



### **Hak cipta dan penggunaan kembali:**

Lisensi ini mengizinkan setiap orang untuk menggubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

### **Copyright and reuse:**

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

## **BAB III**

### **PELAKSANAAN KERJA MAGANG**

#### **3.1. Kedudukan dan Koordinasi**

Program kerja magang yang dilakukan penulis dilaksanakan pada kantor cabang Karawaci, Tangerang dari PT RIGT Software Indonesia. Waktu kerja magang dilakukan selama 7 bulan 14 hari, dimana total hari kerja terhitung selama 46 hari. Jam kerja ditentukan dengan persetujuan dengan perusahaan.

Selama penulis melakukan kerja magang pada PT RIGT Software Indonesia, penulis bekerja pada departemen *development* sebagai *programmer web*. Penulis diberi tugas untuk membuat sistem *project management* yang digunakan untuk kepentingan internal perusahaan. Dalam melakukan pembuatan program, penulis mengerjakan tugas kerja magang di bawah bimbingan *project manager* Patriazka Odie Amri, S.Kom. yang ditunjuk oleh *director* sendiri Richard Giovanni, S.Kom.

#### **3.2. Tugas Yang Dilakukan**

Pada kerja magang kali ini, penulis mempunyai pekerjaan-pekerjaan khusus yang diberikan oleh direktur PT RIGT Software Indonesia. Berikut pekerjaan yang diberikan kepada penulis selama kerja magang berlangsung:

1. Pembangunan sistem *project management*
2. Fotografi 360 Ruko Golden 8, Gading Serpong

### 3.2.1. Pembangunan Sistem *Project Management*

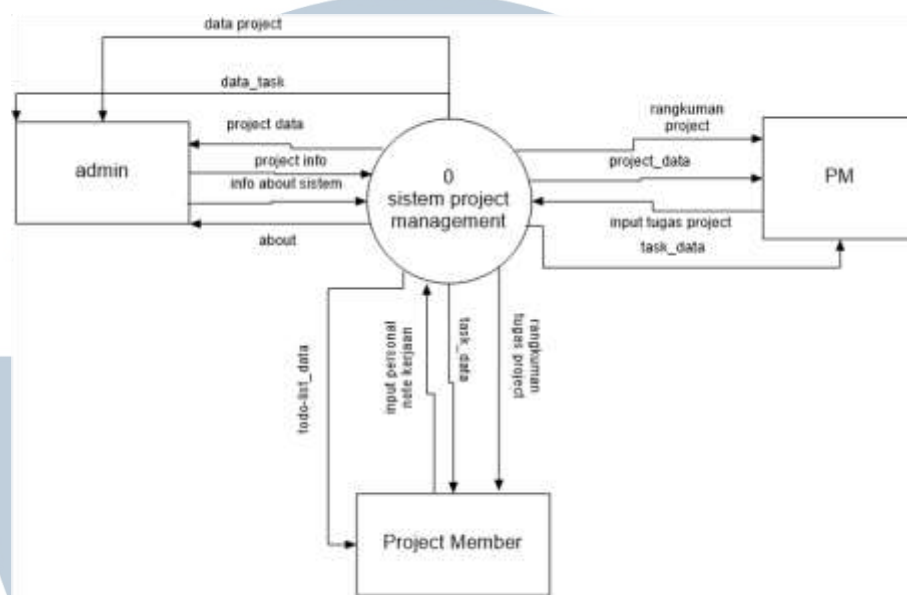
Pada hari pertama masuk, penulis dijelaskan bagaimana tugas yang diberikan selama program kerja magang ini. Penulis mendapatkan tugas dari kepala perusahaan untuk membuat sebuah sistem *project management*. Sistem *project management* ini mempunyai fungsi untuk mendokumentasi semua proyek yang didapatkan oleh perusahaan, beserta pembagian tugas dari proyek yang diberikan kepada masing-masing anggota (ditentukan secara manual).

Dalam sistem ini terdapat 2 level pengguna yaitu non-administrator dan administrator. Secara hirarkial, akun administrator (admin) adalah orang yang memegang kendali penuh terhadap sistem. Di dalam struktur organisasi perusahaan administrator dipegang oleh direktur perusahaan.

Akun non-administrator terdiri dari akun *general manager* (GM) dan akun *user*. Akun GM merupakan akun yang diperuntukkan untuk semua kepala divisi yang ada di perusahaan. Akun *user* diperuntukkan kepada setiap anggota perusahaan selain direktur dan *general manager*.

UMN

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA



Gambar 3.1 Gambar *Context Diagram System Project Management*

Pada gambar 3.1 pada diagram konteks entitas yang terlibat dalam sistem yaitu admin, PM( *project manager*), dan *project member*. Pada masing-masing eksternal entitas tersebut merupakan orang yang ada di dalam perusahaan. Seorang admin merupakan orang yang menggunakan akun administrator, seorang PM dan *project member* bisa diduduki oleh semua jenis akun. Sistem ini berfokus pada pencatatan dari *project* dan *task* yang dimiliki. Fokus yang pertama adalah *project*, data ini merupakan informasi proyek yang didapatkan perusahaan dan dicatat sesuai dengan MOU (*memorandum of understanding*) tersetujui. Data *project* diinputkan oleh seorang administrator sistem, hasil data *project* yang telah diinputkan pengguna akan ditampilkan kepada pengguna yang berposisi sebagai PM. Pada akun administrator yang posisinya sebagai PM maka data yang ditampilkan tidak hanya data proyek yang dipimpin saja. Pada akun non administrator yang posisinya sebagai PM, data yang ditampilkan hanya data *project* yang sesuai dengan kepemilikannya.

Berdasarkan perencanaan dari proyek yang telah dibuat oleh PM, maka pengguna yang posisinya sebagai *project member* akan ditugaskan kepada masing-masing anggota sesuai dengan perencanaan. Perencanaan penyelesaian *project* tersebut terdiri dari beberapa tugas yang akan menjadi data *task* pada sistem.

Pada akun administrator, data *task* menampilkan semua tugas dari proyek yang dimiliki perusahaan. Bagi orang yang merupakan *project member* akan ditampilkan semua tugas yang menjadi bagian dari proyek sesuai dengan penugasan. Pada akun GM dan akun *user* sebagai *project member*, data *task* terbagi menjadi 2 jenis yaitu *assigned task* dan *project task*. *Assigned task* merupakan tugas proyek yang harus dikerjakan oleh pengguna. *Project task* merupakan detail kegiatan proyek yang dipimpin pengguna sebagai manajer proyek, dan setiap *task* tersebut dikerjakan oleh pengguna lain. Bagi Administrator, jika pengguna berposisi sebagai *project member* selain menampilkan data *task* yang dipunyai juga dapat menampilkan semua data *task* yang ada di dalam perusahaan.

Untuk membantu pengguna yang berposisi sebagai *project member*, dalam hal pengaturan jadwal juga dapat sebagai pengingat pekerjaan, maka pengguna disediakan *to-do list*. Baik akun admin, GM, maupun *user* yang berlaku sebagai *project member*, *todo-list* dapat memperlihatkan semua data dan dapat diedit. Hasil rangkuman semua data *project* dan *task* dapat dilihat pada *dashboard* sebagai rangkuman yang dimiliki.

Selain dijelaskan pembagian tugas yang ada pada sistem, penulis diberikan desain UI dalam bentuk *excel file*. Dari rancangan *design UI (User Interface)* penulis dapat menyimpulkan, sistem dapat terbagi menjadi beberapa modul.

Modul-modul tersebut terdiri dari:

1. Modul *project*
2. Modul *task*
3. Modul *todo-list*
4. Modul *Dashboard* dan *Setting-Dashboard*
5. Modul *About*

Modul di atas merupakan modul yang dirancang oleh manajer proyek untuk penulis kerjakan dalam kerja magang ini.

Penulis melakukan perancangan *database* yang nantinya digunakan dalam membangun sistem *project management*. Dalam membangun perancangan *database* penulis membuat beberapa tabel untuk penyimpanan pada masing-masing modul. Tabel tersebut dapat dilihat dalam penjelasan berikut:

1. Tabel *project*

Merupakan tabel yang digunakan untuk mencatat data proyek yang perusahaan. Tabel 3.1 menunjukkan struktur data tabel *project* yang dibuat:

Tabel 3.1 Struktur Tabel *Project*

<b>Data Name</b>	<b>Data Type</b>	<b>Data Size</b>	<b>Data Info</b>
project_id	Int	11	<i>Primary key</i>
project_name	Varchar	150	

project_pmid	Varchar	100	<i>FK Login</i>
project_issuedby	Varchar	100	
project_priority	Int	11	
project_startdate	Varchar	30	
project_deadline	Varchar	30	
project_enddate	Varchar	30	
project_progress	Int	11	
project_projectStatus	Varchar	50	
project_description	Text		
project_status	Int	11	

## 2. Tabel *task*

Merupakan tabel yang digunakan dalam mencatat data tugas dari *project* yang dibuat. Tabel 3.2 adalah tabel dari struktur data *task* yang dibuat:

Tabel 3.2 Struktur Tabel *Task*

<b>Data Name</b>	<b>Data Type</b>	<b>Data Size</b>	<b>Data Info</b>
task_id	Int	11	<i>Primary key</i>
task_name	Varchar	150	
project_id	Int	11	<i>FK Project</i>
task_assignTo	Varchar	100	<i>FK Login</i>
task_issuedBy	Varchar	100	<i>FK Login</i>
task_priority	Int	11	
task_startDate	Varchar	30	
task_deadline	Varchar	30	
task_endDate	Varchar	30	
task_progress	Int	11	
task_taskStatus	Varchar	50	

task_description	Text		
task_status	Int	11	

Pada tabel 3.2 di atas, tabel *task* mempunyai korelasi dengan tabel *project* dalam pembagian tugas dari proyek yang dimiliki dengan merefrensikan *project\_id*. Selain korelasi dengan tabel *project*, *Task\_assignTo* dan *Task\_issuedBy* pada tabel *task* juga mempunyai korelasi dengan tabel *login*. Korelasi ditunjukkan dengan refrensi *username* pada tabel *task*. Refrensi *username* pada kolom *task\_assignTo* di tabel *task*, mengartikan data tugas tersebut ditujukan kepada siapa tugas diberikan. Refrensi *username* pada kolom *task\_issuedBy* di tabel *task*, mengartikan data tugas tersebut dibuat oleh pengguna yang memberikan tugas.

### 3. Tabel *todo-list*

Tabel *todo-list* dipakai untuk mencatat data yang dibuat oleh pengguna secara personal sebagai penjadwalan pribadi juga sebagai pengingat dalam pengerjaan *tugas* dan akhir dari *project* yang dimiliki.

