

# PERANCANGAN APLIKASI PENCATATAN KEUANGAN BERBASIS *WEBSITE* DENGAN MENGGUNAKAN METODE AGILE DEVELOPMENT STUDI KASUS PADA WARUNG NASI SUNDA MANG UJANG

## WEBSITE-BASED FINANCIAL RECORDING APPLICATION USING THE AGILE DEVELOPMENT METHOD CASE STUDY ON THE WARUNG NASI SUNDA MANG UJANG

Muhamad Arief Damanhuri<sup>1</sup>, Ir. Budi Praptono, M.M.<sup>2</sup>, Maria Dellarosawati I, S.T, M.B.A<sup>3</sup>

<sup>123</sup>Prodi S1 Teknik Industri, Fakultas Rekayasa Industri Telkom

<sup>1</sup>[mariefdamanhuri@student.telkomuniversity.ac.id](mailto:mariefdamanhuri@student.telkomuniversity.ac.id), <sup>2</sup>[budipraptono35@gmail.com](mailto:budipraptono35@gmail.com),

<sup>3</sup>[mariadellarosawati@telkomuniversity.ac.id](mailto:mariadellarosawati@telkomuniversity.ac.id)

### Abstrak

Pada zaman yang sekarang ini teknologi berkembang semakin pesat, suatu informasi dan perkembangan teknologi sendiri sangatlah berpengaruh dan mempunyai peran penting bahkan menjadi kebutuhan di mayoritas masyarakat. Tujuan penelitian ini adalah untuk membantu pengelolaan keuangan dan pembukuan pada Warung Nasi Sunda Mang Ujang dengan cara merancang aplikasi pencatatan keuangan dengan menggunakan metode *Agile Development*.

Hasil dari penelitian ini menunjukkan bahwa dengan menggunakan Aplikasi Pencatatan Keuangan, Warung Nasi Sunda Mang Ujang dapat lebih bisa mengontrol data pengeluaran dan pemasukan yang dilakukan dan juga dapat menentukan harga lebih optimal untuk penjualan produk yang dilakukan.

**Kata kunci :** scrum, aplikasi pencatatan keuangan, laporan keuangan

### Abstract

*In this day and age technology develops more rapidly, the development of information and technology itself is very influential and has an important role and even becomes a necessity in most societies. The purpose of this study is to assist financial management and bookkeeping at Sunda Mang Ujang Warung Nasi by designing financial recording applications using the Agile Development method.*

*The results of this study indicate that by using the Financial Recording Application, Sunda Mang Ujang Rice Stalls can be better able to control the expenditure and income data conducted and also can determine a more optimal price for product sales.*

**Keywords:** Scrum, financial recording application, financial statements

### 1. Pendahuluan

Pada zaman yang sekarang ini teknologi berkembang semakin pesat, teknologi canggih bermunculan dan persaingan semakin tinggi karena kebutuhan terhadap suatu informasi dan perkembangan teknologi sendiri sangatlah berpengaruh dan mempunyai peran penting bahkan menjadi kebutuhan di mayoritas masyarakat.



Gambar 1. Pendapatan Rata – Rata Perhari Warung Nasi Sunda Mang Ujang

Berdasarkan dari grafik yang didapat ditunjukkan bahwa dari tahun 2007 pendapatan rata – rata per hari Warung Nasi Sunda Mang Ujang mengalami peningkatan setiap tahunnya. Warung Nasi Sunda Mang Ujang membutuhkan aplikasi guna membantu untuk mencatat keuangan dengan baik agar catatan keuangan dapat dikontrol dan tidak hilang jika dibandingkan dengan pencatatan dengan manual menggunakan kertas. Beberapa faktor yang dijadikan rumusan yaitu

