BAB III
PEMBAHASAN

3.1 Analisis Sistem

Tahap pertama yang dilakukan adalah menganalisis sistem. Sistem yang penulis analisis adalah sistem yang sedang berjalan (dalam hal ini sistem yang sudah ada). Analisis sistem dilakukan dengan cara memecah bagian-bagian yang ada pada sistem kemudian diamati. Berikut ini akan dijelaskan mengenai analisis sistem yang dilakukan terhadap manajemen proyek menggunakan PARKIT yang meliputi analisis masalah, analisis fungsional, analisis nonfungsional, analisis kebutuan nonfungsional, dan analisis kebutuhan fungsional.

3.1.1 Analisis Masalah

DCISTEM di Universitas Padjajaran merupakan salah satu unit pelaksana teknis yang bergerak dalam bidang pengeloaan dan pusat data dan teknologi informasi, tidak hanya itu DCISTEM juga merupakan unit pelaksana teknis yang memiliki tugas sebagai pengembang teknologi untuk Universitas Padjajaran. Namun seperti yang telah penulis bahas pada bab satu, DCISTEM membutuhkan manajemen proyek dalam menjalankan tugas-tugasnya sebagai unit pelaksana teknis. Dalam hal ini DCISTEM sendiri telah memiliki perangkat lunak sebagai lembar kerja manajemen proyek yang berbasis web bernama PARKIT, namun penggunaan PARKIT di DCISTEM belum optimal dikarenakan belum adanya sumber daya manusia yang secara khusus menangani manajemen proyek dan mengoperasikan PARKIT. Sehingga dari analisis masalah yang dilakukan, permasalahan yang dihadapi oleh DCISTEM dalam manajemen proyek menggunakan PARKIT adalah :

1. Manajemen proyek DCISTEM – UNPAD yang belum terstruktur.
2. Pengguaan PARKIT yang belum optimal.

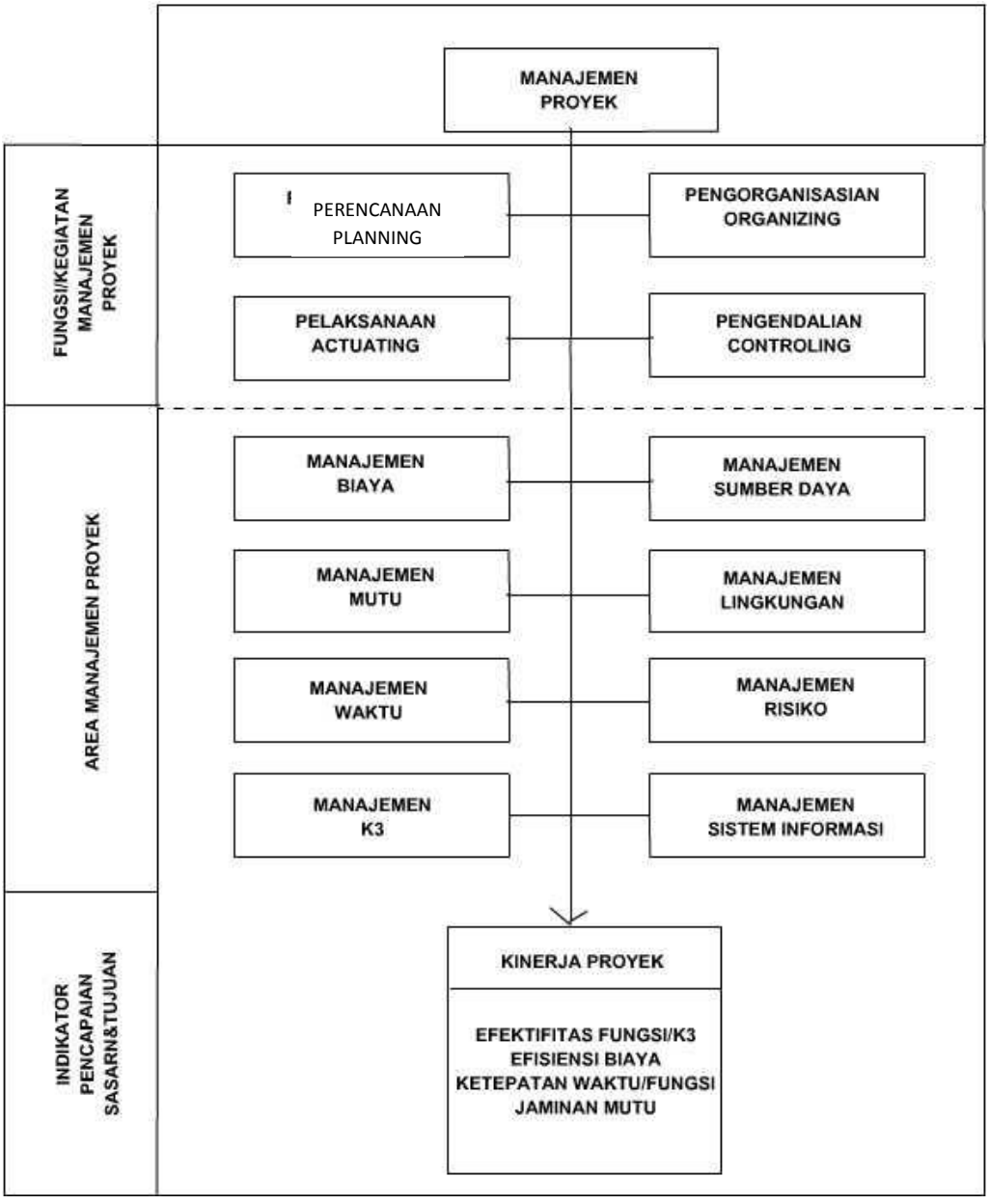
3.1.2 Analisis Fungsional

Sistem yang sedang berjalan di DCISTEM-UNPAD memiliki batasan prosedur. Berdasarkan hasil analisis, prosedur yang harus dijalani dalam

memperbaharui manajemen proyek yang ada di DCISTEM adalah sebagai berikut :

1. Memulai dari perencanaan suatu proyek yang akan dibangun.
2. Melakukan pengorganisasian yang pasti.
3. Melaksanakan proyek yang telah direncanakan dan diorganisasikan.
4. Melakukan pengendalian dalam segala aspek manajemen proyek tersebut.

Untuk menggambarkan aliran manajemen proyek pada prosedur sistem yang sedang berjalan dapat dilihat pada gambar di bawah ini :



Gambar 3.1 Proses Manajemen Proyek dan Pencapaian Kinerjanya.

3.1.3 Analisis Non-Fungsional

Setelah melakukan analisis fungsional untuk mengetahui prosedur manajemen proyek yang berjalan di DCISTEM-UNPAD, maka langkah selanjutnya yaitu melakukan analisis non-fungsional. Analisis non-fungsional dilakukan untuk mengetahui siapa saja yang menggunakan sistem dan untuk mengetahui perangkat keras ataupun perangkat lunak yang digunakan pada sistem tersebut. Analisis non-fungsional yang dilakukan dibagi dalam tiga tahap, yaitu :

1. Analisis Pengguna
2. Analisis Perangkat Keras
3. Analisis Perangkat Lunak

3.1.3.1 Analisis Pengguna

Perangkat keras dan perangkat lunak yang ada tidak akan berguna apabila tidak ada pengguna yang mengoperasikannya. Hak akses pengguna PARKIT di DCISTEM dibagi menjadi tiga bagian, yaitu :

1. Super Admin

   Pengguna ini memiliki karakteristik yang sangat khusus, karena tidak semua orang dapat menjadi super admin dalam sistem PARKIT. Pengguna ini memiliki hak akses untuk memperbarui seluruh bagian PARKIT. Pengguna ini dituntut untuk mampu menguasai HTML, PHP, maupun bahasa pemrograman lain yang mendukung pembaruan PARKIT.
2. Administrator

   Pengguna ini bertugas untuk memperbarui informasi mengenani proyek, task, maupaun *subtask* yang ada pada PARKIT dimana proyek ini akan di tag kepada karyawan atau pegawai yang bersangkutan. Pengguna ini dituntut untuk selalu mengecek dan meng*update* perkembangan kinerja proyek.
3. Karyawan/Pegawai

   Pengguna ini memiliki karakteristik umum, karena siapapun yang terlibat dalam proyek DCISTEM dapat menjadi pengguna ini. Pengguna ini bertugas untuk selalu melakukan laporan *progress* kinerja proyek yang telah diselesaikan pada halaman *task* dan *subtask.*

3.1.3.2 Analisis Perangkat Keras

Dalam membangun PARKIT, DCISTEM menggunakan perangkat keras dengan spesifikasi sebagai berikut :

Tabel 3.1 Analisis Perangkat Keras

Processor	Intel Pentium 4 dengan kecepatan 1,4 GHZ
RAM	1 GB.
VGA	512 Mb
Harddisk	80 GB
Alat Input	Mouse optical, keyboard
Monitor	LCD

3.1.3.3 Analisis Perangkat Lunak

Perangkat lunak yang digunakan DCISTEM-UNPAD untuk membangun PARKIT adalah sebagai berikut :

Tabel 3.2 Analisis Perangkat Lunak

Sistem Operasi	Microsoft *Windows* XP, Windows 7
Web Editor	Collabtive
Web browser	Mozilla Firefox, Google Crome, dll

3.1.4 Kebutuhan Non-Fungsional

Setelah dilakukan analisis non-fungsional, dapat ditentukan kebutuhan non-fungsional yang meliputi kebutuhan pengguna, perangkat keras dan perangkat lunak untuk menjalankan PARKIT di DCISTEM-UNPAD.

3.1.4.1 Kebutuhan Pengguna

Untuk menggunakan PARKIT, DCISTEM-UNPAD membutuhkan pengguna dengan karakteristik sebagai berikut :

1. Admin Umum : Administrator umum adalah admin yang bertugas untuk menambah admin dan mengawasi PARKIT.
2. Admin Rekayasa Perangkat Lunak : Administrator Rekayasa Perangkat Lunak bertugas untuk mengawasi PARKIT dan mengelola proyek PARKIT dalam ruang lingkup Divisi Rekayasa Perangkat Lunak.

3. Admin Infrastruktur : Administrator Infrastruktur bertugas untuk mengawasi PARKIT dan mengelola proyek PARKIT dalam ruang lingkup Divisi Infrastruktur.
4. Admin Layanan dan Konten : Administrator Infrastruktur bertugas untuk mengawasi PARKIT dan mengelola proyek PARKIT dalam ruang lingkup Divisi Layanan dan Konten.
5. Pegawai/karyawan : Pegawai/karyawan bertugas untuk melaporkan seluruh *progress* kinerja proyek yang telah dikerjakan pada halaman *task* dan *subtask.*

3.1.4.2 Kebutuhan Perangkat Keras

Spesifikasi perangkat keras yang dibutuhkan dalam pengoperasian PARKIT DCISTEM-UNPAD adalah :

Tabel 3.3 Kebutuhan Perangkat Keras

Processor	Intel Pentium 4 dengan kecepatan 1,4 GHZ
RAM	1 GB.
VGA	512 Mb
Harddisk	80 GB
Alat Input	Mouse optical, keyboard
Monitor	LCD

3.1.4.3 Kebutuhan Perangkat Lunak

Untuk membangun PARKIT DCISTEM-UNPAD dibutuhkan perangkat lunak sebagai berikut :

Tabel 3.4 Kebutuhan Perangkat Lunak

Sistem Operasi	Microsoft *Windows* XP, Windows 7
Bahasa Pemrograman	PHP, Collabtive
Web Editor	Collabtive
Web Server	Localhost
Web Browser	Mozilla Firefox, Google Crome, dll
Database Server	MySQL

Sedangkan spesifikasi perangkat lunak yang akan digunakan untuk mengakses PARKIT DCISTEM-UNPAD adalah :

Tabel 3.5 Kebutuhan Perangkat Lunak untuk mengakses PARKIT

Sistem Operasi	Microsoft *Windows* XP,Windows 7
Web Browser	Mozilla Firefox, Google Crome, dll

3.1.4.4 Analisis Jaringan

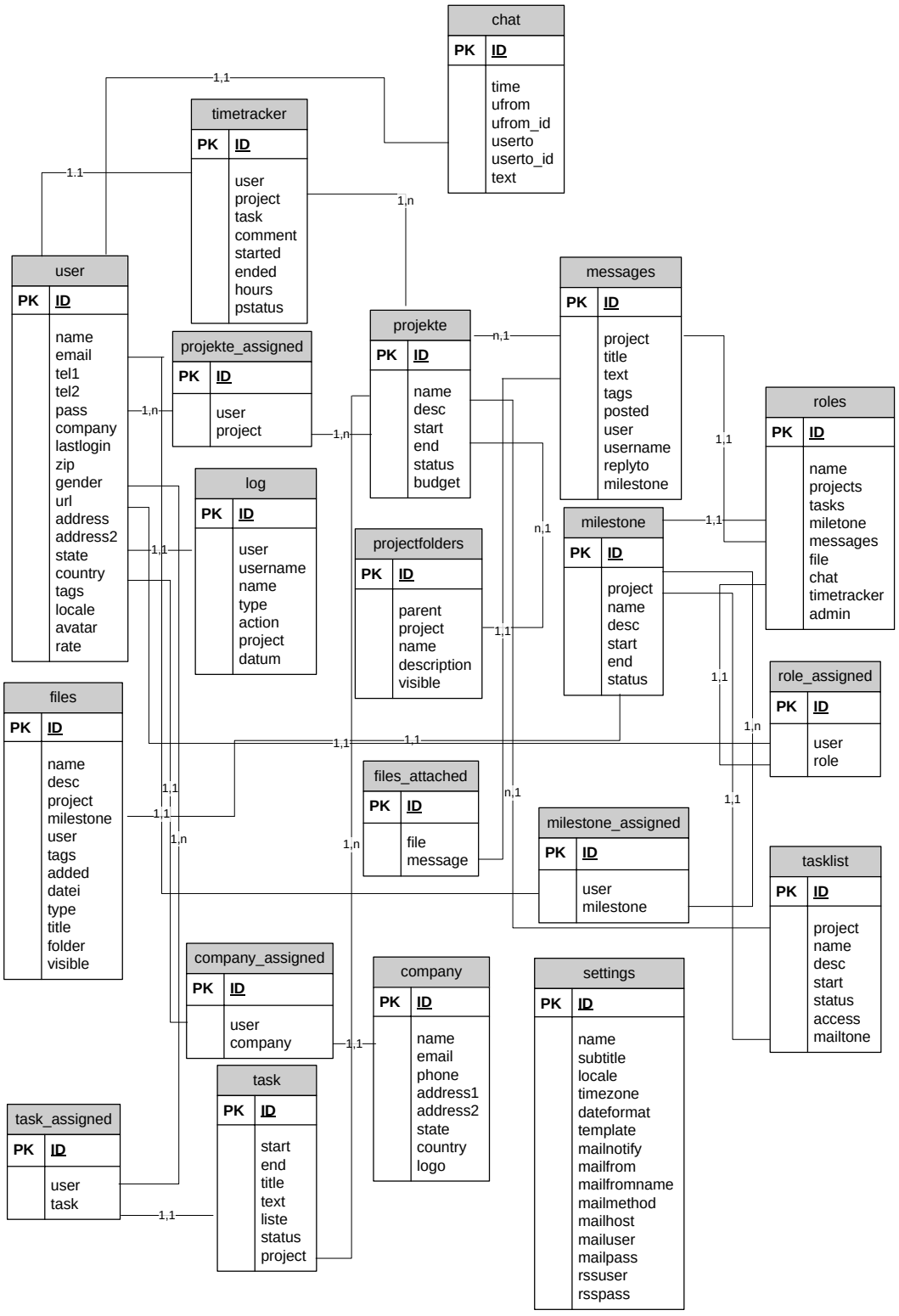
Analisis jaringan internet dan analisis denah ruangan setiap Divisi yang ada di DCISTEM-UNPAD dimaksudkan untuk mempermudah dalam menggambarkan jaringan yang akan dibangun.

