

BAB IV

IMPLEMENTASI DAN EVALUASI SISTEM

4.1 Kebutuhan Sistem

Tahap kebutuhan sistem merupakan tahap menjelaskan kebutuhan-kebutuhan yang harus dipenuhi untuk menguji coba *user interface* serta menjalankan aplikasi. Adapun kebutuhan sistem ini harus dipenuhi sehingga aplikasi ini dapat berjalan dengan baik. Kebutuhan sistem tersebut yaitu:

1. Kebutuhan Perangkat Keras: Kebutuhan perangkat keras merupakan komponen fisik peralatan yang membentuk suatu sistem komputer, serta peralatan-peralatan lain yang mendukung komputer dalam menjalankan tugasnya. Perangkat keras yang digunakan sebagai penyedia harus memiliki kinerja yang baik, sehingga aplikasi yang tersedia dapat diakses dengan baik user. Spesifikasi perangkat keras yang dibutuhkan adalah:

- a. laptop dengan prosessor CPU 2.30 GHz.
- b. Memori 2 gigabytes (GB) RAM.
- c. VGA 802MB 32bit dengan resolusi 1366 x 768
- d. Printer
- e. Mouse

2. Kebutuhan Perangkat Lunak: Perangkat lunak adalah suatu program komputer yang diperlukan para pengguna untuk mengoperasikan fungsi dari perangkat

keras. Adapun perangkat lunak yang digunakan dalam pembuatan aplikasi ini adalah:

- a. Sistem Operasi *Microsoft Windows 7*.
- b. *Microsoft SQL Server 2008*
- c. *Microsoft Visual Studio 2010*

4.2 Pembuatan Aplikasi dan Instalasi Aplikasi

Tahapan ini dilakukan setelah melakukan pembuatan desain dan menjelaskan kebutuhan sistem. Tahapan ini menjelaskan tentang tahapan membuat aplikasi dan tahapan instalasi aplikasi. Aplikasi perhitungan kelayakan investasi ini dibuat dengan menggunakan bahasa pemrograman Visual Studio 2010. Aplikasi ini hanya digunakan untuk satu komputer atau laptop. Untuk tahapan membuat aplikasi sebagai berikut :

1. Pembuatan *Database*

Database yang digunakan untuk pembuatan program ini menggunakan *database SQL Server 2008*. Dengan rincian tabel ada 22 tabel.

2. Coding Aplikasi

Sebelum melakukan coding, aplikasi ini terdiri dari 22 *form* dan terdapat 3 modul/proses inti utama, yaitu :

- Modul/proses menghitung aliran kas bersih dan laba rugi.
- Modul/proses menghitung penilaian investasi
- Modul/proses kelayakan investasi

a. Modul/proses menghitung aliran kas bersih dan laba rugi

Diproses ini terdapat sub proses menghitung penyusutan yang dilakukan di *form* mesin, sub proses menghitung rencana pesanan produksi yang dilakukan di *form* rencana produksi, sub proses menghitung biaya produksi yang dilakukan di *form* biaya produksi, sub proses menghitung laba rugi yang dilakukan di *form* laba rugi dan akan menghasilkan laporan laba rugi di *form* laporan laba rugi serta sub proses menghitung aliran kas bersih yang dilakukan di *form* aliran kas bersih dan akan menghasilkan laporan di *form* laporan aliran kas bersih.

b. Modul/proses menghitung penilaian investasi

Diproses ini terdapat sub proses menghitung *average rate of return* yang dilakukan di *form* hitung *ARR* dan menghasilkan laporan di *form* laporan *ARR*, sub proses menghitung *payback period* yang dilakukan di *form* hitung *PP* dan menghasilkan laporan di *form* laporan *PP*, sub proses menghitung *net present value* yang dilakukan di *form* hitung *NPV* dan menghasilkan laporan di *form* laporan *NPV*, sub proses menghitung *internal rate of return* yang dilakukan di *form* hitung *IRR* dan menghasilkan laporan *IRR* di *form* laporan *IRR* serta sub proses menghitung *profitability index* yang dilakukan di dalam *form* hitung *PI* dan akan menghasilkan laporan *PI* di *form* laporan *PI*.

c. Modul/proses kelayakan investasi

Proses ini adalah proses untuk membandingkan hasil perhitungan dengan yang ditargetkan. Untuk melakukan proses kelayakan investasi dibutuhkan *form* kelayakan investasi yang akan menghasilkan laporan kelayakan investasi di *form* kelayakan investasi.

Setelah menjelaskan tentang alur dari modul/proses aplikasi dan *form* maka selanjutnya dilakukan tahapan coding aplikasi. Coding aplikasi adalah memberikan nilai/fungsi di *form* aplikasi agar aplikasi bisa berjalan atau berfungsi sesuai dengan alur yang sudah dirancang.

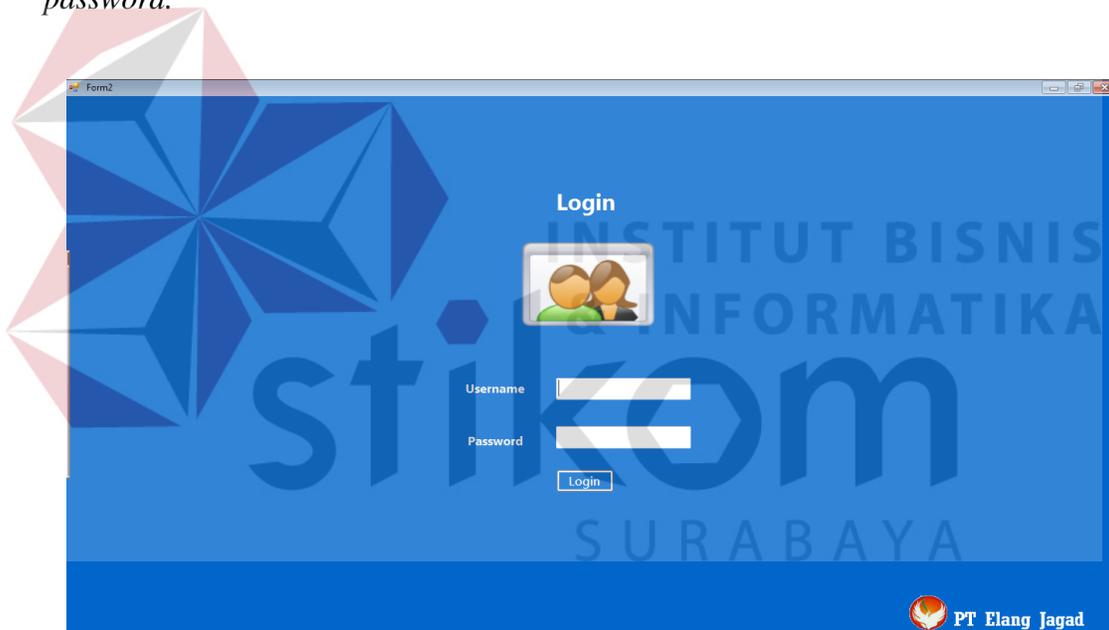
3. Instalasi Aplikasi

Untuk menjalankan aplikasi tersebut agar berjalan dengan baik diperlukan tahapan-tahapan install, tahapan instalasi sebagai berikut :

- a. Install sistem operasi *Microsoft Windows 7* pada laptop atau komputer.
- b. Install aplikasi perhitungan kelayakan investasi mesin pada laptop atau komputer.
- c. Install aplikasi *database* microsoft SQL Server 2008 pada laptop atau komputer.
- d. *Attacth database* aplikasi perhitungan kelayakan investasi pada laptop atau komputer.
- e. Kemudian jalankan aplikasi perhitungan kelayakan investasi.

4.3 Uji Coba *User Interface*

Setelah kebutuhan sistem terpenuhi, langkah selanjutnya adalah mengimplementasikan ke dalam Aplikasi Perhitungan Kelayakan Investasi Aset. Pada saat sistem dijalankan yang pertama kali tampil adalah tampilan *login* seperti pada gambar 4.1. Dari tampilan ini, pengguna harus *login* terlebih dulu dengan memasukkan *username* dan *password* agar dapat melanjutkan ke halaman menu selanjutnya dengan menekan tombol *login* atau menekan enter pada setelah mengisi *password*.



Gambar 4.1 Tampilan *Login*

4.3.1 Tampilan Menu Utama

Tampilan menu utama pada gambar 4.2 ini berisi tampilan menu-menu yaitu menu *form* master, menu periode, menu rencana produksi, menu biaya produksi, menu laba rugi, menu aliran kas bersih, menu hitung *ARR*, menu hitung *PP*, menu

hitung *NPV*, menu hitung *IRR*, menu hitung *PI*, menu kelayakan investasi, menu laporan aliran kas bersih, menu laporan laba rugi, menu laporan *ARR*, menu laporan *PP*, menu laporan *NPV*, menu laporan *IRR*, menu laporan *PI*, menu laporan kelayakan investasi serta menu *logout*. Jika salah satu menu tersebut diklik maka akan muncul *form*.

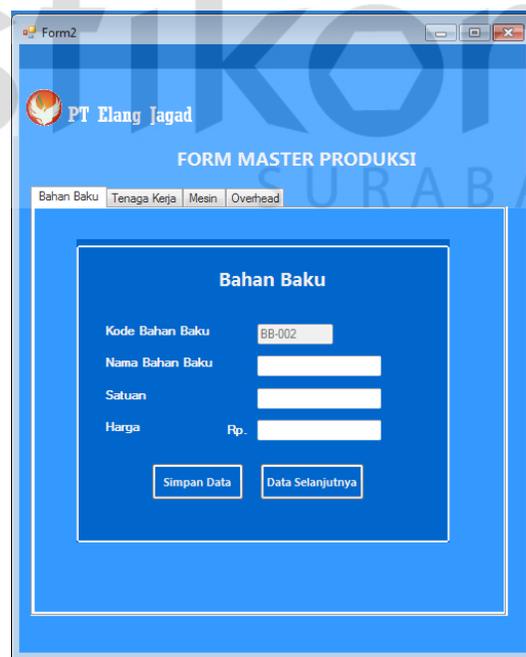


Gambar 4.2 Tampilan Menu Utama

4.3.2 Tampilan *Form* Master

Tampilan *form* master muncul ketika pengguna memilih *icon form* master pada menu utama dan sebelumnya sudah melakukan *login*. Tampilan *form* master ini terbagi menjadi 4 *tab page* yaitu master bahan baku, master tenaga kerja, master mesin serta master *overhead*. Di *form* master bahan baku terdapat 4 *textbox* : (Kode bahan baku, nama bahan baku, satuan, serta harga) dan 2 *button* : *button* simpan untuk menyimpan data bahan baku baru, *button* data selanjutnya untuk membersihkan *textbox* dan melanjutkan ke data berikutnya. Di *form* master tenaga

kerja terdapat 3 *textbox* : (Kode tenaga kerja, nama tenaga kerja, serta upah) 1 *combobox* yaitu jenis tenaga kerja serta 2 *button* : *button* simpan untuk menyimpan data tenaga kerja, *button* data selanjutnya untuk membersihkan *textbox* dan melanjutkan ke data berikutnya. Di *form* master mesin terdapat 8 *textbox*, 1 *numericupdown* serta 2 *button* yaitu *button* simpan untuk menyimpan data tenaga kerja, *button* data selanjutnya untuk membersihkan *textbox* dan melanjutkan ke data berikutnya. Di *form* master *overhead* terdapat 3 *textbox*, 1 *combobox* yaitu jenis *overhead* serta 2 *button* : *button* simpan untuk menyimpan data tenaga kerja, *button* data selanjutnya untuk membersihkan *textbox* dan melanjutkan ke data berikutnya. Adapun tampilan *form* master bahan baku dapat dilihat pada gambar 4.3, *form* master tenaga kerja dapat dilihat pada gambar 4.4, *form* master mesin dapat dilihat pada gambar 4.5 serta *form* master *overhead* dapat dilihat pada gambar 4.6.



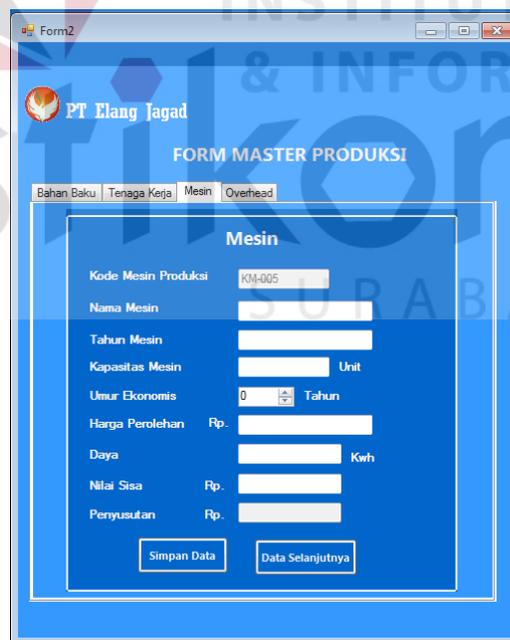
The image shows a screenshot of a web application window titled 'Form2'. The application is for 'PT Elang Jagad' and is titled 'FORM MASTER PRODUKSI'. It has four tabs: 'Bahan Baku', 'Tenaga Kerja', 'Mesin', and 'Overhead'. The 'Bahan Baku' tab is active. The form contains the following fields and buttons:

- Kode Bahan Baku:** A text box containing 'BB-002'.
- Nama Bahan Baku:** An empty text box.
- Satuan:** An empty text box.
- Harga:** A text box with 'Rp.' to its left.
- Buttons:** 'Simpan Data' and 'Data Selanjutnya'.

Gambar 4.3 Tampilan *Form* Bahan Baku *Tab Page* 1



The screenshot shows a web application window titled 'Form2'. At the top left is the logo of PT Elang Jagad. The main heading is 'FORM MASTER PRODUKSI'. Below this is a navigation bar with four tabs: 'Bahan Baku', 'Tenaga Kerja', 'Mesin', and 'Overhead'. The 'Tenaga Kerja' tab is selected. The form itself is titled 'Tenaga Kerja' and contains the following fields: 'Kode Tenaga Kerja' with the value 'TK-006', 'Jenis Tenaga Kerja' with a dropdown arrow, 'Nama Tenaga Kerja' with an empty text box, and 'Upah' with a sub-label 'Rp.' and 'PerHari'. At the bottom of the form are two buttons: 'Simpan Data' and 'Data Selanjutnya'.

Gambar 4.4 Tampilan *Form* Tenaga Kerja *Tab Page 2*

The screenshot shows the same web application window as in Gambar 4.4, but with the 'Mesin' tab selected. The form is titled 'Mesin' and contains the following fields: 'Kode Mesin Produksi' with the value 'KM-005', 'Nama Mesin' with an empty text box, 'Tahun Mesin' with an empty text box, 'Kapasitas Mesin' with an empty text box and the unit 'Unit', 'Umur Ekonomis' with the value '0' and the unit 'Tahun', 'Harga Perolehan' with a sub-label 'Rp.' and an empty text box, 'Daya' with an empty text box and the unit 'Kwh', 'Nilai Sisa' with a sub-label 'Rp.' and an empty text box, and 'Perawatan' with a sub-label 'Rp.' and an empty text box. At the bottom of the form are two buttons: 'Simpan Data' and 'Data Selanjutnya'.

Gambar 4.5 Tampilan *Form* Mesin *Tab Page 3*

Gambar 4.6 Tampilan *Form Overhead Tab Page 4*

4.3.3 Tampilan periode

Tampilan data periode muncul ketika pengguna memilih *icon* periode pada menu utama dan sebelumnya sudah melakukan *login*. Tampilan periode ini terbagi menjadi dua *tab* yaitu periode rencana produksi dan periode investasi. Di *form* periode rencana produksi terdapat: 1 *numericupdown* dan 4 *button* yaitu: *button* simpan untuk menyimpan data periode, *button* ubah untuk mengubah data periode jika ada perubahan, *button* hapus untuk menghapus data periode, *button* cari untuk mencari data periode yang ingin diubah atau dihapus dan terdapat 3 *textbox* yaitu: *textbox* kode periode yang otomatis terisi sendiri, *textbox* bulan serta *textbox* cari kode periode. Di *form* periode investasi terdapat : 1 *numericupdown* dan 4 *button* yaitu: *button* simpan untuk menyimpan data periode, *button* ubah untuk mengubah

data periode jika ada perubahan, *button* hapus untuk menghapus data periode, *button* cari untuk mencari data periode yang ingin diubah atau dihapus dan terdapat 2 *textbox* yaitu: *textbox* kode periode yang otomatis terisi sendiri serta *textbox* cari kode periode. Adapun tampilan periode dapat dilihat pada gambar 4.7 dan gambar 4.8.

Kode_Periode_Proc	Periode_Produksi	Bulan
P-016	16	april
P-017	17	mei
P-018	18	juni
P-019	19	juli
P-020	20	agustus
P-021	21	september

Gambar 4.7 Tampilan Periode Produksi *Tab Page 1*

Kode_Periode_Inve	Periode_Investasi
P-001	1
P-002	2
P-003	3
P-004	4
*	

Gambar 4.8 Tampilan Periode Investasi *Tab Page 2*

4.3.4 Tampilan Rencana Produksi

Tampilan rencana produksi muncul ketika pengguna memilih *icon* rencana produksi pada menu utama dan sebelumnya sudah melakukan *login*. Tampilan rencana produksi pada gambar 4.9, gambar 4.10 dan gambar 4.11 ini terbagi menjadi tiga bagian *tab page* yaitu *tab page* perhitungan *history* permintaan pesanan, *tab page* perhitungan mencari koefisien regresi dan harga konstan serta *tab page* perhitungan peramalan produksi.

Di *form tab page* perhitungan *history* permintaan pesanan ini terdapat: 10 *textbox* yaitu kode rencana produksi yang otomatis akan terisi kodenya, kode periode yang otomatis terisi kode periodenya, periode yang otomatis terisi periodenya disesuaikan dengan kode periode, permintaan pesanan, hasil perkalian periode yang otomatis terisi hasilnya dari perhitungan antara permintaan pesanan dengan periode, hasil kuadrat periode yang otomatis terisi sendiri hasilnya dari kuadrat periode, akumulasi periode yang otomatis terisi sendiri hasilnya dari akumulasi inputan perperiode, akumulasi permintaan pesanan yang otomatis terisi sendiri hasilnya dari akumulasi permintaan pesanan perperiode, akumulasi perkalian periode yang otomatis terisi sendiri hasilnya dari akumulasi perkalian permintaan pesanan perperiode, akumulasi kuadrat periode yang otomatis terisi sendiri hasilnya dari akumulasi kuadrat perperiode dan terdapat 3 *button* yaitu: *button* hitung untuk menghitung permintaan pesanan dikali dengan periode serta untuk menghitung kuadrat periode, *button* simpan perperiode untuk menyimpan perperiodenya dari kode rencana produksi, kode periode, permintaan pesanan, hasil perkalian periode, hasil

kuadrat periode, *button* periode selanjutnya adalah untuk melanjutkan inputan periode selanjutnya. Adapun gambar *tab page* perhitungan *history* permintaan pesanan dapat dilihat pada gambar 4.9.

The screenshot displays the 'FORM RENCANA PRODUKSI' interface for PT Elang Jagad. It features a blue header with the company logo and name. Below the header, there are three tabs: 'TabPage1', 'TabPage2', and 'TabPage3'. The main content area is divided into several sections:

- Input Fields:**
 - Kode Rencana Produksi: RP-001
 - Periode: P-012
 - Permintaan Pesanan: 12
 - Permintaan Pesanan: 7.500 Urit
- Buttons:**
 - Hitung
 - Simpan PerPeriode
 - Periode Selanjutnya
 - Simpan Data
 - Data Selanjutnya
- Results:**
 - Hasil Perkalian Periode: 90.000
 - Hasil Kuadrat Periode: 144
 - Akumulasi periode: 66
 - Akumulasi permintaan pesanan: 82.500
 - Akumulasi perkalian periode: 495.000
 - Akumulasi kuadrat periode: 506
- Data Table:**

Kode_Rencana_Pr	Kode_Periode_Pro	Permintaan_Pesan	Hasil_Perkalian_Pe	Hasil_Kl
RP-001	P-001	7500	7500	1
RP-001	P-002	7500	15000	4
RP-001	P-003	7500	22500	9
RP-001	P-004	7500	30000	16
RP-001	P-005	7500	37500	25
RP-001	P-006	7500	45000	36
RP-001	P-007	7500	52500	49

Gambar 4.9 Tampilan Rencana Produksi *Tab Page 1* Perhitungan *History* Permintaan Pesanan

Di *form tab page* perhitungan mencari koefisien regresi dan harga konstan ini terdapat 14 *textbox* yang semuanya otomatis terisi sendiri dan terdapat 2 *button* yaitu: *button* hitung untuk menghitung koefisien regresi, *button* hitung untuk menghitung harga konstan. Adapun gambar *tab page* perhitungan koefisien regresi dan harga konstan dapat dilihat pada gambar 4.10.

Gambar 4.10 Tampilan Rencana Produksi *Tab Page 2* Perhitungan Koefisien Regresi dan Harga Konstan

Di form *tab page* perhitungan peramalan produksi ini terdapat: 9 *textbox* yaitu kode rencana produksi yang otomatis akan terisi kodenya, kode periode yang otomatis terisi kode periodenya, periode yang otomatis terisi periodenya disesuaikan dengan kode periode, harga jual, harga konstan yang otomatis terisi sendiri yang didapatkan dari perhitungan harga konstan, koefisien regresi yang otomatis terisi sendiri yang didapatkan dari perhitungan koefisien regresi, hasil peramalan perperiode, rata-rata peramalan yang otomatis terisi sendiri yang didapatkan dari rata-rata hasil peramalan perperiode, hasil peramalan dan terdapat 5 *button* yaitu: *button* hitung untuk menghitung koefisien regresi dikali dengan periode ditambah dengan harga konstan, *button* simpan perperiode untuk menyimpan perperiodenya dari kode rencana produksi, kode periode, hasil peramalan periode, *button* periode selanjutnya

adalah untuk melanjutkan inputan periode selanjutnya, *button* simpan data untuk menyimpan akumulasi periode, akumulasi permintaan pesanan, akumulasi perkalian periode, akumulasi kuadrat periode, harga konstan, koefisien regresi serta harga jual, rata-rata peramalan serta hasil peramalan. Adapun gambar *tab page* perhitungan peramalan produksi dapat dilihat pada gambar 4.11.

The screenshot displays the 'FORM RENCANA PRODUKSI' interface for PT Elang Jagad, specifically the 'Perhitungan Peramalan Produksi' tab. The form includes several input fields and buttons for calculating and saving production forecast data.

Input Fields:

- Kode Rencana Produksi: RP-002
- Harga Jual: Rp. 10.000
- Harga Konstan: 7500
- Koefisien Regresi: 0
- Periode: P-025
- Hasil: [Empty field]

Buttons: Hitung, Simpan Perperiode, Periode Selanjutnya, Simpan Data, Data Selanjutnya.

Table 1: Production Forecast Data

Kode_Rencana_Pr	Harga_Jual	Hasil_Peramalan	Rata_Rata_Hasil_F	Akumulasi
RP-001	10000	7500	7500	78

Table 2: Detailed Production Forecast Data

Kode_Rencana_Pr	Kode_Periode_Proc	Hasil_Peramalan_P
RP-001	P-013	7500
RP-001	P-014	7500
RP-001	P-015	7500
RP-001	P-016	7500
RP-001	P-017	7500

Summary Fields:

- Rata-Rata Peramalan: 7500 Unit
- Hasil Peramalan: 7500 Unit

Gambar 4.11 Tampilan Rencana Produksi *Tab Page 3* Perhitungan Peramalan Produksi

4.3.5 Tampilan Perhitungan Biaya Produksi

Tampilan perhitungan biaya produksi akan muncul ketika pengguna memilih *icon* biaya produksi pada menu utama dan sebelumnya sudah melakukan *login*. Tampilan perhitungan pendapatan ini terdapat 36 *textbox*, 8 *combobox* serta 2 *button* yaitu: *button* simpan data untuk menyimpan data biaya produksi baru, *button* data selanjutnya untuk mengkosongkan *textbox* yang sudah disimpan. Adapun tampilan perhitungan biaya produksi dapat dilihat pada gambar 4.12.

The screenshot shows a web-based form titled 'Form Biaya Produksi' for PT Elang Jagad. The form is organized into three columns:

- Column 1 (PENDAPATAN & BIAYA VARIABEL):** Contains fields for 'Volume Produksi' (with a dropdown 'RP-001'), 'Harga Jual', 'Total Pendapatan', 'Bahan Baku' (with a dropdown 'Plat Logam'), 'Biaya Bahan Baku' (Rp. 3,000), 'Overhead Variabel' (with a dropdown 'Enamel'), 'Biaya Overhead' (Rp. 125), and various 'Total' and 'TOTAL BIAYA VARIABEL' fields.
- Column 2 (BIAYA TETAP):** Contains fields for 'Gaji Tenaga Kerja Langsung' (Operator, Rp. 50,000), 'Gaji Tenaga Tidak Langsung' (Operator, Rp. 50,000), and 'Total Gaji TKTL'.
- Column 3 (Listrik & Overhead):** Contains fields for 'Listrik' (Ginding Surface, Daya 3.5, Tarif Dasar Listrik, Waktu, Biaya Listrik), 'Overhead Tetap' (Enamel), 'Biaya Overhead' (Rp. 125), and 'Penyusutan' (Ginding Surface, Rp. 3,250,000).

At the bottom, there are summary fields for 'TOTAL BIAYA TETAP' and 'TOTAL BIAYA PRODUKSI'. Two buttons are visible: 'Simpan Data' and 'Data Selanjutnya'.

Gambar 4.12 Tampilan Perhitungan Biaya Produksi

4.3.6 Tampilan Perhitungan Laba Rugi

Tampilan perhitungan laba rugi akan muncul ketika pengguna memilih *icon* laba rugi pada menu utama dan sebelumnya sudah melakukan *login*. Tampilan perhitungan laba rugi ini terdapat: 3 *combobox* kode mesin yang digunakan untuk memilih total investasi berdasarkan kode mesin, kode biaya produksi yang digunakan untuk memilih hasil pendapatan berdasarkan kode biaya produksi dan periode serta kode biaya produksi yang digunakan untuk memilih biaya variabel dan biaya tetap. Ada 20 *textbox* yaitu kode laba rugi yang secara otomatis terisi, kode periode dan periode investasi yang otomatis terisi sendiri, pendapatan, biaya variabel, biaya tetap, total biaya, laba sebelum pajak, pajak, serta laba setelah pajak, akumulasi pendapatan, akumulasi biaya variabel, akumulasi biaya tetap, akumulasi total biaya, akumulasi laba sebelum pajak, akumulasi pajak, akumulasi laba setelah pajak serta rata-rata laba setelah pajak. Terdapat 4 *button* yaitu: *button* simpan perperiode untuk menyimpan data pendapatan, biaya variabel, biaya tetap, total biaya, laba sebelum pajak, pajak

serta laba setelah pajak perperiode. *Button* periode selanjutnya untuk melanjutkan ke inputan periode selanjutnya. *Button* simpan data untuk menyimpan data akumulasi pendapatan, akumulasi biaya variabel, akumulasi biaya tetap, akumulasi total biaya, akumulasi laba sebelum pajak, akumulasi pajak, akumulasi laba setelah pajak serta untuk menyimpan rata-rata laba setelah pajak. *Button* data selanjutnya untuk melanjutkan inputan baru. Adapun tampilan perhitungan laba rugi dapat dilihat pada gambar 4.13.

