COVER BUKU

(Desain cover original)

Penulis :

Mahasiswa +

Dosen Pembimbing

Implementasi Aplikasi Algoritma Regresi Linier Berganda Untuk Memprediksi Gaji Pegawai

Syafrial Fachri Pane, Bachtiar Ramadhan

Nur Tri Ramadhanti Adiningrum

# D4 Teknik Informatika, Politeknik Pos Indonesia

Jalan Sari Asih No.54, Kota Bandung, Jawa Barat

Judul :

Implementasi Aplikasi Algoritma Regresi Linear Berganda Untuk Memprediksi Gaji Pegawai

2022,

... Halaman, 16 cm x 23 cm

Penulis :   
Bachtiar Ramadhan

Nur Tri Ramadhanti Adiningrum

Syafrial Fachri Pane

Penyunting :

Roni Habibi

Layout :

Bachtiar Ramadhan

Nur Tri Ramadhanti Adiningrum

**Sanksi Pelanggaran Pasal 113**

**Undang-Undang Nomor 28 Tahun 2014**

**tentang Hak Cipta**

1. Setiap Orang yang dengan tanpa hak melakukan pelanggaran hak ekonomi sebagaimana dimaksud dalam Pasal 9 ayat (1) huruf i untuk Penggunaan Secara Komersial dipidana dengan pidana penjara paling lama 1 (satu) tahun dan/atau pidana denda paling banyak Rp100.000.000 (seratus juta rupiah).
2. Setiap Orang yang dengan tanpa hak dan/atau tanpa izin Pen cipta atau pemegang Hak Cipta melakukan pelanggaran hak ekonomi Pencipta sebagaimana dimaksud dalam Pasal 9 ayat (1) huruf c, huruf d, huruf t dan/atau huruf h untuk Penggunaan Secara Komersial dipidana dengan pidana penjara paling lama 3 (tiga) tahun dan/atau pidana denda paling banyak Rp500.000.000,00 (lima ratus juta rupiah).
3. Setiap Orang yang dengan tanpa hak dan/atau tanpa izin Pencip ta atau pemegang Hak Cipta melakukan pelanggaran hak ekonomi Pencipta sebagaimana dimaksud dalam Pasal 9 ayat (1) huruf a, huruf b, huruf e, dan/atau huruf g untuk Penggunaan Secara Komersial dipidana dengan pidana penjara paling lama 4 (empat) tahun dan/atau pidana denda paling banyak Rp1.000.000.000,00 (satu miliar rupiah).
4. Setiap Orang yang memenuhi unsur sebagaimana dimaksud pada ayat (3) yang dilakukan dalam bentuk pembajakan, dipidana dengan pidana penjara paling lama 10 (sepuluh) tahun dan/atau pidana denda paling banyak Rp4.000.000.000,00 (empat miliar rupiah).

# Kata Sambutan

Puji syukur kami haturkan kehadirat Allah SWT. yang telah melimpahkan kekuatan dan kesabaran, karena berkat rahmat dan hidayah-Nya, sehingga penulisan buku dengan judul “Implementasi Aplikasi Algoritma Regresi Linier Berganda Untuk Memprediksi Gaji Pegawai” yang telah lama dipersiapkan dapat diselesaikan dengan baik. Penyusunan buku ini mendapatkan banyak masukan dan motivasi dari berbagai pihak terutama rekan-rekan yang telah berkolaborasi dalam penulisan buku ini. Buku ini bertujuan untuk memberikan wawasan tentang penerapan regresi linier berganda pada bidang informatika terutama dalam pembuatan aplikasi prediksi. Maka dari itu, pada kesempatan ini kami ingin mengucapkan terima kasih yang sebesar-besarnya kepada seluruh pihak atas keihklasannya dalam melakukan proses penyusunan buku ini hingga selesai sesuai rencana serta sesuai dengan kaedah-kaedah penulisan karya ilmiah. Penulis juga menghanturkan terima kasih kepada orang tua dan keluarga yang sangat kami cintai, yang selalu memberikan motivasi dan selalu berdoa atas setiap langkah kebenaran yang kami lakukan. Semoga buku ini kelak bermanfaat untuk penulis maupun siapa saja yang membaca buku ini.

Penulis menyadari bahwa dalam penyusunan buku ini terdapat banyak kekurangan dan jauh dari sempurna serta menyadari bahwa buku ini hanya merupakan sebagian kecil dari luasnya pengetahuan dalam algoritma regresi linier berganda. Oleh karena itu, segala bentuk masukan baik saran ataupun kritik yang berharga dari berbagai pihak untuk membangun kesempurnaan buku ini sangat kami harapkan.

Terima kasih Penulis

# Kata Pengantar

Penulisan buku ini dimaksudkan untuk memberikan arahan atau sarana pembelajaran kepada pembaca mengenai implementasi algoritma regresi linier berganda untuk melakukan prediksi gaji pegawai dengan menggunakan bahasa pemrograman python dan framework Django.

Penyusunan buku ini sangat membutuhkan waktu dan pemikiran yang mendalam. Oleh karenan itu, segala bentuk kritik dan saran yang membangun dari berbagai pihak akan begitu bermanfaat guna penyempurnaan buku di masa yang akan datang.

Akhir kata, penulis mengucapkan terima kasih yang setinggi-tingginya atas apresiasinya kepada berbagai pihak yang telah turut berpartisipasi dalam penyusunan dan penyempurnaan buku ini.

Semoga buku ini kelak memberikan manfaat bagi para pembaca dan penulis khususnya.

Penulis

# Daftar Isi

# Daftar Gambar

# Daftar Tabel

# Daftar Simbol

Bab 1

Pengenalan Aplikasi

## Latar Belakang

Perkembangan ilmu pengetahuan dan teknologi pada Revolusi Industri 4.0 semakin berkembang pesat. Revolusi Industri 4.0 sendiri mulai terjadi melalui rekayasa intelegensia dan internet of thing sebagai tulang punggung pergerakan dan konektivitas antara manusia dengan mesin[1]. Sehingga, terdapat penggabungan teknologi digital dan internet dengan industri konvensional, yang bertujuan untuk meningkatkan produktivitas, efisiensi dan layanan konsumen secara signifikan[2]. Era revolusi ini akan mendisrupsi berbagai kegiatan diberbagai bidang seperti pada bidang teknologi, ekonomi, sosial, dan politik[1]. Saat ini, kehidupan berada diawal revolusi yang secara mendasar mengubah cara hidup, bekerja, dan berhubungan satu sama lain [3].

Perubahan karakteristik pekerjaan adalah salah satu dampak tersendiri dari datangnya revolusi industri 4.0[4]. Karakteristik pekerjaan yang berubah akan mendisrupsi pekerjaan yang telah ada dan menggantikanya dengan pekerjaan dengan karakteristik baru [5]. Karakteristik baru pada pekerjaan juga membutuhkan kompetensi baru kepada para pekerja[6]. Tentunya perusahaan harus siap untuk saling bersaing dengan perusahaan yang lain[7]. Selanjutnya, perusahaan perlu memiliki keunggulan dan manajemen yang efektif untuk menghadapi persaingan tersebut[7]. Dengan demikian salah astu aspek yang berpengaruh besar terhadap kemajuan dan keberhasilan sebuah perusahaan adalah kinerja pegawainya[7]. Walaupun perusahaan tersebut memiliki teknologi yang canggih, namun tidak terdapat tenaga kerja didalamnya, perusahaan tidak akan dapat mencapai tujuannya[7].

Oleh karena itu, penentuan gaji yang tepat oleh perusahaan kepada pegawai adalah salah satu faktor yang berpengaruh secara internal terhadap kemajuan perusahaan. Selain itu, perusahaan juga harus bersedia mengeluarkan gaji bonus bagi pegawainya yang telah bekerja dengan maksimal dan sesuai dengan apa yang dibutuhkan oleh sebuah perusahaan. Sangat disayangkan, perkembangan perusahaan saat ini belum memliki suatu media keputusan untuk melakukan prediksi gaji pegawai berdasarkan kualitas data.

Karakteristik dataset yang digunakan untuk memprediksi gaji pegawai terdiri dari parameter-parameter berdasarkan faktor-faktor spesifik. Selanjutnya faktor-faktor tersebut akan diuji validitas dan korelasinya menggunakan pendekatan machine learning. Faktor-faktor tersebut akan diambil berdasarkan pedoman interpretasi koefisien korelasi [8]. Untuk menentukan faktor yang dominan terhadap prediksi gaji, maka koefisien korelasi yang akan digunakan adalah tingkat hubungan sedang, kuat, dan sangat kuat. Metode yang digunakan pada machine learning yaitu regression. Regression digunakan untuk melakukan prediksi gaji pegawai. Tentunya hasil prediksi gaji pegawai perlu divisualisasikan secara realtime untuk dapat digunakan oleh perusahaan dalam menentukan keputusan dengan cepat. Visualisasi hasil prediksi tersebut akan ditampilkan berbasis web base dengan framework Django.

## Identifikasi Masalah

Berikut adalah identifikasi permasalahan dari penelitian ini :

1. Bagaimana menganalisis karakteristik data gaji pegawai pada perusahaan?
2. Bagaimana cara menganalisis uji validitas dan korelasi dari dataset gaji pegawai pada perusahaan?
3. Bagaimana cara membuat model prediksi gaji pegawai yang tepat?
4. Bagaimana cara melakukan visualisasi data dari hasil model prediksi gaji pegawai?

## Tujuan

Tujuan penilitian ini sebagai berikut :

1. Menganalisis data kepegawaian yang berkaitan dengan gaji pegawai.
2. Menganalisis uji validitas dan korelasi dari dataset gaji yang terdiri dari parameter gaji pegawai dan faktor-faktor terhadap prediksi gaji.
3. Membuat model prediksi dengan pendekatan machine learning menggunakan regresi.
4. Merancang sistem berbasis web base dengan framework Django.

## Manfaat

Manfaat penilitian ini sebagai berikut :

1. Sebagai wadah untuk memberikan inovasi baru dalam hal melakukan prediksi gaji pegawai.
2. Bagi perusahaan, penelitian ini diharapkan dapat membantu dalam menentukan keputusan yang tepat dalam penentuan gaji pegawai di masa yang akan datang.
3. Penelitian ini dapat menjadi memberikan kontribusi untuk melakukan pengembangan lebih lanjut.

## Ruang Lingkup

Ruang lingkup pada penelitian ini sebagai berikut :

1. Dataset yang digunakan adalah gaji pegawai yang terdiri dari parameter-parameter yang spesifik.
2. Framework yang digunakan untuk membuat visualisasi prediksi dari machine learning yaitu Django.
3. Metode regresi digunakan untuk menguji korelasi dari parameter yang digunakan untuk memprediksi gaji pegawai.

## Sistematika Penulisan

Dalam buku ini, terdapat dari lima bab yang mana setiap bab-nya memiliki pembahasan yg berbeda-beda. Berikut ini adalah pemaparan setiap bab.

1. BAB I PENGENALAN APLIKASI

Bab ini menjelaskan terkait dengan bagaimana cara untuk melakukan prediksi gaji pegawai.

1. BAB II LANDASAN TEORI

Bab ini menjelaskan keterhubungan penelitian-penelitian terdahulu yang mengkaji prediksi gaji dengan pendekatan machine learning.

1. BAB III GAMBARAN OBJEK STUDI

Bab ini menjelaskan mengenai objek studi dilakukan pada penelitian.

1. BAB IV METODOLOGI PENELITIAN

Bab ini menjelaskan metodologi yang digunakan dalam melakukan penyelesaian masalah.

1. BAB V PEMBUATAN APLIKASI

Bab ini menjelaskan tentang proses perancangan aplikasi prediksi menggunakan bahasa pemrograman python dengan framework django.

1. BAB VI KESIMPULAN DAN SARAN

Bab ini berisikan kesimpulan dan saran dari penelitian yang sudah dilakukan.

Bab 2

Landasan Teori

## Teori Umum

### Prediksi

Prediksi yaitu suatu proses untuk memperkirakan secara sistematis tentang sesuatu yang paling mungkin terjadi di masa yang akan datang, berdasarkan informasi dari masa lalu dan sekarang yang dimiliki, agar suatu kesalahan (selisih antara sesuatu yang terjadi dengan hasil perkiraan) mampu diperkecil. Prediksi tidak harus memberikan jawaban secara tepat kejadian yang akan terjadi, melainkan berusaha untuk mencari jawaban sedekat mungkin di masa yang akan terjadi. [20]

Prediksi adalah salah satu unsur yang dapat digunakan untuk mendukung dalam pengambilan keputusan yang tepat. Prediksi dalam pengambilan keputusan didasarkan berdasar data yang ada pada waktu sekarang dan waktu lampau (telah terjadi), sehingga dapat digunakan untuk mendeskripsikan suatu kondisi yang sesuai dengan target yang ingin dicapai. [21]

### Gaji

Menurut Sukirno (2013:350), “Gaji dapat diartikan sebagai pembayaran (upah) kepada tiap pekerja-pekerja tetap dan tenaga kerja professional, seperti pegawai pemerintah, dosen, guru, manajer, dan akuntan” [22]

### Pegawai

Seorang ahli bernama Soedaryono (2000:6) yang terdapat di dalam bukunya dengan berjudul “Tata Laksana Kantor”, disebutkan bahwa pengertian dari pegawai yaitu “Seseorang yang sedang melakukan penghidupannya dengan cara bekerja atau melakukan kegiatan berbayar dalam kesatuan organisasi, baik dari kesatuan kerja pemerintah ataupun dari kesatuan kerja swasta”. Berikut, menurut Wijaya (2002:15), istilah pegawai mengandung pengertian sebagai berikut:

1. Menjadi anggota dari suatu kerjasama (organisasi) dengan maksud untuk memperoleh balas jasa/imbalan kompensasi atas jasa yang telah diberikan.
2. Berada dalam sistem kerja yang bersifat lugas/pamrih.
3. Berkedudukan selaku penerima kerja serta berhadapan dengan pihak memberi kerja.
4. Kedudukan sebagai penerima kerja itu diperoleh setelah melalui proses penerimaan dari perekrutan.
5. Akan menghadapi masa pemberhentian (pemutusan hubungan kerja).

Lalu kemudian menurut Robbins pada tahun 2006 yang di dalam bukunya dengan berjudul “Perilaku Organisasi” pengertian pegawai didefinisikan sebagai “Orang pribadi yang melakukan kerja kepada pemberi kerja, baik sebagai pegawai tetap ataupun tidak berdasarkan kesepakatan kerja baik tertulis ataupun tidak tertulis, untuk melangsungkan suatu pekerjaan dalam jabatan atau kegiatan tertentu yang telah ditetapkan oleh pemberi kerja.” [19]

### Aplikasi

Aplikasi adalah ….

### Data

Data merupakan komponen utama data sistem informasi perusahaan karena semua informasi untuk pengambilan keputusan berasal dari data. Ada berbagai gagasan dalam pendefinisian data, salah satu sumber menyebutkan bahwa data adalah fakta-fakta, pemikiran, ataupun pendapat yang tidak atau belum mempunyai arti kegunaannya. Sedangkan pengertian lain dari data dapat didefinisikan sebagai kelompok teratur simbol-simbol yang mewakili, tindakan, benda, dan sebagainya. Data dapat terbentuk dari karakter yang dapat berupa bentukan alphabet, angka, maupun simbol khusus dan merupakan bentukan yang masih mentah, sehingga diperlukan pengolahan lebih lanjut melalui suatu model untuk menghasilkan informasi. [18]

### Unifed Modeling Language (UML)

UML adalah singkatan dari Unified Modeling Language yang didefinisikan sebagai sekumpulan alat yang digunakan untuk melakukan abstraksi terhadap sebuah sistem atau software berbasis objek. UML juga dapat dikatakan sebagai bahasa spesifikasi standar yang digunakan untuk mendokumentasikan, menspesifikasikan, dan membangun suatu sistem perangkat lunak.

UML adalah suatu bentuk metodologi untuk mengembangkan sistem OOP dan sekelompok perangkat tools untuk mendukung pengembangan sistem tersebut. Sehingga, UML juga dapat menjadi salah satu cara untuk mempermudah dalam melakukan pengembangan aplikasi yang berkelanjutan. Oleh karenanya, UML juga dapat menjadi suatu alat bantu untuk transfer ilmu tentang sistem yang akan dikembangkan dari satu developer ke developer lainya.

## Bahasa Pemrograman

### Python



Python adalah ....

### HTML



Html adalah …

### CSS



CSS adalah ….

### Django



Django ialah sebuah web framework berbasis bahasa pemrograman Python yang didesain untuk membuat suatu aplikasi web yang dinamis, kaya fitur dan aman. [27] Django yang dikembangkan oleh Django Software Foundation terus mendapatkan perbaikan sehingga membuat web framework yang satu ini menjadi pilihan utama bagi banyak pengembang aplikasi web. [27]

### Model View Template

### Model

### View

### Template

## Metode Yang Digunakan

### Teknik Pengumpulan Data

Dalam penelitian ini, dataset yang digunakan dalam proses regresi linier berganda adalah dataset kepegawaian yang bersumber dari Kaggle. Dataset yang digunakan adalah employee\_attrition\_train.csv sebanyak 1029 baris dan 35 kolom, dan employee\_attrition\_test.csv sebanyak 441 baris dan 34 kolom. Dataset tersebut dapat dilihat pada link sebagai berikut : https://www.kaggle.com/colearninglounge/employee-attrition

### Machine Learning

Machine learning dapat diartikan sebagai aplikasi komputer dan algoritma matematika yang diadopsi dengan cara pembelajaran yang berasal dari data dan dapat menghasilkan suatu prediksi di masa yang akan datang. [24]

### Regresi Linier

Regresi linier (*linear regression*) adalah metode yang dipakai untuk memperoleh model hubungan antara variabel dependen dan variabel independen (Walpole et al., 2012). Terdapat dua jenis regresi linier, yaitu terdiri dari regresi linier sederhana (*simple linear regression*) dan regresi linear berganda (*multiple linier regression*). Perbedaan terdapat pada banyaknya variabel independen. Regresi linier sederhana adalah metode yang dipakai untuk mendapatkan model hubungan antara satu variabel dependen dan satu variabel independent lainnya, sedangkan regresi linier ganda adalah metode yang digunakan untuk memperoleh model hubungan antara satu variabel dependen dengan lebih dari satu variabel independen lainnya (Walpole et al., 2012). [25]

### Metode Regresi Linier Berganda

Regresi berganda adalah perpanjangan dari regresi linier sederhana. [25] Analisis regresi linier berganda dapat digunakan untuk memprediksi hubungan antara satu variabel independen berdasarkan nilai dari dua atau lebih variabel dependen. [25] Analisis regresi linier berganda juga menghasilkan persamaan matematis. [25] Jika ada lebih dari dua variabel maka hubungan linier dapat dinyatakan dalam persamaan regresi linier berganda yang dikutip pada persamaan 1 dan persamaan 2 sebagai berikut :

(1)

(2)

Keterangan :

= nilai-nilai hasil pengamatan

= nilai regresi

i = 1,2,3, k

Pada persamaan di atas ada satu variabel dependen, yaitu Y’ dan ada n variabel independen X1, X2, …Xn. [25]

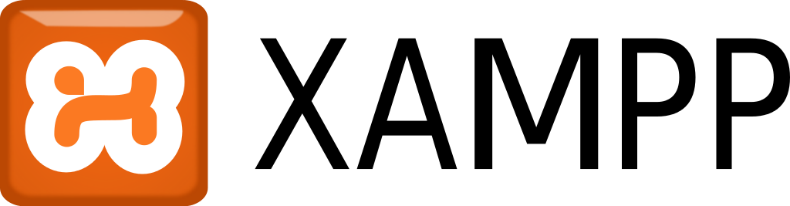
## Software

* + 1. **Visual Studio Code**



Pengertian

* + 1. **XAMPP**



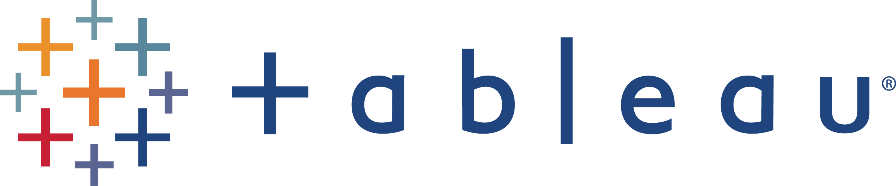
Pengertian

* + 1. **Jupyter Notebook**



Pengertian

* + 1. **Tableau**



Pengertian

* + 1. **Anaconda**



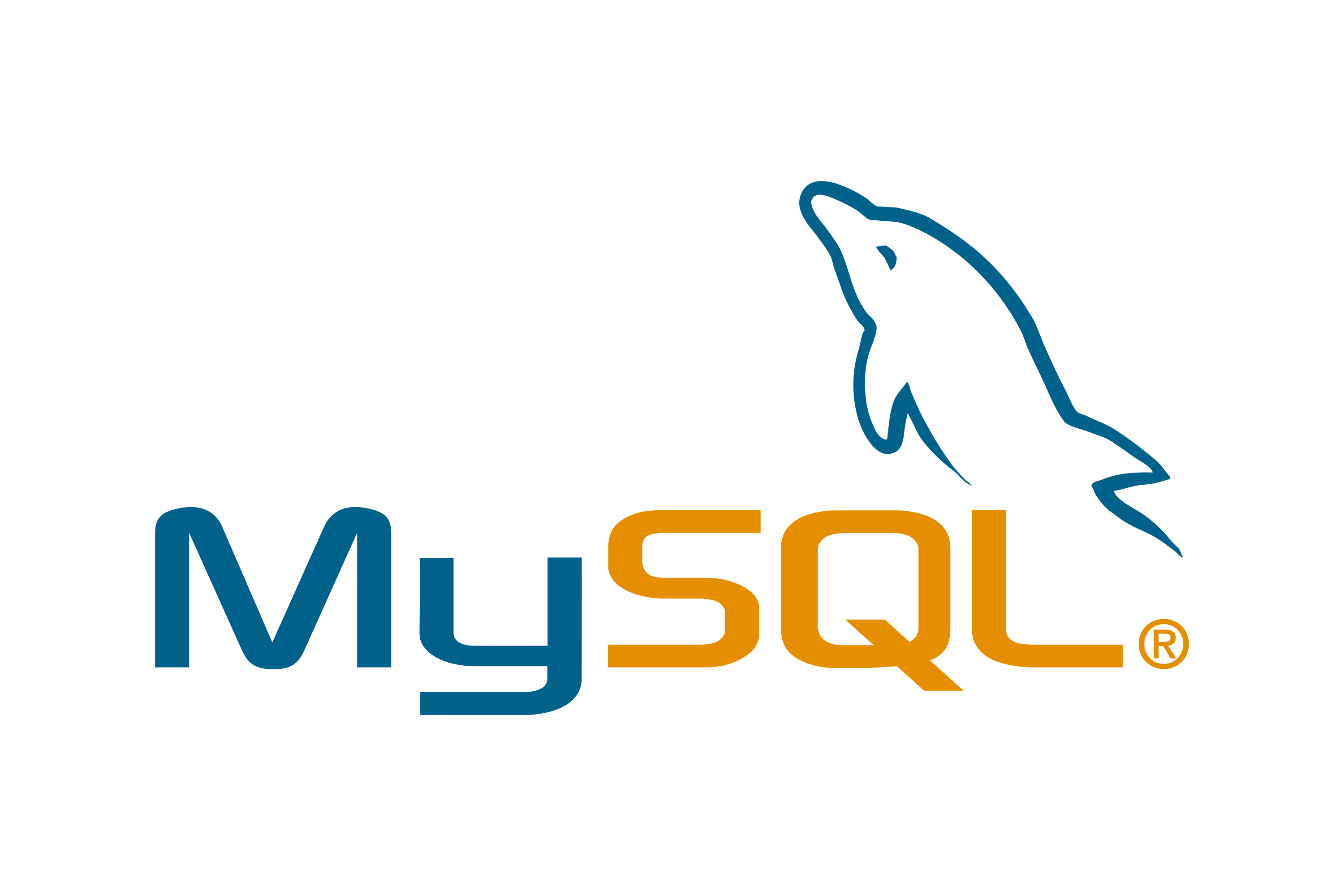
Pengertian

## Penyimpanan

### Database

Database atau basis data adalah kumpulan informasi yang disimpan di dalam komputer secara sistematik sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut. Kegunaan utama sistem basis data adalah agar pemakai mampu menyusun suatu pandangan (*view*) abstraksi data. Hal ini bertujuan untuk menyederhanakan interaksi antara pengguna dengan sistemnya dan basis data dapat mempresentasikan pandangan yang berbeda kepada para pengguna, programmer, dan administratornya. [28]

### MySQL



Mysql adalah…..

## Tinjauan Pustaka

### Deskripsi Metode Yang Sama

1. **Implementasi Model Regresi Linear Sederhana Untuk Prediksi Gaji Berdasarkan Pengalaman Lama Bekerja**

Yayan Adrianova Eka Tuah, Anyan (Program Studi Pendidikan Komputer, STKIP Persada Khatulistiwa Sintang)

Suatu perusahaan tidak dapat dipisahkan dari tenaga kerja. Salah satu faktor terpenting dalam mendongkrak kinerja tenaga kerja ialah pemberian gaji atau upah yang sesuai berdasarkan waktu lama karyawan tersebut bekerja. Penelitian ini bertujuan untuk mengetahui prediksi gaji karyawan berdasarkan tahun lama masa kerja dari seorang karyawan. Penelitian ini menggunakan metode pengumpulan data dan juga kuisioner. Teknik analisis data yaitu analisis regresi linear dalam python machine learning untuk mengetahui seperti apa pengaruh variabel bebas terhadap variabel terikat. Hasil dari penelitian ini ialah adanya suatu pengaruh positif dan signifikan antara gaji dengan masa kerja terhadap kinerja dari karyawan. Berdasarkan pengujian dengan menggunakan fungsi regresi linear untuk memprediksi besarnya gaji dari pengalaman bekerja, diketahui bahwa semakin bertambahnya pengalaman bekerja seorang karyawan, maka gaji karyawan tersebut juga akan bertambah setiap tahunnya. Hasil tersebut memperlihatkan bahwasannya pengalaman bekerja memberikan pengaruh positif terhadap besarnya gaji karyawan. [1]

1. **Data Mining Menggunakan Regresi Linear Untuk Prediksi Harga Saham Perusahaan Pelayaran**

Ekka Puji Ariesanto Akhmad (Program Studi Ketatalaksanaan Pelayaran Niaga, Program Diploma Pelayaran, Universitas Hang Tuah)

Pergerakan harga penutupan saham dari PT. BULL cenderung mengalami variasi setiap harinya. Investor perlu melakukan tindakan yang tepat, sehingga risiko yang ada dapat dikurangi dengan mengetahui naik turunnya harga saham pada masa yang akan datang, dan mampu memprediksi langkah kebijakan yang optimal untuk membuat keputusan pembelian/penjualan saham yang sesuai. Tujuan penelitian ini untuk menerapkan data mining menggunakan regresi linear untuk prediksi harga saham pada perusahaan pelayaran. Dalam penelitian ini menggunakan metodologi berupa Cross Industry Standard Process for Data Mining (CRISP-DM). Hasil dari penelitian tersebut menunjukkan bahwa masih ada selisih antara harga penutupan saham luaran data testing dengan harga penutupan saham aktual yang terdapat di bursa saham. Evaluasi dari nilai Root Mean Square Error menunjukkan angka plus 7,522 dari data aktual harga penutupan saham periode harian PT. BULL. [2]

1. **Machine Learning Untuk Model Prediksi Harga Sembako Dengan Metode Regresi Linier Berganda**

Kandari Puteri dan Astried Silvianie (Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Institut Bisnis dan Informatika (IBI) Kosgoro 1957)

Harga sembilan bahan pokok atau yang biasa disebut sembako, setiap waktu dapat naik dan turun, serta kebutuhan akan informasi harga dari sembako harian. Oleh karena itu diperlukan suatu peramalan harga pada sembako harian untuk diketahui pada beberapa waktu ke depan. Salah satu metode yang dapat digunakan untuk melakukan prediksi harga dengan tipe numerik kontinu ialah dengan menggunakan metode regresi. Penelitian ini bertujuan untuk mengimplementasikan metode regresi linier berganda dalam memprediksi harga dari sembako. Data yang digunakan adalah menggunakan sampel data sembako di DKI Jakarta. Hasil yang didapatkan, sistem machine learning dapat digunakan sebagai alat bantu untuk memprediksi harga sembako harian baik harga pada masa lampau, saat ini maupun masa yang akan datang (masa depan yang akan terjadi). [3]

1. **Prediksi Harga Rumah Menggunakan Web Scrapping Dan Machine Learning Dengan Algoritma Linear Regression**

Andi Saiful, Septi Andryana, Aris Gunaryati (Universitas Nasional; Jl. Sawo Manila No.61, Pejaten Barat, Pasar Minggu, Jakarta Selatan, DKI Jakarta 12520, Jurusan Informatika, Universitas Nasional, DKI Jakarta).

