

**EKSPANSI APLIKASI AQUA BREEDING DENGAN  
PENAMBAHAN FITUR INVENTARISASI UNTUK PENENTUAN  
HARGA DASAR PRODUK PERIKANAN BERBASIS *MOBILE***

**Proposal Skripsi**

**Disusun untuk memenuhi salah satu syarat  
memperoleh gelar Sarjana Komputer**



*Mencerdaskan dan  
Memartabatkan Bangsa*

**Oleh:  
Akbar Maulana Alfatih  
1313619003**

**PROGRAM STUDI ILMU KOMPUTER  
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS NEGERI JAKARTA**

**2023**

## **HALAMAN PERSEMBAHAN**

*Untuk Keluargaku dan Diriku Sendiri.*

## KATA PENGANTAR

Puji syukur penulis panjatkan ke hadirat Allah SWT, karena dengan rahmat dan karunia-Nya, penulis dapat menyelesaikan proposal skripsi yang berjudul **"Ekspansi Aplikasi Aqua Breeding Dengan Penambahan Fitur Inventarisasi Untuk Penentuan Harga Dasar Berbasis *Mobile*"**.

Keberhasilan dalam menyusun proposal skripsi ini tidak lepas dari bantuan berbagai pihak yang mana dengan tulus dan ikhlas memberikan masukan guna sempurnanya proposal skripsi ini. Oleh karena itu dalam kesempatan ini, dengan kerendahan hati penulis mengucapkan banyak terima kasih kepada:

1. Yth. Para petinggi di lingkungan FMIPA Universitas Negeri Jakarta.
2. Yth. Ibu Ria Arafiya, M.Si selaku Koordinator Program Studi Ilmu Komputer.
3. Yth. Bapak Muhammad Eka Suryana, M.Kom selaku Dosen Pembimbing I yang telah membimbing, mengarahkan, serta memberikan saran dan koreksi terhadap proposal skripsi ini.
4. Yth. Bapak Med Irzal, M.Kom selaku Dosen Pembimbing II yang telah membimbing, mengarahkan, serta memberikan saran dan koreksi terhadap proposal skripsi ini.

Penulis menyadari bahwa penyusunan proposal skripsi ini masih jauh dari sempurna karena keterbatasan ilmu dan pengalaman yang dimiliki. Oleh karenanya, kritik dan saran yang bersifat membangun akan penulis terima dengan senang hati. Akhir kata, penulis berharap tugas akhir ini bermanfaat bagi semua pihak khususnya penulis sendiri. Semoga Allah SWT senantiasa membalas kebaikan semua pihak yang telah membantu penulis dalam menyelesaikan proposal skripsi ini.

Jakarta, 24 Mei 2023

Akbar Maulana Alfatih

## DAFTAR ISI

<b>HALAMAN PERSEMBAHAN</b>	<b>iii</b>
<b>KATA PENGANTAR</b>	<b>iv</b>
<b>DAFTAR ISI</b>	<b>v</b>
<b>DAFTAR GAMBAR</b>	<b>vii</b>
<b>DAFTAR TABEL</b>	<b>viii</b>
<b>I PENDAHULUAN</b>	<b>1</b>
1.1 Latar Belakang Masalah . . . . .	1
1.2 Rumusan Masalah . . . . .	3
1.3 Pembatasan Masalah . . . . .	3
1.4 Tujuan Penelitian . . . . .	4
1.5 Manfaat Penelitian . . . . .	4
<b>II KAJIAN PUSTAKA</b>	<b>5</b>
2.1 Pengertian Persediaan dan Manajemen Persediaan . . . . .	5
2.2 Jenis-jenis Manajemen Persediaan . . . . .	6
2.3 Biaya Persediaan . . . . .	6
2.4 Pengendalian Persediaan . . . . .	7
2.5 Penentuan Harga Transfer . . . . .	8
2.5.1 Syarat Terpenuhi Harga Transfer . . . . .	8
2.5.2 Tujuan Penentuan Harga Transfer . . . . .	9
2.5.3 Kebijakan Penentuan Harga Transfer . . . . .	9
2.5.4 Prinsip Dasar Penentuan Harga Transfer . . . . .	10
2.6 Aqua Breeding . . . . .	12
2.7 <i>Frontend</i> dan <i>Backend</i> . . . . .	12
2.8 Flutter . . . . .	13
2.9 Flask . . . . .	20
2.10 MongoDB . . . . .	20
2.11 REST API . . . . .	20
2.12 Scrum . . . . .	21

2.13	<i>Unit Testing</i>	22
2.14	<i>User Analytics</i>	22
<b>III</b>	<b>METODOLOGI PENELITIAN</b>	<b>23</b>
3.1	Keterhubungan Penelitian	23
3.2	Depresiasi Aset Bahan Organik	24
3.3	Metode Penentuan Harga Dasar	25
3.4	Tahapan Penelitian	26
3.5	Analisa Kebutuhan	27
3.6	Perancangan Sistem	28
3.7	Pengujian	47
	<b>DAFTAR PUSTAKA</b>	<b>51</b>

## DAFTAR GAMBAR

Gambar 2.1	Logo Aqua Breeding . . . . .	12
Gambar 2.2	Halaman dengan Scaffold . . . . .	15
Gambar 2.3	Halaman tanpa Scaffold . . . . .	15
Gambar 2.4	List View . . . . .	17
Gambar 2.5	Elevated Button . . . . .	19
Gambar 2.6	Skema REST API . . . . .	20
Gambar 3.1	Diagram Alur Penelitian <i>Aquaculture</i> . . . . .	23
Gambar 3.2	Alur Tahapan Penelitian . . . . .	26
Gambar 3.3	<i>Use Case</i> Aplikasi . . . . .	28
Gambar 3.4	Tahapan Perancangan Sistem dengan Metode Scrum . . . . .	29
Gambar 3.5	Skema Database Fitur Inventaris . . . . .	31
Gambar 3.6	Integrasi Skema Database Inventaris dengan Skema Database Iterasi 1 . . . . .	33
Gambar 3.7	Halaman Dashboard . . . . .	34
Gambar 3.8	Halaman Menu Inventaris . . . . .	34
Gambar 3.9	Halaman Data Inventaris Pakan . . . . .	35
Gambar 3.10	Halaman Input Inventaris Pakan . . . . .	35
Gambar 3.11	Halaman Detail Inventaris Pakan . . . . .	35
Gambar 3.12	Halaman Data Inventaris Bahan Budidaya . . . . .	36
Gambar 3.13	Halaman Input Inventaris Bahan Budidaya . . . . .	36
Gambar 3.14	Halaman Detail Inventaris Bahan Budidaya . . . . .	36
Gambar 3.15	Halaman Data Inventaris Tagihan Listrik . . . . .	37
Gambar 3.16	Halaman Input Inventaris Tagihan Listrik . . . . .	37
Gambar 3.17	Halaman Detail Inventaris Tagihan Listrik . . . . .	37
Gambar 3.18	Halaman Data Inventaris Benih . . . . .	38
Gambar 3.19	Halaman Input Inventaris Benih . . . . .	38
Gambar 3.20	Halaman Detail Inventaris Benih . . . . .	38
Gambar 3.21	Halaman Data Inventaris Aset . . . . .	39
Gambar 3.22	Halaman Input Inventaris Aset . . . . .	39
Gambar 3.23	Alur Inventaris Pakan . . . . .	41
Gambar 3.24	Alur Inventaris Bahan Budidaya . . . . .	42
Gambar 3.25	Alur Inventaris Listrik . . . . .	43

Gambar 3.26 Alur Inventaris Benih . . . . .	44
Gambar 3.27 Alur Inventaris Aset . . . . .	45
Gambar 3.28 Update Skema Database Inventaris . . . . .	46

## DAFTAR TABEL

Tabel 3.1	Product Backlog . . . . .	30
Tabel 3.2	Sprint 1 Backlog . . . . .	31
Tabel 3.3	Sprint 2 Backlog . . . . .	40
Tabel 3.4	Skenario Unit Testing . . . . .	48



# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang Masalah**

Perikanan merupakan suatu sumber penghasilan terbesar yang ada di Indonesia dikarenakan Indonesia sendiri disebut sebagai negara maritim yang memiliki arti negara kepulauan. Oleh karena itu, banyak penduduk di Indonesia yang bermata pencaharian sebagai pembudidaya ikan. Namun, jika terlalu banyak menangkap ikan akan menyebabkan *over fishing* yang membuat kemampuan bereproduksi ikan akan jauh lebih kecil daripada jumlah ikan hasil tangkapan. Hal ini akan menyebabkan langkanya spesies ikan tersebut dan berkurangnya angka produksi ikan. Dengan demikian, untuk mengatasi hal tersebut diperlukan budidaya perikanan yang berguna untuk menjaga ikan sampai masa panen tiba, serta dapat meningkatkan nilai ekonomi para pembudidaya ikan.

Dalam menjalankan budidaya perikanan, kebanyakan pembudidaya ikan masih melakukan cara manual dalam mengelola budidayanya. Hal ini tentunya kurang efektif dalam jangka panjang dan akan menyulitkan dalam pengelolaan budidayanya. Oleh karena itu, dalam penelitian yang dibuat oleh (Lin, 2019) dan (Ouyang, 2021) dapat berguna dalam menerapkan budidaya perikanan modern.

Yi-Bing Lin dan timnya membuat *smart aquarium* yang bertujuan untuk meningkatkan kualitas akuarium yang bernama FishTalk. FishTalk memungkinkan sebuah sensor pada akuarium untuk menggerakkan aktuator secara real time. Kegunaan dari *smart aquarium* ini seperti sistem pemberian pakan otomatis dan pengendalian air dalam kolam secara otomatis. (Lin, 2019)

Sementara itu, Bing Ouyang dan timnya membuat sebuah sistem yang dibentuk dan digunakan untuk monitoring serta *decision making* pada tambak perikanan, sistem ini dinamakan HAUCS (*Hybrid Aerial Underwater Robotic System*). Pemantauan ini dilakukan dengan memanfaatkan sistem robotik, mesin, dan operator manusia. Tujuan dibentuknya HAUCS ini adalah untuk meringankan pekerjaan manusia dari tugas yang berat, terlalu banyak biaya, dan memakan waktu dalam operasi pelaksanaan budidaya *aquaculture* melalui platform pemanfaatan sistem robotik. (Ouyang, 2021)

Dari kedua penelitian diatas, dapat disimpulkan bahwa alat yang digunakan

dapat bermanfaat bagi para pembudidaya ikan karena dapat mempermudah pengelolaan budidaya. Namun, tentunya alat dan bahan yang dibutuhkan cukup banyak dan pasti mematok harga yang tidak sedikit.

Oleh karena itu, penelitian yang dilakukan oleh (Chen, 2020) dan timnya mungkin dapat mengatasi masalah tersebut. Penelitian ini bertujuan untuk membuat big data dengan *framework* SpringBoot dan Java Persistence API (JPA) yang didalamnya terdapat data kualitas air pada setiap perkembangbiakan ikan ternak. Platform ini dapat digunakan untuk memprediksi kualitas air dari setiap kolam dan memberikan notifikasi langsung ketika ada masalah pada kolam tersebut. Namun, penelitian ini hanya berfokus pada pendataan kualitas air saja sehingga rincian lain dari budidaya tersebut masih belum lengkap. (Chen, 2020)

Tapi, tidak seperti dua penelitian yang sudah dirujuk sebelumnya, penelitian (Chen, 2020) ini berbasis aplikasi sehingga tidak ada biaya peralatan tambahan. Dengan demikian, pembudidaya ikan akan lebih terbantu jika terdapat aplikasi yang dapat membantu mereka dalam mengembangkan budidayanya tanpa perlu mengeluarkan biaya tambahan.

Pada penelitian yang dilakukan oleh (Hadi, 2021), (Maghriza, 2022) dan (Rahmanto, 2022), mereka membuat suatu aplikasi bernama *Aqua Breeding* yang berfungsi untuk mencatat pendetailan dari setiap budidaya para pembudidaya ikan. Detail yang dimaksud seperti pencatatan pakan ikan, pencatatan angka kematian ikan, pengendalian kualitas air, dan pencatatan lainnya yang berhubungan pada musim budidaya ikan tersebut. Aplikasi ini tentunya dapat membantu para pembudidaya ikan dan juga dapat meningkatkan ekonomi pembudidaya ikan sejalan dengan lancarnya musim budidaya.

Penelitian yang terkait dalam aplikasi tersebut adalah penelitian Fadhil Perwira Hadi yang berjudul “Rancang Bangun Web Service dan Website sebagai Storage Engine dan Monitoring Data Sensing untuk Budidaya Ikan Air Tawar” menghasilkan suatu sistem web service yang dapat menerima data yang dikirimkan oleh *embedded device*, dengan menerapkan konsep IoT (Hadi, 2021). *Web service* tersebut kemudian dilanjutkan dengan penelitian Andri Rahmanto dengan judul “Perancangan Arsitektur Aplikasi Budidaya Perikanan Modern pada Backend yang bertanggung jawab dalam melayani Transaksi Query Webservice dengan menggunakan Teknologi Flask Microservice”. *Web service* ini menghasilkan *output* berupa arsitektur aplikasi budidaya perikanan modern pada *backend* berupa *endpoint* yang dapat digunakan untuk pendataan budidaya perikanan air tawar (Rahmanto,

2022). Dalam pengolahan *backend* ini, Gian Chiesa Maghriza dengan penelitiannya yang berjudul “Perancangan Frontend Aplikasi Pendukung Teknologi Perikanan Modern dengan menggunakan Framework Flutter yang mentarget Multi Platform” membuat *user interface* serta konfigurasi fitur pencatatan dari aplikasi teknologi perikanan modern. Fitur-fitur yang ada pada aplikasi ini didasari pada penggunaan *endpoint* yang sudah disediakan pada *backend* buatan Andri (Maghriza, 2022). Namun pada aplikasi tersebut masih terdapat kekurangan seperti belum tersedia fitur inventarisasi sebagai *storage* dalam berbudidaya.

Hal tersebut tentunya masih belum memecahkan masalah dari pembudidaya ikan dalam menjalankan budidayanya. Masalah yang paling berdampak pada pembudidaya ikan adalah saat harga komoditas mengalami kenaikan sedangkan harga jual ikan tidak mengalami perubahan dikarenakan harga yang sudah ditetapkan oleh Kemendagri sehingga pembudidaya ikan bisa mengalami kerugian. Hal ini tentunya akan membawa dampak negatif dalam nilai ekonomi perikanan.

Oleh karena itu, penelitian ini bertujuan untuk memecahkan masalah tersebut dengan menambahkan sistem inventarisasi pada aplikasi *Aqua Breeding* ini untuk menentukan harga dasar pada produk perikanan. Dengan demikian, harga dasar tersebut dapat digunakan oleh pembudidaya ikan untuk penjualan hasil panen mereka. Selain itu, fitur inventarisasi ini juga dapat membantu para pembudidaya ikan dalam mengolah dan mengontrol kebutuhan serta pengeluaran dalam setiap musim budidaya. Berdasarkan fitur baru yang sudah dijelaskan sebelumnya, aplikasi ini diharapkan dapat membantu para pembudidaya ikan untuk berbudidaya dalam hal penentuan harga dasar dan pengendalian kebutuhan saat budidaya berlangsung.

