**PEMBANGUNAN APLIKASI MONITORING TANAMAN MEMANFAATKAN API DARI ALAT SIRAM TANAMAN OTOMATIS**

**SKRIPSI**

Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana (S1)

**Prayogi Sukmana**

**10118165**



**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS TEKNIK DAN ILMU KOMPUTER**

**UNIVERSITAS KOMPUTER INDONESIA**

**2022**

DAFTAR ISI

[BAB 1 PENDAHULUAN 6](#_Toc111049974)

[1.1. Latar Belakang Masalah 6](#_Toc111049975)

[1.2. Identifikasi Masalah 7](#_Toc111049976)

[1.3. Maksud dan Tujuan 8](#_Toc111049977)

[1.4. Batasan Masalah 8](#_Toc111049978)

[1.5. Metodologi Penelitian 9](#_Toc111049979)

[1.5.1. Pengumpulan Data 10](#_Toc111049980)

[1.5.2. Analisis Masalah 10](#_Toc111049981)

[1.5.3. Pembangunan Perangkat Lunak 10](#_Toc111049982)

[1.6. Sistematika Penulisan 12](#_Toc111049983)

[BAB 2 TINJAUAN PUSTAKA 14](#_Toc111049984)

[2.1. Tanaman 14](#_Toc111049985)

[2.1.1. Tanaman Kaktus 14](#_Toc111049986)

[2.1.2. Tanaman Sukulen 15](#_Toc111049987)

[2.2. Vue JS 16](#_Toc111049988)

[2.3. Laravel 16](#_Toc111049989)

[2.4. Vue-Router 17](#_Toc111049990)

[2.5. Axios 17](#_Toc111049991)

[2.6. MySQL 17](#_Toc111049992)

[2.7. API Alat Siram Tanaman Otomatis 18](#_Toc111049993)

[2.8. MQTT 18](#_Toc111049994)

[BAB 3 ANALISIS DAN PERANCANGAN 19](#_Toc111049995)

[3.1. Analisis 19](#_Toc111049996)

[3.1.1 Analisis Masalah 19](#_Toc111049997)

[3.1.2 Analisis Proses Bisnis 19](#_Toc111049998)

[3.1.3 Analisis Arsitektur Sistem 22](#_Toc111049999)

[3.1.4 Analisis Teknologi Yang Digunakan 23](#_Toc111050000)

[3.1.5 Analisis SKPL 26](#_Toc111050001)

[3.1.6 Analisis Kebutuhan Non-Fungsional 27](#_Toc111050002)

[3.1.7 Analisis Kebutuhan Fungsional 32](#_Toc111050003)

[3.1.8 Analisis Basis Data 64](#_Toc111050004)

[3.2. Perancangan 66](#_Toc111050005)

[3.2.1 Perancangan Menu 66](#_Toc111050006)

[3.2.2 Perancangan Basis Data 67](#_Toc111050007)

[3.2.3 Perancangan Antarmuka 69](#_Toc111050008)

[3.2.4 Perancangan Pesan 73](#_Toc111050009)

[BAB IV IMPLEMENTASI DAN PENGUJIAN 76](#_Toc111050010)

[4.1. Implementasi 76](#_Toc111050011)

[4.2.1 Perangkat Lunak Peneliti 76](#_Toc111050012)

[4.2.2 Perangkat Keras Peneliti 76](#_Toc111050013)

[4.2.3 Implementasi Antarmuka 77](#_Toc111050014)

[4.2.4 Implementasi Kelas 77](#_Toc111050015)

[4.2.5 Implementasi Basis Data 78](#_Toc111050016)

[4.2.6 Implementasi Teknologi 79](#_Toc111050017)

[4.2 Pengujian 85](#_Toc111050018)

[4.2.1 Rencana Pengujian BlackBox 85](#_Toc111050019)

[4.2.2 Hasil Pengujian 86](#_Toc111050020)

[BAB V KESIMPULAN DAN SARAN 90](#_Toc111050021)

[5.1. Kesimpulan 90](#_Toc111050022)

[5.2. Saran 90](#_Toc111050023)

[DAFTAR PUSTAKA 91](#_Toc111050024)

# DAFTAR GAMBAR

[Gambar 1. 1 Metode Penelitian 9](file:///G:\Semester8\SKRIPSI\Revisi%202-8-2022\10118165_PrayogiSukmana_SKRIPSI.docx#_Toc111977238)

[Gambar 1. 2 Metode Waterfall 10](file:///G:\Semester8\SKRIPSI\Revisi%202-8-2022\10118165_PrayogiSukmana_SKRIPSI.docx#_Toc111977239)

[Gambar 3. 1 Analisis Arsitektur Desain Sistem 23](#_Toc111977243)

[Gambar 3. 2 Skrip import Axios ke sistem 24](#_Toc111977244)

[Gambar 3. 3 Skrip URL dari API yang digunakan 25](#_Toc111977245)

[Gambar 3. 4 Skrip FIeld yang akan ditampilkan 25](#_Toc111977246)

[Gambar 3. 5 Skrip deklarasi field 25](#_Toc111977247)

[Gambar 3. 6 Skrip Axios untuk mengambil data dari API 26](#_Toc111977248)

[Gambar 3. 7 Use Case Diagram 34](#_Toc111977249)

[Gambar 3. 8 Activity Diagram UC-01 51](#_Toc111977250)

[Gambar 3. 9 Activity Diagram UC-02 53](#_Toc111977251)

[Gambar 3. 10 Activity Diagram UC-03 55](#_Toc111977252)

[Gambar 3. 11 Activity Diagram UC-04 56](#_Toc111977253)

[Gambar 3. 12 Activity Diagram UC-05 57](#_Toc111977254)

[Gambar 3. 13 Activity Diagram UC-06 59](#_Toc111977255)

[Gambar 3. 14 Activity Diagram UC-07 60](#_Toc111977256)

[Gambar 3. 15 Activity Diagram UC-08 61](#_Toc111977257)

[Gambar 3. 16 Activity Diagram UC-09 62](#_Toc111977258)

[Gambar 3. 17 Activity Diagram UC-10 63](#_Toc111977259)

[Gambar 3. 18 Activity Diagram UC-11 64](#_Toc111977260)

[Gambar 3. 19 Activity Diagram UC-12 65](#_Toc111977261)

[Gambar 3. 20 Activity Diagram UC-13 66](#_Toc111977262)

[Gambar 3. 21 Activity Diagram UC-14 67](#_Toc111977263)

[Gambar 3. 22 Class Diagram 68](#_Toc111977264)

[Gambar 3. 23 Sequence Diagram UC-01 69](#_Toc111977265)

[Gambar 3. 24 Sequence Diagram UC-02 70](#_Toc111977266)

[Gambar 3. 25 Sequence Diagram UC-03 70](#_Toc111977267)

[Gambar 3. 26 Sequence Diagram UC-04 71](#_Toc111977268)

[Gambar 3. 27 Sequence Diagram UC-05 71](#_Toc111977269)

[Gambar 3. 28 Sequence Diagram UC-06 72](#_Toc111977270)

[Gambar 3. 29 Sequence Diagram UC-07 72](#_Toc111977271)

[Gambar 3. 30 Sequence Diagram UC-08 73](#_Toc111977272)

[Gambar 3. 31 Sequence Diagram UC-09 73](#_Toc111977273)

[Gambar 3. 32 Sequence Diagram UC-10 74](#_Toc111977274)

[Gambar 3. 33 Sequence Diagram UC-11 74](#_Toc111977275)

[Gambar 3. 34 Sequence Diagram UC-12 75](#_Toc111977276)

[Gambar 3. 35 Sequence Diagram UC-13 75](#_Toc111977277)

[Gambar 3. 36 Sequence Diagram UC-14 76](#_Toc111977278)

[Gambar 3. 37 Entity Relationship Diagram 77](#_Toc111977279)

[Gambar 3. 38 Perancangan Menu 79](#_Toc111977280)

[Gambar 3. 39 Skema Tabel Relasional 80](#_Toc111977281)

[Gambar 3. 40 Tampilan Antarmuka Halaman Utama 83](#_Toc111977282)

[Gambar 3. 41 Tampilan Antarmuka Halaman Detail 84](#_Toc111977283)

[Gambar 3. 42 Tampilan Antarmuka Halaman Tambah Tanaman 85](#_Toc111977284)

[Gambar 3. 43 Perancangan Pesan Kesalahan Username / Password 86](#_Toc111977285)

[Gambar 3. 44 Perancangan Pesan Login Berhasil 86](#_Toc111977286)

[Gambar 3. 45 Perancangan Pesan Ketika Tanah Kering 86](#_Toc111977287)

[Gambar 3. 46 Perancangan Pesan Ketika Suhu Dingin 87](#_Toc111977288)

[Gambar 3. 47 Perancangan Pesan Tanmbah Tanaman 87](#_Toc111977289)

[Gambar 3. 48 Perancangan Pesan Ubah Tanaman 87](#_Toc111977290)

[Gambar 3. 49 Perancangan Pesan Hapus Tanaman 88](#_Toc111977291)

[Gambar 3. 50 Perancangan Pesan Cahaya Kurang 88](#_Toc111977292)

# DAFTAR TABEL

[Tabel 3. 1 SKPL Fungsional 27](#_Toc111974845)

[Tabel 3. 2 SKPL Non Fungsional 28](#_Toc111974846)

[Tabel 3. 3 Analisis Kebutuhan Perangkat Lunak 29](#_Toc111974847)

[Tabel 3. 4 Fakta Spesifikasi Perangkat Lunak Peneliti 29](#_Toc111974848)

[Tabel 3. 5 Analisis Kebutuhan Perangkat Keras 30](#_Toc111974849)

[Tabel 3. 6 Fakta Spesifikasi Perangkat Keras Peneliti 30](#_Toc111974850)

[Tabel 3. 7 Analisis Kebutuhan Perangkat Pikir 31](#_Toc111974851)

[Tabel 3. 8 Definisi Aktor 34](#_Toc111974852)

[Tabel 3. 9 Definisi Use Case 35](#_Toc111974853)

[Tabel 3. 10 Use Case Scenario UC-01 40](#_Toc111974854)

[Tabel 3. 11 Use Case Scenario UC-02 41](#_Toc111974855)

[Tabel 3. 12 Use Case Scenario UC-03 41](#_Toc111974856)

[Tabel 3. 13 Use Case Scenario UC-04 42](#_Toc111974857)

[Tabel 3. 14 Use Case Scenario UC-05 43](#_Toc111974858)

[Tabel 3. 15 Use Case Scenario UC-06 44](#_Toc111974859)

[Tabel 3. 16 Use Case Scenario UC-07 44](#_Toc111974860)

[Tabel 3. 17 Use Case Scenario UC-08 45](#_Toc111974861)

[Tabel 3. 18 Use Case Scenario UC-09 46](#_Toc111974862)

[Tabel 3. 19 Atribut Entity Relationship Diagram 75](#_Toc111974863)

[Tabel 3. 20 Struktur Tabel Kotak Tanaman 78](#_Toc111974864)

[Tabel 3. 21 Struktur Tabel Suhu 78](#_Toc111974865)

[Tabel 3. 22 Struktur Tabel Cahayas 79](#_Toc111974866)

[Tabel 3. 23 Struktur Tabel Kelembapans 79](#_Toc111974867)

[Tabel 3. 24 Struktur Tabel Users 79](#_Toc111974868)

[Tabel 4. 1 Perangkat Lunak Peneliti 87](#_Toc111974869)

[Tabel 4. 2 Perangkat Keras Peneliti 87](#_Toc111974870)

[Tabel 4. 3 Implementasi Antarmuka 88](#_Toc111974871)

[Tabel 4. 4 Implementasi Kelas 88](#_Toc111974872)

[Tabel 4. 5 Implementasi Basis Data 89](#_Toc111974873)

# ABSTRAK

Penelitian ini bertujuan untuk membantu pengelola *greenhouse* Cactus ThreeD dalam mengontrol tanaman yang kemudian dapat melakukan penyiraman tanaman, menyalakan *heater* dan menyalakan lampu untuk kotak tanaman secara otomatis, maupun manual melalui sistem dan membantu pengelola *greenhouse* dalam memperoleh informasi kondisi jumlah kelembaban, intensitas cahaya, dan suhu pada media tanaman secara langsung. Tanaman hias dikenal dengan keindahannya yang memiliki nilai estetis dan nilai seni. Tidak hanya nilai estetis dan nilai seni, tanaman hias juga berfungsi sebagai stabilisator dan pemeliharaan lingkungan, pendidikan, pemelihara kesehatan, serta ekonomi dan sosial. Agar fungsi dari tanaman hias tersebut dapat maksimal maka diperlukan perawatan yang rutin. Hal yang harus diperhatikan dalam merawat tanaman hias ini diantaranya adalah suhu, intensitas cahaya, dan juga kelembapan tanah. Penelitian ini menggunakan metode waterfall dalam pembangunan sistemnya. Berdasarkan penelitian *greenhouse* Cactus ThreeD masih menggunakan cara perawatan yang manual pada kotak tanaman dan belum adanya informasi terkait suhu, intensitas cahaya dan kelembapan tanah pada tanaman hias. Dengan adanya Sistem Monitoring Tanaman Memanfaatkan API Alat Siram Tanaman Otomatis dapat memudahkan *greenhouse* dalam melakukan perawatan tanaman hias secara mudah dan mendapatkan hasil tanaman hias yang baik dan indah. Selain itu walaupun alat ini dapat memudahkan pengelola *greenhouse*, namun dalam penggunaannya, alat ini perlu adanya kalibrasi jika ada penambahan kotak tanaman yang baru.

Kata Kunci : Sistem Monitoring Tanaman, Siram Tanaman Otomatis, Tanaman Hias

# *ABSTRACT*

*This research aim to help Cactus ThreeD greenhouse owner in controlling plants which can then water plants, turn on heaters and turn on lights for plants section automatically, or manually through the system and help greenhouse owner in obtaining information on conditions of soil moisture, light intensity, and temperature directly on plant media. Ornamental plants are known for their beauty which has aesthetic value and artistic value. Not only aesthetic value and artistic value, ornamental plants also function as a stabilizer and maintenance of the environment, education, health care, as well as economic and social. In order for the function of these ornamental plants to be maximized, regular maintenance is needed. Things that must be considered in caring for these ornamental plants include temperature, light intensity, and soil moisture. This research using the waterfall method in the development of the system. Based on research, the Cactus ThreeD greenhouse still uses manual maintenance methods on plants section and there is no information regarding temperature, light intensity and soil moisture in ornamental plants. The existence of a Plant Monitoring System Utilizing API Automatic Plant Watering Tool can make it easier for greenhouses to carry out ornamental plant care easily and get good and beautiful ornamental plant results. In addition, although this tool can make it easier for greenhouse owner, in its use, this tool needs calibration if there is a new plants section added*

*Keywords : Plant Monitoring System, Automatic Plant Watering, Ornamental Plants*

# BAB 1 PENDAHULUAN

## Latar Belakang Masalah

Setiap rumah memiliki pekarangan sebagai penunjang keindahan rumah tersebut. Biasanya pekarangan rumah dipenuhi oleh tanaman, baik tanaman hias, tanaman obat maupun sayur-sayuran. Jenis tanaman yang ditanam tersebut berskala kecil dalam bentuk pot. Namun bagi rumah yang tidak memiliki pekarangan namun ingin tetap menanam tanaman, bisa menanam tanaman hias dalam skala yang lebih kecil lagi.

Tanaman merupakan salah satu makhluk hidup yang banyak memiliki manfaat untuk lingkungan sekitar. Berbagai macam manfaat dapat didapatkan dari sebuah tanaman, yaitu sebagai obat herbal, bahan makanan, dan yang paling utama adalah tanaman dapat menghasilkan oksigen yang berguna sebagai kelangsungan hidup manusia. Banyak sekali jenis-jenis tanaman, dari mulai bentuk fisiknya yang bermacam-macam, hingga manfaatnya yang berbeda-beda. Dari banyaknya tanaman tersebut tanaman dibagi menjadi beberapa kelompok. Salah satunya adalah kelompok tanaman hias.

Tanaman Hias atau Ornamental Plants diartikan sebagai semua tanaman yang memiliki nilai hias dari mulai bagian bunga, akar, daun, aroma dan batangnya sendiri, dan tanaman tersebut memiliki nilai estetis dan nilai seni [10]. Jadi dapat dikatakan bahwa segala tanaman yang memiliki fungsi sebagai keindahan dapat digolongkan menjadi kelompok tanaman hias. Tanaman hias sendiri memiliki jenis yang sangat banyak sekali diluar sana, dengan kondisi tersebut manfaat tanaman hias pun semakin meluas. Adapun fungsi lain tanaman hias selain fungsi estetis atau keindahan. Diantaranya adalah tanaman hias dapat berfungsi sebagai stabilisator dan pemeliharaan lingkungan, pendidikan, pemelihara kesehatan, serta ekonomi dan sosial (Rukmana, 2012, h.11).

Beberapa faktor penunjang tanaman agar dapat tumbuh dengan sempurna, yaitu suhu, intensitas cahaya, dan juga kelembapan tanah. Apabila suhu dari lingkungan tanaman melebihi batas optimal atau kurang dari batas optimal, maka akan menyebabkan tumbuhan tidak tumbuh secara optimal. Begitu juga apabila kelembapan tanah dan intensitas cahaya yang tidak mencukupi kebutuhan tanaman tersebut, maka tanaman tidak tumbuh secara optimal pula.

Pada Cactus ThreeD menyediakan tanaman hias berupa kaktus dan sukulen, berdasarkan hasil wawancara dan observasi, tanaman hias yang berada di Cactus ThreeD masih banyak yang tidak tumbuh secara optimal, dikarenakan lokasi dari greenhouse tersebut tidak mendapatkan factor penunjang tanaman yang optimal, yang menyebabkan kebanyakan tanaman hias yang berada disana menjadi busuk

Dari permasalahan diatas, maka dibutuhkan suatu solusi yang dapat membantu pihak pengelola dalam pengoptimalan pertumbuhan tanaman dan memantau kondisi tanaman yang mereka tanam.

## Identifikasi Masalah

Berdasarkan latar belakang di atas, maka didapatkan masalah yang dapat diidentifikasi, yaitu adalah sebagai berikut :

1. Tanaman harus selalu dijaga dan dipantau kondisi nya, salah satu cara nya dengan menyiram tanaman tersebut. Namun dalam *greenhouse* tanaman dikelompokan dalam kotak, sehingga dalam penyiraman terkadang tidak ada beberapa tanaman di dalam kotak yang tidak tersiram.
2. Selain dipantau kondisi melalui penyiraman tanaman, tanaman juga dipantau berdasarkan suhu. Namun masih banyak orang yang hanya memantau kondisi dengan menyiram saja, tidak memperhatikan tanaman berdasarkan kondisi suhu.
3. Selain dipantau kondisi berdasarkan suhu, tanaman juga dipantau berdasarkan intensitas cahaya.

## Maksud dan Tujuan

1. Maksud

Maksud dari penelitian ini adalah untuk membangun sistem atau aplikasi *monitoring* padatanaman dalam proses pertumbuhan berdasarkan suhu, kelembapan tanah dan intensitas cahaya, yang dimana sistem ini dapat melakukan penyiraman tanaman, menyalakan lampu dan menyalakan *element heater* secara otomatis, dimana nanti sistem ini juga dihubungkan dengan alat untuk menyiram tanaman sehingga pengguna dapat menyiram tanaman, menyalakan lampu, dan menyalakan *element heater* melalui sistem.

1. Tujuan

Tujuan dari penelitian akan dijabarkan sebagai berikut :

1. Membantu pengelola greenhouse Cactus ThreeD dalam mengontrol pertumbuhan tanaman kaktus
2. Membantu pengelola greenhouse dalam proses penyiraman secara otomatis pada kotak media tanam tanaman kaktus agar mengurangi keterlambatanan dalam penyiraman
3. Membantu pengelola greenhouse dalam proses pemanasan kotak tanaman untuk mencapai suhu yang optimal dalam kotak tanaman
4. Membantu pengelola greenhouse dalam proses pencahayaan kotak tanaman agar kebutuhan tanaman terhadap cahaya terpenuhi
5. Membantu pengelola greenhouse memperoleh informasi kondisi jumlah kelembaban, intensitas cahaya, dan suhu sehingga kondisi media tanam tanaman kaktus bisa sesuai dengan kondisi optimal.

## Batasan Masalah

Adapun batasan dalam penelitian ini, yaitu adalah sebagai berikut :

1. Sistem yang dibangun untuk monitoring kelembaban tanah, intensitas cahaya, dan suhu pada media tanam greenhouse
2. Jumlah kelembaban yang dijadikan standar adalah berkisar 50%
3. Jumlah suhu yang dijadikan standar adalah 26˚C hingga 32˚C
4. Sistem digunakan pada media tanam untuk tanaman kaktus dan sukulen
5. Sistem hanya dapat memonitoring kondisi jumlah kelembaban, suhu dan intensitas cahaya pada kotak media tanam budidaya tanaman kaktus dan tanaman sukulen
6. Sistem yang dibangun menggunakan koneksi internet agar dapat memonitoring secara *real-time*.
7. API yang digunakan terdapat data – data tanaman, yang meliputi
8. Data Kelembapan Tanah
9. Data Intensitas Cahaya
10. Data Suhu
11. Sistem yang dibangun hanya memberi informasi kondisi kelembaban, intensitas cahaya dan suhu bebasis website

## Metodologi Penelitian

Metodologi Penelitian yang akan dilaksanakan merupakan sebuah rangkaian tahapan yang disusun secara sistematis, berikut adalah gambaran dari metode penelitian yang akan berjalan.

Gambar 1. 1 Metode Penelitian

Dari *flow map* di atas dapat didefinisikan sebagai berikut :

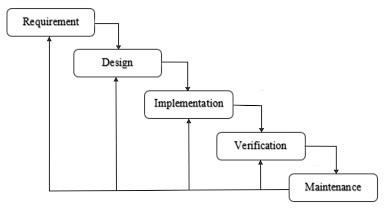
### Pengumpulan Data

Tahapan pertama pada metode penelitian ini adalah pengumpulan informasi. Untuk pengumpulan informasi, akan melalui berbagai cara seperti mencari informasi melalui media buku, atau jurnal. Bahkan peneliti akan melakukan wawancara dan melakukan penelitian di Greenhouse Cactus ThreeD untuk mengumpulkan informasi yang mungkin tidak bisa didapatkan selain dengan mencari informasi melalui buku atau jurnal.

### Analisis Masalah

Selanjutnya adalah tahapan analisis masalah. Sebelum membuat ide, hasil informasi yang didapat dari pengumpulan perlu dianalisis terlebih dahulu.

### Pembangunan Perangkat Lunak

Pada tahapan ini, metode dalam pembangunan perangkat lunak yang digunakan adalah metode Waterfall. Karena metode ini cocok digunakan pada penelitian yang skalanya kecil dan juga memiliki tahapan-tahapan yang sederhana. Ada empat tahapan dalam metode ini yaitu Analisis kebutuhan, Desain, Implementasi dan Pengujian.

Gambar 1. 2 Metode Waterfall

Dari gambar metode Waterfall di atas maka prosedur *waterfall* yang diterapkan pada Aplikasi ini sebagai berikut :

1. Analisis Kebutuhan, merupakan tahapan paling awal yang ada dalam metode *waterfall*. Pada tahapan ini diperoleh informasi-informasi yang dibutuhkan untuk keperluan penelitian akan didapatkan melalui media buku, atau jurnal. Bahkan peneliti akan melakukan wawancara dan melakukan penelitian di Cactus ThreeD untuk mengumpulkan informasi yang mungkin tidak bisa didapatkan selain dengan mencari informasi melalui buku atau jurnal.
2. Desain Sistem, tahapan ini mengambil masukan-masukan dari informasi yang telah didapatkan dari tahapan analisis kebutuhaan yang sebelumnya dilakukan. Tahapan ini merupakan perencanaan solusi perangkat lunak yang bisa mencakup desain sistem dan kebutuhan fungsional atau non-fungsionalnya. Penulis menggunakan perangkat keras yaitu laptop. Sedangkat *software*nya penulis menggunakan Bahasa pemrograman Vue.js untuk bagian desain antarmuka, dan Bahasa pemrograman Laravel untuk bagian desain sistem, IDE akan menggunakan Visual Studio Code, sedangkan desain prototype menggunakan aplikasi Figma.
3. Implementasi, pada tahap inilah pengembangan aktual sistem terjadi sesuai dengan spesifikasi desain. Hasil dari langkah ini adalah *prototype* sistem yang dibangun dengan harapan dapat memenuhi semua kebutuhan fungsionalnya.
4. Pengujian, pada tahap ini sistem yang dilakukan oleh penguji terlebih dahulu. Pada tahapan pengujian, peneliti akan menjalanakan sistem yang telah terhubung dengan alat siram otomatis.
5. Maintenance, tahap dimana suatu perangkat lunak yang sudah selesai dapat mengalami perubahan - perubahan atau penambahan sesuai dengan kritik dan saran yang diberikan. Pada aplikasi ini, nanti jika aplikasi telah dirilis dan ada beberapa fitur yang harus dikembangkan maka sangat diperlukan proses *Maintenance*

## Sistematika Penulisan

Untuk memudahkan penulisan dalam penyusunan skripsi ke arah yang dimaksud, maka digunakan sistematika penulisan yang nantinya akan mempermudah penulisan skripsi, meliputi :

1. BAB 1 - Pendahuluan

Bab ini berisi tentang latar belakang, rumusan masalah, maksud dan tujuan, manfaat, batasan masalah, metodologi penelitian dan sistematika penulisan. Singkatnya bab ini menjelaskan permasalahan yang sedang terjadi dan menjelaskan solusi untuk menyelesaikan masalah tersebut. Terlihat Seperti yang telah ditulis diatas Bab 1 merupakan sebuah uraian tentang latar belakang masalah terbuatnya sistem monitoring tanaman, maksud dan tujuan dibuatnya sistem tersebut, batasan masalah yang digunakan dalam proses pembuatan sistem ini, dan sistematika penulisan laporan ini.

1. BAB 2 – Tinjauan Pustaka

Bab ini membahas teori-teori pendukung yang digunakan dalam proses perencanaan dan proses pembuatan pada penelitian, seperti landasan teori tentang tanaman, tanaman hias, Vue JSdan yang lainnya yang berhubungan dengan penelitian ini

1. BAB 3 – Analisis dan Perancangan Sistem

Bab ini berisi tentang analisis, dan perancangan sistem pada penelitian. Untuk penelitian ini, analisis akan menggunakan tool tambahan berupa UML atau Unified Modelling Language. Bab 3 ini bertujuan untuk memudahkan pembaca atau peneliti selanjutnya dalam memahami fungsionalitas apa saja yang dimiliki oleh sistem.

1. BAB 4 – Implementasi dan Pengujian Sistem

Bab ini berisi tentang implementasi dan juga pengujian sistem. Pengujian yang akan dilakukan adalah pengujian secara black box. Selain itu, bab ini juga akan membahas tentang hasil pengujian yang dilakukan langsung oleh pengguna.

1. BAB 5 – Kesimpulan dan Saran

Bab ini berisi kesimpulan yang diperoleh selama melakukan penelitian dan menghasilkan saran-saran untuk memperbaiki sistem di penelitian berikutnya apabila ada yang melanjutkan

# BAB 2 TINJAUAN PUSTAKA

## Tanaman

Tanaman merupakan salah satu makhluk hidup yang banyak memiliki manfaat untuk lingkungan sekitar. Berbagai macam manfaat dapat didapatkan dari sebuah tanaman, yaitu sebagai obat herbal, bahan makanan, dan yang paling utama adalah tanaman dapat menghasilkan oksigen yang berguna sebagai kelangsungan hidup manusia. Banyak sekali jenis-jenis tanaman, dari mulai bentuk fisiknya yang bermacam-macam, hingga manfaatnya yang berbeda-beda. Dari banyaknya tanaman tersebut tanaman dibagi menjadi beberapa kelompok. Salah satunya adalah kelompok tanaman hias.

### Tanaman Kaktus

Tumbuhan kaktus menjadi salah satu tanaman yang cukup terkenal di masyarakat. Tidak hanya di negara asalnya, melainkan di berbagai macam negara mengetahui tanaman kaktus. Mendengar kata kaktus orang akan mengingat sebuah tumbuhan yang memiliki duri pada batang tubuhnya. Berikut adalah pandangan penulis mengenai tumbuhan kaktus, yakni, Kaktus adalah tumbuhan yang termasuk kedalam keluarga Cactaceae yang memiliki lebih dari 2000 varietas dengan berbagai warna dan bentuk.