Tabel 3.3 adalah tabel struktur data dari tabel *todo-list*:

Tabel 3.3 Struktur Tabel *Todo-list*

Data Name	Data Type	Data Size	Data Info
Todo_id	Int	11	<i>Primary key</i>
Todo_userid	Varchar	100	<i>FK Login</i>
Todo_taskName	Varchar	150	
Todo_priority	Int	11	
Todo_deadLine	Varchar	30	



Todo_progress	Int	11	
Todo_taskStatus	Varchar	50	
Todo_description	Text		
Todo_status	Int	11	

Pada tabel 3.3, *todo\_userid* merefrensikan *username* yang ada di tabel *login* sebagai penanda khusus kepemilikan data personal *todo-list*.

#### 4. Tabel *Status*

Tabel *status* dipakai untuk memberi urutan *ranking* status. Tabel 3.4 merupakan struktur data dari tabel *status* yang dibuat:

Tabel 3.4 Struktur Tabel *Status*

Data Name	Data Type	Data Size	Data Info
Id	Int	11	<i>Primary key</i>
Status	Varchar	50	
Rank	Int	11	

#### 5. Tabel *Login*

Tabel *login* dipakai untuk mencatat pengguna sistem dan juga posisi masing-masing pengguna. Selain mencatat posisi, tabel *login* dipakai untuk mencatat pengaturan penampilan *dashboard* masing-masing akun. Tabel

3.5 merupakan struktur data dari tabel *login* yang dibuat:

Tabel 3.5 Struktur Tabel *Login*

Data Name	Data Type	Data Size	Data Info
Username	Varchar	150	<i>Primary key</i>

Name	Varchar	50	
Password	Varchar	50	
Type	Varchar	20	
Setting	Varchar	100	
settingProjectTable	Varchar	50	
settingTaskbyProject	Varchar	50	
settingAssginedTask	Varchar	50	
Status	Int	11	

6. Tabel *about*

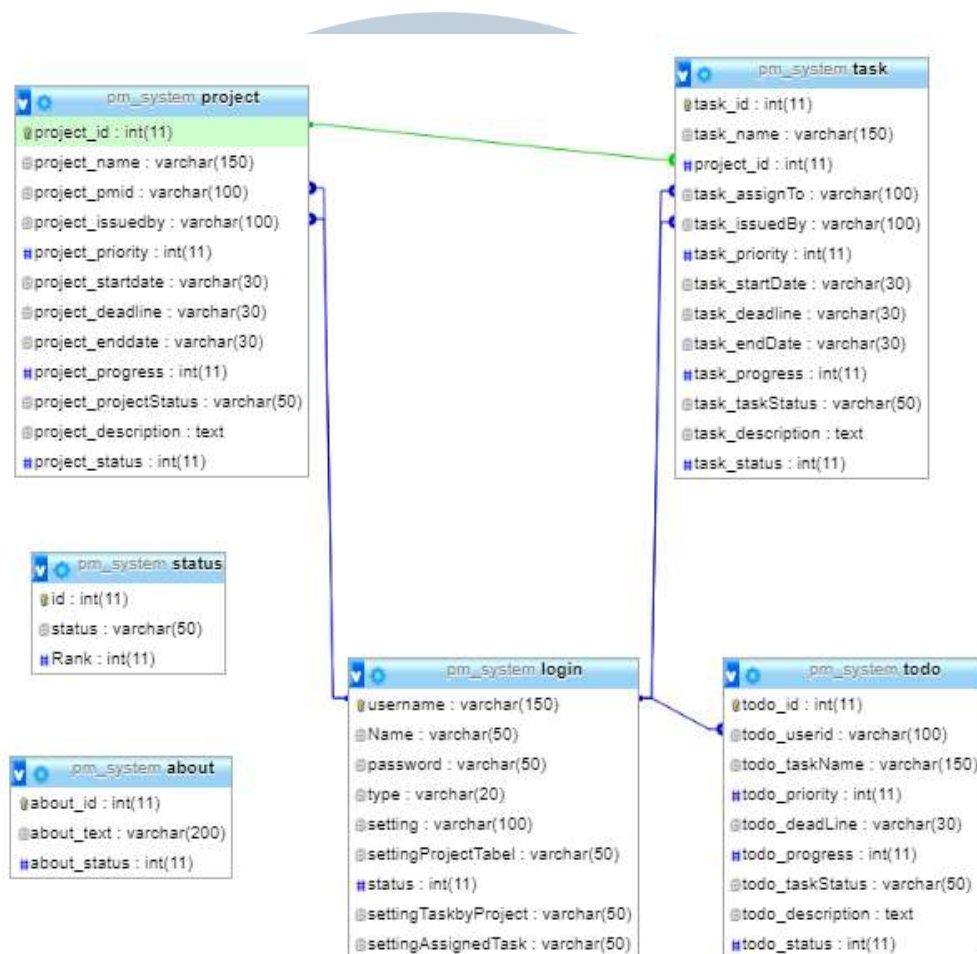
Tabel *about* dipakai untuk menampung pengaturan tulisan yang dipakai pada halaman *about* untuk mendeskripsikan sistem. Tabel 3.6 merupakan struktur data dari tabel *about* yang dibuat:

Tabel 3.6 Struktur Tabel *About*

Data Name	Data Type	Data Size	Data Info
about_id	Int	11	<i>Primary key</i>
about_status	Varchar	200	
about_status	Int	11	

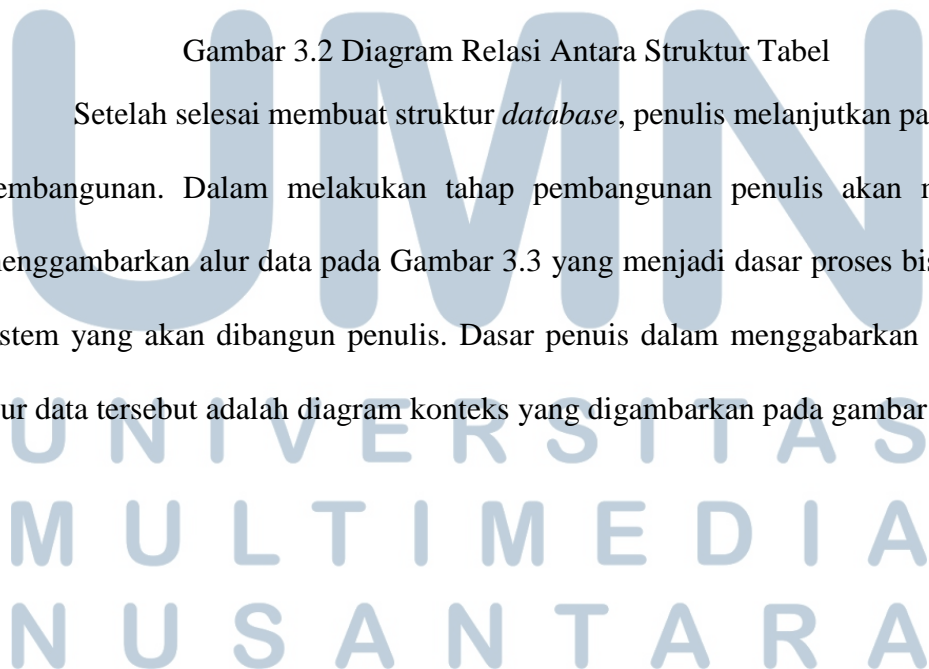
Setelah menyusun struktur tabel, didapatkan relasi antar tabel dari tabel yang dibuat sebagai. Relasi antar tabel tersebut dapat dilihat pada Gambar 3.2

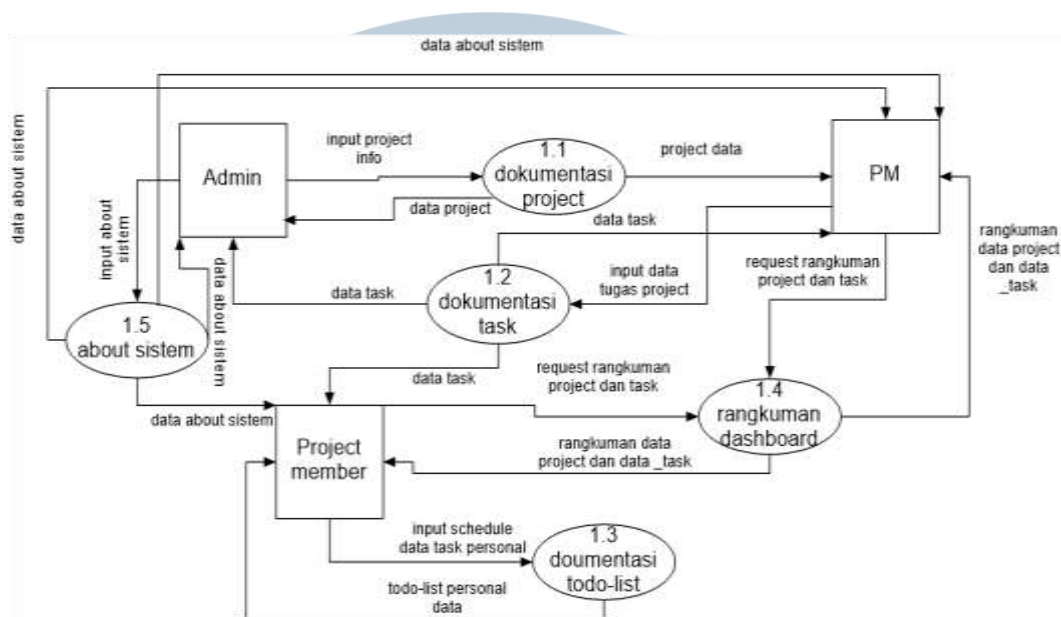
U N I V E R S I T A S  
M U L T I M E D I A  
N U S A N T A R A



Gambar 3.2 Diagram Relasi Antara Struktur Tabel

Setelah selesai membuat struktur *database*, penulis melanjutkan pada tahap pembangunan. Dalam melakukan tahap pembangunan penulis akan mencoba menggambarkan alur data pada Gambar 3.3 yang menjadi dasar proses bisnis dari sistem yang akan dibangun penulis. Dasar penulis dalam menggambarkan diagram alur data tersebut adalah diagram konteks yang digambarkan pada gambar 3.1.