## 1.2 Rumusan Masalah

Dari uraian latar belakang di atas, perumusan masalah pada penelitian ini ialah “Bagaimana rancangan sistem inventaris yang digunakan untuk menentukan harga dasar dan harga jual minimum dari produk perikanan berbasis *mobile*?”.

## 1.3 Pembatasan Masalah

Pembatasan masalah pada penelitian ini antara lain:

1. Aplikasi hanya menentukan harga dasar dan harga jual minimum dari produk perikanan.

#### **1.4 Tujuan Penelitian**

Penelitian ini dilakukan dengan tujuan untuk membuat aplikasi perikanan berbasis *mobile* dengan sistem inventaris untuk menentukan harga dasar dan harga jual minimum dari produk perikanan.

#### **1.5 Manfaat Penelitian**

1. Bagi penulis

Meningkatkan pengetahuan sistem inventaris pada produk perikanan, menambah pengalaman dalam mengembangkan aplikasi, memperoleh gelar sarjana di bidang Ilmu Komputer, serta menjadi media untuk penulis dalam mengaplikasikan ilmu yang didapat dari kampus.

2. Bagi Universitas Negeri Jakarta

Menjadi pedoman untuk penelitian di masa depan, dan dapat memberikan panduan bagi mahasiswa program studi Ilmu Komputer tentang penentuan harga dasar pada produk perikanan dengan sistem inventaris.

3. Bagi masyarakat

Membantu masyarakat yang ingin dan sedang menggeluti bidang budidaya perikanan dalam proses penentuan harga dasar pada produk perikanan.

## **BAB II**

### **KAJIAN PUSTAKA**

#### **2.1 Pengertian Persediaan dan Manajemen Persediaan**

Pada buku Dasar-Dasar Manajemen (Sim, 2022), dijelaskan bahwa persediaan adalah sebuah stok aset yang dimiliki oleh perusahaan. Aset ini dapat berupa bahan mentah, bahan baku, barang jadi, barang dalam proses, hingga bahan pembantu. Persediaan merupakan aset yang berharga, karena berkaitan dengan proses produksi. Persediaan yang tidak teratur dapat menyebabkan kerugian, sehingga menerapkan manajemen persediaan dalam suatu bisnis merupakan hal yang penting.

Manajemen persediaan merupakan suatu cara untuk melakukan pengawasan, kontrol, pengelolaan terhadap persediaan yang dimiliki oleh perusahaan. Berbagai macam kegiatan yang berkaitan dengan memperoleh, menyimpan, hingga menggunakan persediaan merupakan bagian dari manajemen persediaan.

Manajemen persediaan memiliki beberapa fungsi, yaitu:

1. Mencegah terjadinya kekurangan persediaan.
2. Mencegah barang dari supplier tidak sesuai kebutuhan.
3. Memastikan proses produksi berjalan dengan lancar.
4. Mengantisipasi permintaan yang mendadak.
5. Menyesuaikan pembelian dengan jadwal produksi.

Selain beberapa fungsi yang sudah disebutkan diatas, Manajemen persediaan juga memiliki tujuan. Beberapa tujuan dari manajemen persediaan adalah sebagai berikut.

1. Mengantisipasi kenaikan harga bahan baku.
2. Memastikan persediaan selalu tersedia.
3. Mengurangi resiko bahan baku yang datang terlambat.
4. Menjaga jumlah persediaan tetap stabil.

5. Mengantisipasi kemungkinan adanya perubahan dari segi penawaran ataupun permintaan.

## **2.2 Jenis-jenis Manajemen Persediaan**

Manajemen persediaan dibagi menjadi beberapa jenis, diantaranya:

### **1. Bahan Mentah**

Bahan mentah atau bahan baku merupakan bahan utama atau dasar dari dibuatnya suatu produk. Tanpa adanya bahan baku, maka produk tidak bisa masuk ke tahap produksi. Oleh karena itu manajemen persediaan diperlakukan untuk mengelola bahan baku agar selalu tersedia dan siap untuk diproses.

### **2. Barang Setengah Jadi**

Barang setengah jadi atau barang dalam proses merupakan barang yang belum sepenuhnya bisa digunakan, sehingga perlu untuk diproses lebih lanjut untuk menjadi barang jadi, yang nantinya siap untuk digunakan. Dalam hal ini, manajemen persediaan digunakan untuk menghitung banyaknya barang setengah jadi tersebut untuk memenuhi kebutuhan pasar.

### **3. Barang Jadi**

Barang jadi merupakan barang yang sudah siap untuk dijual. Manajemen persediaan berguna untuk mengatur pengiriman barang tersebut ke pasar sehingga keadaan produk di pasar tetap stabil.

## **2.3 Biaya Persediaan**

Penetapan biaya persediaan atau evaluasi persediaan memungkinkan perusahaan untuk memberikan nilai moneter untuk barang-barang dalam persediaan mereka. Inventaris perusahaan seringkali merupakan aset terbesarnya dan pengukuran yang tepat untuk memastikan keakuratan laporan keuangan.

Untuk menentukan biaya persediaan, diperlukan lima langkah-langkah sebagai berikut.

1. Menentukan periode waktu tertentu yang dimana perlu menemukan nilai inventaris.

2. Memastikan persediaan selalu tersedia.
3. Mengurangi resiko bahan baku yang datang terlambat.
4. Menjaga jumlah persediaan tetap stabil.
5. Mengantisipasi kemungkinan adanya perubahan dari segi penawaran ataupun permintaan.

## 2.4 Pengendalian Persediaan

Pengendalian persediaan adalah suatu tahapan yang digunakan untuk menyelesaikan masalah terkait dengan pengendalian barang pada perusahaan.

Persediaan yang terlalu berlebihan akan merugikan, karena akan lebih banyak modal yang diperlukan.

Menurut Sunyoto (2012:225), Sistem pengendalian persediaan merupakan serangkaian pengendalian untuk menentukan tingkat persediaan yang harus dijaga. Sistem ini menentukan dan menjamin tersedianya persediaan yang tepat dalam kualitas dan waktu yang tepat. Jika persediaan terlalu sedikit dapat mengakibatkan resiko terjadinya kekurangan persediaan atau bisa dibilang *stockout*. Bila persediaan dilebihkan, maka biaya penyimpanan dan modal yang diperlukan akan bertambah. Sebaliknya, jika persediaan dikurangi maka akan mengalami *stockout* (kehabisan barang).

Menurut Assauri (2004), Pengendalian persediaan dapat dikatakan sebagai suatu kegiatan untuk menentukan tingkat dan komposisi dari persediaan sehingga perusahaan dapat melindungi kelancaran produksi dan penjualan serta kebutuhan-kebutuhan perusahaan dengan efisien.

Dengan kata lain, pengendalian persediaan akan mempermudah perusahaan untuk memproduksi barang, disimpan di gudang dan sampai ke konsumen. Persediaan yang terlalu besar (*overstock*) merupakan pemborosan karena menyebabkan tingginya beban biaya untuk inventaris barang-barang tersebut, sementara jika persediaan terlalu kecil maka dapat menyebabkan proses produksi terhenti sehingga konsumen akan pergi karena permintaannya tidak terpenuhi.

Agar pengendalian persediaan dapat dilakukan dengan maksimal, menurut Assauri (2004:176) ada faktor-faktor yang harus dipertimbangkan dalam menjalankan pengendalian persediaan, antara lain:

1. Adanya fasilitas pergudangan yang cukup luas dan teratur
2. Adanya sistem administrasi pencatatan dan pemeriksaan atas penerimaan dan pengeluaran barang
3. Sumber daya yang menguasai sistem administrasi pengendalian persediaan yang digunakan perusahaan
4. Perencanaan untuk mengganti barang yang telah digunakan dan barang yang sudah lama berada dalam gudang sehingga usang
5. Informasi dari bagian produksi tentang sifat teknis barang, daya tahan produk dan lamanya produksi, untuk melakukan perencanaan pengendalian persediaan
6. Informasi dari bagian penjualan tentang tingkat penjualan produk perusahaan, sehingga bagian persediaan bisa menentukan besarnya persediaan yang seharusnya ada sehingga tidak terjadi kekurangan persediaan yang mengakibatkan pesanan konsumen tidak terpenuhi.

## **2.5 Penentuan Harga Transfer**

Pada buku *Management Control* (Supitriyani, 2022), dijelaskan bahwa penentuan harga transfer atau *transfer pricing* merupakan proses harga penentuan harga yang ditetapkan dalam transaksi penjualan dan pembelian diantara berbagai unit organisasi pada kelompok perusahaan atau instansi yang sama.

Dalam menentukan harga transfer, prinsip dasarnya adalah bahwa harga transfer sebaiknya serupa dengan harga yang akan dikenakan seandainya produk tersebut dijual ke konsumen luar atau dibeli dari pemasok luar. Ketika suatu perusahaan membeli atau menjual produk

### **2.5.1 Syarat Terpenuhiya Harga Transfer**

Menurut (Silalahi et al., 2019), syarat-syarat yang harus dijalankan agar terpenuhiya harga transfer adalah:

1. Sistem harus dapat memberikan informasi yang relevan yang dibutuhkan oleh suatu pusat laba untuk dapat menemukan trade-off yang optimum antara biaya dan pendapatan perusahaan.



2. Laba yang dihasilkan harus dapat menggambarkan dengan baik pengaturan trade-off antara biaya-pendapatan yang telah ditetapkan. Setiap pusat laba harus dapat memaksimalkan laba perusahaan dengan jalan memaksimalkan laba divisinya.
3. Tingkat laba yang diperlihatkan oleh masing-masing pusat laba harus dapat mencerminkan besarnya kontribusi laba dari masing-masing pusat laba terhadap laba perusahaan secara keseluruhan.

### **2.5.2 Tujuan Penentuan Harga Transfer**

Menurut (Silalahi et al., 2019), harga transfer harus dirancang sedemikian rupa sehingga dapat mencapai tujuan berikut :

1. Memberikan informasi yang relevan kepada masing-masing unit usaha untuk menentukan imbal balik yang optimum antara biaya dan pendapatan perusahaan.
2. Menghasilkan keputusan yang selaras dengan cita-cita, maksudnya sistem harus dirancang sedemikian rupa sehingga keputusan yang meningkatkan laba unit usaha juga akan meningkatkan laba perusahaan.
3. Membantu pengukuran kinerja ekonomi dari unit usaha individual.
4. Sistem tersebut harus mudah dimengerti dan dikelola.

### **2.5.3 Kebijakan Penentuan Harga Transfer**

Menurut (Hansen and Mowen, 2009), dalam penyusunan sebuah kebijakan penetapan harga transfer, kedua pandangan dari divisi penjual dan divisi pembeli harus dipertimbangkan. Pendekatan biaya peluang (opportunity cost approach) mencapai tujuan tersebut dengan mengidentifikasi harga minimum yang ingin diterima divisi penjual dan harga maksimum yang ingin dibayar divisi pembeli. Berikut harga-harga yang ditetapkan di setiap divisi:

1. Harga transfer minimum

Harga transfer minimum adalah harga transfer yang akan membuat keadaan divisi penjual tidak menjadi lebih buruk jika barang yang dijual pada divisi internal daripada dijual pada pihak luar.

## 2. Harga transfer maksimum

Harga transfer maksimum adalah harga transfer yang akan membuat keadaan divisi pembeli tidak menjadi lebih buruk jika suatu input dibeli dari divisi internal daripada jika barang yang sama dibeli secara eksternal.

### 2.5.4 Prinsip Dasar Penentuan Harga Transfer

Menurut (Anthony and Govindarajan, 2018), masalah penentuan harga transfer sebenarnya merupakan penentuan harga pada umumnya, dengan sedikit modifikasi untuk mempertimbangkan faktor-faktor tertentu yang unik dalam transaksi internal. Prinsip dasarnya adalah bahwa harga transfer sebaiknya serupa dengan harga yang akan dikenakan seandainya produk tersebut dijual ke konsumen luar atau dibeli dari pemasok luar.

Ketika suatu pusat laba di suatu perusahaan membeli produk dan menjual ke satu sama lain, maka dua keputusan yang harus diambil untuk setiap produk adalah:

1. Apakah perusahaan harus memproduksi sendiri produk tersebut atau membelinya dari pemasok luar. Hal ini merupakan keputusan sourcing.
2. Jika diproduksi secara internal, pada tingkat harga berapakah produk tersebut akan ditransfer antarpusat laba. Hal ini merupakan keputusan harga transfer.

Sistem harga transfer dapat bervariasi dari yang paling sederhana sampai yang paling rumit, tergantung dari sifat usahanya. Berikut merupakan beberapa jenis situasi dalam menentukan sistem harga transfer.

#### 1. Situasi Ideal

Menurut (Anthony and Govindarajan, 2018), harga transfer berdasarkan harga pasar akan menghasilkan keselarasan jika kondisi-kondisi berikut ada, yaitu :

- (a) Orang-orang Kompeten.
- (b) Atmosfer yang baik.
- (c) Harga Pasar.
- (d) Kebebasan Memperoleh Sumber Daya.
- (e) Informasi Penuh.
- (f) Negoisasi

## 2. Hambatan dalam Perolehan Sumber Daya

Seorang manajer pembelian bebas mengambil keputusan sourcing. Demikian halnya dengan manajer penjualan, ia harus bebas untuk menjual produknya ke pasar yang paling menguntungkan. Menurut (Anthony and Govindarajan, 2018), akibat-akibat yang terjadi jika para manajer pusat laba tidak memiliki kebebasan dalam mengambil keputusan sourcing adalah sebagai berikut.

### (a) Pasar yang terbatas

Beberapa alasan pasar terbatas bagi pusat laba (pembeli dan penjual):

- i. Keberadaan kapasitas internal mungkin membatasi pengembangan penjualan eksternal.
- ii. Jika suatu perusahaan merupakan produsen tunggal dari produk yang terdifferensiasi, tidak ada sumber dari luar.
- iii. Jika perusahaan telah melakukan investasi yang besar, cenderung tidak akan menggunakan sumber daya dari luar kecuali harga jual di luar mendekati biaya variabel perusahaan.

Dalam kondisi pasar yang terbatas, harga transfer yang paling memenuhi persyaratan sistem pusat laba adalah harga kompetitif.

Perusahaan dapat mengetahui tingkat harga kompetitif jika perusahaan tersebut tidak membeli atau menjual produknya ke pasar bebas melalui cara-cara dibawah ini.

- i. Jika ada harga pasar diterbitkan, maka harga tersebut dapat digunakan untuk menentukan harga transfer.
- ii. Harga pasar mungkin ditentukan berdasarkan penawaran.
- iii. Jika pusat laba pembelian membeli produk yang serupa dari pasar luar/bebas maka pusat laba tersebut dapat meniru untuk harga kompetitif untuk produk-produk eksklusifnya.