PT Elang Jagad

FORM LABA RUGI

Kode Laba Rugi: LB-002

Periode Laba Rugi: P-001

Investasi: Rp. 25,000,000

Pendapatan: Rp. 75,000,000

Biaya Variabel: Rp. 24,562,500

Biaya Tetap: Rp. 31,874,125

Total Biaya: Rp. 56,436,625

Laba Sebelum Pajak: Rp. 18,563,375

Pajak: 30%

Laba Setelah Pajak: Rp. 12,994,363

Kode_Laba_Rugi	Periode_Investasi	Total_Investasi	Pendapatan	Total_Biaya_Variab	Total_Biaya_Tetap	Total_Biaya	Laba_Sebelum_Paj	Pajak
LB-001	1	20000000	75000000	24562500	31874125	56436625	18563375	5569012
LB-001	2	20000000	100000000	32750000	35569957	68319957	31680043	9504013
LB-001	3	20000000	130000000	42575000	35745377	78320377	51679623	15503887
LB-001	4	20000000	150000000	49125000	35863733	84988733	65011267	19503380

Laba Kumulatif Pendapatan: Rp. _____

Laba Kumulatif Biaya Variabel: Rp. _____

Laba Kumulatif Biaya Tetap: Rp. _____

Laba Kumulatif Total Biaya: Rp. _____

Laba Kumulatif Laba Sebelum Pajak: Rp. _____

Laba Kumulatif Pajak: Rp. _____

Laba Kumulatif Laba Setelah Pajak: Rp. _____

Rata-Rata Laba Setelah Pajak: Rp. _____

Simpan Perperiode Periode Selanjutnya

Simpan Data Data Selanjutnya

Gambar 4.13 Tampilan Perhitungan Laba Rugi

4.3.7 Tampilan Perhitungan Aliran Kas Bersih

Tampilan perhitungan aliran kas bersih akan muncul ketika pengguna memilih *icon* aliran kas bersih pada menu utama dan sebelumnya sudah melakukan *login*. Tampilan perhitungan aliran kas bersih ini terdapat: 2 *combobox* kode laba rugi yang digunakan untuk memilih laba setelah pajak berdasarkan kode laba rugi dan periodenya, kode mesin yang digunakan untuk memilih hasil penyusutan berdasarkan kode mesin. Ada 7 *textbox* : kode aliran kas bersih yang secara otomatis terisi, kode periode dan periode investasi yang otomatis terisi sendiri, laba setelah pajak otomatis terisi sendiri jika memilih *combobox* kode laba rugi, penyusutan otomatis jika memilih *combobox* kode mesin, hasil aliran kas bersih perperiode otomatis didapatkan dari penjumlahan laba setelah pajak perperiode ditambah dengan penyusutan, akumulasi aliran kas bersih, total aliran kas bersih. Terdapat 5 *button* yaitu: *button* simpan perperiode untuk menyimpan kode laba rugi, kode mesin, periode, serta hasil aliran kas bersih perperiode. *Button* periode selanjutnya untuk melanjutkan ke inputan periode selanjutnya. *Button* simpan data untuk menyimpan data kode aliran kas bersih serat akumulasi aliran kas bersih dan total aliran kas bersih. Adapun tampilan perhitungan aliran kas bersih dapat dilihat pada gambar 4.14.

PT Elang Jagad

FORM ALIRAN KAS BERSIH

Kode Aliran Kas Bersih: KAS-002

Periode: P-001

Laba Setelah Pajak: LB-001

Penyusutan: KM-002

Hasil: Rp. 15,244,363

Kode_Aliran_Kas_E	Periode_Investasi	Laba_Setelah_Pajak	Penyusutan	Hasil
KAS-001	1	12994363	2250000	15244363
KAS-001	2	22176030	2250000	24426030
KAS-001	3	36175736	2250000	38426030
KAS-001	4	45507887	2250000	47776030

Akumulasi Aliran Kas Bersih: Rp. []

Total Aliran Kas Bersih: Rp. []

Simpan Perperiode, Periode Selanjutnya, Simpan Data, Data Selanjutnya

Gambar 4.14 Tampilan Perhitungan Aliran Kas Bersih

4.3.8 Tampilan Perhitungan *Average Rate of Return (ARR)*

Tampilan perhitungan *average rate of return* akan muncul ketika pengguna memilih *icon* hitung *ARR* pada menu utama dan sebelumnya sudah melakukan *login*. Tampilan perhitungan *average rate of return* ini terdapat: 2 *combobox* kode laba rugi yang digunakan untuk memilih rata-rata laba setelah pajak berdasarkan kode laba rugi, kode mesin yang digunakan untuk memilih total investasi berdasarkan kode mesin. Ada 4 *textbox* : kode *ARR* yang secara otomatis terisi, laba setelah pajak otomatis terisi sendiri jika memilih *combobox* kode laba rugi, total investasi otomatis terisi jika memilih *combobox* kode mesin, hasil *ARR* yang didapatkan dari pembagaian antara rata-rata laba setelah pajak dengan total investasi. Terdapat 2 *button* yaitu: *button* hitung untuk menghitung pembagian antara rata-rata laba setelah pajak dengan total investasi, *button* simpan untuk menyimpan data *ARR* baru.

Adapun tampilan perhitungan *average rate of return (ARR)* dapat dilihat pada gambar 4.15.

Kode_ARR	rata_rata_Laba_Sei	Total_Investasi	Hasil_ARR
ARR-001	29213504	20000000	146.07
*			

Gambar 4.15 Tampilan Perhitungan *Average Rate of Return (ARR)*

4.3.9 Tampilan Perhitungan *Payback Period (PP)*

Tampilan perhitungan *payback period* akan muncul ketika pengguna memilih *icon* hitung *PP* pada menu utama dan sebelumnya sudah melakukan *login*. Tampilan perhitungan *payback period* ini terdapat: 5 *combobox* kode mesin yang digunakan untuk memilih total investasi berdasarkan kode mesin, kode aliran kas bersih periode n yang digunakan untuk memilih hasil aliran kas bersih perperiode berdasarkan periode dan kode aliran kas bersih, kode aliran kas bersih periode $n+1$ yang digunakan untuk memilih hasil aliran kas bersih perperiode berdasarkan periode dan kode aliran kas bersih. Ada 7 *textbox* : kode *PP* yang secara otomatis terisi, aliran kas bersih perperiode periode n otomatis terisi sendiri jika memilih *combobox* kode aliran kas bersih periode n , aliran kas bersih perperiode periode $n+1$ otomatis terisi sendiri

jika memilih *combobox* kode aliran kas bersih periode n+1, total investasi otomatis terisi jika memilih *combobox* kode mesin serta hasil *PP*. Terdapat 3 *button* yaitu: *button* hitung untuk menghitung *payback period (PP)*, *button* simpan data untuk menyimpan data *PP* baru, *button* data selanjutnya untuk melanjutkan ke inputan data baru selanjutnya. Adapun tampilan perhitungan *payback period (PP)* dapat dilihat pada gambar 4.16.

The screenshot shows the 'Form Payback Period' interface. At the top, it displays 'PT Elang Jagad' and 'Form Payback Period'. The form contains several dropdown menus: 'Kode PP' (PP-002), 'Investasi' (KM-002), 'Periode n' (P-001), 'Periode n + 1' (P-002), and 'Aliran Kas Bersih' (KAS-001). Below these, there is a calculation area with the following values: 20,000,000 - 15,244,363 = 4,755,637, and 4,755,637 / 39,670,389 = 0.119, resulting in 1.19 Tahun. There are two data tables. The first table has columns: Kode_Aliran_Kas_E, Periode_Investasi, Laba_Setelah_Pajak, and Penyusutan. The second table has columns: Kode_PP, Kode_Periode_Inve, Kode_Mesin, and Kode_Aliran_Kas_E. At the bottom, there are buttons for 'Simpan Data' and 'Data Selanjutnya'.

Gambar 4.16 Tampilan Perhitungan *Payback Period (PP)*

4.3.10 Tampilan Perhitungan *Net Present Value (NPV)*

Tampilan perhitungan *net present value* akan muncul ketika pengguna memilih *icon* hitung *NPV* pada menu utama dan sebelumnya sudah melakukan *login*. Tampilan perhitungan *net present value* ini terdapat: 2 *combobox* kode mesin yang digunakan untuk memilih total investasi berdasarkan kode mesin, kode aliran kas bersih yang digunakan untuk memilih hasil aliran kas bersih perperiode berdasarkan periode dan kode aliran kas bersih. Ada 9 *textbox* : kode *NPV* yang secara otomatis terisi, kode periode dan periode yang secara otomatis terisi, aliran kas bersih

perperiode otomatis terisi sendiri jika memilih *combobox* kode aliran kas bersih periode, *discount factor*, hasil *PV* kas masuk perperiode, akumulasi *PV* kas masuk yang otomatis terisi sendiri berdasarkan jumlah hasil *NPV*, total investasi otomatis terisi jika memilih *combobox* kode mesin, hasil *NPV*. Terdapat 6 *button* yaitu: *button* hitung untuk menghitung *PV* kas masuk yang didapatkan dari perkalian antara aliran kas bersih perperiode dengan *discount factor*, *button* hitung untuk menghitung hasil *NPV* yang didapatkan dari pengurangan antara akumulasi *PV* kas masuk dengan total investasi, *button* simpan perperiode untuk menyimpan kode *NPV*, periode, kode aliran kas bersih, *discount factor*, serta hasil *PV* kas masuk perperiode, *button* periode selanjutnya untuk melanjutkan ke inputan periode selanjutnya, *button* simpan data untuk menyimpan data kode *NPV* serta akumulasi *PV* kas masuk dan total investasi serta hasil *NPV*. Adapun tampilan perhitungan *Net Present Value (NPV)* dapat dilihat pada gambar 4.17.

PT Elang Jagad Form Net Present Value

Kode NPV: NPV-002
 Periode NPV: P-001
 Aliran Kas Bersih: KAS-001
 Rp. 15,244.363
 Discount Factor: 0.847
 PV Kas Masuk: Rp. 12,311.975
 Hitung
 Simpan Perperiode Periode Selanjutnya

Kode_NPV	akumulasi_PV_Kas	Total_Investasi	Hasil_NPV
NPV-001	78494208	20000000	58494208

Kode_NPV	Periode_Investasi	Hasil_Aliran_Kas_B	Discount_Factor	PV_Kas_Masuk
NPV-001	1	15244363	0.847	12911975
NPV-001	2	24426030	0.718	17537890
NPV-001	3	38425736	0.609	23401273
NPV-001	4	47757887	0.516	24643070

Akumulasi PV Kas Masuk:
 Investasi: KM-001
 Rp.
 Hasil NPV: Rp. Hitung
 Simpan Data Data Selanjutnya

Gambar 4.17 Tampilan Perhitungan *Net Present Value (NPV)*

4.3.11 Tampilan Perhitungan *Internal Rate of Return (IRR)*

Tampilan perhitungan *internal rate of return* akan muncul ketika pengguna memilih *icon* hitung *IRR* pada menu utama dan sebelumnya sudah melakukan *login*. Tampilan perhitungan *internal rate of return* ini terdapat: 3 *combobox* kode mesin yang digunakan untuk memilih total investasi berdasarkan kode mesin, kode aliran kas bersih yang digunakan untuk memilih hasil aliran kas bersih perperiode berdasarkan periode dan kode aliran kas bersih, kode *NPV* yang digunakan untuk memilih hasil *NPV* berdasarkan kode *NPV*. Ada 15 *textbox* : kode *IRR* yang secara otomatis terisi, kode periode dan periode yang secara otomatis terisi, aliran kas bersih perperiode otomatis terisi sendiri jika memilih *combobox* kode aliran kas bersih periode, *discount factor*, hasil *PV* kas masuk perperiode, akumulasi *PV* kas masuk yang otomatis terisi sendiri berdasarkan jumlah hasil *NPV*, total investasi otomatis terisi jika memilih *combobox* kode mesin, hasil *NPV*, hasil *IRR*. Terdapat 7 *button* yaitu: *button* hitung *PV* kas masuk untuk menghitung *PV* kas masuk yang didapatkan dari perkalian antara aliran kas bersih perperiode dengan *discount factor*, *button* hitung *NPV* untuk menghitung hasil *NPV* berlawanan yang didapatkan dari pengurangan antara akumulasi *PV* kas masuk dengan total investasi, *button* simpan perperiode untuk menyimpan kode *NPV*, periode, kode aliran kas bersih, *discount factor*, serta hasil *PV* kas masuk perperiode, *button* periode selanjutnya untuk melanjutkan ke inputan periode selanjutnya, *button* hitung *IRR* untuk menghitung hasil *IRR* yang hasilnya berupa persentase, *button* simpan data untuk menyimpan data kode *IRR* serta akumulasi *PV* kas masuk dan total investasi serta hasil *NPV*

berlawanan dan hasil *IRR*. Adapun tampilan perhitungan *Internal Rate of Return* (*IRR*) dapat dilihat pada gambar 4.18.

PT Elang Jagad
Form Internal Rate Of Return

NPV Berlawanan

Kode IRR: IRR-002
 Periode IRR: P-001
 Aliran Kas Bersih: KAS-001
 Rp: 15.244.363
 Discount Factor: 0.735
 PV Kas Masuk: Rp: 11.204.607

Hitung

Simpan Perperiode | Periode Selanjutnya

Kode_IRR	Hasil_NPV	Total_Investasi	Akumulasi_PV_Kas	Hasil_NPV_Berlaw	Hasil_IRR
IRR-001	58494208	20000000	53657835	33657835	60.39

Kode_IRR	Periode_Investasi	Hasil_Aliran_Kas_B	Discount_Factor_B	PV_Kas_Masuk_B
IRR-001	1	15244363	0.735	11204607
IRR-001	2	24426030	0.541	13214482
IRR-001	3	38425736	0.398	15293443
IRR-001	4	47757887	0.292	13945303

Akumulasi PV Kas Masuk: Rp: NPV: NPV-001

Investasi: KM-001
 Rp:

Hasil NPV Berlawanan: Rp: Hitung

Hitung IRR

Discount Factor NPV: + × - = % Hitung

Simpan Data | Data Selanjutnya

Gambar 4.18 Tampilan Perhitungan *Internal Rate of Return* (*IRR*)

4.3.12 Tampilan Perhitungan *Profitability Index* (*PI*)

Tampilan perhitungan *profitability index* akan muncul ketika pengguna memilih *icon* hitung *PI* pada menu utama dan sebelumnya sudah melakukan *login*. Tampilan perhitungan *profitability index* ini terdapat: 2 *combobox* kode *NPV* yang digunakan untuk memilih akumulasi *PV* kas masuk berdasarkan kode *NPV*, kode mesin yang digunakan untuk memilih total investasi berdasarkan kode mesin. Ada 4 *textbox* : kode *PI* yang secara otomatis terisi, akumulasi *PV* kas masuk otomatis terisi sendiri jika memilih *combobox* kode *NPV*, total investasi otomatis terisi sendiri jika memilih *combobox* kode mesin, hasil *PI*. Terdapat 3 *button* yaitu: *button* hitung *PI* untuk menghitung *PI* yang didapatkan dari pembagian antara akumulasi *PV* kas

masuk dengan total investasi, *button* simpan data untuk menyimpan kode *PI*, kode *NPV*, kode mesin, serta hasil *PI*, *button* data selanjutnya untuk melanjutkan ke inputan *PI* baru. Adapun tampilan perhitungan *Profitability Index (PI)* dapat dilihat pada gambar 4.19.

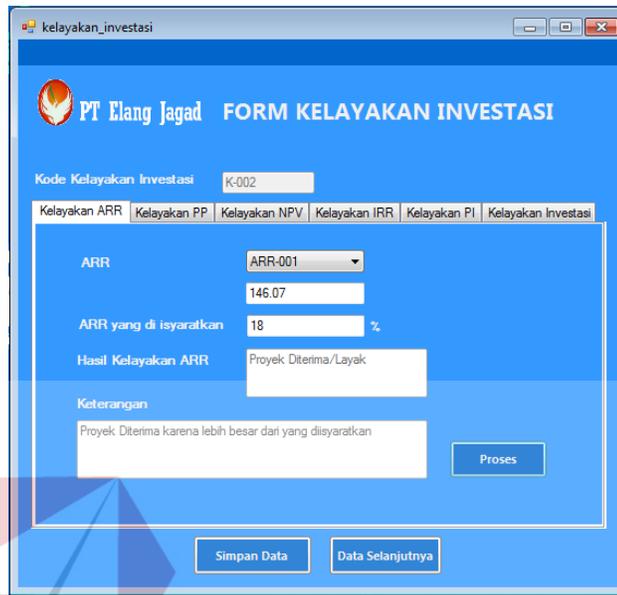
Kode_PI	Akumulasi_PV_Kas	Total_Investasi	Hasil_PI
PI-001	78494208	20000000	3.92
*			

Gambar 4.19 Tampilan *Profitability Index (PI)*

4.3.13 Tampilan Kelayakan Investasi

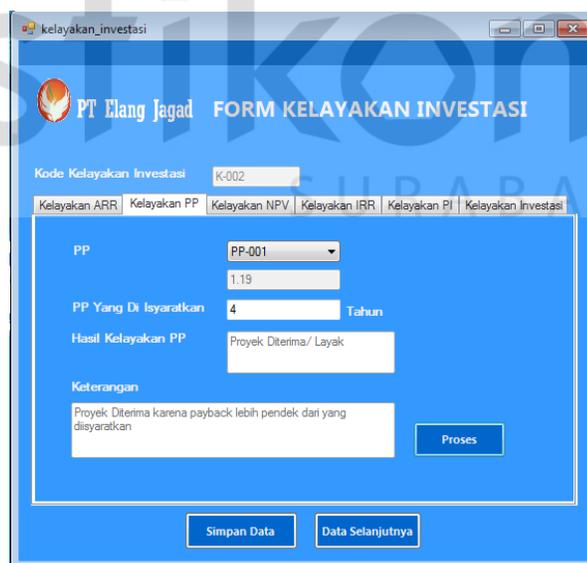
Tampilan kelayakan investasi akan muncul ketika pengguna memilih *icon* kelayakan investasi pada menu utama dan sebelumnya sudah melakukan *login*. Tampilan kelayakan investasi ini merupakan perbandingan antara hasil perhitungan dengan yang diisyaratkan atau ditargetkan yang hasilnya layak atau tidak layak. Tampilan kelayakan investasi dibagi menjadi 6 *tab page* : *tab page* kelayakan *ARR*, *tab page* kelayakan *PP*, *tab page* kelayakan *NPV*, *tab page* kelayakan *IRR*, *tab page* kelayakan *PI*. *Button* proses akan membandingkan hasil perhitungan dengan yang diisyaratkan atau ditargetkan. *Button* simpan data akan menyimpan semua data

kelayakan investasi. Adapun tampilan kelayakan investasi dapat dilihat pada gambar 4.20, gambar 4.21, gambar 4.22, gambar 4.23, gambar 4.24 dan gambar 4.25.



The screenshot shows a web application window titled 'kelayakan_investasi'. The header includes the PT Elang Jagad logo and the title 'FORM KELAYAKAN INVESTASI'. Below the header, there is a text input field for 'Kode Kelayakan Investasi' with the value 'K-002'. A navigation bar contains several tabs: 'Kelayakan ARR', 'Kelayakan PP', 'Kelayakan NPV', 'Kelayakan IRR', 'Kelayakan PI', and 'Kelayakan Investasi'. The 'Kelayakan ARR' tab is active. The form fields include: 'ARR' (dropdown menu with 'ARR-001'), a numerical input field with '146.07', 'ARR yang di isyaratkan' (input field with '18' and a '%' symbol), and 'Hasil Kelayakan ARR' (text box with 'Proyek Diterima/Layak'). A 'Keterangan' section contains the text 'Proyek Diterima karena lebih besar dari yang diisyaratkan'. At the bottom, there are buttons for 'Proses', 'Simpan Data', and 'Data Selanjutnya'.

Gambar 4.20 Tampilan Kelayakan Investasi *Tab Page* Kelayakan ARR



The screenshot shows the same web application window as in Gambar 4.20, but with the 'Kelayakan PP' tab selected. The 'Kode Kelayakan Investasi' remains 'K-002'. The 'Kelayakan PP' tab is active. The form fields include: 'PP' (dropdown menu with 'PP-001'), a numerical input field with '1.19', 'PP Yang Di Isyaratkan' (input field with '4' and the word 'Tahun'), and 'Hasil Kelayakan PP' (text box with 'Proyek Diterima/ Layak'). The 'Keterangan' section contains the text 'Proyek Diterima karena payback lebih pendek dari yang diisyaratkan'. At the bottom, there are buttons for 'Proses', 'Simpan Data', and 'Data Selanjutnya'.

Gambar 4.21 Tampilan Kelayakan Investasi *Tab Page* Kelayakan PP

Kelayakan Investasi

Kode Kelayakan Investasi K-002

Kelayakan ARR Kelayakan PP Kelayakan NPV Kelayakan IRR Kelayakan PI Kelayakan Investasi

NPV NPV-001

58.494.208

Hasil Kelayakan NPV Proyek Diterima/Layak

Keterangan
Proyek Diterima karena bersifat positif

Proses

Simpan Data Data Selanjutnya

Gambar 4.22 Tampilan Kelayakan Investasi *Tab Page* Kelayakan NPV

Kelayakan Investasi

Kode Kelayakan Investasi K-002

Kelayakan ARR Kelayakan PP Kelayakan NPV Kelayakan IRR Kelayakan PI Kelayakan Investasi

IRR IRR-001

60.39

IRR Yang Di Isyaratkan 18 %

Hasil Kelayakan IRR Proyek Diterima/Layak

Keterangan
Proyek Diterima karena lebih dari suku bunga

Proses

Simpan Data Data Selanjutnya

Gambar 4.23 Tampilan Kelayakan Investasi *Tab Page* Kelayakan IRR

Kode Kelayakan Investasi: K-002

Kelayakan ARR | Kelayakan PP | Kelayakan NPV | Kelayakan IRR | Kelayakan PI | Kelayakan Investasi

PI: PI-001

Hasil Kelayakan PI: 3.92

Hasil Kelayakan PI: Proyek Diterima/Layak

Keterangan: Proyek Diterima karena bersifat lebih dari satu

Proses

Simpan Data | Data Selanjutnya

Gambar 4.24 Tampilan Kelayakan Investasi *Tab Page* Kelayakan PI

Kode Kelayakan Investasi: K-002

Kelayakan ARR | Kelayakan PP | Kelayakan NPV | Kelayakan IRR | Kelayakan PI | Kelayakan Investasi

Kode_Kelayakan	Hasil_ARR	Hasil_PP	Hasil_NPV	Hasil
K-001	146.07	1.19	58494208	60.39

Kesimpulan Kelayakan Investasi Ini Adalah Layak

Simpan Data | Data Selanjutnya

Gambar 4.25 Tampilan Kelayakan Investasi *Tab Page* Hasil Kelayakan

4.3.14 Tampilan Laporan Laba Rugi

Tampilan laporan laba rugi ini merupakan hasil dari perhitungan laba rugi. Laporan laba rugi ini digunakan untuk mengetahui proyeksi laba rugi perusahaan setelah melakukan investasi mesin. Untuk membuat laporan ini tinggal klik *icon* laporan laba rugi di menu utama. Di laporan ini akan menampilkan beberapa laba rugi, untuk memilih laba rugi per kode tinggal ketik kode laba ruginya kemudian klik *button* tampilkan di *form* tampilan laporan laba rugi. Adapun tampilan *form* laporan laba rugi dapat dilihat pada gambar 4.26 dan gambar 4.27 laporan laba rugi setelah dicetak.

Periode	Pendapatan	Biaya Variabel	Biaya Tetap	Penyusutan	Total Biaya	Laba Sebelum Pajak	Pajak	Laba Setelah Pajak
1	75.000.000	24.562.500	31.874.125	2.250.000	56.436.625	18.563.375	5.589.012	12.974.363
2	100.000.000	32.750.000	35.569.957	2.250.000	68.319.957	31.680.043	9.504.013	22.176.030
3	130.000.000	42.575.000	36.745.377	2.250.000	78.320.377	51.679.623	15.503.887	36.175.736
4	150.000.000	49.125.000	35.863.733	2.250.000	84.988.733	65.011.267	19.503.380	45.507.887
Total Laba Setelah Pajak								116.854.016.00

Gambar 4.26 Tampilan *Form* Laporan Laba Rugi

Kode Laba Rugi LB-001		Laporan Laba Rugi							2/2/2016
Periode	Pendapatan	Biaya Variabel	Biaya Tetap	Penyusutan	Total Biaya	Laba Sebelum Pajak	Pajak	Laba Setelah Pajak	
1	75.000.000	24.562.500	31.874.125	2.250.000	56.436.625	18.563.375	5.569.012	12.994.363	
2	100.000.000	32.750.000	35.569.957	2.250.000	68.319.957	31.680.043	9.504.013	22.176.030	
3	130.000.000	42.575.000	35.745.377	2.250.000	78.320.377	51.679.623	15.503.887	36.175.736	
4	150.000.000	49.125.000	35.863.733	2.250.000	84.988.733	65.011.267	19.503.380	45.507.887	
Total Laba Setelah Pajak								116,854,016.00	

Gambar 4.27 Laporan Laba Rugi

4.3.15 Tampilan Laporan Aliran Kas Bersih

Tampilan laporan aliran kas bersih ini merupakan hasil dari perhitungan aliran kas bersih. Laporan aliran kas bersih ini digunakan untuk mengetahui proyeksi penerimaan-penerimaan dan pengeluaran-pengeluaran kas pada saat investasi mesin. Untuk membuat laporan ini tinggal klik *icon* laporan aliran kas bersih di menu utama. Di laporan ini akan menampilkan beberapa aliran kas bersih, untuk memilih aliran kas bersih perkode tinggal ketik kode aliran kas bersih kemudian klik *button* tampilkan di *form* tampilan laporan aliran kas bersih. Adapun tampilan *form* laporan aliran kas bersih dapat dilihat pada gambar 4.28 dan gambar 4.29 laporan aliran kas bersih setelah dicetak.

Form2

KAS-001

Main Report

PT. ELANG JAGAD
KONTRAKTOR INDUSTRI DAN PERDAGANGAN
Jalan Kolonel Soegiono No 56 Ngingas RT. 05 RW. 02
Waru Sidoarjo

Laporan Aliran Kas Bersih 2/2/2016

Kode	Periode	Laba Setelah Pajak	Penyusutan	Aliran Kas Bersih
KAS-001	1	12,994,363	2,250,000	15,244,363
KAS-001	2	22,176,030	2,250,000	24,426,030
KAS-001	3	36,175,736	2,250,000	38,425,736
KAS-001	4	45,507,887	2,250,000	47,757,887
Total Aliran Kas Bersih				125,854,016.00
				125,854,016.00

Current Page No.: 1 Total Page No.: 1 Zoom Factor: 100%

Gambar 4.28 Tampilan *Form* Aliran Kas Bersih

PT. ELANG JAGAD
KONTRAKTOR INDUSTRI DAN PERDAGANGAN
Jalan Kolonel Soegiono No 56 Ngingas RT. 05 RW. 02
Waru Sidoarjo

Laporan Aliran Kas Bersih 2/2/2016

Kode	Periode	Laba Setelah Pajak	Penyusutan	Aliran Kas Bersih
KAS-001	1	12,994,363	2,250,000	15,244,363
KAS-001	2	22,176,030	2,250,000	24,426,030
KAS-001	3	36,175,736	2,250,000	38,425,736
KAS-001	4	45,507,887	2,250,000	47,757,887
Total Aliran Kas Bersih				125,854,016.00
				125,854,016.00

Gambar 4.29 Laporan Aliran Kas Bersih

4.3.16 Tampilan Laporan *Average Rate of Return (ARR)*

Tampilan laporan *average rate of return* ini merupakan hasil dari perhitungan *average rate of return*. Laporan *average rate of return* ini digunakan untuk mengetahui tingkat keuntungan investasi mesin. Untuk membuat laporan ini tinggal klik *icon* laporan *ARR* di menu utama. Di laporan ini akan menampilkan beberapa hasil *average rate of return*, untuk memilih *average rate of return* perkode tinggal ketik kode *ARR* kemudian klik *button* tampilkan di *form* tampilan laporan *average rate of return*. Adapun tampilan *form* laporan *average rate of return* dapat dilihat pada gambar 4.30 dan gambar 4.31 laporan *average rate of return* setelah dicetak.

The screenshot displays the SAP Crystal Reports interface for the 'ARR-001' report. The report header identifies the company as PT. ELANG JAGAD, a contractor and trader, located at Jalan Kolonel Soegiono No 56, Ngings RT.05 RW.02, Waru Sidoarjo. The report title is 'Laporan AverageRateOfReturn' dated 2/2/2016. The main data is presented in a table with three columns: Kode, Periode Investasi, and Laba Setelah Pajak. Below the table, summary statistics are provided for Total Investasi, Hasil ARR, and Akumulasi.

Kode	Periode Investasi	Labas Setelah Pajak
ARR-001	1	12.994.363
ARR-001	2	22.176.030
ARR-001	3	36.175.736
ARR-001	4	45.507.887
		116.854.016,00
Total Investasi		20.000.000
Hasil ARR		146,07 %
Akumulasi		116.854,016.00

Gambar 4.30 Tampilan *Form* Laporan *Average Rate of Return (ARR)*

 PT. ELANG JAGAD KONTRAKTOR INDUSTRI DAN PERDAGANGAN Jalan Kolonel Soegiono No 56 Ngingas RT.05 RW.02 Waru Sidoarjo		
Laporan Average Rate Of Return		
Kode	Periode Investasi	Laba Setelah Pajak
ARR-001	1	12,994,363
ARR-001	2	22,176,030
ARR-001	3	36,175,736
ARR-001	4	45,507,887
		116,854,016.00
Total Investasi		20,000,000
Hasil ARR		146.07 %
Akumulasi		116,854,016.00

Gambar 4.31 Laporan *Average Rate of Return (ARR)*

4.3.17 Tampilan Laporan *Payback Period (PP)*

Tampilan laporan *payback period* ini merupakan hasil dari perhitungan *payback period*. Laporan *payback period* ini digunakan untuk mengukur seberapa cepat investasi mesin itu bisa kembali. Untuk membuat laporan ini tinggal klik *icon* laporan *PP* di menu utama. Di laporan ini akan menampilkan beberapa hasil *payback period*, untuk memilih *payback period* perkode tinggal ketik kode *PP* kemudian klik *button* tampilkan di *form* tampilan laporan *payback period*. Adapun tampilan *form* laporan *payback period* dapat dilihat pada gambar 4.32 dan gambar 4.33 laporan *payback period* setelah dicetak.