Tempat tinggal ialah kebutuhan primer yang sangat penting untuk dimiliki manusia. Maka, sangat penting dalam membuat perencanaan agar nantinya setiap keluarga dapat memiliki tempat tinggal pribadi. Dalam perencanaan tersebut dibutuhkanlah suatu prediksi harga di masa yang akan datang. Tujuan dari penelitian ini yaitu untuk membuat suatu model prediksi harga rumah menggunakan metode machine learning adapun algoritmanya yaitu linear regression. Dengan dilakukannya web scraping untuk mengumpulkan data, melalui beberapa website yang bergelut pada bidang jual beli rumah. Adapun menurut developer rumah yang ditanyakan dilapangan terkait variabel yang mempengaruhi harga rumah. Dari hasil penerapan prediksi harga rumah, disimpulkan bahwa pengolahan awal data yang dilakukan pada data set 7442 data menjadi 794 data amat mempengaruhi pada tingkat akurasi dari prediksi harga rumah tersebut. Dengan menggunakan algoritma linear regression untuk memprediksi harga rumah dapat memberikan hasil keakuratan prediksi harga rumah dengan baik. [4]

1. **Metode Regresi Linier Untuk Prediksi Pengadaan Inventaris Barang**

Melisa Winda Pertiwi, Richardus Eko Indrajit

Teknik prediksi data mining pada persediaan barang diperlukan di beberapa tempat, dalam beberapa kasus ada masalah pengadaan, misalnya pada Dinas Pariwisata Pemuda Dan Olahraga Kota Tasikmalaya provinsi Jawa Barat. Hal ini disebabkan oleh kurangnya metode ilmiah untuk memprediksi pengadaan. Prediksi yang digunakan untuk pengadaan tahun depan diharapkan menghasilkan pengetahuan yang dapat digunakan untuk mengambil keputusan. Metode regresi linier merupakan salah satu metode yang dapat memprediksi (forecasting). Pada resource ini berbasis business intelligence yang berfokus untuk memprediksi persediaan barang dengan menggunakan metode regresi linier berbasis pada data sebelumnya, sehingga dapat diketahui keakuratan metode regresi linier terhadap data yang digunakan oleh RMSE (Root Mean Square Error). Penelitian ini berbasis business intelligence yang mana akan muncul suatu pengetahuan baru berdasarkan proses bisnis dari studi kasus untuk melakukan peramalan khususnya pada inventaris barang. Berdasarkan uji coba dataset inventaris barang menggunakan metode regresi linear, maka dapat disimpulkan bahwa metode ini baik terhadap dataset yang digunakan dengan menunjukan akurasi RMSE 0.94. [5]

1. **Prediksi Kasus Covid-19 Di Indonesia Menggunakan Metode Backpropagation Dan Regresi Linear**

Wahyudin, Heri Purwanto (Konsentrasi Teknik Informatika, Program Studi Manajemen Informatika, STMIK LPKIA Bandung)

COVID-19 pertama kali menyerang manusia di Wuhan, China sedangkan di Indonesia sendiri mulai menyerang dalam lepas 2 Maret 2020 terdeteksi dua orang telah terkonfirmasi positif. Dari kasus tersebut tiap harinya mengalami pertambahan yang relatif signifikan. Hingga saat ini masih belum menemukan obat atau vaksin yang bisa digunakan untuk mengatasi penyebaran virus COVID-19. Adapun maksud dilakukannya penelitian ini yaitu untuk dapat memperkirakan jumlah kasus aktif pada penambahan kasus COVID-19 di Indonesia. Pada penelitian ini akan dicobakan dengan menggunakan metode Backpropagation dan Regresi Linear. Hasil prediksi kasus aktif dengan Backpropagation memberikan hasil penambahan dan penurunan yang tidak terlalu signifikan sedangkan hasil prediksi kasus aktif dengan Regresi Linear menunjukan bahwa penambahan kasus untuk tiap harinya mengalami penambahan kasus aktif. Setelah penjelasan serta penjabaran pada bab sebelumnya, dapat dijelaskan bahwa, secara umum hasil prediksi kasus aktif dengan Regresi Linear menunjukan bahwa penambahan kasus untuk tiap harinya mengalami pertambahan kasus aktif COVID-19. [6]

1. **Perbandingan Regresi Linear, Backpropagatian Dan Fuzzy Mamdani Dalam Prediksi Harga Emas.**

Nur Nafi’iyah

Tujuan dari penelitian ini yaitu untuk melihat hasil dari regresi linear, backpropagation dan fuzzy mamdani dalam memprediksi harga emas. Regresi linear adalah suatu persamaan garis dari berbagai data yang dikumpulkan. Fuzzy mamdani yaitu algoritma fuzzy yang menggunakan nilai yang crips dengan rentang 0-1. Sedangkan backpropagation ialah suatu algoritma nueral network yang sangat kompleks. Emas merupakan benda yang dapat digunakan untuk melakukan investasi. Sehingga apabila kita dapat memahami bagaimana pergeseran harga emas, maka kita mampu mendapatkan untung. Hasil dari ketiga metode tersebut menunjukkan bahwa korelasi yang didapatkan dari regresi linear sangat bagus yaitu 0,929. Dan nilai korelasi tertinggi dari ketiga metode berasal dari metode backpropagation. Hal ini dapat terbukti bahwa dalam memprediksi harga emas dengan menggunakan backpropagation lebih sedikit errornya ± 0,05. [7]

1. **Implementasi Algoritma Regresi Linear Sederhana Dalam Memprediksi Besaran Pendapatan Daerah (Studi Kasus : Dinas Pendapatan Kab. Deli Serdang).**

Fransiskus Ginting, Efori Buulolo, Edward R Siagian.

Data Mining yaitu suatu penemuan informasi dengan cara melakukan penggalian pola informasi yang dapat berisi pencarian trend dalam sejumlah data yang sangat besar serta dapat membantu dalam proses penyimpanan data untuk mengambil suatu keputusan diwaktu yang akan datang. Dalam menentukan pola dilakukan suatu teknik klasifikasi dengan mengumpulkan record (Training set). Pendapatan daerah umumnya berasal dari pajak dan retribusi daerah. Pajak daerah merupakan salah satu dari sumber pendanaan untuk daerah secara rata-rata nasional belum mampu memberikan kontribusi yang besar dalam pembentukan pendapatan asli daerah. Dengan memanfaatkan data Pendapatan Daerah dapat menghasilkan peramalan dan prediksi penghasilan Pendapatan Daerah kedepannya agar sesuai dengan kenyataan/realitas sehingga RAPBD yang telah direncanakan dapat berjalan dengan lancar. Regresi Linear Sederhana atau SLR (Simple Linear Regression) adalah salah satu metode statistik yang dipergunakan dalam produksi untuk melakukan peramalan atau prediksi mengenai karakteristik kualitas maupun kuantitas untuk menggambarkan proses yang terkait dengan pengolahan data perolehan besaran pendapatan daerah. Sehingga dalam tahap pengujian dengan visual basic net dapat membantu dalam mengolah data Besaran Pendapatan Daerah yang valid. Berdasarkan proses hasil perhitungan yang telah dilakukan, maka dapat diketahui nilai prediksi besaran pendapatan daerah dinas pendapatan daerah kab. Deli Serdang. Dengan menggunakan algoritma regresi linear sederhana, dinilai dapat memprediksi besaran pendapatan daerah dinas pendapatan daerah kab. deli serdang untuk tahun berikutnya sehingga program-program yang telah direncanakan sebelumnya oleh dinas pendapatan dapat berjalan dengan lancar, dan juga dapat membuat program-program yang baru agar dapat meningkatkan pendapatan daerah untuk memajukan daerah tersebut. [8]

1. **Prediksi Tingkat Produksi Kopi Menggunakan Regresi Linear**

Petrus Katemba, Rosita Koro Djoh

Kopi dari daerah ini menjadi andalan ekspor hasil perkebunan, yangtelah menembus pasar internasional dengan harga tinggi karena mutunya yang begitu baik. Namun produksi kopi cenderung menurun yang mengakibatkan permintaan akan kopi mengalmi penurunan yangdisebabkan oleh beberapa faktor, baik faktor alam dan sistem yang digunakan masih tradisional. Ada banyak upaya peningkatan produksi kopi telah dilakukan pemerintah, namun lemahnya teknologi pendukung menjadi salah satu kendala peningkatan produksi kopi. Tujuannya ialah untuk mengetahui apakah produksi kopi mengalami peningkatan atau justru mengalami penurunan dari waktu ke waktu. Untuk memenuhi kebutuhan kopi, maka dilakukan prediksi dengan menggunakan Regresi linear sederhana yang merupakan salah satu metode statistik yang dipergunakan dalam produksi untuk melakukan prediksi tentang karakteristik kuantitas maupun kualitas tersebut. Simple Regresi Linear yang terdiri dari satu buah variable bebas (x) dengan satu buah variabel terikat (y). Dengan dilakukannya prediksi menggunakan Metode Regresi Linear, informasi dapat diberikan dan membantu para petani dan pemerintah dalam mengambil kebijakan guna meningkatkan produksi kopi di Kabupaten Manggarai. Hasil yang diperoleh dari penelitian yang dimana melibatkan 5 periode yaitu dari tahun 2011-2015, nilai tertinggi pada tahun 2015 adalah sebesar 1.537,38ton dan nilai terendah pada tahun 2011 adalah sebesar 1.109. Setelah dilakukannya pengujian dengan menggunakan MSE dan MAPE, diperoleh nilai MSE 43,112% dan MAPE20,001%. Sehingga pengujian menggunakan MAPE jauh lebih baik dalam menghitung akurasi prediksi produksi kopi. [9]

1. **Peramalan Penerimaan Mahasiswa Baru Universitas Samudra Menggunakan Metode Regresi Linear Sederhana.**

Tri Novriza Putri, Adam Yordan, Dara Havisha Lamkaruna.

Peramalan penerimaan mahasiswa baru di Universitas Samudra menggunakan metode regresi linear sederhana. Penelitian ini bertujuan untuk meramalkan atau memprediksi jumlah penerimaan mahasiswa baru dengan menganalisis data-data yang ada pada tahun-tahun sebelumnya, yang nantinya akan dipresentasikan ke kejadian masa depan dengan model matematis, perhitungan ini dapat berupa perhitungan yang menggunakan pendekatan baik kuantitatif atau kualitatif. Data yang digunakan pada penelitian ini yaitu data penerimaan mahasiswa di Universitas Samudra pada tahun 2014, 2015, 2016, 2017 dan 2019. Pada masalah ini digunakanlah suatu metode dalam data mining yaitu regresi linear sederhana. Hasil yang didapatkan pada penelitian ini yaitu penggunaan metode regresi linear sederhana dapat dipertimbangkan karena jumlah error yang didapat pada hasil prediksi penerimaan mahasiswa baru satu tahun kedepan tidak terlalu besar yaitu hanya sebesar 21 dari 1428 mahasiswa hasil prediksi, dimana data ini diuji pada data di tahun 2018 jika diprediksi pada tahun tersebut, banyak mahasiswa yang masuk pada tahun tersebut adalah sebanyak 1449. [10]

1. **Salary Prediction Using Regression**

Sayan Das, Rupashri Barik, Ayush Mukherjee.

Tujuan dari jurnal ini adalah untuk memprediksi gaji seseorang setelah tahun tertentu. Representasi grafis dari memprediksi gaji adalah proses yang bertujuan untuk mengembangkan sistem komputerisasi untuk mempertahankan seluruh pekerjaan sehari-hari grafik pertumbuhan gaji di bidang apapun dan dapat memprediksi gaji setelah jangka waktu tertentu. Aplikasi ini dapat mengambil database untuk sistem penggajian dari organisasi dan membuat grafik melalui informasi ini dari database. Model yang digunakan adalah model regresi linear dan regresi polinomial. Memilih gaji dari grafik x-y membutuhkan banyak percobaan karena mungkin ada lebih dari satu grafik yang sesuai. Prediksi ini benar hingga waktu tertentu. Akurasi dapat diperoleh dengan menerapkan regresi k-terdekat.

1. **Empirical Analysis Of Regression Techniques By House Price And Salary Prediction**

U. Bansa, A. Narang, A. Sachdeva, I. Kashyap, S.P. Panda.

Tujuan utama dari jurnal ini adalah untuk membandingkan kinerja dua teknik regresi yaitu algoritma Regresi Linier Sederhana (SLR) dan Regresi Linier Berganda (MLR) dengan dua kasus: memprediksi gaji karyawan setelah tahun-tahun tertentu dan memprediksi harga real estat. Gaji karyawan tergantung pada banyak faktor, seperti pengalaman total karyawan, sertifikasi, dan pengalaman keseluruhan sebagai pemimpin dan manajer. Faktor-faktor dalam memprediksi harga rumah adalah luas tanah (sqft\_living), kondisi, tepi laut, jumlah kamar tidur, dan sebagainya. Dataset yang digunakan dalam eksperimen ini adalah dataset open-source dari KaggleInc. Algoritma dibandingkan menggunakan parameter seperti nilai R-kuadrat, Mean absolute error (MAE), Mean Squared Error (MSE), Median Absolute Error (MDAE), Variance Score, dan Root Mean Square Error (RMSE). Hasil telah menunjukkan bahwa MLR memberikan efisiensi yang lebih baik dibandingkan dengan SLR. Karena pada data House price, MLR memiliki R-Square 0,67 dan SLR 0,49. Serta pada data prediksi gaji, MLR memiliki R-Square 0,92 dan SLR 0,75.

1. **PREDIKSI KASUS COVID-19 DI INDONESIA MENGGUNAKAN METODE BACKPROPAGATION DAN REGRESI LINEAR**

Wahyudin, Heri Purwanto.

Penelitian ini dimaksudkan untuk dapat memperkirakan jumlah kasus aktif pada penambahan kasus COVID-19 di Indonesia. Metode yang digunakan dalam penelitian ini adalah metode backpropagation dan regresi linear. Penelitian ini menggunakan data dari databooks yang terdiri dari atribut total kasus, kasus sembuh, kasus meninggal dan kasus aktif . Pada penelitian ini hanya menggunakan data kasus aktif atau pola kasus aktif besok (mendatang) atau besar kasus aktif per hari, namun akan dibatasi hanya 15 hari. Hasil dari penelitian tersebut ialah *monetary salary* dan *equity salary of senior managers, ordinary employee salary,* dan *the internal salary gap of the senior management team* semuanya berkorelasi positif secara signifikan pada tingkat 1% dengan efisiensi inovasi perusahaan. Korelasi antara *salary gap* antara *senior managers* dan *ordinary employees* dengan *enterprise innovation efficiency* tidak jelas. Koefisien korelasi regresi antar variabel berada dalam kisaran yang dapat diterima. Hal ini dapat menunjukkan bahwa model yang digunakan tidak memiliki multikolinearitas yang signifikan. Hasil prediksi dari kasus aktif dengan metode Backpropagation memberikan hasil berupa penambahan dan penurunan yang tidak terlalu signifikan sedangkan hasil prediksi kasus aktif dengan Regresi Linear menunjukan bahwa penambahan kasus untuk tiap harinya mengalami penambahan kasus aktif.

1. **REGRESI LINEAR BIVARIAT SIMPEL DAN APLIKASINYA PADA DATA CUACA DI CILACAP**

Saniyah dan Budi Pratikno.

Penelitian ini bertujuan untuk membahas mengenai metode regresi linear bivariat simpel pada data cuaca di Kabupaten Cilacap untuk mengetahui hubungan unsur-unsur dalam cuaca dan iklim yaitu temperatur udara, kelembaban udara, dan curah hujan dengan model regresilinear bivariat. Model regresi linear bivariat ini dapat memprediksi dan menganalisis hubungan antar dua variabel respon sekaligus yang saling berkorelasi dan perhitungan regresi linear bivariat ini biasanya menggunakan matrik. Variabel pada model regresi linear bivariat simpel ini menggunakan dua variabel respon, yaitu variabel curah hujan dan variabel kelembaban udara suatu wilayah, dan satu variabel prediktor, temperatur udara. Metode pengujian model persamaan regresi adalah uji Wilk’s Lamda, dengan nilai Wilk’s Lamda = 0,881101 bernilai lebih kecil dari lambda tabel 0,903. Model peramalan untuk Y1 adalah Y(1) = −894,130 + 45,892X yang berarti jika tempertur udara naik sebesar satu derajat Celcius maka curah hujan akan naik sebesar 45,892mm serta nilai rata-rata error bulanan = -0,00697mm dan MDEE(1) = 151,2132. Model peramalan untuk Y2 adalah Y(2) = 78,0433+0,1581X yang artinya jika tempertur udara naik sebesar satu derajat Celcius maka kelembaban udara akan naik 0,1581 persen seta nilai rata-rata error bulanan = 0,000441 persen dan MDE(2) = 1,206636. Hasil uji model [Y(1)  Y(2)] = [-894,130+45,892X 78,0433+0,1581X] menunjukan bahwa parameter keduanya significant, dengan mean deviation error model tersebut adalah [151, 2132 1, 206636].

1. **Analisis Regresi Linier Berganda Dalam Estimasi Produktivitas Tanaman Padi Di Kabupaten Karawang**

Tesa Nur Padilah dan Riza Ibnu Adam.

Karawang yang sebagai pusat penanaman padi, memang sudah seharusnya produktivitas padi dari tahun ke tahun mengalami peningkatan. Sayangnya, produktivitas padi ternyata tidak konsisten, hal itu dibuktikan ketika tahun 2015 mengalami kenaikan sedangkan tahun 2016 mengalami penurunan. Oleh karena itu, diperlukannya suatu estimasi sehingga dapat diketahui bagaimana produktivitas padi untuk tahun-tahun berikutnya. Oleh karena itu, masalah produktivitas padi di Kabupaten Karawang dapat diestimasi dengan menggunakan analisis regresi linier berganda. Model regresi linier berganda dapat dinyatakan dalam bentuk perkalian matriks. Selanjutnya, perhitungan nilai-nilai koefisien regresi dapat dicari dengan menggunakan eliminasi Gauss. Data yang digunakan adalah data perbulan yang berjumlah 57data. Variabel terikat pada penelitian ini adalah produktivitas padi (kw/ha). Berdasarkan model regresi didapat 80,46% faktor-faktor produktivitas padi. Variabel-variabel yang mempengaruhi peningkatan jumlah produktivitas padi yaitu variabel produksi dan curah hujan, sedangkan variabel-variabel yang mempengaruhi penurunan jumlah produktivitas yaitu variabel luas panen, luas tanam, dan hari hujan. kesalahan relatif regresi yang diperoleh yaitu 0,04642 atau 4,642%.

1. **MODEL REGRESI MULTIVARIAT ANALISIS KESEJAHTERAAN PEDAGANG KAKI LIMA BERDASARKAN KARAKTERISTIK SOSIAL EKONOMI**

Desak Putu Eka Nilakusmawati dan Made Susilawati.

Kurangnya penelitian terhadap pekerja sektor informal, khususnya PKL telah membuat diremehkannya kontribusi sektor ini terhadap pertumbuhan ekonomi dan pembangunan berkelanjutan. Sehingga penelitian ini dapat memberikan kontribusi untuk mengisi kesenjangan tersebut. Penelitian ini memiliki suatu tujuan yaitu untuk mengetahui bagaimana model tingkat kesejahteraan pedagang kaki lima berdasarkan karakteristik sosial ekonomi. Data dalam penelitian ini merupakan data kuantitatif yang diperoleh dari sumber primer, yaitu diambil secara langsung oleh peneliti menggunakan angket dan pengambilan sampel dilakukan dengan cara *purposive sampling.* Model yang digunakan dalam penelitian adalah analisis *multivariate* yaitu analisis regresi linier dan *logistic ordinal*. Variabel yang signifikan berpengaruh terhadap rata-rata pendapatan pedagang kaki lima adalah tingkat pendidikan, curahan jam kerja, dan jumlah tenaga kerja diluar tenaga kerja keluarga yang ikut membantu. Model rata-rata pendapatan pedagang kaki lima adalah Y = -1.982 + 0.654pendidikan + 0.134curahan jam kerja + 0.817Jumlah tenaga kerja non keluarga. Variabel yang berpengaruh signifikan terhadap status pekerjaan pedagang kaki lima yaitu status perkawinan, sifat layanan dagangan, curahan jam kerja, serta jumlah tenaga kerja diluar tenaga kerja keluarga. Model dugaan yang menjelaskan status pekerjaan pedagang kaki lima yaitu G(x) = -16.308 - 0.519 status kawin + 0,739 sifat layanan + 1,19663 curahan jam kerja - 1,062 Jumlah tenaga kerja non keluarga.

Bab 3

Gambaran Objek Studi

## Objek Studi

## A

## B

## C

Bab 4

Metodologi Penelitian

## Diagram Alur Metodologi Penelitian

## Tahapan Diagram Alur Metodologi Penelitian

### Perancangan Sistem

### Pengujian Sistem

Bab 5

Pembuatan Aplikasi

## Analisis

Analisis sistem ialah penjabaran dari suatu sistem informasi yang utuh ke berbagai macam bagian-bagian komponennya yang bertujuan untuk mengevaluasi permasalahan atau kendala yang terjadi pada suatu sistem, sehingga nantinya dapat dilakukan perbaikan ataupun pengembangan pada sistem tersebut.

Perancangan sistem merupakan kegiatan merancang dan mendesain suatu sistem yang baik yang dimana kegiatan tersebut adalah langkah-langkah operasi dalam proses pengolahan data dan prosedur-prosedur untuk mendukung operasi sistem tersebut. Tujuan dari perancangan sistem ialah untuk memenuhi kebutuhan para pemakai sistem serta memberikan gambaran yang jelas dan rancang bangun yang lengkap kepada programmer dan ahli-ahli yang terlibat didalamnya.

Pada bagian ini, dibahas tentang analisis prosedur yang digambarkan dalam bentuk flowmap BPMN, pengkodean, analisis sistem fungsional, dan analisis sistem non fungsional yang meliputi perangkat keras dan perangkat lunak yang digunakan. Selain itu pada bagian ini juga akan dibahas mengenai analisis user yang terlibat dalam aplikasi tersebut. Tahapan ini sangat penting dalam membantu melanjutkan tahapan yang selanjutnya yaitu tahapan perancangan.

### Analisis Sistem Yang Sedang Berjalan

Pada sub bab ini akan dijelaskan mengenai proses prediksi gaji pegawai yang sedang berjalan. Analisa sistem yang sedang berjalan bertujuan untuk mengetahui dan menggambarkan lebih lanjut mengenai bagaimana cara kerja sistem tersebut. Sistem yang berjalan saat ini menjelaskan tentang flowmap proses prediksi gaji pegawai. Flowmap tersebut merupakan gambaran alur proses prediksi gaji pegawai yang sedang berjalan pada aplikasi yang nantinya akan dibangun.



### Analisis Sistem Yang Akan Dibangun

Pada sub bab ini akan dijelaskan mengenai aplikasi website prediksi gaji pegawai berdasarkan faktor-faktor spesifik akan dibangun. Analisa sistem yang sedang berjalan memiliki tujuan untuk memberikan gambaran dan mengetahui lebih lanjut bagaimana cara kerja sistem tersebut. Strategi yang digunakan dalam menganalisis sistem yang akan dibangun ini, adalah dengan membongkar atau menterjemahkan dalam bentuk flowmap BPMN.

