baku | |

| Test Case ID | Tujuan | Input | Output Yang Diharapkan | Status |
|--------------|--|-----------------|--|--------|
| 2. | Melakukan pemberian <i>review</i> bahan baku dengan format salah | Tidak ada input | Muncul pesan : data tidak boleh kosong | |

11. Fungsi Penjualan Bahan Baku

Fungsi ini digunakan oleh UMKM untuk dapat melihat data hasil penjualan bahan baku yang telah mereka lakukan dan melihat data bahan baku apa saja yang mereka jual pada aplikasi pemenuhan bahan baku UMKM. Penjelasan mengenai perancangan fungsi cek pengguna aktif adalah sebagai berikut.

a. *Flow Of Event* Penjualan Bahan Baku

Proses ini dilakukan oleh UMKM untuk dapat melihat data hasil penjualan bahan baku yang telah mereka lakukan dan melihat data bahan baku apa saja yang mereka jual pada aplikasi pemenuhan bahan baku UMKM. Seperti pada Tabel 3.25 berikut.

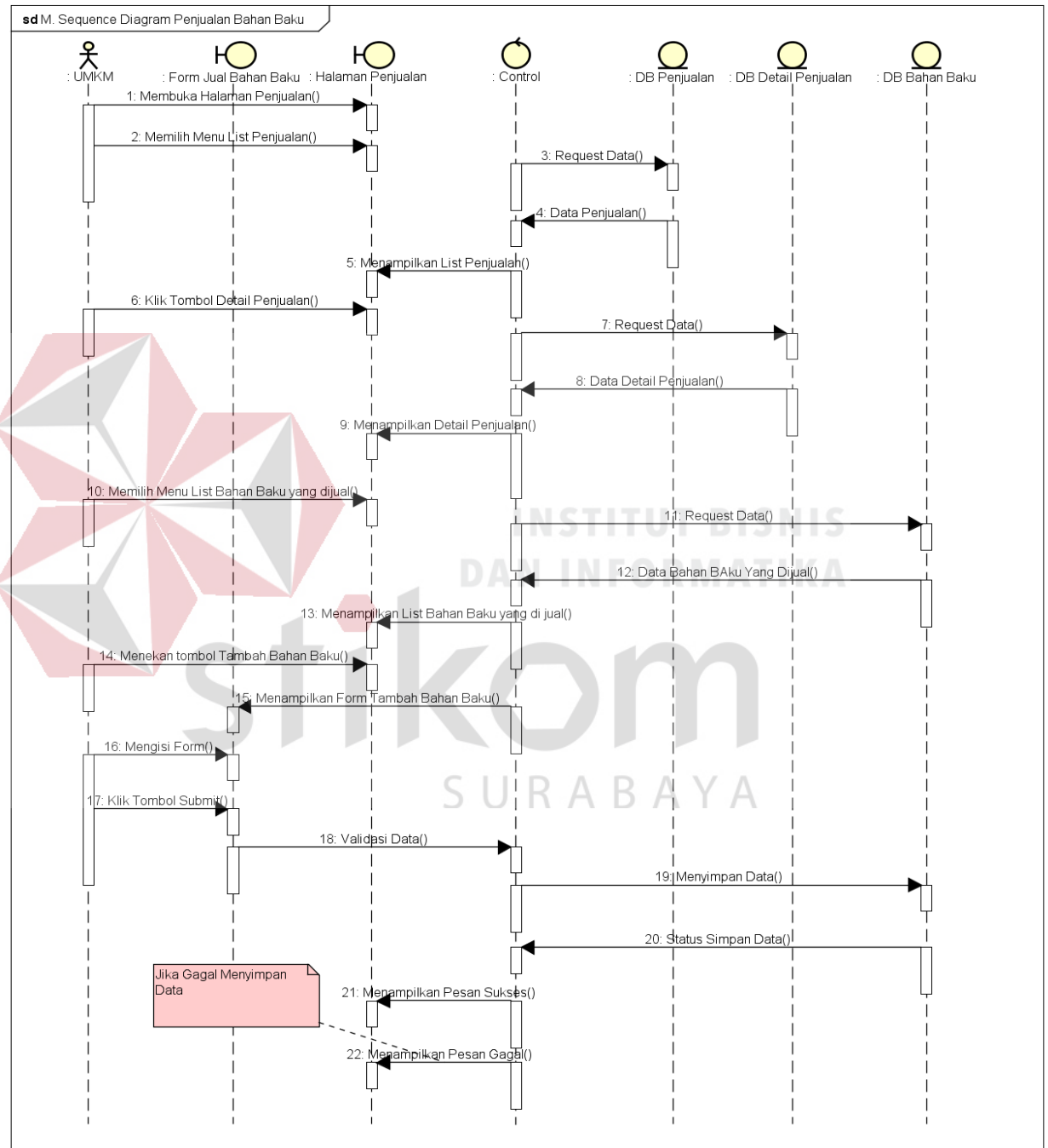
Tabel 3.25 *Flow Of Event* Penjualan Bahan Baku

| Nama <i>Flow Of Event</i> | Penjualan Bahan Baku | | |
|---------------------------|--|------------------------|---|
| Aktor | UMKM. | | |
| Deskripsi | Pihak aktor dapat melihat data hasil penjualan bahan baku yang telah mereka lakukan dan melihat data bahan baku apa saja yang mereka jual pada aplikasi pemenuhan bahan baku UMKM. | | |
| Kondisi Awal | Aktor membuka halaman penjualan. | | |
| Kondisi Akhir | Sistem menyimpan data bahan baku yang ditambahkan dan menampilkan halaman penjualan. | | |
| Aliran Kegiatan Utama 1 | No. | Aktor | Sistem |
| | 1. | Membuka menu penjualan | Sistem menampilkan halaman penjualan dan list penjualan |

| Nama <i>Flow Of Event</i> | Penjualan Bahan Baku | | |
|--------------------------------|----------------------|---------------------------------|---|
| Aliran Kegiatan Utama 2 | 1. | Membuka menu penjualan | Sistem menampilkan halaman penjualan dan list penjualan |
| | 2. | Memilih menu bahan baku | Sistem menampilkan list bahan baku |
| | 3. | Memilih fitur tambah bahan baku | Sistem menampilkan halaman form tambah bahan baku |
| | 4. | Mengisi form tambah bahan baku | Sistem melakukan validasi data bahan baku. Jika sesuai maka sistem akan menampilkan pesan berhasil menambahkan bahan baku |
| Aliran Kegiatan Utama 3 | 1. | Membuka menu penjualan | Menampilkan halaman penjualan dan list penjualan |
| | 2. | Memilih menu bahan baku | Sistem menampilkan list bahan baku |
| | 3. | Memilih fitur ubah bahan baku | Sistem menampilkan halaman form ubah bahan baku |
| | 4. | Mengubah data bahan baku | Sistem melakukan validasi data bahan baku. Jika sesuai maka sistem akan menampilkan pesan data bahan baku berhasil diubah |
| Aliran Kegiatan Utama 4 | 1. | Membuka menu penjualan | Menampilkan halaman penjualan dan list penjualan |
| | 2. | Memilih menu bahan baku | Sistem menampilkan list bahan baku |
| | 3. | Memilih fitur hapus bahan baku | Sistem menampilkan pesan konfirmasi hapus bahan baku |
| | 4. | Menyetujui hapus bahan baku | Sistem melakukan proses hapus bahan baku dan sistem akan kembali ke halaman penjualan |

b. Sequence Diagram Penjualan Bahan Baku

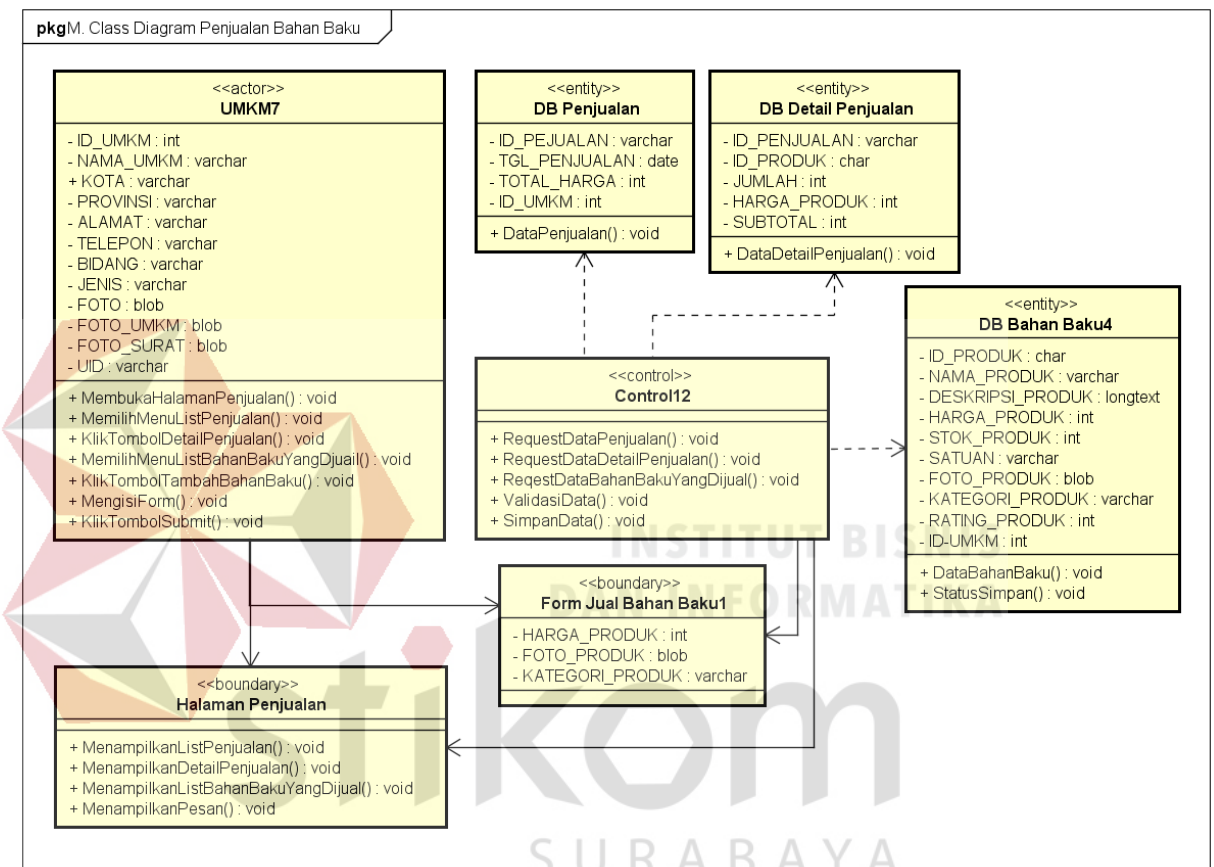
Pada diagram ini menggambarkan alur proses penjualan bahan baku lebih detail. Seperti pada Gambar 3.35 berikut.



Gambar 3.35 Sequence Diagram Penjualan Bahan Baku

c. **Class Diagram Penjualan Bahan Baku**

Pada diagram ini menggambarkan *attribute*, *operation* dan *relation* yang ada pada setiap *actor*, *boundary*, *control* dan *entity* pada fungsi penjualan bahan baku berdasarkan alur di *sequence diagram*. Seperti pada Gambar 3.36 berikut.



Gambar 3.36 Class Diagram Penjualan Bahan Baku

d. **Desain Test Case Penjualan Bahan Baku**

Desain uji coba pada fungsi penjualan bahan baku apakah sudah berjalan sesuai keinginan atau tidak. Desain uji coba fungsi penjualan bahan baku dapat dilihat pada Tabel 3.26 berikut.

Tabel 3.26 Desain *Test Case* Penjualan Bahan Baku

| Test Case ID | Tujuan | Input | Output Yang Diharapkan | Status |
|--------------|--|--|--|--------|
| 1. | Melakukan tambah data bahan baku | Nama_Produk, Deskripsi_Produk, Harga_Produk, Stok_Produk, Satuan, Foto_Produk, Kategori_Produk | Data bahan baku berhasil ditambahkan | |
| 2. | Melakukan tambah data bahan baku dengan format salah | Tidak ada input | Muncul pesan : data tidak boleh kosong | |
| 3. | Melakukan ubah data bahan baku | Nama_Produk, Deskripsi_Produk, Harga_Produk, Stok_Produk, Satuan, Foto_Produk, Kategori_Produk | Data bahan baku berhasil diubah | |
| 4. | Melakukan ubah data bahan baku dengan format salah | Tidak ada input | Muncul pesan : Data tidak boleh kosong | |

12. Fungsi Melihat Laporan

Fungsi ini digunakan oleh UMKM untuk dapat melihat dan mencetak laporan hasil dari transaksi penjualan dan pembelian yang telah dilakukan UMKM. Penjelasan mengenai perancangan lebih detail mengenai fungsi ini dapat dilihat sebagai berikut.

a. *Flow Of Event* Melihat Laporan

Proses ini dilakukan oleh UMKM untuk dapat melihat dan mencetak laporan hasil dari transaksi penjualan dan pembelian yang telah dilakukan UMKM. Seperti pada Tabel 3.27 berikut.

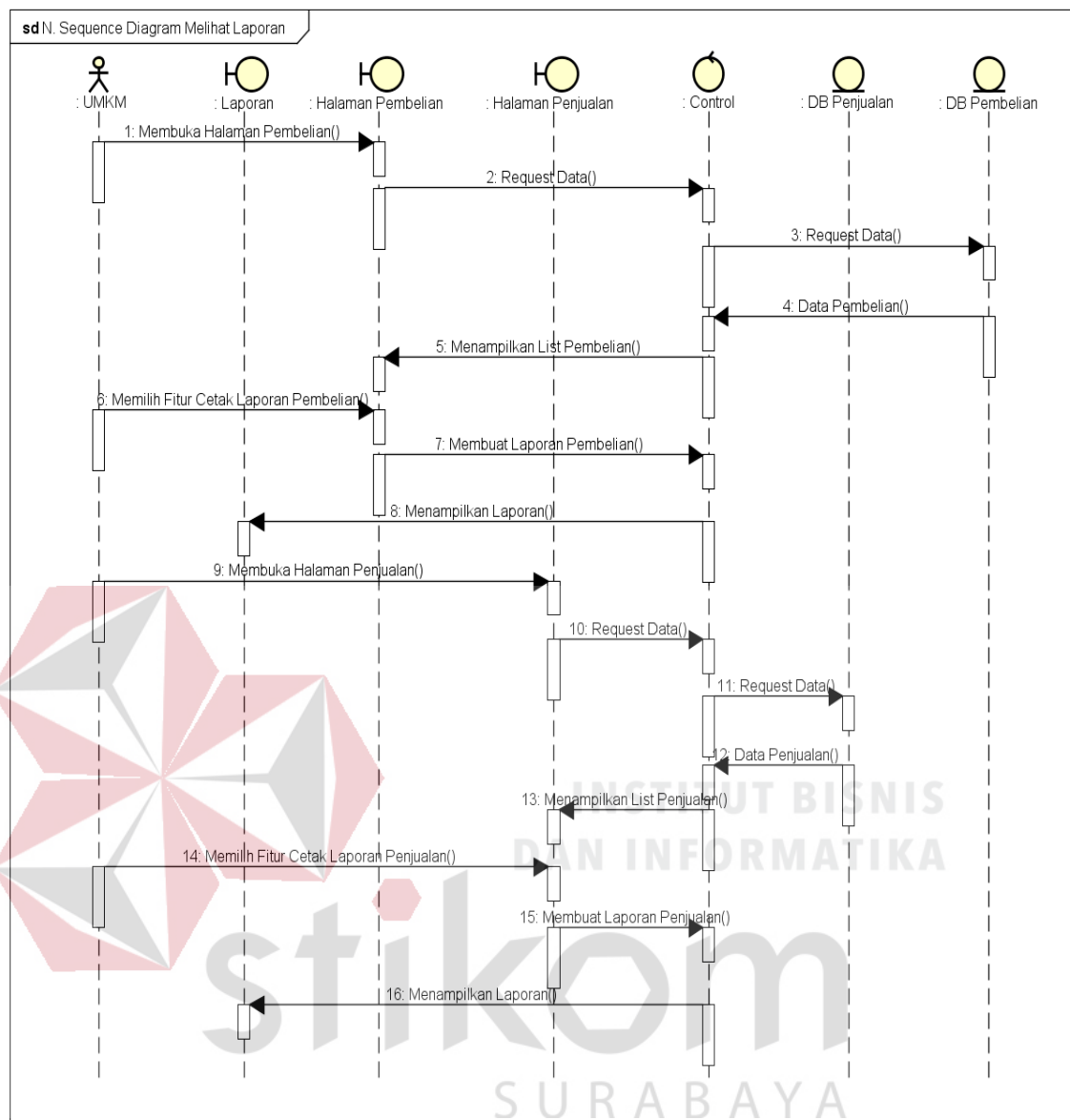
Tabel 3.27 *Flow Of Event* Melihat Laporan

| Nama <i>Flow Of Event</i> | | Melihat Laporan | |
|---------------------------|-----|---|--|
| Aktor | | UMKM. | |
| Deskripsi | | Pihak aktor dapat melihat dan mencetak laporan hasil dari transaksi penjualan dan pembelian yang telah dilakukan. | |
| Kondisi Awal | | 1. Aktor membuka halaman penjualan. 2. Aktor membuka halaman pembelian. | |
| Kondisi Akhir | | 1. Sistem menampilkan laporan penjualan. 2. Sistem menampilkan laporan pembelian. | |
| Aliran Kejadian Utama 1 | No. | Aktor | Sistem |
| | 1. | UMKM membuka halaman penjualan | Sistem menampilkan halaman penjualan dan list penjualan |
| Aliran Kejadian Utama 2 | 2. | UMKM memilih fitur cetak laporan | Sistem membuat laporan dan menampilkan laporan penjualan |
| | 1. | UMKM membuka halaman pembelian | Sistem menampilkan halaman pembelian dan list pembelian |
| | 2. | UMKM memilih fitur cetak laporan | Sistem membuat laporan dan menampilkan laporan pembelian |

b. *Sequence Diagram* Melihat Laporan

Pada diagram ini menggambarkan alur proses melihat laporan lebih detail.

