BAB 4
IMPLEMENTASI DAN PENGUJIAN SISTEM

4.1 Implementasi Sistem

4.1.1 Implementasi Perangkat keras

Perangkat keras yang digunakan pada saat mengimplementasikan dari Sistem Pengklasifikasian Objek Wisata di Dinas Pariwisata dan Kebudayaan Jawa Barat adalah sebagai berikut :

1. Processor Intel® Core™ CPU i3 @ 2.27 GHz,
2. RAM 4 GB,
3. VGA Intel® Graphics Media Accelator HD (Core i3),
4. Hard Disk 300 GB,
5. Monitor 14 Inch, dan
6. Modem Smartfren 153 Kbps.

4.1.2 Implementasi Perangkat Lunak

Perangkat keras yang digunakan pada saat mengimplementasikan dari Sistem Pengklasifikasian Objek Wisata di Dinas Pariwisata dan Kebudayaan Jawa Barat adalah sebagai berikut :

1. Sistem Operasi Windows 7 Ultimate,
2. Bahas Pemrograman PHP 5.3.8,
3. Database MySQL 5.5.16,
4. Browser Google Chrome,
5. Macromedia Dreamweaver Cs5 dan Notepad ++,
6. Tools Diagram Star UML dan,
7. Tools Perancangan antarmuka CorelDraw X5,

4.1.3 Implementasi Basis Data

Pembuatan database dilakukan dengan menggunakan aplikasi DBMS MySQL. Implementasi database dalam bahasa SQL adalah sebagai berikut :

Tabel 4.1 Implementasi Basis Data

No.	Nama Tabel	SQL
1	Tabel_akun	CREATE TABLE IF NOT EXISTS `akun` (`nama_pengguna` char(30) NOT NULL, `kata_kunci` char(100) NOT NULL, PRIMARY KEY (`nama_pengguna`)) ENGINE=InnoDB DEFAULT CHARSET=latin1;
2	Tabel_Data_Latih	CREATE TABLE IF NOT EXISTS `data_latih` (`id_data_latih` tinyint(4) NOT NULL AUTO_INCREMENT, `id_karakter` tinyint(4) NOT NULL,`nama_objek_wisata` varchar(150) NOT NULL,`deskripsi` longtext NOT NULL, `keyword` longtext NOT NULL, PRIMARY KEY (`id_data_latih`), KEY `fk_id_karakter2` (`id_karakter`)) ENGINE=InnoDB DEFAULT CHARSET=latin1;
3	Tabel_Karakter	CREATE TABLE IF NOT EXISTS `karakter` (`id_karakter` tinyint(4) NOT NULL AUTO_INCREMENT, `nama_karakter` varchar(30) NOT NULL, PRIMARY KEY (`id_karakter`), UNIQUE KEY `nama_karakter` (`nama_karakter`)) ENGINE=InnoDB DEFAULT CHARSET=latin1;
4	Tabel_Kata_Dasar	CREATE TABLE IF NOT EXISTS `kata_dasar` (`id_kata_dasar` int(10) NOT NULL AUTO_INCREMENT, `kata` varchar(20) NOT NULL, `tipe` varchar(20) NOT NULL, PRIMARY KEY (`id_kata_dasar`)) ENGINE=InnoDB DEFAULT CHARSET=latin1;
5	Tabel_Kota_Kabupaten	CREATE TABLE IF NOT EXISTS `kota_kabupaten` (`kota_kabupaten` varchar(100) NOT NULL, `propinsi_id` tinyint(4) NOT NULL, PRIMARY KEY (`kota_kabupaten`),