### (b) Kelebihan atau Kekurangan Kapasitas Industri

Jika pusat laba penjualan tidak dapat menjual seluruh produk ke pasar bebas atau memiliki kapasitas produksi yang berlebih. Perusahaan mungkin tidak akan mengoptimalkan labanya jika pusat laba pembelian membeli produk dari pemasok luar sementara sementara kapasitas produksi di dalam masih memadai. Dan sebaliknya, jika pusat laba

pembelian tidak dapat memperoleh produk yang diperlukan dari luar sementara pusat laba penjualan menjual produknya ke pihak luar. Situasi ini terjadi ketika terdapat kekurangan kapasitas produksi di dalam industri, sehingga pusat laba pembelian terhalang dan laba perusahaan tidak optimal.

Meskipun ada hambatan dalam perolehan sumber daya, harga pasar tetap merupakan harga transfer yang baik. Meskipun demikian, jika tidak ada cara untuk memperkirakan harga kompetitif, pilihan lainnya adalah mengembangkan harga transfer berdasarkan biaya (*cost based transfer price*). Biasanya, perusahaan akan mengeliminasi unsur iklan, pendanaan, atau pengeluaran lain yang tidak dikeluarkan oleh pihak penjual dalam transaksi internal saat penentuan harga transfer.

## 2.6 Aqua Breeding



**Gambar 2.1:** Logo Aqua Breeding

Aqua Breeding merupakan aplikasi yang digunakan pembudidaya ikan untuk mengontrol budidaya ikan mereka. Aplikasi ini memiliki fitur seperti mencatat pemberian pakan, perhitungan kualitas air, mencatat total panen, serta aktivitas budidaya ikan lainnya. Dari logo pada **Gambar 2.1**, dapat diartikan aplikasi Aqua Breeding memiliki arti menyatukan para pembudidaya ikan menjadi satu kesatuan dalam aplikasi ini.

## 2.7 Frontend dan Backend

Dalam pengembangan aplikasi, terdapat 2 sisi pengembangan yaitu *frontend* dan *backend*. *Frontend* merupakan bagian yang ditampilkan kepada user seperti

contoh halaman dashboard, menu aplikasi, dan sebagainya. Oleh sebab itu, sisi *frontend* ini juga bisa dibilang sebagai *user-side* atau *client-side*.

Sementara itu, *backend* merupakan bagian yang mengurus koneksi antara server dengan *database* aplikasi. *Backend* juga bertugas untuk membuat penghubung antara aplikasi dengan server dalam bentuk *endpoint* yang nantinya akan digunakan pada sisi *frontend* untuk ditampilkan pada aplikasi. Oleh karena itu, sisi *backend* bisa dibilang sebagai *server-side*.

## 2.8 Flutter

Flutter merupakan salah satu *framework* yang digunakan dalam pengembangan aplikasi *mobile* (Flutter, 2017). Flutter dikembangkan dengan bahasa pemrograman Dart yang dirancang oleh Lars Bak dan Kasper Lund dan memiliki struktur berbasis *class* dan berorientasi terhadap objek. *Framework* ini diresmikan pada tahun 2015 oleh perkumpulan Dart *developer summit* dan merilis versi stabilnya yaitu Flutter 1.0 pada tahun 2018 di acara Flutter Live. Pengembangan *framework* ini tergolong cukup besar karena bersifat *open source* sehingga banyak komunitas dan orang-orang yang ikut mengembangkan *framework* ini.

Beberapa keunggulan penggunaan Flutter sebagai *framework* pengembangan aplikasi *mobile* adalah sebagai berikut.

### 1. Multiplatform

*Multiplatform* berarti bahwa *framework* ini dapat digunakan untuk mengembangkan dua sisi *mobile* yaitu Android dan iOS dengan satu basis kode. Hal ini tentunya dapat mempersingkat waktu serta mengurangi biaya pada bagian *development*.

### 2. Open Source

*Open Source* berarti bahwa *framework* ini dapat dimodifikasi oleh pengguna sehingga user juga dapat berkontribusi dalam pengembangan *framework* ini.

### 3. Dokumentasi lengkap

Dokumentasi lengkap *framework* Flutter dapat diakses pada situs resmi Flutter di <https://flutter.dev/>.

Dalam Flutter, bagian-bagian yang membentuk sebuah halaman disebut sebagai widget. Beberapa widget Flutter yang umum digunakan dalam pembuatan aplikasi adalah sebagai berikut.

### 1. Scaffold

Scaffold merupakan bagian utama dari sebuah halaman pada Flutter. Bisa dibilang Scaffold ini adalah *parent* dari semua widget. Berikut contoh penerapan Scaffold pada Flutter.

---

```
import 'package:flutter/material.dart';

/// Flutter code sample for [Scaffold].

void main() => runApp(const
  ScaffoldExampleApp());

class ScaffoldExampleApp extends
  StatelessWidget {
  const ScaffoldExampleApp({super.key});

  @override
  Widget build(BuildContext context) {
    return const MaterialApp(
      home: ScaffoldExample(),
    );
  }
}

class ScaffoldExample extends StatefulWidget {
  const ScaffoldExample({super.key});

  @override
  State<ScaffoldExample> createState() =>
    _ScaffoldExampleState();
}
```

```

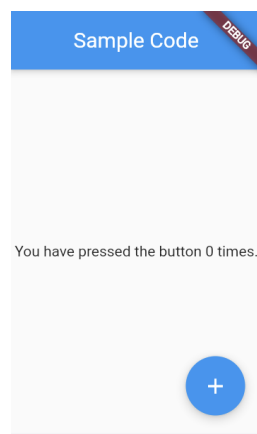
class _ScaffoldExampleState extends
  State<ScaffoldExample> {
  int _count = 0;

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('Sample_Code'),
      ),
      body: Center(child: Text('You_have_pressed_
        the_button_$_count_times.')),
      floatingActionButton: FloatingActionButton(
        onPressed: () => setState(() => _count++),
        tooltip: 'Increment_Counter',
        child: const Icon(Icons.add),
      ),
    );
  }
}

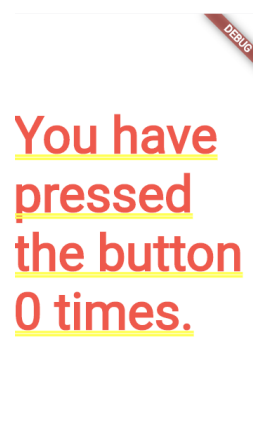
```

---

Berikut merupakan hasil layout dari code sebelumnya.



**Gambar 2.2:**  
Halaman dengan Scaffold



**Gambar 2.3:**  
Halaman Scaffold

Dapat dilihat pada perbandingan **Gambar 2.2** dan **Gambar 2.3**, tanpa menggunakan Scaffold maka halaman tidak akan terproses dengan baik.

## 2. List View

List View merupakan salah satu widget yang digunakan untuk menampilkan list dari data. Berikut contoh penggunaan List View.

---

```

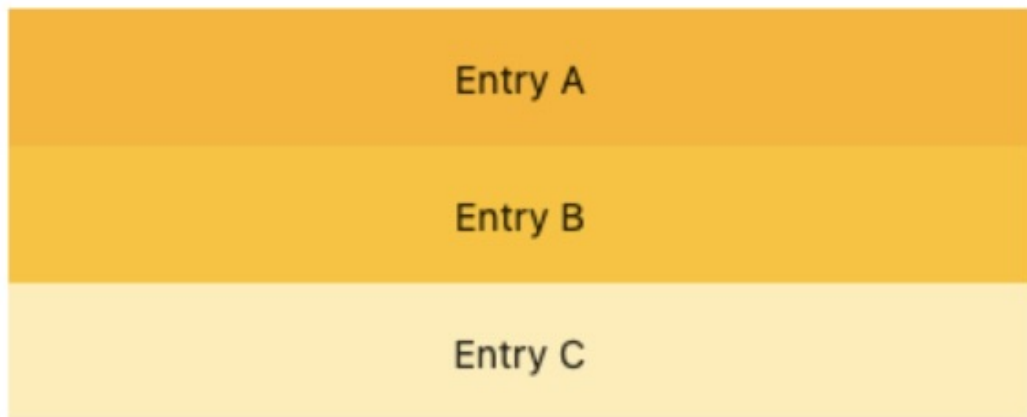
ListView(
  padding: const EdgeInsets.all(8),
  children: <Widget>[
    Container(
      height: 50,
      color: Colors.amber[600],
      child: const Center(child: Text('Entry_A')),
    ),
    Container(
      height: 50,
      color: Colors.amber[500],
      child: const Center(child: Text('Entry_B')),
    ),
    Container(
      height: 50,
      color: Colors.amber[100],
      child: const Center(child: Text('Entry_C')),
    ),
  ],
)

```

---

Berikut merupakan hasil layout dari code sebelumnya.





**Gambar 2.4:** List View

### 3. Elevated Button

Elevated Button merupakan salah satu widget button yang digunakan pada Flutter. Berikut contoh penggunaan Elevated Button.

---

```
import 'package:flutter/material.dart';

/// Flutter code sample for [ElevatedButton].

void main() => runApp(const
  ElevatedButtonExampleApp());

class ElevatedButtonExampleApp extends
  StatelessWidget {
  const ElevatedButtonExampleApp({super.key});

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home: Scaffold(
        appBar: AppBar(title: const
          Text('ElevatedButton_Sample')),
        body: const ElevatedButtonExample(),
      ),
    ),
  ),
}
```

```

    );
  }
}

class ElevatedButtonExample extends
  StatefulWidget {
  const ElevatedButtonExample({super.key});

  @override
  State<ElevatedButtonExample> createState() =>
    _ElevatedButtonExampleState();
}

class _ElevatedButtonExampleState extends
  State<ElevatedButtonExample> {
  @override
  Widget build(BuildContext context) {
    /// Set up ElevatedButton Style

    final ButtonStyle style =
      ElevatedButton.styleFrom(textStyle: const
        TextStyle(fontSize: 20));

    /// Render ElevatedButton

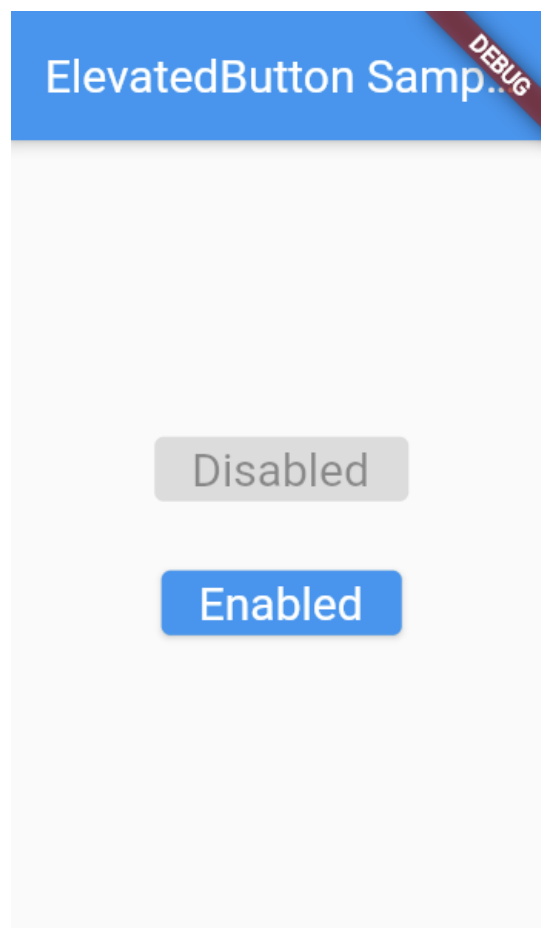
    return Center(
      child: Column(
        mainAxisAlignment: MainAxisAlignment.min,
        children: <Widget>[
          ElevatedButton(
            style: style,
            onPressed: null,
            child: const Text('Disabled'),
          ),
          const SizedBox(height: 30),

```

```
ElevatedButton(  
  style: style,  
  onPressed: () {},  
  child: const Text('Enabled'),  
),  
],  
),  
);  
}  
}
```

---

Berikut merupakan hasil layout dari code sebelumnya.



**Gambar 2.5:** Elevated Button

## 2.9 Flask

Flask merupakan *microframewok* yang digunakan pada sisi *backend* dengan basis bahasa pemrogramannya yaitu Phyton (Flask, 2010). Flask dirilis pada tahun 2010 dan dikembangkan oleh Armin Ronacher, seorang python *entusiast*. Flask disebut *microframewok* karena Flask tidak memerlukan alat bantu lain atau *library* dalam penggunaannya.

Salah satu keuntungan menggunakan Flask adalah basis bahasa pemrogramannya menggunakan Phyton. Dengan ini, Flask dapat diintegrasikan dengan beberapa library Python seperti Machine Learning, AI, dan sebagainya.

## 2.10 MongoDB

MongoDB merupakan *database* NoSQL yang dikembangkan oleh MongoDB Inc yang rilis pada tahun 2009 (MongoDB, 2009). *Database* ini disebut NoSQL karena berbasis objek atau bisa disebut JSON (JavaScript Object Notation), berbeda dengan MySQL yang berbasis tabel dalam penggunaannya.

## 2.11 REST API

Dalam pengembangan aplikasi, untuk menghubungkan antara *frontend* dengan *backend* dapat digunakan API sebagai perantaranya. API merupakan singkatan dari *Application Programming Interface* yang berfungsi menerima request dan response dari sisi *frontend* dan *backend*.

REST merupakan singkatan dari *Representational State Transfer*. API dapat disebut sebagai REST API jika memiliki standar kriteria dari REST. Kriteria tersebut bertujuan untuk menjadi standar dalam proses komunikasi antar aplikasi dan pengguna sehingga menjadi lebih fleksibel.



**Gambar 2.6:** Skema REST API

Pada **Gambar 2.6**, dapat dilihat skema penggunaan REST API yang

melibatkan *client* dan *database*. REST API bekerja dengan cara menerima *request* yang melibatkan *database* dan memberikan *response* kepada *client* dengan perantara komunikasi seperti HTTP.

*Request client* berupa CRUD (*Create, Read, Update, Delete*) pada HTTP berupa GET, POST, PUT, DELETE yang digunakan untuk berkomunikasi terhadap server serta *response* untuk client yang diterima berbentuk JSON. Metode GET berfungsi untuk mengambil data, POST dan PUT berfungsi untuk membuat dan memperbarui data, serta DELETE berfungsi untuk menghapus data pada *database*.

## 2.12 Scrum

Scrum merupakan metode pengembangan aplikasi yang digunakan untuk kolaborasi antar tim pengembangan (Scrum, 2010). Pada jurnal yang berjudul The Scrum Guide (Ken Schwaber, 2020), dijelaskan bahwa Scrum merupakan kerangka kerja yang digunakan untuk membantu orang, tim, serta organisasi dalam menyelesaikan suatu masalah. Scrum menerapkan prinsip Agile yang dimana berfokus pada kepuasan konsumen dalam masa pengembangan.

