



## SISTEM PENCARIAN KOS BERBASIS WEB DI WILAYAH KOTA BANJARMASIN

Maghfur Maulana<sup>1\*</sup>, Nurhaeni<sup>1</sup>, Muhammad Zulfadhilah<sup>2</sup>, Bayu Nugraha<sup>1</sup>

<sup>1</sup>Program Studi Sistem Informasi, Universitas Sari Mulia, Indonesia

<sup>2</sup>Program Studi Teknologi Informasi, Universitas Sari Mulia, Indonesia

**Abstrak:** Kos merupakan jasa yang memberikan penawaran sebuah kamar atau tempat yang untuk dihuni dengan sejumlah pembayaran bulanan yang sudah ditentukan dalam periode tertentu. di kota Banjarmasin terdapat 178 kos yang tersebar diseluruh kecamatan dengan berbagai *range* harga dan fasilitas yang ditentukan oleh pemilik kos, di sisi lain kota Banjarmasin terdapat 30 Perguruan Tinggi yang menerima mahasiswa baru disetiap tahun dari dalam maupun dari luar kota Banjarmasin. Hal ini dapat menimbulkan permasalahan terutama bagi mahasiswa baru dari luar kota dan ingin melanjutkan ke perguruan tinggi di kota Banjarmasin, mahasiswa tidak mengetahui secara detail informasi kos. Tujuan dari penelitian ini yaitu membangun sistem pencarian kos berbasis web di wilayah kota Banjarmasin untuk mempermudah masyarakat terkhusus bagi mahasiswa yang berasal dari luar daerah Banjarmasin. Metode penelitian yang digunakan yaitu metode pengembangan *Waterfall*, dengan UML sebagai desain sistem, PHP sebagai bahasa pemrograman dan MySQL sebagai *database*. Hasil dari penelitian ini adalah sistem yang dihasilkan dapat menjalankan perintah sesuai dengan apa yang diinginkan, sistem ini digunakan untuk mempermudah user atau pencari kos untuk mempermudah proses pencarian kos dan sebagai wadah bagi pemilik kos di wilayah kota Banjarmasin.

**Kata kunci:** Banjarmasin, kos, mahasiswa, metode *waterfall*, sistem pencarian.

### I. PENDAHULUAN

Kos merupakan tempat tinggal sementara bagi masyarakat, pelajar mahasiswa, ataupun siswa tingkat SMA dan SMK. Pada era teknologi saat ini, tempat kos ingin menyajikan informasi sewa kepada masyarakat dan pemuda-pemudi secara cepat dalam pencarian tempat kos (Sahidu & Purwanto, 2022). Berdasarkan hasil temuan observasi, di kota Banjarmasin terdapat 178 kos yang tersebar di 5 kecamatan dengan berbagai *range* harga dan fasilitas yang ditentukan oleh

pemilik kos. Jumlah kos akan terus bertambah dikarenakan wilayah kota Banjarmasin terdapat 30 Perguruan Tinggi yang menerima mahasiswa baru dari dalam maupun dari luar kota Banjarmasin disetiap tahun. Hal ini juga dijadikan peluang oleh masyarakat untuk menambah penghasilan salah satunya dengan membuka usaha kos. (LLDIKTI, 2023).

Jika dilihat dari data kos yang tersebar di wilayah kota Banjarmasin, hal ini dapat menimbulkan permasalahan terutama bagi mahasiswa baru yang berasal dari luar daerah wilayah kota Banjarmasin dan ingin mencari kos di wilayah kota Banjarmasin yang sesuai dengan keinginannya. Mahasiswa merasa

\*) maghfurmaulana45@unism.ac.id

Diterima: 9 September 2023

Direvisi: 26 Mei 2024

Disetujui: 29 Mei 2024

DOI: 10.23969/infomatek.v26i1.10309

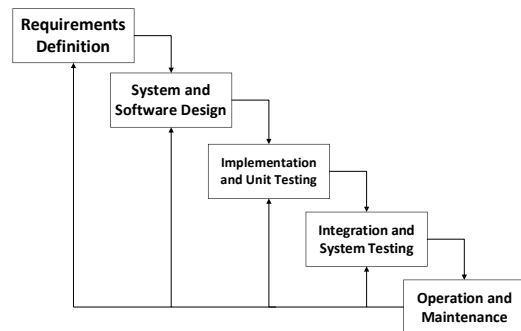
kesulitan menemukan detail informasi kos yang menyebabkan mahasiswa tersebut merasa kebingungan, serta pemilik kos yang kurang efektif dan maksimal dalam mempromosikan kos miliknya yang tidak banyak diketahui oleh mahasiswa (Laila et al., 2019). Maka dari itu diperlukan sistem yang menampung seluruh pemilik kos menjadi satu wadah yang memberikan detail informasi berupa alamat, fasilitas, dan harga yang sesuai, sehingga mahasiswa dapat dengan mudah mencari kos hanya dengan menggunakan sistem tersebut (Nizar, 2021).

