

## BAB II

### LANDASAN TEORI

#### 2.1. Kajian Terhadap Penelitian Yang Terkait Sebelumnya

Pada penelitian (Mardiyono, dkk, 2022) menjelaskan bahwa prediksi (forecasting) merupakan alat bantu yang sangat penting dalam perencanaan yang efektif dan efisien khususnya dalam bidang ekonomi dan organisasi bisnis dalam setiap pengambilan keputusan yang sangat signifikan. Saat ini permasalahan yang paling sering terjadi dalam penerimaan siswa baru adalah tidak tercapainya kuota siswa yang mendaftar. Sehingga masih terdapat bangku kosong di banyak sekali sekolah. Pada Penelitian ini menggunakan metode *Autoregressive* untuk melakukan prediksi terhadap jumlah pendaftar pada periode yang akan mendatang. Hasil pengujian menggunakan MAPE (Mean Absolute Percentage Error) sebesar 19,94%.

Pada penelitian (Hartanto, dkk, 2021) menjelaskan bahwa teknik prediksi merupakan suatu sistem yang mempunyai tujuan untuk melakukan perencanaan yang lebih efektif dalam melakukan proses bisnis dalam suatu perusahaan. Saat ini permasalahan yang paling sering terjadi dalam penerimaan siswa baru adalah tidak tercapainya kuota siswa yang mendaftar. Sehingga masih terdapat bangku kosong di banyak sekali sekolah. Pada Penelitian ini menggunakan metode *Autoregressive* untuk melakukan prediksi terhadap jumlah pendaftar pada periode yang akan mendatang. Hasil pengujian menggunakan MAPE (Mean Absolute Percentage Error) sebesar 18,18%.

Pada penelitian (Harsiti, dkk, 2022) membuat prediksi terhadap pengelolaan persediaan obat. Pengelolaan persediaan obat pada klinik ini dilakukan setiap satu minggu sekali dengan pencatatan manual. Transaksi penjualan obat pun dicatat dalam pembukuan biasa yaitu dicatat dalam buku besar, sehingga proses kontrol terhadap persediaan obat belum dapat dilakukan dengan baik, tidak dapat dilakukan proses prediksi terhadap persediaan obat yang harus dibeli untuk periode berikutnya. Hal ini disebabkan karena catatan transaksi penjualan yang tersedia hanya digunakan sebagai catatan dan arsip saja dan belum digunakan sebagai data yang dapat mendukung proses prediksi. Permasalahan ini dapat diatasi dengan mengimplementasikan metode regresi linier sederhana yang dapat digunakan sebagai salah satu pendekatan untuk menghasilkan informasi berupa hasil prediksi untuk persediaan obat dalam proses penjualan obat jenis tablet. Berdasarkan perhitungan dengan menggunakan metode regresi linier diperoleh akurasi sebesar 98,505% yang artinya metode ini dapat diimplementasikan dalam membuat prediksi stok obat jenis tablet (ibu profen) pada Klinik Teluk.

Pada penelitian (Baihaqi, dkk, 2019) menerangkan bahwa seiring bertambahnya jumlah populasi manusia dan keadaan perekonomian yang semakin maju, maka tingkat kesadaran masyarakat terhadap kesehatan semakin meningkat. Maka diperlukan sebuah metode untuk memprediksi jumlah kunjungan pasien pada RSUD Cilacap. Tujuan dari penelitian ini adalah untuk mengetahui hasil prediksi jumlah kunjungan pasien pada RSUD Cilacap menggunakan metode regresi linier. Y. Pada penelitian ini Metode prediksi regresi linier dapat menghasilkan prediksi dengan beberapa kriteria nilai error MAPE, dimana terdapat 26 model prediksi regresi linier yang memiliki nilai error kurang dari

20% artinya mempunyai akurasi sebesar 80%. Akan tetapi, terdapat 3 model prediksi regresi linier yang masuk dalam kategori buruk yaitu nilai errornya lebih dari 50%, dan terdapat 1 model prediksi regresi linier yang termasuk dalam kategori cukup atau mempunyai nilai error sebesar 20% sampai 50%.