Untuk menjalankan metode Scrum, diperlukan Scrum Master, Product Owner, serta Developer. Scrum Master bertugas sebagai pemimpin serta bertanggung jawab dalam menjalankan prinsip Scrum pada tim, Product Owner bertanggung jawab dalam membuat dan mengontrol pekerjaan tim agar sesuai dengan kebutuhan, dan Developer yang bertugas untuk menjalankan list tugas yang sudah dibuat dan disepakati.

Dalam menjalankan Scrum, tentunya terdapat beberapa aktivitas yang digunakan agar pengembangan dapat dilakukan secara teratur. Aktivitas tersebut antara lain Sprint, Sprint Planning, Daily Scrum, Sprint Review, serta Sprint Restropective. Untuk penjelasan lebih lanjut dapat dilihat pada jurnal "The Scrum Guide" karya (Ken Schwaber, 2020) dan website Scrum yang bisa diakses pada <https://www.scrum.org/>. Adapun penelitian dari Andri Rahmanto yang menggunakan metode Scrum ini dalam pengembangan aplikasinya, penelitian tersebut berjudul "Perancangan Arsitektur Aplikasi Budidaya Perikanan Modern pada Backend yang Bertanggung Jawab Melayani Transaksi Query Webservice Dengan Menggunakan Teknologi Flask Microservice".

### **2.13 *Unit Testing***

*Unit testing* merupakan suatu aktivitas dalam pengembangan aplikasi yang bertujuan untuk menguji fungsionalitas serta komponen dari aplikasi yang dikembangkan. *Unit testing* bertujuan untuk memastikan aplikasi dalam kondisi yang sudah sesuai dengan kebutuhan yang sudah disepakati sebelumnya. Proses dalam *unit testing* ini meliputi pengecekan *output* fungsi-fungsi yang ada pada aplikasi berdasarkan *input* dari *tester*.

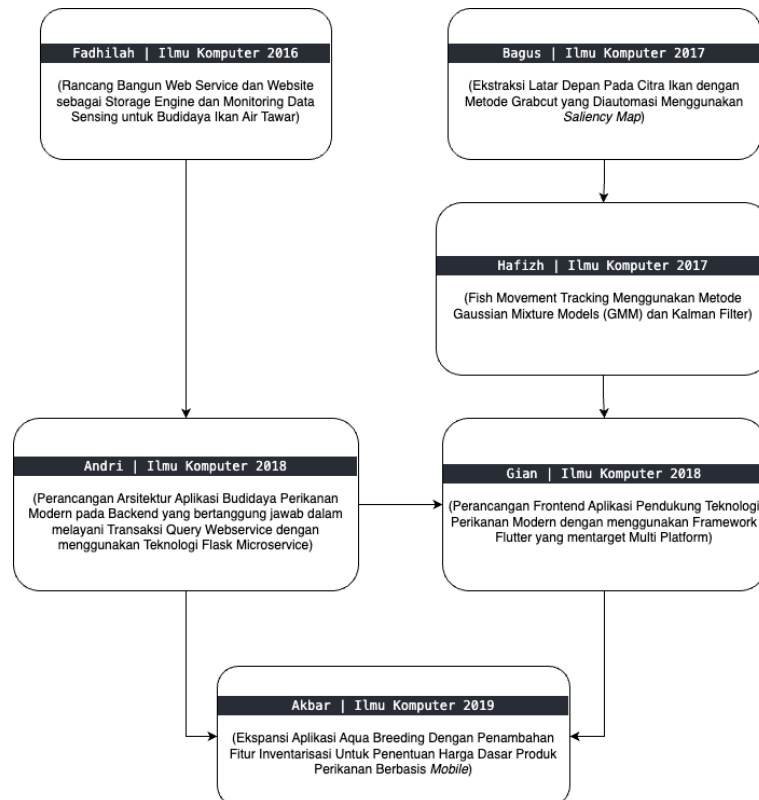
### **2.14 *User Analytics***

*User analytics* adalah proses pengumpulan data aktivitas user selama menggunakan aplikasi. Hal ini mencakup seperti lama penggunaan aplikasi, detail penggunaan fitur, dan sebagainya. Data ini juga dapat menambah wawasan *developer* agar dapat mengembangkan aplikasi yang lebih optimal.

## BAB III

### METODOLOGI PENELITIAN

#### 3.1 Keterhubungan Penelitian



**Gambar 3.1:** Diagram Alur Penelitian *Aquaculture*

Pada diagram diatas, dapat dilihat urutan arah dari topik penelitian Aquaculture. Penelitian pertama kali dimulai oleh (Hadi, 2021) dengan mengembangkan sebuah *web service* serta *website* yang berfungsi sebagai *storage engine* dan *monitoring data sensing* untuk digunakan pada budidaya perikanan air tawar sebagai media penyimpanan data-data sensing dari sensor yang dikirimkan ke sistem serta memonitoringnya dalam bentuk table dan grafik *real-time* serta penelitian yang dilakukan oleh (Nugraha, 2022) dengan tujuan untuk membangun sistem deteksi objek pada citra ikan dengan metode GrabCut yang telah diautomasi menggunakan *saliency map*.

Penelitian (Nugraha, 2022) kemudian dilanjutkan oleh (Alim, 2021) yaitu

merancang dan membangun sebuah sistem yang dapat melakukan pelacakan pergerakan ikan dengan menggunakan metode GMM dan Kalman Filter. Sementara penelitian (Hadi, 2021) belum diterapkan pada aplikasi riset *Aquaculture* dalam waktu dekat sehingga penelitian yang dibuat oleh (Rahmanto, 2022) dilakukan dengan membuat *web service* juga yang bertujuan untuk melayani transaksi *query* berupa monitoring budidaya perikanan yang dibarengi dengan penelitian (Maghriza, 2022) pada bagian perancangan *frontend* sebagai pendukung pada aplikasi yang dikembangkan.

Dalam penelitian yang sudah berjalan ini, penulis mengembangkan penelitian yang dilakukan oleh (Rahmanto, 2022) dan (Maghriza, 2022) dengan membuat fitur baru yaitu manajemen inventaris yang dapat menentukan harga dasar dari produk hasil perikanan.

### 3.2 Depresiasi Aset Bahan Organik

Dalam menyimpan bahan baku pada inventaris, bahan baku tersebut akan mengalami penurunan karena bahan baku mempunyai masa kadaluarsa. Maka dari itu, dapat dimisalkan jika per bulan bahan baku tersebut mengalami penurunan kualitas sebesar 25% dari harga belinya, maka dalam 4 bulan bahan baku tersebut akan kadaluarsa dan sudah tidak lagi berharga. Namun, skala penurunan tersebut bervariasi tergantung pada jenis bahan baku yang digunakan. Variabel  $W_p$  mewakili harga dari bahan baku yang digunakan dan variabel  $\alpha$  mewakili skala depresiasi per bulan dari bahan baku yang digunakan. Formula depresiasi dapat dibuat menjadi persamaan berikut.

$$W_{p\text{sesudah}} = W_{p\text{sebelum}} \times (1 - \alpha) \quad (3.1)$$

- $W_p$  = Harga dari bahan baku yang digunakan
- $\alpha$  = skala depresiasi per bulan dari bahan baku yang digunakan

Ukuran skala depresiasi dari masing-masing bahan organik berbeda-beda, untuk itu diperlukan data lebih lanjut mengenai skala depresiasi dari pakar bahan organik.



### 3.3 Metode Penentuan Harga Dasar

Ketika musim budidaya sudah selesai dan kolam sudah bisa untuk dipanen, maka harga ikan sementara yang bisa digunakan dapat dihitung dengan rumus-rumus berikut.

#### 1. Harga Dasar Produk

$$T = \frac{\sum_{i=1}^3 C_i + \frac{C_l}{n}}{N} \quad (3.2)$$

Dari formula ini, dapat dihasilkan harga dasar produk perikanan per ekornya. Ini dihitung berdasarkan detail pemakaian saat budidaya berlangsung seperti pakan, benih, obat-obatan dan perawatan air, serta pemakaian listrik per kolam.

#### 2. Harga Jual Minimum Produk

$$T = \frac{\sum_{i=1}^3 C_i \frac{C_l}{n} + \frac{C_a}{60}}{N} \quad (3.3)$$

Dari formula ini, dapat dihasilkan harga jual minimum produk perikanan per ekornya. Perhitungan ini sama seperti formula penentuan harga dasar, namun ditambah dengan harga pengeluaran aset yang dibagi dengan rentang waktu penggunaannya (5 tahun = 60 bulan).

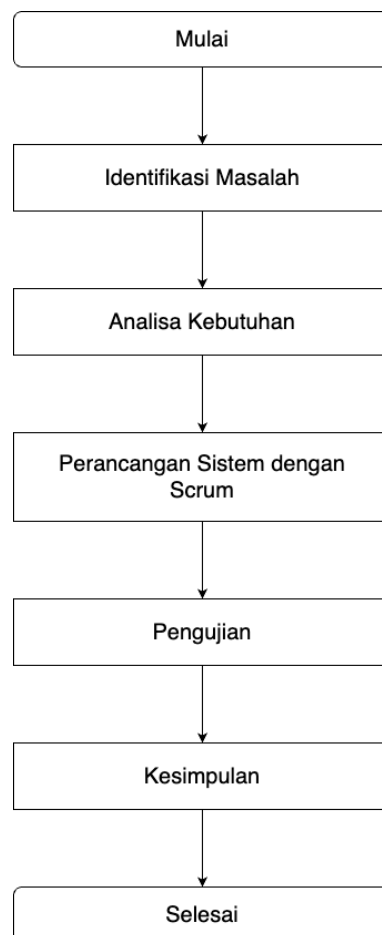
Detail atribut dari dua rumus diatas dapat dilihat sebagai berikut.

- T = Harga jual ikan
- $C_1$  = Harga total benih selama musim berjalan
- $C_2$  = Harga total pakan selama musim berjalan
- $C_3$  = Harga total bahan budidaya selama musim berjalan
- $C_l$  = Harga tagihan listrik
- $C_a$  = Harga total penggunaan aset

- $n$  = Jumlah kolam
- $N$  = Jumlah ikan yang hidup

Harga dari perhitungan tersebut dapat digunakan oleh pembudidaya untuk menjadi harga dasar dalam penjualannya. Tentunya harga tersebut hanya merupakan harga panen dan pembudidaya tidak wajib menggunakan harga tersebut dalam transaksi.

### 3.4 Tahapan Penelitian



**Gambar 3.2:** Alur Tahapan Penelitian

Desain penelitian adalah alur yang dijalankan selama masa pengembangan aplikasi. Pada **Gambar 3.2**, terdapat desain penelitian yang digunakan dalam pembuatan aplikasi ini dengan metode Scrum.

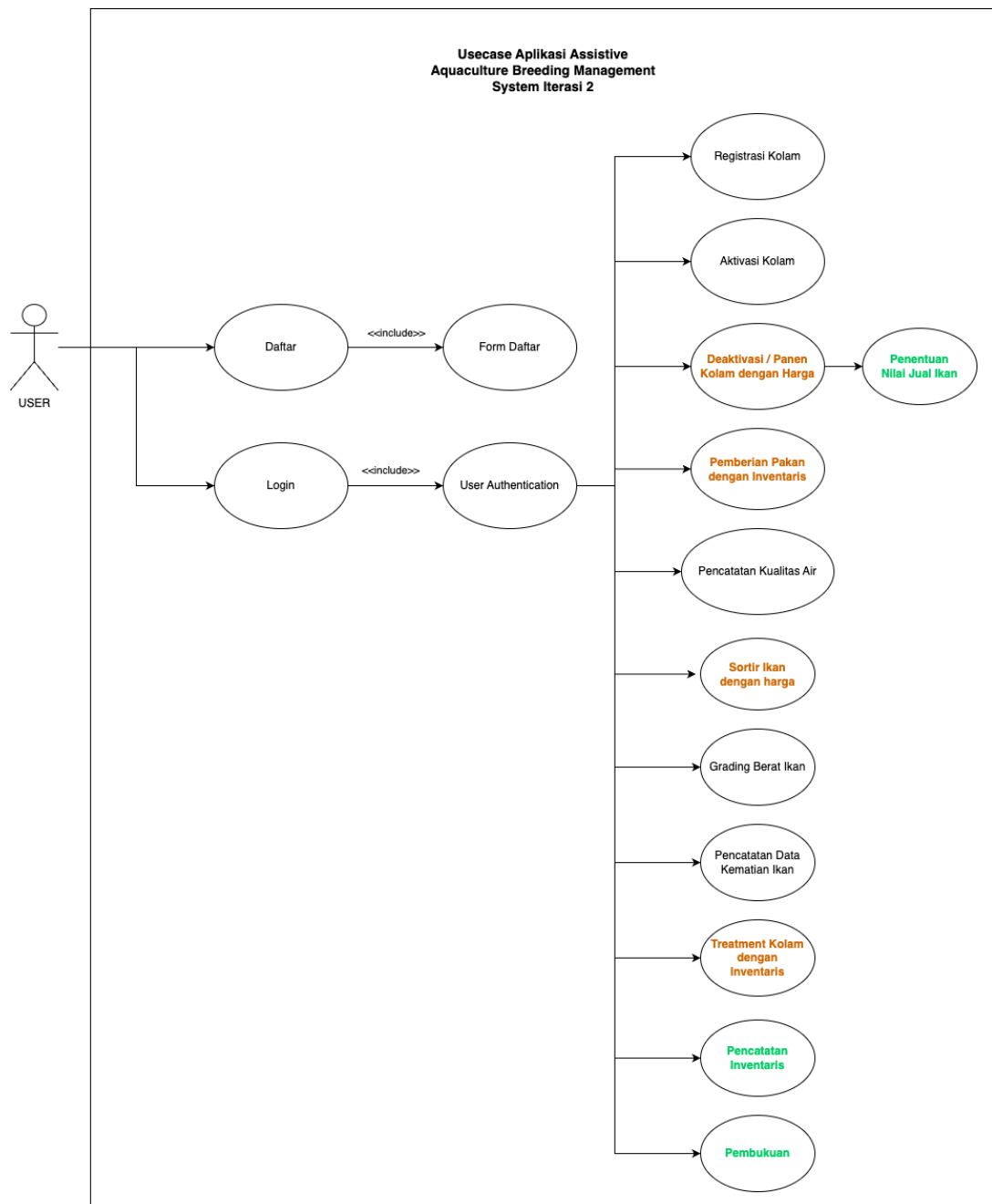
### 3.5 Analisa Kebutuhan

Pada pengembangan aplikasi lanjutan ini, fitur yang ditambahkan adalah fitur manajemen inventaris serta fitur pembukuan. Fitur pencatatan inventaris merupakan fitur yang akan ada pada aplikasi yang berguna untuk para pembudidaya ikan mencatat semua hal yang berhubungan dengan budidaya perikanannya. Hal-hal yang dapat dicatat oleh pembudidaya pada fitur ini seperti bahan baku (termasuk pakan dan bahan budidaya), penggunaan listrik, benih, serta aset yang digunakan selama masa budidaya dilakukan.

