

BAB III PELAKSANAAN KERJA PROFESI

3.1 Bidang Kerja

Saat pelaksanaan KP, praktikan ditempatkan pada divisi Teknologi Informasi / IT (*Information Technology*). Divisi ini memiliki tanggung jawab untuk membantu memecahkan masalah teknologi yang ada pada perusahaan. Pada saat pelaksanaannya praktikan diberikan kepercayaan untuk mengerjakan tugas yang ada pada divisi tersebut.

Salah satu tugas tersebut adalah membuat aplikasi "*Ticketing System*". Aplikasi ini ditujukan untuk orang lapangan, dimana hal ini dapat membantu mereka dalam melaporkan masalah yang ada di lapangan kepada departemen yang ada di kantor. Proyek ini juga berkaitan dengan departemen lain, karena orang lapangan akan mengirimkan laporan kepada orang yang ada pada departemen di kantor. Dengan demikian, mereka dapat terhubung dan berkomunikasi. Selain itu, perusahaan jadi memiliki riwayat data masalah di lapangan dan dapat ditinjau kembali untuk mengurangi masalah kedepannya.

Aplikasi ini memiliki fitur membuat, membaca, memperbaiki dan menghapus, atau biasa disebut CRUD (*Create Read, Update, Delete*). CRUD akan digunakan pada halaman login, tiket, project, departemen dan pengguna. Namun beberapa fitur dapat digunakan tergantung dengan peran pengguna (*user role*). Peran pengguna dapat diubah sesuai dengan kebutuhan departemen masing masing. Pada aplikasi ini praktikan berfokus pada pemotongan data (*slicing data*) status berdasarkan departemen dan PIC (*Project In Charge*) hingga pembuatan sistem peran pengguna (*user role*).

Aplikasi "*Ticketing System*" ini dikembangkan dengan "Appsheet" sebagai platform utama, karena "Appsheet" memiliki pemrosesan data yang cepat namun kompleks. Oleh karenanya, aplikasi ini dapat dibuat dengan waktu yang singkat, lengkap dan memiliki bug yang rendah. "Appsheet" ini juga dapat digunakan untuk *mobile* dan *website*. Selain itu, untuk penyimpanan datanya menggunakan basisdata (*database*) "MariaDB" melalui *idcloudhosting.com*. Praktikan bisa mengimplementasi materi perkuliahan terkait server basisdata kepada proyek aplikasi yang akan dikerjakan.

3.2 Pelaksanaan Kerja

Pelaksanaan kerja merupakan bagian utama dalam mata kuliah KP, tahap ini wajib dilakukan demi menyelesaikan mata kuliah KP. Berikut merupakan tahapan praktikan dalam melaksanakan mata kuliah KP di perusahaan pertanian.

3.2.1 Perancangan Ticketing System

Sebelum melaksanakan proyek, Praktikan melakukan perancangan untuk memudahkan praktikan dalam pengerjaan. Dengan demikian, pekerjaan dapat dilakukan dengan baik dan terstruktur. Perancangan tersebut akan dijelaskan sebagai berikut ini.

(1) Spesifikasi Perangkat Lunak

Berikut merupakan spesifikasi perangkat lunak untuk menunjang pembuatan aplikasi "*Ticketing System*".

- a) Browser, berguna untuk menjalankan "Appsheet" dan "*hosting*".
- b) Appsheet, sebagai platform utama dalam pembuatan sistem ini.

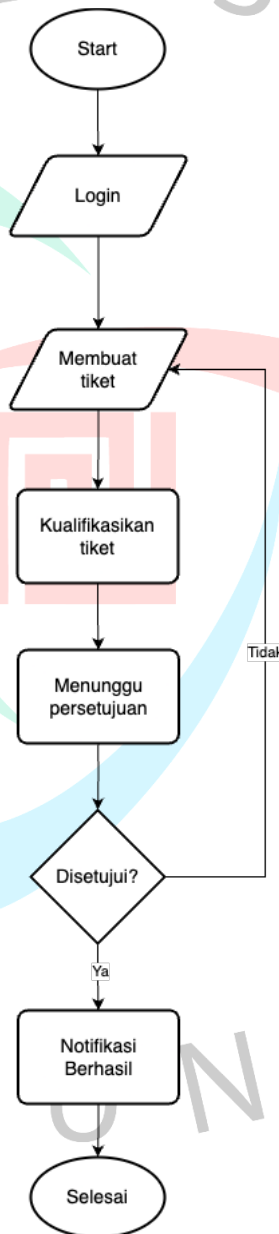
(2) Spesifikasi Perangkat Keras

Berikut merupakan spesifikasi perangkat keras yang digunakan untuk membuat aplikasi "*Ticketing System*".

- a) Laptop, digunakan dalam masa pengembangan (*development*) dengan menjalankan website "Appsheet" dan "*hosting*".
- b) Smartphone, digunakan pada saat pengujian (*testing*). Karena kebanyakan pengguna akan menggunakan smartphone untuk menerima notifikasi.

(3) Diagram Alir

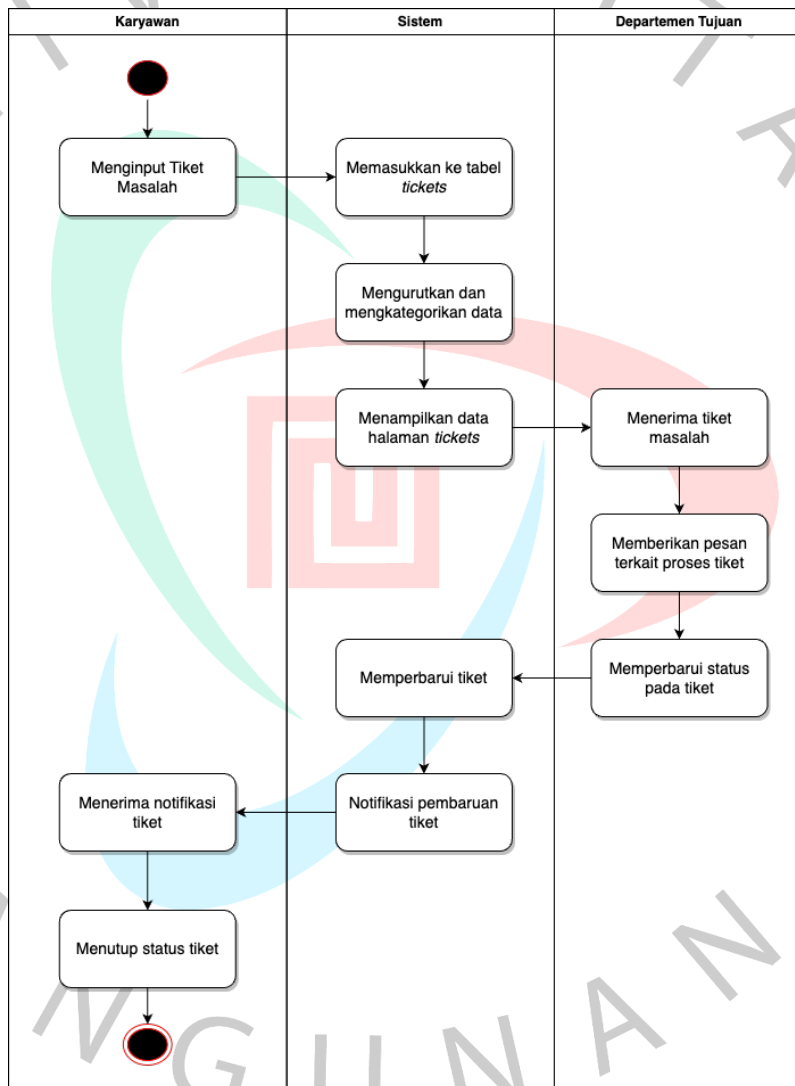
Diagram alir atau biasa disebut *Flowchart Diagram* merupakan diagram yang menunjukkan bagaimana runtutan program berjalan. Dengan adanya diagram ini, praktikan dalam lebih mudah dalam membaca alur dan menerapkannya kedalam program. Diagram alir ditampilkan pada gambar 3.1.



Gambar 3.1 Diagram Alir

(4) Diagram Aktivitas

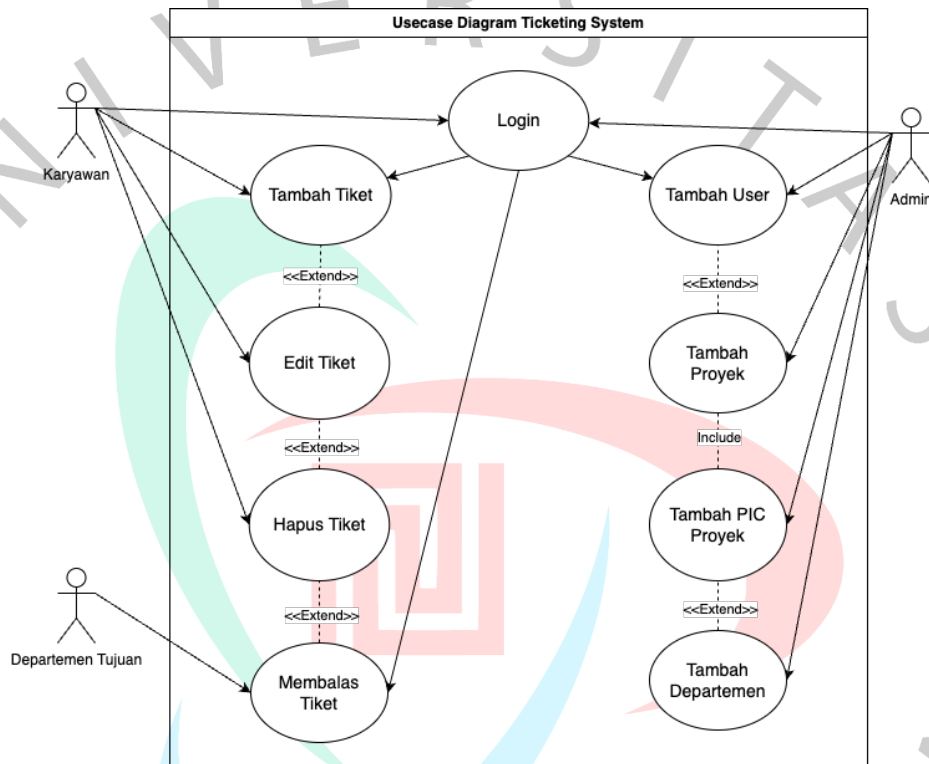
Diagram aktivitas merupakan diagram yang menggambarkan proses-proses data yang sedang dirancang. Proses tersebut diantaranya ialah, Karyawan, Sistem, dan Departemen Tujuan. Diagram aktivitas ditampilkan pada gambar 3.2, dan menjelaskan alur "Ticketing System" dimulai dari karyawan mengisi tiket masalah hingga menerima respon dari departemen tujuan.