Dalam menjalankan PARKIT diperlukan adanya perangkat komputer di setiap Divisi untuk terkoneksi dengan internet. Terdapat tiga Divisi yang ada di DCISTEM-UNPAD yaitu Divisi Rekayasa Perangkat Lunak, Divisi Infrastruktur, dan Divisi Layanan dan Konten.

Agar PARKIT yang dibangun dapat diakses 24 jam perhari selama 7 hari per minggu secara terus menerus, PARKIT yang dibangun harus disimpan pada webhosting dan untuk dapat mengaksesnya diperlukan ISP. Adapun *web hosting* yang digunakan oleh PARKIT adalah www.unpad.ac.id/PARKIT dan ISP yang digunakan adalah Speedy.

3.1.5 Analisis Data

Setiap sistem informasi membutuhkan data untuk menghasilkan informasi yang dibutuhkan. Data yang ada harus saling berhubungan atau berelasi antara satu dengan yang lain untuk menyajikan informasi yang benar. Untuk memodelkan relasi antar data tersebut digunakan diagram E-R (Entity-Relationship). Setelah melakukan analisis terhadap kebutuhan PARKIT, maka terbentuklah tabel-tabel yang berisi data yang berhubungan dengan dokumen dan keperluan sistem. Adapun diagram E-R dari relasi tabel-tabel adalah sebagai berikut :



Gambar 3.2 Diagram E-R PARKIT

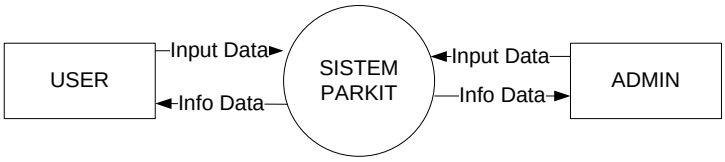
3.1.6 Analisis Kebutuhan Fungsional

Setelah melakukan analisis kebutuhan non fungsional maka dilanjutkan ke langkah berikutnya yaitu analisis kebutuhan fungsional. Langkah ini dimaksudkan untuk menganalisis PARKIT yang diusulkan. Dalam langkah ini dilakukan penentuan entitas-entitas baik entitas internal, serta prosedur-prosedur yang bisa dilakukan oleh masing-masing entitas.

Untuk mempermudah menggambarkan hasilnya, maka dibentuklah hasil analisis kebutuhan non-fungsional dalam bentuk diagram-diagram serta alat bantu berupa diagram konteks, diagram alir data, dan kamus data.

3.1.6.1 Diagram Konteks

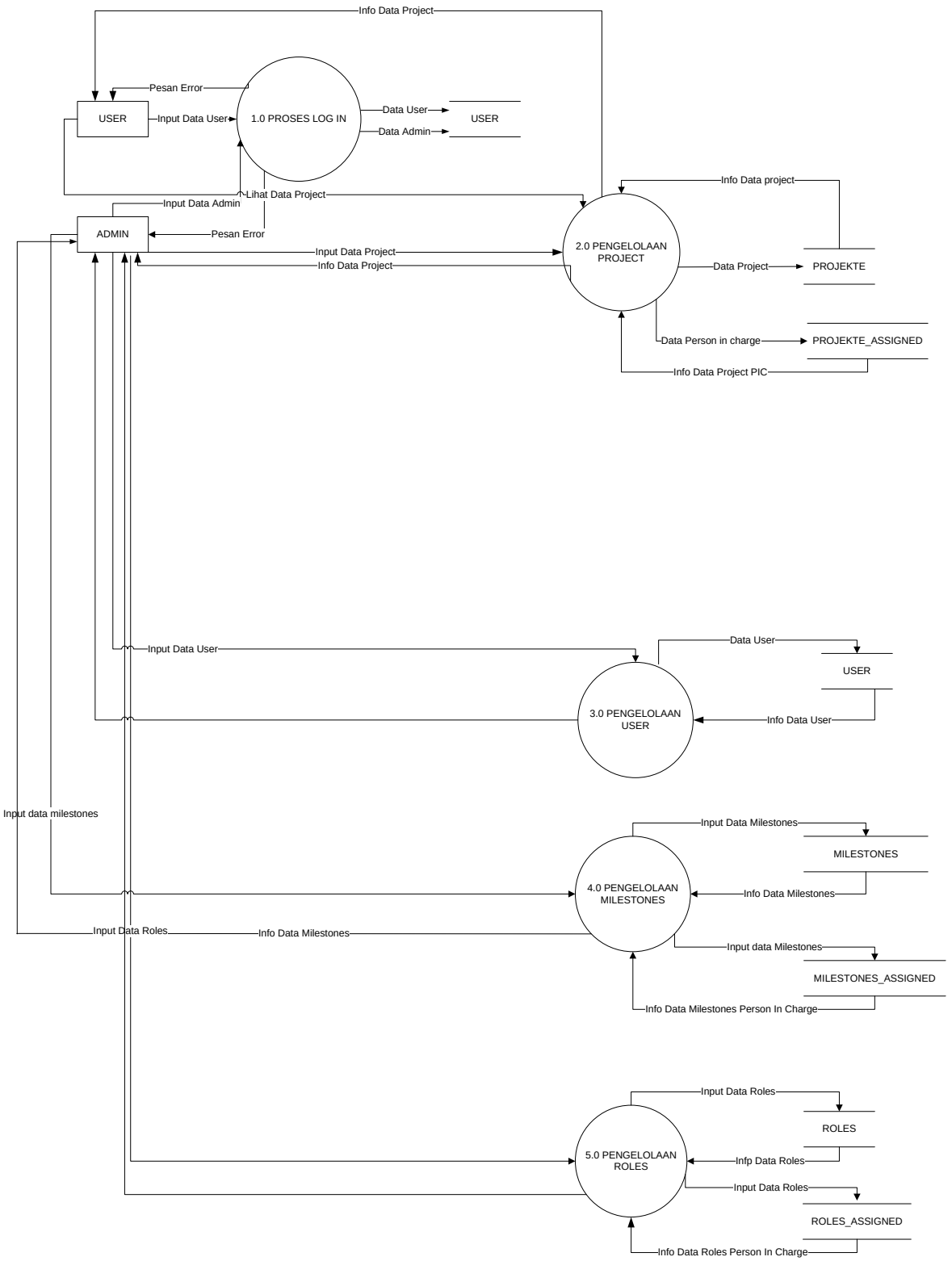
Diagram konteks merupakan salah satu alat bantu dalam melakukan analisis terstruktur. Diagram konteks ini menggambarkan suatu sistem secara garis besarnya atau keseluruhannya saja. Dalam diagram konteks juga digambarkan entitas eksternal yang merupakan perangkat pikir yang menghasilkan data yang akan diolah oleh sistem. Adapun diagram konteks yang diusulkan adalah sebagai berikut.



Gambar 3.3 Diagram Konteks

3.1.6.2 Diagram Alir Data

Diagram alir data merupakan sebuah representasi dari suatu sistem yang menggambarkan bagian-bagian dari sistem tersebut beserta keterkaitan antara bagian-bagian yang ada. Dari diagram alir data ini seseorang bisa mengetahui sumber dari informasi di dalam sistem maupun tujuan dari masukan yang berasal dari entitas. Pada PARKIT diagram alir data adalah sebagai berikut :



Gambar 3.4 Diagram Alir Data Level 1.

1.0 Proses Log In

Proses ini adalah proses untuk memasukan *username* dan *password* sebagai autentikasi ke menu admin.

2.0 Proses pengelolaan project

Proses pengelolaan *project* adalah proses dimana pengolahan data tentang *project* yang akan di buat untuk setiap Departemen.

3.0 Proses pengelolaan user

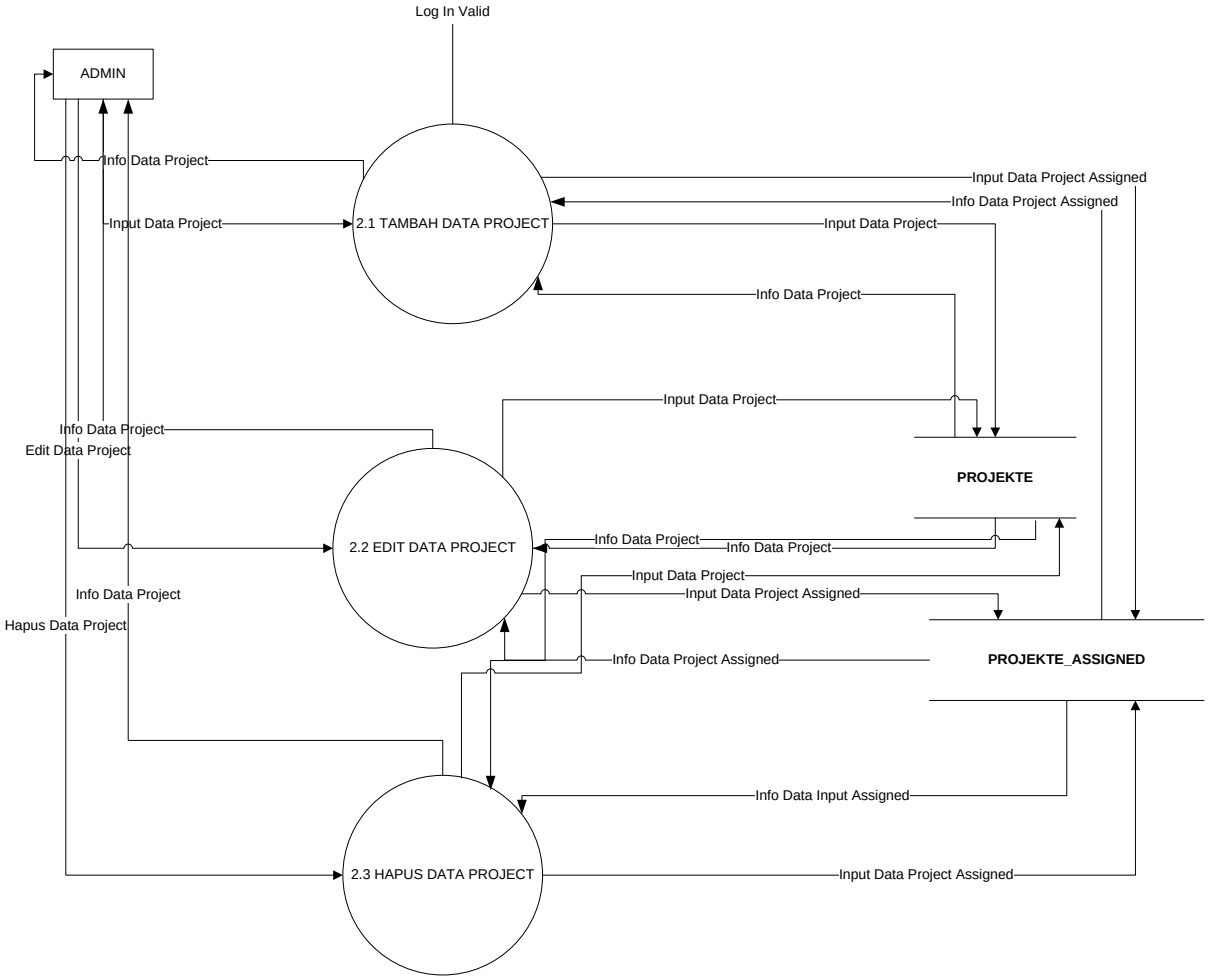
Adalah proses untuk mengolah data data *user*.

4.0 Proses milestones

adalah proses yang digunakan untuk monitoring suatu *project* apakah sudah berjalan seharusnya atau tidak.

5.0 Proses Roles

Proses untuk mengetahui apa saja masalah yang terjadi pada saat *project* berjalan atau dengan kata lain *list* beberapa masalah yang terjadi saat *project* sedang berjalan.



Gambar 3.5 DFD LEVEL 2 Proses Pengelolaan *Project*

2.1 Proses Tambah Data Project

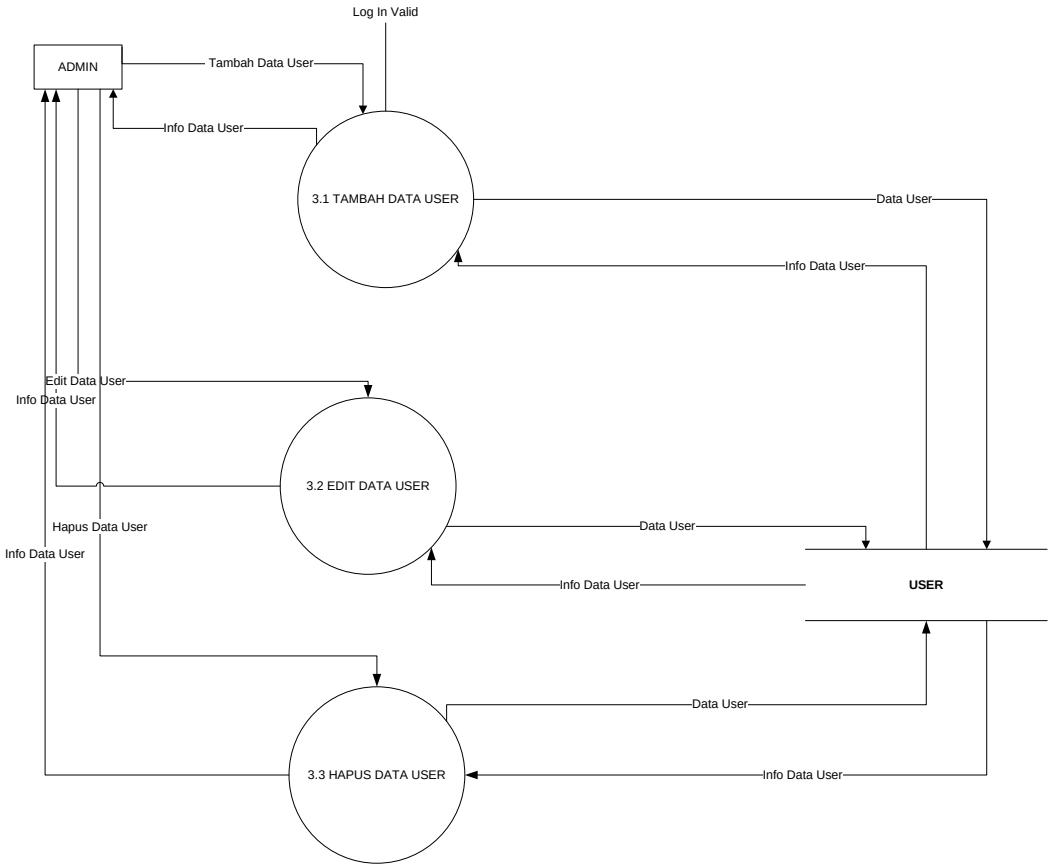
Proses transaksi penambahan data yang dilakukan oleh admin yang disimpan kedalam tabel “PROJEKTE” dan“PROJEKTE_ASSIGNED”.