No.	Nama Tabel	SQL
		KEY `kota_id` (`propinsi_id`)) ENGINE=InnoDB DEFAULT CHARSET=latin1;
6	Tabel_Objek_Wisata	CREATE TABLE IF NOT EXISTS `objek_wisata` (`id_objek_wisata` int(11) NOT NULL AUTO_INCREMENT, `id_karakter` tinyint(4) NOT NULL, `kota_kabupaten` varchar(100) NOT NULL, `nama_objek_wisata` varchar(150) NOT NULL, `deskripsi` longtext NOT NULL, `gambar` varchar(300) DEFAULT NULL, `tanggal_input` datetime NOT NULL, PRIMARY KEY (`id_objek_wisata`), UNIQUE KEY `nama_objek_wisata` (`nama_objek_wisata`), KEY `fk_id_karakter` (`id_karakter`), KEY `fk_kota_kabupaten` (`kota_kabupaten`)) ENGINE=InnoDB DEFAULT CHARSET=latin;
7	Tabel_Probabilitas	CREATE TABLE IF NOT EXISTS `probabilitas` (`id_probabilitas` bigint(20) NOT NULL AUTO_INCREMENT, `id_karakter` tinyint(4) NOT NULL, `kata` varchar(20) NOT NULL,`probabilitas` double NOT NULL, PRIMARY KEY (`id_probabilitas`), KEY `fk_id_karakter3` (`id_karakter`)) ENGINE=InnoDB DEFAULT CHARSET=latin1

4.1.4 Implementasi Kelas

Tabel 4.2 Implementasi Kelas

No.	Nama Kelas	Nama File Fisik	Nama File Executable
1.	Akun	Akun.php	Akun.php
2.	Database	Database.php	Database.php
3.	Data Latih	Data Latih.php	Data Latih.php
4.	Karakter	Karakter.php	Karakter.php

No.	Nama Kelas	Nama File Fisik	Nama File Executable
5.	Kota	Kota.php	Kota.php
6.	Menu	Menu.php	Menu.php
7.	NVB	NVB.php	NVB.php
8.	ObjekWisata	ObjekWisata.php	ObjekWisata.php
9.	Preprocessing	Preprocessing.php	Preprocessing.php
10.	Stemming	Stemming.php	Stemming.php

4.1.5 Implementasi Antarmuka Admin

Implementasi antarmuka dilakukan dengan menampilkan setiap tampilan dari program yang dibangun dan pengkodeannya dalam bentuk file program. Berikut ini adalah file-file yang digunakan dalam membuka tampilan antamuka pengguna admin.

Tabel 4.3 Implementasi Antarmuka Admin

No	Nama Antarmuka Untuk Admin	Deskripsi	Nama File Executable
1.	Login	Menampilkan form isian nama pengguna dan kata kunci.	login.php akun.php reset.css style.css invalid.css jquery-1.3.2.min.js simpla.jquery.configuration.js facebox.js
2.	Beranda	Menampilkan Halaman Utama situs admin.	Index.php Karakter.php ObjekWisata.php DataLatih.php Menu.php reset.css style.css invalid.css jquery-1.3.2.min.js simpla.jquery.configuration.js facebox.js
3.	Per Kategori	Menampilkan informasi objek wisata berdasarkan kategori.	PerKategori.php Menu.php ObjekWisata.php reset.css

No	Nama Antarmuka Untuk Admin	Deskripsi	Nama File Executable
			style.css invalid.css jquery-1.3.2.min.js simpla.jquery.configuration.js facebox.js
4.	Daftar Karakter	Menampilkan daftar nama data karakter.	daftarKarakter.php Karakter.php Menu.php reset.css style.css invalid.css jquery-1.3.2.min.js simpla.jquery.configuration.js facebox.js
5.	up-del-kar	Halaman untuk tambah karakter.	up-del-kar.php Karakter.php daftarKarakter.php
6.	DaftarKota	Menampilkan daftar nama data kota/kab.	DaftarKota.php Kota.php Menu.php reset.css style.css invalid.css jquery-1.3.2.min.js simpla.jquery.configuration.js facebox.js
7.	up-del-kota	Halaman untuk tambah, dan cari kota/kab.	up-del-kota.php DaftarKota.php Kota.php
8.	DaftarDataLatih	Menampilkan daftar nama data latih.	DaftarDataLatih.php Menu.php DataLatih.php Preprocessing.php Stemming.php NVB.php reset.css style.css invalid.css jquery-1.3.2.min.js simpla.jquery.configura