Gambar 3.2 Diagram Aktivitas

(5) Diagram Usecase

Diagram usecase digunakan untuk merincikan apa saja fungsi yang dapat dilakukan sistem. Dengan demikian, praktikan dapat lebih mudah dalam pengembangan aplikasi ini kedepannya. Diagram usecase ditampilkan pada gambar 3.3.



Gambar 3.3 Diagram Usecase

3.2.2 Implementasi Database

Setelah melakukan perancangan, tahap selanjutnya yaitu membuat basisdata menggunakan "MariaDB" di dalam "hosting" perusahaan yaitu *id.cloudhosting.com*.

- (1) Kolom dan tipe data pada tabel tickets ditampilkan pada gambar 3.4, tabel ini nantinya akan berisi laporan masalah yang ada pada lapangan. Kolom tersebut diantaranya sebagai berikut.
 - a) Kolom ticketID, berisikan id unik untuk tiket.
 - b) Kolom tanggal, berisikan tanggal pada saat melaporkan masalah.
 - c) Kolom owner, berisikan pengguna yang akan mengisi tiket dan diinputkan secara otomatis dari sistem.

- d) Kolom status, merupakan status proses untuk tiket.
- e) Kolom project, berisikan proyek yang akan dilaporkan masalahnya. Kolom ini merupakan kunci asing (foreign key) dan terhubung kepada id projects.
- f) Kolom deskripsi_problem, berisikan deskripsi terkait masalah.
- g) Kolom tujuan, akan berisi nama orang yang akan ditunjukan terkait masalah tersebut, nama orang ini merupakan nama PIC dari projects.
- h) Kolom gambar, berisikan lokasi untuk menyimpan gambar.
- i) Kolom closed, berisikan tanggal ketika tiket tersebut ditutup kasusnya.

#	Name	Type	Collation	Attributes	Null	Default
<input type="checkbox"/>	1 ticketID	varchar(10)	latin1_swedish_ci		No	None
<input type="checkbox"/>	2 tanggal	date			No	None
<input type="checkbox"/>	3 owner	varchar(50)	latin1_swedish_ci		No	None
<input type="checkbox"/>	4 status	varchar(30)	latin1_swedish_ci		No	None
<input type="checkbox"/>	5 project	varchar(10)	latin1_swedish_ci		No	None
<input type="checkbox"/>	6 deskripsi_problem	text	latin1_swedish_ci		No	None
<input type="checkbox"/>	7 tujuan	varchar(10)	latin1_swedish_ci		No	None
<input type="checkbox"/>	8 gambar	text	latin1_swedish_ci		No	None
<input type="checkbox"/>	9 closed	date			No	current_timestamp()

Gambar 3.4 Tabel Tickets

(2) Kolom dan tipe data tabel “projects” ditampilkan pada gambar 3.5. Tabel ini digunakan untuk menyimpan proyek apa saja yang sedang berjalan, proyek ini nantinya akan terhubung kepada tabel “tickets”. Maka dari itu, data dapat ditarik dan ditampilkan dalam form input tiket. Berikut merupakan kolom pada tabel “projects”.

- a) Kolom projectID, berisikan id unik untuk “project”.
- b) Kolom project_name, berisikan nama dari proyek yang sedang berjalan.
- c) Kolom tanggal_project, berisikan tanggal proyek dimulai.
- d) Kolom status_project, berisikan status proyek ketika sedang aktif.
- e) Kolom pic, berisikan kumpulan id para PIC yang bertanggung jawab pada proyek tersebut.
- f) Kolom count, berisikan jumlah PIC yang ada pada proyek tersebut.

#	Name	Type	Collation	Attributes	Null	Default	Comments
<input type="checkbox"/>	1 projectID	varchar(10)	utf8_general_ci		No	None	
<input type="checkbox"/>	2 project_name	varchar(50)	utf8_general_ci		No	None	
<input type="checkbox"/>	3 tanggal_project	date			No	None	
<input type="checkbox"/>	4 status_project	varchar(10)	utf8_general_ci		No	None	
<input type="checkbox"/>	5 pic	text	utf8_general_ci		No	None	
<input type="checkbox"/>	6 count	varchar(3)	utf8_general_ci		No	None	

Gambar 3.5 Tabel Projects

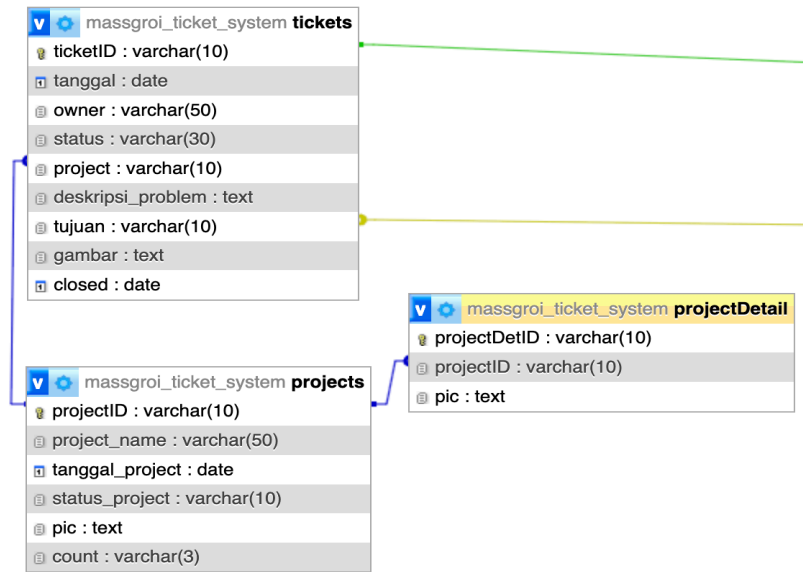
(3) Kolom dan tipe data tabel “projectDetail” ditampilkan pada gambar 3.6. Tabel ini nantinya berisi PIC dari tabel “projects” yang telah dipecah, sehingga “id pic” dapat ditampilkan kedalam “form input ticket” pada bagian tujuan. Berikut merupakan penjelasan kolomnya.

- a) Kolom projectDetID, berisikan id unik untuk tabel “projectDetail”.
- b) Kolom projectID, merupakan kunci asing yang terhubung kepada “id projects”.
- c) Kolom pic, merupakan kunci asing dan berisi “id pic” yang telah dipecah dari tabel “projects”.

#	Name	Type	Collation	Attributes	Null	Default	Comments
<input type="checkbox"/>	1 projectDetID	varchar(10)	utf8_general_ci		No	None	
<input type="checkbox"/>	2 projectID	varchar(10)	utf8_general_ci		No	None	
<input type="checkbox"/>	3 pic	text	utf8_general_ci		No	None	

Gambar 3.6 Tabel “ProjectDetail”

(4) Relasi yang dibuat untuk ketiga tabel tersebut dijelaskan pada gambar 3.7, relasi ini berguna untuk memanggil data dari tabel 1 ke tabel 2, sehingga tabel bisa saling berhubungan.

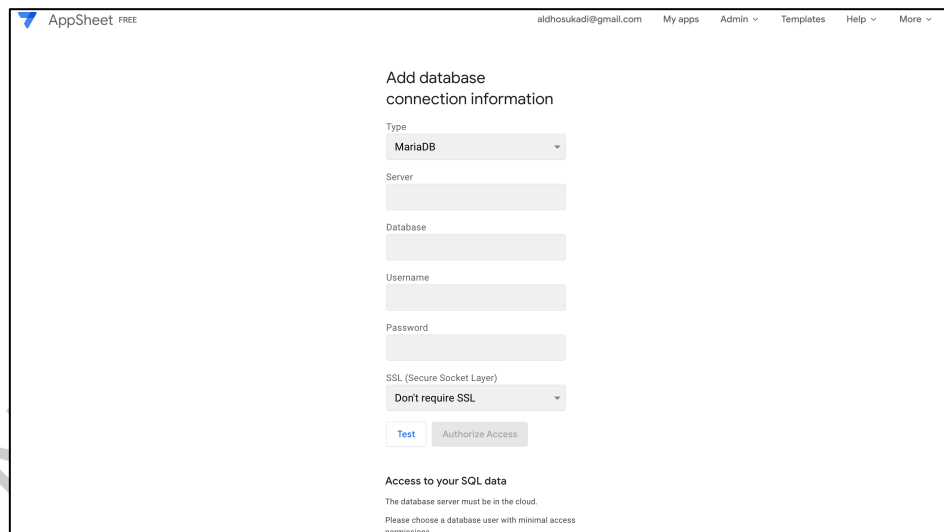


Gambar 3.7 Relasi Basisdata

3.2.3 Menghubungkan “Appsheat” dengan Basisdata

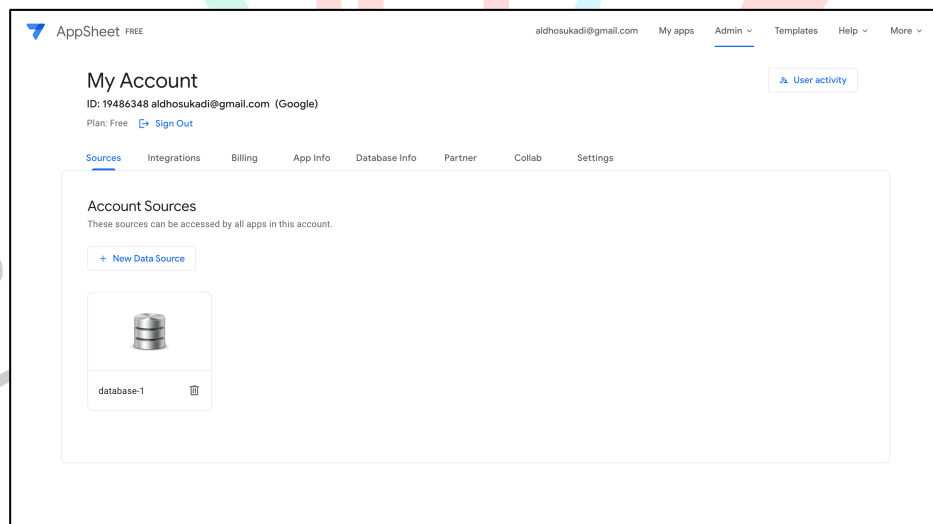
“Appsheat” memiliki banyak metode penyimpanan data. Sebagai penyimpanan bawaan, “Appsheat” menggunakan format csv dan disimpan kedalam “Google Drive”. Namun, format tersebut memiliki performa yang lemah ketika menampung data yang banyak. Oleh karena itu, praktikan menggunakan server basisdata “MariaDB” milik perusahaan dan dihubungkan dengan “Appsheat”. Dengan demikian, data bisa tersimpan dengan baik dan berjalan dengan lancar. Untuk bisa menambahkan database, praktikan harus menambahkannya pada akun “Appsheat” milik perusahaan. Tahapan tersebut akan dijelaskan sebagai berikut.