Berdasarkan uraian tersebut maka dalam penelitian ini ingin membangun sistem pencarian kos berbasis *web* di wilayah kota Banjarmasin dengan menggunakan metode pengembangan *Waterfall* dan mampu menjadikan solusi terkhusus mahasiswa baru (perantaraan) yang ingin melanjutkan pendidikan ke perguruan tinggi dan mencari tempat tinggal di wilayah kota Banjarmasin, dan membantu pemilik kos untuk mempermudah untuk mempromosikan usaha kos miliknya.

## II. METODOLOGI

### 2.1. Metode Pengembangan Perangkat Lunak

Metode sistem yang digunakan untuk pengembangan adalah air terjun model (*Waterfall Model*). *Waterfall* biasanya disebut dengan model *sequential linier* atau *classic cycle*. Berikut merupakan metode pengembangan *Waterfall* menurut (Pressman, 2008 dalam Wijaya and Astuti, 2019) dapat dilihat pada Gambar 1.



**Gambar 1.** Siklus Pengembangan Metode *Waterfall* (Pressman, 2008 dalam Wijaya and Astuti, 2019)

#### 1. *Requirement Definition*

Proses pencarian kebutuhan diintensifkan dan difokuskan pada *software*. Untuk mengetahui sifat dari program yang akan dibuat, maka para *software engineer* harus mengerti tentang domain informasi dari *software*, misalnya fungsi yang dibutuhkan, *user interface*. Dari 2 aktivitas tersebut (pencarian kebutuhan sistem dan *software*) harus didokumentasikan dan ditunjukkan kepada pelanggan.

#### 2. *System and Software Design*

Proses ini digunakan untuk mengubah kebutuhan di atas menjadi representasi ke dalam bentuk "*blueprint*" *software* sebelum *coding* dimulai. Desain harus dapat mengimplementasikan kebutuhan yang telah disebutkan pada tahap sebelumnya. Seperti 2 aktivitas sebelumnya, maka proses ini juga harus didokumentasikan sebagai konfigurasi dari *software*.

#### 3. *Implementation and Unit Testing*

Untuk dapat dimengerti oleh mesin, dalam hal ini adalah komputer, maka desain tadi harus diubah bentuknya menjadi bentuk yang dapat dimengerti

oleh mesin, yaitu ke dalam bahasa pemrograman melalui proses *coding*. Tahap ini merupakan implementasi dari tahap design yang secara teknis nantinya dikerjakan oleh *programmer*.

#### 4. *Integration and System Testing*

Sesuatu yang dibuat haruslah diujicobakan. Demikian juga dengan *software*. Semua fungsi fungsi *software* harus di uji coba, agar *software* bebas dari *error*, dan hasilnya harus benar-benar sesuai dengan kebutuhan yang sudah didefinisikan sebelumnya.

#### 5. *Operation and Maintenance*

Pemeliharaan suatu *software* diperlukan, termasuk di dalamnya adalah pengembangan, karena *software* yang dibuat tidak selamanya hanya seperti itu. Ketika dijalankan mungkin saja masih ada *error* kecil yang tidak ditemukan sebelumnya, atau ada penambahan fitur-fitur yang belum ada pada *software* tersebut. Pengembangan diperlukan ketika adanya perubahan dari eksternal perusahaan seperti ketika ada pergantian sistem operasi, atau perangkat lainnya (Pressman, 2008 dalam Wijaya and Astuti, 2019).

## 2.2. Teknik Pengumpulan Data

Teknik pengumpulan data adalah cara cara yang dapat digunakan oleh peneliti untuk mengumpulkan data . Beberapa metode yang digunakan sebagai berikut :

#### 1. Observasi

Dalam hal ini proses yang dilakukan yaitu mengamati dan mencari jumlah pemilik kos yang ada di wilayah kota Banjarmasin.

#### 2. Wawancara

Wawancara dilakukan kepada pemilik kos yang ada di wilayah kota

Banjarmasin dengan memberikan masukan dari sistem kos yang akan dirancang dan dibuat serta pengambilan data yang berkaitan dengan kos yang dimiliki.

#### 3. Studi Pustaka

Studi Pustaka dilakukan untuk memperoleh informasi dari jurnal penelitian terdahulu yang berkaitan dengan sistem pencarian kos dengan bahasa pemrograman PHP dan MySql sebagai *database*.