2.2 Proses Edit Data Project

Proses transaksi yang dilakukan terhadap *project* yang ada didalam *database* dengan data yang sudah ada kemudian merubah data tersebut lalu menyimpan kembali kedalam *database* tersebut.

2.3 Penghapusan Data Project

Proses penghapusan data dimana admin dapat melakukan penghapusan data *project*.



Gambar 3.6 DFD LEVEL 3 Proses Pengelolaan *User*

3.1 Proses Tambah Data User

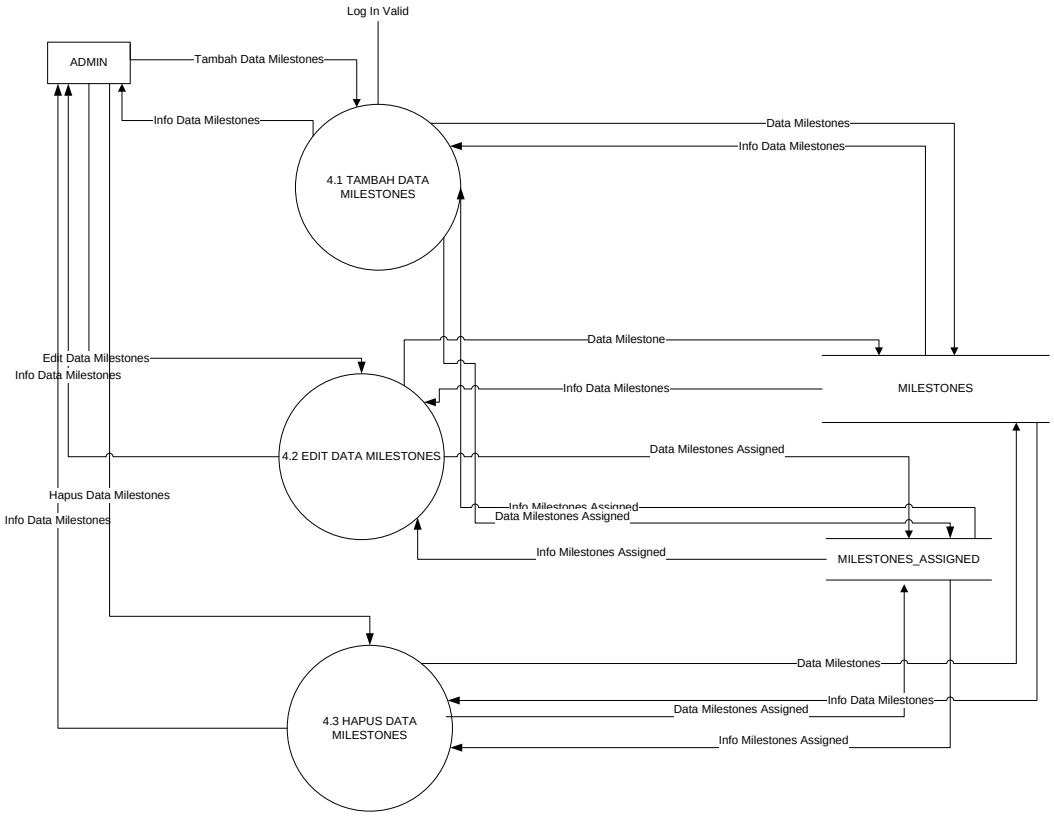
Proses dimana admin dapat memasukan *user* baru baik admin/*user* biasa yang akan menjalankan aplikasi ini.

3.2 Proses Edit Data user

Proses dimana admin dapat melakukan perubahan data baik itu *user*/admin.

3.3 Penghapusan Data User

Proses dimana admin dapat menghapus admin/*user* yang sudah tidak aktif lagi.



Gambar 3.7 DFD LEVEL 4 Proses Pengelolaan *Milestones*

4.1 Proses Tambah Data Milestones

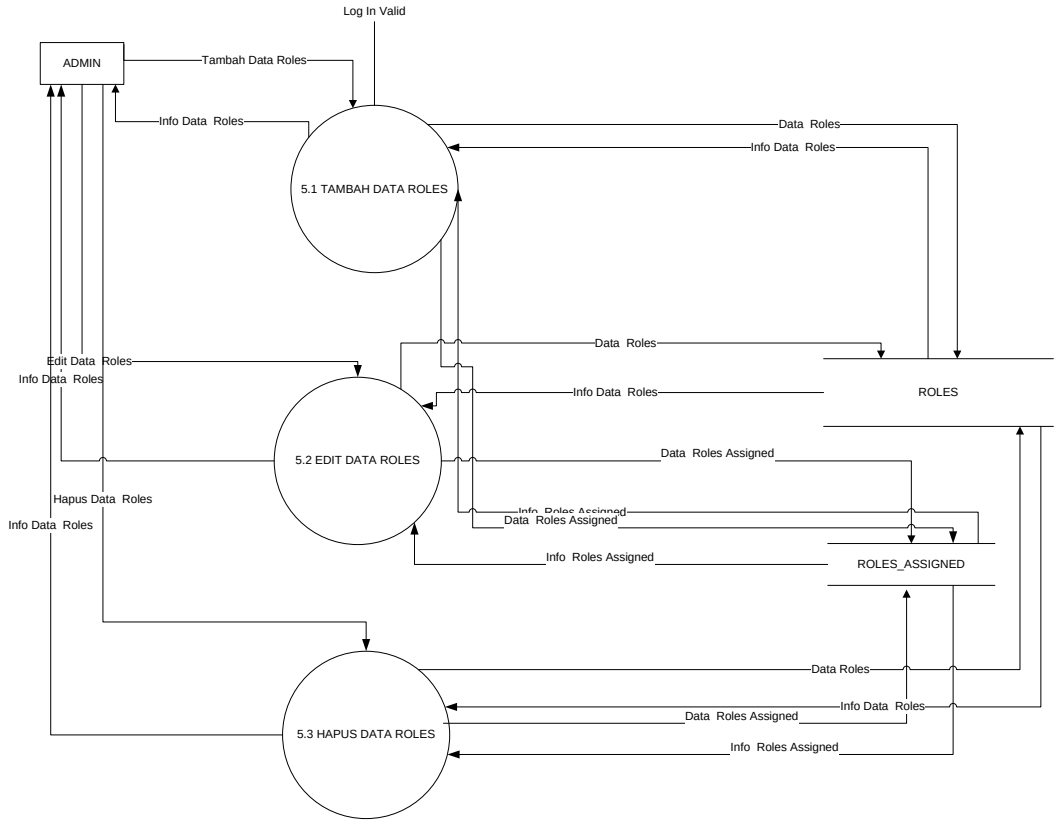
Proses dimana admin dapat menambahkan data *milestone* terbaru untuk dapat diketahui oleh semua *user*

4.2Proses Edit Data Milestones

Admin dapat mengubah suatu deskripsi dari *milestone* yang telah di *input*kan.

4.3Proses HapusData Milestones

Admin dapat menghapus data *milestone* yang telah di*inputkan*.



Gambar 3.8 DFD LEVEL 5 Proses Pengelolaan *Roles*

5.1 Proses Tambah Data Roles

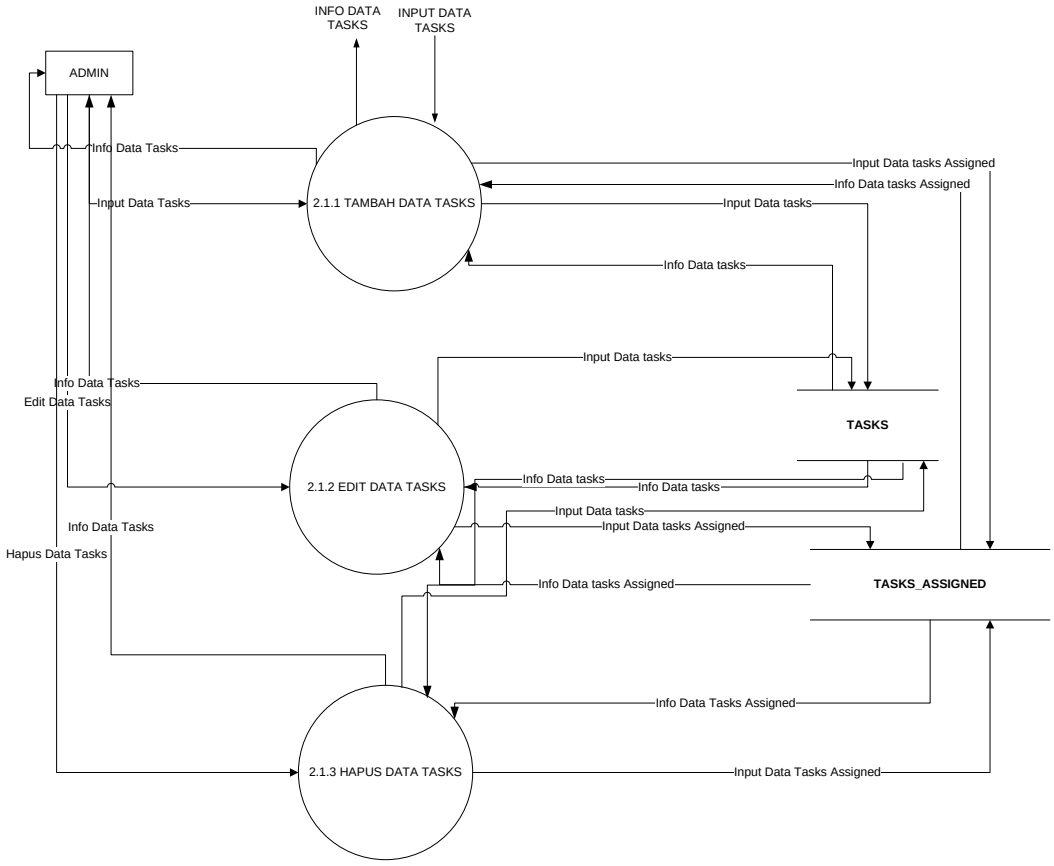
Proses dimana admin dapat memasukan data *role* untuk dapat dilihat oleh semua *user*.

5.2Proses Edit Data Roles

Admin dapat merubah data *role* yang telah di*input*kan lalu kemudian menyimpan kembali kedalam *Database*.

5.3Proses HapusData Roles

Admin dapat menghapus data suatu *role* yang telah di*input*kan kedalam *Database*.



Gambar 3.9 DFD LEVEL 2.1.1 Proses Pengelolaan *tasks*

2.1.1 Proses Tambah Data Tasks

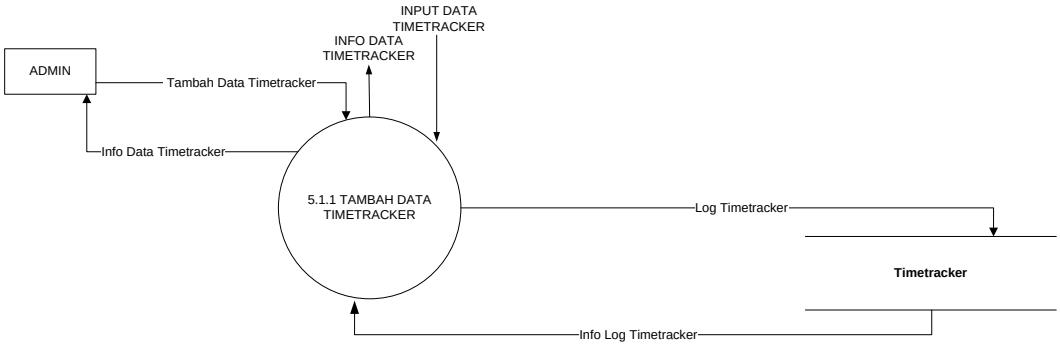
Adalah proses dimana admin dapat menambahkan setelah admin menambahkan data *project* sehingga akan terdapat *list tasks* didalam data *project*

2.1.2Proses Edit Data *Tasks*

Adalah proses dimana admin dapat merubah data *list tasks* kemudian menyimpan kembali ke dalam *Database*.

2.1.3Proses Hapus Data *Tasks*

Admin dapat menghapus list dari *tasks* yang telah di *input*kan.



Gambar 3.10 DFD LEVEL 5.1.1 Proses Pengelolaan *Timetracker*

5.1.1 Proses Tambah Data Timetracker

Admin dapat menambahkan data *timetracker* setelah data *tasks* di isikan yang berfungsi untuk mengetahui status projek *list tasks* apa saja yg telah diisikan dan akan disimpan dalam *database* berbentuk *LOG* yang tidak dapat diubah maupun dihapus.

3.1.6.3 Kamus Data

Kamus data merupakan penjelasan data yang dipergunakan pada suatu sistem. Berikut ini adalah kamus data yang digunakan dalam PARKIT.

Tabel 3.6 Kamus Data (Data Project)

No.	Kamus	Keterangan
1	Nama Data Aliran	Data *Project*
	Where used / how used	Proses 2.0 Proses pengelolaan *Project* Proses 2.1 Proses penambahan data *project* Proses 2.2 Proses Edit data *project* Proses 2.3 Proses Hapus *Projec*t
	Deskripsi	Data *user* admin merupakan data yang digunakan untuk mengisi proses pengelolaan *project*

Tabel 3.7 Kamus Data (Data User)

No.	Kamus	Keterangan
2	Nama Data Aliran	Data *User*
	Where used / how used	Proses 3.0 Pengelolaan *User* Proses 3.1 Proses Tambah *User* Proses 3.2 Proses Edit *User* Proses 3.3 Proses Hapus *User*
	Deskripsi	Hanya admin yang dapat menambahkan data *user* maupun admin.