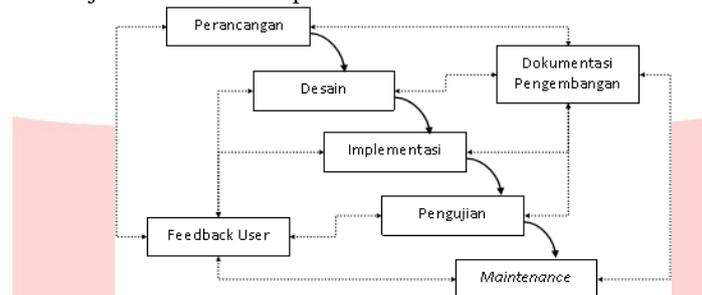
(1) bagaimana cara merancang aplikasi pencatatan keuangan, (2) bagaimana merancang aplikasi yang dapat mengelola keuangan

Perancangan aplikasi pencatatan keuangan untuk Warung Nasi Sunda Mang Ujang ini dilakukan dengan metode Agile scrum. Penulis memilih metode Agile Scrum dikarenakan karakteristik Agile Scrum sendiri yang dimana pada setiap prosesnya harus menerima persetujuan dengan user agar aplikasi sesuai dengan keinginan user dan meminimalisir kesalahan serta terjadinya *miss communication*. Tujuan dilakukannya perancangan aplikasi ini guna membantu warung nasi mang ujang untuk (1) membantu pencatatan keuangan, (2) membantu mengelola keuangan.

## 2. Tinjauan Pustaka

### 2.1. Pengertian Agile Development dan Scrum

Menurut M. Arif Firdaus (2017), Agile merupakan seperangkat metode yang dapat membantu tim untuk berfikir lebih efektif dan efisien didalam bekerja dan membuat keputusan.



Gambar 2. System Development Life Cycle Agile

Pada Gambar II.1 dijelaskan tahapan *System Development Life Cycle* metode Agile yang dimana tahapannya sebagai berikut :

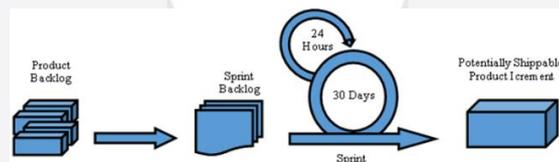
1. Perancangan
2. Desain
3. Implementasi
4. Pengujian
1. Maintenance

Pada setiap tahap SDLC dilakukan tahap feedback user yang dimana user akan melihat dan menyetujui tentang apa yang telah diranacang oleh perancang aplikasi.

Menurut Ken Schwaber dan Jeff Sutherland ( 2017 : 11 ), Scrum merupakan sebuah kerangka kerja yang dimana orang dapat mengatasi masalah kompleks adaptif, dimana pada saat yang bersamaan itu mereka juga dapat menghantarkan produk dengan nilai yang tinggi secara produktif dan kreatif.

Sifat *Scrum* sendiri yaitu :

1. Ringan
2. Mudah Dipelajari
3. Sulit dikuasai



Gambar 3. Siklus Scrum

Pada Gambar II.2 dijelaskan bahwa siklus scrum memiliki beberapa tahapan yang dimana sebagai berikut :

1. *Product Backlog*

Pada tahapan *product backlog* adalah proses dimana penentuan daftar item yang diperlukan untuk membuat suatu perancangan sistem.