Kaktus dikenal sebagai tumbuhan sukulen karena bagian tubuhnya seperti batang, akar, dan daun mampu menyimpan air untuk bertahan hidup. Kaktus adalah tanaman yang memiliki bentuk dan cara bertahan hidup yang unik dan berbeda dari jenis tanaman lain. Kaktus dikenal memiliki duri yang tak lain adalah daunnya. Daun kaktus tumbuh dari struktur khusus yang disebut areole. Ukuran duri bervariasi dan bisa mencapai panjang 15 cm. Daun yang berubah menjadi duri bertujuan untuk mencegah penguapan air berlebih. Sebagian duri kaktus bahkan berwarna terang yang membantu mereka memantulkan sinar matahari sehingga tanaman tetap dingin.

Tumbuhan kaktus memiliki bunga yang sangat indah. Pertumbuhan bunga kaktus terbilang lambat. Butuh waktu sekitar 2 tahun untuk bunga kaktus dapat mekar sempurna. Beberapa bunga kaktus dapat bertahan hidup selama beberapa hari setelah mekar, adapula yang hanya bertahan hidup selama sehari setelah mekar. Keadaan waktu yang dibutuhkan bunga untuk mekarpun berbeda tergantung jenis kakstunya. Sebagian bunga hanya dapat mekar pada saat malam hari dan sebagian lain hanya dapat mekar pada saat siang hari.

Keunikan terjadi pula pada akar kaktus. Tanaman akar yang sangat panjang. Kedalaman akar kaktus tidak lebih dari 10 cm berada di bawah permukaan tanah. Namun, akar kaktus dapat menyebar ke sekeliling hingga meliputi diameter 2 meter. Hal tersebut dilakukan untuk mencari sumber air tanah yang berada di wilayah sekitar kaktus berada. Kaktus merupakan tanaman dengan cara bertahan hidup yang luar biasa. Kaktus mampu hidup di daerah kering seprti gurun pasir, semi gurun dan padang rumput. hidup daerah karena memiliki beberapa kemampuan unik untuk kaktus mampu bertahan hidup dengan menyesuaikan keadaan di lingkungan sekitarnya dengan jumlah air yang sedikit. Kemampuan tersebut menjadikan tumbuhan kaktus istimewa, unik dan amat berbeda dengan jenis tanaman lain.

### Tanaman Sukulen

Sukulen adalah istilah umum untuk tumbuhan yang dapat menyimpan air di dalamnya. daun. Sukulen atau succulents berasal dari kata latin “Sukkot”. Berarti jus. Dalam bahasa Inggris artinya Berair. Berair Ia memiliki kemampuan untuk menyimpan air di batang dan pulp (Coyne & Knutzen, 2014, hal.2). Di sisi lain, menurut Sueno, sukulen ada di Jati (2016). Tumbuhan dengan batang berdaging tetapi daunnya tebal Itu bukan duri, tapi masih berbentuk seperti daun. Daun sukulen juga berbentuk Lemak dan cair (hal.8). Daun kaya air Tanaman sukulen ini berfungsi sebagai adaptasi terhadap iklim kering Tanaman ini dapat bertahan lama dalam keadaan kering, Itu bisa menahan penguapan. Namun, ada beberapa jenis sukulen Ini dapat menahan kondisi kering dan dingin seperti yucca, houseleek, sedum, dan beberapa agave dan kaktus.

## Vue JS

Vue JS merupakan sebuah kerangka kerja berbasis Javascript yang digunakan untuk membangun tampilan antarmuka pada aplikasi web yang dikembangkan oleh Evan You pada tahun 2013.

Dalam penggunaannya, Vue JS dapat diintregasikan dengan kerangka kerja atau *library* Javascript yang lainnya.

Vue ini dapat menjalankan aplikasi ­*Single Page Application* yang canggih, dimana apabila pengguna melakukan pergantian halaman, maka *browser* yang digunakan tidak akan melakukan *refresh* ketika membuka halaman yang baru, sehingga akan terasa lebih cepat dalam penggunaan web.

Dalam pembangunan aplikasi monitoring tanaman ini menggunakan framework VueJS karena dalam penggunannya menggunakan teknologi *Single Page Application* yang dapat melakukan perubahan tanpa harus melakukan *refresh* pada setiap halaman

## Laravel

Laravel merupakan *framework* dari Bahasa pemrograman PHP yang bersifat *open-souce* menggunakan konsep *Model-View-Controller*(MVC) yang dikembangkan pada tahun 2011 oleh Taylor Otwell.

Laravel diciptakan unutk menyederhanakan proses pengembangan perangkat lunak, dimana dalam pembuatan suatu aplikasi menggunakan Laravel, semua basic nya sudah disediakan dalam dokumentasi Laravel, contoh nya jika ingin membuat *Controller*, kita cukup memasukkan syntax pada command-prompt, maka otomatis laravel akan membuatkannya.

## Vue-Router

Vue-Router merupakan sebuah teknologi yang digunakan dari framework Vue JS agar dapat melakukan perpindahan halaman dengan teknologi *Single Page Application*, jadi ketika pengguna website memilih menu halaman lain, website tidak harus melakukan *refresh* untuk mengakses halaman yang dipilih oleh pengguna

## Axios

Axios merupakan sebuah *library* yang bersifat terbuka (*open-source)* yang digunakan untuk mengambil atau menampilkan data dari sebuah API. *Library* Axios ini biasa digunakan dalam beberapa Bahasa pemrograman, seperti Vue JS, Axios ini lebih popular dalam pendekatan untuk permintaan atau pengambilan data dari sebuah API.

Axios digunakan dalam pemanggilan data backend dari Laravel, data yang dipanggil pada backend adalah data tanaman yang terdapat pada halaman home, dan juga digunakan dalam pemanggilan data data tanaman dari API Alat Siram Tanaman Otomatis

## MySQL

MySQL dapat dikategorikan perangkat lunak yang membuat database bersifat berbuka (open source). MySQL adalah sistem yang berguna untuk melakukan proses pengaturan koleksi-koleksi struktur data (database) baik meliputi proses pembuatan atau proses pengelolaan database [8]. MySQL banyak digunakan pada kalangan promgrammer karena open source dan mudah dipahami. MySQL bisa disebut perangkat lunak yang membuat database sebagai sumber dari pengolahan data untuk membangun aplikasi ataupun website.

MySQL digunakan dalam menyimpan data tanaman, data kelembapan tanah, intensitas cahaya, dan suhu yang telah didapat dari API yang mengirimkan data ke sistem

## API Alat Siram Tanaman Otomatis

API Alat Siram Tanaman Otomatis merupakan API yang dibuat oleh peneliti untuk digunakan dalam sistem ini agar dapat memanfaatkan *Service* pada aplikasi, API Alat Siram Tanaman Otomatis ini digunakan oleh sistem untuk menampilkan data - data dari suatu tanaman, seperti data kelembapan tanah, suhu dan intensitas cahaya, juga untuk menyalakan dan mematikan kran yg terdapat pada tanaman.

## MQTT

MQTT merupakan sebuah protokol yang diterapkan pada IOT. Protokol ini sangat mendukung untuk jaringan WAN, karena WAN mencakup area yang luas. Protokol MQTT mempunyai kelebihan yaitu dapat bekerja dengan energi dan media penyimpanan yang minimum.

MQTT digunakan dalam penghubung sistem dengan alat, teknologi yang digunakan dalam penghubungan sistem dengan alat ini adalah menggunakan WebSocket, cara penggunaan WebSocket ini dengan memanggil url host, port dan endpoint dari alat, MQTT

MQTT digunakan dalam pembuatan API Alat Siram Tanaman Otomatis, MQTT bekerja sebagai wadah untuk mengirimkan data – data dari alat ke API

# BAB 3 ANALISIS DAN PERANCANGAN

## Analisis

Tahapan analisis bertujuan untuk menganalisis permasalahan-permasalahan yang ada serta menentukan kebutuhan apa saja yang dibutuhkan oleh sistem yang akan dibangun. Analisis yang akan dibahas adalah analisis masalah, analisis teknologi yang digunakan, analisis sistem yang dibangun, analisis perancangan arsitektur sistem, analisis kebutuhan fungsional dan analisis kebutuhan non fungsional. Analisis ini bertujuan untuk membuat sistem yang akan dibangun dapat menyelesaikan semua permasalahan yang ada di Bab 1.

### Analisis Masalah

Analisis masalah merupakan tahapan penjabaran mengenai masalah yang ada sebelum pengembangan sistem dibuat. Dalam penelitian ini, permasalahan pertama yang dihadapi yaitu sulitnya untuk mengontrol tanaman kaktus dan sukulen, dikarenakan belum adanya sistem monitoring tanaman kaktus dan sukulen berdasarkan data suhu, kelembapan tanah dan intensitas cahaya yang dapat mengolah atau memproses data suhu, kelembapan tanah dan intensitas cahaya untuk didapatkan hasil kondisi tanaman.

### Analisis Proses Bisnis

Analisis proses bisnis merupakan tahapan yang bertujuan untuk mengetahui proses bisnis apa saja yang aka nada pada sistem yang akan dibangun

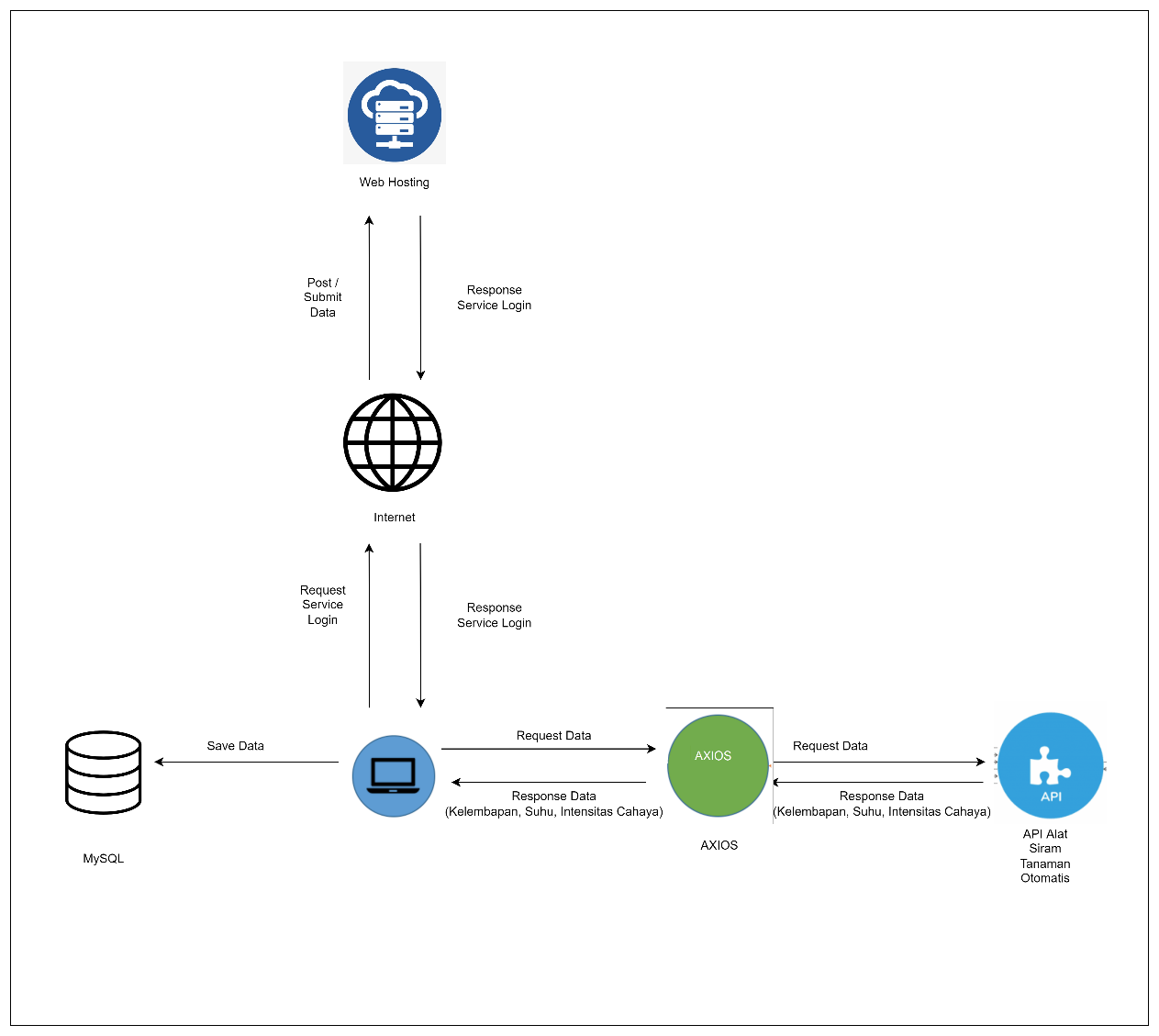
#### Analisis Proses Bisnis Sistem Yang Akan Dibangun

Analisis proses bisnis dari sistem yang akan dibangun merupakan gambaran dari bagaimana cara kerja sistem yang akan dibangun pada penelitian ini. Adapuun alur dari sistem yang akan dibangun adalah sebagai berikut :

1. Proses Bisnis Deteksi Kelembapan Tanah
2. Alat melakukan deteksi pada media tanam
3. Alat mengirimkan data ke API melalui MQTT
4. Sistem melakukan *request* data kelembapan tanah ke API
5. API menerima *request* dari sistem
6. API mengirimkan data kelembapan tanah ke sistem
7. Sistem menerima data kelembapan tanah dari API
8. Sistem menampilkan keterangan berdasarkan data kelembapan tanah yang didapat dari API
9. Sistem menyimpan data dari API ke database
10. Proses Bisnis Penyiraman Tanaman Otomatis
11. Alat melakukan deteksi pada media tanam
12. Alat mengirimkan data ke API melalui MQTT
13. Sistem melakukan *request* data kelembapan tanah ke API
14. API menerima *request* dari sistem
15. API mengirimkan data kelembapan tanah ke sistem
16. Sistem menerima data kelembapan tanah dari API
17. Sistem mengolah informasi kelembapan tanah, apabila kelembapan tanah dibawah 50%, maka kran akan menyala selama 15 detik, jika kelembapan diatas 50%, maka kran akan mati
18. Proses Bisnis Deteksi Intensitas Cahaya
19. Alatmelakukan deteksi pada sekitar tanaman
20. Alat mengirimkan data ke API melalui MQTT
21. Sistem melakukan *request* data intensitas cahaya ke API
22. API menerima *request* dari sistem
23. API mengirimkan data intensitas cahaya ke sistem
24. Sistem menerima data intensitas cahaya tanah dari alat
25. Sistem menampilkan keterangan berdasarkan data intensitas cahaya yang didapat dari alat
26. Proses Bisnis Menyalakan Lampu CFL Otomatis
27. Sensor *Light Dependant Resistor* melakukan deteksi pada sekitar tanaman
28. Sistem melakukan *request* data intensitas cahaya ke alat
29. Alat Menerima *request* dari sistem
30. Alat mengirimkan data kelembapan intensitas cahaya ke sistem
31. Sistem menerima data intensitas cahaya tanah dari alat
32. Sistem mengolah informasi intensitas cahaya, apabila intensitas cahaya mendapat suhu dibawah 26C maka lampu LED akan menyala otomatis, dan apabila intensitas cahaya diatas 26C, maka lampu akan otomatis mati
33. Proses Bisnis Deteksi Suhu
34. Alatmelakukan deteksi pada sekitar tanaman
35. Sistem melakukan *request* data suhu ke alat
36. Alat Menerima *request* dari sistem
37. Alat mengirimkan data suhu ke sistem
38. Sistem menerima data suhu dari alat
39. Sistem menampilkan keterangan berdasarkan data suhu yang didapat dari alat
40. Proses Bisnis Menyalakan *Element Heater* dan Kipas Otomatis
41. Alatmelakukan deteksi pada sekitar tanaman
42. Sistem melakukan *request* data suhu ke alat
43. Alat Menerima *request* dari sistem
44. Alat mengirimkan data suhu ke sistem
45. Sistem menerima data suhu dari alat
46. Sistem mengolah informasi suhu, apabila suhu diatas 32˚C, maka kipas akan menyala dan pemanas akan mati, apabila suhu dibawah 25˚C, maka kipas akan mati dan pemanas akan menyala

### Analisis Arsitektur Sistem

Arsitektur dibuat untuk mendifinisikan komponen – komponen yang ada dalam sistem secara lebih spesifik. Pada penelitian ini, arsitektur sistem nya adalah sebagai berikut :



Gambar 3. 1 Analisis Arsitektur Desain Sistem

Keterangan :

1. Sistem mengirimkan request service untuk login melalui internet
2. Web Hosting menerima request login dari sistem, melakukan proses pengecekan data username yang dikirimkan sistem, dan mengirimkan response datanya kembali kepada sistem
3. Melalui Axios, sistem mengirimkan *request* untuk mengambil data tanaman dari alat siram tanaman otomatis
4. Alat siram tanaman otomatis menerima permintaan dari sistem, dan mengirimkan data kepada sistem
5. Sistem menyimpan data yang diterima dari alat siram tanaman ke dalam basis data untuk nantinya diproses dan ditampilkan kepada pengguna

### Analisis Teknologi Yang Digunakan

Analisis teknologi dilakukan bermaksud untuk memberi gambaran teknologi apa saja yang digunakan dalam penelitian ini

#### API Alat Siram Tanaman Otomatis

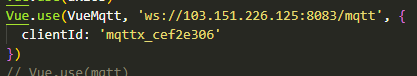
API Alat Siram Tanaman Otomatis merupakan API yang dibuat oleh peneliti untuk digunakan dalam sistem ini agar dapat memanfaatkan *Service* pada aplikasi, API Alat Siram Tanaman Otomatis ini digunakan oleh sistem untuk menampilkan data - data dari suatu tanaman, seperti data kelembapan tanah, suhu dan intensitas cahaya, juga untuk menyalakan dan mematikan kran yg terdapat pada tanaman. Untuk mengimplementasikan API Alat Siram Tanaman Otomatis ke sistem, langkah – langkah nya adalah sebagai berikut

1. Menginstall plugin Axios terlebih dahulu
2. Setelah menginstall plugin Axios, panggil fungsi Axios pada halaman yang akan menampilkan data – data dari API, untuk pemanggilan fungsi Axios dapat dilihat pada gambar berikut

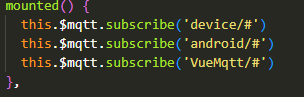


Gambar 3. 2 Skrip import Axios ke sistem

1. Langkah selanjutnya tambahkan *method* ambil data dengan menambahkan skirp url dari API yang digunakan, dan tambahkan *field* apa saja yang akan ditampilkan pada sistem

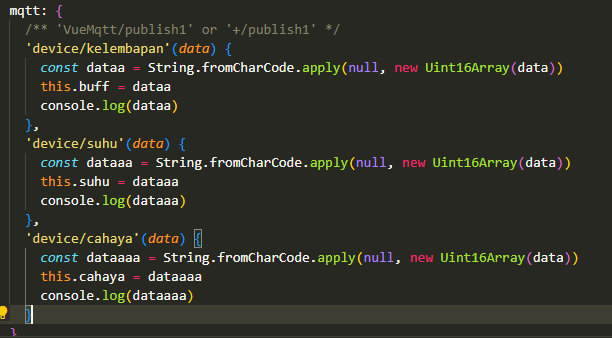


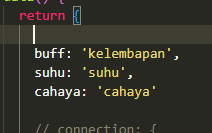
Gambar 3. 3 Skrip URL dari API yang digunakan



Gambar 3. 4 Skrip FIeld yang akan ditampilkan

1. Setelah menambahkan url dari API yang digunakan, maka deklarasikan dan halaman utama, tampilkan *field* yang dipanggil dari skrip url sebelumnya



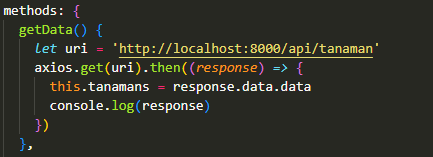


Gambar 3. 5 Skrip deklarasi field

#### Axios

Axios merupakan sebuah *library* yang bersifat terbuka (*open-source)* yang digunakan untuk mengambil atau menampilkan data dari sebuah API. *Library* Axios ini biasa digunakan dalam beberapa Bahasa pemrograman, seperti Vue JS, Axios ini lebih popular dalam pendekatan untuk permintaan atau pengambilan data dari sebuah API.

Singkatnya Axios digunakan pada setiap website yang akan dihubungkan dengan data yg terdapat pada API, maka website harus mengirimkan request ke API melalui Axios, untuk melakukan *request* data kepada API, dapat dilihat dari gambar berikut ini



Gambar 3. 6 Skrip Axios untuk mengambil data dari API

#### Vue-Router

Vue-Router termasuk kedalam teknologi yang dimiliki dari framework Vue.js , Vue-Router yang diimplementasikan pada sistem yang dibangun pada penelitian ini bertujuan agar sistem yang dibangun dapat berpindah halaman ketika pengguna menekan menu lainnya. Pada penggunaannya pada aplikasi yang dibuat, Vue-Router ini digunakan ketika actor akan memilih halaman tertentu yang terdapat pada aplikasi, maka aplikasi akan mengarahkan ke halaman yang dituju.

### Analisis SKPL

Analisis SKPL atau spesifikasi kebutuhan perangkat linak merupakan tahapan yang menjelaskan hasil dari analisis kebutuhan – kebutuhan fungsional apa saja yang harus ada pada sistem supaya dapat memecahkan masalah – masalah yang ada di identifikasi masalah. Selain kebutuhan fungsional, tahapan ini juga membahasa kebutuhan non-fungsional

#### Analisis Spesifikasi Kebutuhan Fungsional

Berdasarkan dari analisis masalah, maka didapatkan spesifikasi kebutuhan fungsional, yaitu dapat dilihat pada tabel di bawah ini.

Tabel 3. 1 SKPL Fungsional

|  |  |  |
| --- | --- | --- |
| No | SKPL-ID | Spesifikasi Kebutuhan Fungsional |
| 1 | SKPL-F01 | Sistem dapat mengambil data kelembapan tanah pada tanaman |
| 2 | SKPL-F02 | Sistem dapat mengambil data intensitas cahaya di sekitar tanaman |
| 3 | SKPL-F03 | Sistem dapat mengambil data suhu di sekitar tanaman |
| 4 | SKPL-F04 | Sistem dapat menampilkan informasi apabila terdapat salah satu parameter bernilai kurang dari nilai optimal |
| 5 | SKPL-F05 | Sistem dapat melakukan penyiraman tanaman |
| 6 | SKPL-F06 | Sistem dapat menyalakan lampu CFL |
| 7 | SKPL-F07 | Sistem dapat menyalakan *element heater* |

#### Analisis Spesifikasi Kebutuhan Non-Fungsional

Sementara untuk kebutuhan non-fungsional, bisa dilihat pada tabel berikut

Tabel 3. 2 SKPL Non Fungsional

|  |  |  |
| --- | --- | --- |
| No | SKPL-ID | Spesifikasi Kebutuhan Perangkat Lunak |
| 1 | SKPL-NF01 | Sistem yang dibuat berbasis website |
| 2 | SKPL-NF02 | Bahasa pemrograman yang dibangun untuk membangun sistem yaitu Javascript untuk *Front-end* dengan menggunakan framework Vue.JS dan PHP untuk *Back-end* dengan menggunakan framework Laravel |
| 3 | SKPL-NF03 | Basis data yang digunakan untuk sistem yaitu MySQL |
| 4 | SKPL-NF04 | Sistem dapat berjalan dalam browser Mozilla Firefox, Google Chrome, dan Microsoft Edge |
| 5 | SKPL-NF05 | Untuk perangkat keras, spesifikasi minimal supaya sistem dapat berjalan yaitu prosesor Intel Core i3 5005U , RAM 2GB, Harddisk 100GB, Monitor mendukung 16 bit *colour* |

### Analisis Kebutuhan Non-Fungsional

Analisis kebutuhan non fungsional menggambarkan kebutuhan luar sistem yang diperlukan untuk menjalankan perangkat lunak yang dibangun. Kebutuhan non fungsional untuk menjalankan sistem yang dibangun meliputi kebutuhan perangkat lunak, kebutuhan perangkat keras dan kebutuhan perangkat pikir.

#### Analisis Kebutuhan Perangkat Lunak

Analisis kebutuhan perangkat lunak merupakan proses analisis yang lebih menekankan kepada aspek pemanfaatan software. Kebutuhan non fungsional perangkat lunak dapat dilihat pada tabel 3.7 berikut

Tabel 3. 3 Analisis Kebutuhan Perangkat Lunak

|  |  |  |
| --- | --- | --- |
| No | SKPL-ID | Spesifikasi Kebutuhan Perangkat Lunak |
| 1 | SKPL-NF01 | Sistem yang dibuat berbasis website |
| 2 | SKPL-NF02 | Bahasa pemrograman yang dibangun untuk membangun sistem yaitu Javascript untuk *Front-end* dengan menggunakan framework Vue.JS dan PHP untuk *Back-end* dengan menggunakan framework Laravel |
| 3 | SKPL-NF03 | Basis data yang digunakan untuk sistem yaitu MySQL |
| 4 | SKPL-NF04 | Sistem dapat berjalan dalam sistem operasi Windows 10 |

##### Fakta Spesifikasi Perangkat Lunak Penliti

Dalam pembuatan sistem, perangkat lunak yang dipakai oleh peneliti dilampirkan dalam tabel berikut.

Tabel 3. 4 Fakta Spesifikasi Perangkat Lunak Peneliti

|  |  |
| --- | --- |
| Sistem Operasi | Microsoft Windows 10 |
| DBMS | MySQL |
| Aplikasi Pembuatan | Visual Studio Code |

#### Analisis Kebutuhan Perangkat Keras

Analisis kebutuhan perangkat keras merupakan proses analisis yang lebih menekankan kepada aspek pemanfaatan hardware. Kebutuhan non fungsional perangkat keras dapat dilihat pada tabel 3.10 berikut.

Tabel 3. 5 Analisis Kebutuhan Perangkat Keras

|  |  |  |
| --- | --- | --- |
| No | SKPL-ID | Spesifikasi Kebutuhan Perangkat Lunak |
| 1 | SKPL-NF05 | Untuk spesifikasi kebutuhan perangkat keras, spesifikasi minimal supaya sistem dapat berjalan optimal yaitu, Prosesor Intel® Pentium® 4, RAM 2GB, Harddisk 80GB |
| 2 | SKPL-NF06 | Untuk kebutuhan sensor, perangkat yang dipastikan kompatibel yaitu yang memiliki board yang dapat terhubung dengan Node MCU |

##### Fakta Spesifikasi Perangkat Keras Peneliti

Fakta spesifikasi perangkat keras (komputer) yang peneliti gunakan selama penelitian dapat dilihat pada tabel 3.11 berikut

Tabel 3. 6 Fakta Spesifikasi Perangkat Keras Peneliti

|  |  |  |
| --- | --- | --- |
| No | Nama Item | Spesifikasi |
| 1 | CPU | Intel Core i3 5005U CPU @ 2.00 GHz |
| 2 | Memory | 6GB |
| 3 | Storage | 500GB |
| 4 | GPU | Intel® HD Graphic 5500 |

#### Analisis Kebutuhan Perangkat Pikir

Analisis kebutuhan perangkat pikir yang dilakukan pada penelitian ini akan ditargetkan dengan empat kategori yaitu tanggung jawab, tingkat pendidikan, tingkat keterampilan yang dimiliki, dan pengalaman menggunakan komputer. Perangkat pikir yang terlibat dalam sistem ini adalah pengguna biasa atau individu yang ingin kondisi kesehatan jantungnya selalu di monitoring, dan olahragawan. Setiap perangkat piker pastinya memiliki hak akses, namun karena ini sistem yang akan dibangun untuk memecahkan semua masalah yang ada di batasan masalah, maka setiap perangkat pikir dapat mengakses semua hak akses yang ada. Untuk kriteria yang ditargetkannya, dapat dilihat pada tabel 3.15 berikut.