CRPaybackPeriod

PP-001

Main Report

Cari Kode Tampilkan

SAP CRYSTAL REPORTS®

PT. ELANG JAGAD
 KONTRAKTOR INDUSTRI DAN PERDAGANGAN
 Jalan Kolonel Soegiono No 56 Ngingas RT.05 RW.02
 Waru Sidoarjo

Laporan Payback Period 2/2/2016

Kode PP	Periode	Laba Setelah Pajak	Penyusutan	Aliran Kas Bersih
PP-001	1	12.994.363	2.250.000	15.244.363
PP-001	2	22.176.030	2.250.000	24.426.030
PP-001	3	36.175.736	2.250.000	38.425.736
PP-001	4	45.507.887	2.250.000	47.757.887
		116.854.016.00		125.854.016.00
			Total_Investasi	20.000.000
			Hasil PP	1.19 Tahun

Gambar 4.32 Tampilan Form Laporan Payback Period (PP)

PT. ELANG JAGAD
 KONTRAKTOR INDUSTRI DAN PERDAGANGAN
 Jalan Kolonel Soegiono No 56 Ngingas RT.05 RW.02
 Waru Sidoarjo

Laporan Payback Period 2/2/2016

Kode PP	Periode	Laba Setelah Pajak	Penyusutan	Aliran Kas Bersih
PP-001	1	12.994.363	2.250.000	15.244.363
PP-001	2	22.176.030	2.250.000	24.426.030
PP-001	3	36.175.736	2.250.000	38.425.736
PP-001	4	45.507.887	2.250.000	47.757.887
		116.854.016.00		125.854.016.00
			Total_Investasi	20.000.000
			Hasil PP	1.19 Tahun

Gambar 4.33 Laporan Payback Period (PP)

4.3.18 Tampilan Laporan *Net Present Value* (NPV)

Tampilan laporan *net present value* ini merupakan hasil dari perhitungan *net present value*. Laporan *net present value* ini digunakan untuk mengetahui selisih penerimaan-penerimaan kas bersih dimasa yang akan datang dengan investasi sekarang. Untuk membuat laporan ini tinggal klik *icon* laporan NPV di menu utama. Di laporan ini akan menampilkan beberapa hasil *net present value*, untuk memilih *net present value* perkode tinggal ketik kode NPV kemudian klik *button* tampilkan di *form* tampilan laporan *net present value*. Adapun tampilan *form* laporan *net present value* dapat dilihat pada gambar 4.34 dan gambar 4.35 laporan *net present value* setelah dicetak.

The screenshot shows a SAP Crystal Reports window titled 'CRNetPresentValuevb'. The report is for 'PT. ELANG JAGAD', a trading and industry contractor, located at 'Jalan Kolonel Soegiono No 56 Ngingas RT. 05 RW 02 Waru Sidoarjo'. The report is dated 2/2/2016 and is titled 'Laporan Net Present Value'. The data is as follows:

Kode	Periode	Aliran Kas Bersih	Discount Factor	PV Kas Masuk
NPV-001	1	15,244,363	0.85	12,911,975
NPV-001	2	24,426,030	0.72	17,537,890
NPV-001	3	38,425,736	0.61	23,401,273
NPV-001	4	47,797,887	0.52	24,843,070
Akumulasi				78,494,208.00
Hasil NPV				58,494,208

The report footer indicates 'Current Page No.:1', 'Total Page No.:1', and 'Zoom Factor:100%'.

Gambar 4.34 Tampilan *Form* Laporan *Net Present Value* (NPV)

 PT. ELANG JAGAD KONTRAKTOR INDUSTRI DAN PERDAGANGAN Jalan Kolonel Soegiono No 56 Ngingas RT. 05 RW. 02 Waru Sidoarjo				
Laporan Net Present Value				
2/2/2016				
Kode	Periode	Aliran Kas Bersih	Discount Factor	PV Kas Masuk
NPV-001	1	15.244.363	0.85	12.911.975
NPV-001	2	24.426.030	0.72	17.537.890
NPV-001	3	38.425.736	0.61	23.401.273
NPV-001	4	47.757.887	0.52	24.643.070
				78.494.208.00
Akumulasi				78.494.208.00
Hasil NPV				58.494.208

Gambar 4.35 Laporan *Net Present Value* (NPV)

4.3.19 Tampilan Laporan *Internal Rate of Return* (IRR)

Tampilan laporan *internal rate of return* ini merupakan hasil dari perhitungan *internal rate of return*. Laporan *internal rate of return* ini digunakan untuk mengetahui tingkat bunga yang menyamakan selisih penerimaan-penerimaan kas bersih dimasa yang akan datang dengan investasi sekarang. Untuk membuat laporan ini tinggal klik *icon* laporan *IRR* di menu utama. Di laporan ini akan menampilkan beberapa hasil *internal rate of return*, untuk memilih *internal rate of return* perkode tinggal ketik kode *IRR* kemudian klik *button* tampilkan di *form* tampilan laporan *internal rate of return*. Adapun tampilan *form* laporan *internal rate of return* dapat dilihat pada gambar 4.36 dan gambar 4.37 laporan *internal rate of return* setelah dicetak.

PT. ELANG JAGAD
KONTRAKTOR INDUSTRI DAN PERDAGANGAN
Jalan Kolonel Soegiono No 56 Ngingas RT.05 RW.02
Waru Sidoarjo

Laporan Internal Rate of return 2/2/2016

Kode	Periode	Aliran Kas Bersih	Discount_Factor	PV Kas Masuk	Discount Factor_Berlawanan	PV Kas Masuk Berlawanan
IRR-001	1	15.244.363	0.85	12.911.975	0.74	11.204.607
IRR-001	2	24.426.030	0.72	17.537.890	0.54	13.214.482
IRR-001	3	38.425.736	0.61	23.401.273	0.40	15.293.443
IRR-001	4	47.757.887	0.52	24.643.070	0.29	13.945.303
				78.494.208,00		53.657.835,00
Akumulasi				78.494.208,00	Akumulasi	53.657.835,00
Hasil_NPV				58.494.208	Hasil NPV Berlawanan	33.657.835
Hasil_IRR				60.39%		

Current Page No.:1 Total Page No.:1 Zoom Factor:100%

Gambar 4.36 Tampilan Form Laporan Internal Rate of Return (IRR)

PT. ELANG JAGAD
KONTRAKTOR INDUSTRI DAN PERDAGANGAN
Jalan Kolonel Soegiono No 56 Ngingas RT.05 RW.02
Waru Sidoarjo

Laporan Internal Rate of return 2/2/2016

Kode	Periode	Aliran Kas Bersih	Discount_Factor	PV Kas Masuk	Discount Factor_Berlawanan	PV Kas Masuk Berlawanan
IRR-001	1	15.244.363	0.85	12.911.975	0.74	11.204.607
IRR-001	2	24.426.030	0.72	17.537.890	0.54	13.214.482
IRR-001	3	38.425.736	0.61	23.401.273	0.40	15.293.443
IRR-001	4	47.757.887	0.52	24.643.070	0.29	13.945.303
				78.494.208,00		53.657.835,00
Akumulasi				78.494.208,00	Akumulasi	53.657.835,00
Hasil_NPV				58.494.208	Hasil NPV Berlawanan	33.657.835
Hasil_IRR				60.39%		

Gambar 4.37 Laporan Internal Rate of Return (IRR)

4.3.20 Tampilan Laporan *Profitability Index* (PI)

Tampilan laporan *profitability index* ini merupakan hasil dari perhitungan *profitability index*. Laporan *profitability index* ini digunakan untuk mengetahui perbandingan penerimaan-penerimaan kas bersih dimasa yang akan datang dengan investasi sekarang. Untuk membuat laporan ini tinggal klik *icon* laporan *PI* di menu utama. Di laporan ini akan menampilkan beberapa hasil *profitability index*, untuk memilih *profitability index* perkode tinggal ketik kode *PI* kemudian klik *button* tampilkan di *form* tampilan laporan *profitability index*. Adapun tampilan *form* laporan *profitability index* dapat dilihat pada gambar 4.38 dan gambar 4.39 laporan *profitability index* setelah dicetak.

The screenshot shows a SAP Crystal Reports window titled 'CRProfitabilityIndex'. The report is for 'PT. ELANG JAGAD', a contractor and trader, located at 'Jalan Kolonel Soegiono No 56 Ngings RT. 05 RW. 02 Waru Sidoarjo'. The report is dated 2/2/2016. It displays a table of 'PV Kas Masuk' (Present Value of Cash Inflows) for four periods. The table is as follows:

Kode	Periode	PV Kas Masuk
PI-001	1	12.911.975
PI-001	2	17.537.830
PI-001	3	23.401.273
PI-001	4	24.643.070
		78.494.208.00

Below the table, the following summary statistics are shown:

Total Investasi	20.000.000
Hasil PI	3.92
Akumulasi	78.494.208.00

The report footer indicates 'Current Page No.: 1', 'Total Page No.: 1', and 'Zoom Factor: 100%'.

Gambar 4.38 Tampilan *Form* Laporan *Profitability Index* (PI)

PT. ELANG JAGAD		
 PT. ELANG JAGAD KONTRAKTOR INDUSTRI DAN PERDAGANGAN Jalan Kolonel Soegiono No 56 Ngingas RT.05 RW.02 Waru Sidoarjo		
Laporan Profitability Index		
		2/2/2016
Kode	Periode	PV Kas Masuk
PI-001	1	12.911.975
PI-001	2	17.537.890
PI-001	3	23.401.273
PI-001	4	24.643.070
		78.494.208.00
Total Investasi		20.000.000
Hasil PI		3.92
Akumulasi		78.494.208.00

Gambar 4.39 Laporan *Profitability Index* (PI)

4.3.21 Tampilan Laporan Kelayakan Investasi

Tampilan laporan kelayakan investasi ini merupakan hasil dari perhitungan kelayakan investasi. Laporan kelayakan investasi ini digunakan untuk mengetahui layak atau tidak layak investasi mesin tersebut dengan dilihat dari hasil perhitungan dan target atau syarat yang ditentukan. Untuk membuat laporan ini tinggal klik *icon* laporan kelayakan investasi di menu utama. Di laporan ini akan menampilkan beberapa hasil kelayakan investasi, untuk memilih kelayakan investasi perkode tinggal ketik kode kelayakan investasi kemudian klik *button* tampilkan di *form* tampilan laporan kelayakan investasi. Adapun tampilan *form* laporan kelayakan investasi dapat dilihat pada gambar 4.40 dan gambar 4.41 laporan kelayakan investasi setelah dicetak.

Current Page No.:1 Total Page No.:1 Zoom Factor:100%

Gambar 4.40 Tampilan *Form* Kelayakan Investasi

PT. ELANG JAGAD
KONTRAKTOR INDUSTRI DAN PERDAGANGAN
Jalan Kolonel Soegiono No 56 Ngingas RT.05 RW.02
Waru Sidoarjo

Kelayakan Investasi Mesin 2/2/2016

K-001	Hasil	Target	Hasil Kelayakan	Keterangan
ARR	146.07 %	18%	Proyek Diterima/Layak	Proyek Diterima karena lebih besar dari yang diisyaratkan
PP	1.19 Tahun	4 Tahun	Proyek Diterima/ Layak	Proyek Diterima karena payback lebih pendek dari yang diisyaratkan
NPV	58,494,208	0	Proyek Diterima/Layak	Proyek Diterima karena bersifat positif
IRR	60.39 %	18%	Proyek Diterima/Layak	Proyek Diterima karena lebih dari suku bunga
PI	3.92	1	Proyek Diterima/Layak	Proyek Diterima karena bersifat lebih dari satu

Kesimpulan Kelayakan Investasi Ini Adalah Layak Di investasikan atau dibeli

Gambar 4.41 Laporan Kelayakan Investasi

4.4 Uji Coba

Setelah melakukan kebutuhan sistem, pembuatan program, instalasi program dan uji coba *user interface*, tahap selanjutnya adalah melakukan evaluasi terhadap sistem. Evaluasi dilakukan dengan dua tahap yaitu evaluasi terhadap *form* pendukung dengan evaluasi dasar serta *form-form* yang sesuai tujuan aplikasi dibuat dengan hasil analisis. Tujuan evaluasi ini adalah untuk mengetahui apakah fungsi pada aplikasi yang telah dibuat ini sudah berjalan dengan baik dan sesuai dengan tujuan atau *output* yang diharapkan. Adapun evaluasi uji coba yang dilakukan adalah sebagai berikut:

4.4.1 Uji Coba *Form* Master Mesin

Form master mesin digunakan untuk menyimpan data mesin yang ada dan yang akan diinvestasikan atau dibeli. Di *form* master mesin terdapat perhitungan penyusutan. Perhitungan penyusutan diperoleh dari harga perolehan dikurangi dengan nilai sisa kemudian hasilnya dibagi dengan umur ekonomis mesin. Di *form* ini terdapat :

- *Textbox* hasil penyusutan

Umur Ekonomis	4	Tahun
Harga Perolehan	Rp. 20,000,000	
Daya	1.5	Kwh
Nilai Sisa	Rp. 11,000,000	
Penyusutan	Rp. 2,250,000	

Gambar 4.42 *Textbox* Hasil Penyusutan

Penjelasan: *User* menginputkan data umur ekonomis, harga perolehan, serta nilai sisa dan *user* tidak dapat melakukan *input* pada *textbox* hasil

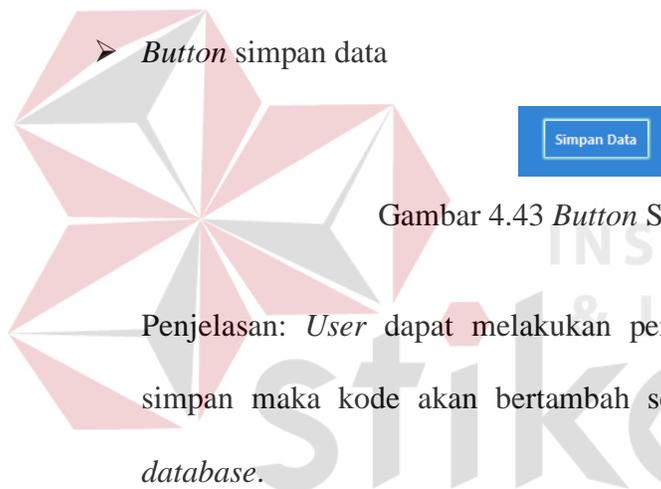
penyusutan karena *disable* atau data yang ditampilkan berasal dari perhitungan:

Penyusutan =

(harga perolehan – nilai sisa) / umur ekonomis(4.1)

Tabel 4.1 Pengujian *Textbox* Hasil Penyusutan

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan



Gambar 4.43 *Button* Simpan Data

Penjelasan: *User* dapat melakukan penyimpanan dengan menekan *button* simpan maka kode akan bertambah serta data akan tersimpan ke dalam *database*.

Tabel 4.2 Pengujian *Button* Simpan Data

No.	Pengujian	Hasil
1.	Melakukan penyimpanan jika ada data yang kosong	Tidak dapat dilakukan

Langkah selanjutnya yaitu melakukan *test* data. *Test* data diperlukan untuk mengetahui apakah fungsi pada aplikasi yang telah dibuat ini sudah berjalan dengan baik dan sesuai dengan tujuan atau *output* yang diharapkan. Sebagai perbandingan, perlu dilakukan uji coba.

Diketahui:

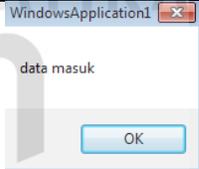
- *Textbox* harga perolehan: Rp 20.000.000
- *Textbox* nilai sisa : Rp 11.000.000
- *Numeric updown* umur ekonomis: 4 tahun
- *Textbox* penyusutan: Rp 2.250.000

Dari contoh data master dan perhitungan penyusutan di atas, lalu tahap selanjutnya yaitu menguji *form* master mesin dan perhitungan penyusutan.

Hasil dari pengujian akan dibandingkan dengan rancangan desain uji coba perhitungan penyusutan yang sebelumnya telah dibuat. Seperti pada tabel 4.3.



Tabel 4.3 Hasil Uji Coba *Form* Master Mesin

<i>Test Case ID</i>	<i>Test Case</i>	<i>Test Data</i>	<i>Output yang diharapkan</i>	<i>Actual Output</i>	<i>Kesimpulan</i>
1	Menampilkan hasil perhitungan penyusutan	Setelah terisi <i>textbox</i> harga perolehan, nilai sisa serta <i>numericupdown</i> umur ekonomis dan klik enter	(Rp 20.000.000 – Rp 11.000.000) / 4 = Rp 2.250.000,00		Sesuai
2	Menyimpan data master mesin dan perhitungan penyusutan	Menekan <i>button</i> simpan data	Tersimpan ke <i>database</i> mesin dan tampil kotak dialog “data masuk”		Sesuai

4.4.2 Uji Coba *Form* Perhitungan Rencana Produksi

Form perhitungan rencana produksi digunakan untuk mengetahui proyeksi produksi setelah melakukan investasi. Di *form* rencana produksi terdapat perhitungan peramalan produksi. Perhitungan peramalan produksi diperoleh dari harga konstan ditambah dengan koefisien regresi dikali dengan subyek variabel. Di *form* ini terdapat:

- *Textbox* hasil peramalan

Harga Konstan	<input type="text" value="7500"/>
Koefisien Regresi	<input type="text" value="0"/>
Periode	<input type="text" value="P-014"/>
	<input type="text" value="14"/>
Hasil	<input type="text" value="7,500"/> Unit

Gambar 4.44 *Textbox* Hasil Peramalan

Penjelasan: *user* tidak dapat melakukan *input* pada *textbox* hasil peramalan karena *disable* atau data yang ditampilkan berasal dari perhitungan:

Peramalan =

$$Y = a + bX \dots \dots \dots (4.2)$$

Dimana : Y = Subyek dalam variabel dependen yang diprediksikan

a = Harga Y bila X=0 (harga konstan)

b = Koefisien regresi

X= Subyek pada variabel independen yang mempunyai nilai tertentu (periode)

Tabel 4.4 Pengujian *Textbox* Hasil Peramalan

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

➤ *Button* simpan dataGambar 4.45 *Button* Simpan Data

Penjelasan: *User* dapat melakukan penyimpanan dengan menekan *button* simpan maka kode akan bertambah serta data akan tersimpan ke dalam *database*.

Tabel 4.5 Pengujian *Button* Simpan Data

No.	Pengujian	Hasil
1.	Melakukan penyimpanan jika ada data yang kosong	Tidak dapat dilakukan

Langkah selanjutnya yaitu melakukan *test* data. *Test* data diperlukan untuk mengetahui apakah fungsi pada aplikasi yang telah dibuat ini sudah berjalan dengan baik dan sesuai dengan tujuan atau *output* yang diharapkan. Sebagai perbandingan, perlu dilakukan uji coba.

Diketahui:

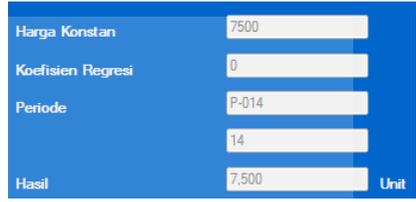
- *Textbox* harga konstan: Rp 7.500
- *Textbox* koefisien regresi : Rp 0
- *Textbox* periode: 14
- *Textbox* hasil peramalan: Rp 7.500

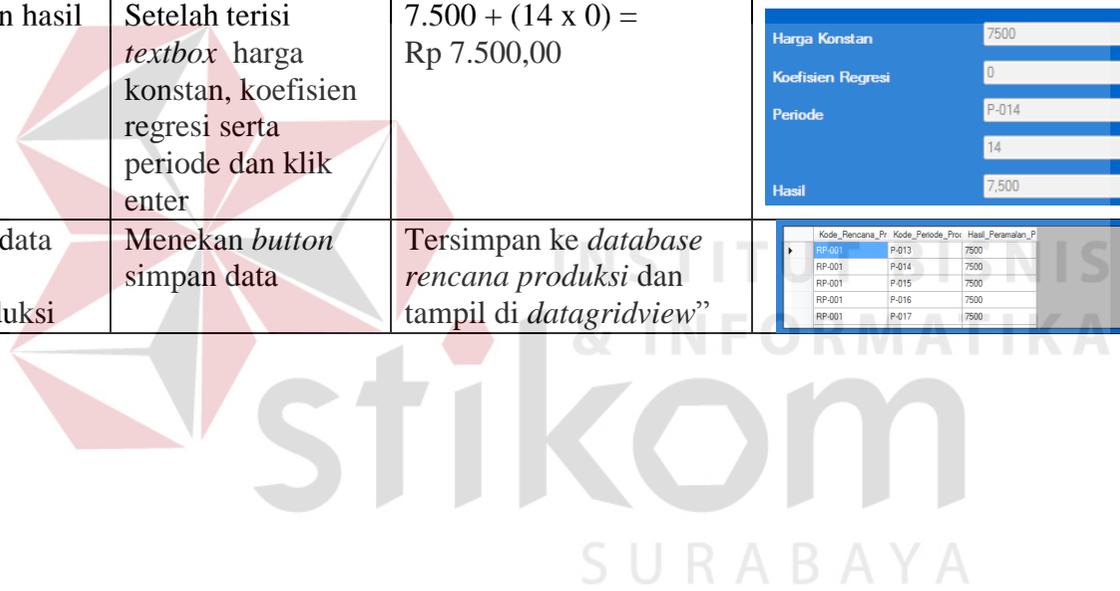
Dari contoh data perhitungan rencana produksi di atas, lalu tahap selanjutnya yaitu menguji perhitungan rencana produksi. Hasil dari

pengujian akan dibandingkan dengan rancangan desain uji coba perhitungan rencana produksi yang sebelumnya telah dibuat. Seperti pada tabel 4.6.



Tabel 4.6 Hasil Uji Coba *Form* Perhitungan Rencana Produksi

<i>Test Case ID</i>	<i>Test Case</i>	<i>Test Data</i>	<i>Output yang diharapkan</i>	<i>Actual Output</i>	<i>Kesimpulan</i>
1	Menampilkan hasil perhitungan peramalan	Setelah terisi <i>textbox</i> harga konstan, koefisien regresi serta periode dan klik enter	$7.500 + (14 \times 0) =$ Rp 7.500,00		Sesuai
2	Menyimpan data perhitungan rencana produksi	Menekan <i>button</i> simpan data	Tersimpan ke <i>database rencana produksi</i> dan tampil di <i>datagridview</i> "		Sesuai



4.4.3 Uji Coba *Form* Perhitungan Biaya Produksi

Form perhitungan biaya produksi digunakan untuk mengetahui biaya-biaya yang mempengaruhi penambahan dari kapasitas produksi setelah melakukan investasi. Perhitungan biaya produksi diperoleh dari pendapatan, biaya bahan baku, biaya *overhead* variabel, biaya tenaga kerja langsung, tenaga kerja tidak langsung serta biaya *overhead* tetap. Di *form* ini terdapat :

➤ *Combobox* kode rencana produksi

Gambar 4.46 *Combobox* Kode Rencana Produksi RP-001

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* kode rencana produksi, data ditampilkan berasal dari *database* rencana produksi. Total pendapatan didapatkan dari volume produksi dikali dengan harga jual yang otomatis terisi ketika memilih *combobox*. Adapun untuk kode selanjutnya dapat dilihat pada gambar 4.47, gambar 4.48 serta gambar 4.49.

Gambar 4.47 *Combobox* Kode Rencana Produksi RP-002

Gambar 4.48 *Combobox* Kode Rencana Produksi RP-003

Gambar 4.49 *Combobox* Kode Rencana Produksi RP-004Tabel 4.7 Pengujian *Combobox* Kode Rencana Produksi

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

➤ *Combobox* nama bahan baku

Gambar 4.50 *Combobox* Nama Bahan Baku

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* nama bahan baku, data ditampilkan berasal dari *database* master bahan.

Tabel 4.8 Pengujian *Combobox* Nama Bahan Baku

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

➤ *Combobox overhead variabel*



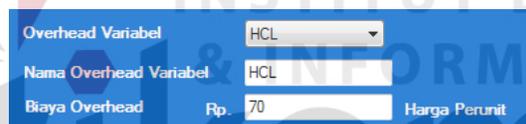
Overhead Variabel Enamel
 Nama Overhead Variabel Enamel
 Biaya Overhead Rp. 125 Harga Perunit

Gambar 4.51 *Combobox Overhead Variabel Enamel*

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox overhead variabel* dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari *database master overhead*.

Tabel 4.9 Pengujian *Combobox Overhead Variabel Enamel*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan



Overhead Variabel HCL
 Nama Overhead Variabel HCL
 Biaya Overhead Rp. 70 Harga Perunit

Gambar 4.52 *Combobox Overhead Variabel HCL*

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox overhead variabel* dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari *database master overhead*.

Tabel 4.10 Pengujian *Combobox Overhead Variabel HCL*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

Gambar 4.53 *Combobox Overhead Variabel Soda Api*

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox overhead* variabel dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari *database* master *overhead*.

Tabel 4.11 Pengujian *Combobox Overhead Variabel Soda Api*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

➤ *Combobox* tenaga kerja langsung

Gambar 4.54 *Combobox Tenaga Kerja Langsung Operator*

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* gaji tenaga kerja langsung dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari *database* master tenaga kerja.

Tabel 4.12 Pengujian *Combobox Tenaga Kerja Langsung Operator*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

Gambar 4.55 *Combobox* Tenaga Kerja Langsung Teknisi

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* gaji tenaga kerja langsung dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari *database* master tenaga kerja.

Tabel 4.13 Pengujian *Combobox* Tenaga Kerja Langsung Teknisi

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

➤ *Combobox* tenaga kerja tidak langsung

Gambar 4.56 *Combobox* Tenaga Kerja Tidak Langsung Akuntansi

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* gaji tenaga kerja tidak langsung dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari *database* master tenaga kerja.

Tabel 4.14 Pengujian *Combobox* Tenaga Kerja Tidak Langsung Akuntansi

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

Gambar 4.57 *Combobox* Tenaga Kerja Tidak Langsung Administrasi

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* gaji tenaga kerja tidak langsung dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari *database* master tenaga kerja.

Tabel 4.15 Pengujian *Combobox* Tenaga Kerja Tidak Langsung Administrasi

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan



Gambar 4.58 *Combobox* Tenaga Kerja Langsung *Engineering*

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* gaji tenaga kerja tidak langsung dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari *database* master tenaga kerja.

Tabel 4.16 Pengujian *Combobox* Tenaga Kerja Tidak Langsung *Engineering*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

➤ *Combobox* daya listrik mesin



Gambar 4.59 *Combobox* Daya Listrik Mesin *Grinding*

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* mesin dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari *database* master mesin.

Tabel 4.17 Pengujian *Combobox* Daya Listrik Mesin *Grinding*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2	Penginputan yang sama	Dapat dilakukan

Gambar 4.60 *Combobox* Daya Listrik Mesin *Milling*

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* mesin dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari *database* master mesin.

Tabel 4.18 Pengujian *Combobox* Daya Listrik Mesin *Milling*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2	Penginputan yang sama	Dapat dilakukan

Gambar 4.61 *Combobox* Daya Listrik Mesin *Pon Manual*

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* mesin dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari *database* master mesin.

Tabel 4.19 Pengujian *Combobox* Daya Listrik Mesin *Pon* Manual

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	Penginputan yang sama	Dapat dilakukan

Gambar 4.62 *Combobox* Daya Listrik Mesin Las Listrik

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* mesin dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari *database* master mesin.

Tabel 4.20 Pengujian *Combobox* Daya Listrik Mesin Las Listrik

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	Penginputan yang sama	Dapat dilakukan

➤ *Combobox* overhead tetap

Gambar 4.63 *Combobox* Overhead Tetap Gas

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* overhead tetap dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari *database* master overhead.

