

BAB II

LANDASAN TEORI

2.1 Tinjauan Pustaka

Tinjauan pustaka ditujukan untuk mendapatkan perbandingan dan referensi, juga untuk menghindari asumsi kesamaan dengan penelitian ini. Maka dicantumkan beberapa penelitian sebelumnya.

Penelitian Muslimin *et al.* (2022) yang berjudul “Pembuatan Aplikasi Penjualan Binatang Ternak Berbasis *Android* Di Peternakan Bebek Alfalah Desa Padaulun” bisnis peternakan Bebek Alfalah yang masih belum efektif dan efisien dikarenakan dalam proses jual beli masih dilakukan secara langsung di tempat karena jumlah transaksi penjualan masih relatif rendah karena banyak usaha yang masih melakukan promosi dan penjualan masih menggunakan brosur ataupun baliho, maka penulis memutuskan untuk merancang sebuah aplikasi *e-commerce* untuk usaha pemasaran peternakan tersebut yang nantinya dapat memudahkan peternakan Bebek Alfalah dalam melakukan penjualan dan promosi ternak bebek dan daging yang dapat di harapkan memberikan calon pembeli memudahkan untuk mendapatkan informasi tentang peternakan dan daging bebek tersebut. Kemudian perancangan diimplementasikan dengan menggunakan *Laravel* untuk membangun aplikasi untuk dikelola oleh admin dan menggunakan *flutter* untuk membangun aplikasi untuk digunakan oleh *customer* dalam memesan produk, *backend* menggunakan *PHP* dan *My SQL* sebagai database.

Penelitian (Sudrajat, 2021) yang berjudul “Penggunaan Teknologi *Flutter* dalam Aplikasi” Kedai kopi merupakan tempat untuk para penikmat kopi dan juga sebagai tempat bersantai bersama teman yang lainnya. Namun dimasa pandemi covid-19 seperti saat ini, hal tersebut sudah sangat sulit dilakukan. Untuk itu perlu dibuat sebuah aplikasi yang bisa membantu dalam memberikan pelayanan kepada para penikmat kopi. Aplikasi ini akan menjembatani antara para pemilik kedai kopi dengan para penikmat kopi dalam transaksi dimasa pandemi saat ini. Aplikasi yang dibangun berbasis mobile dengan sistem operasi android untuk pengguna. Metode yang digunakan dalam penyusunan penelitian

ini menggunakan model *Waterfall*. Sistem aplikasi ini menggunakan *Android Studio* dan *Visual Studio Code* sebagai *Integrated Development Environment (IDE)* serta XAMPP sebagai *bundle* untuk *web server (Apache)* dan database (*MySQL*) dengan *tools Unified Modelling Language (UML)*. Aplikasi ini bisa membantu bagi para pemilik kedai kopi untuk memberikan informasi mengenai produk yang dijualnya juga promo serta diskon yang ada dan penikmat kopi bisa tetap menikmati kopi karena adanya kemudahan dalam mengakses aplikasi.

