

# SKPL

## SPEKIFIKASI KEBUTUHAN PERANGKAT LUNAK

### Pembangunan Sistem Informasi Geografis Perkebunan Propinsi Jambi Berbasis Web

Dipersiapkan oleh:


S. Thya Safitri / 1111

Wahyu Adi Prabowo / 2222

Program Studi Teknik Informatika

Fakultas Teknologi Industri dan Informatika

Institut Teknologi Telematika Telkom Purwokerto

	Program Studi Teknik Informatika  FTII	Nomor Dokumen		Halaman
		<b>SKPL</b>		1/39

## DAFTAR PERUBAHAN

Revisi	Deskripsi
<b>A</b>	
<b>B</b>	
<b>C</b>	
<b>D</b>	
<b>E</b>	
<b>F</b>	

INDEX	-	A	B	C	D	E	F	G
TGL								
Ditulis oleh								
Diperiksa oleh								
Disetujui oleh								

### Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi

## Daftar Isi

1. Pendahuluan	6
1.1. Tujuan	6
1.2. Lingkup Masalah	6
1.3. Definisi, Akronim dan Singkatan	7
1.4. Deskripsi umum ( Overview )	8
2. Deskripsi Kebutuhan	8
2.1. Perspektif Produk	8
Gambar 1 Arsitektur Aplikasi SIGJ	9
2.1.1 Antarmuka Pemakai	9
2.1.2 Antarmuka Perangkat Keras	10
2.1.3 Antarmuka Perangkat Lunak	10
2.1.4 Antarmuka Komunikasi	11
2.2. Fungsi Produk	11
2.3. Karakteristik Pengguna	15
2.4. Batasan – batasan	16
2.5. Asumsi dan Ketergantungan	16
3 Kebutuhan Fungsional Perangkat Lunak	17
3.1 Use Case Diagram	17
3.2 Use Case Specification	18
3.2.1 Use case Spesification : Login	18
3.2.2 Use case Spesification : Ubah Password	18
3.2.3 Use case Spesification : Kelola Data Perkebunan	20
3.2.4 Use case Spesification : Kelola Data Pemilik Perkebunan	22
3.2.5 Use case Spesification : Kelola Data Hasil Kebun	24
3.2.6 Use case Spesificatio : Kelola Data Pengguna	25
3.2.7 Use case Spesification : Kelola Jumlah Hasil Kebun	27
3.2.8 Use case Spesification : Tampil Peta Perkebunan	29
3.2.9 Use case Spesification : Tambah Point Perkebunan	30
4 Analisis Kebutuhan Data	32
4.1 ERD (Entity Relationship Diagram)	32
4.2 Data Definition	33
4.2.1 Data DATAUSER	33
4.2.2 Data DATAPEMILIKPERKEBUNAN	33
4.2.3 Data DATAPERKEBUNAN	34
4.2.4 Data DATAHASILKEBUN	36
4.2.5 Data DATAKABUPATEN	36
4.2.6 Data DATAKECAMATAN	37
4.2.7 Data DATAROLE	37
4.2.8 Data DATAJUMLAH	38

## Daftar Gambar

1. Arsitektur Aplikasi SIGJ.....	9
2. Use Case Diagram .....	18
3. Entity Relationship Diagram (ERD) .....	33

## **1. Pendahuluan**

### **1.1. Tujuan**

Tujuan dari dokumen spesifikasi kebutuhan perangkat lunak ini merupakan dokumen spesifikasi kebutuhan perangkat lunak SIGJ (Pembangunan Sistem Informasi Geografis berbasis Web untuk Pendataan Perkebunan dan Pelaporan Produksi Hasil Perkebunan di Propinsi Jambi) untuk mendefinisikan kebutuhan perangkat lunak yang meliputi antarmuka eksternal (antarmuka antara sistem dengan perangkat lunak dan perangkat keras, dan pengguna), performansi (kemampuan perangkat lunak dari segi kecepatan, tempat penyimpanan yang dibutuhkan, serta keakuratan), dan atribut tambahan yang dimiliki sistem, serta mendefinisikan fungsi perangkat lunak. SKPL-SIGJ ini juga mendefinisikan batasan perancangan perangkat lunak, karakteristik program, serta asumsi dan ketergantungan perangkat lunak ini.

### **1.2. Lingkup Masalah**

Perangkat lunak SIGJ dikembangkan dengan tujuan untuk :

1. Menangani fungsi login
2. Menangani ubah password
3. Menangani pengelolaan data perkebunan
4. Menangani pengelolaan data hasil kebun
5. Menangani pengelolaan data pemilik perkebunan
6. Menangani pengelolaan data pengguna
7. Menangani pengelolaan data jumlah hasil kebun
8. Menampilkan peta perkebunan

Dan berjalan pada lingkungan dengan platform Windows XP.

### **1.3. Definisi, Akronim dan Singkatan**

Daftar definisi dan akronim yang digunakan :

<b>Keyword atau Phrase</b>	<b>Definisi</b>
SKPL	Dokumen yang berisi tentang spesifikasi kebutuhan pengembangan perangkat lunak.
SKPL-SIGJ-XX	Kode yang merepresentasikan kebutuhan pada SIGJ (Sistem Informasi Geografis Propinsi Jambi) dimana XXX merupakan nomor fungsi produk.
SIGJ	Sistem Informasi Geografis Propinsi Jambi untuk Membantu Keputusan Penataan dan Pengelolaan Perkebunan di Propinsi Jambi.
Server	Komputer yang menyediakan sumber daya bagi klien yang terhubung melalui jaringan.
Internet	Istilah umum yang digunakan untuk menunjuk <i>Network Global</i> yang terdiri dari komputer dan layanan servis dengan sekitar 30 sampai 50 juta pemakai computer dan puluhan layanan informasi termasuk <i>e-mail</i> , <i>FTP</i> , dan <i>World Wide Web</i> .
Perkebunan	Sebidang tanah yang digunakan untuk menanam hasil - hasil perkebunan oleh sekelompok masyarakat ataupun perorangan maupun perusahaan.

#### **1.4. Deskripsi umum ( Overview )**

Secara umum dokumen SKPL ini terbagi 4 bagian utama. Bagian pertama berisi penjelasan mengenai dokumen SKPL tersebut yang mencakup tujuan pembuatan SKPL, ruang lingkup masalah dalam pengembangan perangkat lunak, definisi, referensi, dan deskripsi umum tentang dokumen SKPL ini.

Bagian kedua berisi penjelasan umum tentang perangkat lunak SIGJ yang akan dikembangkan mencakup perspektif produk yang, fungsi produk perangkat lunak, karakteristik pengguna, batasan dalam penggunaan perangkat lunak dan asumsi yang terpakai dalam pengembangan perangkat lunak SIGJ.

Bagian ketiga berisi penjelasan secara lebih rinci tentang kebutuhan perangkat lunak SIGJ yang akan dikembangkan. Pada bagian terakhir atau bagian keempat berisi tentang spesifikasi kebutuhan data.