Tabel 4.21 Pengujian *Combobox Overhead Tetap Gas*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan


Gambar 4.64 *Combobox Overhead Tetap Air*

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox overhead* tetap dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari *database* master *overhead*.

Tabel 4.22 Pengujian *Combobox Overhead Tetap Air*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan


Gambar 4.65 *Combobox Overhead Tetap Telepon*

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox overhead* tetap dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari *database* master *overhead*.

Tabel 4.23 Pengujian *Combobox Overhead Tetap Telepon*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

Gambar 4.66 *Combobox Overhead Tetap Sewa Bangunan*

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox overhead* tetap dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari *database* master *overhead*.

Tabel 4.24 Pengujian *Combobox Overhead Tetap Sewa Bangunan*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

Gambar 4.67 *Combobox Overhead Tetap Alat Tulis Kantor*

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox overhead* tetap dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari *database* master *overhead*.

Tabel 4.25 Pengujian *Combobox Overhead Tetap Alat Tulis Kantor*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

➤ *Combobox* penyusutan mesin



Gambar 4.68 *Combobox* Penyusutan Mesin *Grinding*

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* mesin dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari proses perhitungan penyusutan yang didapatkan dari *database* mesin.

Tabel 4.26 Pengujian *Combobox* Penyusutan Mesin *Grinding*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan



Gambar 4.69 *Combobox* Penyusutan Mesin *Milling*

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* mesin dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari proses perhitungan penyusutan yang didapatkan dari *database* mesin.

Tabel 4.27 Pengujian *Combobox* Penyusutan Mesin *Milling*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan



Gambar 4.70 *Combobox* Penyusutan Mesin *Pon Manual*

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* mesin dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari proses perhitungan penyusutan yang didapatkan dari *database* mesin.

Tabel 4.28 Pengujian *Combobox* Penyusutan Mesin *Pon Manual*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

Gambar 4.71 *Combobox* Penyusutan Mesin Las Listrik

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* mesin dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari proses perhitungan penyusutan yang didapatkan dari *database* mesin.

Tabel 4.29 Pengujian *Combobox* Penyusutan Mesin Las Listrik

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

➤ *Textbox* hasil biaya bahan baku

Gambar 4.72 *Textbox* Hasil Biaya Bahan Baku

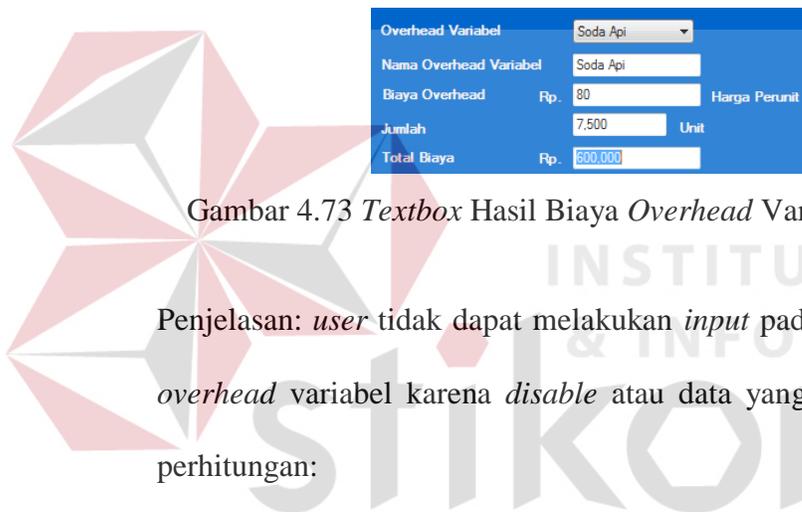
Penjelasan: *user* tidak dapat melakukan *input* pada *textbox* hasil total biaya bahan baku karena *disable* atau data yang ditampilkan berasal dari perhitungan:

Total biaya bahan baku =
 biaya bahan baku perunit x unit yang diproduksi.....(4.3)

Tabel 4.30 Pengujian *Textbox* Hasil Biaya Bahan Baku

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

➤ *Textbox* hasil biaya *overhead* variabel



Gambar 4.73 *Textbox* Hasil Biaya *Overhead* Variabel Soda Api

Penjelasan: *user* tidak dapat melakukan *input* pada *textbox* hasil total biaya *overhead* variabel karena *disable* atau data yang ditampilkan berasal dari perhitungan:

Total biaya =
 biaya *overhead* per unit x unit yang diproduksi.....(4.4)

Tabel 4.31 Pengujian *Textbox* Hasil *Overhead* Variabel Soda Api

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

Gambar 4.74 *Textbox* Hasil Biaya *Overhead* Variabel Enamel

Penjelasan: *user* tidak dapat melakukan *input* pada *textbox* hasil total biaya *overhead* variabel karena *disable* atau data yang ditampilkan berasal dari perhitungan:

$$\text{Total biaya} = \text{biaya } \textit{overhead} \text{ per unit} \times \text{unit yang diproduksi} \dots\dots\dots(4.5)$$

Tabel 4.32 Pengujian *Textbox* Hasil *Overhead* Variabel Enamel

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

Gambar 4.75 *Textbox* Hasil Biaya *Overhead* Variabel HCL

Penjelasan: *user* tidak dapat melakukan *input* pada *textbox* hasil total biaya *overhead* variabel karena *disable* atau data yang ditampilkan berasal dari perhitungan:

Total biaya =

$$\text{biaya overhead per unit} \times \text{unit yang diproduksi} \dots\dots\dots(4.6)$$

Tabel 4.33 Pengujian *Textbox* Hasil *Overhead* Variabel HCl

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

➤ *Textbox* hasil total biaya *overhead* variabel



Gambar 4.76 *Textbox* Hasil Total Biaya *Overhead* Variabel

Penjelasan: *user* tidak dapat melakukan *input* pada *textbox* hasil total biaya *overhead* variabel karena *disable* atau data yang ditampilkan berasal dari perhitungan:

$$\text{Total biaya overhead} = \text{Total biaya enamel} + \text{total biaya HCl} + \text{total biaya soda api} \dots\dots\dots(4.7)$$

Tabel 4.34 Pengujian *Textbox* Hasil Total Biaya *Overhead* Variabel

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

➤ *Textbox* hasil total biaya variabel



Gambar 4.77 *Textbox* Hasil Total Biaya Variabel

Penjelasan: *user* tidak dapat melakukan *input* pada *textbox* hasil total biaya *overhead* variabel karena *disable* atau data yang ditampilkan berasal dari perhitungan:

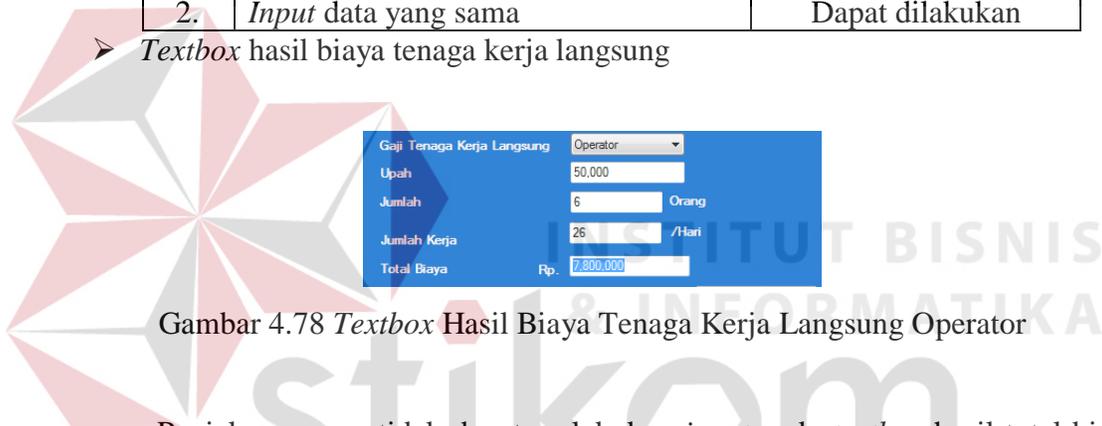
Total biaya variabel =

Total biaya bahan baku + total biaya *overhead* variabel(4.8)

Tabel 4.35 Pengujian *Textbox* Hasil Total Biaya Variabel

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

➤ *Textbox* hasil biaya tenaga kerja langsung



Gambar 4.78 *Textbox* Hasil Biaya Tenaga Kerja Langsung Operator

Penjelasan: *user* tidak dapat melakukan *input* pada *textbox* hasil total biaya tenaga kerja karena *disable* atau data yang ditampilkan berasal dari perhitungan:

Biaya tenaga kerja =

upah x jumlah orang x jumlah hari kerja(4.9)

Tabel 4.36 Pengujian *Textbox* Hasil Biaya Tenaga Kerja Langsung Operator

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

Gambar 4.79 *Textbox* Hasil Biaya Tenaga Kerja Langsung Teknisi

Penjelasan: *user* tidak dapat melakukan *input* pada *textbox* hasil total biaya tenaga kerja langsung karena *disable* atau data yang ditampilkan berasal dari perhitungan:

$$\text{Biaya tenaga kerja} = \text{upah} \times \text{jumlah orang} \times \text{jumlah hari kerja} \dots\dots\dots(4.10)$$

Tabel 4.37 Pengujian *Textbox* Hasil Biaya Tenaga Kerja Langsung Teknisi

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

➤ *Textbox* hasil total biaya tenaga kerja langsung

Gambar 4.80 *Textbox* Hasil Total Biaya Tenaga Kerja Langsung

Penjelasan: *user* tidak dapat melakukan *input* pada *textbox* hasil total biaya tenaga kerja langsung karena *disable* atau data yang ditampilkan berasal dari perhitungan:

$$\text{Total biaya tenaga kerja langsung} = \text{biaya tenaga kerja operator} + \text{biaya tenaga kerja teknisi} \dots\dots\dots(4.11)$$

Tabel 4.38 Pengujian *Textbox* Hasil Total Biaya Tenaga Kerja Langsung

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

➤ *Textbox* hasil biaya tenaga kerja tidak langsung

Gambar 4.81 *Textbox* Hasil Biaya Tenaga Kerja Tidak Langsung Akuntansi

Penjelasan: *user* tidak dapat melakukan *input* pada *textbox* hasil total biaya tenaga kerja tidak langsung karena *disable* atau data yang ditampilkan berasal dari perhitungan:

$$\text{Biaya tenaga kerja} = \text{upah} \times \text{jumlah orang} \times \text{jumlah hari kerja} \dots\dots\dots(4.12)$$

Tabel 4.39 Pengujian *Textbox* Hasil Biaya Tenaga Kerja Tidak Langsung Akuntansi

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

Gambar 4.82 *Textbox* Hasil Biaya Tenaga Kerja Tidak Langsung Administrasi

Penjelasan: *user* tidak dapat melakukan *input* pada *textbox* hasil total biaya tenaga kerja tidak langsung karena *disable* atau data yang ditampilkan berasal dari perhitungan:

Biaya tenaga kerja =

upah x jumlah orang x jumlah hari kerja(4.13)

Tabel 4.40 Pengujian *Textbox* Hasil Tenaga Kerja Tidak Langsung Administrasi

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan



Gambar 4.83 *Textbox* Hasil Biaya Tenaga Kerja Tidak Langsung *Engineering*

Penjelasan: *user* tidak dapat melakukan *input* pada *textbox* hasil total biaya tenaga kerja tidak langsung karena *disable* atau data yang ditampilkan berasal dari perhitungan:

Biaya tenaga kerja =

upah x jumlah orang x jumlah hari kerja(4.14)

Tabel 4.41 Pengujian *Textbox* Hasil Biaya Tenaga Kerja Tidak Langsung *Engineering*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

- *Textbox* hasil total biaya tenaga kerja tidak langsung



Gambar 4.84 *Textbox* Hasil Total Biaya Tenaga Kerja Tidak Langsung

Penjelasan: *user* tidak dapat melakukan *input* pada *textbox* hasil total biaya tenaga kerja langsung karena *disable* atau data yang ditampilkan berasal dari perhitungan:

$$\begin{aligned} \text{Total biaya tenaga kerja tidak langsung} &= \\ &\text{biaya tenaga kerja administrasi} + \text{biaya tenaga kerja akuntansi} + \text{biaya} \\ &\text{tenaga kerja engineering} \dots\dots\dots(4.15) \end{aligned}$$

Tabel 4.42 Pengujian *Textbox* Hasil Total Biaya Tenaga Kerja Tidak Langsung

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

- *Textbox* hasil total biaya listrik



Gambar 4.85 *Textbox* Hasil Total Biaya Listrik

Penjelasan: *user* tidak dapat melakukan *input* pada *textbox* hasil total biaya listrik karena *disable* atau data yang ditampilkan berasal dari perhitungan:

$$\begin{aligned} \text{Total biaya listrik} &= \\ &\text{Akumulasi penjumlahan dari beberapa biaya listrik mesin} \dots\dots\dots(4.16) \end{aligned}$$

Tabel 4.43 Pengujian *Textbox* Hasil Total Biaya Listrik

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

- *Textbox* hasil total biaya *overhead* tetap


Gambar 4.86 *Textbox* Hasil Total Biaya *Overhead* Tetap

Penjelasan: *user* tidak dapat melakukan *input* pada *textbox* hasil total biaya *overhead* tetap karena *disable* atau data yang ditampilkan berasal dari perhitungan:

Total biaya *overhead* tetap =
akumulasi penjumlahan dari beberapa biaya *overhead* tetap.....(4.17)

Tabel 4.44 Pengujian *Textbox* Hasil Total Biaya *Overhead* Tetap

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

- *Textbox* hasil total biaya penyusutan


Gambar 4.87 *Textbox* Hasil Total Biaya Penyusutan

Penjelasan: *user* tidak dapat melakukan *input* pada *textbox* hasil total biaya penyusutan karena *disable* atau data yang ditampilkan berasal dari perhitungan:

Total biaya penyusutan =

akumulasi penjumlahan dari beberapa biaya penyusutan.....(4.18)

Tabel 4.45 Pengujian *Textbox* Hasil Total Biaya Penyusutan

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

➤ *Textbox* hasil total biaya tetap



TOTAL BIAYA TETAP Rp. 31.874.125

Gambar 4.88 *Textbox* Hasil Total Biaya Tetap

Penjelasan: *user* tidak dapat melakukan *input* pada *textbox* hasil total biaya tetap karena *disable* atau data yang ditampilkan berasal dari perhitungan:

Total biaya tetap =

total gaji tenaga kerja langsung + tidak langsung + total biaya listrik + total

biaya *overhead* tetap + total biaya penyusutan.....(4.19)

Tabel 4.46 Pengujian *Textbox* Hasil Total Biaya Tetap

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

➤ *Textbox* hasil total biaya produksi



Gambar 4.89 *Textbox* Hasil Total Biaya Produksi

Penjelasan: *user* tidak dapat melakukan *input* pada *textbox* hasil total biaya produksi karena *disable* atau data yang ditampilkan berasal dari perhitungan:

$$\text{Total biaya produksi} = \text{total biaya variabel} + \text{total biaya Tetap} \dots \dots \dots (4.20)$$

Tabel 4.47 Pengujian *Textbox* Hasil Total Biaya Produksi

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

➤ *Button* simpan data



Gambar 4.90 *Button* Simpan Data

Penjelasan: *User* dapat melakukan penyimpanan dengan menekan *button* simpan maka kode akan bertambah serta data akan tersimpan ke dalam *database*.

Tabel 4.48 Pengujian *Button* Simpan Data

No.	Pengujian	Hasil
1.	Melakukan penyimpanan jika ada data yang kosong	Tidak dapat dilakukan

Langkah selanjutnya yaitu melakukan *test* data. *Test* data diperlukan untuk mengetahui apakah fungsi pada aplikasi yang telah dibuat ini sudah berjalan dengan baik dan sesuai dengan tujuan atau *output* yang diharapkan. Sebagai perbandingan, perlu dilakukan uji coba.

Diketahui:

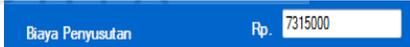
- *Textbox* total pendapatan: Rp 75.000.000
- *Textbox* biaya bahan baku : Rp 22.500.000
- *Textbox* biaya *overhead* variabel: Rp 2,062,500
- *Textbox* total biaya variabel: Rp 24,562,500
- *Textbox* total biaya tenaga kerja langsung : Rp 10,400,000
- *Textbox* total biaya tenaga kerja tidak langsung: Rp 7,280,000
- *Textbox* total biaya listrik: Rp 1,729,125
- *Textbox* total biaya *overhead* tetap: Rp 5,150,000
- *Textbox* total biaya penyusutan: Rp 7.315.000
- *Textbox* total biaya tetap: Rp 31,874,125
- *Textbox* total biaya produksi : Rp 56,436,625

Dari contoh data perhitungan biaya produksi di atas, lalu tahap selanjutnya yaitu menguji perhitungan biaya produksi. Seperti pada tabel 4.49, 4.50, 4.51.

Tabel 4.49 Hasil Uji Coba *Form* Perhitungan Biaya Produksi

<i>Test Case ID</i>	<i>Test Case</i>	<i>Test Data</i>	<i>Output yang diharapkan</i>	<i>Actual Output</i>	<i>Kesimpulan</i>
1	Menampilkan hasil total pendapatan	Setelah memilih <i>combobox</i>	Harga jual x volume produksi $10.000 \times 7.500 = \text{Rp } 75.000.000$		Sesuai
2	Menampilkan hasil total biaya bahan baku	Setelah memilih <i>combobox</i> dan memasukan jumlah unit	Biaya bahan baku x unit $3.000 \times 7.500 = \text{Rp } 22.500.000$		Sesuai
3	Menampilkan hasil total biaya <i>overhead</i> variabel	Setelah memilih <i>combobox</i> dan memasukan jumlah unit	Akumulasi dari biaya <i>overhead</i> variabel $937,500 + 525,000 + 600,000 = 2,062,500$		Sesuai
4	Menampilkan hasil total biaya variabel	Setelah semua <i>textbox</i> terisi dan klik enter	Biaya bahan baku + biaya <i>overhead</i> variabel $3.000 \times 7.500 = \text{Rp } 22.500.000$		Sesuai

Tabel 4.50 Hasil Uji Coba *Form* Perhitungan Biaya Produksi (Lanjutan)

5	Menampilkan hasil total biaya tenaga kerja langsung	Setelah terisi <i>textbox</i> dan klik enter	Akumulasi dari biaya tenaga kerja langsung $50.000 \times 8 \text{ orang} \times 26 \text{ hari} = 10,400,000$		Sesuai
6	Menampilkan hasil total biaya tenaga kerja tidak langsung	Setelah terisi <i>textbox</i> dan klik enter	Akumulasi dari biaya tenaga kerja tidak langsung $70.000 \times 4 \text{ orang} \times 26 \text{ hari} = 7,280,000$		Sesuai
7	Menampilkan hasil total biaya listrik	Setelah terisi <i>textbox</i> dan klik enter	Akumulasi dari biaya listrik $1,025,752 + 439,608 + 263,765 = 1,729,125$		Sesuai
8	Menampilkan hasil total biaya <i>overhead</i> tetap	Setelah terisi <i>textbox</i> dan klik enter	Akumulasi dari biaya <i>overhead</i> tetap $1,750,000 + 500,000 + 300,000 + 2,500,000 + 100,000 = 5,150,000$		Sesuai
9	Menampilkan hasil total biaya penyusutan	Setelah terisi <i>textbox</i> dan klik enter	Akumulasi dari biaya penyusutan $3,250,000 + 2,250,000 + 1,250,000 + 565,000 = 7.315.000$		Sesuai

Tabel 4.51 Hasil Uji Coba *Form* Perhitungan Biaya produksi (Lanjutan)

10	Menampilkan hasil total biaya tetap	Setelah terisi <i>textbox</i> dan klik enter	Biaya tenaga kerja langsung + biaya tenaga tidak langsung + biaya listrik + biaya <i>overhead</i> tetap + biaya penyusutan $10,400,000 + 7,280,000 + 1,729,125 + 5,150,000 + 7.315.000 = 31,874,125$		Sesuai
11	Menampilkan hasil total biaya produksi	Setelah terisi <i>textbox</i> dan klik enter	Penjumlahan biaya variabel + biaya tetap $24,562,500 + 31,874,125 = 56,436,625$		Sesuai

4.4.4 Uji Coba *Form* Perhitungan Laba Rugi

Form perhitungan laba rugi digunakan sebagai mengetahui estimasi keuangan setelah melakukan investasi. Data perhitungan laba rugi diperoleh dari perhitungan pendapatan, biaya variabel, biaya tetap, kemudian laba sebelum pajak, pajak dan laba setelah pajak. Di *form* ini terdapat:

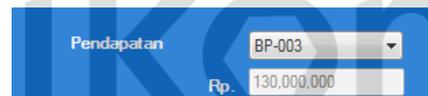
➤ *Combobox* Kode Pendapatan



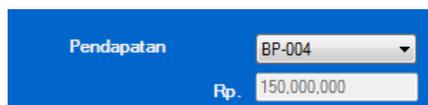
Gambar 4.91 *Combobox* Kode Pendapatan BP-001



Gambar 4.92 *Combobox* Kode Pendapatan BP-002



Gambar 4.93 *Combobox* Kode Pendapatan BP-003



Gambar 4.94 *Combobox* Kode Pendapatan BP-004

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* kode pendapatan, data ditampilkan berasal dari *database* biaya produksi.

Tabel 4.52 Pengujian *Combobox* Kode Pendapatan

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

➤ *Combobox* kode biaya produksi

Gambar 4.95 *Combobox* Kode Biaya Produksi BP-001

Gambar 4.96 *Combobox* Kode Biaya Produksi BP-002

Gambar 4.97 *Combobox* Kode Biaya Produksi BP-003

Gambar 4.98 *Combobox* Kode Biaya Produksi BP-004

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* kode mesin, data ditampilkan berasal dari *database* mesin.

Tabel 4.53 Pengujian *Combobox* Kode Biaya Produksi

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

➤ *Textbox* hasil laba setelah pajak



Gambar 4.99 *Textbox* Hasil Laba Setelah Pajak Periode 1



Gambar 4.100 *Textbox* Hasil Laba Setelah Pajak Periode 2



Gambar 4.101 *Textbox* Hasil Laba Setelah Pajak Periode 3



Gambar 4.102 *Textbox* Hasil Laba Setelah Pajak Periode 4

Penjelasan: *User* tidak dapat melakukan *input* pada *textbox* hasil laba setelah pajak karena data ditampilkan berasal dari perhitungan:

Laba setelah pajak =

$$(\text{pendapatan} - \text{biaya total}) \times 30\% \dots\dots\dots(4.21)$$

Tabel 4.54 Pengujian *Textbox* Hasil Laba Setelah Pajak

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

➤ *Button* simpan data



Gambar 4.103 *Button* Simpan Data

Penjelasan: *User* dapat melakukan penyimpanan dengan menekan *button* simpan maka kode akan bertambah serta data akan tersimpan ke dalam *database*.

Tabel 4.55 Pengujian *Button* Simpan Data

No.	Pengujian	Hasil
1.	Melakukan penyimpanan jika ada data yang kosong	Tidak dapat dilakukan

Langkah selanjutnya yaitu melakukan *test* data. *Test* data diperlukan untuk mengetahui apakah fungsi pada aplikasi yang telah dibuat ini sudah berjalan dengan baik dan sesuai dengan tujuan atau *output* yang diharapkan. Sebagai perbandingan, perlu dilakukan uji coba.

Diketahui:

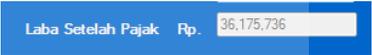
- Laba setelah pajak periode 1 : 12.994.363
- Laba setelah pajak periode 2 : 22.176.030
- Laba setelah pajak periode 3 : 36.175.736
- Laba setelah pajak periode 4 : 45.507.887

Dari contoh data perhitungan laba rugi di atas, lalu tahap selanjutnya yaitu menguji *form* perhitungan laba rugi dan hasil pengujiannya akan di bandingkan. Seperti pada tabel 4.56 dan 4.57.

Tabel 4.56 Hasil Uji Coba *Form* Perhitungan Laba Rugi

<i>Test Case ID</i>	<i>Test Case</i>	<i>Test Data</i>	<i>Output yang diharapkan</i>	<i>Actual Output</i>	<i>Kesimpulan</i>
1	Menampilkan nilai hasil laba setelah pajak periode 1	<ul style="list-style-type: none"> Memilih kode biaya produksi : BP-001 Kemudian klik enter Inputkan nilai pajak 	Laba setelah pajak = (pendapatan – (biaya variabel + biaya tetap)) - pajak $(75.000.000 - (24.562.500 + 31.874.125)) - 30\% =$ Rp 12.994.363		Sesuai
2	Menampilkan nilai hasil laba setelah pajak periode 2	<ul style="list-style-type: none"> Memilih kode biaya produksi : BP-002 Kemudian klik enter Inputkan nilai pajak 	Laba setelah pajak = (pendapatan – (biaya variabel + biaya tetap)) - pajak $(100.000.000 - (32.750.000 + 35.569.957)) - 30\% =$ Rp 22.176.030		Sesuai

Tabel 4.57 Hasil Uji Coba *Form* Perhitungan Laba Rugi (Lanjutan)

3	Menampilkan nilai hasil laba setelah pajak periode 3	<ul style="list-style-type: none"> • Memilih kode biaya produksi : BP-003 • Kemudian klik enter • Inputkan nilai pajak 	<p>Laba setelah pajak = (pendapatan – (biaya variabel + biaya tetap)) - pajak $(130.000.000 - (42.575.000 + 35.745.377)) - 30\% = \text{Rp } 36.175.736$</p>		Sesuai
4	Menampilkan nilai hasil laba setelah pajak periode 4	<ul style="list-style-type: none"> • Memilih kode biaya produksi : BP-004 • Kemudian klik enter • Inputkan nilai pajak 	<p>Laba setelah pajak = (pendapatan – (biaya variabel + biaya tetap)) – pajak $(150.000.000 - (49.125.000 + 35.863.733)) - 30\% = \text{Rp } 45.507.887$</p>		Sesuai

4.4.5 Uji Coba *Form* Perhitungan Aliran Kas Bersih

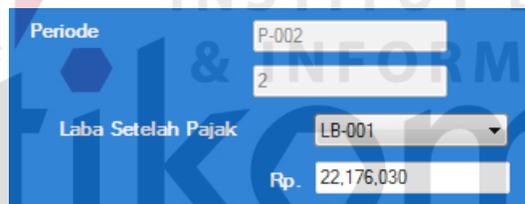
Form perhitungan aliran kas bersih digunakan sebagai mengetahui estimasi penerimaan-penerimaan atau pengeluaran-pengeluaran pendapatan setelah melakukan investasi. Data perhitungan aliran kas bersih diperoleh dari perhitungan laba setelah pajak perperiode dengan penyusutan. Di *form* ini terdapat:

➤ *Textbox* hasil laba setelah pajak



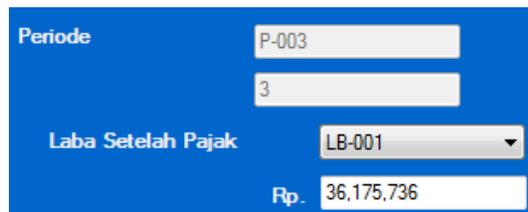
Periode	P-001
	1
Laba Setelah Pajak	LB-001
Rp.	12,994,363

Gambar 4.104 *Textbox* Hasil Laba Setelah Pajak Periode 1



Periode	P-002
	2
Laba Setelah Pajak	LB-001
Rp.	22,176,030

Gambar 4.105 *Textbox* Hasil Laba Setelah Pajak Periode 2



Periode	P-003
	3
Laba Setelah Pajak	LB-001
Rp.	36,175,736

Gambar 4.106 *Textbox* Hasil Laba Setelah Pajak Periode 3

Gambar 4.107 *Textbox* Hasil Laba Setelah Pajak Periode 4

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* kode laba rugi, data ditampilkan berasal dari *database* laba rugi.

Tabel 4.58 Pengujian *Textbox* Hasil Laba Setelah Pajak Perperiode

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

➤ *Textbox* hasil aliran kas bersih perperiode

Gambar 4.108 *Textbox* Hasil Aliran Kas Bersih Periode 1

Gambar 4.109 *Textbox* Hasil Aliran Kas Bersih Periode 2

Gambar 4.110 *Textbox* Hasil Aliran Kas Bersih Periode 3

Gambar 4.111 *Textbox* Hasil Aliran Kas Bersih Periode 4

Penjelasan: *User* tidak dapat melakukan *input* pada *textbox* hasil laba setelah pajak karena data ditampilkan berasal dari perhitungan:

Aliran kas bersih =

laba setelah pajak perperiode + penyusutan mesin(4.22)

Tabel 4.59 Pengujian *Textbox* Hasil Aliran Kas Bersih

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

➤ *Button* simpan & hitung



Gambar 4.112 *Button* Simpan & Hitung

Penjelasan: *User* dapat melakukan penyimpanan dengan menekan *button* simpan dan menghitung hasil aliran kas bersih dengan menekan *button* hitung.