- (1) Koneksi basisdata ditampilkan pada gambar 3.8, tahap ini merupakan tahapan yang penting untuk dilakukan dan tahapan yang paling pertama dalam menjalankan pembuatan aplikasi ini. Dengan cara koneksi diisi sesuai dengan kebutuhan “Appsheat” seperti server hosting (ip address), *username*, dan *password* basisdata. Kredensial pengguna ini didapat dari server yang telah dibuat di “*hosting*” perusahaan dan bersifat rahasia.



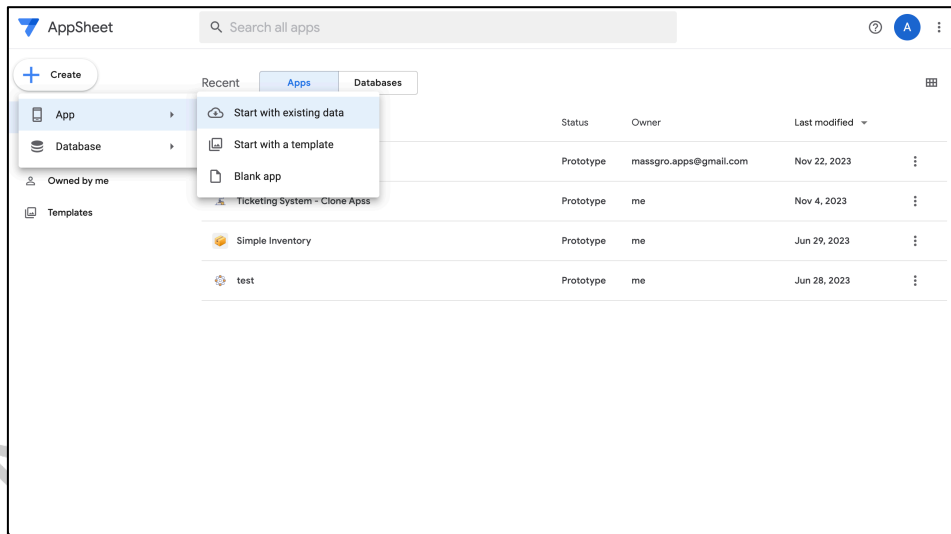
Gambar 3.8 Koneksi Basisdata

- (2) Setelah koneksi berhasil dilakukan pada basisdata di “Appsheet”, maka basisdata akan tampil seperti gambar 3.9. Basisdata tersebut nantinya akan digunakan sebagai penyimpanan utama aplikasi dari “*Ticketing System*” ini. Sebagai peringatan, basisdata pada server harus diisi dahulu dengan tabel dan kolom yang akan digunakan nantinya.



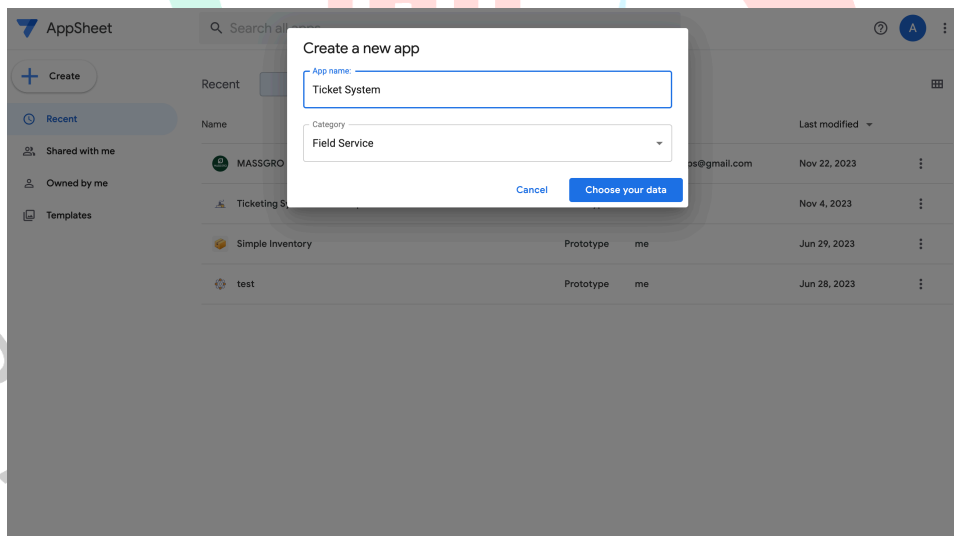
Gambar 3.9 Appsheet Basisdata

- (3) Pembuatan aplikasi baru bisa dilihat pada gambar 3.10, dengan cara memilih menu *Create – App – Start with existing data*. Menu selanjutnya akan ditampilkan dalam bentuk form isian seperti nama aplikasi dan kategori aplikasi.



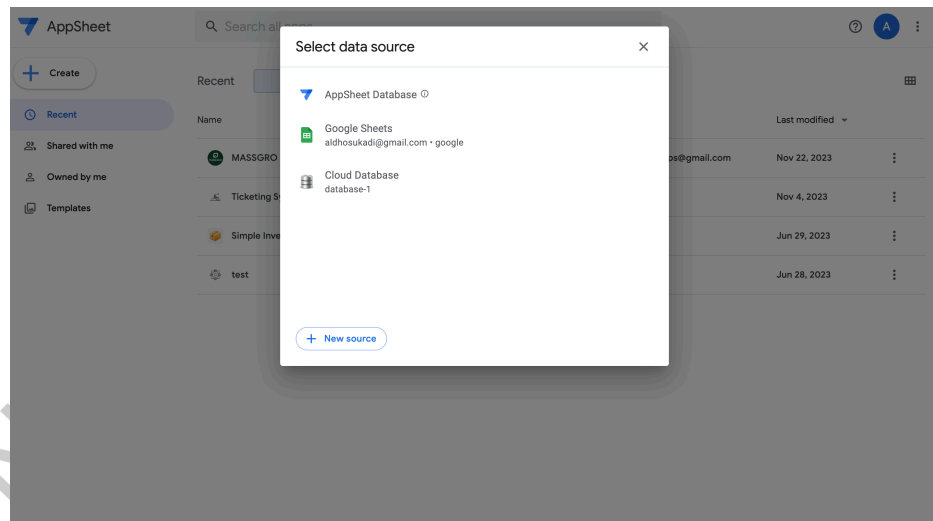
Gambar 3.10 Membuat Aplikasi Baru

- (4) Tahap selanjutnya adalah dengan mengisi nama aplikasi dan kategori aplikasi yang ingin dibuat. Pengisian nama aplikasi ditampilkan pada gambar 3.11.



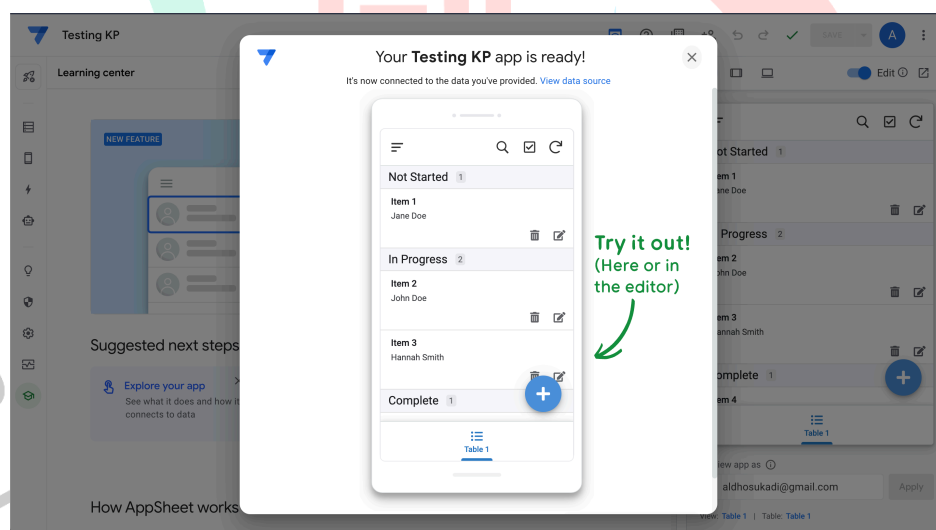
Gambar 3.11 Mengisi Nama Aplikasi

- (5) Setelah selesai mengisi data aplikasi, tahap selanjutnya adalah dengan memilih sumber data yang ingin digunakan. Pemilihan sumber data dapat dilihat pada gambar 3.12, dan telah terisi database yang telah dimasukkan oleh praktikan sebelumnya.



Gambar 3.12 Memilih Sumber Data

- (6) Ketika halaman yang ditampilkan seperti pada gambar 3.13, maka pengkoneksian antara basisdata “MariaDB” dan “Appsheet” telah berhasil.