2. *Sprint Backlog*

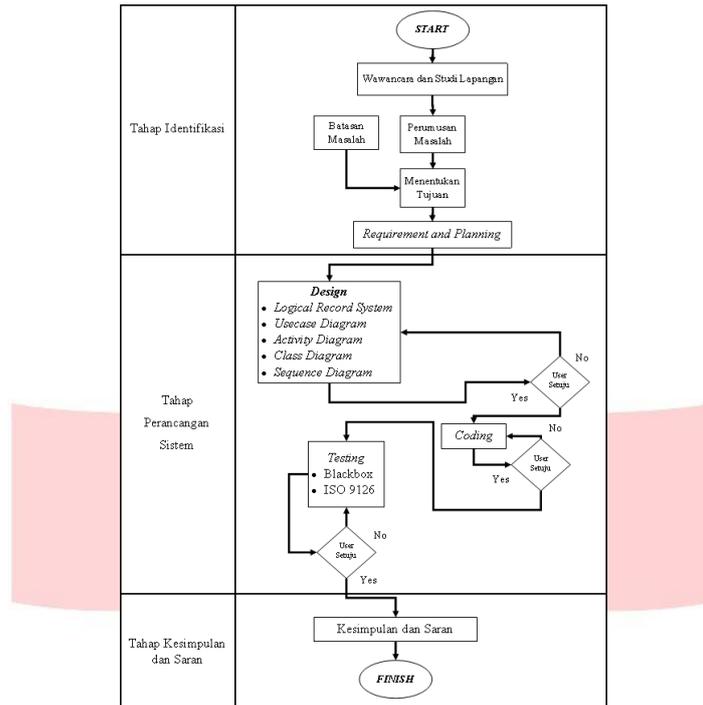
*Sprint backlog* ini merupakan sekumpulan item dari produk *backlog* yang telah dipilih untuk dirancang di tahap *sprint*, dan juga didalamnya terdapat rencana untuk mengembangkan produk dan merealisasikan di *sprint goal*.

3. *Sprint*

*Sprint* merupakan jantung atau inti dari scrum itu sendiri yang berarti bahwa batasan waktu selama satu bulan atau kurang, dimana pekerjaan dianggap selesai, bisa digunakan dan berpotensi untuk dirilis. *Sprint* biasanya memiliki durasi yang tetap sepanjang proses pengembangan sistem hingga sistem itu selesai.

### 3. Metode Pemecahan Masalah

Berikut merupakan sistematika pemecahan masalah yang digunakan dalam penelitian ini :



Gambar 4. Sistematika Pemecahan Masalah

Pada Gambar 4 dapat diketahui bahwa tahap pertama dalam penelitian yaitu tahap identifikasi dimana wawancara dan studi lapangan pada Warung Nasi Sunda Mang Ujang, menetapkan rumusan masalah yang didapat, menentukan batasan masalah dan tujuannya serta membuat *planning* dari kebutuhan *user*.

Tahap selanjutnya yaitu perancangan sistem dimana dimulai dari tahap desain UML, lanjut kepada tahap pemrograman atau *coding* setelah program selesai program diuji menggunakan *Blackbox* dan ISO:9126, pada setiap tahap dilakukan verifikasi terhadap *user* agar proses bisa dilanjutkan sesuai keinginan *user*.

Tahap terakhir yaitu kesimpulan dan saran dimana dijelaskan kembali mengenai apa yang telah dilakukan pada penelitian secara singkat dan menyeluruh dari proses awal hingga proses akhir dan juga saran untuk pengembangan aplikasi yang lebih menunjang.

## 4. Pembahasan

### 4.1 Gambaran Umum Sistem Lama

Tabel 1. Gambaran Umum Sistem Lama

Sistem Lama	Masalah	Usulan
Perhitungan <i>cash in</i> dan <i>cash out</i> dilakukan dengan cata manual	Perhitungan menjadi lama dan kadang tidak sesuai dengan yang ada.	Membuat aplikasi pencatatan keuangan agar perhitungan menjadi lebih efektif dan efisien dan mengurangi kesalahan data.
<i>Cash out</i> dan <i>cash in</i> tidak ditulis atau didokumentasikan oleh buku.	Tidak adanya keterangan yang jelas mengenai <i>cash in</i> dan <i>cash out</i> .	Dengan adanya aplikasi catatan <i>cash in</i> dan <i>cash out</i> tidak akan hilang.
Perhitungan dilakukan pada saat proses penjualan telah selesai dilakukan	Menambah waktu kerja untuk menghitung pendapatan.	Perhitungan dapat dilakukan kapan saja dengan cepat.

