**PROYEKSI CURAH HUJAN DAERAH PADANG PARIAMAN  
MENGGUNAKAN DEEP LEARNING DENGAN  
METODE LONG SHORT-TERM MEMORY**

**SKRIPSI**

*Untuk Memenuhi Salah Satu Syarat*

*Memperoleh Gelar Sarjana Komputer*

**Program Studi : Teknik Informatika  
Jenjang Pendidikan : Strata 1 (S1)**



**OLEH :**

**EDO SULAIMAN**  
**NIM. 18101152630092**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS ILMU KOMPUTER**

**UNIVERSITAS PUTRA INDONESIA “YPTK” PADANG**

**2021**

# BAB I PENDAHULUAN

## Latar Belakang

Dewasa ini data merupakan penunjang pengambilan keputusan secara cepat, atau dikenal dengan istilah *Data Driven Decision Making (DDDM)* dimana kemajuan teknologi berperan besar dalam memanfaatkan data dan informasi tersebut (Aditya, Mulyana, Eka, & Widianto, 2020). Dalam berbagai aspek termasuk melakukan prediksi mengenai informasi curah hujan yang akurat di mana pemodelan tersebut masih memiliki kekurangan seperti penggunaan jumlah parameter, asumsi matematis, dan rumusan persamaan yang cenderung rumit, untuk menghasilkan sebuah model prediksi yang mendekati keakuratan optimal harus memiliki banyak *paremeter* dan *variabel* input untuk memenuhi sebuah asumsi prediksi (Supriyadi, 2019).

Mengatasi perihal tersebut, dikembangkanlah sebuah sebuah *Kecerdasan Buatan (Artificial Inteligence)* yang memiliki kemampuan untuk melakukan pembelajaran untuk menganalisa berbagai macam asumsi dan aspek yang berpengaruh untuk menarik kesimpulan (Supriyadi, 2019)*. Artificial Intelligence (AI)* atau *kecerdasan buatan* didefinisikan secara berbeda dalam konteks yang berbeda pula dalam disiplin ilmu komputer AI adalah mempelajari cara mensimulasikan untuk melakukan tugas yang biasanya membutuhkan pemahaman seperti manusia (Knowledge@Wharton, 2018). AI juga suatu cabang ilmu komputer yang menggunakan lebih banyak simbol daripada angka, dan memproses informasi berdasarkan jumlah aturan dalam merepresentasikan pengetahuan (Swarnkar & Swarnkar, 2019).

*Machine Learning* adalah bagian dari AI di mana mesin digunakan untuk belajar dari pengalaman masa lalu (Aditya, Mulyana, Eka, & Widianto, 2020). Algoritma *Machine Learning* digunakan dalam berbagai aplikasi, seperti dalam kedokteran, pengenalan email, pengenalan suara, dan visi komputer, di mana sulit atau tidak mungkin untuk mengembangkan algoritma konvensional untuk melakukan tugas yang diperlukan (Hu, Niu, Carrasco, Lennox, & Arvin, 2020).

Beberapa implementasi *Machine Learning* menggunakan data dan *Neural Network (Jaringan Saraf)* dengan cara yang meniru kerja otak biologis manusia (Zhou, 2019). *Deep learning* dapat dipahami sebagai bentuk *Neural Network* layer berganda yang merupakan bagian dari *Machine Learning* yang dapat digunakan dalam tugas termasuk *computer vision, speech recognition, natural language processing, machine translation, bioinformatics, drug design, medical image analysis, material inspection* dan *board game programs*, di mana mereka telah menghasilkan hasil yang sebanding dan dalam beberapa kasus melebihi kinerja para pakar (Hu, Niu, Carrasco, Lennox, & Arvin, 2020).

Misalnya, sebuah komputasi yang menggunakan *deep learning*, akan mampu memahami konsep seperti garis, bentuk, tekstur, dan juga pengaruhnya dengan melihat data-data citra tanpa bantuan tambahan dari manusia (Schneiderman & Kanade, 2002). *Machine Learning* senantiasa bekerja menggunakan 1 layer dimana *deep learning* bekerja lebih dari 1 *layer*. untuk batasan layer dari *deep learning* itu sendiri sebagai *Neural Network* biasanya memiliki 3 *layer* atau lebih, makin bayak *layer* yang digunakan akan memengaruhi lama / delay waktu yang terpakai untuk komputer mengkalkulasi (Hinton, et al., 2012).

*Layer* pada *deep learning* dapat di gambarkan seperti lapisan neuron pada otak manusia *layer* itu nantinya akan menggambarkan jarak atau vector menggunakan Fungsi matematika yaitu fungsi *sigmoid (σ)* (Putra J. G., 2020)*.* Alasan fungsi sigmoid digunakan karena dalam fungsi ini membutuhkan perhitungan yang relatif mudah dan cepat. Selain itu, fungsi sigmoid dapat diartikan sebagai nilai peluang karena nilainya antara 0 dan 1 (Putra J. G., 2020).

Salah satu pendekatan *deep learning* yang mampu secara otomatis mempelajari fitur yang dideskripsikan dalam bentuk *vektor* adalah *Recurrent Neural Networks (RNN)* (Puspaningrum, Bunga, & Iryanto, 2020). Pada *RNN* sendiri teknik learning bekerja dengan menyimpan layer dari output kembali sebagai input pada hidden layer berikutnya hingga memprediksi hasil akhir (Tarkus, Sompie, & Jacobus, 2020). Kelemahan *RNN* adalah tidak mampu lagi untuk belajar menghubungkan informasi ketika ada kesenjangan yang terus tumbuh, memori yang tersimpan akan semakin tidak relevan seiring waktu berjalan karena tertimpa dengan memori baru (Putra, Osmond, & Ansori, 2020), di karenakan kelemahan dari *RNN* sendiri tidak dapat mempelajari informasi yang terlalu jauh atau *Long-Term Dependencies*, yang cukup jauh pada masukannya (Wibisono & Khodra, 2018).

Penelitian *deep learning* terdahulu yang di lakukan Juanda, Jondri, & Rohmawati, tentang *Prediksi Harga Bitcoin Dengan Menggunakan Recurrent Neural Network* menurutnya, masalah prediksi *time series* adalah jenis pemodelan prediktif yang sulit, tidak seperti pemodelan predikftif regresi, *time series* juga menambah kompleksitas ketergantungan urutan antar variable input. *Recurrent Neural Network* terbukti berhasil digunakan untuk prediksi data *time series* karena *RNN* mampu menggunakan informasi yang telah direkam sebelumnya yang panjang urutannya atau sequence-nya beragam-ragam. Oleh karena itu, pembangunan sistem ini dibuat dengan metode *Recurrent Neural Network* dengan menggunakan algoritma *Backpropagation Through Time*. Hasil akhir Prediksi harga Bitcoin dapat dilakukan menggunakan Recurrent Neural Network. Akurasi rata-rata terbaik yang didapatkan adalah 98.76% pada data latih dan 97.46% pada data uji, dengan parameter jumlah pola input terbaik adalah 5, jumlah epoch 1000, nilai learning rate 0.001 dan jumlah hidden unit 50 (Juanda, Jondri, & Rohmawati, 2018).

*Long Short-Term Memory (LSTM)* merupakan sebuah pengembangan metode dari arsitektur *Recurrent Neural Network (RNN)*, Banyak peneliti yang mengembangkan metode *LSTM* di berbagai bidang seperti dalam bidang prediksi deret waktu atau *forecasting* dikarenakan metode *LSTM* mampu mengatasi kekurangan tersebut karena metode ini dapat mengatur memori pada setiap masukannya dengan menggunakan memory cells dan gate units pada setiap neurons yang berfungsi sebagail pengatur memori (Putra, Osmond, & Ansori, 2020). Contoh penggunaan deep learning untuk data *time series* yang banyak dihasilkan dari pengamatan cuaca adalah *LSTM*, *LSTM* sendiri diciptakan oleh Hochreiter dan Schmidhuber pada tahun 1997 (Supriyadi, 2019).

Penelitian *LSTM* terdahulu yang yang di teliti oleh Poornima & Pushpalatha, mengenai *Prediction of Rainfall Using Intensified LSTM Based Recurrent Neural Network with Weighted Linear Units* dalam penelitian tersebut menyajikan *Long Short-Term Memory (LSTM)* berbasis *Recurrent Neural Network (RNN)* untuk memprediksi *curah hujan*. *Neural Network* dilatih dan diuji menggunakan dataset standar curah hujan. Jaringan yang dilatih akan menghasilkan atribut prediksi curah hujan. Parameter yang dipertimbangkan untuk evaluasi kinerja dan efisiensi model prediksi curah hujan yang diusulkan adalah *Root Mean Square Error (RMSE), akurasi, jumlah epoch, loss*, dan *learning rate*. Menurutnya *LSTM* dapat menyimpan data besar ke dalam memorinya dan dapat menghindari gradient yang hilang lebih baik dari pada *RNN* dan menunjukan akurasi lebih baik di bandingkan *RNN*, *LSTM* pun juga mempertahankan akurasi di masa depan seiring dengan mempertimbangkan nilai *Root Mean Square Error (RMSE), akurasi, jumlah epoch, loss, dan learning rate* (Poornima & Pushpalatha, 2019)*.*

Penilituan *LSTM* terdahulu yang juga di lakukan oleh Supriyadi, mengenai metode *deep learning LSTM* untuk memprediksi parameter cuaca, seperti suhu udara, kelembaban, kecepatan angin, dan tekanan udara. Metode ini bekerja dengan memanfaatkan fungsi matematika seperti fungsi *tanh* dan *sigmoid* yang berada dalam layer *LSTM*. Adapun jumlah layer yang digunakan sebanyak 200 buah. Sedangkan jumlah datanya dibagi dua menjadi training data dan test data dengan rasio 9:1.pada bulan Januari 2019. Diperoleh *RMSE* parameter suhu udara, kelembaban, kecepatan angin, dan tekanan udara nilainya semakin baik ketika menggunakan deep learning *LSTM* dengan update dibandingkan *LSTM* tanpa update. Diperoleh hasil prediksi suhu udara, kelembaban, kecepatan angin, dan tekanan udara 1 hari ke depan memiliki *RMSE* yang baik. Dari parameter cuaca tersebut hanya parameter suhu dan kelembaban udara yang mengalami pertambahan *RMSE* seiring bertambahnya waktu. Sedangkan parameter kecepatan angin dan tekanan udara mengalami penurunan di hari ketiga dan meningkat secara kontinu hingga 1 bulan ke depan (Supriyadi, 2019).

