

## BAB III

### PELAKSANAAN KERJA MAGANG

#### 3.1 Kedudukan dan Koordinasi

Dalam program kerja magang yang dilaksanakan di PT Telkom ini, peserta magang diposisikan sebagai *Junior Developer* di dalam lingkup departemen *Human Capital Business Partner*. Peserta kerja magang ditugaskan untuk mengembangkan aplikasi berbasis web. *Project* yang dikembangkan berfungsi sebagai monitor dan sarana untuk *booking* ruangan yang ada di dalam PT Telkom Indonesia sesuai dengan kebutuhan pengguna (*user*) dari aplikasi tersebut.

Dalam pengerjaannya, peserta magang dibimbing dan diarahkan secara langsung oleh Bapak Aceng Komarudin (NIK 876843) selaku *Officer 2 System Deployment* di departemen *Human Capital Business Partner*. Pemberian tugas dan agenda kerja magang diberikan langsung oleh Bapak Aceng Komarudin. Pemberian dan pengarahan tugas dilaksanakan dengan cara *briefing* via *online meeting* yang dihadiri oleh peserta magang dan juga Bapak Aceng Komarudin sendiri.

Untuk komunikasi terkait bimbingan dalam pengerjaan *project*, peserta magang dapat menghubungi langsung Bapak Aceng Komarudin melalui whatsapp atau merencanakan *online meeting* melalui *google meet*. Setelah melaksanakan tugas dan agenda yang diberikan, peserta magang melakukan pelaporan atas selesainya tugas dan agenda yang telah diberikan. Pelaporan dilaksanakan via *online meeting*, serta penyampaian revisi jika didapati ketidaksesuaian hasil yang diharapkan. Pertemuan tersebut biasanya disertai dengan pemberian agenda selanjutnya yang harus dilakukan.

## 3.2 Tugas dan Uraian Kerja Magang

### 3.2.1 Tugas yang Dilakukan

Selama melaksanakan praktik kerja magang, di project aplikasi booking ruangan di PT Telkom Indonesia, peserta magang dilibatkan dalam pembuatan dan perancangan aplikasi sesuai *roadmap* dan *sprint* pengembangan project. Keterlibatan peserta magang dapat di uraian seperti pembuatan dan perancangan alur (*flowchart*) dari aplikasi, desain dan pembuatan wireframe diagram sebagai gambaran dan acuan pembentukan aplikasi, perancangan struktur database aplikasi, pembuatan database, perancangan *logic* aplikasi, penulisan code dari aplikasi hingga perealisasi rancangan menjadi aplikasi berbasis web.

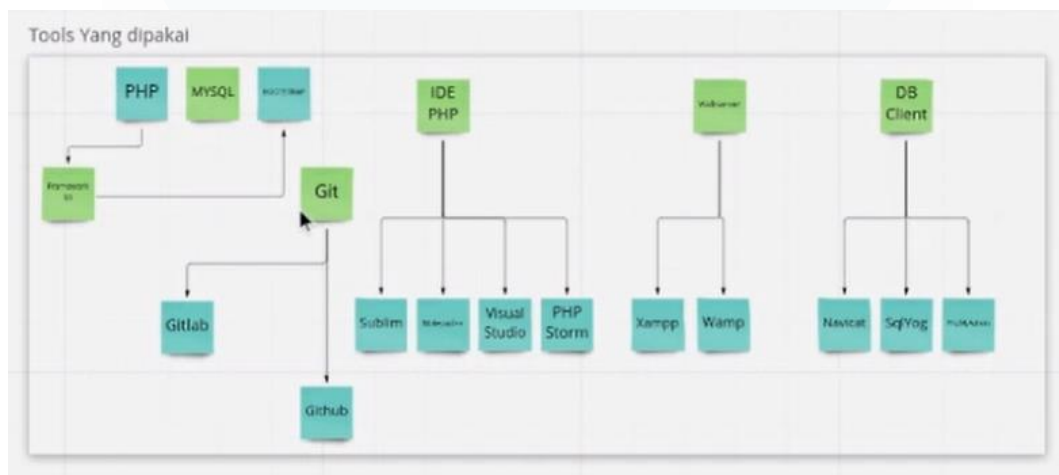
### 3.2.2 Uraian Kerja Magang

Tabel 3. 1 Tabel uraian kerja magang

NO	AKTIVITAS	TANGGAL MULAI	TANGGAL SELESAI
1	<b>Pengenalan Terkait Framework Yii</b>		
	Controllers	4 Juli 2022	4 Juli 2022
	Models	4 Juli 2022	4 Juli 2022
	Views	4 Juli 2022	4 Juli 2022
2	<b>Studi Kasus</b>		
	Pembuatan menu penjualan	5 Juli 2022	7 Juli 2022
	Form penjualan	8 Juli 2022	27 Juli 2022
	Page Keranjang	28 Juli 2022	29 Juli 2022
3	<b>Requirement Booking Apps</b>		
	Pembuatan Flowchart	4 Agustus 2022	24 Agustus 2022
	Pembuatan Wireframe aplikasi	8 Agustus 2022	15 Agustus 2022
	Pembuatan Wireflow	9 Agustus 2022	16 Agustus 2022
4	<b>Design databases</b>		
	Pembuatan ER Diagram	25 Agustus 2022	14 September 2022
	Generate Diagram ke databases	15 September 2022	19 September 2022
	Migrasi database	20 September 2022	26 September 2022
5	<b>Development Aplikasi</b>		
	Login Page	27 September 2022	18 Oktober 2022
	Pembuatan page master data	19 Oktober 2022	6 Desember 2022
	Booking Ruangan	7 Desember 2022	22 Desember 2022
	Page Approval	22 Desember 2022	-
	Page Informasi	-	-

Pada tampilan tabel 3.1 uraian kerja magang, dapat digambarkan tugas yang dilakukan berfokus pada pembuatan dan juga pengembangan aplikasi berbasis web yang berfungsi untuk memmanagement dan monitoring proses pemesanan ruangan di kantor PT Telkom Indonesia. Framework yang digunakan dalam pembuatan project ini adalah framework Yii 2. Alasan menggunakan framework tersebut dikarenakan *project* dalam perusahaan telah menggunakan Yii Framework sebagai sarana pengembangan source code nya. Pembuatan project tersebut memiliki beberapa alur yang harus dilakukan sesuai roadmap dan sprint pengembangan *project*. Alur dan tahapan pembuatan *project* dapat diuraikan seperti berikut.

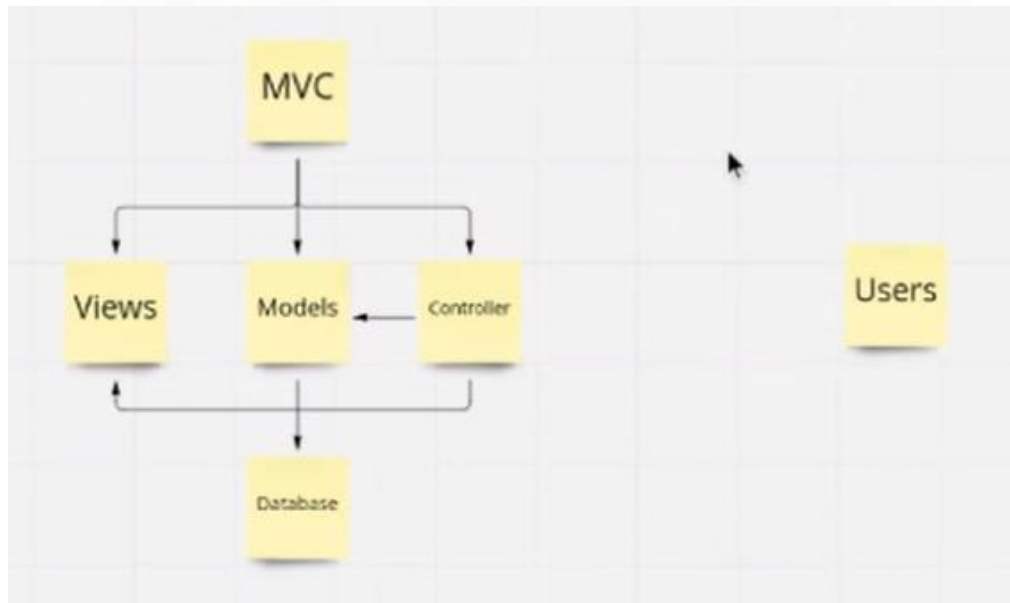
### 3.2.2.1 Pengenalan Terkait Framework Yii



Gambar 3. 1 Briefing mengenai project

Setelah mendapatkan informasi mengenai *supervisor* dan kontak nya, peserta magang diminta untuk menghadiri *online meeting* yang membahas mengenai *project* yang akan dikerjakan. Seperti pada gambar 3.1 yang merupakan dokumentasi *briefing* mengenai *project* pertemuan ini membahas tentang aplikasi apa yang akan dikembangkan serta menjelaskan fungsi dan fitur yang akan ada didalamnya. Kemudian bahasan selanjutnya mengenai gambaran umum dan penjelasan atas framework yang akan digunakan [10]. Hal ini dilakukan agar

peserta magang mengerti tentang *flow* dan alur data yang terjadi didalam framework yang digunakan. Dalam pembahasan mengenai framework, pembimbing menjelaskan mengenai alur (flow) data yang terbagi atas *Model, View, Controller* (MVC), serta *package* yang digunakan didalam Yii Framework.



Gambar 3. 2 Gambaran mengenai flow MVC

Seperti di dokumentasi gambar 3.2 dalam penjelasan yang dilakukan, poin poin yang ditekankan dibagi menjadi tiga bagian, yakni *Model, View, Controller* (MVC). Poin tersebut didahulukan dan diutamakan karena merupakan hal dasar yang harus dimengerti dalam framework Yii. Dari hasil pembahasan, dapat dijelaskan Model merupakan dasar atas kelas kelas yang ada di dalam Framework Yii. Model mempresentasikan atau mewakili bagaimana metode atau atribut akan dipanggil atau dengan kata lain, model mewakili sebuah objek. *View* adalah skrip yang berisi elemen atas tampilan antarmuka dengan pengguna. Skrip yang ada di bagian *view*, akan langsung mengatur dan mengubah bagaimana tampilan dari program web yang dibuat. *Controller* dapat digambarkan sebagai kelas turunan dari sebuah model, karena fungsi fungsi yang ada di controller dapat digunakan jika ada *request* (action) dari model maupun *view*.

Selain penjelasan tentang aplikasi, di pertemuan *supervisor* (Bapak Aceng Komarudin) juga menjelaskan bagaimana peserta magang dan pembimbing dapat saling berkoordinasi dan berkomunikasi satu sama lain mengenai pengerjaan *project*. Setelah melakukan pertemuan bersama Bapak Aceng Komarudin, peserta magang mempelajari metode, bahasa, dan package dari Yii Framework melalui dokumentasi di web resmi Yii Framework, jurnal, modul dan juga forum diskusi online.

### 3.2.2.2 Studi Kasus

Sebelum terjun ke project perusahaan, peserta magang diminta membuat program latihan yaitu program kasir sebagai simulasi dari pengerjaan *project* aslinya, sehingga peserta magang dapat mengenali lebih dalam pengerjaan menggunakan Yii Framework. *Project* latihan yang dibuat mengangkat tema program kasir berisi penjualan barang yang memiliki database dan juga kode barang sebagai id nya. Simulasi tersebut mirip dengan *project* yang akan dibuat, mulai dari struktur database maupun flow aplikasinya. Pengerjaan studi kasus dilaksanakan selama sebulan di awal kegiatan kerja magang.

Berdasarkan rancangan dan diskusi atas program studi kasus ini, program akan memiliki dua role yakni *admin* dan *user*. User akan berperan seperti pembeli atas barang yang diperjualbelikan. Sedangkan admin dapat mengatur (create, update, delete) atas data penjual, data barang hingga harga barang yang ditawarkan. Pengerjaan program latihan *supervisor* secara langsung membina dan membimbing pengerjaan pembuatan. Pelaporan update dan juga hasil juga selalu disampaikan kepada Bapak Aceng Komarudin.

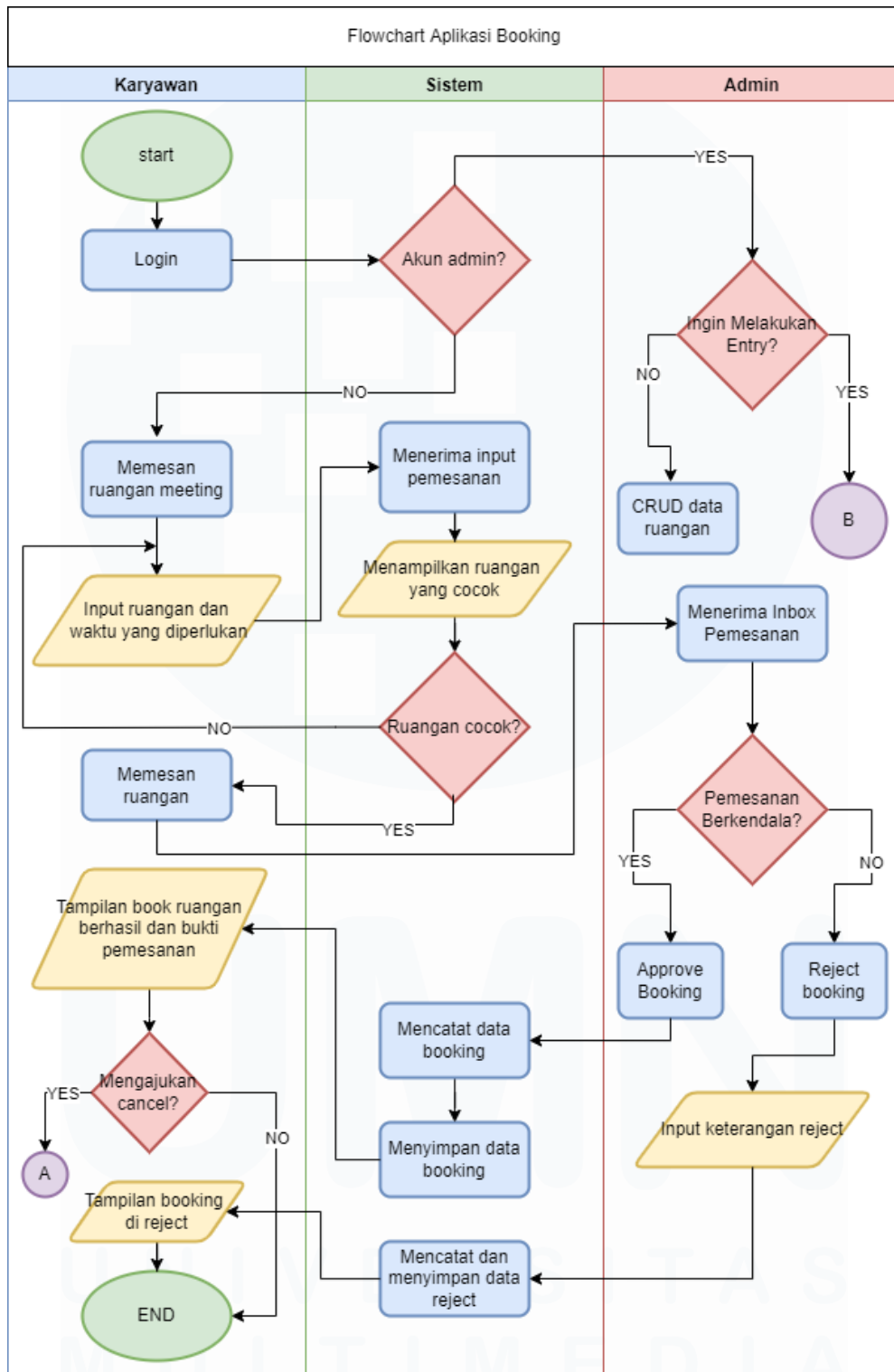
Tahapan tahapan yang telah dilakukan dalam studi kasus ini peserta magang diminta untuk membuat menu penjualan, sebagai halaman awal dimana program menampilkan menu (landing page). Kemudian untuk tugas selanjutnya diminta untuk membuat *form* penjualan, agar pemahaman atas fungsi dan *package* Framework Yii dapat digunakan secara langsung oleh peserta magang. Contoh

penerapan yang ada pada saat membuat *form* penjualan adalah bagaimana pengembang akan mengenali halaman edit *form* yang disediakan oleh framework Yii, pemanggilan fungsi yang memiliki berbagai macam jenis seperti *textbox*, *dropdown*, dan *date time picker*.