Tabel 3.8 Kamus Data (Data Mailstones)

No.	Kamus	Keterangan
3	Nama Data Aliran	Data *Milestones*
	Where used / how used	Proses 4.0 Pengelolaan Milestones Proses 4.1 Proses Tambah Milestones Proses 4.2 Proses Edit Milestones Proses 4.3 Proses Hapus *Milestones*
	Deskripsi	Admin dapat memonitor perkembangan dan kemajuan *project*

Tabel 3.9 Kamus Data (Data Roles)

No.	Kamus	Keterangan
4	Nama Data Aliran	Data *Roles*
	Where used / how used	Proses 5.0 Pengelolaan *Roles* Proses 5.1 Proses Tambah *Roles* Proses 5.2 Proses Edit *Roles* Proses 5.3 Proses Hapus *Roles*
	Deskripsi	Hanya admin yang dapat memasukan,mengedit data *role* untuk dapat dilihat oleh semua *user*, dan dapat juga menghapusnya.

Tabel 3.10 Kamus Data (Data Tasks)

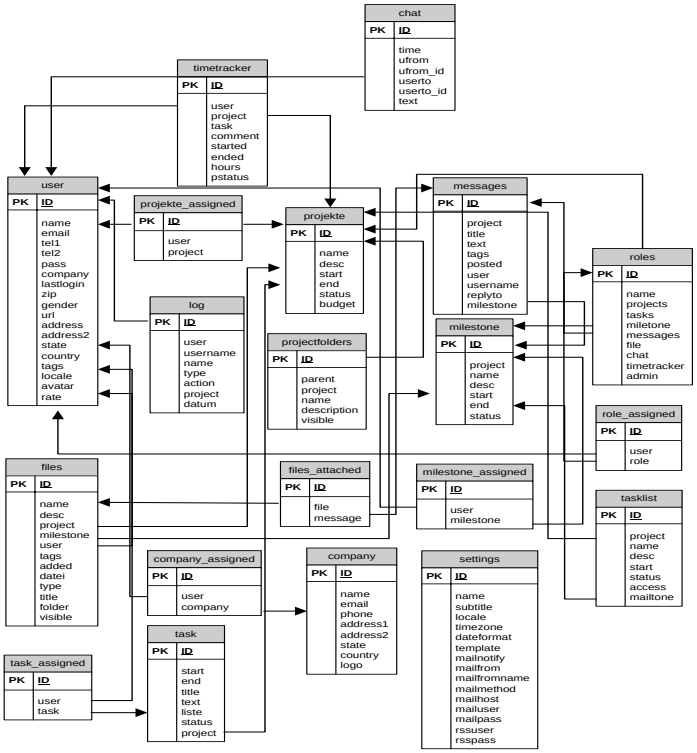
No.	Kamus	Keterangan
5	Nama Data Aliran	Data *Tasks*
	Where used / how used	Proses 2.1.1 Pengelolaan *Tasks* Proses 2.1.2 Proses Tambah *Tasks* Proses 2.1.3 Proses Edit *Tasks* Proses 2.1.4 Proses Hapus *Tasks*
	Deskripsi	Admin dapat menambahkan dan mengedit setelah admin menambahkan data *project* sehingga akan terdapat *list tasks* di dalam data *project*

Tabel 3.11 Kamus Data (Data Timetracker)

No.	Kamus	Keterangan
6	Nama Data Aliran	Data *Timetracker*
	Where used / how used	Proses 5.1.1 Pengelolaan *Timetracker* Proses 5.1.2 Proses Tambah *Timetracker*
	Deskripsi	*Timetracker* digunakan untuk mengontrol *tasklist* secara terperinci

3.1.6.4 Relasi Tabel

Berikut relasi tabel yang terjadi pada pada PARKIT.



Gambar 3.11 Relasi Tabel

3.1.6.5 Struktur Tabel

Berikut adalah struktur tabel yang digunakan sebagai penyimpanan data yang diolah pada PARKIT.

3.1.6.5.1 Tabel *Chat*

Chat memiliki field ID, time, ufrom,ufrom_id, userto, userto_id, text. Untuk melihat detail dari struktur tabel *chat* dapat dilihat ditabel di bawah ini

Tabel 3.12 Chat

Nama field	Tipe	Size	Keterangan
ID	int	10	auto_increment
time	varchar	255	
unfrom	varchar	255	
ufrom_id	int	10	
userto	varchar	255	
userto_id	int	10	
text	varchar	255	

3.1.6.5.2 Tabel *Company*

Company memiliki field ID, name, email, phone, address1, address2, state, country, logo. Untuk melihat detail dari struktur tabel *company* dapat dilihat ditabel di bawah ini

Tabel 3.13 Company

Nama field	Tipe	Size	Keterangan
ID	int	10	Auto_increment
name	varchar	255	
email	varchar	255	
phone	varchar	255	
address1	varchar	255	
address2	varchar	255	
state	varchar	255	
country	varchar	255	
logo	varchar	255	

3.1.6.5.3 Tabel *Company_assigned*

Company_assigned memiliki field ID, user, company. Untuk melihat detail dari struktur tabel *company_assigned* dapat dilihat ditabel di bawah ini

Tabel 3.14 Company_assigned

Nama field	Tipe	Size	Keterangan
ID	int	10	Auto_increment
user	int	10	
company	int	10	

3.1.6.5.4 Tabel *Files*

Files memiliki field ID, name, desc, project, milestones, user, tags, added, datei, type, folder, visible. Untuk melihat detail dari struktur tabel *files* dapat dilihat ditabel di bawah ini

Tabel 3.15 files

Nama field	Tipe	Size	Keterangan
ID	int	10	auto_increment
Name	varchar	255	
Desc	varchar	255	
project	int	10	
milestones	int	10	
User	int	10	

Tags	varchar	255	
Added	varchar	255	
Datei	varchar	255	
Type	varchar	255	
Folder	int	10	
Visible	text		

3.1.6.5.5 Tabel *files_attached*

Files_attached memiliki field ID, file, messages. Untuk melihat detail dari struktur tabel *files_attached* dapat dilihat ditabel di bawah ini

Tabel 3.16 files_attached

Nama field	Tipe	Size	Keterangan
ID	int	10	auto_increment
File	int	10	
message	int	10	

3.1.6.5.6 Tabel *log*

Log memiliki field ID, user, username, name, type, action, project, datum. Untuk melihat detail dari struktur tabel *log* dapat dilihat ditabel di bawah ini

Tabel 3.17 log

Nama field	Tipe	Size	Keterangan
ID	int	10	auto_increment
User	int	10	
username	varchar	255	
Name	varchar	255	
Type	varchar	255	
Action	int	1	
project	int	10	
Datum	varchar	255	

3.1.6.5.7 Tabel *Messages*

Messages memiliki field ID, project, title, text, tags, posted, user, username, replyto, milestone. Untuk melihat detail dari struktur tabel *messages* dapat dilihat ditabel di bawah ini

Tabel 3.18 Messages

Nama field	Tipe	Size	Keterangan
ID	int	10	auto_increment
project	int	10	
Title	varchar	255	
Text	text		
Tags	varchar	255	
Posted	varchar	255	
User	int	10	
username	varchar	255	
Replyto	int	11	
milestone	int	10	

3.1.6.5.8 Tabel *Milestone*

Milestone memiliki field ID, project, name, desc, start, end, status. Untuk melihat detail dari struktur tabel *milestone* dapat dilihat ditabel di bawah ini

Tabel 3.19 Milestone

Nama field	Tipe	Size	Keterangan
ID	int	10	auto_increment
project	Int	10	
Name	varchar	255	
Desc	text		
Start	varchar	255	
End	varchar	255	
Status	tinyint	1	

3.1.6.5.9 Tabel *milestone_assigned*

Milestones_assigned memiliki field ID, user, milestone. Untuk melihat detail dari struktur tabel *Milestones_assigned* dapat dilihat ditabel di bawah ini

Tabel 3.20 milestone_assigned

Nama field	Tipe	Size	Keterangan
ID	int	10	auto_increment
User	int	10	
milestone	int	10	

3.1.6.5.10 Tabel *projectfolders*

Projectfolders memiliki field ID, parent, project, name, description, visible. Untuk melihat detail dari struktur tabel *projectfolders* dapat dilihat ditabel di bawah ini

Tabel 3.21 projectfolders

Nama field	Tipe	Size	Keterangan
ID	int	10	auto_increment
Parent	int	10	
Project	int	11	
Name	text		
description	varchar	255	
Visible	text		

3.1.6.5.11 Tabel *projecte*

Projecte memiliki field ID, name, desc, start, end, status, budget. Untuk melihat detail dari struktur tabel *projecte* dapat dilihat ditabel di bawah ini

Tabel 3.22 Projecte

Nama field	Tipe	Size	Keterangan
ID	int	10	auto_increment
Name	varchar	255	
Desc	text		
Start	varchar	255	
End	varchar	255	
Status	tinyint	1	
budget	float		

3.1.6.5.12 Tabel *projecte_assigned*

projecte_assigned memiliki field ID, user, project. Untuk melihat detail dari struktur tabel *projecte_assigned* dapat dilihat ditabel di bawah ini

Tabel 3.23 projecte_assigned

Nama field	Tipe	Size	Keterangan
ID	int	10	auto_increment
User	int	10	
project	int	10	

3.1.6.5.13 Tabel *roles*

Roles memiliki field ID, name, projects, tasks, milestone, messages, files, chat, timetracker, admin . Untuk melihat detail dari struktur tabel *roles* dapat dilihat ditabel di bawah ini

Tabel 3.24 roles

Nama field	Tipe	Size	Keterangan
ID	int	10	auto_increment
Name	varchar	255	
projects	text		
Tasks	text		
milestones	text		
messagess	text		
Files	text		
Chat	text		
timetracker	text		
Admin	text		

3.1.6.5.14 Tabel *roles_assigned*

Roles_assigned memiliki field ID, user, role . Untuk melihat detail dari struktur tabel *roles_assigned* dapat dilihat ditabel di bawah ini

Tabel 3.25 roles_assigned

Nama field	Tipe	Size	Keterangan
ID	int	10	auto_increment
User	int	10	
Role	int	10	

3.1.6.5.15 Tabel *settings*

Settings memiliki field ID, name, subtitle, locale, timezone, datefrom,template, mailnotify, mailfrom, mailfromname, mailmethod, mailhost, mailuser, mailpass, rssuser, rsspass . Untuk melihat detail dari struktur tabel *settings* dapat dilihat ditabel di bawah ini

Tabel 3.26 settings

Nama field	Tipe	Size	Keterangan
ID	tinyint	1	auto_increment
Name	varchar	255	

subtitle	varchar	255	
Locale	varchar	6	
timezone	varchar	60	
datefrom	varchar	50	
template	varchar	255	
mailnotify	tinyint	1	
mailfrom	varchar	255	
mailfromname	varchar	255	
mailmethod	varchar	5	
mailhost	varchar	255	
mailuser	varchar	255	
mailpass	varchar	255	
Rssuser	varchar	255	
Rsspass	varchar	255	

3.1.6.5.16 Tabel *tasklist*

*Tasklis*t memiliki field ID, project, name, desc, start, status, access, milestone . Untuk melihat detail dari struktur tabel *tasklist* dapat dilihat ditabel di bawah ini

Tabel 3.27 tasklist

Nama field	Tipe	Size	Keterangan
ID	int	10	auto_increment
project	int	10	
Name	varchar	255	
Desc	text		
Start	varchar	255	
Status	tinyint	1	
Access	tinyint	4	
milestone	int	10	

3.1.6.5.17 Tabel *task*

Task memiliki field ID, start, end, title, text, liste, status, project . Untuk melihat detail dari struktur tabel *task* dapat dilihat ditabel di bawah ini

Tabel 3.28 task

Nama field	Tipe	Size	Keterangan
ID	int	10	auto_increment
Start	varchar	255	
End	varchar	255	
Title	varchar	255	
Text	text		

Liste	int	10	
Status	tinyint	1	
project	int	10	

3.1.6.5.18 Tabel *tasklist_assigned*

Tasklist_assigned memiliki field ID, user, task . Untuk melihat detail dari struktur tabel *tasklist_assigned* dapat dilihat ditabel di bawah ini

Tabel 3.29 tasklist_assigned

Nama field	Tipe	Size	Keterangan
ID	int	10	auto_increment
User	int	10	
Task	int	10	

3.1.6.5.19 Tabel *timetracker*

Timetracker memiliki field ID, user, project, task, comment, started, ended, hours, pstatus. Untuk melihat detail dari struktur tabel *timetracker* dapat dilihat ditabel di bawah ini

Tabel 3.30 timetracker

Nama field	Tipe	Size	Keterangan
ID	int	10	auto_increment
User	int	10	
project	int	10	
task	int	10	
comment	text		
started	varchar	255	
ended	varchar	255	
hours	float		
pstatus	tinyint	1	

3.1.6.5.20 Tabel *user*

User memiliki field ID, name, email, tel1, tel2, pass, company, lastlogin, zip, gender, url, address, address2, state, country, tags, locale, avatar, rate. Untuk melihat detail dari struktur tabel *user* dapat dilihat ditabel di bawah ini

Tabel 3.31 user

Nama field	Tipe	Size	Keterangan
ID	int	10	auto_increment

name	varchar	255	
email	varchar	255	
tel1	varchar	255	
tel2	varchar	255	
pass	varchar	255	
company	varchar	255	
lastlogin	varchar	255	
zip	varchar	10	
gender	char	1	
url	varchar	255	
address	varchar	255	
address2	varchar	255	
state	varchar	255	
country	varchar	255	
tags	varchar	255	
locale	varchar	6	
avatar	varchar	255	
rate	varchar	10	

3.2 Keterkaitan Siklus Manajemen Proyek dengan PARKIT

Setiap proyek harus direncanakan secara detail dan terkontrol. Kontrol berhubungan dengan membandingkan *progress* kerja aktual dengan rencana sebelumnya dan memberikan koreksi ketika rencana tidak sesuai dengan aksi. Rencana proyek akan disiapkan oleh manajer proyek dan menspesifikasikan kebutuhan pekerjaan yang akan dilakukan.

Karena DCISTEM merupakan bagian dari UPT UNPAD maka proyek yang dikerjakan oleh DCISTEM dibuat dan dikerjakan hanya untuk UNPAD sehingga *stakeholder* keseluruhan dipegang oleh UNPAD. Dalam hal ini yang berperan sebagai manajer proyek adalah Sekretaris Bidang, pelanggan adalah *civitas* UNPAD, organisasi proyek adalah DCISTEM, dan sponsor adalah UNPAD itu sendiri.

Berikut adalah langkah – langkah yang dilakukan pada saat melakukan manajemen proyek di DCISTEM – UNPAD mengikuti siklus manajemen proyek :

1. Fase pelingkupan

   Fase pelingkupan merupakan fase penentuan batas sebuah proyek. Proses negosiasi lingkup proyek mendominasi fase ini. Hasil proses negosiasi

lingkup proyek dituangkan dalam sebuah *statement of work* atau pernyataan kerja.