Tabel 4.60 Pengujian *Button* Simpan & Hitung

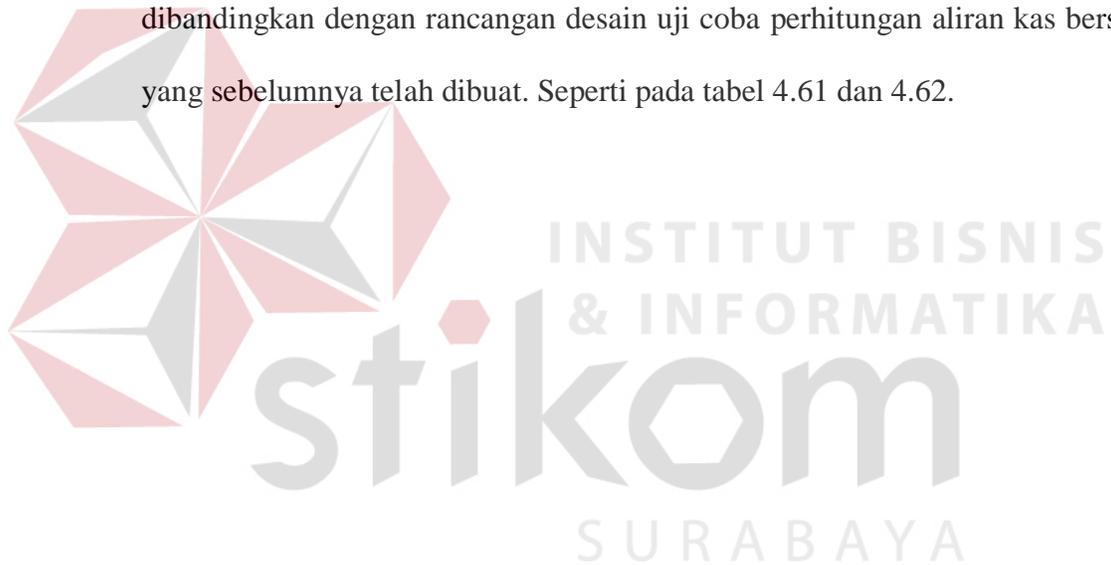
No.	Pengujian	Hasil
1.	Melakukan penyimpanan jika ada data yang kosong	Tidak dapat dilakukan
2.	Melakukan proses hitung hasil jika ada data yang kosong	Tidak dapat dilakukan

Langkah selanjutnya yaitu melakukan *test* data. *Test* data diperlukan untuk mengetahui apakah fungsi pada aplikasi yang telah dibuat ini sudah berjalan dengan baik dan sesuai dengan tujuan atau *output* yang diharapkan. Sebagai perbandingan, perlu dilakukan uji coba.

Diketahui:

- Laba setelah pajak periode 1 : 12.994.363
- Laba setelah pajak periode 2 : 22.176.030
- Laba setelah pajak periode 3 : 36.175.736
- Laba setelah pajak periode 4 : 45.507.887
- Penyusutan : 2.250.000

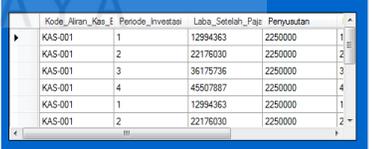
Dari contoh data perhitungan aliran kas bersih di atas, lalu tahap selanjutnya yaitu menguji *form* perhitungan aliran kas bersih. Hasil dari pengujian akan dibandingkan dengan rancangan desain uji coba perhitungan aliran kas bersih yang sebelumnya telah dibuat. Seperti pada tabel 4.61 dan 4.62.



Tabel 4.61 Hasil Uji Coba *Form* Perhitungan Aliran Kas Bersih

<i>Test Case ID</i>	<i>Test Case</i>	<i>Test Data</i>	<i>Output yang diharapkan</i>	<i>Actual Output</i>	<i>Kesimpulan</i>
1	Menampilkan nilai hasil aliran kas bersih periode 1	<ul style="list-style-type: none"> Pastikan <i>textbox</i> periode : P-001 Memilih kode laba rugi : LB-001 Memilih kode mesin : KM-002 Kemudian klik enter 	Aliran kas bersih = laba setelah pajak + penyusutan $12.994.363 + 2.250.000 =$ Rp 15.244.363		Sesuai
2	Menampilkan nilai hasil aliran kas bersih periode 2	<ul style="list-style-type: none"> Pastikan <i>textbox</i> periode : P-002 Memilih kode laba rugi : LB-001 Memilih kode mesin : KM-002 Kemudian klik enter 	Aliran kas bersih = laba setelah pajak + penyusutan $22.176.030 + 2.250.000 =$ Rp 24.426.030		Sesuai

Tabel 4.62 Hasil Uji Coba *Form* Perhitungan Aliran Kas Bersih (Lanjutan)

3	Menampilkan nilai hasil aliran kas bersih periode 3	<ul style="list-style-type: none"> • Pastikan <i>textbox</i> periode : P-003 • Memilih kode laba rugi : LB-001 • Memilih kode mesin : KM-002 • Kemudian klik enter 	<p>Aliran kas bersih = laba setelah pajak + penyusutan</p> $36.175.736 + 2.250.000 = \text{Rp } 38.425.736$		Sesuai
4	Menampilkan nilai hasil aliran kas bersih periode 4	<ul style="list-style-type: none"> • Pastikan <i>textbox</i> periode : P-004 • Memilih kode laba rugi : LB-001 • Memilih kode mesin : KM-002 • Kemudian klik enter 	<p>Aliran kas bersih = laba setelah pajak + penyusutan</p> $45.507.887 + 2.250.000 = \text{Rp } 47.757.887$		Sesuai
5	Menyimpan data perhitungan data aliran kas bersih	Menekan <i>button</i> simpan data	Tersimpan ke <i>database</i> aliran kas bersih dan tampil di <i>datagridview</i>		Sesuai

Setelah melakukan pengujian terhadap *form* pendukung perhitungan dengan pengujian dasar maka selanjutnya akan dilakukan pengujian *form* perhitungan berdasarkan hasil analisa yang telah dibuat.

4.4.6 Uji Coba Hasil Analisa *Form* Master Mesin

Form master mesin digunakan untuk menyimpan data mesin yang ada dan yang akan diinvestasikan atau dibeli. Di *form* master mesin terdapat perhitungan penyusutan. Perhitungan penyusutan diperoleh dari harga perolehan dikurangi dengan nilai sisa kemudian hasilnya dibagi dengan umur ekonomis mesin. Di *form* ini terdapat :

➤ *Textbox* perhitungan penyusutan

Umur Ekonomis	4	Tahun
Harga Perolehan	Rp. 20,000,000	
Daya	1.5	Kwh
Nilai Sisa	Rp. 11,000,000	
Penyusutan	Rp. 2,250,000	

Gambar 4.113 *Textbox* Perhitungan Penyusutan

Penjelasan: *User* menginputkan data umur ekonomis, harga perolehan, serta nilai sisa. *User* tidak dapat melakukan *input* pada *textbox* hasil penyusutan karena *disable* atau data yang ditampilkan berasal dari perhitungan:

Penyusutan =

(harga perolehan – nilai sisa) / umur ekonomis(4.23)

Tabel 4.63 Pengujian *Textbox* Hasil Penyusutan

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

➤ *Button* simpan data

Gambar 4.114 *Button* Simpan Data

Penjelasan: *User* dapat melakukan penyimpanan dengan menekan *button* simpan maka kode akan bertambah serta data akan tersimpan ke dalam *database*.

Tabel 4.64 Pengujian *Button* Simpan Data

No.	Pengujian	Hasil
1.	Melakukan penyimpanan jika ada data yang kosong	Tidak dapat dilakukan

Langkah selanjutnya yaitu melakukan *test* data. *Test* data diperlukan untuk mengetahui apakah fungsi pada aplikasi yang telah dibuat ini sudah berjalan dengan baik dan sesuai dengan tujuan atau *output* yang diharapkan. Sebagai perbandingan, perlu dilakukan uji coba.

Diketahui:

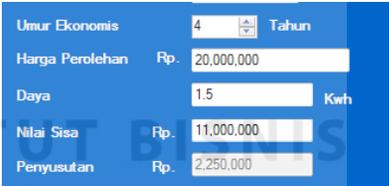
- *Textbox* harga perolehan: Rp 20.000.000
- *Textbox* nilai sisa : Rp 11.000.000
- *Numeric updown* umur ekonomis: 4 tahun

- *Textbox* penyusutan: Rp 2.250.000

Dari contoh data master dan perhitungan penyusutan di atas, lalu tahap selanjutnya yaitu menguji *form* master mesin dan perhitungan penyusutan. Hasil dari pengujian akan dibandingkan dengan rancangan desain uji coba perhitungan penyusutan yang sebelumnya telah dibuat. Seperti pada tabel 4.65.



Tabel 4.65 Hasil Uji Coba *Form* Master Mesin

<i>Test Case ID</i>	<i>Test Case</i>	<i>Test Data</i>	<i>Output yang diharapkan</i>	<i>Actual Output</i>	<i>Kesimpulan</i>
1	Menampilkan hasil perhitungan penyusutan	Setelah terisi <i>textbox</i> harga perolehan, nilai sisa serta <i>numericupdown</i> umur ekonomis dan klik enter	(investasi – nilai sisa)/ umur ekonomis (Rp 20.000.000 – Rp 11.000.000) / 4 = Rp 2.250.000,00		Sesuai
2	Menyimpan data master mesin dan perhitungan penyusutan	Menekan <i>button</i> simpan data	Tersimpan ke <i>database</i> mesin dan tampil kotak dialog “data masuk”		Sesuai

SURABAYA

4.4.7 Uji Coba *Form* Hasil Analisa Perhitungan Rencana Produksi

Form perhitungan rencana produksi digunakan untuk mengetahui proyeksi produksi setelah melakukan investasi. Di *form* rencana produksi terdapat perhitungan peramalan produksi. Perhitungan peramalan produksi diperoleh dari harga konstan ditambah dengan koefisien regresi dikali dengan subyek variabel (periode). Di *form* ini terdapat

➤ *Textbox* hasil peramalan

Harga Jual	Rp.	10,000
Harga Konstan		10000
Koefisien Regresi		0
Periode		P-013
Hasil		13
		Unit

Gambar 4.115 *Textbox* Hasil Peramalan

Penjelasan: *User* menginputkan harga jual, *user* tidak dapat melakukan *input* pada *textbox* harga konstan dan koefisien regresi karena data berasal dari *database* detail peramalan, *user* tidak bisa menginputkan periode karena otomatis muncul dari *database* periode produksi dan *user* tidak dapat menginputkan hasil peramalan karena *disable* atau data yang ditampilkan berasal dari perhitungan:

Peramalan =

$$Y = a + bX \dots \dots \dots (4.24)$$

Dimana : Y = Subyek dalam variabel dependen yang diprediksikan

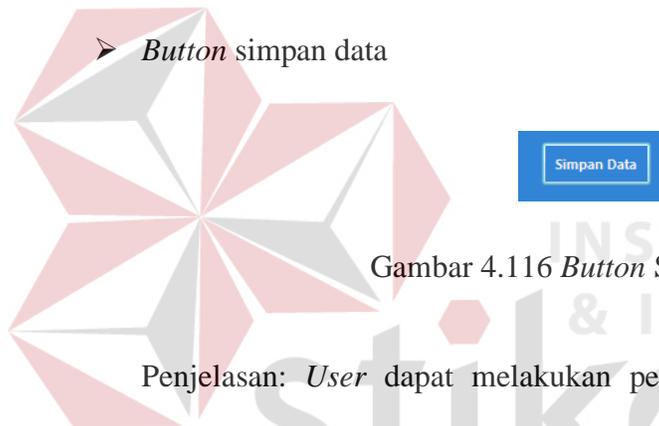
a = Harga Y bila X=0 (harga konstan)

b = Koefisien regresi

X=Subyek pada variabel independen yang mempunyai nilai tertentu (Periode)

Tabel 4.66 Pengujian *Textbox* Hasil Peramalan

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Tidak dapat dilakukan



Gambar 4.116 *Button* Simpan Data

Penjelasan: *User* dapat melakukan penyimpanan dengan menekan *button* simpan data maka kode akan bertambah serta data akan tersimpan ke dalam *database*.

Tabel 4.67 Pengujian *Button* Simpan Data

No.	Pengujian	Hasil
1.	Melakukan penyimpanan jika ada data yang kosong	Tidak dapat dilakukan

Langkah selanjutnya yaitu melakukan *test* data. *Test* data diperlukan untuk mengetahui apakah fungsi pada aplikasi yang telah dibuat ini sudah

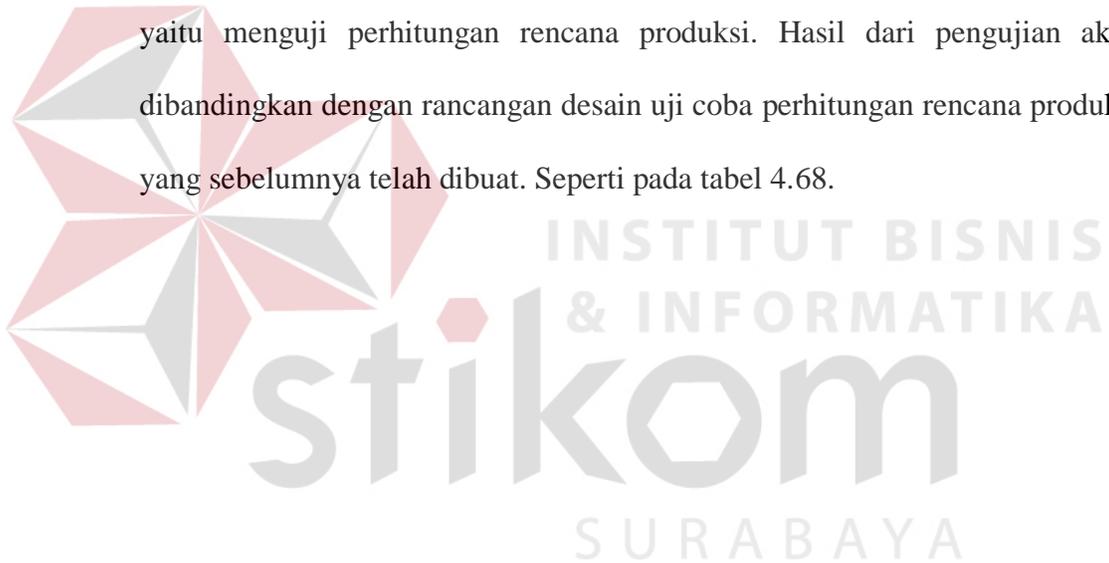
berjalan dengan baik dan sesuai dengan tujuan atau *output* yang diharapkan.

Sebagai perbandingan, perlu dilakukan uji coba.

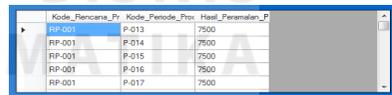
Diketahui:

- *Textbox* harga konstan: Rp 10.000
- *Textbox* koefisien regresi : Rp 0
- *Textbox* periode: 13
- *Textbox* hasil peramalan: Rp 10.000

Dari contoh data perhitungan rencana produksi di atas, lalu tahap selanjutnya yaitu menguji perhitungan rencana produksi. Hasil dari pengujian akan dibandingkan dengan rancangan desain uji coba perhitungan rencana produksi yang sebelumnya telah dibuat. Seperti pada tabel 4.68.



Tabel 4.68 Hasil Uji Coba *Form* Perhitungan Rencana Produksi

<i>Test Case ID</i>	<i>Test Case</i>	<i>Test Data</i>	<i>Output yang diharapkan</i>	<i>Actual Output</i>	<i>Kesimpulan</i>
1	Menampilkan hasil perhitungan peramalan	Setelah terisi <i>textbox</i> harga konstan, koefisien regresi serta periode dan klik enter	Harga konstan + (periode produksi x koefisien regresi) $10.000 + (13 \times 0) =$ Rp 10.000,00		Sesuai
2	Menyimpan data perhitungan rencana produksi	Menekan <i>button</i> simpan data	Tersimpan ke <i>database</i> rencana produksi dan tampil di <i>datagridview</i>		Sesuai



4.4.8 Uji Coba *Form* Hasil Analisa Perhitungan Biaya Produksi

Form perhitungan biaya produksi digunakan untuk mengetahui biaya-biaya yang mempengaruhi penambahan dari kapasitas produksi setelah melakukan investasi. Perhitungan biaya produksi diperoleh dari pendapatan, biaya bahan baku, biaya *overhead* variabel, biaya tenaga kerja langsung, tenaga kerja tidak langsung serta biaya *overhead* tetap. Di *form* ini terdapat :

➤ *Combobox* kode rencana produksi

Gambar 4.117 *Combobox* Kode Rencana Produksi

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* kode rencana produksi, data ditampilkan berasal dari *database* rencana produksi. Total pendapatan didapatkan dari volume produksi dikali dengan harga jual yang otomatis terisi ketika memilih *combobox*.

Tabel 4.69 Pengujian *Combobox* Kode Rencana Produksi

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

➤ *Combobox* nama bahan baku

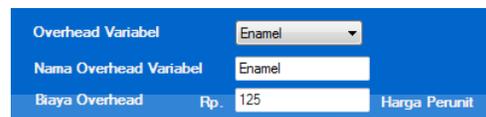
Gambar 4.118 *Combobox* Nama Bahan Baku

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* nama bahan baku, data ditampilkan berasal dari *database* master bahan.

Tabel 4.70 Pengujian *Combobox* Nama Bahan Baku

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

➤ *Combobox overhead* variabel



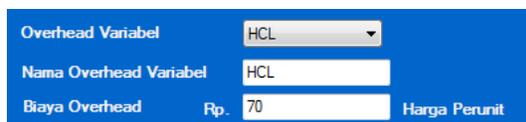
The screenshot shows a blue header bar with the text 'Overhead Variabel'. Below it, there is a dropdown menu with 'Enamel' selected. Underneath the dropdown is a text box containing 'Enamel'. At the bottom, there is a text box with 'Rp. 125' and the label 'Harga Perunit' to its right.

Gambar 4.119 *Combobox Overhead* Variabel Enamel

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox overhead* variabel dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari *database* master *overhead*.

Tabel 4.71 Pengujian *Combobox Overhead* Variabel Enamel

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan



The screenshot shows a blue header bar with the text 'Overhead Variabel'. Below it, there is a dropdown menu with 'HCL' selected. Underneath the dropdown is a text box containing 'HCL'. At the bottom, there is a text box with 'Rp. 70' and the label 'Harga Perunit' to its right.

Gambar 4.120 *Combobox Overhead* Variabel HCL

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox overhead* variabel dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari *database* master *overhead*.

Tabel 4.72 Pengujian *Combobox Overhead Variabel HCl*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

Gambar 4.121 *Combobox Overhead Variabel Soda Api*

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox overhead* variabel dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari *database* master *overhead*.

Tabel 4.73 Pengujian *Combobox Overhead Variabel Soda Api*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

➤ *Combobox* tenaga kerja langsung

Gambar 4.122 *Combobox Tenaga Kerja Langsung Operator*

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* gaji tenaga kerja langsung dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari *database* master tenaga kerja.

Tabel 4.74 Pengujian *Combobox Tenaga Kerja Langsung Operator*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

Gambar 4.123 *Combobox* Tenaga Kerja Langsung Teknisi

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* gaji tenaga kerja langsung dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari *database* master tenaga kerja.

Tabel 4.75 Pengujian *Combobox* Tenaga Kerja Langsung Teknisi

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

➤ *Combobox* tenaga kerja tidak langsung

Gambar 4.124 *Combobox* Tenaga Kerja Tidak Langsung Akuntansi

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* gaji tenaga kerja tidak langsung dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari *database* master tenaga kerja.

Tabel 4.76 Pengujian *Combobox* Tenaga Kerja Tidak Langsung Akuntansi

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

Gambar 4.125 *Combobox* Tenaga Kerja Tidak Langsung Administrasi

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* gaji tenaga kerja tidak langsung dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari *database* master tenaga kerja.

Tabel 4.77 Pengujian *Combobox* Tenaga Kerja Tidak Langsung Administrasi

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan



Gambar 4.126 *Combobox* Tenaga Kerja Langsung *Engineering*

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* gaji tenaga kerja tidak langsung dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari *database* master tenaga kerja.

Tabel 4.78 Pengujian *Combobox* Tenaga Kerja Tidak Langsung *Engineering*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

➤ *Combobox* daya listrik mesin



Gambar 4.127 *Combobox* Daya Listrik Mesin *Grinding*

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* mesin dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari *database* master mesin.

Tabel 4.79 Pengujian *Combobox* Daya Listrik Mesin *Grinding*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2	Penginputan yang sama	Dapat dilakukan

Gambar 4.128 *Combobox* Daya Listrik Mesin *Milling*

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* mesin dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari *database* master mesin.

Tabel 4.80 Pengujian *Combobox* Daya Listrik Mesin *Milling*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2	Penginputan yang sama	Dapat dilakukan

Gambar 4.129 *Combobox* Daya Listrik Mesin *Pon Manual*

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* mesin dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari *database* master mesin.

Tabel 4.81 Pengujian *Combobox* Daya Listrik Mesin *Pon Manual*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

2	Penginputan yang sama	Dapat dilakukan
---	-----------------------	-----------------

Gambar 4.130 *Combobox* Daya Listrik Mesin Las Listrik

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* mesin dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari *database* master mesin.

Tabel 4.82 Pengujian *Combobox* Daya Listrik Mesin Las Listrik

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2	Penginputan yang sama	Dapat dilakukan

➤ *Combobox* overhead tetap

Gambar 4.131 *Combobox* Overhead Tetap Gas

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* overhead tetap dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari *database* master overhead.

Tabel 4.83 Pengujian *Combobox* Overhead Tetap Gas

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

Overhead Tetap: Air
 Nama Overhead Tetap: Air
 Biaya Overhead: Rp. 500,000 Harga Perunit

Gambar 4.132 *Combobox Overhead Tetap Air*

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox overhead* tetap dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari *database* master *overhead*.

Tabel 4.84 Pengujian *Combobox Overhead Tetap Air*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

Overhead Tetap: Telepon
 Nama Overhead Tetap: Telepon
 Biaya Overhead: Rp. 300,000 Harga Perunit

Gambar 4.133 *Combobox Overhead Tetap Telepon*

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox overhead* tetap dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari *database* master *overhead*.

Tabel 4.85 Pengujian *Combobox Overhead Tetap Telepon*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

Overhead Tetap: Sewa Bangunan
 Nama Overhead Tetap: Sewa Bangunan
 Biaya Overhead: Rp. 2,500,000 Harga Perunit

Gambar 4.134 *Combobox Overhead Tetap Sewa Bangunan*

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* *overhead* tetap dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari *database* master *overhead*.

Tabel 4.86 Pengujian *Combobox* *Overhead* Tetap Sewa Bangunan

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan



Gambar 4.135 *Combobox* *Overhead* Tetap Alat Tulis Kantor

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* *overhead* tetap dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari *database* master *overhead*.

Tabel 4.87 Pengujian *Combobox* *Overhead* Tetap Alat Tulis Kantor

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

➤ *Combobox* penyusutan mesin



Gambar 4.136 *Combobox* Penyusutan Mesin *Grinding*

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* mesin dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari proses perhitungan penyusutan yang didapatkan dari *database* mesin.

Tabel 4.88 Pengujian *Combobox* Penyusutan Mesin *Grinding*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan



Gambar 4.137 *Combobox* Penyusutan Mesin *Milling*

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* mesin dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari proses perhitungan penyusutan yang didapatkan dari *database* mesin.

Tabel 4.89 Pengujian *Combobox* Penyusutan Mesin *Milling*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan



Gambar 4.138 *Combobox* Penyusutan Mesin *Pon Manual*

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* mesin dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari proses perhitungan penyusutan yang didapatkan dari *database* mesin.

Tabel 4.90 Pengujian *Combobox* Penyusutan Mesin *Pon* Manual

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

Gambar 4.139 *Combobox* Penyusutan Mesin Las Listrik

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* mesin dan *textbox* akan terisi secara otomatis, data ditampilkan berasal dari proses perhitungan penyusutan yang didapatkan dari *database* mesin.

