

## **Laporan Tubes**

**“APLIKASI LAYANAN DOKTER HEWAN”**



Dikerjakan oleh:

Anna Kurniaty – 201524001

Dimas Wisnu Saputro – 201524005

Hasanah – 201524010

1AD4 Jurusan Teknik Komputer dan Informatika

**Program Studi D4 Teknik Informatika**

**Jurusan Teknik Komputer dan Informatika**

**Politeknik Negeri Bandung**

**2020/2021**

## Bab I

### Deskripsi Aplikasi

#### 1. Deskripsi Aplikasi

QueenPet adalah suatu aplikasi layanan dokter hewan yang berfungsi untuk mempermudah layanan pada sebuah klinik dokter hewan. Aplikasi ini memiliki fitur menambahkan pasien, menampilkan antrian, memanggil antrian, bantuan, dan credit. Pada aplikasi ini customer akan mendapatkan informasi mengenai waktu estimasi pelayanan, waktu estimasi mulai pemeriksaan, dan waktu selesai tergantung dari jenis penyakit yang diderita kucing. Pada aplikasi ini juga terdapat pengkategorian penyakit kucing berdasarkan penyakit yang diderita kucing.

Dibawah ini adalah pengkategorian penyakit kucing, yaitu:

- 1) Ringan : penyakit kulit, luka ringan, bersin
- 2) Sedang : cacingan, diare, luka dalam
- 3) Berat : gangguan kerongkongan yang mengeluarkan lendir yang berbau busuk, kuning, terkena virus

Selain itu pada aplikasi ini Customer akan dilayani sesuai dengan antrian dari urutan kedatangannya dan aplikasi akan memprioritaskan Customer berdasarkan parameter prioritas terbesar. Penentuan parameter prioritas akan dijelaskan di poin berikutnya.

#### 2. Spesifikasi dan Fitur Studi Kasus

##### 2.1. Spesifikasi Studi Kasus

Waktu pelayanan pada aplikasi ini pun beragam, tergantung dari kategori penyakit hewan tersebut.

1. Ringan : 15 menit
2. Sedang : 30 menit
3. Berat : 45 menit

**Tabel Parameter Prioritas**

| NO | Parameter Prioritas                | Nilai Prioritas | Deskripsi   |
|----|------------------------------------|-----------------|---|
| 1  | Memiliki salah satu penyakit berat | 4               | Kondisi dimana jika Customer yang membawa kucing dengan salah satu penyakit berat maka akan didahulukan namun hal tersebut tergantung waktu kedatangannya. Memiliki nilai |

|   |  |   |   |
|---|--|---|---|
|   |  |   | prioritas paling tinggi yaitu 4 dibanding dengan penyakit lainnya   |
| 2 | Memiliki minimal 2 buah penyakit penyakit sedang | 3 | Kondisi dimana jika Customer membawa kucing dengan kategori penyakit sedang minimal 2 buah penyakit maka akan didahulukan sesudah penyakit berat dan memiliki poin prioritas dengan nilai 3 |
| 3 | Memiliki minimal 3 buah penyakit ringan          | 2 | Kondisi dimana jika Customer membawa kucing dengan kategori penyakit ringan minimal 3 buah penyakit   |
| 4 | Waktu kedatangan                                 | 1 | Waktu kedatangan memiliki nilai prioritas paling rendah yaitu 1   |

## 2.2. Spesifikasi Fitur Pada program

### 2.2.1. Menu

Menu merupakan tahapan pertama dari program yang kami buat. Menu adalah daftar perintah atau pilihan yang ditampilkan pada layar monitor pertama kali. Menu pada program yang kami buat berisi 6 pilihan. Customer diharuskan memilih 1 pilihan yang terdapat di menu dengan menginput angka sesuai nomor menu yang akan dipilih.

- Jika customer menginputkan angka 1, maka customer akan diarahkan ke fitur menambah pasien yang akan dijelaskan pada subbab 2.2.2.
- Jika customer menginputkan angka 2, maka customer akan diarahkan ke fitur tampil pendaftar yang akan dijelaskan pada subbab 2.2.3.
- Jika customer menginputkan angka 3, maka customer akan diarahkan ke fitur memanggil pendaftar yang akan dijelaskan pada subbab 2.2.4.

- Jika customer menginputkan angka 4, maka customer akan diarahkan ke fitur help yang akan dijelaskan pada subbab 2.2.5
- Jika customer menginputkan angka 5, maka customer akan diarahkan ke fitur credits yang akan dijelaskan pada subbab 1.2.6
- Jika customer menginputkan angka 6, maka customer akan keluar dari aplikasi.

#### 2.2.2. Menambahkan Pasien

Fitur Menambahkan pasien berfungsi untuk mendaftarkan data pasien ke antrian. Pada Fitur ini Customer diminta menginputkan nama, waktu kedatangan, dan penyakit pasien .

#### 2.2.3. Menampilkan Antrian

Fitur Menampilkan Antrian berfungsi untuk menampilkan data antrian pasien yang sudah terdaftar dalam bentuk tabel atau list.

#### 2.2.4. Memanggil Antrian

Fitur Memanggil Antrian berfungsi untuk memanggil data antrian pertama dan mengeluarkannya dari antrian. Fitur ini akan menampilkan informasi nama pasien dan penyakitnya.

#### 2.2.5. Help

Fitur Help adalah tampilan yang akan berisikan pilihan aturan prioritas dan daftar penyakit. Help bertujuan untuk membantu customer dalam menyelesaikan masalah yang dihadapi pada saat menggunakan aplikasi ini.

#### 2.2.6. Credit

Fitur ini berfungsi untuk menampilkan credit .Credit adalah tampilan yang akan menampilkan nama *Developer* aplikasi, program studi, dan jurusan.