1. Flowmap Login

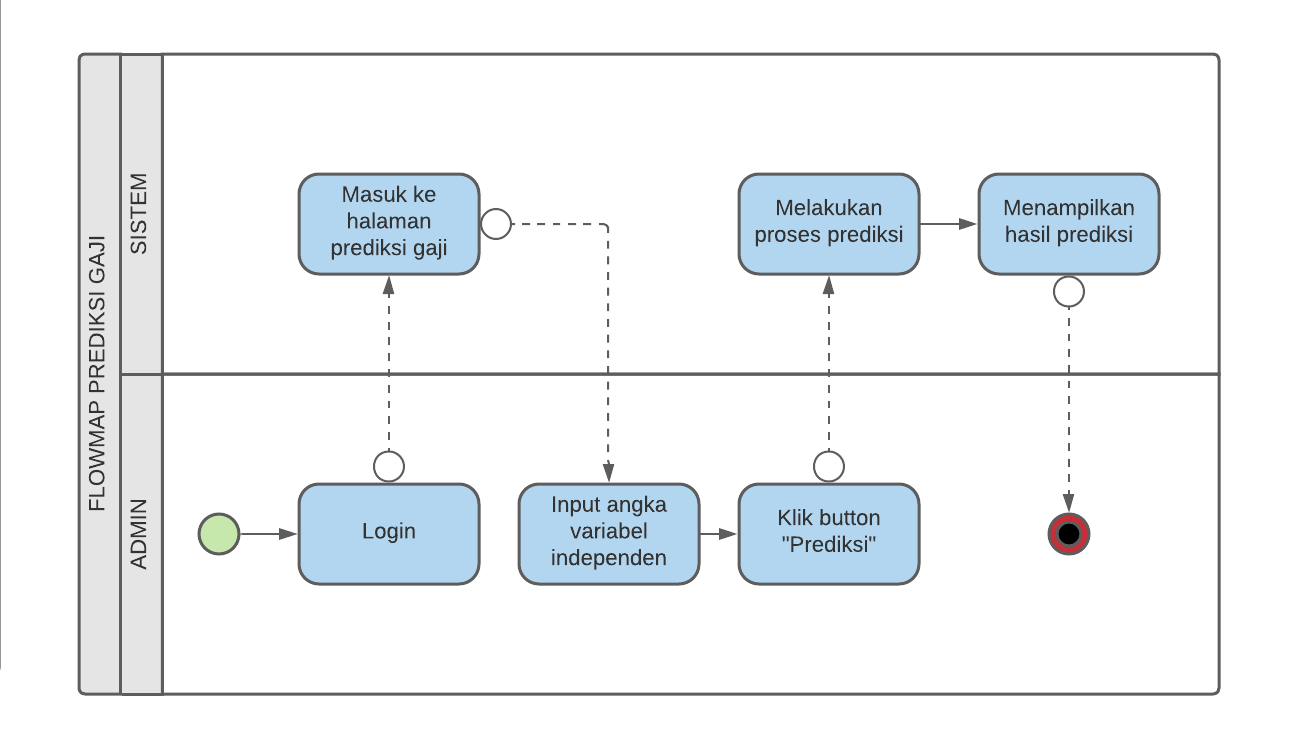
Pada bagian ini menjelaskan tentang flowmap proses login admin untuk masuk ke aplikasi website prediksi gaji. Flowmap tersebut merupakan gambaran alur proses login admin yang akan dibangun pada aplikasi ini.



Keterangan :

1. Admin dapat membuka Aplikasi Prediksi Gaji Pegawai.
2. Aplikasi menampilkan halaman login .
3. Admin dapat melakukan penginputan username serta password.
4. Setelah itu, sistem dapat melakukan pengecekan apakah username serta password yang diinputkan admin valid atau tidak.
5. Jika username dan password sesuai, maka admin dapat masuk ke halaman .utama aplikasi. Sedangkan jika username dan password tidak sesuai, maka admin akan tetap berada di halaman login.
6. Flowmap Prediksi Gaji

Pada bagian ini menjelaskan tentang flowmap proses prediksi gaji pegawai yang dilakukan oleh admin. Flowmap tersebut merupakan gambaran alur proses prediksi gaji pegawai oleh admin yang akan dibangun pada aplikasi ini.



Keterangan :

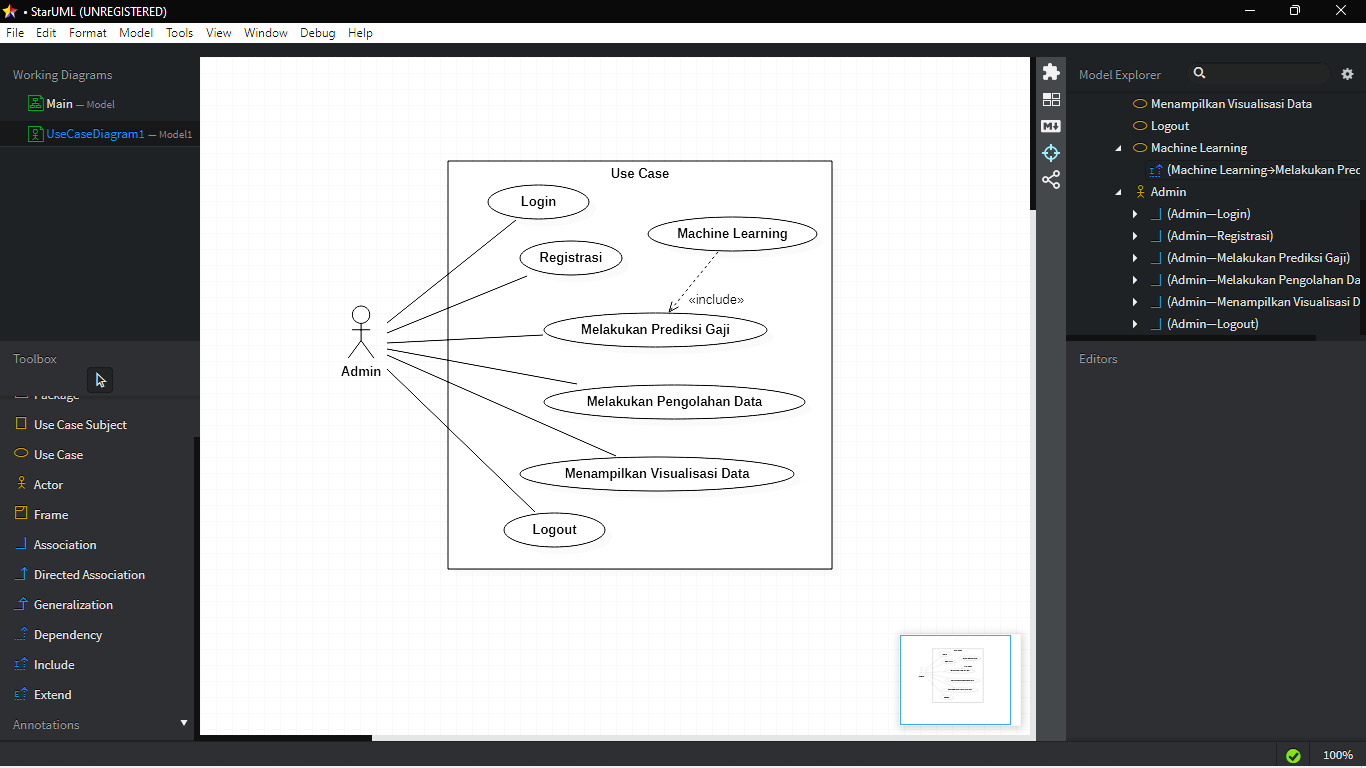
1. Admin dapat melakukan login.
2. Setelah admin berhasil login, aplikasi akan menampilkan halaman dashboard.
3. Setelah itu, admin dapat memilih menu Prediksi untuk beralih ke halaman prediksi.
4. Pada halaman prediksi, admin dapat menginputkan angka berupa variabel independent (usia, level pekerjaan, total tahun bekerja, dan tahun di perusahaan) pada form yang disediakan.
5. Setelah diinputkan, aplikasi akan menampilkan hasil prediksi gaji pegawai.

## Perancangan Sistem (UML)

UML adalah singkatan dari Unified Modeling Language yang didefinisikan sebagai sekumpulan alat yang digunakan untuk melakukan abstraksi terhadap sebuah sistem atau software berbasis objek. UML juga dapat dikatakan sebagai bahasa spesifikasi standar yang digunakan untuk mendokumentasikan, menspesifikasikan, dan membangun suatu sistem perangkat lunak.

UML adalah suatu bentuk metodologi untuk mengembangkan sistem OOP dan sekelompok perangkat tools untuk mendukung pengembangan sistem tersebut. Sehingga, UML juga dapat menjadi salah satu cara untuk mempermudah dalam melakukan pengembangan aplikasi yang berkelanjutan. Oleh karenanya, UML juga dapat menjadi suatu alat bantu untuk transfer ilmu tentang sistem yang akan dikembangkan dari satu developer ke developer lainya.

### Use case Diagram



Use case diagram yaitu suatu gambaran graphical untuk memodelkan seluruh proses bisnis berdasarkan perspektif pengguna sistem dari beberapa atau semua aktor, use case, dan interaksi yang memperkenalkan suatu sistem. Use case diagram secara sederhana merupakan sebuah sarana bantu untuk melakukan pendefinisian apa yang ada di luar sistem (aktor) dan apa yang harus dilakukan oleh sistem yang sedang dikembangkan.

1. Definisi Aktor

Pada bagian ini akan dijelaskan aktor-aktor yang terlibat dalam Sistem.

Tabel 3. 1 Definisi Aktor

|  |  |  |
| --- | --- | --- |
| No | Aktor | Deskripsi |
| 1 | Admin | * Login * Melakukan Prediksi Gaji * Melakukan Pengolahan Data * Melakukan Visualisasi Data * Logout |

1. Definisi *Use case*

Tabel 3. 2 Definisi Use case

|  |  |  |
| --- | --- | --- |
| No | Aktor | Deskripsi |
| 1 | Machine learning | Merupakan aktivitas interaksi antara manusia dengan mesin. Dalam hal ini, admin membuat model prediksi Machine learning. |
| 2 | Login | Merupakan aktivitas login yang dilakukan oleh admin. Sebelum admin masuk ke aplikasi ia harus melakukan login terlebih dahulu. |
| 3 | Registrasi | Merupakan aktivias registrasi yang dilakukan oleh admin. Apabila admin belum memiliki akun, maka admin harus melakukan registrasi terlebih dahulu. |
| 4 | Melakukan Prediksi Gaji | Merupakan aktivitas memprediksi gaji karyawan yang dilakukan oleh admin dengan parameter lama bekerja seorang karyawan. |
| 5 | Melakukan Pengolahan Data | Merupakan aktivitas insert, read, update dan delete data karyawan yang dilakukan oleh admin. |
| 6 | Menampilkan Visualisasi Data | Merupakan aktivitas visualisasi grafik data karyawan yang dilakukan oleh admin. |
| 7 | Logout | Merupakan aktivitas logout yang dilakukan oleh admin. Admin dapat keluar dari aplikasi jika telah selesai melakukan pekerjaan pada aplikasi. |

1. Skenario *Use case*

Skenario use case diharapkan setelah berjalannya fungsional use case. Selain itu juga diberikan ulasan yang berkaitan dengan tanggapan dari sistem atas suatu aksi yang dilakukan oleh aktor. Setiap use case akan diberikan sebuah scenario yang akan menjelaskan secara detail interaksi yang ada di dalamnya.

Tabel 3. 3 Skenario Use case Machine learning

|  |  |
| --- | --- |
| Identifikasi | |
| Nomor | 1 |
| Nama | Machine learning |
| Tujuan | Pembuatan model prediksi |
| Deskripsi | |
| Aktor | Admin |
| Skenario Utama | |
| Kondisi Awal |  |
| Aksi Aktor | Reaksi Sistem |
| Membuat model prediksi | Melakukan olah data gaji karyawan |
| Kondisi Akhir | Masuk pada aplikasi |

Tabel 3. 4 Skenario Use case Login Admin

|  |  |
| --- | --- |
| Identifikasi | |
| Nomor | 1 |
| Nama | Login |
| Tujuan | Sebelum masuk pada aplikasi |
| Deskripsi | |
| Aktor | Admin |
| Skenario Utama | |
| Kondisi Awal |  |
| Aksi Aktor | Reaksi Sistem |
| Input username dan password | Mengecek data dan menentukan aktor tersebut admin atau user |
| Kondisi Akhir | Masuk pada aplikasi |

Tabel 3. 5 Skenario Use case Registrasi

|  |  |
| --- | --- |
| Identifikasi | |
| Nomor | 1 |
| Nama | Registrasi |
| Tujuan | Membuat akun admin |
| Deskripsi | |
| Aktor | Admin |
| Skenario Utama | |
| Kondisi Awal |  |
| Aksi Aktor | Reaksi Sistem |
| Input username, dan password | Menyimpan data admin atau user |
| Kondisi Akhir | Masuk pada halaman login |

Tabel 3. 6 Skenario Use case Melakukan Prediksi Gaji Karyawan

|  |  |
| --- | --- |
| Identifikasi | |
| Nomor | 1 |
| Nama | Melakukan Predikis Gaji Karyawan |
| Tujuan | Melakukan Predikis Gaji Karyawan |
| Deskripsi | |
| Aktor | Admin |
| Skenario Utama | |
| Kondisi Awal |  |
| Aksi Aktor | Reaksi Sistem |
| Menginput Data | Melakukan Prediksi Gaji |
| Kondisi Akhir | Admi dapat memprediksi gaji |

Tabel 3. 7 Skenario Use case Melakukan Visualisasi Data

|  |  |
| --- | --- |
| Identifikasi | |
| Nomor | 1 |
| Nama | Melakukan Visualisasi Data |
| Tujuan | Melakukan visualisasi data karyawan |
| Deskripsi | |
| Aktor | Admin |
| Skenario Utama | |
| Kondisi Awal |  |
| Aksi Aktor | Reaksi Sistem |
| Input data karyawan | Menampilkan grafik visualisasi data karyawan |
| Kondisi Akhir | Admin dapat memvisualisasikan data |

Tabel 3. 8 Skenario Use case Melakukan Pengolahan Data

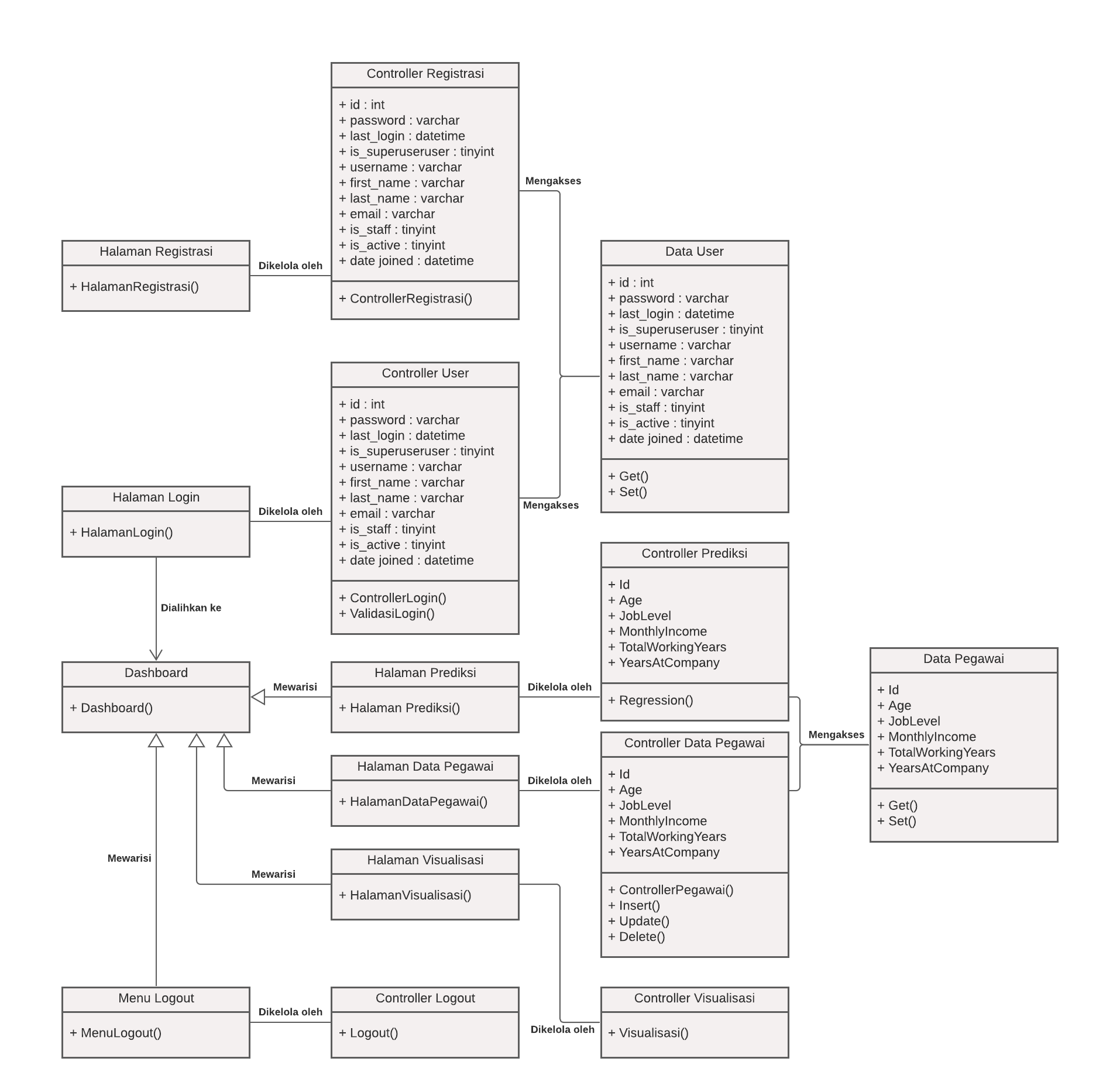
|  |  |
| --- | --- |
| Identifikasi | |
| Nomor | 1 |
| Nama | Melakukan Pengolahan Data |
| Tujuan | Melakukan pengolahan data karyawan |
| Deskripsi | |
| Aktor | Admin |
| Skenario Utama | |
| Kondisi Awal |  |
| Aksi Aktor | Reaksi Sistem |
| Insert, read, update, dan delete data karyawan. | Mengubah data karyawan yang diubah oleh admin |
| Kondisi Akhir | Admin dapat mengelola data |

Tabel 3. 9 Skenario Use case Logout Admin

|  |  |
| --- | --- |
| Identifikasi | |
| Nomor | 1 |
| Nama | Logout |
| Tujuan | Keluar dari aplikasi |
| Deskripsi | |
| Aktor | Admin |
| Skenario Utama | |
| Kondisi Awal |  |
| Aksi Aktor | Reaksi Sistem |
| Menekan tombol logout | Melakukan proses keluar dari aplikasi |
| Kondisi Akhir | Keluar dari aplikasi |

### Class diagram

Class diagram ialah diagram UML yang mendeskripsikan suatu struktur dari sebuah sistem yang dibuat dari kelas-kelas dengan relasi - relasinya. Class diagram juga dapat menggambarkan jenis-jenis objek yang terdapat pada sistem dan berbagai hubungan statis yang terdapat di antara mereka. Class diagram ini dapat menunjukkan properti dan operasi sebuah kelas serta batasan-batasan yang terdapat dalam hubungan-hubungan objek tersebut.



Gambar 3. 5 Class diagram

### Sequence Diagram

Sequence Diagram adalah diagram yang mendeskripsikan serta menjelaskan suatu interaksi objek yang berdasarkan urutan waktu. Interaksi tersebut diawali dari apa yang memicu aktivitas tersebut, proses dan perubahan apa saja yang terjadi secara internal dan output apa yang dihasilkan. Sequence Diagram dapat digunakan untuk menggambarkan urutan atau prosedur yang harus dilakukan untuk dapat menghasilkan sesuatu seperti pada use case diagram.

1. Sequence Diagram Login Admin

Berikut ini merupakan Sequence Diagram Login Admin menjelaskan hubungan antara admin pada aplikasi.

Berikut Sequence Diagram Login Admin pada gambar.



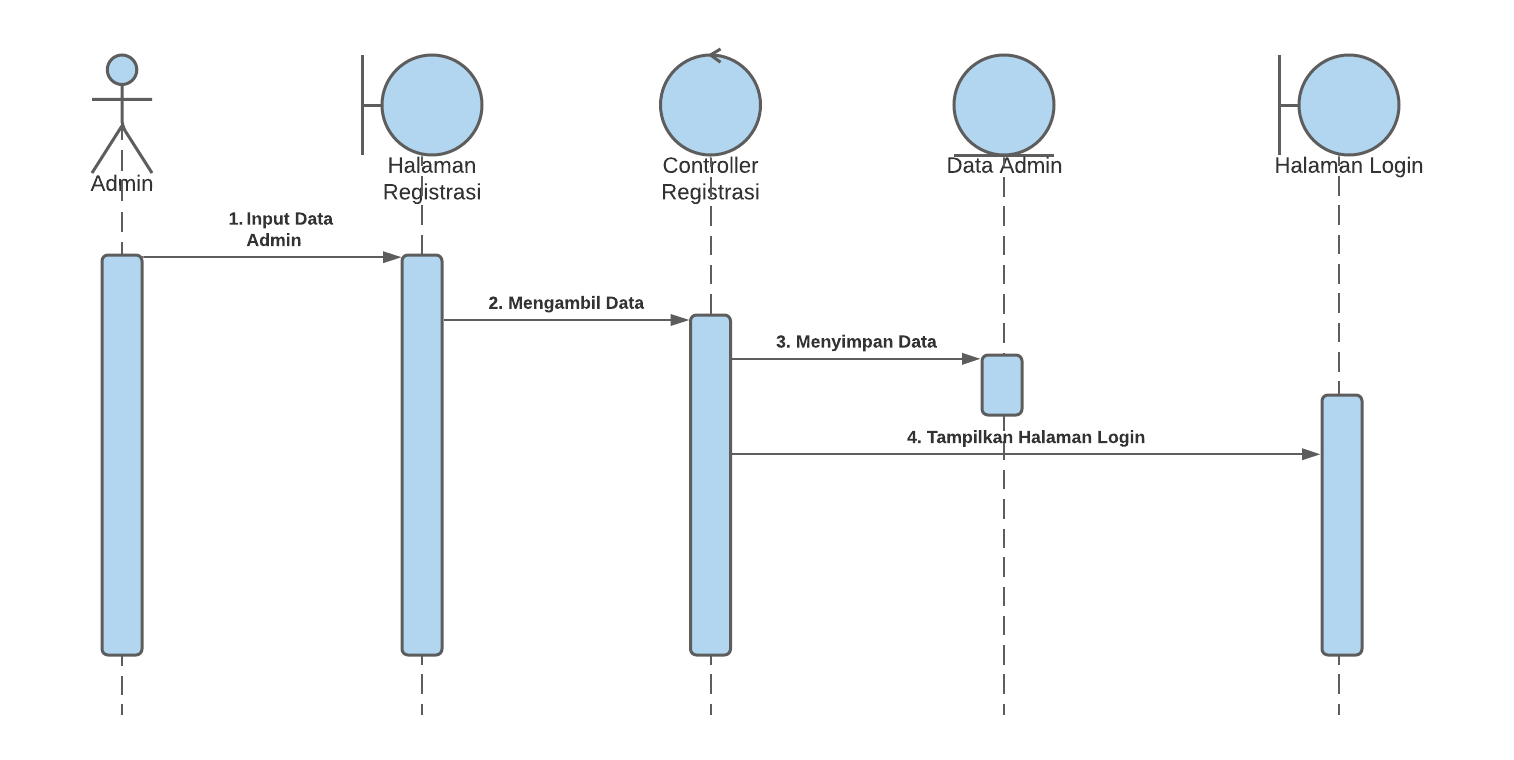
Gambar 3. 6 Sequence Diagram Login Admin

Keterangan :

1. Admin menginputkan username dan password di halaman login.
2. Validasi username dan password diteruskan ke Controller login.
3. Controller login mengambil data ke database data admin untuk validasi login.
4. Jika username dan password valid pada data admin maka akan ditampilkan halaman dashboard.
5. Jika username dan password tidak valid pada data admin maka akan ditampilkan halaman login.
6. Sequence Diagram Melakukan Registrasi

Berikut ini merupakan Sequence Diagram Registrasi Admin menjelaskan hubungan antara admin pada aplikasi.

Berikut Sequence Diagram Melakukan Registrasi pada gambar.

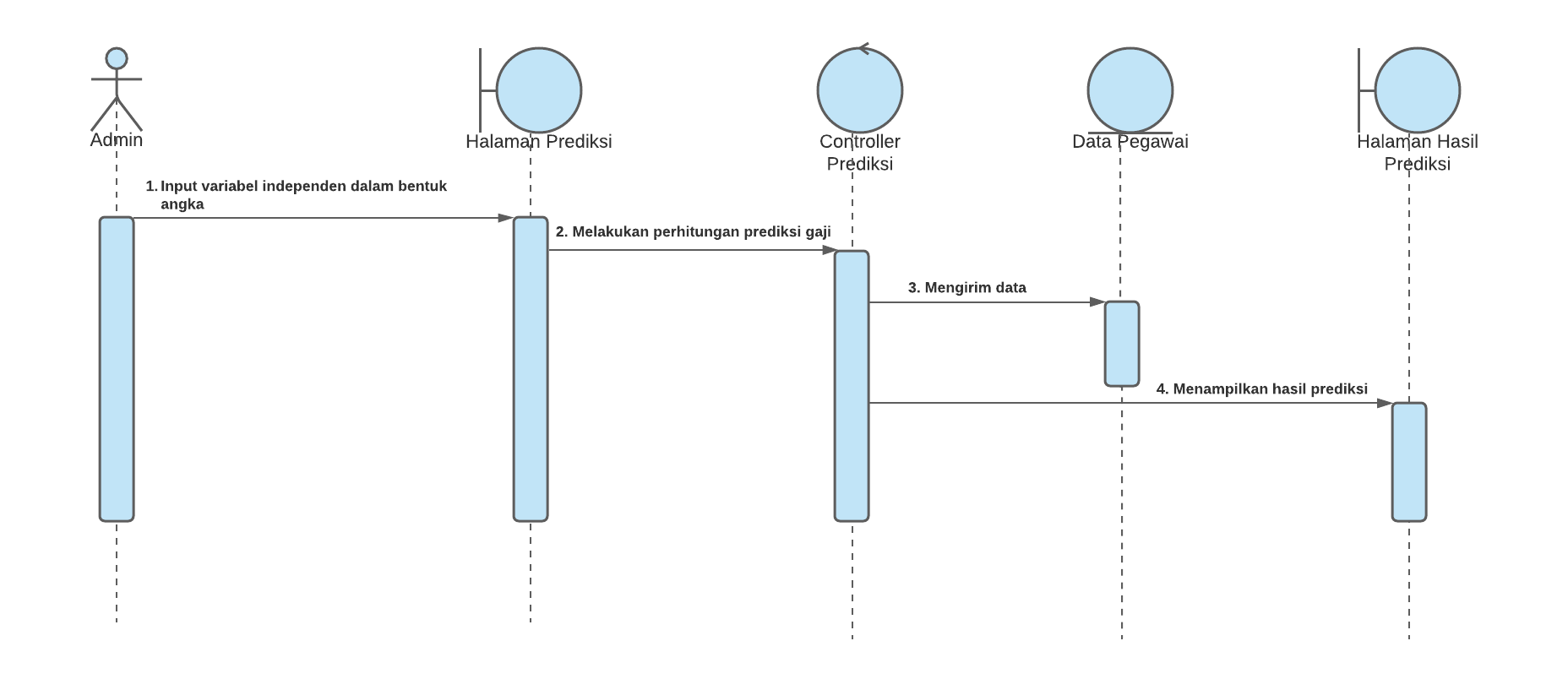


Keterangan :

1. Admin menginputkan data admin di halaman registrasi.
2. Mengambil data oleh Controller registrasi.
3. Controller login mengirimkan data ke data admin kemudian data tersebut disimpan.
4. Jika data admin telah tersimpan pada data admin maka akan ditampilkan halaman login.
5. Sequence Diagram Melakukan Prediksi Gaji

Berikut ini merupakan Sequence Diagram melakukan predikis gaji menjelaskan hubungan antara admin dan aplikasi. Admin dapat melakukan prediksi gaji dengan cara menginput data.

Berikut Sequence Diagram Melakukan Prediksi Gaji pada gambar.