## **2. Deskripsi Kebutuhan**

### **2.1. Perspektif Produk**

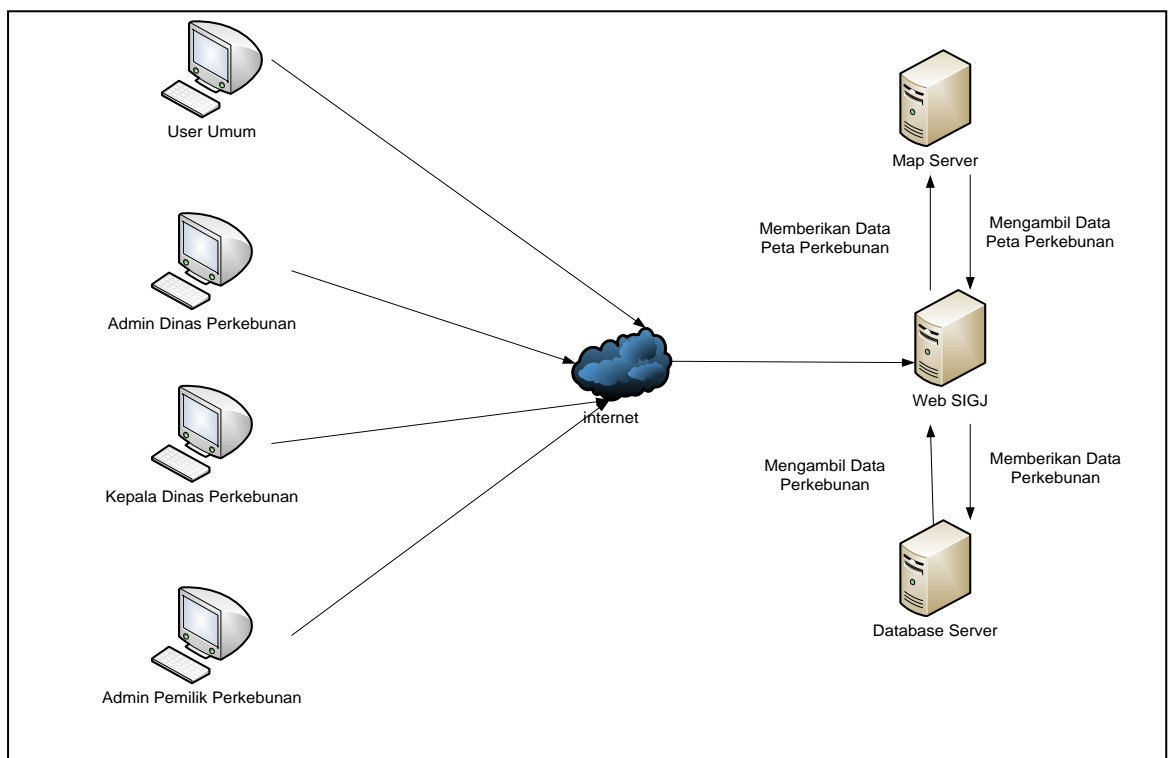
SIGJ merupakan perangkat lunak yang menyediakan informasi mengenai data perkebunan dan data hasil perkebunan di Propinsi Jambi. Informasi disajikan dalam bentuk peta perkebunan beserta hasil perkebunan dalam kurun waktu tertentu.

Perangkat lunak SIGJ ini berjalan pada platform Windows XP untuk komputer PC atau *notebook*, yang memiliki *Internet Explorer* dan terhubung secara *on-line* ke internet. Bahasa pemrograman yang digunakan adalah



ASP.NET. Sedangkan untuk lingkungan pemrogramannya menggunakan Microsoft Visual Studio .NET 2005.

Pengguna akan berinteraksi dengan sistem melalui antarmuka GUI (*Graphical User Interface*). Aplikasi SIGJ mendapatkan data-data perkebunan dari berbagai pemilik perkebunan secara perorangan maupun perusahaan dengan bantuan *web service*. Data-data perkebunan tersebut kemudian diolah untuk kemudian disimpan dalam database server. Pengolahan data tersebut akan menghasilkan sebuah informasi yang akan ditampilkan dengan tabel informasi juga didukung dengan tampilan peta.



Gambar 1Arsitektur Aplikasi SIGJ

### 2.1.1 Antarmuka Pemakai

Pengguna berinteraksi dengan antarmuka yang ditampilkan dalam bentuk halaman web.

### **2.1.2 Antarmuka Perangkat Keras**

Piranti perangkat keras yang digunakan dalam perangkat lunak ini adalah :

1. PC
2. Mouse
3. Keyboard
4. Modem atau LAN Card

### **2.1.3 Antarmuka Perangkat Lunak**

Perangkat lunak yang dibutuhkan dalam mengoperasikan perangkat lunak SIGJ adalah:

1. Nama : Windows XP Professional SP2  
Sumber : Microsoft  
Sebagai Sistem Operasi komputer.
2. Nama : Microsoft SQL Server 2005  
Sumber : Microsoft  
Sebagai DBMS yang dibutuhkan dalam mengoperasikan perangkat lunak SIGJ.
3. Nama : .Net Framework  
No Versi : 3.5  
Sumber : Microsoft  
Sebagai pustaka pemrograman yang dibutuhkan dalam menjalankan perangkat lunak WIG.
4. Nama : Mozilla Firefox  
Sumber : Mozilla  
Sebagai internet browser di mana perangkat lunak WIG dijalankan.
5. Nama : IIS (Internet Information System)  
Sumber : Microsoft  
Sebagai web server.

6. Nama : Map Server  
Sumber : Map Server  
Sebagai framework Web GIS.

#### **2.1.4 Antarmuka Komunikasi**

Antarmuka komunikasi suatu protokol komunikasi yang digunakan dalam mengoperasikan perangkat lunak SIGJ yaitu protokol TCP/IP.

#### **2.2. Fungsi Produk**

Fungsi produk perangkat lunak SIGJ akan dibagi berdasarkan pengguna atau aktor yang menggunakan perangkat lunak ini :

##### **1. Fungsi Login (SKPL-01)**

Merupakan fungsi yang digunakan oleh user untuk masuk ke dalam sistem yang akan digunakan.

##### **2. Fungsi Ubah Password (SKPL-02)**

Merupakan fungsi yang digunakan untuk mengubah password pengguna.

##### **a. Admin Pemilik Perkebunan**

##### **1. Fungsi Kelola Data Pemilik Perkebunan (SKPL-03)**

Merupakan fungsi yang digunakan untuk mengelola data pribadi pemilik lahan perkebunan.

Fungsi kelola data pemilik perkebunan mencakup:

##### **a. Fungsi Tampil Data Pemilik (SKPL-03-01)**

Merupakan fungsi yang digunakan untuk menampilkan data pemilik perkebunan pribadi.

##### **b. Fungsi Ubah Data Pemilik (SKPL-03-02)**

Merupakan fungsi yang digunakan untuk mengubah data pemilik perkebunan.

##### **c. Fungsi Hapus Data Pemilik (SKPL-03-03)**

Merupakan fungsi yang digunakan untuk menghapus data pemilik perkebunan pribadi.

2. Fungsi *Kelola Jumlah Hasil Kebun* (**SKPL-04**)

a. Fungsi *Tampil Jumlah Hasil Kebun* (**SKPL-04-01**)

Merupakan Fungsi yang digunakan untuk menampilkan data jumlah hasil perkebunan pribadi. Tampil data dapat dilakukan dengan memilih kategori tampilan yaitu berdasarkan Hasil Kebun, Tahun dan Nama Perkebunan.

b. Fungsi *Tambah Jumlah Hasil* (**SKPL-04-02**)

Merupakan Fungsi yang digunakan untuk melakukan penambahan jumlah hasil perkebunan pada perkebunan milik pribadi.

b. Admin Dinas Perkebunan

1. Fungsi *Kelola Data Pengguna* (**SKPL-05**)

Merupakan fungsi yang digunakan untuk mengelola data - data hasil perkebunan.