Selain mencatat inventaris pada musim budidaya, fitur pencatatan inventaris ini juga dapat menentukan rekomendasi harga jual dari ikan yang dipanen oleh pembudidaya ikan berdasarkan perhitungan dari pengeluaran biaya selama musim budidaya berjalan. Beberapa fitur yang sudah ada di penelitian sebelumnya juga harus diperbarui dengan adanya manajemen inventaris ini seperti panen, pemberian pakan, sortir ikan, serta treatment kolam.

Kemudian untuk fitur pembukuan berguna untuk pembudidaya ikan melihat riwayat musim budidaya yang sudah mereka jalankan. Terdapat beberapa rincian yang ditampilkan seperti biaya pengeluaran sampai berapa total ikan yang terpanen pada musim budidaya tersebut.

Fitur-fitur tersebut dapat dibuat menjadi *use case* pada **Gambar 3.3**. Pada *use case* tersebut, font warna hitam merupakan fitur yang sudah ada pada penelitian sebelumnya yang tidak berubah dan font warna cokelat merupakan fitur sebelumnya yang akan diperbarui pada penelitian ini. Sementara itu, untuk font warna hijau merupakan fitur baru yang akan tersedia pada aplikasi dan dikembangkan pada penelitian ini.

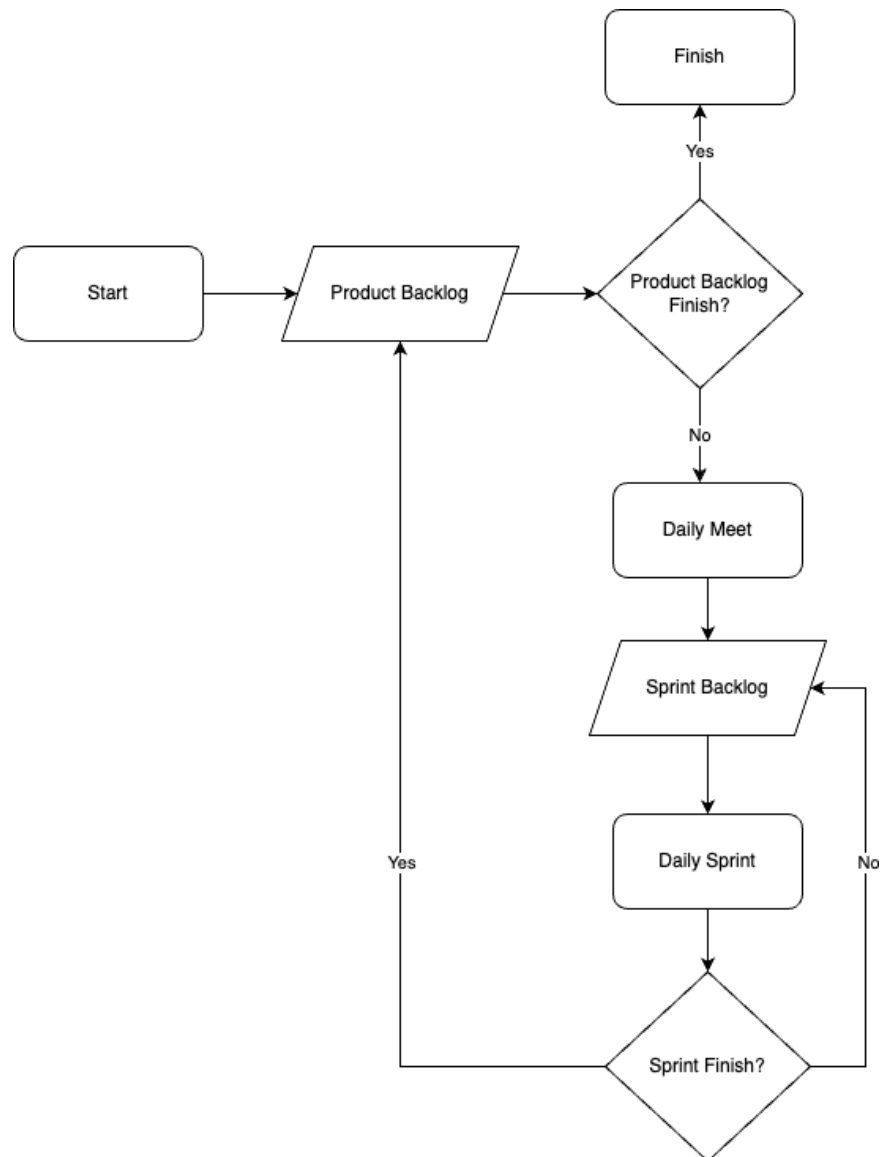


**Gambar 3.3:** *Use Case Aplikasi*

### 3.6 Perancangan Sistem

Pada aplikasi yang akan dibuat pada penelitian ini dikembangkan dengan metode Scrum. Beberapa komponen scrum seperti product backlog, sprint backlog, daily sprint, serta daily meet digunakan agar terwujudnya ketertiban dalam masa

pengembangan aplikasi. Berikut penjelasan dari masing-masing elemen yang ada pada metode Scrum.



**Gambar 3.4:** Tahapan Perancangan Sistem dengan Metode Scrum

#### 1. Product Backlog

Product Backlog adalah tugas-tugas yang **akan** dijalankan pada penelitian dan hal yang pertama kali dilakukan sebelum memulai riset. Daftar tugas yang ada pada Product Backlog ini akan dipindahkan pada Sprint Backlog tergantung pada skala prioritas dari task itu sendiri. Berikut adalah tabel dari Product Backlog yang sudah berjalan.

**Tabel 3.1:** Product Backlog

No	Stories	Sprint	Status
1	Pencatatan inventaris	1, 2	On Progress
2	Aktivasi kolam dengan inventaris	-	Uncomplete
3	Depresiasi aset dalam inventaris	-	Uncomplete
4	Pemberian pakan yang terkoneksi dengan inventaris	-	Uncomplete
5	Treatment kolam yang terkoneksi dengan inventaris	-	Uncomplete
6	Sortir termasuk harga nilai jual ikan	-	Uncomplete
7	Panen termasuk harga nilai jual ikan	-	Uncomplete
8	Pembukuan pencatatan pengeluaran per musim budidaya	-	Uncomplete

## 2. Sprint Backlog

Sprint Backlog adalah daftar tugas yang **harus** dijalankan selama masa Sprint berlangsung. Tugas yang ada pada Sprint Backlog bersifat fleksibel seiring dengan berjalannya Sprint.

## 3. Sprint

Progres sprint dilaksanakan ketika list task pada sprint backlog sudah disepakati bersama. Periode pengerjaan sprint bervariasi tergantung pada kesulitan task dari sprint backlog tersebut.

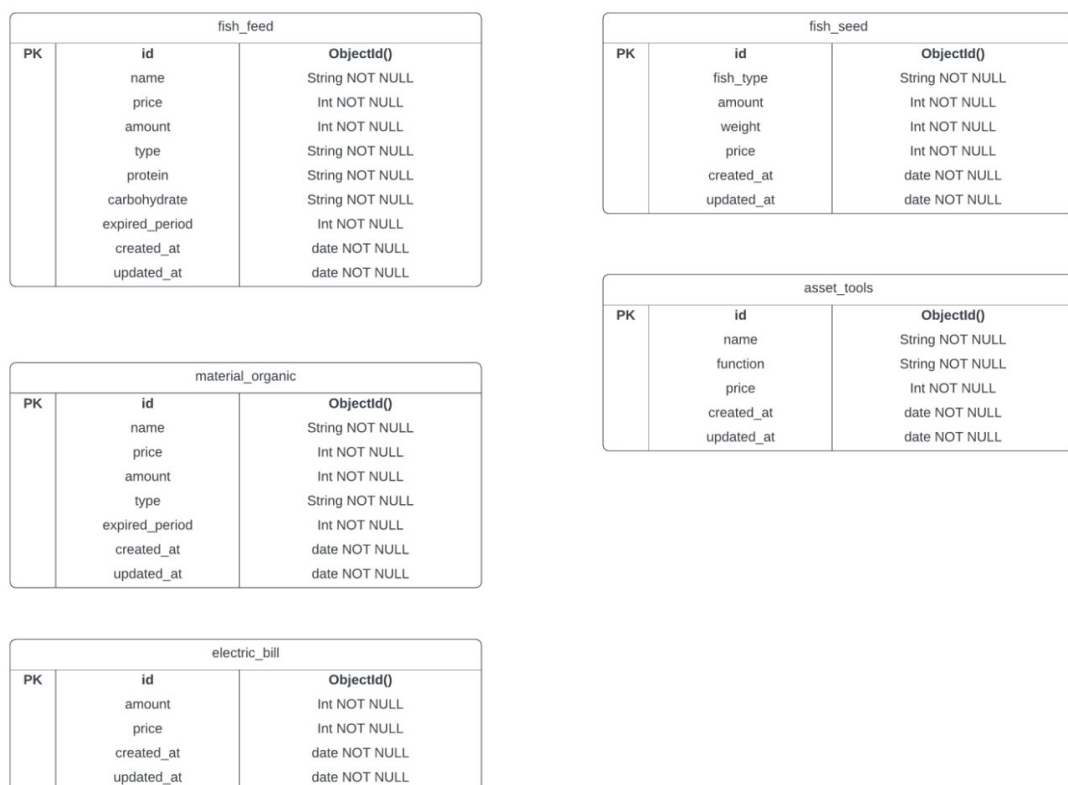
### (a) Sprint 1

Sprint 1 dilaksanakan pada tanggal 07 Maret 2023 - 29 Maret 2023. Detail dari Sprint 1 ini adalah mengerjakan tugas yang ada pada Sprint 1 Backlog di tabel berikut.

**Tabel 3.2:** Sprint 1 Backlog

No	Stories	Task	Status
1	Fitur pencatatan inventaris	<ul style="list-style-type: none"> <li>- Membuat skema database dari pencatatan inventaris</li> <li>- Membuat integrasi skema database dengan skema database sebelumnya</li> <li>- Membuat mockup dari fitur inventaris</li> </ul>	Complete  Complete  Complete

Berikut merupakan skema database yang mewakili fitur inventaris dapat dilihat pada **Gambar 3.5** berikut.

**Gambar 3.5:** Skema Database Fitur Inventaris

Dari skema database tersebut, terdapat lima opsi kategori inventaris yang sudah dijelaskan sebelumnya. Pada skema database ini, masing-masing

kategori memiliki kebutuhan yang berbeda antara lain.

- i. fish\_feed (Pakan Ikan)
- ii. material\_organic (Bahan Organik)
- iii. electric\_bill (Tagihan Listrik)
- iv. fish\_seed (Benih Ikan)
- v. asset\_tools (Peralatan)

Dalam tabel database tersebut, pada kolom pertama terdapat jenis *key* yang dijadikan patokan dalam tabel database tersebut. Kemudian kolom kedua dan ketiga merupakan hubungan antara nama data dan tipe data yang mewakili nama data tersebut.

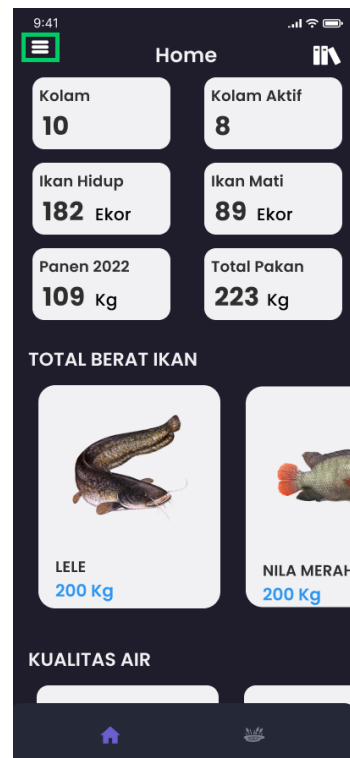
Setelah skema database dari inventaris telah dibuat, tabel-tabel database tersebut harus diintegrasikan dengan skema database sebelumnya untuk menyesuaikan kebutuhan fitur yang akan dibuat nantinya. Berikut merupakan skema database yang telah diintegrasikan dengan skema database dari inventaris dapat dilihat pada **Gambar 3.6**.



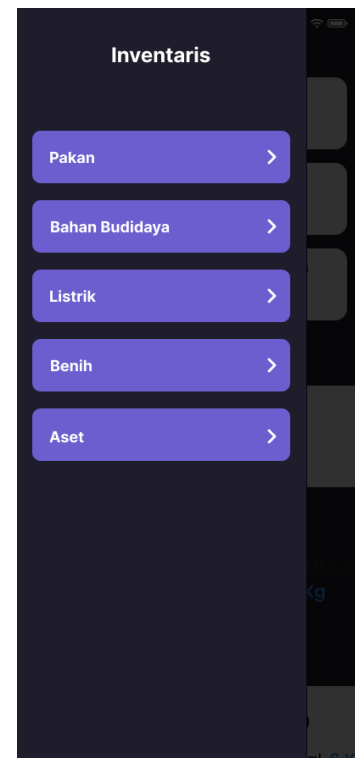


Dari integrasi diatas, tabel fish\_feed diintegrasikan dengan tabel feed\_type yang digunakan untuk input pakan dan tabel material\_organic diintegrasikan dengan tabel pond\_treatment karena dalam fitur treatment kolam diperlukan data dari tabel material organik tersebut. Lalu tabel fish\_seed diintegrasikan dengan tabel pond\_activation dan fish\_log untuk aktivasi kolam dan perhitungan jumlah ikan. Sementara itu, tabel electric\_bill dan asset\_tools merupakan individu yang tidak terintegrasi dengan tabel yang lain. Hal ini dikarenakan tabel tersebut hanya untuk menyimpan datanya saja dan tidak digunakan di dalam fitur.

Berikut merupakan mockup dari fitur inventaris yang mencakup skema database sebelumnya.