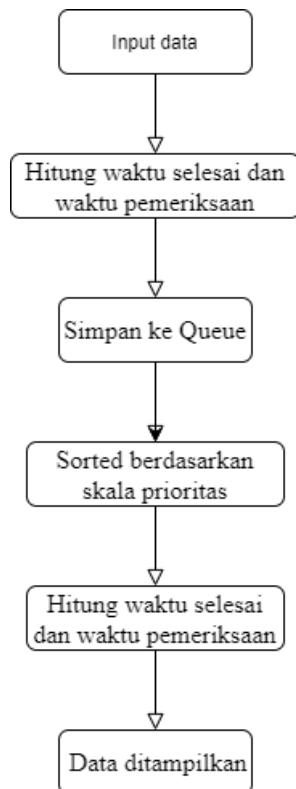
## Bab II

### Design atau Perancangan

#### 3. Perancangan Program

Desain atau perancangan adalah penggambaran, perencanaan, dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah dari satu kesatuan yang utuh mengenai aplikasi QueenPet yang kami buat. Desain atau perancangan ini memuat tentang visualisasi penyelesaian solusi permasalahan, daftar ADT yang digunakan, atribut data (sebagai header ADT), modul behaviour data, rancangan user interface, dan perancangan algoritma pada setiap modul yang akan dijelaskan pada subbab-subbab selanjutnya.

##### 3.1. Visualisasi penyelesaian solusi permasalahan



##### 3.2. Daftar ADT yang digunakan

ADT yang akan kami gunakan adalah ADT Queue disebabkan karena aplikasi ini berupa sebuah antrian. Queue adalah struktur data abstrak yang terbuka di kedua ujungnya. Satu ujung digunakan untuk memasukkan data(enqueue) dan ujung yang lainnya digunakan untuk menghapus data(dequeue). Queue mengikuti metode First-In-First-Out, yaitu item data yang disimpan terlebih dahulu akan diakses terlebih dahulu.

### **3.3. Atribut Data (sebagai header ADT)**

#### **3.3.1. Data Pasien**

Data pasien berupa nama, waktu kedatangan, nama penyakit, kategori penyakit, waktu pelayanan, waktu mulai dilayani, waktu estimasi selesai, dan prioritas. Data pasien akan dicatat dalam struct data. Struct data terdiri dari delapan field, yakni:

- Array satu dimensi bertipe char dengan ukuran 255 yang digunakan untuk menampung nama pasien. Nama pasien berperan sebagai identitas pasien.
- Variabel waktuKedatangan yang bertipe integer digunakan untuk menampung waktu kedatangan pasien.
- Array dua dimensi bertipe char dengan ukuran 10 dan 255 yang digunakan untuk menampung daftar penyakit pasien.
- Array satu dimensi bertipe char dengan ukuran 10 yang digunakan untuk menampung kategori penyakit.
- Variabel WaktuPelayanan bertipe integer yang digunakan untuk menampung waktu pelayanan.
- Variabel WaktuMulai bertipe integer yang digunakan untuk menampung waktu mulai dilayani.
- Variabel WaktuSelesai bertipe integer yang digunakan untuk menampung waktu estimasi selesai dilayani.
- Variabel Prioritas bertipe integer yang digunakan untuk menampung parameter prioritas.

```
typedef struct{
    int waktuKedatangan;
    char nama[255]; //Nama majikan
    char dataPenyakit[10][255]; //Daftar Nama Penyakit
    char kategoriPenyakit[10]; //Ringen || Sedang || Berat
    int WaktuPelayanan; //Waktu estimasi Pelayanan
    int WaktuMulai; //Waktu estimasi mulai Pemeriksaan
    int WaktuSelesai; //Waktu selesai Pemeriksaan
    int prioritas;
} data;
```

#### **3.3.2. Struktur Data Queue**

Struktur Data Queue digunakan untuk menyambungkan data pasien dan menghubungkannya seperti sebuah antrian. Struktur Data Queue terdiri dari dua struct, yakni struct Queue dan struct NodeQueue. Dalam Struct Queue terdapat field Front dan Rear bertipe addrNQ yang bertujuan untuk menunjuk awal dan akhir antrian pasien. Sedangkan, pada Struct NodeQueue terdapat field info bertipe infotype yang berfungsi menyimpan data

pasien dan field next bertipe addrNQ yang berfungsi menunjuk struktur node di antrian selanjutnya.

```
/* Definisi elemen dan address */
typedef data infotypeQueue; // Data
typedef struct NodeQueue *addrNQ; // sebuah pointer
yang menunjuk ke struct NodeQueue
typedef struct NodeQueue {
    infotypeQueue info; // address NodeQueue
    addrNQ next;
} NodeQueue;
typedef struct {
    addrNQ Front; // tag antrian depan
    addrNQ Rear; // tag antrian belakang
} Queue;
```

### 3.4. Modul behavior data (sebagai header ADT)

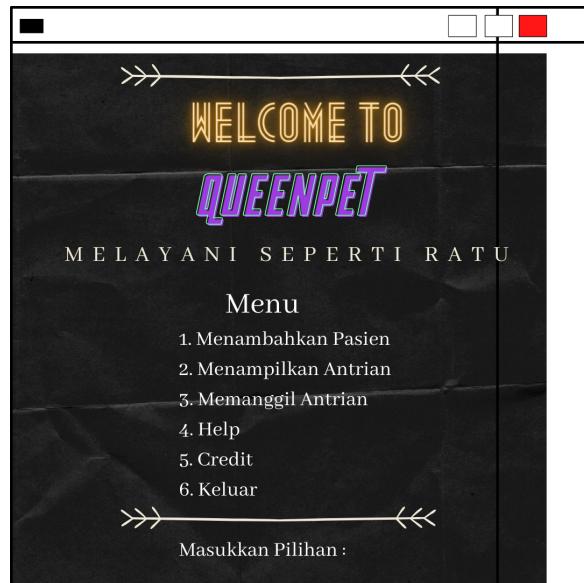
| NO | Nama Modul       | Jenis Modul | Deskripsi   | Penanggung Jawab |
|----|------------------|-------------|---|------------------|
| 1  | Main             | Function    | Modul untuk menjalankan program                                     | Hasanah          |
| 2  | Tambah Pendaftar | Procedure   | Modul untuk menambahkan data pendaftar                              | Dimas W S        |
| 3  | Create Queue     | Procedure   | Modul untuk membuat queue kosong                                    | Dimas W S        |
| 4  | Sort             | Procedure   | Modul untuk mengurutkan queue berdasarkan prioritas                 | Dimas W S        |
| 5  | Tampil Pendaftar | Procedure   | Modul untuk menampilkan data pendaftar                              | Hasanah          |
| 6  | Set              | Procedure   | Modul ini berfungsi untuk menghitung waktu mulai pemeriksaan hingga | Dimas W S        |