Fungsi kelola data hasil perkebunan mencakup:

a. Fungsi *Tambah Data Pengguna* (**SKPL-05-01**)

Merupakan fungsi yang digunakan untuk memasukkan data pengguna.

b. Fungsi *Ubah Data Pengguna* (**SKPL-05-02**)

Merupakan fungsi yang digunakan untuk melakukan perubahan data pengguna.

c. Fungsi *Hapus Data Pengguna* (**SKPL-05-03**)

Merupakan fungsi yang digunakan untuk melakukan penghapusan data pengguna.

2. Fungsi *Kelola Data Perkebunan* (**SKPL-06**)

Merupakan fungsi yang digunakan untuk mengelola data perkebunan.

Fungsi kelola data perkebunan mencakup:

a. Fungsi *Tambah Data Perkebunan* (**SKPL-06-01**)

Merupakan fungsi yang digunakan untuk menambah data perkebunan perkebunan.

b. Fungsi *Ubah Data Perkebunan* (**SKPL-06-02**)

Merupakan fungsi yang digunakan untuk mengubah data perkebunan.

c. Fungsi *Hapus Data Perkebunan* (**SKPL-06-03**)

Merupakan fungsi yang digunakan untuk menghapus data perkebunan.

3. Fungsi *Kelola Data Hasil Kebun* (**SKPL-07**)

Merupakan fungsi yang digunakan untuk mengelola data hasil perkebunan.

Fungsi kelola data perkebunan mencakup:

a. Fungsi *Tambah Data Hasil Kebun* (**SKPL-07-01**)

Merupakan fungsi yang digunakan untuk menambah data hasil perkebunan.

b. Fungsi *Ubah Data Kebun* (**SKPL-07-02**)

Merupakan fungsi yang digunakan untuk mengubah data hasil perkebunan.

c. Fungsi *Hapus Data Kebun* (**SKPL-07-03**)

Merupakan fungsi yang digunakan untuk menghapus data hasil perkebunan.

4. Fungsi *Kelola Data Pemilik Perkebunan* (**SKPL-08**)

Merupakan fungsi yang digunakan untuk mengelola data hasil perkebunan.

Fungsi kelola data perkebunan mencakup:

a. Fungsi *Tambah Data Pemilik Perkebunan* (**SKPL-08-01**)

Merupakan fungsi yang digunakan untuk menambah data pemilik perkebunan.

b. Fungsi *Ubah Data Pemilik Perkebunan* (**SKPL-08-02**)

Merupakan fungsi yang digunakan untuk mengubah data pemilik perkebunan.

c. Fungsi *Hapus Data Pemilik Perkebunan* (**SKPL-08-03**)

Merupakan fungsi yang digunakan untuk menghapus data pemilik perkebunan.

c. Kepala Dinas Perkebunan Wilayah

1. Fungsi *Tampil Peta Perkebunan* (**SKPL-09**)

Merupakan fungsi yang digunakan untuk menampilkan peta dan informasi perkebunan yaitu data perkebunan dan data hasil perkebunan di Propinsi Jambi.

Fungsi tampil data perkebunan mencakup:

a. Fungsi *Tampil Data Perkebunan Berdasarkan Kabupaten* (**SKPL-09-01**)

Merupakan fungsi yang digunakan untuk menampilkan data perkebunan berdasarkan kabupaten di Propinsi Jambi.

b. Fungsi *Tampil Data Perkebunan berdasarkan Kecamatan* (**SKPL-09-02**)

Merupakan fungsi yang digunakan untuk menampilkan data perkebunan berdasarkan kecamatan di Propinsi Jambi.

c. Fungsi Tampil Data Perkebunan Berdasarkan Nama Perkebunan (**SKPL-07-03**)

Merupakan fungsi yang digunakan untuk menampilkan data perkebunan berdasarkan hasil perkebunan.

2. Fungsi Tambah Titik Perkebunan (**SKPL-10**)

Merupakan Fungsi yang digunakan untuk melakukan penambahan Titik / Point perkebunan sebagai Simbol data Perkebunan.

d. User Umum

1. Fungsi Tambah Titik Perkebunan (**SKPL-11**)

Merupakan Fungsi yang digunakan untuk melakukan penambahan Titik / Point perkebunan sebagai Simbol data Perkebunan.

### **2.3. Karakteristik Pengguna**

Karakteristik pengguna yang menggunakan perangkat lunak WIG yaitu :

1. Admin Pemilik Perkebunan

- a. Mengerti pengoperasian komputer
- b. Memahami sistem komputer tempat perangkat lunak dijalankan
- c. Memahami perangkat lunak yang digunakan
- d. Memiliki sedikit kemampuan pemrograman
- e. Mengerti proses pengelolaan data
- f. Memahami konsep basis data

2. Admin Dinas Perkebunan

- a. Mengerti pengoperasian komputer
- b. Memahami sistem komputer tempat perangkat lunak dijalankan

- c. Memahami perangkat lunak yang digunakan
  - d. Memiliki sedikit kemampuan pemrograman
  - e. Mengerti proses pengelolaan data
  - f. Memahami konsep Basis Data
  - g. Memahami pengelolaan mapscript
3. Kepala Dinas Perkebunan Wilayah
- a. Mengerti pengoperasian komputer
  - b. Memahami sistem komputer tempat perangkat lunak dijalankan
  - c. Dapat menggunakan aplikasi yang berbasis web
4. User Umum
- a. Mengerti pengoperasian komputer

#### **2.4. Batasan - batasan**

Batasan yang digunakan dalam pengembangan perangkat lunak SIGJ yaitu :

1. Kebijakan umum

Mengacu pada tujuan dari pengembangan perangkat lunak SIGJ yaitu untuk membantu pengambilan keputusan dinas terkait dalam pengadaan perizinan pembukaan lahan baru sebagai perkebunan.

2. Keterbatasan perangkat keras

Ditentukan kemudian setelah setelah sistem ini berjalan (sesuai dengan kebutuhan).