Pada fase pelingkupan, manajer proyek melakukakn analisis penuh terhadap sistem harus diadakan untuk memproduksi kebutuhan – kebutuhan fungsional seluruh sistem. Sebelum membangun pekerjaan tersendiri dalam menyelesaikan proyek, termasuk menetapkan masalah yang terjadi dalam sistem yang sedang berjalan, menetapkan tujuan dan gol, dan mendaftar kendala – kendala atau keterbatasan. Menentukan dan mendefinisikan interface dengan sistem berjalan lainnya, dan kebutuhan – kebutuhan di dalam atau di luar bagian. Seluruh hasil analisis kemudian didiskusikan dengan para staf DCISTEM yang bersangkutan sampai menghasilkan pernyataan kerja.

2. Fase perencanaan

   Fase perencanaan adalah fase mengidentifikasikan tugas – tugas yang terlibat dalam proyek. Merinci tugas menjadi tingkatan – tingkatan tugas (subtugas). Hasil proses identifikasi tugas adalah *work breakdown structure* atau struktur rincian tugas.

   Pada fase perencanaan, struktur rincian tugas dituangkan dalam PARKIT. Proyek, *milestone, taklist, dan subtask list* tersusun dalam PARKIT.

3. Fase perkiraan

   Fase perkiraan adalah fase perkiraan waktu masing – masing tugas. Pada fase ini dilakukan penentuan durasi setiap tugas sesuai kapasitas tim proyek.

4. Fase penjadwalan

   Fase penjadwalan adalah fase menentukan ketergantungan antar tugas yang membangun proyek secara keseluruhan.

5. Fase pengorganisasian

   Fase pengorganisasian adalah fase penempatan sumber daya pada masing – masing tugas dan menyeimbangkan (*leveling*) sumber daya. Pada PARKIT terdapat fasilitas yang dapat menunjukkan secara langsung kepada siapa tugas akan diberikan.

6. Fase pengarahan

   Fase pengarahan adalah fase yang dilakukan pada saat implementasi proyek. Implementasi proyek perlu pengarahan tim akan rencana yang telah disepakati sehingga proyek dapat diselesaikan secara optimal.

7. Fase pengontrolan

   Fase pengontrolan adalah fase memonitor dan mengontrol kemajuan proyek, penyampaian laporan perkembangan proyek dan perubahan proyek. Pada PARKIT fase pengontrolan terhitung secara otomatis.

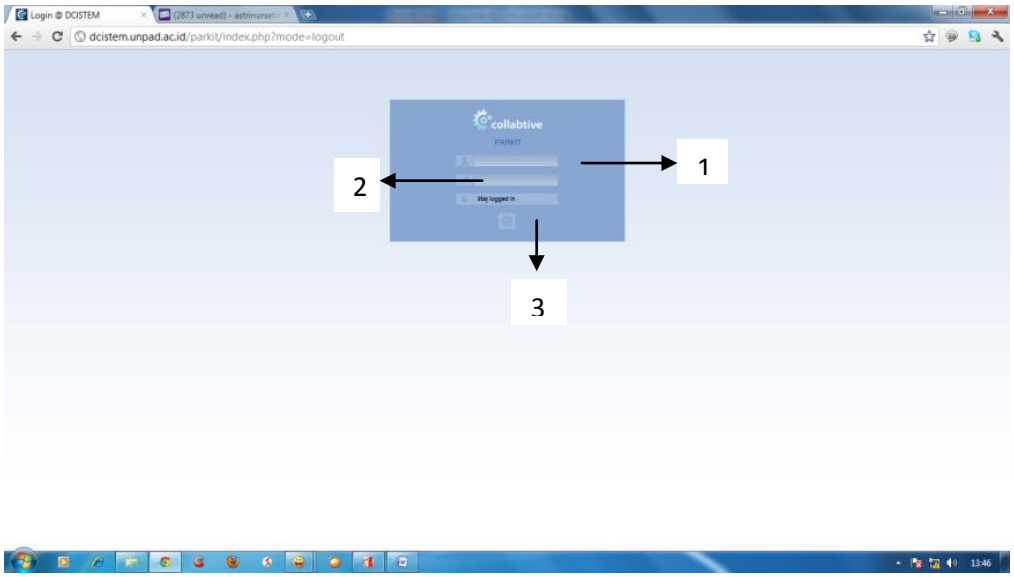
8. Fase penutupan

   Fase penutupan adalah fase menilai hasil akhir proyek dan masukan atau pengalaman yang didapat yang akan berguna bagi proyek – proyek selanjutnya.

3.3 PARKIT (Padjajaran Kolaborasi IT)

3.3.1. Jelajah Halaman PARKIT

Berikut akan dijelaskan tampilan – tampilan dari sistem PARKIT. Pertama – tama untuk dapat menggunakan sistem PARKIT ini, pengguna harus telah terdaftar sebagai pengguna di sistem PARKIT, sehingga halaman yang pertama kali muncul pada sistem PARKIT ni adalah **halaman *login*.** Berikut adalah tampilan dari halaman *login* PARKIT.

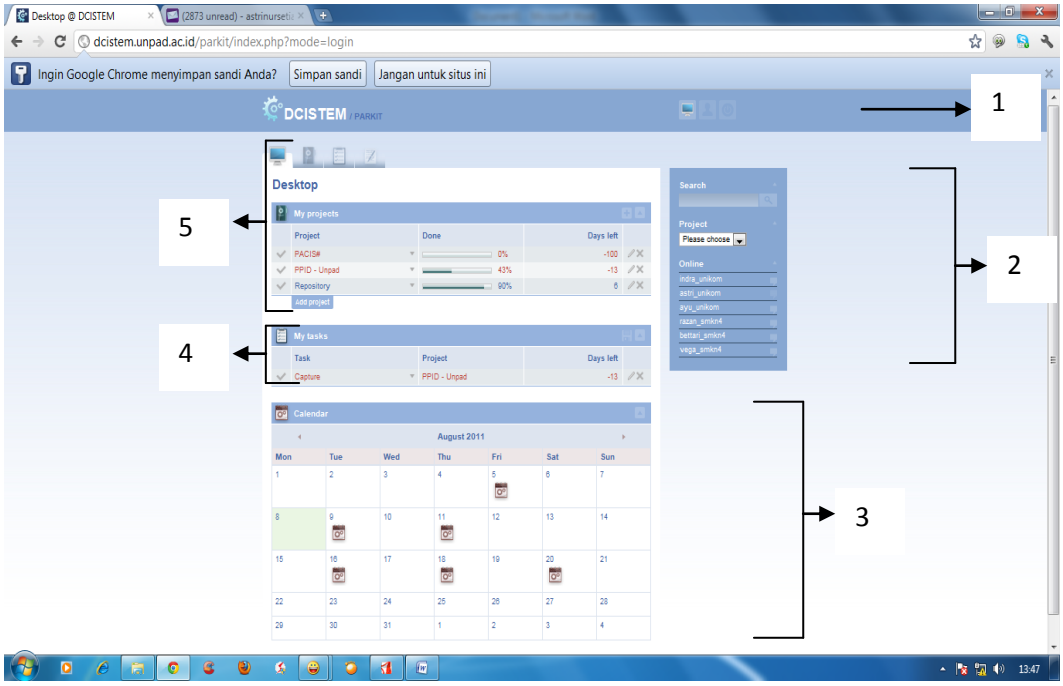


Gambar 3.12 Halaman login.

Keterangan :

1. *Field* untuk mengetikkan *username.*
2. *Field* untuk mengtikkan *password.*
3. Tombol *login* yang berfungsi untuk menghubungkan pengguna ke halaman berikutnya apabila *username* dan *password* telah diisi dengan benar.

Setelah pengguna berhasil *login* dengan menggunakan *username* dan *password* yang benar, maka akan muncul **halaman *desktop***. Berikut adalah tampilan dari halaman *desktop*.



Gambar 3.13 *Halaman desktop.*

Keterangan :

1. Bagian ini menjelaskan *icon – icon* dan fungsi *icon* tersebut.

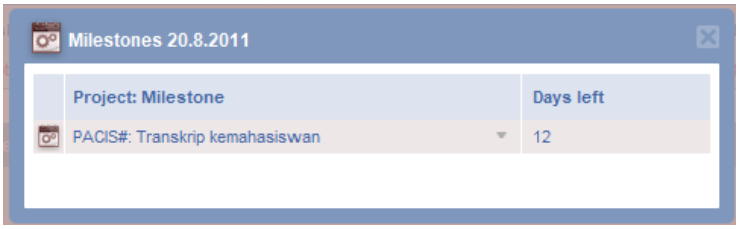
   Icon  adalah ***icon desktop***. Apabila *pointer* mouse disorot ke *icon* ini maka disamping *icon* ini akan muncul *hint* bertulisan "*Desktop*", namun apabila diklik *icon* ini akan menghubungkan atau mengembalikan ke halaman *desktop*.

Icon  adalah ***icon My account***. Apabila *pointer* mouse disorot ke *icon* ini maka disamping *icon* ini akan muncul hint bertulisan "*My account*", namun apabila diklik *icon* ini akan menghubungkan ke halaman *My account* yang berisi profil pengguna. Halaman *My account* akan dibahas pada bahasan selanjutnya.

Gambar  adalah *iconLogout*. Apabila *pointer* mouse disorot ke *icon* ini maka disamping *icon* ini akan muncul *hint* bertulisan "*Logout*", namun apabila diklik *icon* ini akan mengeluarkan pengguna dari sistem PARKIT.

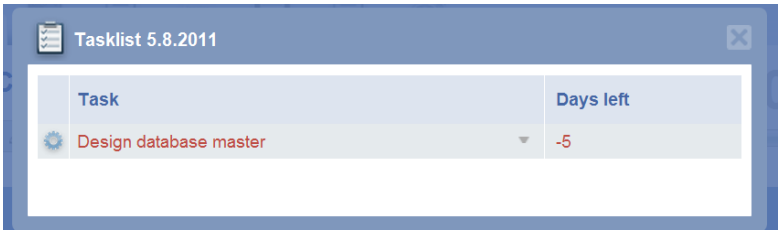
2. Bagian ini terdiri dari *search*, *project*, dan *online*.
   - *Search* adalah fasilitas untuk mempermudah pengguna untuk melakukan pencarian baik pencarian *project*, *milestone*, maupun pencarian *task*.
   - *Project* adalah fasilitas untuk mempermudah pengguna untuk melakukan pencarian *project*.
   - *Online* adalah fasilitas untuk mengetahui siapa saja pengguna yang sedang menggunakan sistem PARKIT ini. Fasilitas ini juga memungkinkan pengguna untuk melihat profil pengguna lain dengan cara mengklik *username* yang muncul.

3. *Calendar*, adalah fasilitas untuk memudahkan pengguna melihat rencana projek. Apabila ada projek yang harus dikerjakan maka pada tanggal dikalender terdapat *icon*  dan  . Apabila *icon*  di klik maka akan muncul pemberitahuan berbentuk *showmessage* berupa keterangan *milestone* pada projek tersebut. Contoh tampilan *showmessage* tersebut adalah sebagai berikut :



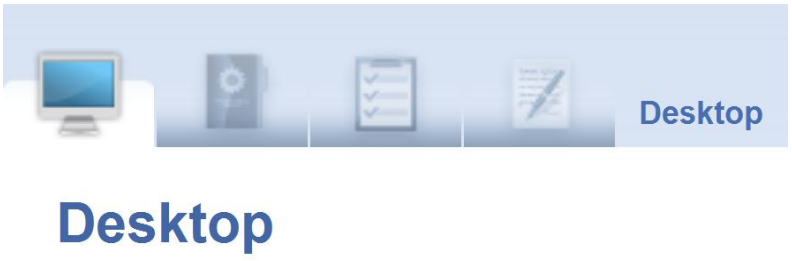
Gambar 3.14 Showmessage Milestones.

Apabila *icon*  di klik maka akan muncul pemberitahuan berbentuk *showmessage* berupa keterangan *Tasklist*. Contoh tampilan *showmessage* tersebut adalah sebagai berikut :



Gambar 3.15 Showmessage Tasklist.

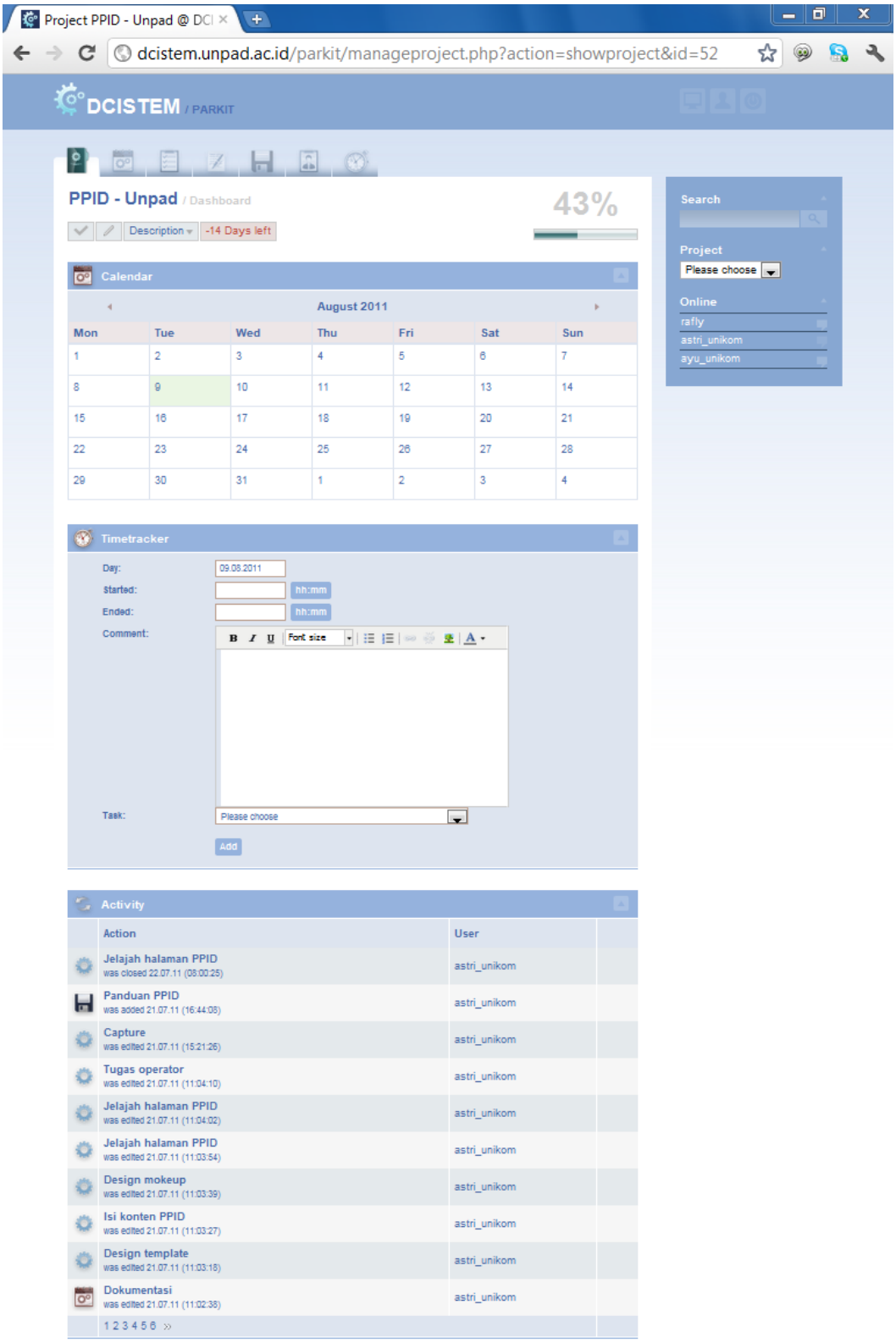
4. *My task*, bagian ini merupakan fasilitas pemberitahuan pengguna mengenai tugas apa saja yang harus diselesaikan oleh pengguna.
5. Bagian ini merupakan fasilitas pemberitahuan pengguna mengenai projek yang berkaitan dengan pengguna.