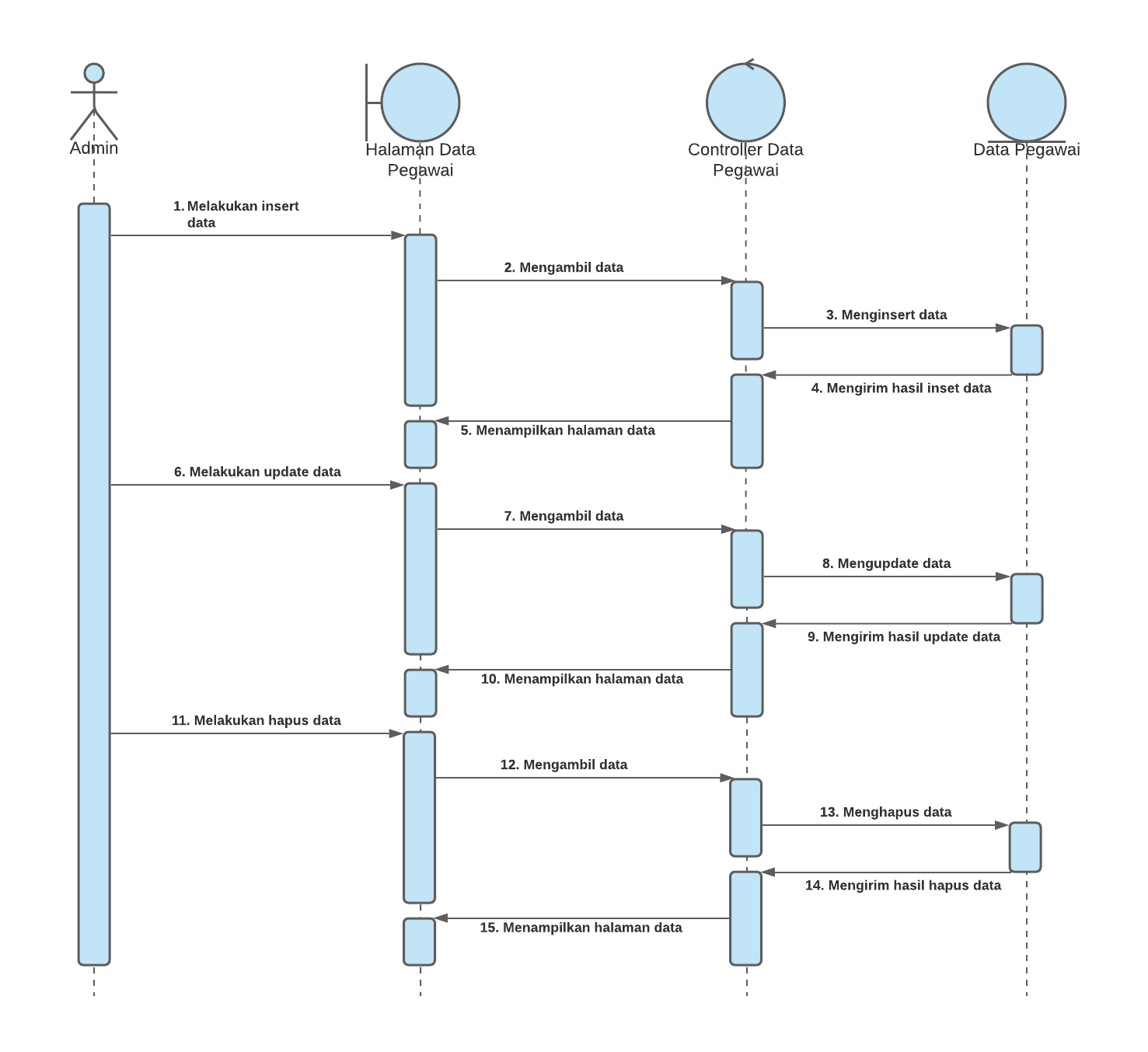
Gambar 3. 8 Sequence Diagram Melakukan Prediksi Gaji

Keterangan :

1. Admin menginputkan independen dalam bentuk angka.
2. Sistem melakukan perhitungan prediksi gaji.
3. Controller prediksi mengirim data ke database data pegawai dan memproses data.
4. Hasil data yang dihitung akan ditampilkan pada halaman hasil prediksi.
5. Sequence Diagram Melakukan Pengolahan Data

Berikut ini merupakan Sequence Diagram mengelola data menjelaskan hubungan antara admin pada aplikasi. Admin dapat insert, read, update, dan delete pada data alat.

Berikut Sequence Diagram Melakukan Pengolahan Data pada gambar.



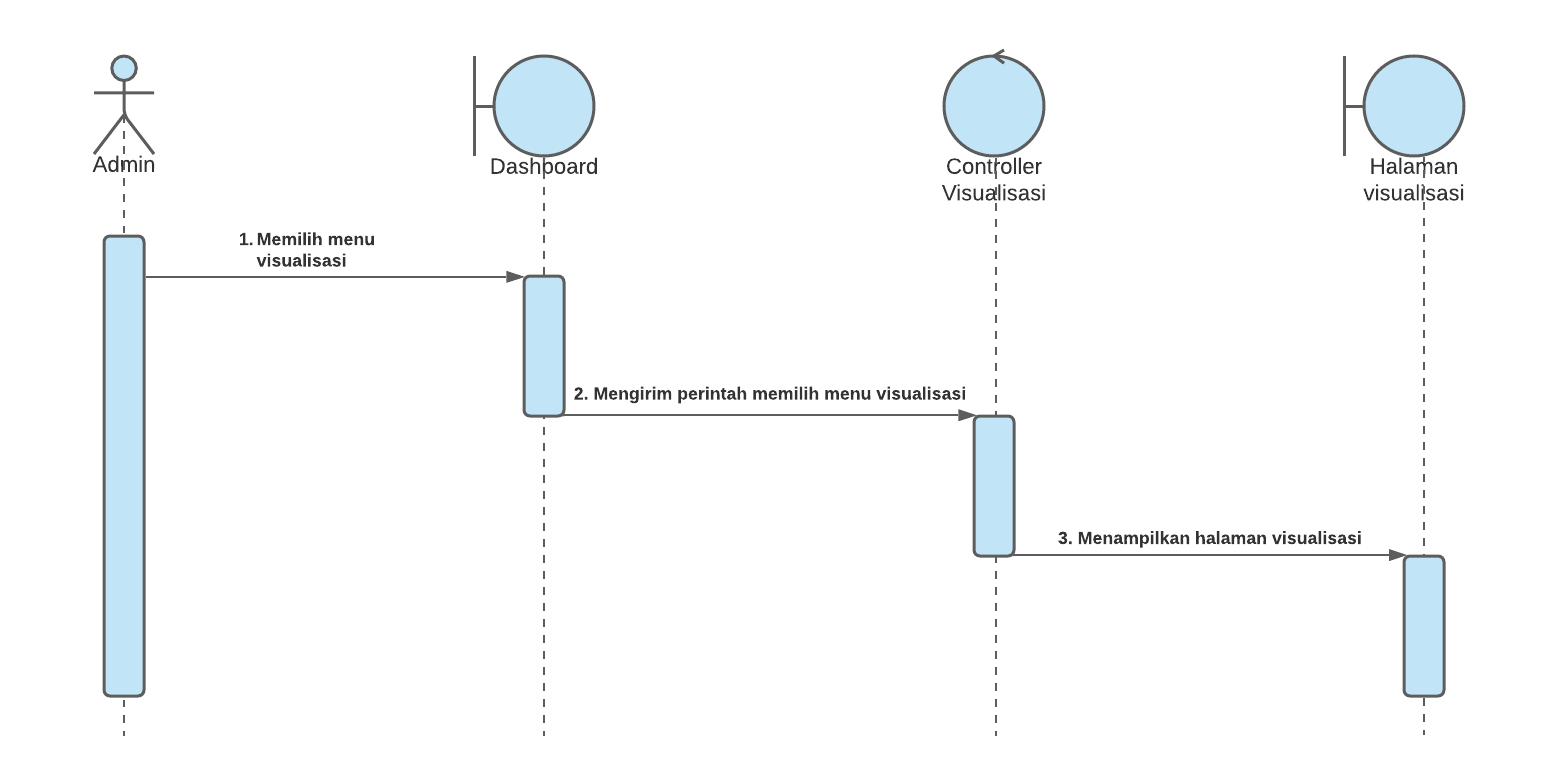
Gambar 3. 9 Sequence Diagram Melakukan Pengolahan Data

Keterangan :

1. Admin memilih menu yang akan dikelola pada menu di halaman data pegawai. Menu yang ada yaitu tambah data, update data, dan hapus data.
2. Halaman data pegawai mengambil data (baik antara input, update atau hapus) untuk diteruskan ke controller data pegawai.
3. Controller data pegawai mengirim hasil olah data ke data pegawai.
4. Data pegawai mengirim hasil olah data ke controller data pegawai.
5. Controller data pegawai menampilkan halaman data pegawai.
6. Sequence Diagram Melakukan Visualisasi Data

Berikut ini merupakan Sequence Diagram melakukan visualisasi data menjelaskan hubungan antara admin pada aplikasi.

Berikut Sequence Diagram Melakukan Visualisasi Data pada gambar.



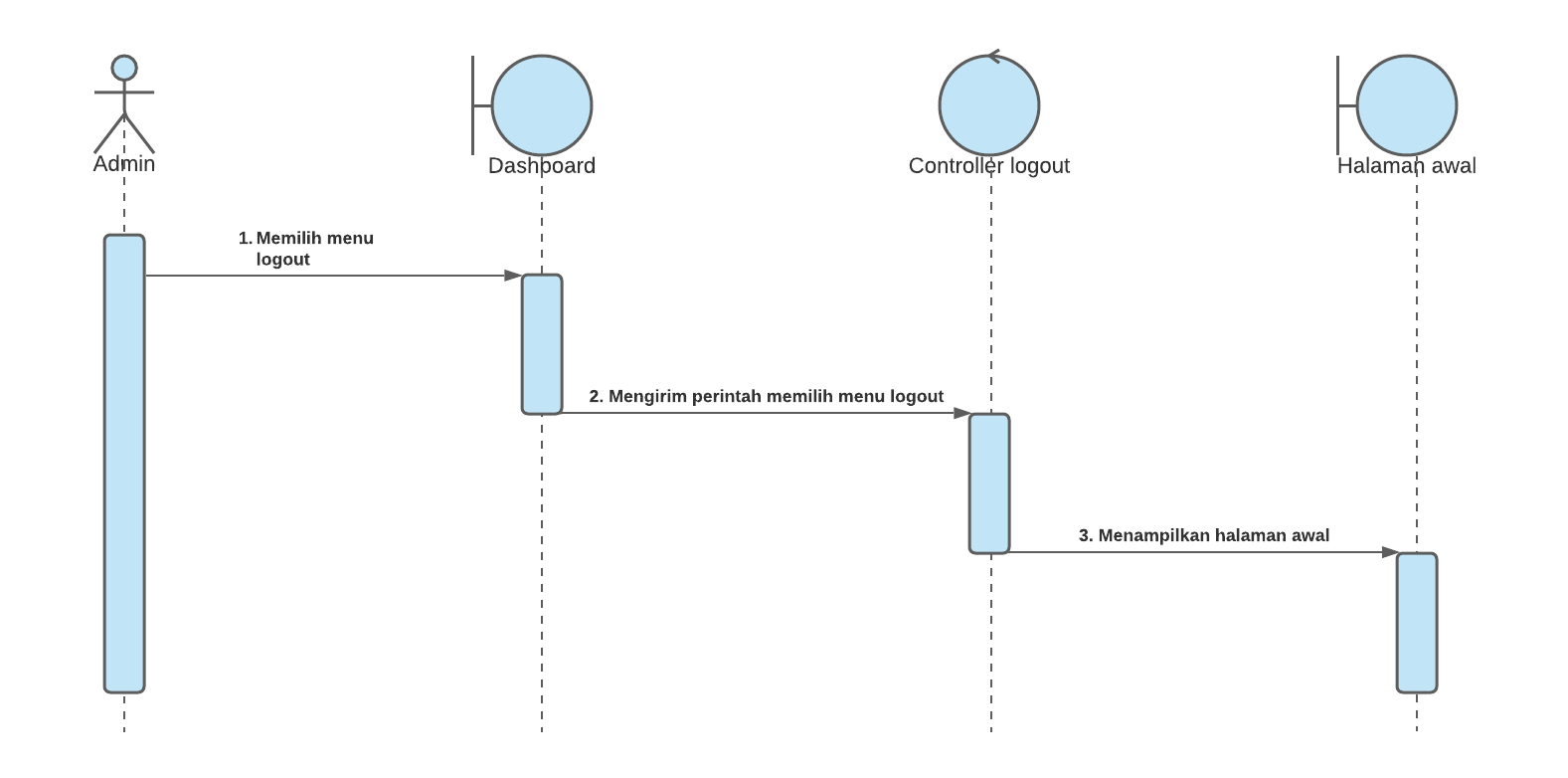
Gambar 3. 10 Sequence Diagram Melakukan Visualisasi Data

Keterangan :

1. Admin memilih menu halaman visualisasi pada dashboard.
2. Dashboard mengirim perintah memilih menu visualisasi ke controller visualisasi.
3. Controller visualisasi menampilkan halaman visualisasi.
4. Sequence Diagram Melakukan Logout

Berikut ini merupakan Sequence Diagram logout Admin menjelaskan hubungan antara admin pada aplikasi.

Berikut Sequence Diagram logout Admin pada gambar.



Gambar 3. 11 Sequence Diagram logout admin

Keterangan :

1. Admin memilih menu logout pada dashboard.
2. Dashboard mengirim perintah memilih menu logout ke controller logout.
3. Controller logout melakukan proses logout kemudian menuju untuk menampilkan halaman awal.

### Collaboration Diagram

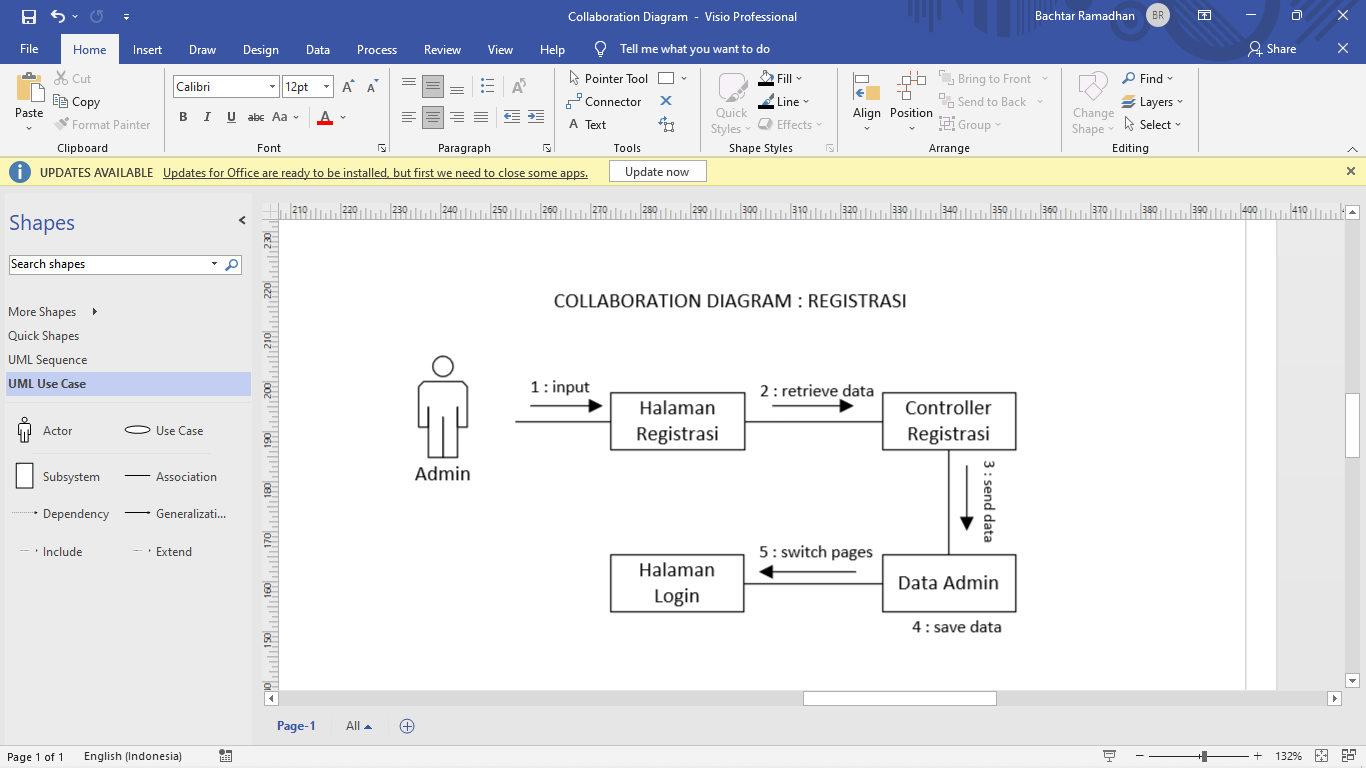
Collaboration Diagram dapat dipakai untuk memodelkan interaksi antar objek yang ada di dalam sistem. Berbeda dari Sequence Diagram yang lebih menunjukkan kronologis dari operasi-operasi yang dilakukan, Collaboration Diagram ini lebih fokus pada pemahaman atas keseluruhan operasi yang dilakukan oleh objek.

1. Collaboration Diagram Melakukan Login



Gambar 3. 12 Collaboration Diagram Melakukan Login

1. Collaboration Diagram Melakukan Registrasi



Gambar 3. 13 Collaboration Diagram Registrasi

1. Collaboration Diagram Melakukan Prediksi Gaji



Gambar 3. 14 Collaboration Diagram Melakukan Prediksi Gaji

1. Collaboration Diagram Melakukan Pengolahan Data



Gambar 3. 15 Collaboration Diagram Melakukan Visualisasi Data

1. Collaboration Diagram Melakukan Visualisasi Data

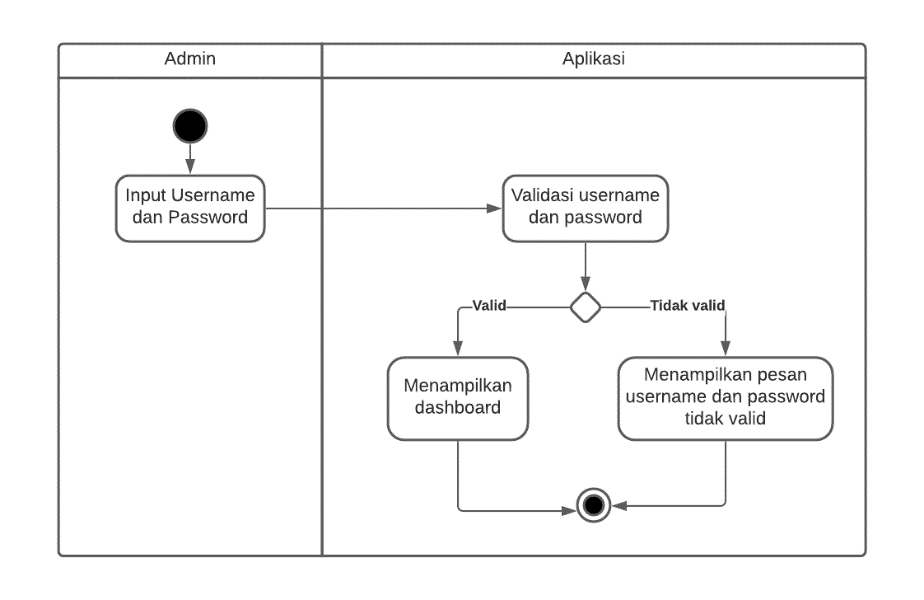


Gambar 3. 16 Collaboration Diagram Melakukan Visualisasi Data

### Activity Diagram

Activity Diagram dapat didefinisikan sebagai diagram yang menggambarkan tentang berbagai aktifitas yang terjadi pada suatu sistem. Activity Diagram adalah suatu teknik yang digunakan untuk mendeskripsikan logika procedural, proses bisnis serta aliran kerja dalam banyak kasus. Activity Diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, hasil akhir yang mungkin terjadi, hingga bagaimana mereka berakhir.

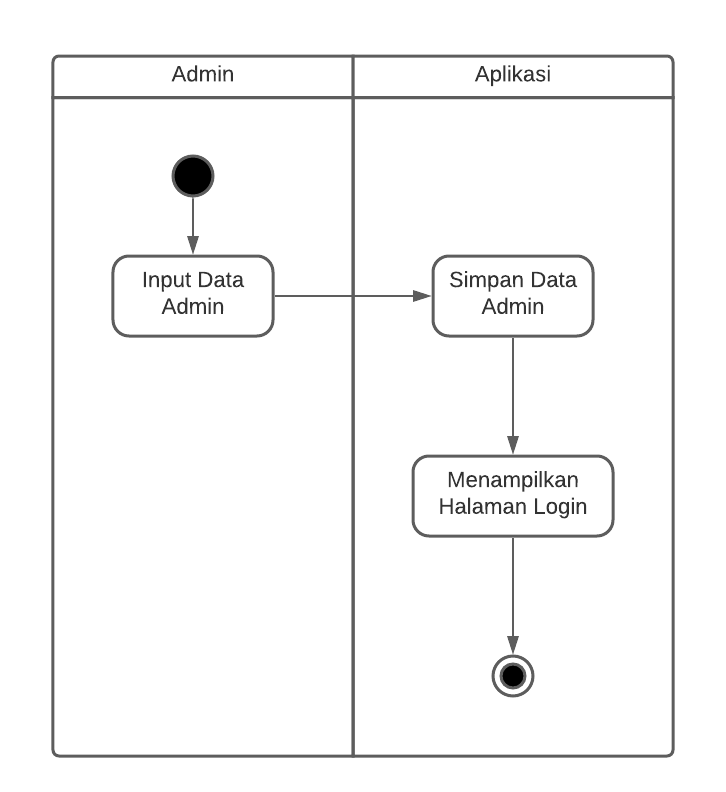
1. Activity Diagram Login Admin



Gambar 3. 17 Activity Diagram Login Admin

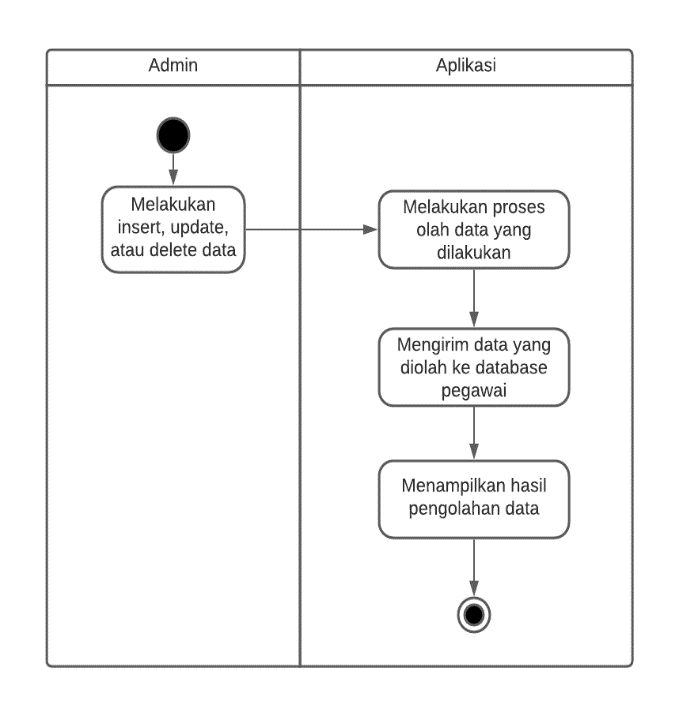
Keterangan :

1. Admin menginput username dan password.
2. Aplikasi melakukan validasi username dan password.
3. Jika valid akan menampilkan halaman dashboard.
4. Jika tidak valid menampilkan pesan username dan password tidak valid.
5. Activity Diagram Melakukan Registrasi Admin



Keterangan :

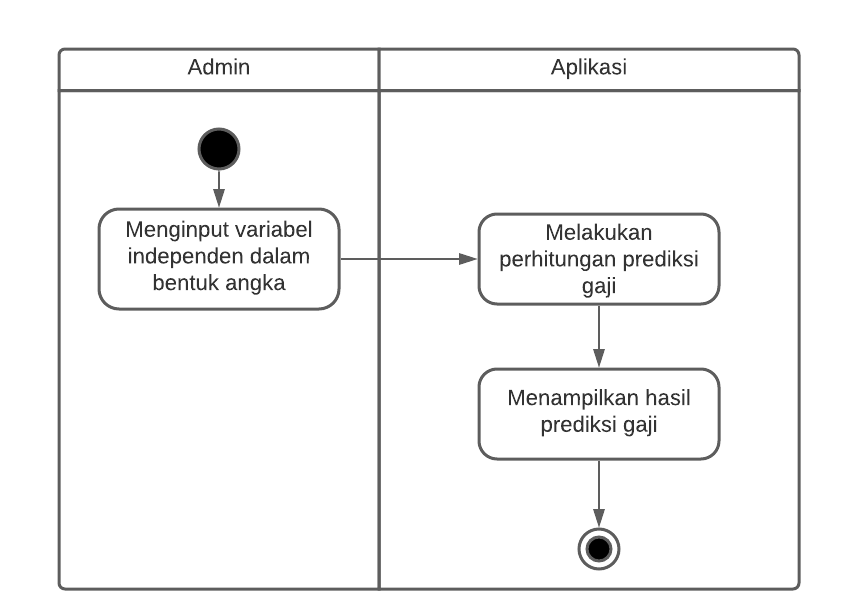
1. Admin menginput data admin.
2. Aplikasi menyimpan data admin.
3. Aplikasi menampilkan halaman login.
4. Activity Diagram Melakukan Pengolahan Data



Gambar 3. 19 Activity Diagram Melakukan Pengolahan Data

Keterangan :

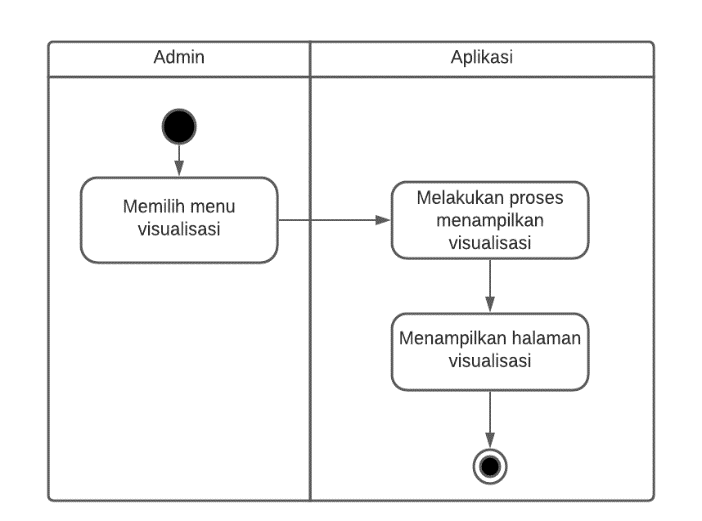
1. Admin melakukan olah data seperti insert, update, dan delete data.
2. Aplikasi melakukan proses olah data yang dilakukan.
3. Aplikasi mengirim data yang diolah ke database pegawai.
4. Aplikasi menampilkan data hasil create/update/delete.
5. Activity Diagram Melakukan Prediksi Gaji



Gambar 3. 20 Activity Diagram Melakukan Prediksi Gaji

Keterangan :

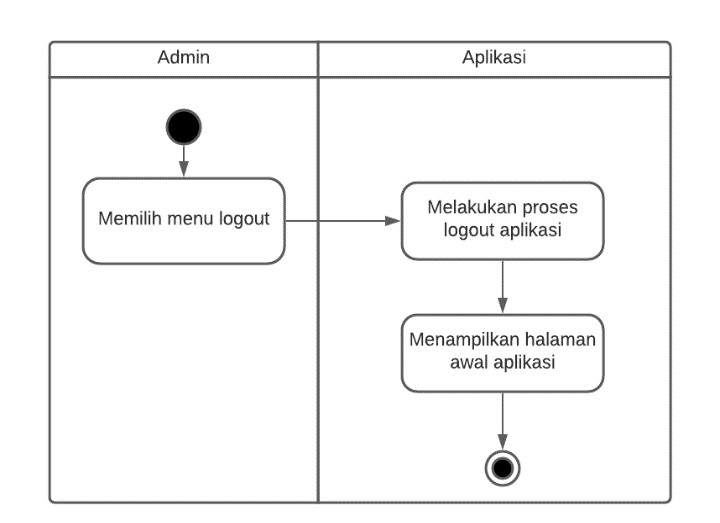
1. Admin menginputkan variabel independent dalam bentuk angka pada form prediksi.
2. Aplikasi melakukan proses perhitungan prediksi gaji.
3. Aplikasi menampilkan hasil prediksi.
4. Activity Diagram Melakukan Visualisasi



Gambar 3. 21 Activity Diagram Melakukan Visualisasi

Keterangan :

1. Admin memilih menu visualisasi.
2. Aplikasi melakukan proses menampilkan visualisasi.
3. Aplikasi menampilkan halamann visualisasi.
4. Activity Diagram Logout Admin



Gambar 3. 22 Activity Diagram Logout Admin

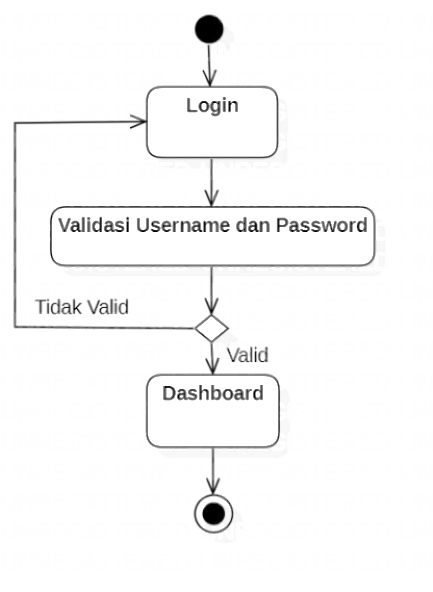
Keterangan :

* + - 1. Admin memilih menu logout.
      2. Aplikasi melakukan proses logout aplikasi.
      3. Aplikasi menampilkan halaman awal aplikasi.