Tabel 3. 7 Analisis Kebutuhan Perangkat Pikir

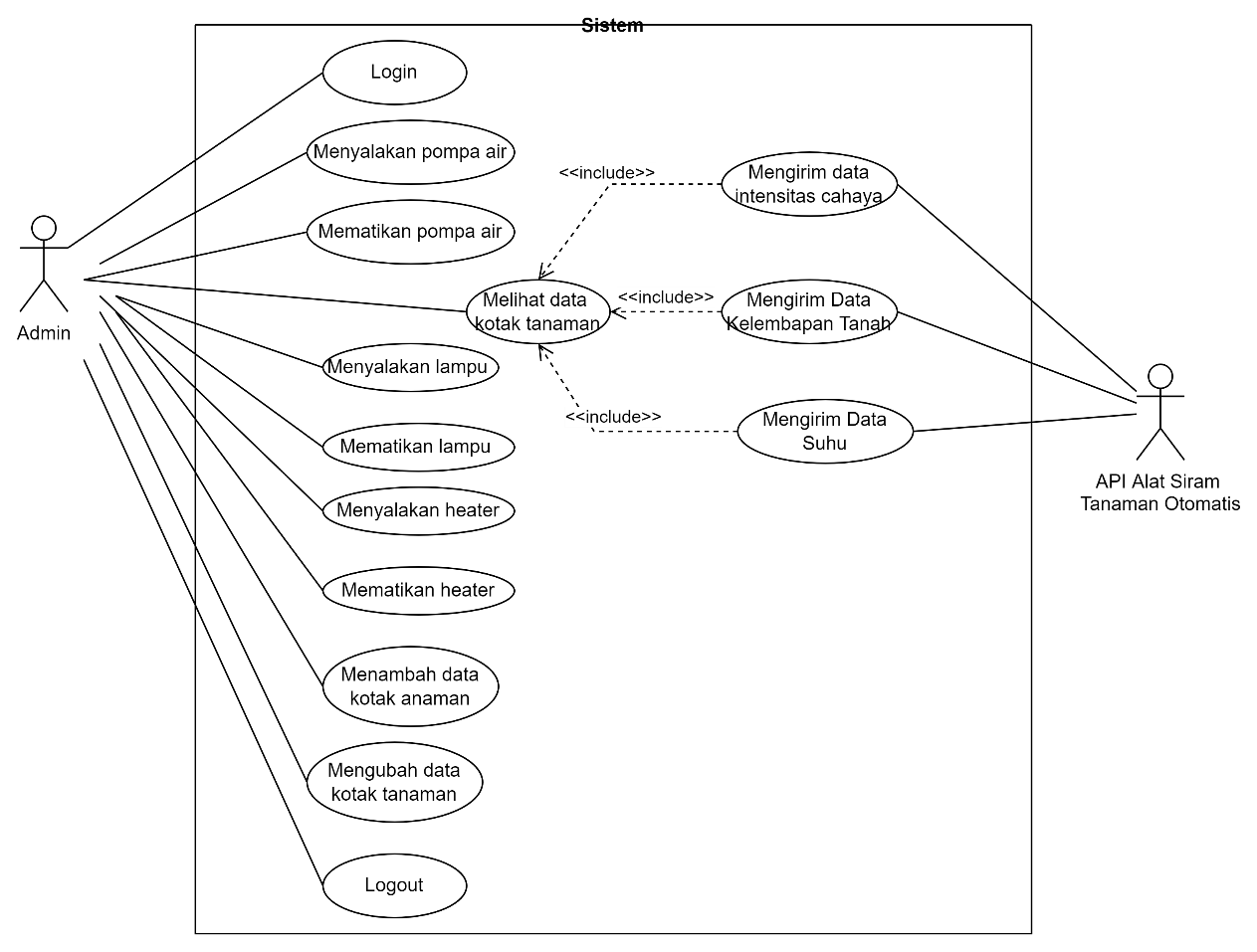
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Pengguna Sistem | Hak Akses | Tingkat Pendidikan | Tingkat Keterampilan yang Harus Dimiliki | Pengalaman Menggunakan Komputer |
| 1 | Pengelola Greenhouse | 1. Dapat mengakses menu tanaman 2. Dapat mendapatkan notifikasi ketika media tanam sebuah tanaman kering 3. Dapat mendapatkan notifikasi ketika intensitas cahaya pada suatu tanaman kurang dari nilai optimal 4. Dapat mendapatkan notifikasi ketika suhu di sekitar tanaman kurang dari nilai optimal 5. Dapat melakukan penyiraman tanaman 6. Dapat melihat riwayat data tanaman | SMP/SMA/SMK/Sederajat/D3/S1/Jenjang Lebih Tinggi | Memiliki kemampuan untuk menggunakan computer atau smartpohne | Baik, |

### Analisis Kebutuhan Fungsional

Analisis kebutuhan fungsional dilakukan menggunakan bantuan tools tambahan, ayitu UML, adapun tahapan analisis sistem menggunakan UML 70 meliputi Use Case Diagram, Use Case Scenario, Activity Diagram, Sequence Diagram, dan Class Diagram.

#### Use Case Diagram

Use case diagram adalah pemodelan yang digunakan untuk menggambarkan kelakuan sistem yang dibuat. Use case diagram menggambarkan sebuah interaksi antara satu atau lebih aktor dengan sistem yang dibuat. Secara sederhana, use case diagram digunakan untuk memahami fungsi apa saja yang ada di dalam sebuah sistem dan siapa saja yang dapat menggunakan fungsi-fungsi tersebut. Berikut use case diagram untuk sistem yang akan dibangun pada gambar berikut ini.



Gambar 3. 7 Use Case Diagram

#### Definisi Aktor

Tabel 3. 8 Definisi Aktor

|  |  |  |
| --- | --- | --- |
| No | Aktor | Deskripsi |
| 1 | Aktor | Aktor dengan role ini memiliki wewenang untuk menambahkan dan memanipulasi data tanaman serta melakukan penyiraman tanaman. |
| 2 | API Alat Siram Tanaman Otomatis | Aktor dengan role ini memiliki wewenang untuk mengirimkan data pada tanaman. |

#### Definisi Use Case

Tabel 3. 9 Definisi Use Case

|  |  |  |
| --- | --- | --- |
| **No** | **Use Case** | **Deskripsi** |
| 1 | Login | **Kode Use Case**: UC-01  **Deskripsi Singkat**: Sistem menyediakan halaman login sebelum memasuki laman website  **Aktor**: Admin  **Trigger**: Aktor memasukan username dan password  **Prekondisi**: Halaman utama web ditampilkan  **Pascakondisi**: Sistem menampilkan halaman awal website berupa informasi beberapa tanaman yang sudah ditanam |
| 2 | Menyalakan Pompa Air | **Kode Use Case**: UC-02  **Deskripsi Singkat**: Sistem menyediakan halaman untuk melakukan penyiraman tanaman.  **Aktor**: Admin  **Trigger**: Aktor mengetuk tombol “Siram” pada halaman detail tanaman  **Prekondisi**: Halaman untuk detail ditampilkan  **Pascakondisi**: Pompa air akan menyala |
| 3 | Mematikan Pompa Air | **Kode Use Case**: UC-03  **Deskripsi Singkat**: Sistem menyediakan halaman untuk melakukan penyiraman tanaman.  **Aktor**: Admin  **Trigger**: Aktor mengetuk tombol “Matikan Pompa” pada halaman detail tanaman  **Prekondisi**: Halaman untuk detail ditampilkan  **Pascakondisi**: Pompa air akan mati |
| 4 | Melihat Data Kotak Tanaman | **Kode Use Case**: UC-04  **Deskripsi Singkat**: Sistem menampilkan data tanaman yang sudah ditanam, dan riwayat data – data tanaman.  **Aktor**: Admin  **Trigger**: Aktor harus login  **Prekondisi**: Halaman login ditampilkan  **Pascakondisi**: Sistem menampilkan data tanaman. |
| 5 | Menyalakan Lampu | **Kode Use Case**: UC-05  **Deskripsi Singkat**: Sistem menyediakan tombol untuk menyalakan dan mematikan lampu  **Aktor**: Admin  **Trigger**: Aktor menekan tombol “Nyalakan lampu” pada halaman detail website  **Prekondisi**: Halaman detail akan ditampilkan  **Pascakondisi**: Lampu menyala. |
| 6 | Mematikan Lampu | **Kode Use Case**: UC-06  **Deskripsi Singkat**: Sistem menyediakan tombol untuk menyalakan dan mematikan lampu  **Aktor**: Admin  **Trigger**: Aktor menekan tombol “Mematikan lampu” pada halaman detail website  **Prekondisi**: Halaman detail akan ditampilkan  **Pascakondisi**: Lampu mati. |
| 7 | Menyalakan *Element Heater* | **Kode Use Case**: UC-07  **Deskripsi Singkat**: Sistem menyediakan tombol untuk menyalakan dan mematikan *element heater*  **Aktor**: Admin  **Trigger**: Aktor menekan tombol “Nyalakan Pemanas” pada halaman detail website  **Prekondisi**: Halaman detail akan ditampilkan  **Pascakondisi**: *Element Heater* menyala. |
| 8 | Mematikan *Element Heater* | **Kode Use Case**: UC-08  **Deskripsi Singkat**: Sistem menyediakan tombol untuk menyalakan dan mematikan *element heater*  **Aktor**: Admin  **Trigger**: Aktor menekan tombol “Matikan Pemanas” pada halaman detail website  **Prekondisi**: Halaman detail akan ditampilkan  **Pascakondisi**: *Element Heater* mati. |
| 9 | Menambah Data Kotak Tanaman | **Kode Use Case**: UC-09  **Deskripsi Singkat**: Sistem menyediakan tombol untuk menambah data tanaman.  **Aktor**: Admin  **Trigger**: Aktor memilih icon “+” pada halaman utama website dan mengisi form lalu memilih tombol “simpan” pada halaman tambah tanaman.  **Prekondisi**: Halaman home akan ditampilkan  **Pascakondisi**: Sistem menambahkan data kedalam database. |
| 10 | Mengubah Data Kotak Tanaman | **Kode Use Case**: UC-10  **Deskripsi Singkat**: Sistem menyediakan pilihan untuk menghapus data tanaman.  **Aktor**: Admin  **Trigger**: Aktor memilih tombol “Edit” pada halaman detail website dan mengisi form yang telah disediakan.  **Prekondisi**: Halaman edit akan ditampilkan  **Pascakondisi**: Sistem mengubah data kedalam database. |
| 11 | Mengirimkan data intensitas cahaya | **Kode Use Case:** UC-11  **Deskripsi Singkat:** Alat akan mengirimkan data setiap ada perubahan intensitas cahaya  **Aktor:** API Alat Siram Tanaman Otomatis  **Trigger:** Perubahan intensitas cahaya pada tanaman  **Prekondisi:** Alat mengirimkan data pada sistem  **Pascakondisi:** Alat mengedit data yang ada di database ketika terjadi perubahan. |
| 12 | Mengirimkan data kelembapan tanah | **Kode Use Case:** UC-12  **Deskripsi Singkat:** Alat akan mengirimkan data setiap ada perubahan kelembapan tanah  **Aktor:** API Alat Siram Tanaman Otomatis  **Trigger:** Perubahan kelembapan tanah pada media tanam  **Prekondisi:** Alat mengirimkan data pada sistem  **Pascakondisi:** Alat mengedit data yang ada di database ketika terjadi perubahan. |
| 13 | Mengirimkan data suhu | **Kode Use Case:** UC-13  **Deskripsi Singkat:** Alat akan mengirimkan data setiap ada perubahan suhu di sekitar tanaman.  **Aktor:** API Alat Siram Tanaman Otomatis  **Trigger:** Perubahan suhu di sekitar tanaman  **Prekondisi:** Alat mengirimkan data pada sistem  **Pascakondisi:** Alat mengedit data yang ada di database ketika terjadi perubahan. |
| 14 | Logout | **Kode Use Case**: UC-014  **Deskripsi Singkat**: Sistem menyediakan halaman logout untuk mengakhiri *session*  **Aktor**: Admin  **Trigger**: Aktor menekan tombol Logout  **Prekondisi**: Halaman utama web ditampilkan  **Pascakondisi**: Halaman login ditampilkan |

#### Use Case Scenario

Pada bagian ini akan diisi dengan skenario (flow of event) untuk tiap use case yang menggambarkan urutan interaksi aktor dengan use case tersebut dari awal sampai akhir. Untuk usecase scenario pertama, yaitu usecase scenario untuk usecase dengan kode UC-01 dapat dilihat pada tabel 3.18 berikut.

Tabel 3. 10 Use Case Scenario UC-01

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Login | |
| Goal In Context | Pengguna dapat masuk ke dalam sistem dengan menggunakan akun yang telah terdaftar | |
| Description | Fungsionalitas ini digunakan oleh pengguna untuk masuk atau login ke dalam sistem dengan menggunakan akun yang telah terdaftar di sistem | |
| Successful End Condition | Pengguna berhasil masuk ke dalam sistem | |
| Failed End Condition | Pengguna tidak berhasil masuk ke dalam sistem | |
| Actors | Admin, Web Service | |
| Trigger | Pengguna menekan tombol masuk dan sistem mengirimkan request login menggunakan API | |
| Main Flow | Step | Action |
|  | 1 | Pengguna membuka aplikasi |
|  | 2 | Sistem menampilkan halaman login |
|  | 3 | Pengguna menekan tombol Login |
|  | 4 | Sistem mengecek apakah akun pengguna sesuai dengan database |
|  | 5 | Pengguna berhasil masuk ke dalam sistem |
|  | 6 | Sistem mengalihkan pengguna ke halaman utama |

Untuk usecase scenario selanjutnya, yaitu usecase scenario untuk usecase dengan kode UC-02 dapat dilihat pada tabel berikut.

Tabel 3. 11 Use Case Scenario UC-02

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Menyalakan pompa air | |
| Goal In Context | Admin dapat menyalakan pompa air melalui sistem | |
| Description | Fungsionalitas ini digunakan oleh pengguna untuk menyalakan pompa air untuk menyiram tanaman. | |
| Successful End Condition | Pompa air berhasil menyala | |
| Failed End Condition | Pompa air gagal menyala | |
| Actors | Admin | |
| Trigger | Admin menekan tombol “Nyalakan Pompa” | |
| Main Flow | Step | Action |
|  | 1 | Admin masuk ke detail |
|  | 2 | Admin menekan tombol nyalakan pompa |
|  | 3 | Pompa akan menyala |

Untuk usecase scenario selanjutnya, yaitu usecase scenario untuk usecase dengan kode UC-03 dapat dilihat pada tabel berikut.

Tabel 3. 12 Use Case Scenario UC-03

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Mematikan pompa air | |
| Goal In Context | Admin dapat mematikan pompa air melalui sistem | |
| Description | Fungsionalitas ini digunakan oleh pengguna untuk mematikan pompa air untuk menyiram tanaman. | |
| Successful End Condition | Pompa air berhasil mati | |
| Failed End Condition | Pompa air gagal mati | |
| Actors | Admin | |
| Trigger | Admin menekan tombol “Matikan Pompa” | |
| Main Flow | Step | Action |
|  | 1 | Admin masuk ke detail |
|  | 2 | Admin menekan tombol matikan pompa |
|  | 3 | Pompa akan mati |

Untuk usecase scenario selanjutnya, yaitu usecase scenario untuk usecase dengan kode UC-04 dapat dilihat pada tabel berikut.

Tabel 3. 13 Use Case Scenario UC-04

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Melihat Data Kotak Tanaman | |
| Goal In Context | Admin dapat melihat data kotak tanaman yang terdiri dari data isi tanaman, data suhu, data cahaya dan data kelembapan tanah | |
| Description | Fungsionalitas ini digunakan oleh pengguna untuk menambahkan data tanaman agar masuk kedalam sistem | |
| Successful End Condition | Admin berhasil melihat data – data kotak tanaman | |
| Failed End Condition | Admin gagal melihat data – data kotak tanaman | |
| Actors | Admin, API Alat Siram Tanaman Otomatis | |
| Trigger | Admin masuk ke halaman utama | |
| Main Flow | Step | Action |
|  | 1 | Admin masuk ke halaman utama |
|  | 2 | Sistem menampilkan data – data kotak tanaman |

Untuk usecase scenario selanjutnya, yaitu usecase scenario untuk usecase dengan kode UC-05 dapat dilihat pada tabel berikut.

Tabel 3. 14 Use Case Scenario UC-05

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Menyalakan Lampu | |
| Goal In Context | Admin dapat menyalakan lampu melalui sistem | |
| Description | Fungsionalitas ini digunakan oleh Admin untuk menyalakan lampu | |
| Successful End Condition | Lampu berhasil menyala | |
| Failed End Condition | Lampu gagal menyala | |
| Actors | Admin | |
| Trigger | Admin menekan tombol “Nyalakan Lampu” | |
| Main Flow | Step | Action |
|  | 1 | Admin membuka halaman detail |
|  | 2 | Admin menekan tombol “Nyalakan Lampu” |
|  | 3 | Lampu akan menyala |

Untuk usecase scenario selanjutnya, yaitu usecase scenario untuk usecase dengan kode UC-06 dapat dilihat pada tabel berikut.

Tabel 3. 15 Use Case Scenario UC-06

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Mematikan Lampu | |
| Goal In Context | Admin dapat mematikan lampu melalui sistem | |
| Description | Fungsionalitas ini digunakan oleh Admin untuk mematikan lampu | |
| Successful End Condition | Lampu berhasil mati | |
| Failed End Condition | Lampu gagal mati | |
| Actors | Admin | |
| Trigger | Admin menekan tombol “Matikan Lampu” | |
| Main Flow | Step | Action |
|  | 1 | Admin membuka halaman detail |
|  | 2 | Admin menekan tombol “Matikan Lampu” |
|  | 3 | Lampu akan mati |

Untuk usecase scenario selanjutnya, yaitu usecase scenario untuk usecase dengan kode UC-07 dapat dilihat pada tabel berikut.

Tabel 3. 16 Use Case Scenario UC-07

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Menyalakan *Heater* | |
| Goal In Context | Admin dapat menyalakan *heater* melalui sistem | |
| Description | Fungsionalitas ini digunakan oleh Admin untuk menyalakan *heater* | |
| Successful End Condition | *Heater* berhasil menyala | |
| Failed End Condition | *Heater* gagal menyala | |
| Actors | Admin | |
| Trigger | Admin menekan tombol “Nyalakan Pemanas” | |
| Main Flow | Step | Action |
|  | 1 | Admin membuka halaman detail |
|  | 2 | Admin menekan tombol “Nyalakan Pemanas” |
|  | 3 | Pemanas akan menyala |

Untuk usecase scenario selanjutnya, yaitu usecase scenario untuk usecase dengan kode UC-08 dapat dilihat pada tabel berikut.

Tabel 3. 17 Use Case Scenario UC-08

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Mematikan *Heater* | |
| Goal In Context | Admin dapat mematikan *heater* melalui sistem | |
| Description | Fungsionalitas ini digunakan oleh Admin untuk mematikan *heater* | |
| Successful End Condition | *Heater* berhasil mati | |
| Failed End Condition | *Heater* gagal mati | |
| Actors | Admin | |
| Trigger | Admin menekan tombol “Matikan Pemanas” | |
| Main Flow | Step | Action |
|  | 1 | Admin membuka halaman detail |
|  | 2 | Admin menekan tombol “Matikan Pemanas” |
|  | 3 | Pemanas akan mati |

Untuk usecase scenario selanjutnya, yaitu usecase scenario untuk usecase dengan kode UC-09 dapat dilihat pada tabel berikut.

Tabel 3. 18 Use Case Scenario UC-09

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Menambah data kotak tanaman | |
| Goal In Context | Admin dapat menambahkan data kotak tanaman seperti nama kotak, isi tanaman dan catatan | |
| Description | Fungsionalitas ini digunakan oleh Admin untuk menambahkan data baru | |
| Successful End Condition | Data berhasil ditambahkan dan masuk ke database | |
| Failed End Condition | Data gagal ditambahkan dan gagal masuk ke database | |
| Actors | Admin | |
| Trigger | Admin menekan icon “+” dan mengisi form yang disediakan | |
| Main Flow | Step | Action |
|  | 1 | Admin menekan icon “+” pada halaman utama |
|  | 2 | Admin mengisi form yang terdapat pada halaman |
|  | 3 | Admin menekan tombol “Simpan” |
|  | 4 | Sistem berhasil menyimpan data |

Untuk usecase scenario selanjutnya, yaitu usecase scenario untuk usecase dengan kode UC-10 dapat dilihat pada tabel berikut.

Tabel 3. 19 Use Case Scenario UC-10

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Mengubah data kotak tanaman | |
| Goal In Context | Admin dapat mengubah data kotak tanaman seperti nama kotak, isi tanaman dan catatan | |
| Description | Fungsionalitas ini digunakan oleh Admin untuk mengubah data yang sudah ada | |
| Successful End Condition | Data berhasil diubah dan terubah ke database | |
| Failed End Condition | Data gagal diubah dan gagal terubah ke database | |
| Actors | Admin | |
| Trigger | Admin menekan tombol Edit dan mengisi form yang disediakan | |
| Main Flow | Step | Action |
|  | 1 | Admin menekan Edit pada halaman detail |
|  | 2 | Admin mengisi form yang terdapat pada halaman |
|  | 3 | Admin menekan tombol “Simpan” |
|  | 4 | Sistem berhasil mengubah data |

Untuk usecase scenario selanjutnya, yaitu usecase scenario untuk usecase dengan kode UC-11 dapat dilihat pada tabel berikut.

Tabel 3. 20 Use Case Scenario UC-11

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Mengirimkan Data Intensitas Cahaya | |
| Goal In Context | API dapat mengirimkan data intensitas cahaya ke sistem | |
| Description | Fungsionalitas ini digunakan oleh Admin untuk mengetahui data intensitas cahaya pada tanaman | |
| Successful End Condition | API berhasil mengirimkan data intensitas cahaya | |
| Failed End Condition | API gagal mengirimkan data intensitas cahaya | |
| Actors | API Alat Siram Tanaman Otomatis | |
| Trigger | API akan mengirimkan data setiap 5 menit sekali ke sistem | |
| Main Flow | Step | Action |
|  | 1 | Sensor mendeteksi intensitas cahaya |
|  | 2 | Sensor mengirimkan data ke API |
|  | 3 | API menerima data dari sensor |
|  | 4 | API mengirimkan data ke sistem |
|  | 5 | Sistem menampilkan data dari API ke Admin |

Untuk usecase scenario selanjutnya, yaitu usecase scenario untuk usecase dengan kode UC-12 dapat dilihat pada tabel berikut.

Tabel 3. 21 Use Case Scenario UC-12

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Mengirimkan Data Kelembapan Tanah | |
| Goal In Context | API dapat mengirimkan data kelembapan tanah ke sistem | |
| Description | Fungsionalitas ini digunakan oleh Admin untuk mengetahui data kelembapan tanah tanaman | |
| Successful End Condition | API berhasil mengirimkan data kelembapan tanah | |
| Failed End Condition | API gagal mengirimkan data kelembapan tanah | |
| Actors | API Alat Siram Tanaman Otomatis | |
| Trigger | API akan mengirimkan data setiap 5 menit sekali ke sistem | |
| Main Flow | Step | Action |
|  | 1 | Sensor mendeteksi kelembapan tanah |
|  | 2 | Sensor mengirimkan data ke API |
|  | 3 | API menerima data dari sensor |
|  | 4 | API mengirimkan data ke sistem |
|  | 5 | Sistem menampilkan data dari API |

Untuk usecase scenario selanjutnya, yaitu usecase scenario untuk usecase dengan kode UC-13 dapat dilihat pada tabel berikut.

Tabel 3. 22 Use Case Scenario UC-13

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Mengirimkan Data Suhu | |
| Goal In Context | API dapat mengirimkan data suhu ke sistem | |
| Description | Fungsionalitas ini digunakan oleh Admin untuk mengetahui data suhu tanaman | |
| Successful End Condition | API berhasil mengirimkan data suhu | |
| Failed End Condition | API gagal mengirimkan data suhu | |
| Actors | API Alat Siram Tanaman Otomatis | |
| Trigger | API akan mengirimkan data setiap 5 menit sekali ke sistem | |
| Main Flow | Step | Action |
|  | 1 | Sensor mendeteksi suhu |
|  | 2 | Sensor mengirimkan data ke API |
|  | 3 | API menerima data dari sensor |
|  | 4 | API mengirimkan data ke sistem |
|  | 5 | Sistem menampilkan data dari API |

Untuk usecase scenario selanjutnya, yaitu usecase scenario untuk usecase dengan kode UC-14 dapat dilihat pada tabel berikut.

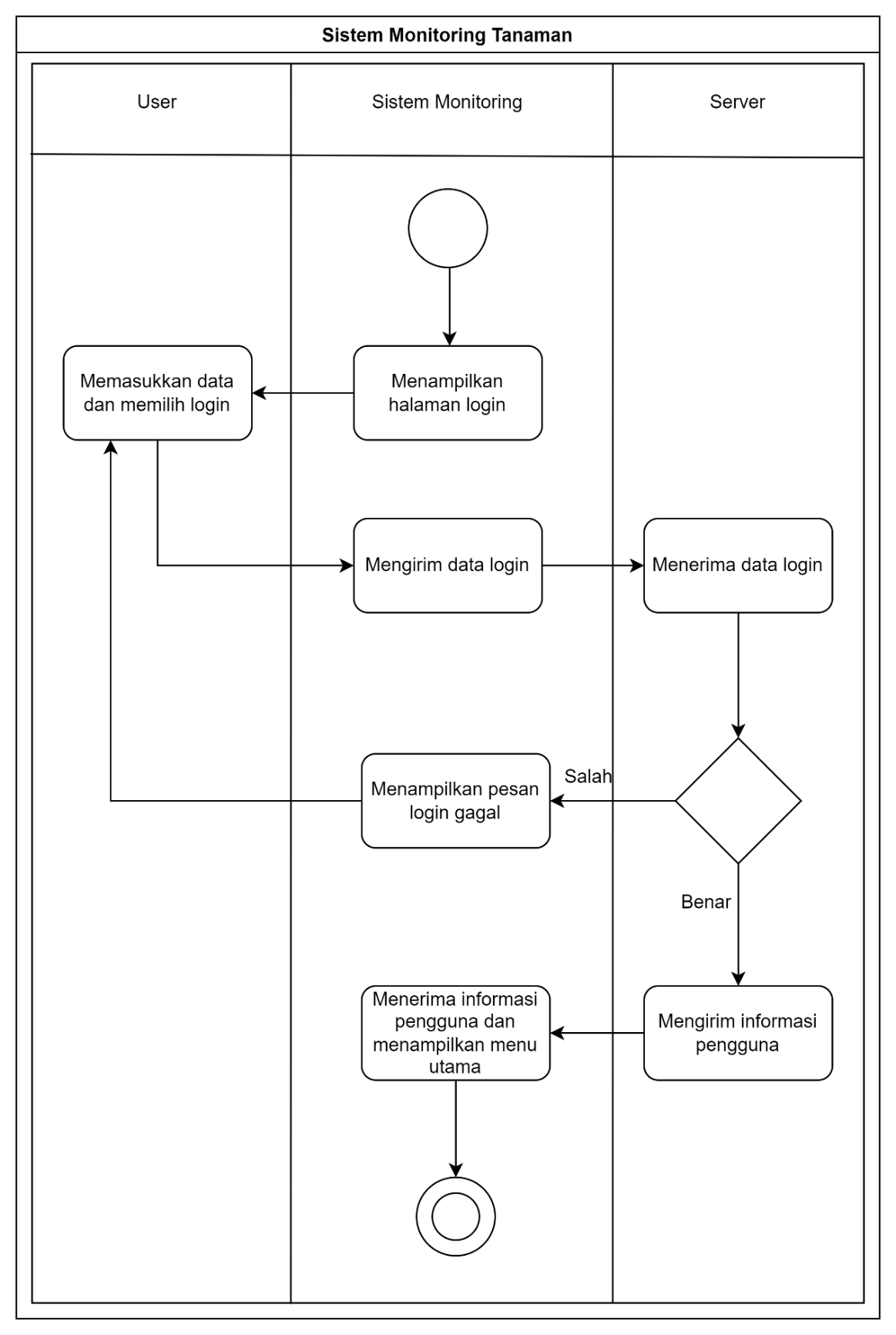
Tabel 3. 23 Use Case Scenario UC-14

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Logout | |
| Goal In Context | API dapat mengirimkan data suhu ke sistem | |
| Description | Proses ini dilakukan untuk melakukan logout dan mengakhiri sesi login yang aktif. | |
| Successful End Condition | Berhasil logout dan menampilkan halaman login | |
| Failed End Condition | Gagal logout dan tidak menampilkan halaman login | |
| Actors | Admin | |
| Trigger | Admin menekan tombol logout | |
| Main Flow | Step | Action |
|  | 1 | Admin menekan tombol logout |
|  | 2 | Aplikasi akan melakukan logout dan mengakhri sesi login yang aktif. |

#### Activity Diagram

Activity diagram memodelkan aliran kerja atau workflow dari urutan aktivitas dalam suatu proses yang mengacu pada use case diagram yang ada. Untuk activity diagram untuk usecase dengan kode UC-01 dapat dilihat pada gambar berikut.