Gambar 3.16 Tabulasi pilihan desktop**.**

Apabila tabulasi *desktop* diklik maka tampilan yang muncul adalah (Gambar 2) Halaman *desktop*.

Apabila nama projek di klik maka akan muncul tampilan seperti gambar dibawah ini :



Gambar 3.17 Tampilan bagian dalam project

Pada tampilan diatas terbagi menjadi 3 bagian yaitu :

- *Calendar* yang fungsinya sama dengan *calendar* yang berada pada halaman *desktop*yaitu fasilitas untuk memudahkan pengguna melihat rencana projek. Apabila ada projek yang harus dikerjakan maka pada tanggal dikalender terdapat *icon*  dan . Apabila *icon*  di klik maka akan muncul pemberitahuan berbentuk *showmessage* berupa keterangan *milestone* pada projek tersebut. Contoh tampilan *showmessage* tersebut adalah Seperti *(Gambar 3) Showmessage Milestones* :

  Apabila *icon*  di klik maka akan muncul pemberitahuan berbentuk *showmessage* berupa keterangan *Tasklist*. Contoh tampilan *showmessage* tersebut adalah Seperti *(Gambar 4) Showmessage Tasklist* :

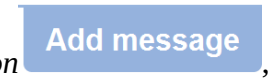
- *Timetracker* berfungsi untuk mengontrol *Tasklist* secara terperinci berdasarkan waktu mulai dan waktu selesai. Cara menggunakannya isikan *day* dengan tanggal, *started* dengan waktu dimulainya pengerjaan *Tasklist*, *ended* dengan waktu penyelesaian *task*tist, *comment* dengan komentar mengenai *Tasklist* yang dikerjakan, *task* dengan memilih *task* yang akan dikerjakan. Apabila semua bagian yang perlu diisi telah diisi klik 

.
- *Activity* berfungsi sebagai pemberitahuan aktifitas apa saja yang dilakukan pengguna, *activity* ini seperti *log* sehingga tidak dapat dihapus.

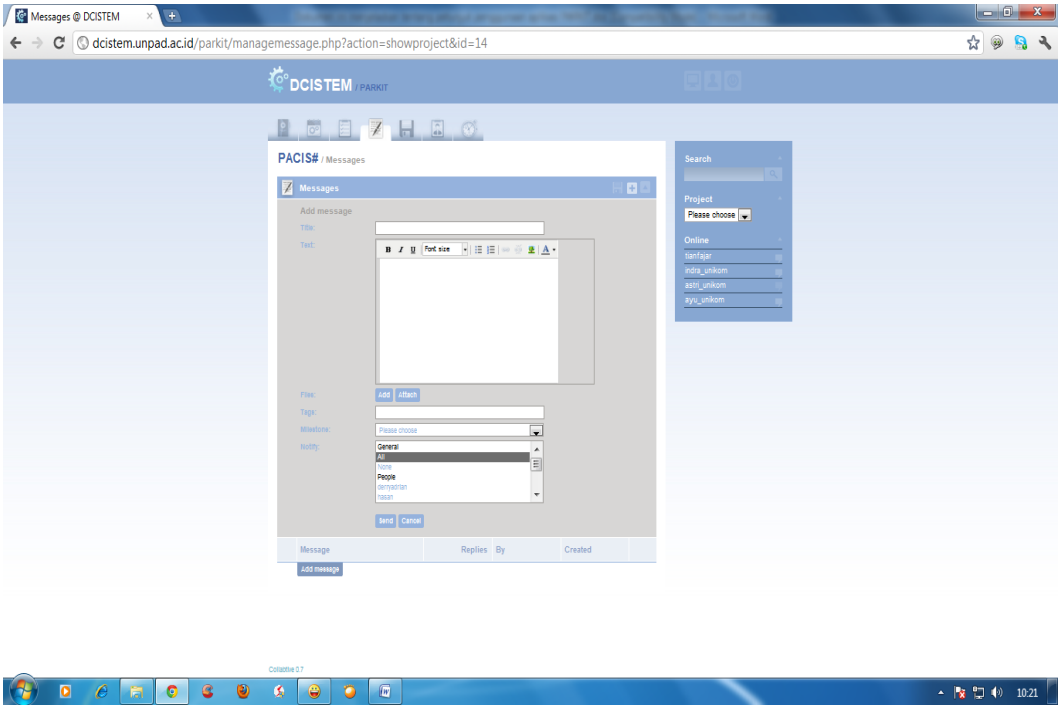
Tabulasi *milestone* pada halaman dalam projek ini berisi *milestone* projek, untuk bahasan yang lebih terperinci akan dijelaskan pada bahasan cara menambahkan *milestone* pada projek dalam sistem PARKIT.

Tabulasi *Tasklist* pada halaman dalam projek ini berisi *Tasklist* dan *subtasklist* projek, untuk bahasan yang lebih terperinci akan dijelaskan pada

bahasan cara menambahkan *Tasklist* dan *subtasklist* pada projek dalam sistem PARKIT.

Tabulasi *Messages* pada halaman dalam projek ini berisi fasilitas untuk saling berhubungan dan bertukar data dalam ruang lingkup projek tersebut. Cara mengirimkan pesan menggunakan fasilitas ini adalah dengan cara mengklik *icon* 

, maka akan muncul tampilan seperti berikut :



Gambar 3.18 Tampilan dalam message

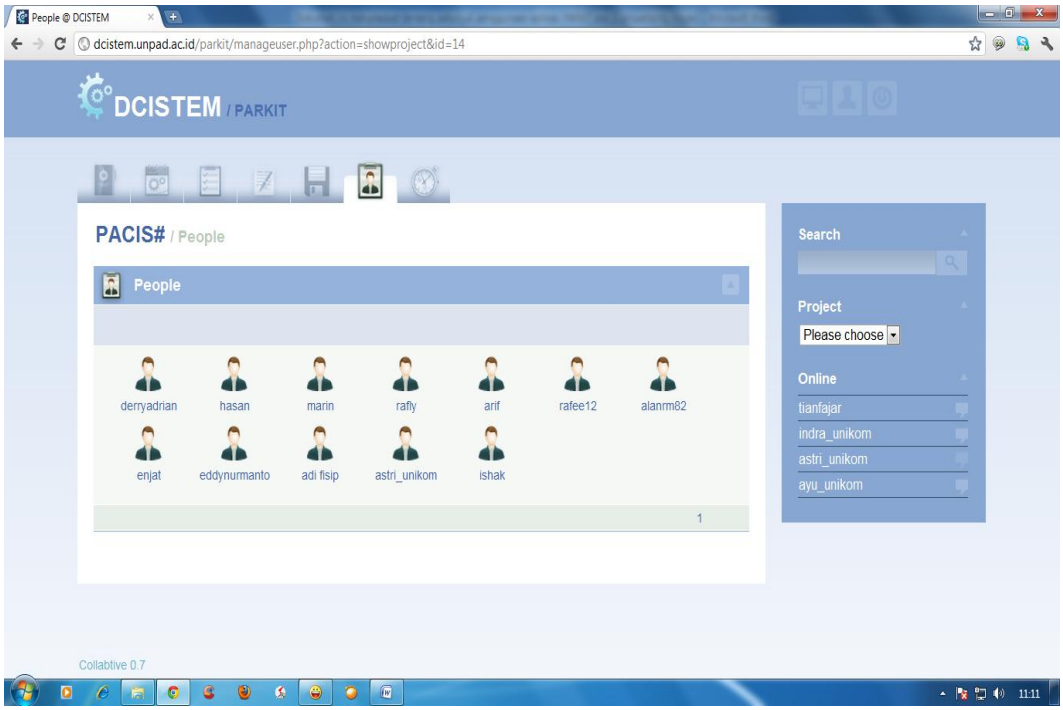
Isikan *title* dengan judul pesan, *text* dengan isi pesan, *file*s apabila perlu mengirimkan data yang harus di *upload*, *tags* dengan keterangan, *milestone* dengan memilih *milestone* yang bersangkutan, *notify* kepada siapa pesan ini ditujukan. Apabila semua komponen pesan yang dibutuhkan telah diisi klik 

.

Pengguna juga dapat meng*eksport message* ke bentuk RSS maupun bentuk PDF, dengan cara mengklik *icon* , secara otomatis akan muncul pilihan RSS ataupun PDF.

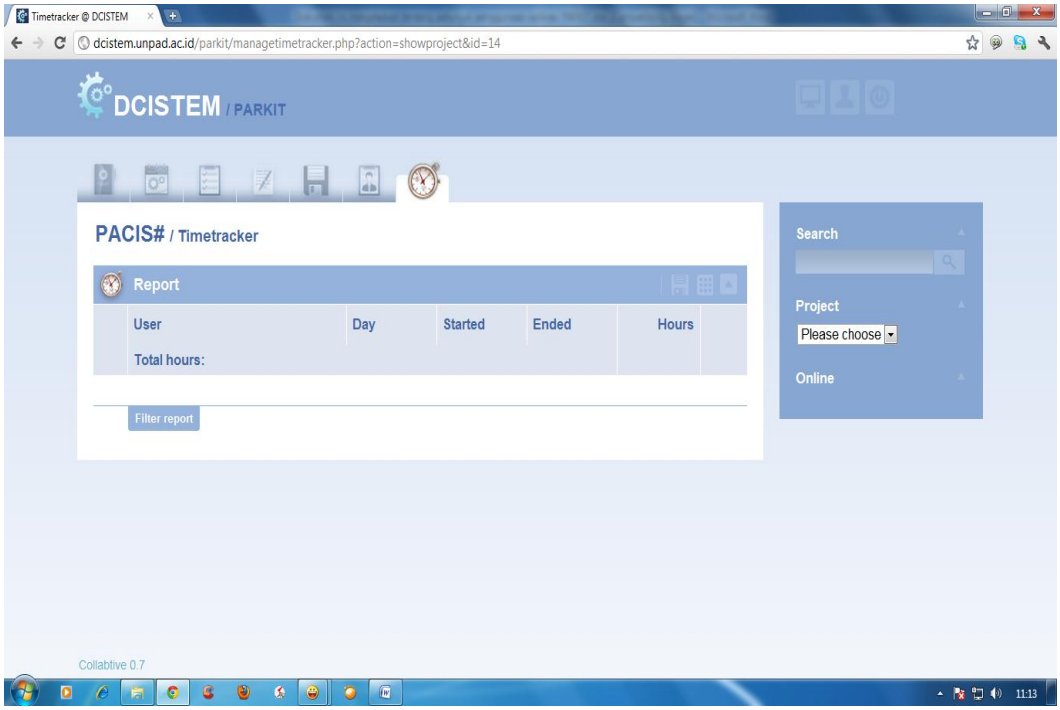
Tabulasi *file*s pada halaman dalam projek ini berfungsi untuk saling berkirim data sama seperti fungsi *attachment* pada tab *message*s, namun *file* akan langsung tersimpan pada *root directory*.

Tabulasi *user* memberikan informasi mengenai semua pengguna yang menggunakan sistem PARKIT. Berikut adalah tampilan yang muncul apabila pengguna mengklik tabulasi *user* :



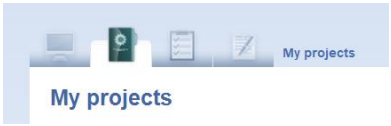
Gambar 3.19 Tampilan pengguna PARKIT

Terakhir adalah tabulasi *timetracker* yang berfungsi untuk memberikan infomasi mengenai keseluruhan waktu yang terpakai apabila pengguna menggunakan *timetracker* dalam mengerjakan *task*. Berikut adalah tampilan halamannya apabila pengguna mengklik tabulasi *timetracker* :



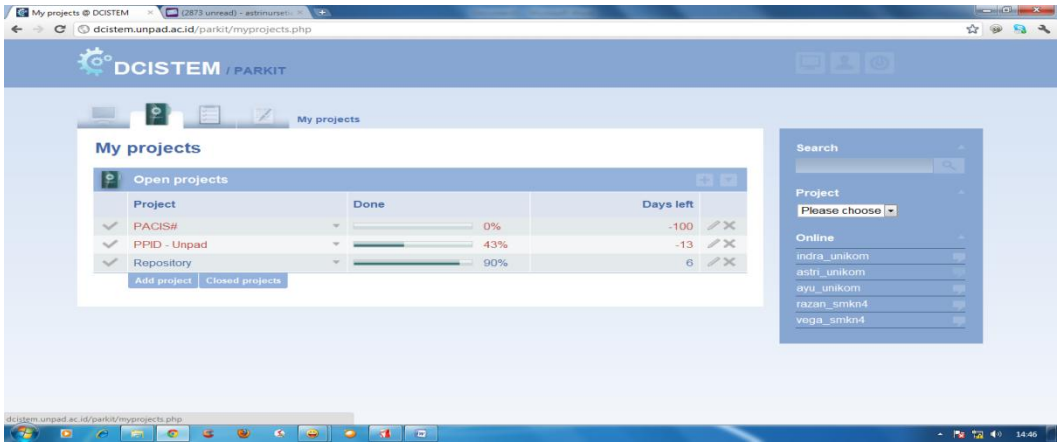
Gambar 3.20 Tampilan bagian dalam timetracker

Fasilitas *filterreport* mempermudah pengguna untuk mendapatkan laporan sesuai yang diingikan.



Gambar 3.21 Tabulasi pilihan My project.