No	Nama Antarmuka Untuk Admin	Deskripsi	Nama File Executable
			tion.js facebox.js
9.	up-del-dat	Halaman untuk tambah, edit, dan hapus data latih.	up-del-dat.php DaftarDataLatih.php DataLatih.php
10	DaftarObjekWisata	Menampilkan daftar nama data uji objek wisata hasil klasifikasi.	DaftarObjekWisata.php Menu.php ObjekWisata.php Preprocessing.php Stemming.php NVB.php reset.css style.css invalid.css jquery-1.3.2.min.js simpla.jquery.configuration.js facebox.js
11	up-del-obj	Halaman untuk tambah, edit, cari dan hapus data uji objek wisata.	up-del-obj.php DaftarObjekWisata.php Menu.php ObjekWisata.php
12	logout	Keluar dari sisterm dan menampilkan *form login.*	Logout.php

4.2 Pengujian Sistem

Pengujian sistem merupakan tahapan untuk melakukan serangkaian tes untuk mencoba sistem yang telah dibangun dengan tujuan untuk menemukan kesalahan-kesalahan atau kekurangan-kekurangan pada perangkat lunak yang diuji.

Dalam melakukan pengujian terhadap sistem pengklasifikasian objek wisata dilakukan dengan dua macam teknik pengujian, yaitu pengujian *black box*. Adapun pengujian *black box* berfokus pada persyaratan fungsional dari sebuah sistem yang dibangun dan menemukan kesalahan program.

4.2.1 Pengujian *Alpha*

Pada tahap ini pengujian dilakukan untuk memastikan sistem pengklasifikasian objek wisata Jawa Barat telah berjalan dengan benar dan sesuai dengan kebutuhan yang diinginkan.Pengujian *alpha* menitikberatkan kepada persyaratan fungsional perangkat lunak.

4.2.1.1 Skenario Pengujian Aplikasi

Skenario pengujian menjelaskan pengujian terhadap sistem yang ada pada sistem pengklasifikasian objek wisata Jawa Barat. Skenario pengujian dari sistem ini dapat dilihat pada tabel 4.4.

Tabel 4.4 Rencana Penguji Admin

NO	Kelas Uji	Butir Uji	Jenis Pengujian
1.	Login	a. Mengisi nama pengguna dan kata kunci	*Black Box*
		b. Verifikasi data login (nama pengguna dan kata kunci)	*Black Box*
		c. Percobaan masuk ke salah satu halaman admin tanpa melakukan pengisian ke proses login.	*Black Box*
2.	Menu Beranda	Menampilkan informasi jumlah objek wisata secara keseluruhan, menampilkan informasi objek wisata sudah dikategorikan berdasarkan 8 karakter dan menampilkan informasi jumlah data latih.	*Black Box*
3.	Menu Data Karakter	a. Menampilkan data Karakter keseluruhan.	*Black Box*
		b. Melakukan proses tambah data karakter.	*Black Box*
4.	Menu Data Kota/Kab	a. Menampilkan data Kota/kab keseluruhan.	*Black Box*
		b. Melakukan proses tambah data Kota/kab.	*Black Box*
		c. Melakukan proses pencarian data Kota/kab.	*Black Box*
		a. Menampilkan data latih keseluruhan.	*Black Box*

NO	Kelas Uji	Butir Uji	Jenis Pengujian
5.	Menu Data Latih	b. Melakukan proses tambah data latih dengan melakukan proses *text mining* yaitu *preprocessing* terlebih dahulu seperti proses *case folding*, *tokenizing*, *filtering*, dan *stemming*. dan diklasifikasikan nvb sebagai tahap pembelajaran lalu dibuat keyword dan disimpan ke dalam tabel model probabilitas.	*Black Box*
		c. Melakukan proses ubah data latih dengan melakukan proses *text mining* yaitu *preprocessing* terlebih dahulu seperti proses *case folding*, ,*tokenizing*, *filtering*, dan *stemming*. dan diklasifikasikan nvb sebagai tahap pembelajaran lalu dibuat keyword baru dan disimpan ke dalam tabel model probabilitas.	*Black Box*
		d. Melakukan proses hapus data latih dengan menghapus juga keyword yang ada di tabel model probabilitas.	*Black Box*
		e. Menampilkan informasi secara detail dari salah satu data latih.	*Black Box*
6.	Menu Data Uji	a. Menampilkan data uji objek wisata keseluruhan.	*Black Box*
		b. Melakukan proses tambah data uji objek wisata dengan melakukan proses *text mining* yaitu *preprocessing* terlebih dahulu seperti proses *case folding*, *tokenizing*, *filtering*, dan *stemming*. dan diklasifikasikan nvb sebagai	*Black Box*