## **2.2. Prediksi**

Prediksi atau peramalan merupakan kegiatan yang memperkirakan kejadian di masa yang akan datang. Prediksi (forecasting) merupakan sebuah alat yang sangat dibutuhkan dalam perencanaan yang efektif dan efisien khususnya dalam bidang ekonomi dan organisasi bisnis ketika terjadi pengambilan keputusan-keputusan penting. Prediksi menjadi dasar bagi perencanaan jangka pendek maupun jangka panjang bagi perusahaan. Dalam area fungsional keuangan, prediksi memberikan dasar dalam menentukan anggaran dan pengendalian biaya. Pada bagian pemasaran, prediksi dibutuhkan untuk merencanakan penjualan produk baru, kompensasi tenaga dan beberapa keputusan penting lainnya. Pada bagian produksi dan operasi menggunakan data-data untuk perencanaan kapasitas, fasilitas, produksi, penjadwalan, dan pengendalian persediaan (Hartanto, dkk, 2021).

## **2.3. Siswa Baru**

Penerimaan merupakan penyambutan, proses, perbuatan atau sikap terhadap seseorang. Siswa merupakan pelajar pada akademi atau perguruan tinggi. Baru merupakan suatu hal belum ada sebelumnya (Situngkir, dkk, 2020). Sedangkan menurut pasal 1 ayat 4 UU RI No.20 tahun 2003 tentang Sistem Pendidikan Nasional, peserta didik adalah anggota masyarakat yang berusaha

mengembangkan dirinya melalui proses pendidikan pada jalur jenjang dan jenis pendidikan tertentu (Widiati & Putri, 2018).

Dari definisi tersebut disimpulkan bahwa siswa baru adalah anggota masyarakat yang mengembangkan dirinya melalui proses pendidikan yang telah melewati proses penyambutan atau pendaftaran yang sebelumnya bukan merupakan pelajar pada akademi.

#### **2.4. *Autoregressive***

Konsep dari metode perkiraan atau peramalan *Autoregressive* (AR) adalah suatu teknik model rata-rata yang dapat menggambarkan hasil dari pengamatan pada waktu  $t$  yang dipengaruhi oleh nilai-nilai dari proses pengamatan sepanjang  $p$  periode sebelumnya (Hartanto, dkk, 2021). Pemodelan *Autoregressive* adalah teknik stokastik banyak digunakan untuk menganalisis dan memprediksi deret waktu. *Autoregressive* memiliki tanda menyajikan pola perilaku periodik (Tavares, dkk, 2019).

Model *autoregressive* (AR) digunakan dalam pemrosesan statistik ketika nilai deret waktu diregresi pada nilai sebelumnya dari deret waktu yang sama. Model *autoregressive* menetapkan bahwa variabel output tergantung secara linier pada nilai sebelumnya dan pada istilah stokastik sebagai istilah yang dapat diprediksi secara tidak sempurna (Ulloa, 2018). Model stokastik yang bisa sangat berguna dalam representasi tertentu deret yang terjadi secara praktis di mana nilai saat ini dari proses dinyatakan sebagai agregat linier terbatas dari nilai sebelumnya. Model ini paling populer digunakan dalam analisis deret waktu, sebuah struktur model yang elegan memfasilitasi estimasi parameter dengan kuadrat terkecil atau metode sederhana lainnya (Balakrishna, 2021). Jika kita

menyatakan nilai-nilai dari suatu proses pada jarak waktu yang sama  $t, t-1, t-2, \dots$  oleh  $X_t, X_{t-1}, X_{t-2}, \dots$ , maka  $X_t$  dapat dijelaskan dengan ekspresi:

Model *autoregressive* orde  $p$  secara umum dapat dituliskan sebagai berikut (Mardiyono, dkk, 2022):

$$Y_t = \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \dots + \phi_p Y_{t-p} + e_t \quad (2.1)$$

Atau secara ekuivalen, dapat dinyatakan dengan model (Balakrishna, 2021).

$$\varphi(B)X_t = a_t \text{ with } \varphi(B) = 1 - \phi_1 B - \phi_2 B^2 - \dots - \phi_p B^p \quad (2.2)$$