Berdasarkan rincian penjelasan sebelumnya, sangat dimungkinkan untuk menggunakan *deep learning* dengan metode *LSTM* dikarenakan mendukung kegiatan proyeksi curah hujan. Karena data pengamatan meteorologi umumnya berupa *vektor* dan *time* series, untuk itu penulis membuat Penelitian dalam bentuk skripsi dengan judul “**PROYEKSI CURAH HUJAN DAERAH PADANG PARIAMAN MENGGUNAKAN DEEP LEARNING DENGAN METODE LONG SHORT-TERM MEMORY.**

## Perumusan Masalah

Berdasarkan uraian pada latar belakang masalah maka yang menjadi perumusan masalah pada skripsi ini adalah sebagai berikut :

1. Bagaimana melakukan proses prediksi curah hujan dengan mengunakan pendekatan Deep Learning menggunakan metode *Long Short-Term Memory* dapat menghasilkan alternatif dalam pengambilan sebuah keputusan ?
2. Bagaimana penerapkan Deep Learning menggunakan metode *Long Short-Term Memory* dapat melakukan prediksi curah hujan dengan parameter cuaca seperti suhu, kelembaban, kecepatan angin dan tekanan udara ?
3. Bagaimana pengujian Deep Learning dengan metode Long Short-Term Memory di implementasikan kedalam sebuah sistem yang dibangun untuk memprediksi curah hujan di daerah padang pariaman ?

## Hipotesis

Berdasarkan rumusan masalah yang telah ditentukan maka dapat diambil kesimpulan sebagai berikut :

1. Dengan melakukan proses prediksi curah hujan dengan mengunakan pendekatan Deep Learning diharapak dapat menghasilkan alternatif dalam pengambilan sebuah keputusan.
2. Dengan menggunakan konsep pendekatan Deep Learning metode *Long Short-Term Memory* diharapkan dapat melakukan prediksi curah hujan dengan parameter cuaca seperti suhu, kelembaban, kecepatan angin dan tekanan udara.
3. Penerapan Deep Learning dengan metode Long Short-Term Memory diharapkan dapat di implementasikan kedalam sebuah sistem yang dibangun untuk memprediksi curah hujan di daerah padang pariaman.

## Batasan Masalah

Adapun batasan masalah pada penelitian ini adalah sebagai berikut :

1. Data yang digunakan adalah data BMKG Klimatologi Padang Pariaman dari tahun 1985 sampai tahun 2021.
2. Metode yang digunakan dalam penelitian ini Long Short-Term Memory.
3. Menggunakan Bahasa Pemograman Python.
4. Fungsi Aktivasi yang digunakan adalah fungsi sigmoid (σ) dan tanh.
5. Untuk fitur variabel yang digunakan dalam penelitian ini terdiri dari curah hujan, kecepatan angin, temperatur, dan kelembapan udara.
6. Data dibagi menjadi dua bagian, yaitu data training dan data test dengan rasio 9:1 di mana 9 untuk training dan 1 untuk test.

## Tujuan Penelitian

Adapun Tujuan dari penelitian ini adalah sebagai berikut :

1. Menerapkan Deep Learning menggunakan metode Long Short-Term Memory untuk melakukan proses prediksi curah hujan dalam menghasilkan alternatif dalam pengambilan sebuah keputusan.
2. Menerapkan pendekatan Deep Learning menggunakan metode Long Short-Term Memory untuk melakukan prediksi curah hujan dengan parameter cuaca seperti suhu, kelembaban, kecepatan angin dan tekanan udara.
3. Melakukan pengujian Deep Learning dengan metode Long Short-Term Memory di implementasikan kedalam sebuah sistem yang dibangun untuk memprediksi curah hujan di daerah padang pariaman.

## Manfaat Penelitian

Hasil dari penelitian di harapkan dapat memberikan manfaat sebagai berikut :

1. Menambah pengetahuan penulis dalam memprediksi / memproyeksi data berbentuk *time series* yang di selesaikan dengan cara *deep learning* dengan menggunakan metode *long short-term memory*.
2. Membuktikan keakuratan metode *long short-term memory* dalam melakukan peramalan khususnya curah hujan.
3. Memberikan informasi tambahan mengenai peramalan curah hujan di daerah padang pariaman yang akan terjadi pada masa mendatang.

## Gambaran Umum Objek Penelitian

BMKG Klimatologi Padang Pariaman merupakan Stasiun Klimatologi BMKG Kelas II yang terletak di daerah *Jalan Raya Padang – Bukittinggi KM. 51 Kapalo Hilalang Sumatera Barat*. Berikut ini adalah gambaran umum tentang BMKG Klimatologi :

### Sejarah BMKG

Pengamatan meteorologi dan geofisika di Indonesia dimulai pertama kali pada tahun 1841 diawali dengan pengamatan yang dilakukan secara individual oleh Dr. Pieter Loth Onnen, Kepala Rumah Sakit di Bogor. Dari tahun ke tahun kegiatannya berkembang seiring dengan meningkatnya kebutuhan akan data hasil pengamatan cuaca dan geofisika. Pada tahun 1866, pemerintah Hindia Belanda meresmikan kegiatan pengamatan perorangan sebagai lembaga pemerintah dengan nama Observatorium Magnetisch en Meteorologisch atau Observatorium Magnetik dan Meteorologi yang dipimpin oleh Dr. Pieter Adrian Bergsma.

Pada tahun 1879, 74 jaringan alat pengukur hujan dibangun di Jawa. pengamatan medan magnet bumi dipindahkan dari Jakarta ke Bogor Pada tahun 1902. Pada tahun 1908 pemantauan gempa dimulai dengan pemasangan komponen horizontal seismograf Wiechert di Jakarta, sedangkan pemasangan komponen vertikal dilakukan pada tahun 1928. Pada tahun 1912, pengamatan meteorologi ditata ulang dengan menambahkan jaringan sekunder. Sedangkan pada tahun 1930 jasa meteorologi mulai digunakan untuk penerangan.

Pada masa pendudukan Jepang antara tahun 1942 dan 1945, nama badan meteorologi dan geofisika diubah menjadi Kisho Kauso Kusho atau Lembaga Meteorologi. Pada tahun 1945 Setelah proklamasi kemerdekaan Indonesia, badan tersebut dibagi menjadi dua: Di Yogyakarta dibentuk Badan Meteorologi yang berkedudukan di lingkungan Mabes TNI khusus untuk melayani kepentingan Angkatan Udara. Di bawah Kementerian Pekerjaan Umum dan Energi Badan Meteorologi dan Geofisika dibentuk Di Jakarta.

Pada tanggal 21 Juli 1947 Biro Meteorologi dan Geofisika diambil alih oleh Pemerintah Belanda dan namanya diubah menjadi Meteorologisch en Geofisiche Dienst. Badan Meteorologi dan Geofisika yang dikelola oleh Pemerintah Republik Indonesia berada di Jl. Gondangdia, Jakarta. Pada tahun 1949, setelah penyerahan kedaulatan Republik Indonesia dari Belanda, Meteorologisch en Geofisiche Dienst diubah menjadi Biro Meteorologi dan Geofisika di bawah Departemen Perhubungan dan Pekerjaan Umum. Selanjutnya pada tahun 1950 Indonesia resmi masuk sebagai anggota Organisasi Meteorologi Dunia dan Kepala Badan Meteorologi dan Geofisika menjadi Wakil Tetap Indonesia dengan WMO.

Pada tahun 1955 Biro Meteorologi dan Geofisika berubah nama menjadi Lembaga Meteorologi dan Geofisika di bawah Kementerian Perhubungan, dan pada tahun 1960 namanya dikembalikan menjadi Biro Meteorologi dan Geofisika di bawah Kementerian Perhubungan Udara.

Pada tahun 1965, namanya diubah menjadi Direktorat Meteorologi dan Geofisika, posisinya tetap di bawah Kementerian Perhubungan Udara. statusnya dinaikkan menjadi lembaga setingkat eselon I dengan nama Badan Meteorologi dan Geofisika, dengan jabatan tetap di bawah Kementerian Perhubungan. Badan Meteorologi dan Geofisika.

Melalui Peraturan Presiden Nomor 61 Tahun 2008, BMG berganti nama menjadi Badan Meteorologi, Klimatologi, dan Geofisika dengan status tetap sebagai Lembaga Pemerintah Non Departemen.

### Struktur Organisasi dan Nama-Nama Pejabatnya

Berikut adalah bentuk struktu organisasi BMKG Padang Pariaman :

(Sumber : Sta. Klim. Kelas II Padang Pariaman, 2021)

Gambar 1. Struktur BMKG Padang Pariaman

BMKG Klimatologi Padang Pariaman dipimpin oleh Kepala Stasiun Klimatologi (Kaslim) bertanggung jawab terhadap seluruh bidang yang ada pada instansi BMKG, Staf yang membantu Kaslim dalam menjalankan aktivitas di kantor adalah sebagai berikut :

1. Tata usaha yang bertanggung jawab terhadap administrasi kantor.
2. Bagian analisa yang bertanggung jawab terhadap pengolahan data dan analisis data-data yang dikirim ke Balai Wilayah I.
3. Tenaga teknis yang bertanggung jawab terhadap data-data klimatologi yang ada di BMKG Klimatologi Padang Pariaman.
4. Tim pengamat yang bertanggung jawab terhadap pengaturan jadwal pengamatan di BMKG Klimatologi Padang Pariaman.
5. Tim komunikasi dan peralatan yang bertanggung jawab terhadap pengiriman informasi kondisi peralatan yang ada di BMKG Klimatologi Padang Pariaman.

### Visi

Mewujudkan BMKG yang handal, tanggap dan mampu dalam rangka mendukung keselamatan masyarakat serta keberhasilan pembangunan nasional, dan berperan aktif di tingkat Internasional.

Terminologi di dalam visi tersebut dapat dijelaskan sebagai berikut :

1. Pelayanan informasi meteorologi, klimatologi, kualitas udara, dan geofisika yang handal ialah pelayanan BMKG terhadap penyajian data, informasi pelayanan jasa meteorologi, klimatologi, kualitas udara, dan geofisika yang akurat, tepat sasaran, tepat guna, cepat, lengkap, dan dapat dipertanggungjawabkan.
2. Tanggap dan mampu dimaksudkan BMKG dapat menangkap dan merumuskan kebutuhan stakeholder akan data, informasi, dan jasa meteorologi, klimatologi, kualitas udara, dan geofisika serta mampu memberikan pelayanan sesuai dengan kebutuhan pengguna jasa.