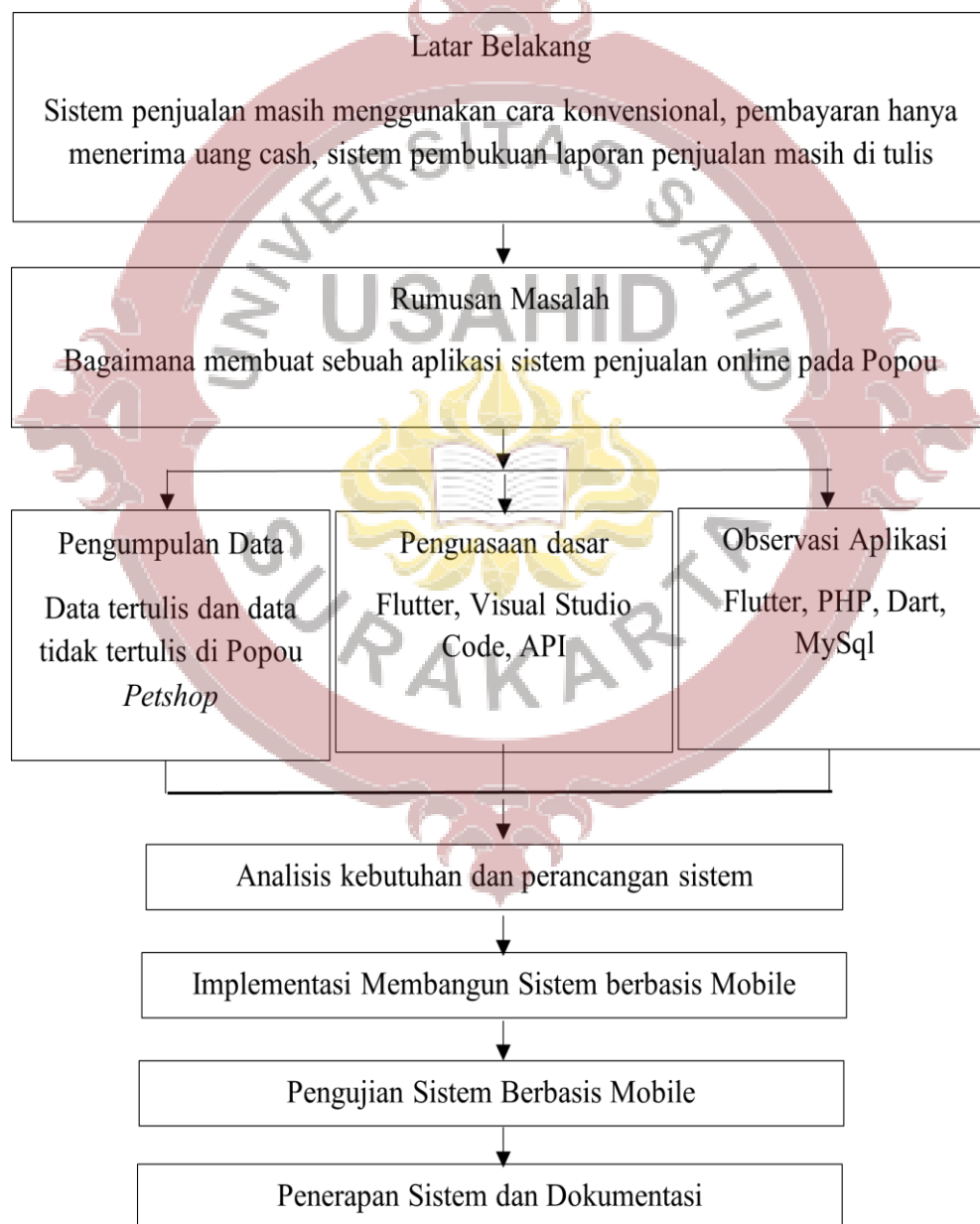
Penelitian (Ainun Najib et al, 2020) yang berjudul “Aplikasi *Point Of Sale Multi Outlet Dan Multi Payment* Berbasis Web Dan *Android*” Pada penelitian ini dilakukan beberapa tahapan dalam perancangan aplikasi *point of sale multi outlet* dan *multi payment* berbasis web dan *android* yaitu pengumpulan data, analisis sistem, perancangan sistem, rancangan basis data, dan rancangan antarmuka. Hasil dari penelitian ini yaitu Aplikasi *point of sale multi outlet* dan *multi payment* berbasis web dan *android* ini dapat memberikan data dan informasi penjualan yang akurat yang dibutuhkan oleh pemilik usaha restoran, dapat menjadi alternatif dalam memberikan kemudahan bagi kasir atau pelayan dalam mencatat pesanan karena cukup melalui aplikasi POS berbasis *android* dan menggunakan ponsel pintar yang menggunakan sistem operasi *android*, membantu manajer atau pemilik usaha untuk melihat laporan barang yang terjual perhari, perbulan dan juga pertahun melalui aplikasi *back office point of sale*.