#### **2.5. Asumsi dan Ketergantungan**

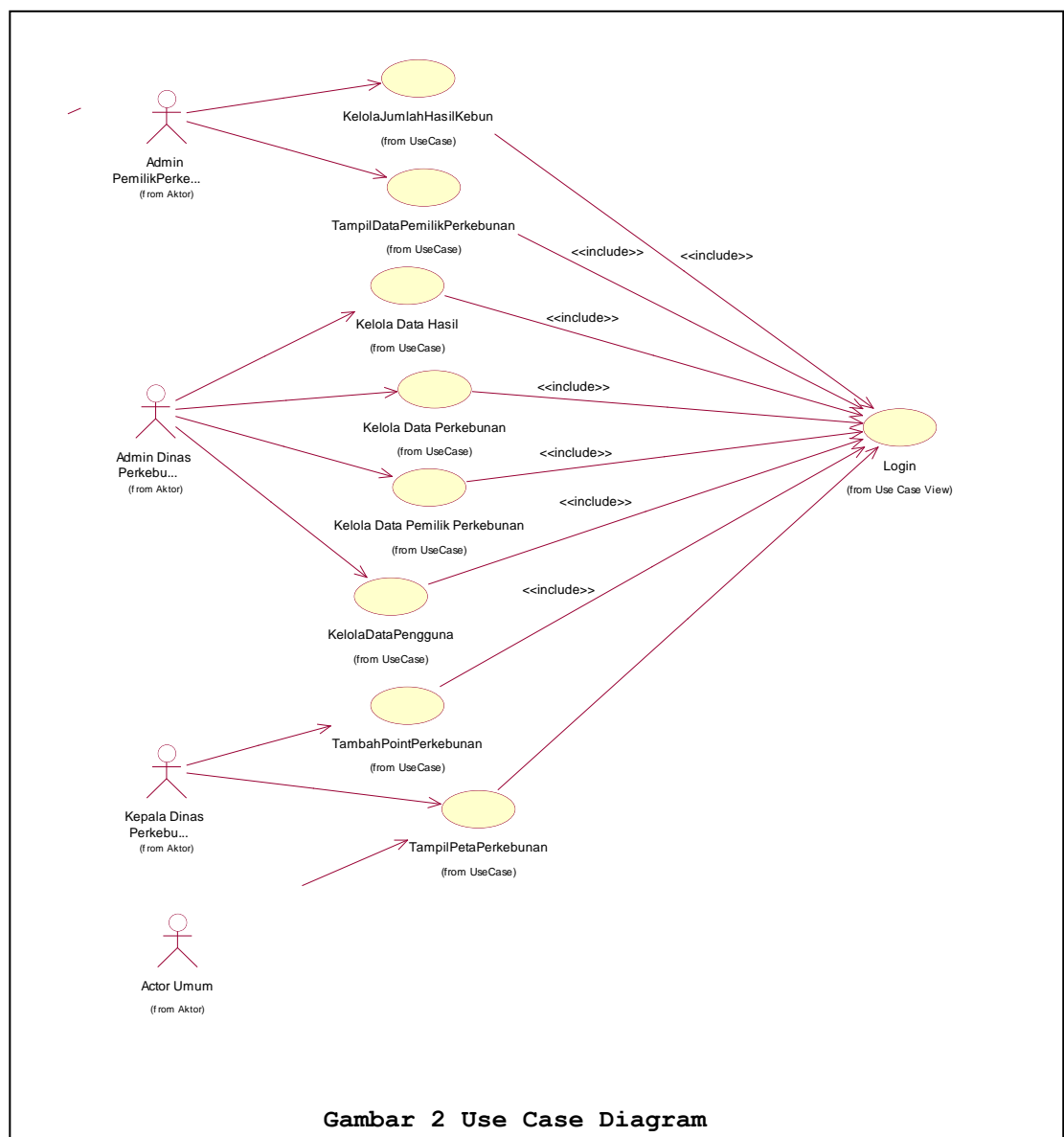
Kebutuhan dari SIGJ ini yaitu yang sifatnya non-fungsionalitas adalah :



- a. Tersedia perangkat lunak yang sesuai dengan kebutuhan untuk mengoperasikan produk perangkat lunak SIGJ
- b. Tersedianya web browser atau web service untuk akses internet

### 3 Kebutuhan Fungsional Perangkat Lunak

#### 3.1 Use Case Diagram



### 3.2 Use Case Specification

#### 3.2.1 Use case Spesification :Login

<b>Brief Description</b>	Use Case ini digunakan oleh aktor untuk memperoleh akses ke sistem. Login didasarkan pada sebuah id unik yaitu user name dari user dan password yang berupa rangkaian karakter.
<b>Primary Actor</b>	Admin Pemilik Perkebunan, Admin Dinas Perkebunan dan Kepala Dinas Perkebunan
<b>Supporting Actor</b>	-
<b>Basic Flow</b>	1.Use Case ini dimulai ketika aktor memilih untuk melakukan login 2.Sistem menampilkan antarmuka untuk login 3.Aktor memasukkan username dan password 4.Sistem memeriksa username dan password yang diinputkan aktor E-1 Usernameatau Password tidak sesuai 5.Sistem memberikan akses ke aktor 6.Use Case ini selesai
<b>Alternative Flow</b>	-
<b>Error Flow</b>	E-1 Usernameatau Password tidak sesuai 1. Sistem menampilkan peringatan bahwa user_name atau password tidak sesuai 2. Kembali ke Basic Flow langkah ke 3
<b>PreConditions</b>	-
<b>PostConditions</b>	Aktor memasuki sistem dan dapat menggunakan fungsi-fungsi dalam sistem sesuai dengan hak akses yang diberikan.

#### 3.2.2 Use case Spesification :Ubah Password

<b>Brief</b>	Use Case ini digunakan oleh aktor untuk
--------------	---

<b>Description</b>	mengubah password yang digunakan saat login ke sistem.
<b>Primary Actor</b>	Admin Pemilik Perkebunan, Admin Dinas Perkebunan dan Kepala Dinas Perkebunan
<b>Supprting Actor</b>	-
<b>Basic Flow</b>	<ol style="list-style-type: none"> <li>1. Use Case ini dimulai ketika aktor memilih untuk melakukan ubah password</li> <li>2. Sistem menampilkan antarmuka ubah password</li> <li>3. Sistem menampilkan username user saat login</li> <li>4. Aktor menginputkan password lama, password baru, dan <i>confirm</i> password</li> <li>5. Aktor meminta sistem untuk menyimpan data yang telah diubah</li> <li>6. Sistem melakukan pengecekan terhadap data yang telah diubah <ul style="list-style-type: none"> <li>E-1 Username dan password lama salah</li> <li>E-2 Password baru tidak berhasil dikonfirmasi</li> </ul> </li> <li>7. Sistem menyimpan data yang telah diubah ke database</li> <li>8. Use Case selesai</li> </ol>
<b>Alternative Flow</b>	-
<b>Error Flow</b>	<p>E-1 Username dan password lama salah</p> <ol style="list-style-type: none"> <li>1. Sistem menampilkan peringatan bahwa password lama salah</li> <li>2. Kembali ke Basic Flow langkah ke-4</li> </ol> <p>E-2 Password baru tidak berhasil dikonfirmasi</p> <ol style="list-style-type: none"> <li>1. Sistem menampilkan peringatan bahwa password baru tidak berhasil dikonfirmasi</li> <li>2. Kembali ke Basic Flow langkah ke-4</li> </ol>
<b>PreConditions</b>	1. Use Case Login telah dilakukan

	2. Aktor telah berhasil memasuki sistem
<b>PostConditions</b>	Password telah diubah.

### 3.2.3 Use case Spesification :Kelola Data Perkebunan

<b>Brief Description</b>	Use Case ini digunakan oleh aktor untuk melakukan pengelolaan data perkebunan.
<b>Primary Actor</b>	Admin Dinas Perkebunan
<b>Supporting Actor</b>	-
<b>Basic Flow</b>	<ol style="list-style-type: none"> <li>1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan data perkebunan</li> <li>2. Sistem menampilkan antarmuka pengelolaan data perkebunan</li> <li>3. Sistem memberikan pilihan untuk melakukan tambah perkebunan atau ubah data perkebunan</li> <li>4. Aktor memilih untuk melakukan tambah data perkebunan <ol style="list-style-type: none"> <li>A-1 Aktor memilih untuk melakukan ubah data perkebunan</li> <li>A-2 Aktor memilih untuk melakukan hapus data perkebunan</li> </ol> </li> <li>5. Aktor menginputkan data perkebunan</li> <li>6. Aktor meminta sistem untuk menyimpan data perkebunan yang telah diinputkan</li> <li>7. Sistem mengecek data perkebunan yang telah diinputkan <ol style="list-style-type: none"> <li>E-1 Data perkebunan yang diinputkan sudah ada</li> </ol> </li> <li>8. Sistem menyimpan data perkebunan ke database</li> <li>9. Use Case selesai</li> </ol>
	A-1 Aktor memilih untuk melakukan ubah data