Tabel 4.91 Pengujian *Combobox* Penyusutan Mesin Las Listrik

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

➤ *Textbox* hasil biaya bahan baku

Gambar 4.140 *Textbox* Hasil Biaya Bahan Baku

Penjelasan: *User* menginputkan jumlah unit dan *user* tidak dapat melakukan *input* pada *textbox* hasil total biaya bahan baku karena *disable* atau data yang ditampilkan berasal dari perhitungan:

Total biaya bahan baku =

biaya bahan baku perunit x unit yang diproduksi.....(4.25)

Tabel 4.92 Pengujian *Textbox* Hasil Biaya Bahan Baku

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

➤ *Textbox* hasil biaya *overhead* variabel

Gambar 4.141 *Textbox* Hasil Biaya *Overhead* Variabel Soda Api

Penjelasan: *user* tidak dapat melakukan *input* pada *textbox* hasil total biaya *overhead* variabel karena *disable* atau data yang ditampilkan berasal dari perhitungan:

$$\text{Total biaya} = \text{biaya } \textit{overhead} \text{ perunit} \times \text{unit yang diproduksi} \dots\dots\dots (4.26)$$

Tabel 4.93 Pengujian *Textbox* Hasil *Overhead* Variabel Soda Api

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

Gambar 4.142 *Textbox* Hasil Biaya *Overhead* Variabel Enamel

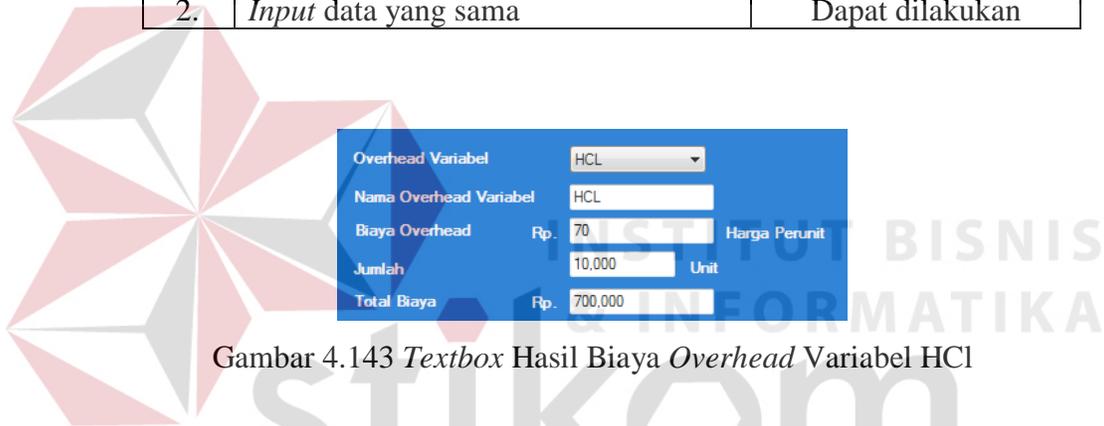
Penjelasan: *user* tidak dapat melakukan *input* pada *textbox* hasil total biaya *overhead* variabel karena *disable* atau data yang ditampilkan berasal dari perhitungan:

Total biaya =

biaya *overhead* perunit x unit yang diproduksi.....(4.27)

Tabel 4.94 Pengujian *Textbox* Hasil *Overhead* Variabel Enamel

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan



Gambar 4.143 *Textbox* Hasil Biaya *Overhead* Variabel HCl

Penjelasan: *user* tidak dapat melakukan *input* pada *textbox* hasil total biaya *overhead* variabel karena *disable* atau data yang ditampilkan berasal dari perhitungan:

Total biaya =

biaya *overhead* perunit x unit yang diproduksi.....(4.28)

Tabel 4.95 Pengujian *Textbox* Hasil *Overhead* Variabel HCl

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

➤ *Textbox* hasil total biaya *overhead* variabel



Gambar 4.144 *Textbox* Hasil Total Biaya *Overhead* Variabel

Penjelasan: *user* tidak dapat melakukan *input* pada *textbox* hasil total biaya *overhead* variabel karena *disable* atau data yang ditampilkan berasal dari perhitungan:

Total biaya *overhead* =

$$\text{total biaya enamel} + \text{total biaya HCl} + \text{total biaya soda api} \dots\dots\dots(4.29)$$

Tabel 4.96 Pengujian *Textbox* Hasil Total Biaya *Overhead* Variabel

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

➤ *Textbox* hasil total biaya variabel



Gambar 4.145 *Textbox* Hasil Total Biaya Variabel

Penjelasan: *user* tidak dapat melakukan *input* pada *textbox* hasil total biaya *overhead* variabel karena *disable* atau data yang ditampilkan berasal dari perhitungan:

Total biaya variabel =

$$\text{total biaya bahan baku} + \text{total biaya } \textit{overhead} \textit{ variabel} \dots\dots\dots(4.30)$$

Tabel 4.97 Pengujian *Textbox* Hasil Total Biaya Variabel

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

➤ *Textbox* hasil biaya tenaga kerja langsung

Gambar 4.146 *Textbox* Hasil Biaya Tenaga Kerja Langsung Operator

Penjelasan: *user* tidak dapat melakukan *input* pada *textbox* hasil total biaya tenaga kerja karena *disable* atau data yang ditampilkan berasal dari perhitungan:

$$\text{biaya tenaga kerja} = \text{upah} \times \text{jumlah orang} \times \text{jumlah hari kerja} \dots\dots\dots(4.31)$$

Tabel 4.98 Pengujian *Textbox* Hasil Biaya Tenaga Kerja Langsung Operator

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

Gambar 4.147 *Textbox* Hasil Biaya Tenaga Kerja Langsung Teknisi

Penjelasan: *user* tidak dapat melakukan *input* pada *textbox* hasil total biaya tenaga kerja langsung karena *disable* atau data yang ditampilkan berasal dari perhitungan:

biaya tenaga kerja =

upah x jumlah orang x jumlah hari kerja(4.32)

Tabel 4.99 Pengujian *Textbox* Hasil Biaya Tenaga Kerja Langsung Teknisi

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

➤ *Textbox* hasil total biaya tenaga kerja langsung



Gambar 4.148 *Textbox* Hasil Total Biaya Tenaga Kerja Langsung

Penjelasan: *user* tidak dapat melakukan *input* pada *textbox* hasil total biaya tenaga kerja langsung karena *disable* atau data yang ditampilkan berasal dari perhitungan:

Total biaya tenaga kerja langsung =

Biaya tenaga kerja operator + biaya tenaga kerja teknisi(4.33)

Tabel 4.100 Pengujian *Textbox* Hasil Total Biaya Tenaga Kerja Langsung

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

- *Textbox* hasil biaya tenaga kerja tidak langsung

Gambar 4.149 *Textbox* Hasil Biaya Tenaga Kerja Tidak Langsung Akuntansi

Penjelasan: *user* tidak dapat melakukan *input* pada *textbox* hasil total biaya tenaga kerja tidak langsung karena *disable* atau data yang ditampilkan berasal dari perhitungan:

Biaya tenaga kerja =

upah x jumlah orang x jumlah hari kerja(4.34)

Tabel 4.101 Pengujian *Textbox* Hasil Biaya Tenaga Kerja Tidak Langsung Akuntansi

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

Gambar 4.150 *Textbox* Hasil Biaya Tenaga Kerja Tidak Langsung Administrasi

Penjelasan: *user* tidak dapat melakukan *input* pada *textbox* hasil total biaya tenaga kerja tidak langsung karena *disable* atau data yang ditampilkan berasal dari perhitungan:

Biaya tenaga kerja =

upah x jumlah orang x jumlah hari kerja(4.35)

Tabel 4.102 Pengujian *Textbox* Hasil Biaya Tenaga Kerja Tidak Langsung Administrasi

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

The screenshot shows a blue-themed form with the following fields and values:

- Gaji Tenaga Tidak Langsung: Engineering (dropdown menu)
- Upah: 70.000
- Jumlah: 1 Orang
- Jumlah Kerja: 26 /Hari
- Total Biaya: Rp. 1.820.000

Gambar 4.151 *Textbox* Hasil Biaya Tenaga Kerja Tidak Langsung *Engineering*

Penjelasan: *user* tidak dapat melakukan *input* pada *textbox* hasil total biaya tenaga kerja tidak langsung karena *disable* atau data yang ditampilkan berasal dari perhitungan:

Biaya tenaga kerja =

upah x jumlah orang x jumlah hari kerja(4.36)

Tabel 4.103 Pengujian *Textbox* Hasil Biaya Tenaga Kerja Tidak Langsung *Engineering*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

- *Textbox* hasil total biaya tenaga kerja tidak langsung



Gambar 4.152 *Textbox* Hasil Total Biaya Tenaga Kerja Tidak Langsung

Penjelasan: *user* tidak dapat melakukan *input* pada *textbox* hasil total biaya tenaga kerja langsung karena *disable* atau data yang ditampilkan berasal dari perhitungan:

Total biaya tenaga kerja tidak langsung =

biaya tenaga kerja administrasi + biaya tenaga kerja akuntansi + biaya tenaga kerja *engineering*.....(4.37)

Tabel 4.104 Pengujian *Textbox* Hasil Total Biaya Tenaga Kerja Tidak Langsung

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

- *Textbox* hasil total biaya listrik



Gambar 4.153 *Textbox* Hasil Total Biaya Listrik

Penjelasan: *user* tidak dapat melakukan *input* pada *textbox* hasil total biaya listrik karena *disable* atau data yang ditampilkan berasal dari perhitungan:

Total biaya listrik =

akumulasi penjumlahan dari beberapa biaya listrik mesin.....(4.38)

Tabel 4.105 Pengujian *Textbox* Hasil Total Biaya Listrik

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

- *Textbox* hasil total biaya *overhead* tetap


Gambar 4.154 *Textbox* Hasil Total Biaya *Overhead* Tetap

Penjelasan: *user* tidak dapat melakukan *input* pada *textbox* hasil total biaya *overhead* tetap karena *disable* atau data yang ditampilkan berasal dari perhitungan:

Total biaya *overhead* tetap =
akumulasi penjumlahan dari beberapa biaya *overhead* tetap.....(4.39)

Tabel 4.106 Pengujian *Textbox* Hasil Total Biaya *Overhead* Tetap

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

- *Textbox* hasil total biaya penyusutan


Gambar 4.155 *Textbox* Hasil Total Biaya Penyusutan

Penjelasan: *user* tidak dapat melakukan *input* pada *textbox* hasil total biaya penyusutan karena *disable* atau data yang ditampilkan berasal dari perhitungan:

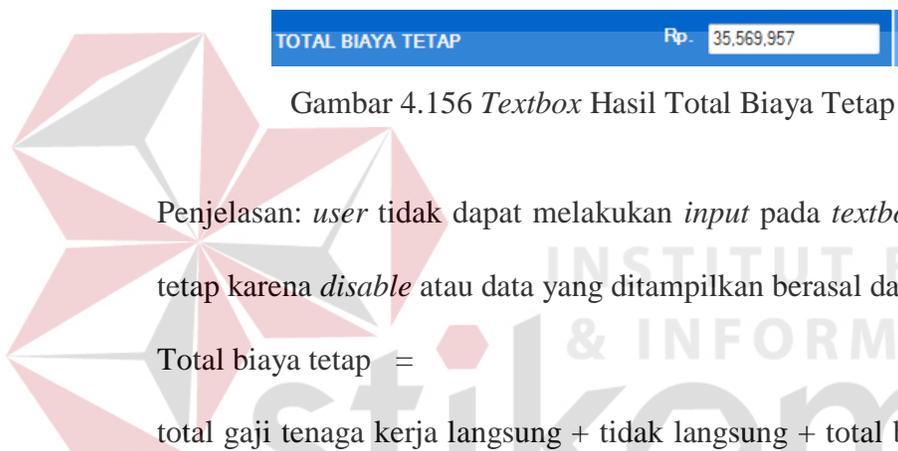
Total biaya penyusutan =

akumulasi penjumlahan dari beberapa biaya penyusutan.....(4.40)

Tabel 4.107 Pengujian *Textbox* Hasil Total Biaya Penyusutan

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

➤ *Textbox* hasil total biaya tetap



Gambar 4.156 *Textbox* Hasil Total Biaya Tetap

Penjelasan: *user* tidak dapat melakukan *input* pada *textbox* hasil total biaya tetap karena *disable* atau data yang ditampilkan berasal dari perhitungan:

Total biaya tetap =
total gaji tenaga kerja langsung + tidak langsung + total biaya listrik + total biaya *overhead* tetap + total biaya penyusutan.....(4.41)

Tabel 4.108 Pengujian *Textbox* Hasil Total Biaya Tetap

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

➤ *Textbox* hasil total biaya produksi



Gambar 4.157 *Textbox* Hasil Total Biaya Produksi

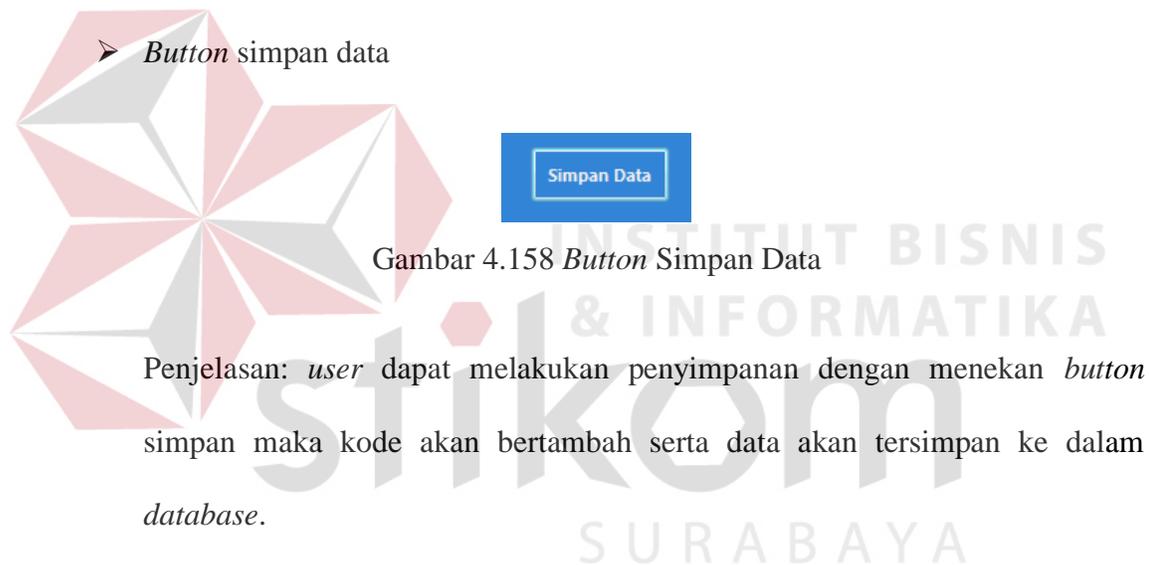
Penjelasan: *user* tidak dapat melakukan *input* pada *textbox* hasil total biaya produksi karena *disable* atau data yang ditampilkan berasal dari perhitungan:

Total biaya produksi =

total biaya variabel + total biaya tetap.....(4.42)

Tabel 4.109 Pengujian *Textbox* Hasil Total Biaya Produksi

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan



Gambar 4.158 *Button* Simpan Data

Penjelasan: *user* dapat melakukan penyimpanan dengan menekan *button* simpan maka kode akan bertambah serta data akan tersimpan ke dalam *database*.

Tabel 4.110 Pengujian *Button* Simpan Data

No.	Pengujian	Hasil
1.	Melakukan penyimpanan jika ada data yang kosong	Tidak dapat dilakukan

Langkah selanjutnya yaitu melakukan *test* data. *Test* data diperlukan untuk mengetahui apakah fungsi pada aplikasi yang telah dibuat ini sudah berjalan dengan baik dan sesuai dengan tujuan atau *output* yang diharapkan. Sebagai perbandingan, perlu dilakukan uji coba.

Diketahui:

- *Textbox* total pendapatan: Rp 100.000.000
- *Textbox* biaya bahan baku : Rp 30.000.000
- *Textbox* biaya *overhead* variabel: Rp 2,750,000
- *Textbox* total biaya variabel: Rp 32,750,000
- *Textbox* total biaya tenaga kerja langsung : Rp 11,700,000
- *Textbox* total biaya tenaga kerja tidak langsung: Rp 7,280,000
- *Textbox* total biaya listrik: Rp 1,874,957
- *Textbox* total biaya *overhead* tetap: Rp 5,150,000
- *Textbox* total biaya penyusutan: Rp 9.565.000
- *Textbox* total biaya tetap: Rp 35,569,957
- *Textbox* total biaya produksi : Rp 68,319,957

Dari contoh data perhitungan biaya produksi di atas, lalu tahap selanjutnya yaitu menguji perhitungan biaya produksi. Hasil dari pengujian akan dibandingkan dengan rancangan desain uji coba perhitungan biaya produksi yang sebelumnya telah dibuat. Seperti pada tabel 4.111, 4.112 dan 4.113.

Tabel 4.111 Hasil Uji Coba *Form* Perhitungan Biaya Produksi

<i>Test Case ID</i>	<i>Test Case</i>	<i>Test Data</i>	<i>Output yang diharapkan</i>	<i>Actual Output</i>	<i>Kesimpulan</i>
1	Menampilkan hasil total pendapatan	Setelah memilih <i>combobox</i>	Harga jual x volume produksi $10.000 \times 10.000 = \text{Rp } 100.000.000$		Sesuai
2	Menampilkan hasil total biaya bahan baku	Setelah memilih <i>combobox</i> dan memasukkan jumlah unit	Biaya bahan baku x unit $3.000 \times 10.000 = \text{Rp } 30.000.000$		Sesuai
3	Menampilkan hasil total biaya <i>overhead</i> variabel	Setelah memilih <i>combobox</i> dan memasukkan jumlah unit	Akumulasi dari biaya <i>overhead</i> variabel $1,250,000 + 700,000 + 800,000 = 2,750,000$		Sesuai
4	Menampilkan hasil total biaya variabel	Setelah semua <i>textbox</i> terisi dan klik enter	Biaya bahan baku + biaya <i>overhead</i> variabel $30.000.000 + 2.750.000 = \text{Rp } 32,750,000$		Sesuai

Tabel 4.112 Hasil Uji Coba *Form* Perhitungan Biaya Produksi (Lanjutan)

5	Menampilkan hasil total biaya tenaga kerja langsung	Setelah terisi <i>textbox</i> dan klik enter	Akumulasi dari biaya tenaga kerja langsung $50.000 \times 9 \text{ orang} \times 26 \text{ hari}$ $= 11,700,000$		Sesuai
6	Menampilkan hasil total biaya tenaga kerja tidak langsung	Setelah terisi <i>textbox</i> dan klik enter	Akumulasi dari biaya tenaga kerja tidak langsung $70.000 \times 4 \text{ orang} \times 26 \text{ hari}$ $= 7,280,000$		Sesuai
7	Menampilkan hasil total biaya listrik	Setelah terisi <i>textbox</i> dan klik enter	Akumulasi dari biaya listrik $1,025,752 + 439,608 +$ $263,765 + 145,832 =$ $1,874,957$		Sesuai
8	Menampilkan hasil total biaya <i>overhead</i> tetap	Setelah terisi <i>textbox</i> dan klik enter	Akumulasi dari biaya <i>overhead</i> tetap $1,750,000 + 500,000 +$ $300,000 + 2,500,000 +$ $100,000 = 5,150,000$		Sesuai

Tabel 4.113 Hasil Uji Coba *Form* Perhitungan Biaya produksi (Lanjutan)

9	Menampilkan hasil total biaya penyusutan	Setelah terisi <i>textbox</i> dan klik enter	Akumulasi dari biaya penyusutan $3,250,000 + 2,250,000 + 1,250,000 + 565,000 + 2,250,000 = \text{Rp } 9.565.000$		Sesuai
10	Menampilkan hasil total biaya tetap	Setelah terisi <i>textbox</i> dan klik enter	Biaya tenaga kerja langsung + biaya tenaga tidak langsung + biaya listrik + biaya <i>overhead</i> tetap + biaya penyusutan $11,700,000 + 7,280,000 + 1,874.957 + 5,150,000 + 9.565.000 = \text{Rp } 35,569,957$		Sesuai
11	Menampilkan hasil total biaya produksi	Setelah terisi <i>textbox</i> dan klik enter	Penjumlahan biaya variabel + biaya tetap $32.750.000 + 35,569,957 = \text{Rp } 68.319.957$		Sesuai

4.4.9 Uji Coba *Form* Hasil Analisa Perhitungan Laba Rugi

Form perhitungan laba rugi digunakan sebagai mengetahui estimasi keuangan setelah melakukan investasi. Data perhitungan laba rugi diperoleh dari perhitungan pendapatan, biaya variabel, biaya tetap, kemudian laba sebelum pajak, pajak dan laba setelah pajak. Di *form* ini terdapat:

➤ *Combobox* Kode Pendapatan



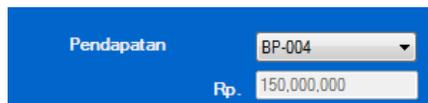
Gambar 4.159 *Combobox* Kode Pendapatan BP-001



Gambar 4.160 *Combobox* Kode Pendapatan BP-002



Gambar 4.161 *Combobox* Kode Pendapatan BP-003



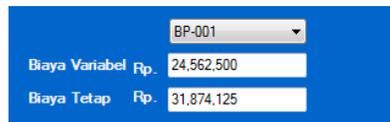
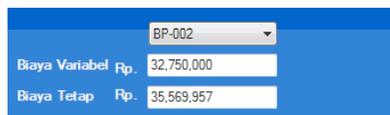
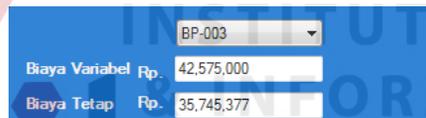
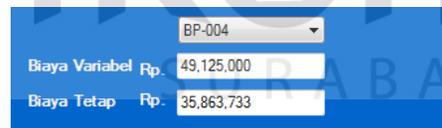
Gambar 4.162 *Combobox* Kode Pendapatan BP-004

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* kode pendapatan, data ditampilkan berasal dari *database* biaya produksi.

Tabel 4.114 Pengujian *Combobox* Kode Pendapatan

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

➤ *Combobox* kode biaya produksi


Gambar 4.163 *Combobox* Kode Biaya Produksi BP-001

Gambar 4.164 *Combobox* Kode Biaya Produksi BP-002

Gambar 4.165 *Combobox* Kode Biaya Produksi BP-003

Gambar 4.166 *Combobox* Kode Biaya Produksi BP-004

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* kode mesin, data ditampilkan berasal dari *database* mesin.

Tabel 4.115 Pengujian *Combobox* Kode Biaya Produksi

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

➤ *Textbox* hasil laba setelah pajak

Gambar 4.167 *Textbox* Hasil Laba Setelah Pajak Periode 1

Gambar 4.168 *Textbox* Hasil Laba Setelah Pajak Periode 2

Gambar 4.169 *Textbox* Hasil Laba Setelah Pajak Periode 3

Gambar 4.170 *Textbox* Hasil Laba Setelah Pajak Periode 4

Penjelasan: *User* tidak dapat melakukan *input* pada *textbox* hasil laba setelah pajak karena data ditampilkan berasal dari perhitungan:

Laba setelah pajak =

$$(\text{pendapatan} - \text{biaya total}) \times 30\% \dots\dots\dots(4.43)$$

Tabel 4.116 Pengujian *Textbox* Hasil Laba Setelah Pajak

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

➤ *Button* simpan data



Gambar 4.171 *Button* Simpan Data

Penjelasan: *User* dapat melakukan penyimpanan dengan menekan *button* simpan maka kode akan bertambah serta data akan tersimpan ke dalam *database*.

Tabel 4.117 Pengujian *Button* Simpan Data

No.	Pengujian	Hasil
1.	Melakukan penyimpanan jika ada data yang kosong	Tidak dapat dilakukan

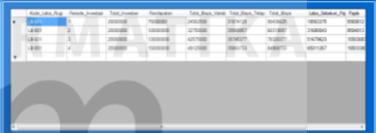
Langkah selanjutnya yaitu melakukan *test* data. *Test* data diperlukan untuk mengetahui apakah fungsi pada aplikasi yang telah dibuat ini sudah berjalan dengan baik dan sesuai dengan tujuan atau *output* yang diharapkan. Sebagai perbandingan, perlu dilakukan uji coba.

Diketahui:

- BP-002
- Pendapatan : Rp 100.000.000
- Biaya Variabel : Rp 32.750.000
- Biaya Tetap : Rp 35.569.957
- Pajak : 30%

Dari contoh data perhitungan laba rugi di atas, lalu tahap selanjutnya yaitu menguji *form* perhitungan laba rugi. Seperti pada tabel 4.118.

Tabel 4.118 Hasil Uji Coba Perhitungan Laba Rugi

<i>Test Case ID</i>	<i>Test Case</i>	<i>Test Data</i>	<i>Output yang diharapkan</i>	<i>Actual Output</i>	<i>Kesimpulan</i>
1	Menampilkan nilai hasil laba setelah pajak periode 2	<ul style="list-style-type: none"> • Memilih kode biaya produksi : BP-002 • Kemudian klik enter • Inputkan nilai pajak 	Laba setelah pajak = (pendapatan – (biaya variabel + biaya tetap)) - pajak $(100.000.000 - (32.750.000 + 35.569.957)) - 30\% = \text{Rp } 22.176.030$		Sesuai
2	Menyimpan data Laba rugi	Menekan <i>button</i> simpan data	Tersimpan ke <i>database</i> Laba rugi dan tampil di <i>datagridview</i>		Sesuai

4.4.10 Uji Coba *Form* Hasil Analisa Perhitungan Aliran Kas Bersih

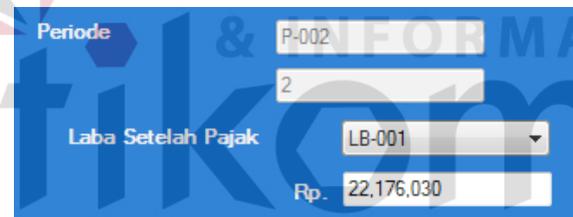
Form perhitungan aliran kas bersih digunakan sebagai mengetahui estimasi penerimaan-penerimaan atau pengeluaran-pengeluaran pendapatan setelah melakukan investasi. Data perhitungan aliran kas bersih diperoleh dari perhitungan laba setelah pajak perperiode dengan penyusutan. Di *form* ini terdapat:

➤ *Textbox* hasil laba setelah pajak



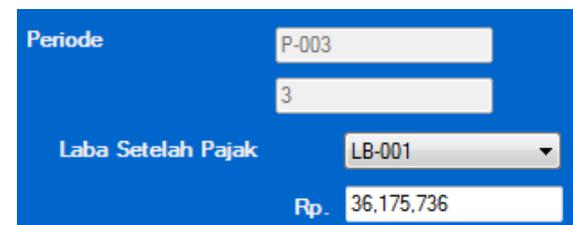
Periode	P-001
	1
Laba Setelah Pajak	LB-001
Rp.	12,994,363

Gambar 4.172 *Textbox* Hasil Laba Setelah Pajak Periode 1



Periode	P-002
	2
Laba Setelah Pajak	LB-001
Rp.	22,176,030

Gambar 4.173 *Textbox* Hasil Laba Setelah Pajak Periode 2



Periode	P-003
	3
Laba Setelah Pajak	LB-001
Rp.	36,175,736

Gambar 4.174 *Textbox* Hasil Laba Setelah Pajak Periode 3

Gambar 4.175 *Textbox* Hasil Laba Setelah Pajak Periode 4

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* kode laba rugi, data ditampilkan berasal dari *database* laba rugi

Tabel 4.119 Pengujian *Textbox* Hasil Laba Setelah Pajak

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

➤ *Textbox* hasil aliran kas bersih perperiode

Gambar 4.176 *Textbox* Hasil Aliran Kas Bersih Periode 1

Gambar 4.177 *Textbox* Hasil Aliran Kas Bersih Periode 2

Gambar 4.178 *Textbox* Hasil Aliran Kas Bersih Periode 3

Gambar 4.179 *Textbox* Hasil Aliran Kas Bersih Periode 4

Penjelasan: *User* tidak dapat melakukan *input* pada *textbox* hasil laba setelah pajak karena data ditampilkan berasal dari perhitungan:

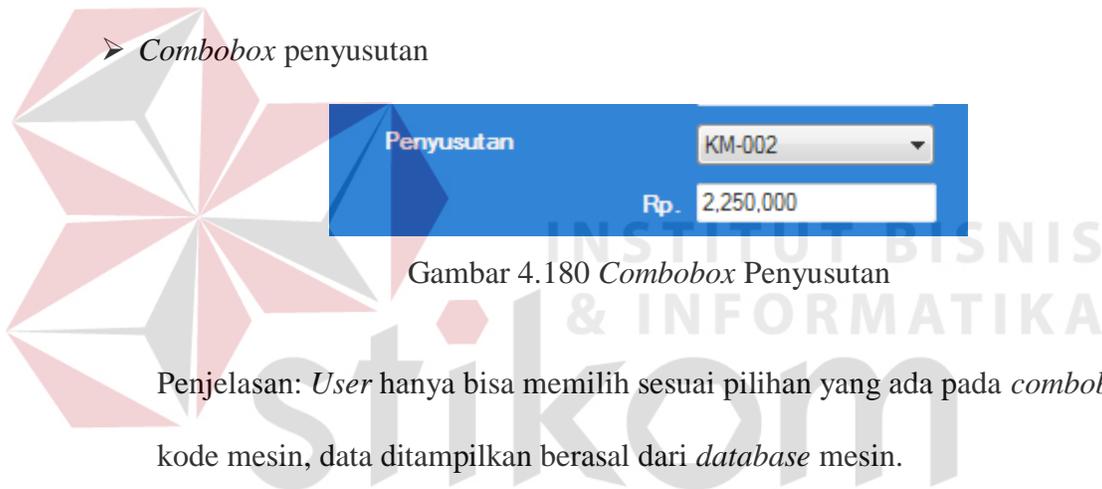
Aliran kas bersih =

laba setelah pajak perperiode + penyusutan mesin(4.44)

Tabel 4.120 Pengujian *Textbox* Hasil Aliran Kas Bersih

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

➤ *Combobox* penyusutan



Gambar 4.180 *Combobox* Penyusutan

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* kode mesin, data ditampilkan berasal dari *database* mesin.

Tabel 4.121 Pengujian *Combobox* Kode Penyusutan

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

➤ *Button* simpan & hitung



Gambar 4.181 *Button* Simpan & Hitung

Penjelasan: *User* dapat melakukan penyimpanan dengan menekan *button* simpan dan menghitung hasil aliran kas bersih dengan menekan *button* hitung.

Tabel 4.122 Pengujian *Button* Simpan & Hitung

No.	Pengujian	Hasil
1.	Melakukan penyimpanan jika ada data yang kosong	Tidak dapat dilakukan
2.	Melakukan proses hitung hasil jika ada data yang kosong	Tidak dapat dilakukan

Langkah selanjutnya yaitu melakukan *test* data. *Test* data diperlukan untuk mengetahui apakah fungsi pada aplikasi yang telah dibuat ini sudah berjalan dengan baik dan sesuai dengan tujuan atau *output* yang diharapkan. Sebagai perbandingan, perlu dilakukan uji coba.

Diketahui:

- Laba setelah pajak periode 1 : 12.994.363
- Laba setelah pajak periode 2 : 22.176.030
- Laba setelah pajak periode 3 : 36.175.736
- Laba setelah pajak periode 4 : 45.507.887
- Penyusutan : 2.250.000

Dari contoh data perhitungan aliran kas bersih di atas, lalu tahap selanjutnya yaitu menguji *form* perhitungan aliran kas bersih. Hasil dari pengujian akan dibandingkan dengan rancangan desain uji coba perhitungan aliran kas bersih yang sebelumnya telah dibuat. Seperti pada tabel 4.123 dan 4.124.