Gambar 3.3 DFD Lv0 Sistem *Project Management*

Berdasarkan DFD Lv 0 gambar 3.3 maka berikut proses bisnis yang berjalan dari sistem:

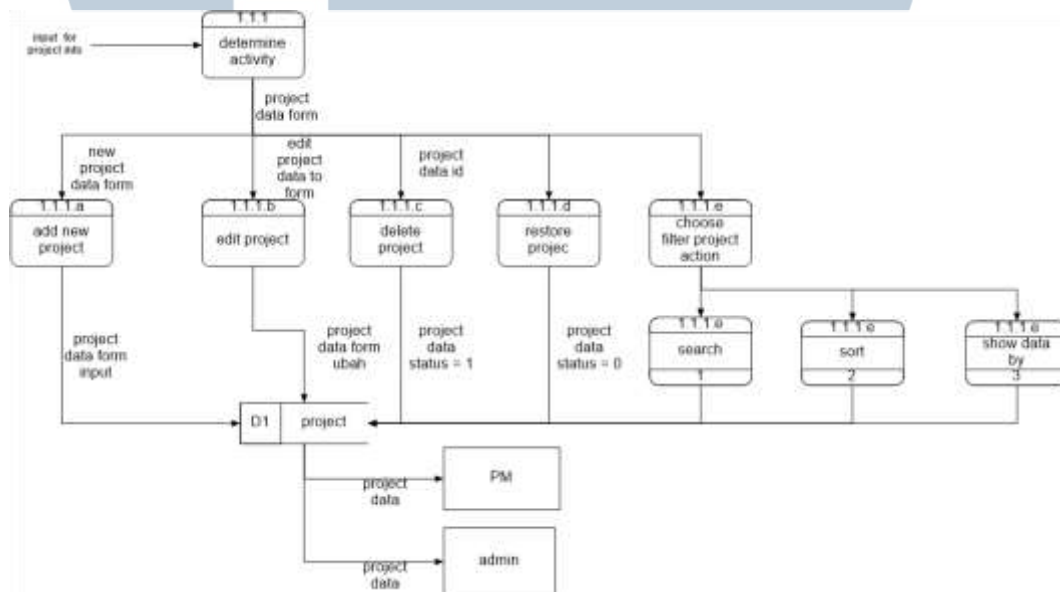
1. PT RIGT Software Indonesia berhasil mendapatkan proyek dari pihak ketiga yang telah disetujui oleh kedua belah pihak. Proyek tersebut merupakan proyek resmi dan menjadi proyek perusahaan.
2. Setelah menjadi proyek resmi perusahaan, seseorang ditunjuk untuk bertanggung jawab terhadap proyek tersebut sebagai manajer proyek. Orang yang ditunjuk tersebut membuat perencanaan terhadap proyek untuk kemudian dijalankan.
3. Setelah perencanaan dibuat oleh manajer proyek, maka proyek dilaksanakan dan penugasannya disesuaikan dengan perencanaannya. Setiap orang boleh mendapat lebih dari satu tugas. Penugasan berlaku bagi siapapun dalam organisasi.

4. Tugas-tugas proyek dikerjakan mengikuti dengan *deadline* yang telah dibuat. Setiap proyek dan tugas-tugas yang belum dikerjakan maka dianggap *not started* di dalam persen di anggap 0%. Untuk setiap proyek dan tugas yang sedang dikerjakan maka dianggap sebagai *in progress* di dalam persen dianggap 1% hingga 99 %. Untuk proyek dan tugas yang selesai dikerjakan hingga selesai dianggap sebagai *complete*, dalam persen dianggap 100 %. Status di atas berubah secara otomatis jika progres telah terisi.
5. Setiap proyek dan tugas-tugas yang tidak berhasil selesai sesuai *deadline* dianggap menjadi *overdue*. Jika tidak bisa diselesaikan ataupun bermasalah, dan ada kesepakatan terhadap proyek maupun tugas untuk ditutup, maka diganti secara manual menjadi status *closed*.
6. Untuk membantu pengguna dalam mengatur pekerjaannya maka disediakan saran untuk mengatur jadwal personal dalam mengerjakan tugas.
7. Kegiatan proyek dan tugas di atas beserta progress dapat dilihat sebagai rangkuman bagi semua pengguna.

Setelah mencoba mengerti alur penulis mulai membuat program berdasarkan alur dari sistem yang telah dijelaskan. Pembuatan program dilakukan oleh penulis menggunakan bahasa PHP (*Hypertext Processor*), HTML (*Hyper Text Markup Language*), JavaScript sebagai sarana untuk membangun sistem. Berikut dibahas alur sistem yang dibangun penulis per modul:

## 1. Modul Project

Seperti yang telah dijelaskan diatas modul *project* merupakan fokus yang paling utama pada sistem ini. Fungsi dari modul ini merupakan pencatatan data proyek berdasarkan MOU (*Memorandum of Understanding*) yang telah disepakati antara pihak pemberi proyek dan juga perusahaan. Pencatatan rangkuman MOU tentang proyek tersebut dicatat ke dalam sistem. Untuk menjelaskan alur sistem yang berikut diagram alur data *level 1* dari modul *project* pada gambar 3.4:



Gambar 3.4 DFD level 1 Project

Pada akun administrator (*admin*) pengguna dapat melihat semua data proyek yang dimiliki perusahaan baik yang sudah selesai, sedang berjalan, tertutup, telat, belum dimulai. Pada akun non administrator (*GM* dan *user*) pengguna hanya dapat melihat data proyek yang ditugaskan kepada pengguna sebagai *project manager*.

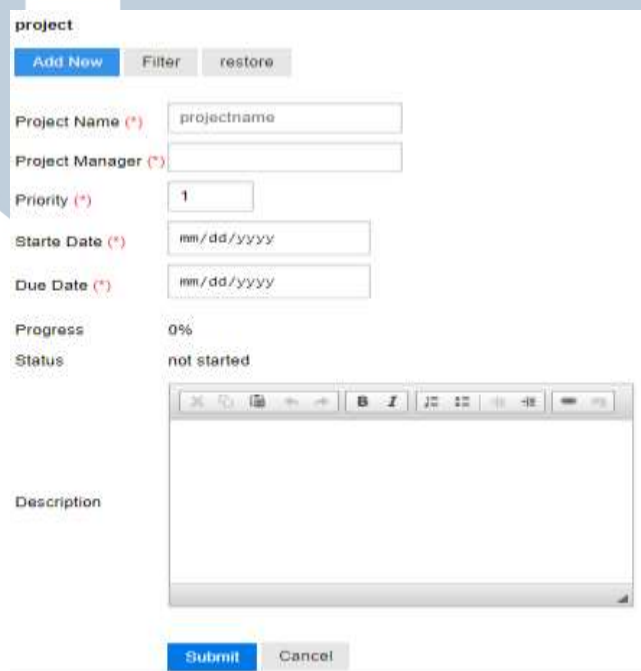
Penambahan sebuah data proyek ke dalam sistem didasarkan pada dokumen MOU yang sudah disepakati. *Project manager* yang berakun non-administrator atau administrator dapat menginputkan data ke dalam modul *project*. Tidak semua akun non-administrator dapat menambahkan data hanya (*general manager*) GM yang dapat menambahkan proyek baru.

Untuk memulai penginputan data proyek pengguna dapat menekan *add new* untuk membuka *form* penambahan. *Form* penambahan proyek terdiri dari inputan sebagai berikut:

1. *Project Name*: inputan ini digunakan untuk menambahkan nama proyek.
2. *Project Manager*: inputan ini digunakan untuk menambahkan info kepada siapa proyek akan diserahkan untuk diawasi.
3. *Priority*: inputan ini digunakan untuk menentukan kepentingan dari proyek tersebut.
4. *Start Date*: inputan ini digunakan untuk mengisi tanggal proyek dimulai.
5. *Due Date*: inputan ini digunakan untuk mengisi kapan proyek berakhir
6. *Progress*: digunakan untuk mengisi kemajuan dari progress pada awal penambahan maka form memberikan nilai 0. Nilai progress pada *project* berubah berdasarkan dari penyelesaian dari tugas yang merupakan bagian dari proyek tersebut.

7. *Status*: digunakan untuk mengisi status dari proyek, pada awal penambahan diberikan nilai *not started*.
8. *Description*: digunakan untuk memberikan deskripsi.

*Field required* atau *not null* perlu diisi sehingga permintaan penambahan dapat diproses. Jika *field required* atau *not null* ada yang tidak lengkap maka penambahan tidak dapat diproses. Gambaran *form* penambahan data dapat dilihat pada gambar 3.5



Gambar 3.5 *Form* Penambahan

Untuk menambahkan hasil *input* dari *form* penambahan data maka pengguna melakukan *submit*. Hasil penginputan yang sudah disubmit, kembali dicek bahwa data yang perlu diisi telah lengkap atau tidak ketika melakukan penambahan. Jika lengkap penambahan data berhasil, jika tidak penambahan gagal dilakukan. Hasil penambahan ditampilkan di dalam tabel proyek.



Jika ada data yang perlu diubah maka pengguna dapat menekan *edit* pada data yang ingin diubah pada tabel untuk memunculkan *form* perubahan, *form* perubahan mempunyai *field* yang sama dengan *form* penambahan. Gambaran *form* penambahan dapat dilihat pada gambar 3.6. *Field* yang ada pada *form* perubahan sudah terisi sesuai dengan data yang ingin diubah. Untuk melakukan penambahan maka pengguna menekan *submit*.

project

Add New Filter restore

Project Name (\*) abccdd

Project Manager (\*) Patriazka Odie

Priority (\*) 4

Start Date (\*) 05/31/2017

Due Date (\*) 06/01/2017

End Date 06/13/2017

Progress (\*) 100 %

status (\*) completed

Description (\*)

baoddasdkjsalifsdca

Submit Cancel

(\*) : required/not null

Gambar 3.6 *Form Edit Project*

Sama seperti *form* penambahan, *field* yang sifatnya *required* atau *not null* tidak boleh dikosongkan sehingga bisa diproses. Hasil perubahan yang dilakukan dapat dilihat di dalam tabel *proyek*.

Untuk melakukan penghapusan pada data yang dituju maka pengguna menekan *delete*. Penghapusan data dapat dilakukan oleh administrator yang berakun admin dan juga akun GM bagi non-admin. Setelah *delete* ditekan maka muncul dialog konfirmasi tentang penghapusan dilakukan atau tidak, jika pengguna menekan *ok* pada saat penghapusan dilakukan maka dicek apakah data proyek tersebut mempunyai *task* atau tidak. Jika ada *task* yang berhubungan maka penghapusan digagalkan jika tidak maka penghapusan dilakukan. Jika ditekan *cancel* maka penghapusan tidak dilakukan oleh pengguna. Data yang terhapus dapat dikembalikan dengan membuka tabel *resotre*.

No	projectname	Project Manager	Issued By	Priority	Start Date	Due Date	End Date	Progress	Status	Description	Related Task
1	abc	Malvin Michael	Richard	5	2017-06-02	2017-09-05	2017-06-02	90 %	overdue	adodjikasjopadpa #	0

Gambar 3.7 Tabel *Restore* Proyek

Gambar 3.7, merupakan gambar tabel *restore* tempat penampungan data yang terhapus. Untuk mengembalikan data maka pengguna menekan *restore* sehingga data yang sudah terhapus dapat kembali ke dalam tabel *project*. Jika ingin menghapus permanen maka pengguna dapat menekan *permanent delete* pada tabel *restore*. Data yang sudah terhapus permanen tidak muncul lagi di dalam sistem.

Ketika sudah selesai melakukan pengembalian atau penghapusan secara permanen maka pengguna dapat kembali ke dalam tabel proyek dengan menekan tombol *return*.

Untuk membantu pengguna mencari data, maka disediakan fasilitas *filter* yang digunakan menyaring data. Terdapat 3 jenis *filter* yang disediakan dalam modul ini yaitu *searching data*, *sorting tabel*, *show data by*.