<b>Alternative Flow</b>	<p>perkebunan</p> <ol style="list-style-type: none"> <li>1. Aktor menginputkan luas data perkebunan dan nama daerah perkebunan</li> <li>2. Sistem menampilkan detail data perkebunan</li> <li>3. Aktor mengubah data perkebunan</li> <li>4. Aktor meminta sistem untuk menyimpan data perkebunan yang telah diubah</li> <li>5. Sistem menyimpan data perkebunan yang telah diubah ke database</li> <li>6. Berlanjut ke Basic Flow langkah ke-9</li> </ol> <p>A-2 Aktor memilih untuk melakukan hapus data perkebunan</p> <ol style="list-style-type: none"> <li>1. Aktor menginputkan id_perkebunan dan id_pemilik</li> <li>2. Sistem menampilkan detail data perkebunan</li> <li>3. Aktor memilih data perkebunan yang akan dihapus</li> <li>4. Sistem melakukan konfirmasi bahwa data akan dihapus</li> <li>5. Sistem menghapus data dari database</li> <li>6. Berlanjut ke Basic Flow langkah ke-9</li> </ol>
<b>Error Flow</b>	<p>E-1 Data perkebunan yang diinputkan sudah ada</p> <ol style="list-style-type: none"> <li>1. Sistem menampilkan peringatan bahwa data perkebunan yang diinputkan sudah ada</li> <li>2. Kembali ke Basic Flow langkah 4</li> </ol>
<b>PreConditions</b>	<ol style="list-style-type: none"> <li>1. Use Case Login telah dilakukan</li> <li>2. Aktor berhasil memasuki sistem</li> </ol>
<b>PostConditions</b>	Data perkebunan bertambah, diubah atau telah terhapus

### 3.2.4 Use case Spesification :Kelola Data Pemilik Perkebunan

<b>Brief Description</b>	Use Case ini digunakan oleh aktor untuk melakukan pengelolaan data pemilik perkebunan.
<b>Primary Actor</b>	Admin Pemilik Perkebunan dn Admin Dinas Perkebunan
<b>Supporting Actor</b>	-
<b>Basic Flow</b>	<ol style="list-style-type: none"> <li>1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan data pemilik perkebunan</li> <li>2. Sistem menampilkan antarmuka pengelolaan data pemilik perkebunan</li> <li>3. Sistem memberikan pilihan untuk melakukan tambah pemilik perkebunan dan ubah pemilik perkebunan</li> <li>4. Aktor memilih untuk melakukan tambah data pemilik perkebunan <ol style="list-style-type: none"> <li>A-2 Aktor memilih untuk melakukan ubah data pemilik perkebunan</li> <li>A-3 Aktor memilih untuk melakukan tambah <i>point</i> perkebunan</li> </ol> </li> <li>5. Aktor menginputkan data pemilik perkebunan</li> <li>6. Aktor meminta sistem untuk menyimpan data pemilik perkebunan yang telah diinputkan</li> <li>7. Sistem mengecek data pemilik perkebunan yang telah diinputkan <ol style="list-style-type: none"> <li>E-1 Nama pemilik perkebunan yang diinputkan telah terpakai</li> </ol> </li> <li>8. Sistem menyimpan data pemilik perkebunan ke database</li> <li>9. Use Case selesai</li> </ol>

<p><b>Alternative Flow</b></p>	<p>A-1 Aktor memilih untuk melakukan ubah data pemilik perkebunan</p> <ol style="list-style-type: none"> <li>1. Aktor menginputkan nama pemilik perkebunan yang akan diubah</li> <li>2. Sistem menampilkan detail data pemilik perkebunan</li> <li>3. Aktor mengubah data pemilik perkebunan</li> <li>4. Aktor meminta sistem untuk menyimpan data pemilik perkebunan yang telah diubah</li> <li>5. Sistem menyimpan data pemilik perkebunan yang telah diubah ke database</li> <li>6. Berlanjut ke Basic Flow langkah ke-9</li> </ol> <p>A-2 Aktor memilih untuk melakukan tambah <i>point</i> data perkebunan</p> <ol style="list-style-type: none"> <li>1. Aktor menginputkan nama pemilik perkebunan</li> <li>2. Sistem menampilkan detail data perkebunan</li> <li>3. Aktor memilih data perkebunan yang mau ditambahkan <i>point</i> ke dalam peta E-2Point untuk data perkebunan yang dipilih aktor telah ditambahkan</li> <li>4. Aktor menambahkan <i>point</i> ke dalam peta</li> <li>5. Sistem menyimpan <i>point</i> bersama data perkebunan</li> <li>6. Berlanjut ke Basic Flow langkah ke-9</li> </ol>
<p><b>Error Flow</b></p>	<p>E-1 Nama pemilik perkebunan yang diinputkan telah terpakai</p> <ol style="list-style-type: none"> <li>1. Sistem menampilkan peringatan bahwa nama pemilik perkebunan yang diinputkan telah terpakai</li> <li>2. Kembali ke Basic Flow langkah 4</li> </ol>

	<p>E-2 Point untuk data perkebunan yang dipilih aktor telah ditambahkan</p> <ol style="list-style-type: none"> <li>1.Sistem menampilkan peringatan bahwa <i>point</i> sudah ada</li> <li>2.Kembali ke Alternatif Flow A-2.3</li> </ol>
<b>PreConditions</b>	<ol style="list-style-type: none"> <li>1. Use Case Login telah dilakukan</li> <li>2. Aktor berhasil memasuki sistem</li> </ol>
<b>PostConditions</b>	Datapemilik perkebunan bertambah atau telah berubah

### 3.2.5 Use case Spesification :Kelola Data Hasil Kebun

<b>Brief Description</b>	Use Case ini digunakan oleh aktor untuk menampilkan data hasil kebun.
<b>Primary Actor</b>	Admin Dinas Perkebunan
<b>Supprting Actor</b>	-
<b>Basic Flow</b>	<ol style="list-style-type: none"> <li>1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan data hasil kebun</li> <li>2. Sistem menampilkan antarmuka pengelolaan data hasil kebun</li> <li>3. Sistem memberikan pilihan untuk melakukan tambah dan ubah hasil kebun</li> <li>4. Aktor memilih untuk melakukan tambah datahasil kebun</li> <li>A-2 Aktor memilih untuk melakukan ubah data hasil kebun</li> <li>5. Aktor menginputkan data hasil kebun</li> <li>6. Aktor meminta sistem untuk menyimpan data hasil kebun yang telah diinputkan</li> <li>7. Sistem mengecek data hasil kebun yang telah diinputkan</li> <li>E-1 Id Hasil Kebun yang diinputkan telah terpakai</li> <li>8. Sistem menyimpan data hasil kebun ke database</li> </ol>



	9. Use Case selesai
<b>Alternative Flow</b>	<p>A-1 Aktor memilih untuk melakukan ubah hasil kebun</p> <ol style="list-style-type: none"> <li>1. Aktor menginputkan id hasil kebun yang akan diubah</li> <li>2. Sistem menampilkan detail data hasil kebun</li> <li>3. Aktor mengubah data hasil kebun</li> <li>4. Aktor meminta sistem untuk menyimpan data hasil kebun yang telah diubah</li> <li>5. Sistem menyimpan data hasil kebun yang telah diubah ke database</li> <li>6. Berlanjut ke Basic Flow langkah ke-9</li> </ol>
<b>Error Flow</b>	<p>E-1 Id hasil kebun yang diinputkan telah terpakai</p> <ol style="list-style-type: none"> <li>1. Sistem menampilkan peringatan bahwa Id hasil kebun yang diinputkan telah terpakai</li> <li>2. Kembali ke Basic Flow langkah 4</li> </ol>
<b>PreConditions</b>	<ol style="list-style-type: none"> <li>1. Use Case Login telah dilakukan</li> <li>2. Aktor berhasil memasuki sistem</li> </ol>
<b>PostConditions</b>	Data hasil kebun bertambah atau telah berubah