NO	Kelas Uji	Butir Uji	Jenis Pengujian
		tahap klasifikasi lalu diberi label kategori dari 8 karakter yang paling maksimum.	
		c. Melakukan proses ubah data latih dengan melakukan proses *text mining* yaitu *preprocessing* terlebih dahulu seperti proses *case folding*, *tokenizing*, *filtering*, dan *stemming*. dan diklasifikasikan nvb sebagai tahap klasifikasi lalu diberi label kategori yang baru dari 8 karakter yang paling maksimum.	*Black Box*
		d. Melakukan proses hapus data uji objek wisata.	*Black Box*
		e. Menampilkan informasi secara detail dari salah satu data uji objek wisata.	*Black Box*
		f. Melakukan proses pencarian data uji objek wisata berdasarkan nama objek wisata.	*Black Box*

4.2.1.2 Kasus dan Hasil Pengujian *Black Box*

Kasus dan hasil uji perangkat lunak yang sudah dibangun dengan menggunakan metode *black box* berdasarkan pengamatan yang dilakukan dan menghasilkan kesimpulan dari pengujian tersebut, diantaranya sebagai berikut :

1. Pengujian *Login*

Pengujian login dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian login dapat dilihat pada tabel 4.5.

Tabel 4.5 Pengujian Login Admin

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
Data Login berupa Nama pengguna: dimasrickyf kata kunci : 10108401	Data login dicari dalam *database*, data nama pengguna dan kata kunci *valid* serta dapat menampilkan halaman utama admin.	Nama pengguna dan kata kunci benar dan dapat masuk ke menu halaman admin	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Nama pengguna dan kata kunci tidak diisi	Tidak dapat login/masuk dan menampilkan pesan kesalahan	Sistem menampilkan pesan kesalahan "nama pengguna dan kata kunci harus diisi"	[√] Diterima [ ] Ditolak
Memasukan data login nama pengguna dan kata kunci yang tidak sesuai	Tidak dapat login/masuk dan menampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "Kombinasi nama pengguna dan kata kunci salah !!"	[√] Diterima [ ] Ditolak
Memasukan data login nama pengguna dan tidak mengisi kata kunci.	Tidak dapat login/masuk dan menampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "kata kunci harus diisi"	[√] Diterima [ ] Ditolak
Memasukan data login hanya mengisi kata kunci saja	Tidak dapat login/masuk dan menampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "nama pengguna harus diisi"	[√] Diterima [ ] Ditolak

2. Pengujian Tambah Data Karakter

Pengujian tambah data karakter dilakukan dengan memasukkan data untuk mendapatkan hasil yang diinginkan. Pengujian dapat dilihat pada tabel 4.6 dibawah ini.

Tabel 4.6 Pengujian Tambah Data Karakter

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi pada data Karakter contoh : Nama Karakter : Hutan	Menyimpan data karakter ke *database*	Sistem menampilkan pesan "karakter Hutan telah berhasil disimpan"	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Karakter dengan nama karakter tidak diisi contoh : nama karakter : -	Menolak menyimpan data karakter ke *database*	Sistem akan menampilkan pesan kesalahan " Gagal menyimpan data karakter nama karakter belum diisi"	[√] Diterima [ ] Ditolak
Data karakter yang diisi nama karakter yang konflik. Contoh Nama Karakter : Hutan	Menolak menyimpan data karakter ke *database*	Sistem akan menampilkan pesan kesalahan " Gagal menyimpan data karakter karakter Hutan sudah tersedia"	[√] Diterima [ ] Ditolak

3. Pengujian Tambah Data Kota/Kab

Pengujian tambah data kota/kab dilakukan dengan memasukkan data untuk mendapatkan hasil yang diinginkan. Pengujian dapat dilihat pada tabel 4.7 dibawah ini.