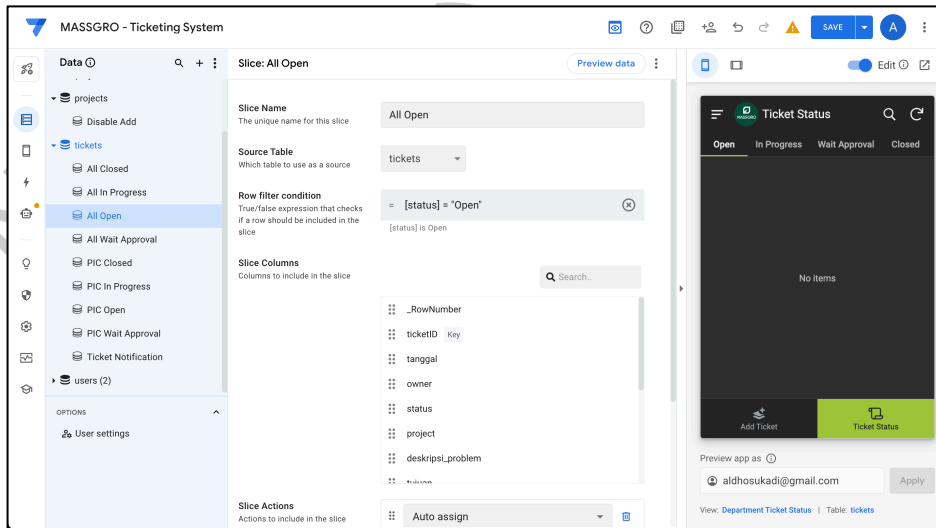


Gambar 3.13 Halaman Sukses

3.2.4 Implementasi Sistem Pemotongan Data

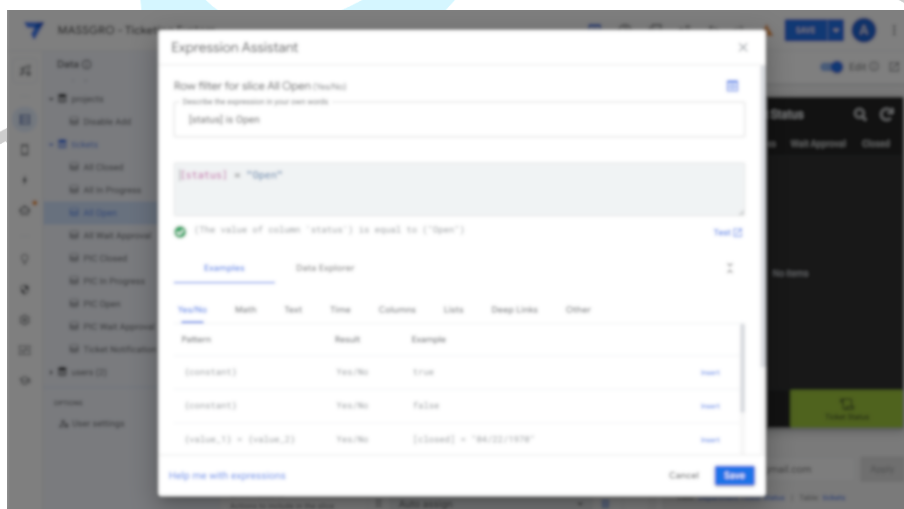
Sistem yang digunakan pada aplikasi ini adalah pemotongan data (slicing data). Pemotongan data merupakan penyaring atau pemisah data berdasarkan kualifikasinya. Kualifikasi ini disebut sebagai status tiket dan diantaranya adalah buka (open), kemajuan (progress), menunggu persetujuan (wait approval), dan ditutup (closed). Status tersebut akan menjadi pemisah.

- (1) Pemotongan data status dapat dilakukan seperti gambar 3.14, langkahnya dengan memilih tabel “ticket” karena tabel tersebut merupakan tabel utama yang akan digunakan pada “form input tiket”. Selanjutnya, buat “slice data” dan tuliskan nama “slicing”.



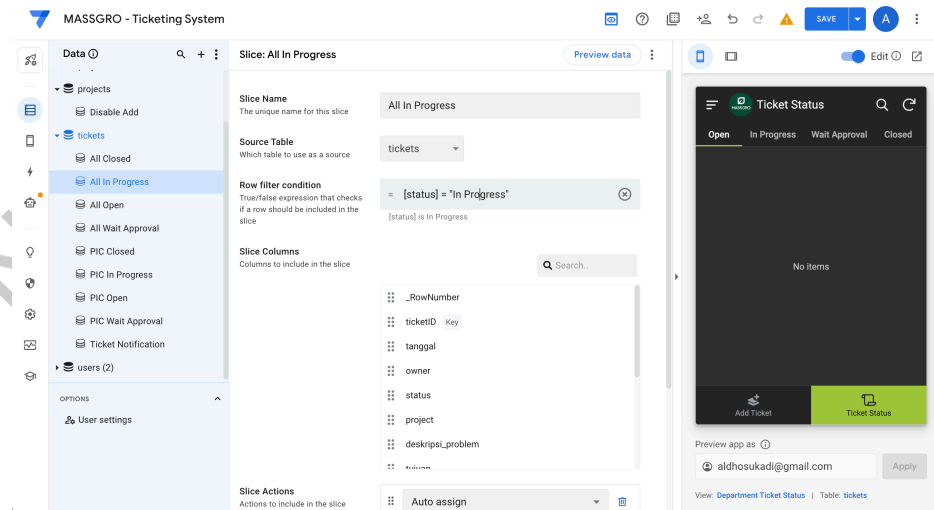
Gambar 3.14 Pemotongan Data

- (2) Setelah itu, sintaks pada bagian “Row filter condition” dapat digunakan seperti gambar 3.15. Sintaks yang digunakan pada gambar tersebut adalah [status] = “open” yang dapat diartikan bahwa data akan dipisah berdasarkan kolom status pada tabel tiket dan memiliki isi yang sama yaitu “open”.



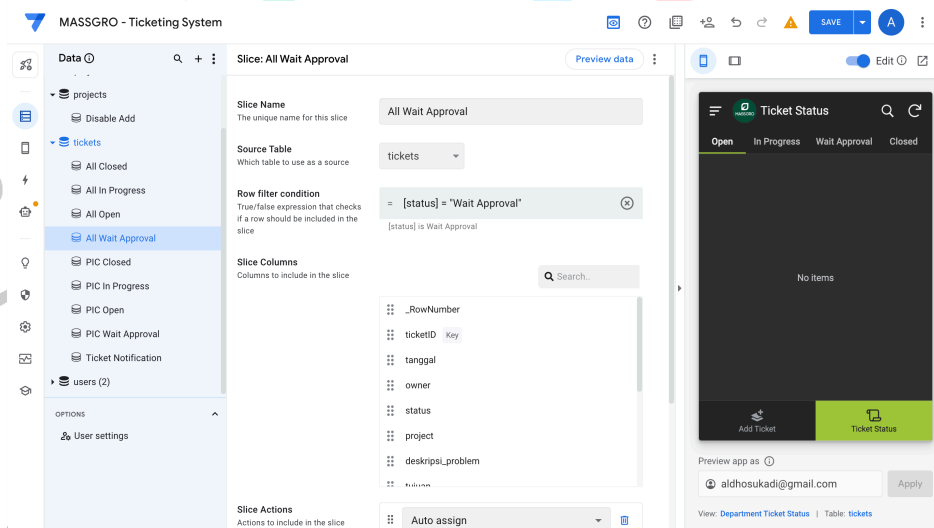
Gambar 3.15 Pemotongan “Status Open”

- (3) Pemotongan “status progress” ditampilkan pada gambar 3.16, pemotongan ini akan memisahkan tiket berdasarkan status yang sedang dalam “progress”.



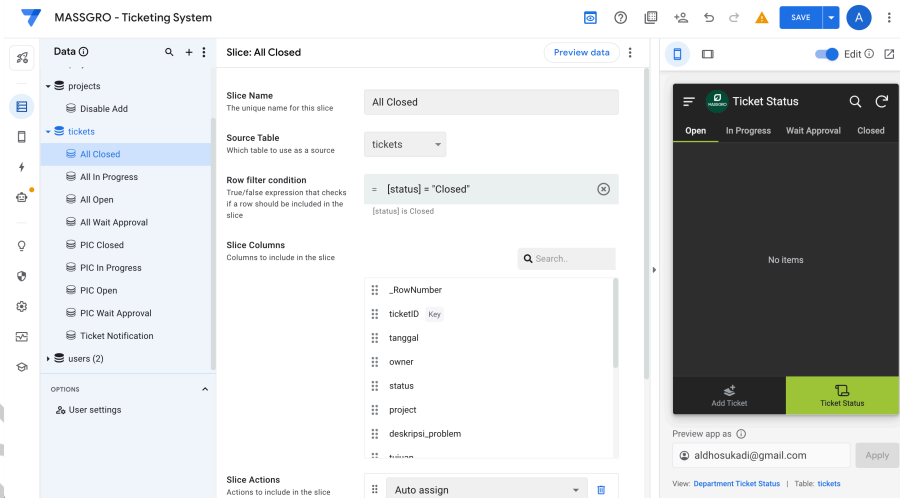
Gambar 3.16 Pemotongan “Status Progress”

- (4) Pemotongan status “wait approval” ditampilkan pada gambar 3.17, pemotongan ini akan memisahkan tiket berdasarkan status yang sedang “wait approval”.



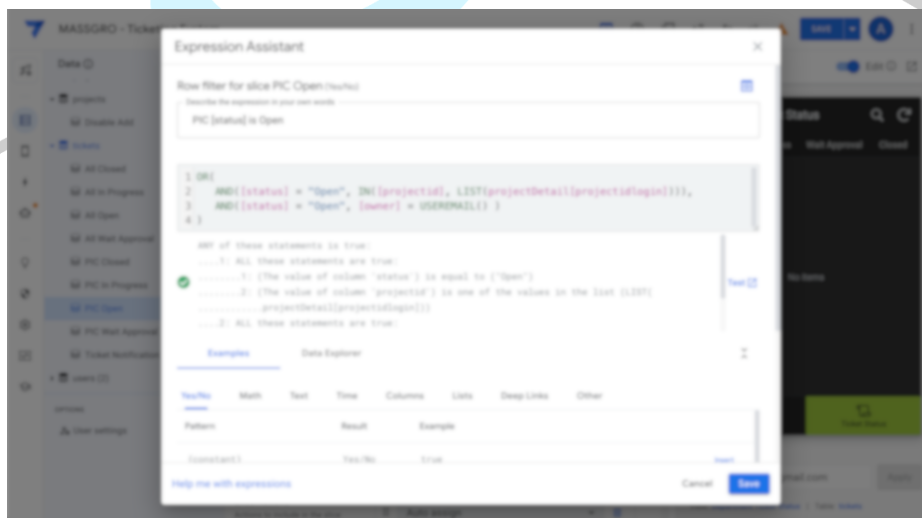
Gambar 3.17 Pemotongan “Status Wait Approval”

- (5) Pemotongan status “closed” ditampilkan pada gambar 3.18, pemotongan ini akan memisahkan tiket berdasarkan status yang sedang “closed”.