### 3.2.6 Use case Spesification :Kelola Data Pengguna

<b>Brief Description</b>	Use Case ini digunakan oleh aktor untuk menampilkan data pengguna.
<b>Primary Actor</b>	Admin Dinas Perkebunan
<b>Supprting Actor</b>	-
<b>Basic Flow</b>	<ol style="list-style-type: none"> <li>1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan data pengguna</li> <li>2. Sistem menampilkan antarmuka pengelolaan data pengguna</li> <li>3. Sistem memberikan pilihan untuk melakukan tambah, ubah atau hapus data pengguna</li> </ol>

	<p>4. Aktor memilih untuk melakukan tambah data pengguna</p> <p>A-2 Aktor memilih untuk melakukan ubah data pengguna</p> <p>A-3 Aktor memilih untuk melakukan hapus data pengguna</p> <p>5. Aktor menginputkan data pengguna</p> <p>6. Aktor meminta sistem untuk menyimpan data pengguna yang telah diinputkan</p> <p>7. Sistem mengecek data pengguna yang telah diinputkan</p> <p>E-1 Nama pengguna yang diinputkan telah terpakai</p> <p>8. Sistem menyimpan data pengguna ke database</p> <p>9. Use Case selesai</p>
<b>Alternative Flow</b>	<p>A-1 Aktor memilih untuk melakukan ubah data pengguna</p> <p>1. Aktor menginputkan nama pengguna yang akan diubah</p> <p>2. Sistem menampilkan detail data pengguna</p> <p>3. Aktor mengubah data pengguna</p> <p>4. Aktor meminta sistem untuk menyimpan data pengguna yang telah diubah</p> <p>5. Sistem menyimpan data pengguna yang telah diubah ke database</p> <p>6. Berlanjut ke Basic Flow langkah ke-9</p> <p>A-2 Aktor memilih untuk melakukan hapus data pengguna</p> <p>1. Aktor memilih melakukan hapus data pengguna</p> <p>2. Sistem menampilkan semua data pengguna yang diperlukan oleh user</p> <p>3. Aktor memilih data yang akan dihapus</p>

	4. Sistem melakukan konfirmasi kepada aktor 5. Sistem menghapus data dari database 6. Berlanjut ke Basic Flow langkah ke-9
<b>Error Flow</b>	E-1 Nama pengguna yang diinputkan telah terpakai  1. Sistem menampilkan peringatan bahwa nama pengguna yang diinputkan telah terpakai 2. Kembali ke Basic Flow langkah 4
<b>PreConditions</b>	1. Use Case Login telah dilakukan 2. Aktor berhasil memasuki sistem
<b>PostConditions</b>	Data pengguna bertambah, berubah atau telah terhapus

### 3.2.7 Use case Spesification :Kelola Jumlah Hasil Kebun

<b>Brief Description</b>	Use Case ini digunakan oleh aktor untuk melakukan pengelolaan data jumlah hasil kebun.
<b>Primary Actor</b>	Admin Pemilik Perkebunan
<b>Supprting Actor</b>	-
<b>Basic Flow</b>	1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan data jumlah hasil kebun milik pribadi seusai dengan user tertentu.  2. Sistem menampilkan antarmuka pengelolaan data jumlah hasil kebun  3. Sistem memberikan pilihan untuk melakukan tambah, ubah atau hapus data jumlah hasil kebun  4. Aktor memilih untuk melakukan tambah data jumlah hasil kebun  A-2 Aktor memilih untuk melakukan ubah data jumlah hasil kebun  A-3 Aktor memilih untuk melakukan

	<p>hapusdata jumlah hasil kebun</p> <ol style="list-style-type: none"> <li>5. Aktor menginputkan data jumlah hasil kebun</li> <li>6. Aktor meminta sistem untuk menyimpan data jumlah hasil kebun yang telah diinputkan</li> <li>7. Sistem mengecek data jumlah hasil kebun yang telah diinputkan</li> </ol> <p>E-1 Nama jumlah hasil kebun yang diinputkan telah terpakai</p> <ol style="list-style-type: none"> <li>8. Sistem menyimpan data jumlah hasil kebun ke database</li> <li>9. Use Case selesai</li> </ol>
<b>Alternative Flow</b>	<p>A-1 Aktor memilih untuk melakukan ubah data jumlah hasil kebun</p> <ol style="list-style-type: none"> <li>1. Aktor menginputkan nama jumlah hasil kebun yang akan diubah</li> <li>2. Sistem menampilkan detail data jumlah hasil kebun</li> <li>3. Aktor mengubah data jumlah hasil kebun</li> <li>4. Aktor meminta sistem untuk menyimpan data jumlah hasil kebun yang telah diubah</li> <li>5. Sistem menyimpan data jumlah hasil kebun yang telah diubah ke database</li> <li>6. Berlanjut ke Basic Flow langkah ke-9</li> </ol> <p>A-2 Aktor memilih untuk melakukan hapus data jumlah hasil kebun</p> <ol style="list-style-type: none"> <li>7. Aktor memilih melakukan hapus data jumlah hasil kebun</li> <li>8. Sistem menampilkan semua data jumlah hasil kebun yang diperlukan oleh user</li> <li>9. Aktor memilih data yang akan dihapus</li> <li>10. Sistem melakukan konfirmasi kepada aktor</li> <li>11. Sistem menghapus data dari database</li> </ol>

	12. Berlanjut ke Basic Flow langkah ke-9
<b>Error Flow</b>	E-1 Data Jumlah hasil kebun yang diinputkan telah terpakai 1. Sistem menampilkan peringatan bahwa nama data tersebut telah ada 2. Kembali ke Basic Flow langkah 4
<b>PreConditions</b>	1. Use Case Login telah dilakukan 2. Aktor berhasil memasuki sistem
<b>PostConditions</b>	Data jumlah hasil kebun bertambah, berubah atau telah terhapus

### 3.2.8 Use case Spesification :Tampil PetaPerkebunan

<b>Brief Description</b>	Use Case ini digunakan oleh aktor untuk menampilkan peta dan data perkebunan.
<b>Primary Actor</b>	Kepala Dinas Perkebunan dan User Umum
<b>Supprting Actor</b>	-
<b>Basic Flow</b>	1. Use Case ini dimulai ketika aktor memilih untuk melakukan tampil data perkebunan 2. Sistem menampilkan antarmuka tampil data perkebunan 3. Aktor memilih menampilkan data perkebunan berdasarkan wilayah kabupaten A-1 Aktor memilih menampilkan data perkebunan berdasarkan wilayah kecamatan A-2 Aktor memilih menampilkan data perkebunan berdasarkan hasil perkebunan 4. Sistem menampilkan <i>report</i> data perkebunan dari wilayah kabupaten yang dipilih aktor 5. Use Case selesai
<b>Alternative Flow</b>	A-1 Aktor memilih menampilkan data perkebunan berdasarkan wilayah kecamatan

	<ol style="list-style-type: none"> <li>1. Aktor memilih menampilkan <i>report</i> hasil perkebunan</li> <li>2. Aktor meminta sistem untuk menampilkan <i>report</i> wilayah kecamatan</li> <li>3. Sistem menampilkan <i>report</i> berdasarkan hasil perkebunan</li> <li>4. Berlanjut ke Basic Flow langkah ke-7</li> </ol> <p>A-2 Aktor memilih menampilkan data perkebunan berdasarkan hasil perkebunan</p> <ol style="list-style-type: none"> <li>1. Aktor memilih menampilkan <i>report</i> hasil perkebunan</li> <li>2. Aktor meminta sistem untuk menampilkan <i>report</i> hasil perkebunan</li> <li>3. Sistem menampilkan <i>report</i> berdasarkan hasil perkebunan</li> <li>4. Berlanjut ke Basic Flow langkah ke-7</li> </ol>
<b>Error Flow</b>	-
<b>PreConditions</b>	<ol style="list-style-type: none"> <li>1. Use Case Login telah dilakukan</li> <li>2. Aktor berhasil memasuki sistem</li> </ol>
<b>PostConditions</b>	<i>Report</i> data perkebunan berdasarkan wilayah dan hasil perkebunan dapat ditampilkan.

