

BAB IV

ANALISIS KERJA PRAKTEK

4.1 Analisis Sistem yang Berjalan

4.1.1 Prosedur/Tahapan yang Dianalisis Berdasarkan Bagiannya

Sebelumnya petani yang kemudian nanti disebutnya adalah pengguna, pengguna melakukan pendaftaran terlebih dahulu ke bagian *admin eFishery feeder* untuk mendapatkan akses login pada aplikasi *Smart Feeder* nya. Setelah mendapatkan *username* dan *password* baru pengguna bisa mengakses dan langsung mencoba untuk menggunakan produknya. Kemudian di bawah ini adalah prosedur dalam aplikasi *Smart Feeder*-nya.

Prosedur Menambah Feeder Baru :

1. Pengguna membuka aplikasi
2. Pilih *button* tambah di *homescreen*
3. *Scan barcode* yang ada di *Feeder*
4. Isi *form input* informasi *feed*
5. Isi *form input* jadwal harian dan jumlah pakan (kg)
6. Isi *form input* informasi budidaya
7. Selesai.

Prosedur mengatur Kalibrasi dan Jarak lontar :

1. Pilih kolam yang mau diatur *feeder*-nya
2. Pilih *feeder*
3. Pilih pengaturan *feeder*
4. Lalu kemudian pilih perbarui kalibrasi

5. Atur waktu kalibrasi
6. Setelah itu, kembali ke menu pengaturan *feeder*, lalu kemudian pilih jarak lontan
7. Atur jarak lontan
8. Jalankan *feeder*
9. Selesai.

Prosedur mengatur Jadwal *Feeder* :

1. Pilih kolam yang mau diatur di *homescreen*
2. Lalu kemudian pilih Jadwal
3. Pilih tombol ubah
4. Memasukan informasi jadwal harian
5. Memilih jenis jadwal harian yang terdapat 3 pilihan yaitu *basic*, *advance* dan *continuous*
6. Pilih tombol Tambah
7. Memasukan informasi Jam Dimulai, Besaran *feed* dalam (kg), dan informasi Jam Selesai
8. Tekan Simpan
9. Lalu kemudian, tekan *synchronize feeder*
10. Selesai.

Prosedur mengatur informasi tentang *Feeder* dan *Farming* :

1. Pilih kolam di *homescreen*
2. Tekan *Info*
3. Pilih tombol Ubah *farming information*

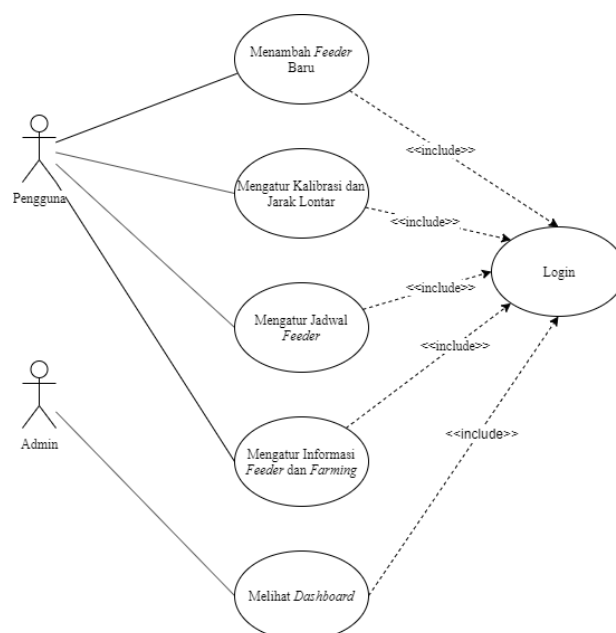
4. Lalu Masukan informasi *farming*
5. Tekan tombol Simpan
6. Selesai.

Prosedur melihat Informasi *Dashboard Smart Feeder* :

1. Pengguna membuka aplikasi *smart feeder*
2. Di halaman *home screen* ada tombol *dashboard eFishery*, lalu tekan tombol tersebut
3. Kemudian muncul informasi tentang *dashboard eFishery* tersebut
4. Selesai.

4.1.2 Use case Diagram

Dalam menganalisis sistem yang berjalan tentunya harus membuat suatu pemodelan atau gambaran sistem yaitu *use case diagram*. Berikut ini adalah *use case diagram* yang digunakan pada sistem yang berjalan.



Gambar 5 Use case diagram sistem yang berjalan

4.1.2.1 Definisi Aktor dan Deskripsinya

Berikut ini adalah Definisi Aktor dan Deskripsinya, penulis menuliskan karena agar supaya lebih bisa mengerti dan mudah dipahami definisinya.

Tabel 2 Definisi aktor

No.	Aktor	Deskripsi
1.	Pengguna	Pengguna adalah orang yang menggunakan aplikasi tersebut untuk keperluan tertentu, seperti yang dibahas di sini adalah untuk tujuan pengelolaan pakan ikan atau udang.
2.	Admin	Admin adalah orang yang melakukan tugas untuk mengurus hal-hal terkait pengelolaan di dalam sistem itu sendiri.

4.1.2.2 Definisi Use Case dan Deskripsinya

Berikut ini penulis deskripsikan definisi *Use Case* yang penulis gunakan dalam laporan ini untuk tujuan lebih melihat detail deskripsi serta juga agar supaya lebih bisa dimengerti.

Tabel 3 Definisi Use Case

No.	Use Case	Deskripsi
1.	Menambah <i>Feeder</i> Baru	Proses untuk menambah <i>feeder</i> baru yang dibeli oleh pengguna untuk digunakan pada kolam nya dengan cara <i>scan barcode</i> .
2.	Mengatur Kalibrasi dan Jarak Lontar	Proses pengelolaan Data yang ada di aplikasi oleh pengguna untuk kemudian nanti diproses oleh mesin sebelum dilakukan kalibrasi dan juga sebagai bahan laporan.

3.	Mengatur Jadwal <i>Feeder</i>	Proses untuk pengaturan jadwal feeder, jadi pengguna dapat melakukan pengaturan untuk kapan-kapan saja <i>feeder</i> akan berjalan.
4.	Mengatur Informasi <i>Feeder</i> dan <i>Farming</i>	Proses untuk melakukan pengaturan di dalam aplikasi terkait <i>feeder</i> dan <i>farming</i> .
5.	Melihat <i>Dashboard</i>	Proses untuk melihat informasi yang ada pada dashboard terkait pemberian pakan, pengelolaan data, kesehatan ikan dan lainnya.
5.	<i>Login</i>	Proses untuk melakukan <i>login</i> sebelum masuk ke sistem aplikasinya.

4.1.3 Skenario *Use case*

Pada skenario *use case* penulis melakukan penilaian pada skenario tersebut untuk memudahkan analisis skenario pada tahapan-tahapan selanjutnya. Berikut ini adalah tahapan-tahapan skenario *use case* pada aplikasi *Smart Feeder* :

4.1.3.1 Skenario Menambah *Feeder* Baru

Berikut ini adalah skenario menambah *feeder* baru pada aplikasi *smart feeder* eFishery app yang sedang berjalan saat ini.

Tabel 4 Skenario Menambah *Feeder* Baru

Identifikasi	
Nama	Menambah <i>Feeder</i> Baru
Tujuan	Untuk melakukan penambahan <i>feeder</i> baru dengan cara <i>scan barcode</i> pada

	<i>feeder.</i>
Deskripsi	
Tipe	Aktivitas pendukung
Aktor	Pengguna
<i>Main Flow</i>	
Pengguna	Sistem
1. Pengguna membuka aplikasi	
	2. Menampilkan halaman utama
3. Memilih tombol Tambah di halaman utama	
	4. Menampilkan halaman tambah <i>feeder</i> dan <i>form input</i>
5. Menekan tombol Tambah dan kemudian Melakukan <i>scan barcode</i>	
	6. Menampilkan halaman <i>scan barcode</i>
7. Mengisi <i>form input</i> informasi <i>feeder</i>	
	8. Menyimpan data di sistem dan menampilkan <i>feeder</i> baru
9. Pengguna mendapatkan <i>feeder</i> baru dan langsung bisa mengelola <i>feeder</i>	
Kondisi Akhir	
Pengguna mendapatkan <i>feeder</i> baru dan bisa langsung melakukan pengelolaan <i>feeder</i> untuk pemberian pakan.	

4.1.3.2 Skenario Mengatur Kalibrasi dan Jarak Lontar

Berikut ini adalah skenario mengatur kalibrasi dan jarak lontar pada aplikasi *smart feeder* eFishery app yang sedang berjalan saat ini.