|    |                        |           |   |           |
|----|------------------------|-----------|---|-----------|
|    |                        |           | waktu selesai pasien  |           |
| 7  | Panggil Pendaftar      | Procedure | Modul untuk memanggil data pendaftar  | Anna K    |
| 8  | Help                   | Procedure | Modul untuk menampilkan data help   | Anna K    |
| 9  | Credit                 | Procedure | Modul untuk menampilkan data Credit   | Hasanah   |
| 10 | Hitung Waktu Pelayanan | Function  | Modul untuk menghitung waktu pelayanan yang sudah di set berdasarkan kategori penyakit                                | Dimas W S |
| 11 | HitungPrioritas        | Function  | Modul ini berfungsi untuk memprioritaskan data pasien sesuai kategori penyakit yaitu berat sedang dan ringan          | Hasanah   |
| 12 | PrintQueue             | Procedure | Modul ini berfungsi untuk menampilkan keseluruhan data pendaftar berupa tabel (nama pasien,waktu datang,kategori dll) | Anna K    |
| 13 | Header                 | Procedure | Modul untuk menampilkan tampilan di main menu   | Anna K    |
| 14 | PrintFormat            | Procedure | Modul ini berfungsi untuk menampilkan keseluruhan data pendaftar berupa Format(nama pasien,waktu datang,kategori dll) | Hasanah   |
| 15 | PrintPenyakit          | Procedure | Modul untuk menampilkan penyakit pasien   | Anna K    |