### Misi

Dalam rangka mewujudkan Visi BMKG, maka diperlukan visi yang jelas yaitu berupa langkah-langkah BMKG untuk mewujudkan Misi yang telah ditetapkan yaitu :

1. Mengamati dan memahami fenomena meteorologi, klimatologi, kualitas udara dan geofisika.
2. Menyediakan data, informasi dan jasa meteorologi, klimatologi, kualitas udara dan geofisika yang handal dan terpercaya.
3. Mengkoordinasikan dan memfasilitasi kegiatan di bidang meteorologi, klimatologi , kualitas udara dan geofisika.
4. Berpartisipasi aktif dalam kegiatan internasional di Bidang meteorologi, klimatologi, kualitas udara dan geofisika.

# BAB II LANDASAN TEORI

## Rekayasa Perangkat Lunak

IEEE Computer Society mendefinisikan Rekayasa Perangkat Lunak (RPL) sebagai penerapan suatu pendekatan yang sistematis, disiplin dan terkuantifikasi atas pengembangan, penggunaan dan pemeliharaan perangkat lunak, serta studi atas pendekatan-pendekatan ini, yaitu penerapan pendekatan engineering atas perangkat lunak (Hasanah & Untari, 2020).

RPL sendiri adalah suatu disiplin ilmu yang membahas semua aspek produksi perangkat lunak, mulai dari tahap awal yaitu analisa kebutuhan pengguna, menentukan spesifikasi dari kebutuhan pengguna, disain, pengkodean, pengujian sampai pemeliharaan sistem setelah digunakan (Hasanah & Untari, 2020).

RPL lebih fokus pada praktik pengembangan perangkat lunak dan mengirimkan perangkat lunak yang bermanfaat kepada pelanggan (customer). Adapun ilmu computer lebih fokus pada teori dan konsep dasar perangkat computer. Rekayasa perangkat lunak lebih fokus pada bagaimana membuat perangkat lunak yang memenuhi kriteria berikut (Hasanah & Untari, 2020) :

1. Dapat terus dipelihara setelah perangkat lunak selesai dibuat seiring berkembangnya teknologi dan lingkungan (maintainability)
2. Dapat diandalkan dengan proses bisnis yang dijalankan perubahan yang terjadi (dependability robust)
3. Efisien dari segi sumber daya dan penggunaan
4. Kemampuan untuk dipakai sesuai dengan kebutuhan (usability)

### Proses Rekayasa Perangkat Lunak

Kerangka kerja proses membangun dasar bagi proses rekayasa perangkat lunak yang lengkap dengan cara mengidentifikasi sejumlah kecil aktivitas kerangka kerja yang cocok bagi semua proyek rekayasa perangkat lunak (Setiyani, 2018). Kerangka kerja proses pada rekayasa perangkat lunak terdiri atas lima aktivitas berikut (Setiyani, 2018) :

#### Komunikasi

Komunikasi, bertujuan untuk memahami tujuan-tujuan stakeholder atas proyek perangkat lunak yang sedang dikembangkan dan mengumpulkan kebutuhan-kebutuhan yang akan membantu mendefinisikan fitur-fitur perangkat lunak berikut dengan fungsi-fungsinya (Setiyani, 2018) :

#### Perencanaan

Kegiatan perencanaan menciptkan suatu peta yang dapat membantu membimbing tim perangkat lunak. Rencana proyek perangkat lunak menggambarkan risiko – risiko yang mungkin muncul, sumber daya yang akan dibutuhkan, produk – produk kerja yang harus dihasilkan dan schedule kerja (Setiyani, 2018) :

#### Pemodelan

Pemodelan, dilakukan bertujuan untuk membuat sketsa sehingga tim perangkat lunak dapat memahami gambaran besar produk yang akan di buat (Setiyani, 2018) :

#### Konstruksi

Konstruksi, sediri adalah kegiatan yang menggabungkan pengkodean dan pengujian (Setiyani, 2018) :

#### Penyerahan

Penyerahan di sini merupakan Penyerahan perangkat lunak kepada user, penyajian perangkat lunak kepada user untuk di evaluasi (Setiyani, 2018) :

### Software Development Life Cycle (SDLC)

System Development Life Cycle (SDLC) adalah metodologi klasik yang digunakan untuk mengembangkan, memelihara dan menggunakan sistem informasi. Siklus hidup sistem itu sendiri merupakan metodologi, tetapi polanya lebih dipengaruhi oleh kebutuhan untuk mengembangkan sistem yang lebih cepat. Pengembangan sistem yang lebih cepat dapat dicapai dengan peningkatan siklus hidup dan penggunaan peralatan pengembangan berbasis computer (Wahyudi, 2018).

#### Tahap-Tahap SDLC

Secara umum tahap-tahap dalam System Development Life Cycle (SDLC)terbagi dalam beberapa tahap (Wahyudi, 2018):

##### Tahap Perencanaan Sistem (system planning)

Tahap Planning Merupakan tahap awal dari pengembangan sistem, tahap ini bertujuan untuk mengidentifikasi dan memprioritaskan sistem informasi apa yang akan dikembangkan, sasaran-sasaran yang ingin dicapai, jangka waktu pelaksanaan serta mempertimbangkan dana yang tersedia dan siapa yang melaksanakan (Wahyudi, 2018).

##### Tahap Analisis Sistem (system analysis)

Tahap Analisis system adalah penelitian atas sistem yang telah ada dengan tujuan untuk merancang system baru atau memperbaharui system yang sudah ada (Wahyudi, 2018).

##### Tahap Perancangan/Desain Sistem (system design)

Tahap Rancangan system adalah penentuan proses dan data yang diperlukan oleh system baru. Jika system ini berbasis komputer, rancangan dapat menyerta kan spesifikasi jenis peralatan yang akan digunakan (Wahyudi, 2018).

##### Tahap Penerapan/Implementasi Sistem (system implementation)

Tahap Penerapan merupakan kegiatan memperoleh dan mengintegrasikan sumber daya fisik dan konseptual yang menghasilkan suatu sistem yang bekerja. Pada tahapan ini dilakukan beberapa hal yaitu: Coding, Testing, Instalasi. Dan Output dari tahapan ini adalah : source code, prosedur, pelatihan (Wahyudi, 2018).

##### Tahap Pemeliharaan/Perawatan Sistem

Tahap pemeliharaan/perawatan sistem merupakan tahap yang dilakukan setelah tahap implementasi yang meliputi penggunaan sistem, audit sistem, penjagaan sistem, perbaikan sistem dan peningkatan sistem (Wahyudi, 2018).

#### Model Waterfall

Model SDLC air terjun (waterfall) sering juga disebut model sekuensial linier (sequential linier) atau alur hidup klasik (classic life cycle). Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuential atau terurut dimulai dari analisis kebutuhan perangkat lunak, desain, pembuatan kode program, pengujian, dan pemeliharaan (maintenance) (Tabrani & Pudjiarti, 2021). Berikut tahap tahapanya sebagai beikut (Tabrani & Pudjiarti, 2021):

##### Analisis Kebutuhan Perangkat Lunak

Proses pengumpulan kebutuhan dilakukan secara intensif untuk menspesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh user. Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu untuk didokumentasikan (Tabrani & Pudjiarti, 2021).

##### Desain

Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengkodean. Tahap ini mentrannslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya. Desain perangkat lunak yang dihasilkan pada tahap ini juga perlu didokumentasikan (Tabrani & Pudjiarti, 2021).

##### Pembuatan Kode Program

Desain harus ditranslasikan kedalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain (Tabrani & Pudjiarti, 2021).

##### Pengujian

Pengujian fokus pada perangkat lunak secara dari segi lojik dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (error) dan memastika keluaran yang dihasilkan sesuai dengan yang diinginkan (Tabrani & Pudjiarti, 2021).

##### Pendukung atau Pemeliharaan (maintenance)

Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirimkan ke user. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru. Tahap pendukung atau pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan perangkat lunak yang sudah ada, tapi tidak untuk membuat perangkat lunak baru (Tabrani & Pudjiarti, 2021).

## Unified Modelling Language *(UML)*

Pemodelan dalam suatu rekayasa perangkat lunak merupakan suatu hal yang dilakukan di tahapan awal. Pemodelan delam perangkat lunak merupakan suatu yang harus dikerjakan di bagian awal dari rekayasa, dan pemodelan ini akan mempengaruhi perkerjaan-pekerjaan dalam rekayasa perangkat lunak tersebut. Salah satu perangkat pemodelan adalah Unified Modelling Language (UML). UML merupakan salah satu standart bahasa yang banyak digunakan di dunia industry untuk mendefinisikan requirement, membuat analisis & desain, sert amenggambarkan arsitektur dalam pemrograman berorientasi objek. UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan mendokumentasikan sistem perangkat lunak (Hasanah & Untari, 2020).



(Sumber : Hasanah & Untari, 2020)*.*

Gambar 2. Bagan UML

Berdasarkan **Gambar 2** berikut penjelasan singkat dari pembagian kategori tersebut (Hasanah & Untari, 2020) :

1. Behavior diagram, merupakan kumpulan diagram yang digunakan untuk menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi pada suatu sistem.
2. Interaction diagram, merupakan kumpulan diagram yang digunakan untuk menggambarkan interaksi sistem dengan sistem lain maupun interaksi antar subsistem pada suatu sistem.
3. Structure diagram, merupakan kumpulan diagram yang digunakan untuk menggambarkan strutur statis dari sistem yang dimodelkan.

### Behavior Diagrams

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

Table 1. Notasi Use Case Diagram

|  |  |  |
| --- | --- | --- |
| No | Simbol | Keterangan |
| 1 | *Use case* | Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor. |
| 2 | *Actor*  *Description: Description: actor* | Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem itu. |
| 3 | *Association* | use case yang memiliki interaksi dengan aktor. |
| 4 | *Extend* | Relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan dapat berdiri sendiri. |
| 5 | *Generalization* | Relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan memerlukan ini untuk menjalankan fungsinya. |

(Sumber : Hasanah & Untari, 2020)

#### Activity Diagram

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 digunakan untuk menggambarkan langkahlangkah atau aktivitas pada suatu sistem (Hasanah & Untari, 2020).