### Statechart Diagram

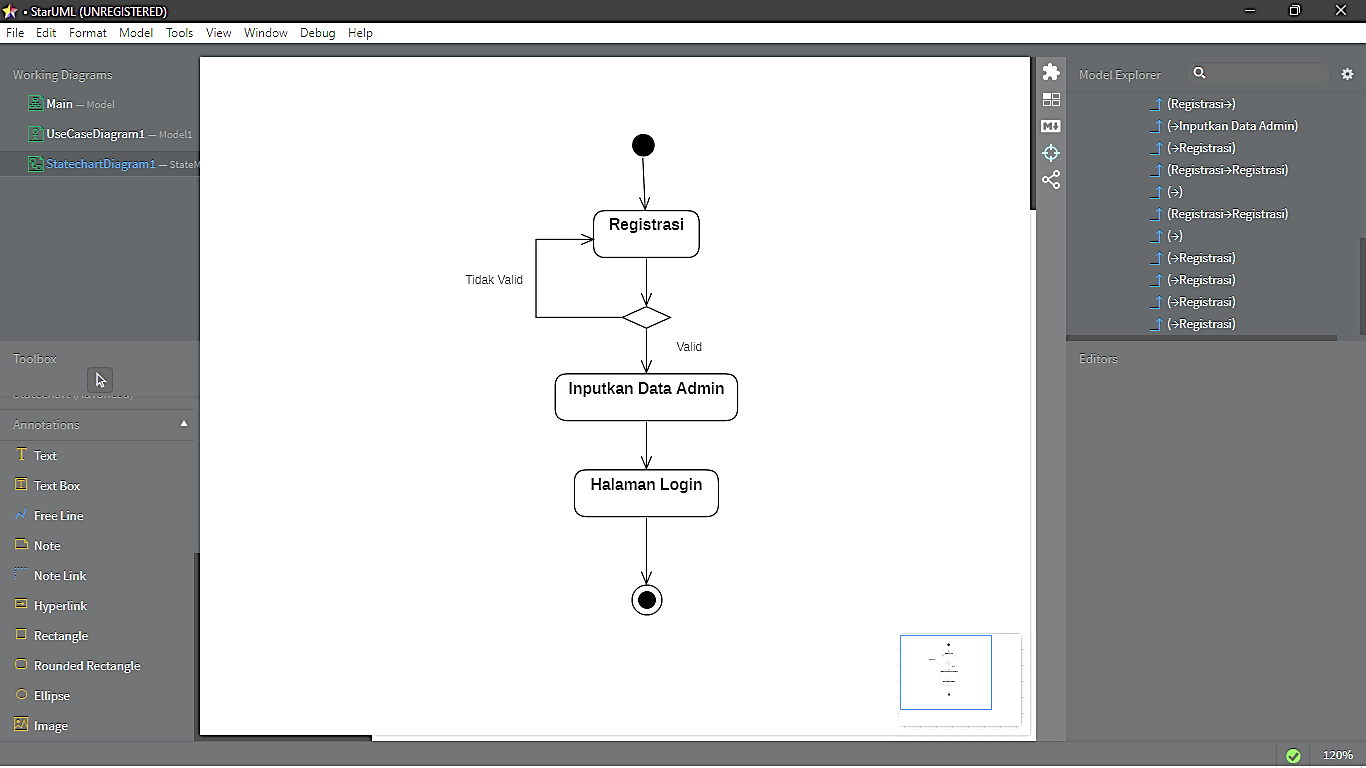
Statechart Diagram dapat menggambarkan transisi dan perubahan keadaan (dari satu state ke state lainnya) dari suatu objek pada sistem sebagai akibat dari stimuli yang diterimanya. Pada umumnya Statechart Diagram dapat menjelaskan atau menggambarkan class tertentu (satu class dapat memiliki lebih dari satu Statechart Diagram).

1. Statechart Diagram Login Admin



Gambar 3. 23 Statechart Diagram Login Admin

1. Statechart Diagram Registrasi Admin



1. Statechart Diagram Melakukan Prediksi



Gambar 3. 25 Statechart Diagram Melakukan Prediksi

1. Statechart Diagram Pengolahan Data



Gambar 3. 26 Statechart Diagram Pengolahan Data

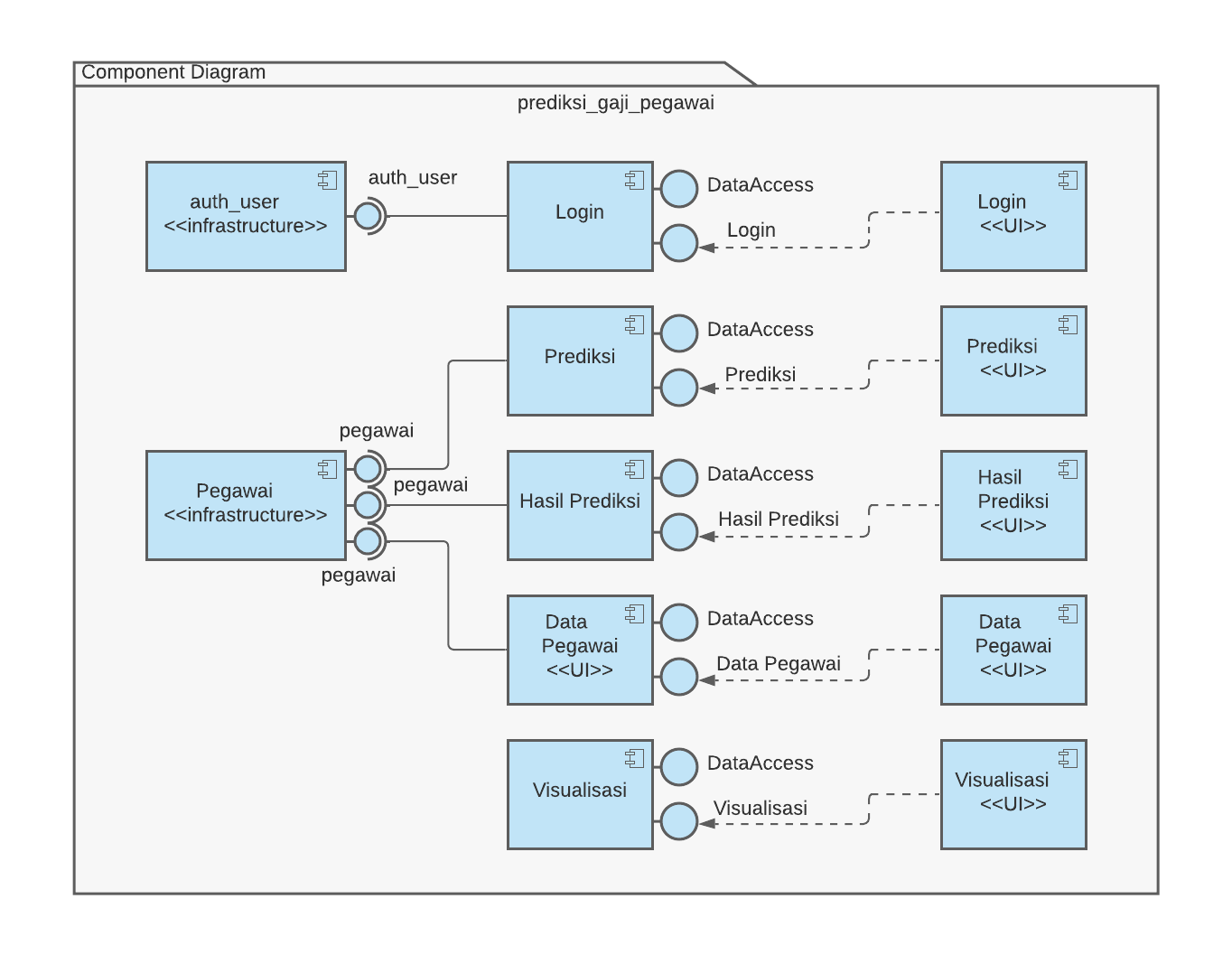
1. Statechart Diagram Visualisasi



Gambar 3. 27 Statechart Diagram Visualisasi

### Component Diagram

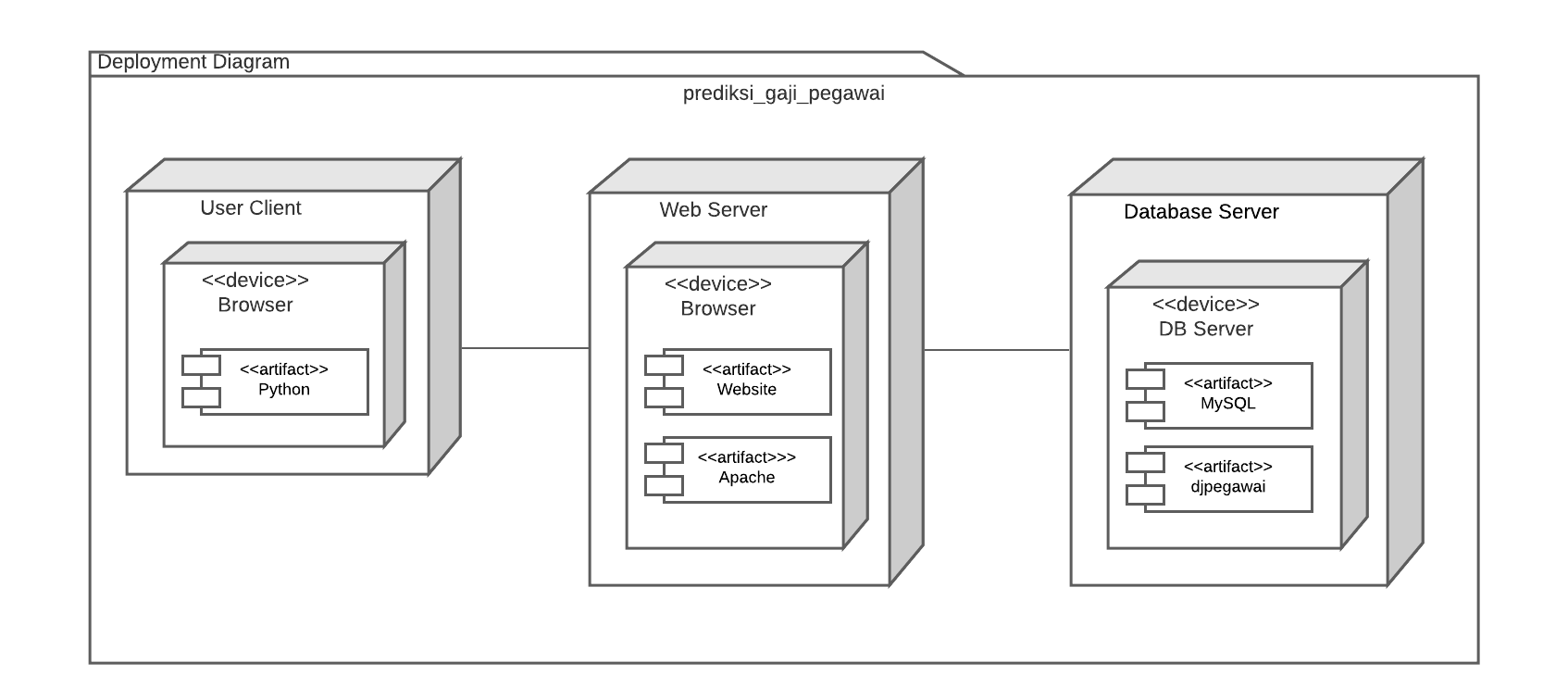
Component diagram dapat menggambarkan struktur dan hubungan antar komponen piranti lunak, termasuk ketergantungan (dependency) diantaranya. Komponen piranti lunak atau yang biasa disebut dengan perangkat lunak adalah modul berisi code, baik berisi source code maupun binary code, baik library maupun executable, baik yang muncul pada compile time, link time, maupun runtime. Pada umumnya komponen dapat terbentuk dari beberapa class dan atau package, tapi dapat juga dibentuk dari komponen-komponen yang lebih kecil.



Gambar 3. 28 Component Diagram

### Deployment Diagram

Deployment/physical diagram menggambarkan detail bagaimana suatu komponen di-deploy dalam infrastruktur sistem, dimana komponen akan terletak (pada mesin, server atau piranti keras apa), dan bagaimana kemampuan jaringan pada lokasi tersebut, spesifikasi, server, serta hal-hal lain yang bersifat fisik. Sebuah node terdiri dari server, workstation, atau piranti keras lain yang digunakan untuk melakukan deploy komponen dalam lingkungan sebenarnya. Hubungan antar node (misalnya TCP/IP) dan requirement dapat pula didefinisikan dalam diagram ini. (Dharwiyanti : 2003)



Gambar 3. 29 Deployment Diagram

## Perancangan Basis Data/ Data Sistem

Perancangan database merupakan proses untuk menentukan dan pengaturan data yang dibutuhkan untuk mendukung rancangan sistem, agar teciptanya pemrosesan data yang lebih efisien. Struktur tabel meliputi nama tabel, tipe data, nama atribut dan data relasi seperti primary dan foreign key.

### CDM (Conceptual Data Model)



### PDM (Physical Data Model)



### ERD (Entity Relationship Diagram)



## Struktur Menu

Struktur menu ialah bentuk umum dari suatu rancangan aplikasi dalam memudahkan pengguna untuk menjalankan aplikasi. Sehingga saat menjalankan aplikasi, pengguna tidak mengalami kesulitan dalam memilih menumenu yang diinginkan. Berikut adalah struktur menu dari aplikasi Prediksi Gaji Pegawai secara keseluruhan.



## Perancangan User Interface

Pencil

### User Interface Halaman Depan

### User Interface Halaman Login

### User Interface Halaman Registrasi

### User Interface Halaman Dashboard

### User Interface Halaman Prediksi

### User Interface Halaman Hasil Prediksi

### User Interface Halaman Data Pegawai

### User Interface Halaman Tambah Data Pegawai

### User Interface Halaman Edit Data Pegawai

### User Interface Halaman Visualisasi

## Halaman Antarmuka

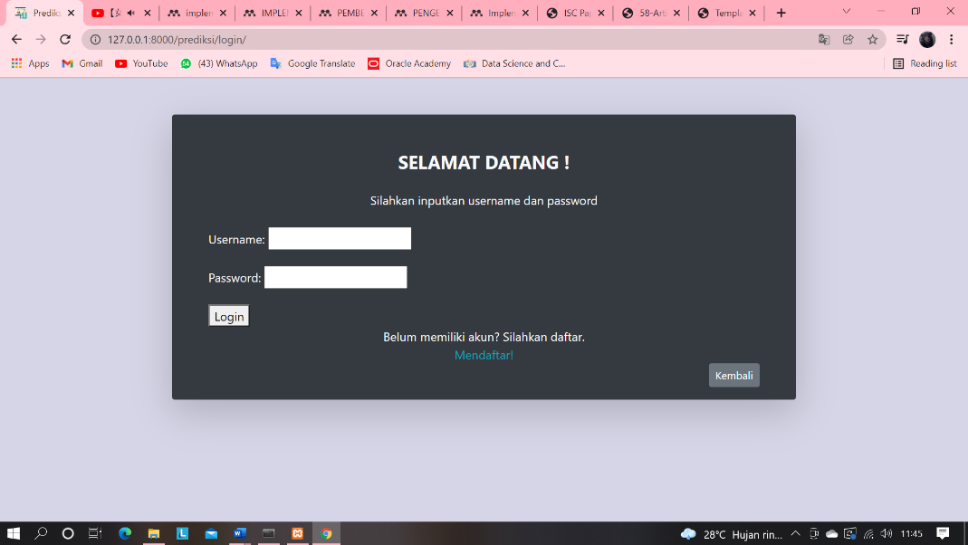
Impelementasi sistem memiliki bagian berupa pemaparan mengenai tampilan pada sistem serta kegunaan dari setiap halaman. Implementasi dari hasil perancanggan menggunakan bahasa pemrograman python dengan framework Django.

### Implementasi Halaman Depan



Gambar diatas merupakan implementasi dari halaman depan sistem. Pada halaman ini, user dapat menekan button “Login” pada navbar untuk masuk ke dalam sistem.

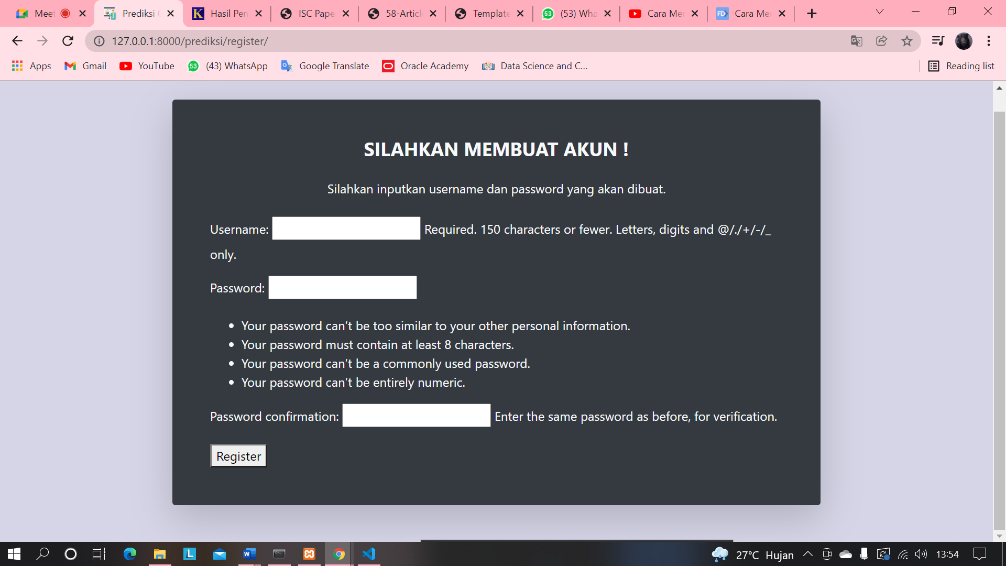
### Antarmuka Halaman Login



Gambar 3. 35 Halaman Login Sistem

Gambar diatas merupakan implementasi dari halaman login. Sebelum masuk ke halaman dashboard, user harus menginputkan username dan password yang sesuai. Jika sesuai, maka user akan di arahkan ke halaman dashboard. Jika tidak user akan diminta kembali memasukkan username dan password yang sesuai.

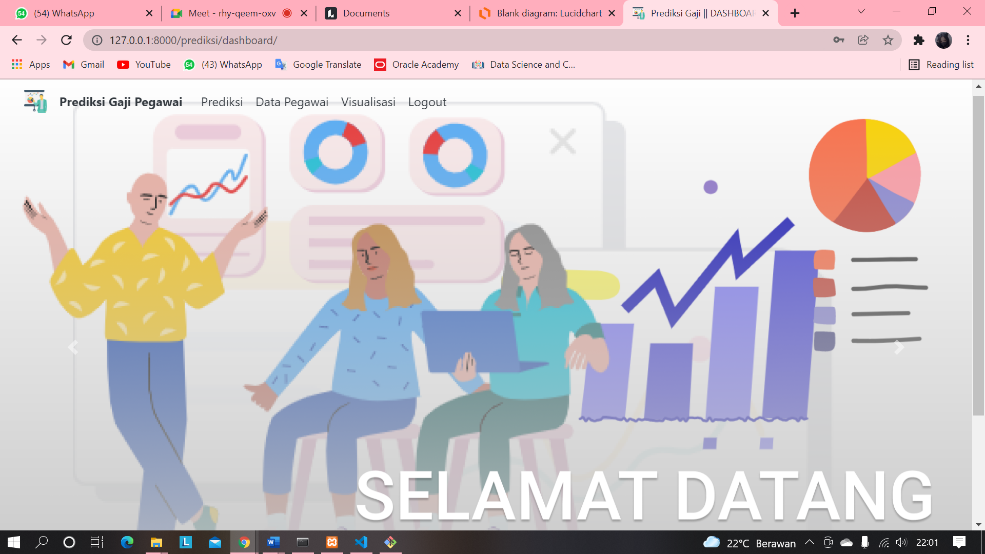
### Antarmuka Halaman Registrasi



Gambar 3. 36 Halaman Registrasi Sistem

Gambar diatas merupakan implementasi dari halaman registrasi. Jika user belum memiliki akun untuk masuk sistem, maka user dapat melakukan registrasi dengan cara menginputkan username dan password lalu menekan tombol register.

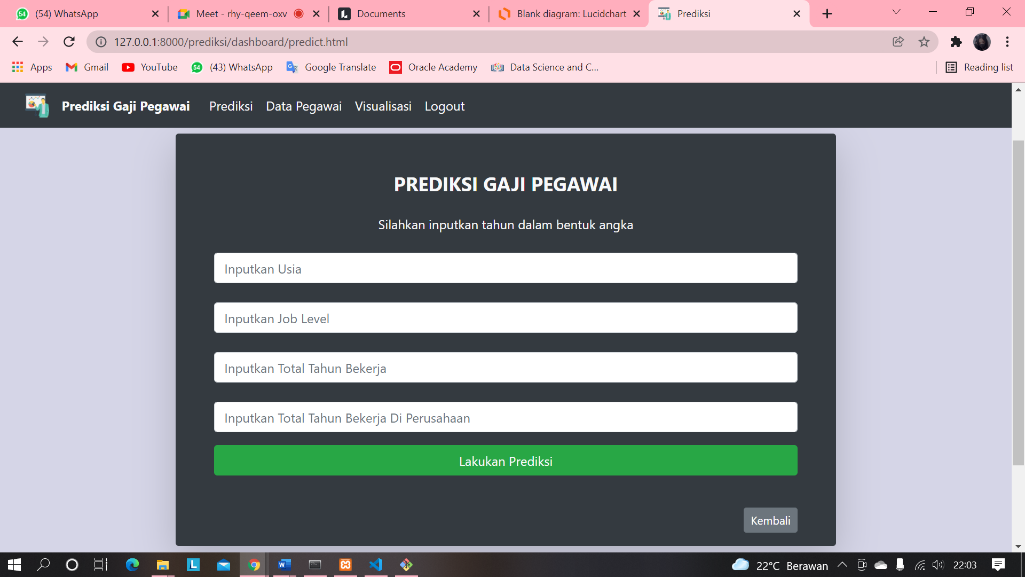
### Antarmuka Halaman *Dashboard*



Gambar 3. 37 Halaman Dashboard Sistem

Gambar diatas merupakan implementasi dari halaman dashboard dari sistem yang dirancang. Pada bagian navbar, terdapat pilihan submenu untuk mengakses suatu halaman yang lainnya. Pilihan submenu tersebut terdiri dari Prediksi, Data Pegawai, Visualisasi, dan Logout.