Tabel 5 Skenario Mengatur Kalibrasi dan Jarak Lontar

Identifikasi	
Nama	Mengatur Kalibrasi dan Jarak Lontar
Tujuan	Mengatur dan sekaligus menjalankan mesin <i>smart feeder</i>
Deskripsi	
Tipe	Aktivitas utama
Aktor	Pengguna
Main Flow	
Pengguna	Sistem
1. Pengguna membuka aplikasi	
	2. Menampilkan halaman <i>home screen</i>
3. Memilih kolam yang mau diatur di <i>home screen</i>	
	4. Menampilkan Detail kolam beserta data <i>feeder</i>
5. Pilih <i>feeder</i>	
6. Pilih <i>setting feeder</i>	
	7. Menampilkan Perbarui kalibrasi dan atur jadwal kalibrasi

8. Mengatur jadwal kalibrasi dan jarak lontar	
	9. Menampilkan halaman <i>setting feeder</i>
10. Pengguna selesai menjalankan mesin <i>smart feeder</i> -nya	
Kondisi Akhir	
Pengguna mendapatkan informasi dan sudah bisa menjalankan mesin <i>smart feeder</i> eFishery dan cukup menikmati, ikan dikasih makan.	

4.1.3.3 Skenario Mengatur Jadwal *Feeder*

Berikut ini adalah skenario mengatur jadwal *feeder* pada aplikasi *smart feeder* eFishery app yang gunanya untuk pengguna bisa memberikan jadwal kapan mesin akan berjalan sesuai tingkat kelaparan ikan atau udang.

Tabel 6 Skenario Mengatur Jadwal Feeder

Identifikasi	
Nama	Mengatur Jadwal <i>Feeder</i>
Tujuan	Untuk mengatur jadwal kapan mesin akan berjalan sesuai dengan keinginan si pengguna dan info tingkat kelaparan ikan dan udangnya
Deskripsi	
Tipe	Aktivitas pendukung
Aktor	Pengguna
Main Flow	

Pengguna	Sistem
1. Pengguna membuka aplikasi	
	2. Menampilkan halaman <i>home screen</i>
3. Pilih kolam yang mau diatur jadwal <i>feeder</i> -nya	
	4. Menampilkan halaman detail kolam
5. Kemudian pilih Jadwal	
	6. Menampilkan halaman Jadwal
7. Pilih <i>button</i> ubah	
	8. Menampilkan halaman ubah dan menampilkan <i>form input</i>
9. Memasukan informasi jadwal harian sesuai si pengguna mau	
	10. Menyimpan data dan mensinkronisasi mesin
11. Pengguna selesai mengatur jadwal	
Kondisi Akhir	
Pengguna selesai mengatur jadwal <i>feeder</i> dan mesin siap untuk digunakan serta otomatis berjalan sesuai fungsinya berdasarkan jadwal yang diberikan.	

4.1.3.4 Skenario Mengatur Informasi Tentang *Feeder* dan *Farming*

Berikut ini adalah skenario mengatur informasi tentang *feeder* dan *farming* pada aplikasi *smart feeder* eFishery app yang kemudian digunakan untuk bahan data pengoperasian mesin agar berjalan sesuai dengan data tersebut.

Tabel 7 Skenario Mengatur Informasi *Feeder* dan *Farming*

Identifikasi	
Nama	Mengatur Informasi <i>Feeder</i> dan <i>Farming</i>
Tujuan	Untuk melakukan pengaturan informasi dan data kemudian digunakan untuk menjalankan mesin
Deskripsi	
Tipe	Aktivitas pendukung
Aktor	Pengguna
Main Flow	
Pengguna	Sistem
1. Pengguna membuka aplikasi	
	2. Menampilkan halaman utama <i>home screen</i>
3. Pilih kolam	
	4. Menampilkan halaman <i>detail</i> kolam
5. Pilih info	
	6. Menampilkan halaman info
7. Pilih <i>edit farming information</i>	

	8. Menampilkan halaman <i>edit farming information</i>
9. Melihat, mengubah dan memasukan data terkait informasi <i>farming</i>	
	10. Menyimpan data
Kondisi Akhir	
Pengguna bisa mendapatkan data <i>farming</i> dan mengubah informasi tersebut.	

4.1.3.5 Skenario Melihat *Dashboard*

Berikut ini adalah skenario melihat dashboard pada aplikasi *smart feeder* eFishery app yang digunakan untuk hanya menampilkan informasi seputar kalibrasi, kesehatan ikan dan udang, jadwal, serta informasi lain terkait *farming* dan *feeder*.

Tabel 8 Skenario Melihat *Dashboard*

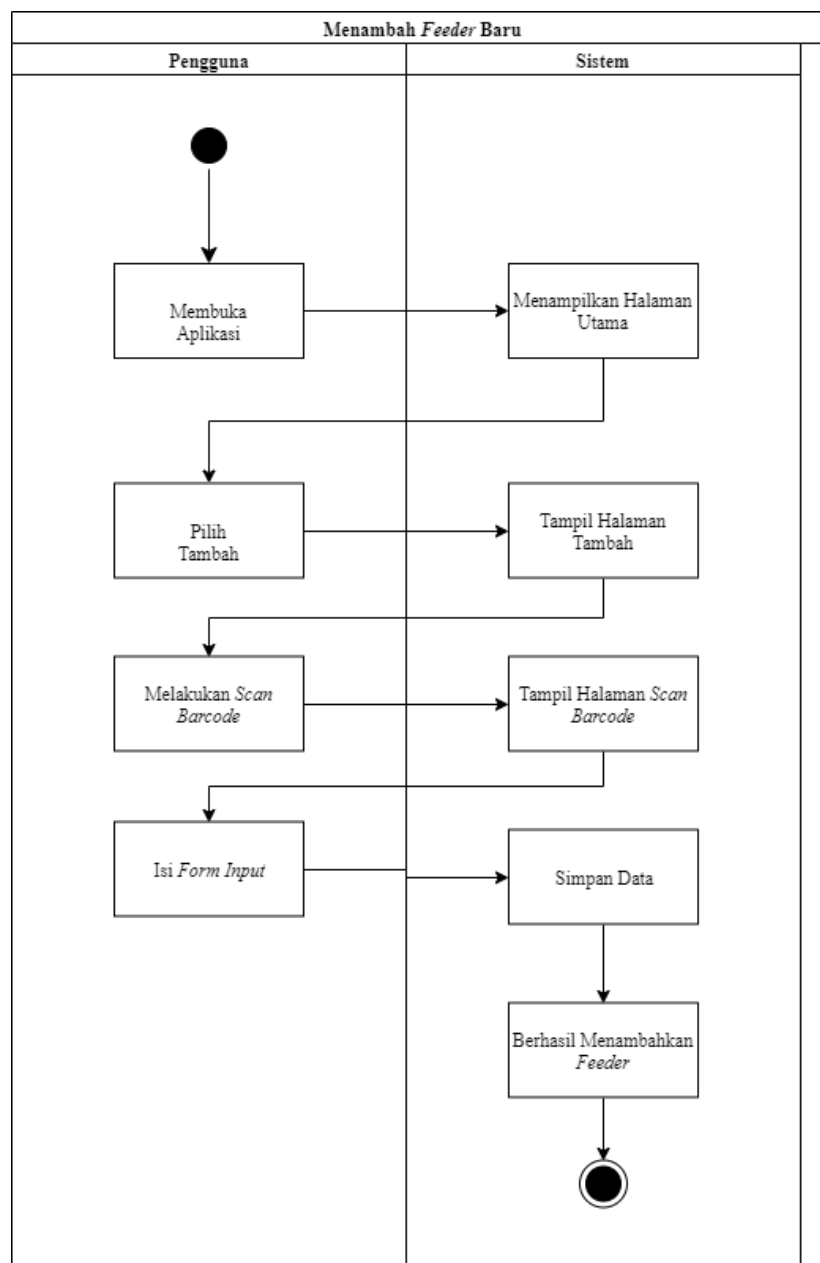
Identifikasi	
Nama	Melihat <i>Dashboard</i>
Tujuan	Untuk melihat informasi di <i>dashboard</i>
Deskripsi	
Tipe	Aktivitas utama
Aktor	Pengguna, Admin
Main Flow	
Pengguna	Sistem
1. Pengguna dan admin membuka	

aplikasi dan <i>website dashboard</i>	
	2. Menampilkan halaman utama pada aplikasi
3. Pengguna memilih menu <i>dashboard</i> pada aplikasi, sedangkan admin langsung membuka <i>website dashboard</i>	
	4. Menampilkan halaman penuh <i>website dashboard</i>
5. Pengguna dan admin mendapatkan informasi pada <i>dashboard</i>	
Kondisi Akhir	
Pengguna mendapatkan informasi <i>dashboard</i> dari website.	