### **3.5. Rancangan user interface**

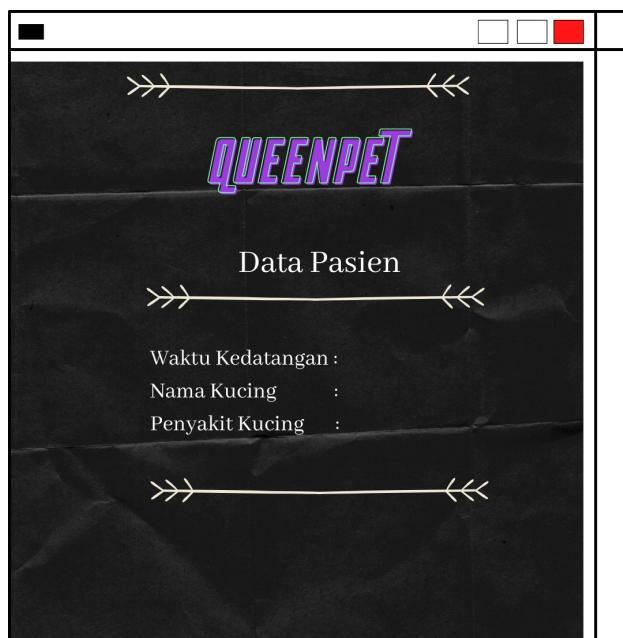
#### **3.5.1. Main Menu**

- Rancangan Tampilan awal pada Main Menu

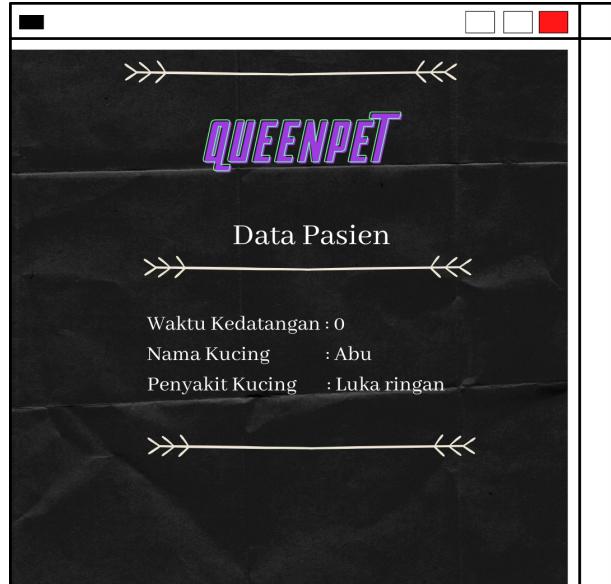


#### **3.5.2. Menambahkan Pasien**

- Tampilan awal

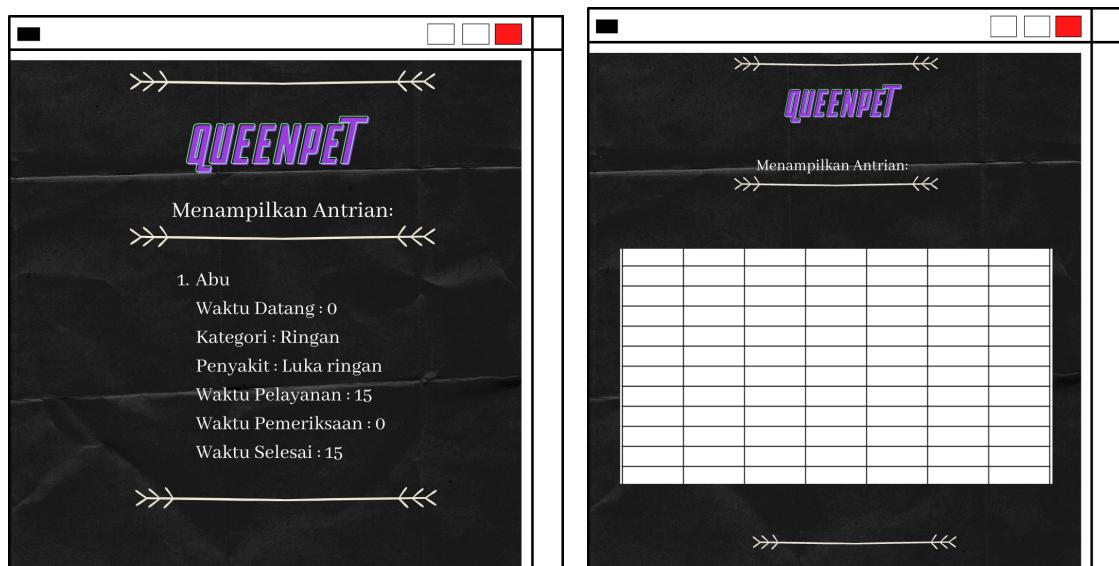


- Tampilan setelah diisi



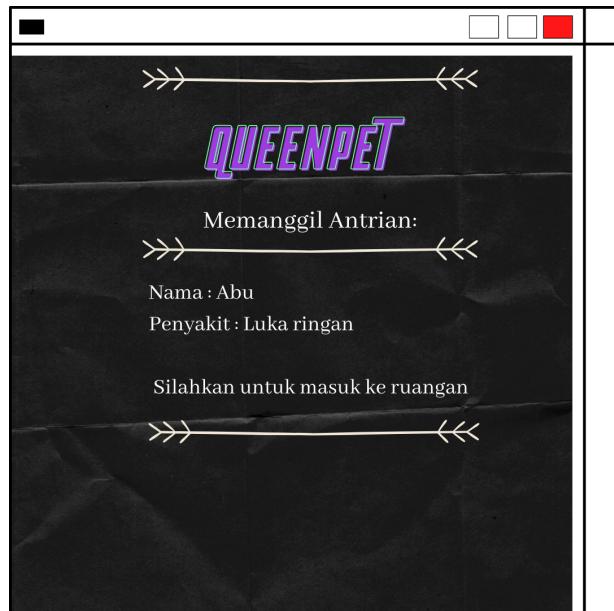
### **3.5.3. Menampilkan Antrian**

- Rancangan tampilan awal untuk menampilkan antrian



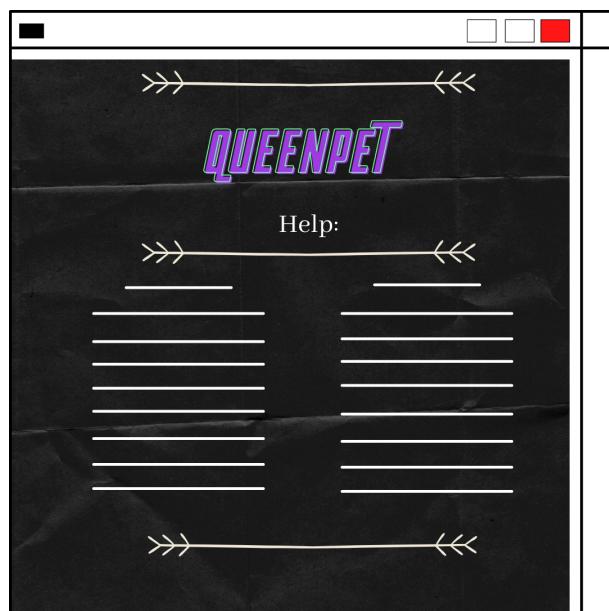
### **3.5.4. Memanggil Antrian**

- Rancangan tampilan awal untuk memanggil antrian



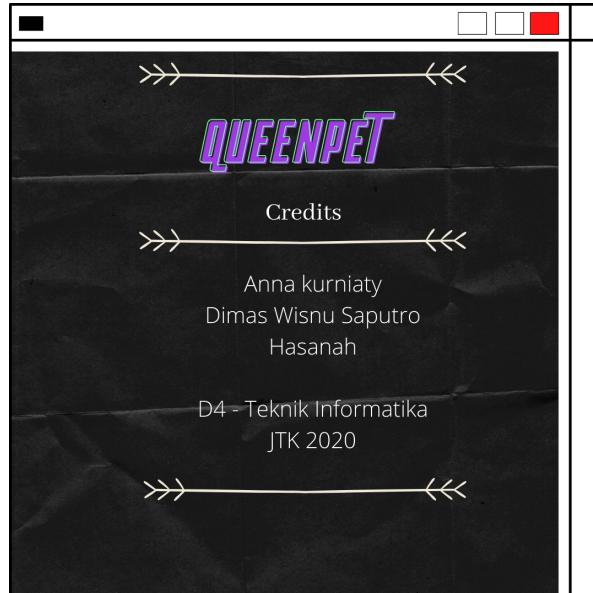
### **3.5.5. Help**

- Rancangan tampilan awal untuk memanggil antrian



### 3.5.6. Credit

- Rancangan tampilan awal untuk Credit



## 3.6. Rancangan algoritma pada setiap modul

### 3.6.1. Function Main

|                             |  |
|-----------------------------|--|
| Deskripsi                   | Modul untuk menjalankan program  |
| Parameter Input             | -  |
| Parameter Output / Keluaran | -  |
| Pseudocode                  | <pre>Function Main( ) → integer //Kamus Data choice : char myQueue : Queue  //Algoritma Begin CreateQueue(&amp;myQueue) Repeat     system("cls");     write(layar)"Program Layanan Dokter Hewan\n"     write(layar)"1. Tambah Pendaftar\n"     write(layar)"2. Tampilkan Pendaftar\n"     write(layar)"3. Panggil Pendaftar\n"</pre> |

```

        write(layar)"4. Help\n"
        write(layar)"5. Credit\n"
        write(layar)"6. Keluar\n"
        write(layar)"Masukkan Pilihan: "
        read(keyboard) choice
        if(choice ='1'){
            tambahPendaftar(&myQueue);
        }else if(choice ='2'){
            sort(&myQueue);
            tampilPendaftar(myQueue);
        }else if(choice ='3'){
            panggilPendaftar(&myQueue);
        }else if(choice ='4'){
            help();
        }else if(choice ='5'){
            credit();
        }
    until(choice = '6')
    return 0
End Function

```

### 3.6.2. Procedure tambahPendaftar

|                             |  |
|-----------------------------|--|
| Deskripsi                   | Modul untuk menambahkan data pendaftar ke Queue  |
| Parameter Input             | Q : Queue  |
| Parameter Output / Keluaran | Q : Queue  |
| Pseudocode                  | <pre> Procedure tambahPendaftar(input/output Q : Queue) //Kamus customer : data p : pointer of NodeQueue i : integer temp : char  //Algoritma Begin     p ← Q^.Rear     //Input data antrian     write(layar) "Nama      : "     read(keyboard) customer.nama     write(layar) "Waktu Datang : "     read(keyboard) customer.waktuKedatangan      For ( I ← 1 to 10 ) Do         write(layar) "Nama Penyakit %d : "         read(keyboard) customer.dataPenyakit[i] </pre> |

|  |  |
|--|--|
|  | <pre> write(layar)"Apakah ada penyakit lain? (y/n): " read(keyboard) temp if(temp='n'  temp='N')     then i ← i + 10;  Endfor  //Proses Menghitung Waktu Pelayanan, Waktu Mulai, dan Estimasi Selesai customer.WaktuPelayanan ← HitungWaktuPelayanan(customer.dataPenyakit) if(p=nil)     customer.WaktuSelesai ← customer.waktuKedatangan + customer.WaktuPelayanan customer.WaktuMulai ← 0;  if(p!=nil)     if(p-&gt;info.WaktuSelesai &gt; customer.waktuKedatangan)         then customer.WaktuMulai ← p-&gt;info.WaktuSelesai         customer.WaktuSelesai ← p-&gt;info.WaktuSelesai + customer.WaktuPelayanan     else customer.WaktuMulai ← customer.waktuKedatangan         customer.WaktuSelesai ← customer.waktuKedatangan + customer.WaktuPelayanan  //Memasukkan ke antrian enQueue(Q,customer)  write (layar) "Press any key to continue.. " getch()  End Procedure </pre> |
|--|--|