Pada aplikasi POS ini terdapat tiga metode pembayaran yang akan digunakan, salah satunya dengan memanfaatkan salah satu *fintech* yang sering digunakan masyarakat terutama di daerah pamekasan yaitu *go-pay*. Untuk menggunakan layanan *go-pay* pada aplikasi, *Go-pay* Indonesia sendiri merekomendasikan Api yang disediakan oleh *midtrans*. Jadi penulis akan menggunakan Api *midtrans* untuk mengintegrasikan layanan *go-pay* dengan aplikasi POS *multi outlet* dan *multi payment* berbasis web dan *android*. Aplikasi *back-end point of sale multi outlet dan multi payment* dibangun menggunakan *GraphQL API* dan *framework Angular*, sedangkan *front-end* POS dibangun menggunakan *framework Flutter*. Aplikasi *back-end* POS berbasis web digunakan untuk; mengelola (menambah, mengubah, melihat, dan menghapus)

data *outlet*, pengguna (owner, manajer, kasir), pelanggan, barang, dan saldo lokal; melihat statistik toko (penjualan dan penghasilan); dan melihat daftar transaksi pesanan. Aplikasi *front-end* POS berbasis *android* digunakan untuk melakukan transaksi antara kasir dengan pelanggan

2.2 Kerangka Pemikiran

Berdasarkan permasalahan yang telah dikemukakan dapat dibuat sebuah kerangka pemikiran yang dapat dilihat pada Gambar 2.1:



Gambar 2. 1 Kerangka Pemikiran

Penjelasan dari kerangka pemikiran tersebut adalah :

1. Latar Belakang Masalah.

Sistem penjualan Popou *Petshop* masih menggunakan cara konvensional, pembayaran hanya menerima uang cash, sistem pembukuan laporan penjualan masih di tulis manual pada buku.

2. Rumusan Masalah.

Bagaimana membuat sebuah aplikasi sistem penjualan online pada Popou *Petshop* berbasis *mobile*.

3. Pengumpulan data tertulis dan tidak tertulis

Pada penelitian dilakukan pengumpulan data secara tertulis dan tidak tertulis pada Popou *Petshop*. Pengumpulan data ini menggunakan metode observasi, wawancara dan studi pustaka.

4. Penguasaan dasar

Penguasaan dasar dalam pembuatan aplikasi sistem penjualan online Berbasis *mobile* akan membantu dalam melakukan pembuatan aplikasi sistem penjualan online berbasis *Mobile* meliputi *code editor* Visual Studio Code, API.

5. Observasi Aplikasi

Pada tahap ini aplikasi penjualan online berbasis *mobile* menggunakan teknologi pembangun meliputi *framework* Flutter, Bahasa pemrograman *Dart* dan *PHP*, serta Database *Mysql*.

6. Analisis dan Perancangan aplikasi sistem penjualan online Berbasis *mobile*

Ini merupakan hal yang paling utama dalam penelitian ini, menganalisis sesuai data yang di peroleh. Perancangan sistem menggunakan metode berorientasi objek (*UML*).

7. Implementasi aplikasi sistem penjualan online Berbasis *mobile*.

Proses penerapan analisis dan desain sistem baik itu logika dan tampilan dengan menggunakan *framework* Flutter, *code editor* Visual Studio Code, bahasa pemrograman PHP, bahasa pemrograman Dart, database MySQL, API (Laravel, Cek ongkir, Midtrans).

8. Pengujian aplikasi sistem penjualan online Berbasis *mobile*.

Proses pengujian sistem secara internal dapat dilakukan baik itu secara verifikasi ataupun validasi data. Metode pengujian yang diambil adalah metode pengujian BlackBox dan pengujian beta. Pengujian BlackBox digunakan untuk menguji fungsi-fungsi khusus dari perangkat lunak yang dirancang. Pengujian beta digunakan untuk tanggapan pengguna terhadap aplikasi sistem penjualan online, dengan melakukan kuesioner.

9. Penerapan sistem dan Dokumentasi

Sistem yang sudah diimplementasikan dan diuji coba kemudian diterapkan pada Popou *Petshop* dan setelah itu di buatnya dokumentasi dari keseluruhan kegiatan penyusunan Tugas akhir.