Table 2. Notasi Activity Diagram

|  |  |  |
| --- | --- | --- |
| No | Simbol | Keterangan |
| 1 | *Initial State* | Sebuah diagram aktivitas memiliki sebuah status awal |
| 2 | *Activity* | Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja |
| 3 | *Decision* | Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu |
| 4 | *Join* | Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu |
| 5 | *Final State* | Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir. |
| 6 | *Swimlane*    or | Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi. |

(Sumber : Hasanah & Untari, 2020)

#### Statechart Diagram

Statechart diagram menelusuri individu-individu objek melalui keseluruhan daur hidupnya, menypesifikasikan semua urutan yang mungkin dari pesan-pesan yang akan diterima objek tersebut, bersama-sama dengan tanggapan atas pesan-pesan tersebut (Ilham & Fajri, 2020).

Table 3. Notasi Statechart Diagram

|  |  |  |
| --- | --- | --- |
| No | Simbol | Keterangan |
| 1 | *Initial State* | Kelas pada struktur sistem |
| 2 | *Final State* | Sama dengan konsep interface dalam beberapa pemrograman berorientasi objek. |
| 3 | *Event* | Relasi antarkelas dengan makna umum.  Asosiasi biasanya juga disertai dengan multiplicity |
| 4 | *State* | Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain. |

(Sumber : Ilham & Fajri, 2020).

### Interaction Diagrams

#### Sequence Diagram

Sequence diagrammenggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, display, dan sebagainya) berupa messageyang digambarkan terhadap waktu. Sequence diagramterdiri atas dimensi vertikal (waktu) dan dimensi horizontal (objek-objek yang terkait). Sequence diagrambiasa digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai respons dari sebuah eventuntuk menghasilkan outputtertentu. Diawali dari apa yang men-triggeraktivitas tersebut, proses dan perubahan apa saja yang terjadi secara internal dan outputapa yang dihasilkan (Rinaldi, 2019).

Table 4. Notasi State Machine Diagram

|  |  |  |
| --- | --- | --- |
| No | Simbol | Keterangan |
| 1 | *Actor*  *Description: Description: actor*  Or  Nama aktor | Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi biasanya dinyatakan menggunakan kata benda di awal frasa nama actor |
| 2 | *Lifeline* | Menyatakan kehidupan suatu objek |
| 3 | *Object*  objek : kelas | Menyatakan objek yang berinteraksi pesan |
| 4 | *Active Time* | Menyatakan objek dalam keadaan aktif berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya, |
| 5 | *Message type C****reate***  <<*create*>> | Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat |
| 6 | *Message type* ***Call***  1:nama\_metode() | Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri |
| 7 | *Message type* ***Send***  1: masukan | Menyatakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim |
| 8 | *Message type* ***Return***  1: keluaran  ---------------------- | Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian |
| 9 | *Message type* ***Destroy***  <<*destroy*>> | Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, jika ada create maka ada destroy. |

(Sumber : Rinaldi, 2019)

#### Collaboration Diagram

Collaboration diagram adalah cara alternative untuk mengetahui tahap-tahap terjadinya suatu aktivitas. Perbedaan antara collaboration dan sequence diagram adalah collaboration diagram memperlihatkan bagaimana hubungan antara beberapa objek berdasarkan urutan dari pesan, sedangkan sequence diagram memperlihatkan bagaimana urutan kejadian berdasarkan waktu (Ilham & Fajri, 2020).

Table 5. Notasi Collaboration Diagram

|  |  |  |
| --- | --- | --- |
| No | Simbol | Keterangan |
| 1 | *Object*  objek : kelas | Objek yang melakukan interaksi pesan. |
| 2 | *Link* | Relasi antara objek yang menghubungkan objek satu dengan lainya atau dengan diri sendiri. |
| 4 | *State* | Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain. |

(Sumber : Ilham & Fajri, 2020)

### Structure Diagrams

#### Class Diagram

Class diagram adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. Class diagram menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metode/fungsi). Class diagram menggambarkan struktur dan deskripsi class, package dan objek beserta hubungan satu sama lain seperti containment, pewarisan, asosiasi, dan lain-lain (Rinaldi, 2019).

Tabel 6. Notasi Class Diagram

|  |  |  |
| --- | --- | --- |
| No | Simbol | Keterangan |
| 1 | *Class*  **nama\_kelas**  + atribut  + operasi() | Kelas pada struktur sistem |
| 2 | *Interface* | Sama dengan konsep interface dalam beberapa pemrograman berorientasi objek. |
| 3 | *Association* | Relasi antar kelas dengan makna umum.  Asosiasi biasanya juga disertai dengan multiplicity |
| 4 | *Directed Association* | Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain. |
| 5 | *Dependency* | Relasi antar kelas dengan ketergantungan antar kelas |
| 6 | *Aggregation* | Relasi antar kelas dengan makna semua-bagian (whole-part) |

(Sumber : Rinaldi, 2019)

#### Deployment Diagram

Menggambarkan secara lengkap bagaimana komponen di deployment dalam infrastruktur sistem, dimana komponen akan terletak, bagaimana kemampuan jaringan pada kondisi tertentu, spesifikasi server, dan hal-hal lain yang bersifat fisikal (Wijaya, Masriadi, & Ikhlas, 2020).

Tabel 7. Notasi Deployment Diagram

|  |  |  |
| --- | --- | --- |
| No | Simbol | Keterangan |
| 1 | *Package*  [Description: http://4.bp.blogspot.com/-sc_hs41py7c/VR5gbgH2yHI/AAAAAAAAAXU/xAgOjf7Nc-g/s1600/Snap%2B2015-04-03%2Bat%2B16.41.02.png](http://4.bp.blogspot.com/-sc_hs41py7c/VR5gbgH2yHI/AAAAAAAAAXU/xAgOjf7Nc-g/s1600/Snap+2015-04-03+at+16.41.02.png) | Package merupakan sebuah bungkusan dari satu atau lebih komponen. |
| 2 | *Node*  Nama\_node | Biasanya mengacu pada perangkat keras(hardware), perangkat lunak yang tidak dibuat sendiri (software), jika didalam node disertakan komponen untuk mengkonsistensikan rancangan maka komponen yang diikutsertakan harus sesuai dengan komponen yang telah didefinisikan sebelumnya pada diagram komponen. |
| 4 | *Dependency* | Ketergantungan antar komponen, arah panah mengarah pada komponen yang dipakai. |
| 5 | *Link* | Relasi antar komponen. |

(Sumber : Wijaya, Masriadi, & Ikhlas, 2020).

## Artificial Intelligence

*Andreas Kaplan* dan *Michael Haenlein* mendefinisikan kecerdasan buatan / *Artificial Intelligence (AI)* sebagai kemampuan system untuk menafsirkan data eksternal dengan benar, untuk belajar dari data tersebut, dan menggunakan pembelajaran tersebut guna mencapai tujuan dan tugas tertentu melalui adaptasi yang fleksibel. Sistem seperti ini umumnya dianggap sebagai computer (Siahaan, et al., 2020).

Lebih lanjut *Budiharto* menyatakan bahwa *Intelligence* merupakan istilah yang kompleks yang dapat didefinisikan dengan ungkapan yang berbeda seperti logika, pemahaman, self-awareness, pembelajaran, perencanaan, dan problem solving. Sedangkan “Artificial” adalah sesuatu yang tidak nyata, seperti tipuan karena merupakan hasil simulasi (Sihombing & Syaputra, 2020).

Savitri menguraikan bahwa Kecerdasan buatan / Artificial intelligence (AI) merupakan bidang ilmu komputer yang menekankan pada penciptaan mesin cerdas yang bekerja dan bereaksi seperti manusia yang perkembangannya terjadi sangat pesat di era revolusi industri keempat (Sihombing & Syaputra, 2020).

Kecerdasan diciptakan dan dimasukkan kedalam suatu mesin / komputer agar dapat melakukan pekerjaan seperti yang dapat dilakukan oleh manusia. Beberapa macam bidang yang menggunakan kecerdasan buatan antara lain sistem pakar, permainan komputer (games), logika fuzzy, jaringan saraf tiruan dan robotika (Siahaan, et al., 2020).

Tujuan utama dari pembuatan AI adalah untuk membuat mesin memiliki fungsi yang memiliki kriteria-kriteria kecerdasan didalamnya sehingga mesin tersebut mampu melakukan pekerjaan manusia yang lebih kompleks, tergantung dari tingkat kecerdasan AI yang digunakan (Gunova, 2021).

Dalam penerapannya, terdapat 6 kemampuan utama yang dapat diklasifikasikan sebagai AI (Gunova, 2021), Kemampuan tersebut antara lain :

* Representasi Informasi/Pengetahuan (Knowledge Representation)
* Perencanaan (Planning)
* Persepsi (Perception)
* Machine Learning / Deep Learning
* Pemahaman Bahasa (Natural Language Processing)
* Robotics

Meskipun 6 kemampuan diatas memiliki fungsi yang berbeda-beda, kemampuan kemampuan tersebut saling berhubungan satu sama lain saat pengaplikasiaannya. Bahkan hubungan antar kemampuan utama diatas akan membentuk kemampuan baru (Gunova, 2021).

## Machine Learning

Machine Learning merupakan Disiplin ilmu yang menggunakan berbagai pendekatan untuk mengajarkan komputer untuk menyelesaikan tugas-tugas di mana tidak ada algoritma yang sepenuhnya memuaskan tersedia. Dalam kasus di mana terdapat sejumlah besar jawaban potensial, satu pendekatan adalah memberi label beberapa jawaban yang benar sebagai valid. Ini kemudian dapat digunakan sebagai data pelatihan bagi komputer untuk meningkatkan algoritme yang digunakannya untuk menentukan jawaban yang benar. Misalnya, untuk melatih sistem untuk tugas pengenalan karakter digital, dataset MNIST dari angka tulisan tangan sering digunakan (Alpaydin, 2020).

