BAB 4
IMPLEMENTASI DAN PENGUJIAN SISTEM

4.1 Implementasi Sitem

Implementasi sistem merupakan tahap yang dilakukan untuk mengetahui sistem yang telah dibangun dapat beroperasi dengan baik. Tahap ini merupakan dimana sistem siap untuk digunakan. Setelah implementasi maka dilakukan pengujian terhadap sistem yang telah dibangun, pengujian tersebut dilakukan untuk mengetahui kekurangan yang ada pada sistem. Dari hasil pengujian tersebut bisa dijadikan sebagai bahan untuk pengembangan sistem selanjutnya.

Spesifikasi perangkat keras yang digunakan dalam pembangunan Manajemen Proyek di PT. Dua Ribu Satu Pangripta dapat lihat pada tabel dibawah ini :

Tabel 4. 1 Spesifikasi Perangkat Keras Yang Digunakan

No	Perangkat Keras	Spesifikasi
1	*Prosesor*	*Core 2 duo, Processo*r 2,00 Ghz
2	*Memory*	2 GB
3	*VGA Card*	512 MB (*standart*)
4	*Harddisk*	320 GB
5	*Monitor*	*Monitor* 14”, Resolusi 1024 x 768
6	*Mouse*	Standart
No	**Pernagakat Keras**	**Spesifikasi**
7	Printer	Standart
8	Koneksi	Kecepatan diatas 1 *Mbp/s*

4.1.1 Perangkat Lunak yang digunakan

Spesifikasi perangkat lunak yang digunakan dalam pembangunan Manajemen Proyek di PT. Dua Ribu Satu Pangripta dapat lihat pada table dibawah ini :

Tabel 4. 2 Perangkat Lunak Yang Digunakan

No	Perangkat Lunak	Spesifikasi
1	Sistem Operasi	Microsoft Windows 7
2	*Web Browser*	Google Chrome
3	Bahasa Pemrograman	PHP
4	DBMS	MySQL
5	*Code Editor*	Sublime Text 3
6	*UML Modeler*	Microsoft Visio 2010
7	*Web Server*	XAMPP 3.2.1

4.1.2 Implementasi Basis Data

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

1. Tabel User

Tabel admin digunakan untuk menyimpan data admin. Struktur tabel users dapat dilihat sebagai berikut.

Tabel 4. 3 Implementasi Tabel Users

Nama Tabel	Perintah
user	CREATE TABLE IF NOT EXISTS `user` (`id` int(11) NOT NULL AUTO_INCREMENT, `username` varchar(30) NOT NULL, `password` varchar(50) NOT NULL, `role` enum NOT NULL, PRIMARY KEY (`id_user`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;

2. Tabel Proyek

Tabel proyek digunakan untuk menyimpan data proyek yang digunakan oleh manajer proyek. Struktur tabel proyek dapat dilihat sebagai berikut.

Tabel 4. 4 Implementasi Tabel Proyek

Nama Tabel	Perintah
Proyek	CREATE TABLE IF NOT EXISTS `proyek` (`id_proyek` varchar(11) NOT NULL, `kode_proyek` varchar(50) NOT NULL, `id_klien` varchar(11) NOT NULL, `nama_proyek` varchar(100) NOT NULL, `biaya` float NOT NULL, `status` varchar(50) NOT NULL, `tanggal` date NOT NULL, PRIMARY KEY (`id_proyek`)) ENGINE=InnoDB DEFAULT CHARSET=latin1;

3. Tabel Klien

Berikut adalah Struktur tabel klien dapat dilihat seperti table dibawah ini.

Tabel 4. 5 Implementasi Tabel Klien

Nama Tabel	Perintah
Klien	CREATE TABLE IF NOT EXISTS `klien` (`id_klien` int(11) NOT NULL AUTO_INCREMENT, `id_user` int(11) NOT NULL, `nama_klien` varchar(100) NOT NULL, `telepon` varchar(20) NOT NULL, `instansi` varchar(50) NOT NULL, PRIMARY KEY (`id_klien`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=2 ;

4. Tabel Jadwal

Berikut adalah Struktur tabel jadwal dapat dilihat seperti table dibawah ini.