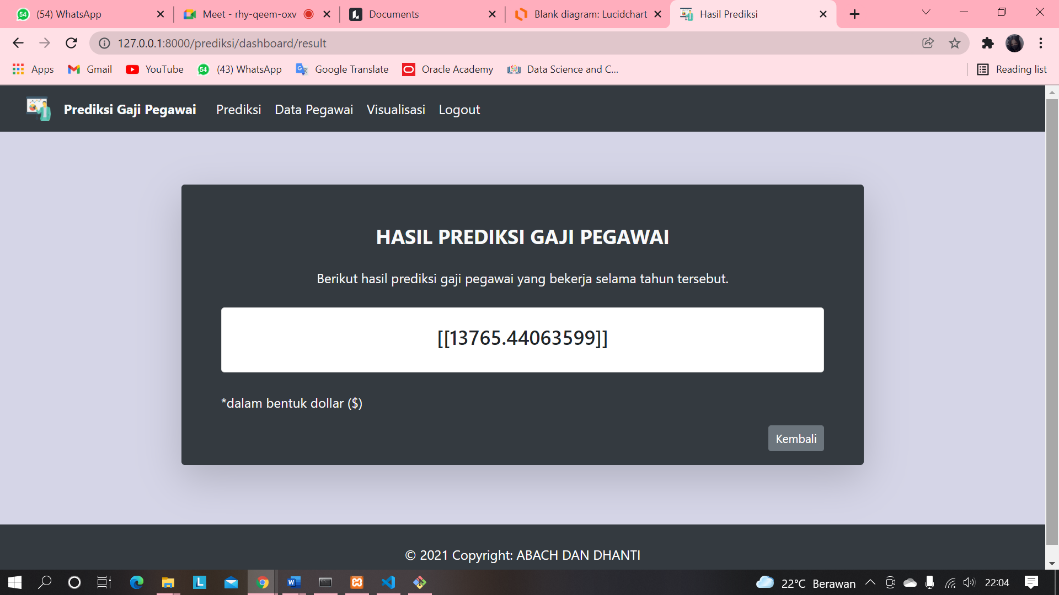
### Antarmuka Halaman Prediksi



Gambar 3. 38 Halaman Prediksi Sistem

Gambar diatas merupakan implementasi dari halaman prediksi dari sistem yang dirancang. Pada halaman ini, terdapat form untuk melakukan prediksi gaji pegawai. User dapat menginputkan angka berupa usia, job level, total tahun bekerja, dan total tahun bekerja di perusahan ke dalam form untuk mengetahui hasil prediksi gaji pegawai. Kemudian user dapat menekan button “Lakukan Prediksi” untuk melihat hasil prediksi berdasarkan tahun yang diinputkan.

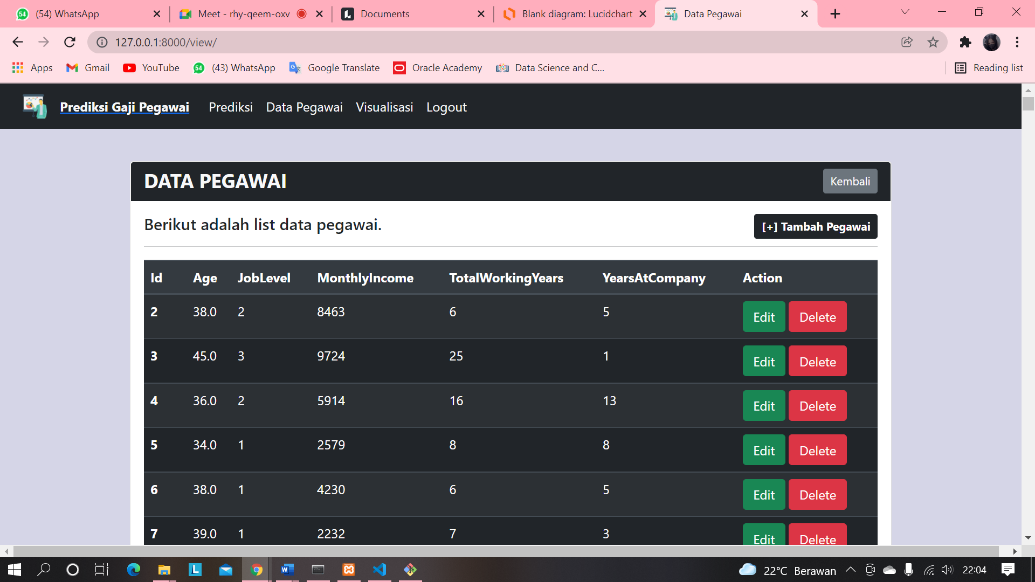
### Antarmuka Halaman Hasil Prediksi



Gambar 3. 39 Halaman Hasil Prediksi Sistem

Gambar diatas merupakan implementasi dari halaman hasil prediksi dari sistem yang dirancang. Pada halaman ini, akan ditampilkan hasil prediksi gaji dari user yang sebelumnya telah menginputkan angka tahun pada form prediksi.

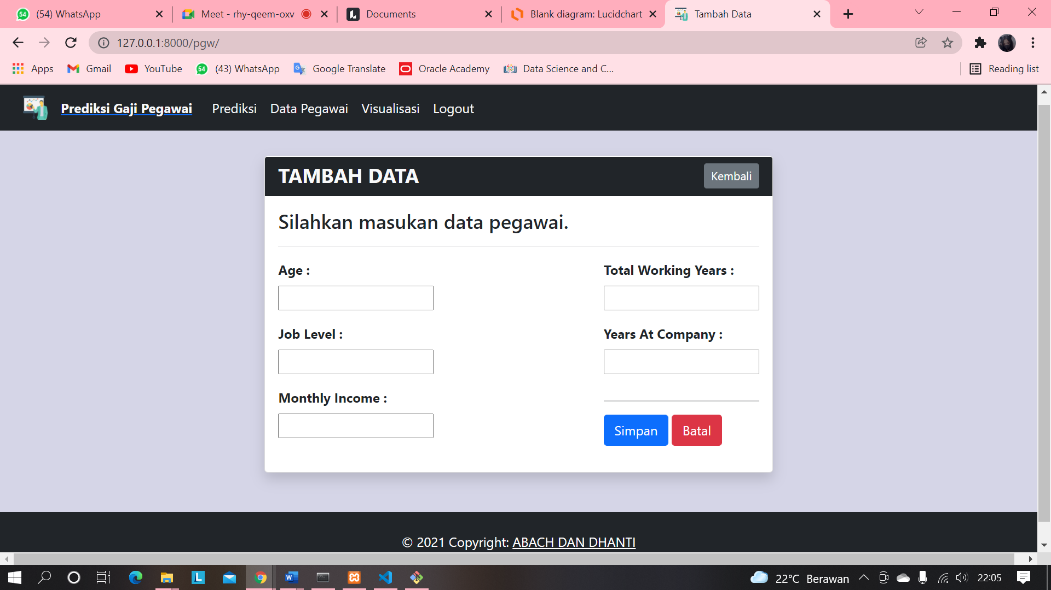
### Antarmuka Halaman Data Pegawai



Gambar 3. 40 Halaman Data Pegawai Sistem

Gambar diatas merupakan implementasi dari halaman data pegawai dari sistem yang dirancang. Pada halaman ini, akan ditampilkan data dalam bentuk tabel yang terdiri dari atribut Id, Age, JobLevel, MonthlyIncome, dan Action. Record yang ditampilkan terdiri dari 439 record. Terdapat button Tambah Pegawai yang berfungsi untuk menambah data pegawai, button Edit berfungsi untuk mengubah suatu record yang dipilih, dan button Delete untuk menghapus suatu record yang dipilih.

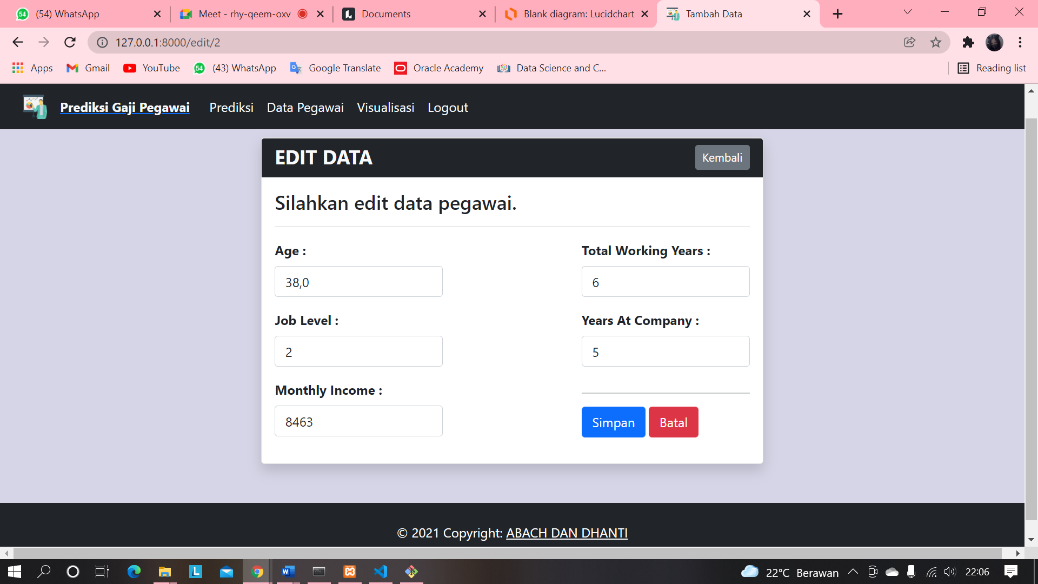
### Antarmuka Halaman Tambah Data Pegawai



Gambar 3. 41 Halaman Tambah Data Pegawai Sistem

Gambar diatas merupakan implementasi dari halaman tambah data pegawai dari sistem yang dirancang. Pada halaman ini, akan ditampilkan form tambah data untuk menambah data pegawai. User dapat menginputkan Age, Job Level, Monthly Income, Total Working Years, dan Years At Company pada form. Lalu user dapat menekan button Simpan untuk menyimpan data yang diinputkan. User dapat menekan button Batal untuk me-reset data yang diinputkan pada form.

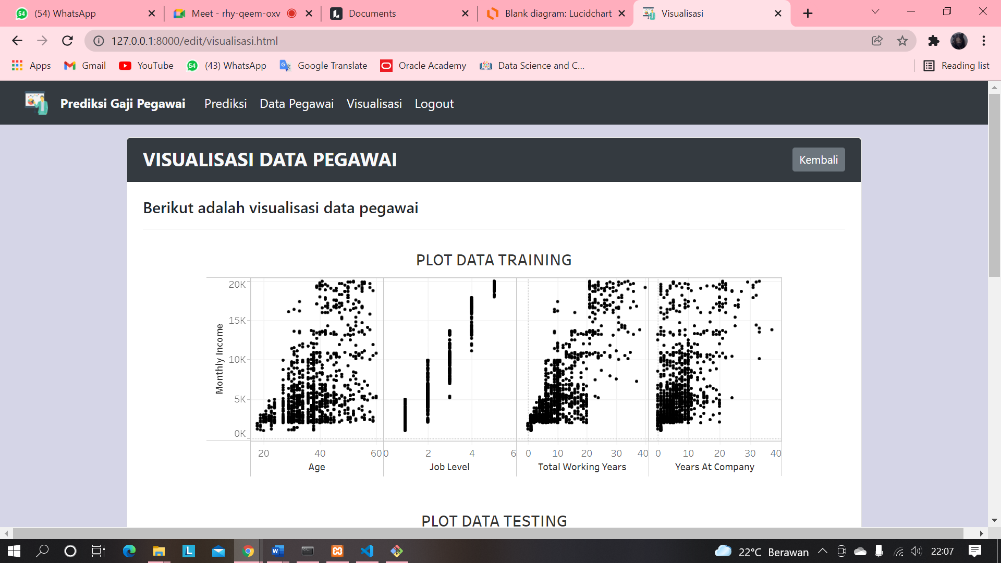
### Antarmuka Halaman Edit Data Pegawai



Gambar 3. 42 Halaman Edit Data Pegawai Sistem

Gambar diatas merupakan implementasi dari halaman edit data pegawai dari sistem yang dirancang. Pada halaman ini, akan ditampilkan form edit data untuk mengubah data pegawai. User dapat menginputkan Age, Job Level, Monthly Income, Total Working Years, dan Years At Company pada form. Lalu user dapat menekan button Simpan untuk menyimpan data yang telah diubah. User dapat menekan button Batal untuk me-reset data yang diinputkan pada form.

### Antarmuka Halaman Visualisasi



Gambar 3. 43 Halaman Visualisasi Sistem

Gambar diatas merupakan implementasi dari halaman visualisasi dari sistem yang dirancang. Pada halaman ini, akan ditampilkan bentuk visualisasi prediksi antara Monthly Income dengan Age, Job Level, Total Working Years, Years At Company. Visualisasi terdiri dari Plot Data Testing, dan Linearitas

## Implementasi Metode Regresi Linier Berganda

### Himpunan Data

Pada tahap ini, hal yang dilakukan adalah memahami dan mempersiapkan data yang dikenal dengan istilah Data Preprocessing. Metode yang digunakan dalam Data Preprocessing pada model ini adalah Data Cleaning. Berikut kode programnya :

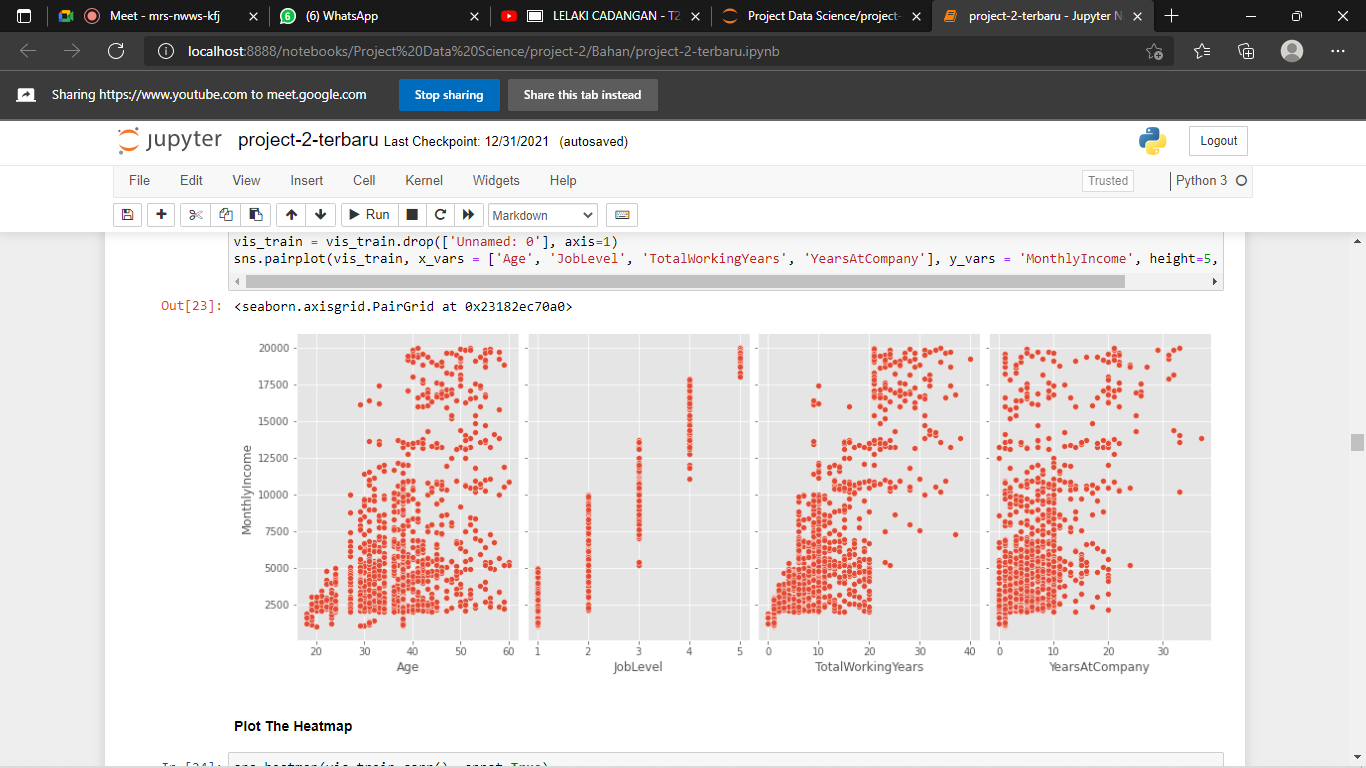
|  |  |
| --- | --- |
| [1] | # Basic Library  import pandas as pd  import numpy as np  # Data Visualization  import matplotlib.pyplot as plt  from mpl\_toolkits.mplot3d import Axes3D  import seaborn as sns  from scipy.stats import skew  # Model Building  from sklearn.linear\_model import LinearRegression  import statsmodels.api as sm |
| [2] | df\_train = pd.read\_csv('E:\Data\_Urang\Mata Kuliah\Semester 3\PROJECT II\Project New\PROJECT-2\dataset\employee\_attrition\_train.csv')  df\_train |
| [3] | df\_train.info() |
| [4] | # Encoder BusinessTravel Variable  # converting type of columns to 'category'  df\_train['BusinessTravel']= df\_train['BusinessTravel'].astype('category')  # Assigning numerical values and storing in another column  df\_train['BusinessTravel']= df\_train['BusinessTravel'].cat.codes  # Encoder Department Variable  df\_train['Department']= df\_train['Department'].astype('category')  # Assigning numerical values and storing in another column  df\_train['Department'] = df\_train['Department'].cat.codes  # Encoder EducationField Variable  df\_train['EducationField']= df\_train['EducationField'].astype('category')  # Assigning numerical values and storing in another column  df\_train['EducationField']= df\_train['EducationField'].cat.codes  # Encoder Gender Variable  df\_train['Gender'] = df\_train['Gender'].astype('category')  # Assigning numerical values and storing in another column  df\_train['Gender'] = df\_train['Gender'].cat.codes  # Encoder JobRole Variable  df\_train['JobRole'] = df\_train['JobRole'].astype('category')  # Assigning numerical values and storing in another column  df\_train['JobRole'] = df\_train['JobRole'].cat.codes  # Encoder MaritalStatus Variable  df\_train['MaritalStatus']= df\_train['MaritalStatus'].astype('category')  # Assigning numerical values and storing in another column  df\_train['MaritalStatus'] = df\_train['MaritalStatus'].cat.codes  # Encoder Over18 Variable  df\_train['Over18'] = df\_train['Over18'].astype('category')  # Assigning numerical values and storing in another column  df\_train['Over18'] = df\_train['Over18'].cat.codes  # Encoder OverTime Variable  df\_train['OverTime'] = df\_train['OverTime'].astype('category')  # Assigning numerical values and storing in another column  df\_train['OverTime'] = df\_train['OverTime'].cat.codes  df\_train |
| [5] | df\_train.isnull().values.any() |
| [6] | df\_train.isnull().sum() |
| [7] | Age = df\_train['Age']  Age.describe() |
| [8] | df\_train.Age = df\_train.Age.fillna(value=df\_train.Age.mean()) |
| [9] | DailyRate = df\_train['DailyRate']  DailyRate.describe() |
| [10] | df\_train.DailyRate= df\_train.DailyRate.fillna  (value=df\_train.DailyRate.mean()) |
| [11] | DistanceFromHome = df\_train['DistanceFromHome']  DistanceFromHome.describe() |
| [12] | df\_train.DistanceFromHome= df\_train.DistanceFromHome.fillna  (value=df\_train.DistanceFromHome.mean()) |
| [13] | df\_train\_clean = df\_train  df\_train\_clean.isnull().values.any() |
| [14] | df\_train\_clean.isnull().sum() |
| [15] | def plotCorrelationMatrix(df\_train\_clean, graphWidth):  df\_train\_clean = df\_train\_clean[[col for col in df\_train\_clean if df\_train\_clean[col].nunique() > 1]] # keep columns where there are more than 1 unique values  if df\_train\_clean.shape[1] < 2:  print(f'No correlation plots shown: The number of non-NaN or constant columns ({df\_train\_clean.shape[1]}) is less than 2')  return  corr = df\_train\_clean.corr()  plt.figure(num=None, figsize=(graphWidth, graphWidth), dpi=80, facecolor='w', edgecolor='k')  corrMat = plt.matshow(corr, fignum = 1)  plt.xticks(range(len(corr.columns)), corr.columns, rotation=90)  plt.yticks(range(len(corr.columns)), corr.columns)  plt.gca().xaxis.tick\_bottom()  plt.colorbar(corrMat)  plt.title(f'Correlation Matrix for Data Training', fontsize=15)  plt.show()  plotCorrelationMatrix(df\_train\_clean, 8) |
| [16] | df\_train\_clean.corr().abs() |
| [17] | df\_train\_clean.columns |
| [18] | df\_train\_clean = df\_train\_clean.drop(['Attrition', 'BusinessTravel', 'DailyRate', 'Department','DistanceFromHome', 'Education', 'EducationField', 'EmployeeCount','EmployeeNumber', 'EnvironmentSatisfaction', 'Gender', 'HourlyRate', 'JobInvolvement', 'JobRole', 'JobSatisfaction', 'MaritalStatus','MonthlyRate', 'NumCompaniesWorked','Over18', 'OverTime', 'PercentSalaryHike', 'PerformanceRating', 'RelationshipSatisfaction', 'StandardHours', 'StockOptionLevel', 'TrainingTimesLastYear', 'WorkLifeBalance','YearsInCurrentRole', 'YearsSinceLastPromotion', 'YearsWithCurrManager'], axis=1) |
| [19] | df\_train\_clean.corr() |
| [20] | x\_train = df\_train\_clean[['Age', 'JobLevel', 'TotalWorkingYears', 'YearsAtCompany']]  y\_train = df\_train\_clean[['MonthlyIncome']] |
| [21] | df\_train\_clean.to\_csv('E:\Data\_Urang\Mata Kuliah\Semester 3\PROJECT II\Project New\PROJECT-2\dataset\employee\_attrition\_train\_clean.csv') |

Keterangan :

* Line 1 : Untuk mengimport library yang dibutuhkan.
* Line 2 : Mengimport data ke pyhton kemudian disimpan dalam variabel dengan nama df\_train.
* Line 3 : Menampilkan info detail tabel/data yang disimpan.
* Line 4 : Melakukan encoder, karena machine learning tidak bisa membaca korelasi dari suatu variabel jika nilainya bernilai object/string.
* Line 5 : Cek apakah ada data yang kosong pada tabel.
* Line 6 : Cek berapa record yang terdapat data yag kosong.
* Line 7 : Membuat variabel Age yang berisikan atribut age, kemudian cek deskripsi variabel Age.
* Line 8 : Melakukan pengisian data kosong pada variabel Age dengan nilai mean-nya.
* Line 9 : Membuat variabel DailyRate yang berisikan atribut DailyRate, kemudian cek deskripsi variabel DailyRate.
* Line 10 : Melakukan pengisian data kosong pada variabel DailyRate dengan nilai mean-nya.
* Line 11 : Membuat variabel DistanceFromHome yang berisikan atribut DistanceFromHome, kemudian cek deskripsi variabel DistanceFromHome.
* Line 12 : Melakukan pengisian data kosong pada variabel DistanceFromHome dengan nilai mean-nya.
* Line 13 : Membuat variabel df\_train\_clean yang berisikan data df\_train, kemudian cek apakah masih terdapat data yang kosong atau tidak.
* Line 14 : Cek berapa record yang terdapat data yag kosong.
* Line 15 : Membuat heatmap antar variabel yang terdapat dalam variabel df\_train\_clean.
* Line 16 : Membuat tabel korelasi antar variabel.
* Line 17 : Menampilkan kumpulan variabel yang terdapat dalam df\_train\_clean.
* Line 18 : Melakukan drop variabel yang tidak diperlukan.
* Line 19 : Membuat tabel korelasi setelah proses drop variabel.
* Line 20 : Menetapkan variabel independent (sumbu x) yaitu variabel Age, JobLevel, TotalWorkingYears, dan YearsAtCompany. Kemudian menetapkan variabel variabel dependen (sumbu y) yaitu MonthlyIncome.
* Line 21 : Melakukan export data ke format file csv.

1. Visualisasi Data Training

|  |  |
| --- | --- |
| [22] | %matplotlib inline  plt.style.use('ggplot')  plt.rcParams['figure.figsize'] = (12,8) |
| [23] | vis\_train = pd.read\_csv('E:\Data\_Urang\Mata Kuliah\Semester 3\PROJECT II\Project New\PROJECT-2\dataset\employee\_attrition\_train\_clean.csv')  vis\_train = vis\_train.drop(['Unnamed: 0'], axis=1)  sns.pairplot(vis\_train, x\_vars = ['Age', 'JobLevel', 'TotalWorkingYears', 'YearsAtCompany'], y\_vars = 'MonthlyIncome', height=5, aspect=0.7) |
| [24] | sns.heatmap(vis\_train.corr(), annot=True) |



Keterangan :

* Line 22 : Membuat style untuk diagram plot.
* Line 23 : Melakukan import data kemudian disimpan dalam variabel vis\_train. Drop kolom yang tidak diperlukan. Kemudian menentukan variabel x dan y untuk diagram plot.
* Line 24 : Membuat heatmap untuk melihat korelasi antar variabel.