Tabel 4.7 Pengujian Tambah Data Kota/Kab

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi pada data Kota/Kab contoh. Nama Provinsi : Jawa Barat Nama Kota/Kab : Kabupaten	Menyimpan data kota/kab ke *database*	Sistem menampilkan "data kota/kab kabupaten bandung berhasil	[√] Diterima [ ] Ditolak

Bandung		disimpan”	
Kasus dan Hasil Uji (Data Salah)			
Data Kota/Kab tidak diisi	Menolak menyimpan data kota/kab ke *database*	Sistem akan menampilkan pesan kesalahan “ Gagal menyimpan data kota/kab belum diisi”	[√] Diterima [ ] Ditolak
Data Kota/Kab diisi konflik	Menolak menyimpan data kota/kab ke *database* karena sudah tersedia	Sistem akan menampilkan pesan kesalahan “ Gagal menyimpan data kota/kab sudah tersedia”	[√] Diterima [ ] Ditolak

4. Pengujian Tambah Data Latih (Proses NVB pembelajaran)

Pengujian tambah data latih dilakukan dengan memasukkan data untuk mendapatkan hasil yang diinginkan. Pengujian dapat dilihat pada tabel 4.8 dibawah ini.

Tabel 4.8 Pengujian Tambah Data Latih (Proses NVB pembelajaran)

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi data latih contoh : - Pilih Karakter : Hutan - Nama Objek Wisata : Taman Nasional Gunung Gede Pangrango - Deskripsi :” Taman Nasional Gunung Gede Pangrango ditetapkan oleh unesco sebagai cagar biosfir pada tahun 1977.”	Menyimpan data latih ke *database*, dengan melakukan proses *text mining*, yaitu *tokenizing*, *filtering*, dan *stemming* lalu dibuat indeks pembuatan *keyword* dan juga melakukan klasifikasi nvb pembelajaran data latih dan dibuat ke dalam model probabilitas.	Sistem menampilkan pesan “ Data latih telah berhasil disimpan dan diklasifikasikan NVB sebagai pembelajaran data latih “	[√] Diterima [ ] Ditolak

Kasus dan Hasil Uji (Data Salah)			
Data Latih tidak diisi	Menolak menyimpan data latih ke *database* karena tidak memilih karakter, tidak mengisi nama objek wisata dan deskripsi.	Sistem menampilkan pesan kesalahan " Gagal menyimpan data latih"	[√] Diterima [ ] Ditolak
Hanya memilih pilih karakter pada data latih saja. Contoh : - Pilih Karakter : Hutan - Nama Objek Wisata : - - Deskripsi : -	Menolak menyimpan data latih ke *database*. Karena hanya memilih karakter saja, tidak mengisi nama objek wisata dan deskripsi.	Sistem menampilkan pesan kesalahan " Gagal menyimpan data latih. Isi nama objek wisata terlebih dahulu"	[√] Diterima [ ] Ditolak
Hanya memilih pilih karakter dan nama objek wisata pada data latih saja. Contoh : - Pilih Karakter : Hutan - Nama Objek Wisata : Taman Nasional Gunung Gede Pangrango - Deskripsi : -	Menolak menyimpan data latih ke *database*. Karena hanya memilih karakter dan mengisi nama objek wisata dan tidak mengisi deskripsi.	Sistem menampilkan pesan kesalahan " Gagal menyimpan data latih. Isi deskripsi terlebih dahulu"	[√] Diterima [ ] Ditolak

5. Pengujian Ubah Data Latih

Pengujian ubah data latih dilakukan dengan megubah data untuk mendapatkan hasil yang diinginkan. Pengujian dapat dilihat pada tabel 4.9 dibawah ini.

Tabel 4.9 Pengujian Ubah Data Latih

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengubah deskripsi pada data Latih Contoh : Deskripsi : "Gua Pawon	Data latih dapat diubah dan disimpan kembali ke *database*, dengan melakukan proses *text mining*, yaitu *tokenizing, filtering,*	Sistem menampilkan pesan " data latih berhasil diubah dan diklasifikasikan	[√] Diterima [ ] Ditolak