Filter *searching data* digunakan untuk melakukan pencarian data berdasarkan kategori yang ingin dicari. Untuk melakukan pencarian data, pengguna dapat membuka *form* pencarian dengan menekan tombol *search data* pada pilihan. Gambar 3.8 adalah form pencarian data yang ditampilkan untuk mencari data *project* yang manajer proyeknya Michael Malvin. Pengguna perlu mengisi kategori dan *keyword* sesuai dengan kategori pencarian yang ingin dilakukan. Kedua *field* tersebut perlu diisi sehingga pencarian dapat di proses, jika ada yang tak lengkap pengguna tidak dapat melakukan pencarian. Sesudah mengisi kedua *field*, untuk melakukan pencarian pengguna akan menekan tombol *search*. Hasilnya ditampilkan pada tabel proyek atau tabel *restore* tergantung dimana *filter* dilakukan, jika sudah mengatur *filter sorting data* dan *show data by* maka penampilan hasil pada tabel menyesuaikan.

No	projectname	Project Manager	Issued By	Priority	Start Date	Due Date	End Date	Progress	Status	Description	Related Task
1	login	Malvin Michael	Malvin Michael	5	2017-06-01	2017-06-07	2017-06-08	100 %	complete	testing login sistem project management	3
2	abc	Malvin Michael	Richard	5	2017-06-02	2017-06-05	2017-06-02	90 %	overdue	actdjskicdsjdajca	0

Gambar 3.8 Hasil *Filter Search Data*

Filter *show data by* digunakan untuk menampilkan tabel sesuai dengan kategori yang hanya ingin ditampilkan oleh pengguna. Untuk membuka *form show data by* pengguna dapat menekan tombol *show data by* pada pilihan *filter*, *form show data by* dapat dilihat pada Gambar 3.9. Ketika pengguna sudah membuka *form show data by*, pengguna memilih kategori mana saja yang ingin ditampilkan oleh pengguna didalam tabel proyek. Ketika ingin mengosongkan semua kategori pengguna dapat menekan *unchecked all*, jika ingin mereset penampilan sehingga semua kateogri dipilih kembali maka dapat ditekan *checked all*. Ketika kategori sudah dipilih sesuai dengan keinginan pengguna, maka pengguna dapat menekan *show*. Tabel proyek menampilkan sesuai dengan kategori yang ditampilkan sebagai hasil. Jika telah melakukan *filter search data* dan *sorting table* maka tabel proyek hasil *show data by* akan menyesuaikan.

project

Add New **Filter** restore

**Show Data By** Search Data Sort Table Clear Filter

Show By Column

- Project Name
- Project Manager
- Issued By
- Priority
- Start Date
- Due Date
- End Date
- Progress
- Project Status
- Description
- Related Task

Checked All  Unchecked All

**Show** Cancel

No	projectname	Project Manager	Priority	Start Date	Due Date	End Date	Progress	
1	login	Malvin Michael	5	2017-06-01	2017-06-07	2017-06-06	100 %	Edit   Delete
2	abc	Malvin Michael	5	2017-06-02	2017-06-05	2017-06-02	90 %	Edit   Delete

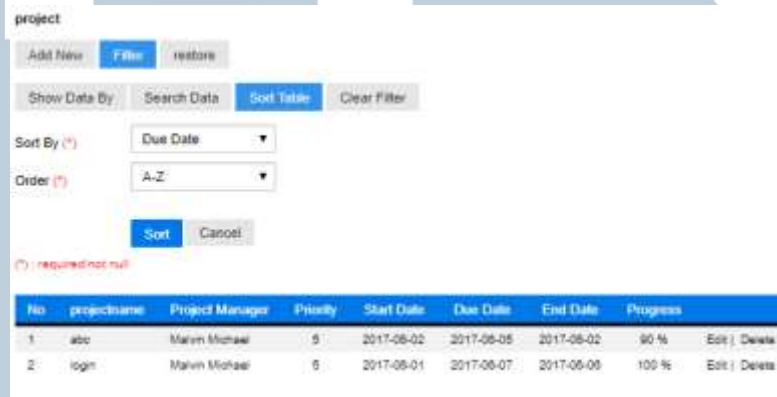
NUSANTARA

Gambar 3.9 Hasil *Filter Show Data by*

*Filter sorting* digunakan untuk melakukan pengurutan terhadap data *proyek*. Untuk melakukan pengurutan dan membuka *form sorting*, maka pengguna dapat menekan *sort table* pada pilihan *filter*. Setelah pengguna menekan tombol *sort table* maka ditampilkan *form* pengurutan seperti pada Gambar 3.10. Pengguna harus mengisi kedua *field* yaitu kategori yang ingin diurutkan dan jenis pengurutan yang ingin dilakukan. Pada *field* jenis pengurutan, terdapat dua jenis yaitu A-Z yang merupakan pengurutan *ascending* dan Z-A yang merupakan pengurutan *descending*. Pengurutan *ascending* mengurutkan data dari terkecil ke terbesar, pengurutan *descending* mengurutkan data dari terbesar ke terkecil. Sesudah mengisi kedua *field* maka pengguna menekan tombol *sorting* untuk memproses pengurutan. Pengurutan menyesuaikan jenis data pada masing-masing kategori. Hasilnya dapat dilihat pada tabel proyek ataupun tabel *restore* sesuai dengan jenis data kategori pengurutan yang dilakukan.

Jika berdasarkan angka maka pengurutan *ascending* (A-Z) dilakukan dari angka terkecil ke besar dan *descending* pengurutan dilakukan angka berdasarkan angka dari terbesar ke kecil. Jika pengurutan huruf maka pengurutan *ascending* dilakukan dari huruf terkecil ke besar dan *descending* huruf terurutkan berdasarkan angka dari terbesar ke kecil. Jika tanggal pengurutan *ascending* dilakukan dari tanggal terkecil ke besar dan *descending* dilakukan tanggal terurutkan berdasarkan angka dari terbesar ke kecil. Pengurutan status diurutkan berdasarkan setting *ranking* yang disepakati oleh perusahaan. Pengurutan *ascending* dilakukan dari *ranking* terkecil ke besar dan *descending* dilakukan pengurutan berdasarkan *ranking*

dari terbesar ke kecil. Jika sudah dilakukan *filter searching* dan *show data by* penampilan tabel menyesuaikan.



Gambar 3.10 Hasil *Filter Sorting*

Ketiga *filter* diatas tidak langsung menghilang ketika pengguna melakukan *filter* yang lainnya, karena setiap *filter* disimpan sementara. Sesudah *filter* dilakukan maka pengguna dapat menghapus *filter* dengan menekan tombol *clear filter*.

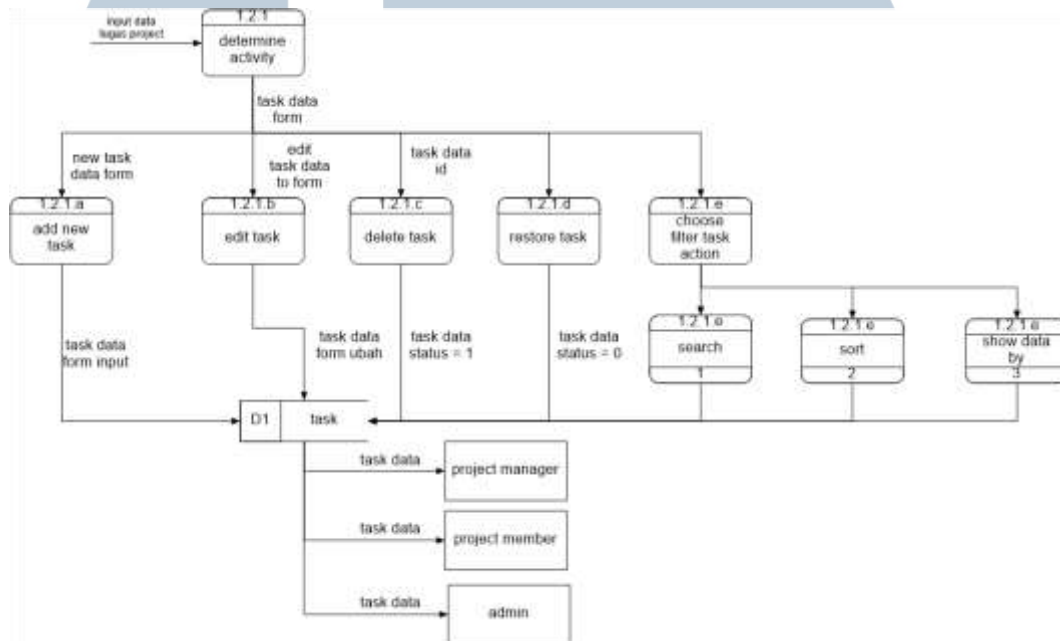
## 2. Modul *Task*

Modul *task* merupakan pencatatan yang digunakan untuk mencatat detail pekerjaan yang dibuat oleh *project manager* untuk menyelesaikan proyek yang diberikan. Modul *task* dapat dibuat setelah ada data *project*, tanpa ada *project* maka *task* tersebut tidak bisa dibuat. Karna perbedaan fungsi penulis memisahkan modul *task* pada administrator dan non administrator.

### A. Modul *Task* Administrator

Pada modul *task* administrator, penulis menamakan modul *task* bagi administrator sebagai *task-all*. Sesuai dengan fungsinya modul *task-all*

menampilkan semua *task* yang dipunyai anggota perusahaan, dan admin berhak melakukan manipulasi data terhadap data tersebut. Gambar 3.11 *DFD level 1* untuk menjelaskan aliran data modul *task-all*



Gambar 3.11 Gambar *DFD level 1* Modul *Task-All*

Admin dapat melihat semua *task* yang ada pada perusahaan milik masing-masing anggota perusahaannya. Ketika ingin menambahkan *task* baru, admin dapat membuka *form* penambahan dapat dilakukan dengan memilih *add new*. *Form* penambahan beserta *field* yang ada pada *form* dapat dilihat pada gambar 3.12. *Field* yang ada pada *form* penambahan terdiri dari:

1. *Task Name*: inputan ini digunakan untuk menambahkan nama tugas proyek.
2. *Project Name*: merupakan inputan yang diisi untuk menyatakan bahwa tugas ini merupakan bagian dari proyek tertentu.

3. *Assign to*: inputan ini digunakan untuk menambahkan info kepada siapa tugas ini ditugaskan.
4. *Priority*: inputan ini digunakan untuk menentukan kepentingan dari tugas tersebut.
5. *Start Date*: inputan ini digunakan untuk mengisi tanggal tugas dimulai.
6. *Due Date*: inputan ini digunakan untuk mengisi kapan tugas berakhir.
7. *Progress*: digunakan untuk mengisi kemajuan dari progress pada awal penambahan maka form memberikan nilai 0. Nilai progress pada *project* berubah berdasarkan dari penyelesaian dari tugas yang merupakan bagian dari proyek tersebut.
8. *Status*: digunakan untuk mengisi status dari tugas, pada awal penambahan diberikan nilai *not started*.
9. *Description*: digunakan untuk memberikan deskripsi.