## III. HASIL DAN PEMBAHASAN

Sistem Pencarian Kos Berbasis Web di Wilayah Kota Banjarmasin dibuat dengan tahap sebagai berikut :

### 3.1 *Requirement Definition*

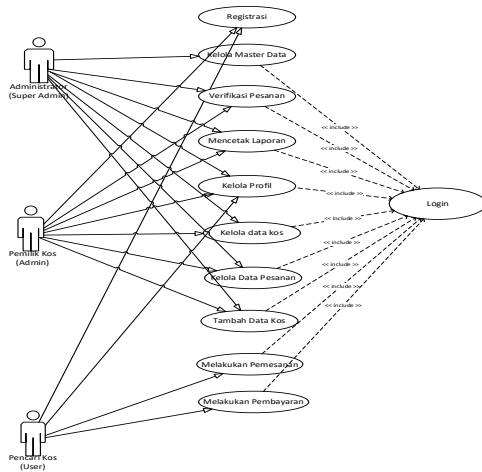
Pada tahap ini dilakukan observasi yang dilakukan oleh peneliti menghasilkan 178 data kos di kota Banjarmasin. Selain observasi peneliti juga melakukan tahap analisis kebutuhan *website* atau fitur-fitur, analisis kebutuhan untuk *software* membangun sistem, dan analisis *hardware* yang digunakan.

### 3.2 *System and Software Design*

Pada tahap ini dilakukan desain sistem dengan menggunakan pemodelan UML (*Unified Modeling Language*). Berikut adalah hasil desain pemodelan sistem.

#### 1. *Use Case Diagram*

*Use Case Diagram* adalah Diagram yang digunakan untuk memodelkan perilaku dinamis dari sistem pada saat beroperasi (*running*). Diagram ini terdiri dari interaksi faktor internal dan eksternal. Agen internal dan eksternal disebut *actor* (Andriyanto, 2022). *Use case diagram* pada sistem pencarian kos berbasis web di wilayah kota Banjarmasin dapat dilihat pada Gambar 2.



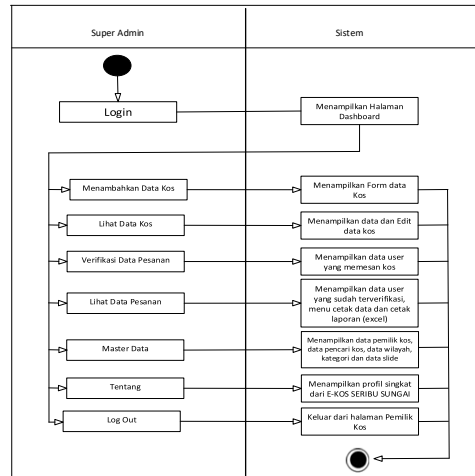
**Gambar2. Use Case Diagram**

Use Case Diagram terdapat 3 aktor yang menjalankan sistem yaitu, *Developer (Super Admin)*, pemilik kos (*Admin*), pencari kos (*user*). *Developer (Super Admin)* memiliki hak akses penuh dalam *website* yang dibuat. Pemilik kos (*admin*) dapat melakukan proses registrasi, jika sudah melakukan proses registrasi maka pemilik kos dapat *login* dan dapat mengakses menu-menu yaitu kelola data kos, melakukan verifikasi pesanan, mengelola data pemesanan dan dapat mencetak laporan. Pencari kos (*user*) dapat melakukan registrasi, jika sudah registrasi maka *user* dapat *login* kedalam sistem dan dapat melakukan pemesanan dan pembayaran.

## 2. Activity Diagram

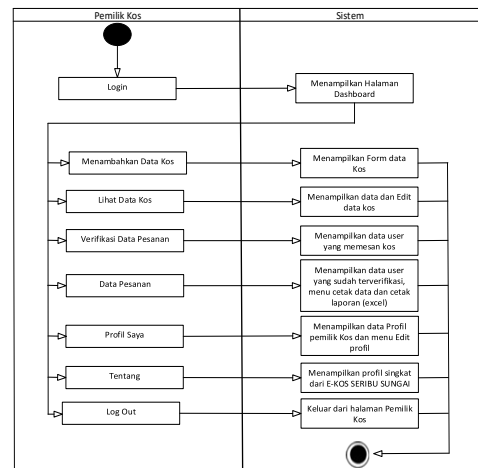
*Activity diagram* atau diagram aktivitas merupakan salah satu pemodelan yang sangat bermanfaat dalam perancangan perangkat lunak. Karena dapat membantu kita dalam memahami proses bisnis sistem secara keseluruhan (Ardiyanto, 2022). Pada tahap ini menjelaskan dan menggambarkan aktivitas yang dilakukan dari 3 aktor dalam pengguna sistem. *Activity diagram*

*super admin* dapat dilihat pada Gambar 3.