Tabel 4. 6 Implementasi Tabel Jadwal

Nama Tabel	Perintah
jadwal	CREATE TABLE IF NOT EXISTS `jadwal` (`id_jadwal` int(11) NOT NULL AUTO_INCREMENT, `kode_proyek` varchar(50) NOT NULL, `bobot` int(11) NOT NULL, `nama_kegiatan` varchar(50) NOT NULL, `durasi` int(11) NOT NULL, `mulai` date NOT NULL, `selesai` date NOT NULL, `id_user` int(11) NOT NULL, PRIMARY KEY (`id_jadwal`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=3 ;

5. Tabel CPM

Berikut adalah Struktur tabel CPM dapat dilihat seperti table dibawah ini.

Tabel 4. 7 Implementasi Tabel CPM

Nama Tabel	Perintah
CPM	CREATE TABLE IF NOT EXISTS `cpm` (`id_cpm` int(11) NOT NULL AUTO_INCREMENT, `kode_proyek` varchar(50) NOT NULL, `durasi_maju` varchar(10) NOT NULL, `durasi_mundur` varchar(10) NOT NULL, `es` int(11) NOT NULL, `ef` int(11) NOT NULL, `ls` int(11) NOT NULL, `lf` int(11) NOT NULL, `t` int(11) NOT NULL, `tf` int(11) NOT NULL) ENGINE=InnoDB DEFAULT CHARSET=latin1;

6. Tabel EVM

Berikut adalah Struktur tabel EVM dapat dilihat seperti table dibawah ini.

Tabel 4. 8 Implementasi Tabel EVM

Nama Tabel	Perintah
EVM	CREATE TABLE IF NOT EXISTS `evm` (`id_evm` int(11) NOT NULL AUTO_INCREMENT, `id_proyek` int(11) NOT NULL `od` double NOT NULL, `ac` double NOT NULL, `bobot_rencana` double NOT NULL, `bobot_aktual` double NOT NULL, `bac` double NOT NULL, `mingguke` int(11) NOT NULL, `pv` double NOT NULL, `ev` double NOT NULL, `cv` double NOT NULL, `sv` double NOT NULL, `cpi` double NOT NULL, `spi` double NOT NULL, `eac` double NOT NULL, `etc` double NOT NULL) ENGINE=InnoDB DEFAULT CHARSET=latin1;

7. Tabel Identifikasi

Berikut adalah Struktur tabel identifikasi dapat dilihat seperti table dibawah ini.

Tabel 4. 9 Implementasi Tabel Identifikasi

Nama Tabel	Perintah
Identifikasi	CREATE TABLE IF NOT EXISTS `identifikasi` (`id_identifikasi` int(11) NOT NULL AUTO_INCREMENT, `kode_proyek` varchar(50) NOT NULL `kode_resiko` varchar(50) NOT NULL, `jenis_resiko` varchar(50) NOT NULL, `deskripsi` text NOT NULL,

	PRIMARY KEY (`id_identifikasi`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=4 ;

8. Tabel Penanganan

Berikut adalah Struktur tabel penanganan dapat dilihat seperti table dibawah ini.

Tabel 4. 10 Implementasi Tabel Penanganan

Nama Tabel	Perintah
Penanganan	CREATE TABLE IF NOT EXISTS `penanganan` (`id_penanganan` int(11) NOT NULL AUTO_INCREMENT, `kode_resiko` varchar(10) NOT NULL, `penanganan` text NOT NULL, PRIMARY KEY (`id_penanganan`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=4 ;

9. Tingkat Kepentingan

Berikut adalah Struktur tabel kepentingan dapat dilihat seperti table dibawah ini.