Program *Machine Learning* dapat melakukan tugas tanpa diprogram secara eksplisit untuk melakukannya. Ini melibatkan komputer belajar dari data yang disediakan sehingga mereka melakukan tugas-tugas tertentu. Untuk tugas-tugas sederhana yang ditugaskan ke komputer, dimungkinkan untuk memprogram algoritme yang memberi tahu mesin bagaimana menjalankan semua langkah yang diperlukan untuk memecahkan masalah yang dihadapi; di bagian komputer, tidak diperlukan pembelajaran. Untuk tugas yang lebih maju, mungkin sulit bagi manusia untuk membuat algoritme yang diperlukan secara manual. Dalam praktiknya, ternyata lebih efektif untuk membantu mesin mengembangkan algoritmenya sendiri, daripada meminta pemrogram manusia menentukan setiap langkah yang diperlukan (Alpaydin, 2020).

Algoritma *Machine Learning* digunakan dalam berbagai macam aplikasi, seperti dalam kedokteran, penyaringan email, pengenalan suara, dan visi komputer, di mana sulit atau tidak mungkin untuk mengembangkan algoritma konvensional untuk melakukan tugas-tugas yang diperlukan (Hu, Niu, Carrasco, Lennox, & Arvin, 2020).

## Deep Learning

Sejak tahun 2006, *Deep Structured Learning* atau yang lebih dikenal dengan *Deep learning* atau *Hierarchical Learning* telah muncul sebagai area baru dalam penelitian *Machine Learning* yang berdasarkan pada suatu set algoritma yang mencoba untuk memodelkan abstraksi tingkat tinggi pada data dengan menggunakan graf yang mendalam dengan beberapa layer pengolahan, yang terdiri dari beberapa transformasi linier dan non-linier (François, 2018).

*Deep Learning* sendiri adalah cabang ilmu *machine learning* berbasis *Neural Network (NN)* atau bisa dikatakan sebagai perkembangan dari *Neural Network* (Ilahiyah & Nilogiri, 2018). Metode pendekatan *deep learning* mengklasifikasi data dalam dua sesi yaitu sesi training dan testing. Pada sesi training mempelajari ekstrasi fitur dari setiap data supaya bisa membedakan suaru label dengan label yang lain. Pada sesi testing data-data yang diuji dapat dianalisa dari hasil sesi training (Azizah, Umayah, & Fajar, 2018).



(Sumber : Savalia & Emamian, 2018)

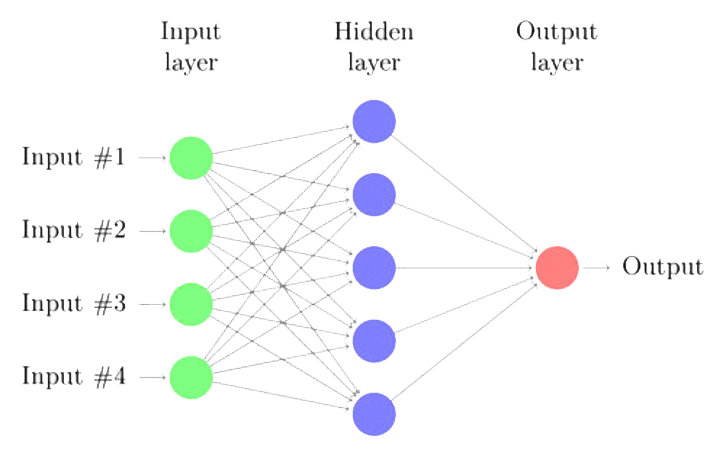
Gambar 3. Perbedaan Simple Neural Network dan Deep Learning

Dalam *deep learning*, sebuah komputer belajar mengklasifikasi secara langsung dari gambar atau suara. Metode *deep learning* menggunakan CPU dan RAM dalam proses komputasi, dan juga memanfaatkan GPU sehingga proses komputasi data yang besar dapat berlangsung lebih cepat (Ilahiyah & Nilogiri, 2018).

## Jaringan Syaraf Tiruan

*Jaringan saraf tiruan (JST) / Artificial Neural Network (ANN) / Neural Nenwork (NN)* adalah jaringan komputasi terinspirasi dari cara kerja otak manusia karena terbukti Otak manusia melakukan sesuatu hal yang sama persis seperti, hirarki pertama pada neuron menerima formasi pada visual cortex yang sensitif terhadap gambaran tepi dan gumpalan khusus (François, 2018).

Menurut Qomariyah, Hamzah, & Mustika *Artificial Neural Network* merupakan metode pemrosesan informasi yang mempunyai karakteristik menyerupai jaringan syaraf biologi. Bagian terpenting dari Artificial Neural Network adalah kemampuannya dalam melakukan pembelajaran (Qomariyah, Hamzah, & Mustika, 2018).



(Sumber : Shekar, S´a, Ferreira, & Soares, 2018)

Gambar 4. Skema Jaringan Saraf Tiruan

Seperti yang terlihat pada **Gambar 4** Lapisan-lapisan penyusun JST tersebut dapat dibagi menjadi 3, yaitu (Satria, 2018) :

1. Lapisan Input, unit-unit di dalam lapisan input disebut unit-unit input. Unit-unit input tersebut menerima pola inputan data dari luar yang menggambarkan suatu permasalahan (Satria, 2018)
2. Lapisan tersembunyi, unit -unit di dalam lapisan tersembunyi disebut unit- unit tersembunyi (Satria, 2018)
3. Lapisan output, unit-unit didalam lapisan outputdisebut unit-unit output (Satria, 2018)

## Long Short-Term Memory

*Long Short-Term Memory (LSTM)* merupakan pengembangan dari *Recurrent Neural Network (RNN)* dengan mengatasi salah satu kekurangan *RNN* yaitu kemampuan pengelolaan informasi dalam periode yang lama yang mana dilakukan modifikasi pada Recurrent Neural Network (RNN) dengan memberi memory cell untuk dapat menyimpan informasi dalam waktu yang lama. Diusulkan oleh Sepp Hochreiter dan Jurgen Schmidhuber pada tahun 1997, *LSTM* banyak dipilih untuk prediksi berbasis waktu atau *time series* karena dikenal lebih unggul dan handal dalam melakukan prediksi dalam waktu lama dibanding algoritma lain (Zahara, Sugianto, & Ilmiddafiq, 2019; Manaswi, 2018).

A screenshot of a video game

Description automatically generated with medium confidence  
Diagram

Description automatically generated

(Sumber : Manu, 2021)

Gambar 5. Perulangan modul dalam LSTM berisi empat layer yang saling berinteraksi.

Kunci LSTM adalah *cell state*, garis horizontal yang melewati bagian atas diagram keadaan sel seperti ban berjalan. Ini berjalan lurus ke bawah seluruh rantai, memiliki beberapa linier kecil interaksi (Ghosh, Bose, Maji, Debnath, & Sen, 2019). Untuk setiap sel memori memiliki tiga layer sigmoid dan satu layer tanh (Qiu, Wang, & Zhou, 2019).

A picture containing text, clock

Description automatically generated

(Sumber : Manu, 2021)

Gambar 6. Alur Informasi Cell State pada LSTM

Keterangan (Pardede & Ibrahim, 2020) :

: *Cell state*

: Nilai *Cell state* sebelum order ke *t*

Pada **Gambar 6**, garis horizontal yang melalui bagian atas diagram dikenal sebagai *cell state* (, ). Ini bertindak seperti ban berjalan yang berjalan di seluruh jaringan. Ini membawa informasi dari sel sebelumnya ke saat ini dan seterusnya (Hiransha, Gopalakrishnan, Menon, & Soman, 2018). Kemampuan untuk menambah atau menghapus informasi ke cell state dikendalikan oleh struktur yang disebut *gate*. *Gate* digunakan untuk secara opsional membiarkan informasi lewat. Informasi yang di saring melalui struktur *gate* yang akan mempertahankan dan memperbarui *cell state* memori, LSTM memiliki tiga di antaranya gerbang, untuk melindungi dan mengontrol cell state, Struktur gerbangnya mencakup *forget gate*, *input gate.* dan *output gate*. *Gate* terdiri dari layer jaring saraf sigmoid dan operasi perkalian pointwise (Ghosh, Bose, Maji, Debnath, & Sen, 2019; Qiu, Wang, & Zhou, 2019).

Graphical user interface, application

Description automatically generated

(Sumber : Manu, 2021)

Gambar 7. Layer sigmoid mengeluarkan angka antara nol dan satu.

Pada **Gambar 7** layer sigmoid mengeluarkan angka antara 0 dan 1, menggambarkan berapa banyak dari setiap komponen yang harus dilewati. Nilai 0 berarti “jangan biarkan apa pun lewat”, sedangkan nilai 1 berarti “biarkan semuanya lewat!” (Ghosh, Bose, Maji, Debnath, & Sen, 2019)*.* Berikut ialah keterangan setiap gate yang ada pada **Gambar 5** :

### Forget Gate

Pada *forget gate* informasi pada setiap data *input* yang akan diolah dan dipilih data mana saja yang akan disimpan atau dibuang pada memory cells. Fungsi aktivasi yang digunakan pada *forget gate* ini adalah fungsi aktivasi sigmoid. Dimana hasil keluarannya antara 0 dan 1. Jika keluarannya adalah 1 maka semua data akan disimpan dan sebaliknya jika keluarannya 0 maka semua data akan dibuang (Aldi, Jondri, & Aditsania, 2018). Dengan rumus seperti pada **Gambar 8** :

Diagram

Description automatically generated

(Sumber : Manu, 2021)

Gambar 8. Alur Informasi Forget Gate pada LSTM

Keterangan (Pardede & Ibrahim, 2020) :

: *Forget gate*

: Fungsi Aktivasi *Sigmoid*

: Nilai *Weight* untuk *Forget gate*

: Nilai *output* sebelum order ke *t*

: Nilai *input* pada order ke *t*

: Nilai bias pada *Forget gate*

Pada **Gambar 8** LSTM memutuskan informasi apa yang akan dibuang dari *cell state*. Keputusan ini dibuat oleh layer sigmoid yang disebut "layer forget gate." (). Terlihat pada dan dan nilai *output* antara angka 0 dan 1 untuk setiap angka dalam *cell state* pada **Gambar 6**. *Output* dari 1 mewakili 'sepenuhnya simpan ini' sementara 0 mewakili 'singkirkan ini sepenuhnya' (Boruah & Barman, 2018).