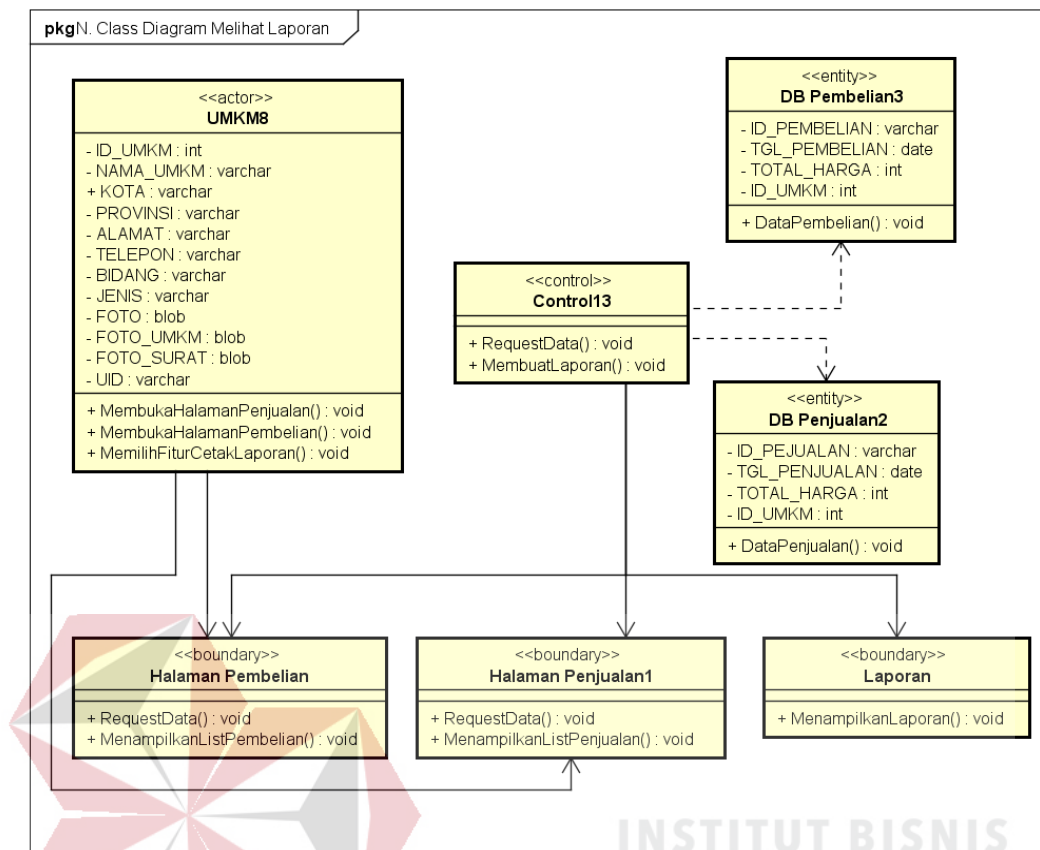
Seperti pada gambar 3.39 berikut.



Gambar 3.37 Sequence Diagram Melihat Laporan

c. **Class Diagram Melihat Laporan**

Pada diagram ini menggambarkan *attribute*, *operation* dan *relation* yang ada pada setiap *actor*, *boundary*, *control* dan *entity* pada fungsi penjualan bahan baku berdasarkan alur di *sequence diagram*. Seperti pada Gambar 3.38 berikut.



Gambar 3.38 Class Diagram Melihat Laporan

d. Desain Test Case Melihat Laporan

Desain uji coba pada fungsi melihat laporan apakah sudah berjalan sesuai keinginan atau tidak. Desain uji coba fungsi melihat laporan dapat dilihat pada Tabel 3.28 berikut.

Tabel 3.28 Desain Test Case Melihat Laporan

| Test Case ID | Tujuan | Input | Output Yang Diharapkan | Status |
|--------------|--|---|--|--------|
| 1. | Melakukan penyetakan laporan pembelian | Menekan tombol cetak pada halaman pembelian | Sistem menampilkan halaman <i>print</i> laporan. | |
| 2. | Melakukan penyetakan laporan penjualan | Menekan tombol cetak pada halaman penjualan | Sistem menampilkan halaman <i>print</i> laporan. | |

3.2.3. Daily Scrum

Tahap ini melaporkan kemajuan setiap *sprint backlog* yang sudah dikerjakan. *Daily scrum* akan dilakukan 24 jam kemudian untuk mengetahui perkembangan setelah melakukan *coding*. Pada kasus ini, *dialy scrum* menggunakan *tools* Trello sebagai alat bantu. Bentuk *daily scrum* dapat dilihat pada Tabel 3.29.

Tabel 3.29 *Daily scrum*

| <i>To do</i> | <i>In Progress</i> | <i>Done</i> |
|--|--|---|
| Membuat sesi <i>login</i> untuk <i>admin</i> | (Diisi apabila fungsi pada kolom <i>to do</i> sedang dikerjakan) | (Diisi apabila fungsi pada kolom <i>to do</i> telah selesai dikerjakan) |
| Membuat user interface halaman pengelolaan <i>bag of word</i> | | |
| Membuat proses CRUD (<i>Create, Read, Update, Delete</i>) data <i>bag of word</i> | | |
| Membuat user interface halaman pengelolaan kategori bahan baku | | |
| Membuat proses CRUD (<i>Create, Read, Update, Delete</i>) data kategori bahan baku | | |
| Membuat user interface halaman validasi pembayaran | | |
| Membuat proses validasi pembayaran | | |
| Membuat sesi <i>login</i> menggunakan <i>Google Sign In</i> untuk UMKM | | |
| Membuat <i>register</i> data-data kelengkapan UMKM | | |
| Membuat user interface untuk katalog bahan baku | | |
| Membuat user interface untuk keranjang bahan baku | | |

| <i>To do</i> | <i>In Progress</i> | <i>Done</i> |
|---|--------------------|-------------|
| Membuat <i>user interface</i> untuk hasil pembelian dan detail pembelian bahan baku | | |
| Membuat <i>user interface</i> untuk pengelolaan bahan baku | | |
| Membuat proses pengelolaan bahan baku | | |
| Membuat <i>user interface</i> untuk hasil penjualan bahan baku | | |
| Membuat proses transaksi pembelian | | |
| Membuat laporan pembelian | | |
| Membuat laporan penjualan | | |
| UMKM dapat memberikan <i>feedback</i> bahan baku yang telah dibeli | | |
| Notifikasi terhadap UMKM apabila ada pemberitahuan terbaru mengenai proses pembelian ataupun penjualan yang dilakukan. | | |
| Melakukan analisis <i>text mining</i> terhadap portofolio berkelanjutan UMKM. | | |
| Menggunakan <i>plug-in</i> dari <i>Sastrawi/tokenizing</i> untuk membantu dalam proses <i>text mining</i> pada tahap <i>tokenizing</i> | | |
| Menggunakan <i>plug-in</i> dari <i>Sastrawi/sastrawi</i> untuk membantu proses <i>text mining</i> pada tahap <i>stemming</i> dan <i>filtering stopwords</i> | | |
| Menggunakan <i>bag of words</i> UMKM untuk | | |

| <i>To do</i> | <i>In Progress</i> | <i>Done</i> |
|---|--------------------|-------------|
| membantu proses <i>text mining</i> | | |
| Menggunakan <i>plug-in</i> algoritma <i>Naïve Bayes Classification</i> (NBC) untuk menentukan <i>cluster</i> kebutuhan atau kelebihan UMKM akan bahan baku. | | |
| Menggunakan algoritma <i>Boyer Moore</i> untuk mencari kesamaan kata kunci hasil dari NBC berdasarkan portofolio UMKM dengan bahan baku yang dijual UMKM lain. | | |
| Menggunakan algoritma <i>Jaro-Winkler</i> untuk mencari tingkat kesamaan kata kunci hasil dari NBC berdasarkan portofolio UMKM dengan bahan baku yang dijual UMKM lain. | | |
| Menggunakan <i>Graphic Rating Scale</i> (GRS) untuk membuat rekomendasi bahan baku yang dibutuhkan UMKM berdasarkan hasil dari algoritma <i>Boyer-Moore</i> ataupun algoritma <i>Jaro-Winkler</i> | | |

3.3. Tahap Akhir

Pada tahap akhir ini terdapat beberapa proses-proses yang harus dilakukan yaitu *sprint review*, *sprint retrospective* dan *product*. untuk penjelasan lebih lanjut mengenai setiap proses-proses tersebut adalah sebagai berikut.

3.3.1. *Sprint Review*

Sprint review merupakan tahap presentasi berupa hasil produk yang telah sesuai dengan hasil awal kepada *product owner* atas pengerjaan yang telah dilakukan. Hasil keluaran yang dihasilkan berupa catatan apakah hasil telah sesuai dengan kesepakatan awal atau belum. Apabila hasil belum sesuai maka akan dilakukan iterasi selanjutnya.

A. *Sprint Review Iterasi Pertama*

Sprint review pada iterasi pertama dapat menghasilkan aplikasi analisis *text mining* berbasis *web* yang dapat melakukan analisis terhadap portofolio berkelanjutan UMKM dibantu dengan beberapa algoritma seperti Naïve Bayes, *Boyer-Moore*, dan *Jaro Winkler* sehingga dapat mengetahui kekurangan berdasarkan portofolio berkelanjutan UMKM dan dapat memberikan rekomendasi bahan baku yang sesuai dengan kebutuhan yang dimiliki oleh UMKM tersebut.

1. *Review Hasil Iterasi Pertama*

Berikut dijelaskan *review* hasil dari iterasi pertama dapat dijelaskan sebagai berikut.

2. *Kesesuaian Story Dengan Fitur Aplikasi*

Pada iterasi pertama telah dilakukan beberapa pengerjaan proses dan menghasilkan bebapa fitur yang dapat digunakan sesuai dengan *story* yang terjadi pada *product backlog*. Beberapa fitur tersebut telah dirilis pada kalangan tertentu seperti *scrum team*, *scrum master*, dan *product owner*. Dapat dilihat pada Tabel 3.30 *story* dan fitur yang dihasilkan pada iterasi pertama.

Tabel 3.30 Kesesuaian *Story* Dengan Fitur Aplikasi

| No | ID Story | Fitur |
|----|----------------------------|--|
| 1. | 1 | Fitur: - <i>Login UMKM</i> Deskripsi: Pengguna aplikasi (UMKM) dapat melakukan <i>register</i> dan <i>login</i> dengan menggunakan akun Google mereka. Dan dapat mengambil beberapa data yang terdapat pada akun Google-nya |
| 2. | 3 | Fitur : - <i>Login Admin</i> Deskripsi : Pengguna aplikasi (<i>admin</i>) dapat melakukan <i>login</i> dengan menggunakan <i>email</i> dan <i>password</i> yang mereka punya. |
| 3. | 10, 11, 12, 13, 23, 24, 26 | Fitur : - Mengisi <i>Timeline Portofolio</i> - Melihat Klasifikasi Status <i>Timeline Portofolio</i> Deskripsi: Klasifikasi status <i>timeline portofolio</i> dengan menggunakan fitur <i>text mining</i> , <i>Naïve bayes classifier</i> , <i>Boyer Moore</i> dan <i>Graphic Rating Scale</i> . Sehingga menghasilkan keluaran berupa bahan baku yang sesuai dengan kebutuhan dan kelebihan UMKM. |
| 4. | 2 | Fitur : - <i>Register UMKM</i> Deskripsi : Pengguna aplikasi (UMKM) dapat melakukan <i>register</i> dengan mengisi data-data kelengkapan UMKM mereka seperti : nama UMKM, alamat, nomor telepon, dan sebagainya. |
| 5. | 14, 15, 16, 17, 18, 19 | Fitur : - <i>Pembelian Bahan Baku</i> - <i>Melihat Laporan Pembelian</i> Deskripsi : Pengguna aplikasi (UMKM) dapat melakukan transaksi pembelian mulai dari melihat katalog bahan baku hingga mencetak laporan pembelian. |
| 6. | 20, 21, 22 | Fitur : - <i>Penjualan Bahan Baku</i> - <i>Mengelola Bahan Baku</i> - <i>Melihat Laporan Penjualan</i> Deskripsi : Pengguna aplikasi (UMKM) dapat melakukan penjualan mulai dari mengelola bahan baku yang |

| No | ID Story | Fitur |
|-----|----------|---|
| | | ingin dijual hingga dapat mencetak laporan penjualan. |
| 7. | 4, 5 | Fitur : - Mengelola Data <i>Bag Of Word</i> Deskripsi : Pengguna aplikasi (<i>admin</i>) dapat melihat, menambah, dan menghapus data <i>bag of word</i> pada halaman <i>bag of word</i> |
| 8. | 6, 7 | Fitur : - Mengelola Data Kategori Bahan Baku Deskripsi : Pengguna aplikasi (<i>admin</i>) dapat melihat, menambah, dan menghapus data kategori bahan baku pada halaman kategori bahan baku |
| 9. | 8, 9 | Fitur : - Validasi Pembayaran Deskripsi : Pengguna aplikasi (<i>admin</i>) dapat melakukan validasi pada bukti pembayaran yang telah dikirim oleh UMKM |
| 10. | 28 | Fitur : - Menerima Notifikasi Deskripsi : Pengguna aplikasi (UMKM) dapat menerima notifikasi baik dari hasil proses pembelian maupun dari hasil proses penjualan |
| 11. | 27 | Fitur : - Memberikan <i>Review</i> Bahan Baku Deskripsi : Pengguna aplikasi (UMKM) dapat memberikan <i>feedback</i> berupa <i>review</i> pada bahan baku yang telah ia beli |

3. Kesimpulan Review

Kesimpulan berdasarkan hasil diskusi dan pengembangan oleh *scrum master*, *product owner*, dan pengembang memiliki beberapa kekurangan pada sistem. Beberapa kekurangan tersebut adalah pada fitur klasifikasi *timeline portofolio* sudah dapat menghasilkan hasil berupa rekomendasi, namun algoritma *string matching* yang digunakan yaitu *Boyer Moore* mempunyai kekurangan yaitu tidak dapat mengatasi kesalahan pengetikan yang dilakukan oleh pengguna. Oleh

karena itu penggunaan *Boyer Moore* perlu dipertimbangkan atau ditambahkan algoritma *string matching* lainnya untuk menyempurnakan proses *string matching*.

B. Sprint Review Iterasi Kedua

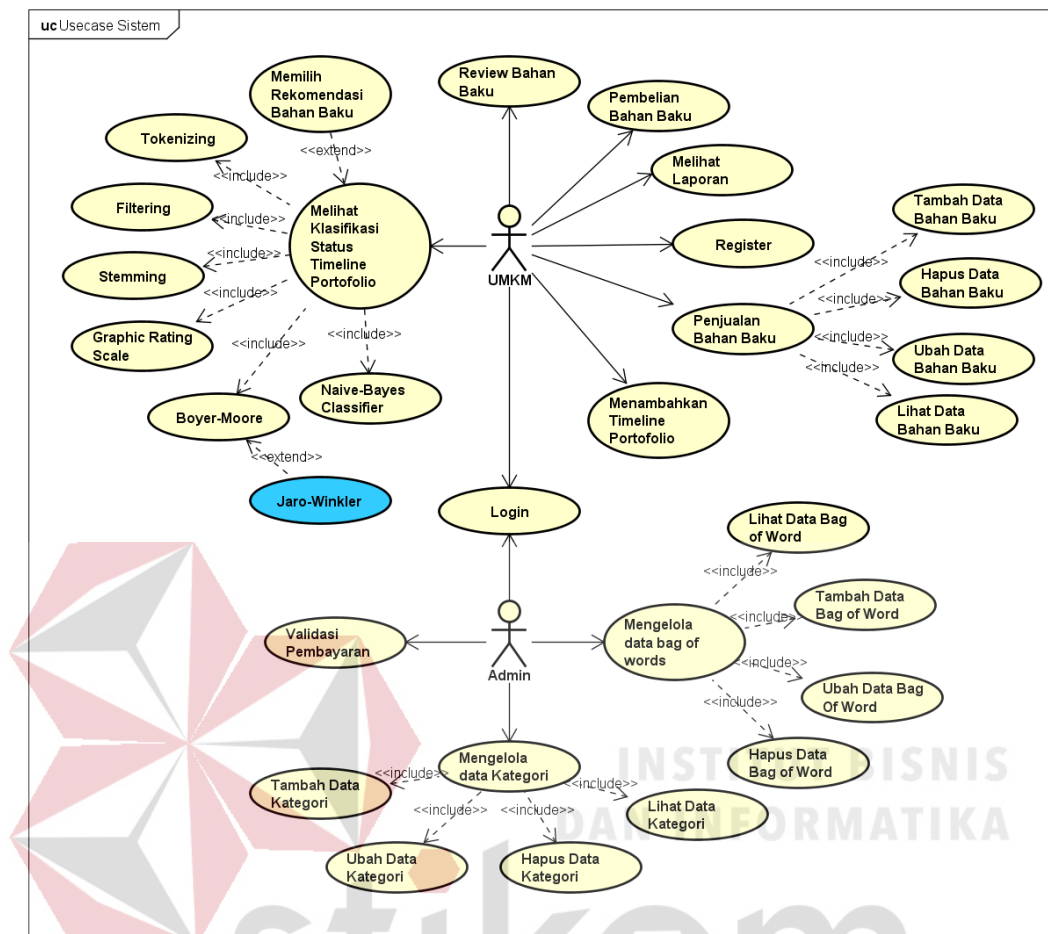
Pada sprint planning iterasi kedua ini yang akan dikerjakan adalah hal yang sama dari sprint planning iterasi pertama, namun dengan tambahan yaitu memperbaiki tampilan dan memperbaiki dari hasil *review* dengan *product owner*.