### 3.6.3. Procedure CreateQueue()

|                             |  |
|-----------------------------|--|
| Deskripsi                   | Modul untuk membuat Queue kosong   |
| Parameter Input             | Q : Queue  |
| Parameter Output / Keluaran | Q : Queue  |
| Pseudocode                  | <pre> Procedure CreateQueue(input/output Q : Queue) //Algoritma Begin     Q^.Front ← NULL </pre> |

|  |  |
|--|--|
|  | $Q^{\wedge}.\text{Rear} \leftarrow \text{NULL}$<br>End Procedure |
|--|--|

### 3.6.4. Procedure sort()

|                             |   |
|-----------------------------|---|
| Deskripsi                   | Modul untuk mengurutkan Queue berdasarkan Prioritas   |
| Parameter Input             | Q : Queue   |
| Parameter Output / Keluaran | Q : Queue   |
| Pseudocode                  | <pre> Procedure sort(input/output Q : Queue)   //Kamus Data   NodeQueue *p, *buff;   p : pointer of NodeQueue   buff : pointer of NodeQueue   temp : data   i : integer    //Algoritma   Begin     p <math>\leftarrow</math> Q<math>^{\wedge}.\text{Front}</math>     if(p=NULL    p<math>^{\wedge}.\text{next}=NULL)</math>       Then return      p = p<math>^{\wedge}.\text{next}</math>;     while(p<math>^{\wedge}.\text{next}!=NULL</math>) do       buff = p<math>^{\wedge}.\text{next}</math>       while(buff!=NULL)do         if(p<math>^{\wedge}.\text{info}.\text{WaktuSelesai} &gt;</math>           buff<math>^{\wedge}.\text{info}.\text{waktuKedatangan}</math>)           then if(p<math>^{\wedge}.\text{info}.\text{prioritas} &lt; buff^{\wedge}.\text{info}.\text{prioritas}</math>)             then temp <math>\leftarrow</math> p<math>^{\wedge}.\text{info}</math>               p<math>^{\wedge}.\text{info} \leftarrow buff^{\wedge}.\text{info}</math>               buff<math>^{\wedge}.\text{info} \leftarrow temp</math>             else if(p<math>^{\wedge}.\text{info}.\text{prioritas} =</math>               buff<math>^{\wedge}.\text{info}.\text{prioritas}</math>)               then temp <math>\leftarrow</math> p<math>^{\wedge}.\text{info}</math>               p<math>^{\wedge}.\text{info} \leftarrow buff^{\wedge}.\text{info}</math>               buff<math>^{\wedge}.\text{info} \leftarrow temp</math>             end if           end if           buff <math>\leftarrow</math> buff<math>^{\wedge}.\text{next}</math>         end while       p <math>\leftarrow</math> p<math>^{\wedge}.\text{next}</math>     end while   end Procedure </pre> |

### 3.6.5. Procedure TampilPendaftar

|                             |  |
|-----------------------------|--|
| Deskripsi                   | Modul untuk menampilkan pendaftar  |
| Parameter Input             | Q : Queue  |
| Parameter Output / Keluaran | Q : Queue  |
| Pseudocode                  | <pre> Procedure tampilPendaftar(input/output Q : Queue)   \\Kamus Data   choice : char    \\Algoritma   Begin     Do       write(layar)" ~ ~ ~ QUEENPET~ ~ ~ "       write(layar)"\n\n~Tampil Pendaftar~"       write(layar)"1. Bentuk Tabel"       write(layar)"2. Bentuk List"       write(layar)"3. Kembali ke Main Menu"       write(layar)"Masukkan Pilihan: "       read(keyboard) "%c", &amp;choice       if(choice='1') then         PrintQueue(Q)       else if(choice='2') then         PrintFormat(Q)       while(choice!='3')       end while     end Procedure   </pre> |

### 3.6.6. Procedure set

|                             |  |
|-----------------------------|--|
| Deskripsi                   | Set Waktu mulai Pemeriksaan dan Waktu Selesai data pasien  |
| Parameter Input             |  |
| Parameter Output / Keluaran |  |
| Pseudocode                  | <pre> Procedure set(input/output Q : Queue, checkpoints : int)   //Kamus Data   p : pointer of NodeQueue   buff : pointer of NodeQueue   temp : data    //Algoritma   Begin     p ← Q-&gt;Front;     if(p==NULL    p-&gt;next=NULL)then       return     end if   </pre> |

```

//Set Waktu mulai Pemeriksaan dan Waktu Selesai;

if(checkpoints=0)then
    p->info.WaktuMulai ←
    p->info.waktuKedatangan
    p->info.WaktuSelesai ←
    p->info.waktuKedatangan + p->info.WaktuPelayanan
    end if
    else
        if(checkpoints >
p->info.waktuKedatangan)the
            p->info.WaktuMulai ← checkpoints
            p->info.WaktuSelesai ←
            checkpoints + p->info.WaktuPelayanan
            else
                p->info.WaktuMulai ←
                p->info.waktuKedatangan
                p->info.WaktuSelesai ←
                p->info.waktuKedatangan + p->info.WaktuPelayanan

                buff ← p
                p ← p->next
                while(p!=NULL)
                    if(buff->info.WaktuSelesai >
p->info.waktuKedatangan)then
                        p->info.WaktuMulai ←
                        buff->info.WaktuSelesai
                        p->info.WaktuSelesai ←
                        buff->info.WaktuSelesai + p->info.WaktuPelayanan
                        end if
                        else
                            p->info.WaktuMulai ←
                            p->info.waktuKedatangan
                            p->info.WaktuSelesai ←
                            p->info.waktuKedatangan + p->info.WaktuPelayanan

                            buff ← buff->next
                            p ← p->next
                        end while
                    end Procedure