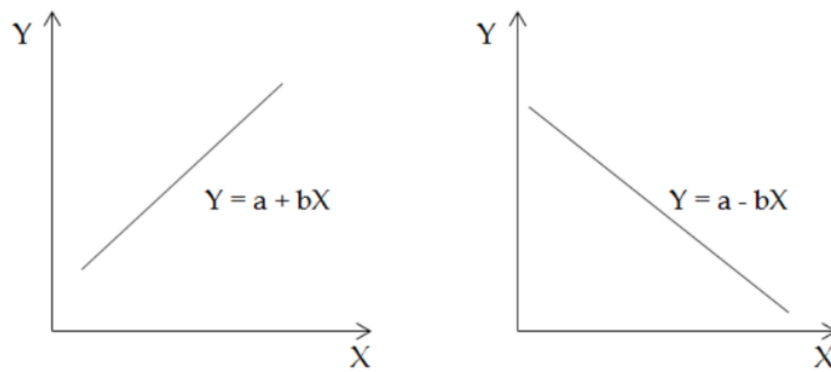
atau atur ulang dengan mengurangkan  $y_{t-1}$  (Zhang, dkk, 2020):

$$\Delta y_t = a + \delta t + \phi_1 y_{t-1} + \phi_2 \Delta y_{t-1} + \dots + \phi_{p-1} \Delta y_{t-p+1} + e_t \quad (2.3)$$

Model *autoregressive* adalah metode pemodelan klasik atau standar yang digunakan pada data deret waktu. Bagian "otomatis" dari *autoregressive* mengacu pada fakta bahwa model ini memanfaatkan korelasi deret waktu dengan dirinya sendiri di masa lalu, oleh karena itu disebut *autoregressive* (Bateman, dkk, 2019).

## 2.5. Regresi Linier Sederhana

Model regresi linier sering digunakan untuk mengeksplorasi hubungan antara hasil yang berkelanjutan dan independent variabel, hasil biner juga mungkin digunakan. Untuk memenuhi asumsi normalitas, penelitian menggunakan metode ini sering melakukan transformasi hasil dengan data yang luar biasa. Misalnya, menggunakan informasi di lebih dari 100.000 mata pelajaran (Schmidt & Finan, 2018). Analisa regresi menganalisa bagaimana hubungan antara dua variabel atau lebih, dikatakan regresi linier karena jika diaplikasikan dalam bentuk pertemuan sumbu  $x$  dan  $y$  akan membentuk kurva garis lurus seperti pada gambar 2.1 (Darma, 2020)



Gambar 2. 1 Regresi Linier

Sumber: (Ayuni & Fitriyah, 2019)

Regresi Linear Sederhana adalah salah satu metode *least square* yang berfungsi untuk menguji hubungan sebab akibat antara Variabel Penyebab (X) terhadap Variabel Akibat (Y). Variabel Penyebab pada umumnya dilambangkan dengan X sedangkan Variabel akibat dilambangkan dengan Y (Suyono, 2018). Setelah mendapatkan data, dari data yang tersedia, hal pertama yang harus ditentukan adalah untuk mengidentifikasi faktor penyebab (disebut X) dan efeknya variabel (disebut Y). X dan persentase dari error sebagai Y. Setelah itu, kami menghitung persamaan regresi linier sederhana (Nasution & Harahap, 2020).

Persamaan umum metode regresi linier sederhana adalah (Harsiti, dkk, 2022):

$$y = a + bx \tag{2.3}$$

Keterangan:

a = Konstanta

b = Koefisien regresi

Y= Variabel dependen (variabel tak bebas)

X = Variabel independen (variabel bebas)

Menghitung konstanta (a) :

$$a = \frac{(\sum y)(\sum x^2) - (\sum x)(\sum xy)}{n(\sum x^2) - (\sum x)^2} \quad (2.4)$$

Menghitung Koefisien

$$b = \frac{n(\sum xy) - (\sum x)(\sum y)}{n(\sum x^2) - (\sum x)^2} \quad (2.5)$$

Titik awal alami untuk mempelajari tentang analisis data statistik adalah dengan sampel independent dan data yang terdistribusi secara identik (selanjutnya i.i.d.), misalkan  $Y = (Y_1, \dots, Y_n)$ . Itu model regresi linier menyederhanakan asumsi identik dan independen dengan (i) memungkinkan berarti  $Y_i$  bergantung, secara linier, pada sekumpulan variabel lain, (ii) memungkinkan  $Y_i$  memiliki varians yang berbeda, dan (iii) memungkinkan korelasi antara  $Y_i$ . Model regresi linier tidak hanya penting secara mendasar dalam berbagai besar kuantitatif disiplin ilmu, tetapi juga merupakan dasar dari sejumlah besar model yang lebih kompleks, seperti yang muncul dalam studi data panel, analisis *time series* dan *generalized linear models* (GLIM), teknik analisis data yang lebih maju (sering disebut sekarang sebagai algoritme) juga memiliki akar dalam regresi, seperti *least absolute shrinkage and selection operator* (LASSO), *elastic net*, dan *least angle regression* (LARS). Metode seperti itu sering digunakan dalam machine learning (Paolella, 2019).