### 4.2 Tahap Requirement User

Berikut merupakan tahap *requirement user* yang didapatkan dari hasil wawancara perancang aplikasi kepada *user* dimana :

1. Pada tahap 1 merupakan data kebutuhan kasaran dari *user*.
2. Pada tahap 2 dilakukan pertimbangan eliminasi kebutuhan yang akan dibuatkan dalam aplikasi, berdasarkan:
  - M ( *Madatory* ) : Apakah sistem dibutuhkan atau penting
  - D ( *Desirable* ) : Apakah sistem diinginkan atau tidak terlalu penting
  - I ( *inessential* ) : Diluar dari sistem dan dieliminasi

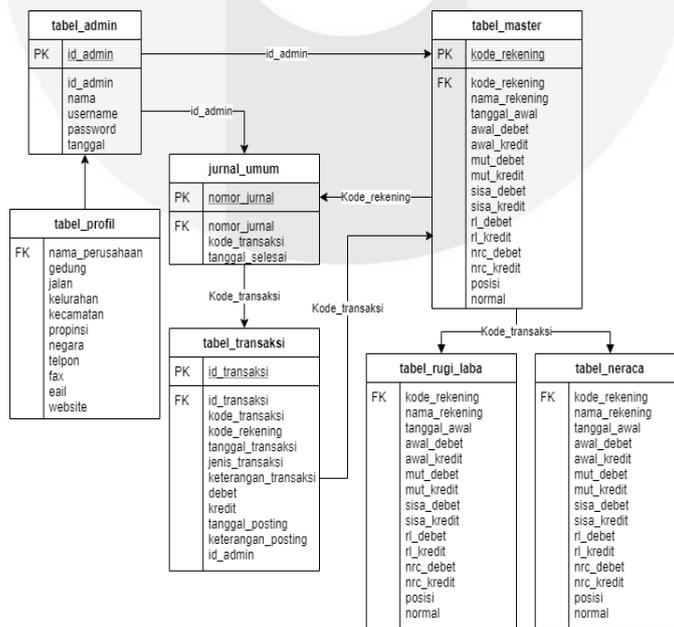
- Pada tahap 3, pemadatan sistem yang diinginkan oleh user dan akan dilakukan eliminasi kembali dengan keterangan lain seperti :
  - T (*technical*) : Apakah sistem dapat dibuat.
  - O (*operational*) : Apakah sistem dapat dikembangkan.
  - E (*economi*) : Apakah biaya pembuatannya mahal.
  - L (*Low*) : Mudah atau murah.
  - M (*Medium*) : Sedikit sulit atau sedikit mahal.
  - H (*High*) : Sulit atau mahal.
- Pada tahap 4 *final draft* kebutuhan *user* yang telah melalui tahap eliminasi.

Tabel 2. Elisitasi tahap 4

Fungsional	
No	Sistem yang diinginkan : Keterangan
1	Sistem login terintegrasi
2	Sistem dapat melakukan pengisian transaksi jurnal umum
3	Sistem dapat menampilkan data jurnal umum yang akan dipost
4	Sistem dapat melakukan <i>posting</i>
5	Sistem dapat menampilkan data buku jurnal
6	Sistem dapat melakukan perhitungan laba rugi dan neraca
7	Sistem dapat menampilkan laba rugi
8	Sistem dapat menampilkan neraca
11	Sistem dapat diakses dimana saja
13	Sistem dapat melakukan hapus data
14	Sistem dapat menampilkan data akun
15	Sistem dapat melakukan pengisian data akun
16	Sistem dapat mencetak data akun menjadi PDF
17	Sistem memiliki kemampuan <i>logout</i>
18	Sistem mempunyai <i>pop-up</i> verifikasi
19	Sistem dapat menunjukkan waktu
Non Fungsional	
No	Sistem yang diinginkan : Keterangan
1	Dapat diakses menggunakan <i>gadget</i>
2	Memiliki tampilan yang <i>simple</i> dan menarik
3	Aplikasi ringan untuk digunakan
4	Memiliki bahasa yang umum

**4.3 Logical Record Structure ( design database )**

Pada gambar 5 ini dijelaskan interaksi antar tabel atau *class* menggunakan referensi dari *primary key* setiap tabel atau *class*.