```

### 3.6.7. Procedure PanggilPendaftar

|                             |                                    |
|-----------------------------|------------------------------------|
| Deskripsi                   | Modul untuk memanggil pendaftar    |
| Parameter Input             | myQueue : Queue, checkpoints : int |
| Parameter Output / Keluaran | myQueue : Queue, checkpoints : int |

|            |   |
|------------|---|
| Pseudocode | <pre> Procedure panggilPendaftar(input/output myQueue : Queue, checkpoints : int)     //Kamus Data     temp : data      //Algoritma     Begin         temp.prioritas ← 6         deQueue(myQueue, &amp;temp)         if(temp.prioritas&gt;5) then             write(layar)"Pendaftar masih kosong!"         end if         else             write(layar)"~~Memanggil Antrian~~"             write(layar)"Nama : %s\n", temp.nama             printPenyakit(temp)             *checkpoints ← temp.WaktuSelesai             write(layar)"Press any key to continue.. "             getch()     end Procedure </pre> |
|------------|---|

### 3.6.8. Procedure help

|                             |   |
|-----------------------------|---|
| Deskripsi                   | Modul untuk menampilkan help  |
| Parameter Input             | -   |
| Parameter Output / Keluaran | -   |
| Pseudocode                  | <pre> Procedure help()     //Kamus Data     pilih : char     a : char      //Algoritma     Begin         pilih ← 0         a ← 0         do             write(layar)   ~QUEENPET~                write(layar)" Help "             write(layar)"&gt;&gt;&gt;-----&lt;&lt;&lt;\n\n"             write(layar)"1. Aturan Prioritas \n"             write(layar)"2. Daftar Penyakit \n"             write(layar)"3. Exit \n"             write(layar)"&gt;&gt;&gt;-----&lt;&lt;&lt;\n\n"             write(layar)"Enter Menu : "             read(keyboard)"%s",&amp;pilih             if( pilih = '1' ) then </pre> |

```

        write(layar)"|||~QUEENPET~|||"
        write(layar)"|Help - Aturan
Prioritas|"

write(layar)">>>-----<<<\n\n"
prioritas \n"
\n"
write(layar)"ditentukan dengan nilai
prioritas \n"
write(layar)"sebagai berikut :\n"
write(layar)"1. Memiliki salah
satu\n"
write(layar)"penyakit berat - 4 \n"
write(layar)"2. Memiliki minimal 2
buah\n"
write(layar)"penyakit sedang - 3\n"
write(layar)"3. Memiliki minimal 3
buah\n"
write(layar)"penyakit ringan - 2 \n"
write(layar)"4. Waktu kedatangan -
1\n"
write(layar)"type anything to return
:"
read(keyboard) "%s", &a

write(layar)">>>-----<<<\n\n"
end if
if ( pilih = '2' ) then
write(layar)"|||~QUEENPET~|||"
write(layar)"~|Help - Daftar
Penyakit|~"

write(layar)">>>-----<<<\n\n"
kulit,\n"
write(layar)"luka ringan, bersin\n"
diare, \n"
write(layar)"luka dalam\n"
yang \n"
write(layar)"3. Berat : gangguan \n"
write(layar)"kerongkongan yang \n"
write(layar)"mengeluarkan lendir
yang \n"
write(layar)"berbau busuk, kuning,
\n"
write(layar)"terkena virus\n"

write(layar)">>>-----<<<\n\n"
write(layar)"type anything to return
:"
read(keyboard) "%s", &a

```

|  |   |
|--|---|
|  | <pre>         end if         while(pilih != '3')             end while         end Procedure     </pre> |
|--|---|

### 3.6.9. Procedure Credit

|                             |   |
|-----------------------------|---|
| Deskripsi                   | Modul untuk menampilkan Credit  |
| Parameter Input             | -   |
| Parameter Output / Keluaran | -   |
| Pseudocode                  | <pre> Procedure Credit() //Kamus Data     kembali : char  //Algoritma Begin     kembali ← 0     do         system("cls")         write(layar)"   ~QUEENPET~   "         write(layar)"~ Credits ~"         write(layar)"&gt;&gt;&gt;-----&lt;&lt;&lt;"          write(layar)" Anna Kurniaty"         write(layar)"Dimas Wisnu Saputro \n"         write(layar)"Hasanah"         write(layar)"D4 - Teknik Informatika"         write(layar)"JTK 2020"         write(layar)"&gt;&gt;&gt;-----&lt;&lt;&lt;"          write(layar)"Type 1 if you want to return to the menu : "         read(keyboard) kembali         while(kembali !='1')     End Procedure     </pre> |

### 3.6.10. Function HitungWaktuPelayanan

|                             |   |
|-----------------------------|---|
| Deskripsi                   | Menghitung waktu pelayanan              |
| Parameter Input             | temp : array of char [0...9] [0....254] |
| Parameter Output / Keluaran | -                                       |

|            |  |
|------------|--|
| Pseudocode | <pre> Function HitungWaktuPelayanan(input temp : array of char [0...9] [0....254]) →     //Kamus     i : integer      //Algoritma     for(i ← 0 to 10) do         if(strcmp(temp[i],"gangguan kerongkongan")==0){             return 45;         }else if(strcmp(temp[i],"kuning")==0){             return 45;         }else if(strcmp(temp[i],"terkena virus")==0){             return 45;         }     end for     for(i ← 0 to 10) do         if(strcmp(temp[i],"cacingan")==0){             return 30;         }else if(strcmp(temp[i],"diare")==0){             return 30;         }else if(strcmp(temp[i],"luka dalam")==0){             return 30;         }     end for      return 15; End Function </pre> |
|------------|--|