Kemudian untuk tahapan pembuatan halaman keranjang, peserta magang dapat memahami bagaimana alur data yang terjadi antara *Model*, *View*, *Controller* (MVC) dan pengiriman data antar tabel tabel yang ada di database. Di dalam halaman keranjang, terjadi pemusatan data yang ada didalam tabel tabel database program. Hal tersebut terjadi karena didalam halaman keranjang memuat informasi penjual, informasi barang, informasi pembeli, yang letaknya diluar tabel keranjang sehingga akan meningkatkan pemahaman data yang terkoneksi melalui *Foreign Key*. Dalam studi kasus yang dilakukan, peserta magang telah menyelesaikan beberapa page dengan fiturnya seperti page pembuatan menu penjualan, *form* penjualan, page keranjang pembelian dan juga page informasi penjual.

### 3.2.2.3 Requirement Booking App

Setelah waktu pengerjaan studi kasus berakhir, *supervisor* menugaskan untuk masuk ke tahapan pengerjaan *project* perusahaan. Sebagai langkah awal pembuatan program, *supervisor* menugaskan untuk merancang flowchart dari *project* tersebut, agar *flow* atau alur dari proses program yang akan dibuat lebih mudah disampaikan. Proses pembuatan flowchart dilakukan di draw.io agar *worksheet* yang digunakan dapat di akses pula oleh supervisor. Terdapat beberapa kali pembuatan flowchart, karena setelah didiskusikan oleh supervisor terjadi beberapa pembenaran dan revisi alur didalam flowchart. Hasil final flowchart dapat ditampilkan sebagai berikut.



Gambar 3. 3 Flowchat Project Booking App

Dapat dilihat pada gambar 3. 3 flowchart project booking app, terdapat tiga segmen interaksi di dalam aplikasi yakni antara karyawan, sistem dan admin. Karyawan merupakan user yang dapat mengakses (menggunakan) aplikasi bermodalkan hak akses login dengan akun internal perusahaan. Akses user karyawan dapat memesan (booking) ruangan melalui aplikasi serta *monitoring* ruangan yang tersedia atau yang sudah digunakan.

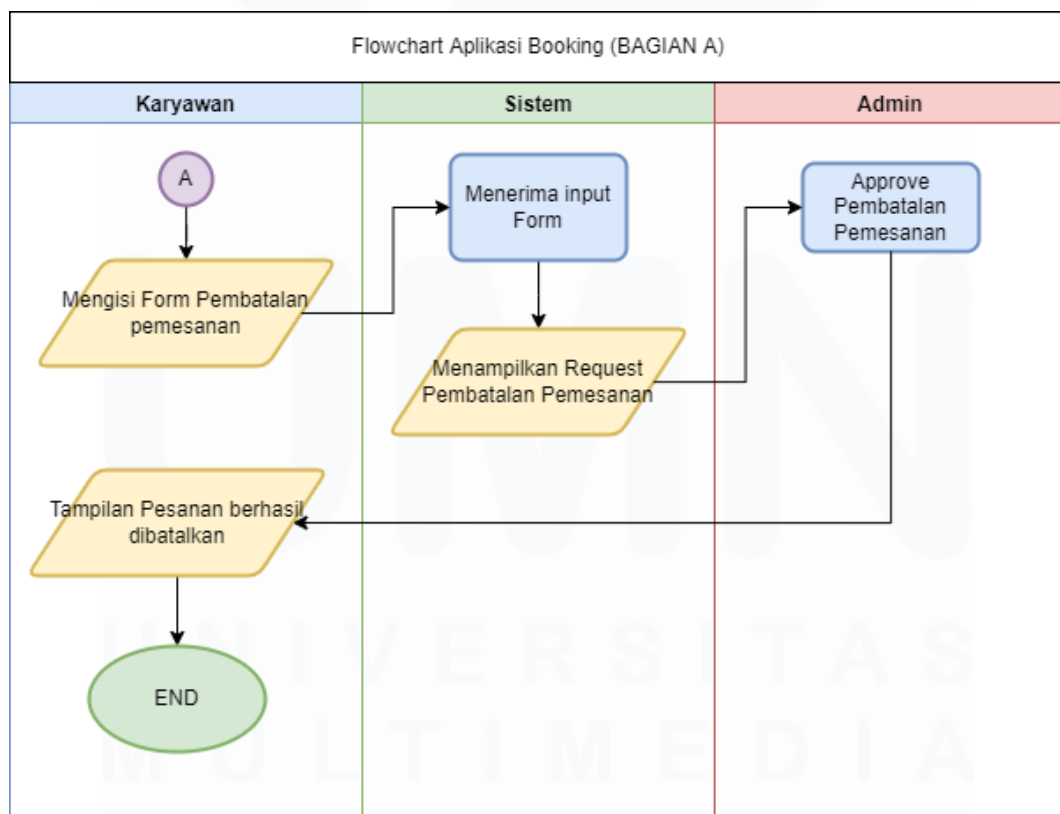
Kemudian di *logic* sistem, memiliki peranan untuk mengorganisir dan memproses permintaan yang telah di *input*. *Input* akan diproses sesuai pemrograman yang telah dibuat untuk menghasilkan *output* sebagai contoh, pemesanan telah berhasil atau pembatalan pemesanan yang sudah dilakukan. Dan yang terakhir terdapat admin, yang sifatnya seperti akun karyawan atau akun user biasa, namun memiliki hak istimewa untuk menambah ganti daftar ruangan, pesanan dan juga *approve* maupun *reject* pemesanan yang diminta.

Untuk alur secara keseluruhan sesuai dengan flowchart yang ditampilkan, dapat dilihat proses berawal dari pengguna (user) yang melakukan *login* kedalam program. Lalu, sistem akan memilah apakah akun yang masuk akun karyawan, atau akun admin. Jika akun tersebut akun karyawan, maka akan diarahkan ke proses pemesanan ruangan. Proses pemesanan dilakukan diawali dengan proses input kebutuhan user seperti tanggal, waktu dan kapasitas ruangan. Kemudian sistem akan menerima input tersebut dan menampilkan ruangan yang sesuai dengan kebutuhan user. Jika cocok maka user dapat memesan ruangan tersebut dan menunggu proses *approve* oleh admin. Jika tidak maka user diberi pilihan untuk *input* ulang kebutuhan ruangan hingga sesuai atau menghentikan proses pemesanan. Apabila proses pemesanan sudah di *approve* oleh pihak admin, karyawan kembali diberi pilihan apakah pemesanan dibatalkan (*cancel*) atau tidak. Jika pesanan tidak dibatalkan maka proses pemesanan ruangan selesai. Sebaliknya jika pemesanan dibatalkan maka proses akan dilanjutkan di flowchart A yang akan dijelaskan selanjutnya. Kemudian alur yang terjadi jika *login* menggunakan akun admin. Hak akses admin ada beberapa hal, yakni dapat memesan ruangan dengan



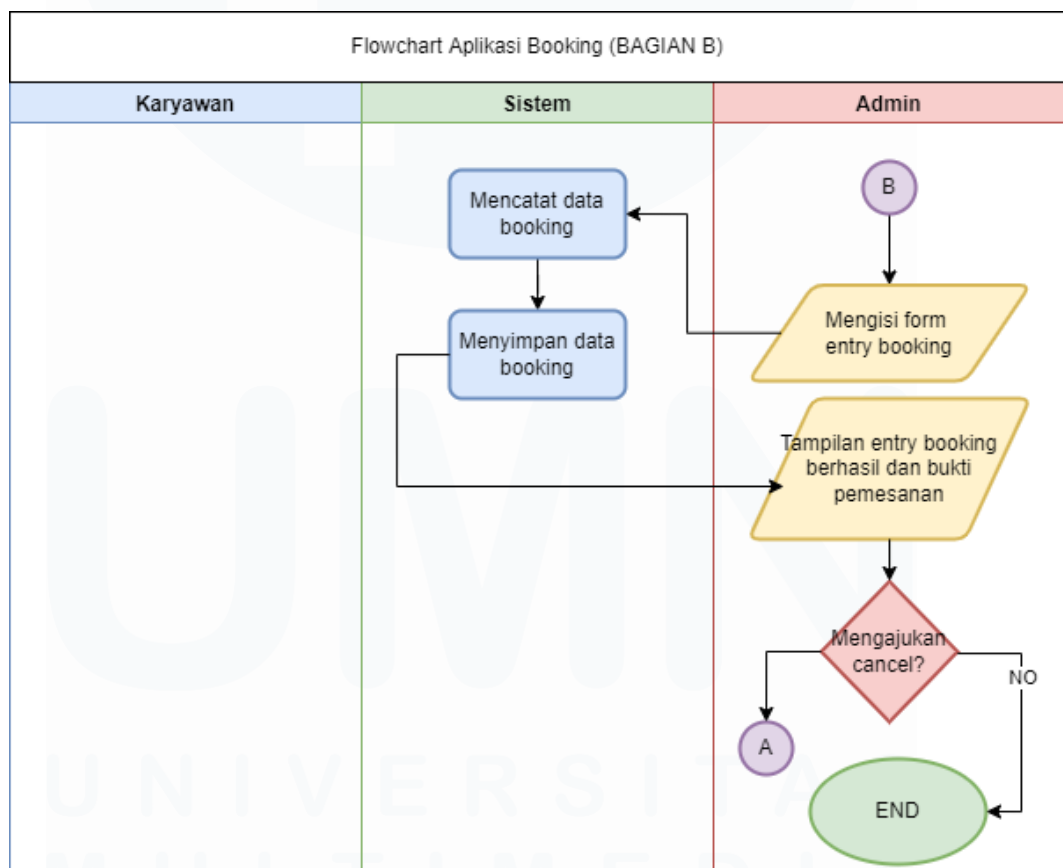
akses *entry* yakni pemesanan ruangan tanpa menunggu di approve pemesanannya, dapat *create update delete* list ruangan, *approve* dan *reject* pemesanan ruangan, dan dapat *monitoring history reject* pemesanan.

Jika berdasarkan alur dari flowchart diatas, setelah *login*, admin diberi pilihan yang dia miliki untuk melakukan beberapa akses tersebut. Sebagai contoh jika admin menerima pemesanan dari karyawan maka admin diberi pilihan untuk *reject* atau *approve* pemesanan tersebut. Jika pesanan diterima maka akan tampil keterangan bahwa pesanan telah berhasil, sebaliknya jika pesanan ditolak maka akan muncul pesan tidak diterima bersama keterangan penolakan. Kemudian jika admin ingin melakukan *entry form*, maka proses akan dilanjutkan ke flowchart B yang akan dibahas selanjutnya. Di flowchart sebelumnya terdapat pembagian *section* flowchart yaitu flowchart A dan flowchart B. berikut merupakan tampilan dan penjelasan dari kedua flowchart tersebut.



Gambar 3. 4 Flowchart bagian A

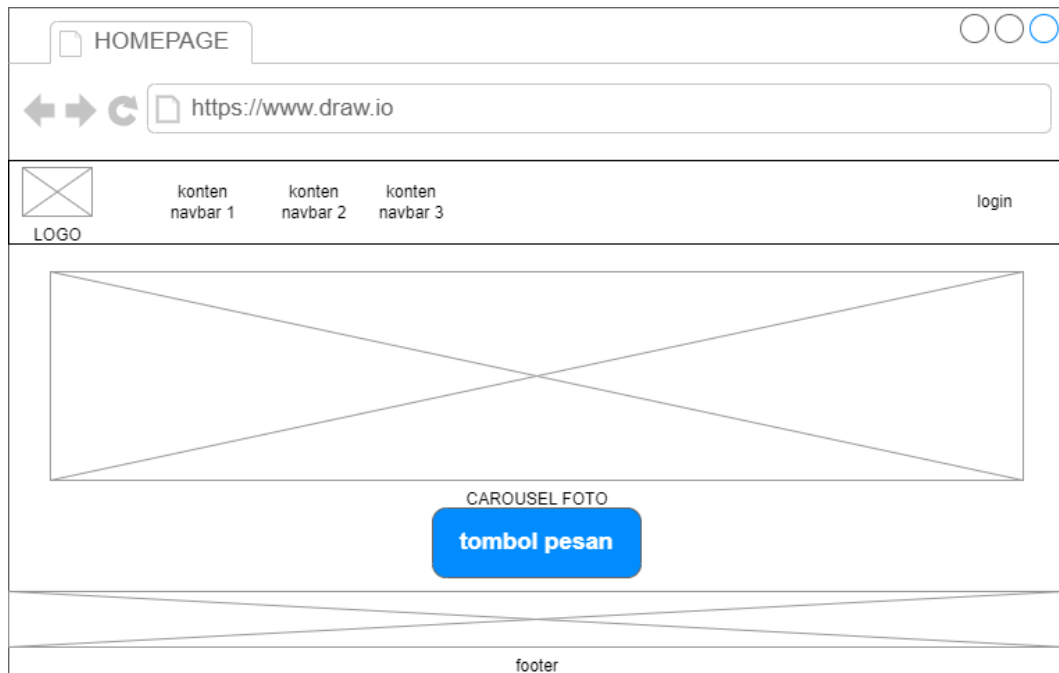
Dapat dilihat dari tampilan flowchart utama, pada gambar 3.4 flowchart bagian A merupakan kelanjutan dari arus aktifitas karyawan jika ingin melakukan pembatalan (cancel) atas pesanan ruangan yang dilakukan. Proses *cancel* sendiri dapat dilakukan sebelum pesanan di *approve* oleh admin. Dari alur aktivitasnya sendiri dapat dilihat karyawan akan diberikan *form* pembatalan pesanan yang berisikan keterangan pembatalan serta nama ruangan yang dibatalkan pemesanannya. Setelah itu, sistem akan menerima input dari *form* dan akan diteruskan ke tampilan admin berupa notifikasi adanya pembatalan. Permintaan pembatalan masuk ke *page* admin dan admin akan menyetujui pembatalan tersebut. Hal yang terjadi disaat admin menyetujui permintaan pembatalan, akan muncul tampilan dibatalkan (di cancel) pada *page history* pesanan karyawan.