Proses penambahan data yang berjalan seperti proses penambahan pada modul *project*, dimana pengguna mengisi data yang perlu diisi pada *field* yang disediakan, setidaknya *Field required* atau *not null* sehingga penambahan dapat diproses. Untuk memproses penambahan admin harus menekan *submit*, ketika dilakukan pemrosesan terlebih dahulu di cek bahwa apakah *field* yang *required* atau *nonnull* tidak kosong. Jika tidak kosong penambahan dilanjutkan, jika ada yang tidak lengkap maka penambahan dibatalkan. Jika berhasil hasilnya tertampil di tabel *task*.



task

Add New Filter restore

task Name (\*)

Project Name (\*)

assign To (\*)

priority (\*)

start date (\*)

due date (\*)

progress  %

status

Description

Submit Cancel

(\*): not null/required

Gambar 3.12 Task-All *Form* Penambahan

Jika ada perubahan terhadap data *task-all* yang ingin diubah, pengguna dapat menekan *edit* untuk memunculkan *form* perubahan. Ketika *form* perubahan data muncul seperti pada gambar 3.13, *field form* perubahan yang tersedia sudah terisi dengan data. *Field* yang disediakan sama seperti pada *form penambahan* hanya saja ada tambahan untuk *end date* yang diisi pada saat tugas selesai. Admin melakukan perubahan pada *field* yang ingin diubah. Perubahan diproses ketika setidaknya *field required* atau *not null* sudah terisi, untuk melakukan perubahan pengguna menekan *submit*. Jika berhasil maka perubahan tertampil pada tabel *task-all*.

task

Add New Filter restore

task Name (\*) cek add task

Project Name (\*) cek task

Assign To (\*) tatatatititi

priority (\*) 5

start date (\*) 06/02/2017

due date (\*) 06/02/2017

end date 06/02/2017

progress (\*) 100 %

status (\*) completed

Description

Submit Cancel

(\*)not null/required

Gambar 3.13 Form Edit Task-All

Kegiatan penghapusan data pada modul ini dilakukan seperti pada modul lainnya. Ketika tombol *delete* ditekan maka muncul konfirmasi penghapusan, jika ditekan *ok* pengguna menghapus data dari tabel *task-all*. Jika ditekan *cancel* penghapusan tidak dilakukan. Setiap data yang terhapus masuk kedalam tabel *restore* sehingga dapat dikembalikan seperti pada gambar 3.14.

task

Filter Return

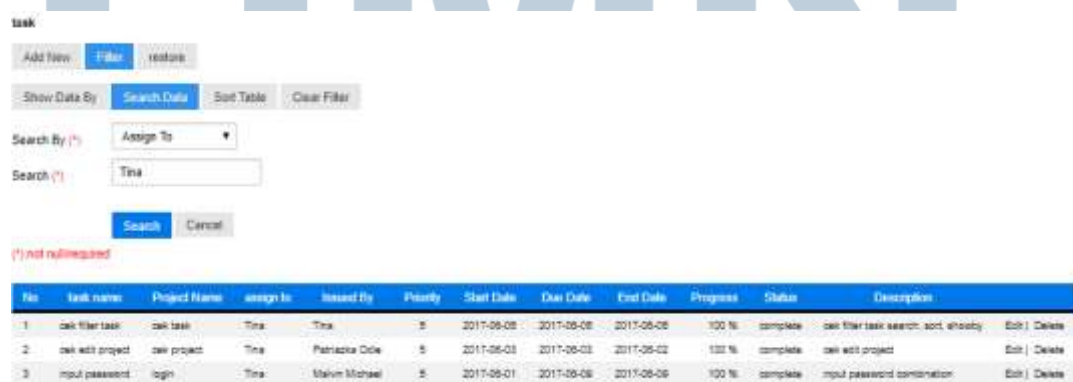
No	task name	Project Name	assign to	Issued By	Priority	Start Date	Due Date	End Date	Progress	Status	Description	
1	login	abccdd	Tina	Patriazka Dole	4	2017-06-02	2017-06-03	2017-06-02	100 %	complete	login system	Restore   Permanent Delete

Gambar 3.14 Gambar Tabel Restore Task-All

Untuk mengembalikan data yang terhapus dari tabel *restore*, pengguna dapat menekan *restore*. Jika ingin menghapus permanen pengguna dapat menekan *permanent delete*. Data yang telah dikembalikan muncul kembali pada tabel *task-all*.

Fasilitas *filter* digunakan untuk mempermudah pengguna dalam menemukan data dan menampilkan data. Sama seperti modul lainnya terdapat 3 jenis kegiatan *filter* yaitu *filter search data*, *sort data*, *show data by*. Penggunaan *filter* pada modul ini digunakan seperti modul *filter* lainnya.

Ketika *filter* yang dipilih adalah *filter search*, maka muncul *form* pencarian seperti gambar 3.15. Ketika *form* pencarian muncul maka dilakukan pencarian berdasarkan *field* kategori dan *keyword* yang diinputkan untuk pencarian. Untuk memulai pencarian maka pengguna menekan *submit*, pencarian diproses ketika kedua *field* sudah terisi lengkap. Hasilnya akan ditampilkan di dalam tabel dan menyesuaikan dengan pengaturan *filter* yang sudah dilakukan *sorting* dan *show data by*.



Gambar 3.15 Filter Searching Data

Ketika *filter* yang dipilih *show data by* maka dilakukan penampilan data berdasarkan kategori yang dipilih. *Filter* dilakukan dengan membuka *form show by* dan memilih kategori yang ingin ditampilkan seperti pada gambar 3.16. Untuk mengosongkan pilihan dapat ditekan *unchecked all*, dan untuk mereset penampilan ditekan *checked all*. Setelah selesai mengatur administrator menekan *show*, hasilnya ditampilkan tabel berdasarkan kategori yang dipilih. Jika pengaturan *filter* lain sudah diatur penampilan menyesuaikan.

The screenshot shows a 'project task' interface with a 'Filter' button. Under 'Show Data By', there are several checkboxes for filtering tasks. The 'Show By Column' section includes checkboxes for 'task name', 'project name', 'assign to', 'issued by', 'priority', 'start date', 'due date', 'end date', 'progress', 'project status', 'description', 'Checked All', and 'Unchecked All'. A 'Show' button is visible below the filter options.

No	task name	Project Name	assign to	Priority	Start Date	End Date	Progress	
1	input username	login	Patriazka Odie	5	2017-06-02	2017-06-02	100 %	Edit   Delete
2	input password	login	Tina	5	2017-06-01	2017-06-09	100 %	Edit   Delete
3	login validate to database	login	Patriazka Odie	5	2017-06-03	2017-06-02	100 %	Edit   Delete

Gambar 3.16 *Filter Show Data by Task All*

Ketika *filter* yang dipilih *sort table*, *filter* yang dilakukan adalah pengurutan data berdasarkan kategori dan jenis pengurutan data. Untuk melakukan pengurutan data seperti *filter* lainnya, admin membuka *form* pengurutan dan mengisi *form* tersebut yang terdiri dari kategori data dan jenis pengurutan seperti pada gambar 3.17. Ada dua jenis pengurutan A-Z pengurutan *ascending*, Z-A pengurutan *descending*. Kedua *field* diatas harus diisi untuk dapat memproses pengurutan jika ada yang kosong maka pengurutan tidak dilakukan. Untuk memproses pengurutan,

maka pengguna menekan *sort*. Hasil pengurutan ditampilkan pada tabel, jika sudah mengatur *filter* lain penampilan menyesuaikan.

task

Add New Filter restore

Show Data By Search Data Sort Table Clear Filter

Sort By (\*) task name ▼

Order (\*) A-Z ▼

Sort Cancel

(\*):not null/required

No	task name	Project Name	assign to	Priority	Start Date	End Date	Progress	
1	oek add task	oek task	tatatatititi	5	2017-06-02	2017-06-02	100 %	Edit   Delete
2	oek delete and restore task	oek task	tataketata	5	2017-06-06	2017-06-06	100 %	Edit   Delete
3	oek filter task	oek task	Tina	5	2017-06-05	2017-06-05	100 %	Edit   Delete

Gambar 3.17 Filter Sorting Data

Ketiga *filter* diatas disimpan sementara sehingga pengaturan yang telah diatur tidak hilang ketika dilakukan *filter* lainnya. Untuk menghapus semua pengaturan *filter* yang sudah dilakuka, admin menekan *clear filter*.

## B. Modul *Task* Non-administrator

Pada modul *task* non-administrator diperuntukkan untuk akun non-administrator yang terdiri dari akun *General Manager* (GM) dan *user*, modul task terbagi menjadi *assigned task* dan *project task*.

*Project task* merupakan modul *task* yang digunakan melakukan pencatatan *task* dari proyek yang dipimpin oleh pengguna. Alur sistem yang berjalan pada *project task* mempunyai alur yang sama dengan *task-all* bagi administrator.

Jika ingin melakukan penambahan, seperti *task-all* yang telah dijelaskan pengguna dapat membuka *form* penambahan dengan menekan *add new*. Setelah *form* penambahan terbuka seperti pada gambar 3.18, maka pengguna mengisi data *task* yang perlu ditambahkan, setidaknya *field* yang sifatnya *required* atau *not null* tidak boleh kosong. *Field* yang ada pada *form* penambahan sama seperti pada *form* penambahan *task-all*.

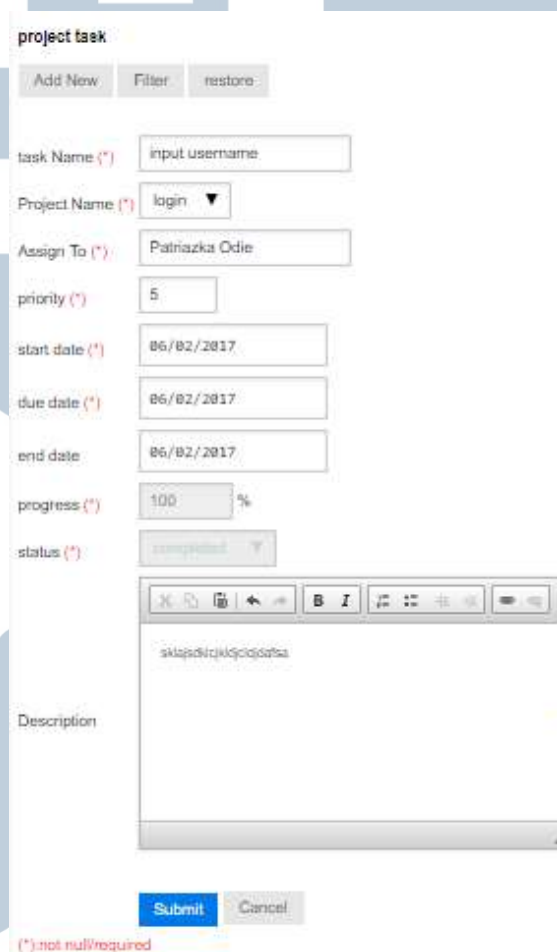
Untuk memproses hasil inputan maka pengguna menekan tombol *submit*, jika *required* atau *not null field* ada yang kosong maka penambahan tidak diproses. Hasil penambahan muncul di dalam tabel.