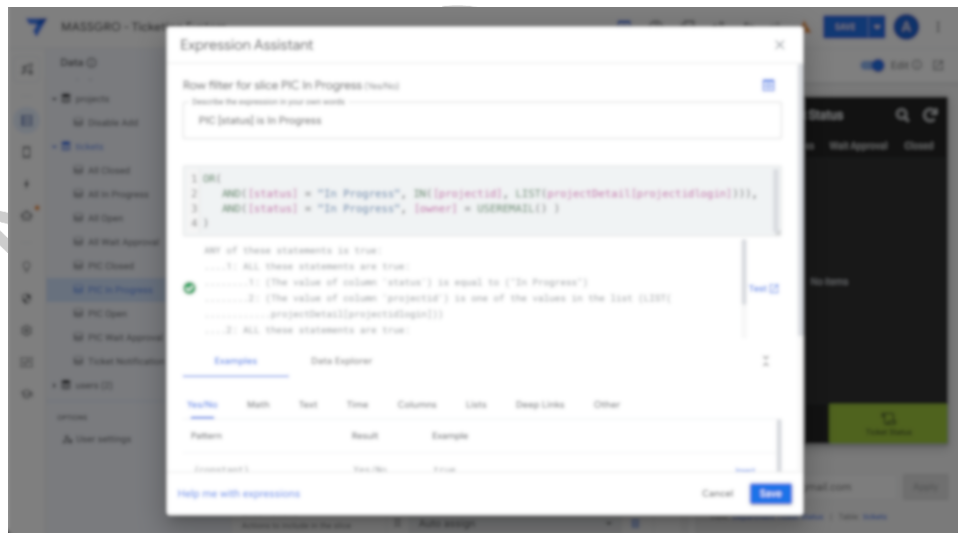
Gambar 3.18 Pemotongan “Status Closed”

- (6) Langkah selanjutnya adalah dengan menerapkan sistem peran pengguna, sistem ini menjadi kualifikasi kedua setelah status. Nantinya data akan difilter berdasarkan PIC yang ada pada tabel “project”. PIC tersebut dapat melihat tiket yang berhubungan dengan dirinya. Selain itu, pemilik tiket atau disebut sebagai pemilik (*owner*) dapat melihat tiket yang telah dimasukkan olehnya. Dengan demikian, pemilik tiket dapat mengubah, menghapus dan memberikan pesan kepada tiket. Sintaks ditampilkan pada gambar 3.19, sintaks tersebut memiliki arti apabila pic dalam tiket proyek sama dengan pengguna yang login dan status tiketnya “open” maka tiket akan ditampilkan kepada pengguna.



Gambar 3.19 Pemotongan “Status PIC Open”

- (7) Pemotongan “status PIC progress” ditampilkan pada gambar 3.20, gambar tersebut memiliki arti sintaks apabila pic dalam tiket proyek sama dengan pengguna yang login dan status tiketnya “progress” maka tiket akan ditampilkan kepada pengguna.



Gambar 3.20 Pemotongan “Status PIC Progress”

- (8) Pemotongan “status PIC wait approval” ditampilkan pada gambar 3.21, gambar tersebut memiliki arti sintaks apabila pic dalam tiket proyek sama dengan pengguna yang login dan status tiketnya “wait approval” maka tiket akan ditampilkan kepada pengguna.



Gambar 3.21 Pemotongan “Status PIC Wait Approval”

- (9) Pemotongan “status PIC closed” ditampilkan pada gambar 3.22, gambar tersebut memiliki arti sintaks apabila pic dalam tiket proyek sama dengan pengguna yang login dan status tiketnya “closed” maka tiket akan ditampilkan kepada pengguna.



Gambar 3.22 Pemotongan “Status PIC Closed”

- (10) Setelah slicing selesai dibuat, maka data slicing dapat ditampilkan kedalam view Appsheet oleh tim front end dan menyesuaikan pengalaman pengguna.

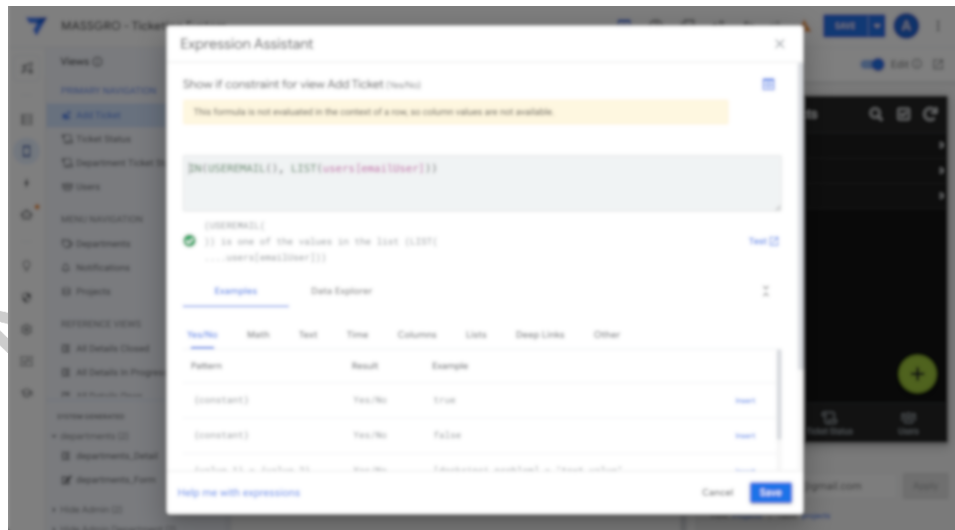
3.2.5 Implementasi Sistem Peran Pengguna

Sistem peran pengguna memungkinkan data atau halaman dapat dilihat sesuai dengan departemen tujuannya, untuk menghindari adanya kesamaan tiket. Sistem ini penting diterapkan pada aplikasi “*Ticketing System*” karena aplikasi ini akan digunakan oleh banyak pengguna dari berbagai departemen. Dengan demikian, menu dapat ditampilkan berdasarkan penggunanya. Implementasi sistem peran pengguna akan dijelaskan sebagai berikut ini.

- (1) Halaman “Add Ticket”

Penerapan peran pengguna pertama berada di halaman ini. Halaman “add ticket” ditampilkan pada gambar 3.23, halaman tersebut nantinya akan digunakan untuk menambahkan data tiket masalah. Dengan demikian, halaman ini hanya boleh dilihat oleh karyawan. Oleh karena itu,

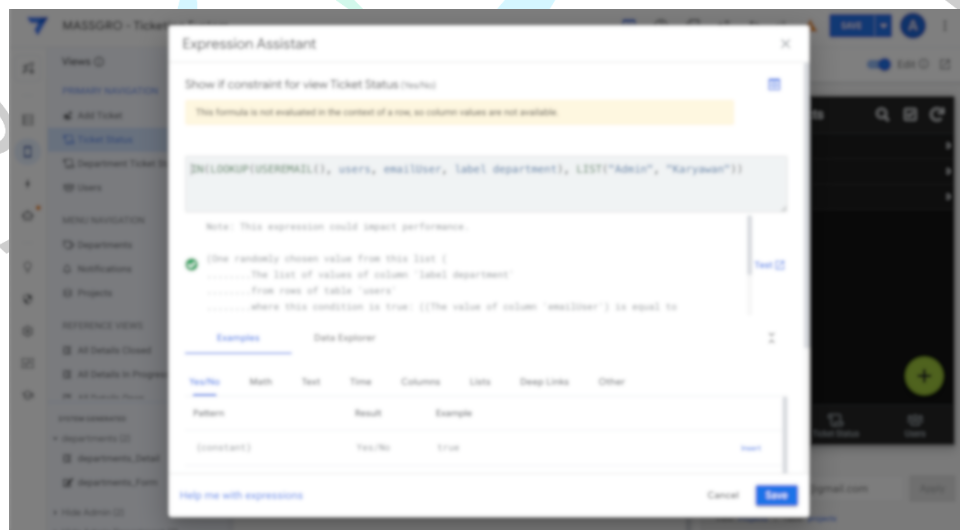
Praktikan membuat sintaks dimana ketika email pengguna yang login sama dengan email pada tabel "user", maka halaman ini dapat ditampilkan.



Gambar 3.23 Halaman "Add Ticket"

(2) Halaman "Ticket Status"

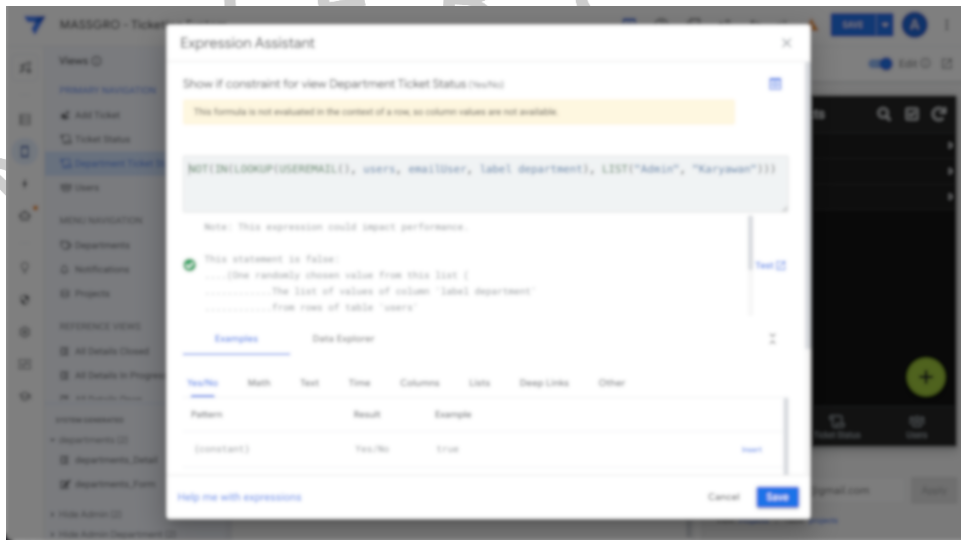
Penerapan selanjutnya berada pada halaman "ticket status". Halaman ini ditampilkan pada gambar 3.24, Halaman ini akan digunakan untuk melihat seluruh data tiket apa saja yang ada berdasarkan statusnya. Oleh karena itu, hanya admin yang diperbolehkan untuk melihat halaman ini.



Gambar 3.24 Halaman "Ticket Status"

(3) Halaman “Departement Ticket Status”

Halaman ini ditampilkan pada gambar 3.25, halaman ini sama seperti halaman sebelumnya. Namun, halaman ini dibuat khusus untuk pengguna departemen. Dengan demikian, hanya pengguna dengan departemen yang dapat melihat halaman ini.