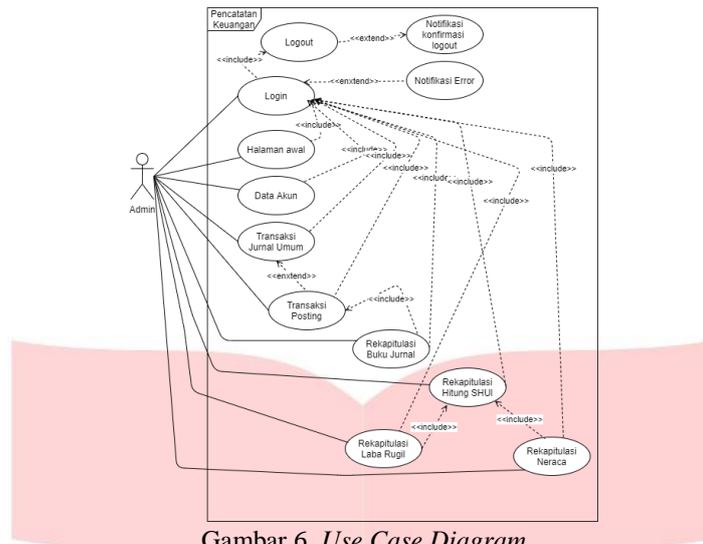


Gambar 5. Logical Record Structure

4.4 Unified Model Language ( UML ).

4.4.1 Use Case Diagram

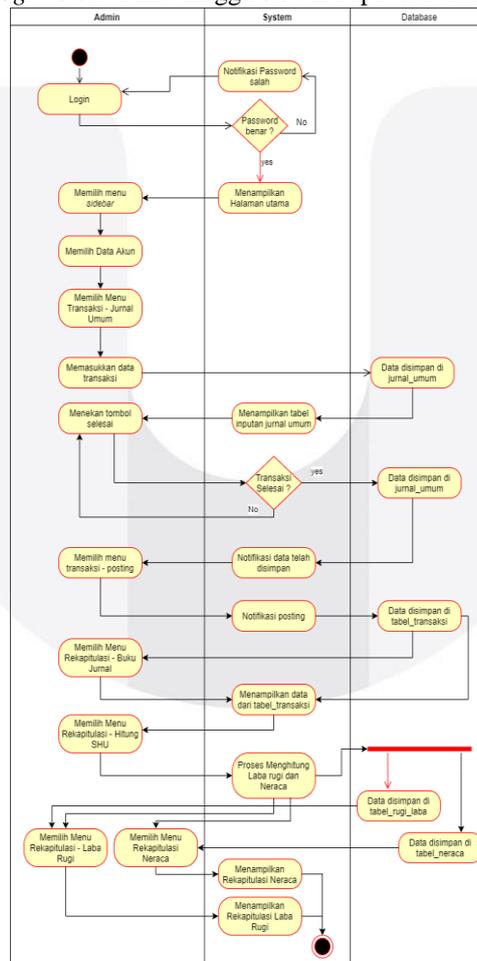
Pada Gambar 6 ini menunjukkan keterkaitan antara actor dengan case dimana menjelaskan user dapat mengakses semua fitur yang ada pada aplikasi tetapi user diharuskan untuk melakukan login terlebih dahulu.