Tabel 4. 11 Implementasi Tabel Tingkat Kepentingan

Nama Tabel	Perintah
Tingkat_Kepentingan	CREATE TABLE IF NOT EXISTS `tingkat_kepentingan` (`id_kepentingan` int(11) NOT NULL AUTO_INCREMENT, `id_identifikasi` int(11) NOT NULL, `probabilitas` double NOT NULL, `dampak` double NOT NULL, `tingkep` double NOT NULL,

	`tingkat_resiko` varchar(50) NOT NULL PRIMARY KEY (`id_kepentingan`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=4 ;

10. Laporan

Berikut adalah Struktur tabel laporan dapat dilihat seperti table dibawah ini.

Tabel 4. 12 Implementasi Tabel Laporan

Nama Tabel	Perintah
Laporan	CREATE TABLE IF NOT EXISTS `laporan` (`id_laporan` int(11) NOT NULL AUTO_INCREMENT, `id_proyek` int(11) NOT NULL, `catatan` text NOT NULL, PRIMARY KEY (`id_laporan`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=5 ;

4.1.3 Implementasi Antarmuka

Implementasi antarmuka dilakukan dengan setiap tampilan program yang dibangunan dan pengkodeannya dalam bentuk *file* program. Adapun untuk tampilan implementasi sistem manajemen proyek di PT. Dua Ribu Satu Pangripta akan dilampirkan pada halaman lampiran. Berikut ini adalah implementasi antarmuka sistem manajemen proyek di PT. Dua Ribu Satu Pangripta.

Tabel 4. 13 Implementasi Antarmuka Admin

No	Menu	Deskripsi	Nama File
1	Halaman *Login*	Digunakan sebagai halaman *login* bagi pengguna system	Login.php
2	Halaman User	Digunakan sebagai halaman pengolahan users	user.php

Implementasi antarmuka untuk manajer dapat dilihat sebagai berikut.

Tabel 4. 14 Implementasi Antarmuka Manajer

No	Menu	Deskripsi	Nama File
1	Halaman *Login*	Digunakan sebagai halaman *login* bagi pengguna system	Login.php
2	Halaman Proyek	Digunakan sebagai halaman pengolahan proyek	proyek.php
4	Halaman Jadwal	Digunakan sebagai halaman melihat jadwal, kegiatan, dan hasil analisa menggunakan metode CPM	jadwal.php
5	Halaman EVM	Digunakan sebagai halaman melihat estimasi jadwal dan biaya suatu proyek berdasarkan analisa menggunakan metode EVM	evm.php
6	Halaman Tingkat Kepentingan	Digunakan sebagai halaman pengolahan resiko berdasarkan identifikasi dan level resiko yang mungkin akan terjadi pada suatu proyek	tingkat_ kepentingan.php
11	Halaman Penanganan	Digunakan sebagai halaman penanganan berdasarkan identifikasi dan level resiko suatu proyek	penanganan.php
12	Halaman Progress Proyek	Digunakan sebagai halaman dalam *progress* pekerjaan	progress_ kerja.php
13	Halaman Identifikasi Resiko	Digunakan untuk mengidentifikasi resiko	Identifikasi.php
14	Halaman Struktur Kerja	Digunakan untuk melihat pekerjaan yang dikerjakan	jadwal_kerja.php

Implementasi antarmuka untuk tenaga ahli dapat dilihat sebagai berikut.

Tabel 4. 15 Implementasi Antarmuka Tenaga Ahli

No	Menu	Deskripsi	Nama File
1	Halaman Login	Digunakan sebagai halaman *login* bagi pengguna system	Login.php
2	Halaman Utama	Digunakan sebagai halaman utama pada Manajer	Home.php
4	Halaman Struktur Kerja	Digunakan sebagai halaman untuk pengolahan struktur kerja proyek	jadwal_kerja.php
6	Halaman Pencapaian Proyek	Digunakan sebagai halaman dalam *progress* pekerjaan	progress_ kerja.php

4.2 Pengujian Sistem

Pengujian sistem merupakan hal terpenting yang bertujuan untuk menemukan kesalahan dan kekurangan pada perangkat lunak yang diuji. Pengujian bermaksud untuk mengetahui perangkat lunak yang dibuat sudah memenuhi kriteria yang sesuai dengan tujuan perancangan. Pengujian perangkat lunak ini menggunakan pengujian *black box*. Pengujian *black box* berfokus pada persyaratan fungsional perangkat lunak.