1. Perbaikan Dari Iterasi Pertama

a. Penambahan Algoritma String Matching Jaro Winkler

Hasil dari sprint *review* iterasi pertama adalah adanya penambahan algoritma yaitu algoritma *jaro winkler*. Penambahan algoritma tersebut digunakan untuk mengatasi kesalahan masukan (*typo*) yang dilakukan oleh UMKM pada saat mengisi status di *timeline portofolio*. Algoritma ini dijalankan apabila algoritma *Boyer Moore* tidak dapat menemukan bahan baku yang dibutuhkan oleh UMKM. Dengan adanya penambahan algoritma *jaro winkler*, maka terjadi beberapa perubahan pada perancangan sistem untuk aplikasi pemenuhan bahan baku UMKM ini seperti pada *usecase system diagram* dan pada fungsi melihat klasifikasi status *timeline portofolio* UMKM, untuk lebih jelasnya dapat dilihat sebagai berikut.

i. Perubahan *Use Case Sistem Diagram*



Gambar 3.39 Perubahan *Use Case System Diagram*

ii. Perubahan Fungsi Melihat Klasifikasi Status *Timeline Protofolio*

a) *Flow Of Event* Fungsi Melihat Klasifikasi Status *Timeline Protofolio*

Tabel 3.31 Perubahan *Flow Of Event* Melihat Klasifikasi Status *Timeline Portofolio*

| Nama Flow Of Event | Melihat Klasifikasi Status <i>Timeline Portofolio</i> |
|--------------------|--|
| Aktor | UMKM. |
| Deskripsi | Sistem dapat secara otomatis untuk menentukan kekurangan atau kelebihan bahan baku yang dimiliki oleh aktor berdasarkan <i>timeline portofolio</i> UMKM dengan penggunaan beberapa algoritma yaitu, <i>text mining</i> , <i>Naïve bayes classifier</i> , |

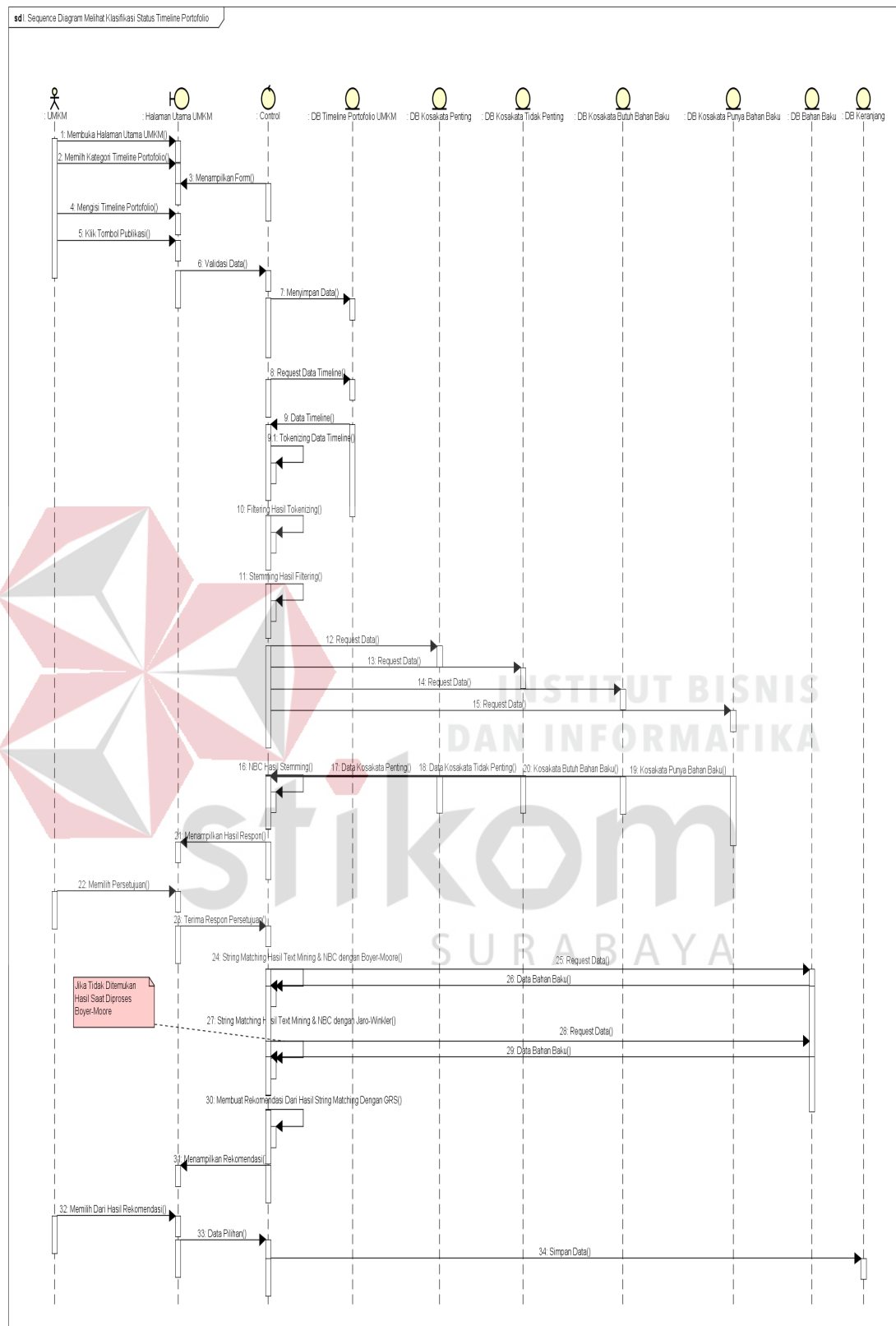
| Nama <i>Flow Of Event</i> | Melihat Klasifikasi Status <i>Timeline Portofolio</i> | | |
|--------------------------------|---|---|--|
| | <i>Boyer Moore</i> , dan <i>Jaro Winkler</i> pada aplikasi pemenuhan bahan baku UMKM. | | |
| Kondisi Awal | Aktor melakukan posting aktivitas UMKMnya. | | |
| Kondisi Akhir | 3. Sistem menampilkan hasil rekomendasi bahan baku yang sesuai dengan kebutuhan UMKM. 4. Sistem menampilkan halaman form untuk UMKM menjual bahan bakunya. | | |
| Aliran Kejadian Utama 1 | No | User | Sistem |
| | 1. | UMKM melakukan posting <i>timeline portofolio</i> | Sistem melakukan validasi data posting <i>timeline portofolio</i> . Jika ya maka sistem akan menyimpan data <i>timeline portofolio</i> |
| | 2. | | Sistem melakukan proses <i>tokenizing</i> , <i>filtering</i> dan <i>stemming</i> pada data posting <i>timeline portofolio</i> |
| | 3. | | Sistem melakukan proses <i>Naïve-Bayes Classifier</i> pada hasil <i>stemming</i> . Jika hasilnya adalah butuh maka sistem akan melanjutkan ke proses <i>String Matching</i> |
| | 4. | | Sistem melakukan proses <i>string matching</i> dengan bahan baku yang dijual UMKM lain menggunakan <i>algoritma Boyer-Moore</i> |
| | 5. | | Jika bahan baku ditemukan maka sistem akan melakukan proses <i>Graphic Rating Scale</i> untuk menentukan rekomendasi. Dan sistem akan menampilkan rekomendasi bahan baku kepada UMKM |
| | 6. | UMKM memilih bahan baku dari hasil rekomendasi | Sistem akan menyimpan data bahan baku kedalam keranjang untuk dilakukan proses pembelian |
| Aliran Kejadian Utama 2 | 1. | UMKM melakukan posting <i>timeline portofolio</i> | Sistem melakukan validasi data posting <i>timeline portofolio</i> . Jika ya maka sistem akan menyimpan data <i>timeline portofolio</i> |

| Nama <i>Flow Of Event</i> | Melihat Klasifikasi Status <i>Timeline Portofolio</i> | | |
|------------------------------|---|---|--|
| | 2. | | Sistem melakukan proses <i>tokenizing, filtering</i> dan <i>stemming</i> pada data posting <i>timeline portofolio</i> |
| | 3. | | Sistem melakukan proses <i>Naïve-Bayes Classifier</i> pada hasil <i>stemming</i> . Jika hasilnya adalah butuh maka sistem akan melanjutkan ke proses <i>String Matching</i> |
| | 4. | | Sistem melakukan proses <i>string matching</i> dengan bahan baku yang dijual UMKM lain menggunakan algoritma <i>Boyer-Moore</i> |
| | 5. | | Jika bahan baku tidak ditemukan saat proses <i>string matching</i> dengan algoritma <i>boyer-moore</i> maka sistem akan melakukan proses <i>string matching</i> dengan algoritma <i>Jaro-Winkler</i> |
| | 6. | | Jika bahan baku ditemukan maka sistem akan melakukan proses <i>Graphic Rating Scale</i> untuk menentukan rekomendasi. Dan sistem akan menampilkan rekomendasi bahan baku kepada UMKM |
| | 7. | UMKM memilih bahan baku dari hasil rekomendasi | Sistem akan menyimpan data bahan baku kedalam keranjang untuk dilakukan proses pembelian |
| Alur Kegiatan Utama 3 | 1. | UMKM melakukan posting <i>timeline portofolio</i> | Sistem melakukan validasi data posting <i>timeline portofolio</i> . Jika ya maka sistem akan menyimpan data <i>timeline portofolio</i> |
| | 2. | | Sistem melakukan proses <i>tokenizing, filtering</i> dan <i>stemming</i> pada data posting <i>timeline portofolio</i> |
| | 3. | | Sistem melakukan proses <i>Naïve-Bayes Classifier</i> pada hasil <i>stemming</i> . |

| Nama <i>Flow Of Event</i> | Melihat Klasifikasi Status <i>Timeline Portofolio</i> | | |
|---------------------------|---|---|---|
| | 4. | | Jika hasilnya adalah kelebihan maka sistem akan menampilkan form tambah bahan baku yang ingin dijual |
| Alur Kegiatan Utama 4 | 1. | UMKM melakukan posting <i>timeline portofolio</i> | Sistem melakukan validasi data posting <i>timeline portofolio</i> . Jika ya maka sistem akan menyimpan data <i>timeline portofolio</i> |
| | 2. | | Sistem melakukan proses <i>tokenizing, filtering</i> dan <i>stemming</i> pada data posting <i>timeline portofolio</i> |
| | 3. | | Sistem melakukan proses <i>Naïve-Bayes Classifier</i> pada hasil <i>stemming</i> . Jika hasilnya adalah butuh maka sistem akan melanjutkan ke proses <i>String Matching</i> |
| | 4. | | Sistem melakukan proses <i>string matching</i> dengan bahan baku yang dijual UMKM lain menggunakan <i>algoritma Boyer-Moore</i> |
| | 5. | | Jika bahan baku tidak ditemukan maka sistem akan menampilkan pesan bahan baku tidak ditemukan kepada UMKM |

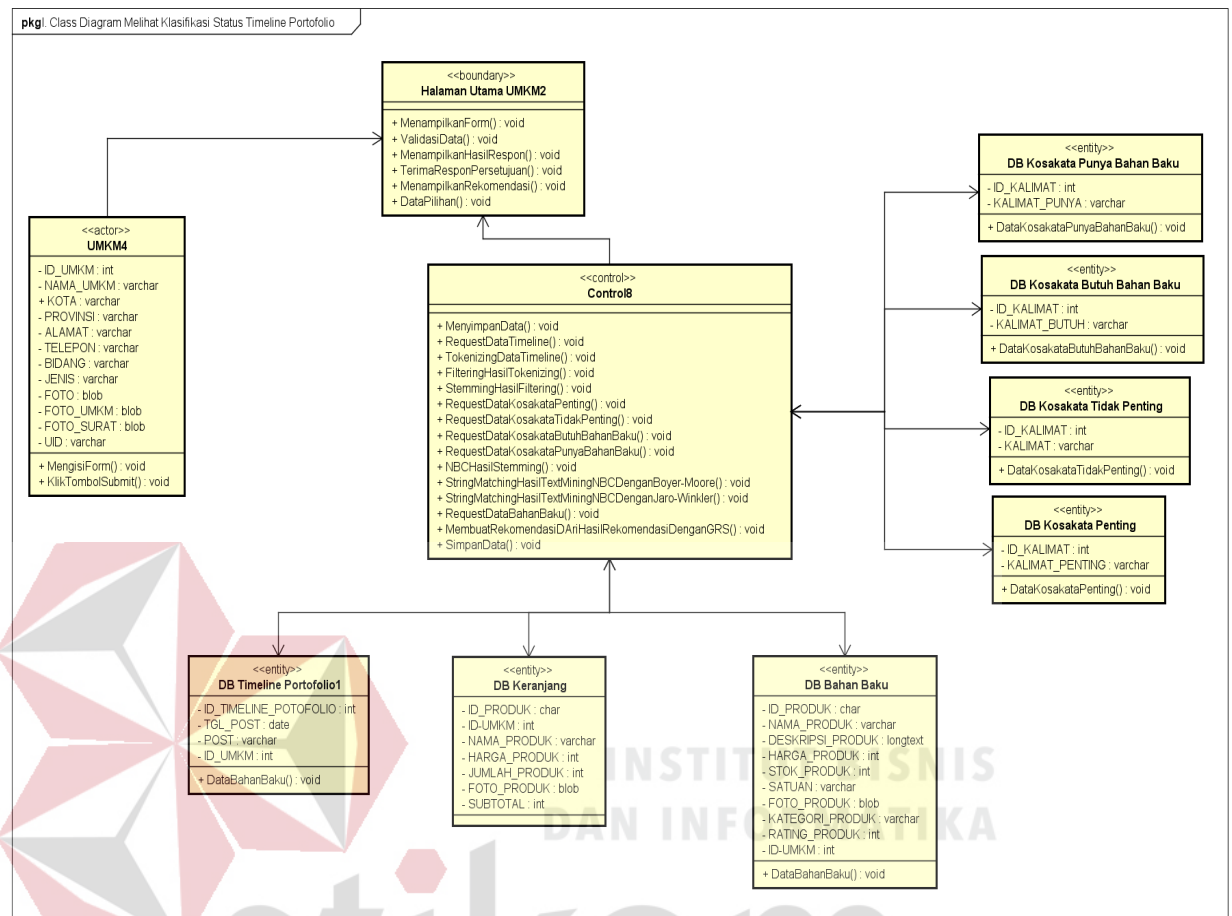
b) *Sequence Diagram Fungsi Melihat Klasifikasi Status Timeline Protofolio*

Untuk perubahan pada *sequence diagram* fungsi melihat klasifikasi status *timeline portofolio* dapat dilihat pada gambar Gambar 3.40.



Gambar 3.40 Perubahan *Sequence Diagram* Melihat Klasifikasi Status *Timeline Portfolio*

c) **Class Diagram Fungsi Melihat Klasifikasi Status Timeline Protofolio**



Gambar 3.41 Perubahan *Class Diagram* Melihat Klasifikasi Status Timeline Portofolio

2. Review Hasil Iterasi Kedua

Berikut dijelaskan *review* hasil dari iterasi kedua dapat dijelaskan sebagai berikut.

a. Kesesuaian dengan *story* dengan fitur

Pada iterasi pertama telah dilakukan beberapa pengerjaan proses dan menghasilkan bebapa fitur yang dapat digunakan sesuai dengan *story* yang terjadi

pada *product backlog* dan perbaikan dari iterasi pertama. Beberapa fitur tersebut telah dirilis pada kalangan tertentu seperti *scrum team*, *scrum master*, dan *product owner*. Dapat dilihat pada Tabel 3.32 *story* dan fitur yang dihasilkan pada iterasi kedua.