Gambar 3. 5 Flowchart bagian B

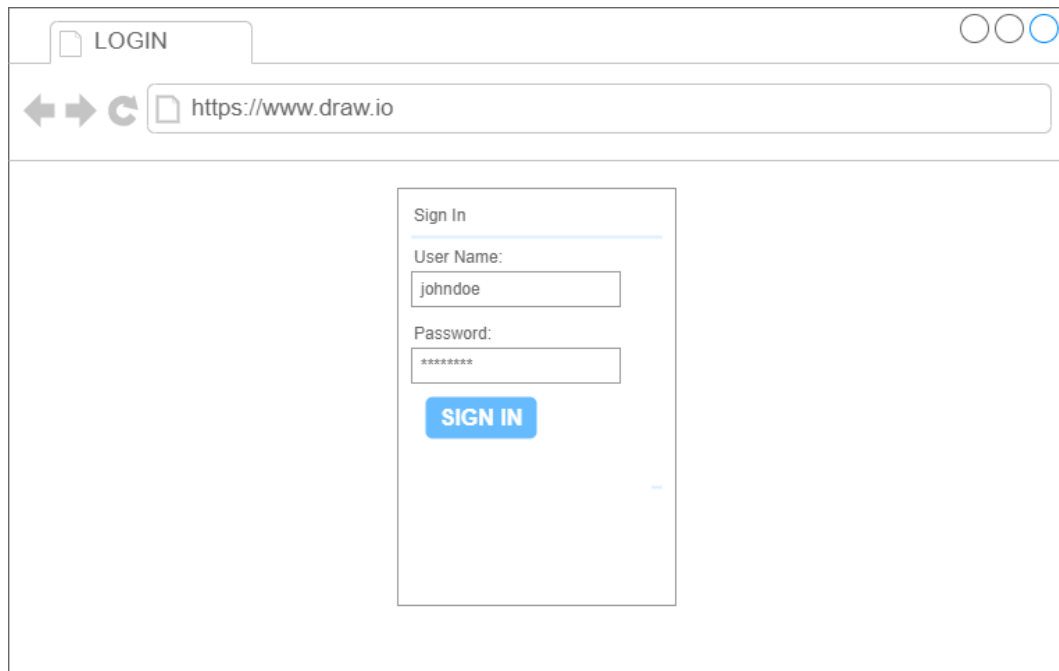
Kemudian pada bagian flowchart yang terakhir yang ada di gambar 3.5, yakni flowchart bagian B, menunjukkan alur ketika admin ingin melakukan proses pemesanan langsung (entry). Pemesanan ini bersifat langsung karena tanpa adanya permintaan (*request*) kepada admin agar pesanan disetujui atau ditolak. Proses yang terjadi pada tahapan ini yakni admin mengisi *form* pemesanan yang disediakan. *Form* yang diberikan hampir sama dengan *form* pemesanan yang diisi oleh karyawan. Pemesanan melalui admin ini dapat memesan untuk diri sendiri maupun atas nama karyawan lain. Setelah pengisian *form* selesai, *input* tersebut akan disimpan dan dicatat oleh sistem, dan sistem akan menampilkan *output* jika pemesanan berhasil. Saat pemesanan telah berhasil, admin berhak mengajukan *cancel* terhadap pesanan yang telah dibuat dengan proses yang sama dengan flowchart A, namun perbedaannya proses pembatalan di *entry form* tidak memerlukan *approve* dari admin. Jika admin memilih untuk tidak melakukan pembatalan maka proses akan selesai.

Setelah menyelesaikan proses pembuatan *flowchart*, peserta magang diminta untuk membuat *wireframe* untuk melanjutkan proses selanjutnya. Sebagai gambaran dan realisasi visual dari hasil pemikiran, peserta magang merancang *wireframe* di web draw.io yang kelak dapat menjadi acuan dalam desain *UI* dan fungsi aplikasi. Hasil berupa rancangan kasar dan tampilan dari poin poin penting yang sekiranya akan digunakan dalam perancangan aplikasi. Berikut merupakan gambaran dari hasil rancangan *wireframe*.



Gambar 3. 6 Tampilan wireframe Homepage

Gambar 3. 6 merupakan tampilan dari *wireframe* di halaman *Homepage*. Dapat dilihat tampilan dari *homepage* yang tergambaran seperti halaman halaman *homepage* pada umumnya. Didalam *homepage* menampilkan satu tombol yang sangat menonjol yang merupakan fungsi utama dari aplikasi web yang akan dirancang yakni tombol memesan ruangan. Dalam halaman tersebut juga terdapat fitur seperti *navbar* yang berisi logo perusahaan, navigasi navigasi ke *page* lainnya, serta tombol untuk melakukan *login* kedalam aplikasi. Selain fitur dan fungsinya, halam ini juga ditambahkan beberapa tambahan dari segi tampilan seperti *carousel* (slide foto) di atas tombol pemesanan serta *footer* dari web dibagian bawah halaman.



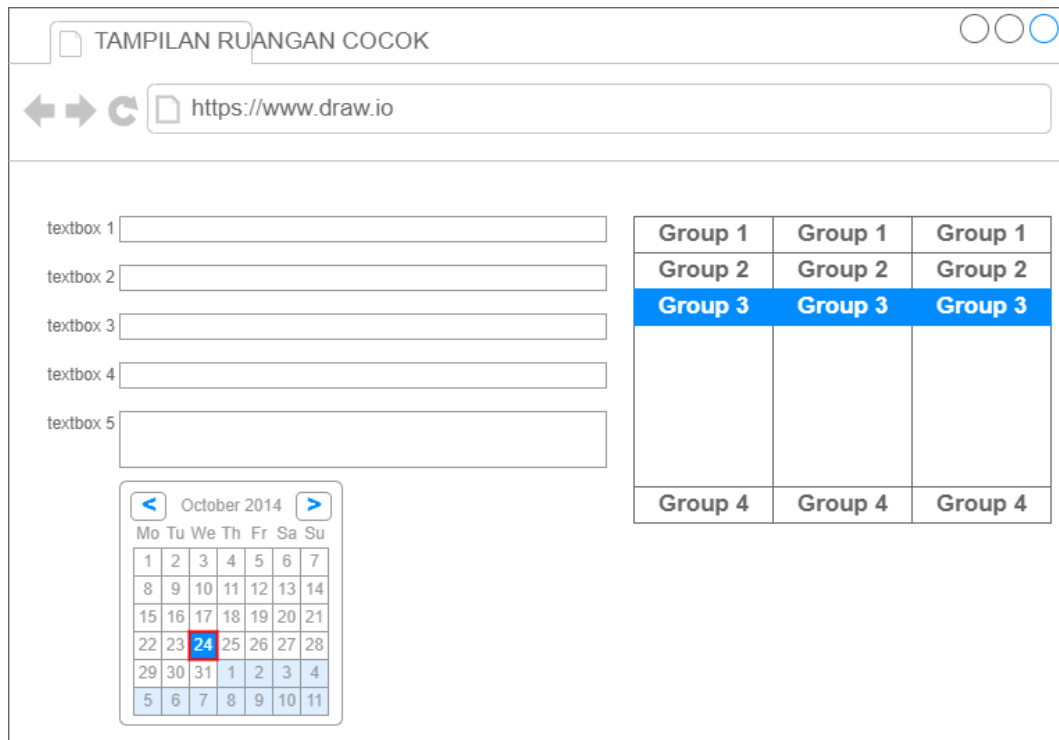
Gambar 3. 7 Wireframe Halaman Login

Selanjutnya dapat dilihat halaman *login* (Gambar 3. 7) yang akan disajikan sama seperti halaman *login* pada umumnya. Terdapat kolom untuk mengisi *id* dan *password* dari akun karyawan maupun admin. Dalam *page* ini, di implementasikan pula *logic* untuk memisahkan antara akun karyawan dan juga akun admin.

The screenshot shows a web browser window with the title "FORM PEMESANAN" and the URL "https://www.draw.io". The form contains five text input fields labeled "textbox 1" through "textbox 5". Below the textboxes is a calendar for October 2014. The calendar shows the days of the week (Mo, Tu, We, Th, Fr, Sa, Su) and the dates from 1 to 31. The date 24 is highlighted in red, indicating it is the selected date.

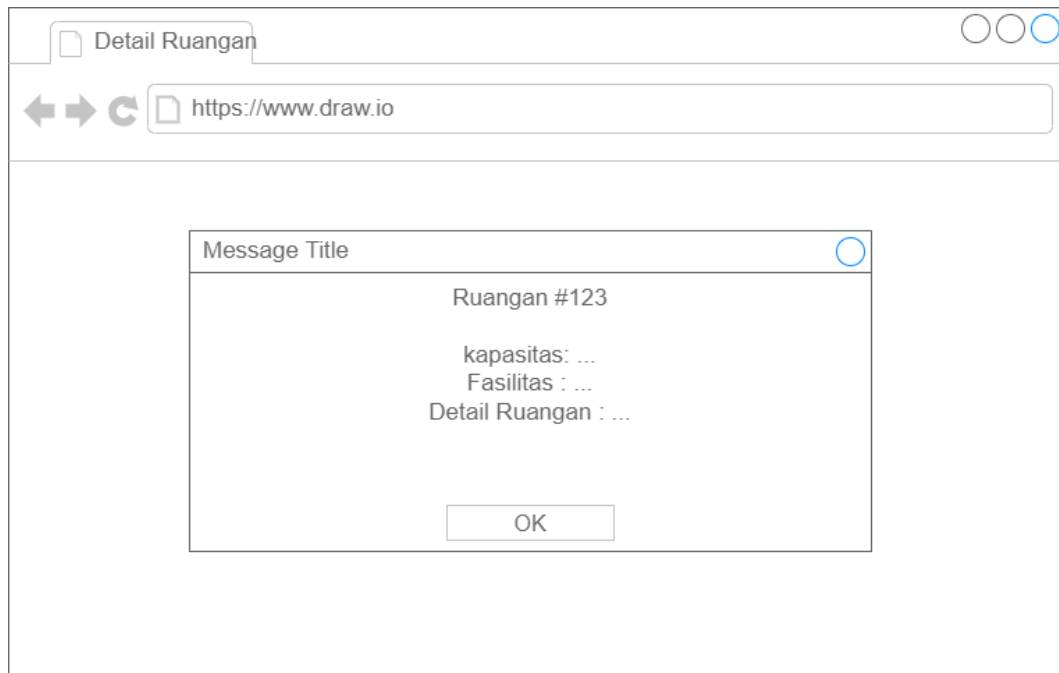
Gambar 3. 8 Form Pemesanan

Kemudian di page selanjutnya (gambar 3. 8) terdapat tampilan dari *form* pemesanan. Dalam halaman ini menampilkan formulir yang berisi tempat pengisian data atas ruangan yang akan dipesan. Data yang dibutuhkan dapat di isi dengan tanggal dan waktu pemesanan, nama pemesan, keperluan pemesanan, pemilihan nama ruangan dan fasilitas tambahan yang diperlukan. Gambaran *textbox* masih belum terlalu tergambar karena keperluan data dan juga isi database masih belum direncanakan.



Gambar 3. 9 Tampilan Ruangan Cocok

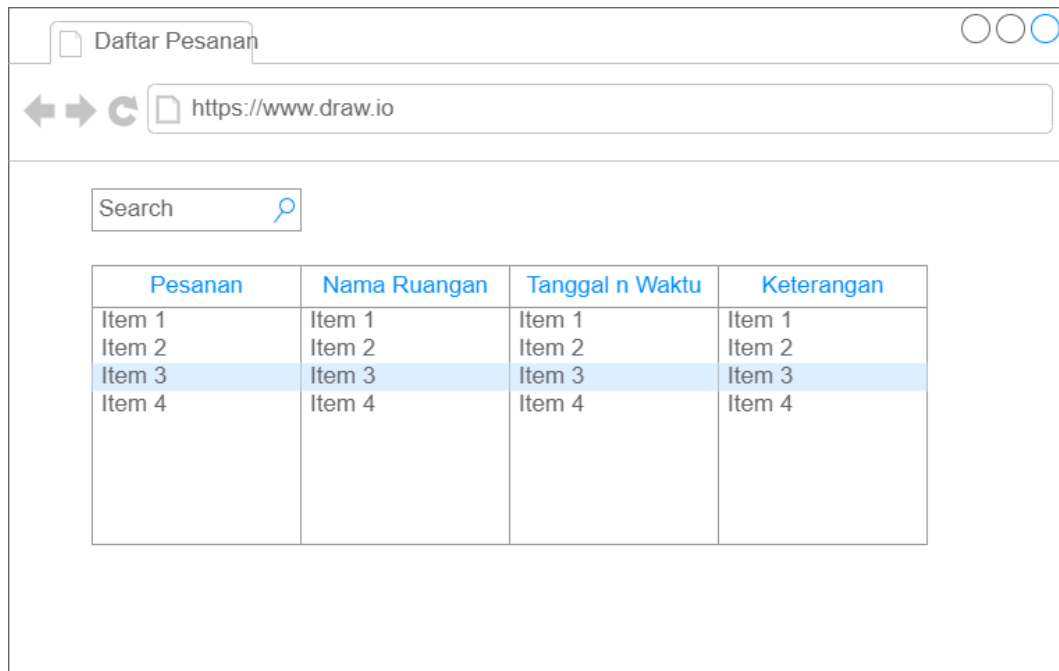
Setelah mengisi kelengkapan dan data yang diperlukan, maka daftar (list) ruangan yang cocok akan muncul akan ditampilkan di sebelah *form* yang disediakan. Daftar ruangan yang muncul ditampilkan seperti pada gambar 3. 9. Tampilan ruangan berisi rincian ruangan, lokasi ruangan, fasilitas, kapasitas ruangan dan juga tombol untuk memilih ruangan dari daftarnya. Penempatan daftar ruangan ditempatkan sebelah *form* pengisian agar jika karyawan ingin mengubah data yang ada di *form* dapat langsung dilakukan tanpa berpindah ke halaman lainnya.



Gambar 3. 10 Detail Ruangan

Dalam daftar ruangan yang muncul di halaman sebelumnya, terdapat tombol untuk menampilkan informasi lebih lanjut mengenai ruangan yang disediakan. Seperti pada gambar 3. 10, setelah menekan tombol tersebut, akan muncul berupa *pop-up* atas informasi ruangan yang tidak muat ditampilkan dalam daftar ruangan sebagai contoh gambar atau foto atas ruangan yang dipilih.