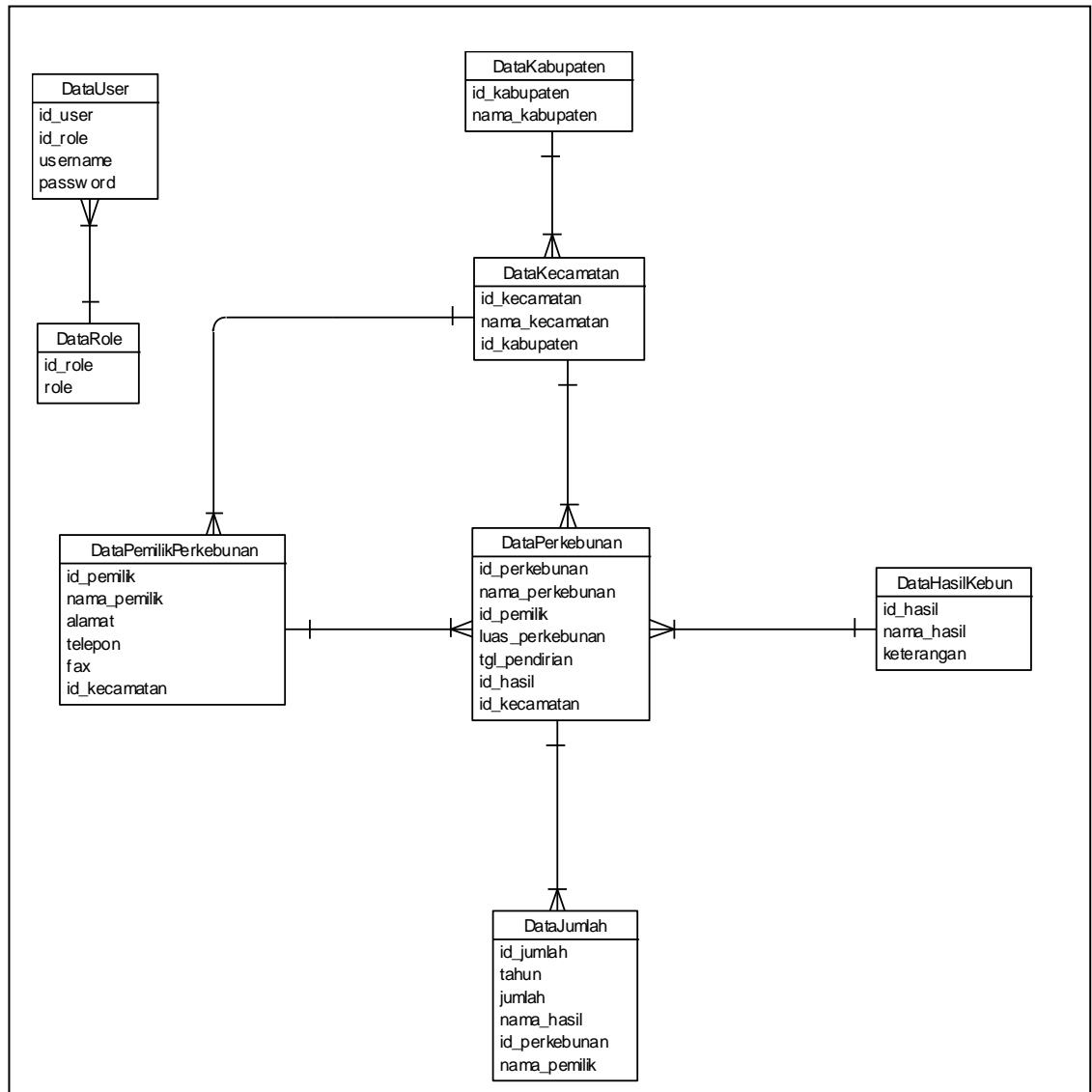
### 3.2.9 Use case Spesification :Tambah Point Perkebunan

<b>Brief Description</b>	Use Case ini digunakan oleh aktor untuk menambahkan point pada peta perkebunan.
<b>Primary Actor</b>	Kepala Dinas Perkebunan
<b>Supprting Actor</b>	-
<b>Basic Flow</b>	<ol style="list-style-type: none"> <li>1. Use Case ini dimulai ketika aktor memilih untuk melakukan add point pada peta perkebunan</li> </ol>

	<p>2. Sistem menampilkan antarmuka peta perkebunan</p> <p>3. Aktor memilih nama perkebunan yang akan dilakukan penambahan point pada peta perkebunan</p> <p>4. Sistem menampilkan nama kecamatan dan kabupaten dari nama perkebunan yang dipilih</p> <p>5. Aktor melakukan penambahan titik nama perkebunan tersebut</p> <p>E-1 Aktor salah melakukan add point pada kecamatan tertentu</p> <p>E-2 Aktor salah melakukan add point pada kabupaten tertentu</p> <p>6. Use Case selesai</p>
<b>Alternative Flow</b>	-
<b>Error Flow</b>	<p>1. E-1 Aktor salah melakukan add point pada kecamatan tertentu</p> <p>1. Sistem menampilkan pesan kesalahan input kecamatan</p> <p>2. Kembali Ke Basic Flow langkah ke-3</p> <p>2. E-2 Aktor salah melakukan add point pada kabupaten tertentu</p> <p>1. Sistem menampilkan pesan kesalahan input kabupaten</p> <p>2. Kembali Ke Basic Flow langkah ke-3</p>
<b>PreConditions</b>	<p>1. Use Case Login telah dilakukan</p> <p>2. Aktor berhasil memasuki sistem</p>
<b>PostConditions</b>	<i>Point</i> atau Titik Nama Perkebunan tertentu telah berhasil dimasukkan dalam peta perkebunan.