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
merupakan salah satu temuan prasejarah yang dilindungi keberadaannya . Disini adanya temuan benda-benda prasejarah seperti alat-alat budaya peninggalan manusia”	dan *stemming* lalu dibuat indeks pembuatan *keyword* dan juga melakukan klasifikasi nvb pembelajaran data latih dan dibuat ke dalam model probabilitas.	ulang (nvb) pembelajaran dimasukkan ke model probabilitas “	
Kasus dan Hasil Uji (Data Salah)			
Mengubah Data Latih namun tidak diisi/kosong	Menolak menyimpan data latih ke *database*.	Sistem menampilkan pesan kesalahan“ Gagal menyimpan data latih.pilih karakter terlebih dahulu “	[√] Diterima [ ] Ditolak
Mengubah Data Latih namun diisi tidak lengkap	Menolak menyimpan data latih ke *database*.	Sistem menampilkan pesan kesalahan“ Gagal menyimpan data latih.pilih karakter terlebih dahulu / isi terlebih dahulu nama objek wisata / isi deskripsi terlebih dahulu“	[√] Diterima [ ] Ditolak

6. Pengujian Tambah Data Uji Klasifikasi (Classify) NVB

Pengujian tambah data uji objek wisata dilakukan dengan memasukkan data untuk mendapatkan hasil yang diinginkan. Pengujian dapat dilihat pada tabel 4.13 dibawah ini.

Tabel 4.10 Pengujian Tambah data uji klasifikasi nvb *(classify)*

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi data Uji Objek Wisata. Contoh : -Nama Objek Wisata : Telaga Nilam -Pilih Kota/Kab : Kabupaten kuningan -Deskripsi :" Telaga Nilam masih berada dikawasan Telaga Remis." file gambar : telaga.jpg	Menyimpan data uji objek wisata ke *database*, dengan melakukan proses *text mining*, yaitu *tokenizing, filtering,* dan *stemming* lalu dibuat indeks pembuatan *keyword* dan tahp terakhir melakukan klasifikasi nvb data uji sebagai proses pengklasifikasian dan berhasil menghasilkan label karakter yang sesuai.	Sistem akan menampilkan pesan " Objek Wisata dengan nama objek wisata taman nastelah diklasifikasikan NVB ke dalam karakter hutan dan telah berhasil disimpan"	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Uji Objek Wisata tidak diisi	Menolak menyimpan data uji objek wisata ke *database*.	Sistem menampilkan pesan kesalahan "Gagal mengklasifikasikan data uji objek wisata "	[√] Diterima [ ] Ditolak
Data Uji Objek Wisata diisi konflik	Menolak menyimpan data uji objek wisata ke *database*. karena sudah tersedia	Sistem menampilkan pesan kesalahan "Gagal mengklasifikasikan data uji objek wisata . Objek wisata telaga nilam sudah tersedia"	[√] Diterima [ ] Ditolak
Data Uji Objek Wisata diisi tidak lengkap	Menolak menyimpan data uji objek wisata ke *database*.	Sistem menampilkan pesan kesalahan "Gagal mengklasifikasikan data uji objek wisata . nama objek wisata belum diisi"	[√] Diterima [ ] Ditolak

7. Pengujian Ubah Data Uji

Pengujian ubah data uji objek wisata dilakukan dengan memasukkan data untuk mendapatkan hasil yang diinginkan. Pengujian dapat dilihat pada tabel 4.11 dibawah ini.

Tabel 4.11 Pengujian Ubah Data Uji

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengubah dan mengisi benar pada Data Uji Objek Wisata (Nama Objek Wisata, Pilih Kota/Kab, Deskripsi)	Data Uji Objek Wisata telah diubah dan disimpan ke *database*, dengan melakukan proses *text mining*, yaitu *tokenizing, filtering,* dan *stemming* lalu dibuat indeks pembuatan *keyword* dan tahp terakhir melakukan klasifikasi nvb data uji sebagai proses pengklasifikasian dan berhasil menghasilkan label karakter yang sesuai.	Sistem akan menampilkan pesan " Objek Wisata dengan nama objek wisata taman nastelah diklasifikasikan NVB ke dalam karakter hutan dan telah berhasil diubah"	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Mengubah Data Uji Objek Wisata namun tidak diisi	Menolak menyimpan data uji objek wisata ke *database*.	Sistem menampilkan pesan kesalahan "Gagal mengklasifikasikan data uji objek wisata "	[√] Diterima [ ] Ditolak
Mengubah Data Uji Objek Wisata dan diisi tidak lengkap	Menolak menyimpan data uji objek wisata ke *database*.	Sistem menampilkan pesan kesalahan "Gagal mengklasifikasikan data uji objek wisata . nama objek wisata belum diisi"	[√] Diterima [ ] Ditolak

8. Pengujian Hapus Data Uji

Pengujian hapus data uji objek wisata dilakukan dengan memilih data untuk mendapatkan hasil yang diinginkan. Pengujian dapat dilihat pada tabel 4.12 dibawah ini.