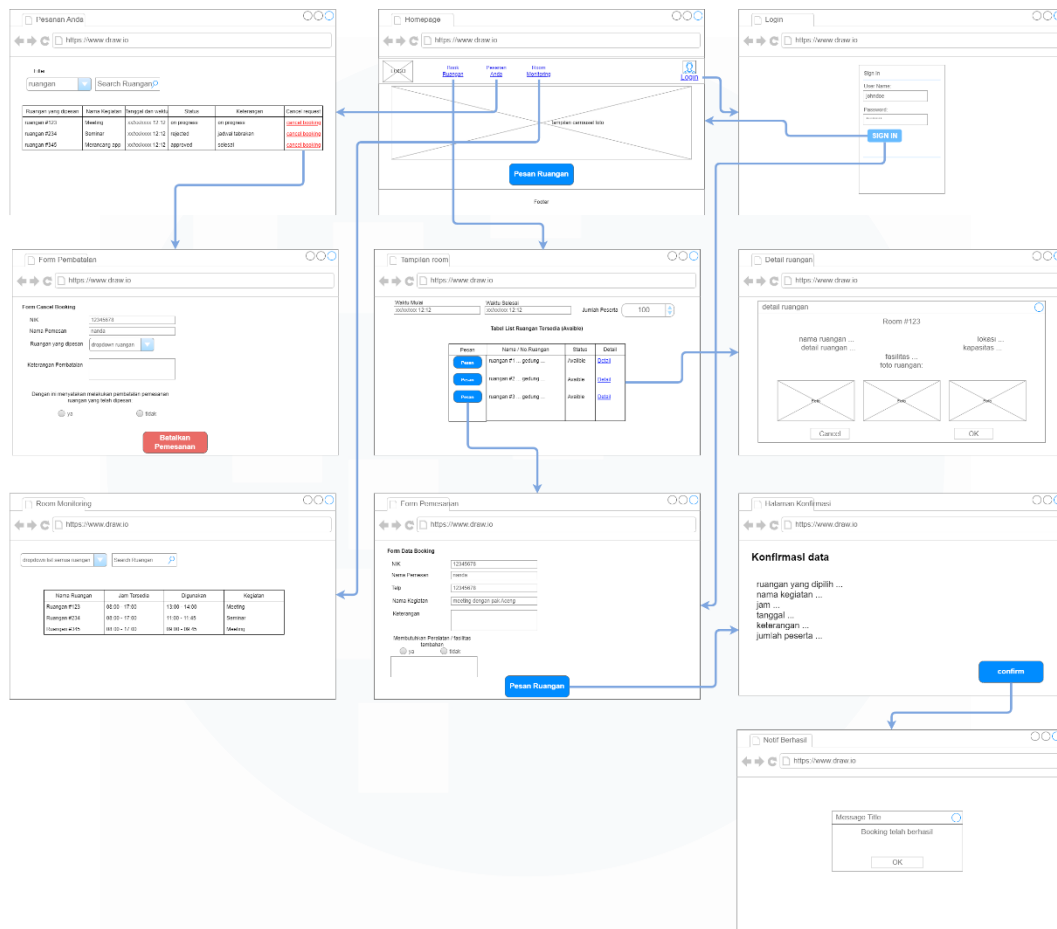




Gambar 3. 11 Halaman Daftar Pemesanan

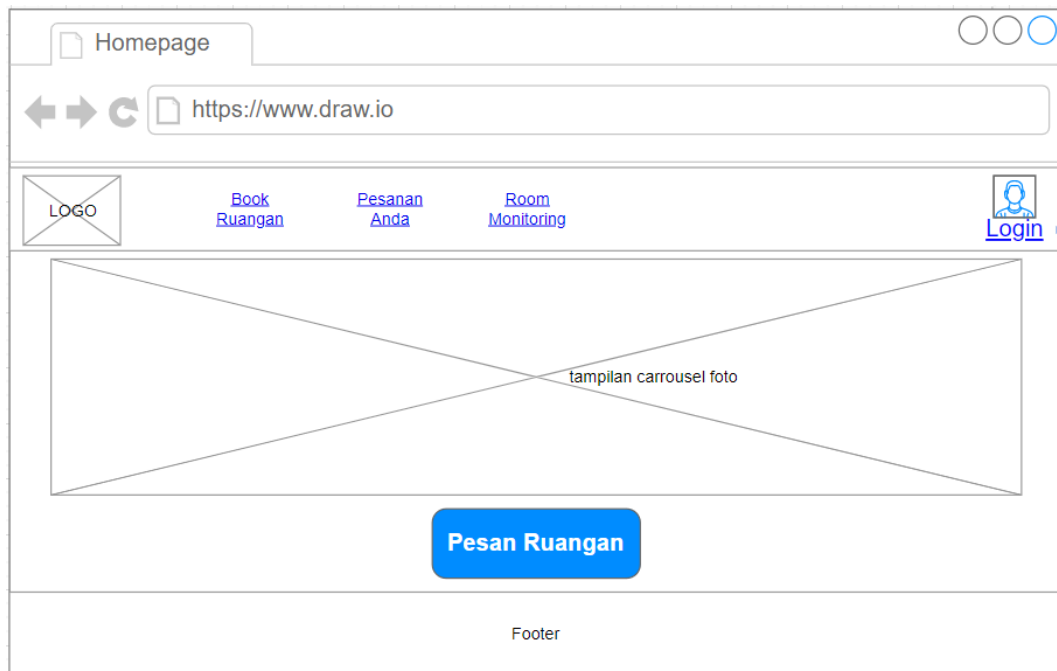
Kemudian untuk halaman terakhir tampilan gambar 3. 11, hasil rancangan *wireframe* adalah halaman daftar pemesanan. Disini karyawan disajikan tabel atas pemesanan ruangan yang telah dibuat. Daftar ini menampilkan pesanan atau dapat juga dipakai sebagai kontrol riwayat pemesanan yang sudah dilakukan. Karyawan juga akan menggunakan halaman ini sebagai pengecekan atas pesanannya apakah sudah di *approve* oleh admin atau belum.

Setelah pengerjaan *wireframe* terselesaikan, terdapat pembuatan *wireflow* aplikasi untuk memenuhi *requirement* dari tahap pembuatan aplikasi Booking App. *Wireflow* yang dimaksud ini memiliki tampilan yang mirip dengan *wireframe*, namun dengan atribut yang lebih detail dan tampilan tampilan web sudah terhubung satu sama lain.



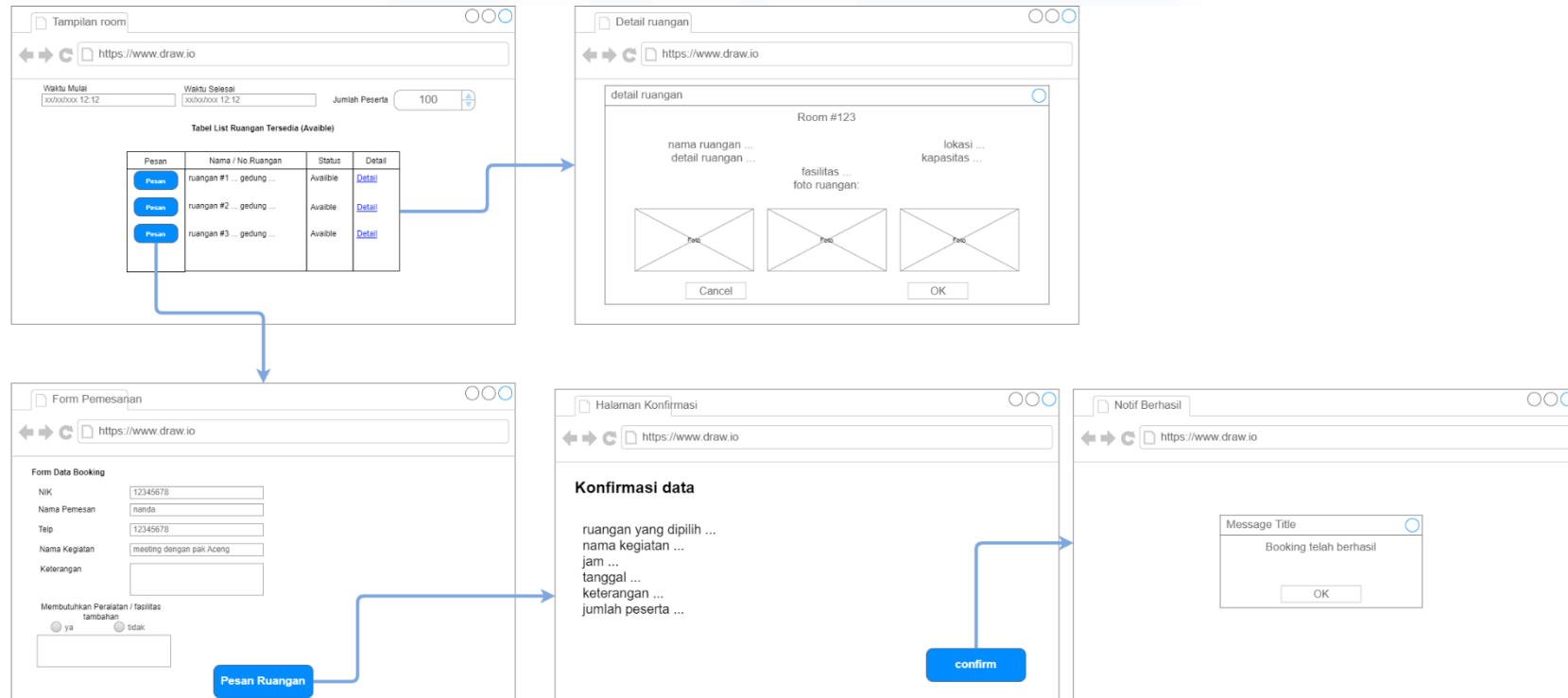
Gambar 3. 12 Tampilan Wireflow Aplikasi Bagian User

Dapat dilihat dari gambar 3. 12, merupakan tampilan keseluruhan dari wireflow bagian *user*. Di tampilan tersebut menunjukkan adanya koneksi antar halaman yang dihubungkan melalui tombol maupun fungsi yang ada di dalam *page*. Dari rancangan yang telah dibuat dan disetujui pendamping praktik kerja magang, halaman user memiliki 10 tampilan yang dapat digunakan. Berikut merupakan penjelasan antar tampilan tampilan tersebut.



Gambar 3. 13 Homepage

Dari gambar 3. 13, merupakan tampilan landing page saat membuka aplikasi Booking App. Di halaman ini disajikan menu pilihan untuk memesan ruangan, menampilkan daftar pemesanan yang dilakukan karyawan, *room monitoring* serta tombol *login* untuk mendapat hak akses sebagai karyawan ataupun admin dari aplikasi. Dari gambaran wireflow ini terdapat tampilan logo perusahaan dan juga tampilan *carousel* foto (slide foto) untuk menambahkan daya tarik visual di halaman ini. Seperti pada tampilan Gambar 3. 12, terdapat koneksi atau alur antar *page*. Untuk memperjelas alurnya akan berikut penjelasan lebih lanjutnya.



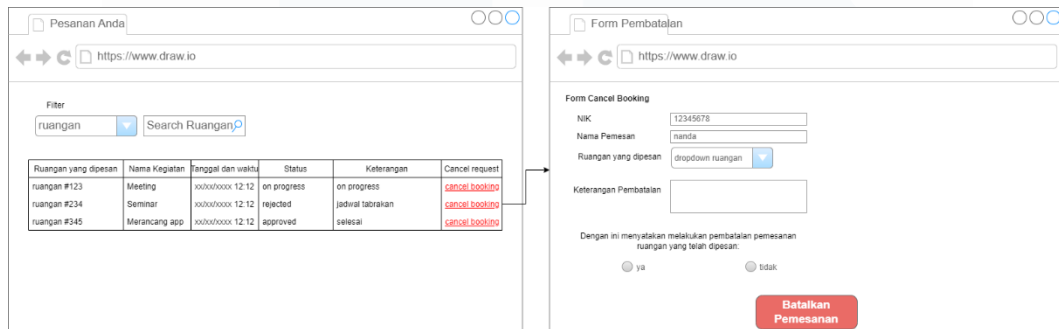
Gambar 3. 14 Wireflow Bagian Booking Ruangan

Dari gambar 3. 14 menunjukkan tampilan saat karyawan menekan bagian pesan ruangan (book ruangan). Tahapan awal yang disajikan adalah tabel tabel ruangan yang tersedia saat ini dan juga *textbox* yang meminta *input* tanggal dan waktu selesai serta jumlah peserta yang akan menggunakan ruangan. *Input* yang diperlukan tersebut berfungsi sebagai filtering ulang atas ruangan yang cocok atas kebutuhan karyawan, dan akan ditampilkan ulang di bagian tabel *list* ruangan. Selain tampilan dari ruangan, terdapat juga tombol “Detail” dalam tabel ruangan, yang berfungsi sebagai *Trigger* (tombol untuk memunculkan) *pop-up* informasi lebih lanjut ruangan yang dipilih. Detail tersebut berisi seperti nama ruangan, lokasi ruangan, kapasitas, fasilitas hingga foto dari ruangan tersebut.

Kemudian jika karyawan menekan tombol pesan yang berada di dalam tabel ruangan, maka karyawan akan langsung diarahkan untuk mengisi data lebih lanjut atas pemesanan yang dilakukan. Karyawan yang telah menekan tombol pesan, berarti sudah mengetahui informasi dan kecocokan atas ruangan yang dipilih. Tampilan form data lebih lanjut tersebut memiliki beberapa input *textbox* untuk mendapatkan informasi mengenai nama pemesan, Nomor Induk Karyawan, nama kegiatan, nomor telepon, keterangan (*additional*), serta permintaan fasilitas tambahan yang mungkin diperlukan.

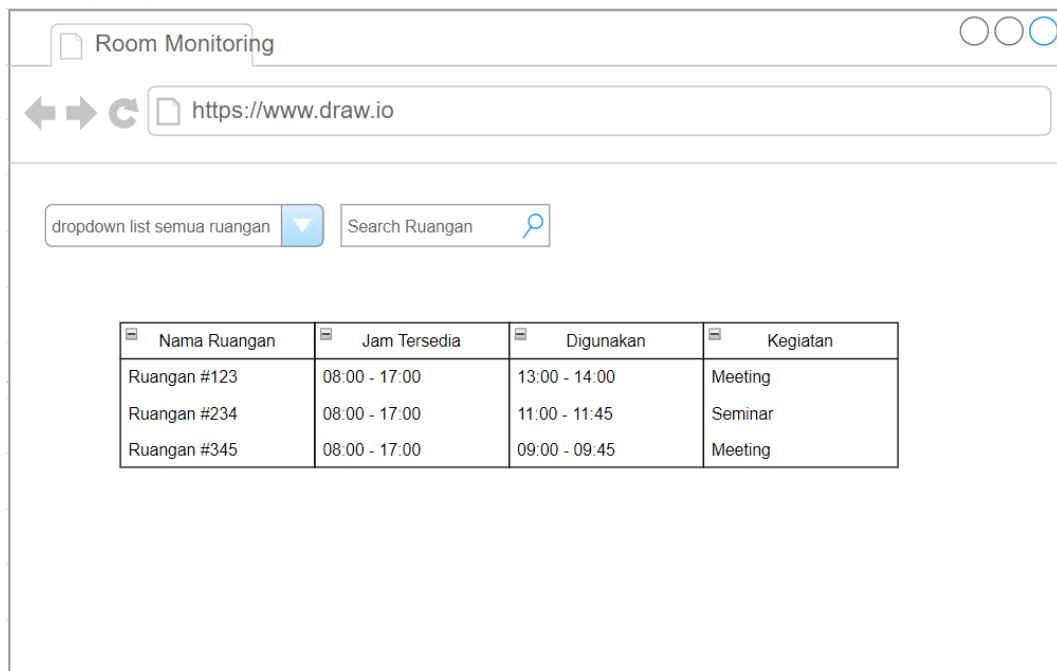
Setelah data dimasukan, karyawan akan diarahkan ke halaman konfirmasi. Halaman ini menampilkan data yang sudah dimasukan oleh karyawan agar karyawan dapat membaca dan mengkonfirmasi ulang data yang telah dimasukan. Jika karyawan belum memasukan data dengan benar atau terdapat salah pengetikan, maka karyawan akan diarahkan kembali ke bagian *form input* data untuk membenahi data yang dimasukan. Sebaliknya, jika karyawan merasa data yang dimasukan sudah benar maka karyawan diminta untuk menekan tombol “confirm”, lalu akan muncul *pop-up* yang berisi pemberitahuan jika pemesanan atau *Booking* telah berhasil. Setelah tampilan dan alur dari proses pemesanan selesai maka tampilan yang dilihat oleh karyawan akan kembali ke menu awal (*homepage*) dari aplikasi Booking App. Setelah tampilan kembali ke *homepage*, karyawan memiliki

hak untuk memilih beberapa aktivitas lain yang dapat dipilih, seperti melihat pesanan dan *room monitoring*. Untuk lebih lengkapnya, berikut tampilan dan alur dari fitur daftar pesanan.