Rumus regresi linier sederhana (SLR) (Olive, 2017) :

$$Y_i\beta_1 + \beta_2X_i + e_i = \alpha + \beta X_i + e_i \quad (2.6)$$

dimana  $e_i$  adalah iid dengan  $E(e_i) = 0$  dan  $VAR(e_i) = 2$  untuk  $i = 1, \dots, n$ .  $Y_i$  dan  $e_i$  adalah variabel acak sedangkan  $X_i$  diperlakukan sebagai diketahui konstanta. Parameter  $\beta_1$ ,  $\beta_2$ , dan  $\beta_0$  adalah konstanta yang tidak diketahui yang perlu diperkirakan. (Jika  $X_i$  adalah variabel acak, maka modelnya adalah tergantung pada  $X_i$  asalkan kesalahan  $e_i$  tidak tergantung pada  $X_i$  Oleh karena itu,  $X_i$  masih diperlakukan sebagai konstanta.) Model SLR (*Simple Linier Regression*) adalah kasus khusus dari model MLR dengan  $p = 2$ ,  $x_{i,1} = 1$ , dan  $x_{i,2} = X_i$ . Model SLR normal menambahkan asumsi bahwa  $e_i$  adalah iid  $N(0, 2)$ . Artinya, distribusi kesalahan normal dengan rata-rata nol dan varians konstan 2. Variabel respon  $Y$  adalah variabel yang Anda inginkan untuk memprediksi sedangkan variabel prediktor  $X$  adalah variabel yang digunakan untuk memprediksi tanggapan. Untuk SLR,  $E(Y_i) = \beta_0 + \beta_1 + 2X_i$  dan garis  $E(Y) = \beta_0 + \beta_1 + 2X$  adalah fungsi regresi.  $VAR(Y_i) = 2$  (Olive, 2017).

## 2.6. Time Series

*Time series* (deret waktu) merupakan sekumpulan data yang diurutkan berdasarkan waktu data tersebut direkam. Frekuensi waktu yang digunakan dapat berupa tahunan, bulanan, mingguan, harian, hingga per jam. Sebuah time series dapat memiliki satu fitur (*univariate*) maupun beberapa fitur (*multivariate*) (Seruni, dkk, 2020). *Time series* pada umumnya memiliki tiga pola, yaitu:

1. Trend

Time series yang datanya cenderung terus naik dalam jangka waktu yang panjang.

2. Seasonal

Time series yang nilai datanya dipengaruhi oleh faktor musiman, di mana perubahan pola selalu terjadi dalam frekuensi yang tetap.



### 3. Cyclic





Time series yang bersifat cyclic datanya memiliki siklus tidak tentu di mana data tersebut dapat naik atau turun dalam frekuensi yang tidak tetap. Kita sering tertarik pada pengamatan yang dilakukan pada waktu-waktu sebelumnya, yang disebut waktu jeda atau lag. Waktu di masa lalu adalah negatif relatif terhadap waktu saat ini. Misalnya waktu sebelumnya adalah  $t-1$  dan waktu sebelum itu adalah  $t-2$ . Pengamatan pada waktu tersebut adalah  $obs(t-1)$  dan  $obs(t-2)$  masing-masing. Waktu di masa depan adalah apa yang kami minati untuk diramalkan dan positif relatif terhadap waktu saat ini. Misalnya waktu berikutnya adalah  $t+1$  dan waktu setelah itu adalah  $t+2$ . Pengamatan pada waktu-waktu tersebut berturut-turut adalah  $obs(t+1)$  dan  $obs(t+2)$ . Untuk mempermudah, kita sering membuang notasi  $obs(t)$  dan menggunakan  $t+1$  sebagai gantinya dan menganggap kita adalah berbicara tentang pengamatan pada waktu daripada indeks waktu itu sendiri. Selain itu, kami dapat merujuk pada pengamatan pada lag dengan singkatan seperti lag 10 atau  $lag=10$  yang akan menjadi sama dengan  $t-10$  (Brownlee, 2020).