Tabel 3.32 Kesesuaian Fitur Dengan *Story*

| No | ID <i>Story</i> | Fitur |
|----|-----------------|--|
| 1 | 25 | Fitur: - Penerapan algoritma <i>Jaro Winkler</i> pada klasifikasi <i>timeline portofolio</i> Deskripsi: Algoritma diterapkan setelah algoritma <i>Boyer Moore</i> . Dimana algoritma dapat menutupi kekurangan dari algoritma <i>Boyer Moore</i> yaitu tidak bisa mengatasi kesalahan pemasukan data oleh pengguna. |

b. Hasil Desain

i. Halaman *Login Admin*

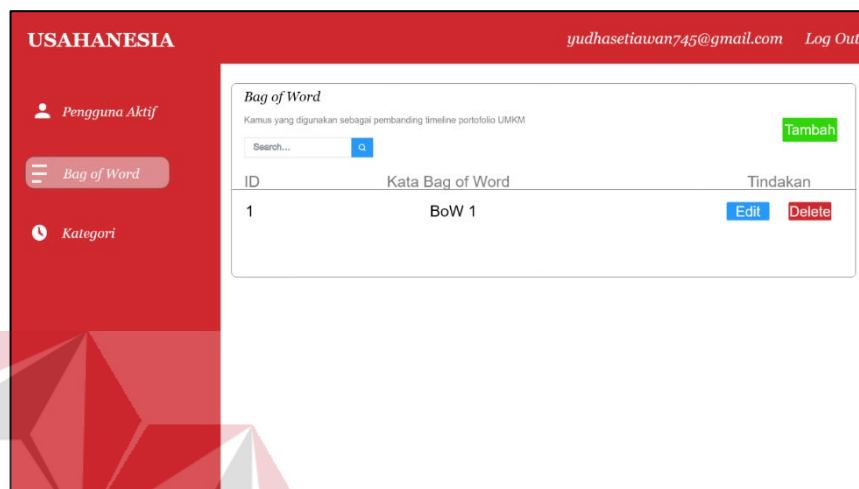
Pada halaman *login admin*, *admin* akan mengisi email dan password. Gambaran dari rancangan desain *login admin* dapat dilihat pada Gambar 3.42 berikut.

Gambar 3.42 Desain Halaman *Login Admin*

ii. Halaman Mengelola *Bag Of Word*

Pada menu pengolahan *bag of word* di sini yang akan digunakan sebagai kamus pembanding pada saat proses *text mining* status *timeline portfolio*.

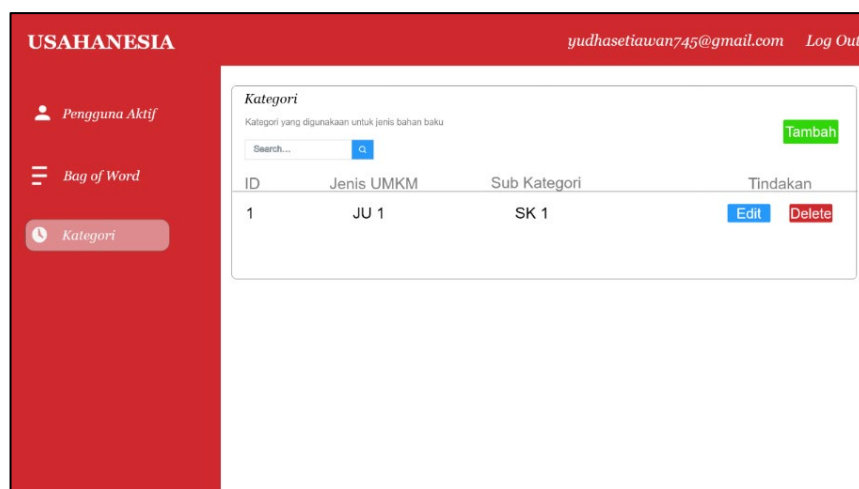
Gambaran desain halaman mengelola *bag of word* ada Gambar 3.43 berikut



Gambar 3.43 Desain Halaman Mengelola Data *Bag Of Word*

iii. Halaman Mengelola Kategori

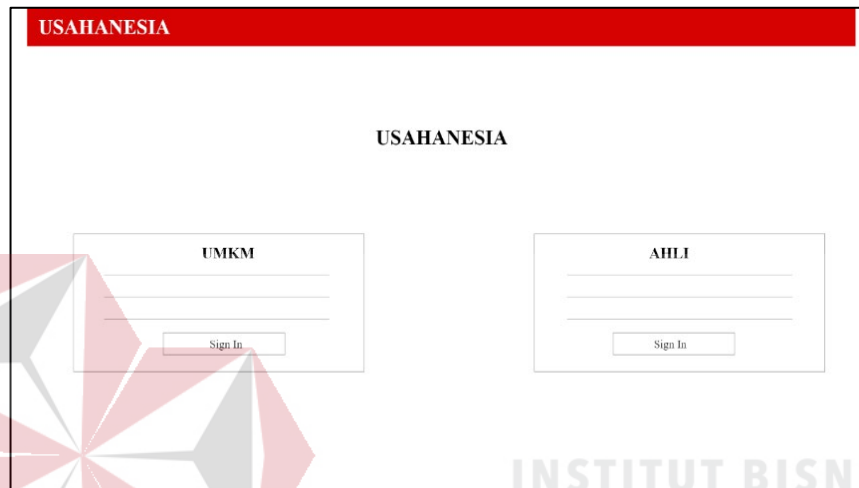
Menu kategori digunakan oleh *admin* untuk menambah data challenge pada aplikasi ini. Gambaran dari rancangan desain menu kategori dapat dilihat pada Gambar 3.44.



Gambar 3.44 Desain Halaman Mengelola Data Kategori

iv. Halaman *Login* UMKM

Pada halaman *login*, UMKM dan ahli akan diberikan satu halaman yang sama. *Login* untuk UMKM akan menggunakan sistem dari google untuk mendapatkan beberapa data. Gambaran dari rancangan desain *login* dapat dilihat pada Gambar 3.45 berikut.



Gambar 3.45 Desain Halaman *Login* UMKM

v. Halaman *Register* UMKM

UMKM yang belum pernah mendaftarkan akun *google*-nya pada aplikasi ini akan diarahkan pada halaman pendaftaran UMKM. Halaman pendaftaran UMKM terdapat 2 jenis, yaitu halaman data profil UMKM, dan halaman data struktur organisasi UMKM. Halaman data profil UMKM memuat seluruh data umum UMKM yang akan diisi oleh UMKM. Gambaran dari rancangan desain data profil dapat dilihat pada Gambar 3.46 berikut.

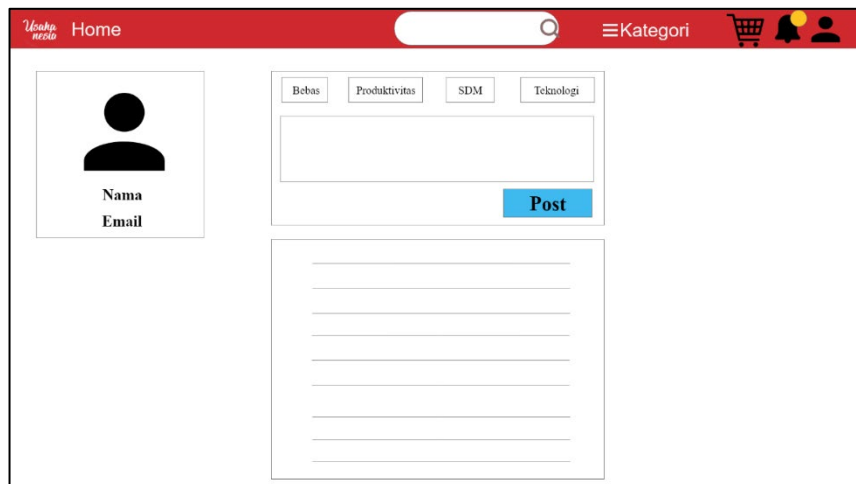
Gambar 3.46 Desain Halaman *Register* Profil UMKM

Halaman data struktur organisasi UMKM memuat data sumber daya manusia pada suatu UMKM. Gambaran dari rancangan desain data struktur UMKM dapat dilihat pada Gambar 3.47 berikut.

Gambar 3.47 Desain Halaman *Register* Struktur Organisasi UMKM

vi. Halaman Utama UMKM

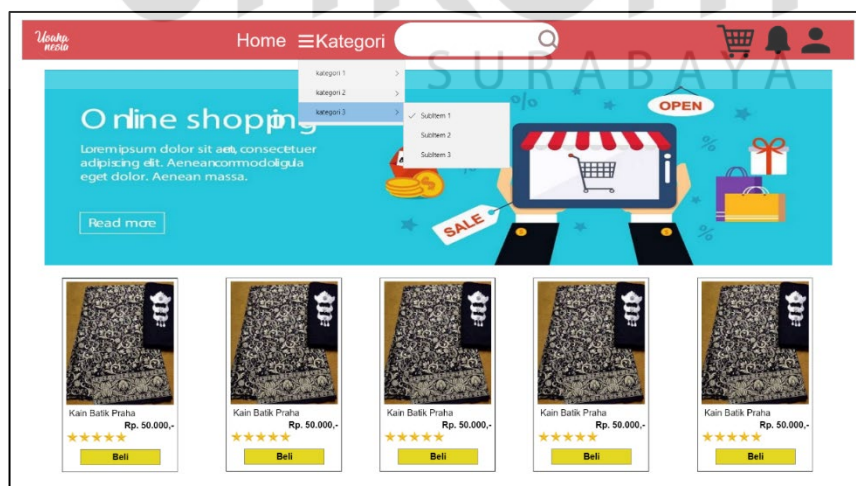
Pada halaman utama UMKM, UMKM dapat menambahkan kegiatan dan kinerja mereka. Terdapat 4 jenis kategori yang ada pada *Timeline Portofolio*, yaitu bebas, produktivitas, sumber daya manusia, serta teknologi. Gambaran dari rancangan desain halaman utama UMKM dapat dilihat pada Gambar 3.48.



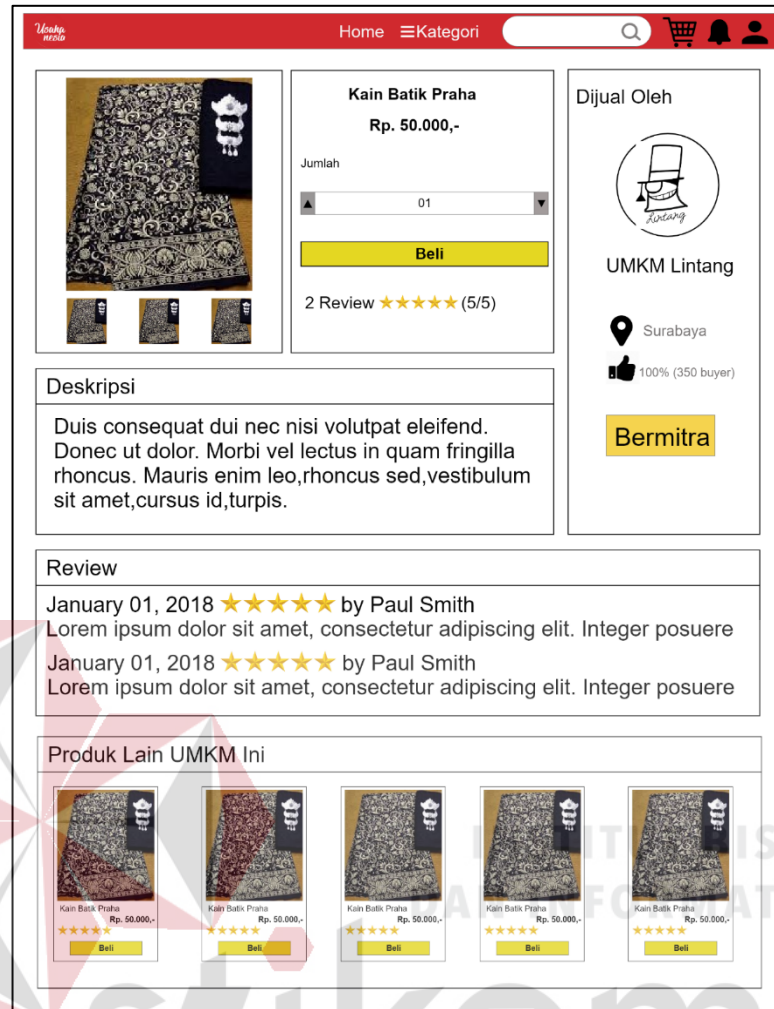
Gambar 3.48 Desain Halaman Utama UMKM

vii. Halaman Katalog Bahan Baku

Pada halaman katalog bahan baku UMKM dapat melihat informasi atau promo yang terjadi di aplikasi pemenuhan bahan baku produksi UMKM ini. Gambaran dari rancangan desain halaman katalog bahan baku dapat dilihat pada Gambar 3.49 dan untuk detail dari bahan baku tersebut dapat dilihat pada Gambar 3.50.

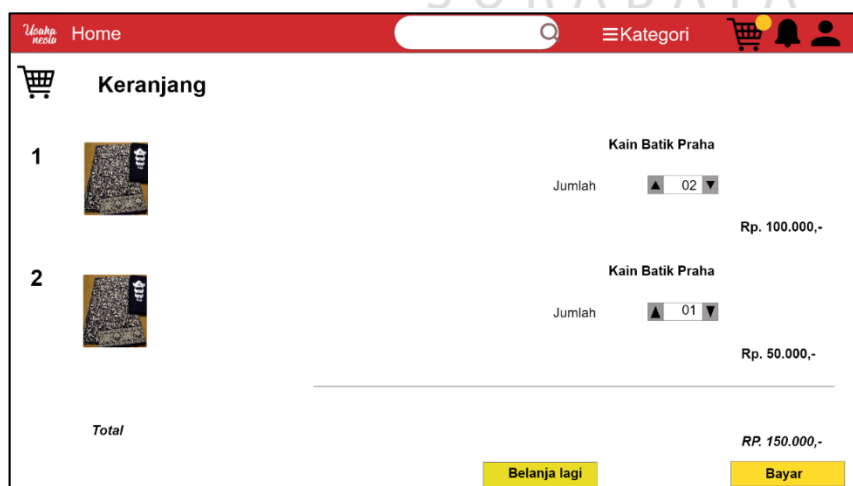


Gambar 3.49 Desain Halaman Katalog Bahan Baku



Gambar 3.50 Desain Halaman Detail Bahan Baku UMKM

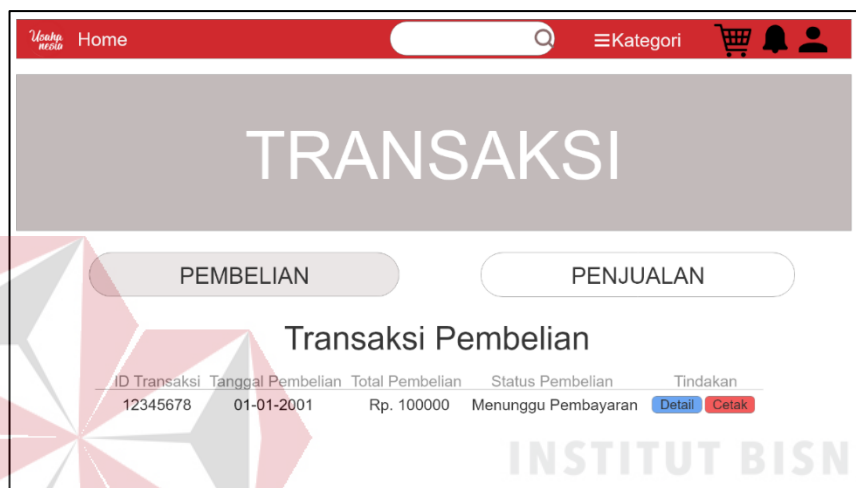
viii. Halaman Keranjang



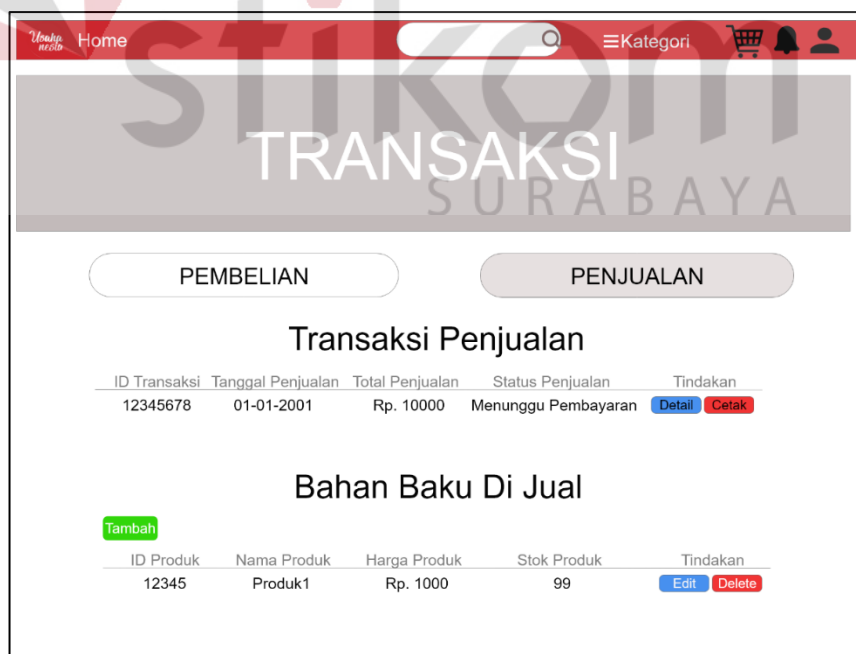
Gambar 3.51 Desain Halaman Keranjang UMKM

ix. Halaman Transaksi

Pada halaman transaksi disini terdapat dua *tab* yaitu pembelian dan penjualan jadi di halaman ini UMK dapat melihat *history* dan status dari pembelian yang telah mereka lakukan. Gambaran dari halaman transaksi penjualan dan transaksi pembelian dapat dilihat pada Gambar 3.52 dan Gambar 3.53.