### 3.6.11. Function HitungPrioritas

|                             |  |
|-----------------------------|--|
| Deskripsi                   | Menghitung parameter prioritas pasien  |
| Parameter Input             | temp : array of char [0...9] [0....254]  |
| Parameter Output / Keluaran | -  |
| Pseudocode                  | <pre> Function HitungPrioritas (input temp : array of char [0...9] [0....254]) →     //Kamus     i : integer     tempSedang : integer     tempRingan : integer      //Algoritma Begin     tempSedang ← 0     tempRingan ← 0 </pre> |

```

for(i ← 0 to 10) Do
    if(strcmp(temp[i],"gangguan
kerongkongan")==0){
        return 4;
    } else if(strcmp(temp[i],"kuning")==0){
        return 4;
    } else if(strcmp(temp[i],"terkena
virus")==0){
        return 4;
    } else if(strcmp(temp[i],"cacingan")==0){
        tempSedang++;
    } else if(strcmp(temp[i],"diare")==0){
        tempSedang++;
    } else if(strcmp(temp[i],"luka dalam")==0){
        tempSedang++;
    } else if(strcmp(temp[i],"penyakit
kulit")==0){
        tempRingan++;
    } else if(strcmp(temp[i],"luka ringan")==0){
        tempRingan++;
    } else if(strcmp(temp[i],"bersin")==0){
        tempRingan++;
    }
}
End For

if(tempSedang>=2){
    return 3;
} else if(tempRingan>=3){
    return 2;
}
return 1;
End Function

```

### 3.6.12. Procedure PrintQueue

|                             |  |
|-----------------------------|--|
| Deskripsi                   | Modul untuk menampilkan keseluruhan data berbentuk tabel   |
| Parameter Input             | data : Queue   |
| Parameter Output / Keluaran | -  |
| Pseudocode                  | <pre> Procedure PrintQueue(input data : Queue)     //Kamus     p : pointer of NodeQueue     i : int     //Algoritma     Begin         i ← 1 </pre> |

```

        write(layar)"Data Pendaftar Layanan Dokter Hewan
- Queue"

write(layar)=====
=====
");

write(layar)"No Nama Waktu Datang Kategori
Penyakit Wpelayanan Wpemerikasaan Wselesai Prioritas";
if(!IsQueueEmpty(data))then
    p ← data.Front
    while (p != nil)
        write(layar)"%d. %s %d %s %s %d
%d %d %d\n", i, p->info.nama, p->info.waktuKedatangan,
p->info.kategoriPenyakit, p->info.dataPenyakit[0],
p->info.WaktuPelayanan, p->info.WaktuMulai,
p->info.WaktuSelesai, p->info.prioritas
        i++
        p ← next(p)
    end while
end if
if(i=1) then
    write(layar)"[ Antrian masih kosong ]\n"
end if
write(layar)"Press any key to continue.. "
getch()
end Procedure

```

### 3.6.13. Procedure Header

|                             |   |
|-----------------------------|---|
| Deskripsi                   | Modul untuk menampilkan Tampilan di menu  |
| Parameter Input             | -   |
| Parameter Output / Keluaran | -   |
| Pseudocode                  | <pre> Procedure header() //Algoritma Begin  write(layar)_ write(layar)"Selamat Datang pada Layanan Dokter Hewan" write(layar)"       QUEENPET       " write(layar)" _____ " }  end procedure </pre> |

### 3.6.14. Procedure PrintFormat

|                             |   |
|-----------------------------|---|
| Deskripsi                   | Modul untuk menampilkan keseluruhan data berbentuk list   |
| Parameter Input             | data : Queue  |
| Parameter Output / Keluaran | -   |
| Pseudocode                  | <pre> Procedure PrintFormat(input data : Queue)     //Kamus     p : pointer of NodeQueue     i : integer      //Algoritma'     i ← 1     write(layar)"Data Pendaftar Layanan Dokter Hewan - Queue"      write(layar)=====     if(!IsQueueEmpty(data)) then         p ← data.Front         while (p != nil) do             write(layar) i, p-&gt;info.nama             write(layar)"Waktu Datang : ",             p-&gt;info.waktuKedatangan             write(layar)"Kategori : ",             p-&gt;info.kategoriPenyakit             printPenyakit(p-&gt;info)             write(layar)"Waktu Pelayanan : ",             p-&gt;info.WaktuPelayanan             write(layar)"Waktu Mulai Pemeriksaan : ", p-&gt;info.WaktuMulai             write(layar)"Waktu Selesai : ",             p-&gt;info.WaktuSelesai             write(layar)"Prioritas : ",             p-&gt;info.prioritas             i ← i + 1             p ← next(p)         end while     end if     if(i=1) then         write(layar)"[ Antrian masih kosong ]"     end if     write(layar)"Press any key to continue.. " End Procedure </pre> |

### 3.6.15. Procedure PrintPenyakit

|                             |   |
|-----------------------------|---|
| Deskripsi                   | Modul untuk menampilkan penyakit seekor pasien  |
| Parameter Input             | buff : data   |
| Parameter Output / Keluaran | -   |
| Pseudocode                  | <pre> Procedure printPenyakit(input buff : data)     //Kamus     i : integer      //Algoritma Begin     write(layar) "\t\t\t\tNama Penyakit : "     for(i&lt;--0 to 10)do         write(layar) buff.dataPenyakit[i]         if(buff.dataPenyakit="Kosong")=0             then break         end if         write(layar)", "     }     write(layar)"\n" End Procedure </pre> |

### 3.7. Implementasi

#### 3.7.1. Tampilan Menu



### 3.7.2. Menambahkan Pasien

```
|~|~|~|~QUEENPET~|~|~|~|  
~|Masukan Data Antrian|~  
>>>-----<<<  
      Nama          : Abu  
      Waktu Datang   : 0  
      Nama Penyakit 1 : Bersin  
  
      Apakah ada penyakit lain? (y/n): n  
  
>>>-----<<<  
  
      Press any key to continue.. -
```

### 3.7.3. Memanggil Antrian

```
|~|~|~|~QUEENPET~|~|~|~|  
~|Tampil Pendaftar|~  
>>>-----<<<  
      1. Bentuk Tabel  
      2. Bentuk List  
      3. Kembali ke Main Menu  
  
>>>-----<<<  
      Masukkan Pilihan:
```

#### 3.7.3.1. Tabel

```
|~|~|~|~QUEENPET~|~|~|~|  
~|Tampil Pendaftar|~  
>>>-----<<<  
      1. Bentuk Tabel  
      2. Bentuk List  
      3. Kembali ke Main Menu  
  
>>>-----<<<  
      Masukkan Pilihan: 1  
  
      Data Pendaftar Layanan Dokter Hewan - Queue  
=====  
No Nama      Waktu Datang  Kategori  Penyakit  Wpelayanan Wpemeriksaan Wselesai Prioritas  
1. Merah Api    21        Berat     kuning     45           21       66       4  
2. Putih Mawar   4         Sedang     diare      30           66       96       1  
3. Hitam Manis  20        Ringan    bersin     15           96       111      1  
4. Hijau Daun    30        Sedang     diare      30           111      141      1  
  
      Press any key to continue.. -
```