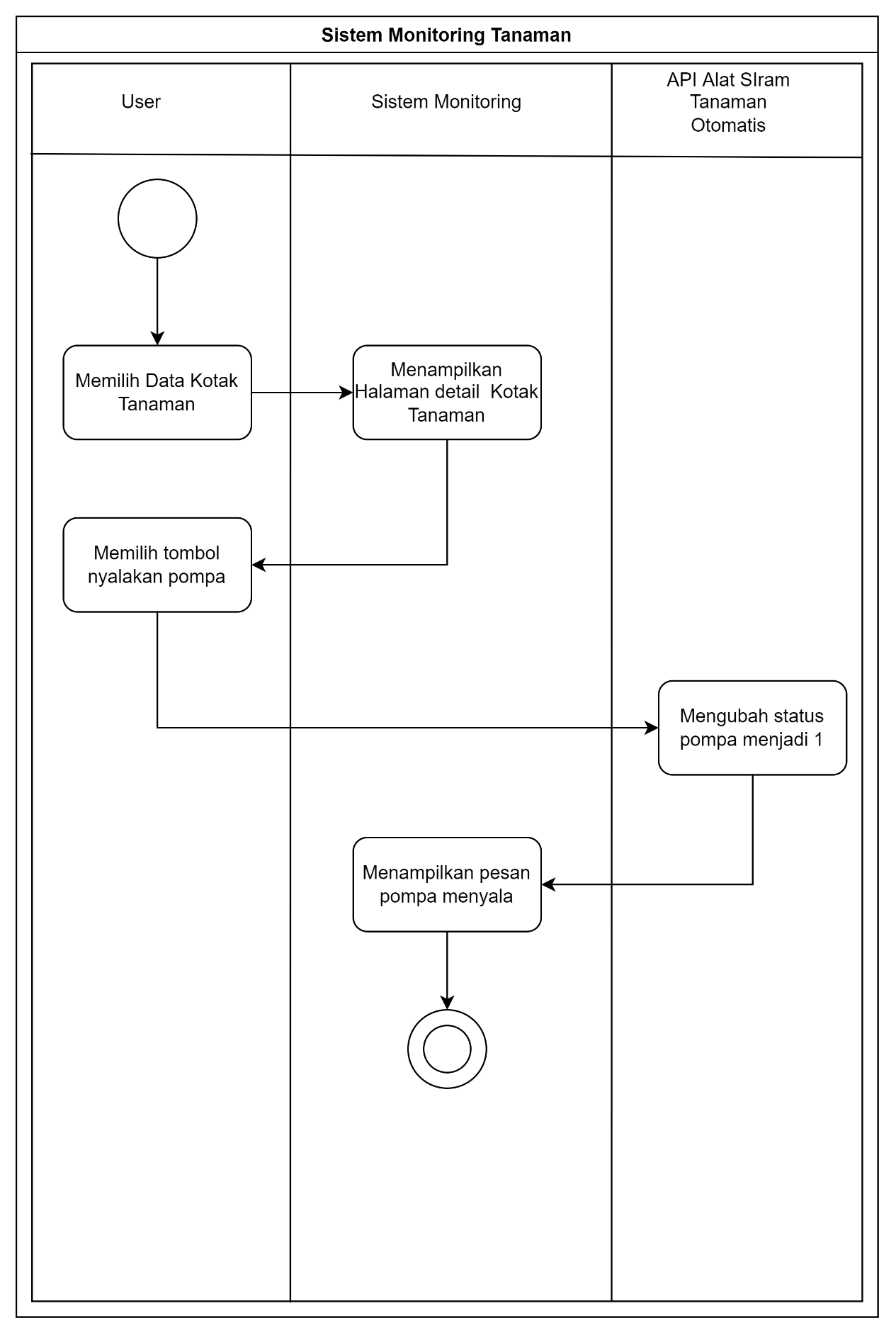
1. Activity Diagram untuk Use Case UC-01 Login



Gambar 3. 8 Activity Diagram UC-01

1. Activity Diagram Menyalakan Pompa Air

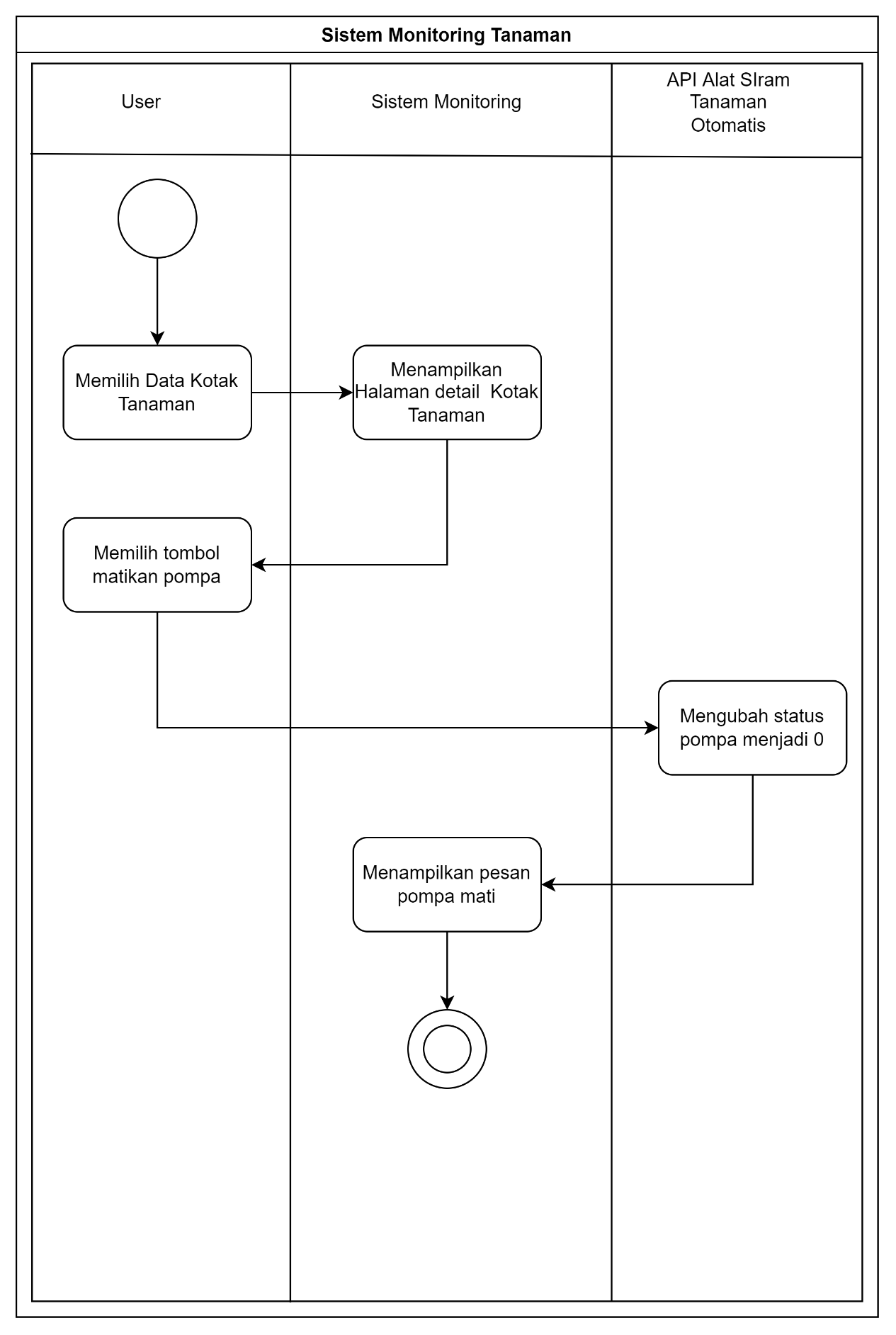
Untuk activity diagram untuk usecase dengan kode UC-02 dapat dilihat pada gambar berikut



Gambar 3. 9 Activity Diagram UC-02

1. Activity Diagram Mematikan Pompa Air

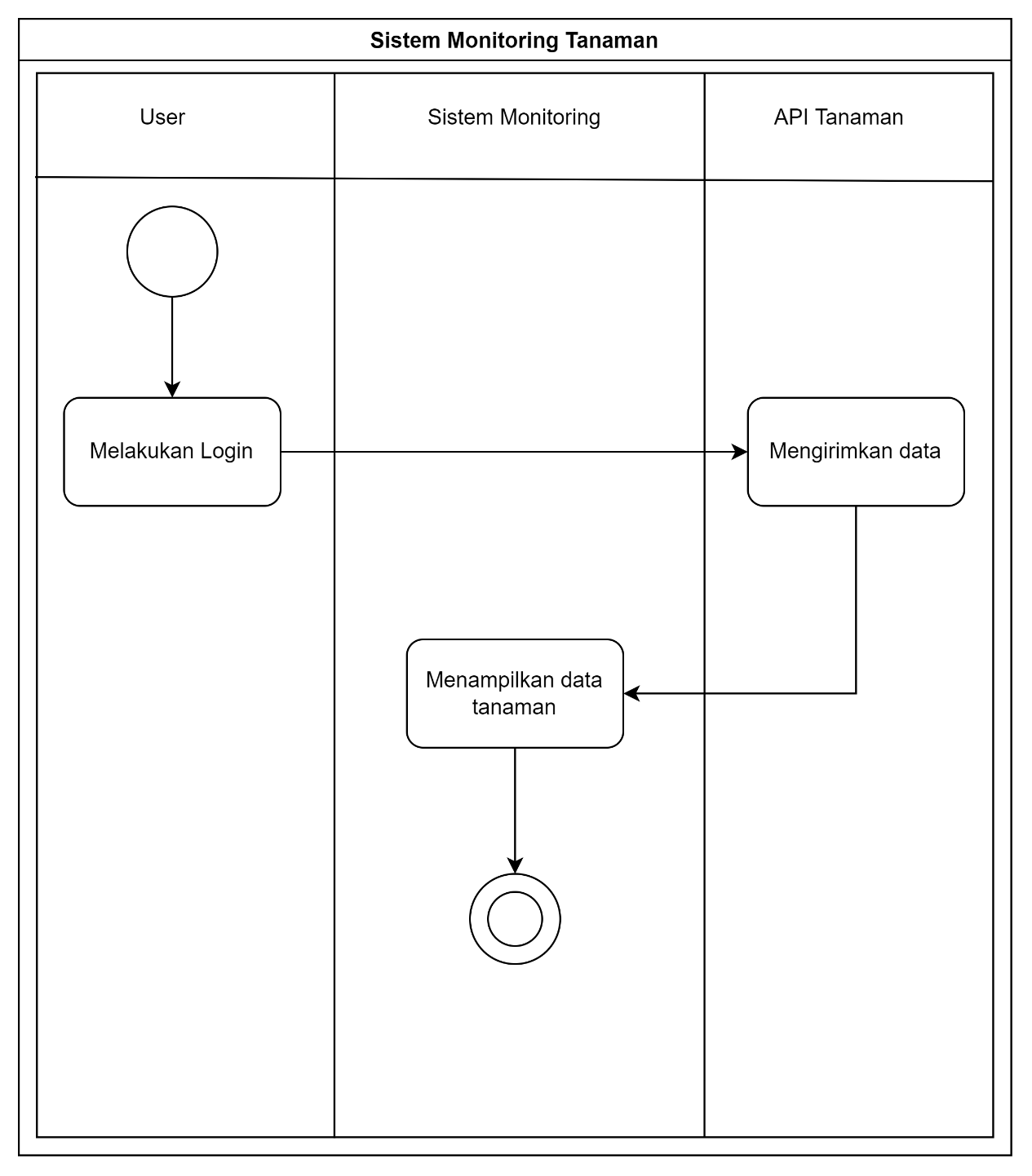
Untuk activity diagram untuk usecase dengan kode UC-03 dapat dilihat pada gambar berikut



Gambar 3. 10 Activity Diagram UC-03

1. Activity Diagram Melihat Data Kotak Tanaman

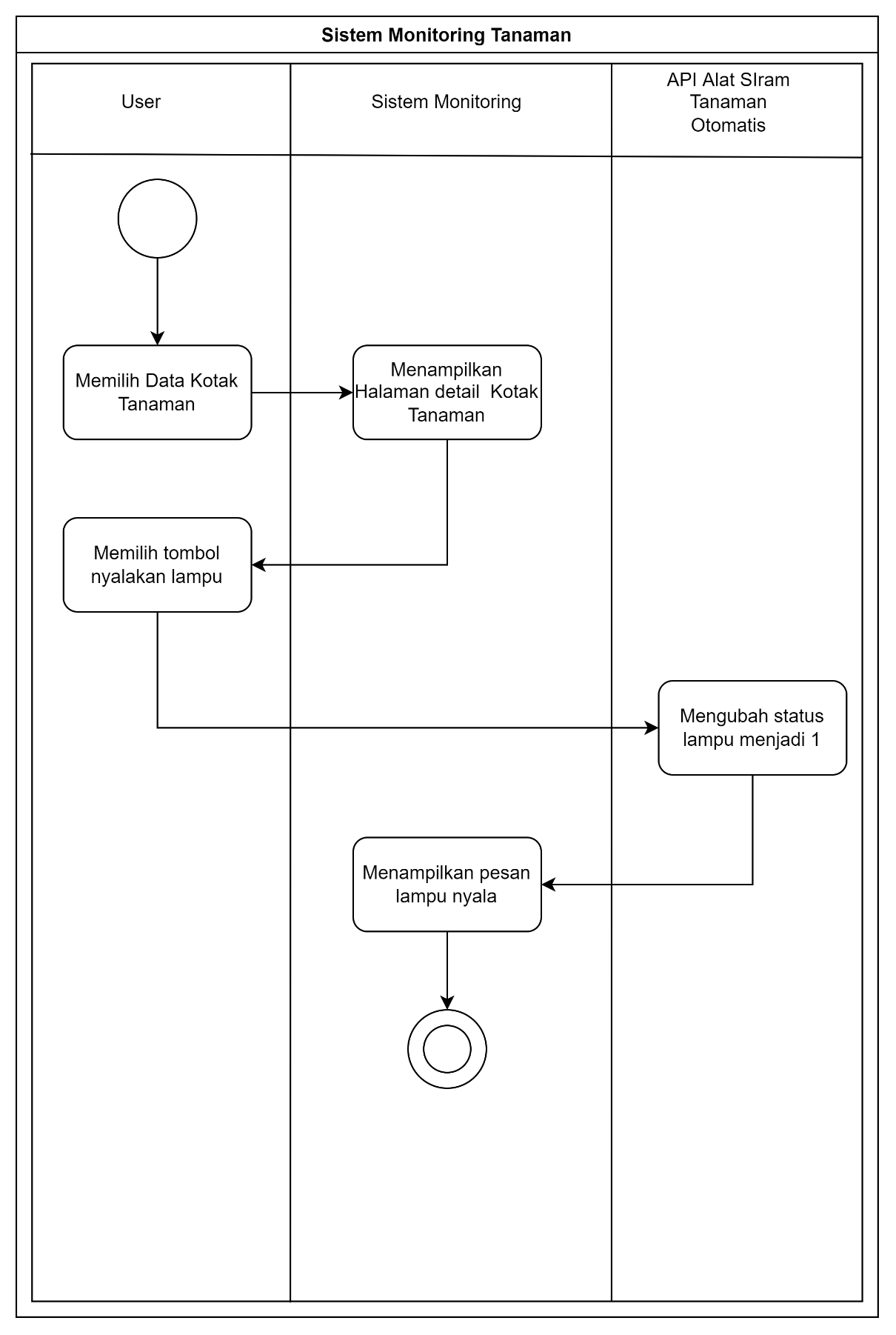
Untuk activity diagram untuk usecase dengan kode UC-04 dapat dilihat pada gambar berikut



Gambar 3. 11 Activity Diagram UC-04

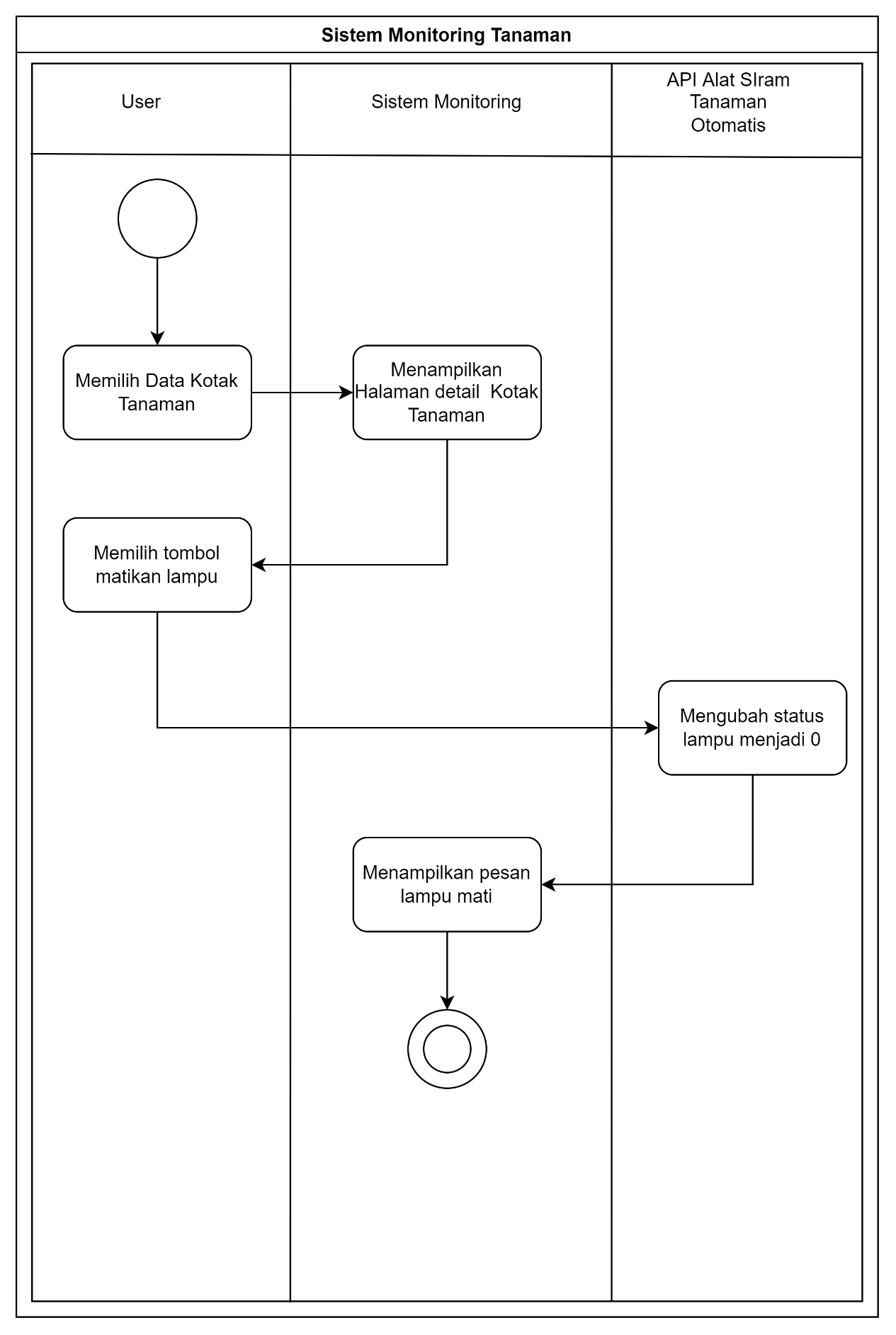
1. Activity Diagram Menyalakan Lampu

Untuk activity diagram untuk usecase dengan kode UC-05 dapat dilihat pada gambar berikut