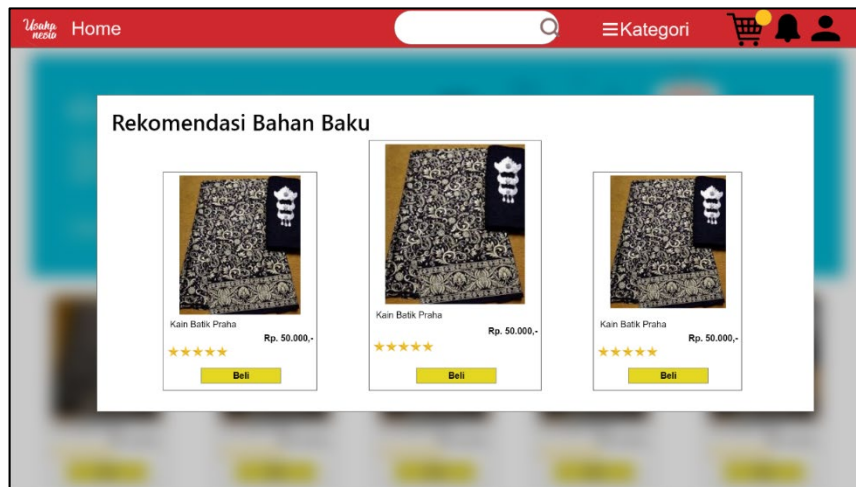


Gambar 3.52 Desain Halaman Transaksi Pembelian UMKM



Gambar 3.53 Desain Transaksi Penjualan UMKM

x. Halaman Rekomendasi



Gambar 3.54 Desain Halaman Rekomendasi UMKM

c. Kesimpulan Review

Dengan adanya perbaikan pada iterasi kedua, telah menyelesaikan masalah yang telah didiskusikan pada iterasi pertama dan telah sesuai dengan *product backlog*.

3.3.2. Sprint Retrospective

Tahapan ini melakukan pembahasan kekurangan yang dilakukan secara personal oleh *scrum master* terhadap individu dari *development team*. Proses ini menghasilkan catatan berupa kekurangan yang dimunculkan oleh *development team*. Pencatatan tersebut dilakukan oleh *scrum master* yang berdasarkan analisis terhadap target pengerjaan serta pengerjaan pelaksanaan yang terjadi.

Pada iterasi pertama terdapat beberapa fungsi dan tampilan yang dijabarkan sudah sesuai dengan rencana pengembangan awal. Namun terdapat beberapa kekurangan yang terjadi seperti ketidaklengkapan pemberian fungsi pada

iterasi pertama sehingga diperlukan adanya penambahan fungsi lainnya. Fungsi yang ditambahkan adalah algoritma *Jaro Winkler* pada klasifikasi *timeline portofolio* dimana algoritma ini memiliki diperlukan untuk mengatasi apabila user melakukan kesalahan pengetikan terhadap *timeline portofolio* mereka

3.3.3. *Product*

Pada tahapan ini adalah hasil keluaran dari produk yang dihasilkan. Beberapa produk tersebut adalah aplikasi penerapan *text mining* untuk pemenuhan bahan baku UMKM berbasis *web* pada startup Usahanesia dan buku laporan tugas akhir serta dokumen-dokumen yang dibutuhkan sebagai penunjang tugas akhir.



BAB IV

IMPLEMENTASI DAN EVALUASI SISTEM

4.1. Implementasi

4.1.1. Kebutuhan *Software* dan *Hardware*

Adapun kebutuhan *software* dan *hardware* untuk mengaplikasikan aplikasi ini dari sisi *client* ataupun *server* seperti yang tampak pada Tabel 4.1 berikut.

Tabel 4.1 Tabel Kebutuhan *Software* Dan *Hardware*

| | <i>Client</i> | <i>Server</i> |
|----------------------------------|--|--|
| Kebutuhan Perangkat Lunak | <ol style="list-style-type: none">1. Minimum <i>Windows XP</i>2. <i>Browser</i> direkomendasikan <i>Google Chrome</i> | <ol style="list-style-type: none">1. Sistem Operasi minimum <i>Windows XP</i>2. <i>Web Server Apache/2.4</i>3. Versi <i>PHP</i> (5,6-7,3)4. <i>Database MariaDB 10.2, PHPMyadmin</i>5. <i>Composer</i> (1.7.3) |
| Kebutuhan Perangkat Keras | <ol style="list-style-type: none">1. Jaringan internet2. Minimal RAM 2 | <ol style="list-style-type: none">1. Jaringan Internet2. Minimal RAM 8 |

Penjelasan detail mengenai sisi *client* dan *server* akan dijelaskan sebagai berikut:

a) Client

Untuk membuka aplikasi diperlukan untuk mempunyai akses internet dan aplikasi berupa *browser*. Penggunaan *browser* direkomendasikan menggunakan *Google Chrome* agar tampilan terlihat baik dan proses berjalan dengan lancar. Untuk menggunakan *Google Chrome* dibutuhkan minimum *Windows XP* agar aplikasi dapat di *install* dan digunakan.

b) *Server*

Penggunaan kebutuhan pada sisi *server* lebih memperhatikan beberapa fitur seperti *web server Apache v.2.4* dimana penggunaan *web server* ini memiliki beberapa kelebihan seperti dapat lintas platform, kemudahan konfigurasi, dan stabil. Penggunaan versi *PHP* dapat mengatasi jenis versi mulai dari 5.6-7.3. Penggunaan *MariaDB* dan *PHPMyAdmin* yang digunakan untuk mengintegrasikan data. Serta *composer* yang dapat mengelola *dependency* pada *PHP*. Penggunaan kebutuhan telah dicoba dan dirasa dapat mengatasi penggunaan pada aplikasi.

4.1.2. Aplikasi

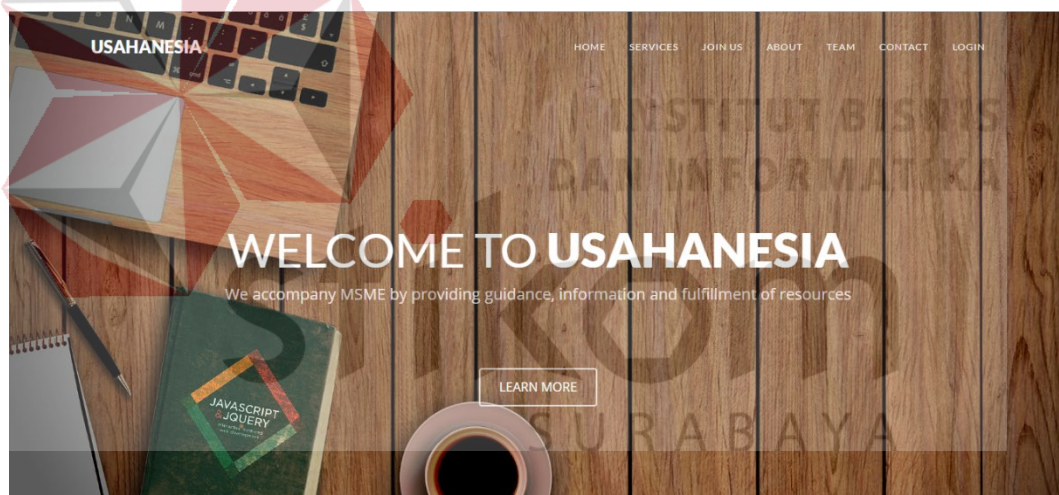
Pada pengerjaan aplikasi ini dibagi menjadi dua kali iterasi sesuai dengan rencana awal. Pada iterasi pertama memiliki tujuan berupa penerapan beberapa metode seperti metode *text mining*, metode *naïve-bayes classifier* (NBC), algoritma *boyer-moore*, dan metode *graphic rating scale* (GRS). Metode *text mining* yang diterapkan terhadap portofolio berkelanjutan UMKM untuk menemukan masalah yang terdapat pada UMKM dan penggunaan metode NBC untuk menentukan apakah UMKM sedang kelebihan atau kekurangan bahan baku. Kemudian penggunaan algoritma *Boyer-Moore* untuk mencari kesamaan kata kunci yang didapat dari hasil *text mining*, hasil pencarian tersebut yang nantinya akan digunakan pada metode GRS untuk dapat memberikan rekomendasi bahan baku yang sesuai dengan kebutuhan UMKM. Beberapa metode dan algoritma tersebut digunakan untuk mendukung dalam melakukan penerapan proses *crowdsourcing* yaitu menghubungkan UMKM yang memiliki kelebihan bahan baku dengan UMKM

yang membutuhkan bahan baku. Pada iterasi pertama ini juga mendapatkan *feedback* untuk memperbaiki sistem dan perubahan tampilan aplikasi.

Pada iterasi kedua menerapkan perbaikan yang terjadi pada iterasi pertama. Beberapa perbaikannya adalah penambahan metode *Jaro Winkler*

A. Halaman Utama/*Home* Usahanesia

Pada Gambar 4.1 terdapat tampilan halaman utama dari Usahanesia. Terdapat beberapa menu pada bagian atas. Untuk melakukan *login* pada aplikasi, pengguna dapat menekan tombol *login* pada bagian menu sebelah kanan.

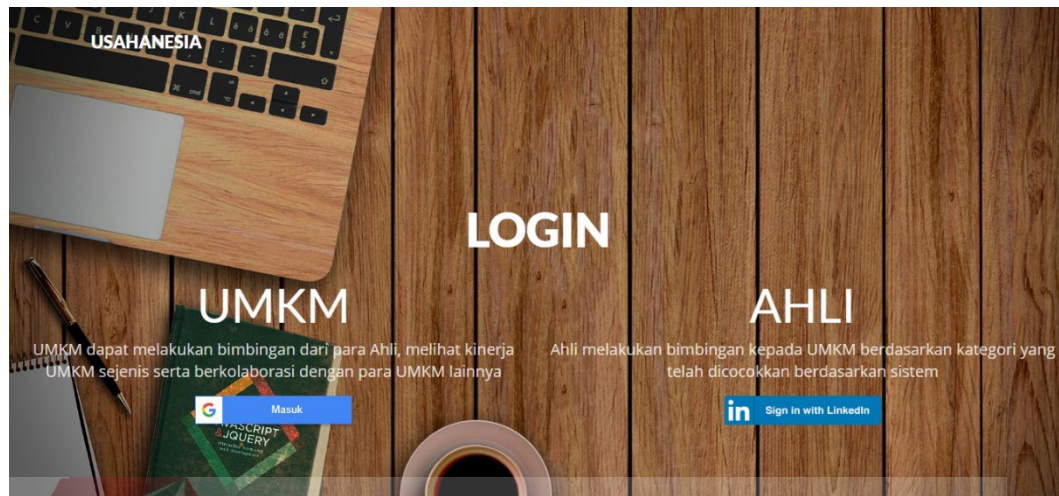


Gambar 4.1 Halaman Utama

B. Halaman *Login* UMKM

Pada halaman *login*, pengguna akan dihadapkan dengan tampilan seperti yang tampak pada Gambar 4.2. Pada tampilan tersebut, pengguna dapat memilih untuk melakukan *login* sebagai UMKM atau sebagai ahli. Baik pengguna baru ataupun pengguna lama dapat memasuki aplikasi dengan cara yang sama. System

yang akan membedakan apakah user termasuk pengguna lama atau pengguna baru yang mewajibkan untuk melakukan registrasi terlebih dahulu.



Gambar 4.2 Halaman *Login*

C. Halaman *Register* UMKM

Apabila pengguna memasuki aplikasi sebagai UMKM maka akan diarahkan untuk mengisi identitas *login* dengan menggunakan akun Google. Apabila akun terverifikasi, user akan diarahkan kepada halaman isi data UMKM untuk melengkapi data seperti yang tampak pada Gambar 4.3.

Setelah mengisi data profil UMKM selanjutnya akan diarahkan kepada halaman pengisian struktur organisasi UMKM seperti yang terdapat pada Gambar 4.4. Pada halaman ini, UMKM dapat mengisi para pemilik atau pekerja yang bekerja pada UMKM tersebut.

Harap Isi Profil Anda Terlebih Dahulu. x

Isi Profil UMKM Anda

Nama UMKM

Nama UMKM

Alamat UMKM

Alamat

Kota

Kota

Provinsi

Provinsi

Email address

isacaeports1@gmail.com

Telepon

No Telp

Bidang

Mikro (Aset Max 50 Juta)

Jenis

Perdagangan

Tahun Berdiri

Tahun Berdiri

Deskripsi

Deskripsi

Upload Foto

Pilih File Tidak ada file yang dipilih

Simpan

Gambar 4.3 Halaman *Register* Profil UMKM

Harap Isi Struktur Organisasi Anda Terlebih Dahulu. x

Isi Struktur Organisasi UMKM Anda

Nama Depan

Nama Depan

Nama Belakang

Nama Belakang

Jenis Kelamin

Pria

Pendidikan Terakhir

S3

Jabatan

Chief Executive Organizer

Simpan

Struktur Organisasi UMKM

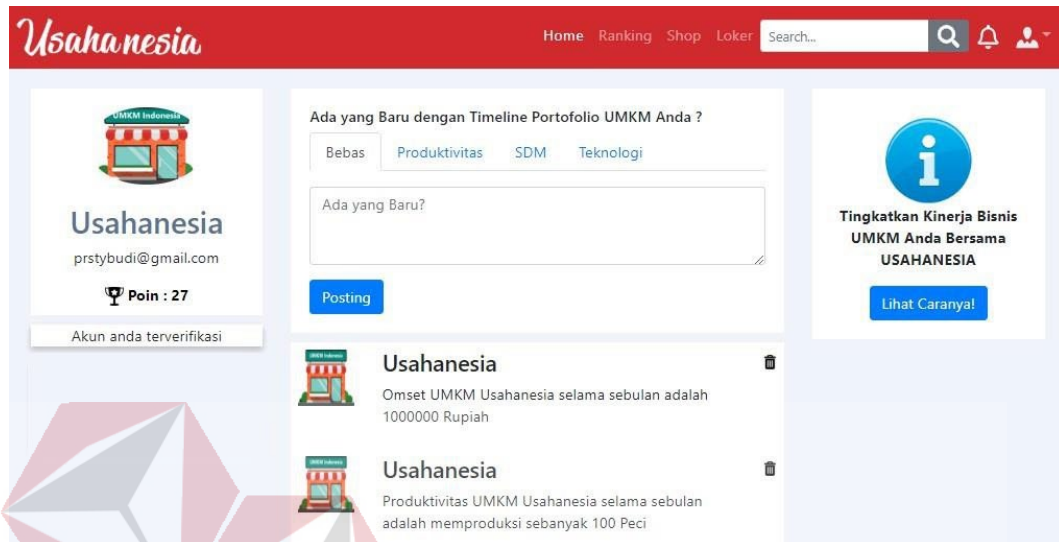
| Nama Lengkap | Jabatan |
|--------------|---------|
| | |

Selesai

Gambar 4.4 Halaman *Register* Struktur Organisasi UMKM

D. Halaman Utama UMKM

Setelah melakukan *login* sebagai UMKM, maka pengguna akan diarahkan pada halaman utama UMKM seperti yang tampak pada gambar 4.5.



Gambar 4.5 Halaman Utama UMKM

E. Menambah Timeline Portfolio

Untuk menambahkan *timeline portfolio* UMKM dapat dilakukan pada halaman utama UMKM seperti yang tampak pada gambar 4.6.



Gambar 4.6 Form Menambahkan *Timeline Portfolio*

F. Menampilkan Hasil Klasifikasi *Timeline Portofolio*

Hasil klasifikasi timeline portfolio berdasarkan hasil iterasi pertama yaitu memberikan rekomendasi ahli yang sesuai dengan hasil kekurangan yang dimiliki oleh ahli berdasarkan timeline portfolio yang dimasukkan oleh UMKM.

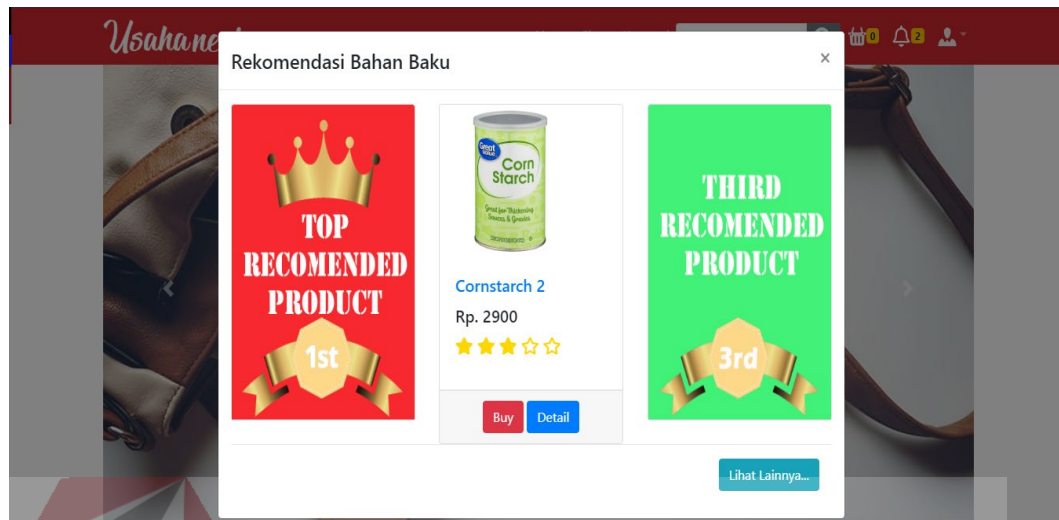
Setelah memasukan timeline portfolio, masukan akan di analisa menggunakan metode *text mining* dan *Naïve bayes classifier* (NBC) sehingga akan menghasilkan keluaran apakah UMKM membutuhkan bahan baku, kelebihan bahan baku atau tidak. Apabila hasil dari *Naïve bayes classifier* menyatakan UMKM membutuhkan bahan baku, maka aplikasi akan menampilkan pemberitahuan kepada UMKM seperti yang tampak pada gambar 4.7.