Gambar 3. 15 Wireflow Halaman Daftar Pesanan

Dapat dilihat di gambar 3. 15, disebelah kiri terdapat daftar atau tabel yang menampilkan data pemesanan yang pernah dilakukan oleh karyawan. Daftar tersebut berisi informasi yang telah di *input* oleh karyawan di tahap sebelumnya serta status penerimaan atau *approval* dari admin aplikasi. Daftar pemesanan yang ada dapat di filter melalui *dropdown* ruangan yang tersedia maupun fitur *search* ruangan yang tersedia di halaman tersebut. Untuk pesanan yang belum disetujui atau di *approve* oleh admin, pesanan dapat dibatalkan oleh karyawan yang mengajukan pemesanan. Pengajuan pemesanan dapat dilakukan di halaman daftar pesanan dengan menekan tombol “cancel booking” yang tersedia didalam tabel daftar pesanan. Setelah menekan tombol tersebut, karyawan akan disajikan dengan *form* yang berisi *textbox* yang berfungsi sebagai *input* alasan pembatalan pemesanan. Proses pembatalan pemesanan nantinya akan tetap muncul di daftar pesanan sebagai catatan riwayat (*history*) dari pemesanan yang telah dilakukan.



Gambar 3. 16 Wireflow Bagian Room Monitoring

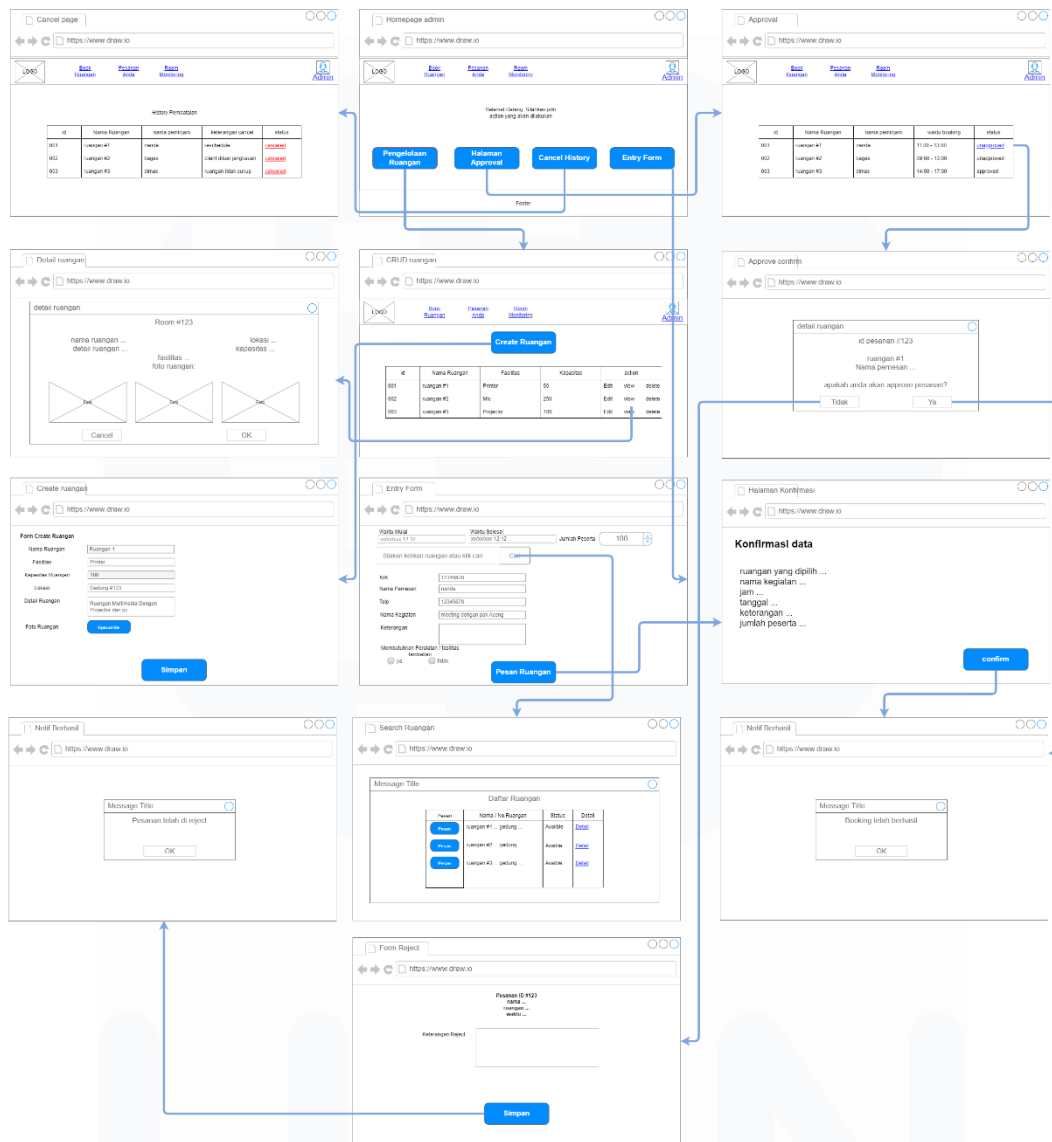
Lalu fitur terakhir yang dapat diakses oleh karyawan yakni “Room Monitoring.” Dalam tampilan di Gambar 3. 16 dapat dilihat halaman ini berisi tabel atau daftar dimana ruangan yang tertera sedang dipesan atau digunakan. Daftar tersebut menampilkan ruangan apa yang sedang digunakan beserta keterangan tanggal dan waktu mulai hingga selesai digunakan. Selain tanggal dan waktu daftar ini menyajikan keterangan kegiatan yang dilakukan didalam ruangan tersebut. Karyawan dapat sortir *list* dari daftar ruangan yang disajikan menggunakan *search bar* maupun *dropdown* dari ruangan.

Sekian merupakan penjelasan dari wireflow dan juga alur proses yang terjadi di tampilan karyawan saat mengakses program Booking App. Dapat dilihat dari wireflow yang disajikan, alur yang dilakukan tidaklah sulit dan cukup mudah dimengerti. Dari penggunaan fitur dan prosedur pemesanan hingga pembatalan dirangkum dalam alur yang sederhana untuk tidak menyulitkan karyawan saat menggunakan aplikasinya. Desain *User Interface* yang dirancang juga mempertimbangkan beberapa pengamatan dari program program aplikasi web yang

cukup terkenal, hingga mendapatkan kenyamanan dari penempatan fungsi aplikasi, namun disesuaikan dengan kebutuhan perusahaan pula.

Setelah penjelasan mengenai halaman yang digunakan karyawan, terdapat pula rancangan wireflow yang ditampilkan dan dapat diakses oleh admin. Kedua *role* ini memiliki tampilan yang berbeda dikarenakan admin memiliki beberapa hak khusus dalam mengakses aplikasi ini seperti mengubah data ruangan, menerima pemesanan ruangan, hingga memesan ruangan tanpa memerlukan persetujuan (approval). Berikut merupakan tampilan dari Wireflow di bagian admin.

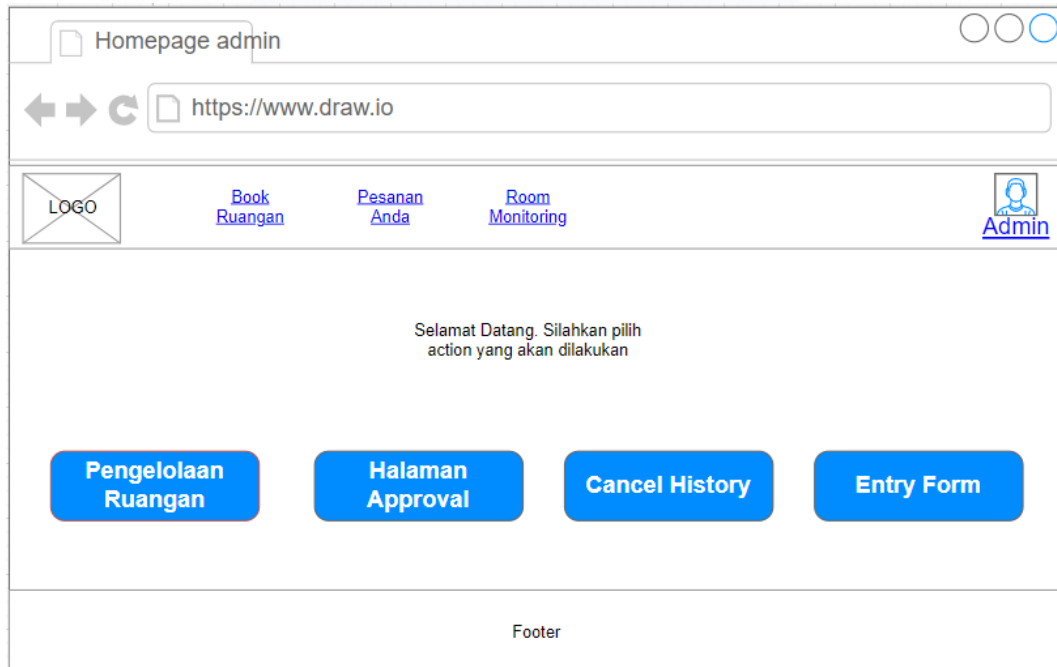




Gambar 3. 17 Wireflow Bagian Admin

Sekilas jika memperhatikan gambar 3. 17, Wireflow dari admin memiliki halaman yang lebih banyak serta alur yang sedikit lebih kompleks dari tampilan Wireflow karyawan. Dilihat dari *homepage* saja, terdapat pilihan menu yang lebih banyak dari *homepage* yang dimiliki oleh karyawan. Hal tersebut dibuat karena admin memiliki peranan yang lebih banyak dalam proses jalannya aplikasi. Walau terlihat lebih banyak dan kompleks, terdapat fitur yang dirasa sama dan mirip dalam

proses yang disajikan dalam wireflow karyawan. Untuk informasi lebih lanjut berikut penjelasan detail mengenai wireflow bagian admin.



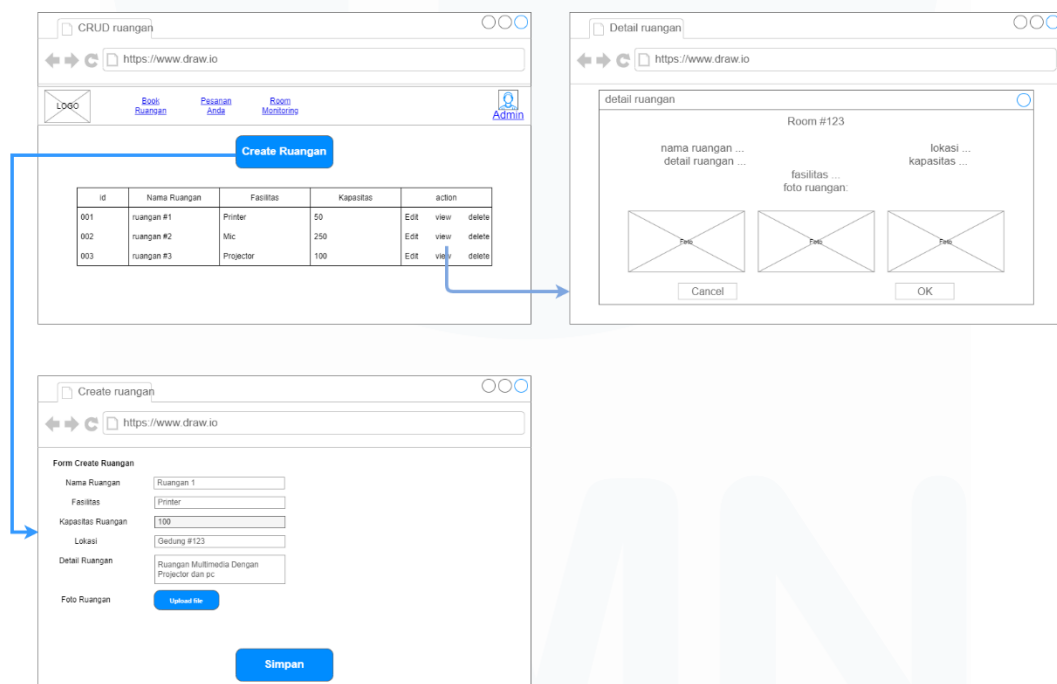
Gambar 3. 18 Tampilan HomePage Admin

Dari tampilan gambar 3. 18 terlihat gambar *homepage* yang mirip dengan halaman karyawan, mulai dari *navbar* hingga *footer* yang dimasukkan. Namun terdapat perbedaan yang sangat menonjol di bagian tombol fungsinya. Empat tombol tersebut mewakili beberapa hak istimewa yang dimiliki admin dalam menjalankan aplikasi Booking App. Tombol Pengelolaan Ruangan, merupakan tombol yang berfungsi untuk memulai aktivitas edit data ruangan seperti *create*, *update* dan *delete* data ruangan yang tersimpan di database.

Tombol Halaman Approval, merupakan tombol yang berfungsi untuk menempatkan admin dalam halaman penerimaan atau penolakan pemesanan ruangan. Dalam halaman ini, admin akan disajikan oleh beberapa pesanan yang dilakukan oleh karyawan. Kemudian admin memiliki pilihan untuk menerima maupun menolak pemesanan ruangan tersebut. Penolakan pesanan juga berdasarkan beberapa pertimbangan seperti, jadwal yang bertabrakan (tidak sesuai),

fasilitas yang tidak siap, ketidak muatan jumlah peserta, hingga data yang dimasukan tidak valid.