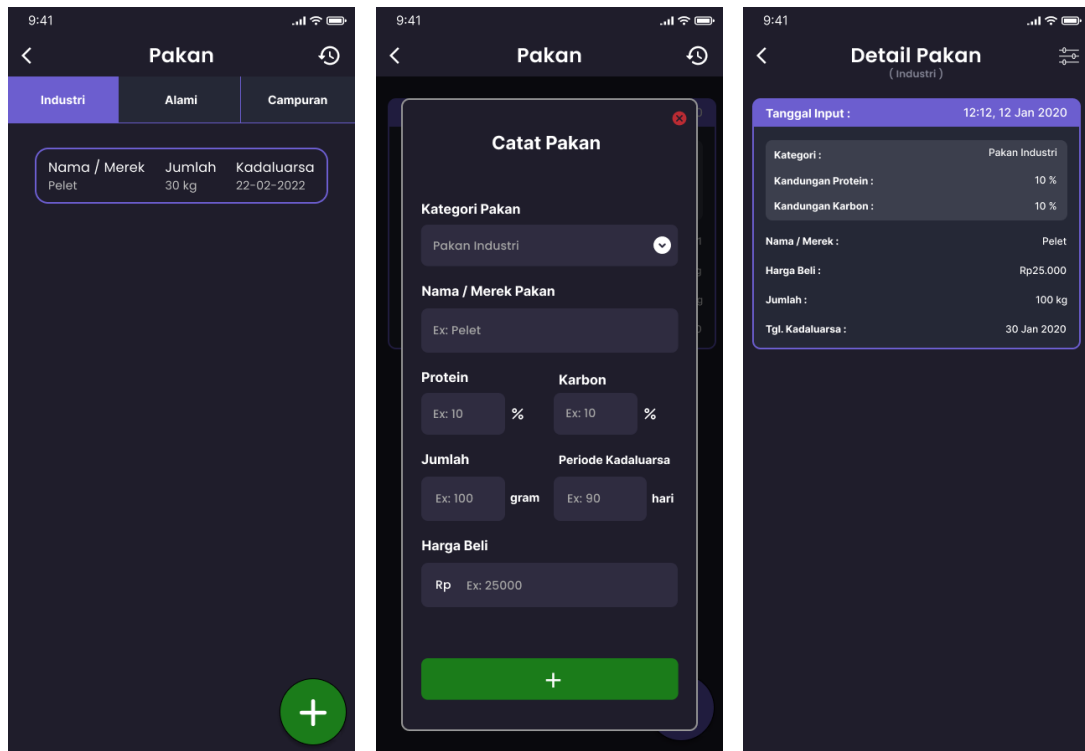


**Gambar 3.7:** Halaman Dashboard



**Gambar 3.8:** Halaman Menu Inventaris

Pada halaman dashboard, dipojok kiri atas terdapat ikon *hamburger* atau list yang ketika ditekan akan menampilkan halaman menu inventaris seperti **Gambar 3.8**. Masing-masing list menu yang ada pada halaman menu inventaris memiliki fungsi yang sesuai dengan skema inventaris.

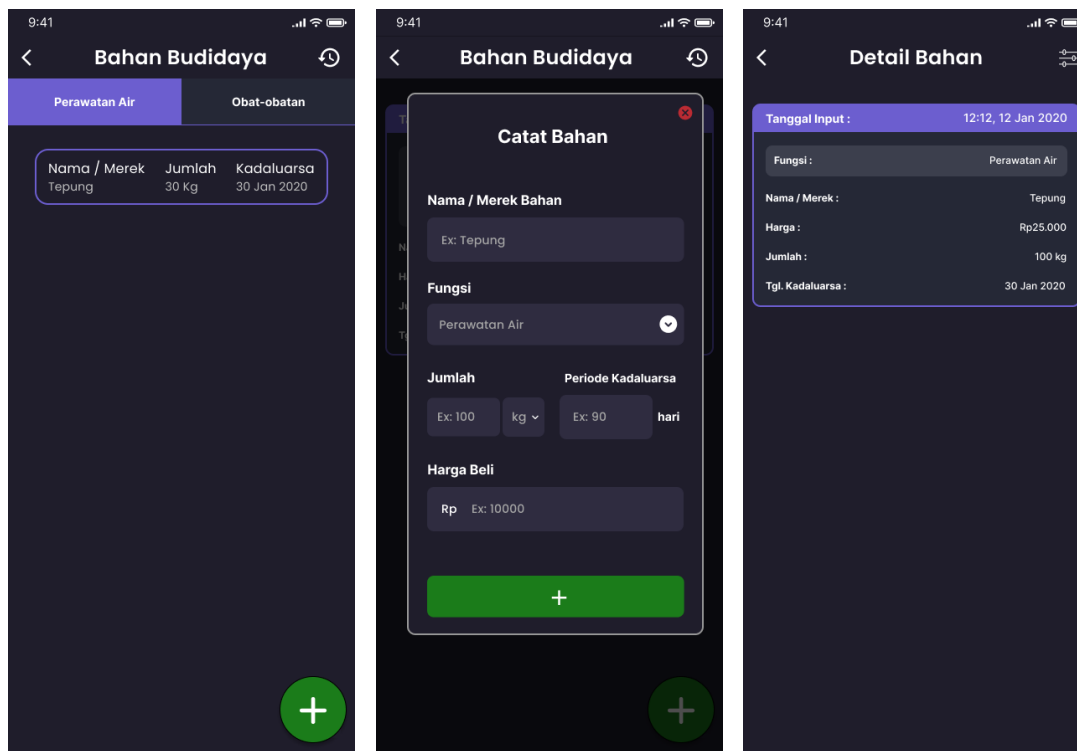


**Gambar 3.9:** Halaman Data Inventaris Pakan      **Gambar 3.10:** Halaman Input Inventaris Pakan      **Gambar 3.11:** Halaman Detail Inventaris Pakan

Jika pada halaman menu inventaris sebelumnya dipilih menu "Pakan", maka akan masuk ke halaman data inventaris pakan. Pada halaman ini terdapat 3 jenis pakan yaitu pakan industri (pelet), alami (tumbuh-tumbuhan), serta campuran (tepung, terigu, dll). Masing-masing jenis pakan memiliki detail data yaitu nama atau merek pakan, total jumlah pakan yang tersedia, serta tanggal kadaluarsa dari pakan tersebut.

Tombol (+) yang ada di pojok kanan bawah akan mengarahkan ke halaman input dari inventaris pakan. Disini diberikan form input yang beragam seperti yang ada pada **Gambar 3.10**.

Sementara tombol riwayat yang ada di pojok kanan atas akan mengarahkan ke halaman detail dari inventaris pakan. Di halaman ini ditampilkan detail pemasukkan pakan ke sistem inventaris.



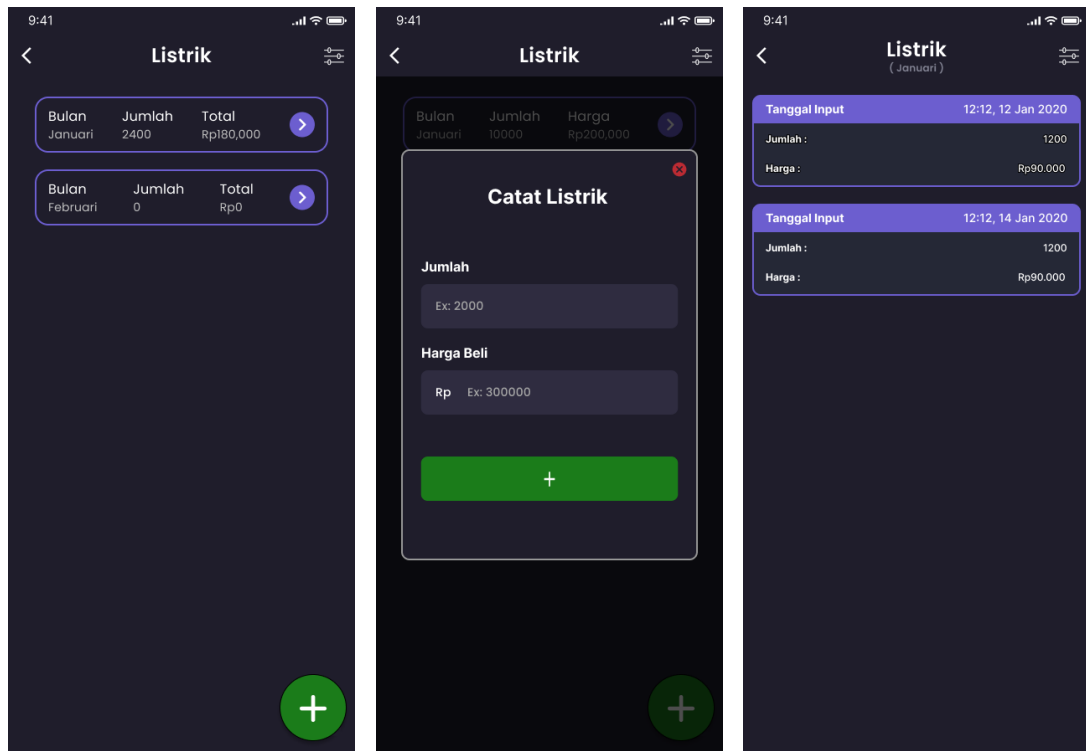
**Gambar 3.12:** Halaman Data Inventaris Bahan Budidaya

**Gambar 3.13:** Halaman Input Inventaris Bahan Budidaya

**Gambar 3.14:** Halaman Detail Inventaris Bahan Budidaya

Jika pada halaman menu inventaris sebelumnya dipilih menu "Bahan Budidaya", maka akan masuk ke halaman data inventaris bahan budidaya. Pada halaman ini terdapat 2 jenis bahan budidaya yang dibagi berdasarkan fungsi yaitu perawatan air dan obat-obatan (Methylene Blue, dll). Sama seperti pada inventaris pakan, masing-masing jenis bahan budidaya memiliki detail data yaitu nama atau merek, total jumlah yang tersedia, serta tanggal kadaluarsa.

Sama seperti halaman inventaris pakan, tombol (+) mengarahkan ke halaman input seperti **Gambar 3.13** dan tombol riwayat akan mengarahkan ke halaman detail inventaris seperti pada **Gambar 3.14**.



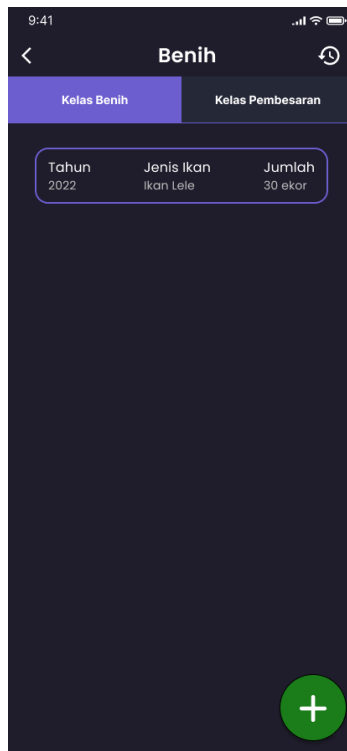
**Gambar 3.15:** Halaman Data Inventaris Tagihan Listrik

**Gambar 3.16:** Halaman Input Inventaris Tagihan Listrik

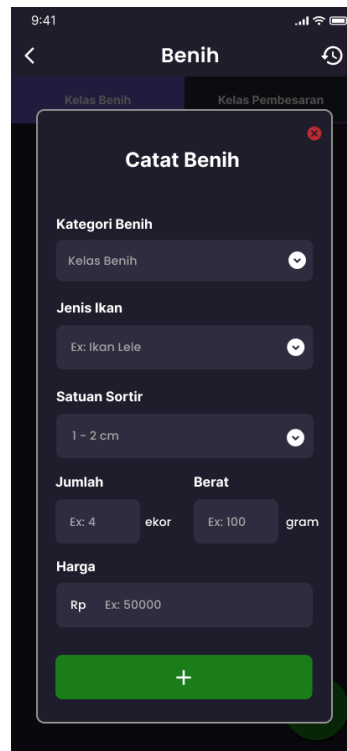
**Gambar 3.17:** Halaman Detail Inventaris Tagihan Listrik

Jika pada halaman menu inventaris sebelumnya dipilih menu "Listrik", maka akan masuk ke halaman data inventaris tagihan listrik. Pada halaman ini terdapat list dari tagihan listrik perbulannya yang digunakan oleh pembudidaya, data yang ditampilkan berupa bulan, jumlah listrik, serta total biaya tagihan. Jika list bulan tersebut ditekan, maka akan pindah ke halaman detail dari tagihan listrik dibulan tersebut.

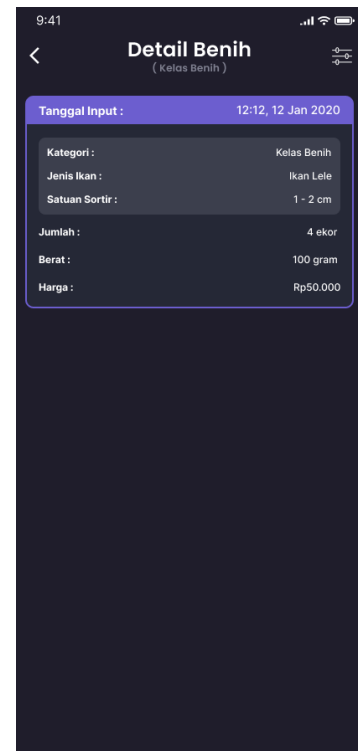
Untuk tombol (+) yang ada di pojok kanan bawah, jika ditekan akan masuk ke halaman input tagihan. Form yang harus diisi hanya jumlah token listrik dan harga belinya. Sementara tombol filter yang ada di pojok kanan atas berfungsi untuk memfilter data sesuai keinginan user.



**Gambar 3.18:** Halaman Data Inventaris Benih



**Gambar 3.19:** Halaman Input Inventaris Benih



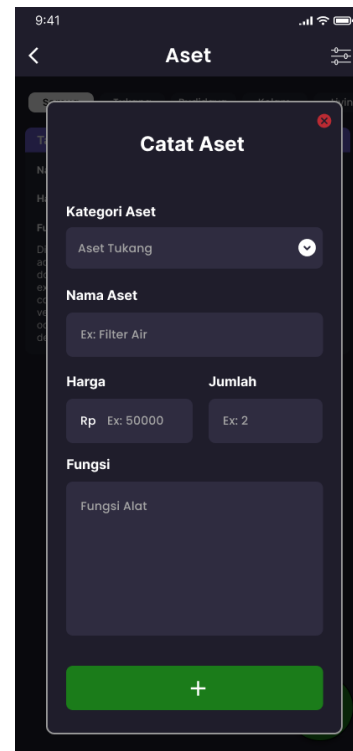
**Gambar 3.20:** Halaman Detail Inventaris Benih

Jika pada halaman menu inventaris sebelumnya dipilih menu "Benih", maka akan masuk ke halaman data inventaris benih. Pada halaman ini terdapat list dari benih yang sudah diinput pada sistem inventaris yang terbagi menjadi dua jenis yaitu benih jenis kelas benih (kecil) dan kelas pembesaran (besar). Masing-masing data memiliki detail seperti tahun benih di input, jenis benih, dan jumlah dari benih.

Untuk tombol (+) yang ada di pojok kanan bawah, jika ditekan akan dinavigasikan ke halaman input benih ikan. Halaman input ini memiliki form seperti pada **Gambar 3.19**. Terdapat dua jenis form yang berbeda berdasarkan kategori yang dipilih, untuk kategori kelas benih pada bagian ukurannya menggunakan satuan sortir sementara untuk kategori kelas pembesaran menggunakan panjang dan lebar untuk ukurannya.



**Gambar 3.21:** Halaman Data Inventaris Aset



**Gambar 3.22:** Halaman Input Inventaris Aset

Jika pada halaman menu inventaris sebelumnya dipilih menu "Aset", maka akan masuk ke halaman data inventaris aset. Pada halaman ini, ditampilkan jenis dari aset-aset yang digunakan selama masa budidaya.

Aset dibagi menjadi empat jenis kategori yaitu aset tukang (aset yang diperlukan pembudidaya), aset budidaya (aset yang dibutuhkan selama budidaya berlangsung), aset kolam (aset yang digunakan dalam kolam budidaya), dan aset living (aset yang diperlukan selama berlangsungnya musim budidaya).