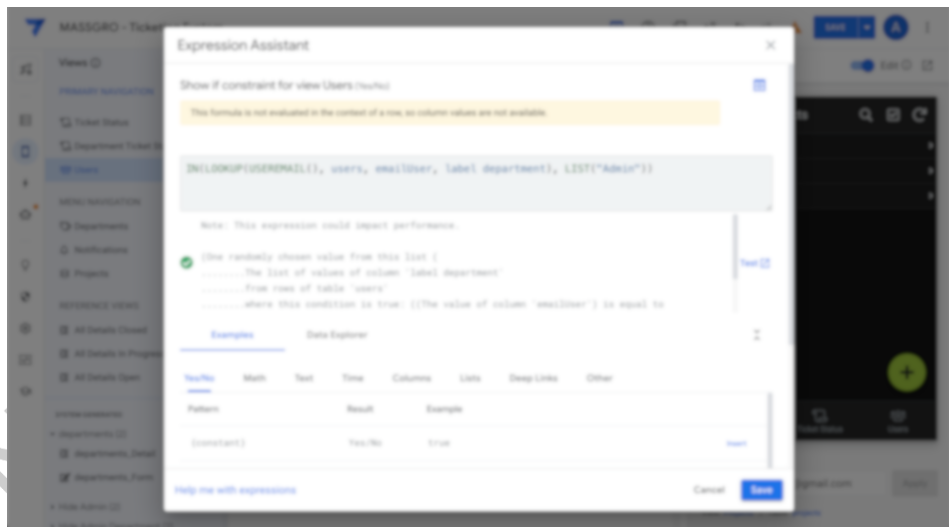
Gambar 3.25 Halaman “Departemen Ticket Status”

(4) Halaman “Users”

Halaman ini ditampilkan pada gambar 3.26, halaman ini berisi data-data penting “user” yang berhubungan dengan sistem user role ini dan hanya pengguna tertera yang diperbolehkan menggunakan aplikasi ini. Tanpa adanya halaman “user”, maka dapat dipastikan bahwa aplikasi ini tidak dapat berjalan sebagaimana semestinya. Dengan demikian, halaman ini hanya dapat dilihat oleh admin.

(5) Halaman “Departements”

Halaman “departements” ditampilkan pada gambar 3.27, halaman ini merupakan departemen yang menampung pengguna. Pada aplikasi ini, departemen dimiliki oleh setiap pengguna. Dengan demikian, departemen dapat terhubung dan digunakan sebagai acuan pemotongan data. Halaman ini hanya dapat dilihat oleh admin, sehingga admin bisa memiliki akses penuh terhadap departemen seperti menambah, mengubah dan menghapus departemen.



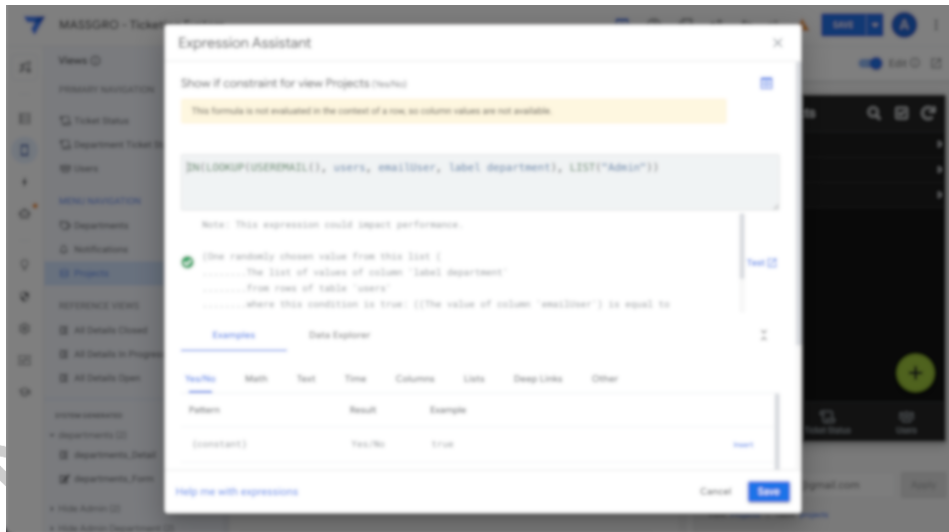
Gambar 3.26 Halaman “Users”



Gambar 3.27 Halaman “Departments”

(6) Halaman “Projects”

Halaman “projects” ditampilkan pada gambar 3.28, halaman ini digunakan untuk membuat proyek baru dan memasukkan PIC pada proyek tersebut. Dengan demikian, PIC tersebut bisa dipanggil berdasarkan idnya dan dihubungkan kepada tabel tiket.

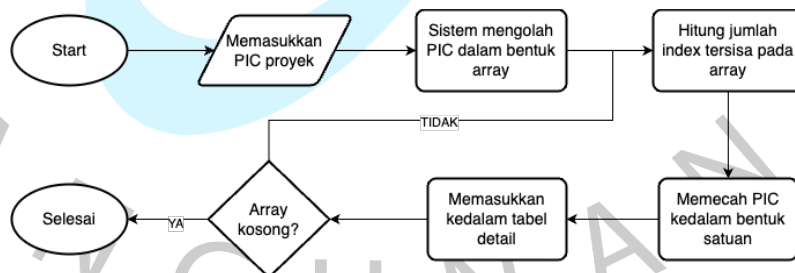


Gambar 3.28 Halaman “Projects”

3.2.6 Implementasi Multi PIC

Pada saat penambahan proyek didalam aplikasi “Ticketing System”, dibutuhkan PIC dalam jumlah banyak, karena orang orang tersebut bertanggung jawab pada proyek yang akan dibuat. Dengan demikian, dibutuhkan *form input* PIC dalam jumlah banyak yang bisa menampung pengguna dari beberapa departemen, sehingga pengguna tersebut bisa mengakses kedalam proyek tersebut. Rangkaian pembuatannya akan dijelaskan sebagai berikut ini.

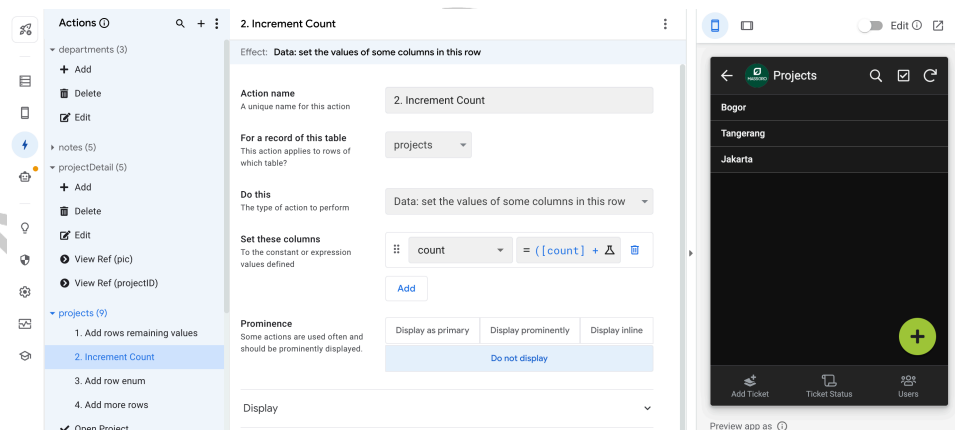
(1) Diagram alir untuk kasus multi PIC ditampilkan pada gambar 3.29.



Gambar 3.29 Diagram Alir Multi PIC

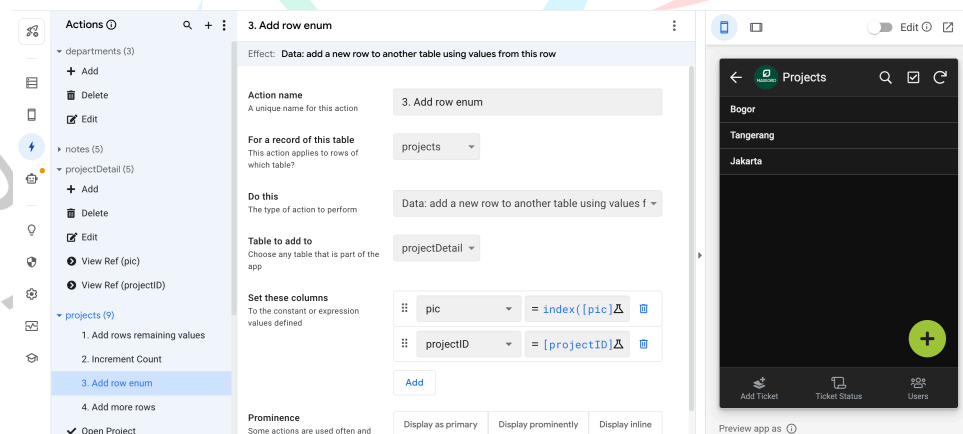
(2) Pada gambar 3.30 ditampilkan gambar pembuatan multi PIC, tahapnya adalah dengan membuat “Actions” pada aplikasi. Perintah yang pertama, ketika “Admin” memilih PIC untuk proyek maka akan dihitung berapa jumlah PIC nya. Jumlah tersebut akan dimasukkan kedalam kolom “Count”

yang ada di tabel “project”. Dengan demikian, kolom tersebut sebagai acuan jumlah untuk PIC yang ada pada proyek tersebut, kolom ini nantinya juga akan digunakan untuk memisahkan PIC dalam “array” dari tabel “project” kepada tabel “projectDetail”.



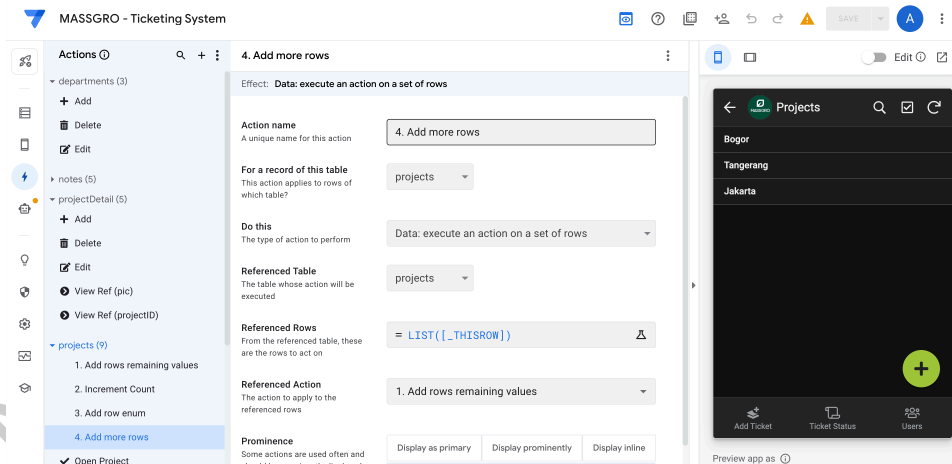
Gambar 3.30 Membuat Kolom Penjumlahan PIC

- (3) Gambar 3.31 adalah proses memecah array dengan memilih kolom apa yang akan dimasukkan kedalam tabel “projectDetail”. Kolom tersebut diantaranya adalah “pic” dan “projectID”. Hasil dari array yang sudah dipecah akan dimasukkan satu satu kedalam tabel “projectDetail”.



Gambar 3.31 Memecah Array Kolom “pic”

- (4) Tahap terakhir adalah dengan membuat sistem perulangan untuk memecah array hingga selesai. Jadi, tabel “projectDetail” akan terus diisi selama array pada kolom “pic” masih tersisa. Pembuatannya ditampilkan pada gambar 3.32.