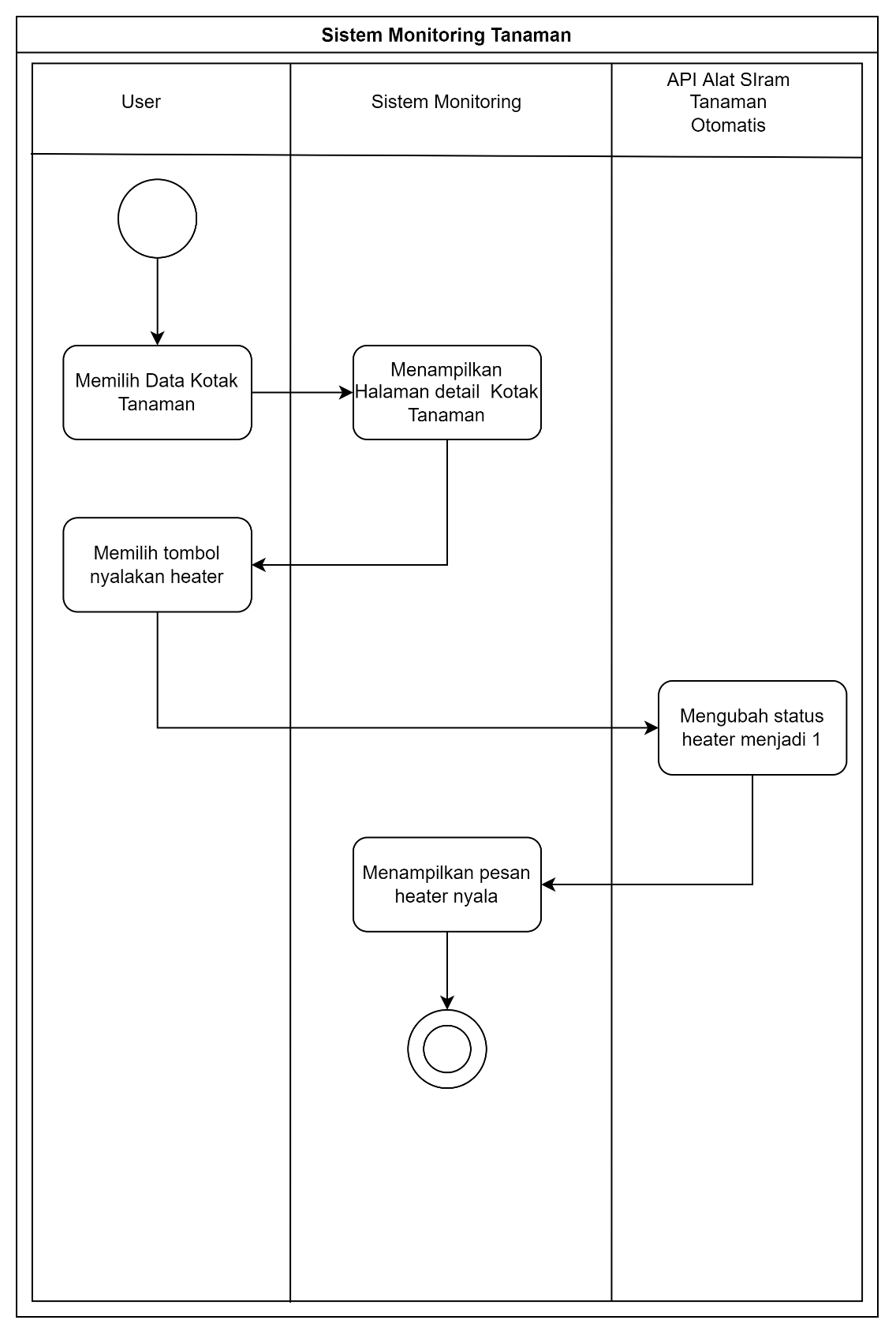
Gambar 3. 12 Activity Diagram UC-05

1. Activity Diagram Mematikan Lampu



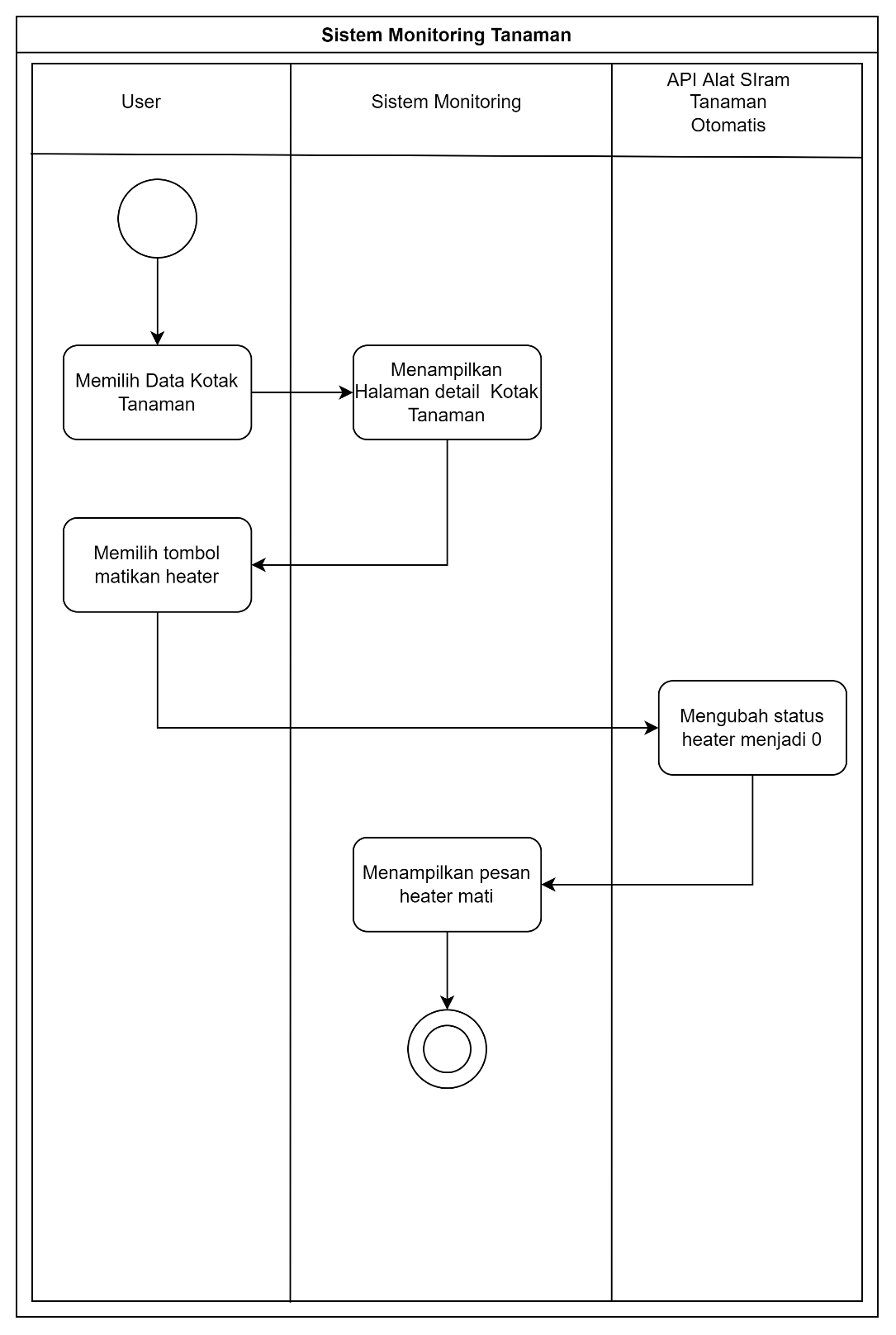
Gambar 3. 13 Activity Diagram UC-06

1. Activity Diagram Menyalakan *Heater*



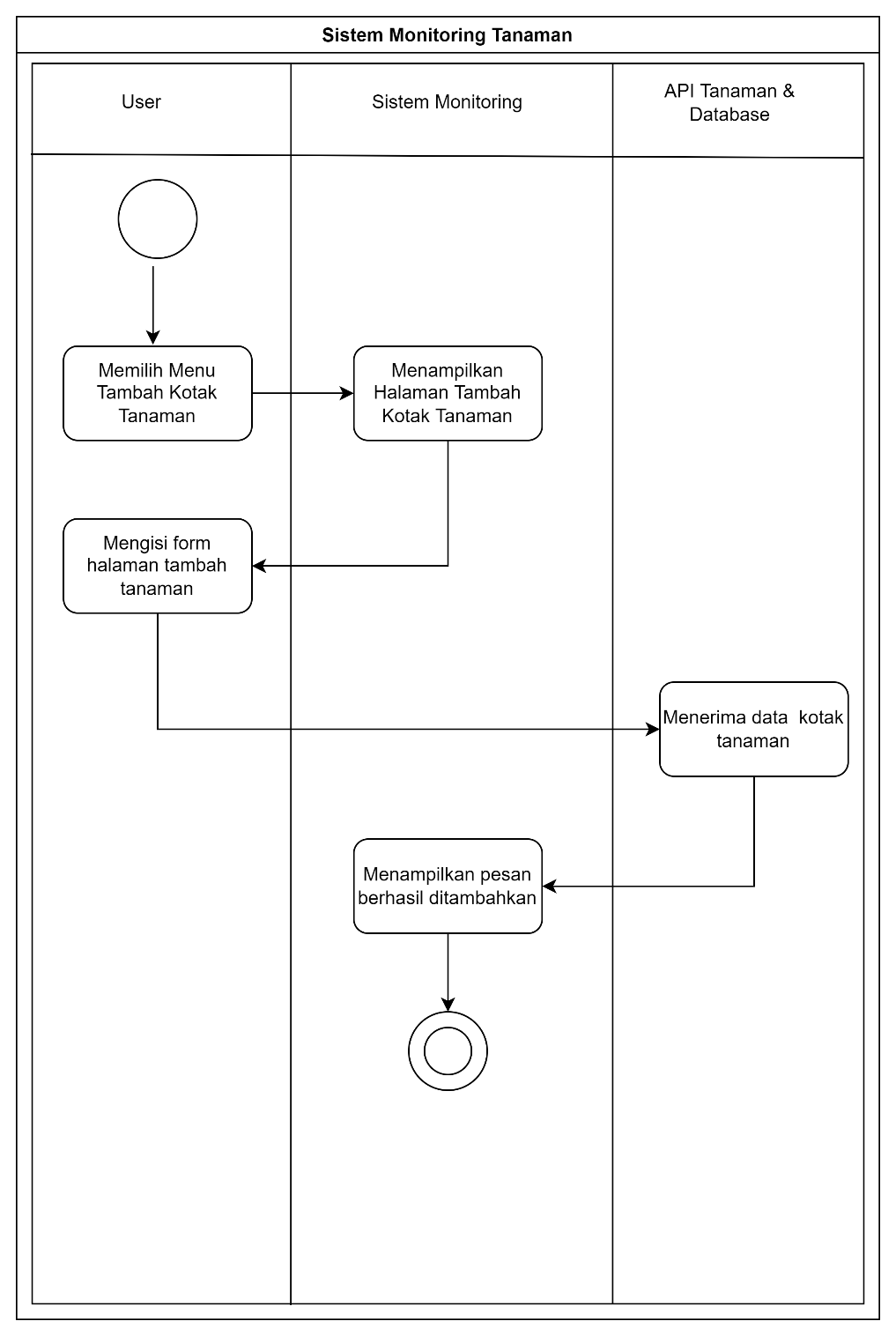
Gambar 3. 14 Activity Diagram UC-07

1. Activity Diagram Mematikan *Heater*



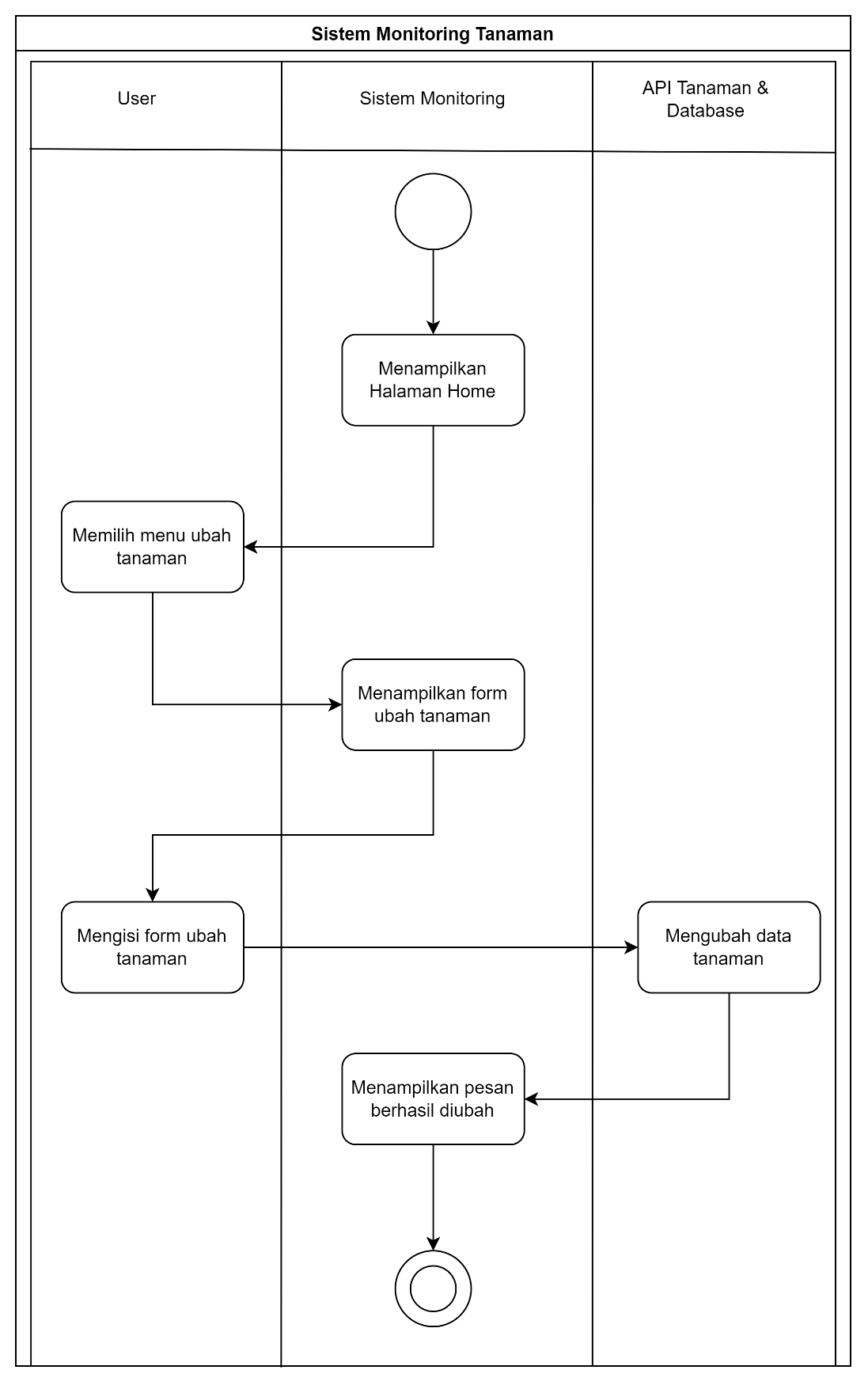
Gambar 3. 15 Activity Diagram UC-08

1. Activity Diagram Menambah Data Kotak Tanaman



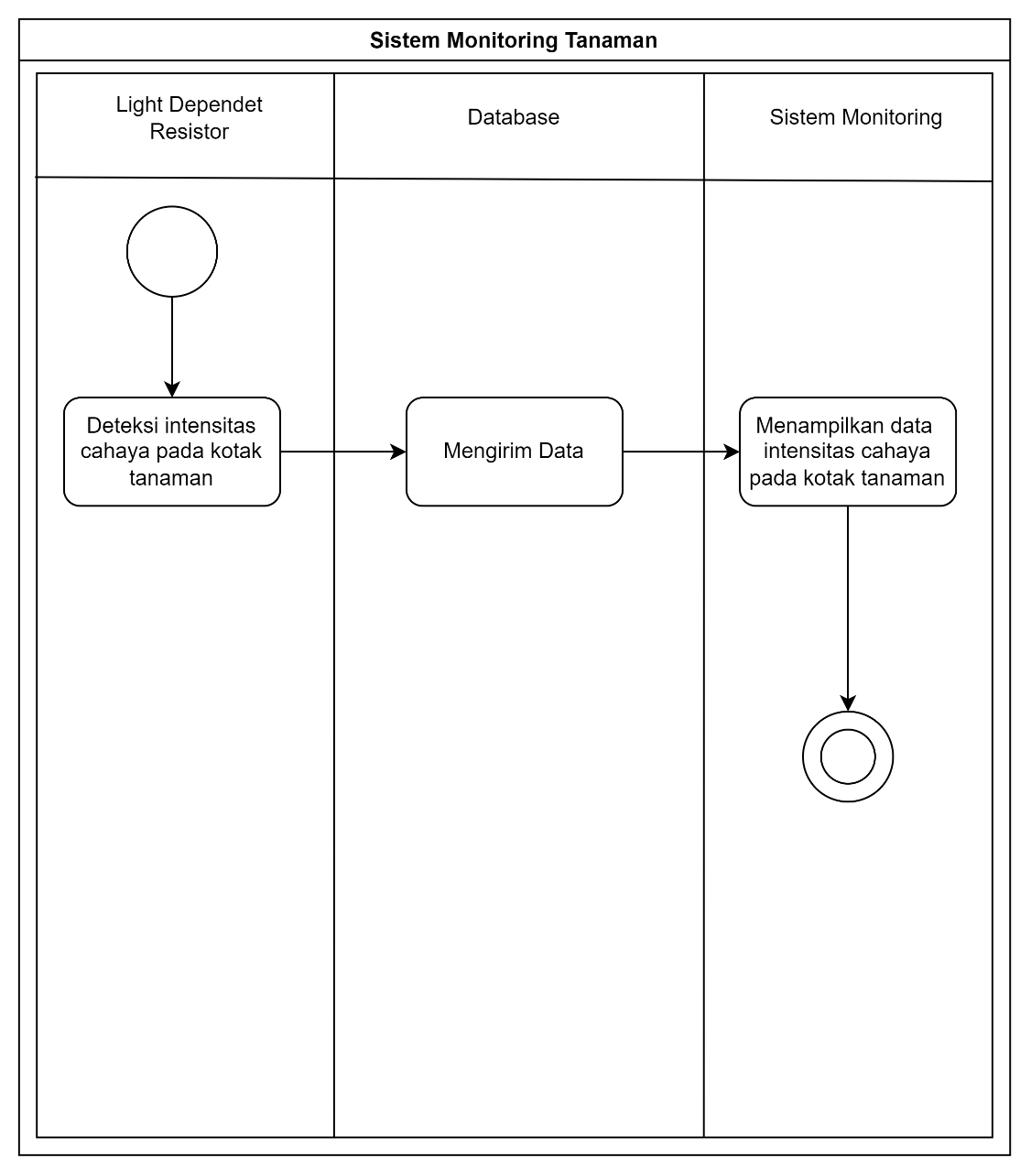
Gambar 3. 16 Activity Diagram UC-09

1. Activity Diagram Mengubah Data Kotak Tanaman



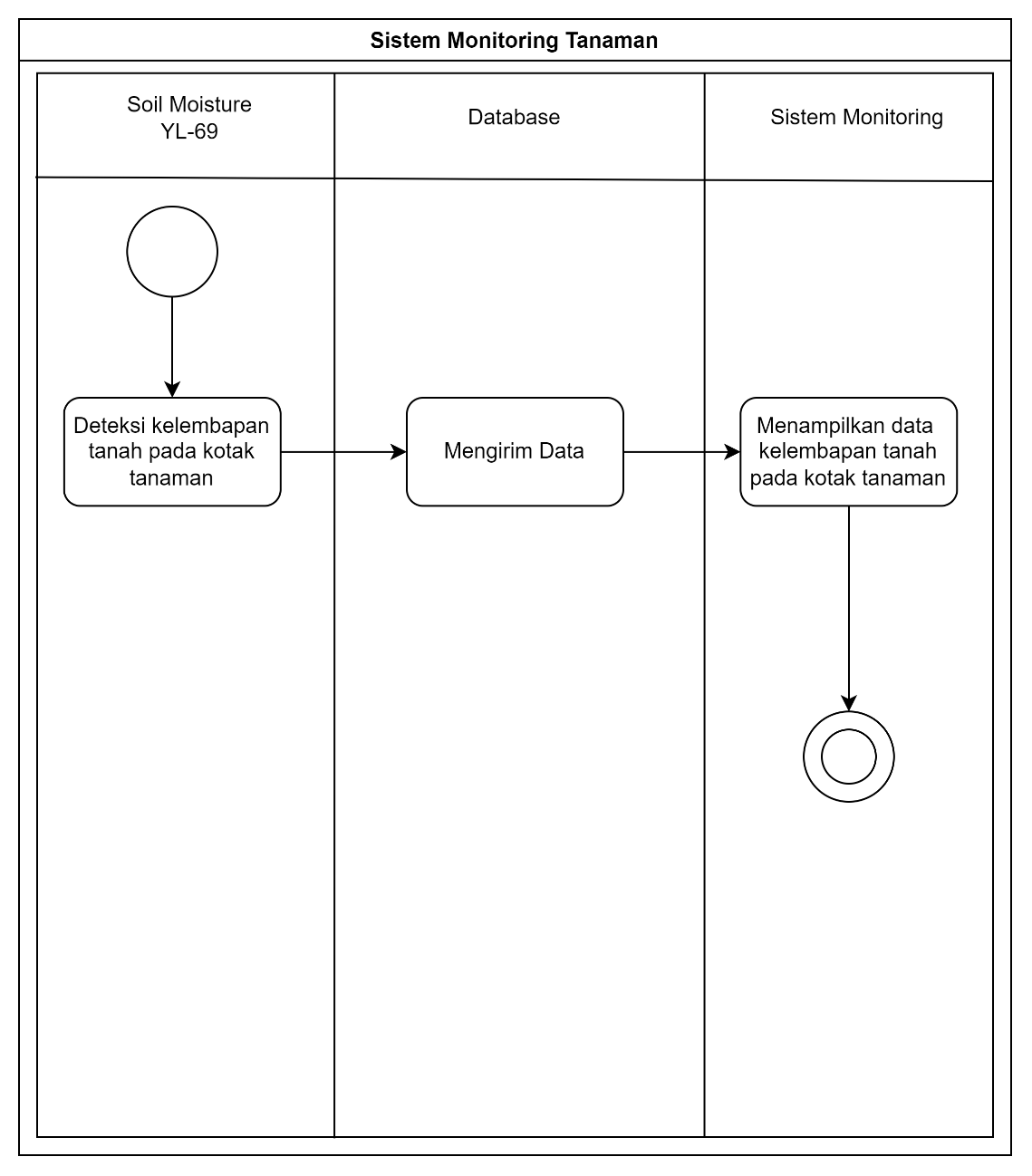
Gambar 3. 17 Activity Diagram UC-10

1. Activity Diagram Mengirim Data Intensitas Cahaya



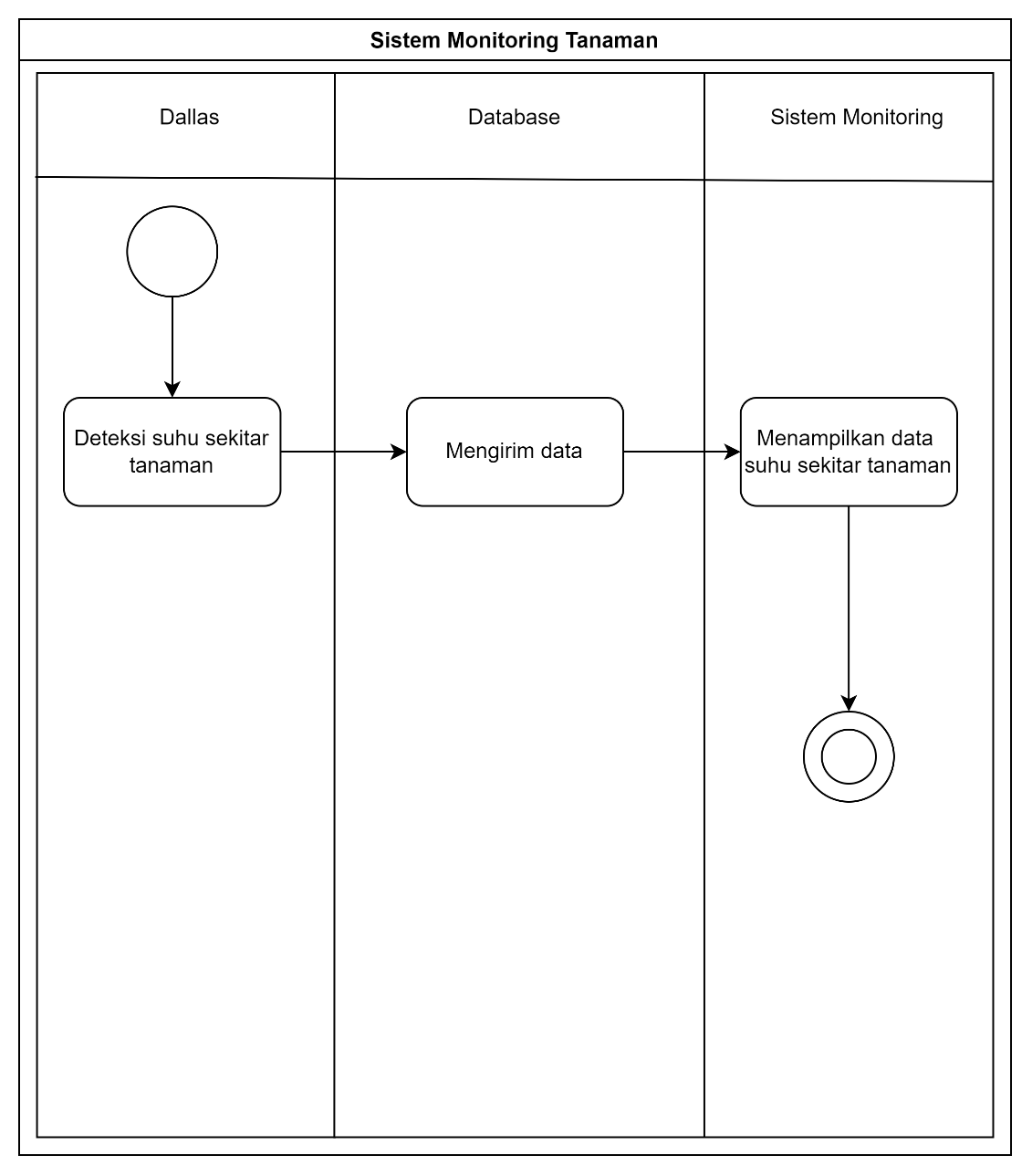
Gambar 3. 18 Activity Diagram UC-11

1. Activity Diagram Mengirim Data Kelembapan Tanah



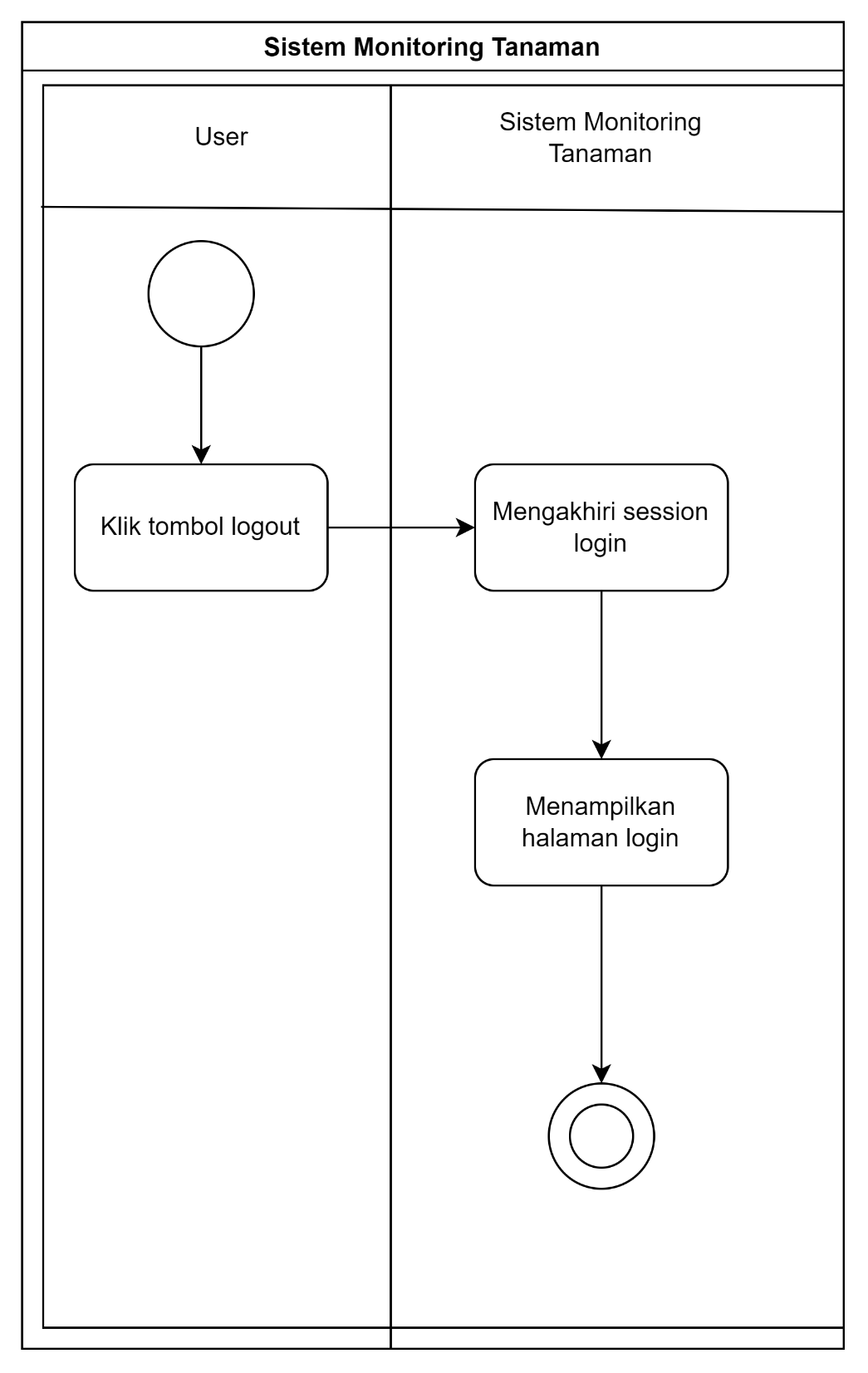
Gambar 3. 19 Activity Diagram UC-12

1. Activity Diagram Mengirim Data Suhu



Gambar 3. 20 Activity Diagram UC-13

1. Activity Diagram Logout



Gambar 3. 21 Activity Diagram UC-14

#### Class Diagram

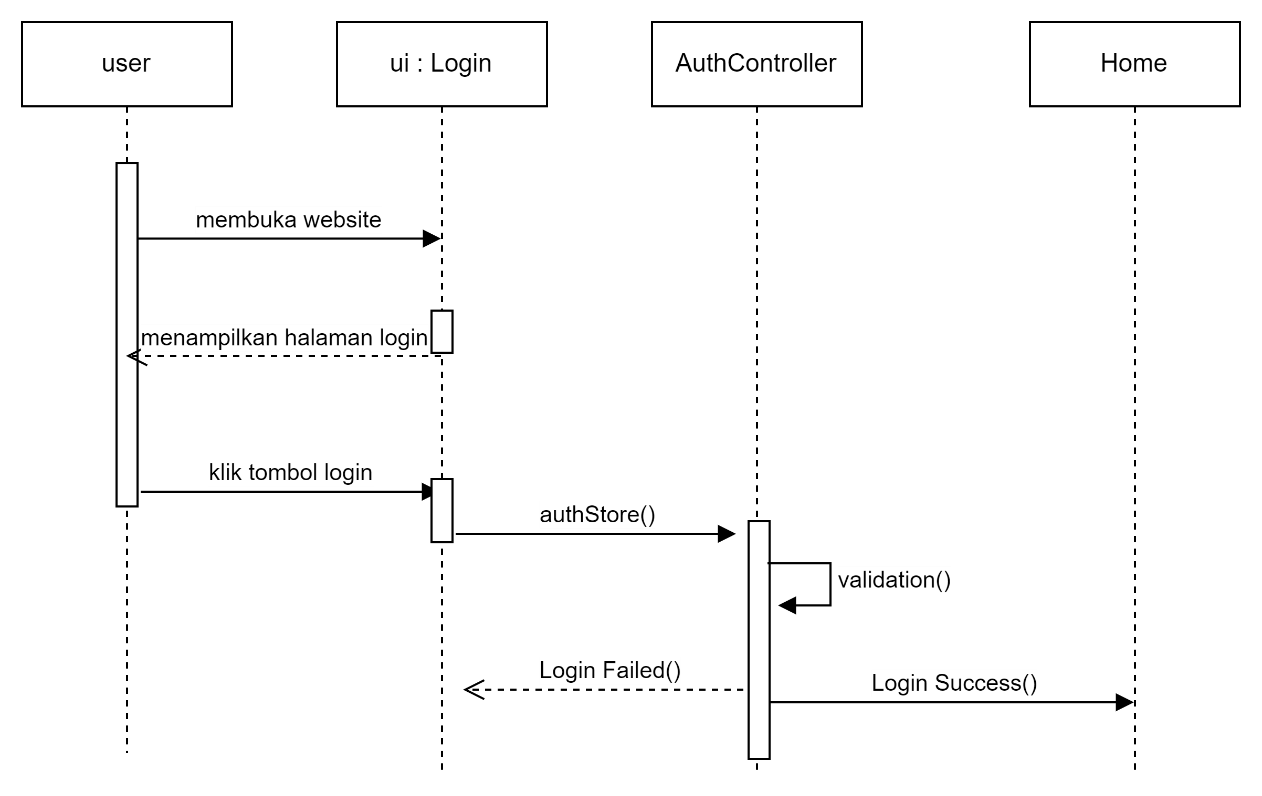
Class Diagram dibuat untuk menunjukan hubungan antar class yang di dalamnya terdapat atribut dan fungsi atau metode dari suatu objek. Berikut adalah diagram kelas sistem yang akan dibangun dapat dilihat pada gambar di bawah ini



Gambar 3. 22 Class Diagram

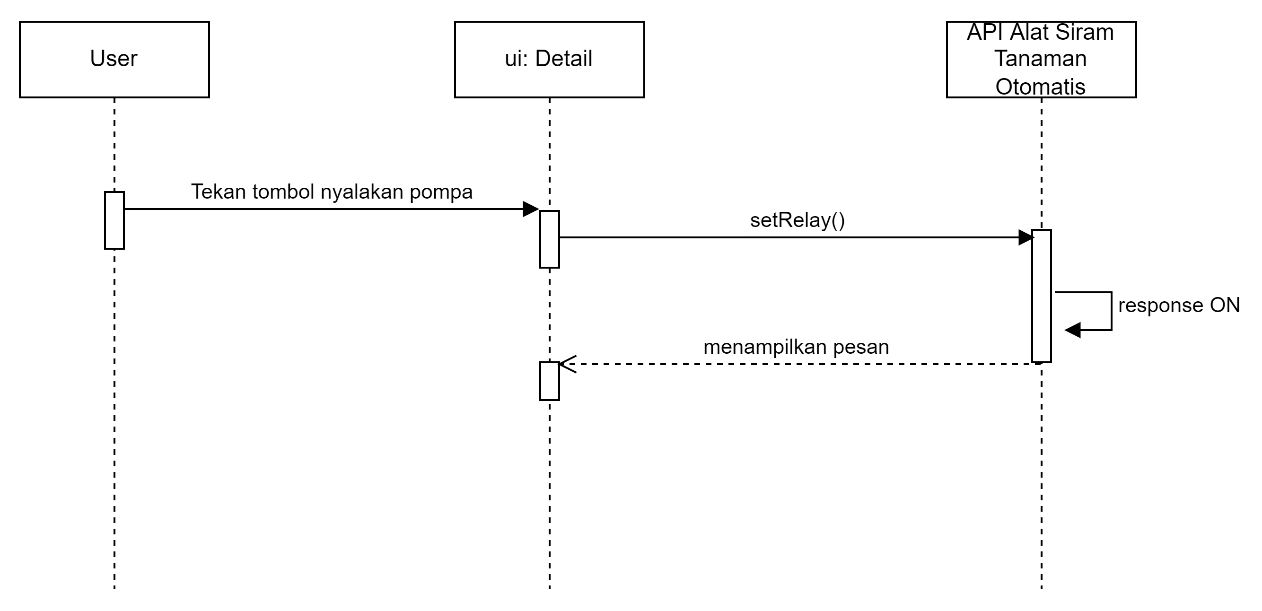
#### Sequence Diagram

Sequence diagram menggambarkan pengaplikasian dari masing-masing skenario use case yang ada pada sistem. Bagaimana sebuah use case dapat berjalan ketika skenario benar terjadi. Untuk sequence diagram yang pertama, yaitu sequence diagram untuk usecase dengan kode UC-01 dapat dilihat pada gambar berikut.