### Input Gate

Pada *input gate* terdapat dua gates yang akan dilaksanakan, pertama akan diputuskan nilai mana yang akan diperbarui menggunakan fungsi aktivasi sigmoid. Selanjutnya fungsi aktivasi tanh akan membuat vector nilai baru yang akan disimpan pada memory cell (Aldi, Jondri, & Aditsania, 2018). Dengan rumus seperti pada **Gambar 9** :

Diagram

Description automatically generated

(Sumber : Manu, 2021)

Gambar 9. Alur Informasi yang melewati Input Gate pada LSTM

Keterangan (Pardede & Ibrahim, 2020) :

: *Input gate*

: Nilai baru yang dapat ditambahkan ke *cell state*

: Fungsi Aktivasi *Sigmoid*

: Fungsi Aktivasi *Tanh*

: Nilai *Weight* untuk *Input gate*

: Nilai *Weight* untuk *Cell state*

: Nilai *output* sebelum order ke *t*

: Nilai *input* pada order ke *t*

: Nilai bias pada *Input gate*

: Nilai bias pada *cell state*

Pada langkah berikutnya di **Gambar 9** LSTM memutuskan informasi apa yang akan disimpan dari *cell state*. Pertama layer *sigmoid* yang disebut "layer *input gate*" () memutuskan nilai mana yang akan diperbarui. Setelah itu, layer membuat vektor nilai kandidat baru, , yang dapat ditambahkan ke state (Ghosh, Bose, Maji, Debnath, & Sen, 2019).

### Cell State / Memory State

Pada *cell state* gates akan mengganti nilai pada memory cell sebelumnya dengan nilai memory cell yang baru. Dimana nilai ini didapatkan dari menggabungkan nilai yang terdapat pada forget gate dan input gate (Aldi, Jondri, & Aditsania, 2018). Dengan rumus seperti pada **Gambar 10** :

Diagram

Description automatically generated with medium confidence

(Sumber : Manu, 2021)

Gambar 10. Alur Memperbaharui Cell State pada LSTM

Keterangan (Pardede & Ibrahim, 2020) :

: *Cell state*

: *Forget gate*

: Nilai *Cell state* sebelum order ke *t*

: *Input gate*

: Kandidat konteks baru yang dapat ditambahkan ke *cell state*

Pada langkah selanjutnya di **Gambar 10**, kedua layer di gabungkan digabungkan untuk membuat pembaruan ke *cell state*. Pada Langkah inilah nilai *cell state* lama (), akan di perbaharui ke nilai dari *cell state* baru () dimana LSTM akan mengalikan *cell state* lama dengan () kemudian ditambahkan dengan () . Ini adalah nilai kandidat baru, yang diskalakan berdasarkan seberapa banyak kami memutuskan untuk memperbarui setiap nilai negara bagian. (Ghosh, Bose, Maji, Debnath, & Sen, 2019).

### Output Gate

Pada *output gate* terdapat dua gate yang akan dilaksanakan, pertama akan diputuskan nilai pada bagian memory cell mana yang akan dikeluarkan dengan menggunakan fungsi aktivasi sigmoid. Selanjutnya akan ditempatkan nilai pada memory cell dengan menggunakan fungsi aktivasi tanh. Terakhir kedua gate tersebut di dikalikan sehingga menghasilkan nilai yang akan dikeluarkan (Aldi, Jondri, & Aditsania, 2018). Dengan rumus seperti pada **Gambar 11** :

A picture containing text, clock

Description automatically generated

(Sumber : Manu, 2021)

Gambar 11. Alur Informasi melewati Output Gate pada LSTM

Keterangan (Pardede & Ibrahim, 2020) :

: *Output gate*

: *Cell state*

: Fungsi Aktivasi *Sigmoid*

: Fungsi Aktivasi *Tanh*

: Nilai *Weight* untuk *Output gate*

: Nilai *output* sebelum order ke *t*

: Nilai *input* pada order ke *t*

: Nilai bias pada *Output gate*

: Nilai *output* pada order ke *t*

Terakhir pada **Gambar 11** adalah tahap dimana perlu memutuskan apa yang akan hasilkan. *Output* akan didasarkan pada *cell state*, tetapi akan menjadi versi yang difilter. Pertama, kita menjalankan layer sigmoid yang memutuskan bagian mana dari *cell state* yang akan kita hasilkan. Kemudian, *cell state* di tempatkan melalui *tanh* (untuk mendorong nilai menjadi antara -1 dan 1) dan mengalikannya dengan *output* *gate* layer *sigmoid*, sehingga hanya akan menampilkan bagian yang putuskan (Ghosh, Bose, Maji, Debnath, & Sen, 2019).

## Prediksi / Forecasting

Menurut Sucipto & Syaharuddin Prediksi / Peramalan (forecasting) adalah kegiatan mengestimasi apa yang akan terjadi pada masa yang akan datang. Peramalan diperlukan karena adanya kesenjangan waktu (timelag) antara kesadaran dibutuhkannya suatu kebijakan baru dengan waktu pelaksanaan kebijakan tersebut (Sucipto & Syaharuddin, 2018).

Menurut Putro, Furqon, & Wijoyo prediksi merupakan suatu proses untuk meramalkan atau memperkirakan suatu variabel di masa yang akan datang. Dalam kasus prediksi biasanya data yang sering digunakan adalah data kuantitatif. Prediksi tidak harus menghasilkan suatu jawaban yang pasti kejadian, melainkan berusaha untuk mencari jawaban yang sedekat mungkin dengan kejadian yang akan terjadi (Putro, Furqon, & Wijoyo, 2018).

Secara umum, ada dua jenis prediksi yaitu kualitatif dan kuantitatif. Prediksi kualitatif merupakan prediksi yang bersifat subjektif, hal ini karena didasarkan pada pengalaman empiris, intuisi pengambilan keputusan dan emosi manusia.Sedangkan, prediksi kuantitatif merupakan prediksi yang bersifat objektif sebab didasarkan pada data aktual dan diolah menggunakan metode tertentu (Surtiningsih, Furqon, & Adinugroho, 2018).

Jenis sesuai yang di jelaskan oleh para ahli sebelumnya prediksi yang digunakan dalam penelitian adalah prediksi kuantitatif. Dikarenakan data yang digunakan adalah data dari masa lalu berupa angka dengan runtutan waktu.

## Penelitian Terdahulu

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Pengarang, Tahun | Metode | Data | Hasil |
| 1 | (Moghar & Hamiche, 2020) | Long Short Term Memory | Dokumentasi saham NYSE  (GOOGL dan NKE) dari website yahoo finance | Setelah melatih NN, hasil pengujian peneliti menunjukkan hasil yang berbeda, jumlah epoch serta panjang data memiliki pengaruh yang signifikan terhadap hasil pengujian. Setelah mengamati data peneliti, peneliti dapat melihat bahwa pada awalnya data kurang stabil dan memiliki nilai yang lebih rendah, setelah NKE mulai mengintip nilai yang lebih besar, aset menjadi lebih fluktuatif, maka sifat aset ini berubah. Model peneliti telah kehilangan jejak harga pembukaan sekitar 600 hingga 700 hari pengujian yang sesuai dengan perubahan sifat data. |
| 2 | (Karno, Hastomo, Nisfiani, & Lukman, 2020) | Long Short Term Memory berbasis *Gated Recurrent Units* | Dokumentasi saham bank di Indonesia dari website yahoo finance | Dengan menggunakan paket library seborn di python, hasil perhitungann korelasi dapat ditunjukkan di matrik heatmap dalam bentuk numerik dan tingkatan warna. Hasil numerik korelasi berupa bilangan dengan rentang -1 dan 1, dimana nilai 1 menunjukan hubungan yang kuat, dan nilai 0 menunjukan hubungan yang rendah antara dua data. Jadi dapat dilihat bahwa RMSE mampu meredam perubahan kesalahan yang besar, sebaliknya MSE mampu melihat perubahan kesalahan yang kecil. GRU berupa sel yang berisi hanya 2 gate dengan rangkaian yang lebih sederhana dibandingkan dengan LSTM. |
| 3 | (Boruah & Barman, 2018) | Long Short Term Memory | Identifikasi kalimat pada Novel *Bhanumati* ditulis oleh Assamese  scholar *Padmanath Gohainbaruah* | Epoch dengan akurasi maksimum untuk model yang berbeda diberikan. Akurasi rata-rata yang terlihat setelah setiap seribu epoch menampilkan kerugian rata-rata setelah setiap seribu epoch dari Uji 1, yang memiliki akurasi maksimum di antara model berbeda yang memiliki konfigurasi berbeda. Tetapi untuk bahasa Assam yang ditranskripsi secara fonetis pada peneliti dapat melihat bahwa akurasi rata-rata meningkat ketika peneliti meningkatkan jumlah neuron dari 128 menjadi 256 di setiap lapisan. Tetapi peningkatan lebih lanjut dari neuron menjadi 512 di setiap lapisan menurunkan akurasi. |
| 4 | (Supriyadi, 2019) | Long Short Term Memory | Observasi sinoptik Stasiun  Meteorologi Maritim Tanjung Priok | Metode deep learning LSTM digunakan untuk memprediksi parameter cuaca, seperti suhu udara, kelembaban, kecepatan angin, dan tekanan udara. Sedangkan jumlah datanya dibagi dua menjadi training data dan test data dengan rasio 9:1.pada bulan Januari 2019. Diperoleh RMSE parameter suhu udara, kelembaban, kecepatan angin, dan tekanan udara nilainya semakin baik ketika menggunakan deep learning LSTM dengan update dibandingkan LSTM tanpa update. Dari parameter cuaca tersebut hanya parameter suhu dan kelembaban udara yang mengalami pertambahan RMSE seiring bertambahnya waktu. |
| 5 | (Poornima & Pushpalatha, 2019) | LSTM dengan *Weighted Linear Units* | Dokumentasi kumpulan data *China Meteorological Administration* | Dalam penelitian tersebut menyajikan Metode Long Short-Term Memory dengan Weighted Linear Units untuk memprediksi curah hujan. Neural Network dilatih dan diuji menggunakan dataset standar curah hujan. Jaringan yang dilatih akan menghasilkan atribut prediksi curah hujan. Parameter yang dipertimbangkan untuk evaluasi kinerja dan efisiensi model prediksi curah hujan yang diusulkan adalah Root Mean Square Error, akurasi, jumlah epoch, loss, dan learning rate. |
| 6 | (Zahara, Sugianto, & Ilmiddafiq, 2019) | Long Short Term Memory | Dokumentasi secara online dari website *Dinas Perdagangan dan Perindustrian Provinsi Jawa Timur* | Pembangunan model prediksi dilakukan di lingkungan cloud computing Amazon Web Service tipe EC2. IHK prediksi pada bulan Desember 2016 yaitu 133.98, mempunyai nilai prediksi yang paling mendekati nilai IHK aktual yaitu 133.98 Dari hasil pengujian nilai RMSE tiap algoritma optimasi LSTM yang masih tergolong besar, terlihat bahwa metode LSTM belum bisa disebut metode yang maksimal dalam melakukan prediksi IHK. |
| 7 | (Aldi, Jondri, & Aditsania, 2018) | Long Short Term Memory | Dokumentasi dari website *blockchain.info* | Pada penelitian tersebut dibangun model LSTM untuk memprediksi harga Bitcoin dengan pengujian parameter komposisi data, jumlah pola time series, jumlah hidden neuron dan max epoch. Pada pengujian tersebut didapatkan hasil yang terbaik yaitu dengan komposisi data latih 70% dan data uji 30%, parameter 1 pola time series, jumlah  25 neuron hidden, dan max epoch adalah 100 dengan akurasi rata-rata pada data latih 95.36% dan data testing 93.5%. |
| 8 | (Putra, Osmond, & Ansori, 2020) | Long Short Term Memory | Dokumentasi data mentah dari Dinas Perindustrian dan Perdagangan Provinsi Jawa Barat | Ada 4 parameter yang akan diuji untuk setiap komoditas, yang terdiri dari Nilai Epoch, Nilai Lookback, Jumlah Hidden Layer, dan Nilai Train Batch. Jumlah hidden layer adalah banyaknya layer LSTM pada model, terdapat Single-Layer dan Multi-Layer dimana Multi-Layer sendiri terdiri dari 2 dan 3 hidden layer. Tujuan pengujian prediksi disini yaitu untuk menguji tingkat akurasi dan kecocokan model LSTM dalam memprediksi harga masa depan dengan menggunakan konfigurasi dari hasil pada pengujian model. |
| 9 | (Ghosh, Bose, Maji, Debnath, & Sen, 2019) | Long Short Term Memory | Dokumentasi dari website resmi *Bombay Stock Exchange* | Dalam penelitian tersebut, peneliti menganalisis pertumbuhan perusahaan dari berbagai sektor dan mencoba mencari tahu yang merupakan rentang waktu terbaik untuk memprediksi harga saham di masa depan. Prediksi bisa lebih akurat jika model akan berlatih dengan jumlah data yang lebih banyak mengatur. Kerangka kerja ini secara luas membantu dalam analisis pasar dan prediksi pertumbuhan perusahaan yang berbeda dalam rentang waktu yang berbeda. |
| 10 | (Rizki, Basuki, & Azhar, 2020) | Long Short Term Memory | Dokumentasi dari website BMKG | Aplikasi berhasil memproses prediksi curah hujan kota Malang dengan parameter curah hujan. Jumlah neuron hidden layer dengan hasil paling optimal yaitu dengan 256 neuron hidden layer. Jumlah epoch dengan hasil paling optimal yaitu 150 epoch. Komposisi Data Train dan Data Test dengan hasil yang paling optimal yaitu dengan komposisi data train 50% dan data test 50%. Hal ini karena komposisi data train 50% dan data test 50% memiliki tingkat error yang paling rendah yaitu pada data train sebesar 12.079 dan pada data test sebesar 11.288. |