### 2.7. UML

*Unified Modelling Language* (UML) merupakan notasi grafis berupa meta-model yang dapat digunakan untuk menggambarkan dan mendesain sistem perangkat lunak, khususnya sistem pemrograman yang berorientasi objek. Dengan menggunakan *Unified Modelling Language* (UML), pendefinisian masalah dapat dilakukan dengan notasi grafis, sehingga memudahkan dalam pemahaman sistem kompleks. Dalam penelitian ini, penulis hanya menggunakan beberapa diagram

Unified Modelling Language (UML) yaitu *use case diagram*, *activity diagram*, *sequence diagram*, dan *class diagram* (Rosa & Shalahuddin, 2019).

Tabel 2. 1. Simbol-simbol *Use Case Diagram* (Ayu & Permatasari, 2018)




No	Simbol	Keterangan
1	<p><i>Use Case</i></p> 	<p>Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya di-nyatakan dengan meng-gunakan kata kerja di awal frase nama <i>use case</i>.</p>
2	<p>Aktor / <i>Actor</i></p>  <p>Nama Aktor</p>	<p>Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.</p>
3	<p>Asosiasi / <i>Association</i></p> 	<p>Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.</p>
4	<p>Ekstensi / <i>Extend</i></p>  <p>&lt;&lt;extend&gt;&gt;</p>	<p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu; mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek; biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan.</p>


No	Simbol	Keterangan
5	Generalisasi / <i>Generalization</i> 	Hubungan generalisasi dan spesialisasi (umum- khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
6	Menggunakan / <i>Include / uses</i>  <<include>>	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini. <i>Include</i> berarti <i>use case</i> yang ditambahkan akan selalu dipanggil saat <i>usecase</i> tambahan dijalankan.

### 2.7.1. Activity Diagram

*Activity diagram* menggambarkan berbagai alur aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alur berawal, dan *decision* yang mungkin terjadi, dan bagaimana alur berakhir. Adapun simbol dari *activity diagram* dijelaskan pada tabel 2. 2.

Tabel 2. 2. Simbol *Activity Diagram* (Ayu & Permatasari, 2018)



No	Gambar	Keterangan
1.	 <i>Activity</i>	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja
2.	 <i>Decision</i>	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
3.	 Status Awal	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki status awal.


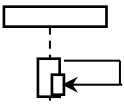
No	Gambar	Keterangan
4.	 Status Akhir	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
5.	<i>Join</i>	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.

### 2.7.2. Sequence Diagram

*Sequence diagram* menggambarkan interaksi antar objek di dalam dan disekitar sistem (termasuk pengguna dan *display*) berupa *message* yang digambarkan terhadap waktu. *Sequence diagram* terdiri atas dimensi vertikal (waktu) dan dimensi horizontal (objek-objek yang terkait). Setiap pesan yang dikirimkan bisa memberikan respon (*return*) relatif pada skenario yang dirancang di *usecase diagram*. Interaksi yang terjadi bisa bersifat instansiasi sebuah objek maupun *static method* dari sebuah *class*. Berikut ini adalah simbol-simbol dari *sequence diagram* dijelaskan pada tabel 2. 3.

Tabel 2. 3. Simbol *Sequence Diagram* (Ayu & Permatasari, 2018)

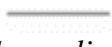

No	Gambar	Keterangan
1.	 Aktor	Aktor adalah pengguna sistem, pengguna dapat berarti manusia, mesin atau sistem lain atau subsistem dari model apapun yang berinteraksi dengan sistem dari <i>boundary system</i>
2.		Peserta individu dalam interaksi (yaitu jalur hidup yang tidak dapat memiliki multiplisitas)






No	Gambar	Keterangan
	<i>Lifeline</i>	
3.	 <i>Message</i>	Menunjukkan aliran informasi atau kendali transaksi antar elemen
4.	 <i>Self-Message</i>	Mencerminkan proses baru atau metode pemanggilan operasi <i>lifeline</i> . Ini adalah spesifikasi pesan biasanya dalam <i>sequence diagram</i>

### 2.7.3. Class Diagram

*Class diagram* menggambarkan hubungan antara obyek-obyek yang terlibat didalam sistem, *class diagram* dapat menunjukkan operasi maupun properti didalam sebuah objek. Berikut ini adalah simbol-simbol dari *class diagram* dijelaskan pada tabel 2. 4.