4.1.4 Activity Diagram

4.1.4.1 Activity Diagram Menambah Feeder Baru

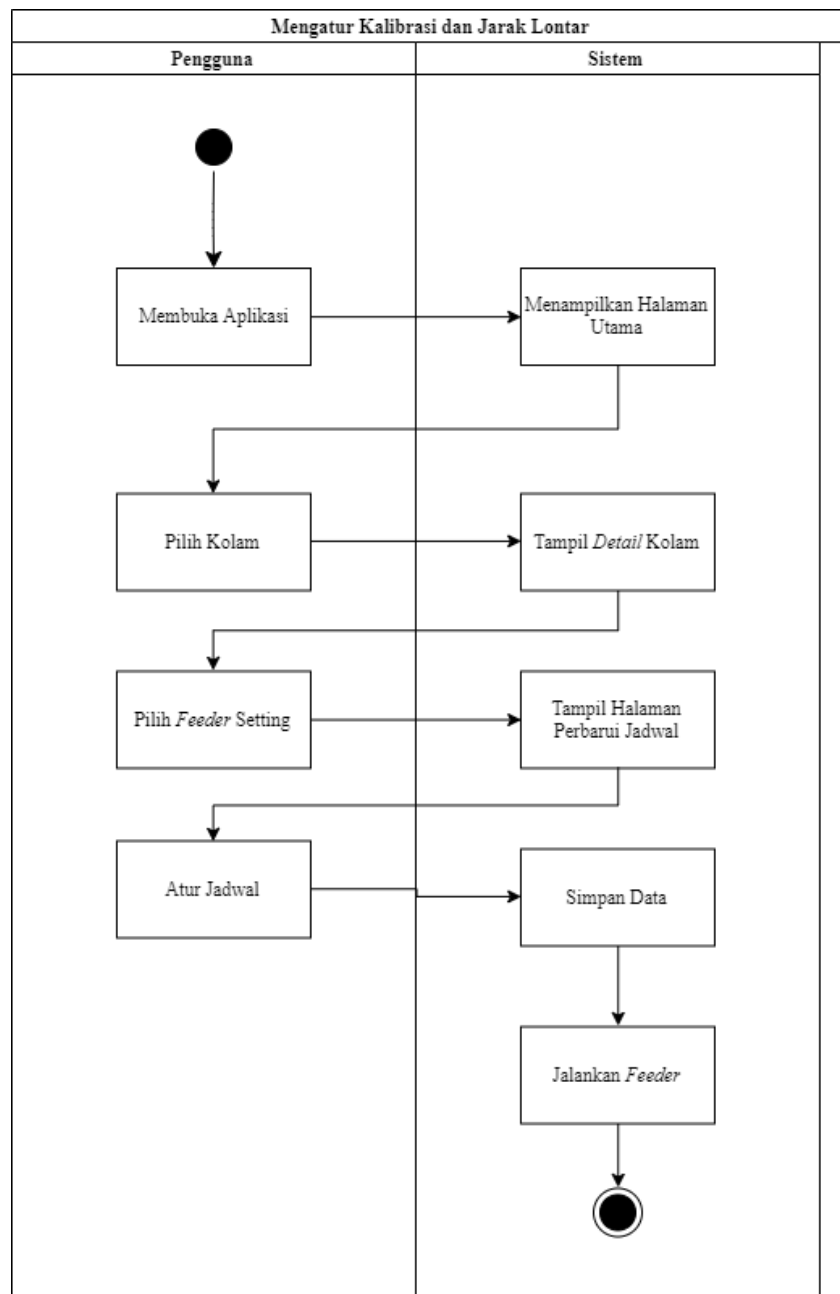
Pada *activity diagram* menambah *feeder* baru, pengguna harus melakukan scan terlebih dahulu dan terkadang pengguna yang awam tidak tahu harus bagaimana.



Gambar 6 Activity Diagram Menambah Feeder Baru

4.1.4.2 Activity Diagram Mengatur Kalibrasi dan Jarak Lontar

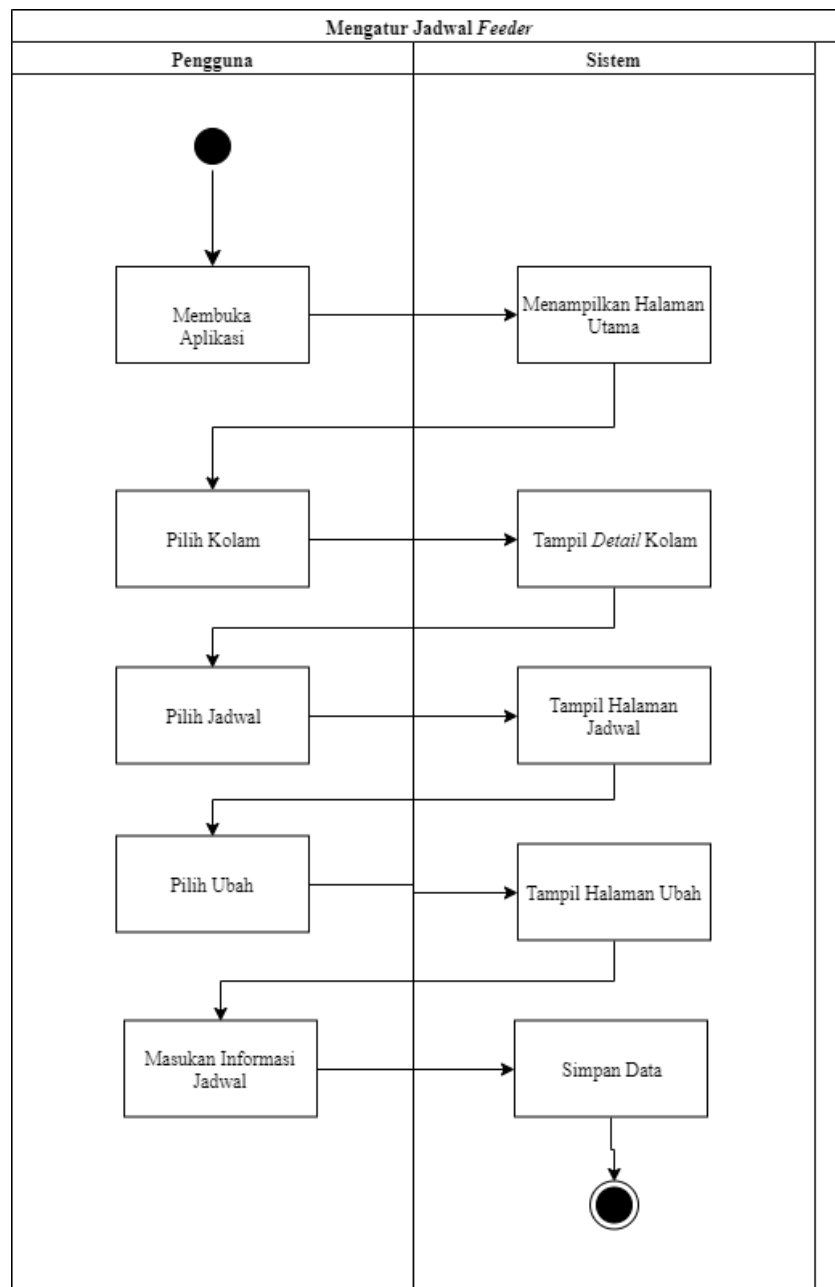
Pada *activity diagram* mengatur kalibrasi dan jarak lontar, pada tahapan ini pengguna sering kesulitan melakukan kalibrasi dan tidak tahu apa sebenarnya kalibrasi.