Tombol (+) pada pojok kanan bawah berfungsi untuk navigasi ke halaman input sementara tombol filter pada pojok kanan atas berfungsi untuk filter data.

#### (b) **Sprint 2**

Sprint 2 dilaksanakan pada tanggal 30 Maret 2023 - 15 April 2023. Detail dari Sprint 2 ini adalah mengerjakan tugas yang ada pada Sprint 2 Backlog di tabel berikut.

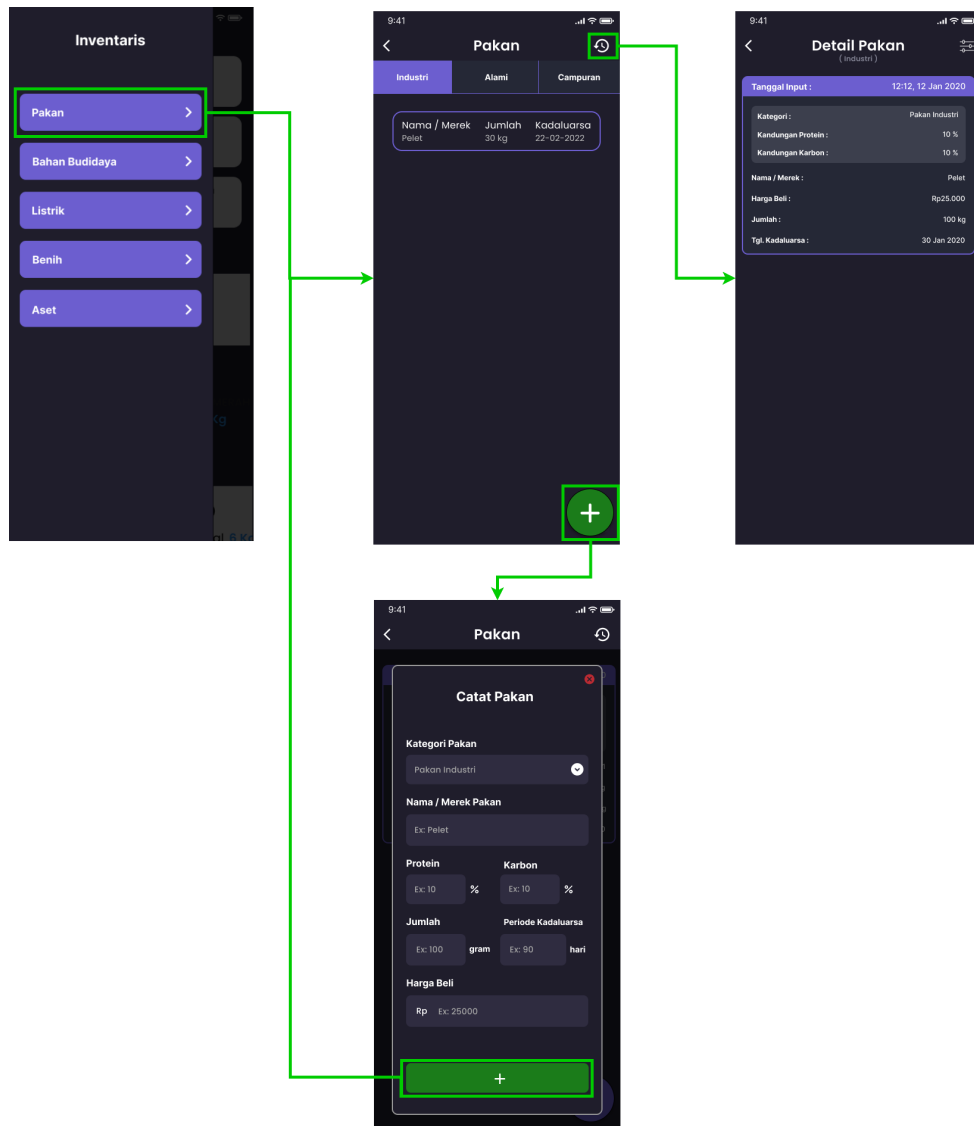
**Tabel 3.3:** Sprint 2 Backlog

No	Stories	Task	Status
1	Fitur pencatatan inventaris	<ul style="list-style-type: none"> <li>- Membuat alur UI/UX dari design aplikasi</li> <li>- Mengupdate skema database pada inventaris</li> </ul>	Complete  Complete

Selama masa Sprint 2 berlangsung, tim bertemu dengan perwakilan dari Dinas Perikanan Bogor. Dari pertemuan itu, salah satunya terdapat beberapa perubahan user requirement sehingga perlu merubah skema database yang sudah dibuat pada Sprint 1. Lalu, terdapat juga pertimbangan perubahan penentuan harga jual ikan dengan ditambahkannya perhitungan aset namun hal ini belum ditentukan akan masuk perhitungan atau tidak.

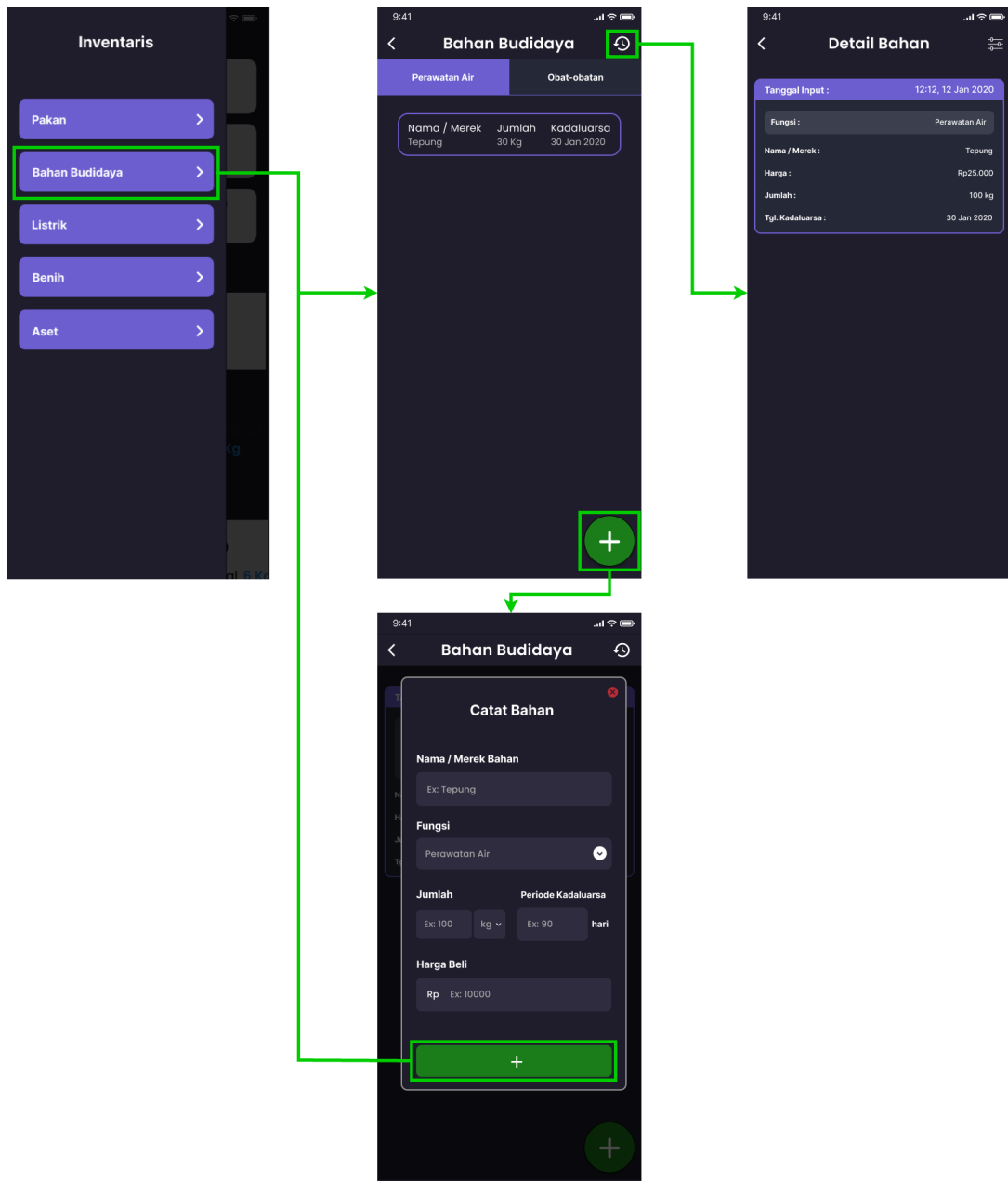
Berikut merupakan alur user sebagai pengguna aplikasi berdasarkan mockup yang sudah dibuat pada Sprint 1.





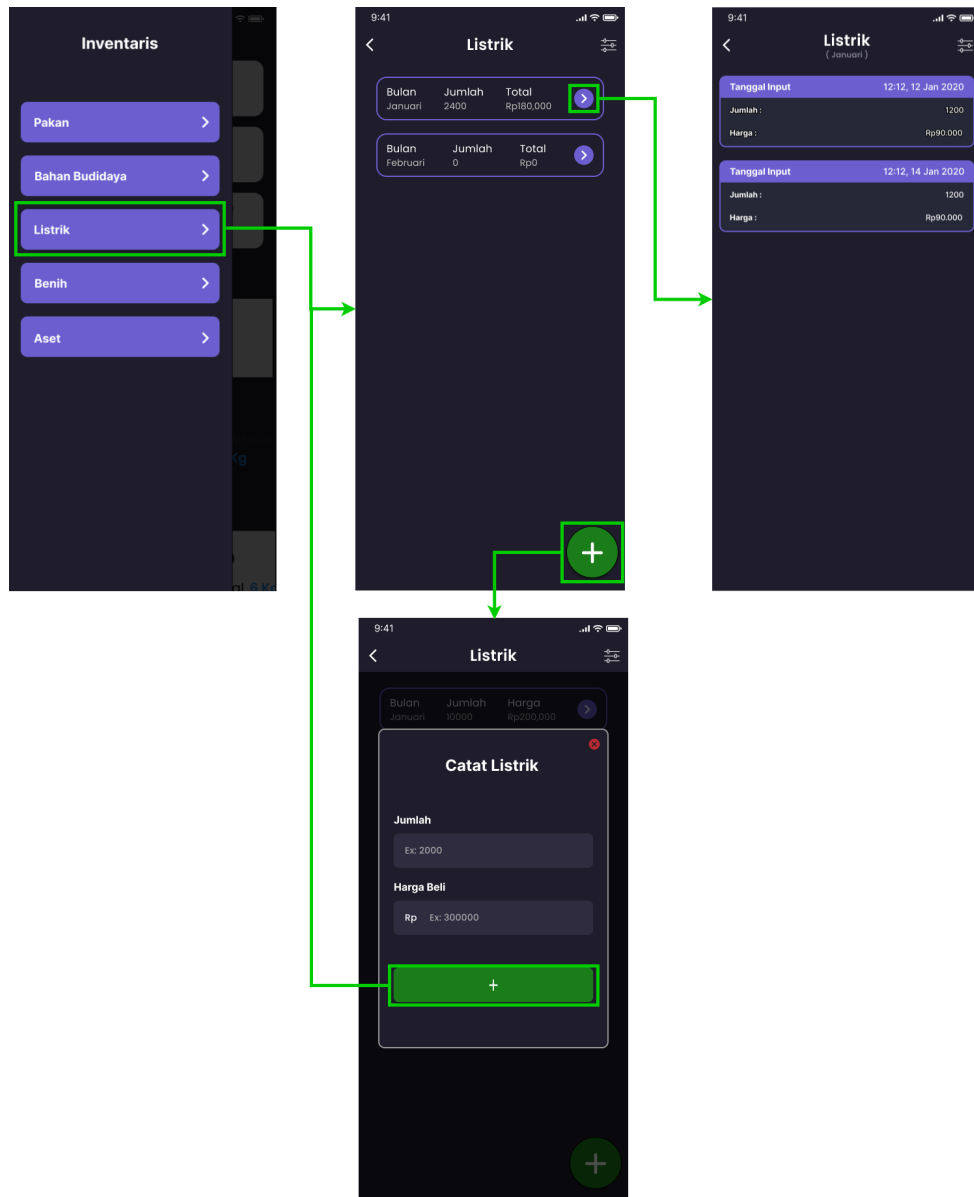
**Gambar 3.23:** Alur Inventaris Pakan

Pada **Gambar 3.23** merupakan alur dari inventaris pakan. Pengguna akan memilih menu "Pakan" dan masuk ke halaman data inventaris pakan, kemudian tombol (+) akan menavigasikan pengguna ke halaman input pakan dan tombol riwayat akan menavigasikan pengguna ke halaman detail input pakan.



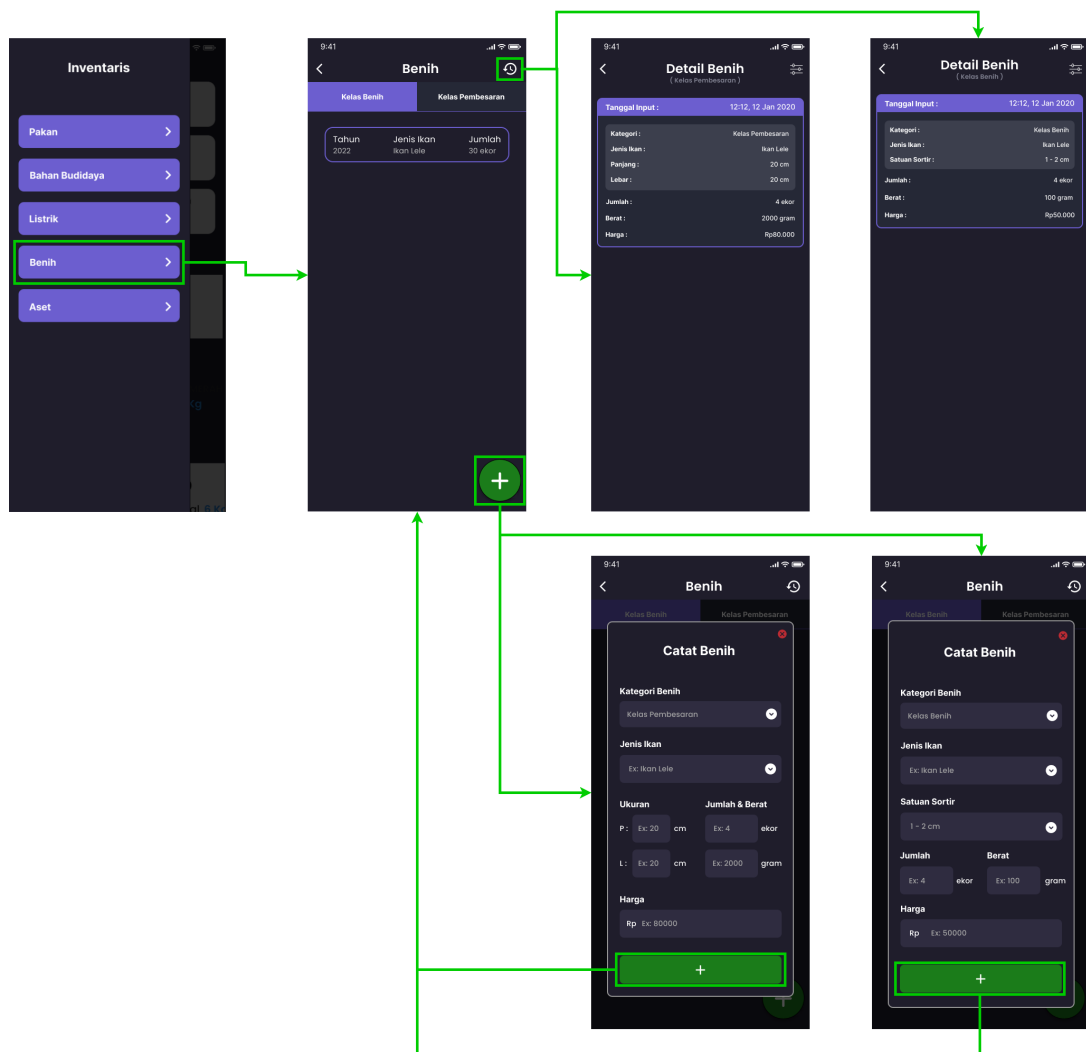
**Gambar 3.24:** Alur Inventaris Bahan Budidaya

Pada **Gambar 3.24** merupakan alur dari inventaris bahan budidaya. Pengguna akan memilih menu "Bahan Budidaya" dan masuk ke halaman data inventaris budidaya, kemudian tombol (+) akan menavigasikan pengguna ke halaman input bahan budidaya dan tombol riwayat akan menavigasikan pengguna ke halaman detail input bahan budidaya.