Tabel 4.12 Pengujian Hapus Data Uji

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memilih pada data uji objek wisata yang akan dihapus contoh : Nama objek wisata : telaga nilam	Menghapus data uji objek wisata dari *database*.	Sistem menampilkan pesan "Objek wisata nama objek wisata telaga nilam berhasil dihapus"	[√] Diterima [ ] Ditolak

9. Pengujian Penyajian detail Data Uji

Pengujian penyajian detail data uji objek wisata dilakukan dengan memilih data untuk mendapatkan hasil yang diinginkan. Pengujian dapat dilihat pada tabel 4.13 dibawah ini.

Tabel 4.13 Pengujian Penyajian detail Data Uji

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memilih salah satu detail dari daftar nama Data uji objek wisata	Menampilkan detail dari data uji objek wisata yang dipilih.	Sistem menampilkan detail penyajian deskripsi data uji objek wisata yang dipilih	[√] Diterima [ ] Ditolak

4.2.1.3 Kesimpulan Hasil Pengujian *Black Box*

Berdasarkan hasil pengujian dengan kasus uji diatas dapat ditarik kesimpulan sebagai berikut :

1. Jika data yang dimasukan benar maka sistem secara fungsional akan mengeluarkan keluaran sesuai dengan harapan.
2. Jika data yang dimasukan salah maka sistem akan mengeluarkan pesan kesalahan dan data keluaran tidak akan sesuai dengan harapan.

4.2.2 Pengujian *Beta*

Pengujian *Beta* merupakan pengujian yang dilakukan secara objektif dimana dilakukan pengujian secara langsung terhadap pengguna sistem dengan menggunakan wawancara mengenai kepada petugas dari pihak disparbud terhadap sistem yang telah dibangun.

4.2.2.1 Wawancara Admin

Wawancara dilakukan kepada petugas yang telah menggunakan sistem, yaitu Bapak Dante Syailendra (Staff IT). Wawancara dilakukan dengan mengajukan 4 pertanyaan, dan hasil wawancara dapat dilihat sebagai berikut :

1. Apakah sistem pengklasifikasian objek wisata Jawa Barat yang dibangun membantu anda dalam proses pengklasifikasian objek wisata berdasarkan kategori 8 karakter ?

   Jawaban : “ Bahwa sistem pengklasifikasian objek wisata sangat membantu sekali dalam melakukan proses pengklasifikasian, dimana objek wisata yang banyak akan sangat mudah jika diklasifikasikan secara otomatis dan tidak secara manual seperti yang dilakukan dulu saat pengklasifikasian.”
2. Bagaimana tampilan dari website ini sudah cukup menarik ?

   Jawaban : “ *Interface* dari website yang simple, dan enak dilihat. “
3. Apakah sistem pengklasifikasian objek wisata jawa barat ini mudah dipelajari dan digunakan?

Jawaban : Secara keseluruhan, halaman administrator sistem pengklasifikasian objek wisata mudah untuk dipelajari, interface yang user friendly, pengguna (admin) tidak mengalami kesulitan dalam menggunakannya.

4. Apakah sistem yang dibangun layak untuk digunakan?
   Jawaban : “ Bahwa sistem pengklasifikasian objek wisata layak digunakan karena mempermudah dalam melakukan proses pengklasifikasian.”

4.2.2.2 Kesimpulan Hasil Pengujian Beta

Hasil dari pengujian beta dapat disimpulkan bahwa sistem pengklasifikasian objek wisata yang telah dibangun dapat membantu pengguna khususnya admin disparbud untuk melakukan proses pengklasifikasian objek wisata berdasarkan kategori ke dalam 8 karakter, selain itu sistem pengklasifikasian objek wisata juga mudah untuk dipelajari dan sangat membantu sekali karena cukup cepat dalam proses pengklasifikasian dimana objek wisata di Jawa Barat sangat banyak sekali.