Gambar 7 Activity Diagram Mengatur Kalibrasi dan Jarak Lontar

4.1.4.3 Activity Diagram Mengatur Jadwal Feeder

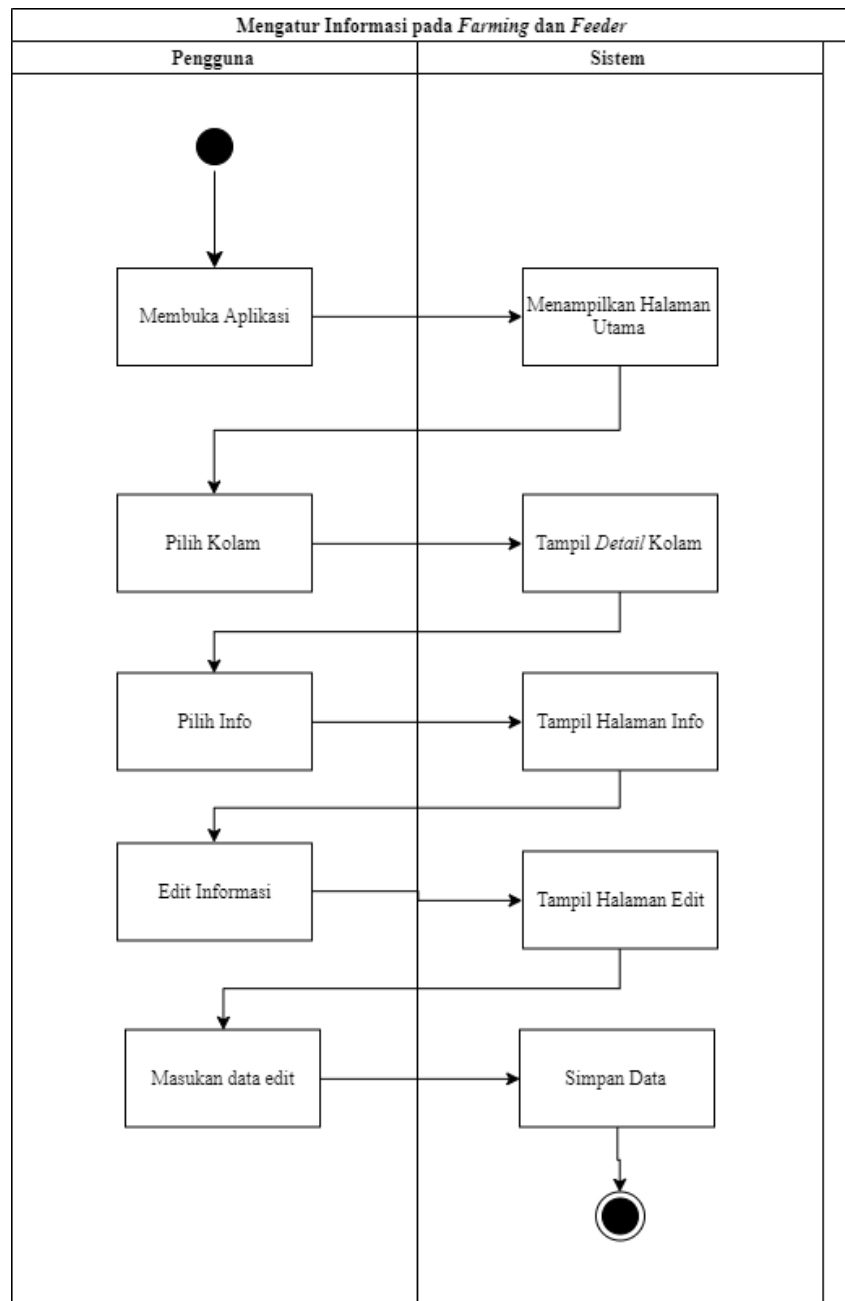
Pada *activity diagram* mengatur jadwal *feeder*, di tahapan ini pengguna sering kali kesulitan disaat membuat jadwal karena harus melewati tahap pilih kolam terlebih dahulu.



Gambar 8 Activity Diagram Mengatur Jadwal Feeder

4.1.4.4 Activity Diagram Mengatur Informasi *Farming* dan *Feeder*

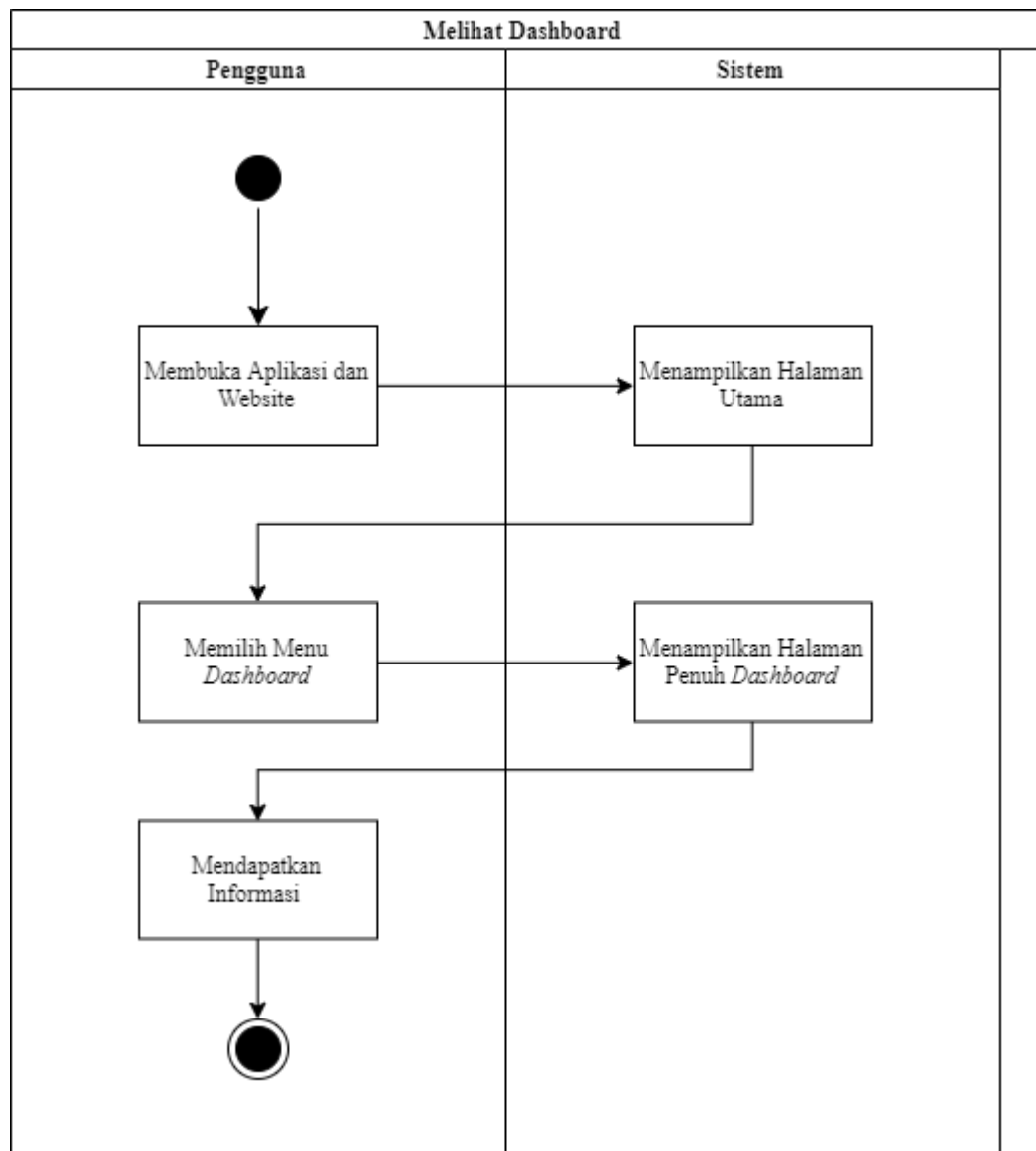
Pada *activity diagram* mengatur informasi *farming* dan *feeder*, pengguna sering kesulitan memahami halaman ini, karena halaman ini dianggap oleh pengguna adalah hanya info, tetapi harus melakukan edit informasi.



Gambar 9 Activity Diagram Mengatur Informasi pada *Farming* dan *Feeder*

4.1.4.5 Activity Diagram Melihat Dashboard

Pada *activity diagram* melihat *dashboard*, pengguna bias melihat informasi mengenai yang terkait dengan mesin, kesehatan ikan dengan cukup klik dashboard di halaman awal aplikasi.



Gambar 10 Activity Diagram Melihat Dashboard

4.1.5 Evaluasi Sistem yang Berjalan

Setelah melakukan analisis pada sistem yang sedang berjalan. Penulis melakukan diskusi dengan Manajer Proyek dari eFishery app itu sendiri terkait evaluasi sistem. Kemudian Penulis menyampaikan bahwa setelah dianalisis sistem tersebut memiliki beberapa permasalahan yang perlu melakukan evaluasi. Berikut permasalahan terkait sistem tersebut :

Tabel 9 Evaluasi Sistem yang Berjalan

No.	Permasalahan	Rencana	Bagian
1.	Aplikasi <i>Smart Feeder</i> eFishery tersebut memiliki kesulitan yang user bingung saat menggunakan <i>flow</i> aplikasi saat ini	Merancang <i>flow</i> baru yang memudahkan pengguna	Sistem Aplikasi
2.	Sistem aplikasi yang sulit dimengerti oleh user terkait <i>user interface</i> -nya	Merancang <i>User Interface</i> yang memudahkan pengguna	Sistem Aplikasi bagi <i>User Interface</i>

4.2 Perancangan Sistem

Pada perancangan sistem yang baru ini, penulis akan membahas mengenai lanjutan dan evaluasi dari sistem yang sedang berjalan tersebut yang sudah dibahas di atas. Tentunya pada pembahasan ini penulis memberikan suatu sistem yang baru tetapi tidak akan jauh dari sistem yang telah ada, dimana penulis akan membuat *flow* aplikasi yang lebih mudah, bisa digunakan oleh pengguna yang *non tech savvy* dan juga akan sedikit merubah sistem dari segi *user interface*-nya. Dengan adanya sistem baru yang penulis akan buat ini, bisa meningkatkan produktivitas petani dalam mengelola pakan ikan maupun udang dan tambak secara keseluruhan, serta

juga bisa meningkatkan pendapatan atau penjualan perusahaan dari peningkatan, perubahan *flow* dan *user interface*-nya.

4.2.1 Tujuan Perancangan Sistem

Tujuan perancangan sistem ini adalah untuk memberikan suatu evaluasi bagi perusahaan dari segi peningkatan dan pembaharuan *flow* aplikasi dan juga *user interface* yang mudah digunakan oleh pengguna yang *non tech savvy*. Selain dari segi sistem, tujuan perancangan sistem ini bisa memberikan suatu gambaran sistem yang pengguna butuhkan dan bisa meningkatkan pendapatan perusahaan.