Gambar 6. Use Case Diagram

4.4.2 Activity Diagram

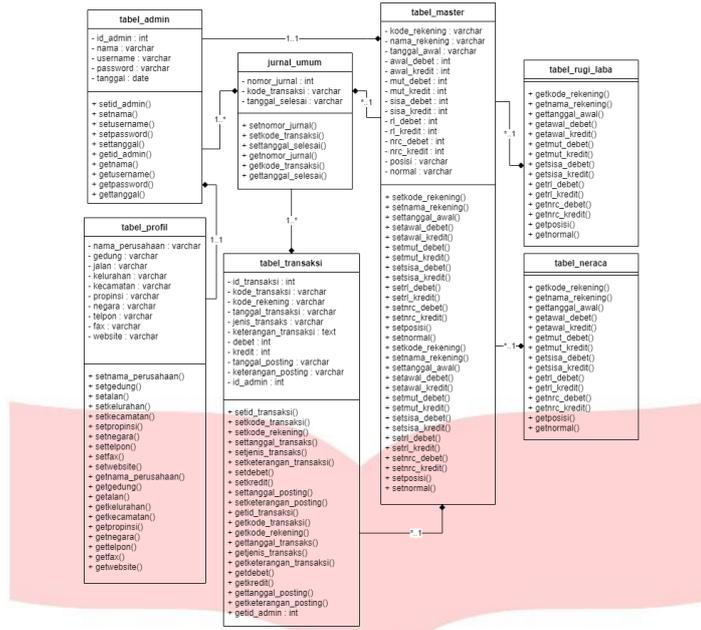
Pada Gambar 7 dimana menggambarkan aktivitas yang akan dilakukan saat akan menggunakan aplikasi pencatatan keuangan dimulai dari melakukan login oleh admin hingga sistem dapat menampilkan laporan laba rugi atau neraca.



Gambar 7. Activity Diagram Penggunaan Aplikasi

4.4.3 Class Diagram

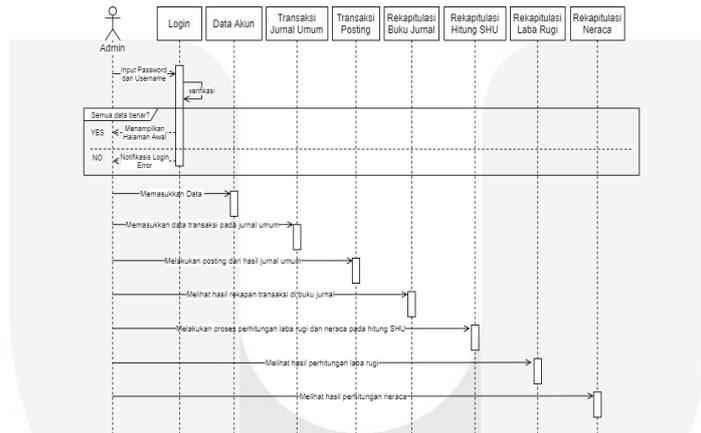
Pada Gambar 8 menunjukkan interaksi antar class atau tabel yang dimana setiap class memiliki 1 relationship untuk class lain atau lebih.



Gambar 8. Class Diagram

4.4.4 Sequence Diagram

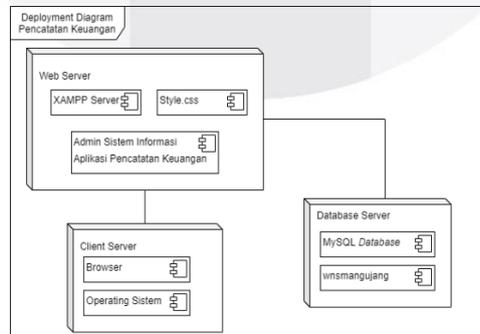
Pada Gambar 9 menunjukkan rangkaian atau urutan perancangan aplikasi yang dimulai dari login hingga pembuatan neraca.



Gambar 9. Sequence Diagram

4.5 Deployment Diagram

Pada Gambar 10 menunjukkan keterkaitan software dan hardware yang digunakan dalam perancangan aplikasi yang dilakukan.