### Proses Data Mining & Pengetahuan

Pada tahapan Proses Data Mining hal yang dilakukan adalah memilih metode yang sesuai dengan karakter data yang dikenal dengan istilah Modelling. Pada model ini digunakan Proses Data Mining Prediction. Pada tahapan Pengetahuan hal yang dilakukan adalah memahami model dan pengetahuan yang sesuai sehingga dapat memilih model. Model yang digunakan adalah Linear Regression menggunakan Scikit Learn.

|  |  |
| --- | --- |
| [25] | df\_test = pd.read\_csv('E:\Data\_Urang\Mata Kuliah\Semester3\PROJECT II\Project New\PROJECT-2\dataset\employee\_attrition\_test.csv')  df\_test |
| [26] | # Encoder BusinessTravel Variable  # converting type of columns to 'category'  df\_test['BusinessTravel'] = df\_test['BusinessTravel'].astype('category')  # Assigning numerical values and storing in another column  df\_test['BusinessTravel'] = df\_test['BusinessTravel'].cat.codes  # Encoder Department Variable  df\_test['Department'] = df\_test['Department'].astype('category')  # Assigning numerical values and storing in another column  df\_test['Department'] = df\_test['Department'].cat.codes  # Encoder EducationField Variable  df\_test['EducationField'] = df\_test['EducationField'].astype('category')  # Assigning numerical values and storing in another column  df\_test['EducationField'] = df\_test['EducationField'].cat.codes  # Encoder Gender Variable  df\_test['Gender'] = df\_test['Gender'].astype('category')  # Assigning numerical values and storing in another column  df\_test['Gender'] = df\_test['Gender'].cat.codes  # Encoder JobRole Variable  df\_test['JobRole'] = df\_test['JobRole'].astype('category')  # Assigning numerical values and storing in another column  df\_test['JobRole'] = df\_test['JobRole'].cat.codes  # Encoder MaritalStatus Variable  df\_test['MaritalStatus'] = df\_test['MaritalStatus'].astype('category')  # Assigning numerical values and storing in another column  df\_test['MaritalStatus'] = df\_test['MaritalStatus'].cat.codes  # Encoder Over18 Variable  df\_test['Over18'] = df\_test['Over18'].astype('category')  # Assigning numerical values and storing in another column  df\_test['Over18'] = df\_test['Over18'].cat.codes  # Encoder OverTime Variable  df\_test['OverTime'] = df\_test['OverTime'].astype('category')  # Assigning numerical values and storing in another column  df\_test['OverTime'] = df\_test['OverTime'].cat.codes  df\_test |
| [27] | df\_test.isnull().values.any() |
| [28] | df\_test.isnull().sum() |
| [29] | # Missing Value in Age  Age = df\_test['Age']  df\_test.Age = df\_test.Age.fillna(value=df\_test.Age.mean())  # Missing Value in DailyDate  DailyRate = df\_test['DailyRate']  df\_test.DailyRate = df\_test.DailyRate.fillna(value=df\_test.DailyRate.mean())  # Missing Value in DistanceFromHome  DistanceFromHome = df\_test['DistanceFromHome']  df\_test.DistanceFromHome = df\_test.DistanceFromHome.fillna(value=df\_test.DistanceFromHome.mean())  # Missing Value in BusinessTravel  BusinessTravel = df\_test['BusinessTravel']  df\_test.BusinessTravel = df\_test.BusinessTravel.fillna(value=df\_test.BusinessTravel.mean())  # Missing Value in MartialStatus  MaritalStatus = df\_test['MaritalStatus']  df\_test.MaritalStatus = df\_test.MaritalStatus.fillna(value=df\_test.MaritalStatus.mean()) |
| [30] | df\_test.isnull().values.any() |
| [31] | df\_test.isnull().sum() |
| [32] | # Eliminasi Variabel yang Tidak akan digunakan  df\_test\_clean = df\_test.drop(['BusinessTravel', 'DailyRate', 'Department','DistanceFromHome', 'Education', 'EducationField', 'EmployeeCount','EmployeeNumber', 'EnvironmentSatisfaction', 'Gender', 'HourlyRate','JobInvolvement','JobRole','JobSatisfaction','MaritalStatus','MonthlyRate','NumCompaniesWorked','Over18','OverTime','PercentSalaryHike','PerformanceRating','RelationshipSatisfaction','StandardHours','StockOptionLevel','TrainingTimesLastYear','WorkLifeBalance','YearsInCurrentRole','YearsSinceLastPromotion','YearsWithCurrManager'], axis=1) |
| [33] | df\_test\_clean.corr().abs() |
| [34] | x\_test = df\_test\_clean[['Age', 'JobLevel', 'TotalWorkingYears', 'YearsAtCompany']]  y\_test = df\_test\_clean[['MonthlyIncome']] |
| [35] | df\_test\_clean.to\_csv('E:\Data\_Urang\Mata Kuliah\Semester 3\PROJECT II\Project New\PROJECT-2\dataset\employee\_attrition\_test\_clean.csv') |
| [36] | vis\_test = pd.read\_csv('E:\Data\_Urang\Mata Kuliah\Semester 3\PROJECT II\Project New\PROJECT-2\dataset\employee\_attrition\_test\_clean.csv')  vis\_test = vis\_test.drop(['Unnamed: 0'], axis=1)  sns.pairplot(vis\_test, x\_vars = ['Age', 'JobLevel', 'TotalWorkingYears', 'YearsAtCompany'], y\_vars = 'MonthlyIncome', height=5, aspect=0.7) |
| [37] | sns.heatmap(vis\_test.corr(), annot=True) |
| [38] | regressor = LinearRegression()  persamaan = regressor.fit(x\_train, y\_train)  print(regressor.coef\_)  print(regressor.intercept\_) |
| [39] | y\_pred = regressor.predict(x\_test)  print(y\_pred) |
| [40] | vis\_test['MonthlyIncome Prediction'] = y\_pred.tolist()  vis\_test |
| [41] | vis\_test.to\_excel('E:\Data\_Urang\Mata Kuliah\Semester 3\PROJECT II\Project New\PROJECT-2\dataset\employee\_vis\_test.xlsx') |

Keterangan :

* Line 25 : Melakukan import data testing kemudian disimpan dalam variabel df\_test.
* Line 26 : Melakukan encoder, karena machine learning tidak bisa membaca korelasi dari suatu variabel jika nilainya bernilai object/string. Line 27 : Cek apakah masih terdapat data yang kosong atau tidak.
* Line 28 : Cek apakah ada data yang kosong pada tabel.
* Line 29 : Membuat variabel kemudian melakukan pengisian data kosong pada variabel dengan nilai mean-nya.
* Line 30 : Cek apakah masih terdapat data yang kosong atau tidak.
* Line 31 : Cek apakah ada data yang kosong pada tabel.
* Line 32 : Melakukan drop variabel yang tidak diperlukan.
* Line 33 : Membuat tabel korelasi setelah proses drop variabel.
* Line 34 : Menetapkan variabel independent (sumbu x) yaitu variabel Age, JobLevel, TotalWorkingYears, dan YearsAtCompany. Kemudian menetapkan variabel variabel dependen (sumbu y) yaitu MonthlyIncome.
* Line 35 : Melakukan export data ke format file csv.
* Line 36 : Melakukan import data kemudian disimpan dalam variabel vis\_train. Drop kolom yang tidak diperlukan. Kemudian menentukan variabel x dan y untuk diagram plot.
* Line 37 : membuat heatmap untuk melihat korelasi antar variabel.
* Line 38 : Membuat variabel regressor yang isinya metode LinearRegression, kemudian membuat variabel persamaan yang isinya terdapat method regressor.fit dengan parameternya x\_train dan y\_train.
* Dari model tersebut didapat nilai koefisien dari variabel Independen. Nilai koefisien dari Age adalah -5,054 yang berarti tiap karyawan yang mengalami pertambahan satu tahun umur kerja, maka akan mengalami penurunan gaji sebesar 5,054. Nilai koefisien dari JobLevel adalah 3871,7530 yang berarti tiap karyawan yang mengalami pertambahan satu tingkat job level akan mengalami kenaikan gaji sebesar 3871,7530. Nilai koefisien TotalWorkingYears adalah 46,9405 yang berarti tiap karyawan yang mengalami pertambahans satu tahun pengalaman bekerja akan mengalami kenaikan kerja sebesar 46,9405. Nilai koefisien YearsAtCompany adalah -9,8460 yang berarti tiap karyawan yang mengalami pertambahan satu tahun akan mengalami penurunan gaji sebesar 9,8460.
* Line 39 : Mencari konstanta/intercept menggunakan regressor, kemudian ditampilkan.
* Intecept merupakan sebuah koefisien dari sebuah persamaan model regresi linear. Untuk kasus ini, berarti untuk variabel X yang bernilai nol atau karyawan yang belum berpengalaman kerja, karyawan tersebut akan menerima gaji sebesar 26611 per tahunnya.
* Line 40 : Melakukan prediksi data testing menggunkana model machine learning. Kemudian buat kolom baru yang Bernama MonthlyIncome Prediction yang berisikan nilai prediksi.
* Berikut persamaan umum dari model linear regresi multivariabel :

Y = β0 + β1X1 + β2X2 + β3X3 + β4X4 +….+ βnXn

β0 adalah nilai intersept dari persamaan linear, dan β1, β2, β3 sampai dengan βn adalah konstanta dari variabel independen. Berdasarkan nilai koefisien variabel independen dan Intersept didapat, maka persamaan regresi linear multivariabel sebagai berikut :

Y = -1728 – 5,054X1 + 3871,7530X2 + 46,9405X3 – 9,8460X4

Keterangan:

Y = Variabel Dependen yaitu MonthlyIncome

X1 = Variabel Independen pertama yaitu Age

X2 = Variabel Independen kedua yaitu JobLevel

X3 = Variabel Independen ketiga yaitu TotalWorkingYears

X4 = Variabel Independen keempat yaitu YearsAtCompany

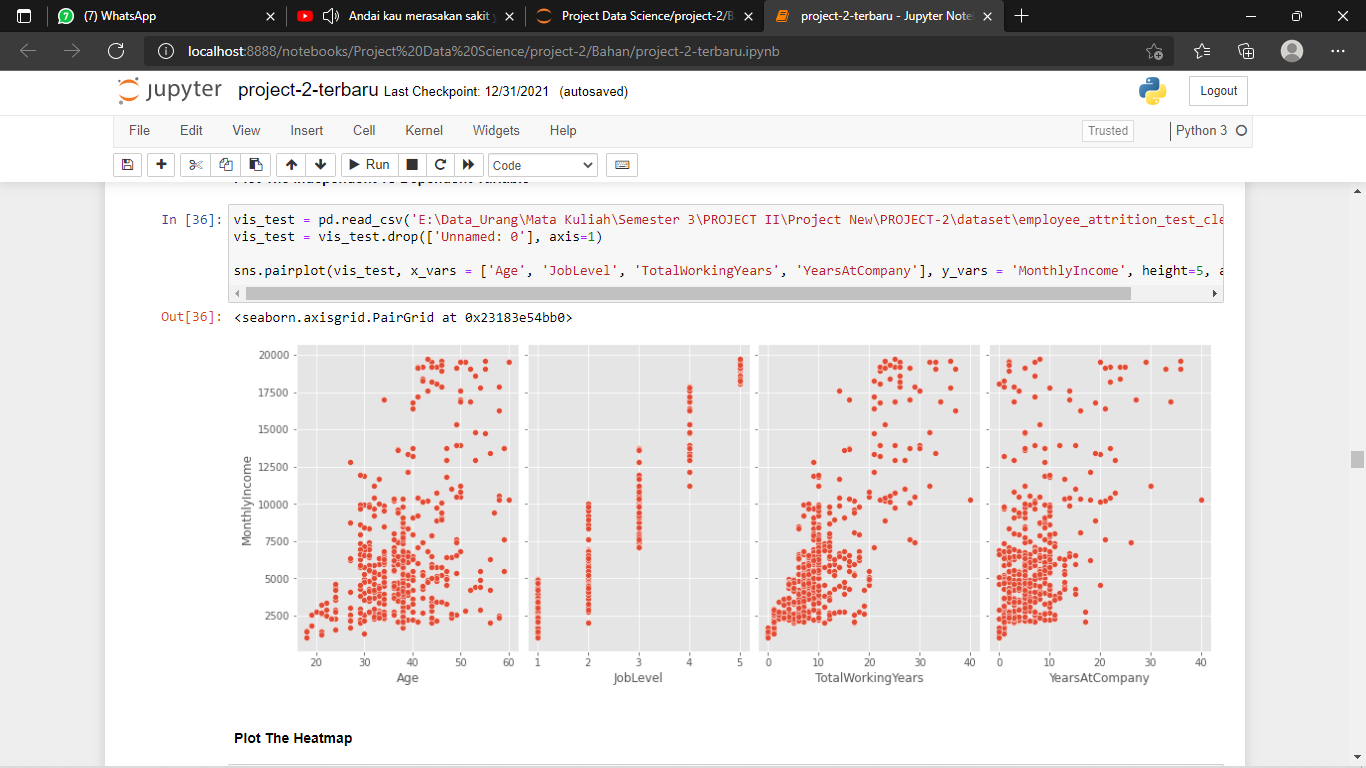
Maka dapat disimpulkan, persamaan regresi linear multivariabel sebagai berikut :

**MonthlyIncome = -1728 – 5,054(Age) + 3871,7530(JobLevel) + 46,9405(TotalWorkingYears) – 9,8460(YearsAtCompany)**

* Line 41 : Melakukan export data setelah prediksi ke dalam format file excel.

1. Visualisasi Data Testing

|  |  |
| --- | --- |
| [36] | vis\_test = pd.read\_csv('E:\Data\_Urang\Mata Kuliah\Semester 3\PROJECT II\Project New\PROJECT-2\dataset\employee\_attrition\_test\_clean.csv')  vis\_test = vis\_test.drop(['Unnamed: 0'], axis=1)  sns.pairplot(vis\_test, x\_vars = ['Age', 'JobLevel', 'TotalWorkingYears', 'YearsAtCompany'], y\_vars = 'MonthlyIncome', height=5, aspect=0.7) |
| [37] | sns.heatmap(vis\_test.corr(), annot=True) |



Keterangan :

* Line 36 : Melakukan import data kemudian disimpan dalam variabel vis\_train. Drop kolom yang tidak diperlukan. Kemudian menentukan variabel x dan y untuk diagram plot.
* Line 37 : membuat heatmap untuk melihat korelasi antar variabel.

### Evaluasi Data

Pada tahap ini hal yang dilakukan adalah melakukan evaluasi terhadap analisis model dan kinerja metode.

1. Validasi Model

|  |  |
| --- | --- |
| [42] | X = df\_train\_clean[['Age', 'JobLevel', 'TotalWorkingYears', 'YearsAtCompany']]  X = sm.add\_constant(X) # adding a constant  olsmod = sm.OLS(df\_train['MonthlyIncome'], X).fit()  print(olsmod.summary()) |



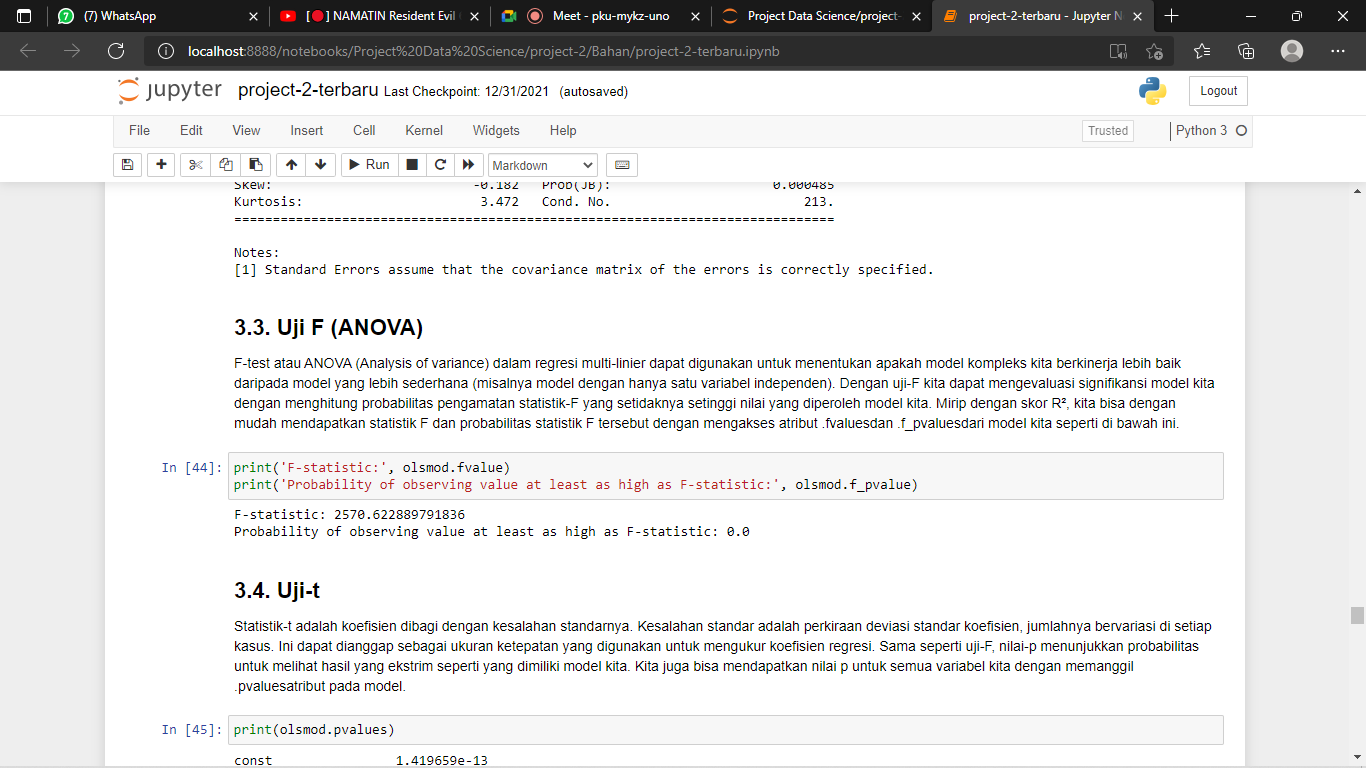
Keterangan :

* Line 42 : Membuat variabel x yang berisikan atribut Age, JobLevel, TotalWorkingYears dan YearsAtCompany. Variabel X digunakan utnuk melihat validasi model machine learning dengan model OLS.

1. Uji F (ANOVA)

F-test atau ANOVA (Analysis of Variance) dalam regresi multi-linear dapat digunakan untuk menentukan apakah model kompleks yang dibuat berkinerja lebih baik daripada model yang lebih sederhana (misalnya model dengan hanya satu variabel independent). Dengan uij-F dapat dievaluasi signifikansi model yang dibuat dengan menghitung probabilitas pengamatan statistic-F yang setidaknya setinggi nilai yang diperoleh model yang dbuat. Mirip dengan skor R2, dapat dengan mudah mendapatkan statistic F dan probabilitas statistic F tersebut dengan mengakses atribut .fvalues dan .f\_pvalues dari model yang dibuat seperti di bawah ini.

|  |  |
| --- | --- |
| [43] | print('F-statistic:', olsmod.fvalue)  print('Probability of observing value at least as high as F-statistic:', olsmod.f\_pvalue) |



Keterangan :

* Line 43 : Untuk menampilkan nilai F-statistik dan P-value dari model machine learning dengan model OLS.

1. P-Value

P-Value adalah nilai probabilitas yang dapat diartikan sebagai besarnya peluang (probabilitas) yang diamati dari statistic uji. Variabel independen (Age, JobLevel, TotalWorkingYears, YearsAtCompany) mempunyai nilai p-value dibawah 0.05, hal ini menandakan ia signifikan melakukan prediksi variabel dependen (MonthlyIncome).

1. Fs

Fs adalah hasil akhir dari analisis ANOVA. Nilai Fs ini yang dikenal dengan F hitung dalam pengujian hipotesa dibandingkan dengan nilai p-value. Jika Fs > P-value, maka dapat dinyatakan bahwa secara simultan (bersama-sama) variabel dependen dan variabel independen berpengaruh signifikan terhadap permintaan. Hipotesa yang didapat dari tabel ANOVA di atas adalah :

H0 = Variabel independen secara simultan bukan penjelas yg signifikan terhadap variabel dependen (Model tidak cocok).

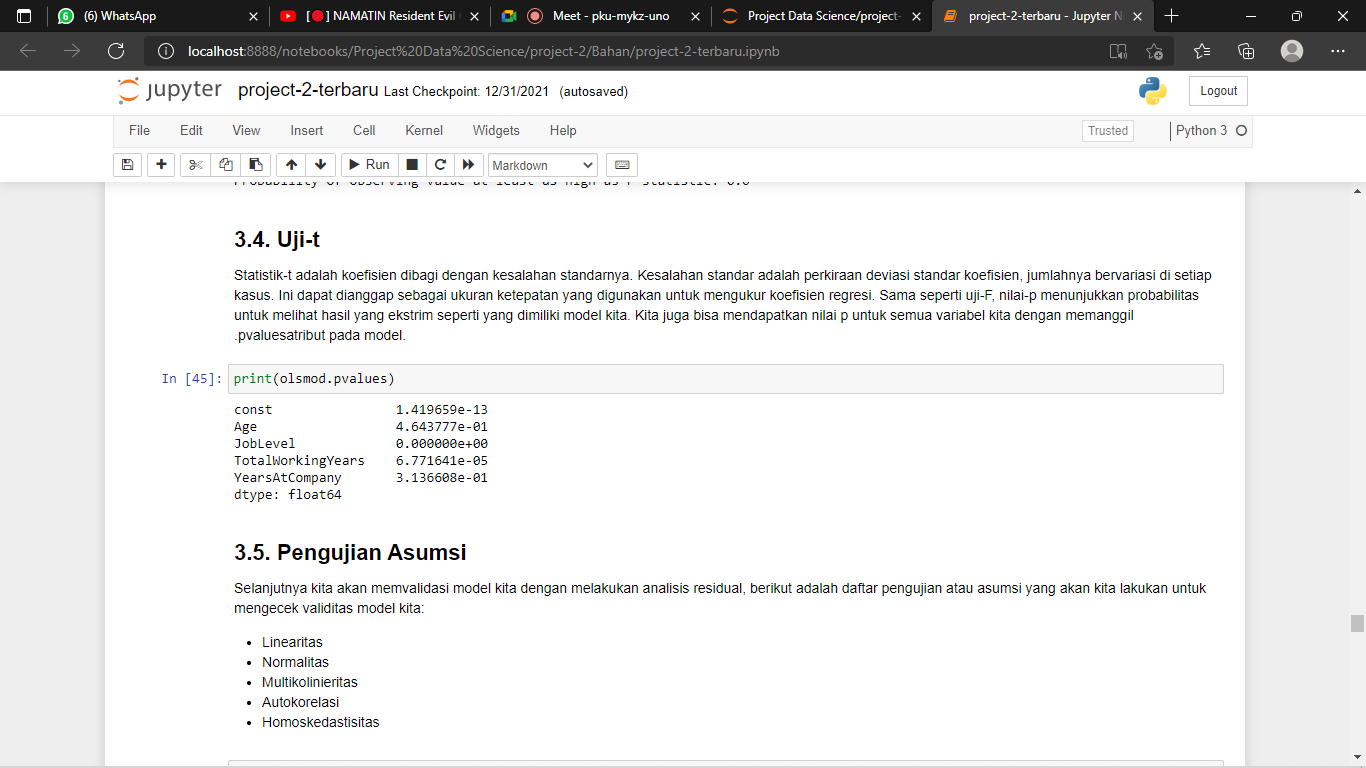
H1 = Variabel independen secara simultan merupakan penjelas yang siginifikan terhadap variabel dependen (Model cocok).

Berdasarkan tabel diatas, dapat diketahui bahwa Fs > P-value, yang artinya hipotesa yang dapat diambil adalah terima H1 dan tolak H0. Dapat dikatakan, variabel Independen (Age, JobLevel, TotalWorkingYears, YearsAtCompany) dan MonthlyIncome berpengaruh signifikan terhadap permintaan. Pada taraf signifikansi 5% (0,05), H0 ditolak karena nilai probabilitasnya yaitu 0,00 yang berarti dibawah dari 5%. Maka dapat disimpulkan, model yang dipakai cocok.

1. Uji-t

Statistik-t adalah koefisien dibagi dengan kesalahan standarnya. Kesalahan standar adalah perkiraan deviasi standar koefisien, jumlahnya bervariasi di setiap kasus. Ini dapat dianggap sebagai ukuran ketetapan yang digunakan untuk mengukur koefisien regresi. Sama seperti uji-F, nilai-p menunjukkan probabilitas untuk melihat hasil yang ekstrim seperti yang dimiliki oleh model yang dibuat. Selain itu, uji-t juga bisa mendapatkan nilai p untuk seluruh variabel dengan memanggil .pvalues atribut pada model.

|  |  |
| --- | --- |
| [44] | print(olsmod.pvalues) |



Keterangan :

* Line 44 : Untuk menampilkan nilai p-values dari variabel independen.

Hipotesa yang dapat diambil adalah :

H0 = Variabel independen tidak berpengaruh signifikan

H1 = Variabel independen berpengaruh signifikan.

α = 0,05 (Taraf signifikansi)

Berdasarkan uji-t dapat diambil hipotesa sebagai berikut :

Nilai variabel X1 (Age) berada di atas taraf signifikansi yang berarti terima H1.

Nilai variabel X2 (JobLevel) berada di bawah taraf signifikansi yang berarti terima H0.

Nilai variabel X3 (TotalWorkingYears) di bawah taraf signifikansi yang berarti terima H0.

Nilai variabel X4 (YearsAtCompany) di atas taraf signifikansi yang berarti terima H1.

Berdasarkan hipotesa di atas, dapat disimpulan variabel independen JobLevel dan TotalWorkingYears adalah variabel yang tidak mempengaruhi variabel dependen. Sedangkan variabel independen Age dan YearsAtCompany adalah variabel yang mempengaruhi variabel dependen.

1. R-Square

R2-Square adalah sebuah nilai yang menyatakan seberapa sesuai hasil prediksi model mendekati data yang sebenarnya. Semakin besar r2-square, maka hasil prediksi semakin dekat dengan data yang sebenarnya, artinya sama saja dengan semakin mendekati 1, maka semakin baik model tersebut.



Berdasarkan tabel ANOVA, nilai koefisien determinasi (R-Square) adalah 0,909 atau 90,9%. Maka, MonthlyIncome dipengarui oleh faktor Age dan YearsAtCompany sebesar 0,909 atau 90,9%. Nilai sisa dari koefisien determinasi adalah 0.091 atau 9,1% dipengaruhi oleh faktor lain yang tidak diketahui.

1. Pengujian Asumsi

Untuk memvalidasi model machine learning dilakukan dengan analisis residual. Berikut adalah daftar pengujian atau asumsi yang akan lakukan untuk mengetahui validitas model :

|  |  |
| --- | --- |
| [45] | df\_test\_new = pd.read\_excel('E:\Data\_Urang\Mata Kuliah\Semester 3\PROJECT II\Project New\PROJECT-2\dataset\employee\_vis\_test.xlsx')  df\_test\_new2 = df\_test\_new.drop(['Unnamed: 0'], axis=1)  df\_test\_new2['MonthlyIncome Prediction'] = olsmod.predict(X)  df\_test\_new2['residual'] = olsmod.resid  df\_test\_new2 |

Keterangan :

* Line 45 : Mengimport data kemudian disimpan dalam variabel df\_test\_new. Setelah itu melakukan drop atribut. Melakukan kolom baru dengan nama MonthlyIncome Prediction yang berisikan dengan hasil prediksi. Kemudian dibuat juga kolom residual untuk menyimpan nilai residualnya.

1. Linearitas

Dengan linearitas dapat diasumsikan bahwa terdapat hubungan linear antara variabel bebas dan variabel terikat. Dengan menggunakan plot pencar dapat diketahui perbandingan nilai prediksi dengan nilai sebenarnya.

|  |  |
| --- | --- |
| [46] | # Plotting the observed vs predicted values  sns.lmplot(x='MonthlyIncome', y='MonthlyIncome Prediction', data=df\_test\_new, fit\_reg=False, size=5)  # Plotting the diagonal line  line\_coords = np.arange(vis\_test[['MonthlyIncome', 'MonthlyIncome Prediction']].min().min()-10,  vis\_test[['MonthlyIncome', 'MonthlyIncome Prediction']].max().max()+10)  plt.plot(line\_coords, line\_coords, # X and y points  color='darkorange', linestyle='--')  plt.ylabel('Predicted MonthlyIncome', fontsize=14)  plt.xlabel('Actual MonthlyIncome', fontsize=14)  plt.title('Linearity Assumption', fontsize=16)  plt.show() |



Keterangan :

* + Line 46 : Untuk menampilkan grafik linearitas dari perbandingan nilai MonthlyIncome dengan MonthlyIncomePrediction.
  + Plot sebar menenujukkan sisa yang tersebar merata di sekitar garis diagonal, sehingga dapat diasumsikan bahwa ada hubungan linier antara variable independent dan dependen.