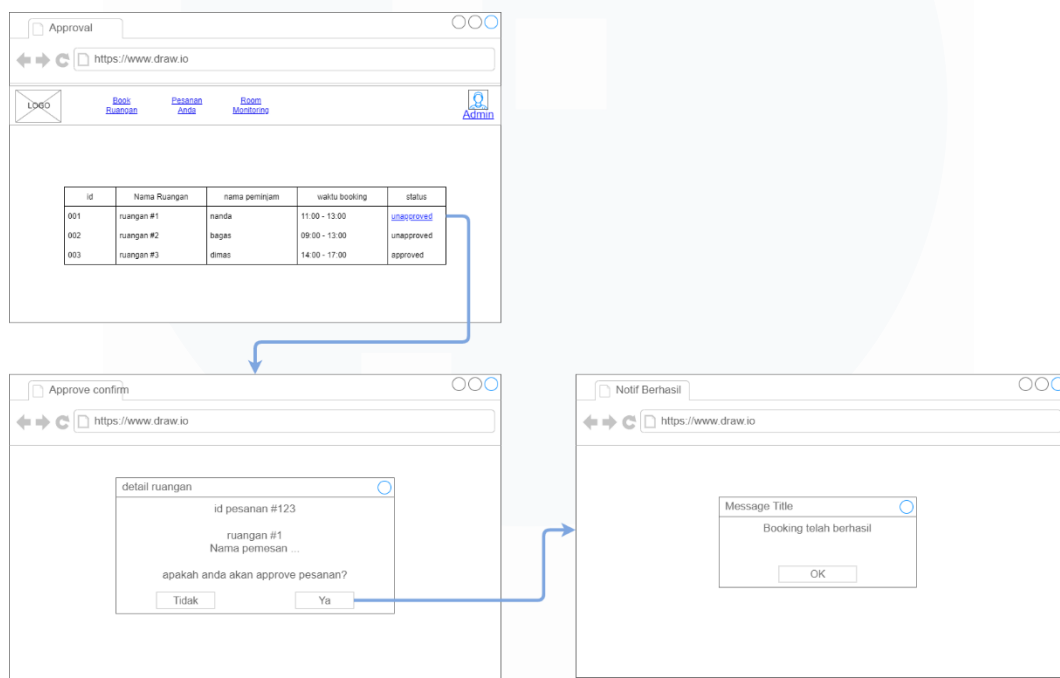
Tombol *cancel history* merupakan tombol yang mengarahkan admin menampilkan daftar dari pesanan yang dibatalkan. Dikarenakan proses pembatalan tidak memerlukan persetujuan admin, maka daftar ini dibuat agar admin dapat melihat dan mengontrol catatan pembatalan yang pernah dilakukan. Lalu terdapat tombol *entry form* merupakan tombol yang mengarahkan admin melakukan pemesanan tanpa melalui langkah persetujuan (*approval*). Pemesanan ini dapat dilakukan untuk diri sendiri maupun untuk karyawan lain.



Gambar 3. 19 Wireflow Pengelolaan Ruangan

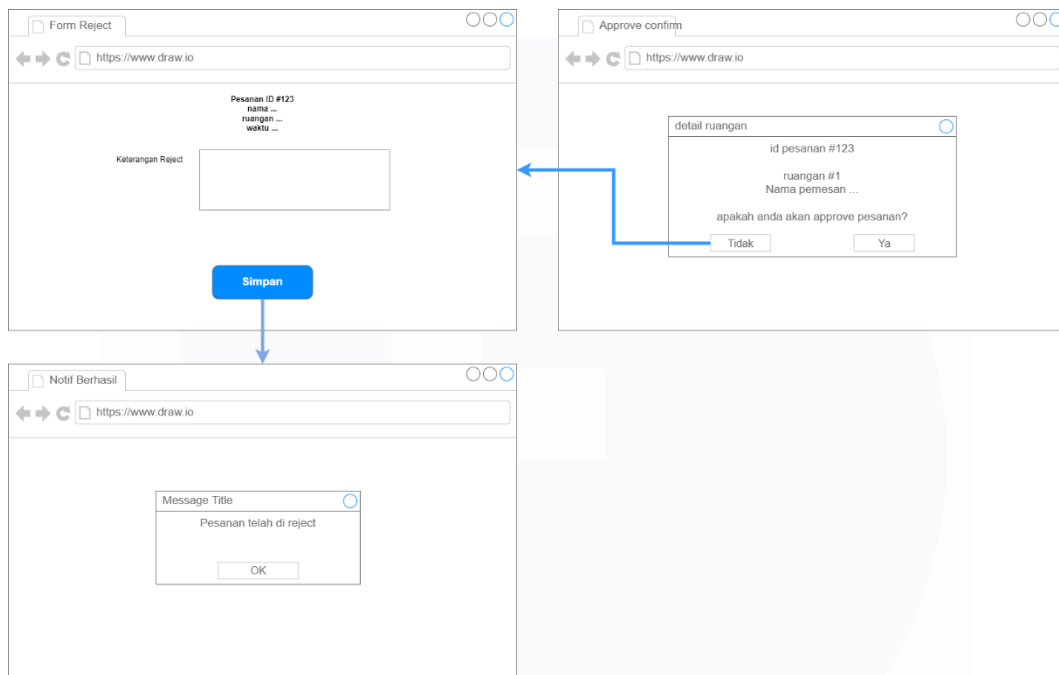
Dari gambar 3.18, jika admin memilih tombol pengelolaan ruangan, maka akan diarahkan ke halaman yang ditampilkan pada gambar 3.19. Didalam tampilan tersebut admin disajikan dengan *list* ruangan yang telah tersedia di dalam aplikasi Booking App. Tampilan berbentuk *list* atau tabel yang berisi informasi mengenai ruangan tersebut seperti nama ruangan, fasilitas, kapasitas dan tombol *action* untuk mengubah, *view* atau hapus ruangan. Jika admin memilih tombol *view* ruangan pada

daftar yang tersedia, maka akan muncul rincian dan semua informasi yang dimiliki dari ruangan tersebut hingga menampilkan foto ruangan yang tersimpan di database. Jika admin memilih tombol “Create Ruangan” maka admin akan diberikan tampilan form yang membutuhkan *input* data untuk generate ruangan baru. Data yang dibutuhkan seperti nama ruangan, fasilitas, kapasitas hingga foto untuk menjadi daftar ruangan yang siap digunakan.



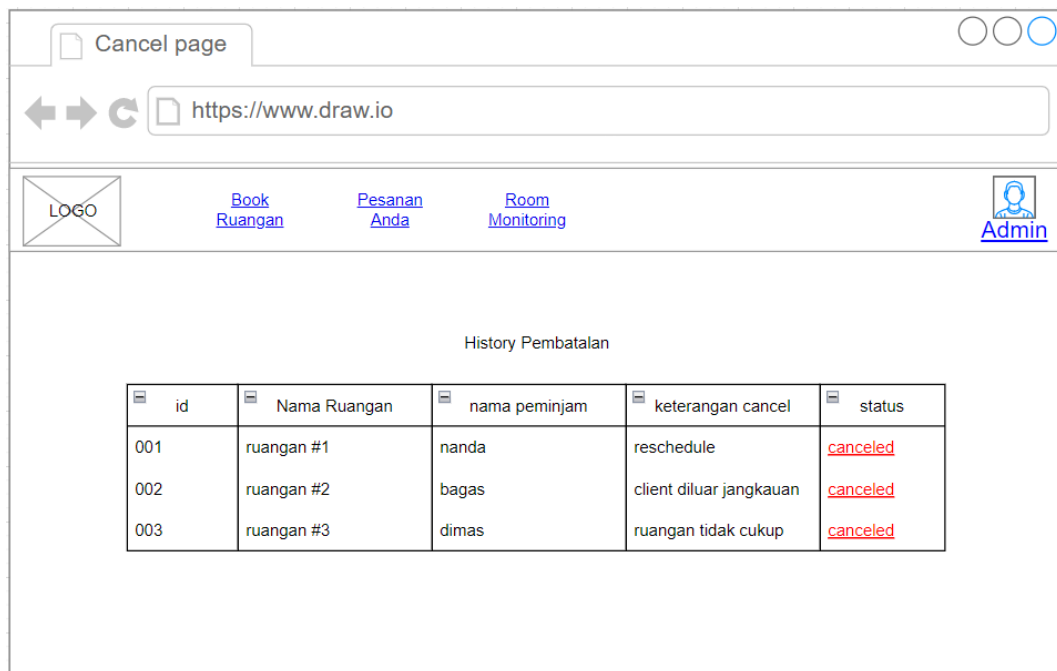
Gambar 3. 20 Wireflow Halaman Approval

Kemudian pada tombol Halaman Approval, admin akan diarahkan pada *page* yang berisi list pesanan karyawan, yang membutuhkan tahap persetujuan. Seperti pada gambar 3. 20, terdapat kolom status seperti “Approved”, “Not Approve” dan “Unapproved.” Jika pesanan berstatus “Unapproved” di pilih, maka admin akan diberi *pop-up* untuk approve atau batalkan pesanan. Saat admin memilih *action* “Ya” maka akan muncul *pop-up* lagi dimana berisi pemberitahuan mengenai pesanan diterima (di *Approve*).



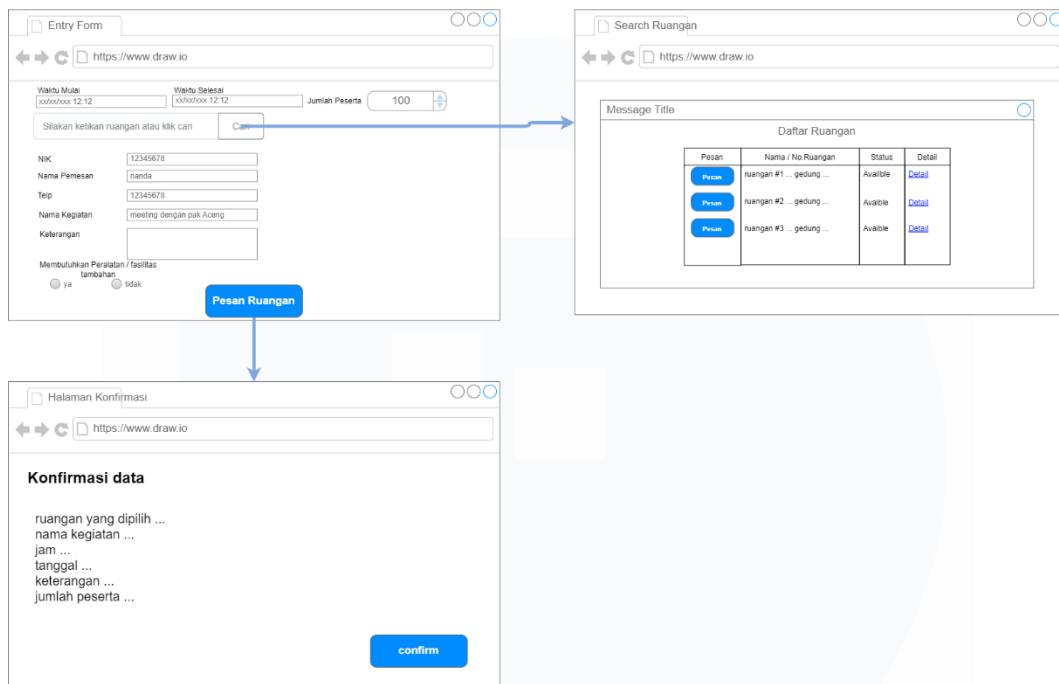
Gambar 3. 21 Wireflow Halaman Approval Bagian Penolakan

Seperti pada gambar 3. 21 jika admin memilih “Tidak” pada pilihan yang muncul, maka akan muncul sebuah form yang berisi *textbox* yang bertujuan untuk meminta data kenapa pesanan ditolak. Keterangan tersebut akan muncul pada halaman pesanan karyawan sehingga karyawan juga mengetahui kenapa pesannya harus dibatalkan.



Gambar 3. 22 Wireflow Cancel History

Di tampilan gambar 3. 22 dalam halaman “Cancel History”, tampilannya lebih sederhana, hanya menampilkan *list* atau daftar yang berisi daftar pesanan dibatalkan baik oleh karyawan maupun oleh admin, beserta keterangan pembatalannya. Di halaman ini menampilkan seluruh data pembatalan agar dapat menjadi catatan dari pembatalan yang telah dilakukan. Dari daftar ini juga diharapkan dapat mempermudah data riwayat dan pencatatan.



Gambar 3. 23 Wireflow Entry Form

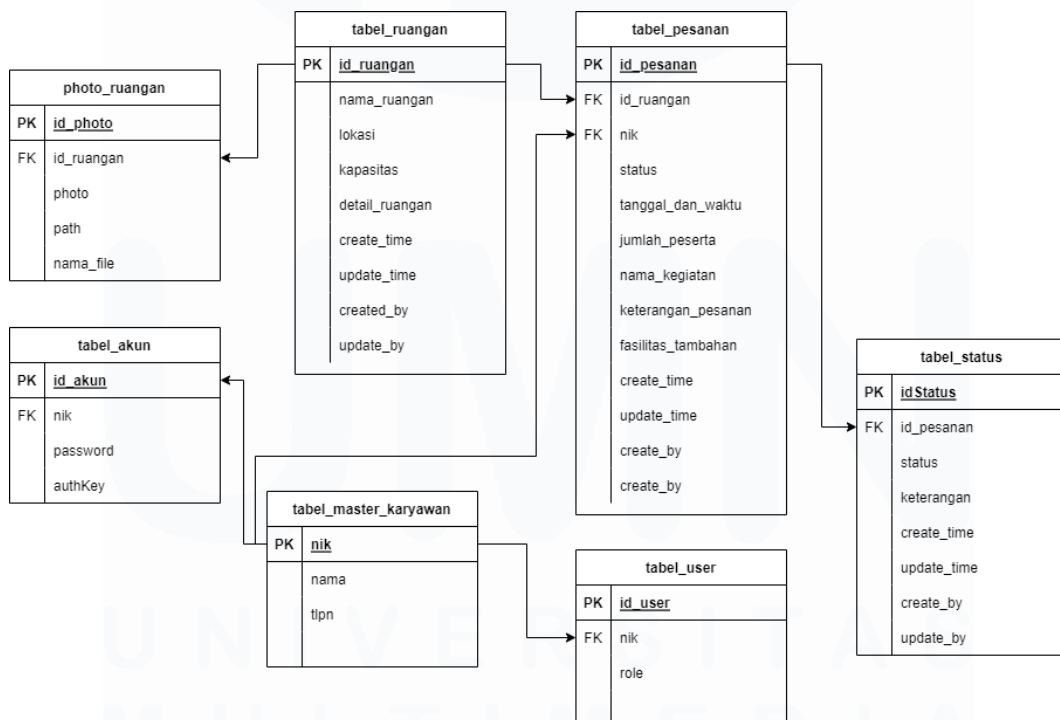
Pada gambar 3. 23 terdapat fitur spesial di akun admin adalah “Entry Form”. Di dalam alur proses *entry form*, admin akan disuguhkan formulir pemesanan, yang kurang lebih sama dengan *form* pemesanan yang dimiliki oleh karyawan. Perbedaannya terletak pada bagian *search* ruangan, dimana saat *search* akan muncul *pop-up* yang berisikan *list* ruangan yang lengkap dan sesuai dengan hasil *filtering* yang dibutuhkan. Setelah pengisian data pemesanan selesai maka akan muncul halaman konfirmasi data untuk mengecek kembali data yang sudah dimasukan (di *Input*). Jika sudah, maka akan muncul *pop-up*, yang berisi pemberitahuan jika pemesanan telah berhasil. Admin juga dapat mengontrol pemesanan melalui halaman pemesanan yang tersedia, dan memiliki tampilan yang sama dengan halaman pemesanan yang ada di bagian karyawan.