4.2.2 Gambaran Umum Sistem yang Diusulkan

Sistem yang ingin diusulkan ini akan sedikit berbeda dari sistem sebelumnya. Ada beberapa *flow* yang dihapus dan ada juga yang ditambah. Kemudian untuk *flow* penambahan *feeder* itu akan penulis hapus dan diganti dengan *flow* yang lebih sederhana sehingga pengguna tidak harus melakukan *scanning* untuk menambahkan *feeder* baru tetapi pengguna ketika mau membeli atau menyewa produk *smart feeder* eFishery akan langsung ditambahkan otomatis mesin ke dalam aplikasi sesuai dengan pembelian atau penyewaannya. Selain itu, pengguna juga akan disediakan *username* dan *password* dari pihak eFishery. Pengurangan tahapan *flow* yang sudah ada agar pengguna lebih mudah melakukan pengelolaan dan agar lebih *friendly* menggunakan aplikasi tersebut serta juga penambahan fitur tips dan tutorial penggunaan dari aplikasi juga info seputar akuakultur dan perawatan mesin.

4.2.3 Perancangan Prosedur yang Diusulkan

Seperti yang telah penulis bahas di sub judul sebelumnya bahwa prosedur menambah *feeder* akan penulis hapus dan diganti dengan otomatis penambahan *feeder* dalam aplikasi saat waktu pembelian maupun penyewaan. Berikut adalah perancangan prosedur yang diusulkan dalam penulisan laporan kerja praktek ini :

Prosedur Mengatur Kalibrasi dan Jarak Lontar :

1. Pengguna membuka aplikasi
2. Pilih menu kalibrasi
3. Pilih Kolam
4. Atur *feeder* yang mana
5. Atur kalibrasi dan jarak lontar pada *feeder* yang dipilih
6. Jalankan *feeder*
7. Selesai

Prosedur Mengatur Jadwal *Feeder* :

1. Pengguna membuka aplikasi
2. Pilih menu Jadwal
3. Tambah Jadwal
4. Masukkan informasi jadwal
5. Simpan jadwal
6. Jalankan
7. Selesai

Prosedur Mengatur Informasi *Feeder* dan *Farming* :

1. Pengguna membuka aplikasi

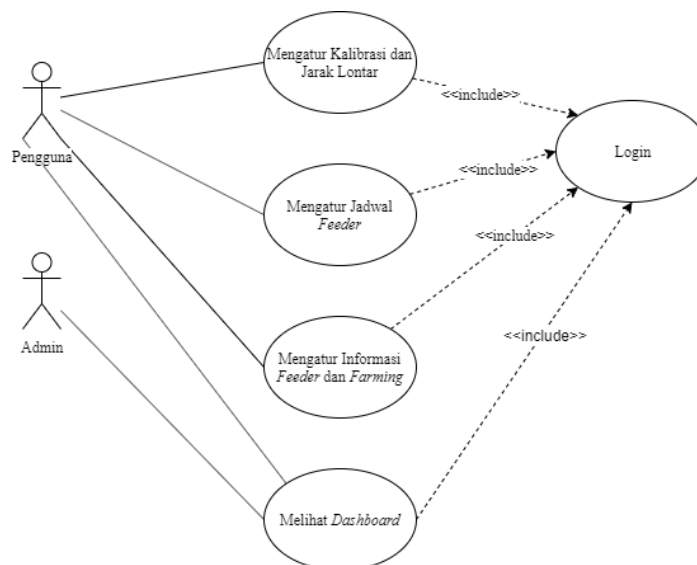
2. Pilih menu info
3. Pilih edit informasi
4. Masukkan informasi
5. Simpan
6. Selesai

Prosedur melihat Informasi *Dashboard Smart Feeder* :

1. Pengguna membuka aplikasi *smart feeder*
2. Di halaman *home screen* ada tombol *dashboard eFishery*, lalu tekan tombol tersebut
3. Kemudian muncul informasi tentang *dashboard eFishery* tersebut
4. Selesai.

4.2.4 Use case Diagram

Setelah sistem yang berjalan sudah dianalisis, kemudian penulis membuat *use case diagram* untuk sistem yang diusulkan. Berikut gambarannya :



Gambar 11 Use Case Diagram Sistem yang Diusulkan

4.2.5 Skenario *Use case*

4.2.5.1 Skenario Mengatur Kalibrasi dan Jarak Lontar yang Diusulkan

Berikut ini adalah skenario mengatur kalibrasi dan jarak lontar pada aplikasi *smart feeder* eFishery app yang diusulkan.

Tabel 10 Skenario Mengatur Kalibrasi dan Jarak Lontar yang Diusulkan

Identifikasi	
Nama	Mengatur Kalibrasi dan Jarak Lontar
Tujuan	Mengatur dan sekaligus menjalankan mesin <i>smart feeder</i>
Deskripsi	
Tipe	Aktivitas utama
Aktor	Pengguna
Main Flow	
Pengguna	Sistem
1. Pengguna membuka aplikasi	
	2. Menampilkan halaman <i>home screen</i>
3. Pilih menu Kalibrasi	
	4. Menampilkan Menu kalibrasi
5. Pilih Kolam	
	6. Menampilkan halaman kolam yang dipilih
7. Atur <i>feeder</i> yang mana	
8. Atur Kalibrasi dan Jarak	

Lontar	
	9. Sistem menyimpan data
	10. Jalankan
Kondisi Akhir	
Menjalankan kalibrasi pada aplikasi dengan mudah.	

4.2.5.2 Skenario Mengatur Jadwal *Feeder*

Berikut ini adalah skenario mengatur jadwal *feeder* pada aplikasi *smart feeder* eFishery app yang diusulkan.

Tabel 11 Skenario Mengatur Jadwal *Feeder* yang Diusulkan

Identifikasi	
Nama	Mengatur Jadwal <i>Feeder</i>
Tujuan	Untuk mengatur jadwal kapan mesin akan berjalan sesuai dengan keinginan si pengguna dan info tingkat kelaparan ikan dan udangnya
Deskripsi	
Tipe	Aktivitas pendukung
Aktor	Pengguna
Main Flow	
Pengguna	Sistem
1. Pengguna membuka aplikasi	
	2. Menampilkan halaman <i>home</i>

	<i>screen</i>
3. Pilih menu Jadwal	
	4. Menampilkan halaman jadwal
5. Kemudian pilih Pilih Kolam yang mana	
	6. Menampilkan Detail halaman kolam
7. Pilih <i>feeder</i>	
	8. Menampilkan halaman detail <i>feeder</i>
9. Pilih Tambah Jadwal	
	10. Menampilkan Halaman Tambah Jadwal
11. Memasukan informasi jadwal	
	12. Simpan data dan Jalankan <i>Feeder</i>
Kondisi Akhir	
Pengguna lebih mudah menyelesaikan mengatur jadwal <i>feeder</i> dan mesin siap untuk digunakan serta otomatis berjalan sesuai fungsinya berdasarkan jadwal yang diberikan.	

4.2.5.3 Skenario Mengatur Informasi *Feeder* dan *Farming*

Berikut ini adalah skenario mengatur informasi *feeder* dan *farming* pada aplikasi *smart feeder* eFishery app yang diusulkan.

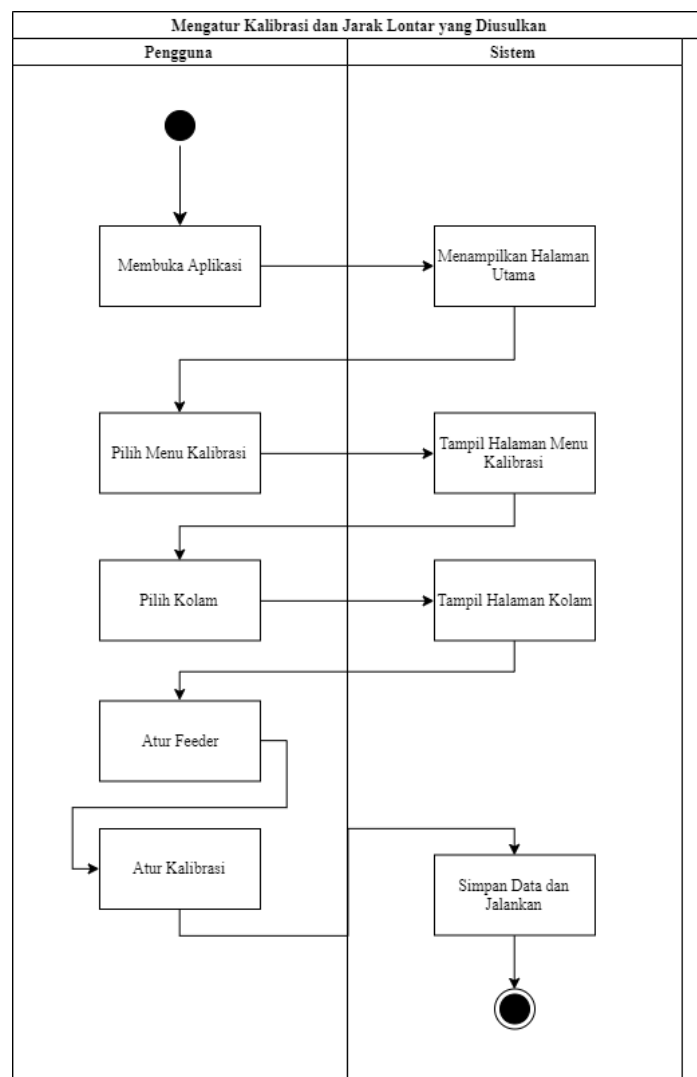
Tabel 12 Skenario Mengatur Informasi *Feeder* dan *Farming* yang Diusulkan