The image shows a web form titled "project task". At the top, there are three buttons: "Add New" (highlighted in blue), "Filter", and "restore". Below these are several input fields, each with a red asterisk (\*) indicating it is required. The fields are: "task Name" (text input), "Project Name" (dropdown menu with "Please Choose..." selected), "assign To" (text input), "priority" (text input with "1" entered), "start date" (text input with "mm/dd/yyyy" placeholder), "due date" (text input with "mm/dd/yyyy" placeholder), "progress" (text input with "0" and a "%" sign), and "status" (dropdown menu with "not started" selected). Below these fields is a large text area for "Description". At the bottom of the form are two buttons: "Submit" (highlighted in blue) and "Cancel". A legend at the bottom left states "(\*) not null/required".

Gambar 3.18 Form Penambahan Project Task

Jika ingin melakukan perubahan, *form* perubahan dapat dimunculkan dengan menekan tombol *edit* pada data tabel. Sesuai dengan gambar 3.19, Pada *form* perubahan data setiap *field* sudah terisi sesuai dengan data yang ingin diubah,

pada data yang ingin diperbarui maka pengguna mengubah data sesuai dengan keinginan. Sesudah dilakukan perubahan untuk memproses pengguna menekan *submit*, jika *field required* atau *not null* lengkap permintaan pembaruan dilakukan. Jika tidak lengkap maka permintaan pembaruan digagalkan. Hasil perubahan tertampil pada tabel *project task*.



The image shows a web form titled "project task". At the top, there are three buttons: "Add New", "Filter", and "restore". Below these are several input fields, each with a red asterisk indicating it is required. The fields are: "task Name (\*)" with the value "input username"; "Project Name (\*)" with a dropdown menu showing "login"; "Assign To (\*)" with the value "Patriazka Odie"; "priority (\*)" with the value "5"; "start date (\*)" with the value "06/02/2017"; "due date (\*)" with the value "06/02/2017"; "end date" with the value "06/02/2017"; "progress (\*)" with a slider set to "100 %"; and "status (\*)" with a dropdown menu showing "completed". Below the status field is a rich text editor for the "Description" field, containing the text "sktjpdqkjdqjcdqsa". At the bottom of the form are two buttons: "Submit" (in blue) and "Cancel". A red note at the bottom left says "(\*) not null/required".

Gambar 3.19 Gambar *Form* Pengubahan *Data*

Jika ingin melakukan penghapusan terhadap data *project task*, maka pengguna bisa menekan *delete* berdasarkan data yang ada pada tabel. Ketika *delete* ditekan maka pengguna mendapatkan konfirmasi dari sistem. Jika pengguna menekan *ok* pada konfirmasi penghapusan dilakukan, jika pengguna memilih

*cancel* maka penghapusan tidak dilakukan. Data yang dihapus dapat dikembalikan oleh pengguna dengan membuka tabel *restore* yang bisa dilihat pada gambar 3.20. Ketika data ingin dikembalikan ke dalam tabel maka pengguna dapat menekan *restore*, jika data ingin dihapus secara permanen maka dapat ditekan *permanen delete*. Sesudah melakukan kegiatan pengembalian maka pengguna dapat kembali ke dalam tabel *project task* menggunakan tombol *return*.



The screenshot shows a web application interface for 'project task'. At the top, there are two buttons: 'Filter' and 'Return'. Below them is a table with the following columns: No, task name, Project Name, assign To, Issued By, Priority, Start Date, Due Date, End Date, Progress, Status, and Description. The table contains one row of data:

No	task name	Project Name	assign To	Issued By	Priority	Start Date	Due Date	End Date	Progress	Status	Description
1	input username	login	Patrizka Odie	Malvin Michael	5	2017-08-02	2017-08-02	2017-08-02	100 %	complete	silajskioikidjoidjaifa Restore   Permanent Delete

Gambar 3.20 Tabel *Restore Project Task*

Fasilitas *filter* disediakan bagi pengguna dalam mencari dan menyaring data, terdapat tiga kegiatan *filter* yaitu *search data*, *show data by*, *sorting table*. *Filter* diatas mempunyai prinsip kerja yang sama dengan *filter* modul lainnya pada sistem. Setiap *filter* diatas disimpan sementara jika sudah melakukan salah satu *filter*, sehingga ketika melakukan *filter* lainnya pengguna tidak kehilangan pengaturan *filter* yang telah dilakukan. Sesudah melakukan *filter*, pengguna bisa menghapus pengaturan dengan menekan *clear filter*.

Modul *task* yang kedua pada akun non-administrator adalah *assigned task* sebagai sarana untuk menampilkan semua *task* dari proyek perlu dikerjakan oleh pengguna. Pada modul *assigned task* ini tidak digunakan sebagai sarana pencatatan *task* melainkan sebagai tempat bagi pengguna untuk memberi *progress* pekerjaan terhadap *task* yang dimiliki.

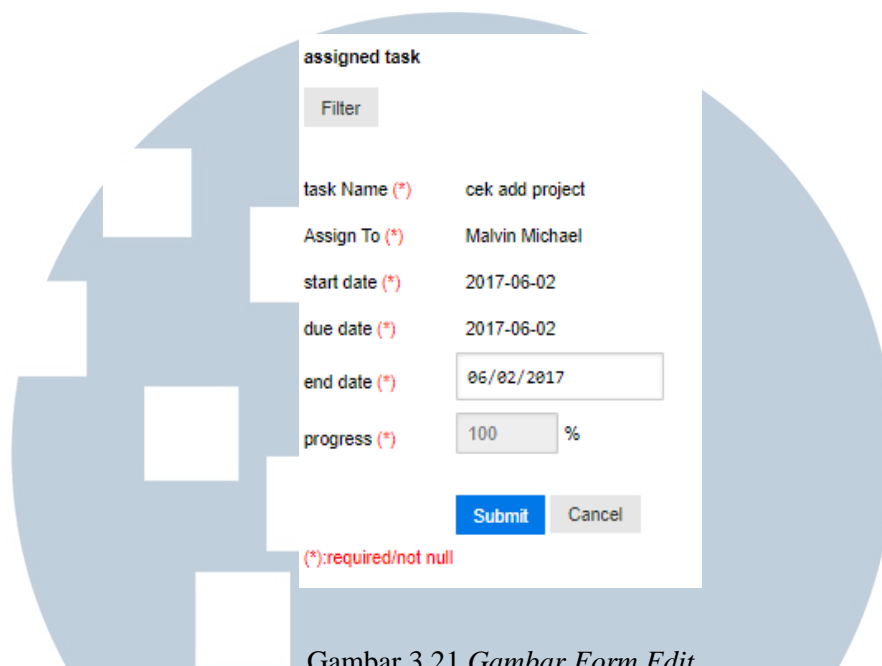


Pengguna baik GM maupun user hanya dapat melakukan perubahan data terhadap data *task* yang ada dalam hal *progress* dan tanggal selesai ketika tugas selesai dijalankan. Untuk menjalankan perubahan data pengguna dapat membuka *form* dengan menekan *edit* pada tabel data. Pada *form* seperti gambar 3.21, *form* perubahan terdapat field sebagai berikut:

1. *Task Name*: berisi data nama dari *task* yang ingin diubah.
2. *Assign To*: berisi data dari orang yang ditugaskan untuk mengerjakan tugas.
3. *Start Date*: berisi data dari tanggal *task* dimulai.
4. *Due Date*: berisi data dari tanggal *task* selesai.
5. *End Date*: berisi data dari tanggal kapan *task* menjadi *complete*. Sesuai dengan yang sudah dijelaskan, *filed* ini dapat dimodifikasi datanya oleh pengguna.
6. *Progress*: merupakan kemajuan *task* yang sudah dikerjakan. Sesuai dengan yang sudah dijelaskan, *filed* ini dapat dimodifikasi datanya oleh pengguna.

Ketika pengguna sudah selesai melakukan pengubahan data maka pengguna bisa melakukan *submit* untuk memproses perubahan tersebut. Penambahan dilakukan memerlukan setidaknya *progress* sehingga bisa diproses. Hasil perubahan dapat dilihat didalam tabel.

U N I V E R S I T A S  
M U L T I M E D I A  
N U S A N T A R A



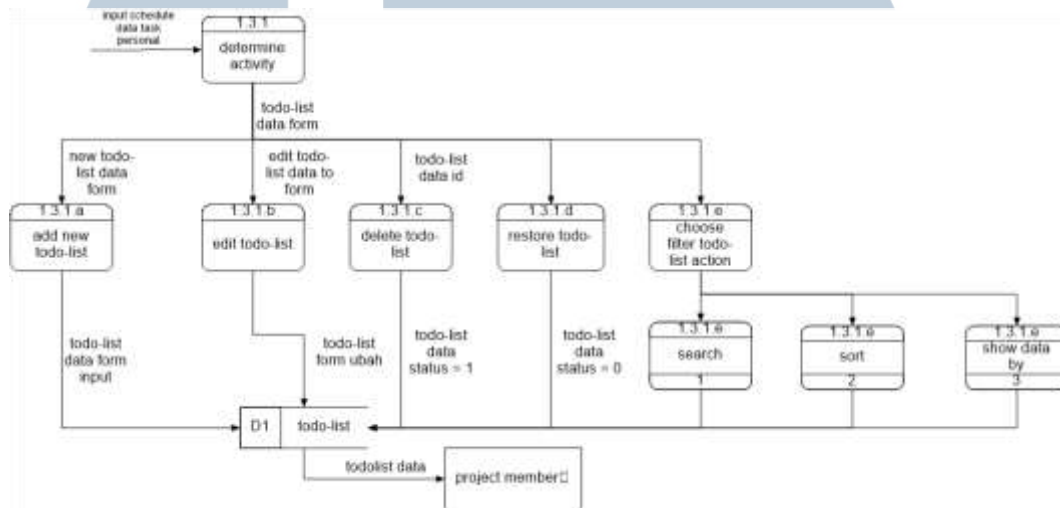
Gambar 3.21 Gambar Form Edit

Untuk mempermudah pencarian pengguna disediakan fasilitas *filter* seperti modul yang lainnya. Fasilitas *filter* terdiri dari *search data*, *show data by*, *sort tabel*. Ketiga *filter* tersebut jika sudah dilakukan pengaturan maka pengaturan disimpan sementara, sehingga ketika dilakukan pada *filter* yang lain tidak hilang. Untuk menghapus pengaturan *filter* yang telah diatur maka pengguna dapat menekan *clear filter*.

### 3. Modul ToDo-List

Modul *todo-list* merupakan modul yang digunakan untuk mencatat semua tugas yang dibuat secara personal. Modul ini berfungsi sebagai *time management* dan *reminder* personal guna membantu pengguna untuk mengerjakan pekerjaan yang dilakukan berdasarkan prioritasnya masing-masing sesuai pengguna.

Untuk menggambarkan alur data dari modul *to-do list* yang telah dibuat, penulis menggambarkan *DFD diagram level 1* pada gambar 3.22 dari sistem yang telah dibuat.