Tabel 4.123 Hasil Uji Coba Perhitungan Aliran Kas Bersih

Test Case ID	Test Case	Test Data	Output yang diharapkan	Actual Output	Kesimpulan
1	Menampilkan nilai hasil aliran kas bersih periode 1	<ul style="list-style-type: none"> • Pastikan <i>textbox</i> periode : P-001 • Memilih kode laba rugi : LB-001 • Memilih kode mesin : KM-002 • Kemudian klik enter 	Aliran kas bersih = laba setelah pajak + penyusutan $12.994.363 + 2.250.000 =$ Rp 15.244.363		Sesuai
2	Menampilkan nilai hasil aliran kas bersih periode 2	<ul style="list-style-type: none"> • Pastikan <i>textbox</i> periode : P-002 • Memilih kode laba rugi : LB-001 • Memilih kode mesin : KM-002 • Kemudian klik enter 	Aliran kas bersih = laba setelah pajak + penyusutan $22.176.030 + 2.250.000 =$ Rp 24.426.030		Sesuai

Tabel 4.124 Hasil Uji Coba Perhitungan Aliran Kas Bersih (Lanjutan)

3	Menampilkan nilai hasil aliran kas bersih periode 3	<ul style="list-style-type: none"> • Pastikan <i>textbox</i> periode : P-003 • Memilih kode laba rugi : LB-001 • Memilih kode mesin : KM-002 • Kemudian klik enter 	Aliran kas bersih = laba setelah pajak + penyusutan $36.175.736 + 2.250.000 =$ Rp 38.425.736		Sesuai
4	Menampilkan nilai hasil aliran kas bersih periode 4	<ul style="list-style-type: none"> • Pastikan <i>textbox</i> periode : P-004 • Memilih kode laba rugi : LB-001 • Memilih kode mesin : KM-002 • Kemudian klik enter 	Aliran kas bersih = laba setelah pajak + penyusutan $45.507.887 + 2.250.000 =$ Rp 47.757.887		Sesuai
5	Menyimpan data perhitungan data aliran kas bersih	Menekan <i>button</i> simpan data	Tersimpan ke <i>database</i> aliran kas bersih dan tampil di <i>datagridview</i>		Sesuai

4.4.11 Uji Coba *Form* Hasil Analisa Perhitungan *Average Rate of Return (ARR)*

Form perhitungan *average rate of return* digunakan untuk mengetahui tingkat keuntungan rata-rata investasi sebagai dasar proses kelayakan investasi. Data perhitungan *average rate of return* diperoleh dari perhitungan laba rugi yaitu rata-rata laba setelah pajak dan total investasi. Di *form* ini terdapat:

➤ *Combobox* kode laba rugi



Gambar 4.182 *Combobox* Kode Laba Rugi

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* kode laba rugi, data ditampilkan berasal dari perhitungan rata rata laba setelah pajak perperiode dari *database* laba rugi.

Tabel 4.125 Pengujian *Combobox* Kode Laba Rugi

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

➤ *Combobox* kode mesin



Gambar 4.183 *Combobox* Kode Mesin

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* kode mesin, data ditampilkan berasal dari *database* mesin.

Tabel 4.126 Pengujian *Combobox* Kode Mesin

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

➤ *Textbox* hasil *average rate of return*

Gambar 4.184 *Textbox* Hasil *Average Rate of Rreturn*

Penjelasan: *User* tidak dapat melakukan *input* pada *textbox* hasil *average*

rate of return karena data ditampilkan berasal dari perhitungan:

average rate of return =

rata-rata laba setelah pajak / total investasi(4.45)

Tabel 4.127 Pengujian *Textbox* Hasil *Average Rate of Return*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

➤ *Button* simpan & hitung

Gambar 4.185 *Button* Simpan & Hitung

Penjelasan: *User* dapat melakukan penyimpanan dengan menekan *button* simpan dan menghitung hasil *average rate of return* dengan menekan *button* hitung.

Tabel 4.128 Pengujian *Button* Simpan & Hitung

No.	Pengujian	Hasil
1.	Melakukan penyimpanan jika ada data yang kosong	Tidak dapat dilakukan
2.	Melakukan proses hitung hasil jika ada data yang kosong	Tidak dapat dilakukan

Langkah selanjutnya yaitu melakukan *test* data. *Test* data diperlukan untuk mengetahui apakah fungsi pada aplikasi yang telah dibuat ini sudah berjalan dengan baik dan sesuai dengan tujuan atau *output* yang diharapkan.

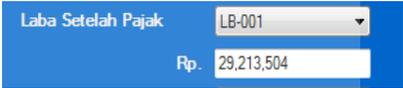
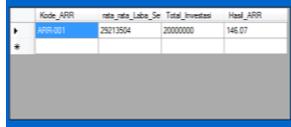
Sebagai perbandingan, perlu dilakukan uji coba.

Diketahui:

- Kode Laba rugi: LB-001
- *Textbox* rata-rata laba setelah pajak : Rp 29.213.504
- Kode mesin: KM-002
- *Textbox* investasi: Rp 20.000.000
- Hasil *average rate of retron* : 146,07 %

Dari contoh data perhitungan *average rate of retron* di atas, lalu tahap selanjutnya yaitu menguji *form* perhitungan *average rate of retron*. Hasil dari pengujian akan dibandingkan dengan rancangan desain uji coba perhitungan *average rate of retron* yang sebelumnya telah dibuat. Seperti pada tabel 4.129.

Tabel 4.129 Hasil Uji Coba Perhitungan *Average Rate of Return*

<i>Test Case ID</i>	<i>Test Case</i>	<i>Test Data</i>	<i>Output yang diharapkan</i>	<i>Actual Output</i>	<i>Kesimpulan</i>
1	Menampilkan nilai rata-rata laba setelah pajak	Memilih kode laba rugi: LB-001	Jumlah laba setelah pajak perperiode / jumlah periode $(12.994.363 + 22.176.030 + 36.175.736 + 45.507.887) / 4 = 29.213.504$		Sesuai
2	Menampilkan nilai total investasi	Memilih kode mesin: KM-002	Rp 20.000.000		Sesuai
3	Menampilkan hasil perhitungan <i>average rate of return</i>	Setelah terisi <i>textbox</i> laba setelah pajak dan total investasi dan menekan <i>button</i> hitung atau klik enter	$ARR = \text{rata rata laba setelah pajak} / \text{investasi} \times 100 \%$ $29.213.504 / 20.000.000 = 146,07 \%$		Sesuai
4	Menyimpan data perhitungan data <i>average rate of return</i>	Menekan <i>button</i> simpan data	Tersimpan ke <i>database</i> <i>Average rate of return</i> dan tampil di <i>datagridview</i>		Sesuai

4.4.12 Uji Coba *Form Hasil Analisa Perhitungan Payback Period (PP)*

Form perhitungan *payback period* digunakan untuk mengetahui seberapa cepat investasi itu bisa kembali dan sebagai dasar dari proses kelayakan investasi. Data perhitungan *payback period* diperoleh dari perhitungan akumulasi aliran kas bersih dan total investasi. Di *form* ini terdapat:

➤ *Combobox* kode aliran kas bersih



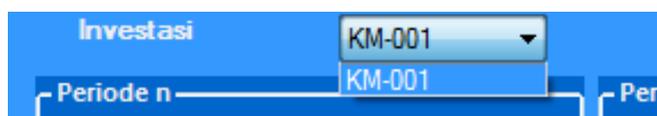
Gambar 4.186 *Combobox* Kode Aliran Kas Bersih

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* kode aliran kas bersih, data ditampilkan berasal dari *database* aliran kas bersih.

Tabel 4.130 Pengujian *Combobox* Kode Aliran Kas Bersih

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

➤ *Combobox* kode mesin



Gambar 4.187 *Combobox* Kode Mesin

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* kode mesin, data ditampilkan berasal dari *database* mesin.

Tabel 4.131 Pengujian *Combobox* Kode Mesin

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

➤ *Textbox* hasil *payback period*

Gambar 4.188 *Textbox* Hasil *Payback Period*

Penjelasan: *User* tidak dapat melakukan *input* pada *textbox* hasil *payback period* karena data ditampilkan berasal dari perhitungan:

Payback period =

$$n + ((a-b)/(c-b)) \times 1 \text{ tahun} \dots \dots \dots (4.46)$$

Dimana :

n: Tahun terakhir dimana jumlah arus kas masih belum bisa menutup investasi mula-mula.

a : Jumlah investasi mula-mula

b: Jumlah kumulatif arus kas pada tahun ke – n

c: Jumlah kumulatif arus kas pada tahun ke n + 1

Tabel 4.132 Pengujian *Textbox* Hasil *Payback Period*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

➤ *Button* Simpan & Hitung

Gambar 4.189 *Button* Simpan & Hitung

Penjelasan: *User* dapat melakukan penyimpanan dengan menekan *button* simpan dan menghitung hasil *payback period* dengan menekan *button* hitung.

Tabel 4.133 Pengujian *Button* Simpan & Hitung

No.	Pengujian	Hasil
1.	Melakukan penyimpanan jika ada data yang kosong	Tidak dapat dilakukan
2.	Melakukan proses hitung hasil jika ada data yang kosong	Tidak dapat dilakukan

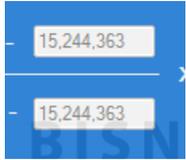
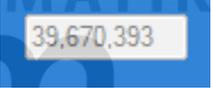
Langkah selanjutnya yaitu melakukan *test data*. *Test data* diperlukan untuk mengetahui apakah fungsi pada aplikasi yang telah dibuat ini sudah berjalan dengan baik dan sesuai dengan tujuan atau *output* yang diharapkan. Sebagai perbandingan, perlu dilakukan uji coba.

Diketahui:

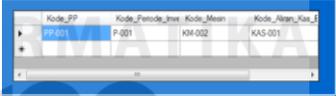
- Kode periode n: P-001
- Kode aliran kas bersih : KAS-001
- *Textbox* aliran kas perperiode : Rp 15.244.363
- Kode periode n + 1: P-002
- Kode aliran kas bersih : KAS-001
- *Textbox* aliran kas perperiode : Rp 39.670.393
- Kode mesin: KM-002
- *Textbox* investasi: Rp 20.000.000
- Hasil *payback period* : 1.19 tahun

Dari contoh data perhitungan *payback period* di atas, lalu tahap selanjutnya yaitu menguji *form* perhitungan *payback period*. Hasil dari pengujian akan dibandingkan dengan rancangan desain uji coba. Seperti pada tabel 4.134 dan 4.135.

Tabel 4.134 Hasil Uji Coba Perhitungan *Payback Period*

<i>Test Case ID</i>	<i>Test Case</i>	<i>Test Data</i>	<i>Output yang diharapkan</i>	<i>Actual Output</i>	Kesimpulan
1	Menampilkan nilai aliran kas bersih perperiode	Memilih kode aliran kas bersih: KAS-001 & kode periode : P-001	Rp 15.244.363		Sesuai
2	Menampilkan nilai aliran kas bersih perperiode n+1	Memilih kode aliran kas bersih: KAS-001 & kode periode : P-002	Rp 39.670.393		Sesuai
3	Menampilkan nilai total investasi	Memilih kode mesin: KM-002	Rp 20.000.000		Sesuai

Tabel 4.135 Hasil Uji Coba Perhitungan *Payback Period* (Lanjutan)

4	Menampilkan hasil perhitungan <i>payback period</i>	Setelah terisi <i>textbox</i> aliran kas perperiode n, aliran kas perperiode n +1 dan total investasi dan menekan <i>button</i> hitung	$n + \frac{(a-b)/(c-b)}{1 + ((20.000.000 - 15.244.363)/(39.670.393 - 15.244.363))} \times 1 \text{ tahun} = 1,19 \text{ Tahun}$		Sesuai
5	Menyimpan data <i>payback period</i>	Menekan <i>button</i> simpan data	Tersimpan ke <i>database payback period</i> dan tampil di <i>datagridview</i>		Sesuai

4.4.13 Uji Coba *Form Hasil Analisa Perhitungan Net Present Value (NPV)*

Form perhitungan *net present value* digunakan untuk mengetahui selisih penerimaan penerimaan kas bersih di masa yang akan datang dengan investasi dan sebagai dasar dari proses kelayakan investasi. Data perhitungan *net present value* diperoleh dari perhitungan aliran kas bersih perperiode dan total investasi. Di *form* ini terdapat:

➤ *Combobox* kode aliran kas bersih

Gambar 4.190 *Combobox* Kode Aliran Kas Bersih Periode 1

Gambar 4.191 *Combobox* Kode Aliran Kas Bersih Periode 2

Gambar 4.192 *Combobox* Kode Aliran Kas Bersih Periode 3

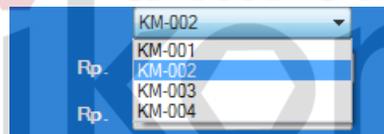

Gambar 4.193 *Combobox* Kode Aliran Kas Bersih Periode 4

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* kode aliran kas bersih, data ditampilkan berasal dari *database* aliran kas bersih.

Tabel 4.136 Pengujian *Combobox* Kode Aliran Kas Bersih

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

➤ *Combobox* kode mesin


Gambar 4.194 *Combobox* Kode Mesin

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* kode mesin, data ditampilkan berasal dari *database* mesin.

Tabel 4.137 Pengujian *Combobox* Kode Mesin

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

➤ *Textbox* hasil *PV* kas masuk

Gambar 4.195 *Textbox* Hasil *PV* Kas Masuk Periode 1

Gambar 4.196 *Textbox* Hasil *PV* Kas Masuk Periode 2

Gambar 4.197 *Textbox* Hasil *PV* Kas Masuk Periode 3

Gambar 4.198 *Textbox* Hasil *PV* Kas Masuk Periode 4

Penjelasan: *User* tidak dapat melakukan *input* pada *textbox* hasil *PV* kas masuk karena data ditampilkan berasal dari perhitungan:

$$PV \text{ kas masuk} = \text{aliran kas perperiode} \times \text{discount factor} \dots\dots\dots(4.47)$$

Tabel 4.138 Pengujian *Textbox* Hasil *PV* Kas Masuk

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

➤ *Textbox* hasil *net present value*

Gambar 4.199 *Textbox* Hasil *Net Present Value*

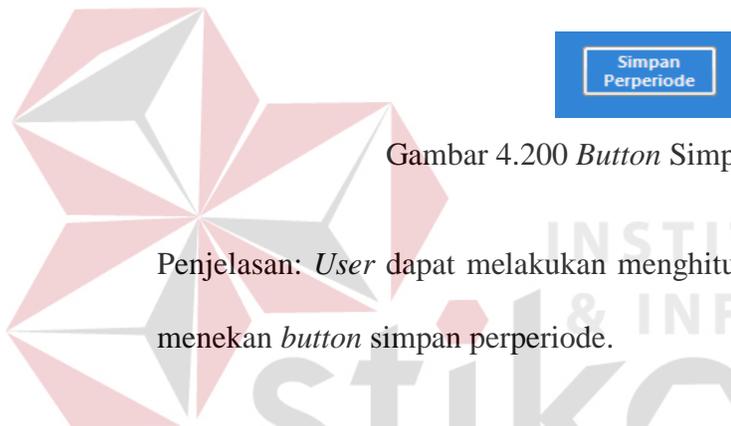
Penjelasan: *User* tidak dapat melakukan *input* pada *textbox* hasil *net present value* karena data ditampilkan berasal dari perhitungan:

$$NPV = \text{akumulasi } PV \text{ kas masuk} - \text{total investasi} \dots\dots\dots(4.48)$$

Tabel 4.139 Pengujian *Textbox* Hasil *Net Present Value*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

➤ *Button* simpan perperiode



Gambar 4.200 *Button* Simpan Perperiode

Penjelasan: *User* dapat melakukan menghitung hasil *PV* kas masuk dengan menekan *button* simpan perperiode.

Tabel 4.140 Pengujian *Button* Simpan Perperiode

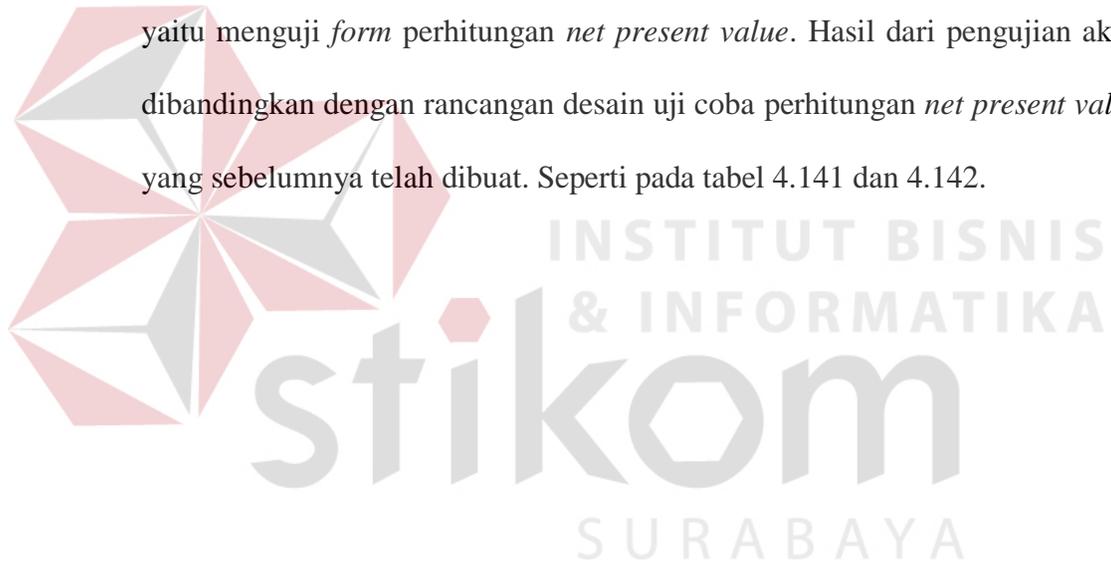
No.	Pengujian	Hasil
1.	Melakukan penyimpanan jika ada data yang kosong	Tidak dapat dilakukan

Langkah selanjutnya yaitu melakukan *test* data. *Test* data diperlukan untuk mengetahui apakah fungsi pada aplikasi yang telah dibuat ini sudah berjalan dengan baik dan sesuai dengan tujuan atau *output* yang diharapkan. Sebagai perbandingan, perlu dilakukan uji coba.

Diketahui:

- Kode periode : P-001
- Kode aliran kas bersih : KAS-001
- *Textbox* aliran kas perperiode : Rp 15.244.363
- Kode mesin: KM-002
- *Textbox* investasi: Rp 20.000.000
- Hasil *PV* kas masuk : Rp 12.911.975
- Hasil *NPV* : Rp 58.494.208

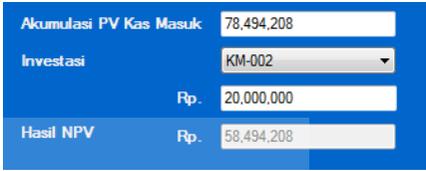
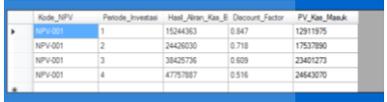
Dari contoh data perhitungan *net present value* di atas, lalu tahap selanjutnya yaitu menguji *form* perhitungan *net present value*. Hasil dari pengujian akan dibandingkan dengan rancangan desain uji coba perhitungan *net present value* yang sebelumnya telah dibuat. Seperti pada tabel 4.141 dan 4.142.

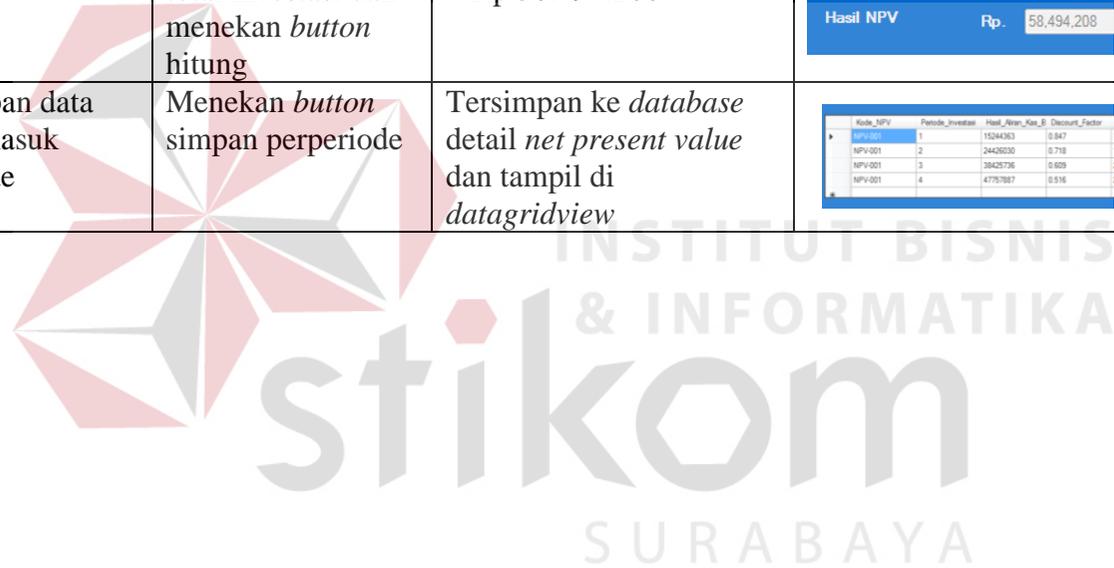


Tabel 4.141 Hasil Uji Coba Perhitungan *Net Present Value*

<i>Test Case ID</i>	<i>Test Case</i>	<i>Test Data</i>	<i>Output yang diharapkan</i>	<i>Actual Output</i>	<i>Kesimpulan</i>
1	Menampilkan nilai aliran kas bersih perperiode	Memilih kode aliran kas bersih: KAS-001 & kode periode : P-001	Rp 15.244.363		Sesuai
2	Menampilkan nilai total investasi	Memilih kode mesin: KM-001	Rp 20.000.000		Sesuai
3	Menampilkan hasil PV kas masuk	Setelah terisi <i>textbox</i> aliran kas perperiode, <i>discount factor</i> dan menekan <i>button</i> hitung atau klik enter	Akumulasi <i>PV</i> kas masuk = aliran kas bersih perperiode x <i>discount factor</i> $15.244.363 \times 0.847 = \text{Rp } 12.911.975$		Sesuai

Tabel 4.142 Hasil Uji Coba Perhitungan *Net Present Value* (Lanjutan)

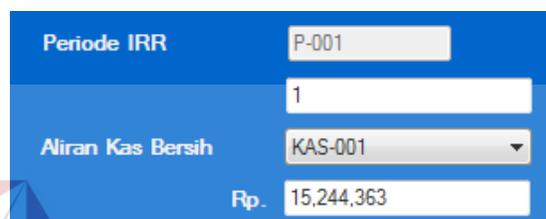
4	Menampilkan hasil <i>NPV</i>	Setelah terisi <i>textbox</i> akumulasi aliran kas bersih , total investasi dan menekan <i>button</i> hitung	$NPV = \text{akumulasi } PV \text{ kas masuk} - \text{investasi}$ 78.494.208 – 20.000.000 = Rp 58.494.208		Sesuai
5	Menyimpan data <i>PV</i> kas masuk perperiode	Menekan <i>button</i> simpan perperiode	Tersimpan ke <i>database</i> detail <i>net present value</i> dan tampil di <i>datagridview</i>		Sesuai



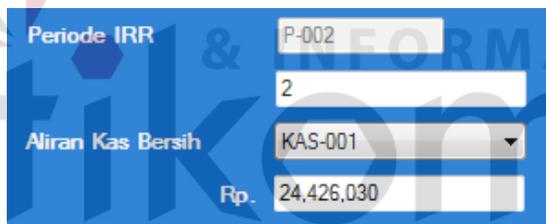
4.4.14 Uji Coba *Form Hasil Analisa Perhitungan Internal Rate of Return (IRR)*

Form perhitungan *internal rate of return* digunakan untuk mengetahui tingkat bunga yang menyamakan penerimaan kas dengan investasi dan sebagai dasar dari proses kelayakan investasi. Data perhitungan *internal rate of return* diperoleh dari perhitungan aliran kas bersih dan total investasi. Di *form* ini terdapat:

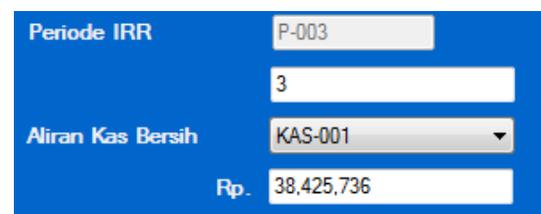
- *Combobox* kode aliran kas bersih



Gambar 4.201 *Combobox* Kode Aliran Kas Bersih Periode 1



Gambar 4.202 *Combobox* Kode Aliran Kas Bersih Periode 2



Gambar 4.203 *Combobox* Kode Aliran Kas Bersih Periode 3

Gambar 4.204 *Combobox* Kode Aliran Kas Bersih Periode 4

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* kode aliran kas bersih, data ditampilkan berasal dari *database* aliran kas bersih.

Tabel 4.143 Pengujian *Combobox* Kode Aliran Kas Bersih

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

➤ *Combobox* kode mesin

Gambar 4.205 *Combobox* Kode Mesin

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* kode mesin, data ditampilkan berasal dari *database* mesin.

Tabel 4.144 Pengujian *Combobox* Kode Mesin

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

➤ *Textbox* hasil *PV* kas masuk berlawanan



Gambar 4.206 *Textbox* Hasil *PV* Kas Masuk Berlawanan Periode 1



Gambar 4.207 *Textbox* Hasil *PV* Kas Masuk Berlawanan Periode 2



Gambar 4.208 *Textbox* Hasil *PV* Kas Masuk Berlawanan Periode 3



Gambar 4.209 *Textbox* Hasil *PV* Kas Masuk Berlawanan Periode 4

Penjelasan: *User* tidak dapat melakukan *input* pada *textbox* hasil *PV* kas masuk berlawanan karena data ditampilkan berasal dari perhitungan:

$$PV \text{ kas masuk berlawanan} = \text{aliran kas bersih perperiode} \times \text{discount factor} \dots \dots \dots (4.49)$$

Tabel 4.145 Pengujian *Textbox* Hasil *PV* Kas Masuk Berlawanan

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

➤ *Textbox* hasil *Internal rate of return*



Gambar 4.210 *Textbox* Hasil *Internal Rate of Return*

Penjelasan: *User* tidak dapat melakukan *input* pada *textbox* hasil *internal rate of return* karena data ditampilkan berasal dari perhitungan:

IRR =

$$rr + \frac{NPV_{rr}}{TPV_{rr} - TPV_{rt}} \times (rt - rr) \dots\dots\dots(4.50)$$

Dimana:

rr = tingkat *discount rate* (*r*) lebih rendah

rt = tingkat *discount rate* (*r*) lebih tinggi

TPV = Total Present Value

NPV = Net Present Value

Tabel 4.146 Pengujian *Textbox* Hasil *Internal Rate of Return*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

➤ *Button* simpan perperiode



Gambar 4.211 *Button* Simpan Perperiode

Penjelasan: *User* dapat melakukan menghitung hasil *PV* kas masuk berlawanan dengan menekan *button* simpan perperiode.

Tabel 4.147 Pengujian *Button* Simpan Perperiode

No.	Pengujian	Hasil
1.	Melakukan penyimpanan jika ada data yang kosong	Tidak dapat dilakukan

Langkah selanjutnya yaitu melakukan *test* data. *Test* data diperlukan untuk mengetahui apakah fungsi pada aplikasi yang telah dibuat ini sudah berjalan dengan baik dan sesuai dengan tujuan atau *output* yang diharapkan. Sebagai perbandingan, perlu dilakukan uji coba.

Diketahui:

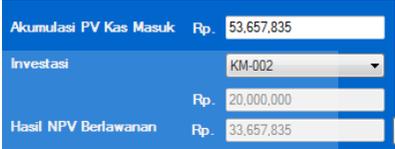
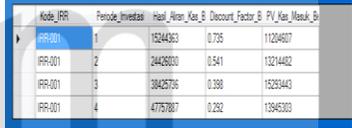
- Kode periode : P-001
- Kode aliran kas bersih : KAS-001
- *Textbox* aliran kas perperiode : Rp 15.244.363
- Kode mesin: KM-002
- *Textbox* investasi: Rp 20.000.000
- Hasil *PV* kas masuk berlawanan : Rp 11.204.607
- Hasil *NPV* berlawanan : Rp 53.657.835
- Hasil *IRR* : 60,39 %

Dari contoh data perhitungan *internal rate of return* di atas, lalu tahap selanjutnya yaitu menguji *form* perhitungan *internal rate of return*. Hasil dari pengujian akan dibandingkan dengan rancangan desain uji coba perhitungan *internal rate of return* yang sebelumnya telah dibuat. Seperti pada tabel 4.148 dan 4.149.