Identifikasi	
Nama	Mengatur Informasi <i>Feeder</i> dan <i>Farming</i>
Tujuan	Untuk melakukan pengaturan informasi dan data kemudian digunakan untuk menjalankan mesin
Deskripsi	
Tipe	Aktivitas pendukung
Aktor	Pengguna
Main Flow	
Pengguna	Sistem
1. Pengguna membuka aplikasi	
	2. Menampilkan halaman utama <i>home screen</i>
3. Pilih menu info	
	4. Tampil <i>screen</i> menu info
5. Pilih Kolam	
	6. Menampilkan halaman Kolam
7. Pilih <i>feeder</i>	
	8. Menampilkan halaman feeder
9. Pilih edit informasi	
	11. Menyimpan data
Kondisi Akhir	
Pengguna mendapatkan data <i>farming</i> dan mengubah informasi dengan mudah.	

4.2.6 Activity Diagram

4.2.6.1 Activity Diagram Mengatur Kalibrasi dan Jarak Lontar yang Diusulkan

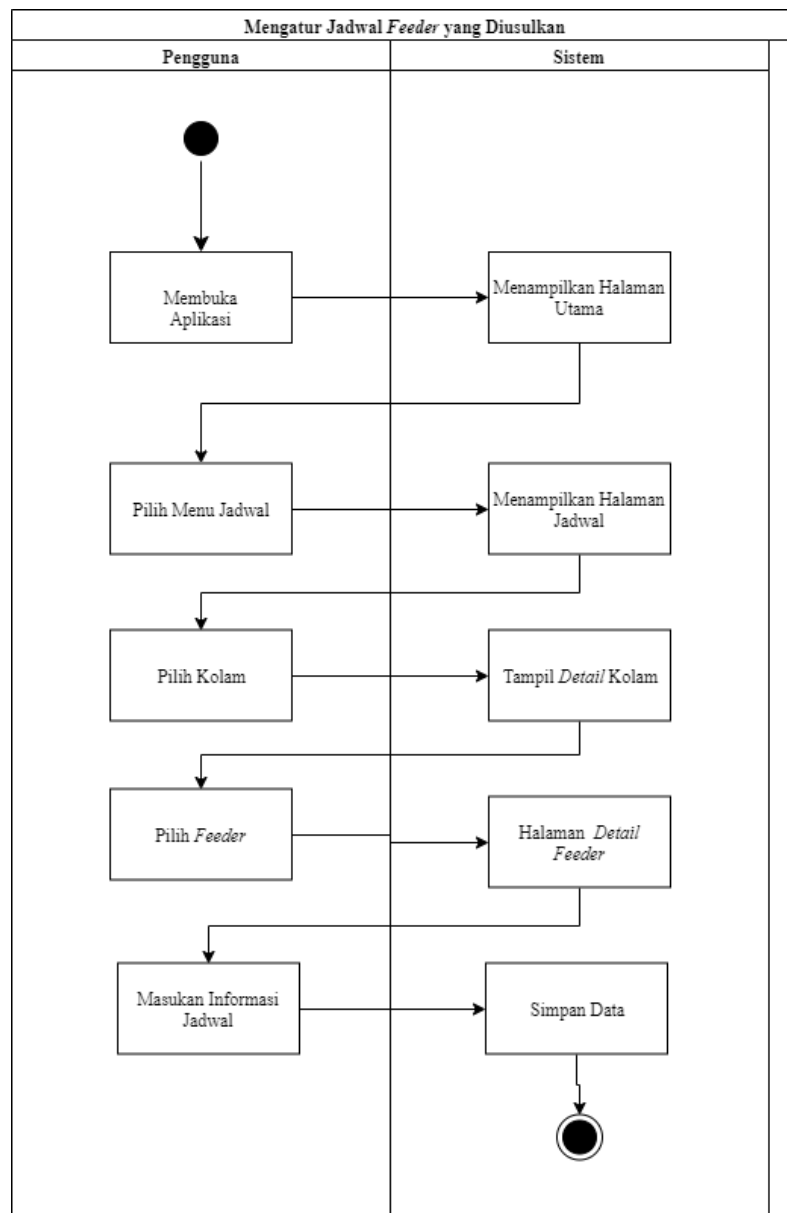
Pada *activity diagram* yang diusulkan mengatur kalibrasi dan jarak lontar, penulis membuat alur yang mudah, yaitu ketika pengguna membuka aplikasi langsung disuguhkan dengan menu utama di halaman *home screen*.



Gambar 12 Activity Diagram Mengatur Kalibrasi dan Jarak Lontar yang Diusulkan

4.2.6.2 Activity Diagram Mengatur Jadwal Feeder yang Diusulkan

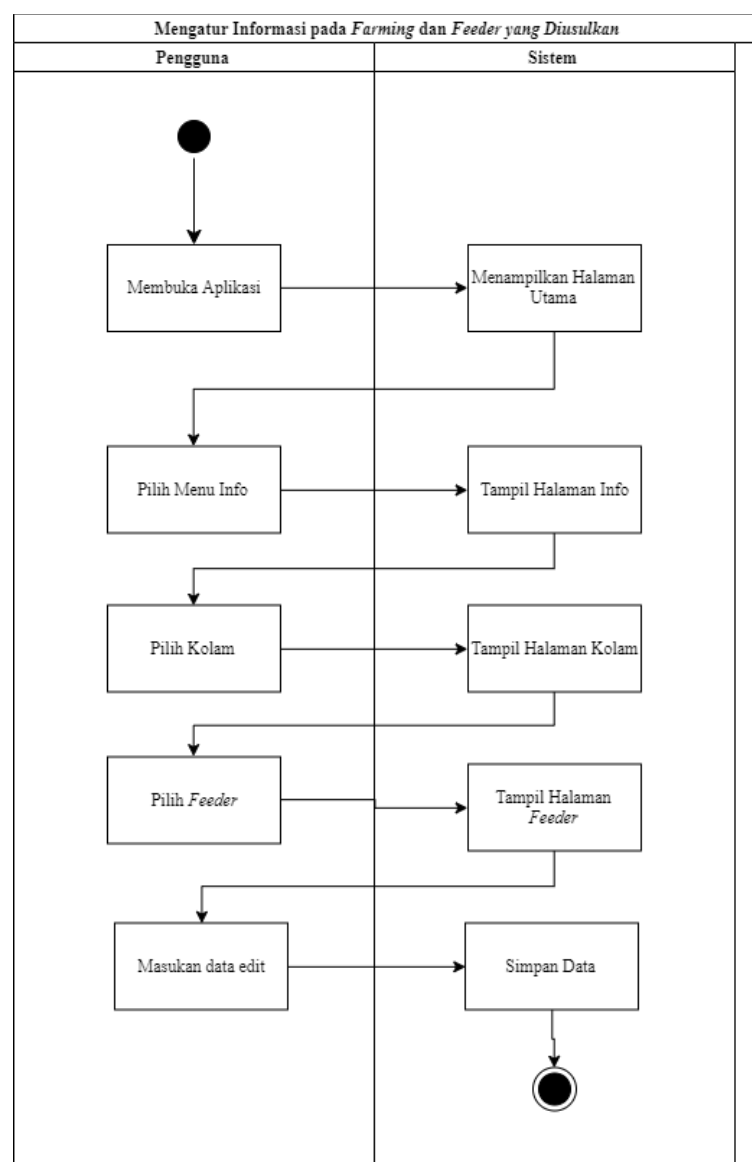
Pada *activity diagram* ini juga, pengguna bias langsung mengakses fitur langsung dari halaman awal atau *home screen*, karena kenapa pengguna lebih bias focus ke aktivitas utama untuk mengatur jadwal feeder lebih mudah juga.



Gambar 13 Activity Diagram Mengatur Jadwal Feeder yang Diusulkan

4.2.6.3 Activity Diagram Informasi Feeder dan Farming yang Diusulkan

Pada *activity diagram* informasi *feeder* dan *farming* system yang diusulkan, pengguna memiliki kemudahan dalam akses dan lebih mudah untuk melihat langsung info terkait ikan atau udang maupun pengelolaannya, kesehatan ikan dan juga pengelolaan air.