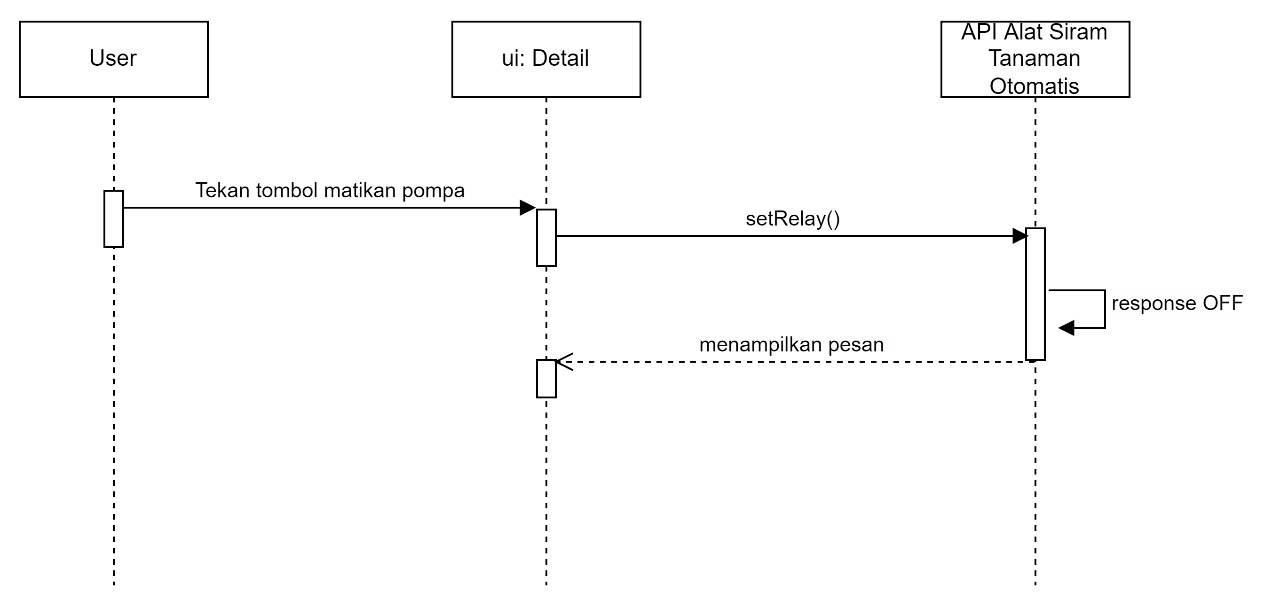
Gambar 3. 23 Sequence Diagram UC-01

Berikutnya sequence diagram untuk use case dengan kode UC-02 dapat dilihat pada gambar berikut



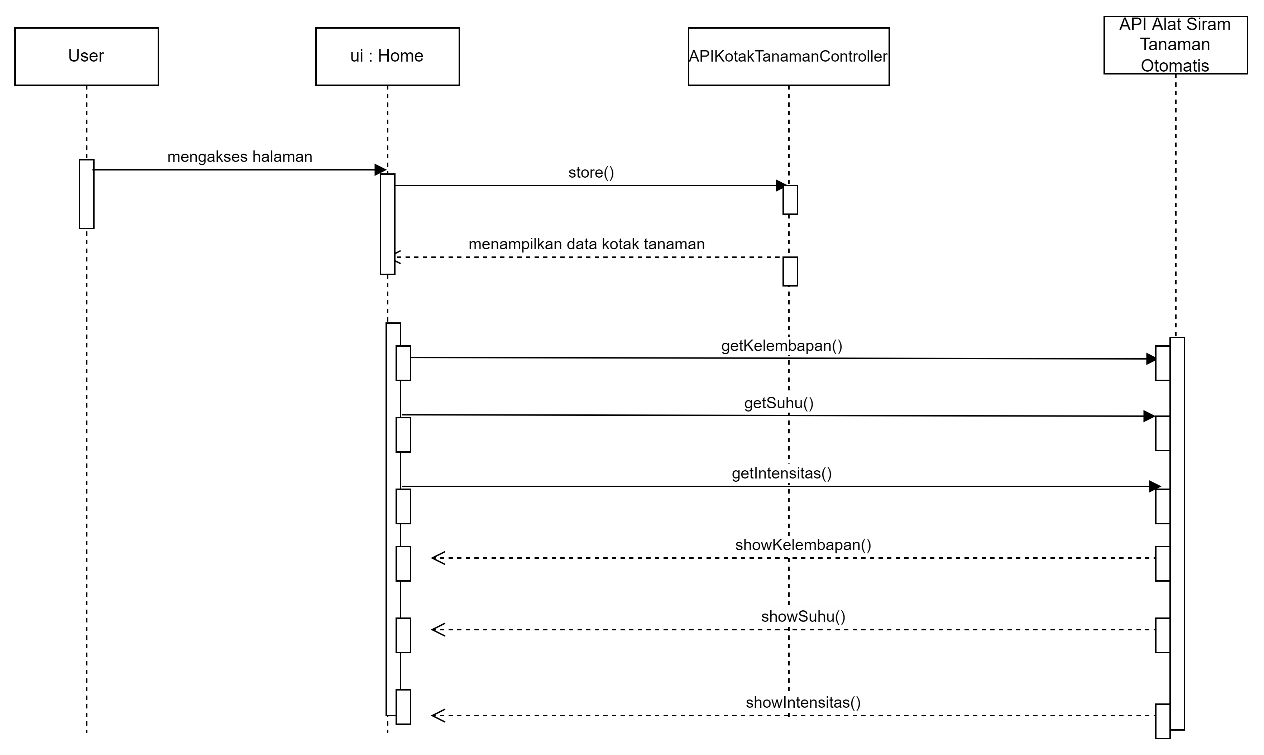
Gambar 3. 24 Sequence Diagram UC-02

Berikutnya sequence diagram untuk use case dengan kode UC-03 dapat dilihat pada gambar berikut



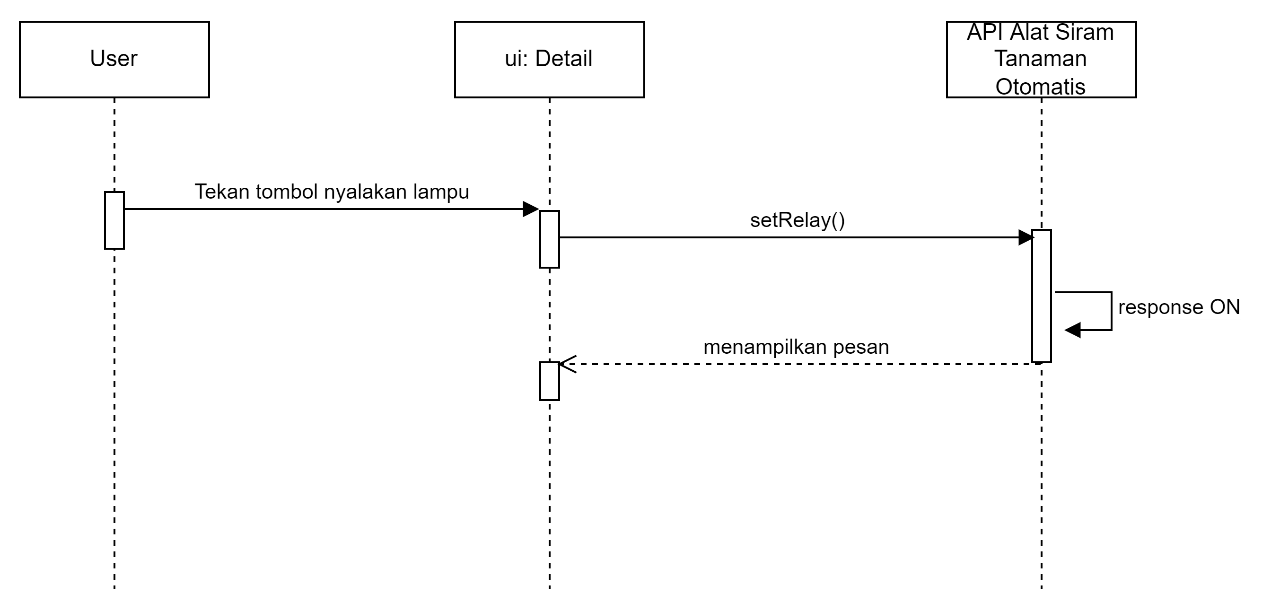
Gambar 3. 25 Sequence Diagram UC-03

Berikutnya sequence diagram untuk use case dengan kode UC-04 dapat dilihat pada gambar berikut



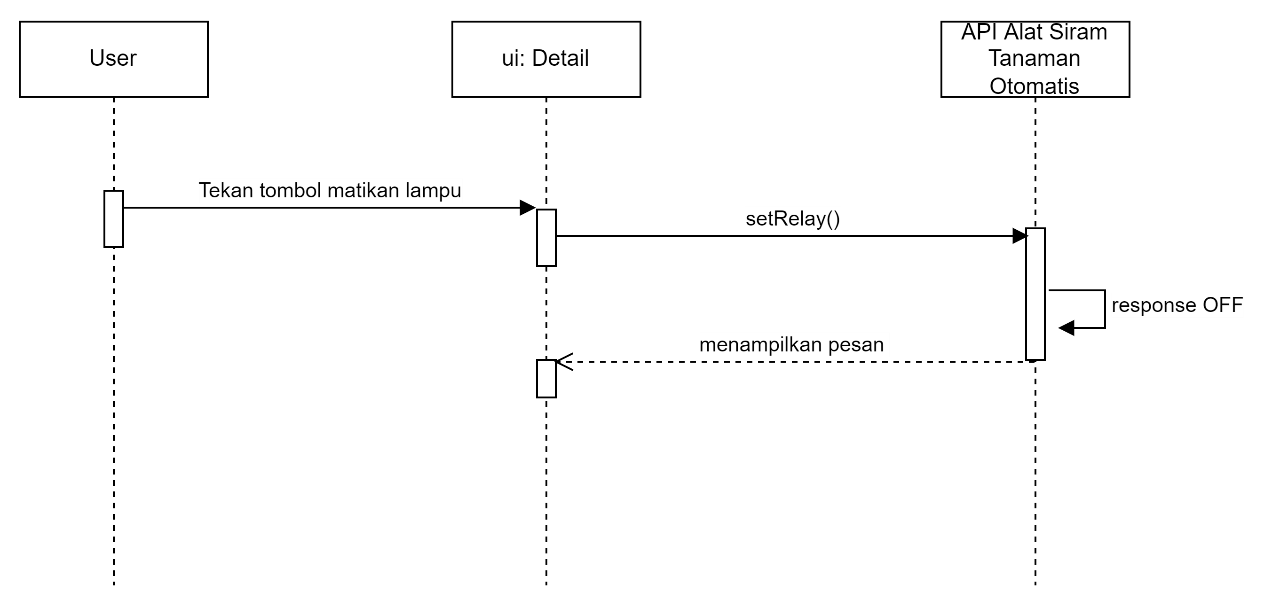
Gambar 3. 26 Sequence Diagram UC-04

Berikutnya sequence diagram untuk use case dengan kode UC-05 dapat dilihat pada gambar berikut



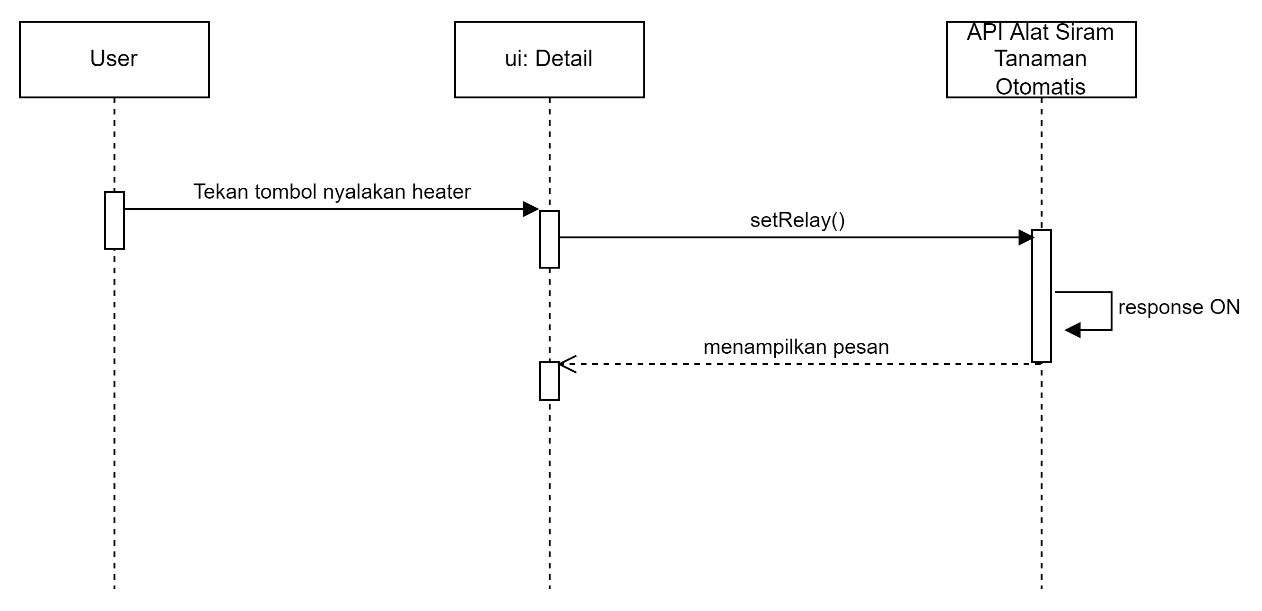
Gambar 3. 27 Sequence Diagram UC-05

Berikutnya sequence diagram untuk use case dengan kode UC-06 dapat dilihat pada gambar berikut



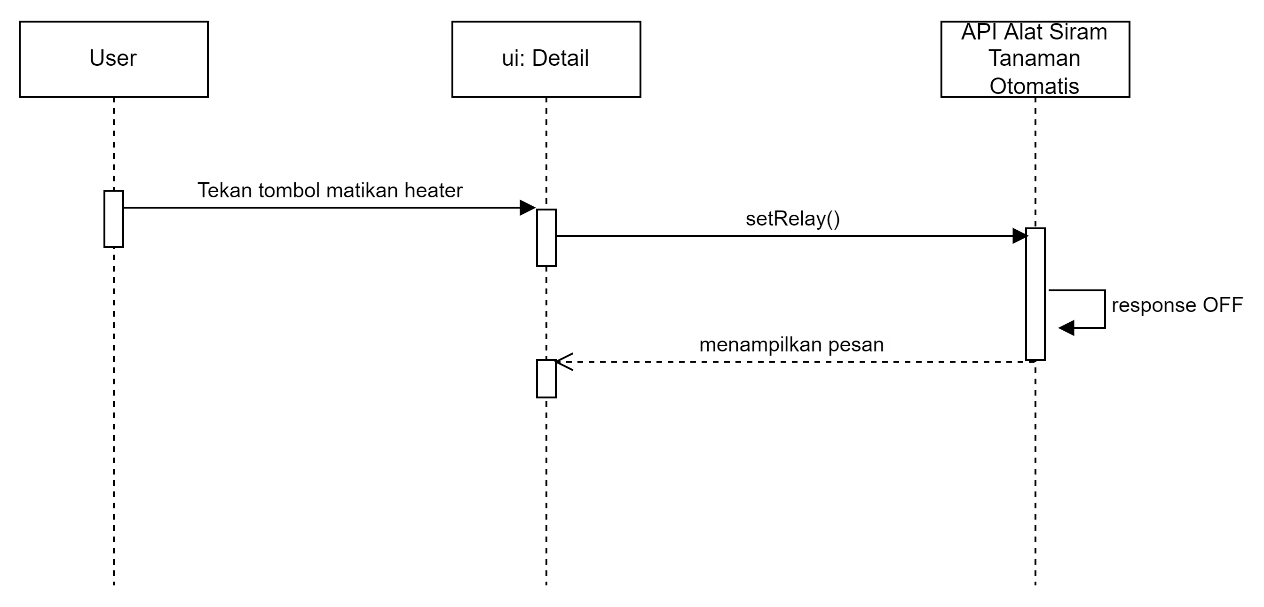
Gambar 3. 28 Sequence Diagram UC-06

Berikutnya sequence diagram untuk use case dengan kode UC-07 dapat dilihat pada gambar berikut



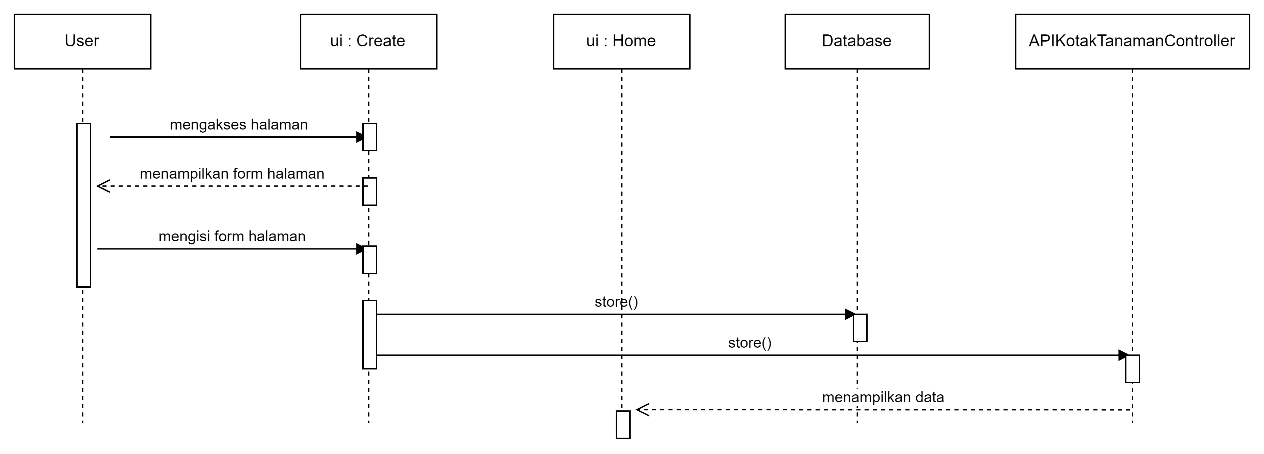
Gambar 3. 29 Sequence Diagram UC-07

Berikutnya sequence diagram untuk use case dengan kode UC-08 dapat dilihat pada gambar berikut



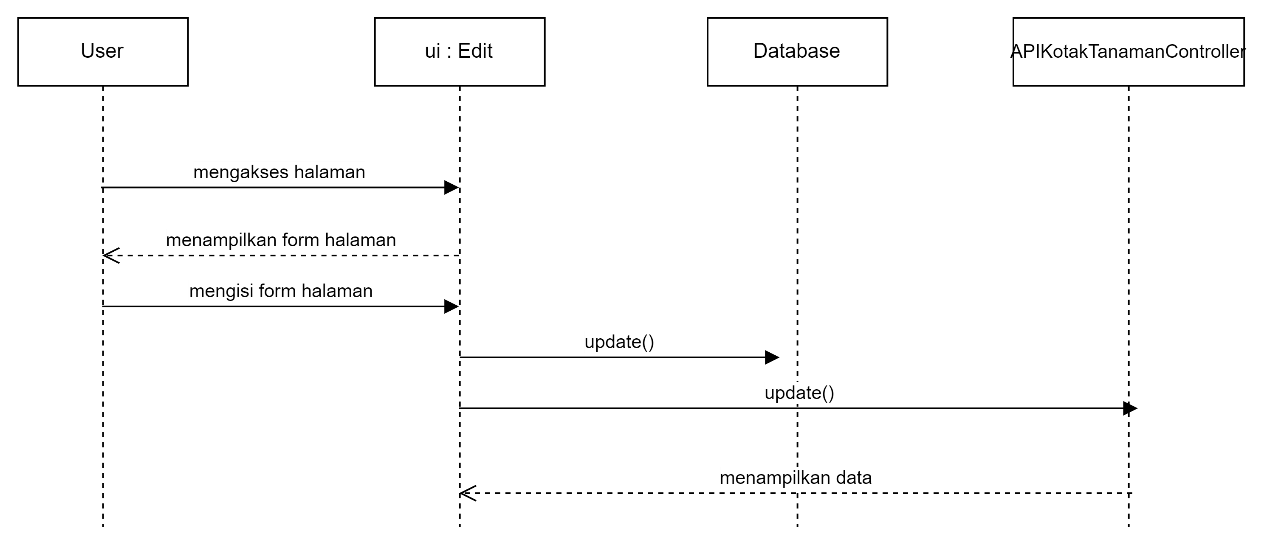
Gambar 3. 30 Sequence Diagram UC-08

Berikutnya sequence diagram untuk use case dengan kode UC-09 dapat dilihat pada gambar berikut



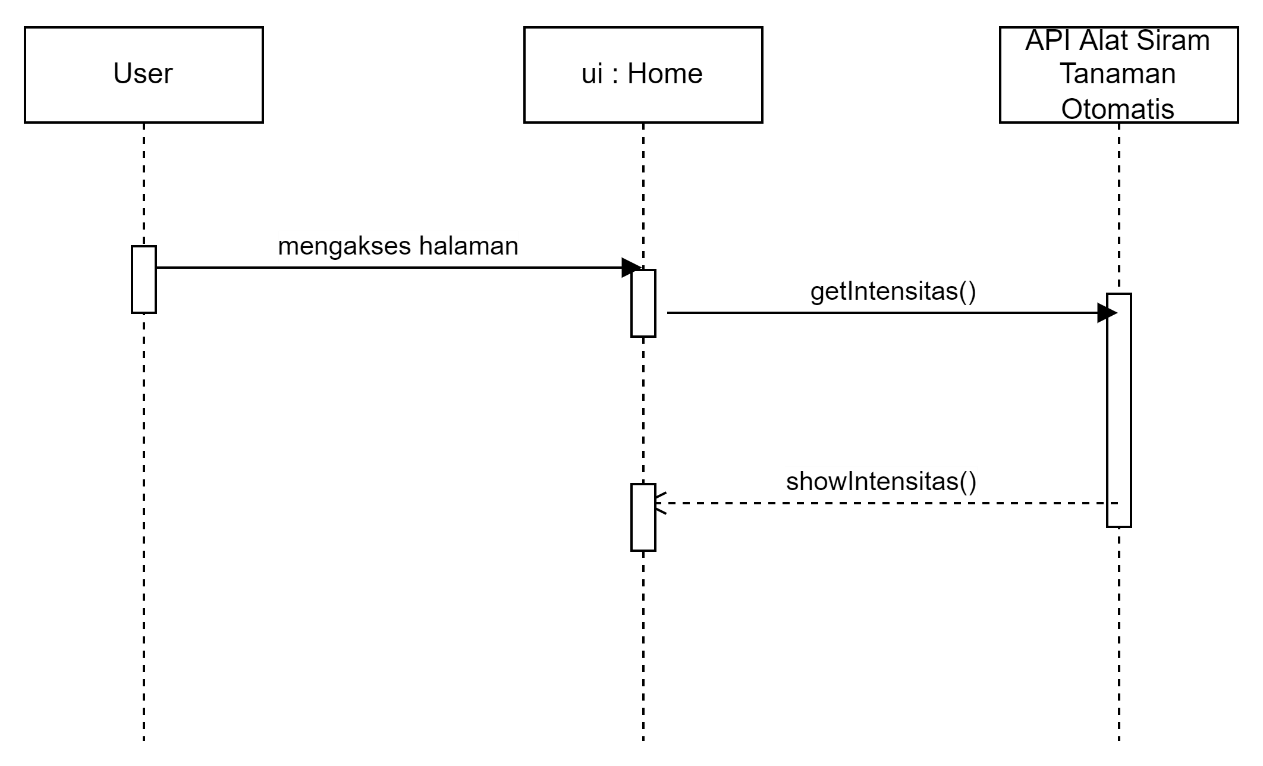
Gambar 3. 31 Sequence Diagram UC-09

Berikutnya sequence diagram untuk use case dengan kode UC-10 dapat dilihat pada gambar berikut



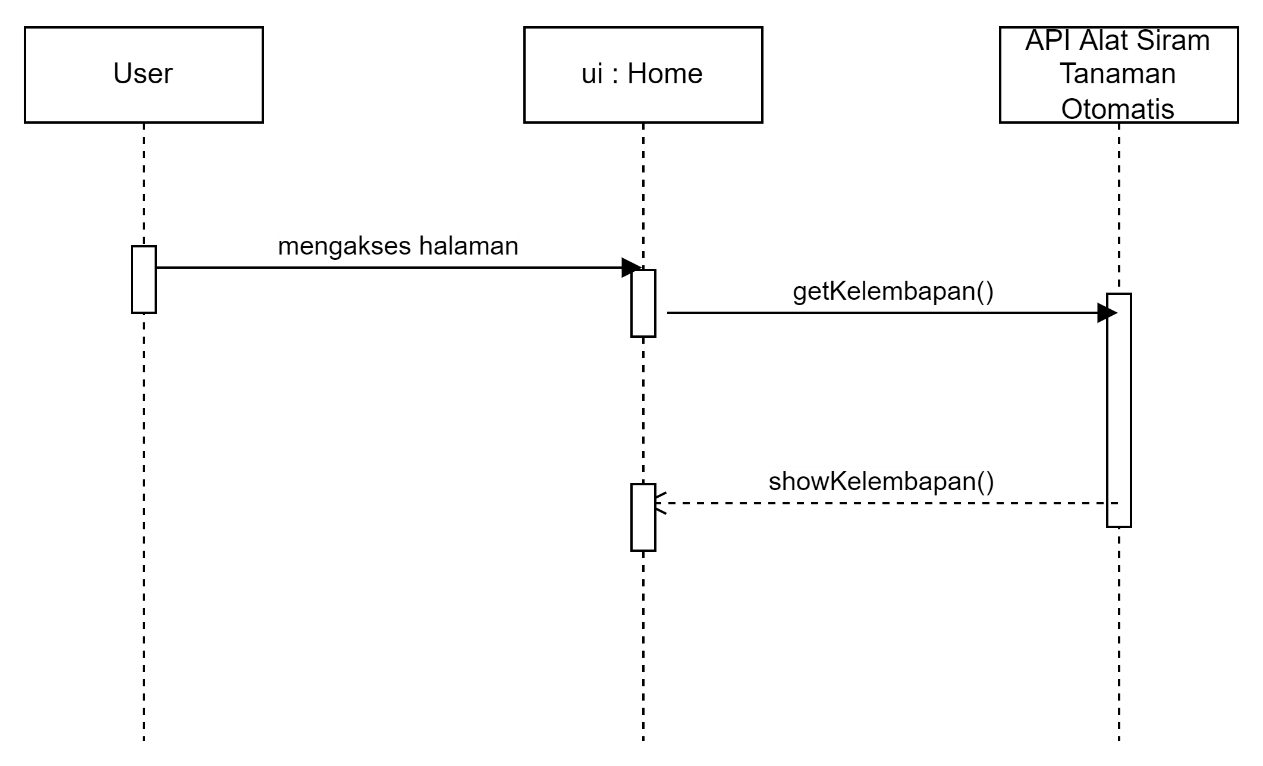
Gambar 3. 32 Sequence Diagram UC-10

Berikutnya sequence diagram untuk use case dengan kode UC-11 dapat dilihat pada gambar berikut