2.3 Landasan Teori

2.3.1 Pemasaran Online

Menurut Yuli dan Marpaung (2020) Promosi online dijelaskan sebagai bentuk kegiatan promosi yang menggunakan internet sebagai sarana utama yang pemanfaatannya dilakukan dengan menggunakan sosial media seperti; *WhatsApp, Messenger, Facebook, dan Instagram*.

2.3.2 Flutter

Chandra dan Tjandra (2020) Flutter adalah SDK untuk pengembangan aplikasi mobile yang dikembangkan oleh Google untuk membangun antarmuka (*user interface*) aplikasi *android* dan *iOS*. Rilis pada juni 2018 sama seperti *react native, framework* ini dapat digunakan untuk membuat atau mengembangkan aplikasi *mobile* yang dapat berjalan pada device *iOS* dan *Android*.

2.3.3 Dart

Menurut Chandra dan Tjandra (2020) Dart merupakan bahasa pemrograman yang dibuat oleh Google untuk menggantikan *Javascript*. Dart menggunakan *static typing* yang berarti sebelum memakai variabel, variabel perlu didefinisikan terlebih dahulu. Dart bisa berjalan pada semua perangkat juga, pada web Dart memakai *dart2js* yang artinya Dart diubah ke *Javascript* agar bisa dimengerti browser. Pada perangkat desktop Dart memakai *dart2aot* yang

mengubah Dart menjadi bahasa mesin. Pada perangkat *mobile* Dart memakai Flutter.

2.3.4 PHP

Menurut Hermanto (2019) PHP diciptakan oleh Rasmus Lerdorf di tahun 1995. Pada masa itu, PHP merupakan kependekan dari *Personal Home Page* atau biasa dikenal sebagai situs personal bernama *Form Interpreted* (FI) memiliki wujud berupa sekumpulan *script* yang digunakan untuk mengolah data formulir dari web. Laravel adalah sebuah *framework* PHP yang dirilis di bawah lisensi MIT, dibangun dengan konsep MVC (*model view controller*). Laravel dirancang untuk meningkatkan kualitas perangkat lunak dengan mengurangi biaya pengembangan awal dan biaya pemeliharaan, dan untuk meningkatkan pengalaman bekerja dengan aplikasi dengan menyediakan sintak yang ekspresif, jelas dan menghemat waktu.

2.3.5 Android

Menurut Huda dan Apriyanto (2019) Android merupakan *toolkit* perangkat lunak open source untuk perangkat *mobile*, android sendiri diciptakan oleh perusahaan Google yang bekerja sama dengan *Open Handset Alliance*. Android adalah sistem operasi berbasis Linux yang dimodifikasi untuk perangkat bergerak (*mobile device*) yang terdiri dari sistem operasi, *middleware*, dan aplikasi-aplikasi. Sistem operasi Android diluncurkan bersamaan dengan dibentuknya organisasi Operasi Handset Alliance tahun 2007.

2.3.6 Visual Studio Code

Menurut Faisal (2017) Visual Studio adalah integrated development enviroment (IDE) yang dikembangkan oleh Microsoft mempermudah software developer mengembangkan aplikasi pada platform milik Microsoft. Visual Studio versi 2015 antara adalah versi stabil terbaru dan sedang dikembangkan Visual Studio 2017 Visual Studio dapat digunakan untuk mengembangkan aplikasi mobile, Web, desktop dan cloud.

2.3.7 Mobile

Menurut Prakarsya,(2019) Menjelaskan bahwa, mobile adalah suatu istilah yang digunakan untuk menggambarkan aplikasi pada piranti berukuran kecil, portable, dan wireless serta mendukung komunikasi. Konsumen menginginkan perangkat yang kecil untuk kenyamanan dan mobilitas mereka dan Perangkat mobile juga hanya menghabiskan sedikit daya dibandingkan dengan mesin desktop.