Gambar 4.7 Pemberitahuan Hasil *Text Mining* & NBC

Selanjutnya, UMKM dapat memilih apakah ingin melihat rekomendasi bahan baku atau tidak. Apabila UMKM ingin melanjutkan, maka sistem akan memproses hasil *text mining* & NBC untuk mencari hasil yang sesuai dengan algoritma *Boyer Moore*. Apabila bahan baku yang sama dengan kebutuhan UMKM

ditemukan, maka sistem akan menampilkan hasil rekomendasi dengan metode *Graphic Rating Scale* tampak seperti gambar 4.8 berikut.



Gambar 4.8 Rekomendasi Bahan Baku

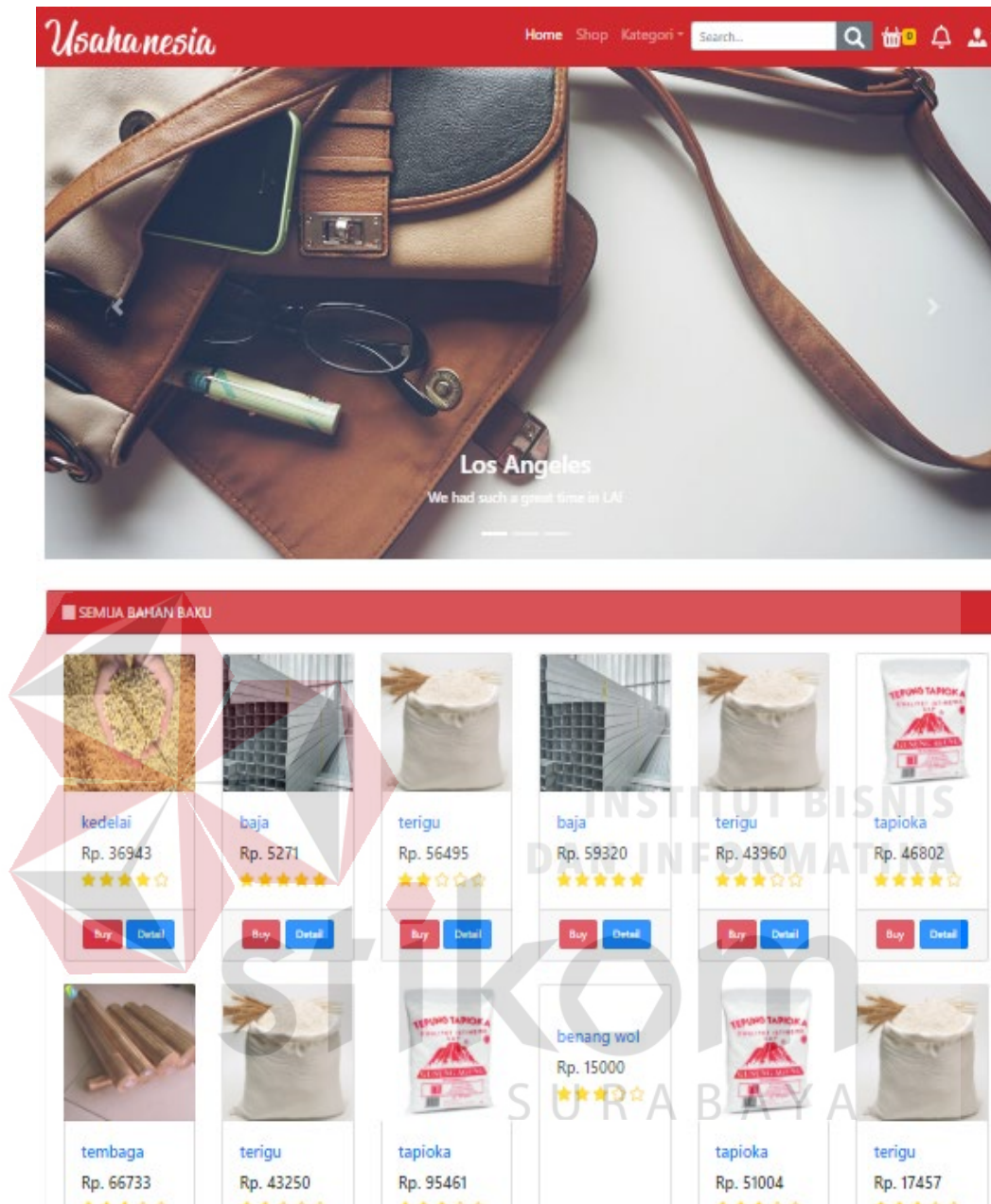
Namun apabila sistem tidak menemukan ahli yang cocok terhadap kebutuhan UMKM tersebut, maka sistem akan menampilkan tampilan seperti pada gambar 4.9.



Gambar 4.9 Pop Up Bahan Baku Tidak Ditemukan

G. Halaman Katalog Bahan Baku

Apabila UMKM memilih menu *shop* pada menu navigasi maka sistem akan langsung menampilkan halaman katalog bahan baku seperti pada gambar 4.10 berikut.



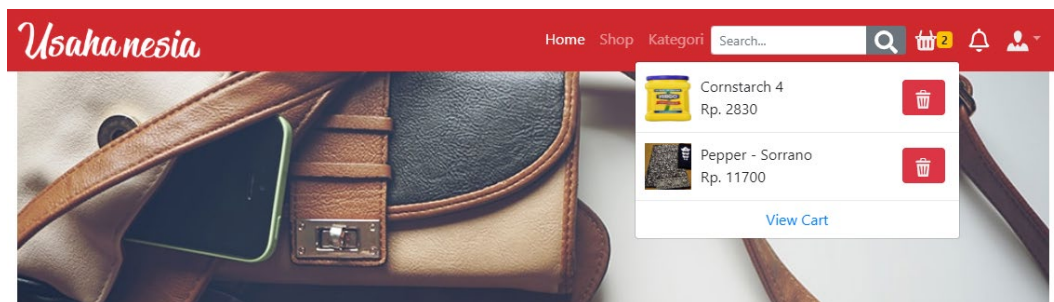
Gambar 4.10 Halaman Katalog Bahan Baku

Kemudian jika UMKM menekan tombol detail pada salah satu bahan baku di halaman katalog maka sistem akan langsung menampilkan halaman detail bahan baku tersebut mulai dari deskripsi bahan baku, *review* bahan baku, UMKM yang menjual bahan baku tersebut hingga bahan baku lain yang dijual oleh UMKM tersebut. Seperti pada gambar 4.11 berikut.



Gambar 4.11 Halaman Detail Bahan Baku

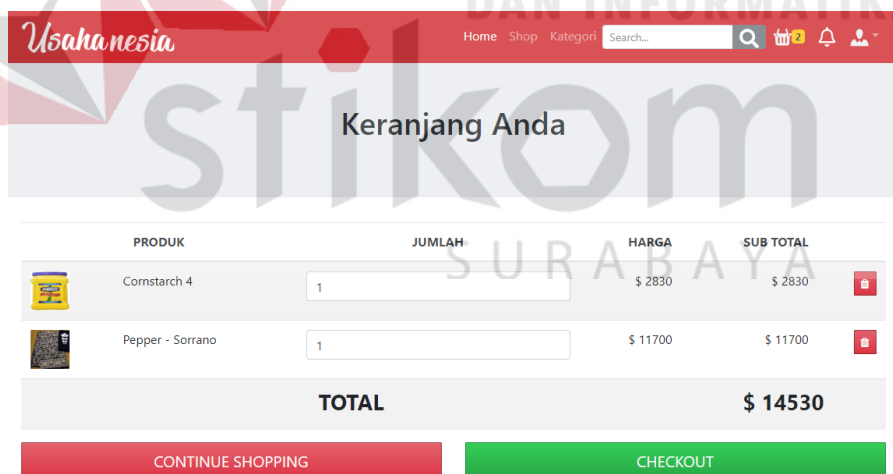
Dan jika UMKM menekan tombol *buy* pada bahan baku di halaman katalog maupun tombol *add to cart* maka sistem akan memasukan bahan baku tersebut kedalam keranjang yang bisa dilihat pada *icon* navigasi keranjang seperti pada gambar 4.12 berikut.



Gambar 4.12 Menu Navigasi Keranjang

H. Halaman Keranjang

Jika pada gambar 4.12 UMKM menekan kata *view cart* maka sistem akan mengarahkan UMKM ke halaman keranjang, dimana pada halaman ini terdapat fitur menghapus bahan baku dari keranjang, mengubah jumlah bahan baku yang ingin dibeli dan menampilkan harga total pembelian bahan baku UMKM tersebut. Seperti yang digambarkan pada gambar 4.13 berikut.



Gambar 4.13 Tampilan Halaman Keranjang

Pada halaman keranjang ini juga terdapat tombol *continue shopping* yang berfungsi untuk mengarahkan UMKM kembali ke halaman katalog untuk melakukan penambahan bahan baku yang ingin dibeli. Sedangkan pada tombol

checkout berfungsi untuk memproses pembelian bahan baku yang dilakukan oleh UMKM tersebut.

I. Halaman Transaksi

Pada aplikasi ini UMKM dapat melihat *history* transaksi yang dilakukan baik transaksi pembelian maupun transaksi penjualan. Pada gambar 4.14 menggambarkan tampilan dari halaman transaksi pembelian yang berisi id pembelian, tanggal pembelian, total pembelian dan status pembelian.

Transaksi

Transaksi Pembelian Transaksi Penjualan

Transaksi Pembelian

| No. | ID Pembelian | Tanggal Pembelian | Total Pembelian | Status Pembelian | |
|-----|--------------|-------------------|-----------------|------------------------------|----------------------------------|
| 1 | 1T74ZWX2IO | 2019-01-20 | Rp. 12700 | Menunggu Persetujuan Penjual | Detail Pembelian |
| 2 | BMQ1UAGJ5W | 2019-01-20 | Rp. 13400 | Menunggu Persetujuan Penjual | Detail Pembelian |
| 3 | T9ICBLWPO4 | 2019-01-20 | Rp. 24400 | Menunggu Pembayaran | Detail Pembelian |

Gambar 4.14 Tampilan Halaman Transaksi Pembelian

Pada gambar 4.14 terdapat tabel transaksi pembelian dimana terdapat fitur untuk melihat detail pembelian dengan memilih detail pembelian. Untuk tampilan dari detail penjualan dapat dilihat pada gambar 4.15 berikut.

Detail Pembelian

No. Transaksi : 1T74ZWX2IO

| No. | ID Bahan Baku | Nama Bahan Baku | Jumlah | Harga Bahan Baku | Sub Total |
|-----|---------------|--------------------------|--------|------------------|-----------|
| 1 | MFH3QCI | Table Cloth 72x144 White | 1 | 12700 | 12700 |

Transaksi Pembelian

| No. | ID Pembelian | Tanggal Pembelian | Total Pembelian | Status Pembelian | |
|-----|--------------|-------------------|-----------------|------------------------------|----------------------------------|
| 1 | 1T74ZWX2IO | 2019-01-20 | Rp. 12700 | Menunggu Pembayaran | Detail Pembelian |
| 2 | BMQ1UAGJ5W | 2019-01-20 | Rp. 13400 | Menunggu Persetujuan Penjual | Detail Pembelian |
| 3 | IAEBL7W9HR | 2019-01-30 | Rp. 14530 | Menunggu Persetujuan Penjual | Detail Pembelian |
| 4 | T9ICBLWPO4 | 2019-01-20 | Rp. 24400 | Menunggu Pembayaran | Detail Pembelian |

Gambar 4.15 Tampilan Halaman Detail Pembelian

Sedangkan pada gambar 4.16 menggambarkan tampilan dari halaman transaksi penjualan. Pada halaman ini terdapat isi yang kurang lebih sama dengan halaman pembelian hanya saja pada halaman ini terdapat tabel untuk mengelola bahan baku yang dijual oleh UMKM.

Transaksi

[Transaksi Pembelian](#) [Transaksi Penjualan](#)

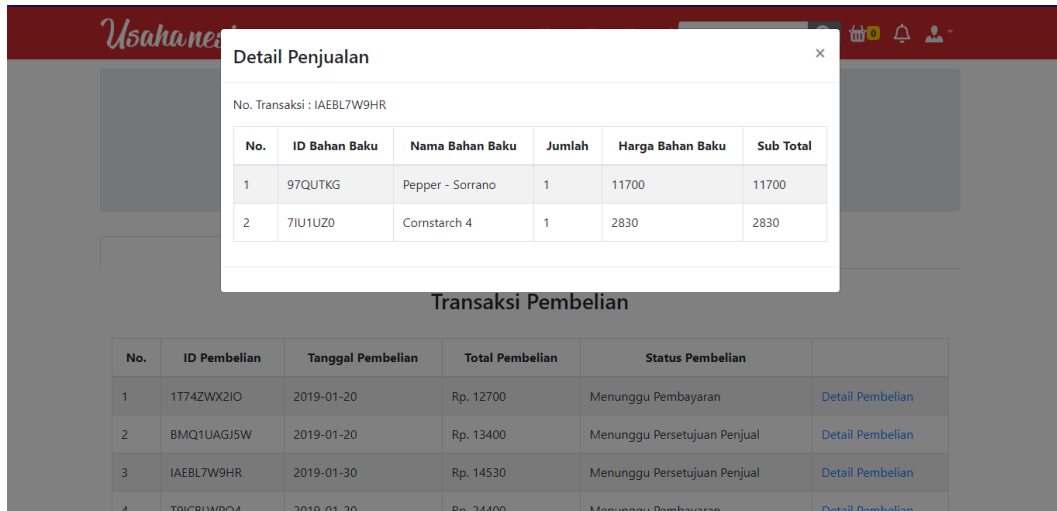
Transaksi Penjualan

| No. | ID Penjualan | Tanggal Penjualan | Total Penjualan | |
|-----|--------------|-------------------|-----------------|----------------------------------|
| 1 | IAEBL7W9HR | 2019-01-30 | Rp. 14530 | Detail Penjualan |
| 2 | T9ICBLWPO4 | 2019-01-20 | Rp. 24400 | Detail Penjualan |

Bahan Baku

Gambar 4.16 Tampilan Halaman Transaksi Penjualan

Pada gambar 4.16 terdapat tabel transaksi penjualan dimana terdapat fitur untuk melihat detail penjualan dengan memilih detail penjualan. Untuk tampilan dari detail penjualan dapat dilihat pada gambar 4.17 berikut.



Detail Penjualan

No. Transaksi : IAEBL7W9HR

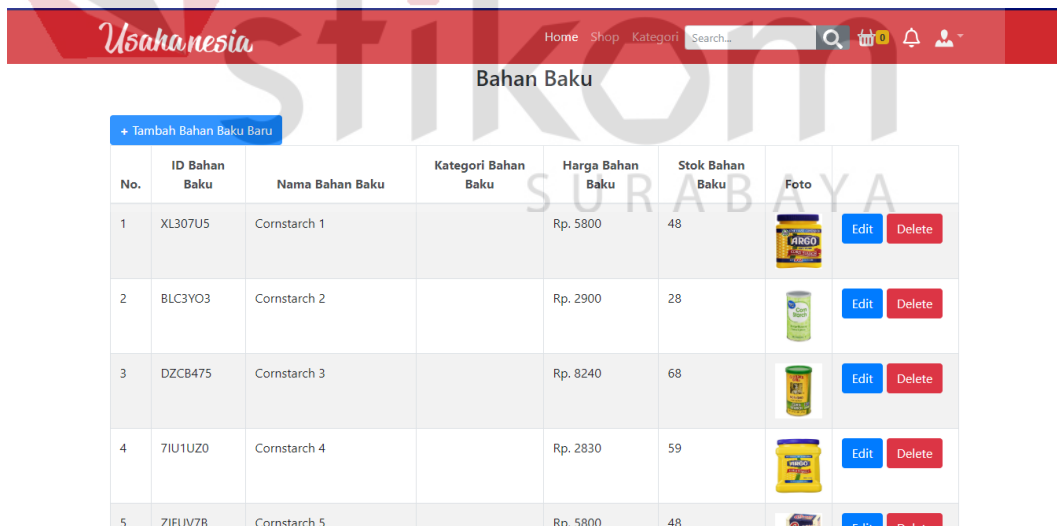
| No. | ID Bahan Baku | Nama Bahan Baku | Jumlah | Harga Bahan Baku | Sub Total |
|-----|---------------|------------------|--------|------------------|-----------|
| 1 | 97QUTKG | Pepper - Sorrano | 1 | 11700 | 11700 |
| 2 | 7IU1UZ0 | Cornstarch 4 | 1 | 2830 | 2830 |

Transaksi Pembelian

| No. | ID Pembelian | Tanggal Pembelian | Total Pembelian | Status Pembelian | |
|-----|--------------|-------------------|-----------------|------------------------------|----------------------------------|
| 1 | 1T74ZX2IO | 2019-01-20 | Rp. 12700 | Menunggu Pembayaran | Detail Pembelian |
| 2 | BMQ1UAGJ5W | 2019-01-20 | Rp. 13400 | Menunggu Persetujuan Penjual | Detail Pembelian |
| 3 | IAEBL7W9HR | 2019-01-30 | Rp. 14530 | Menunggu Persetujuan Penjual | Detail Pembelian |
| 4 | T9ICBLWPO4 | 2019-01-20 | Rp. 24400 | Menunggu Pembayaran | Detail Pembelian |

Gambar 4.17 Tampilan Halaman Detail Penjualan






Halaman kelola bahan baku pada gambar 4.18 ini terdapat 3 fitur yaitu tambah bahan baku dengan memilih tombol +tambah, mengubah bahan baku dengan memilih tombol edit dan menghapus bahan baku dengan memilih tombol delete. Untuk tampilan dari setiap fitur tersebut ada pada gambar 4.19 dan 4.20 berikut.