**Gambar3. Activity Diagram Super Admin**

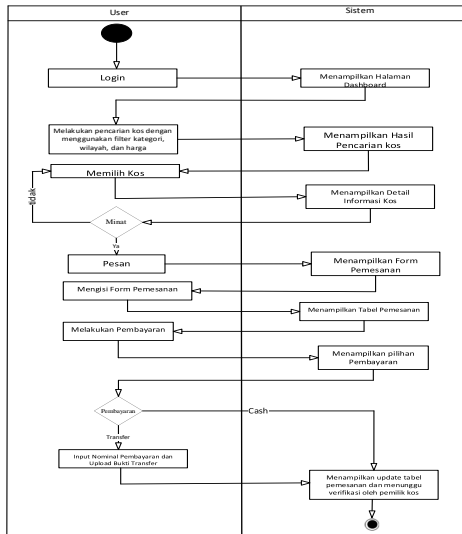
*Activity Diagram Super Admin*, memiliki hak akses penuh untuk mengelola *website*, menjaga privasi data sistem, dan melakukan pekerjaan terkait pengembangan sistem.



**Gambar4. Activity Diagram Admin**

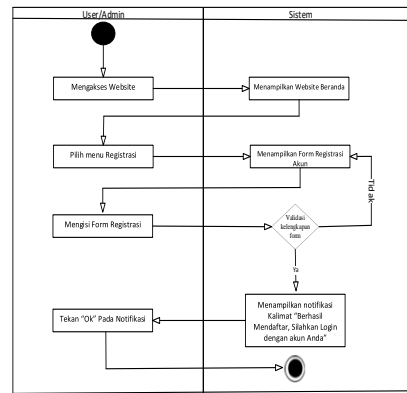
*Activity Diagram Admin*, *admin* melakukan *login* ke sistem kemudian dapat mengakses menu-menu yang disediakan seperti kelola data kos, lihat

data pesanan, verifikasi pesanan kos dan yang terakhir yaitu mencetak laporan sesuai dengan filter waktu yang diinginkan.



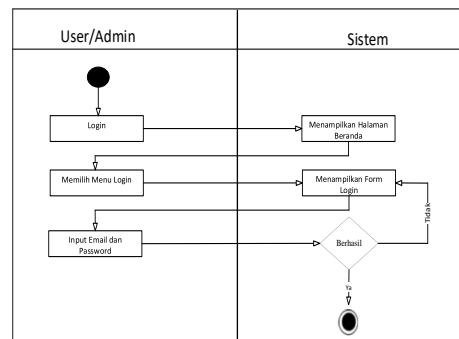
**Gambar5. Activity Diagram User**

Activity Diagram User hal pertama yang dilakukan dengan *login* kedalam sistem dan melakukan proses pencarian kos, setelah pencarian maka *user* berhak memilih kos dan mempertimbangkan. Jika kos yang dipilih tersebut sesuai dengan keinginan maka bisa melakukan ketahap pemesanan, jika tidak maka akan kembali memilih kos. Saat *user* melakukan pemesanan akan ada menu pembayaran, menu pembayaran yang disediakan oleh sistem yaitu menu Transfer dan Cash (Bayar di tempat) saat menggunakan layanan Transfer maka *user* diharapkan untuk upload bukti pembayaran, dan *admin* (Pemilik kos) dapat melakukan konfirmasi terkait pembayaran.



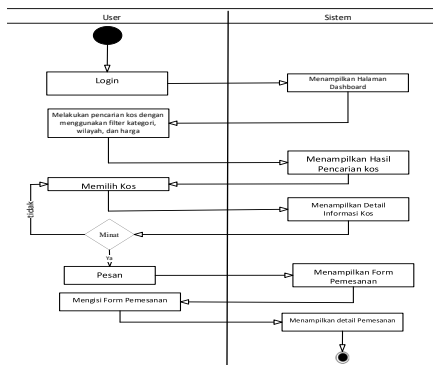
**Gambar6. Activity Diagram Registrasi Admin dan User**

Activity diagram registrasi digunakan pada pemilik kos (*admin*), dan pencari kos (*user*) untuk melakukan pendaftaran sebagai member baik sebagai pencari kos (*user*), atau sebagai pemilik kos (*admin*).



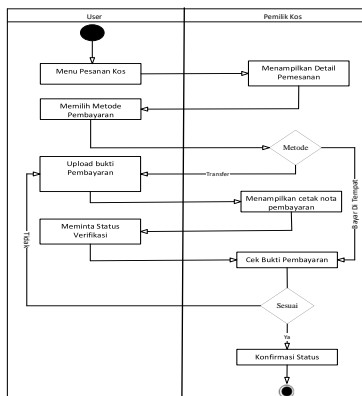
**Gambar7. Activity Diagram Login Admin/User**

Pada Activity Diagram Login, *user/admin* akan ditampilkan menu beranda kemudian memilih menu *login* dan *input email* dan *password* yang sudah terdaftar, jika *login* berhasil maka sistem menampilkan halaman beranda, jika gagal *login* maka akan memasukkan kembali *email* dan *password*.