Gambar 3.22 DFD Level 1 Todo-list Diagram

Untuk melakukan penambahan data *to-do list* maka pengguna memilih menu *add new* untuk membuka *form* penambahan *to-do list*. Pada *form* penambahan seperti gambar 3.23, *form todo-list* terdapat *field* sebagai berikut:

1. *Task Name*: digunakan untuk menambah kegiatan dilakukan oleh pengguna
2. *Priority*: digunakan untuk memberi tingkat kepentingan terhadap tugas personal tersebut.
3. *Due Date*: digunakan untuk memberi akhir kapan tugas harus selesai.
4. *Progress*: digunakan untuk memberi kemajuan dari tugas yang dipunyai.

5. *Description*: digunakan untuk memberi deskripsi dari tugas personal yang dibuat.

Setelah *form* terbuka pengguna mengisi data *todo-list* yang perlu diisi, setidaknya-tidaknya pengguna harus mengisi *field* yang sifatnya *required* atau *not null* sehingga penambahan dapat diproses. Ketika sudah dilakukan pengguna melakukan *submit*. Jika *field* yang sifatnya *required* atau *not null* kosong maka penambahan tidak akan diproses.

The image shows a web form titled "to-do". At the top, there are three buttons: "Add New" (highlighted in blue), "Filter", and "history". Below these are several input fields: "Task Name (\*)" with a text box containing "Task Name"; "Priority (\*)" with a text box containing "1"; "Due Date (\*)" with a text box containing "mm/dd/yyyy"; and "Progress" with a text box containing "0" and a "%" symbol. Below the progress field is a rich text editor for "Description" with a toolbar containing icons for undo, redo, bold, italic, bulleted list, numbered list, link, unlink, and a close button. At the bottom of the form are "Submit" (blue) and "Cancel" (grey) buttons. A red asterisk (\*) is located below the "Submit" button, with the text "required/not null" next to it.

Gambar 3.23 *Form* Penambahan *ToDo-List*

Jika ingin melakukan pembaruan data, pengguna bisa menekan *edit* yang ada pada tabel *todo-list*. Dengan menekan *edit* pada tabel *todo-list* pengguna bisa membuka *form* pembaruan seperti pada gambar 3.24. Pada *form* pembaruan, field yang dipunyai sama seperti *form* penambahan. *Form* pembaruan berisi data yang ingin diperbaharui, untuk melakukan perubahan pengguna menginput pembaruan

sesuai dengan kebutuhan. Setidaknya semua *field* yang *required* atau *not null* harus terisi sehingga dapat diproses. Setelah menginput perubahan sesuai dengan kebutuhan, pengguna menekan tombol *submit* untuk memulai proses perubahan data.

The screenshot shows a web form titled "to-do" with the following fields and controls:

- Buttons: "Add New", "Filter", "history"
- Task Name (\*): Input field containing "perbaiki UAT dan bug"
- Priority (\*): Input field containing "5"
- Due Date (\*): Input field containing "06/12/2017"
- Progress (\*): Input field containing "100" followed by a "%" sign
- Status (\*): Dropdown menu showing "completed"
- Description: Rich text editor containing the text "UAT kacauuuu betulin ya bug masi ada juga"
- Buttons: "Submit" (highlighted in blue), "Cancel"
- Legend: "(\*):required/not null"

Gambar 3.24 Form Pembaruan *ToDo-List*

Jika ingin melakukan penghapusan data, pengguna dapat menekan *delete*. Pengguna mendapatkan konfirmasi untuk melanjutkan penghapusan atau tidak. Jika pengguna ingin melakukan penghapusan maka ditekan *ok*. Jika ditekan *cancel* maka permintaan untuk melakukan penghapusan tidak dilakukan. Seperti pada gambar 3.25, setiap *todo-list* akan ditampilkan ke dalam tabel *history* baik yang telah di hapus ataupun yang masih berjalan.

to-do

Filter Return

No	Task Name	Priority	Due Date	Progress	Status	Description
1	buat laporan	5	2017-05-25	100 %	overdue	buat laporan dari bab 1-4
2	abc	1	2017-04-19	1 %	overdue	acbbdefghijklm
3	bug setting dan checking other bug	5	2017-04-20	100 %	complete	bug checking ada di setting dashboard
4	perbaiki UAT dan bug	5	2017-06-12	100 %	complete	UAT kacauuu betulin ya bug masi ada juga
5		4	2017-04-06	0 %	not started	
6	abc	1	2017-04-29	100 %	overdue	abcccc
7	lengkapi form km-03&km-04	5	2017-04-10	100 %	overdue	lengkapi task form km 03 dan km 04

Gambar 3.25 Gambar Tabel *History ToDo-List*

Untuk mempermudah pengguna dalam melihat data maka diberikan fasilitas *filter*. Terdapat 3 jenis *filter* yaitu *searching data*, *sorting table*, *show data by*. *Filter* dapat dilakukan pada tabel *to-do list* maupun *history*.

Ketika pengguna memilih *filter search* maka pengguna melakukan *filter* pencarian berdasarkan kategori yang dipilih. *Filter search* dilakukan dengan membuka *form* pencarian melalui pilihan *search data* pada pilihan kegiatan *filter*. Ketika *form* pencarian dimunculkan seperti pada gambar 3.26, pengguna perlu mengisi *field* kategori dan *field keyword* untuk memulai pencarian, ketika ada *field* yang tidak lengkap maka pencarian tidak dapat dilakukan. Untuk memproses pengguna dapat menekan *search*. Hasil pencarian ditampilkan ke dalam tabel.

U N I V E R S I T A S  
M U L T I M E D I A  
N U S A N T A R A

to-do

Add New Filter history

Show Data By Search Data Sort Table Clear Filter

Search By (\*) Task Status

Search (\*) overdue

Search Cancel

(\*):require/not null

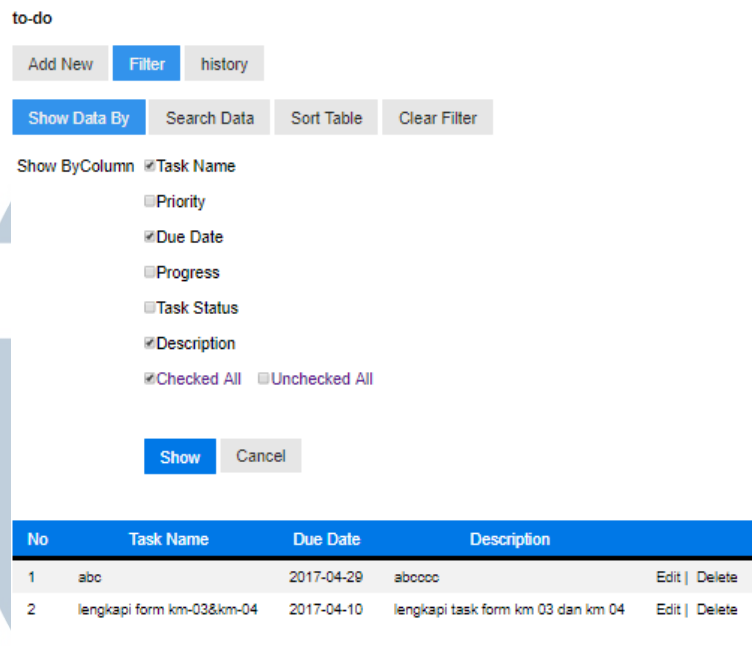
No	Task Name	Priority	Due Date	Progress	Status	Description	
1	abc	1	2017-04-29	100 %	overdue	abc000	Edit   Delete
2	lengkapi form km-03&km-04	5	2017-04-10	100 %	overdue	lengkapi task form km 03 dan km 04	Edit   Delete

Gambar 3.26 Filter Pencarian Data

Ketika pengguna memilih *filter show data by*, pengguna melakukan *filter* penampilan berdasarkan kategori yang ingin ditampilkan di dalam tabel *todo-list*. Untuk memulai *filter* dengan memunculkan *form*, pengguna dapat memilih *show data by* pada pilihan aktifitas *filter*. Ketika *form show by* dimunculkan seperti pada gambar 3.27, pengguna memilih kategori yang ingin digtampilka pengguna. Jika ingin mengosongkan pilihan kategori maka pengguna dapat menekan *unchecked all*, jika ingin mencentang semua dan mereset penampilan ke tabel *todo-list* pengguna dapat menekan *cheked all*. Untuk memproses *filter* penampilan maka pengguna menekan *show*.

Hasil penampilan *filter* pada tabel menyesuaikan dengan kateogri yang dipilih pengguna.

U N I V E R S I T A S  
M U L T I M E D I A  
N U S A N T A R A

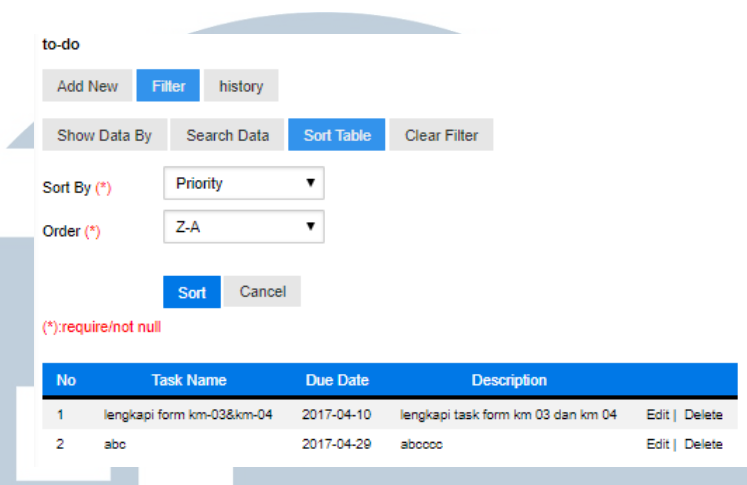


Gambar 3.27 Filter Show Data By

Ketika pengguna memilih filter *Sorting Table*, pengguna melakukan filter dalam bentuk pengurutan data sesuai dengan kategori yang dipilih dan jenis pengurutan yang dipilih. Untuk memulai filter *sorting*, pengguna dapat memilih *sort table* pada pilihan filter sehingga form pengurutan ditampilkan. Pada gambar 3.28 form pengurutan diisi pengguna pada field kategori dan *sort type*. Didalam *sort type* terdapat dua jenis pengurutan yaitu A-Z sebagai *ascending* dan Z-A sebagai *descending*. Pengurutan *ascending* merupakan pengurutan dari terkecil ke terbesar, dan pengurutan *descending* merupakan pengurutan dari terbesar ke terkecil. Hasil pengurutan akan disesuaikan dengan jenis data kategori tersebut.