Usahanesia Home Shop Kategori Search

Bahan Baku

+ Tambah Bahan Baku Baru

| No. | ID Bahan Baku | Nama Bahan Baku | Kategori Bahan Baku | Harga Bahan Baku | Stok Bahan Baku | Foto | |
|-----|---------------|-----------------|---------------------|------------------|-----------------|---|---|
| 1 | XL307U5 | Cornstarch 1 | | Rp. 5800 | 48 |  | Edit Delete |
| 2 | BLC3Y03 | Cornstarch 2 | | Rp. 2900 | 28 |  | Edit Delete |
| 3 | DZCB475 | Cornstarch 3 | | Rp. 8240 | 68 |  | Edit Delete |
| 4 | 7IU1UZ0 | Cornstarch 4 | | Rp. 2830 | 59 |  | Edit Delete |
| 5 | ZIFUV7B | Cornstarch 5 | | Rp. 5800 | 48 |  | Edit Delete |

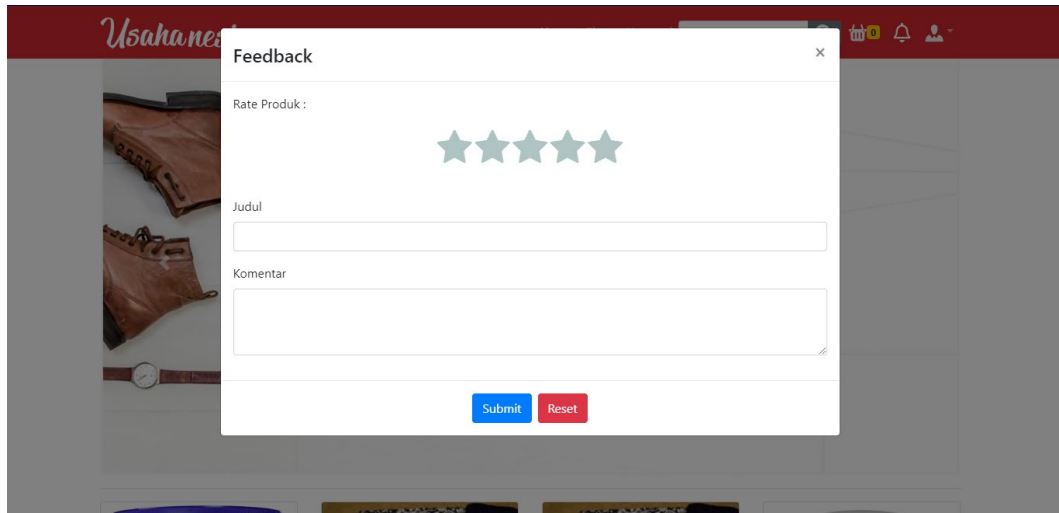
Gambar 4.18 Tampilan Halaman Mengelola Bahan Baku

Gambar 4.19 Tampilan Halaman Tambah Bahan Baku Baru

Gambar 4.20 Tampilan Halaman Ubah Bahan Baku

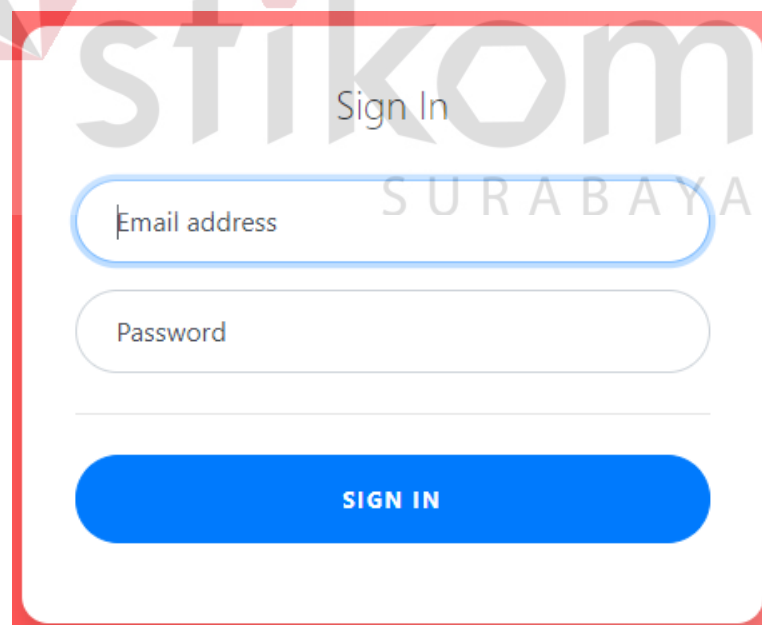
J. Halaman *Review* Bahan Baku

Pada aplikasi ini UMKM dapat memberikan *feedback/review* terhadap bahan baku yang telah dibelinya yang nantinya dapat menjadi bahan pertimbangan UMKM lain dalam membeli bahan baku tersebut. Pada gambar 4.21 menggambarkan tampilan dari halaman *review* bahan baku.

Gambar 4.21 Halaman *Review* Bahan Baku

K. Halaman *Login Admin*

Untuk mengakses halaman *admin* pada aplikasi ini diperlukan otentikasi terlebih dahulu. Proses dilakukan dengan memasukkan alamat email dan sandi dari akun *admin* seperti yang terdapat pada Gambar 4.22.

Gambar 4.22 Halaman *Login Admin*

L. Halaman Mengelola *Bag Of Word*

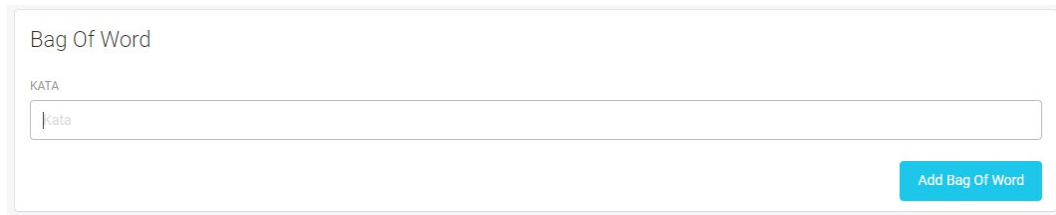
Selain dapat melakukan pengecekan pengguna aktif, *admin* memiliki fitur lainnya yaitu menambahkan *bag of word*. Kegunaan dari pengelolaan *bag of word* ini adalah untuk memperkaya kamus pada sistem guna menunjang kebutuhan pada saat melakukan proses *text mining* pada aplikasi.

Untuk memasuki fitur ini, pengguna dapat memilih menu *bag of word* pada menu navigasi di aplikasi. Dengan demikian sistem akan menunjukkan kata – kata yang dijadikan kamus pada aplikasi ini seperti yang tampak pada Gambar 4.23.



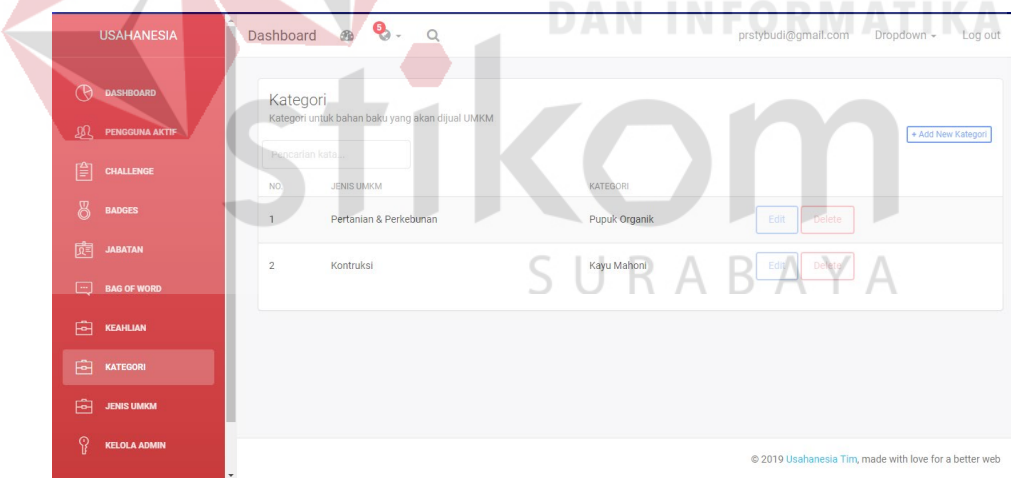
Gambar 4.23 Halaman Mengelola Data *Bag Of Word*

Untuk melakukan penambahan data pada *bag of word*, pengguna dapat melakukan klik [+ Add New Bag of Word](#) pada tombol .Selanjutnya akan muncul form untuk menambahkan *bag of word* seperti yang tampak pada Gambar 4.24.


Gambar 4.24 Tambah Data *Bag Of Word*

M. Halaman Mengelola Kategori

Pada fitur selanjutnya yaitu mengelola data kategori. Untuk memasuki fitur ini, terdapat menu kategori pada sebelah kiri halaman. Data kategori ini digunakan untuk data utama kategori yang akan digunakan oleh UMKM pada saat menambahkan bahan baku yang ingin dijual. Setelah memilih menu kategori pada sebelah kiri, maka akan muncul halaman kategori seperti yang tampak pada Gambar 4.25 berikut.

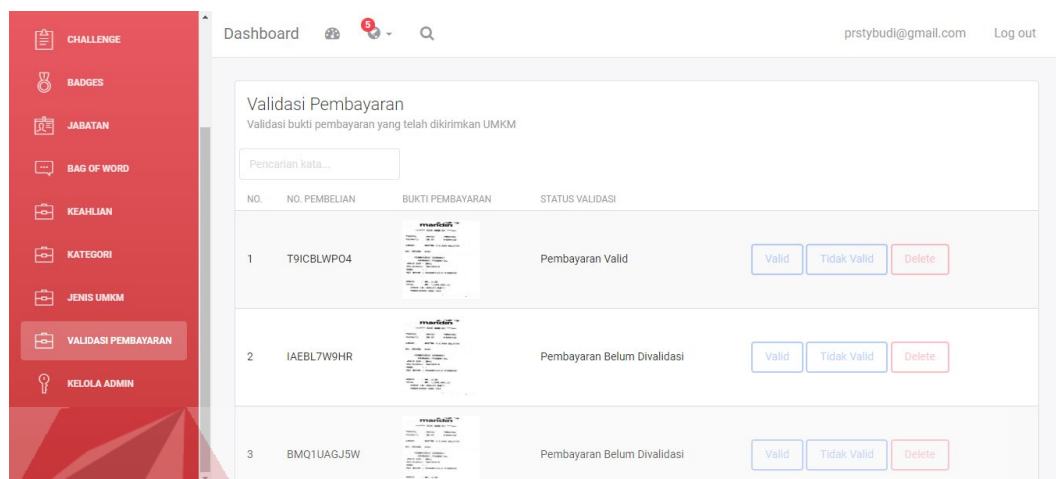


Gambar 4.25 Halaman Mengelola Kategori

N. Halaman Validasi Pembayaran

Pada fitur selanjutnya yaitu melakukan validasi pembayaran. Untuk memasuki fitur ini, terdapat menu kategori pada sebelah kiri halaman. Data validasi

pembayaran ini digunakan agar UMKM dapat melanjutkan proses pembelian. Setelah memilih menu validasi pembayaran pada sebelah kiri, maka akan muncul halaman kategori validasi pembayaran yang tampak pada Gambar 4.26 berikut.



Gambar 4.26 Halaman Validasi Pembayaran

4.2. Evaluasi

4.2.1. Evaluasi Kesesuaian Fitur

Pada akhir pengerjaan telah dihasilkan beberapa fitur yang sesuai dengan *story*. Beberapa fitur yang telah ditulis pada kalangan tertentu seperti *scrum team*, *scrum master*, *product owner*.

Tabel 4.2 Evaluasi kesesuaian fitur dengan *story*

| No. | ID Story | Fitur | Bukti |
|-----|----------|--|-------------|
| 1. | 1 | Fitur: - <i>Login UMKM</i> Deskripsi: Pengguna aplikasi (UMKM) dapat melakukan <i>register</i> dan <i>login</i> dengan menggunakan akun Google mereka. Dan dapat mengambil beberapa data yang terdapat pada akun Google-nya | Gambar 4.2 |
| 2. | 3 | Fitur : - <i>Login Admin</i> Deskripsi : | Gambar 4.16 |

| No. | ID Story | Fitur | Bukti |
|-----|--|--|---|
| | | Pengguna aplikasi (<i>admin</i>) dapat melakukan <i>login</i> dengan menggunakan <i>email</i> dan <i>password</i> yang mereka punya. | |
| 3. | 10, 11, 12, 13, 23, 24, 25, 26 | Fitur : <ul style="list-style-type: none"> - Mengisi <i>Timeline Portofolio</i> - Melihat Klasifikasi Status <i>Timeline Portofolio</i> Deskripsi: Klasifikasi status <i>timeline portofolio</i> dengan menggunakan fitur <i>text mining</i> , <i>Naïve bayes classifier</i> , <i>Boyer Moore</i> , <i>Jaro Winkler</i> dan <i>Graphic Rating Scale</i> . Sehingga menghasilkan keluaran berupa bahan baku yang sesuai dengan kebutuhan dan kelebihan UMKM. | Gambar 4.6, Gambar 4.7, Gambar 4.8, Gambar 4.9 |
| 4. | 2 | Fitur : <ul style="list-style-type: none"> - <i>Register</i> UMKM Deskripsi : Pengguna aplikasi (UMKM) dapat melakukan <i>register</i> dengan mengisikan data-data kelengkapan UMKM mereka seperti : nama UMKM, alamat, nomor telepon, dan sebagainya. | Gambar 4.3, Gambar 4.4 |
| 5. | 14, 15, 16, 17, 18, 19 | Fitur : <ul style="list-style-type: none"> - <i>Pembelian</i> Bahan Baku - Melihat Laporan <i>Pembelian</i> Deskripsi : Pengguna aplikasi (UMKM) dapat melakukan transaksi pembelian mulai dari melihat katalog bahan baku hingga mencetak laporan pembelian. | Gambar 4.10, Gambar 4.11, Gambar 4.12, Gambar 4.13, Gambar 4.14, Gambar 4.15 |
| 6. | 20, 21, 22 | Fitur : <ul style="list-style-type: none"> - <i>Penjualan</i> Bahan Baku - Mengelola Bahan Baku - Melihat Laporan <i>Penjualan</i> Deskripsi : Pengguna aplikasi (UMKM) dapat melakukan penjualan mulai dari mengelola bahan baku yang ingin dijual hingga dapat mencetak laporan penjualan. | Gambar 4.16, Gambar 4.17, Gambar 4.18, Gambar 4.19, Gambar 4.20 |
| 7. | 4, 5 | Fitur : <ul style="list-style-type: none"> - Mengelola Data <i>Bag Of Word</i> Deskripsi : | Gambar 4.24, Gambar 4.25 |

| No. | ID Story | Fitur | Bukti |
|-----|----------|---|-------------|
| | | Pengguna aplikasi (<i>admin</i>) dapat melihat, menambah, dan menghapus data <i>bag of word</i> pada halaman <i>bag of word</i> | |
| 8. | 6, 7 | Fitur : - Mengelola Data Kategori Bahan Baku Deskripsi : Pengguna aplikasi (<i>admin</i>) dapat melihat, menambah, dan menghapus data kategori bahan baku pada halaman kategori bahan baku | Gambar 4.26 |
| 9. | 8, 9 | Fitur : - Validasi Pembayaran Deskripsi : Pengguna aplikasi (<i>admin</i>) dapat melakukan validasi pada bukti pembayaran yang telah dikirim oleh UMKM | Gambar 4.27 |
| 10. | 28 | Fitur : - Menerima Notifikasi Deskripsi : Pengguna aplikasi (UMKM) dapat menerima notifikasi baik dari hasil proses pembelian maupun dari hasil proses penjualan | |
| 11. | 27 | Fitur : - Memberikan <i>Review</i> Bahan Baku Deskripsi : Pengguna aplikasi (UMKM) dapat memberikan <i>feedback</i> berupa <i>review</i> pada bahan baku yang telah ia beli | Gambar 4.21 |

4.2.2. Hasil Testing

Pada tahap ini dilakukan evaluasi terhadap aplikasi pemenuhan bahan baku UMKM dengan melakukan pengujian secara *black box*. Berikut adalah evaluasi yang telah dilakukan untuk lebih detailnya dapat dilihat pada Tabel 4.3. Untuk pengujian pada fungsi melihat klasifikasi status *timeline portfolio* hanya menggunakan 30 data saja yang dapat dilihat lebih detail pada halaman lampiran 3.