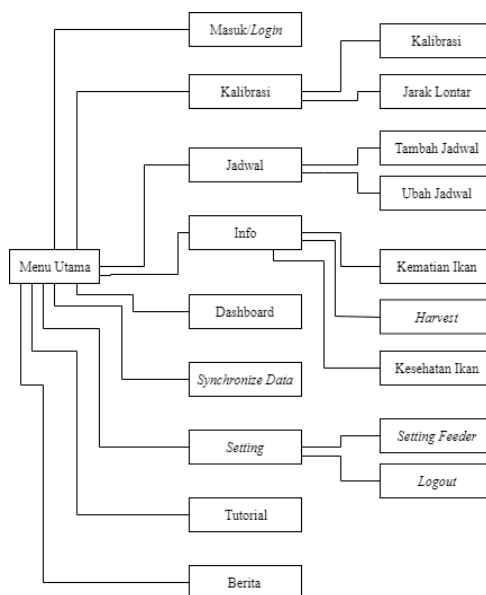
Gambar 14 Activity Diagram Mengatur Informasi pada Farming dan Feeder yang Diusulkan

4.3 Perancangan Antarmuka

Perancangan antarmuka aplikasi *smart feeder* ini penulis cantumkan antarmuka aplikasi yang sekarang dengan antarmuka aplikasi yang penulis rancang sendiri berdasarkan analisis yang penulis lakukan sebelumnya. Sebagian besar penulis mempelajari dan melakukan ini selama kerja praktek di perusahaan eFishery tersebut. Perancangan antarmuka menjadi sangat penting sejak pengguna mulai kesulitan melihat perancangan antarmuka yang telah ada, oleh karena itu penulis membuat versi terbaru dan ini merupakan versi penulis sendiri serta sudah berdiskusi dengan manajer proyek eFishery.

4.3.1 Struktur Menu

Berikut ini adalah struktur menu aplikasi *smart feeder* yang diusulkan :

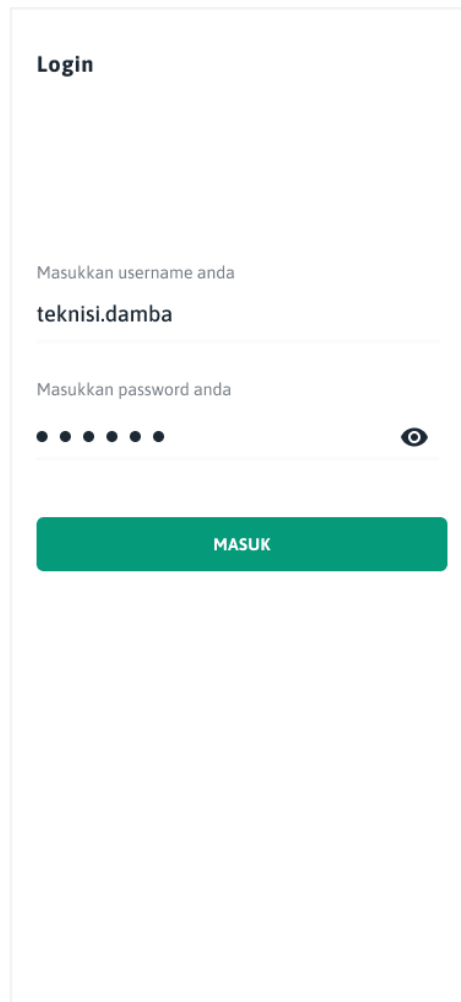


Gambar 15 Struktur Menu Aplikasi Smart Feeder

4.3.2 Perancangan Input

4.3.2.1 Halaman *Login*

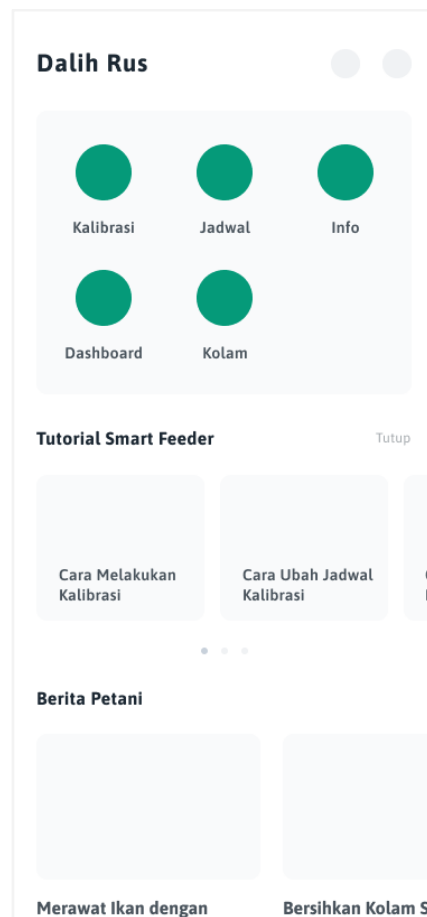
Halaman *login* ini adalah halaman pertama pengguna ketika membuka aplikasi yang sebelumnya belum melakukan *login* terlebih dahulu. Pengguna memasukan *username* dan *password* yang sudah diberikan oleh *admin* eFishery pada saat pembelian.

The image shows a mobile application login screen. At the top, the word "Login" is displayed in bold. Below it, there are two input fields. The first field is for the username, with a placeholder text "Masukkan username anda" and the value "teknisi.damba" entered. The second field is for the password, with a placeholder text "Masukkan password anda" and masked characters "• • • • •" shown. To the right of the password field is an eye icon for toggling visibility. At the bottom of the form is a green button with the text "MASUK" in white capital letters.

Gambar 16 Halaman Masuk / *Login*

4.3.2.2 Halaman *Home Screen*

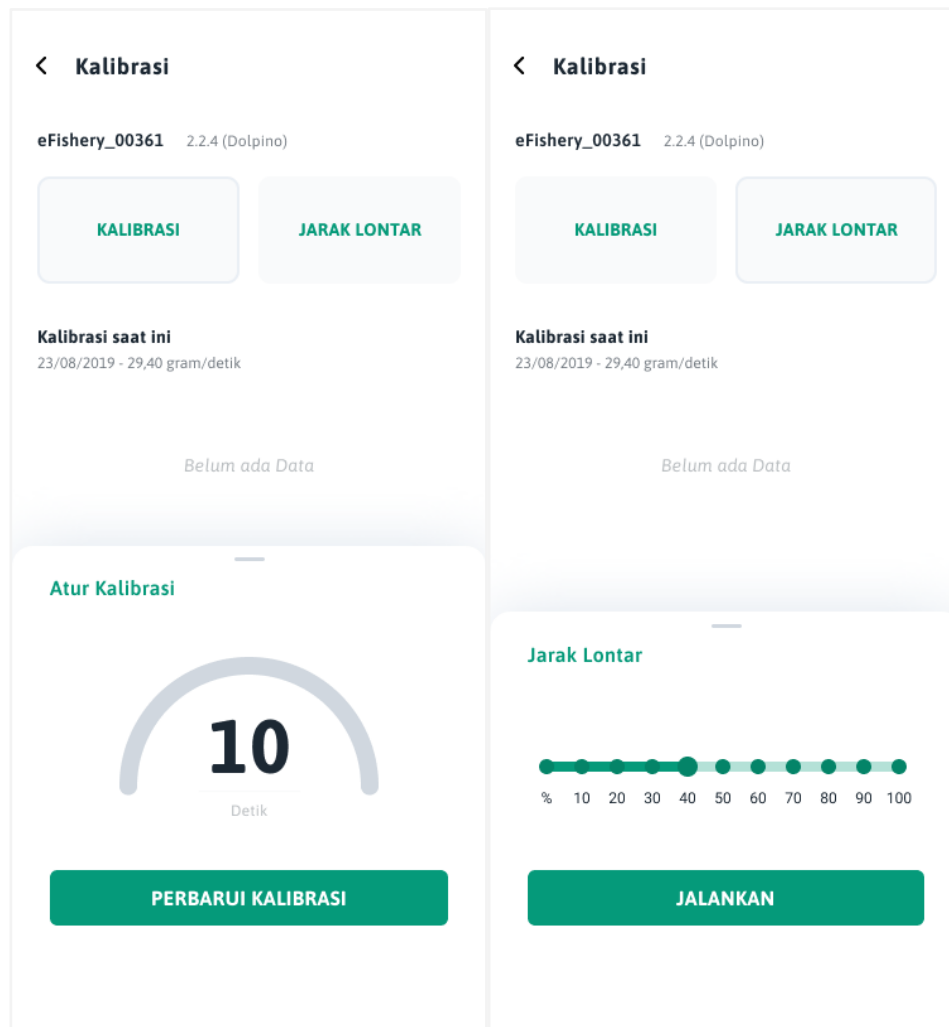
Halaman ini adalah halaman awal sekaligus halaman utama sebuah aplikasi yang terdapat menu aplikasi, menu tutorial dan menu berita.