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA



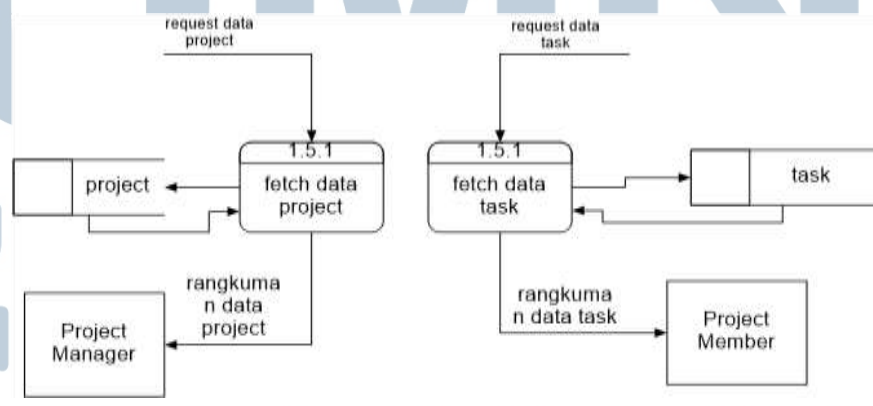


Gambar 3.28 Filter Sorting

Semua *filter* yang diatur disimpan sementara sehingga ketika pengguna melakukan *filter* yang lain, pengaturan *filter* yang sudah dilakukan masih berlaku. Ketika filter sudah selesai dilakukan maka perlu dilakukan *clear filter* sehingga pengaturan *filter* yang tersimpan tidak hilang.

#### 4. Modul Dashboard dan Setting-Dashboard

Modul *Dashboard* merupakan rangkuman dari semua proyek dan tugas-tugas dari proyek yang dipunyai dan ditugaskan. Gambar 3.9 menggambarkan alur data yang ada di dalam dashboard pengguna.



Gambar 3.29 DFD Level 1 Dashboard

Terdapat 3 tabel sebagai laporan yang tentang proyek perusahaan beserta tugas-tugas dari proyek yaitu *project all*, *task by project*, *task by assigned*. Ketiga pelaporan ini dapat diatur penampilannya pada *setting dashboard*. Ketika administrator pertama kali masuk ke dalam sistem *dashboard* akan diisi dengan rangkuman *project all* secara default, bagi non-administrator pengguna tidak ditampilkan apapun di dalam *dashboard*.

Bisa dilihat dari gambar 3.30, *project all* merupakan rangkuman atas semua proyek yang dipunyai, bagi akun administrator *project all* menampilkan semua proyek perusahaan. Bagi non-administrator *project all* menampilkan semua proyek yang dimiliki pengguna. Rangkuman *project all* hanya ditampilkan pada akun administrator dan GM bagi akun non-administrator. Di dalam *setting dashboard* jika pengguna merupakan GM maka akan dicek bahwa pengguna punya project atau tidak, jika tidak maka pilihan *project all* pada setting dashboard diblok sehingga tidak dapat dipilih.

welcome ,Patrizka Odie(GM)

---

Project All

---

No	Project Name	Project Manager	Priority	Start Date	Deadline	Progress	Progress Bar	status
1	cek project	Patrizka Odie	5	2017-06-02	2017-06-03	100%	<div style="width: 100%; height: 10px; background-color: blue;"></div>	complete
2	abocdi	Patrizka Odie	4	2017-06-31	2017-06-01	100%	<div style="width: 100%; height: 10px; background-color: blue;"></div>	complete

Gambar 3.30 Rangkuman *Project All* GM

Pada Gambar 3.31 yang menggambarkan *Task by Project*, merupakan rangkuman yang menampilkan pengelompokan tugas-tugas berdasarkan proyek, pada *task by project* pengguna dapat menampilkan task menggunakan fitur

*fullcalendar* untuk memvisualisasikan tugas-tugas tersebut. Rangkuman *task by project* ditampilkan pada semua akun yang menggunakan sistem. Pada *setting dashboard* jika pengguna berakun GM atau *User* dicek proyek yang dipunyai pengguna, jika pengguna tidak mempunyai proyek maka penampilan *task by project* tidak dapat dipilih.

Task by Project

aboodi[Working Calendar]

No	Task Name	Assign To	priority	Start Date	Deadline	Progress	Status
1	editaswainifava	Patriadka Odie	5	2017-08-04	2017-08-06	100%	complete
2	dasasacadi	Melvin Michal	5	2017-08-03	2017-08-04	100%	complete
3	abc	tatahetata	4	2017-08-02	2017-08-03	100%	complete

cek project[Working Calendar]

No	Task Name	Assign To	priority	Start Date	Deadline	Progress	Status
1	cek delete and restore	awang awang	5	2017-08-08	2017-08-08	100%	complete
2	cek filter	tatahetata	5	2017-08-04	2017-08-04	100%	complete
3	cek edit project	Tina	5	2017-08-03	2017-08-03	100%	complete
4	cek add project	Marik Michal	4	2017-08-02	2017-08-02	100%	complete

Gambar 3.31 Rangkuman *Task by Project*

Rangkuman *Task by assigned* pada gambar 3.32, menampilkan semua tugas yang ditugaskan kepada pengguna. Sama seperti *task by project* semua pengguna dapat menampilkan rangkuman *task by assigned*, pada *setting dashboard* pengguna akan dicek apakah pengguna mempunyai *task* yang ditugaskan kepada pengguna. Jika pengguna tidak memiliki *task* maka penampilan *task by assigned* pada dashboard setting tidak dapat dipilih.

Assign Task

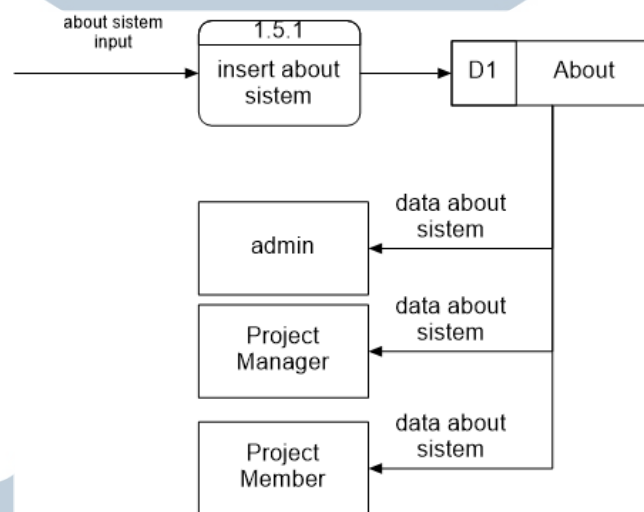
No	Project Name	Task Name	Assign To	Priority	Start Date	Deadline	Progress	Status	Description
1	aboodi	editaswainifava	Patriadka Odie	5	2017-08-04	2017-08-06	100%	complete	disacabacawedatswiozafasf
2	login	input username	Patriadka Odie	5	2017-08-02	2017-08-02	100%	complete	sklasdrijoicjodjofafa
3	login	login validate to database	Patriadka Odie	5	2017-08-03	2017-08-04	100%	complete	login proses found then login

Gambar 3.32 Rangkuman *Task by Assigned*

Di dalam *setting dashboard* untuk mengatur penampilan setiap tabel dalam hal pengurutan dan batas data yang ditampilkan, pengguna dapat mengatur pada *tabel setting*. Ketika ingin melakukan pengaturan penampilan, pengguna perlu mengisi setidaknya *required field* pada ketiga rangkuman laporan sehingga pengaturan ini ditampilkan.

### 5. Modul *About*

Modul *About* merupakan modul yang digunakan untuk memberikan penjelasan tentang sistem yang berjalan. Modul ini merupakan modul tambahan yang dapat dikostum hanya oleh administrator. Gambar 3.33 akan menggambarkan DFD *about* yang dipakai di dalam modul *about*.



Gambar 3.33 DFD level 1 *About*

Untuk membuat penjelasan tentang sistem, maka administrator memilih modul *about setting* pada *sub menu setting*, untuk membuka *form about setting*. Untuk menampilkan tulisan tentang penjelasan sistem halaman *about*, maka pengguna harus mengisi editor yang disediakan dan dilakukan *submit*. Hasil yang

ditampilkan akan menyesuaikan dengan pengaturan *about* yang dibuat. Semua modul diatas telah ditesting melalui UAT (*User Acceptance Test*) yang akan dilampirkan pada akhir laporan ini

### **3.2.2. Fotografi 360 Ruko Golden 8, Gading Serpong**

Selain membuat sistem penulis diberikan tugas sebagai fotografer dari foto 360 pada Ruko Golden 8, Gading Serpong. Foto ini merupakan foto interaktif yang digunakan pada *google streetview*, dimana pengguna dapat berinteraksi dengan foto tersebut. Interaksi yang dapat dilakukan oleh pengguna atas foto tersebut merupakan view dengan sudut pandang seluas 360 derajat. Foto 360 ini dilakukan perusahaan fotografi Ducosky.

Didalam pelaksanaan proyek fotografi 360 Ruko Golden 8 Boulevard, Gading Serpong, penulis melakukan fotografi pada sepanjang jalanan pada ruko. Sepanjang jalanan pada setiap titik foto tersebut pengguna mengambil gambar dengan semua sudut hingga membentuk sudut 360 pada semua sisi. Gambar diambil dimulai dari sudut teratas menghadap ke langit hingga kebawah dan dilakukan hingga membentuk 1 lingkaran penuh, semakin detil semakin bagus. Gambar tersebut diambil pada siang hari dikarenakan dibutuhkan cahaya untuk menghasilkan foto yang sempurna, jika terjadi kesalahan maka penulis harus mengulanginya. Fotografi dilakukan dalam team yang terdiri dari dua orang yaitu penulis dan teman penulis, sehingga dapat tercover dengan efektif daerahnya.

### 3.3. Kendala dalam Proses Kerja Magang

Selama kerja magang, penulis tidak dapat melakukan tugas yang diberikan dengan lancar. Penulis mengalami kesulitan dalam mengerjakan program terutama pada bagian *UI Design (User Interface)*, dan ditemukan *error/bug* pada program yang dibuat. Di samping itu waktu magang terhambat, dikarenakan penulis melakukan pembuatan skripsi bersamaan dengan waktu kerja magang.

Dalam melakukan tugas fotografi 360 penulis mempunyai kesulitan pada prakteknya, karena penulis baru pertama kali melakukan fotografi.

### 3.4. Solusi dari Kendala

Dalam hal mengatasi permasalahan yang ditemui penulis khususnya:

1. *UI Design*, penulis dibimbing oleh *project manager* dalam menggunakan *CSS (Cascading Style Sheet)*. Penulis juga melakukan *browsing* yang berisi artikel tentang permasalahan yang penulis hadapi dalam menyelesaikan permasalahan *UI Design*.
2. Permasalahan *error/bug* sistem, penulis melakukan *browsing* terhadap *troubleshooting error* yang ditemukan.
3. Permasalahan waktu magang yang bersamaan dengan penyelesaian skripsi, penulis diberikan kelonggaran waktu untuk menyelesaikan skripsi terlebih dahulu sehingga waktu magang penulis diperpanjang.
4. Untuk kesulitan dalam melakukan Fotografi 360, penulis dibimbing dan diberikan pelatihan fotografi oleh manajer proyek penulis, sehingga penulis dapat menjalankan tugas fotografi walau masih terdapat kekurangan.