Untuk tahap pembuatan wireflow sendiri didapati beberapa revisi dalam pengerjaannya, melihat dari pengembangan pengembangan rancangan *user interface* hingga penempatan penempatan fungsi agar lebih efektif dan terpakai. Proses pembuatan juga harus disertai persetujuan oleh *supervisor* magang, sehingga

tepat sasaran dengan kebutuhan perusahaan. Untuk tahapan *Requirement* dari aplikasi Booking App, ini sudah berakhir hingga rancangan *wireflow*. Selanjutnya tugas atau agenda yang harus dilaksanakan sudah masuk ke tahapan pembuatan database dari aplikasi.

### 3.2.2.4 Desain Database

Untuk tahapan selanjutnya peserta magang diminta membuat rancangan dari database aplikasi Booking App. Perancangan dilakukan agar saat pembuatan database tidak terjadi banyak pembenaran dan implementasinya ke aplikasi dapat tepat digunakan. Dalam desain pembuatannya dapat dimulai dari pembuatan struktur database seperti tabel tabel beserta isinya. Setelah menentukan tabel dan isinya, struktur dapat dikoneksikan atau dihubungkan satu sama lain berdasarkan relasi khusus di data tabel. Berikut merupakan rancangan struktur database yang telah dibuat oleh peserta magang beserta penjelasannya.



Gambar 3. 24 Struktur Database



Dapat dilihat dalam gambar 3. 24 merupakan wujud dan tampilan struktur database yang telah dibuat dan disetujui oleh *supervisor* kerja magang. Dalam struktur tersebut terdapat tujuh tabel yang menampung isi data berdasarkan penggolongan masing masing. Tabel tabel tersebut juga telah terkoneksi agar dapat mempermudah pemanggilan data antar tabel.

Tabel ruangan (tabel\_ruangan) merupakan tabel yang menyimpan informasi mengenai ruangan yang ada didalam aplikasi Booking App. Dalam tabel ini menyimpan kolom kolom seperti id\_ruangan, yang berisi Id dari setiap ruangan. Id ruangan ini menggunakan angka dan sifatnya *auto increment*, yakni angka yang dibuat secara otomatis, unik (tidak sama) serta berurutan. Kemudian terdapat kolom nama\_ruangan. Kolom ini menyimpan data *varchar* sebagai tempat penyimpanan nama ruangnya, sebagai contoh, “Ruangan Multimedia”, “Ruangan Rapat.” Lalu ada kolom lokasi. Lokasi yang dimaksud disini ialah lokasi letak ruangan tersebut. Selanjutnya ada kolom detail\_ruangan. Kolom ini mampu menyimpan *varchar* dengan kapasitas yang sedikit lebih banyak dari kolom kolom sebelumnya. Kolom ini menyimpan keterangan atau penjelasan dari ruangan yang tersedia. Keterangan tersebut berupa deskripsi (gambaran) singkat dari ruangan. Dan dapat dilihat dari gambar 3. 24, terdapat empat kolom terakhir seperti create\_time, update\_time, create\_by, update\_by yang merupakan kolom *log* yang berfungsi sebagai catatan sistem disaat terjadi *create* (pembuatan) data, atau *update* (mengubah) data. Create\_time dan update\_time akan mencatat tanggal dan waktu Ketika terjadi penambahan atau edit data ruangan. Pencatatan data tersebut diatur secara otomatis berdasarkan tanggal dan waktu pembuatan atau edit data. Hal yang sama juga terjadi di kolom create\_by dan update\_by. Perbedaannya kolom ini akan mengisi data otomatis berdasarkan *username* akun yang sudah masuk kedalam aplikasi. Sebagai contoh, pembuatan data “Ruangan A” menggunakan akun admin dengan *username* “Andi”, maka nama “Andi” akan di ambil oleh sistem dan dicatat dalam kolom create\_by di data “Ruangan A” secara otomatis.

Tabel selanjutnya adalah tabel\_master\_karyawan. Tabel ini berisikan data utama karyawan yang ada di PT Telkom Indonesia. Dikarenakan project ini tidak menggunakan data karyawan asli dari perusahaan, maka kolom dari tabel ini digantikan dengan nik (nomor induk karyawan), nama dan juga nomor telepon. Isi dari tabel ini berupa data simulasi (data dummy) menimbang regulasi privasi perusahaan dan juga agar pengerjaan pembuatan aplikasi di program magang bisa lebih sederhana. Ada pula tabel\_akun. Berbeda dengan tabel\_master\_karyawan, tabel ini berisi tentang akun yang akan menjadi simulasi *id* dan *password* untuk mendapat hak akses dalam aplikasi. Dalam tabel ini berisi *id\_akun* yang merupakan *primary key*, *nik* sebagai *id login*, *password*, dan juga *authKey* yang merupakan komponen penting di Framework Yii 2. Kemudian terdapat tabel\_user. Tabel ini hanya memiliki tiga kolom yang salah satu kolomnya merupakan *foreign key* dari tabel\_master\_karyawan, yakni menggunakan kolom *nik*. Tabel ini memiliki fungsi utama untuk menyimpan role, yang membedakan antara akun admin dan akun karyawan.

Selanjutnya terdapat tabel photo\_ruangan. Tabel ini merupakan pecahan dari tabel\_ruangan dan masih terhubung langsung dengan tabel\_ruangan. Tabel ini berfungsi sebagai tempat penyimpanan foto (gambar) dari ruangan yang tersedia. Dalam tabel ini terdapat kolom kolom seperti, kolom *id\_photo* yang merupakan *primary key* dari tabel yang bersifat *auto increment*. Satu id dari kolom ini mewakili satu gambar yang tersimpan didalam tabel. Kemudian ada *id\_ruangan* yang merupakan *foreign key* dari tabel\_ruangan. Lalu kolom *photo* yang merupakan kolom penyimpanan berupa *blob*. Kemudian ada kolom *path* yang merupakan kolom yang menyimpan path gambar yang disimpan dalam database. Data yang dimasukan kedalam kolom ini akan diacak melalui *logic generate random string* sehingga tidak akan ada duplikasi dalam databasenya. Dan yang terakhir terdapat kolom *nama\_file*. Kolom ini menyimpan nama asli dari gambar yang disimpan di dalam database sehingga nama filenya masih dapat diselamatkan tanpa terkena imbas dari *generate random string* di kolom sebelumnya.

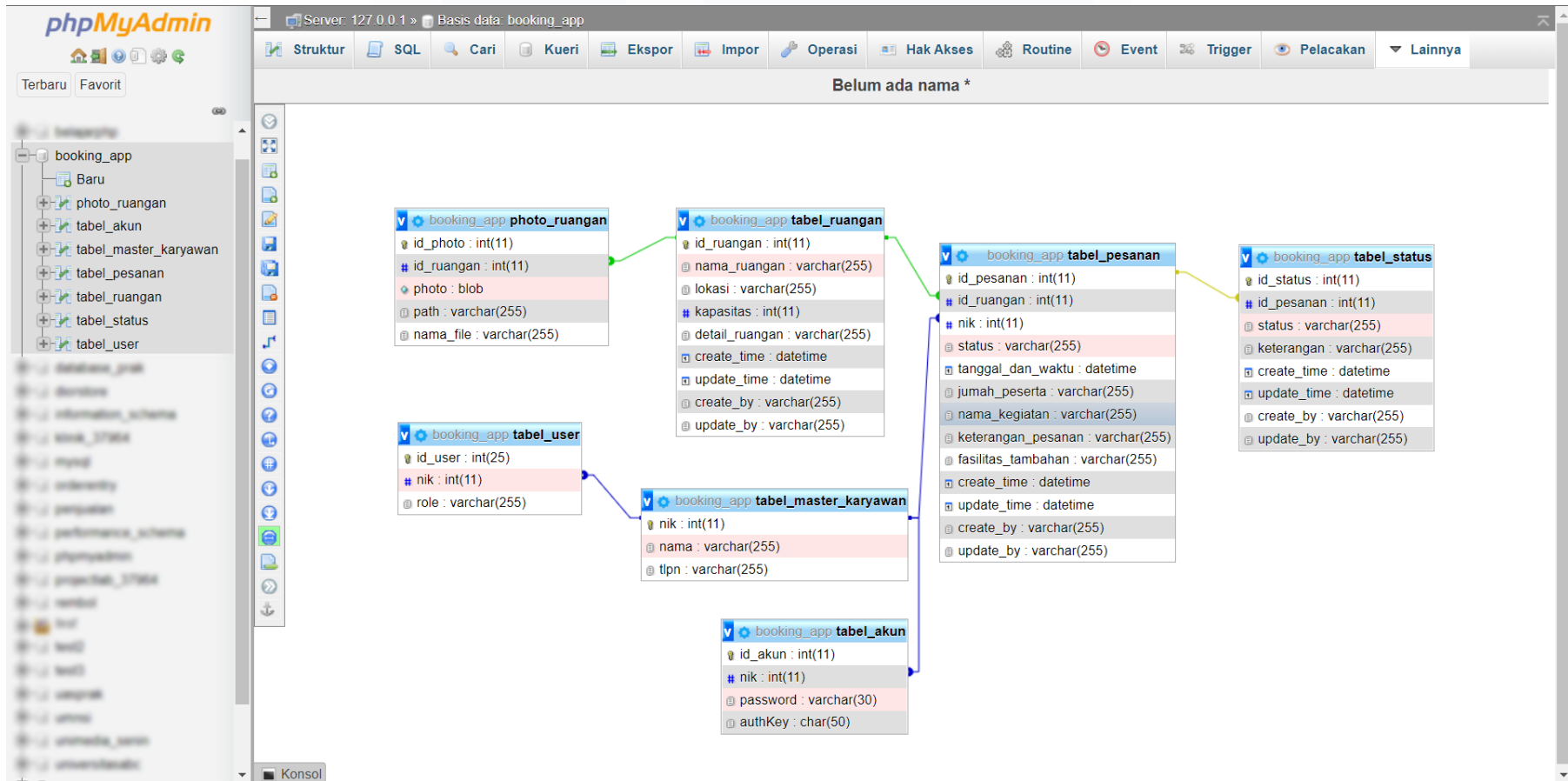
Tabel selanjutnya adalah tabel\_pesanan. Dalam tabel ini memiliki kolom yang lebih banyak dari tabel tabel sebelumnya dan memiliki koneksi paling banyak antar tabel yang ada. Tabel ini menyimpan data data atas pemesanan ruangan dan informasi informasi gabungan yang dapat ditampilkan dalam menu pesanan. Kolom kolom yang ada didalam tabel ini antara lain id\_pesanan, yang merupakan *primary key* dari tabel\_pesanan. Id\_ruangan, yang merupakan *foreign key* dari tabel\_ruangan. Nik, yang merupakan foreign key dari tabel\_master\_karyawan. Kolom status yang menyimpan status dari pemesanan. Kolom tanggal\_dan\_waktu, merupakan kolom yang mencatat dan menyimpan tanggal dan waktu pemesanan. Kemudian terdapat kolom jumlah peserta, yang mencatat banyaknya jumlah peserta yang di input pada saat pemesanan. Koolom nama\_kegiatan, yang menyimpan nama kegiatan yang akan dilakukan pada saat memesan ruangan. Kolom keterangan\_pesanan, kolom ini merupakan kolom tambahan, yang menyimpan keterangan tambahan. Kolom ini bersifat opsional atau boleh *null*. Kolom selanjutnya adalah fasilitas\_tambahan. Di kolom ini pemesan dapat menambahkan catatan jika membutuhkan pemesanan tambahan dalam pemesanan sebuah ruangan. Sama seperti kolom sebelumnya, kolom ini bersifat opsional. Kemudian ke empat kolom terakhir yaitu create\_time, update\_time, create\_by, update\_by sama seperti di tabel sebelumnya yakni kolom *log system* dan memiliki *logic* yang sama.

Dan yang terakhir terdapat tabel\_status. Tabel ini berfungsi sebagai tempat penyimpanan status pemesanan beserta keterangannya. Jika dipikirkan dan ditelaah, kolom status dalam aplikasi sebenarnya tidak memerlukan tabel untuk dideklarasikan. Namun, keterangan *cancel* dan pembatalan memerlukan tempat penyimpanan, sehingga perlu diadakan tabel\_status. Dalam tabel ini pun terdapat beberapa kolom seperti id\_status, id\_pesanan, status, keterangan dan ke empat kolom *log system*.

Setelah struktur database tersusun dan diterima oleh *supervisor* praktik kerja magang, langkah selanjutnya adalah *generate* menjadi database yang dapat digunakan. Proses pembuatan database menggunakan MySQL di program Xampp

didalam php MyAdmin. Semua proses pembuatan dilakukan didalam php MyAdmin mulai dari pembuatan database dan tabel tabelnya. Hal ini dilakukan karena proses didalam php MyAdmin lebih mudah diselesaikan.

Database aplikasi yang dibuat diberi nama “booking\_app” serta ketujuh tabel sesuai dengan struktur database. Tabel dan kolom yang dibuat sama seperti di penjelasan struktur tabel sebelumnya. Selain ketentuan, penghubung antar tabel juga disesuaikan agar data dapat berkorelasi di dalam database. Berikut tampilan database di dalam php MyAdmin.

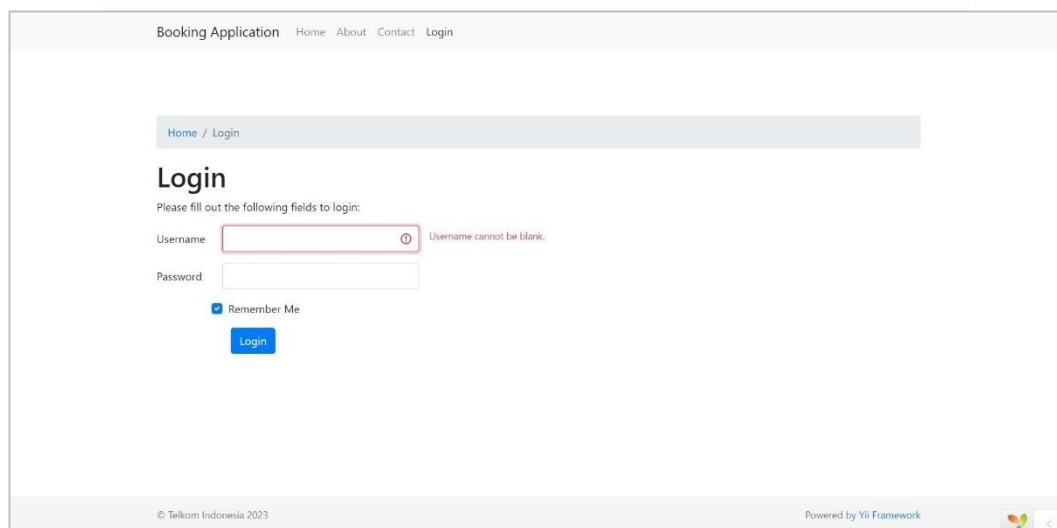


Gambar 3. 25 Tampilan Database di php MyAdmin

Dalam tahap ini, database sudah siap digunakan dan siap di hubungkan atau di migrasi ke aplikasi Booking App. Seperti pada gambar 3. 25 koneksi antar tabel juga sudah tidak ada masalah dan siap digunakan. Untuk tahap desain database telah selesai, peserta magang dapat melanjutkan ke proses selanjutnya yakni tahap *Development* Aplikasi.