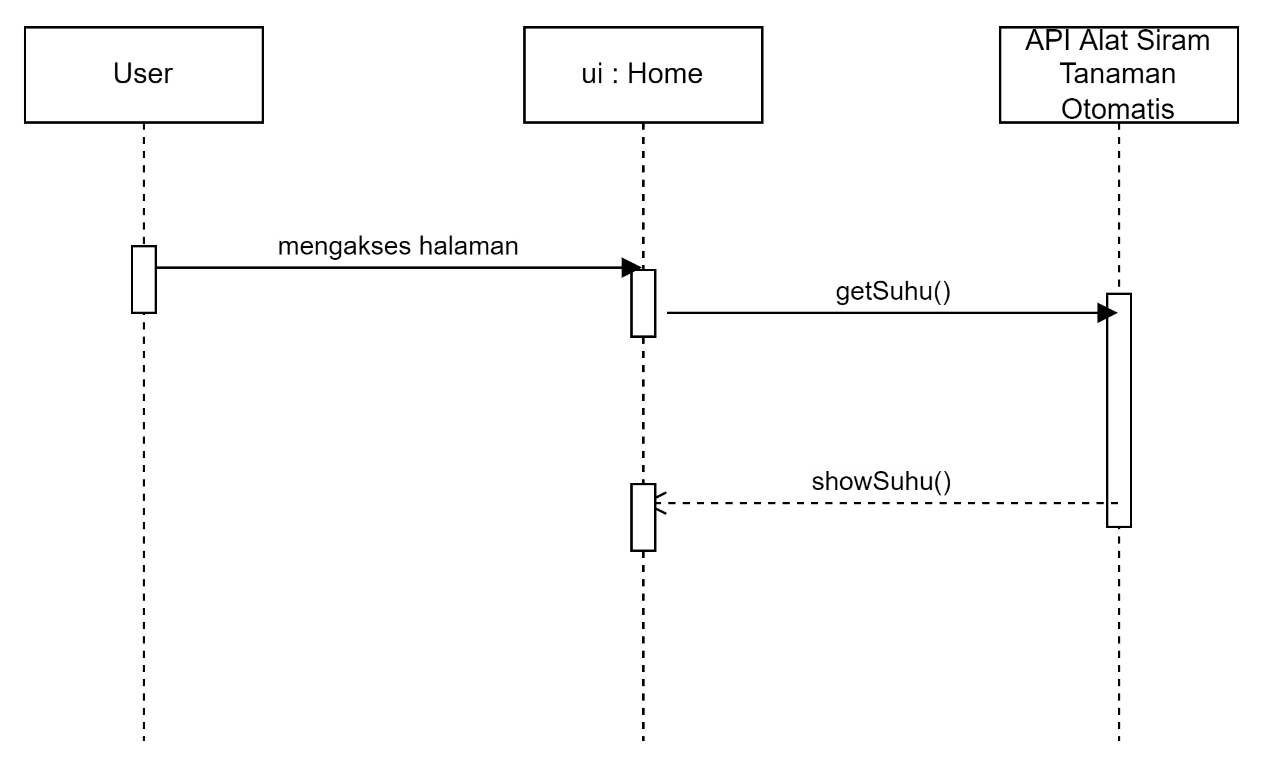
Gambar 3. 33 Sequence Diagram UC-11

Berikutnya sequence diagram untuk use case dengan kode UC-12 dapat dilihat pada gambar berikut



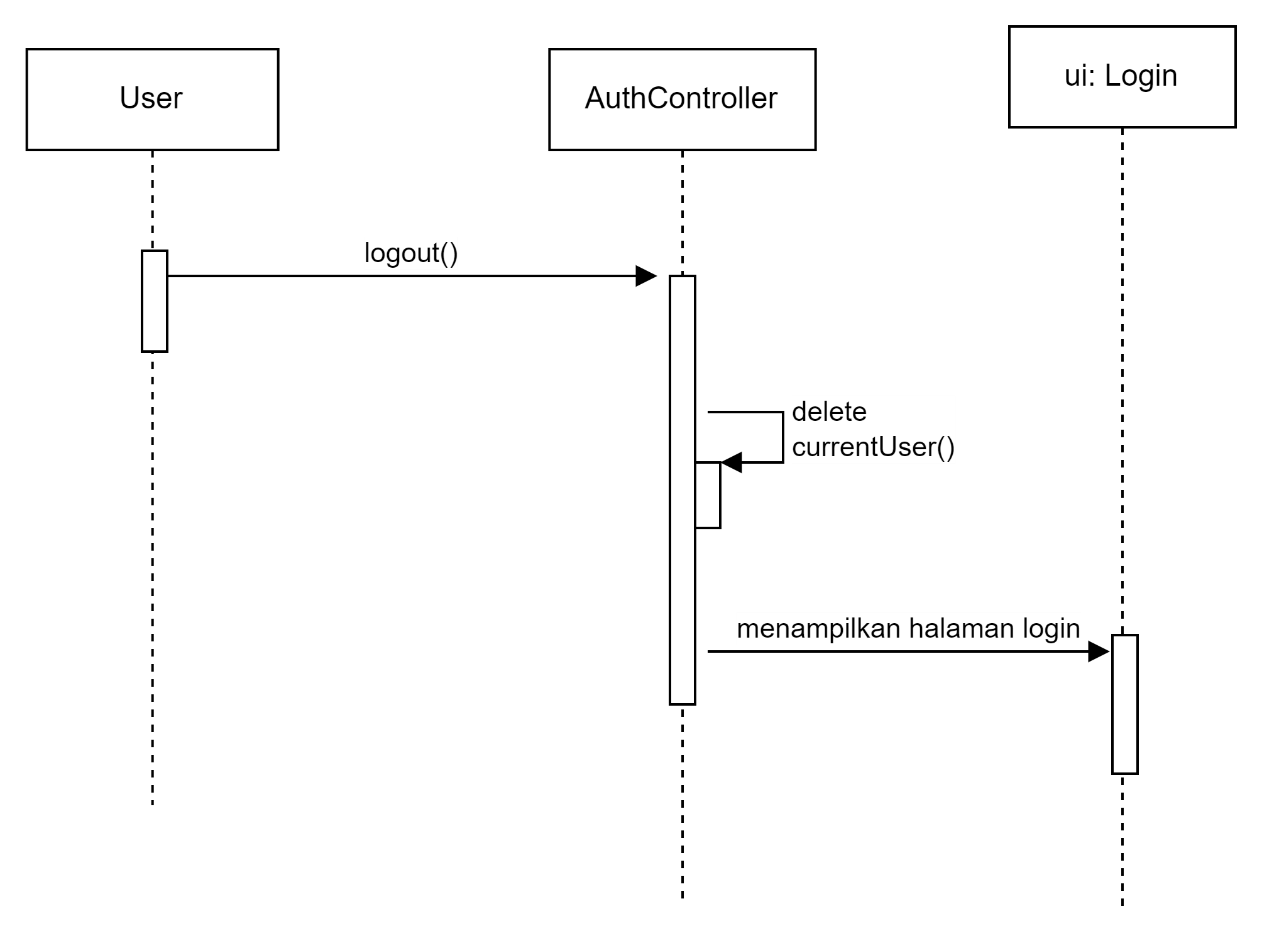
Gambar 3. 34 Sequence Diagram UC-12

Berikutnya sequence diagram untuk use case dengan kode UC-13 dapat dilihat pada gambar berikut



Gambar 3. 35 Sequence Diagram UC-13

Berikutnya sequence diagram untuk use case dengan kode UC-14 dapat dilihat pada gambar berikut



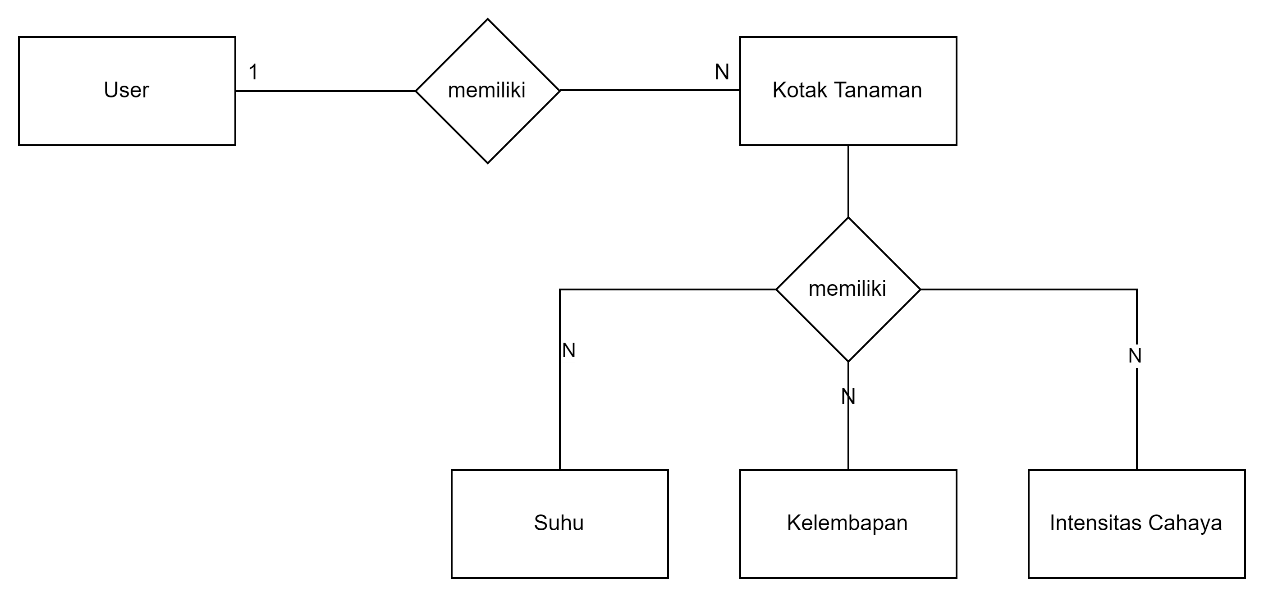
Gambar 3. 36 Sequence Diagram UC-14

### Analisis Basis Data

Setelah berhasil melakukan analisis kebutuhan fungsional, maka tahap analisis basis data bisa dilakukan. Tahapan ini merupakan tahapan untuk menganalisis bagaimana nantinya skema basis data yang akan dibuat.

#### Skema ERD

Skema ERD dibuat dengan bertujuan untuk memudahkan peneliti dalam merancang basis data. Skema ERD pun dibuat berdasarkan dari data – data yang diperlukan untuk memberikan informasi – informasi tanaman kepada pengguna sistem. Untuk lebih jelasnya, bisa dilihat pada gambar berikut



Gambar 3. 37 Entity Relationship Diagram

Gambar diatas hanya menunjukan nama tabel – tabelnya saja, tidak dengan atributnya apa saja. Berikut adalah tabel 3.57 merupakan tabel detail atribut - atribut dari setiap tabel pada gambar ERD 3.63.

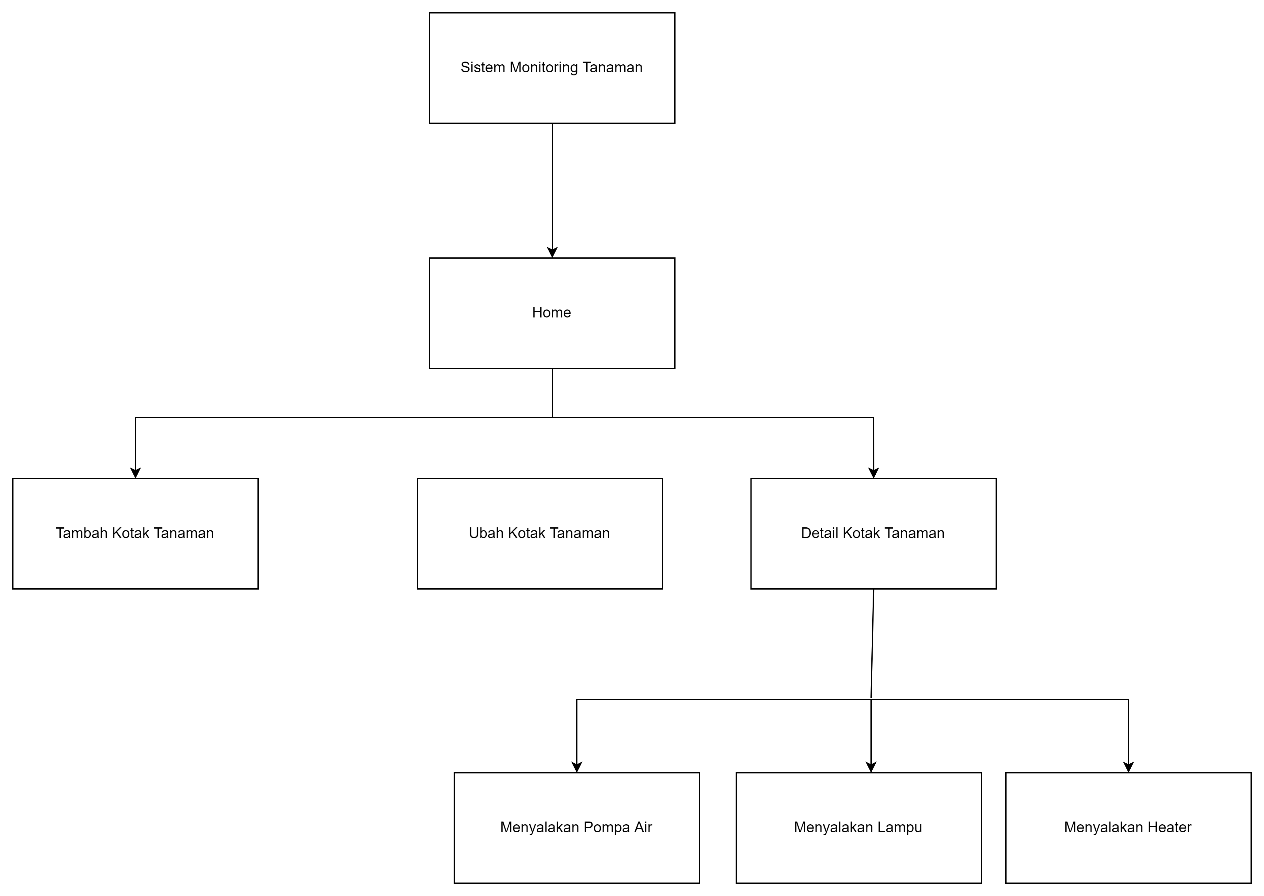
Tabel 3. 24 Atribut Entity Relationship Diagram

|  |  |
| --- | --- |
| Nama Tabel | Atribut |
| Kotaktanamans | Id\_kotak  nama\_kotak  isi\_tanaman  catatan |
| Cahayas | id  id\_kotak  cahaya |
| Kelembapans | id  id\_kotak  kelembapan |
| Suhus | id  id\_kotak  suhu |
| User | username  password  nama |

## Perancangan

### Perancangan Menu

Sistem yang akan dibuat nantinya akan memiliki menu Home sebagai menu utama yang akan ditampilkan setelah login. Di menu Home tersebut, nantinya ada sub-menu lagi seperti menu notifikasi, menu detail tanaman, menu tambah tanaman, dan menu yang lainnya. Untuk lebih jelasnya bisa dilihat pada gambar berikut.

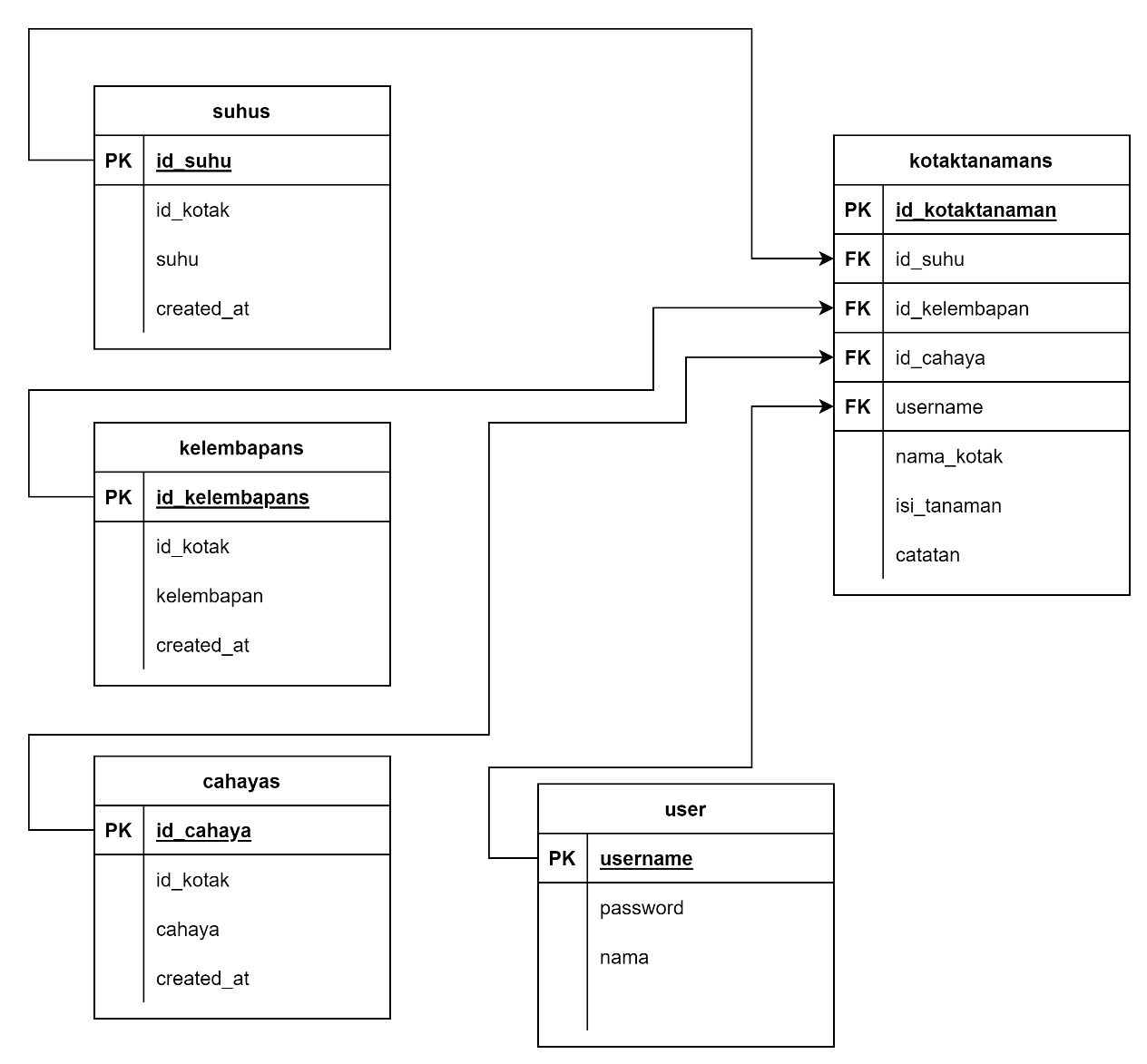


Gambar 3. 38 Perancangan Menu

### Perancangan Basis Data

#### Skema Tabel Relasional

Setelah selesai merancang diagram ERD, selanjutnya penelitian merancang diagram tabel relasional. Diagram ini mirip seperti ERD, tetapi kini ada kumpulan nama – nama variabel di setiap tabelnya. Diagram ini nantinya untuk membantu peneliti dalam menentukan struktur di setiap tabel, seperti menentukan foreign key, ukuran variabel dan tipe data variabel. Untuk lebih jelasnya, dapat dilihat pada gambar berikut.



Gambar 3. 39 Skema Tabel Relasional

#### Struktur Tabel

Untuk menjelaskan lebih jelas mengenai isi tabel, seperti tipe data variabelnya, ukuran variabel, dan keterangan lainnya, dapat dilihat pada struktur tabel. Untuk struktur tabel tanaman, dapat dilihat pada tabel berikut.

Tabel 3. 25 Struktur Tabel Kotak Tanaman

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Nama Data Item | Type | Size | Keterangan |
| 1 | id\_kotaktanaman | Integer | 13 | PRIMARY KEY,  NOT NULL |
| 2 | id\_suhu | Integer | 13 | FOREIGN KEY, NOT NULL |
| 3 | id\_cahaya | Integer | 13 | FOREIGN KEY, NOT NULL |
| 4 | id\_kelembapan | Integer | 13 | FOREIGN KEY, NOT NULL |
| 5 | nama\_kotak | Varchar | 255 | NOT NULL |
| 6 | isi\_tanaman | Varchar | 255 | NOT NULL |
| 7 | catatan | Varchar | 255 | NOT NULL |

Untuk struktur table suhu, dapat dilihat pada table berikut

Tabel 3. 26 Struktur Tabel Suhu

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Nama Data Item | Type | Size | Keterangan |
| 1 | id | Integer | 13 | PRIMARY KEY,  NOT NULL |
| 2 | id\_kotak | Varchar | 255 | NOT NULL |
| 3 | suhu | Int | 13 | NULLABLE |

Untuk struktur table cahaya, dapat dilihat pada table berikut

Tabel 3. 27 Struktur Tabel Cahayas

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Nama Data Item | Type | Size | Keterangan |
| 1 | id | Integer | 13 | PRIMARY KEY,  NOT NULL |
| 2 | id\_kotak | Varchar | 255 | NOT NULL |
| 3 | cahaya | Int | 13 | NULLABLE |

Untuk struktur table kelembapan, dapat dilihat pada table berikut

Tabel 3. 28 Struktur Tabel Kelembapans

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Nama Data Item | Type | Size | Keterangan |
| 1 | id | Integer | 13 | PRIMARY KEY,  NOT NULL |
| 2 | id\_kotak | Varchar | 255 | NOT NULL |
| 3 | kelembapan | Int | 13 | NULLABLE |

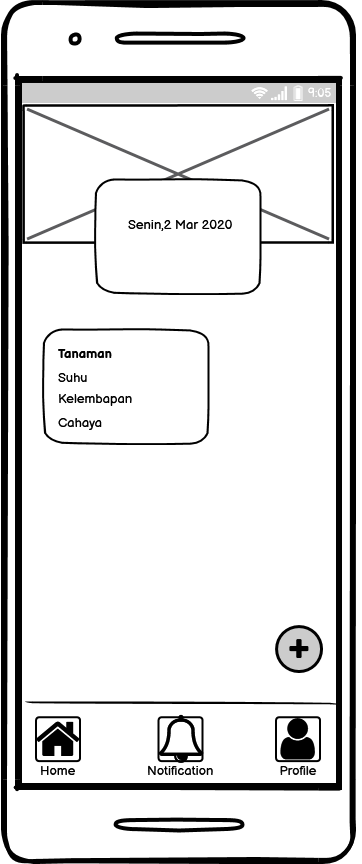
Untuk struktur table user, dapat dilihat pada table berikut

Tabel 3. 29 Struktur Tabel Users

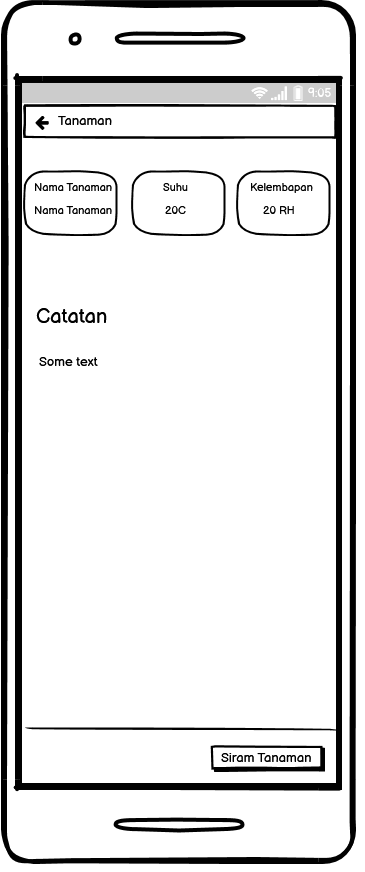
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Nama Data Item | Type | Size | Keterangan |
| 1 | username | varchar | 255 | PRIMARY KEY,  NOT NULL |
| 2 | password | Varchar | 255 | NOT NULL |
| 3 | nama | Varchar | 255 | NOT NULL |

### Perancangan Antarmuka

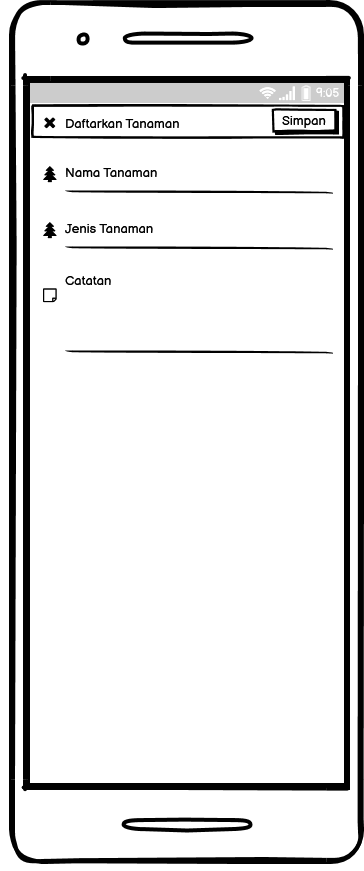
Perancangan antarmuka merupakan tahapan yang sangat penting. Tahapan ini merupakan tahapan mengimplementasikan hasil dari tahapan analisis ke dalam bentuk mockup antarmuka sistem. Berikut adalah rancangan antarmuka dari sistem ini. Gambar pertama dibawah, merupakan tampilan antarmuka dengan kode F01



Gambar 3. 40 Tampilan Antarmuka Halaman Utama



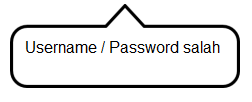
Gambar 3. 41 Tampilan Antarmuka Halaman Detail



Gambar 3. 42 Tampilan Antarmuka Halaman Tambah Tanaman

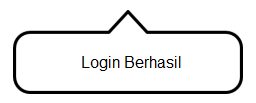
### Perancangan Pesan

Perancangan pesan merupakan tahapan untuk merancang pesan apa saja yang akan ditampilkan oleh sistem kepada pengguna ketikan sedang menggunakan sistem. Pesan yang keluar dapat berupa pesan konfirmasi, pesan sukses, pesan gagal, atau sekedar pesan informasi saja. Untuk perancangan pesan yang pertama dengan kode P01, dapat dilihat pada gambar 3.93 berikut.