**Gambar8.** Activity Diagram Pemesanan Kos

User melakukan *login* kedalam sistem dan melakukan proses pencarian dan memilih kos jika sesuai dengan keinginan maka *user* bisa memesan dan mengisi *form* pemesanan. Jika tidak sesuai dengan keinginan maka *user* mencari kembali kos yang diinginkan.



**Gambar9.** Activity Diagram Pembayaran

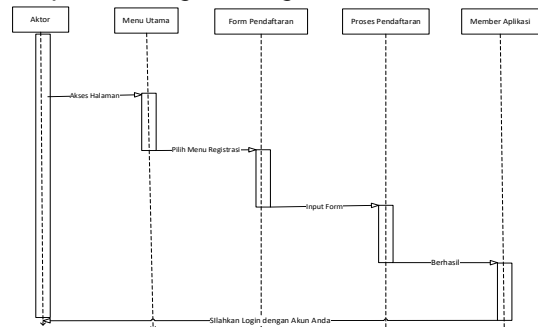
Pada *activity diagram* pembayaran *user* dapat memilih metode pembayaran transfer atau bayar ditempat. Jika menggunakan layanan transfer maka *user* diwajibkan untuk upload bukti pembayaran ke sistem dan menghubungi *admin* (pemilik kos) melalui nomor Whatsapp yang disediakan untuk melakukan verifikasi pesanan, jika bukti

pembayaran yang di *upload* sesuai maka status pesanan pada *user* akan berubah menjadi “Terverifikasi” dan jumlah stok kamar kos akan berkurang. Jika tidak sesuai maka status pesanan akan berubah menjadi “tidak terverifikasi” atau “*upload* ulang bukti dengan jelas”. Jika menggunakan pembayaran *Cash/Bayar* di tempat maka *user* harus datang ke tempat *admin* (pemilik kos) untuk melakukan transaksi, dan *admin* dapat melakukan konfirmasi pesanan pada akun *user* tersebut.

### 3. Sequence Diagram

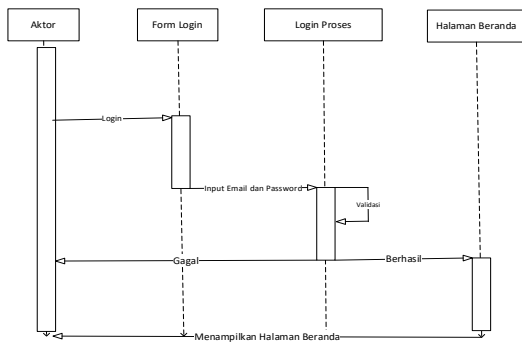
*Sequence diagram* menggambarkan kelakuan/perilaku objek pada *use case* dengan mendeskripsikan waktu hidup objek dan message yang dikirimkan dan diterima antar objek (Murni & Bosker, 2018a).

#### Sequence Diagram Registrasi



**Gambar10.** Sequence Diagram Registrasi Akun

Pada *Sequence Diagram* Registrasi, aktor melakukan akses ke halaman *website* kemudian memilih menu registrasi dan *input* data yang diperlukan dan sistem akan memproses, ketika proses tersebut berhasil maka sistem akan menampilkan tampilan untuk *login* kedalam sistem sebagai *admin/user*.

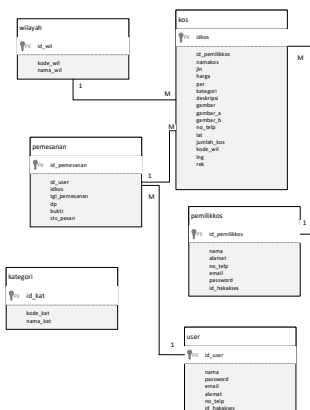


**Gambar11.** Sequence Diagram Login

Pada *Sequence Diagram Login*, Aktor (*admin/user*) menginput *email* dan *password* yang sudah diregistrasikan, kemudian sistem melakukan validasi dan jika benar *email* dan *password* yang di *input* maka sistem akan menampilkan menu beranda (*admin/user*), jika tidak maka akan kembali ke menu awal *login*.

#### 4. Class Diagram

Diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem, Kelas memiliki atribut dan *method* atau operasi (Murni & Bosker, 2018a). Pada *Class Diagram* menampilkan tabel-tabel *database* yang berelasi dengan tabel lain.



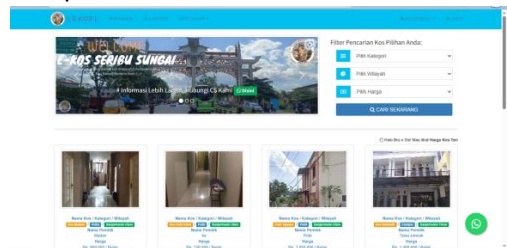
**Gambar12.** Class Diagram

Pada *Class Diagram* menampilkan tabel-tabel *database* yang berelasi dengan tabel lain, seperti pada gambar 12 tabel *user* dan tabel pemesanan memiliki relasi 1 to many, artinya setiap *user* bisa memesan beberapa kos.