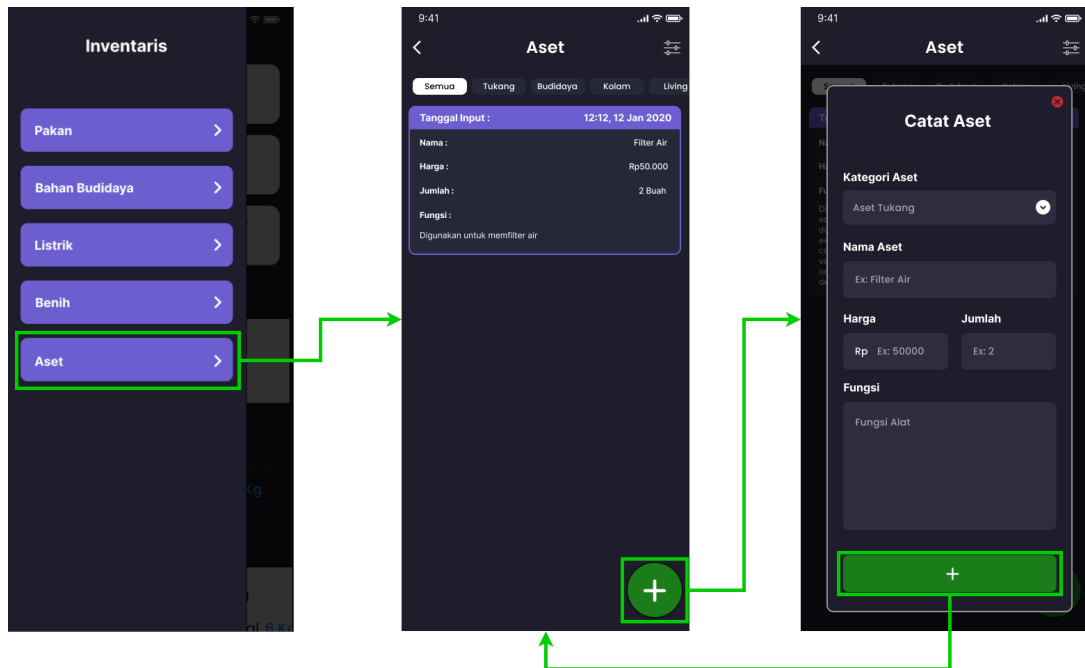
**Gambar 3.25:** Alur Inventaris Listrik

Pada **Gambar 3.25** merupakan alur dari inventaris listrik. Pengguna akan memilih menu "Listrik" dan masuk ke halaman data inventaris listrik, kemudian tombol (+) akan menavigasikan pengguna ke halaman input listrik serta jika pengguna menekan salah satu list bulan pada data listrik, maka akan masuk ke halaman detail input tagihan listrik.



**Gambar 3.26:** Alur Inventaris Benih

Pada **Gambar 3.26** merupakan alur dari inventaris benih. Pengguna akan memilih menu "Benih" dan masuk ke halaman data inventaris benih, kemudian tombol (+) akan menavigasikan pengguna ke halaman input benih dan tombol riwayat akan menavigasikan pengguna ke halaman detail input benih.



**Gambar 3.27:** Alur Inventaris Aset

Pada **Gambar 3.27** merupakan alur dari inventaris aset. Pengguna akan memilih menu "Aset" dan masuk ke halaman data inventaris aset, kemudian tombol (+) akan menavigasikan pengguna ke halaman input aset.

Setelah semua alur telah selesai dibuat, selanjutnya merupakan perubahan skema database inventaris yang dapat dilihat pada **Gambar 3.28** berikut.

fish_feed		
PK	id	Objectid()
	feed_category_id	Int NOT NULL
	name	String NOT NULL
	price	Int NOT NULL
	amount	Int NULL
	protein	Int NOT NULL
	carbon	Int NOT NULL
	expired_period	Int NULL
	created_at	date NOT NULL
	updated_at	date NOT NULL

fish_seed		
PK	id	Objectid()
	fish_seed_category_id	Int NOT NULL
	fish_type	String NOT NULL
	amount	Int NOT NULL
	weight	Int NOT NULL
	sorting_size	String NULL
	length	Int NULL
	width	Int NULL
	price	Int NOT NULL
	created_at	date NOT NULL
	updated_at	date NOT NULL

material_parts		
PK	id	Objectid()
	function_id	Int NOT NULL
	name	String NOT NULL
	price	Int NOT NULL
	amount	Int NOT NULL
	type	String NOT NULL
	expired_period	Int NOT NULL
	created_at	date NOT NULL
	updated_at	date NOT NULL

asset_tools		
PK	id	Objectid()
	asset_category_id	Int NOT NULL
	name	String NOT NULL
	function	String NOT NULL
	amount	Int NOT NULL
	price	Int NOT NULL
	created_at	date NOT NULL
	updated_at	date NOT NULL

electric_bill		
PK	id	Objectid()
	amount	Int NOT NULL
	price	Int NOT NULL
	created_at	date NOT NULL
	updated_at	date NOT NULL

**Gambar 3.28:** Update Skema Database Inventaris

Berdasarkan skema database inventaris tersebut, jika dibandingkan dengan skema database inventaris sebelumnya pada **Gambar 3.5** terdapat pembaruan pada bagian inventaris pakan, bahan budidaya, benih, dan aset.

Pada skema database di inventaris pakan (fish\_feed), ditambahkan *key* **feed\_category\_id** karena pada inventaris pakan diharuskan memilih kategori pakan yang akan dimasukkan. Untuk itu feed\_category\_id berperan untuk menampung jenis kategori pakan yang akan diinput. Sebelumnya jenis inventaris pakan tidak memiliki kategori, sehingga perlu ditambahkan *key* baru untuk jenis data kategori tersebut.

Kemudian, di skema database inventaris bahan budidaya (material\_parts) ditambahkan *key* **function\_id** karena bahan budidaya dibagi menjadi dua fungsi yaitu perawatan air dan obat-obatan. Sebelumnya inventaris bahan budidaya tidak memiliki kategori, sehingga harus ditambahkan *key* baru pada database untuk menampung jenis data tersebut.

Lalu pada skema database inventaris benih ditambahkan *key*

**fish\_seed\_category\_id**, **sorting\_size**, **length**, serta **width**. *key* tersebut ditambahkan karena pada benih dibagi menjadi dua jenis yaitu kelas benih dan kelas pembesaran. Masing-masing kategori memiliki jenis data pengukuran yang berbeda, pada kelas benih digunakan pengukuran satuan sortir dengan *key* **sorting\_size** sementara kelas pembesaran digunakan pengukuran panjang dan lebar dengan *key* **length** dan **width**. Sebelumnya untuk benih tidak terdapat kategori dan jenis ukuran benih sehingga *key* baru diperlukan untuk menampung data tersebut.

Terakhir terdapat perubahan pada skema database inventaris aset yaitu ditambahkan *key* **asset\_category\_id** dan **amount**. Sebelumnya inventaris aset hanya menampung segala jenis aset yang digunakan pada musim budidaya tanpa adanya jenis kategori dan jumlah yang spesifik, namun di skema database sekarang dapat ditentukan jenis kategori pada aset dan berapa jumlah aset yang digunakan sehingga pemantauan aset yang digunakan menjadi lebih detail.

#### 4. Sprint Review

Setelah Sprint berjalan, setiap minggunya diadakan meet bersama tim untuk melaksanakan Sprint Review yang bertujuan untuk melaporkan perkembangan Sprint baik itu proses ataupun hambatan selama pengerjaan Sprint.

#### 5. Deploy Sistem

Ketika semua task sprint yang ada di sprint backlog selesai, maka aplikasi akan di deploy untuk dijalankan pengujian pada aplikasi. Pengujian aplikasi ini dilakukan dengan menggunakan Unit Testing dan User Analytics.

### 3.7 Pengujian

Di tahap pengujian ini, peneliti akan melakukan uji aplikasi menggunakan dua jenis pengujian yaitu unit testing dan User Acceptance Test (UAT). Pengujian unit testing dilakukan oleh tim internal developer aplikasi untuk memastikan kepastian fungsi fitur dan cara kerja fitur agar aplikasi sesuai dengan kebutuhan. Sementara UAT dilakukan oleh user untuk mengetahui apakah aplikasi sudah memenuhi kebutuhan dan layak digunakan.

## 1. Unit Testing

Pengujian dengan unit testing ini dibuat berdasarkan product backlog dan daftar sprint-sprint backlog yang sudah selesai. Pengujian internal ini mengubah status Complete pada task di Sprint Backlog menjadi Tested. Berikut skenario pengujian yang akan dilakukan dapat dilihat pada **Tabel 3.4** berikut.

**Tabel 3.4:** Skenario Unit Testing

Jenis Fitur	Skenario Pengujian
Pencatatan Inventaris	Saat aplikasi dibuka dan sudah terautentikasi, halaman dashboard akan tampil
	Di halaman dashboard, terdapat tombol list yang ada pada pojok kiri atas aplikasi
	Jika tombol list ditekan, maka akan tampil beberapa list menu inventaris
	Ketika salah satu tombol pada list inventaris ditekan, maka akan masuk ke halaman detail data inventaris dari menu yang dipilih
	Pada halaman detail data inventaris, ketika tombol riwayat di pojok kanan atas ditekan akan muncul rincian input pada sistem inventaris
	Pada halaman detail data inventaris, Ketika tombol (+) yang ada di pojok kanan bawah ditekan akan masuk ke halaman input data inventaris
Pencatatan Inventaris (Listrik dan Aset)	Khusus untuk inventaris listrik dan aset, dipojok kanan atas halaman detail data inventaris terdapat tombol filter yang jika ditekan akan memfilter data inventaris berdasarkan jenis filter yang dipilih



## 2. User Analytics

Pengujian ini dilakukan kepada pembudidaya ikan dengan 3 tahap yaitu tutorial, logging / survei penggunaan aplikasi, dan interview. Tahapan tersebut dilakukan secara bertahap dan masing-masing tahap dijelaskan seperti berikut.

### (a) Tahap Tutorial

Pada tahap ini, dilakukan sesi pertemuan dengan para pembudidaya untuk menjelaskan fitur-fitur yang ada pada aplikasi secara keseluruhan.

### (b) Tahap Logging

Pada tahap ini, setiap aktivitas aplikasi yang dilakukan oleh pembudidaya akan dicatat oleh sistem. Hal ini mencakup seperti detail halaman yang diakses dan detail penggunaan fitur yang ada.

### (c) Tahap Interview

Pada tahap ini, terdapat sesi interview dengan pembudidaya mengenai penggunaan aplikasi. Dapat ditanyakan detail dari penggunaan aplikasi berdasarkan data logging yang sudah dikumpulkan sebelumnya.

## DAFTAR PUSTAKA

- Alim, H. (2021). Fish movement tracking dengan menggunakan metode gmm dan kalman filter. *Program Studi Ilmu Komputer Fakultas Matematika Dan Ilmu Pengetahuan Alam Universitas Negeri Jakarta 2021*.
- Chen, Y. (2020). Implementation of water quality management platform for aquaculture based on big data. *2020 International Conference on Computer Information and Big Data Applications (CIBDA)*.
- Flask (2010). <https://flask.palletsprojects.com/en/2.3.x/>.
- Flutter (2017). <https://flutter.dev/>.
- Hadi, F. P. (2021). Rancang bangun web service dan website sebagai storage engine dan monitoring data sensing untuk budidaya ikan air tawar. *Program Studi Ilmu Komputer Fakultas Matematika Dan Ilmu Pengetahuan Alam Universitas Negeri Jakarta 2021*.
- Ken Schwaber, J. S. (2020). The definitive guide to scrum: The rules of the game. *The Scrum Guide*.
- Lin, Y.-B. (2019). Fishtalk: An iot-based mini aquarium system. *IEEE Access*.
- Maghriza, G. C. (2022). Perancangan frontend aplikasi pendukung teknologi perikanan modern dengan menggunakan framework flutter yang mentarget multi platform. *Program Studi Ilmu Komputer Fakultas Matematika Dan Ilmu Pengetahuan Alam Universitas Negeri Jakarta 2022*.
- MongoDB (2009). <https://www.mongodb.com/>.
- Nugraha, B. (2022). Ekstraksi latar depan pada citra ikan dengan metode grabcut yang diautomasi menggunakan saliency map. *Program Studi Ilmu Komputer Fakultas Matematika Dan Ilmu Pengetahuan Alam Universitas Negeri Jakarta 2022*.
- Ouyang, B. (2021). Initial development of the hybrid aerial underwater robotic system (haucs): Internet of things (iot) for aquaculture farms. *IEEE Intenet of Things Journal*.

Rahmanto, A. (2022). Perancangan arsitektur aplikasi budidaya perikanan modern pada backend yang bertanggung jawab dalam melayani transaksi query webservice dengan menggunakan teknologi flask microservice. *Program Studi Ilmu Komputer Fakultas Matematika Dan Ilmu Pengetahuan Alam Universitas Negeri Jakarta 2022*.

Scrum (2010). <https://www.scrum.org/>.

Sim, S. (2022). *Dasar-Dasar Manajemen Keuangan (Fundamentals of Financial Management)*. Uwais Inspirasi Indonesia.

Supitriyani (2022). *Management Control*. Media Sains Indonesia.