Tabel 4.3 *Testing Fungsi*

| No. | Fungsi | | | | |
|-----|---------------------------------------|--|---------------------------------|---|--|
| 1. | Fungsi Login Admin | | | | |
| | Test Case ID | Tujuan | Input | Output Yang Diharapkan | Status |
| | 1. | Melakukan <i>login admin</i> | Email dan password <i>admin</i> | <i>Login</i> berhasil dan sistem menampilkan halaman <i>admin</i> | Sukses (Bukti pada lampiran 2 gambar 1 & gambar 2) |
| | 2. | Melakukan <i>login</i> dengan format salah | Tidak ada input | Muncul pesan : email dan password tidak boleh kosong | Sukses (Bukti pada lampiran 2 gambar 3) |
| | 3. | Melakukan <i>login</i> dengan format salah | Email dan password yang salah | Muncul pesan : email atau password salah | Sukses (Bukti pada lampiran 2 gambar 4) |
| 2. | Fungsi Mengelola Data Kategori | | | | |
| | Test Case ID | Tujuan | Input | Output Yang Diharapkan | Status |
| | 1. | Melakukan tambah data <i>bag of word</i> | Kata <i>bag of word</i> | Data berhasil ditambahkan | Sukses (Bukti pada lampiran 2 gambar 5) |
| | 2. | Melakukan tambah data <i>bag of word</i> dengan format salah | Tidak ada input | Muncul pesan : data tidak boleh kosong | Sukses (Bukti pada lampiran 2 gambar 6) |

| | | | | | |
|----|---------------------------------------|--|---------------------------------|--|--|
| | 3. | Melakukan ubah data <i>bag of word</i> | Kata <i>bag of word</i> | Data berhasil diubah | Sukses (Bukti pada lampiran 2 gambar 7) |
| | 4. | Melakukan ubah data <i>bag of word</i> dengan format salah | Tidak ada input | Muncul pesan : Data tidak boleh kosong | Sukses (Bukti pada lampiran 2 gambar 8) |
| 3. | Fungsi Mengelola Data Kategori | | | | |
| | Test Case ID | Tujuan | Input | Output Yang Diharapkan | Status |
| | 1. | Melakukan tambah data kategori | Jenis UMKM, Kategori Bahan Baku | Data berhasil ditambahkan | Sukses (Bukti pada lampiran 2 gambar 9) |
| | 2. | Melakukan tambah data kategori dengan format salah | Tidak ada input | Muncul pesan : data tidak boleh kosong | Sukses (Bukti pada lampiran 2 gambar 10) |
| | 3. | Melakukan ubah data kategori | Jenis UMKM, Kategori Bahan Baku | Data berhasil diubah | Sukses (Bukti pada lampiran 2 gambar 11) |
| | 4. | Melakukan ubah data kategori dengan format salah | Tidak ada input | Muncul pesan : Data tidak boleh kosong | Sukses (Bukti pada lampiran 2 gambar 12) |
| | Fungsi Validasi Pembayaran | | | | |
| 4. | Test Case ID | Tujuan | Input | Output Yang Diharapkan | Status |

| | | | | | |
|----|-----------------------------|---|---|--|--|
| | 1. | Melakukan validasi pembayaran | Klik tombol valid | Sistem menampilkan pesan : validasi pembayaran berhasil | Sukses (Bukti pada lampiran 2 gambar 13) |
| | 2. | Melakukan penolakan validasi pembayaran | Klik tombol tidak valid | Sistem menampilkan pesan : validasi pembayaran berhasil | Sukses (Bukti pada lampiran 2 gambar 14) |
| 5. | Fungsi Register UMKM | | | | |
| | Test Case ID | Tujuan | Input | Output Yang Diharapkan | Status |
| | 1. | Melakukan <i>register</i> | Nama_Ukm, Provinsi, Alamat, Kota, Telepon, Bidang, Jenis Ukm, Foto, Foto_Ukm, Foto_Surat, | Registrasi UMKM berhasil dan sistem menampilkan halaman utama UMKM | Sukses (Bukti pada lampiran 2 gambar 15 & gambar 16) |
| | 2. | Melakukan <i>register</i> dengan format salah | Tidak ada input | Muncul pesan : data tidak boleh kosong | Sukses (Bukti pada lampiran 2 gambar 17) |
| 6. | Fungsi Login UMKM | | | | |
| | Test Case ID | Tujuan | Input | Output Yang Diharapkan | Status |
| | 1. | Melakukan <i>login</i> UMKM | Email dan password <i>admin</i> | <i>Login</i> berhasil dan sistem menampilkan halaman utama UMKM | Sukses (Bukti pada lampiran 2 gambar 18 & gambar 19) |
| | 2. | Melakukan <i>login</i> dengan format salah | Tidak ada input | Muncul pesan : email dan password tidak boleh kosong | Sukses (Bukti pada lampiran 2 gambar 20) |

| | | | | | |
|----|--|---|-------------------------------|---|--|
| | 3. | Melakukan <i>login</i> dengan format salah | Email dan password yang salah | Muncul pesan : email atau password salah | Sukses (Bukti pada lampiran 2 gambar 21) |
| 7. | Fungsi Mengisi <i>Timeline Portofolio</i> | | | | |
| | Test Case ID | Tujuan | Input | Output Yang Diharapkan | Status |
| | 1. | Mengisi <i>timeline portofolio</i> | Status | Pengisian berhasil dan sistem menampilkan status yang telah dibuat | Sukses (Bukti pada lampiran 2 gambar 22) |
| | 2. | Mengisi <i>timeline portofolio</i> dengan format salah | Tidak ada input | Muncul pesan : status tidak boleh kosong | Sukses (Bukti pada lampiran 2 gambar 23) |
| 8. | Fungsi Melihat Klasifikasi Status <i>Timeline Portofolio</i> (Berdasarkan Dataset 30 Dokumen Pada Lampiran 3) | | | | |
| | Test Case ID | Tujuan | Input | Output Yang Diharapkan | Status |
| | 1. | Melakukan klasifikasi status <i>timeline portofolio</i> | Status | Klasifikasi berhasil dan sistem menampilkan rekomendasi bahan baku | Sukses (Bukti pada lampiran 2 gambar 24 & gambar 25) |
| | 2. | Melakukan klasifikasi status <i>timeline portofolio</i> dengan format salah | Tidak ada input | Menampilkan pesan : status tidak boleh kosong | Sukses (Bukti pada lampiran 2 gambar 26) |
| 9. | 3. | Melakukan klasifikasi status <i>timeline portofolio</i> dengan format salah | Status yang salah | Klasifikasi gagal dan sistem tidak menampilkan rekomendasi bahan baku | Sukses (Bukti pada lampiran 2 gambar 27) |
| | Fungsi Pembelian Bahan Baku | | | | |

| | Test Case ID | Tujuan | Input | Output Yang Diharapkan | Status |
|------------------------------------|--------------|--|----------------------------------|--|--|
| | 1. | Melihat detail pembelian | Menekan tombol detail bahan baku | Sistem menampilkan halaman detail bahan baku | Sukses (Bukti pada lampiran 2 gambar 29) |
| | 2. | Melanjutkan belanja setelah memasukkan bahan baku ke keranjang | Menekan tombol lanjutkan belanja | Sistem menampilkan halaman katalog bahan baku | Sukses (Bukti pada lampiran 2 gambar 30 & gambar 31) |
| | 3. | Menyelesaikan pembelian bahan baku | Menekan tombol selesai belanja | Sistem menampilkan <i>pop up</i> pembelian sedang diproses | Sukses (Bukti pada lampiran 2 gambar 32 & gambar 33) |
| Fungsi Review Bahan Baku | | | | | |
| | Test Case ID | Tujuan | Input | Output Yang Diharapkan | Status |
| 10. | 1. | Melakukan pemberian rating bahan baku | Rating, Judul, Komentar | <i>Review</i> berhasil di simpan dan sistem menampilkan <i>review</i> bahan baku | Sukses (Bukti pada lampiran 2 gambar 34 & gambar 35) |
| | 2. | Melakukan pemberian <i>review</i> bahan baku dengan format salah | Tidak ada input | Muncul pesan : data tidak boleh kosong | Sukses (Bukti pada lampiran 2 gambar 36) |
| Fungsi Penjualan Bahan Baku | | | | | |
| 11. | Test Case ID | Tujuan | Input | Output Yang Diharapkan | Status |

| | | | | | |
|-----|-------------------------------|--|--|--|--|
| | 1. | Melakukan tambah data bahan baku | Nama_Produk, Deskripsi_Produk, Harga_Produk, Stok_Produk, Satuan, Foto_Produk, Kategori_Produk | Data bahan baku berhasil ditambahkan | Sukses (Bukti pada lampiran 2 gambar 37 & gambar 38) |
| | 2. | Melakukan tambah data bahan baku dengan format salah | Tidak ada input | Muncul pesan : data tidak boleh kosong | Sukses (Bukti pada lampiran 2 gambar 39) |
| | 3. | Melakukan ubah data bahan baku | Nama_Produk, Deskripsi_Produk, Harga_Produk, Stok_Produk, Satuan, Foto_Produk, Kategori_Produk | Data bahan baku berhasil diubah | Sukses (Bukti pada lampiran 2 gambar 40) |
| | 4. | Melakukan ubah data bahan baku dengan format salah | Tidak ada input | Muncul pesan : Data tidak boleh kosong | Sukses (Bukti pada lampiran 2 gambar 41) |
| 12. | Fungsi Melihat Laporan | | | | |
| | Test Case ID | Tujuan | Input | Output Yang Diharapkan | Status |
| | 1. | Melakukan penyetakan laporan pembelian | Menekan tombol cetak pada halaman pembelian | Sistem menampilkan halaman <i>print</i> laporan. | Sukses (Bukti pada lampiran 2 gambar 42 & gambar 43) |
| | 2. | Melakukan penyetakan laporan penjualan | Menekan tombol cetak pada halaman penjualan | Sistem menampilkan halaman <i>print</i> laporan. | Sukses (Bukti pada lampiran 2 gambar 44 & gambar 45) |

Berdasarkan hasil pengolahan pengujian *black box* yang telah dilakukan sebelumnya maka persentase keberhasilan dari aplikasi pemenuhan bahan baku UMKM menggunakan metode *text mining* ini dapat dilihat pada Tabel 4.4.

Tabel 4.4 Hasil *Testing*

| No. | Fungsi | Test Case | Test Case Yang Berhasil | Persentase Keberhasilan |
|--------|--|-----------|-------------------------|-------------------------|
| 1. | Fungsi <i>Login Admin</i> | 3 | 3 | 100% |
| 2. | Fungsi Mengelola Data <i>Bag Of Word</i> | 4 | 4 | 100% |
| 3. | Fungsi Mengelola Data Kategori | 4 | 4 | 100% |
| 4. | Fungsi Validasi Pembayaran | 2 | 2 | 100% |
| 5. | Fungsi <i>Register</i> UMKM | 2 | 2 | 100% |
| 6. | Fungsi <i>Login</i> UMKM | 3 | 3 | 100% |
| 7. | Fungsi Mengisi <i>Timeline Portofolio</i> | 2 | 2 | 100% |
| 8. | Fungsi Melihat Klasifikasi Status <i>Timeline Portofolio</i> | 3 | 3 | 100% |
| 9. | Fungsi Pembelian Bahan Baku | 3 | 3 | 100% |
| 10. | Fungsi <i>Review</i> Bahan Baku | 2 | 2 | 100% |
| 11. | Fungsi Penjualan Bahan Baku | 4 | 4 | 100% |
| 12. | Fungsi Melihat Laporan | 2 | 2 | 100% |
| Jumlah | | 34 | 34 | 100% |

BAB V

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Hasil dari tugas akhir ini adalah aplikasi pemenuhan bahan baku UMKM menggunakan *text mining* berbasis web. Aplikasi ini dapat menghasilkan rekomendasi bahan baku yang sesuai dengan kebutuhan UMKM dengan menggunakan metode penerapan *text mining* dan *graphic rating scale* serta penerapan beberapa algoritma seperti *naïve bayes classifier*, *boyer moore*, dan *jaro winker*.

Aplikasi pemenuhan bahan baku UMKM menggunakan *text mining* berbasis web ini telah diuji dengan pengujian *black box*. Hasil evaluasi aplikasi penerapan *text mining* untuk pemenuhan bahan baku UMKM berbasis web ini memiliki tingkat keberhasilan sebesar 100% dari 34 *test case*. Sehingga Aplikasi pemenuhan bahan baku UMKM menggunakan *text mining* berbasis web ini telah sesuai dengan kebutuhan pengguna (*Story*).

5.2. Saran

Aplikasi penerapan *text mining* untuk pemenuhan bahan baku UMKM berbasis web ini dapat dikembangkan dari sisi penerapan *text mining* dan algoritma lainnya untuk menemukan bahan baku yang sesuai dengan kebutuhan UMKM bukan hanya untuk menemukan bahan baku, namun kebutuhan UMKM keseluruhan dengan perhitungan yang tepat.

Adapun saran untuk peneliti selanjutnya adalah :

1. Aplikasi ini masih dalam bentuk website, sehingga dapat dikembangkan lebih lanjut pada *platform android* dan *ios*.
2. Berdasarkan pada batasan masalah aplikasi ini tidak mencakup metode pengiriman dan metode pembayaran hanya melalui transfer antar bank, sehingga dapat dikembangkan lebih lanjut dengan metode pengiriman dengan kurir ekspedisi dan penggunaan metode pembayaran lain seperti kartu kredit, *Dana*, dll.
3. Untuk meningkatkan akurasi metode *naïve bayes classifier* maka dibutuhkan data setiap kategori yang lebih banyak.



DAFTAR PUSTAKA

- Agung, H., & Yogyakarta. (2016). Implementasi *Boyer-Moore* Pada Aplikasi Pencarian Rumus Matematika Dan Fisika. *Jurnal Ilmiah Teknologi Informasi Terapan*, 74-85.
- Appelo, J. (2011). *Management 3.0 Leading Agile Developers, Developing Agile Leaders*. Boston: Pearson Education, Inc.
- Fiege, J. (2016, Februari 2). Dipetik 12 28, 2018, dari <https://github.com/fieg/bayes>
- Hendini, A. (2016). Pemodelan Uml Sistem Informasi Monitoring Penjualan Dan Stok Barang (Studi Kasus: Distro Zhezha Pontianak). *Jurnal Khatulistiwa Informatika*, IV(2), 107-116.
- International Scrum Institute. (t.thn.). *Scrum Revealed Training Book Second Edition*.
- Khan, M. E. (2011). Different Approaches To Black Box Testing Technique For Finding Errors. *International Journal of Software Engineering & Applications (IJSEA)*, 2(4), 31-41.
- Kumar, L., & Bhatia, P. K. (2013). *Text Mining: Concepts, Process And Applications*. *Journal of Global Research in Computer Science*, 36-39.
- Kurniawan, B., Effendi, S., & Sitompul, O. S. (2012). Klasifikasi Konten Berita Dengan Metode *Text Mining*. *Jurnal Dunia Teknologi Informasi*, 14-19.

- Lembaga Pengembangan Perbankan Indonesia & Bank Indonesia. (2015). *Buku Profil Bisnis Usaha Mikro, Kecil, dan Menengah (UMKM)*. Dipetik Agustus 2, 2018, dari <http://www.bi.go.id/id/umkm/penelitian/nasional/kajian/Documents/Profil%20Bisnis%20UMKM.pdf>
- Merina, N. (2016, Agustus 29). *GOukm.id*. Dipetik 10 19, 2018, dari Kamus Istilah Dunia UKM, Bisnis Dan Pemasaran: <http://goukm.id/kamus-bisnis-pemasaran-ukm/>
- Okta'mal, F., Saptono, R., & Sulisty, M. E. (2015). JARO-WINKLER DISTANCE DAN STEMMING UNTUK DETEKSI DINI HAMA DAN PENYAKIT PADI. *Seminar Nasional Sistem Informasi Indonesia*, 305-312.
- Pramono, T. (2015). *Statistik Industri Manufaktur (Bahan Baku) Provinsi Jawa Timur 2015*. Badan Pusat Statistik Provinsi Jawa Timur.
- Ratniasih, N. L., Sudarma, M., & Gunantara, N. (2017). Penerapan *Text Mining* dalam Spam Filtering untuk Aplikasi Chat. *Teknologi Elektro, Vol. 16, No. 3*, 14-15.
- Schwaber, K., & Sutherland, J. (2014). *Scrum-Guide-ID*. Diambil kembali dari <http://www.scrumguides.org/docs/scrumguide/v1/Scrum-Guide-ID>: <http://www.scrumguides.org>
- Shi , M. (2010). Software Functional Testing from the Perspective of Business Practice. *Computer and Information Science*, 3(4), 49-52.

- Sulistiyawan, F., Wuriyanto, T., & Subianto, T. Y. (2013). Sistem Informasi Penilaian Kinerja Pegawai Menggunakan Metode *Graphic Rating Scales* Dan 360 Derajat. *JSIKA* 2, 7-13.
- Syarif, M. (2017). Implementasi Algoritma String Matching Dalam Pencarian Surah Dan Ayat Dalam Al-Quran Berbasis Web. *Indonesian Journal on Networking and Security*, 6(2), 70-76.
- Whitla, P. (2009). *Crowdsourcing* and Its Application in Marketing Activities. *Contemporary Management Research*, Vol. 5, 15-28.