### 3.3 Implementation and Unit Testing

Berikut merupakan tampilan dari pengimplementasian (hasil *coding*) sistem yang telah dirancang. Hasil dapat dilihat sebagai berikut.

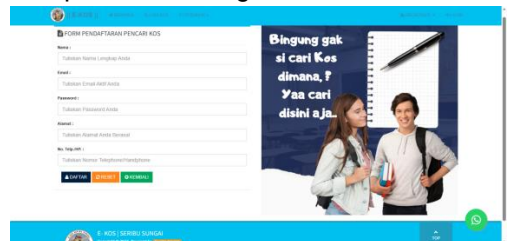
#### 1. Tampilan Menu Beranda



**Gambar13.** Tampilan Beranda Sistem

Menu beranda merupakan tampilan awal saat user mengakses sistem. Menu beranda menampilkan menu diantara registrasi, *login*, cari kos, dan petunjuk bagi calon pemilik atau pencari kos.

#### 2. Tampilan Menu Registrasi



**Gambar14.** Tampilan Menu Registrasi

Menu registrasi difungsikan kepada pencari kos (*user*), dan pemilik kos (*admin*) yang ingin bergabung bersama sebagai *member*.

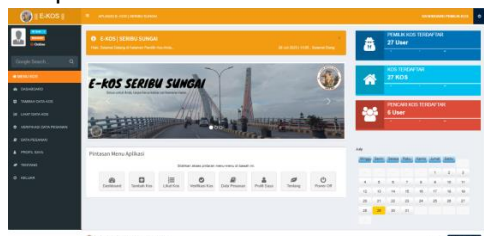
### 3. Tampilan Menu *Login*



**Gambar15.** Menu *Login Admin dan User*

Menu *login* digunakan untuk *admin* ataupun *user* yang ingin masuk kedalam sistem dan mengakses menu menu yang sudah disediakan.

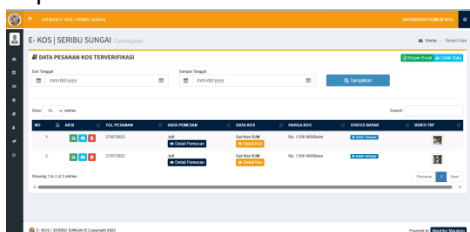
### 4. Tampilan *Dashboard Admin/Pemilik kos*



**Gambar16.** Menu *Dashboard Admin/Pemilik Kos*

Pada *dashboard* pemilik kos, terdapat menu menu yang mampu menunjang aktivitas seperti kelola data kos, verifikasi pesanan, dan lihat data pesanan yang didalamnya terdapat menu cetak laporan sesuai dengan waktu yang diinginkan.

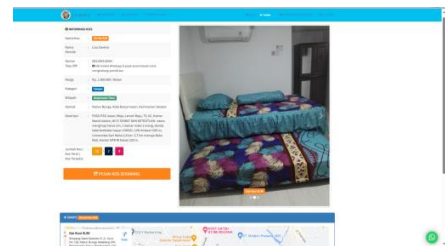
### 5. Tampilan Menu Data Pesanan dan Cetak Laporan



**Gambar17.** Tampilan Halaman Data Pesanan dan Cetak Laporan

Pada halaman data pesanan menampilkan data dari pencari kos yang sudah terverifikasi pembayaran oleh pemilik kos (*admin*), dan terdapat menu cetak laporan terletak dibagian kanan atas beserta sortir waktu yang diinginkan. Hal ini sejalan dengan penelitian Sugiyanto et al (2020) dalam penelitian yang berjudul "Sistem Pencarian dan Penyewaan Tempat Kost di kota Purwokerto Berbasis *Web*" yaitu Menu halaman laporan untuk menampilkan tampilan laporan pemasukan pembayaran dengan jangka waktu yang telah di tentukan.