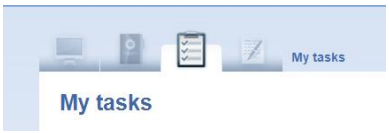
Apabila tabulasi *My project* diklik maka tampilan akan berubah seperti gambar di bawah ini :



Gambar 3.22 Halaman My project.

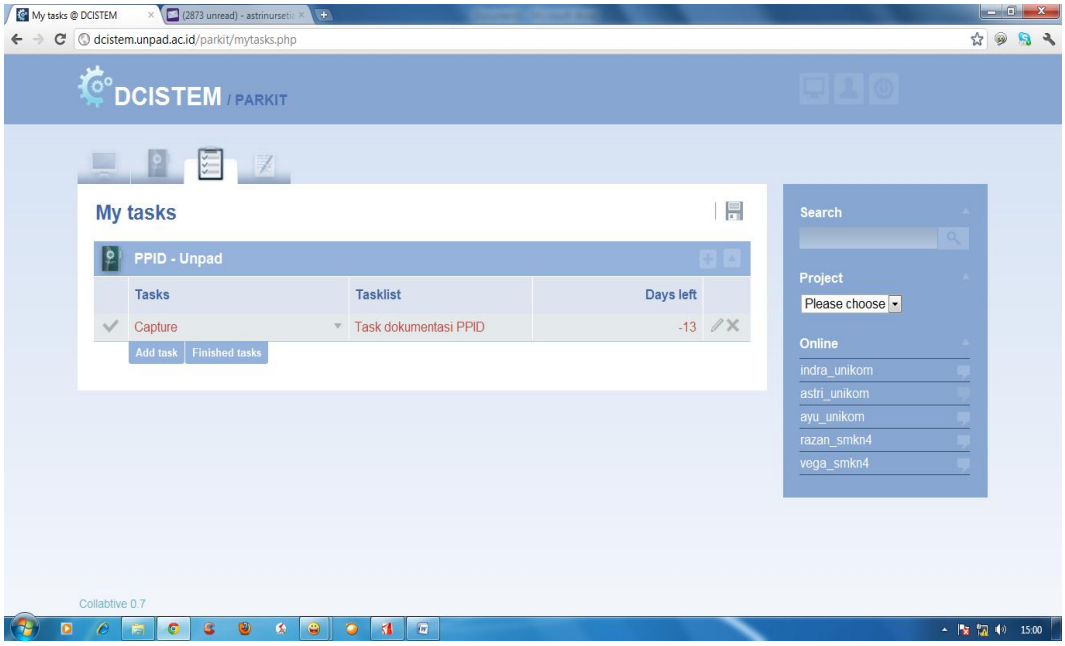
Seperti yang dapat kita lihat, tabulasi *My project* merupakan fasilitas untuk menampilkan seluruh projek yang terhubung dengan pengguna. Pada tampilan ini terdapat tiga kolom. Kolom pertama adalah *project,* menunjukkan nama projek. Kolom kedua adalah *Done,* memberikan informasi berupa keterangan gambar, dimana keterangan ini menunjukkan informasi sejauh mana projek telah diselesaikan dengan melihat persentase yang telah dicapai pada projek tersebut. Kolom ketiga adalah *Days left,* menunjukkan waktu pencapaian target berdasarkan hari.

Pada tampilan isi kolom diatas terdapat warna tulisan atau *font color* yang berbeda. *Font color* merah menunjukkan bahwa projek tersebut sudah melewati target waktu yang ditetapkan, sedangkan *font color* biru menunjukkan bahwa projek tersebut belum melewati atau mendekati target waktu yang telah ditetapkan.



Gambar 3.23 Tabulasi pilihan My task**.**

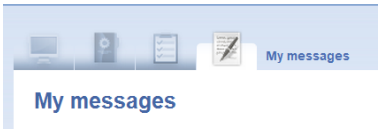
Apabila tabulasi *My task* diklik maka tampilan akan berubah seperti gambar di bawah ini :



Gambar 3.24 Halaman My task.

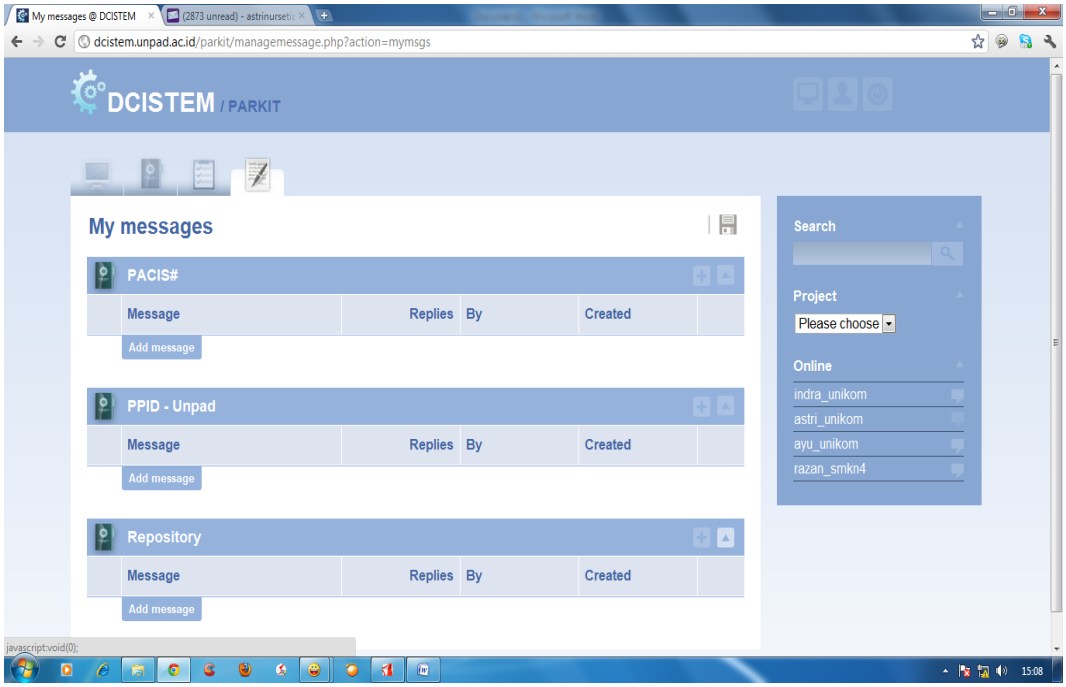
Halaman ini memberikan informasi kepada pengguna mengenai *task* yang harus dikerjakan pengguna yang sedang *login* tersebut. Pada tampilan *My task*ini juga terdapat tiga kolom. Kolom pertama yaitu *Tasks*, merupakan nama *task*. Kolom kedua adalah *Tasklist*, merupakan sub*task* dari *task* sebelah kiri. Kolom ketiga adalah *Days left* menunjukkan waktu pencapaian target berdasarkan hari.

Sama seperti halaman *My project*, *font color* merah menunjukkan bahwa *task* tersebut sudah melewati target waktu yang ditetapkan, sedangkan *font color* biru menunjukkan bahwa *task* tersebut belum melewati atau mendekati target waktu yang telah ditetapkan.



Gambar 3.25 Tabulasi pilihan My Message.

Apabila tabulasi *My task* diklik maka tampilan akan berubah seperti gambar di bawah ini :



Gambar 3.26 Halaman My Message.

Halaman ini berfungsi untuk saling menghubungkan pengguna mengenai projek yang sama ataupun untuk saling mengirim data. Pada tamipilan ini diperlihatkan semua projek yang ada, sehingga pada saat akan mengirimkan pesan pengguna harus memilih projeknya terlebih dahulu. Tata cara mengirimkan pesan sama dengan tata cara mengirimkan pesan yang telah dibahas sebelumnya.

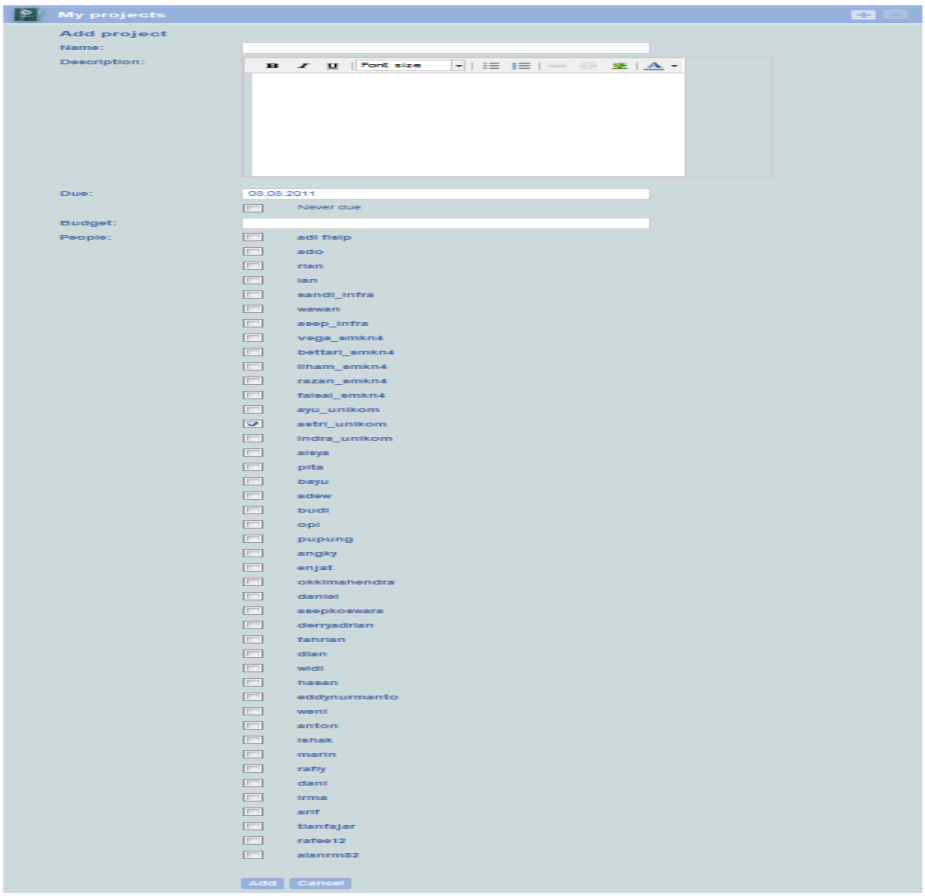
3.3.2 Membuat / Menambahkan Projek Baru di PARKIT

Berikut adalah tata cara menambahkan projek di PARKIT, namun hanya beberapa pengguna yang boleh menambahkan projek pada sistem PARKIT ini :

1. Setelah berhasil melakukan *login* tentu secara otomatis pengguna akan masuk ke halaman *desktop* (gambar 2). Untuk menambahkan projek klik *icon* (+), seperti gambar dibawah ini

    atau klik 

2. Setelah mengklik *icon* diatas akan muncul tampilan seperti di bawah ini :



Gambar 3.27 Halaman untuk menambahkan projek baru

3. Isi *name* dengan nama projek, *description* dengan deskripsi projek, *due* dengan target tanggal projek selesai, *budget* dengan jumlah perkiraan biaya yang akan dikeluarkan, centang nama –nama atau siapa saja yang akan dimasukkan ke dalamprojek yang akan dibuat ini. Apabila telah selesai semua klik 

.

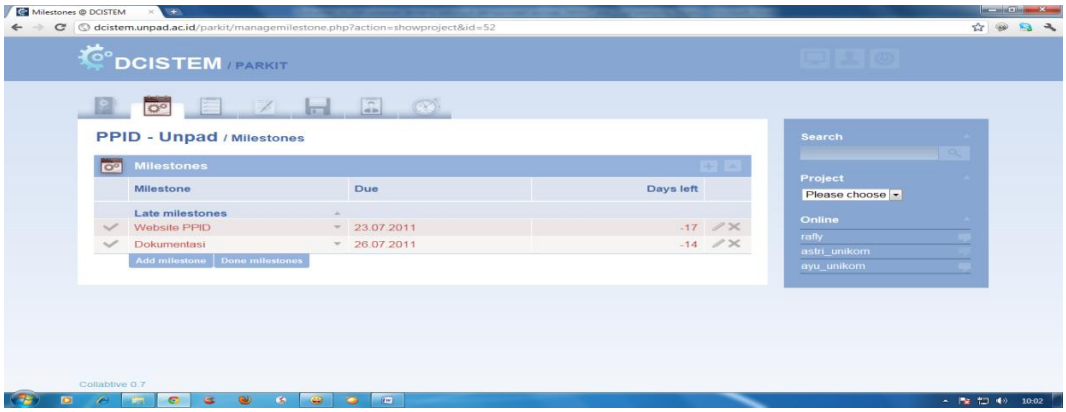
   Apabila terjadi kekeliruan pengguna dapat mengedit projek yang sudah terlanjur dibuat dengan cara pilih nama projek yang akan diedit kemudian klik *icon* . Namun apabila projek yang terlanjur dibuat salah sehingga harus dihapus pengguna dapat menghapus projek tersebut dengan cara mengklik *icon* .

3.3.3 Membuat / Menambahkan Milestone di Proyek PARKIT

Sebelum penulis memaparkan tata cara menambahkan *milestone* di parkit, ada baiknya mengetahui terlebih dahulu apa yang dimaksud dengan *milestone*. *Milestone* adalah suatu referensi sebagai tanda penunjuk suatu peristiwa dalam sebuah projek yang digunakan untuk memonitor atau mengetahui pekembangan dan kemajuan projek.