## 4 Analisis Kebutuhan Data

### 4.1 ERD (Entity Relationship Diagram)



Gambar 3 Entity Relationship Diagram ( ERD )



## 4.2 Data Definition

### 4.2.1 Data DATAUSER

#### 4.2.1.1 Elemen Data id\_user

Representasi	Domain	Range	Format	Presisi	Struktur Data
Untuk id userdari user	Text	0-9,a-z,A-Z	-	-	Varchar (10)

#### 4.2.1.2 Elemen Data username

Representasi	Domain	Range	Format	Presisi	Struktur Data
Untuk usernamedari user	Text	0-9,a-z,A-Z	-	-	Varchar (10)

#### 4.2.1.3 Elemen Data password

Representasi	Domain	Range	Format	Presisi	Struktur Data
Untuk passworddari user	Text	0-9,a-z,A-Z	-	-	Varchar (10)

#### 4.2.1.4 Elemen Data id\_role

Representasi	Domain	Range	Format	Presisi	Struktur Data
Untuk id roledari user	Text	0-9	-	-	Integer

### 4.2.2 Data DATAPEMILIKPERKEBUNAN

#### 4.2.2.1 Elemen Data id\_pemilik

Representasi	Domain	Range	Format	Presisi	Struktur Data
Untuk idpemilik perkebunan	Text	0-9,a-z,A-Z	-	-	Varchar (25)

#### 4.2.2.2 Elemen Data nama\_pemilik

Representasi	Domain	Range	Format	Presisi	Struktur Data

Untuk nama pemilik perkebunan	Text	0-9,a-z,A-Z	-	-	Varchar (50)
-------------------------------------	------	-------------	---	---	--------------

#### 4.2.2.3 Elemen Data alamat

Representasi	Domain	Range	Format	Presisi	Struktur Data
Untuk alamat pemilik perkebunan	Text	0-9,a-z,A-Z	-	-	Varchar (100)

#### 4.2.2.4 Elemen Data telepon

Representasi	Domain	Range	Format	Presisi	Struktur Data
Untuk telepon pemilik perkebunan	Text	0-9,a-z,A-Z	-	-	Integer

#### 4.2.2.5 Elemen Data fax

Representasi	Domain	Range	Format	Presisi	Struktur Data
Untuk faximile pemilik perkebunan	Text	0-9,a-z,A-Z	-	-	Integer

#### 4.2.2.6 Elemen Data id kecamatan

Representasi	Domain	Range	Format	Presisi	Struktur Data
Untuk id kecamatan pemilik perkebunan	Text	0-9,a-z,A-Z	-	-	Varchar (50)

### 4.2.3 Data DATAPERKEBUNAN

#### 4.2.3.1 Elemen Data id\_perkebunan

Representasi	Domain	Range	Format	Presisi	Struktur Data
--------------	--------	-------	--------	---------	---------------

Untuk id perkebunan	Text	0-9,a-z,A-Z	-	-	Varchar (25)
---------------------	------	-------------	---	---	--------------

#### 4.2.3.2 Elemen Data nama\_perkebunan

Representasi	Domain	Range	Format	Presisi	Struktur Data
Untuk nama dari perkebunan	Text	0-9,a-z,A-Z	-	-	Varchar (50)

#### 4.2.3.3 Elemen Data luas\_perkebunan

Representasi	Domain	Range	Format	Presisi	Struktur Data
Untuk luas area perkebunan	Nilai	0-9	-	-	Integer

#### 4.2.3.4 Elemen Data tanggal\_pendirian

Representasi	Domain	Range	Format	Presisi	Struktur Data
Untuk tanggal didirikannya perkebunan	Date	-	mm-dd-yy	-	Integer

#### 4.2.3.5 Elemen Data id\_pemilik

Representasi	Domain	Range	Format	Presisi	Struktur Data
Untuk id pemilikdata perkebunan	Date	-	mm-dd-yy	-	Integer

#### 4.2.3.6 Elemen Data id\_hasil

Representasi	Domain	Range	Format	Presisi	Struktur Data
Untuk id hasilperkebunan	Date	-	mm-dd-yy	-	Integer

#### 4.2.3.7 Elemen Data id\_kecamatan

Representasi	Domain	Range	Format	Presisi	Struktur Data
Untuk tanggal didirikannya	Date	-	mm-dd-yy	-	Integer

perkebunan					
------------	--	--	--	--	--

#### 4.2.4 Data DATAHASILKEBUN

##### 4.2.4.1 Elemen Data id\_hasil

Representasi	Domain	Range	Format	Presisi	Struktur Data
Untuk id jenis dari hasil kebun	Text	0-9,a-z,A-Z	-	-	Varchar (25)

##### 4.2.4.2 Elemen Data nama\_hasil

Representasi	Domain	Range	Format	Presisi	Struktur Data
Untuk nama dari hasil kebun	Text	0-9,a-z,A-Z	-	-	Varchar (50)

##### 4.2.4.3 Elemen Data keterangan

Representasi	Domain	Range	Format	Presisi	Struktur Data
Untuk keterangan tertentu hasil kebun	Nilai	0-9,a-z,A-Z	-	-	Varchar(200)

#### 4.2.5 Data DATAKABUPATEN

##### 4.2.5.1 Elemen Data id\_kabupaten

Representasi	Domain	Range	Format	Presisi	Struktur Data
Untuk id dari kabupaten	Nilai	0-9	-	-	Integer

##### 4.2.5.2 Elemen Data nama\_kabupaten

Representasi	Domain	Range	Format	Presisi	Struktur Data
Untuk nama dari kabupaten	Text	0-9,a-z,A-Z	-	-	Varchar (50)

#### 4.2.6 Data DATAKECAMATAN

##### 4.2.6.1 Elemen Data nama\_kecamatan

Representasi	Domain	Range	Format	Presisi	Struktur Data
Untuk nama dari kecamatan	Text	0-9,a-z,A-Z	-	-	Varchar (30)

##### 4.2.6.2 Elemen Data id\_kecamatan

Representasi	Domain	Range	Format	Presisi	Struktur Data
Untuk id dari kecamatan	Nilai	0-9	-	-	Integer

##### 4.2.6.3 Elemen Data id\_kabupaten

Representasi	Domain	Range	Format	Presisi	Struktur Data
Untuk id dari kabupaten	Nilai	0-9	-	-	Integer

#### 4.2.7 Data DATAROLE

##### 4.2.7.1 Elemen Data id\_role

Representasi	Domain	Range	Format	Presisi	Struktur Data
Untuk id dari jabatan tertentu	Nilai	0-9	-	-	Integer

##### 4.2.6.4 Elemen Data role

Representasi	Domain	Range	Format	Presisi	Struktur Data
Untuk deskripsi dari jabatan tertentu	Text	0-9,a-z,A-Z	-	-	Varchar (50)

#### 4.2.8 Data DATAJUMLAH

##### 4.2.8.1 Elemen Data id\_jumlah

Representasi	Domain	Range	Format	Presisi	Struktur Data
Untuk id jumlah dari data jumlah	Text	0-9,a-z,A-Z	-	-	Varchar (15)

##### 4.2.8.2 Elemen Data tahun

Representasi	Domain	Range	Format	Presisi	Struktur Data
Untuk tahun dari data jumlah	Nilai	0-9	-	-	Integer

##### 4.2.8.3 Elemen Data jumlah

Representasi	Domain	Range	Format	Presisi	Struktur Data
Untuk jumlah dari data jumlah	Nilai	0-9	-	-	Float

##### 4.2.8.4 Elemen Data nama\_hasil

Representasi	Domain	Range	Format	Presisi	Struktur Data
Untuk nama_hasil dari data jumlah	Text	0-9,a-z,A-Z	-	-	Varchar (50)

##### 4.2.8.5 Elemen Data ID\_PERKEBUNAN

Representasi	Domain	Range	Format	Presisi	Struktur Data
Untuk id_perkebunan dari data jumlah	Text	0-9,a-z,A-Z	-	-	Varchar (25)

##### 4.2.8.6 Elemen Data Nama\_Pemilik

Representasi	Domain	Range	Format	Presisi	Struktur Data
Untuk nama_pemilik dari data jumlah	Text	0-9,a-z,A-Z	-	-	Varchar (50)