### 6. Tampilan Detail Informasi Kos



**Gambar18.** Tampilan Halaman Detail Informasi Kos

Tampilan detail informasi kos menyajikan informasi berupa nama pemilik, harga kos, kategori kos, deskripsi kos, jumlah kamar tersedia, alamat yang terhubung dengan *google maps*, dan logo *whatsapp* yang berada dibawah dan jika diklik maka akan otomatis terhubung kepada nomor pemilik kos. Hal ini sejalan dengan penelitian Fitriah & Haryanti (2022) dalam penelitian yang berjudul "Perancangan Sistem Informasi *Website* Sewa Kos (*E-Kos*) Studi Kasus Surabaya" yaitu



Dihalaman ini memuat informasi lebih detail terkait kamar kos yang dipilih yaitu lokasi yang terhubung *google maps* sehingga lebih mudah untuk mencari indekos. Hal serupa yang sejalan dengan penelitian Yuliadi et al (2019) dalam penelitian yang berjudul “Rekayasa aplikasi center Rumah kost berbasis web di Kabupaten Sumbawa” yaitu Terdapat fitur pesan singkat untuk dikirimkan pada pemilik kost berupa Kontak yang tercantum pada aplikasi sudah dapat terhubung langsung ke *whatsapp* pemilik kos.

## 7. Tampilan Menu Pembayaran

**Gambar19.** Menu Pembayaran Transfer

Jika menggunakan layanan transfer maka pencari kos (*user*) diharuskan untuk mengupload bukti pembayaran dengan jelas ke dalam sistem agar memudahkan pemilik kos (*admin*) untuk validasi kemudian dilakukan verifikasi. Hal ini sejalan dengan penelitian Sahidu & Purwanto (2022) dalam penelitian yang berjudul “Sistem Informasi Manajemen Kos Berbasis Web pada Kos Bali” yaitu konfirmasi pembayaran digunakan admin untuk mengkonfirmasi pembayaran jika penyewa telah melakukan transfer kenomor rekening pemilik kos dan melakukan upload bukti pembayaran.

**Gambar20.** Menu Bayar di Tempat (COD)

Jika pencari kos (*user*) menggunakan layanan bayar ditempat maka pencari kos harus mendatangi ke tempat/lokasi pemilik kos yang tertera dan menyerahkan sejumlah uang kemudian pemilik kos (*admin*) akan melakukan konfirmasi di akun pencari kos (*user*).

## 8. Tampilan Status Pembayaran setelah Konfirmasi

**Gambar21.** Tampilan Status Pembayaran setelah konfirmasi

Status pembayaran dapat terverifikasi apabila bukti pembayaran yang di *upload* sesuai dengan harga yang ditentukan, jika bukti tidak sesuai maka pemilik kos akan menolak pesanan dan perintah status tersebut dapat berubah di akun pemesanan pada pencari kos.

## 3.4 Integration and System Testing

Pengujian sistem dilakukan dengan metode pengujian *Black Box* untuk mengetahui jika terdapat kesalahan saat memasukkan data apakah bagian dalam sistem aplikasi akan

menampilkan *error* tersebut, sehingga teknik uji ini menitik beratkan pada fungsi sistem.

Pengujian sistem menggunakan metode *Black Box* dapat dilihat pada Tabel 1.

**Tabel 1.** Pengujian Sistem menggunakan Metode *Black Box*

Data Masukan	Yang Diharapkan	Yang Didapatkan	Kesimpulan
Registrasi terisi lengkap	Berhasil melakukan registrasi	Sistem menerima registrasi dan dapat <i>login</i>	Berhasil
Registrasi tidak terisi lengkap	Registrasi gagal, muncul peringatan diisi semua	Sistem menampilkan pesan 'mohon diisi semua <i>field</i> '	Berhasil
<i>Email</i> dan <i>password</i> sudah diisi dengan benar	Akan masuk ke halaman beranda	Sistem menampilkan pesan <i>login</i> berhasil dan masuk ke halaman beranda	Berhasil
<i>Email</i> dan <i>password</i> tidak terisi dengan lengkap dan salah	Akan muncul peringatan <i>login</i> gagal	Sistem tidak bisa masuk ke beranda dan menampilkan pesan <i>login</i> gagal	Berhasil
Menambah data kos	Akan menyimpan data kos yang ditambahkan	Sistem sukses menyimpan data	Berhasil
Mengupdate data kos	Akan mengupdate data kos	Sistem sukses menyimpan perubahan data	Berhasil
Menghapus data kos	Akan menghapus data kos	Sistem sukses menghapus data kos	Berhasil
Tidak mengisi seluruh data yang diminta oleh sistem	Akan muncul peringatan	Sistem menampilkan pesan "silahkan isi kolom ini"	Berhasil
Mencari data kos dengan pilih kategori	Akan muncul data kos dengan pilihan kategori	Sistem menampilkan data kos dengan kategori yang dipilih	Berhasil
Mencari data kos dengan pilih wilayah	Akan muncul data kos dengan pilihan wilayah	Sistem menampilkan data kos dengan wilayah yang dipilih	Berhasil
Mencari data kos dengan pilih harga	Akan muncul data kos dengan pilihan harga	Sistem menampilkan data kos dengan harga yang dipilih	Berhasil
Pemesanan kos	Akan muncul data pemesanan kos	Sistem menampilkan data pemesanan kos	Berhasil
Pembayaran kos	Akan menampilkan form pembayaran dan dapat mengirimkan bukti pembayaran serta pencetakan nota	sistem menampilkan form pembayaran dan dapat mengirimkan bukti pembayaran serta pencetakan nota	Berhasil
Melakukan cetak Laporan sesuai periode waktu yang ditentukan	Akan mendownload laporan pembukuan dalam bentuk file <i>excel</i>	Sistem mencetak laporan pembukuan sesuai dengan periode waktu yang ditentukan	Berhasil
Tombol <i>icon Whatsapp</i>	Akan terhubung dengan nomor <i>whatsapp</i> admin/pemilik kos	Sistem terhubung ke <i>chat Whatsapp admin</i> /pemilik kos dan menampilkan <i>template chat</i> .	Berhasil