1. Normalitas

Berdasarkan asumsi diatas, dapat dikatakan istilah kesalahan model terdistribusi tidak normal. Memeriksa normalitas residual dengan memplotnya ke dalam histogram dan melihat nilai p dan uji normalitas Anderson-Darling dengan menggunakan normal\_ad( ) fungsi dari statsmodel untuk menghitung p-value dan kemudian membandingkannya dengan threshold 0,05. Jika p-value yang diperoleh lebih tinggi dari threshold maka dapat diasumsikan bahwa residual terdistribusi normal. Jika p-value yang diperoleh lebih kecil dari threshold, maka dapat diasumsikan bahwa residual terdistribusi tidak normal.

|  |  |
| --- | --- |
| [47] | from statsmodels.stats.diagnostic import normal\_ad  # Performing the test on the residuals  p\_value = normal\_ad(df\_test\_new2['residual'])[1]  print('p-value from the test Anderson-Darling test below 0.05 generally means non-normal:', p\_value)  # Plotting the residuals distribution  plt.subplots(figsize=(8, 4))  plt.title('Distribution of Residuals', fontsize=18)  sns.distplot(df\_test\_new2['residual'])  plt.show()  # Reporting the normality of the residuals  if p\_value < 0.05:  print('Residuals are not normally distributed')  else:  print('Residuals are normally distributed') |



Keterangan :

* + Line 47 : Mengimport fungsi normal\_ad untuk mencari nilai normalitas. Kemudian dibuat grafik batang distribusi dari residual data.

Dapat diketahui hipotesa sebagai berikut :

H0 = Residual terdistribusi normal.

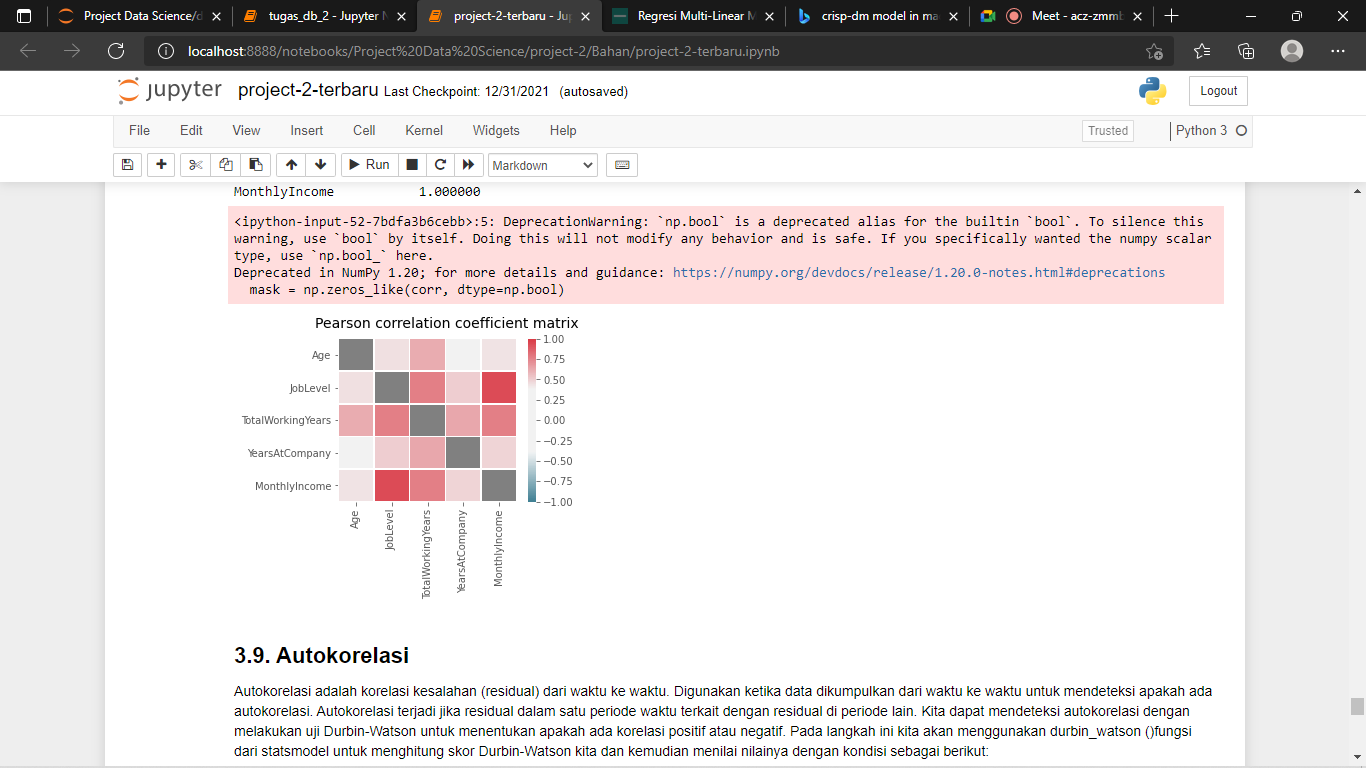
H1 = Residual terdistribusi secara tidak normal.

Dari hasil perhitungan diatas, dapat diketahui bahwa nilai p-value yang dihitung menggunakan metode Anderson-Darling adalah 0,00032261. Angka tersebut berada di bawah nilai threshold yang ditentukan yaitu 0,05, yang berarti tolah H0 terima H1 atau dapat dikatakan residual terdistribusi secara tidak normal. Sehingga disimpulakn asumsi normalitas terpenuhi.

1. Multikolinieritas

Dari hasil asumsi diatas, dapat dikatakan bahwa prediktor yang digunakan dalam regresi berkorelasi satu sama lain. Untuk mengidentifikasi apakah ada korelasi antara prediktor, dapat dihitung koefisien korelasi Pearson antara setiap kolom dalam data menggunakan corr( ) fungsi dari kerangka data Pandas. Kemudian dapat divisualisasikan sebagai peta panas menggunakan heatmap( ) fungsi dari seaborn.

|  |  |
| --- | --- |
| [48] | corr = vis\_test[['Age', 'JobLevel', 'TotalWorkingYears', 'YearsAtCompany', 'MonthlyIncome']].corr()  print('Pearson correlation coefficient matrix of each variables:\n', corr)  # Generate a mask for the diagonal cell  mask = np.zeros\_like(corr, dtype=np.bool)  np.fill\_diagonal(mask, val=True)  # Initialize matplotlib figure  fig, ax = plt.subplots(figsize=(4, 3))  # Generate a custom diverging colormap  cmap = sns.diverging\_palette(220, 10, as\_cmap=True, sep=100)  cmap.set\_bad('grey')  # Draw the heatmap with the mask and correct aspect ratio  sns.heatmap(corr, mask=mask, cmap=cmap, vmin=-1, vmax=1, center=0, linewidths=.5)  fig.suptitle('Pearson correlation coefficient matrix', fontsize=14)  ax.tick\_params(axis='both', which='major', labelsize=10)  # fig.tight\_layout() |

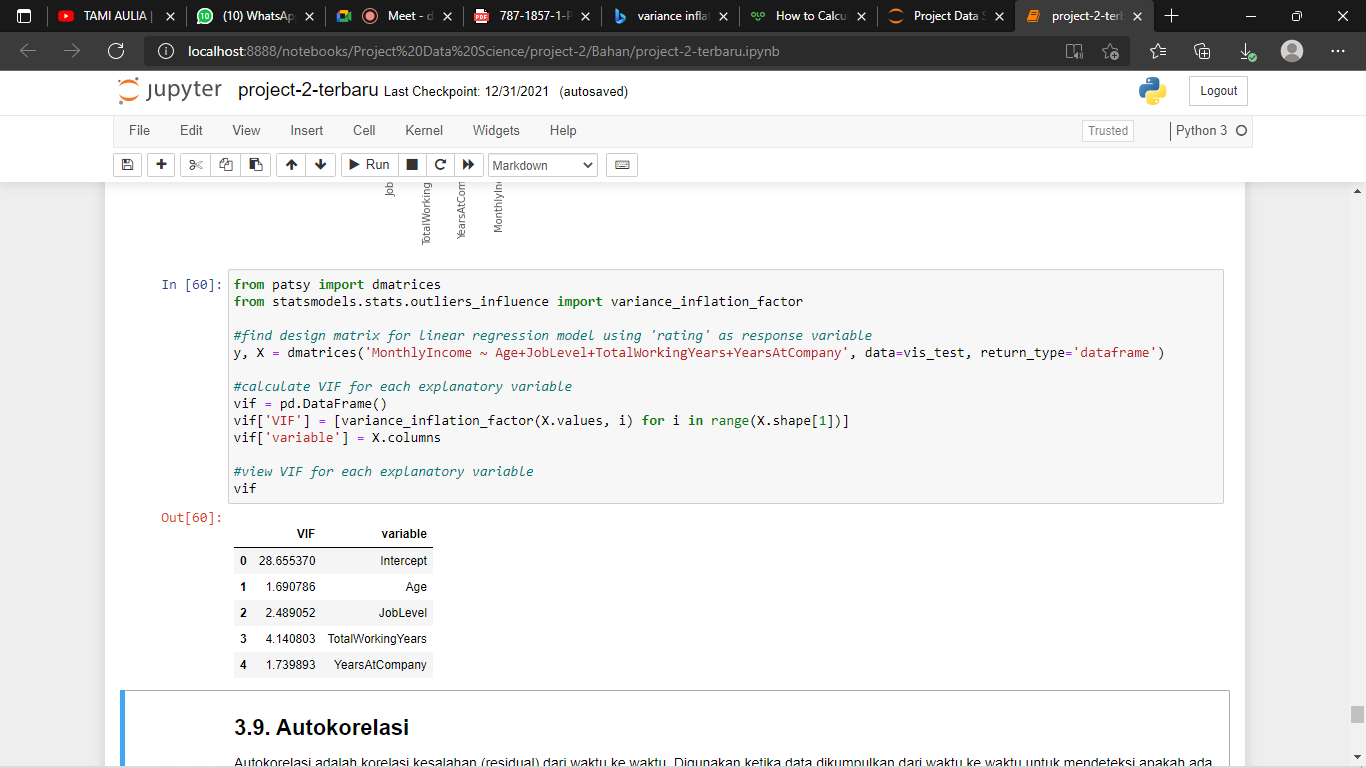


Keterangan :

* + Line 48 : Untuk mencari nilai korelasi antar variabel dan membuat tabel matriks korelasi.

Masalah multikolinearitas itu muncul jika terdapat hubungan yang sempurna pada satu ada lebih variabel independen dalam model. Pengujian multikolinearitas dapat dilakukan dengan melihat nilai koefisien P-if nya.

|  |  |
| --- | --- |
| [49] | from patsy import dmatrices  from statsmodels.stats.outliers\_influence import variance\_inflation\_factor  #find design matrix for linear regression model using 'rating' as response variable  y, X = dmatrices('MonthlyIncome ~ Age+JobLevel+TotalWorkingYears+YearsAtCompany', data=vis\_test, return\_type='dataframe')  #calculate VIF for each explanatory variable  vif = pd.DataFrame()  vif['VIF'] = [variance\_inflation\_factor(X.values, i) for i in range(X.shape[1])]  vif['variable'] = X.columns  #view VIF for each explanatory variable  vif |



Keterangan :

* + Line 49 : Untuk mencari nilai VIF variabel independen dan VIF intersep.

Berdasarkan gambar di atas dapat dilihat nilai variabel Age, JobLevel, TotalWorkingYears, YearsAtCompany memiliki nilia kurang dari 10 sehingga dengan menggunakan tingkat signifikansi sebesar 0,05 dapat disimpulkan bahwa pada data tersebut tidak terdapat multikolinearitas pada variabel-variabel prediktor.

1. Autokorelasi

Autokorelasi adalah korelasi kesalahan (residual) dari waktu ke waktu. Digunakan ketika data dikumpulkan dari waktu ke waktu untuk mendeteksi apakah ada autokorelasi. Autokorelasi terjadi jika residual dalam satu periode waktu terkait dengan residual di periode lain. Autokorelasi dapat dideteksi dengan melakukan uij Durbin-Watson untuk menentukan apakah ada korelasi positif atau negative. Pada langkah ini akan dilakukan perhitungan skor Durbin-Watson menggunakan durbin\_watson( ) fungsi dari statsmodel yang dibuat, kemudian menilainya dengan kondisi sebagai berikut :

1. Jika skor Durbin-Watson kurang dari 1,5 maka terdapat autokorelasi positif dan asumsi tidak terpenuhi.
2. Jika skor Durbin-Watson antara 1,5 – 2,5 maka tidak ada autokorelasi dan asumsi puas.
3. Jika skor Durbin-Watson lebih dari 2,5 maka terdapat autokorelasi negative dan asumsi tidak puas.

|  |  |
| --- | --- |
| [50] | from statsmodels.stats.stattools import durbin\_watson  durbinWatson = durbin\_watson(df\_test\_new2['residual'])  print('Durbin-Watson:', durbinWatson)  if durbinWatson < 1.5:  print('Signs of positive autocorrelation', '\n')  print('Assumption not satisfied')  elif durbinWatson > 2.5:  print('Signs of negative autocorrelation', '\n')  print('Assumption not satisfied')  else:  print('Little to no autocorrelation', '\n')  print('Assumption satisfied') |

|  |  |
| --- | --- |
| [50] | from statsmodels.stats.stattools import durbin\_watson  durbinWatson = durbin\_watson(df\_test\_new2['residual'])  print('Durbin-Watson:', durbinWatson)  if durbinWatson < 1.5:  print('Signs of positive autocorrelation', '\n')  print('Assumption not satisfied')  elif durbinWatson > 2.5:  print('Signs of negative autocorrelation', '\n')  print('Assumption not satisfied')  else:  print('Little to no autocorrelation', '\n')  print('Assumption satisfied') |

Keterangan :

* + Line 50 : Mengimport fungsi durbin\_watson, kemudian fungsi tersebut digunakan untuk mencari nilai autokorelasi dari nilai residualnya.

Didapat hasil perhitungannya adalah 2,160636228. Dapat diasumsikan bahwa terdapat sedikit atau tidak ada autokorelasi, yang berarti asumsi puas.

1. Homoskedastisitas

Dari hasil di atas, ini mengasumsikan homoskedastisitas, yang merupakan varian yang sama dalam istilah kesalahan. Heteroskedastisitas/pelanggaran homoskedastisitas terjadi ketika model tidak memiliki varian genap di seluruh istilah kesalahan. Untuk mendeteksi homoskedastisitas, dapat dilakukan berupa memplot residual dan melihat apakah variansnya tampak seragam.

|  |  |
| --- | --- |
| [51] | # Plotting the residuals  plt.subplots(figsize=(8, 4))  plt.scatter(x=df\_test\_new2.index, y=df\_test\_new2.residual, alpha=0.8)  plt.plot(np.repeat(0, len(df\_test\_new2.index)+2), color='darkorange', linestyle='--')  plt.ylabel('Residual', fontsize=14)  plt.xlabel('Week', fontsize=14)  plt.title('Homescedasticity Assumption', fontsize=16)  plt.show() |



Keterangan :

* + Line 51 : Untuk membuat penyebaran residual pada grafik apakah variansnya seragam atau tidak.

Dari grafik scatterplot di atas, terlihat titik-titik menyebar secara acak, serta tersebar baik di atas maupun di bawah angka 0 (nol) pada sumbu Y. Maka dapat diambil kesimpulan bahwa tidak terdaat gejala heteroskedastisitas pada model regresi yang digunakan.

1. Prediksi Gaji

|  |  |
| --- | --- |
| [52] | # Urutan Inputan : Age, JobLevel (1-5), TotalWorkingYears, YearsAtCompany  salary\_pred = regressor.predict([[20, 1, 3, 1]])  print("Gaji yang terprediksi pada pegawai perbulan yang bekerja sepanjang tahun tersebut adalah ",salary\_pred) |

Keterangan :

* + Line 52 : Menentukan prediksi gaji dengan parameter Age = 20 tahun, JobLevel = 1, TotalWorkingYears = 3 tahun, dan YearsAtCompany = 1 tahun menggunakan metode pada variabel regressor, kemudian simpan di variabel MonthlyIncome. Menampilkan nilai dari variabel MonthlyIncome.

Bab 6

Kesimpulan Dan Saran

## Kesimpulan

Berdasarkan hasil analisis dan pembahasan yang telah dilakukan dapat disimpulkan beberapa poin yaitu :

1. Hasil dari analisis karakteristik data gaji karyawan, didapatkan parameter yang berkaitan dengan gaji karyawan yaitu Age, JobLevel, TotalWorkingYears, dan YearsAtCompany. Parameter-parameter tersebut dipilih berdasarkan pedoman interpretasi koefisien korelasi dari nilai yang sedang hingga sangat kuat.
2. Model yang dibuat berhasil melewati semua pengujian dalam langkah validasi model, sehingga kami dapat menyimpulkan bahwa model kami dapat berperforma baik untuk memprediksi gaji karyawan dengan menggunakan empat variabel independen, yaitu Age, JobLevel, TotalWorkingYears, dan YearsAtCompany. Korelasi yang didapatkan yaitu Age sebesar 0,43; JobLevel sebesar 0,94; TotalWorkingYears sebesar 0,77; dan YearsAtCompany sebesar 0, 48.
3. Model prediksi yang dirancang dengan menggunakan machine learning dengan pendekatan regresi, didapatkan model prediksi gaji karyawan, dimana MonthlyIncome sebagai variabel dependen dan Age, JobLevel, TotalWorkingYears, dan YearsAtCompany variabel independen. Berdasarkan model yang diambil dari model OLS kami mendapatkan nilai akurasi sebesar 0,909. Akurasi tersebut merupakan nilai akurasi yang baik, sehingga dapat dikatakan model machine learning dapat berperforma baik untuk memprediksi gaji.
4. Berdasakan uji validitas, nilai akurasi 0,909 menunjukkan bahwa MonthlyIncome dipengaruhi oleh faktor independen (Age, YearsAtCompany) sebesar 0,909 atau 90,9%. Nilai sisa dari akurasi tersebut adalah 0,091 atau 9,1% yang artinya MonthlyIncome dipengaruhi oleh faktor lain yang tidak diketahui sebesar 9,1%.
5. Visualisasi data dari hasil model prediksi gaji karyawan dapat digunakan menjadi bentuk aplikasi berbasis web base dengan menggunakan framework Django. Dengan aplikasi tersebut, admin dapat melakukan prediksi gaji karyawan dengan mudah dan dengan cepat.

## Saran

Saran yang dapat disampaikan pada peneliti yang akan melanjutkan dan mengembangkan penelitian ini adalah :

1. Pembuatan model prediksi yang digunakan dapat lebih beragam untuk membandingkan performa antara model satu dengan model yang lainnya.
2. Sumber data yang digunakan kurang maksimal. Pada penelitian ini, hanya didapatkan real yang berasal dari Kaggle. Diharapkan kedepannya dapat menggunakan data real langsung dari perusahaan.

## Diskusi

# Daftar Pustaka

# Lampiran - lampiran

# Biodata Penulis

Sinopsis Buku

Hal - hal yang harus ada didalam buku :

Cover

Editorial (penulis/penyunting/UU Hak Cipta

Kata Sambutan

Kata Pengantar

Daftar Isi

Daftar Gambar

Daftar Simbol

Bab 1 Pendahuluan

Bab 2 Landasan Teori

Bab 3 Metodologi Penelitian

Bab 4 Analisis dan hasil pembahasan

Bab 5 Kesimpulan

Daftar Pustaka

Lampiran-lampiran

Sinopsis Buku (pada bagian belakang cover)

Bab 1

Pendidikan Era Digital

**1.1 Pendahuluan**

Seringkali kita mendengar istilah “Revolusi Industri 4.0”, sebuah istilah yang muncul dan dipopulerkan belakangan ini pada saat masyarakat dunia memasuki era milenium baru sejarah peradaban. Banyak orang sebetulnya masih tidak paham apa itu revolusi industri 4.0 dan kenapa mencantumkan angka 4.0 dibelakangnya. Kemudian bagaimana membedakan dengan revolusi pada tingkatan angka-angka sebelumnya? Dan pengaruhnya terhadap dunia pendidikan di Indonesia (Saepudin, 2019).

Secara sederhana, prinsip industri 4.0 menurut Hermann, dkk., (2016) dapat digambarkan sebagai berikut.

**Gambar 1.1:** Prinsip Industri 4.0 (Hermann, dkk., 2016)

## 1.2 Tren Teknologi Pendidikan

Sejak kemunculan komputer, teknologi telah menjadi bagian penting dalam pengembangan tren baru teknologi. Pada tahun 2015, dengan Internet broadband yang tersedia secara luas dan toko aplikasi yang mapan untuk semua jenis perangkat, kemajuan teknologi berjalan terus dan semakin berkembang. Teknologi modern juga telah diterapkan dalam bidang pendidikan dan ruang kelas di seluruh dunia. Selama beberapa dekade proyektor dan papan tulis menjadi semakin canggih, saat ini para guru telah menggunakan materi pendidikan yang lebih berteknologi maju yang mengakibatkan lanskap pendidikan menjadi semakin berubah.

Industri dan publik saat ini mengharapkan dampak besar dari 3D-printing di tahun-tahun mendatang. Sembari perlahan-lahan 3D-printing menjadi tersedia untuk pasar secara massal, 3D-printing membawa peluang pengalaman baru untuk belajar melalui pembuatan prototipe (atau disebut dengan prototyping) dan mendorong peserta didik untuk lebih lagi terlibat dalam proses pembelajaran serta meningkatkan keingintahuan peserta didik untuk berkreasi dan melihat hasilnya dalam bentuk cetak tiga dimensi (3D). Dengan adanya Microsoft HoloLens dan investasi Augmented Reality (AR) secara masif oleh Google, perangkat mobile AR mungkin menjadi teknologi yang akan biasa digunakan di ruang kelas. Fitur visualisasi interaktif mereka membuat AR sangat menarik untuk gamification.



**Gambar 1.2:** Pegawai produsen suku cadang mobil Jerman ZF Fredrichshafen mengenakan kacamata *augmented reality* Hololens (Brigwater, 2018)

Selain itu, kuliah daring terbuka atau Massive Open Online Courses (MOOC) dengan ribuan pembelajar dari seluruh dunia menawarkan kuliah daring gratis melalui internet. Mereka menyediakan interaksi peer-to-peer melalui platform sosial dan popularitas mereka telah sangat meningkat sejak tahun 2011.

## 1.3 Inovasi Teknologi Pendidikan

Perkembangan pesat teknologi digital mendorong kemudahan akses dalam segala bidang. Masyarakat sudah merasakan kemudahan untuk mengakses berbagai keperluan mereka berkat perkembangan pesat internet. Tak terkecuali dalam bidang pendidikan, dengan kemudahan akses internet, sekarang banyak disediakan e-book yang berguna untuk pendidikan dan ilmu pengetahuan.

Contoh pendekatan inovatif dalam pengajaran dan pembelajaran meliputi:

1. Inovasi kelas dan manajemen kelas, termasuk cara mengajar baru yang mengikutkan keterlibatan peserta didik, reorganisasi pembelajaran yang meningkatkan kemampuan peserta didik untuk menerapkan apa yang mereka pelajari, konten belajar yang menjelaskan perubahan historis dalam teori, tugas baru yang mengarah pada peningkatan keterlibatan peserta didik, publikasi peserta didik, dan / atau kegiatan yang menyatukan peserta didik dari berbagai latar belakang.
2. Kepemimpinan dalam inovasi yang membentuk cara baru dan menginspirasi orang lain di dalam dan di luar lembaga, termasuk membimbing kolega tentang pendekatan inovatif, bekerja di posisi administrasi dan layanan untuk memperkenalkan inovasi, berpartisipasi aktif dalam komite untuk mempromosikan atau menciptakan inovasi dan cara lain yang meningkatkan pembelajaran.
3. Memperjuangkan visi baru tentang keunggulan pengajaran melalui pengajaran dan pembelajaran, termasuk kontribusi profesional untuk diskusi, presentasi, buletin, publikasi, dan mode lain untuk berbagi inovasi.

Pada tahun-tahun sebelumnya, para pendidik telah mengadopsi interpretasi yang luas tentang inovasi pengajaran dan pembelajaran dan telah mengakui bahwa pendekatan-pendekatan inovatif juga bisa spesifik untuk disiplin. Suatu pendekatan dalam satu disiplin bisa sangat inovatif, namun pada saat yang sama berpotensi kurang inovatif dalam disiplin lain.

Ada dua konvergensi yang dipublikasikan dengan baik yang dipengaruhi oleh teknologi bergerak:

1. Pertama, konvergensi antara teknologi bergerak sebagai pembelajaran dan desain instruksional, dan pemasaran media komunikasi dan komputasi bergerak digabungkan menjadi satu perangkat. Perangkat ini dapat mengakses Internet, berfungsi sebagai telepon, kamera, video dan pemutar audio dan melakukan tugas-tugas komputasi nirkabel.
2. Kedua, dan sama pentingnya, “konvergensi terjadi antara teknologi personal dan mobile yang baru dengan konsepsi pembelajaran baru sebagai aktivitas seumur hidup yang dikelola secara pribadi” (Sharples dkk., 2007).

**Tabel 1.1:** Konvergensi antara pembelajaran dan teknologi

|  |  |
| --- | --- |
| **Pembelajaran Baru** | **Teknologi Baru** |
| Personalised | Personal |
| Berpusat pada pelajar (learner-centred) | Berpusat pada pengguna (user-centred) |
| Dikondisikan | Bergerak (Mobile) |

**Daftar Isi**

Brigwater, A. (2018) Augmented reality: a vision for the future of manufacturing. Available at: https://www.raconteur.net/manufacturing/augmented-reality-vision-manufacturing (Accessed: 11 July 2019).

Hermann, M., Pentek, T. and Otto, B. (2016) ‘Design Principles for Industrie 4.0 Scenarios’, in 2016 49th Hawaii International Conference on System Sciences (HICSS). IEEE, pp. 3928–3937. doi: 10.1109/HICSS.2016.488.

Holil, M. (2019) Pendidikan Era Revolusi Industri 4.0 oleh MOH HOLIL Halaman all - Kompasiana.com. Available at: https://www.kompasiana.com/holsthea/5c680a2dab12ae76bf4a33e5/pendidikan-era-revolusi-industri-4-o?page=all (Accessed: 21 July 2019).

Malamed, C. (2016) 10 Definitions of Learning. Available at: http://theelearningcoach.com/learning/10-definitions-learning/ (Accessed: 7 August 2019).

Saepudin (2019) REVOLUSI INDUSTRI 4.0 , APAKAH ITU? DAN PENGARUHNYA TERHADAP DUNIA PENDIDIKAN – Dinas Pendidikan Kabupaten Bandung Barat. Available at: http://disdikkbb.org/?news=revolusi-industri-4-0-apakah-itu-dan-pengaruhnya-terhadap-dunia-pendidikan (Accessed: 1 August 2019).

Simarmata, J. et al. (2019) Tren dan Aplikasi: Strategi dan Inovasi dalam Pembelajaran. Denpasar: Jayapangus Press.

-oo00oo-

Daftar isi menggunakan References Management (Mendeley, EndNote, Zotero dll, dengan style HARVARD)