### 3.2.2.5 Development Aplikasi

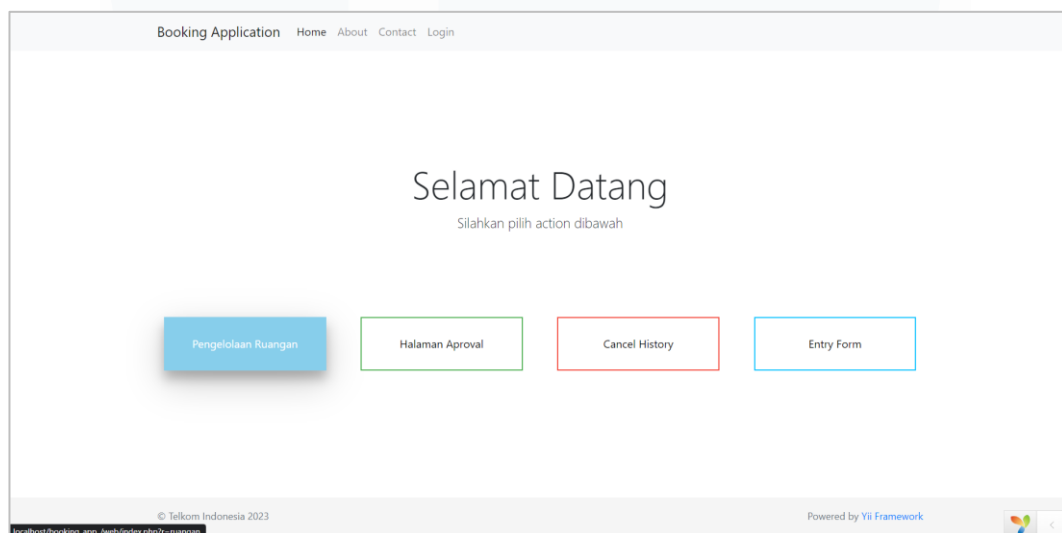
Aplikasi berbasis web yang dibuat dalam program praktik kerja magang ialah aplikasi bernama Booking App. Booking App merupakan aplikasi yang berfungsi untuk mengatur dan memanajemen segala tindak pemesanan ruangan yang dapat dipinjam oleh karyawan perusahaan secara online. Aplikasi Booking App memiliki dua *role*, yakni karyawan dan admin. Karyawan yang dapat mengakses aplikasi merupakan pekerja yang telah terdaftar dalam database perusahaan. Sedangkan *role* admin adalah karyawan yang terpilih sebagai admin untuk menjalankan hak istimewa dalam aplikasi. Berikut merupakan hasil pembuatan aplikasi Booking App pada praktik kerja magang di PT Telkom Indonesia.



Gambar 3. 26 Halaman Login

Gambar 3. 26 merupakan tampilan login yang ada di aplikasi Booking App. Di halaman ini, sistem dapat menentukan apakah akun yang dimasukan merupakan

akun karyawan atau akun admin. Akun yang telah diberi hak akses sebagai admin akan diarahkan ke tampilan home page admin yang sudah disediakan. *User* akan diminta NIK dan juga *password* yang telah terdaftar dalam database sebagai data *login* ke aplikasi. Dapat dilihat dari gambar 3. 26 jika salah satu *textbox* tidak diisi maka proses *login* tidak dapat dilanjutkan dan mengeluarkan pesan “...cannot be blank.” Lalu jika *user* salah memasukan NIK atau *password* kedalam *textbox*, maka proses *login* akan ditolak. Terdapat juga fitur “Remember Me” yang berfungsi untuk mengingat dan menyimpan akun *user* yang sudah masuk sehingga tidak memerlukan *login* ulang ke dalam aplikasi.



Gambar 3. 27 Halaman HomePage Aplikasi

Pada gambar 3. 27, merupakan halaman yang menampilkan fungsi dan navigasi utama pada halaman admin. Halaman ini akan menjadi tampilan pertama yang muncul setelah *login*. Sesuai dengan desain *wireflow* di bagian admin, halaman ini memiliki empat tombol utama yakni Pengelolaan Ruangan, Halaman Approval, Cancel History, dan juga Entry Form.

























Booking Application Home About Contact Login

Home / Pengelolaan Ruangan

## Pengelolaan Ruangan

Tambahkan Ruangan Baru

Showing 1-8 of 8 items.

Id Ruangan	Nama Ruangan	Lokasi	Kapasitas	Detail Ruangan	
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	
1	Ruangan 1	Gedung 123	100	Ruangan Meeting	  
2	Ruangan 2	Gedung 234	150	Ruangan Serbaguna	  
3	Ruangan 3	Gedung 73	200	Ruangan meeting untuk project	  
4	Ruangan 4	Gedung A	50	Ruangan Meeting	  
5	Ruangan 5	Gedung xyz	200	Ruangan Multifungsi	  
6	test	test	20	test	  
7	ruangan 7	jalan bersama	33	ruangan cetak data	  
8	ruangan 8	gedung besar	54	Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.	  

Main Menu

© Telkom Indonesia 2023 Powered by Yii Framework

Gambar 3. 28 Halaman Pengelolaan Ruangan

Kemudian untuk tampilan selanjutnya adalah halaman pengelolaan ruangan. Jika pada homepage (gambar 3. 27) admin menekan tombol pengelolaan ruangan, maka admin akan langsung diarahkan ke halaman seperti pada gambar 3. 28. Pada halaman ini admin disugahi oleh *list* atau daftar dari seluruh ruangan yang aktif dalam aplikasi Booking App. Dalam tampilan pengelolaan ruangan, admin memiliki beberapa pilihan action untuk dilakukan, seperti menambahkan ruangan baru, *view* ruangan, *edit* ruangan, hapus ruangan atau kembali ke main menu.



Booking Application Home About Contact Login

Home / Tabel Ruangans / Tambahkan Ruangan

## Tambahkan Ruangan

Silahkan mengisi form dibawah ini untuk menambahkan ruangan baru

Nama Ruangan

Lokasi

Kapasitas

Detail Ruangan

Image Files  No file chosen

© Telkom Indonesia 2023 Powered by Yii Framework

Gambar 3. 29 Halaman Tambahkan Ruangan

Jika admin menekan tombol tambahkan ruangan baru, maka admin akan diarahkan ke halaman seperti gambar 3. 29. Disini, admin diminta untuk memasukan data agar kelengkapan data ruangan terpenuhi sehingga dapat menjadi ruangan yang aktif dalam aplikasi Booking App. Data yang diminta seperti nama ruangan, lokasi, kapasitas, detail ruangan dan juga gambar dari ruangan. Admin dapat menyimpan lebih dari satu gambar sebagai data pelengkap dalam ruangan. Gambar yang tersimpan dalam aplikasi bersifat penyimpanan lokal di dalam penyimpanan perangkat.

Disaat admin selesai mengisi data, maka admin diberi dua pilihan. Pilihan pertama, jika admin menekan tombol simpan, maka data yang sudah diinput akan tersimpan kedalam database lalu tampilan akan diarahkan kembali ke daftar ruangan. Sebaliknya, jika admin menekan tombol batal, maka data yang dimasukan

tidak akan tersimpan dan admin kembali ke daftar ruangan. Kemudian dalam daftar ruangan terdapat tombol action yang dapat dipilih oleh admin, berikut tampilan yang ada dari tombol *action* tersebut.

The screenshot shows a web application interface for a booking system. At the top, there is a navigation bar with links for 'Home', 'About', 'Contact', and 'Login'. Below the navigation bar, the page title is 'Detail dari : Ruangan 9'. The main content area features a table with the following details:

<b>Id Ruangan</b>	9.
<b>Nama Ruangan</b>	Ruangan 9.
<b>Lokasi</b>	Gedung Telkom 1.
<b>Kapasitas</b>	1000.
<b>Detail Ruangan</b>	Ruangan Theater atau Meeting.

Below the table, there is a section titled 'Foto Ruangan:' which contains a grid of six photographs showing different views of the office space. At the bottom of the page, there are three buttons: 'Update' (blue), 'Delete' (red), and 'Kembali' (blue). The footer of the page includes the copyright notice '© Telkom Indonesia 2023' and the text 'Powered by Yii Framework'.

Gambar 3. 30 Halaman *View* Tombol *Action*

Tampilan dari gambar 3. 30 adalah contoh jika admin memilih tombol *view* pada daftar ruangan. Admin akan disajikan oleh data lengkap suatu ruangan yang berisi nama ruangan, lokasi, kapasitas hingga foto ruangan. Dari halaman ini juga admin dapat mengubah data atau menghapus data ruangan.

Home / Tabel Ruangans / 9 / Update

## Update Data Ruangan: Ruangan 9

Nama Ruangan

Lokasi

Kapasitas

Detail Ruangan

Image Files  No file chosen

Gambar 3. 31 Halaman Update Data

Sama seperti tombol tambahkan ruangan baru, tombol *Update* data akan memberikan *form* untuk memasukan kembali data atas sebuah ruangan. Seperti pada gambar 3. 31 untuk tampilan tombol *delete* (hapus data) memiliki tampilan yang sederhana yakni akan muncul *alert* di bagian atas halaman aplikasi yang berbunyi “Are you sure you want to delete this item?” dengan dua pilihan, “ok” dan “cancel”. Jika admin memilih “ok”, maka data akan terhapus. Sebaliknya jika “cancel”, maka alert akan kembali tertutup dan tidak terjadi apa apa pada data.










Booking Application Home About Contact Login

Home / Form Entry

## Form Entry

Tambahkan Pesanan Baru

Showing 1-3 of 3 items.

#	Id Pesanan	Id Ruang	Nik	Status	Tanggal Dan Waktu	Jumah Peserta	Nama Kegiatan	Keterangan Pesanan	Fasilitas Tambahan	
1	1	1	1234		2022-10-20 13:37:03	20	Rapat	-		  
2	2	4	2345	Aproved	2022-11-24 13:41:08	35	Meeting	Rapat project baru	proyektor	  
3	3	4	2345	Aproved	2022-11-24 13:41:08	35	Meeting	Rapat project baru	proyektor	  

© Telkom Indonesia 2023 Powered by Yii Framework

Gambar 3. 32 Halaman *Form Entry*

Kemudian untuk halaman selanjutnya adalah *form entry*. Seperti pada gambar 3. 32 disaat admin menekan tombol *form entry* di bagian menu utama, akan muncul tampilan yang berisi daftar pemesanan (entry) ruangan. Daftar ini memiliki struktur seperti daftar ruangan di tampilan sebelumnya hanya saja informasi yang diberikan sesuai dengan data pesanan karyawan. Terdapat tombol tambahkan pesanan baru untuk mengisi data *entry* pemesanan dan akan diberikan formulir untuk mengisi kelengkapan datanya. Kemudian untuk tombol *action* di daftar memiliki fungsi yang sama dengan tombol *action* sebelumnya.

Home / Tabel Pesanans / Create Tabel Pesanan

## Create Tabel Pesanan

Id Ruangan

Ruangan 1

Ruangan

Nik

Status

Tanggal Dan Waktu

Jumah Peserta

Nama Kegiatan

Keterangan Pesanan

Fasilitas Tambahan

Save

© Telkom Indonesia 2023 Powered by Yii Framework

Gambar 3. 33 Tampilan Halaman *Form Entry*

Kemudian jika menekan tombol tambahkan pesanan baru, maka akan muncul *form* seperti di gambar 3. 33 yang meminta *input* atas data pemesanan. Pengerjaan pembuatan aplikasi saat menjalani program praktik kerja magang di PT Telkom Indonesia hanya sampai pada halaman menu *entry form*. Pembuatan aplikasi Booking App oleh peserta magang belum bisa diselesaikan hingga tahap penyelesaian. hal tersebut dikarenakan terbatasnya waktu kontrak yang sudah disepakati perusahaan sehingga pengerjaan harus terhenti di halaman admin.

### 3.3 Kendala yang Ditemukan

Saat menjalani praktik kerja magang, terdapat beberapa masalah yang dihadapi oleh peserta magang. Berikut merupakan beberapa kendala yang dihadapi dalam proses praktik kerja magang.

1. Kesulitan melakukan komunikasi secara langsung dengan pembimbing (supervisor) praktik kerja magang. Kendala ini terjadi dikarenakan peserta magang melakukan komunikasi melalui media *online* seperti *online meeting* atau obrolan *online*.
2. Framework Yii 2 belum pernah digunakan oleh peserta magang sebelumnya. Hal tersebut akan mempengaruhi kinerja dari pengembangan aplikasi Booking App.
3. Tidak adanya rekan kerja saat melakukan praktik kerja magang. Dalam praktik perkuliahan, peserta magang terbiasa untuk melakukan pengerjaan *project* bersama tim sehingga pengerjaan dapat disitribusikan (dibagi) Bersama.
4. Praktik kerja magang dilakukan secara *Work From Home* (WFH). Hal ini menjadi kendala dikarenakan peserta magang tidak dapat merasakan praktik kerja magang secara langsung dalam perusahaan.

### 3.4 Solusi atas Kendala yang Ditemukan

Dari kendala yang sudah dijabarkan, berikut adalah solusi yang dilakukan oleh peserta magang untuk mengatasi kendala yang terjadi di praktik kerja magang.

1. Lebih banyak menjadwalkan dan mengumpulkan topik diskusi sebelum melakukan komunikasi. Solusi ini dilakukan agar setiap *online meet* atau melakukan komunikasi menghasilkan pembicaraan yang matang dan poin yang didiskusikan tersampaikan dengan pembimbing magang.
2. Mempelajari dan mendalami lebih banyak mengenai metode dan package dari Framework Yii 2 melalui buku dan media online. Akan

terlambat jika peserta magang berfokus pada pembelajaran dalam Framework Yii 2 saja. Untuk itu solusi yang dilakukan adalah mempelajari metode pengerjaan sembari mengerjakan agenda dan tugas yang diberikan pembimbing magang.

3. Menyesuaikan metode pengerjaan secara individu dalam pengerjaan *project*. Solusi yang dapat dilakukan untuk mengatasi kendala ini adalah mengubah cara mengerjakan sebuah project tanpa adanya tim.
4. Menyediakan lingkungan kerja yang mendukung di lokasi sekarang. Walau tidak berkesempatan terjun langsung di perusahaan, peserta magang dapat mengkondisikan lingkungan kerja saat ini untuk mengerjakan proses praktik kerja magang.