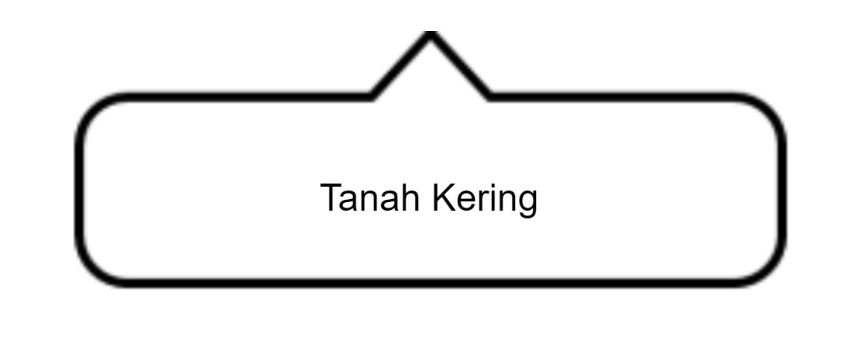
Gambar 3. 43 Perancangan Pesan Kesalahan Username / Password

Untuk perancangan pesan berikutnya dengan kode P02, dapat dilihat pada gambar 3.94 berikut



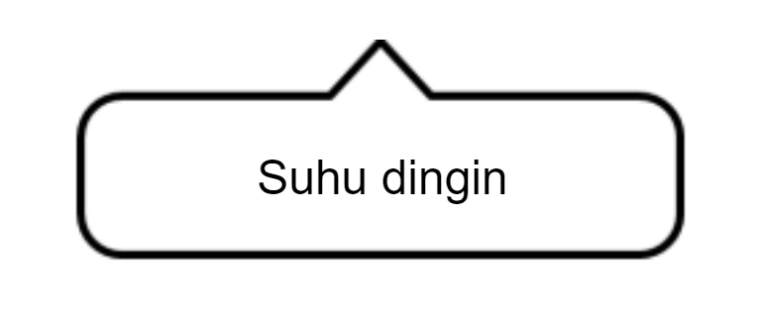
Gambar 3. 44 Perancangan Pesan Login Berhasil

Untuk perancangan pesan berikutnya dengan kode P02, dapat dilihat pada gambar 3.94 berikut



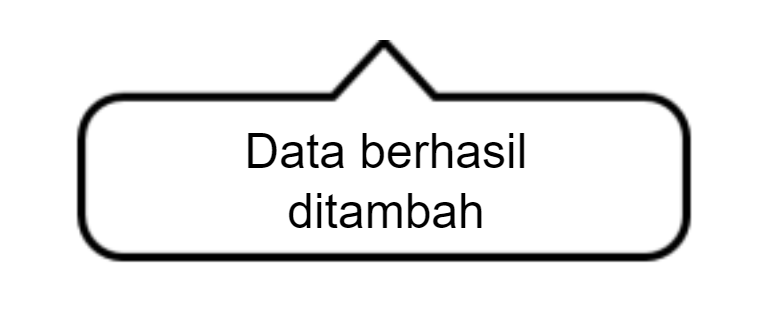
Gambar 3. 45 Perancangan Pesan Ketika Tanah Kering

Untuk perancangan pesan berikutnya dengan kode P02, dapat dilihat pada gambar 3.94 berikut



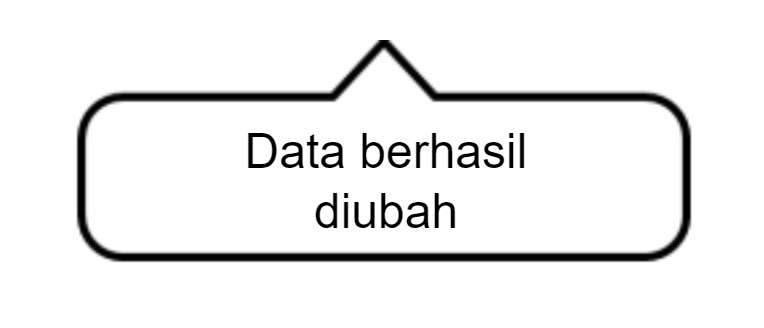
Gambar 3. 46 Perancangan Pesan Ketika Suhu Dingin

Untuk perancangan pesan berikutnya dengan kode P02, dapat dilihat pada gambar 3.94 berikut



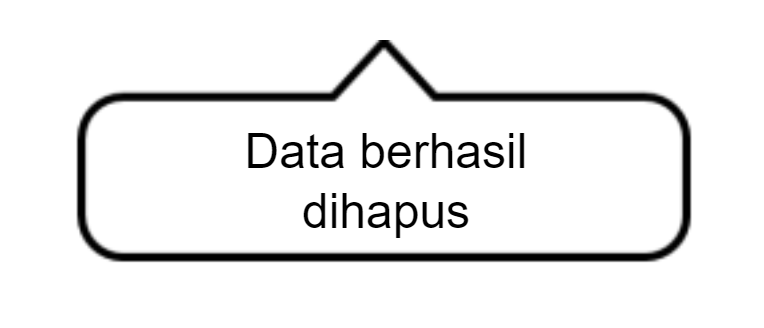
Gambar 3. 47 Perancangan Pesan Tanmbah Tanaman

Untuk perancangan pesan berikutnya dengan kode P02, dapat dilihat pada gambar 3.94 berikut



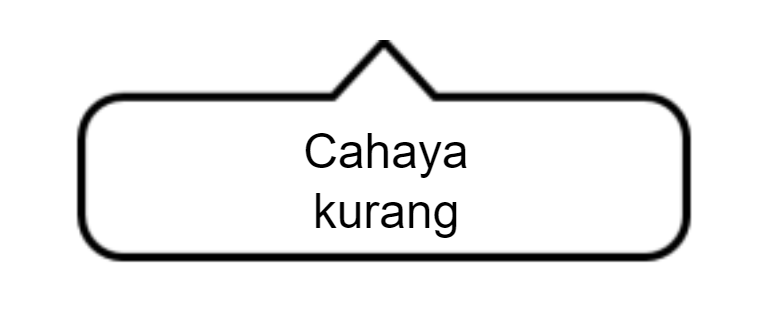
Gambar 3. 48 Perancangan Pesan Ubah Tanaman

Untuk perancangan pesan berikutnya dengan kode P02, dapat dilihat pada gambar 3.94 berikut



Gambar 3. 49 Perancangan Pesan Hapus Tanaman

Untuk perancangan pesan berikutnya dengan kode P02, dapat dilihat pada gambar 3.94 berikut



Gambar 3. 50 Perancangan Pesan Cahaya Kurang

# BAB IV IMPLEMENTASI DAN PENGUJIAN

## Implementasi

Implementasi merupakan tahap menerjemahkan perancangan pada tahap analisis dan perancangan. Tujuan implementasi adalah untuk mengkonfirmasi sistem yang telah dirancang sehingga pengguna dapat memberikan masukkan kepada peneliti. Implementasi sistem menjelaskan tentang kebutuhan perangkat lunak peneliti, perangkat keras peneliti, implementasi antarmuka, dan implementasi basis data.

### Perangkat Lunak Peneliti

Untuk perangkat lunak pada laptop yang peneliti pakai selama melakukan pembangunan dan pengujian sistem, dapat dilihat pada tabel 4.1 berikut

Tabel 4. 1 Perangkat Lunak Peneliti

|  |  |  |
| --- | --- | --- |
| No | Nama | Spesifikasi |
| 1 | Sistem Operasi | Windows 10 Pro 64 Bit |
| 2 | *Tools* Yang Digunakan | Visual Studio Code |
| Draw.io |
| Balsamiq Mockup |
| MySQL |
| 3 | Browser | Google Chrome |

### Perangkat Keras Peneliti

Untuk perangkat keras pada laptop yang peneliti pakai selama melakukan pembangunan dan pengujian sistem, dapat dilihat pada tabel 4.2 berikut.

Tabel 4. 2 Perangkat Keras Peneliti

|  |  |  |
| --- | --- | --- |
| No | Nama Item | Spesifikasi |
| 1 | CPU | Intel Core i3-5005U |
| 2 | Memory | Single Channel DDR3 2GB |
| 3 | Storage | 500GB |
| 4 | GPU | Intel HD Graphic 5500 |

### Implementasi Antarmuka

Implementasi antarmuka menjelaskan tentang implementasi antarmuka sistem pada aplikasi. Implementasi antarmuka diwakili dengan nama 193 antarmuka beserta nama filenya. Implementasi antarmuka sistem dapat dilihat pada tabel 4.3 berikut.

Tabel 4. 3 Implementasi Antarmuka

|  |  |  |
| --- | --- | --- |
| No | Nama Antarmuka | Nama File |
| 1 | Halaman Login | Login.vue |
| 2 | Halaman Utama | Home.vue |
| 3 | Halaman Detail | Detail.vue |
| 4 | Halaman Tambah Kotak Tanaman | Create.vue |
| 5 | Halaman Edit Kotak Tanaman | Edit.vue |

### Implementasi Kelas

Implementasi kelas menggambarkan implementasi dari diagram kelas yang terbentuk dari usecase ke dalam bentuk file dengan ekstensi .php. Berikut adalah implementasi kelas dapat dilihat pada tabel 4.4 berikut.

Tabel 4. 4 Implementasi Kelas

|  |  |  |
| --- | --- | --- |
| No | Nama Kelas | Nama File |
| 1 | Login | AuthController.php |
| 2 | Halaman Utama | KotaktanamanController.php |
| 3 | Halaman Detail | KotaktanamanController.php |
| 4 | Halaman Tambah Tanaman | KotaktanamanController.php |
| 5 | Halaman Edit Tanaman | KotaktanamanController.php |
| 6 | Suhu | SuhuController.php |
| 7 | Cahaya | CahayaController.php |
| 8 | Kelembapan | KelembapanController.php |

### Implementasi Basis Data

Implementasi basis data merupakan implementasi rancangan basis data yang sudah dirancang sebelumnya yang merujuk kepada skema relasinya dan diimplementasikan menggunakan MySQL dengan source code untuk setiap tabel dapat dilihat pada tabel 4.5 berikut.

Tabel 4. 5 Implementasi Basis Data

|  |  |  |
| --- | --- | --- |
| No | Nama Tabel | Query SQL |
| 1 | kotaktanamans | CREATE TABLE Tanaman (  id\_tanaman int NOT NULL PRIMARY KEY,  id\_section int NOT NULL,  username varchar(255) NOT NULL,  nama\_tanaman varchar(255) NOT NULL,  jenis\_tanaman varchar(255) NOT NULL,  catatan varchar(255) NOT NULL,  suhu int,  intensitas\_cahaya int,  kelembapan\_tanah int  ); |
| 2 | cahayas | CREATE TABLE cahayas(  id int NOT NULL PRIMARY KEY,  id\_kotak int (11) NOT NULL,  cahaya int (11) NULLABLE,  created\_at timestamp  ); |
| 3 | Suhus | CREATE TABLE suhus(  id int NOT NULL PRIMARY KEY,  id\_kotak int (11) NOT NULL,  suhu int (11) NULLABLE,  created\_at timestamp  ); |
| 4 | kelembapans | CREATE TABLE kelembapans(  id int NOT NULL PRIMARY KEY,  id\_kotak int (11) NOT NULL,  kelembapan int (11) NULLABLE,  created\_at timestamp  ); |
| 5 | user | CREATE TABLE User (  username varchar(50) NOT NULL PRIMARY KEY,  password varchar(50) NOT NULL,  nama varchar(255) NOT NULL  ); |

### Implementasi Teknologi

Implementasi teknologi adalah pembahasan hasil dari beberapa teknologi yang digunakan dalam pembangunan sistem. Adapun teknologi yang digunakan beserta implementasinya adalah sebagai berikut ini.

#### Implementasi Axios

Implementasi Axios pada pembangunan sistem ini digunakan untuk membuat sistem dapat terhubung dengan sistem *backend* dan terhubung dengan API yang digunakan. Dengan menggunakan Axios, maka sistem yang dibangun akan dapat menerima data dari API yang digunakan. Ketika sistem berhasil terhubung dengan API, maka selanjutnya akan dipilih data – data apa saja yang akan diambil dari API. Semua itu bisa dilakukan dengan cara mengimplementasikan Axios kedalam sistem pada penelitian. Langkah pertama dalam mengimplementasikan Axios ke API dalam sistem, adalah dengan menginstall plugin Axios pada sistem yang dibangun. Adapun *syntax* yang harus dimasukkan untuk menginstall plugin Axios pada sistem, dapat dilihat pada gambar berikut



Setelah menginstall plugin Axios pada sistem, langkah berikutnya yaitu mengimplementasikan kode Javascript untuk Axios agar bisa digunakan dalam sistem. Kode Javascript untuk dihubungkan di sistem terdapat pada halaman Index.js yang terdapat pada *Router*. Untuk kode implementasinya, dapat dilihat pada skrip berikut

import axios from 'axios'

Vue.use(axios)

Potongan skrip diatas merupakan skrip Axios agar dapat terhubung dengan sistem. Untuk Penjelasan dari skrip diatas adalah sebagai berikut

1. Baris 1 melakukan *import* dari plugin Axios yang sudah diinstall
2. Baris 2 melakukan pemanggilan Axios dari *import* sebelumnya agar berfungsi didalam sistem

Setelah berhasil melakukan install plugin dan memanggil plugin pada sistem, langkah berikutnya melakukan pemanggilan data dari API menggunakan Axios, agar data dapat ditampilkan dalam sistem. Kode Javascript untuk menghubungkan data dengan sistem terdapat pada halaman utama Home.vue. Untuk kode implementasinya, dapat dilihat pada skrip berikut

<script>

import axios from 'axios'

export default { -

methods: {

    getData() {

*let* uri = 'http://localhost:8000/api/tanaman'

      axios.get(uri).then((*response*) *=>* {

      this.tanamans = response.data.data

      })

    },

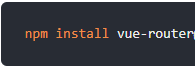
}

Potongan skrip diatas merupakan skrip Axios agar data dari API dapat terhubung dengan sistem. Untuk penjelasan skrip di atas adalah sebagai berikut

1. Baris pertama digunakan untuk melakukan import Axios
2. Baris kedua sampai keempat digunakan untuk membuat *method* yang memiliki nama getData
3. Baris selanjutnya digunakan untuk memanggil *URL* dari API dan mengembalikan data ke halaman utama

#### Implementasi Vue-Router

Vue-Router termasuk kedalam teknologi yang dimiliki dari framework Vue.js , Vue-Router yang diimplementasikan pada sistem yang dibangun pada penelitian ini bertujuan agar sistem yang dibangun dapat berpindah halaman ketika pengguna menekan menu lainnya. Pertama, unutk dapat mengimplementasikan Vue-Router ke dalam sistem, lakukan instalasi Vue-Router pada sistem, adapun *syntax* yang digunakan untuk menginstall Vue-Router kedalam sistem, sebagai berikut



Setelah menginstall Vue-Router pada sistem, langkah berikutnya yaitu mengimplementasikan kode Javascript untuk Vue-Router agar bisa digunakan dalam sistem. Kode Javascript untuk dihubungkan di sistem terdapat pada halaman Index.js yang terdapat pada *Router*. Untuk kode implementasinya, dapat dilihat pada skrip berikut

1 import VueRouter from 'vue-router'

2 Vue.use(VueRouter)

*3 const* routes = [

4  {

5    path: '/',

6    name: 'Home',

7    component: () *=>* import(/\* webpackChunkName: "home" \*/ '@/views/Home.vue'),

8    meta: {

9      layout: 'home'

10    }

  },

  {

    path: '/detail/:id',

    name: 'Detail',

    component: () *=>*

      import(/\* webpackChunkName: "detail" \*/ '@/views/Detail.vue'),

    meta: {

      layout: 'home'

    }

  },

  {

    path: '/create',

    name: 'Create',

    component: () *=>*

      import(/\* webpackChunkName: "create" \*/ '@/views/Create.vue'),

    meta: {

      layout: 'home'

    }

  },

  {

    path: '/login',

    name: 'Login',

    component: () *=>* import(/\* webpackChunkName: "login" \*/ '@/views/Login.vue')

  }

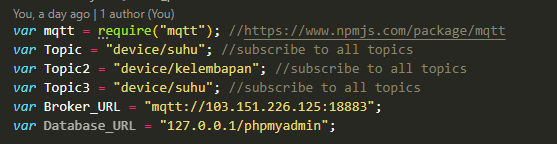
]

Potongan skrip diatas merupakan skrip Vue-Router agar bisa digunakan dalam sistem, dan juga pendaftaran *Routes* dari setiap halaman yang akan dituju. Untuk penjelasan dari skrip di atas adalah sebagai berikut

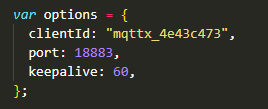
1. Baris 1 melakukan import dari Vue-Router agar dapat digunakan dalam sistem
2. Baris 2 melakukan pemanggilan Vue-Router yang telah di import pada baris pertama
3. Baris 3 melakukan deklarasi dari *Routes* untuk setiap halaman
4. Baris 5 menentukan *path* dari setiap halaman yang didaftarkan
5. Baris 6 menentukan halaman yang dituju dari *path* yang sebelumnya ditentukan
6. Baris 7 memanggil halaman yang akan ditampilkan ketika memasukkan *path* yang dituju

#### Implementasi API Alat Siram Tanaman Otomatis

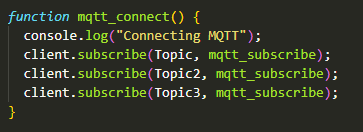
Implementasi API Alat Siram Tanaman Otomatis dalam sistem ini digunakan untuk menghubungkan sistem dengan API untuk mengambil data – data dari tanaman seperti data kelembapan tanah, intensitas cahaya, dan suhu



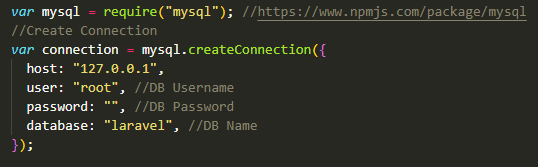
Skrip diatas digunakan untuk mendeklarasikan variable dan isi isinya, baris pertama digunakan untuk memanggil *dependencies* dari MQTT agar dapat digunakan, baris ke 2 sampai baris 4 digunakan untuk deklarasi *topic* yang diambil dari MQTT, baris ke 5 digunakan untuk deklarasi *URL* dari MQTT, dan baris terakhir digunakn untuk deklarasi *URL* database yang nantinya data akan masuk ke database



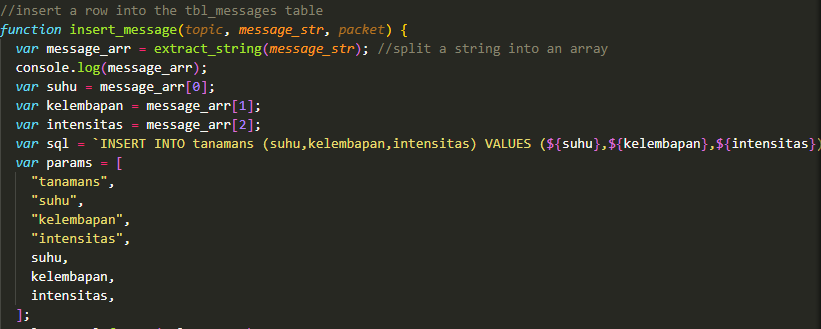
Skrip diatas digunakan untuk mendeklarasikan variable option dan isi isinya, variable options ini digunakan untuk menghubungkan *client id*, *port* dari mqtt ke sistem



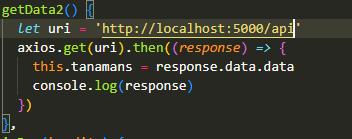
Setelah menghubungkan sistem dengan MQTT, berikutnya menambahkan *function* untuk *subscribe* ke *topic* yang sudah di deklarasikan pada baris awal, bagian ini berfungsi untuk menghubungkan API dengan MQTT agar dapat mengambil data dari MQTT



Setelah berhasil menghubungkan dengan MQTT, berikutnya menghubungkan dengan database, skrip diatas digunakan untuk menghubungkan API dengan database, sehingga ketika MQTT mengirimkan data, akan otomatis masuk ke dalam database



Skrip diatas digunakan untuk memasukkan data dari MQTT ke databse table tanamans,



Setelah menghubungkan API dengan MQTT dan dapat memasukkan data ke database, selanjutnya pada halaman utama akan memanggil *URL* dari API tersebut, yang nantinya data – data dari API akan ditampilkan pada halaman utama

## Pengujian

Pada tahapan pengujian sistem dilakukan untuk memastikan apakah semua fungsi di dalam sistem yang telah dibangun dapat berjalan dengan baik sesuai dengan spesifikasi kebutuhan dan mencari kesalahan yang mungkin terjadi pada sistem. Pengujian yang dilakukan pada sistem yang dibangun adalah pengujian alpha secara fungsional. Metode pengujian yang digunakan adalah metode pengujian blackbox yang berfokus pada persyaratan fungsional dari sistem yang dibangun.

### Rencana Pengujian BlackBox

Untuk rencana pengujian pada sistem yang dibangun, dapat dilihat pada tabel 4.8 berikut.

|  |  |  |
| --- | --- | --- |
| Kelas Uji | Poin Pengujian | Jenis Pengujian |
| Halaman Login | Keberhasilan melakukan login | Blackbox |
| Tambah Kotak Tanaman | Data masukkan | Blackbox |
| Validasi masukkan |
| Edit Tanaman | Data masukkan | Blackbox |
| Melihat Data Kotak Tanaman | Keberhasilan mengambil data dari API | Blackbox |
| Menyalakan Pompa Air | Keberhasilan mengirim data ke API | Blackbox |
| Mematikan Pompa Air | Keberhasilan mengirim data ke API | Blackbox |
| Menyalakan Lampu | Keberhasilan mengirim data ke API | Blackbox |
| Mematikan Lampu | Keberhasilan mengirim data ke API | Blackbox |
| Menyalakan *Heater* | Keberhasilan mengirim data ke API | Blackbox |
| Mematikan *Heater* | Keberhasilan mengirim data ke API | Blackbox |
| Logout | Keberhasilan melakukan logout | Blackbox |

### Hasil Pengujian

Hasil pengujian menampilkan hasi dari pengujian yang dilakukan berdasarkan dari scenario yang telah ditentukan. Pada tabel 4.10 dapat dilihat merupakan hasil dari pengujian login. Didapatkan kesimpulan diterima, karena login sukses dan sekarang aplikasi dapat menampilkan halaman utama

1. Login

Pada tabel ini merupakan hasil pengujian login. Kesimpulan yang didapat yaitu diterima. Karena berhasil login dengan membawa data username yang dilogin kan

|  |  |  |  |
| --- | --- | --- | --- |
| Data Masukan | Hasil yang Diharapkan | Pengamatan | Kesimpulan |
| Username : ayaya | Dapat masuk ke halaman utama | Tampil halaman utama | Diterima |

1. Tambah Tanaman

|  |  |  |  |
| --- | --- | --- | --- |
| Data Masukan | Hasil yang Diharapkan | Pengamatan | Kesimpulan |
| Nama kotak tanaman, isi tanaman, catatan | Aplikasi menampilkan nama kotak tanaman, isi tanaman dan catatan | Aplikasi dapat menampilkan data yang dimasukkan oleh user dan data masuk ke database | Diterima |

1. Ubah Tanaman

|  |  |  |  |
| --- | --- | --- | --- |
| Data Masukan | Hasil yang Diharapkan | Pengamatan | Kesimpulan |
| Id kotak tanaman | Aplikasi menampilkan nama tanaman, jenis tanaman dan catatan yang telah diubah | Aplikasi dapat menampilkan data yang diubah oleh user dan data yang di database akan terubah | Diterima |

1. Menyalakan Pompa Air

|  |  |  |  |
| --- | --- | --- | --- |
| Data Masukan | Hasil yang Diharapkan | Pengamatan | Kesimpulan |
| Kelembapan tanah kering | Pompa air menyala | Pompa air menyala secara otomatis ketika kelembapan tanah kering | Diterima |
| Tekan tombol nyalakan pompa | Pompa air menyala | Pompa air menyala ketika user menekan tombol “Nyalakan Pompa” | Diterima |

1. Mematikan Pompa Air

|  |  |  |  |
| --- | --- | --- | --- |
| Data Masukan | Hasil yang Diharapkan | Pengamatan | Kesimpulan |
| Kelembapan tanah basah | Pompa air mati | Pompa air mati secara otomatis ketika kelembapan tanah basah atau normal | Diterima |
| Tekan tombol matikan pompa | Pompa air mati | Pompa air mati ketika user menekan tombol “Matikan Pompa” | Diterima |

1. Menyalakan Lampu

|  |  |  |  |
| --- | --- | --- | --- |
| Data Masukan | Hasil yang Diharapkan | Pengamatan | Kesimpulan |
| Tidak ada cahaya | Lampu menyala | Lampu menyala secara otomatis ketika tidak ada cahaya | Diterima |
| Tekan tombol nyalakan lampu | Lampu menyala | Lampu menyala ketika user menekan tombol “Nyalakan Lampu” | Diterima |

1. Mematikan Lampu

|  |  |  |  |
| --- | --- | --- | --- |
| Data Masukan | Hasil yang Diharapkan | Pengamatan | Kesimpulan |
| Ada cahaya | Lampu mati | Lampu mati secara otomatis ketika ada cahaya | Diterima |
| Tekan tombol matikan lampu | Lampu mati | Lampu mati ketika user menekan tombol “Matikan Lampu” | Diterima |

1. Menyalakan *Heater*

|  |  |  |  |
| --- | --- | --- | --- |
| Data Masukan | Hasil yang Diharapkan | Pengamatan | Kesimpulan |
| Suhu dibawah 26°C | *Heater* menyala | *Heater* menyala secara otomatis ketika suhu dibawah 26°C | Diterima |
| Tekan tombol nyalakan *heater* | *Heater* menyala | *Heater* menyala ketika user menekan tombol “Nyalakan *Heater*” | Diterima |

1. Mematikan *Heater*

|  |  |  |  |
| --- | --- | --- | --- |
| Data Masukan | Hasil yang Diharapkan | Pengamatan | Kesimpulan |
| Suhu diatas 32°C | *Heater* mati | *Heater* menyala secara otomatis ketika suhu diatas 32°C | Diterima |
| Tekan tombol matikan *heater* | *Heater* mati | *Heater* mati ketika user menekan tombol “Matikan *Heater*” | Diterima |

1. Logout

|  |  |  |  |
| --- | --- | --- | --- |
| Data Masukan | Hasil yang Diharapkan | Pengamatan | Kesimpulan |
| Menekan tombol logout | Aplikasi dapat mengakhiri sesi login terakhir | Aplikasi dapat mengakhiri sesi login terakhir. Data yang sebelumnya tercatat tidak terhapus ketika melakukan logout. | Diterima |

1. Mengirim data tanaman

|  |  |  |  |
| --- | --- | --- | --- |
| Data Masukan | Hasil yang Diharapkan | Pengamatan | Kesimpulan |
| Id tanaman | Aplikasi menampilkan data tanaman yang dikirim dari API | Aplikasi dapat menampilkan data – data tanaman diantaranya suhu, kelembapan dan intensitas cahaya | Diterima |

# BAB V KESIMPULAN DAN SARAN

## Kesimpulan

Berdasarkan dari hasil implementasi, pengujian, serta wawancara kepada pengguna yang telah dilakukan, maka didapatkan kesimpulan bahwa sistem Monitoring Tanaman dapat memudahkan pengguna untuk dapat mengetahui data – data dari kotak tanaman, dan juga untuk mengontrol kotak tanaman

## Saran

Untuk mengembangkan sistem monitoring tanaman memanfaatkan API alat siram otomatis ini, peneliti memberikan saran baik dari peneliti sendiri, maupun dari para pengguna sistem yang diharapkan dapat terwujud pada penelitian selanjutnya. Adapun sarannya adalah sebagai berikut.

1. Memperbaiki kombinasi warna yang dipakai untuk tampilan sistem agar warnanya menjadi lebih seragam lagi.
2. Sistem ini diharapkan dapat digunakan di sector yang lebih besar selain daripada greenhouse.

Demikian kesimpulan dan saran yang dapat peneliti berikan. Semoga saran – saran tersebut dapat dijadikan masukkan yang bermanfaat dan menjadikan penelitian selanjutnya menjadi lebih baik lagi.

# DAFTAR PUSTAKA

|  |  |
| --- | --- |
| [1] | DEWI, NI LUH PRIMA KEMALA. Analisis Tingkat Keuntungan Usahatani Padi Sawah sebagai Dampak dari adanya Subsidi Pupuk di Kabupaten Tabanan. *Journal of Agribusiness and Agritourism*, 2016, 44925. |
| [2] | ENDRA, Robby Yuli, et al. Perancangan Aplikasi Berbasis Web Pada System Aeroponik untuk Monitoring Nutrisi Menggunakan Framework CodeIgniter. *Explore: Jurnal Sistem Informasi dan Telematika (Telekomunikasi, Multimedia dan Informatika)*, 2020, 11.1: 10-16. |
| [3] | HARTONO, Rodi; MALIK, Aditya. Sistem Otomatis Pembuatan Nutrisi Ideal untuk Tanaman Pakcoy Menggunakan Kendali Logika Fuzzy Automatic Sistem for Making Ideal Nutrients for Pakcoy Plant Using Fuzzy Logic Control. |
| [4] | TELAUMBANUA, Juan Peringatan. Pengaruh Berbagai Nutrisi Terhadap Tanaman Bayam Merah (Amaranthus tricolor L.). 2019. |
| [5] | SUTIARSO, Lilik, et al. Aplikasi Sistem Monitoring Pertumbuhan Tanaman Berbasis Web Menggunakan Machine Vision. *Agritech: Jurnal Fakultas Teknologi Pertanian UGM*, 2011, 31.4: 99181. |
| [6] | DONI, Rahmad; RAHMAN, Maulia. Sistem Monitoring Tanaman Hidroponik Berbasis Iot (Internet of Thing) Menggunakan Nodemcu ESP8266. J-SAKTI (Jurnal Sains Komputer dan Informatika), 2020, 4.2: 516-522. |
| [7] | WINSEN, Winsen; SETIAWAN, Alexander; LIM, Resmana. Aplikasi Monitoring Kelembapan Tanah, Suhu, Kadar PH Tanah Serta Penyiraman Dan Pemupukan Otomatis Pada Tanaman Hias Lidah Mertua Berbasis IOT. Jurnal Infra, 2022, 10.1: 85-90. |
| [8] | Ulinuha, A., & Riza, A. G. (2021). SISTEM MONITORING DAN PENYIRAM TANAMAN OTOMATIS BERBASIS ANDROID DENGAN APLIKASI BLYNK. Abdi Teknoyasa, 2(1), 26–31. Retrieved from https://journals2.ums.ac.id/index.php/abditeknoyasa/article/view/318 |
| [9] | R. Y. Endra, A. Cucus, F. N. Affandi, and D. Hermawan, “Implementasi Sistem Kontrol Berbasis Web Pada Smart Room Dengan Menggunakan Konsep Internet Of Things,” Explor. – J. Sist. Inf. dan Telemat. (Telekomunikasi, Multimed. Inform., vol. 10, no. 2, pp. 98–106, 2019, doi: <http://dx.doi.org/10.36448/jsit.v10i2.1316> |
| [10] | WIRYANTA, Bernardinus T. Wahyu. *Media tanam untuk tanaman hias*. AgroMedia, 2007. |