# BAB III METODOLOGI PENELITIAN

## Kerangka Penelitian

Kerangka penelitian merupakan rangkaian kegiatan yang akan dilakukan dalam suatu penelitian berupa rangkaian grafik yang menggambarkan alur proses penelitian proyeksi curah hujan daerah padang pariaman menggunakan deep learning dengan metode long short-term memory, sehingga langkah-langkah yang dilakukan oleh penulis dalam perancangan ini tidak melenceng dari pokok bahasan dan lebih mudah dipahami, yang diilustrasikan seperti pada **Gambar 12** berikut.



Gambar 12. Kerangka Penelitian

## Tahapan Penelitian

Tahapan penelitian merupakan langkah-langkah yang akan dilakukan untuk mempermudah dalam melakukan penelitian. Tahapan dalam penelitian ini terdiri dari penelitian pendahuluan, pengumpulan data, analisa, perancangan, implementasi, pengujian dan kesimpulan yang akan dijelaskan sebagai berikut :

### Penelitian Pendahuluan

Penelitian pendahuluan merupakan proses melakukan pendekatan terhadap objek penelitian. Tujuan dari Penelitian pendahuluan tak lain dapat memberikan solusi terhadap masalah yang di identifikasi. Permasalahan pada proyeksi curah hujan khusus daerah padang pariaman masih terbatas dari masukan data pemodelan, di mana dalam pemodelan tersebut masih memiliki kekurangan seperti penggunaan jumlah parameter, asumsi-asumsi matematis, dan formulasi persamaan yang cenderung rumit. Oleh karena itu dilakukanlah sebuah penelitian yang nantinya dapat membantu BMKG Klimatologi Padang Pariaman dalam melakukan prediksi curah hujan di daerah padang pariaman. Penelitian ini nantinya akan memerikan alternatif lain dalam melakukan prediksi curah hujan di daerah padang pariaman dengam menggunakan metode Long Short-Term Memory.

### Pengumpulan Data

Dalam Penelitian ini semua data bersumber dari Stasiun Klimatologi Kelas II Padang Pariaman dari tahun 1985-2021, dan beberapa pencarian referensi seperti buku-buku, karya-karya ilmiah maupun jurnal, baik yang ada diperpustakaan maupun yang ada di internet yang berhubungan dengan penelitian. Data juga didapat dari studi lapangan dengan melakukan wawancara secara langsung.

#### Waktu Penelitian

Penelitian dilakukan dengan merekap data-data pengamatan unsur-unsur cuaca yang dilakukan oleh Stasiun Klimatologi Kelas II Padang Pariaman secara umum yang di lakukan dari bulan November sampai dengan selesai.

Table 8. Jadwal Penelitian

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **No** | **Kegiatan** | **Bulan Ke** | | | | | | | | | | | | | | | | |
| **1** | | | | **2** | | | | **3** | | | | **4** | | | | |
| 1 | Penelitian Pendahuluan |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2 | Pengumpulan Data |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3 | Analisa |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4 | Perancangan |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5 | Implementasi Sistem |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 6 | Pengujian Sistem |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 7 | Pembuatan Laporan |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

#### Tempat Penelitian

Objek Penelitian di lakukan di Stasium Klimatologi Kelas II Padang Pariaman aygn beralamat di Jalan Raya Padang – Bukittinggi KM. 51 Kapalo Hilalang Sumatera Barat.

#### Metode Penelitian

Metode pada Penelitian ini menggunakan metode kuantitatif yang terdiri dari penelitian perpustakaan, wawancara, penelitian laboratorium. Dengan penjelasan yang di uraikan sebagai berikut :

##### Penelitian Perpustakaan (Library Research)

Pada tahap ini peneliti melakukan studi literatur dengan membaca buku, jurnal, makalah dan laporan penelitian yang terkait dengan topik Penelitian. Tujuannya adalah untuk memperkuat permasalahan serta sebagai dasar teori dalam melakukan studi dan juga menjadi dasar untuk melakukan Penelitian terhadap prediksi curah hujan.

##### Wawancara (Interview)

Pada tahap ini peneliti dan responden berhadapan langsung (face to face) untuk mendapatkan informasi secara lisan dengan tujuan mendapatkan data yang dapat menjelaskan permasalah yang berhubungan dengan penelitian. Wawancara dilakukan dengan dengan supervisor di Stasiun Klimatologi Kelas II Padang Pariaman terkait dengan permasalahan untuk proses menganalisis masalah, menentukan informasi yang di butuhkan oleh peneliti, seperti pengambilan data, dan lain-lain.

##### Penelitian Laboratorium (Laboratorium Reaserch)

Penelitian laboratorium adalah metode penelitian yang di lakukan menggunakan laptop/personal computer (PC). Alat bantu tersebut di gunakan untuk mempraktekan sekaligus menguji keakuratan prediksi curah hujan daerah Padang Pariaman.

Adapun Spesifikasi hardware dan software yang di gunakan dalam melakukan penelitian laboratorium ini adalah sebagai berikut :

###### Perangkat Keras (Hardware)

1. Laptop ASUS Model A445LAB
2. CPU Intel® Core™ i3-5005U CPU @ 2.00GHz (4 CPUs)
3. Memory RAM 12GB
4. Partition Samsung SSD 870 EVO 250GB
5. GPU Intel(R) HD Graphics 5500

###### Perangkat Lunak (Software)

1. Operating System Windows 10 Pro 64bit (10.0, Build 19044)
2. Microsoft Office 365
3. Google Chrome
4. Visual Studio Code
5. Python Programming Language
6. MySQL Database Service
7. StarUML

### Analisa

Analisa dalam penelitian adalah penjabaran dari suatu masalah dari objek yang di teliti yang akhirnya menghasilkan suatu kesimpulan, dalam penelitian ini dimaksudkan untuk mengidentifikasi masalah dalam memprediksi curah hujan berdasarkan data harian klimatologi Dalam proses analisa terdapat tiga tahapan analisa yang harus dilakukan yakni tahapan analisa data, analisa proses dan analisa sistem dengan penjelasan sebagai berikut :

#### Analisa Data

Analisis data adalah sebuah proses untuk memeriksa, membersihkan, mengubah, dan membuat pemodelan data dengan maksud untuk menemukan informasi yang bermanfaat sehingga dapat memberikan petunjuk bagi peneliti untuk mengambil keputusan terhadap pertanyaan-pertanyaan penelitian. Data dalam Penelitian ini di dapat langsung dari Stasiun Klimatologi Kelas II Padang

#### Analisa Proses

Pada tahap analisa proses ini, peneliti akan merancang sebuah sistem dimana penulis data menentukan nilai epoch dan batch untuk metode Long Short Term Memory dan juga akan di lakukan uji validitas dengan melihat nilai error pada standard diviasi agar peneliti dapat melihat nilai ketepatan prediksi yang di lakukan. Dimana sistem tersebut diharapkan dapat menjadi solusi yang tepat untuk memecahkan permasalah curah hujan di daerah Padang Pariaman.