Tabel 2. 4. Simbol *Class Diagram* (Ayu & Permatasari, 2018)

Gambar	Keterangan
 <i>Generalization</i>	Hubungan dimana objek anak berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk.
 <i>Navy</i> <i>Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.

 <i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
 <i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu <i>actor</i> .
 <i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
 <i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri.
 <i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.

## 2.8. MAPE

MAPE merupakan metode untuk menghitung nilai error dalam bentuk persentase (Seruni, dkk, 2020). MAPE merupakan salah satu pengujian kesalahan hasil peramalan yang dihitung dengan cara membagi kesalahan absolut tiap periode dengan nilai observasi yang nyata pada periode tersebut dan merata-ratakan persentase kesalahan absolut tersebut. Rumus dari MAPE dapat dilihat pada persamaan (2.5) (Wardhani, et al., 2022).

$$MAPE = \frac{\sum_{t=1}^n \left| \frac{Y-Y'}{Y} \right|}{n} \cdot 100 \quad (2.5)$$

Keterangan :

Y = Nilai aktual

Y' = Nilai prediksi

n = Jumlah data

Dalam perhitungan MAPE, semua nilai pengurangan antara nilai faktual dan peramalan diabsolutkan atau dijadikan positif. Hasil akhir perhitungan MAPE dapat ditentukan berdasarkan *range* pada tabel 2.5.

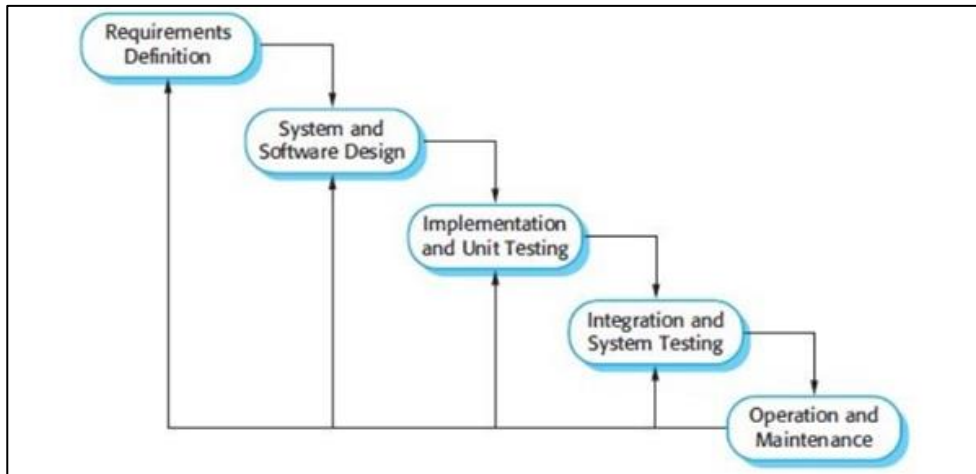
Tabel 2. 5. *Range* Perhitungan MAPE

(Sumber: Maricar, 2019)

<b>Range MAPE</b>	<b>Keterangan</b>
<10%	Model peramalan sangat baik
10-20%	Model peramalan baik
20-50%	Model peramalan cukup
>50%	Model peramalan buruk

## 2.9. *Waterfall*

*Waterfall* atau air terjun merupakan metode yang dikembangkan untuk pengembangan perangkat lunak, membuat perangkat lunak. Model ini berkembang secara sistematis dari satu tahap ke tahap lain dalam mode seperti air terjun. Model *waterfall* ini mengusulkan suatu pendekatan kepada pengembangan software yang sistematis dan sekuensial mulai dari tingkat kemajuan sistem pada seluruh analisis, desain, kode, pengujian, dan pemeliharaan (Wijaya, 2019)



Gambar 2. 2 Tahapan Waterfall

Sumber (Tabrani & Aghniya, 2019)

Gambar 2.2 di atas merupakan tahapan dari metode *Waterfall*. Metode *waterfall* terbagi atas lima tahap (Tabrani & Aghniya, 2019), yaitu:

1. *Requirements 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-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 diujicobakan, 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.