4.2.1 Skenario Pengujian Black Box

Berdasarkan pengujian yang telah dilakukan, dapat diambil kesimpulan bahwa sistem manajemen proyek di PT. Dua Ribu Satu Pangripta telah melalui tahap perbaikan dan sudah dimaksimalkan terhadap proses-proses tersebut.Skenario pengujian *black box* dapat dilihat dibawah ini.

Tabel 4. 16 Skenario Pengujian *Black Box*

Fungsional Uji	Butir Uji	Jenis Pengujian
Login	*Login* user	*Black Box*
Pengelolaan User	Menambah data user	*Black Box*
	Mengubah data user	
	Menghapus data user	
Pengelolaan Identifikasi Resiko	Menambah data identifikasi resiko	*Black Box*
	Mengubah data identifikasi resiko	
	Menghapus data identifikasi resiko	
Pengelolaan jadwal kerja	Menambah data pekerjaan	*Black Box*
	Mengubah data pekerjaan	
	Menghapus data pekerjaan	
Pengelolaan Klien	Menambah data klien	*Black Box*
	Mengubah data klien	
	Menghapus data klien	
Pengelolaan Proyek	Menambah data proyek	*Black Box*
	Mengubah data proyek	
	Menghapus data proyek	
Pengelolaan Struktur Kerja	Menambah data struktur kerja	*Black Box*
	Mengubah data struktur kerja	
	Menghapus data struktur kerja	
Pengelolaan Tingkat Kepentingan	Menambah data tingkat kepentingan	*Black Box*
	Mengubah data tingkat kepentingan	
	Menghapus data tingkat kepentingan	
Pengelolaan	Menambah data penanganan	*Black Box*

Penanganan	Mengubah data penanganan	
	Menghapus data penanganan	

4.2.2 Kasus dan Hasil Pengujian

Berdasarkan rencana pengujian yang disusun, maka dilakukan pengujian sesuai dengan kelas uji yang telah dijelaskan sebelumnya.

1. Pengujian Login

Tabel 4. 17 Pengujian Login Data Benar

Kasus dan Hasil Uji (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memasukan *username* dan *password* dengan benar, contoh masukan, Username : admin Password : admin	*Field* data *username* dan *password* sudah terisi lalu menuju ke halaman utama sesuai hak akses pengguna	Menampilkan halaman utama sesuai hak akses pengguna	[√] Diterima [ ] Ditolak

Tabel 4. 18 Pengujian Login Data Benar

Kasus dan Hasil Uji (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memasukan data *username* dan *password* yang tidak terdaftar atau salah, contoh masukan: *Username* : manajer *Password* : admin	Menampilkan pesan "*Username* atau *password* yang anda masukkan salah"	Menampilkan pesan "*Username* atau *password* yang anda masukkan salah"	[√] Diterima [ ] Ditolak

2. Pengujian Pengelolaan User

Tabel 4. 19 Pengujian Pengelolaan User Data Benar

Kasus dan Hasil Uji (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Contoh masukan tambah data, Username : tenagaahli Nama Lengkap : dodi wijaya Password : tenagaahli	Sistem menyimpan data dan menampilkan pesan "Sukses! Data user telah disimpan"	Sistem menyimpan data dan menampilkan pesan "Sukses! Data user telah disimpan"	[√] Diterima [ ] Ditolak
Contoh masukan yang akan diubah, Username : habibi Password lama : 123456 Password Baru : 654321 Konfirmasi Password Baru : 654321	Sistem mengubah data pada user yang dipilih dan menampilkan pesan "Sukses! Data user telah diubah"	Sistem mengubah data pada user yang dipilih dan menampilkan pesan "Sukses! Data user telah diubah"	[√] Diterima [ ] Ditolak
Menghapus data user	Menampilkan pesan "Yakin menghapus data : Anca Purnama?", dan menampilkan pesan "Sukses! Data telah dihapus" setelah pengguna memilih "ok"	Menampilkan pesan "Yakin menghapus data : Anca Purnama?", dan menampilkan pesan "Sukses! Data telah dihapus" setelah pengguna memilih "ok"	[√] Diterima [ ] Ditolak