Gambar 17 Halaman *Home Screen*

4.3.2.3 Halaman Kalibrasi

Pada halaman ini adalah halaman untuk melakukan kalibrasi, mengatur kalibrasi, merubah kalibrasi dan juga mengatur jarak lontar.



Gambar 18 Halaman Info Feeder dan Farming

4.3.2.4 Halaman Info Feeder dan Farming

Halaman ini adalah halaman menampilkan *form input* untuk edit *feeder* maupun *farming* serta juga dapat memasukan data kematian ikan, kesehatan ikan dan lain-lain.

< Info

eFishery_00361 2.2.4 (Dolpino)

Farming Information

Pond Name
361

Cultivation Commodity
Patin

Total Seed (piece)
5000

Seed Weight When Stocking (gram)
50.0

Cultivation Starting Date
3 Juli 2019

Feeder Activation Date
3 Juli 2019

CANCEL SAVE

Gambar 19 Halaman Info

4.3.2.5 Halama Jadwal *Feeder*

Berikut ini adalah halaman mengatur jadwal *feeder*, jadi pengguna bisa memasukan jadwal untuk menjalankan mesin berdasarkan pengguna mau, dan di halaman ini juga sistem memberikan data mengenai kapan ikan lapar.

Jadwal

eFishery_00361 2.2.4 (Dolpino)

Schedule EDIT

← 30 Augustus 2019

Total Feed/Day (Kg) 1.0 Total Output (Kg) 0.0

Feeding

SYNCRONIZE FEEDER

Feed data not available

Schedule

Feeding Method

Advanced

Schedules ADD

No	Time	Weight	Status
1	12:00-13:54	6802'6"	200 ✓ ✗
2	14:40-16:34	6802'6"	200 ✓ ✗
3	17:21-19:15	6802'6"	200 ✓ ✗

CANCEL **SAVE**

Jadwal

eFishery_00361 2.2.4 (Dolpino)

Schedule EDIT

← 30 Augustus 2019

Jam Mulai 14:00

Berat Pakan 20 Kilogram

Jam Akhir 14:12

Jam Akhir + Jalan

Jam Akhir + Jalan + Jeda

Jalan + Jeda

Debit/detik	Frekuensi	Jalan	Jam Akhir
29 g	5 x 3998 g	0'2'16"	0'0'7"

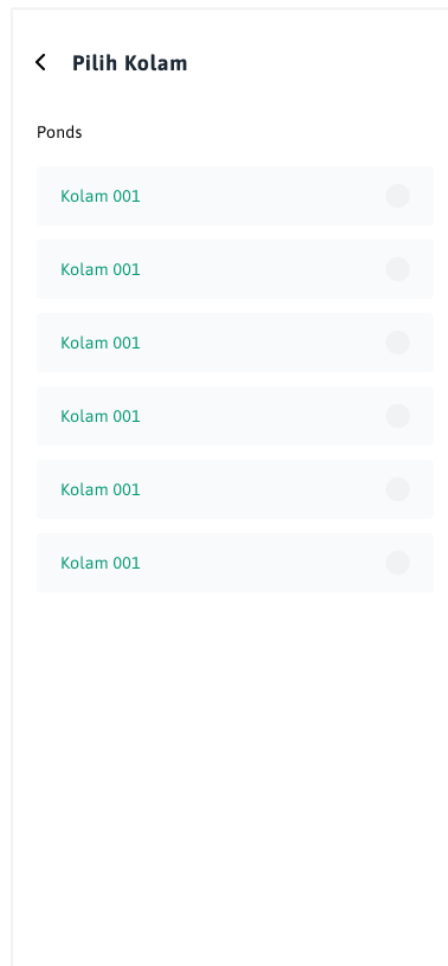
CANCEL **SAVE**

Gambar 20 Halaman Mengatur Jadwal

4.3.3 Perancangan Output

4.3.3.1 Halaman Pilih Kolam

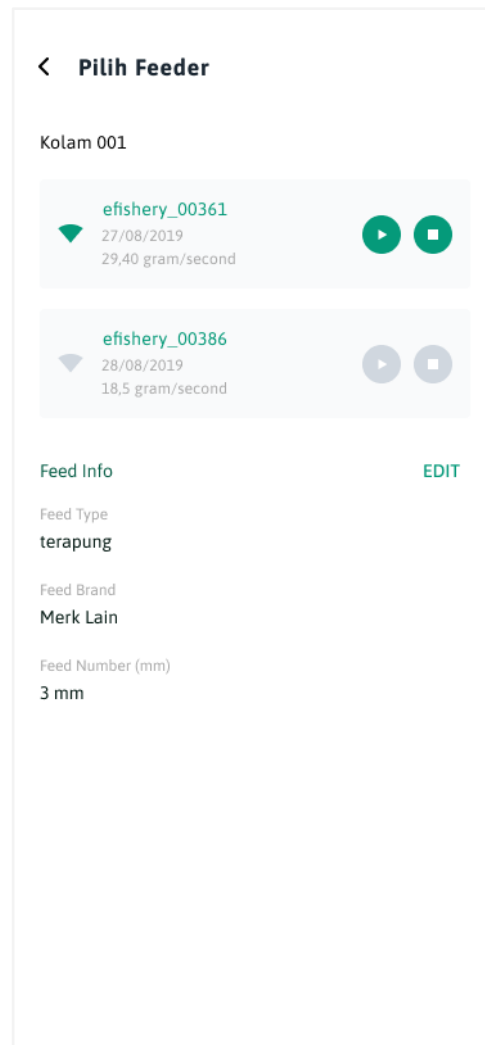
Halaman ini adalah halaman pilih kolam, mesin akan diletakan berdasarkan kolam yang ada, minimal satu kolam satu mesin, dan tiap-tiap mesin itu berbeda pengaturannya. Sehingga pengguna harus tau kolam mana saja yang menggunakan mesin ini untuk pakan ikan atau udangnya.



Gambar 21 Halaman Pilih Kolam

4.3.3.2 Halaman Pilih Feeder

Berikut ini adalah halaman pilih *feeder*, ketika pengguna selesai memilih kolam dan di dalam kolam itu terletak satu mesin, lalu pilihlah feeder tersebut untuk diatur dan dikelola.



Gambar 22 Halaman Pilih Feeder

4.3.3.3 Halaman Menu Tutorial Penggunaan

Berikut ini adalah halaman menu tutorial penggunaan aplikasi *smart feeder*, ini adalah fitur yang akan memudahkan pengguna untuk mengetahui penggunaan mesin tanpa perantara operator.



Gambar 23 Halaman Menampilkan Tutorial

4.3.3.4 Halaman Menu Berita

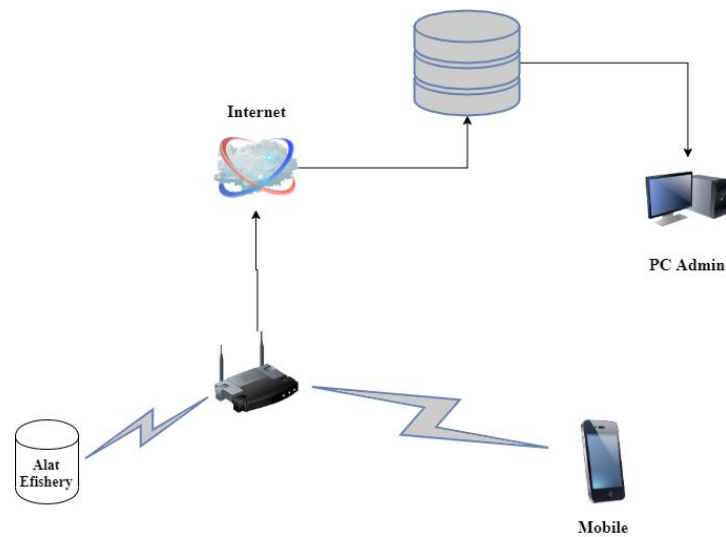
Fitur menu berita ini memberikan kemudahan informasi bagi pengguna dalam hal pengelolaan akuakultur, sehingga tingkat pengetahuan petani atau bisa disebut pengguna akhir meningkat dan pengelolaan akuakultur juga meningkat sehingga produksi jauh lebih tinggi dan memberikan daya saing tinggi serta penjualan yang sangat signifikan.



Gambar 24 Halaman Menampilkan Berita

4.4 Perancangan Arsitektur Jaringan

Berikut ini adalah perancangan arsitektur jaringan yang digunakan pada aplikasi *smart feeder*. Aplikasi ini menggunakan arsitektur jaringan internet melalui ponsel dan menggunakan kuota data internet di ponsel milik pengguna. Berikut ini adalah gambarannya :



Gambar 25 Arsitektur Jaringan pada Aplikasi

Di dalam alat *Smart Feeder* atau alat pemberian pakan ikan atau udang ini terdapat perangkat semacam wifi yang menghubungkan mesin dengan ponsel pengguna untuk kemudian pengguna melakukan pengelolaan dan data hasil tersebut disimpan kedalam *database* dengan cara sinkronisasi data.