2.3.8 MySQL

Menurut Enterprise (2017), MySQL adalah RDBMS yang cepat dan mudah digunakan serta sudah banyak dipakai untuk berbagai kebutuhan dikembangkan oleh MySQL Swedia hampir sebagian besar aplikasi website yang ada di internet dikembangkan menggunakan MySQL dan bahasa pemrograman lainnya seperti PHP. MySQL juga adalah database yang cukup terkenal hampir Sebagai sebagian besar aplikasi berbasis website seperti Wordpress Dilengkapi dengan MySQL. Selain itu, MySQL juga ditawarkan juga ditawarkan dalam berbagai versi termasuk versi gratisan.

2.3.9 Unified Modelling Language (UML)

UML merupakan salah satu standart bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis & desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek. UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan mendokumentasikan sistem perangkat lunak (Hasanah, 2020). Jenis – jenis dari UML antara lain :

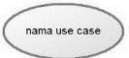
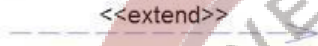
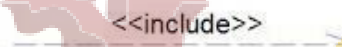
1. Use Case Diagram

Use case diagram adalah teknik untuk merekam persyaratan fungsional sebuah system, menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Use case diagram menekankan kepada “apa” yang diperbuat oleh sistem, dan bukan “bagaimana”. Sebuah use case merepresentasikan sebuah interaksi antara aktor dengan sistem. *Use case* merupakan sebuah pekerjaan tertentu, misalnya login ke sistem, meng-create sebuah daftar belanja, dan sebagainya.

Seorang atau sebuah aktor adalah sebuah entitas dapat berupa manusia atau mesin yang berinteraksi dengan system untuk melakukan pekerjaan-pekerjaan tertentu (Hasanah, 2020).

Berikut adalah simbol-simbol yang ada pada diagram *use case* :

Tabel 2. 1 simbol dan Keterangan Use Case Diagram

Simbol	Keterangan
Use Case 	Abstraksi dan interaksi antara sistem dan aktor.
Ekstend / extend 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri meski tanpa <i>use case</i> tambahan itu arah panah mengarah pada <i>use case</i> yang ditambahkan.
Include 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan membutuhkan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini arah panah <i>include</i> mengarah pada <i>use case</i> yang dipakai (dibutuhkan) atau mengarah pada <i>use case</i> tambahan.
Asosiasi / association 	Penghubung antara aktor dengan <i>Use Case</i>
Actor 	Mewakili peran orang, sistem yang lain, atau alat ketika berkomunikasi dengan <i>Use case</i> .

2. Activity diagram

Menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, decision yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity* diagram juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi (Hasanah, 2020).

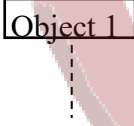
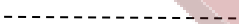
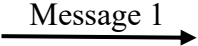
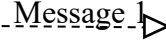
Tabel 2. 2 Simbol dan Keterangan Activity Diagram.

Nama	Simbol	Keterangan
Status Awal		Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
Aktivitas		Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
Percabangan /decision		Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
Status Akhir		Status akhir yang dilakukan system, sebuah diagram aktivitas memiliki sebuah status akhir.

3. Sequence Diagram

Diagram Sequence adalah diagram yang dibuat untuk mengetahui alur dari interaksi antar objek (Hasanah, 2020). Tampilan *sequence* diagram dapat dilihat pada Tabel 2.4.

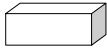
Tabel 2.3 Simbol dan Keterangan Sequence Diagram.

Simbol	Nama	Keterangan
	<i>Actor / Object</i>	Sebuah objek yang berasal dari kelas atau dapat dinamai dengan kelasnya saja. Aktor termasuk objek, Garis putus-putus menunjukkan garis hidup suatu objek.
	<i>lifeline</i>	Menyatakan kehidupan suatu objek.
	Pesan	Interaksi antara satu objek dengan objek lainnya. Objek dapat mengirimkan pesan ke objek lain. Interaksi antar objek ditunjukkan pada bagian operasi pada diagram kelas.
	<i>Return</i>	Pesan kembalian dari komunikasi antar objek
	Aktivasi	Menunjukkan masa hidup dari objek.