Tabel 4. 20 Pengujian Pengelolaan User Data Salah

Kasus dan Hasil Uji (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Menghapus data user (*cancel*)	Menampilkan pesan konfirmasi "Yakin menghapus data?	Menampilkan pesan konfirmasi "Yakin menghapus data ?	[√] Diterima [ ] Ditolak

3. Pengujian Identifikasi Resiko

Tabel 4. 21 Pengujian Pengelolaan Identifikasi Resiko Data Benar

Kasus dan Hasil Uji (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Contoh masukan tambah data, Kode Resiko : R1 Jenis Resiko : Penggambaran Deskripsi : Salah spesifikasi	Sistem menyimpan data dan menampilkan pesan "Sukses! Data identifikasi telah disimpan"	Sistem menyimpan data dan menampilkan pesan "Sukses! Data identifikasi telah disimpan"	[√] Diterima [ ] Ditolak
Contoh masukan yang akan diubah, Kode Resiko : R1 Jenis Resiko : Keuangan Deskripsi : Salah spesifikasi	Sistem mengubah data pada identifikasi yang dipilih dan menampilkan pesan "Sukses! Data identifikasi telah diubah"	Sistem mengubah data pada identifikasi yang dipilih dan menampilkan pesan "Sukses! Data identifikasi telah diubah"	[√] Diterima [ ] Ditolak
Menghapus data identifikasi	Menampilkan pesan "Yakin menghapus resiko : Salah spesifikasi?", dan menampilkan pesan "Sukses! Data telah dihapus" setelah pengguna memilih "ok"	Menampilkan pesan "Yakin menghapus resiko : Salah spesifikasi?", dan menampilkan pesan "Sukses! Data telah dihapus" setelah pengguna memilih "ok"	[√] Diterima [ ] Ditolak

Tabel 4. 22 Pengujian Pengelolaan Identifikasi Resiko Data Salah

Kasus dan Hasil Uji (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Data pada *form* ubah identifikasi kosong, contoh masukan, Kode Resiko : Jenis Resiko : Deskripsi :	Menampilkan pesan "*Form* tidak boleh ada yang kosong"	Menampilkan pesan "*Form* tidak boleh ada yang kosong"	[√] Diterima [ ] Ditolak
Menghapus data identifikasi (*cancel*)	Menampilkan pesan konfirmasi "Yakin menghapus resiko : Salah spesifikasi?" Jika tekan *cancel*, maka data tidak akan terhapus dan tetap di halaman	Menampilkan pesan konfirmasi "Yakin menghapus resiko : Salah spesifikasi?" Jika tekan *cancel*, maka data tidak akan terhapus dan tetap di halaman	[√] Diterima [ ] Ditolak