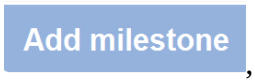
Berikut adalah tata cara menambahkan *milestone* di PARKIT :

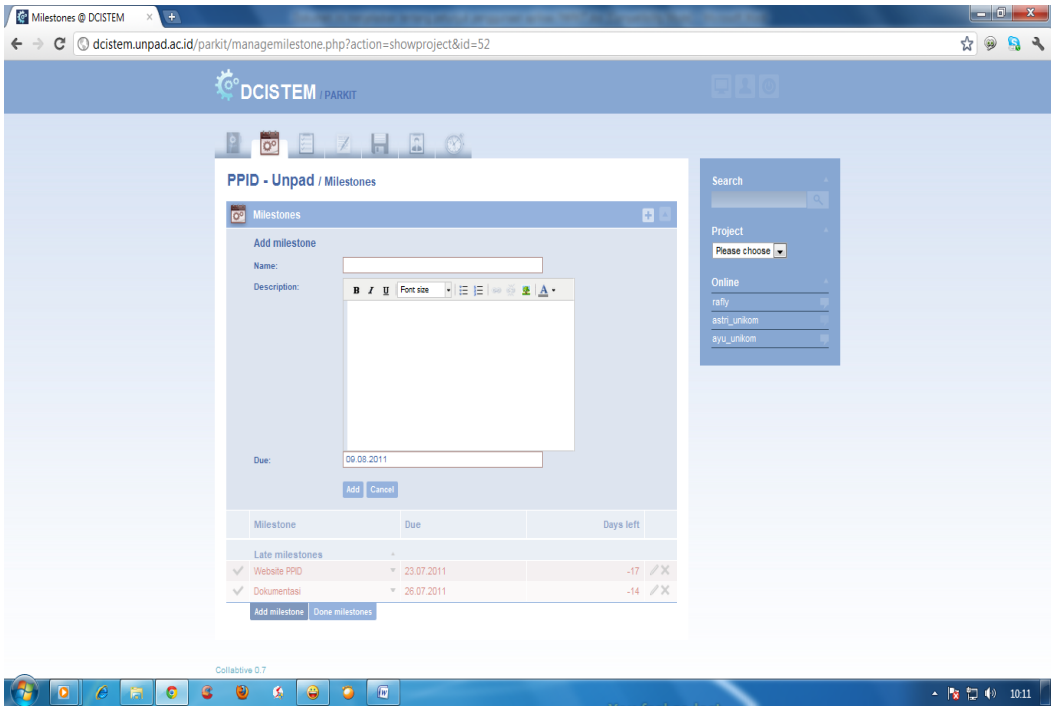
1. Klik projek yang akan diberi *milestone*, maka akan muncul tampilan seperti *(Gambar 12) Halaman projek*
2. Klik tabulasi *milestone*, maka tampilannya akan berubah sebagai berikut :



Gambar 3.28 Halaman milestone

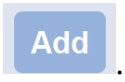
Sama seperti bahasan sebelumnya *font color* merah menunjukkan bahwa *milestone* telah melewati waktu yang telah ditargetkan, sedangkan *font color* biru menunjukkan *milestone* yang belum melewati waktu yang telah ditargetkan.

3. Klik  atau , maka tampilan yang muncul adalah sebagai berikut :



Gambar 3.29 Halaman untuk menambahkan milestone baru

4. Isikan *name* dengan nama *milestone* baru, *description* dengan deskripsi dari *milestone* yang sedang dibuat tersebut, *due* dengan target waktu yang ditentukan berdasarkan tanggal.

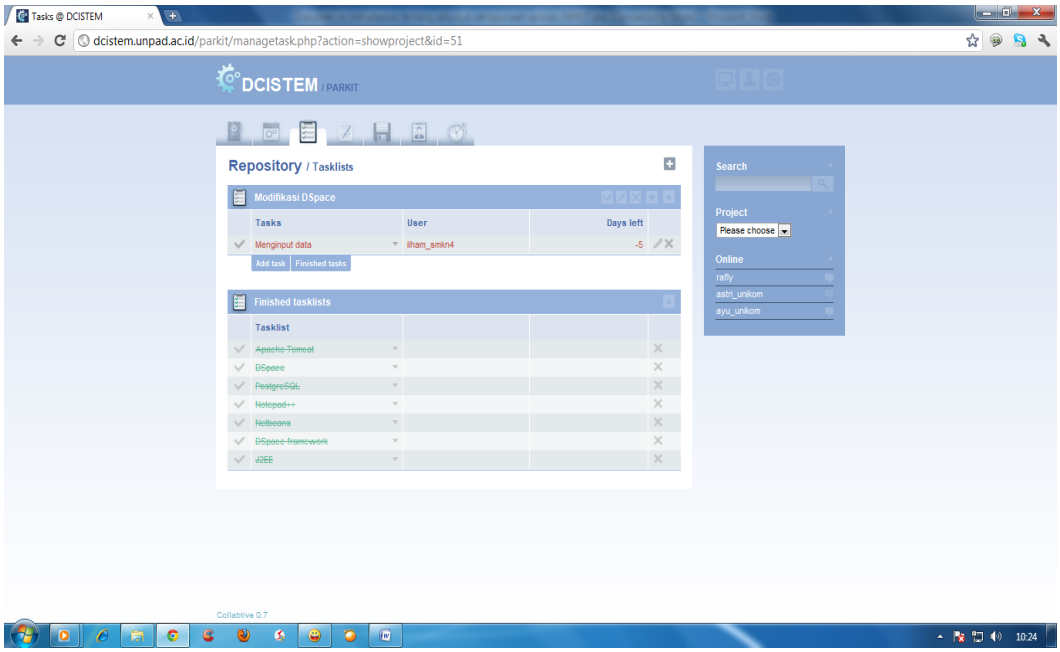
5. Jika nama, deskripsi, dan target waktu telah ditentukan, klik .

   Apabila terjadi kesalahan, pengguna masih dapat mengedit *milestone* dengan memilih *milestone* yang akan diedit kemudian klik *icon* . Pengguna juga dapat menghapus *milestone* dengan cara memilih terlebih dahulu *milestone* yang akan dihapus kemudian klik .

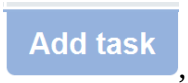
3.3.4 Membuat / Menambahkan Tasklist Pada Milestone di PARKIT

Tasklist dapat dibuat apabila projek dan *milestone* telah dibuat. Tata cara menambahkan *Tasklist* pada *milestone* adalah sebagai berikut :

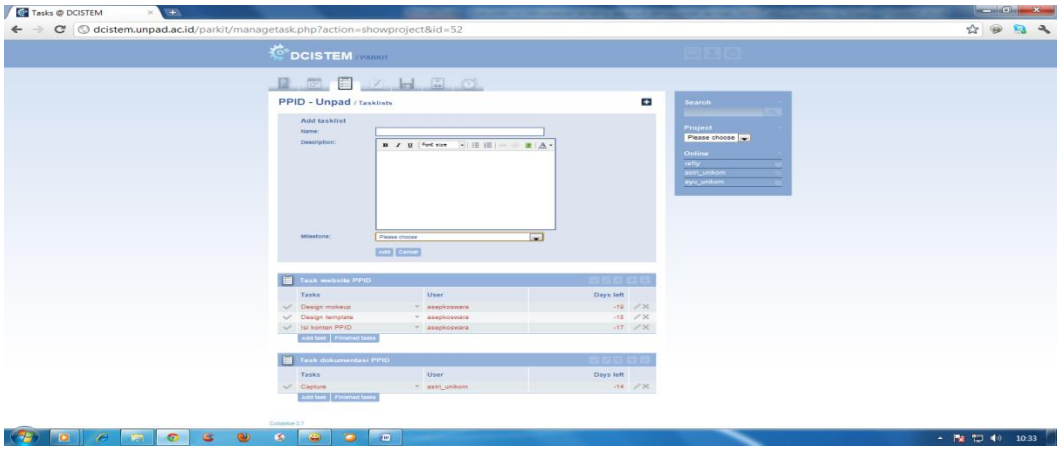
1. Klik projek yang akan ditambahkan *Tasklist*, maka akan muncul tampilan seperti *(Gambar 12) Halaman projek.*
2. Klik tabulasi *Tasklist*, maka akan muncul tampilan sebagai berikut :



Gambar 3.30 Halaman Tasklist

3. Klik  atau 

, maka akan muncul tampilan seperti berikut :



Gambar 3.31 Halaman untuk menambahkan Tasklist baru

4. Isikan *name* dengan nama *Tasklist*, *description* dengan deskripsi *Tasklist*, dan *milestone* dengan nama *milestone* yang dipilih sebagai referensi dari*Tasklist*.

5. Apabila telah selesai mengisi semua bagian yang harus diisi, klik 

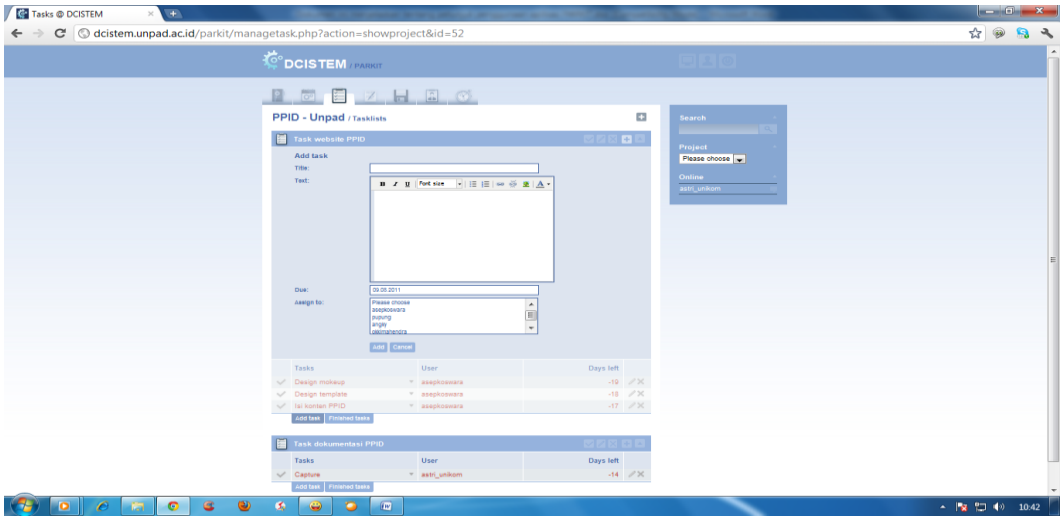
.

Apabila terjadi kesalahan, pengguna masih dapat mengedit *Tasklist* dengan memilih *Tasklist* yang akan diedit kemudian klik *icon* . Pengguna juga dapat menghapus *Tasklist* dengan cara memilih terlebih dahulu *Tasklist* yang akan dihapus kemudian klik .

3.3.5 Membuat / Menambah Subtasklist Pada Tasklist di PARKIT

Subtasklist dapat dibuat apabila projek, *milestone*, dan *tasklist* telah dibuat. Berikut adalah tata cara pembuatan *subtasklist* pada *tasklist* di sistem PARKIT.

1. Klik projek yang akan ditambahkan *Tasklist*, maka akan muncul tampilan seperti *(Gambar 12) Halaman projek*.
2. Klik tabulasi *Tasklist*, maka akan muncul tampilan seperti *(Gambar 15) Halaman Tasklist.*
3. Klik  yang berada didalam tabel, maka akan muncul tampilan sebagai berikut :



Gambar 3.32 Halaman untuk menambahkan subtasklist

4. Isikan *title* dengan judul *subtasklist*, *text* dengan penjelasan dari *subtasklist* tersebut, *due* dengan waktu yang ditargetkan berdasarkan tanggal, *assign to* dengan memilih siapa yang ditugaskan untuk mengerjakan *subtasklist* ini.

5. Apabila semua yang bagian yang perlu diisi telah diisi klik 

.

   Apabila terjadi kesalahan, pengguna masih dapat mengedit *subtasklist* dengan memilih *Tasklist* yang akan diedit kemudian klik *icon* . Pengguna juga dapat menghapus *subtasklist* dengan cara memilih terlebih dahulu *subtasklist* yang akan dihapus kemudian klik .

3.4 HASIL WAWANCARA

Untuk mendapatkan data dan informasi yang dibutuhkan penulis menggunakan salah satu metode untuk mendapatkan informasi yaitu metode wawancara. Berikut adalah hasil wawancara yang telah dilakukan penulis :

Tabel 3.32 Hasil Wawancara

No.	**Pertanyaan**	**Jawaban**
1.	Bagaimana kondisi manajemen proyek di DCISTEM – UNPAD ?	Sistem kerja DCISTEM belum terstruktur dengan baik dan sangat membutuhkan manajemen proyek yang benar untuk membantu manajemen proyek yang lebih terstruktur di DCISTEM.
2.	Mengapa PARKIT dibuat ?	Untuk mengolah lembar kerja manajemen proyek DCISTEM.
4.	Mengapa menggunakan PARKIT ?	Lebih *fleksibel*, lebih *private*
5.	Mengapa menggunakan microsoft project ?	Karena lebih familiar dan lebih lengkap.

3.5 KELEBIHAN DAN KEKURANGAN PARKIT

Ada beberapa kelebihan dan kekurangan PARKIT dengan program manajemen proyek yang telah ada. Dalam laporan ini penulis membandingkan PARKIT dengan Microsoft project.

Tabel 3.33 Kelebihan PARKIT dari Microsoft project

No	PARKIT	Microsoft Project
1	Halaman yang pertama kali muncul pada PARKIT adalah halaman *login*, dalam hal ini PARKIT dibuat dengan menerapkan pengaturan keamanan berupa halaman *login* yang melibatkan *database login user*.	Pada microsoft project dengan mudah *user* dapat masuk ke halaman microsoft project dengan masuk ke windows, klik all program, klik microsoft office, klik microsoft office project.
2	Pada PARKIT jika hendak membuat proyek tidak membutuhkan halaman baru	Pada Microsoft project, harus membuat halaman baru.
3	Hanya pengguna yang bersangkutan saja yang dapatt melihat lembar kerja mengenai suatu proyek.	Semua proyek yang ada dapat dengan mudah diakses oleh semua pengguna.
4	PARKIT berbasis web sehingga pengguna hanya membutuhkan koneksi internet untuk *share* pekerjaan yang harus dikerjakan dan saling berhubungan dengan pengguna lain. Pengguna tidak harus bertatap muka langsung untuk	Microsoft project yang menjadi perbandingan untuk penulis merupakan aplikasi berbasis desktop.

	melaporkan *progress* yang telah dicapai karena akan secara otomatis ter-*update.*	
5	Pada PARKIT setiap *task* memiliki nilai *persentase* yang telah terhitung secara otomatis, sehingga disaat pengguna telah mengklik tanda centang *progress persentase* akan terhitung secara otomatis.	Pada microsoft project progress dibuat berdasarkan perkiraan.

Tabel 3.34 Kekurangan PARKIT dari Microsoft project

1	kalender pada PARKIT mesih belum dapat di*setting* apakah tanggal tersebut merupakan hari libur atau hari besar, sehingga pembuat proyek masih harus memperhatikan tanggal dalam kefektifan jam kerja	pada microsoft project kalender sudah dapat di*setting* sehingga untuk menentukan hari dan jam untuk keefektifan bekerja sudah cukup baik karena ketika pengguna menentukan banyaknya hari yang dibutuhkan perhitungan tanggal secara otomatis menentukan sampai tanggal berapa pekerjaan dapat diselesaikan apabila fasilitas kalendernya telah di*setting* dengan baik.
2	fasilitas *reporting* pada PARKIT belum terkolaborasi dengan program yang umum digunakan untuk reporting seperti excel atau visio dan output seperti printer, secara	Microsoft project yang telah memiliki fasilitas *reporting* seperti *gantt chart* dan fasilitas konverter yang

	keseluruhan *report* masih berada dalam *database* dan hanya dapat dilihat ketika pengguna berada dalam program PARKIT.	berkolaborasi dengan excel dan visio.
3	Fase penjadwalan di PARKIT, tugas masih dapat dipilih sesuai keinginan *user*, sehingga menimbulkan ketidakdisiplinan pekerja.	Fasilitas *predecessor* benar – benar mengatur tahap – tahap yang harus dilakukan.