2.3.10 Component Diagram

Menurut (Sukanto & Shalahudin, 2016), Component Diagram dibuat untuk menunjukkan struktur dan ketergantungan antara sekumpulan komponen dalam suatu sistem. Tabel simbol yang ada di dalam Component Diagram ditunjukkan pada Tabel 2.5 di bawah ini.

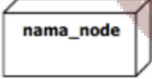
Tabel 2.5 Simbol Component Diagram

SIMBOL	NAMA	KETERANGAN
	Node	Sumber daya yang digunakan pada saat aplikasi dijalankan.
	Association	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan multiplicity.
	Component	Menjelaskan perangkat keras atau objek didalam sistem.

2.3.11 Deployment Diagram

Menurut (Sukanto & Shalahudin, 2016), Deployment Diagram menunjukkan konfigurasi komponen dalam eksekusi aplikasi. Deployment Diagram juga dapat digunakan untuk memodelkan hal-hal seperti tambahan sistem dan sistem klien/server. Tabel simbol yang ada di dalam Deployment Diagram ditunjukkan pada Tabel 2.6 di bawah ini.

Tabel 2.1 Simbol Deployment Diagram

Simbol	Deskripsi
	Sebuah bungkusan dari satu atau lebih node.
	Mengacu pada hardware, software yang tidak dibuat sendiri, jika di dalam node disertakan komponen untuk mmengkonsistenkan rancangan maka komponen yang diikutsertakan harus sesuai dengan komponen yang didefinisikan sebelumnya pada component diagram.
Depedency 	Keberuntungan antar node, arah panah mengarah pada node yang dipakai.
Generalisasi 	Relasi antar node.

2.3.12 Teori Pengujian Sistem

a) Black Box Testing

Black Box Testing atau yang sering dikenal dengan sebutan pengujian spesifikasi fungsional merupakan metode pengujian untuk mengetahui apakah fungsi-fungsi masukan dan keluaran dari perangkat lunak sesuai dengan

spesifikasi yang dibutuhkan. Dalam pengujian ini, tester menyadari apa yang harus dilakukan oleh program tetapi tidak memiliki pengetahuan tentang bagaimana melakukannya (Hasanah, 2020)

b) Beta

Pengujian Beta dilakukan setelah pengujian awal selesai, Pengujian Beta meliputi pengujian aplikasi ke pengguna secara langsung melalui beberapa responden (Saputra *et al.* 2022). Metode ini biasanya melibatkan penyebaran perangkat lunak ke sekelompok orang dan User. Kemudian, kepada sekelompok orang tersebut diberikan sebuah kuesioner yang berisi pernyataan untuk dijawab. Pengumpulan data dilakukan selama pengujian beta melalui Google Form. Formulir ini dibagikan kepada berbagai *customer*, karyawan dan masyarakat umum, untuk memperoleh masukan yang beragam. Data yang dikumpulkan mencakup umpan balik kualitatif dan kuantitatif untuk memberikan gambaran yang lebih menyeluruh tentang performa produk selama fase pengujian (Susanto *et al.*, 2022).

Rumus pengujian Beta :

Tabel 2.4 Skala Linkert

Tingkat Kepuasan	Skala
Tidak Baik (TB)	1
Kurang Baik (KB)	2
Baik (B)	3
Sangat Baik (SB)	4

Setelah melakukan pengujian maka tahap selanjutnya melakukan analisa hasil pengujian. Untuk pengujian beta analisa dilakukan dengan melakukan perhitungan hasil survei dari pertanyaan yang sudah diisi oleh responden. Dimana perhitungan kuesioner diperoleh dengan rumus sebagai berikut:

$$Y = \frac{X}{\text{Skor Ideal}} \times 100\% \quad X = \sum(N \times R)$$

Skor Ideal Skor Ideal = nilai Linkert tertinggi x jumlah responden

Keterangan:

Y = nilai prosentase yang dicari

X = jumlah dari hasil perkalian nilai setiap jawaban dengan responden

N = nilai dari setiap jawaban

R = jumlah responden