Tabel 4.148 Hasil Uji Coba Perhitungan *Internal Rate of Return*

<i>Test Case ID</i>	<i>Test Case</i>	<i>Test Data</i>	<i>Output yang diharapkan</i>	<i>Actual Output</i>	<i>Kesimpulan</i>
1	Menampilkan nilai aliran kas bersih perperiode	Memilih kode aliran kas bersih: KAS-001 & kode periode : P-001	Rp 15.244.363		Sesuai
2	Menampilkan nilai total investasi	Memilih kode mesin: KM-001	Rp 20.000.000		Sesuai
3	Menampilkan hasil PV kas masuk berlawanan	Setelah terisi <i>textbox</i> aliran kas perperiode n, <i>discount factor</i> berlawanan dan menekan <i>button</i> hitung	$PV \text{ kas masuk} = \text{aliran kas bersih} \times \text{discount factor}$ $15.244.363 \times 0.735 = \text{Rp } 11.204.607$		Sesuai

Tabel 4.149 Hasil Uji Coba Perhitungan *Internal Rate of Return* (Lanjutan)

4	Menampilkan hasil <i>NPV</i> berlawanan	Setelah terisi <i>textbox</i> akumulasi aliran kas bersih , total investasi dan menekan <i>button</i> hitung	Akumulasi <i>PV</i> kas masuk – investasi 53.365.835 – 20.000.000 = Rp 33.657.835		Sesuai
5	Menampilkan hasil <i>IRR</i>	Setelah semua <i>textbox</i> terisi	$IRR = rr + \left(\frac{npvrr}{(tpvrr - tpvrt)} \right) \times (rt - rr)$ $18\% \left(\frac{58494208}{(78494208 - 53657835)} \right) \times (36\% - 18\%) = 60,39\%$		Sesuai
6	Menyimpan data <i>PV</i> kas masuk perperiode	Menekan <i>button</i> simpan perperiode	Tersimpan ke <i>database</i> detail <i>internal rate of return</i> dan tampil di <i>datagridview</i>		Sesuai

4.4.15 Uji Coba *Form* Hasil Analisa Perhitungan *Profitability Index (PI)*

Form perhitungan *profitability index* digunakan untuk mengetahui perbandingan antara penerimaan kas bersih dengan investasi dan sebagai dasar dari proses kelayakan investasi. Data perhitungan *profitability index* diperoleh dari perhitungan *net present value* dan total investasi. Di *form* ini terdapat:

➤ *Combobox* kode *NPV*



Gambar 4.212 *Combobox* Kode *NPV*

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* kode *NPV*, data ditampilkan berasal dari *database net present value*.

Tabel 4.150 Pengujian *combobox* kode *NPV*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

➤ *Combobox* kode mesin



Gambar 4.213 *Combobox* Kode Mesin

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* kode mesin, data ditampilkan berasal dari *database* mesin.

Tabel 4.151 Pengujian *Combobox* Kode Mesin

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

➤ *Textbox* hasil *profitability index*

Gambar 4.214 *Textbox* Hasil *Profitability Index*

Penjelasan: *User* tidak dapat melakukan *input* pada *textbox* hasil *profitability*

index karena data ditampilkan berasal dari perhitungan:

$PI =$

akumulasi *PV* kas masuk x total investasi.....(4.51)

Tabel 4.152 Pengujian *Textbox* Hasil *Profitability Index*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan
2.	<i>Input</i> data yang sama	Dapat dilakukan

➤ *Button* simpan data

Gambar 4.215 *Button* Simpan Data

Penjelasan: *User* dapat melakukan menyimpan data *profitability index*.

Tabel 4.153 Pengujian *Button* Simpan Data

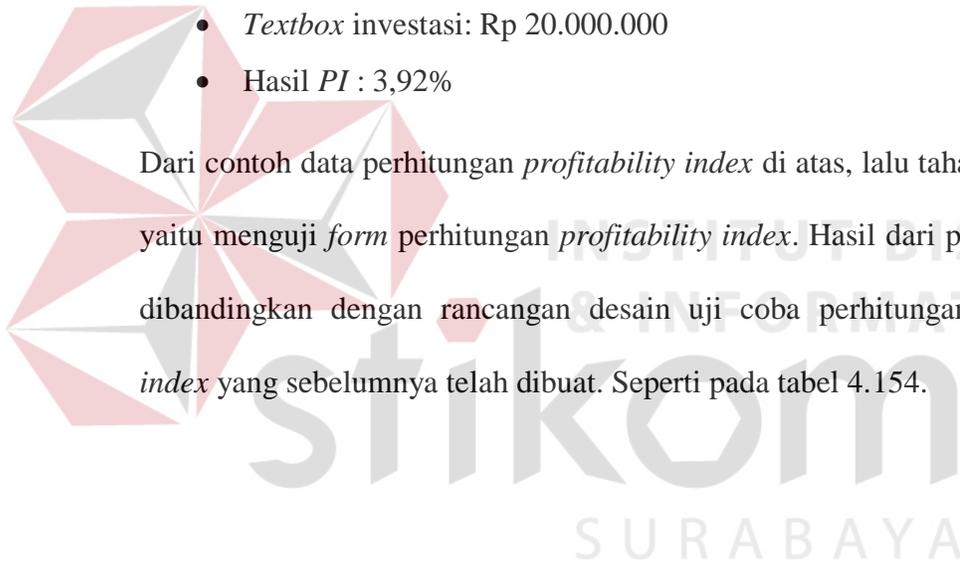
No.	Pengujian	Hasil
1.	Melakukan penyimpanan jika ada data yang kosong	Tidak dapat dilakukan

Langkah selanjutnya yaitu melakukan *test* data. *Test* data diperlukan untuk mengetahui apakah fungsi pada aplikasi yang telah dibuat ini sudah berjalan dengan baik dan sesuai dengan tujuan atau *output* yang diharapkan. Sebagai perbandingan, perlu dilakukan uji coba.

Diketahui:

- Kode *NPV* : *NPV*-001
- *Textbox* akumulasi *PV* kas masuk : Rp 78.494.208
- Kode mesin: *KM*-002
- *Textbox* investasi: Rp 20.000.000
- Hasil *PI* : 3,92%

Dari contoh data perhitungan *profitability index* di atas, lalu tahap selanjutnya yaitu menguji *form* perhitungan *profitability index*. Hasil dari pengujian akan dibandingkan dengan rancangan desain uji coba perhitungan *profitability index* yang sebelumnya telah dibuat. Seperti pada tabel 4.154.



Tabel 4.154 Hasil Uji Coba Perhitungan *Profitability Index*

<i>Test Case ID</i>	<i>Test Case</i>	<i>Test Data</i>	<i>Output yang diharapkan</i>	<i>Actual Output</i>	<i>Kesimpulan</i>
1	Menampilkan nilai <i>PV</i> kas masuk	Memilih kode <i>NPV</i> : <i>NPV-001</i>	Rp 78.494.208		Sesuai
2	Menampilkan nilai total investasi	Memilih kode mesin: <i>KM-001</i>	Rp 20.000.000		Sesuai
3	Menampilkan hasil <i>Profitability Index</i>	Setelah terisi <i>textbox</i> akumulasi <i>pv</i> kas masuk, total investasi dan menekan <i>button</i> hitung atau enter	$PI = PV \text{ kas masuk} / \text{investasi}$ $78.494.208 / 20.000.000 = 3,92 \%$		Sesuai
4	Menyimpan data <i>Profitability Index</i>	Menekan <i>button</i> simpan data	Tersimpan ke <i>database profitability index</i> dan tampil di <i>datagridview</i>		Sesuai

4.4.16 Uji Coba *Form* Hasil Analisa Kelayakan Investasi Mesin

Form kelayakan investasi digunakan sebagai acuan untuk memberi keputusan bagi pihak *engineering* untuk melakukan investasi mesin. Uji coba kelayakan investasi membutuhkan data perhitungan *average rate of return*, *payback period*, *net present value*, *internal rate of return* serta data perhitungan *profitability index*. Di *form* ini terdapat:

➤ *Combobox* kode *ARR*



Gambar 4.216 *Combobox* Kode *ARR*

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* kode *ARR*, data ditampilkan berasal dari *database average rate of return*.

Tabel 4.155 Pengujian *Combobox* kode *ARR*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

➤ *Combobox* kode *PP*



Gambar 4.217 *Combobox* Kode *PP*

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* kode *PP*, data ditampilkan berasal dari *database payback period*.

Tabel 4.156 Pengujian *Combobox* Kode *PP*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

➤ *Combobox* kode *NPV*Gambar 4.218 *Combobox* Kode *NPV*

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* kode *NPV*, data ditampilkan berasal dari *database net present value*.

Tabel 4.157 Pengujian *combobox* Kode *NPV*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

➤ *Combobox* kode *IRR*Gambar 4.219 *Combobox* Kode *IRR*

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* kode *IRR*, data ditampilkan berasal dari *database internal rate of return*.

Tabel 4.158 Pengujian *Combobox* Kode *IRR*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

➤ *Combobox* kode *PI*



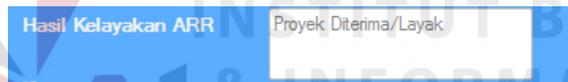
Gambar 4.220 *Combobox* Kode *PI*

Penjelasan: *User* hanya bisa memilih sesuai pilihan yang ada pada *combobox* kode *PI*, data ditampilkan berasal dari *database profitability index*.

Tabel 4.159 Pengujian *Combobox* Kode *PI*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

➤ *Textbox* hasil kelayakan investasi *ARR*



Gambar 4.221 *Textbox* Hasil Kelayakan Investasi *ARR*

Penjelasan: *User* tidak dapat melakukan *input* pada *textbox* hasil kelayakan investasi *ARR* karena data ditampilkan berasal dari perbandingan:

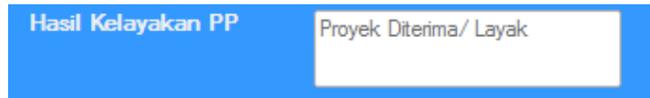
Jika $ARR >$ Tingkat keuntungan yang ditargetkan = layak atau diterima

Jika $ARR <$ Tingkat keuntungan yang ditargetkan = tidak layak atau ditolak

Tabel 4.160 Pengujian *Textbox* Hasil Kelayakan Investasi *ARR*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

➤ *Textbox* hasil kelayakan investasi *PP*



Gambar 4.222 *Textbox* Hasil Kelayakan Investasi *PP*

Penjelasan: *User* tidak dapat melakukan *input* pada *textbox* hasil kelayakan investasi *PP* karena data ditampilkan berasal dari perbandingan:

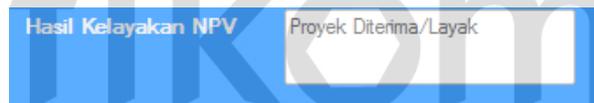
Jika *PP* Lebih pendek yang ditargetkan = layak atau diterima

Jika *PP* Lebih lama yang ditargetkan = tidak layak atau ditolak

Tabel 4.161 Pengujian *Textbox* Hasil Kelayakan Investasi *PP*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

➤ *Textbox* hasil kelayakan investasi *NPV*



Gambar 4.223 *Textbox* Hasil Kelayakan Investasi *NPV*

Penjelasan: *User* tidak dapat melakukan *input* pada *textbox* hasil kelayakan investasi *NPV* karena data ditampilkan berasal dari perbandingan:

Jika $NPV > 0$ = layak atau diterima

Jika $NPV < 0$ = tidak layak atau ditolak

Tabel 4.162 Pengujian *Textbox* Hasil Kelayakan Investasi *NPV*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

- *Textbox* hasil kelayakan investasi *IRR*



Gambar 4.224 *Textbox* Hasil Kelayakan Investasi *IRR*

Penjelasan: *User* tidak dapat melakukan *input* pada *textbox* hasil kelayakan investasi *IRR* karena data ditampilkan berasal dari perbandingan:

Jika $IRR > \text{Tingkat Bunga}$ = layak atau diterima

Jika $IRR < \text{Tingkat Bunga}$ = tidak layak atau ditolak

Tabel 4.163 Pengujian *textbox* hasil kelayakan investasi *IRR*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

- *Textbox* hasil kelayakan investasi *PI*



Gambar 4.225 *Textbox* Hasil Kelayakan Investasi *PI*

Penjelasan: *User* tidak dapat melakukan *input* pada *textbox* hasil kelayakan investasi *PI* karena data ditampilkan berasal dari perbandingan:

Jika $PI > 1$ = layak atau diterima

Jika $PI < 1$ = Tidak layak atau ditolak

Tabel 4.164 Pengujian *Textbox* Hasil Kelayakan Investasi *PI*

No.	Pengujian	Hasil
1.	Melakukan penginputan kosong	Tidak dapat dilakukan

Langkah selanjutnya yaitu melakukan *test* data. *Test* data diperlukan untuk mengetahui apakah fungsi pada aplikasi yang telah dibuat ini sudah berjalan dengan baik dan sesuai dengan tujuan atau *output* yang diharapkan. Sebagai perbandingan, perlu dilakukan uji coba.

Diketahui:

- Kode *ARR* : *ARR-001*
- *Textbox* hasil *ARR* : 146,07 %
- *Textbox* *ARR* yang diisyaratkan : 18 %
- Hasil kelayakan *ARR* : Proyek diterima/layak
- Kode *PP* : *PP-001*
- *Textbox* hasil *PP* : 1,19 Tahun
- *Textbox* *PP* yang diisyaratkan : 4 tahun
- Hasil kelayakan *PP* : Proyek diterima/layak
- Kode *NPV* : *NPV-001*
- *Textbox* hasil *NPV* : Rp. 58.494.208
- Hasil kelayakan *NPV* : Proyek diterima/layak
- Kode *IRR* : *IRR-001*
- *Textbox* hasil *IRR* : 60,39 %
- *Textbox* *IRR* yang diisyaratkan : 18 %
- Hasil kelayakan *IRR* : Proyek diterima/layak
- Kode *PI*: *PI-001*
- *Textbox* hasil *PI*: 3,92 %
- Hasil kelayakan *PI*: Proyek diterima/layak
- Kesimpulan akhir : Layak

Dari contoh data kelayakan investasi di atas, lalu tahap selanjutnya yaitu menguji *form* kelayakan investasi. Hasil dari pengujian akan dibandingkan

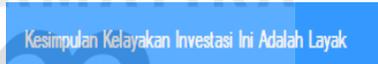
dengan rancangan desain uji coba perhitungan kelayakan investasi yang sebelumnya telah dibuat. Seperti pada tabel 4.165, 4.166.



Tabel 4.165 Hasil Uji Coba Kelayakan Investasi

<i>Test Case ID</i>	<i>Test Case</i>	<i>Test Data</i>	<i>Output yang diharapkan</i>	<i>Actual Output</i>	Kesimpulan
1	Menampilkan nilai <i>ARR</i>	Memilih kode <i>ARR</i> : <i>ARR-001</i> Hasil <i>ARR</i> : 146,07 % Target : 18%	Proyek diterima / layak		Sesuai
2	Menampilkan nilai <i>PP</i>	Memilih kode <i>PP</i> : <i>PP-001</i> Hasil <i>PP</i> : 1,19 Tahun Target : 4 Tahun	Proyek diterima / layak		Sesuai
3	Menampilkan nilai <i>NPV</i>	Memilih kode <i>NPV</i> : <i>NPV-001</i> Hasil <i>NPV</i> : Rp. 58.494.208	Proyek diterima / layak		Sesuai

Tabel 4.166 Hasil Uji Coba Kelayakan Investasi (Lanjutan)

4	Menampilkan nilai <i>IRR</i>	Memilih kode <i>IRR</i> : <i>IRR-001</i> Hasil <i>IRR</i> : 60,39% Target : 18 %	Proyek diterima / layak		Sesuai
5	Menampilkan nilai <i>PI</i>	Memilih kode <i>PI</i> : <i>PI-001</i> Hasil <i>PI</i> : 3,92	Proyek diterima / layak		Sesuai
6	Menampilkan kesimpulan akhir	Setelah semua kelayakan di proses maka akan muncul kesimpulan akhir kelayakan investasi	Kesimpulan kelayakan ini adalah layak		

4.5 Evaluasi Hasil Uji Coba Aplikasi

Setelah dilakukan uji coba terdapat hasil yang sama antara *output* yang diharapkan dengan *output* yang sebenarnya. Hasil yang didapat terdapat beberapa perbandingan antara lain sebagai berikut:

1. Sistem perhitungan penyusutan yang telah dibuat ini dapat melakukan pengolahan data secara efektif dan telah ada perhitungan menurut metode. Perhitungan dimulai dengan nilai sisa dengan total investasi serta umur ekonomis. Seperti contoh data pada saat uji coba, telah diketahui nilai sisa Rp 11.000.000 dan diketahui investasi Rp 20.000.000 dan diketahui umur ekonomis 4 tahun, maka penghitungan penyusutan yaitu :
$$(Rp\ 20.000.000 - 11.000.000 / 4\text{tahun}) = Rp\ 2.250.000,00$$
 dan hasilnya akan muncul secara otomatis oleh sistem. Hasil dari semua penghitungan telah sesuai dan lolos uji dengan *output* yang diharapkan pada saat perencanaan uji coba.
2. Sistem perhitungan rencana produksi yang telah dibuat ini dapat melakukan pengolahan data secara efektif dan telah ada perhitungan menurut metode. Perhitungan dimulai dengan harga konstan dengan koefisien regresi serta periode produksi. Seperti contoh data pada saat uji coba, telah diketahui harga konstan 10.000 unit dan diketahui koefisien regresi 0 dan diketahui periode 14, maka penghitungan rencana produksi yaitu :
$$10.000 + (0 \times 13) = 10.000$$
 unit dan hasilnya akan muncul secara otomatis oleh sistem. Hasil dari semua penghitungan telah sesuai dan lolos uji dengan *output* yang diharapkan pada saat perencanaan uji coba.

3. Sistem perhitungan biaya produksi yang telah dibuat ini dapat melakukan pengolahan data secara efektif dan telah ada perhitungan menurut metode. Perhitungan dimulai dengan biaya variabel dengan biaya tetap. Seperti contoh data pada saat uji coba, telah diketahui biaya variabel Rp 32.750.000 dan diketahui biaya tetap Rp 35.569.957, maka penghitungan biaya produksi yaitu : $32.750.000 + 35.569.957 = 68.319.957$ dan hasilnya akan muncul secara otomatis oleh sistem. Hasil dari semua penghitungan telah sesuai dan lolos uji dengan *output* yang diharapkan pada saat perencanaan uji coba.
4. Sistem perhitungan laba rugi yang telah dibuat ini dapat melakukan pengolahan data secara efektif dan telah ada perhitungan menurut metode. Perhitungan dimulai dengan biaya variabel dengan biaya tetap. Seperti contoh data pada saat uji coba, telah diketahui LB-002 maka pendapatan Rp 100.000.000 biaya variabel Rp 32.750.000 biaya tetap Rp 35.569.957 dan pajak 30%, maka penghitungan laba rugi yaitu : $100.000.000 - (32.750.000 + 35.569.957) - 30\% = 22.176.030$ dan hasilnya akan muncul secara otomatis oleh sistem. Hasil dari semua penghitungan telah sesuai dan lolos uji dengan *output* yang diharapkan pada saat perencanaan uji coba.
5. Sistem perhitungan aliran kas bersih yang telah dibuat ini dapat melakukan pengolahan data secara efektif dan telah ada perhitungan menurut metode. Perhitungan dimulai dengan laba setelah pajak perperiode dengan penyusutan. Seperti contoh data pada saat uji coba, telah diketahui LB-001 maka laba setelah pajak periode 1 Rp 12.994.363 dan KM-002 maka

penyusutan Rp 2.250.000, maka penghitungan aliran kas bersih yaitu :
 $12.994.363 + 2.250.000 = 15.244.363$ dan hasilnya akan muncul secara otomatis oleh sistem. LB-001 maka laba setelah pajak periode 2 Rp 22.176.030 dan KM-002 maka penyusutan Rp 2.250.000, maka penghitungan aliran kas bersih yaitu : $22.176.030 + 2.250.000 = 24.426.030$ dan hasilnya akan muncul secara otomatis oleh sistem. LB-001 maka laba setelah pajak periode 3 Rp 36.175.736 dan KM-002 maka penyusutan Rp 2.250.000, maka penghitungan aliran kas bersih yaitu : $36.175.736 + 2.250.000 = 38.425.736$ dan hasilnya akan muncul secara otomatis oleh sistem. LB-001 maka laba setelah pajak periode 4 Rp 45.507.887 dan KM-002 maka penyusutan Rp 2.250.000, maka penghitungan aliran kas bersih yaitu : $45.507.887 + 2.250.000 = 47.757.887$ dan hasilnya akan muncul secara otomatis oleh sistem. Hasil dari semua penghitungan telah sesuai dan lolos uji dengan *output* yang diharapkan pada saat perencanaan uji coba.

6. Sistem perhitungan *average rate of return* yang telah dibuat ini dapat melakukan pengolahan data secara efektif dan telah ada perhitungan menurut metode. Perhitungan dimulai dengan rata-rata laba setelah pajak dan total investasi. Seperti contoh data pada saat uji coba, telah diketahui kode laba rugi LB-001 maka sistem secara otomatis akan menampilkan rata-rata laba setelah pajak Rp 29.213.504 dan diketahui kode mesin KM-002 maka sistem secara otomatis akan menampilkan total investasi Rp 20.000.000, maka penghitungan *average rate of return* yaitu : $(Rp\ 29.213.504 / Rp\ 20.000.000) \times 100\% = 146,07\%$ dan hasilnya akan

muncul secara otomatis oleh sistem. Hasil dari semua penghitungan telah sesuai dan lolos uji dengan *output* yang diharapkan pada saat perencanaan uji coba.

7. Sistem perhitungan *payback period* yang telah dibuat ini dapat melakukan pengolahan data secara efektif dan telah ada perhitungan menurut metode. Perhitungan dimulai dengan akumulasi aliran kas bersih perperiode dan total investasi. Seperti contoh data pada saat uji coba, telah diketahui kode periode P-001 maka sistem secara otomatis akan menampilkan kode aliran kas bersih KAS-001 dan menampilkan aliran kas bersih perperiode Rp 15.244.363, kode periode P-002 maka sistem secara otomatis akan menampilkan kode aliran kas bersih KAS-001 dan menampilkan aliran kas bersih perperiode Rp 39.670.393 dan diketahui kode mesin KM-002 maka sistem secara otomatis akan menampilkan total investasi Rp 20.000.000, maka penghitungan *payback period* yaitu $1 + ((20.000.000 - 15.244.363) / (39.670.393 - 15.244.363)) \times 1 \text{ tahun} : 1,19 \text{ tahun}$ dan hasilnya akan muncul secara otomatis oleh sistem. Hasil dari semua penghitungan telah sesuai dan lolos uji dengan *output* yang diharapkan pada saat perencanaan uji coba.
8. Sistem perhitungan *net present value* yang telah dibuat ini dapat melakukan pengolahan data secara efektif dan telah ada perhitungan menurut metode. Perhitungan dimulai dengan aliran kas bersih perperiode dan total investasi. Seperti contoh data pada saat uji coba, telah diketahui kode periode P-001 maka sistem secara otomatis akan menampilkan kode aliran kas bersih KAS-001 dan menampilkan aliran kas bersih perperiode

Rp 15.244.363, P-002 maka sistem secara otomatis akan menampilkan kode aliran kas bersih KAS-001 dan menampilkan aliran kas bersih perperiode Rp 24.426.030, P-003 maka sistem secara otomatis akan menampilkan kode aliran kas bersih KAS-001 dan menampilkan aliran kas bersih perperiode Rp 38.425.736, P-004 maka sistem secara otomatis akan menampilkan kode aliran kas bersih KAS-001 dan menampilkan aliran kas bersih perperiode Rp 47.757.887 dan diketahui kode mesin KM-002 maka sistem secara otomatis akan menampilkan total investasi Rp 20.000.000 dan *discount factor* perperiode maka penghitungan *PV* kas masuk yaitu $(15.244.363 \times 0,847) = 12.911.975$ periode 1, $(24.426.030 \times 0,718) = 17.537.890$ periode 2, $(38.425.736 \times 0,609) = 23.401.273$ periode 3, $(47.757.887 \times 0,516) = 24.643.070$ periode 4 dan hasil akumulasi *PV* kas masuk yaitu 78.494.208 dan hasil semuanya akan muncul secara otomatis oleh sistem. Kemudian setelah mengakumulasi *PV* kas masuk maka menghitung *net present value* yaitu akumulasi *PV* kas masuk dikurangi total investasi $(78.494.208 - 20.000.000) = 58.494.208$. Hasil dari semua penghitungan telah sesuai dan lolos uji dengan *output* yang diharapkan pada saat perencanaan uji coba.

9. Sistem perhitungan *internal rate of return* yang telah dibuat ini dapat melakukan pengolahan data secara efektif dan telah ada perhitungan menurut metode. Perhitungan dimulai dengan aliran kas bersih perperiode dan total investasi. Seperti contoh data pada saat uji coba, telah diketahui kode periode P-001 maka sistem secara otomatis akan menampilkan kode aliran kas bersih KAS-001 dan menampilkan aliran kas bersih perperiode

Rp 15.244.363, P-002 maka sistem secara otomatis akan menampilkan kode aliran kas bersih KAS-001 dan menampilkan aliran kas bersih perperiode Rp 24.426.030, P-003 maka sistem secara otomatis akan menampilkan kode aliran kas bersih KAS-001 dan menampilkan aliran kas bersih perperiode Rp 38.425.736, P-004 maka sistem secara otomatis akan menampilkan kode aliran kas bersih KAS-001 dan menampilkan aliran kas bersih perperiode Rp 47.757.887 dan diketahui kode mesin KM-002 maka sistem secara otomatis akan menampilkan total investasi Rp 20.000.000 dan *discount factor perperiode* maka penghitungan *PV* kas masuk berlawanan yaitu $(15.244.363 \times 0,735) = 11.204.607$ periode 1, $(24.426.030 \times 0,541) = 13.214.482$ periode 2, $(38.425.736 \times 0,398) = 15.293.443$ periode 3, $(47.757.887 \times 0,292) = 13.945.303$ periode 4 dan hasil akumulasi *PV* kas masuk berlawanan yaitu 53.657.835 dan hasil semuanya akan muncul secara otomatis oleh sistem. Kemudian dihitung untuk periode selanjutnya. Kemudian setelah mengakumulasi *PV* kas masuk berlawanan maka menghitung *net present value* berlawanan yaitu akumulasi *PV* kas masuk berlawanan dikurangi total investasi $(53.657.835 - 20.000.000) = 33.657.835$. Kemudian menghitung *internal rate of return* : $18\% + (58.494.208 / (78.494.208 - 53.657.875)) \times (36\% - 18\%) = 60,39\%$. Hasil dari semua penghitungan telah sesuai dan lolos uji dengan *output* yang diharapkan pada saat perencanaan uji coba.

10. Sistem perhitungan *profitability index* yang telah dibuat ini dapat melakukan pengolahan data secara efektif dan telah ada perhitungan menurut metode. Perhitungan dimulai dengan akumulasi *PV* kas masuk

dan total investasi. Seperti contoh data pada saat uji coba, telah diketahui kode *NPV: NPV-001* maka sistem secara otomatis akan menampilkan *PV* kas masuk Rp 78.494.208 dan diketahui kode mesin *KM-002* maka sistem secara otomatis akan menampilkan total investasi Rp 20.000.000, maka penghitungan *profitability index* yaitu $\text{Rp } 78.494.208 / \text{Rp } 20.000.000 = 3,92$ dan hasilnya akan muncul secara otomatis oleh sistem. Hasil dari semua penghitungan telah sesuai dan lolos uji dengan *output* yang diharapkan pada saat perencanaan uji coba.

11. Kelayakan investasi merupakan perbandingan yang telah dibuat, dapat melakukan pengolahan data secara efektif dan menurut target yang ditentukan/diisyaratkan. Perbandingan dimulai dengan *average rate of return*, *payback period*, *net present value*, *internal rate of return* dan *profitability index*. Seperti contoh data pada saat uji coba, telah diketahui kode *ARR: ARR-001* maka sistem secara otomatis akan menampilkan *ARR* 146,07 %, kode *PP: PP-001* maka sistem secara otomatis akan menampilkan *PP* 1,19 Tahun, kode *NPV: NPV-001* maka sistem secara otomatis akan menampilkan *NPV* Rp 58.494.208, kode *IRR: IRR-001* maka sistem secara otomatis akan menampilkan *IRR* 60,39%, kode *PI: PI-001* maka sistem secara otomatis akan menampilkan *PI* 3,92, maka Kelayakan *ARR* = 146,07 % > 18% maka proyek layak atau diterima, Kelayakan *PP* = 1,19 tahun < 4 tahun maka proyek layak atau diterima, Kelayakan *NPV* = 58.494.208 > 0 maka proyek layak atau diterima, Kelayakan *IRR* = 60,39% > 18% maka proyek layak atau diterima, Kelayakan *PI* = 3,92 > 1 maka proyek layak atau diterima dan hasilnya

akan muncul secara otomatis oleh sistem. Hasil dari semua perbandingan telah sesuai dan lolos uji dengan *output* yang diharapkan pada saat perencanaan uji coba.