#### Analisa Sistem

Selanjutnya pada tahap analisis sistem ini merupakan tahapan yang sangat kritis dan sangat penting, di karenakan kesalahan pada tahapan analisis sistem akan menyebabkan juga kesalahan ditahap selanjutnya. Tahap analisis sistem merupakan dasar dalam merancang dan merencanakan sistem yang akan dibuat, analisa sistem dilakukan untuk mengetahui apa saja yang dibutuhkan oleh sistem. Analisa sistem dilakukan untuk mengetahui dan mengembangkan sistem yang sedang berjalan. Sistem ini memerlukan beberapa data klimatologi yang harus dicantumkan dalam aplikasi yang nantinya data tersebut akan dimasukkan dalam database agar terorganisir dengan baik.

### Perancangan

#### Perancangan Model

Tahapan perancangan ini, peneliti mengunakan *Unified Modeling Language (UML)* sebagai tools dalam menjelaskan alur analisa yang akan dibuat. Tahapan perancangan bertujuan untuk membuat penelitian dirancang sesuai dengan tujuannya, sehingga tidak melenceng dari tujuan Penelitian. Perancangan akan menggunakan UML sebagai model rancangan agar terorganisasi dan terstruktur dengan rancangan.

##### Usercase Diagram

*Usecase Diagram* memiliki satu actor yaitu, admin yang bertugas mengelolah data klimatologi dan melakukan penilaian untuk memprediksi curah hujan berdasarkan data yang telah di kelola.

##### Class Diagram

*Class Diagram* menggambarkan bagaimana strukturdari aplikasi yang akan dibentuk dalam merancang aplikasi prediksi curah hujan.

##### Sequence Diagram

*Sequence Diagram* menjelaskan urutan-urutan kejadian yang akan terjadi di dalam aplikasi, serta untuk menggambarkan interaksi antara keseluruhan objek di dalam aplikasi. *Sequence Diagram* menjelaskan interaksi objek yang disusun berdasarkan urutan waktu dalam bentuk gambaran tahap demi tahap yang seharusnya dilakukan untuk menghasilkan output sesuai dengan *Use Case diagram*. *Sequence Diagram* juga digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan oleh admin nantinya dalam melakukan input data dan melakukan proyeksi sesuai data curah hujan yang sudah di masukkan ke database.

##### StatechartDiagram

*Statechart Diagram* digunakan untuk menggambarkan perubahan status yang akan terjadi dari aplikasi proyeksi curah hujan dan objeknya selama aplikasi dijalankan sampai dengan selesai.

##### Activity Diagram

*Activity Diagram* hanya terdiri dari admin yang menggambarkan segala aktivitas yang dilakukan oleh admin terhadap sistem yang berupa aktivitas mengelola database dan aplikasi Proyeksi curah hujan. Diagram ini berhubungan dengan *State chart Diagram*. Jika State chart Diagram berfokus pada objek yang dalam suatu proses (atau proses menjadi suatu obyek), *Activity Diagram* berfokus pada aktifitas-aktifitas yang terjadi yang terkait dalam suatu proses tunggal. Jadi dengan kata lain, diagram ini menunjukkan bagaimana aktifitas-aktifitas yang di lakukan oleh admin.

##### Deployment Diagram

*Deployment Diagram* merupakan gambaran proses-proses berbeda pada suatu sistem yang berjalan dan bagaimana relasi didalamnya. Dan digunakan untuk menggambarkan detail bagaimana komponen disusun di infrastruktur sistem.

##### Collaboration Diagram

*Collaboration* menggambarkan interaksi antara objek seperti *sequence diagram*, tetapi lebih menekankan pada peran masing-masing objek dan buka pada waktu penyampaian message.

#### Perancangan Interface

Perancangan interface dibuat untuk memberikan penjelasan tentang tampilan yang dihadapkan pada pengguna saat menggunakan sistem. Pada tahap ini, peneliti melakukan perancangan terhadap tampilan atau desain antar muka (interface) dari aplikasi Proyeksi Curah Hujan.

Pada proses login admin yang sudah terdaftar pada Database dapat mengakses kedalam sistem. Admin yang telah login ke dalam sistem dapat mengakses Database untuk mengelola data informasi di dalam aplikasi tersebut.

### Implementasi

Implementasi sistem merupakan tahap meletakkan sistem sehingga siap untuk dioperasikan. Implementasi bertujuan untuk mengkonfirmasi modul-modul perancangan, sehingga pengguna dapat memberi masukan kepada pengembangan sistem. Pada tahap ini perancangan sistem dilakukan dengan menggunakan Bahasa Pemograman Python dan database MySQL.

### Pengujian

Pengujian sistem merupakan tahap melakukan testing untuk mengetahui kesalahan dalam Sistem Proyeksi Curah Hujan. Pengujian terhadap sistem dilakukan untuk dapat mengetahui Sistem yang dirancang telah berjalan sesuai dengan yang diharapkan. Sehingga memudahkan admin mengetahui informasi terkini dari sistem, dan kemudian mengambil tindakan selanjutnya jika dibutuhkan.

#### Pengujian Lokal

Pengujian Lokal atau LAN, yaitu pengujian yang dilakukan melalui akses jaringan komputer dengan sistem Local Area Network, dengan cara menghubungkan komputer server ke komputer client dengan media kabel, yang terlebih dahulu dilakukan configurasi IP Address pada komputer server dan komputer client. Sehingga client dapat mengakses data pada komputer server, sedangkan pada komputer server menyediakan informasi yang dibutuhkan oleh client.

#### Pengujian Online

Pembuatan program ini nantinya bisa diakses secara online dengan di hosting di tempat penyedia web service. Selanjutnya melakukan konfigurasi database agar terhubung ke web server. Setelah semua proses dilakukan maka website akan bisa diakses oleh admin yang ingin menggunakan aplikasi ini.

#### Pengujian Aplikasi

Pengujian Aplikasi dengan menggunakan Black Box, yaitu pengujian yang dilakukan hanya mengamati hasil eksekusi melalui data uji dan memeriksa fungsional dari perangkat lunak bahwa aplikasi yang dirancang dapat memenuhi kebutuhan yang mendasari perancangan aplikasi tersebut dan berjalan sesuai dengan diharapkan.

#### Pengujian Interface

Pengujian Antarmuka (Interface), yaitu pengujian yang dilakukan untuk melihat bagaimana tampilan akhir dari aplikasi yang telah dirancang dengan pengamatan secara langsung oleh pengguna interaksi secara langsung dengan model interface yang disajikan dalam bentuk prototype. Proses ini dilakukan agar kesalahan dapat terdeteksi dan di rubah secara cepat.

# BAB IV ANALISA DAN PERANCANGAN

## Analisa

Berdasarkan penjelasan yang telah diuraikan pada bab sebelumnya, maka diperlukan suatu proses penganalisan dengan tujuan untuk memberikan hasil yang akurat berdasarkan informasi maupun data yang sudah penulis dapatkan untuk di terapkan pada perancangan.

### Analisa Data

Analisa Data adalah tahap rentan dalam mengembangkan sebuah sistem yang padu. Dimana analisa data merupakan tahap awal agar sistem yang di rancang memiliki gambaran sesuai pola data yang terbentuk

Data dalam Penelitian ini data di dapat langsung dari Stasiun Klimatologi Kelas II Padang Pariaman, data yang di gunakan merupakan data harian dari tanggal **1 January 1985** sampai **21 Desember 2021** data tersebut terdiri dari beberapa fitur variabel seperti yang terlihat pada **Table 9** berikut :

Table 9. Fitur Variabel Data Klimatologi BMKG Padang Pariaman

|  |  |  |
| --- | --- | --- |
| Kode | Keterangan | Satuan |
|  | Temperatur minimum |  |
|  | Temperatur maksimum |  |
|  | Temperatur rata-rata |  |
|  | Kelembapan rata-rata |  |
|  | Curah hujan |  |
|  | Lamanya penyinaran matahari |  |
|  | Kecepatan angin maksimum |  |
|  | Arah angin saat kecepatan maksimum |  |
|  | Kecepatan angin rata-rata |  |
|  | Arah angin terbanyak |  |

Sesuai data Klimatologi yang di tampilakan pada **Table 9** dimana setiap fitur variabel pada data tersebut merupakan variabel in-dependen. Tidak semua fitur data pada **Table 9** akan di gunakan pada Penelitian ini, dan peneliti hanya menyertakan fitur tertentu seperti , , , dan .

Pada data klimatologi, data yang bernilai 8888 berarti data tidak diukur dan data yang bernilai 9999 berarti tidak ada data (tidak dilakukan pengukuran). Nilai tersebut akan dianggap nilai yang hilang “NaN”.

Dalam Penelitian ini data yang di gunakan memiliki beberapa nilai yang hilang (*missing values*) dan juga data bernilai *NaN* dari data yang akan di teliti. Nilai-nilai yang hilang / *NaN* tersebut ini muncul dari banyak factor yang berada di luar kendali staf Stasiun Klimatologi Kelas II Padang Pariaman, yang dimana nantinya peneliti akan melakukan cara penanganan hal dengan melakukan interpolasi linear pada data dengan nilai yang hilang / *NaN*.

### Analisa Proses

Analisis proses dilakukan untuk mengetahui cara pemecahan masalah sehingga dapat menghasilkan solusi dengan menggunakan metode yang tepat, melalui proses yang kooperatif dan interaktif mulai dari menganalisis masalah, mengidentifikasi masalah, hasil akhir pengamatan dalam berbagai format representasi, hingga memeriksa ketepatan pemahaman yang diperoleh.

Setelah dilakukan Analisa data, selanjutnya data yang telah melewati tahap pre-processing akan di persiapkan untuk di gunakan dalam pemodelan. Persiapan ini seperti memisahkan kumpulan data menjadi 2 bagian yaitu *train set* dan *test set,* dan normalisasi kumpulan data. Pada pemodelan *Long Short-Term Memory* input data akan berupa tensor 3D berupa (*batch\_size, timesteps, dan features*). Nilai pobabilitas dropout di set 0.1, Normalisasi yang akan di gunakan adalah normalisasi MinMaxScaller, kemudian fungsi oiptimasi yang di gunakan adalah optimizer ADAM, dan untuk melihat nilai kesalahan / error value dari peramalan digunakan loss function RMSE.

### Analisa Sistem

#### Analisa Sistem Yang Sedang Berjalan

## Perancangan

…

### Perancangan Model

### Perancangan Interface