	master identifikasi	master identifikasi	

4. Pengujian Pengelolaan Jadwal Kerja

Tabel 4. 23 Pengujian Pengelolaan Jadwal Kerja Data Benar

Kasus dan Hasil Uji (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Contoh masukan tambah data, Pekerjaan : Persiapan	Sistem menyimpan data dan menampilkan pesan "Sukses! Data pekerjaan telah disimpan"	Sistem menyimpan data dan menampilkan pesan "Sukses! Data pekerjaan telah disimpan"	[√] Diterima [ ] Ditolak
Contoh masukan yang akan diubah, Pekerjaan : Penggambaran	Sistem mengubah data pada pekerjaan yang dipilih dan menampilkan pesan "Sukses! Data pekerjaan telah diubah"	Sistem mengubah data pada pekerjaan yang dipilih dan menampilkan pesan "Sukses! Data pekerjaan telah diubah"	[√] Diterima [ ] Ditolak
Menghapus data pekerjaan	Menampilkan pesan "Yakin menghapus kegiatan : Persiapan?", dan menampilkan pesan "Sukses! Data telah dihapus" setelah pengguna memilih "ok"	Menampilkan pesan "Yakin menghapus kegiatan : Persiapan?", dan menampilkan pesan "Sukses! Data telah dihapus" setelah pengguna memilih "ok"	[√] Diterima [ ] Ditolak

Tabel 4. 24 Pengujian Pengelolaan Jadwal Kerja Data Salah

Kasus dan Hasil Uji (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Data pada *form* ubah pekerjaan kosong, contoh masukan, Pekerjaan :	Menampilkan pesan "*Form* tidak boleh ada yang kosong"	Menampilkan pesan "*Form* tidak boleh ada yang kosong"	[√] Diterima [ ] Ditolak
Menghapus data pekerjaan	Menampilkan pesan konfirmasi	Menampilkan pesan konfirmasi	[√] Diterima [ ] Ditolak

(*cancel*)	"Yakin menghapus kegiatan : Persiapan?" Jika tekan *cancel,* maka data tidak akan terhapus dan tetap di halaman master pekerjaan	"Yakin menghapus kegiatan : Persiapan?" Jika tekan *cancel,* maka data tidak akan terhapus dan tetap di halaman master pekerjaan	

5. Pengujian Pengelolaan Klien

Tabel 4. 25 Pengujian Pengelolaan Klien Data Benar

Kasus dan Hasil Uji (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Contoh masukan tambah data, Nama Lengkap : Widya L Telepon : 08126661902	Sistem menyimpan data dan menampilkan pesan "Sukses! Data klien telah disimpan"	Sistem menyimpan data dan menampilkan pesan "Sukses! Data klien telah disimpan"	[√] Diterima [ ] Ditolak
Contoh masukan yang akan diubah, Nama Lengkap : Widya L Telepon : 08536561979	Sistem mengubah data pada klien yang dipilih dan menampilkan pesan "Sukses! Data klien telah diubah"	Sistem mengubah data pada klien yang dipilih dan menampilkan pesan "Sukses! Data klien telah diubah"	[√] Diterima [ ] Ditolak
Menghapus data klien	Menampilkan pesan "Yakin menghapus data klien?", dan menampilkan pesan "Sukses! Data telah dihapus" setelah pengguna memilih "ok"	Menampilkan pesan "Yakin menghapus data klien?", dan menampilkan pesan "Sukses! Data telah dihapus" setelah pengguna memilih "ok"	[√] Diterima [ ] Ditolak

Tabel 4. 26 Pengujian Pengelolaan Klien Data Salah

Kasus dan Hasil Uji (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Menghapus data klien (*cancel*)	Menampilkan pesan konfirmasi "Yakin menghapus data klien?" Jika tekan *cancel*, maka data tidak akan terhapus dan tetap di halaman klien	Menampilkan pesan konfirmasi "Yakin menghapus data klien?" Jika tekan *cancel*, maka data tidak akan terhapus dan tetap di halaman klien	[√] Diterima [ ] Ditolak

6. Pengujian Pengelolaan Proyek

Tabel 4. 27 Pengujian Pengelolaan Proyek Data Benar

Kasus dan Hasil Uji (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Contoh masukan tambah data, Kode Proyek : 07082014 Klien : andrian Nama Proyek : ruko Biaya : 7500000 Status Proyek : Proses	Sistem menyimpan data dan menampilkan pesan "Sukses! Data proyek telah disimpan"	Sistem menyimpan data dan menampilkan pesan "Sukses! Data proyek telah disimpan"	[√] Diterima [ ] Ditolak
Contoh masukan tambah data, Kode Proyek : 07082014 Klien : andrian Nama Proyek : ruko Biaya : 7500000 Status Proyek : selesai	Sistem mengubah data pada proyek yang dipilih dan menampilkan pesan "Sukses! Data proyek telah diubah"	Sistem mengubah data pada proyek yang dipilih dan menampilkan pesan "Sukses! Data proyek telah diubah"	[√] Diterima [ ] Ditolak
Menghapus data proyek	Menampilkan pesan "Yakin menghapus proyek?", dan menampilkan	Menampilkan pesan "Yakin menghapus data proyek?", dan menampilkan	[√] Diterima [ ] Ditolak