Gambar 10. Deployment Diagram



### 4.7.2 Blackbox Testing

Tabel 3 *Blacbox Testing Test Case*

No	Skenario	Test Case	Hasil yang diharapkan	Status
1.	Berhasil Login Aplikasi	User atau Admin mengakses halaman login serta memasukkan <i>username</i> : admin <i>password</i> : admin	User atau admin dapat melakukan login dan menampilkan halaman awal	Pass
2.	Berhasil Akses sidebar	User atau admin dapat menggunakan fungsi sidebar dengan menekan tombol sidebar	User atau admin dapat mengakses sidebar dan memunculkan menu pada aplikasi	Pass
3.	Berhasil input data akun	User atau admin memilih menu Data Akun lalu memasukkan data sesuai dengan kebutuhan pada halaman data akun	Data akun berhasil diinputkan	Pass
4.	Berhasil delete data akun	User atau admin memilih menu Data Akun lalu memilih <i>action delete</i>	Data akun terpilih akan terhapus	Pass
5.	Berhasil melakukan input transaksi jurnal umum	User atau admin memilih menu Transaksi – Jurnal umum lalu memasukkan data transaksi yang akan dilakukan sesuai nomor akun yang ada	Nomor akun tersedia dan data berhasil di input	Pass
6.	Berhasil melakukan delete transaksi jurnal umum	User atau admin memilih menu Transaksi – Jurnal umum lalu menekan tombol <i>cancel</i> pada transaksi yang akan dibatalkan atau dihapus	Data transaksi terpilih akan terhapus dari daftar transaksi.	Pass
7.	Berhasil Posting Transaksi	User atau admin memilih menu Transaksi – Posting lalu menekan tombol <i>posting</i>	Data transaksi berhasil disimpan ke jurnal umum dan <i>database tabel_master</i>	Pass
8.	Berhasil Menampilkan Buku Jurnal	User atau admin memilih menu Rekapitulasi – Buku Jurnal	Data transaksi yang telah dilakukan dapat ditampilkan	Pass
9.	Berhasil Menghitung SHU	User atau admin memilih menu Rekapitulasi – Hitung SHU dan menekan tombol proses hitung SHU	Menampilkan notifikasi bahwa data telah dihitung dan tersimpan pada Laba rugi dan Neraca	Pass
10.	Berhasil Menampilkan Laba Rugi	User atau admin memilih menu Rekapitulasi – Laba Rugi	Menampilkan Hasil perhitungan transaksi dalam bentuk Laporan Laba Rugi	Pass
11.	Berhasil Menampilkan Neraca	User atau admin memilih menu Rekapitulasi – Neraca dan memilih tanggal yang ingin dilihat lalu menekan tombol neraca	Menampilkan Hasil perhitungan transaksi dalam bentuk Laporan Neraca	Pass

## 5. Kesimpulan

Berdasarkan penelitian yang telah dilakukan oleh penulis, maka dapat diambil kesimpulan yaitu:

1. Dengan dirancangnya aplikasi pencatatan keuangan berbasis website ini dapat membantu aktifitas pengelolaan keuangan di Warung Nasi Sunda Mang Ujang pada setiap harinya dengan cepat.
2. Dengan dirancangnya aplikasi pencatatan keuangan berbasis website ini Warung Nasi Sunda Mang Ujang dapat memantau pengelolaan keuangan dan juga dapat memantau harga pokok penjualan sesuai dengan perhitungan yang telah dilakukan.

## Daftar Pustaka

- [1] Alim.Yadanur, dkk. 2012. Pengembangan Sistem Informasi Administrasi Pemeriksaan Pasien Di Instalasi Radiologi Rsud Kajen Dengan Unified Process. Semarang: Universitas Diponegoro. Vol. 2, No. 4, ISSN 2086-4930.
- [2] Arief, M. Rudyanto. 2011. Pemrograman Web Dinamis Menggunakan PHP & MySQL. Yogyakarta : Andi
- [3] Ken Schwaber, J. S., 2017. Panduan scrum bahasa indonesia, s.l.: s.n.
- [4] Simanjuntak, O. S., 2013. PEMODELAN PENINGKATAN KUALITAS SISTEM INFORMASI AKADEMIK DENGAN MENGGUNAKAN STANDAR ISO 9126. Seminar Nasional Informatika, pp. A-315.