Gambar 3.32 Sistem Perulangan Kolom

3.2.7 Pengujian Kotak Putih

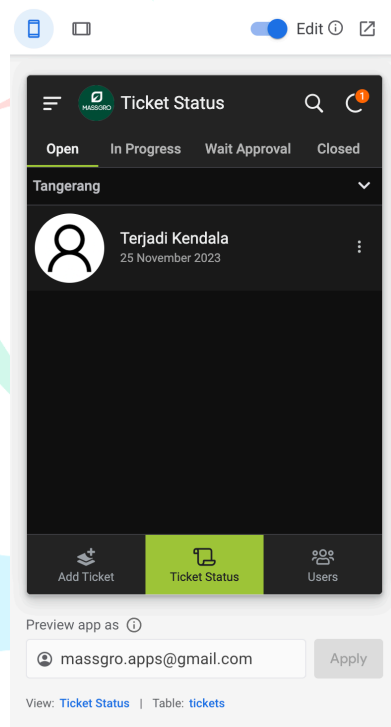
Pengujian kotak putih (*whitebox*) merupakan metode untuk dilakukan pengujian algoritma dan fungsionalitas sebuah aplikasi. Tahap ini penting dilakukan, karena untuk menguji performa daripada aplikasi tersebut dan aman, serta memiliki fungsi yang sesuai dengan target. Penjelasan terkait pengujian ini akan dijelaskan sebagai berikut ini.

- (1) Pengujian yang pertama dilakukan adalah pengujian untuk pemotongan data. Pemotongan yang diuji yaitu status tiket. Pemotongan ini ditampilkan pada gambar 3.33, Ketika tiket telah berhasil dibuat maka status tiket adalah "open" dan akan berada pada halaman "ticket status open". Skenario pengujian "ticket status" dijelaskan pada tabel 3.1 No.1, skenario ini menjelaskan bagaimana program berjalan ketika input telah berhasil dilakukan.

Tabel 3.1 Skenario Pengujian Ticket Status

No.	Skenario Pengujian	Hasil yang diharapkan	Validasi
1	Mengisi form tiket	Tiket yang terinput akan berstatus <i>open</i> dan ditampilkan pada halaman " <i>ticket status open</i> ".	Valid
2	Pemotongan status " <i>progress</i> "	Setelah tiket disetujui maka status tiket akan menjadi " <i>progress</i> " dan	Valid

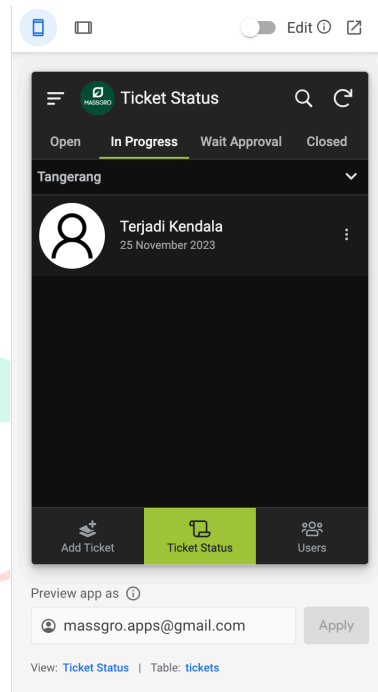
		tampil pada halaman <i>ticket status</i> “ <i>progress</i> ”.	
3	Pemotongan status “ <i>wait approval</i> ”	Setelah melalui status “ <i>progress</i> ”, ketika tiket disetujui akan lanjut ke proses “ <i>wait approval</i> ” dan tampil pada halaman <i>ticket status</i> .	Valid
4	Pemotongan status “ <i>closed</i> ”	Setelah disetujui, karyawan akan menerima notifikasi dan status tiket dapat ditutup oleh karyawan. Sehingga, tiket dapat ditampilkan pada halaman “ <i>ticket status closed</i> .”	Valid



Gambar 3.33 Halaman “Ticket Status Open”

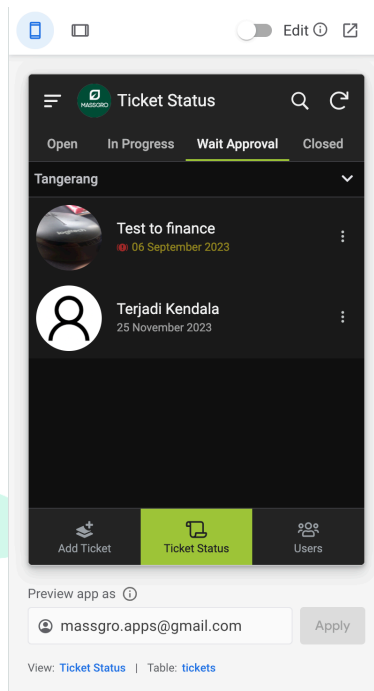
- (2) Pada saat tiket telah diinput oleh karyawan, maka tiket akan menunggu persetujuan oleh departemen yang dituju atau admin. Ketika departemen tujuan telah menyetujui tiket, maka status tiket selanjutnya adalah “*progress*” dan akan ditampilkan pada halaman “*ticket status progress*” seperti gambar 3.34. Skenario pengujian ini dapat dilihat pada tabel 3.1

No.2, skenario tersebut menjelaskan tiket akan di potong kepada halaman “*progress*” ketika telah disetujui.

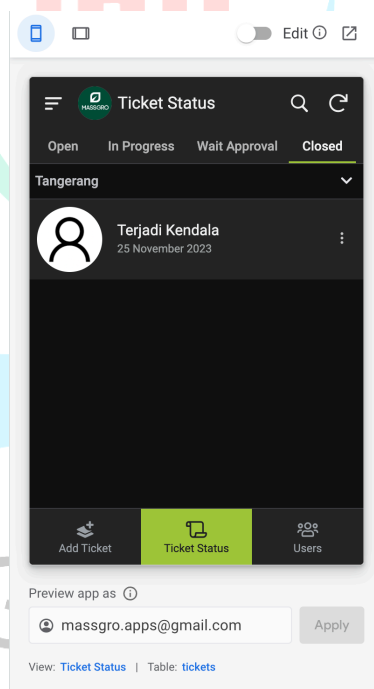


Gambar 3.34 Halaman “Ticket Status Progress”

- (3) Seperti pengujian sebelumnya, status pada tiket akan di potong dan ditampilkan pada halaman “*ticket status progress*”. Pengujian ini ditampilkan pada gambar 3.35 dan dijelaskan pada tabel 3.1 No.3.
- (4) Pengujian halaman “*ticket status closed*” ditampilkan pada gambar 3.36, status tiket akan ditampilkan pada halaman “*closed*” ketika karyawan telah menutup status tiket. Skenario pengujian ini dapat dilihat pada tabel 3.1 No.4, skenario tersebut menjelaskan bahwa halaman “*closed*” berisi tiket dengan status “*closed*” yang telah berhasil.
- (5) Pengujian kedua yang dilakukan adalah sistem peran pengguna. Pengujian ini dilakukan untuk menguji keamanan daripada aplikasi ini, karena ada kemungkinan pengguna diluar departemen mencoba login. Pengujian terkait sistem peran pengguna pada aplikasi “*Ticketing System*” dijelaskan pada tabel 3.2.



Gambar 3.35 Halaman “Ticket Status Wait Approval”

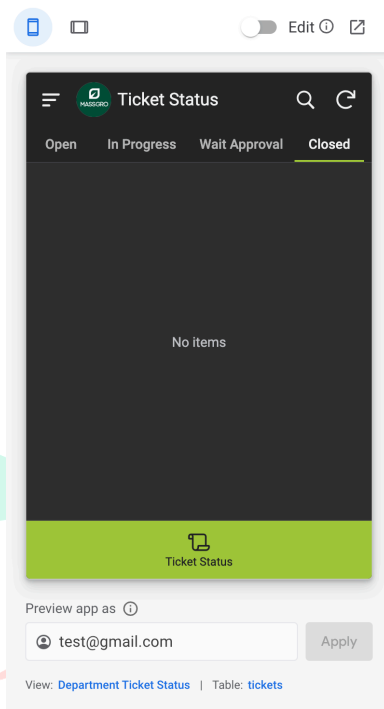


Gambar 3.36 Halaman “Ticket Status Closed”

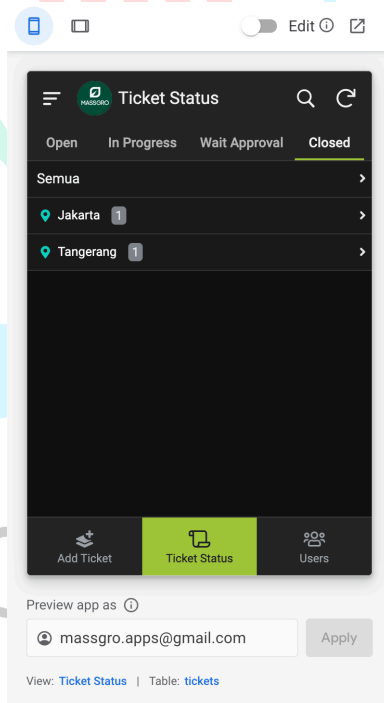
Tabel 3.2 Skenario Pengujian Sistem User Role

No.	Skenario Pengujian	Hasil yang diharapkan	Validasi
1	Halaman "Add Ticket"	Ketika pengguna tidak ada dalam tabel users, maka pengguna tidak dapat menggunakan halaman ini.	Valid
2	Halaman "Ticket Status"	Ketika pengguna yang login bukan merupakan admin, maka pengguna tidak dapat menggunakan halaman ini.	Valid
3	Halaman "Departemen Ticket Status"	Ketika pengguna yang login bukan dari departemen, maka pengguna tidak dapat menggunakan halaman ini.	Valid
4	Halaman "Users"	Ketika pengguna yang login bukan merupakan admin, maka pengguna tidak dapat menggunakan halaman ini.	Valid
5	Halaman "Departement"	Ketika pengguna yang login bukan merupakan admin, maka pengguna tidak dapat menggunakan halaman ini.	Valid
6	Halaman "Projects"	Ketika pengguna yang login bukan merupakan admin, maka pengguna tidak dapat menggunakan halaman ini.	Valid.