### 3.5 Operation and Maintenance

Pada tahap ini dilakukan *maintenance* atau pengembangan yang dilakukan oleh *developer*, dalam hal ini memperbaiki kekurangan pada menu konfirmasi pesanan oleh pemilik kos. Hal yang dapat dilakukan yaitu dapat bekerja sama dengan pihak Bank agar memudahkan proses pembayaran dengan menggunakan kode *Virtual Account*.

## IV. KESIMPULAN

Sistem Pencarian Kos berbasis Web di wilayah kota Banjarmasin dirancang dengan menggunakan desain sistem *Unified Modeling Language* (UML) yang terdiri dari *Use Case Diagram*, *Activity Diagram*, *Sequence Diagram*, dan *Class Diagram*. Bahasa pemrograman yang digunakan adalah PHP dan *Database MySQL*. Sistem berhasil menampilkan menu-menu yang sesuai dan mendukung bagi *user* atau pencari kos untuk mempermudah proses pencarian, pemesanan dan pembayaran. Bagi *admin* atau pemilik kos, sistem ini mampu menjadi bagian dari sarana promosi kos yang dimiliki dan mampu menunjang proses laporan pembukuan. Berdasarkan pengujian sistem yang dilakukan oleh pengguna yang melibatkan 30 mahasiswa diperoleh hasil 100% bahwa sistem mampu menjalankan sesuai dengan fungsi fungsinya.

## DAFTAR PUSTAKA

- Ardiyanto, S. (2022). Pemodelan Perangkat Lunak Behavior Diagram. Politeknik Manufaktur Negeri Bangka Belitung.
- Fitriah, F., & Haryanti, T. (2022). Perancangan Sistem Informasi Website Sewa Kos (E-Kos) Studi Kasus Surabaya. *SinarFe7*, 5(1), 100–105. Retrieved from <https://journal.fortei7.org/index.php/sinarFe7/article/view/370>
- Laila, I., Maulachela, A.B. & Merawati, N.L.P. Pemetaan Tempat Kos Terdekat dari Universitas Bumigora menggunakan Metoda Location Based Service (LBS). *Jurnal Teknologi Informasi dan Multimedia* (JTIM), 1(1):1-7.
- LLDIKTI. (2023). Lembaga Layanan Pendidikan Tinggi Ristekdikti. <https://lldikti11.ristekdikti.go.id/direktori/cari>
- Murni, M., & Bosker, S. (2018a). *Buku Ajar Sistem Pendukung Keputusan*. CV Rudang Mayang.
- Nizar, C. (2021). Rancang Bangun Sistem Informasi Sewa Rumah Kost (E-Kost) Berbasis Website. *Jurnal Sistem Informasi dan Sains Teknologi*, 3(1), 1-10.
- Sahidu, H., & Purwanto, E. (2022). Sistem Informasi Manajemen Kos Berbasis Web Pada Kos Bali. *Jurnal Ilmiah Sistem Manajemen Informatika Dan Komputerisasi Akuntansi*, 5(1), 5–9. <https://doi.org/10.55501/jisimka.v5i1.70>
- Sugiyanto, S., Irfani, R. S., & Fajarwati, S. (2020). Sistem Pencarian Dan Penyewaan Tempat Kost Di Kota Purwokerto Berbasis Web. *Media Pratama. Jurnal Media Pratama*, 14(2), 1–14. <https://jurnal.inf.co.id/index.php/jurnalmedia Pratama/article/view/92>
- Wijaya, Y. D., & Astuti, M. W. (2019). Sistem Informasi Penjualan Tiket Wisata Berbasis Web Menggunakan Metode Waterfall. *Seminar Nasional Teknologi Informasi Dan Komunikasi*, 273–276.
- Yuliadi, Solihat, N. M., & Herfandi. (2021). *Rekayasa Aplikasi Center Rumah Kost*

Berbasis Web Di Kabupaten Sumbawa.  
Jurnal Manajemen Informatika & Sistem

Informasi), 4(2), 141–148.  
<https://doi.org/10.36595/misi.v4i2.529>