	pesan “Sukses! Data telah dihapus” setelah pengguna memilih “ok”	pesan “Sukses! Data telah dihapus” setelah pengguna memilih “ok”	

Tabel 4. 28 Pengujian Pengelolaan Proyek Data Salah

Kasus dan Hasil Uji (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Menghapus data proyek (*cancel*)	Menampilkan pesan konfirmasi “Yakin menghapus data proyek?” Jika tekan *cancel,* maka data tidak akan terhapus dan tetap di halaman proyek	Menampilkan pesan konfirmasi “Yakin menghapus data proyek?” Jika tekan *cancel,* maka data tidak akan terhapus dan tetap di halaman proyek	[√] Diterima [ ] Ditolak

4.2.3 Kesimpulan Hasil Pengujian Black Box

Berdasarkan hasil pengujian *black box* yang dilakukan dengan kasus uji tersebut, dapat ditarik kesimpulan sebagai berikut:

1. Jika data benar maka sistem akan masuk sesuai dengan apa yang telah di inputkan.
2. Jika data salah maka system akan mengeluarkan pesan kesalahan dan meminta ulang untuk menginputkan data yang benar.

4.3 Pengujian Beta

Pengujian beta merupakan pengujian yang dilakukan secara objektif dimana diuji secara langsung ke lapangan yaitu instansi yang bersangkutan mengenai kepuasan pengguna dengan kandungan poin Pengujian beta dilakukan melalui sebuah teknik pengambilan data, yaitu melalui wawancara.

4.3.1 Wawancara Pengujian Beta Manajer Proyek

Wawancara dilakukan untuk pengujian beta terhadap sistem manajemen proyek kepada Bapak Andrian selaku bagian *Project Manager* di perusahaan.

1. Menurut bapak apakah aplikasi ini mudah digunakan ?	
Jawaban Narasumber	Aplikasi ini cukup mudah untuk digunakan karena tampilannya cukup sederhana.
2. Menurut bapak apakah fitur yang ada sudah mencukupi untuk sistem informasi manajemen proyek?	
Jawaban Narasumber	Menurut saya sudah cukup untuk saat ini, walaupun harus ada pengembangan nantinya.
3. Apakah sistem informasi manajemen proyek ini bisa membantu bapak dalam identifikasi dan pengendalian resiko proyek?	
Jawaban Narasumber	Saya harus mencoba terlebih dahulu ke proyek yang lain yang akan dikerjakan.
4. Apakah sistem informasi manajemen proyek ini bisa membantu bapak dalam pengendalian pelaksanaan setiap proyek agar sesuai dengan jadwal ?	
Jawaban Narasumber	Dengan ada nya informasi yang menampilkan data biaya dan waktu itu sangat membantu dalam hal pengawasaan proyek

4.3.2 Kesimpulan Hasil Pengujian Beta

Berdasarkan hasil jawaban dari bapak andrian terhadap pertanyaan yang diajukan pada pengujian beta, maka dapat ditarik kesimpulan bahwa sistem yang dibangun mudah digunakan oleh pengguna, fitur yang ada sudah mencukupi kebutuhan, adanya fitur diagram jaringan kerja (CPM), pengendalian biaya dan waktu (EVM) dan Manjemen Resiko PT. Dua Ribu Satu Pangripta cukup tertarik dengan aplikasi ini. Namun ada beberapa hal yang perlu diperbaiki dalam penyajian informasinya.