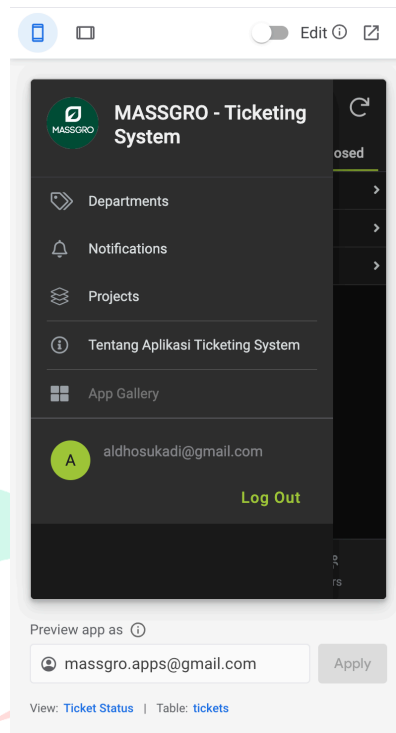
- (6) Pengujian ketiga adalah pengujian tampilan untuk peran pengguna. Pengujian ini ditampilkan pada gambar 3.37. Gambar tersebut menampilkan halaman ketika pengguna bukan salah satu dari karyawan di perusahaan.
- (7) Pengujian untuk halaman admin ditampilkan pada gambar 3.38. Gambar tersebut menampilkan halaman ketika pengguna merupakan admin. peran admin memiliki hak akses penuh, sehingga seluruh menu akan ditampilkan kepada admin. Halaman kedua (*sidebar*) di admin akan ditampilkan pada gambar 3.39



Gambar 3.37 Halaman Peran Tidak Diketahui

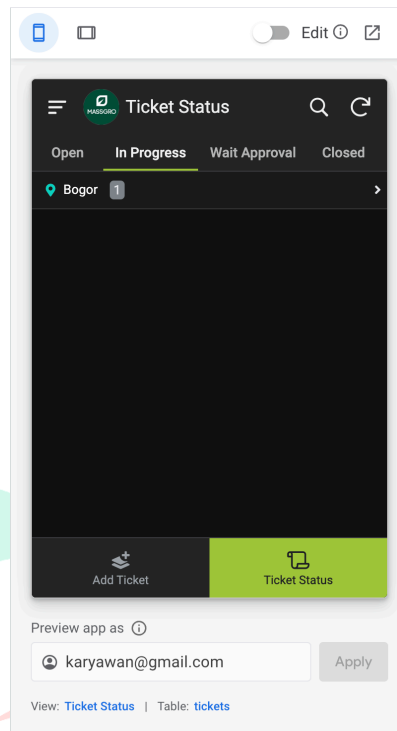


Gambar 3.38 Halaman Peran "Admin"

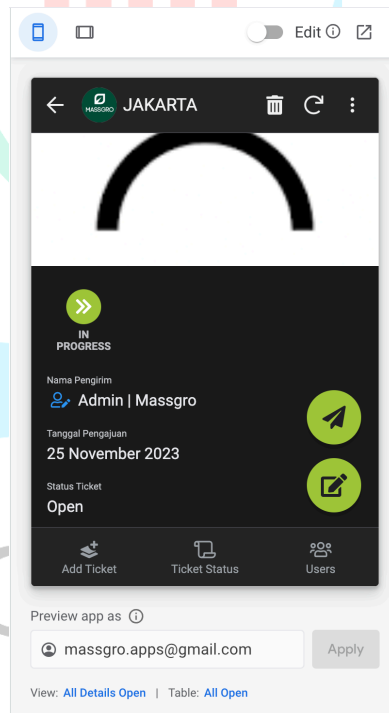


Gambar 3.39 Sidebar Peran "Admin"

- (8) Pengujian untuk halaman karyawan lapangan ditampilkan pada gambar 3.40. Gambar tersebut menampilkan halaman ketika pengguna merupakan karyawan. Peran karyawan memiliki hak akses untuk menambahkan tiket dan melihat status tiket. Dengan demikian, karyawan tidak memerlukan menu yang rumit.
- (9) Pengujian selanjutnya adalah untuk peran departemen. Role ini akan digunakan banyak pengguna dan memiliki berbagai macam departemen, seperti, keuangan, akuntan, teknisi, pemasaran, dll. Halaman peran ini sama seperti pada gambar 3.40, perbedaannya terletak pada hak aksesnya, peran departemen memiliki akses untuk melakukan persetujuan kepada tiket seperti pada gambar 3.41.



Gambar 3.40 Halaman Peran “Karyawan”



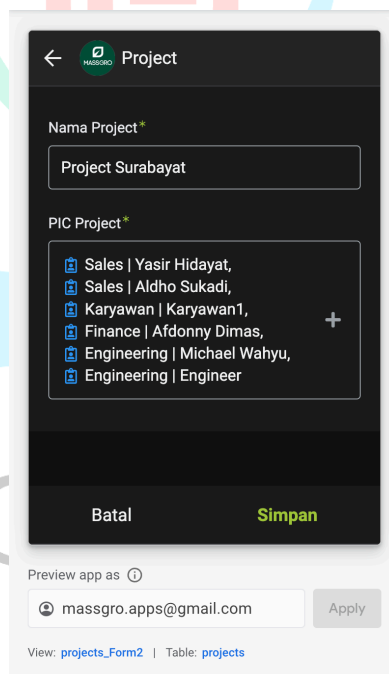
Gambar 3.41 Halaman Peran “Departemen”

(10) Pengujian terakhir adalah untuk multi PIC pada “project”. Penjelasan lebih lengkap terkait pengujian ini dijelaskan pada tabel 3.3.

Tabel 3.3 Pengujian Multi PIC

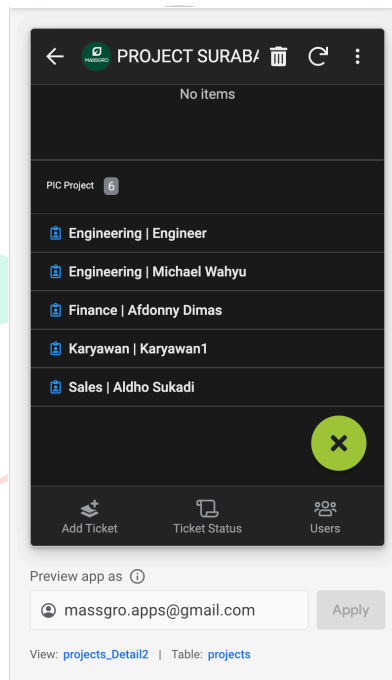
No.	Skenario Pengujian	Hasil yang diharapkan	Validasi
1	Memasukkan PIC dalam jumlah banyak	PIC dapat diinputkan dalam jumlah banyak, sehingga masing masing pengguna dapat melihat proyek yang menjadi tanggung jawabnya	Valid
2	Pengguna dapat melihat tiket status proyek	Ketika pengguna login, maka pengguna dapat melihat tiket status proyek yang berhubungan dengannya	Valid

(11) *Form input* multi PIC ditampilkan pada gambar 3.42, “Admin” dapat memilih seluruh PIC dari departemen. Skenario pengujian ini dapat dilihat pada tabel 3.3 No.1.



Gambar 3.42 Form Input Multi PIC

(12) Data “project” yang telah dibuat, akan berisi PIC yang sudah dimasukkan sebelumnya. PIC ini akan menerima notifikasi apabila ada karyawan yang membuat tiket. Halaman “Project” ditampilkan pada gambar 3.43. Skenario pengujian ini dapat dilihat pada tabel 3.3 No.2



Gambar 3.43 Halaman “Project”

(13) Seluruh tahapan pengujian telah selesai dilakukan dan aplikasi “*ticketing system*” dapat digunakan pada karyawan. Setelah tahap pengujian selesai, hasil kerja Praktikan dipresentasikan pada pembimbing lapangan kerja dan mendapat hasil yang cukup baik dan apresiasi dari pembimbing. Mengingat video pembelajaran terkait “Appsheets” masih sedikit di luar, tetapi Praktikan dapat meneruskan hingga tahap perilsan.

3.3 Kendala Yang Dihadapi

Pelaksanaan program KP di perusahaan pertanian modern yang dijalankan praktikan memiliki beberapa kendala. Diantaranya sebagai berikut ini.

(1) Penggunaan platform Appsheets masih jarang dikenal dan memiliki sedikit pembelajaran di internet. Sehingga, praktikan kesulitan dalam menggunakan platform tersebut.

- (2) Pembimbing kerja memiliki keterbatasan waktu. Dengan demikian, praktikan kesulitan dalam mengatur waktu untuk menemui pembimbing kerja.
- (3) Keterbatasan arahan yang diberikan oleh pembimbing kerja menjadi tantangan bagi praktikan untuk mengembangkan aplikasi tersebut. Hal ini juga dipengaruhi oleh keterbatasan waktu pada poin no.2.
- (4) Aplikasi yang dibuat memiliki algoritma yang cukup rumit bagi praktikan. Oleh karena itu, praktikan membutuhkan waktu yang cukup lama dalam mengembangkan aplikasi ini.

3.4 Cara Mengatasi Kendala

Berdasarkan kendala diatas, berikut ini adalah cara praktikan dalam mencari solusi atas masalah tersebut.

- (1) Praktikan melakukan belajar mandiri dan mencoba keseluruhan daripada platform ini. Praktikan mencoba seluruh fitur yang ada pada platform Appsheet untuk mengetahui lebih dalam.
- (2) Praktikan memilih dan mengatur waktu terbaik untuk menghubungi pembimbing kerja.
- (3) Praktikan mempelajari bagaimana untuk terus aktif dalam berkomunikasi kepada pembimbing kerja.
- (4) Praktikan mempelajari lebih dalam tentang perancangan sistem yang akan dibuat. Dengan demikian, alur akan terbaca dan dapat diterapkan.

3.5 Pembelajaran Yang Diperoleh Dari Kerja Profesi

Dalam pelaksanaan mata kuliah KP di perusahaan pertanian modern, praktikan memperoleh pembelajaran yang banyak. Pembelajaran tersebut diantaranya sebagai berikut ini.

- (1) Praktikan mempelajari cara menggunakan platform Appsheet. Ini merupakan pengalaman yang berharga karena ini adalah pertama kalinya praktikan menggunakan platform tersebut.
- (2) Praktikan mempelajari untuk menghargai waktu. Ketika mengatur waktu dengan baik, maka dapat dipastikan aplikasi dapat selesai tepat pada waktunya seperti yang diharapkan.
- (3) Praktikan mempelajari bagaimana berada di posisi dunia kerja. Pada posisi ini, praktikan diharuskan untuk terus aktif bertanya dan bersosialisasi. Dengan demikian, praktikan dapat terus mengembangkan kemampuan berbicaranya didepan umum.
- (4) Praktikan memahami alur aplikasi yang cukup rumit, sehingga kemampuan praktikan dapat lebih terasah untuk terus bersaing dalam dunia kerja.