### 3.7.3.2. List

```
Masukkan Pilihan: 2
Data Pendaftar Layanan Dokter Hewan - Queue
=====
1. abu
Waktu Datang : 0
Kategori : Ringan
Nama Penyakit : bersin
Waktu Pelayanan : 15
Waktu Mulai Pemeriksaan : 0
Waktu Selesai : 15
Prioritas : 1

2. hijau
Waktu Datang : 4
Kategori : Berat
Nama Penyakit : gangguan kerongkongan
Waktu Pelayanan : 45
Waktu Mulai Pemeriksaan : 15
Waktu Selesai : 60
Prioritas : 4

3. putih
Waktu Datang : 25
Kategori : Berat
Nama Penyakit : kuning, bersin
Waktu Pelayanan : 45
Waktu Mulai Pemeriksaan : 60
Waktu Selesai : 105
Prioritas : 4

4. hitam
Waktu Datang : 21
Kategori : Sedang
Nama Penyakit : diare
Waktu Pelayanan : 30
Waktu Mulai Pemeriksaan : 105
Waktu Selesai : 135
Prioritas : 1

Press any key to continue..
```

### 3.7.4. Help

```
||||~QUEENPET~||||
|Help|
>>>-----<<<
1. Aturan Prioritas
2. Daftar Penyakit
3. Exit

>>>-----<<<
```

#### 3.7.4.1. Parameter Prioritas

```
||||~QUEENPET~||||
|Help - Aturan Prioritas|
>>>-----<<<
Parameter urutan prioritas
ditentukan dengan nilai
sebagai berikut :
1. Memiliki salah satu
penyakit berat - 4
2. Memiliki minimal 2 buah
penyakit sedang - 3
3. Memiliki minimal 3 buah
penyakit ringan - 2
4. Waktu kedatangan - 1

type anything to return :
```

### 3.7.4.2. Daftar Penyakit

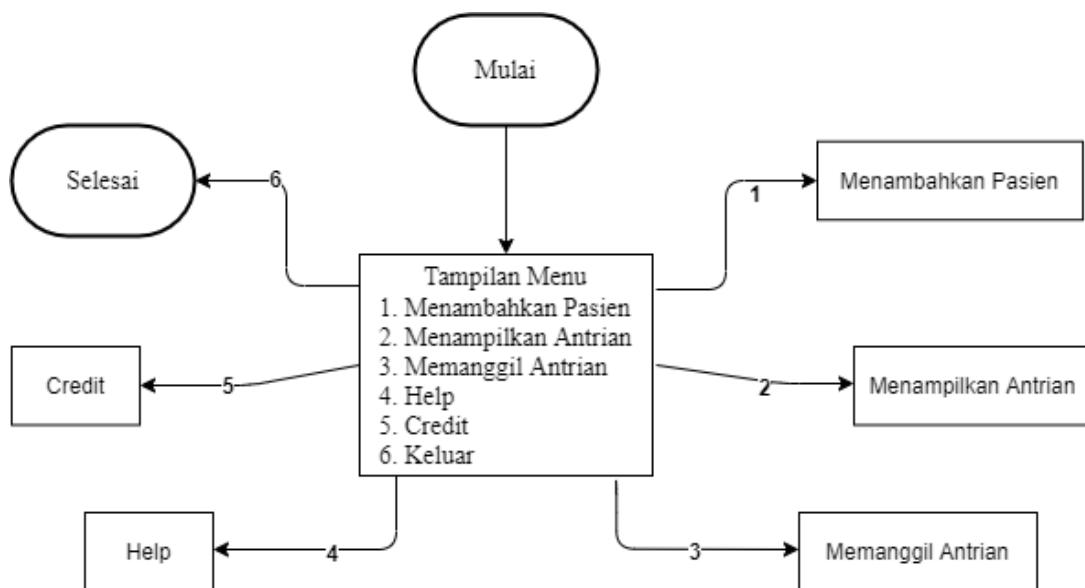
```
||||~QUEENPET~|||  
~|Help - Daftar Penyakit|~  
>>>-----<<  
1. Ringan : penyakit kulit,  
luka ringan, bersin  
2. Sedang : cacingan, diare,  
luka dalam  
3. Berat : gangguan  
kerongkongan yang  
mengeluarkan lendir yang  
berbau busuk, kuning,  
terkena virus  
>>>-----<<  
type anything to return :
```

### 3.7.5. Credit

```
||||~QUEENPET~|||  
~|Credits|~  
>>>-----<<  
Anna Kurniaty  
Dimas Wisnu Saputro  
Hasanah  
D4 - Teknik Informatika  
JTK 2020  
>>>-----<<  
Type 1 if you want to return to the menu :
```

### 3.8. Eksperimen

#### 3.8.1. Rancangan alur skenario simulasi



#### 3.8.2. Rancangan data simulasi

1. Memasukkan data pertama kucing bernama Abu

| Nama | WDatang | Kategori | Penyakit | WPelayanan | WPemeriksaan | WSelesai |
|------|---------|----------|----------|------------|--------------|----------|
| Abu  | 0       | R        | bersin   | 15         | 0            | 15       |

2. Memasukkan data kedua kucing bernama Putih

| Nama  | WDatang | Kategori | Penyakit | WPelayanan | WPemeriksaan | WSelesai |
|-------|---------|----------|----------|------------|--------------|----------|
| Abu   | 0       | R        | bersin   | 15         | 0            | 15       |
| Putih | 4       | S        | diare    | 30         | 15           | 45       |

3. Memasukkan data ketiga kucing bernama Hitam

| Nama  | WDatang | Kategori | Penyakit | WPelayanan | WPemeriksaan | WSelesai |
|-------|---------|----------|----------|------------|--------------|----------|
| Abu   | 0       | R        | bersin   | 15         | 0            | 15       |
| Putih | 4       | S        | diare    | 30         | 15           | 45       |
| Hitam | 20      | R        | bersin   | 15         | 45           | 60       |

4. Memasukkan data keempat kucing bernama Merah

| Nama  | WDatang | Kategori | Penyakit | WPelayanan | WPemeriksaan | WSelesai |
|-------|---------|----------|----------|------------|--------------|----------|
| Abu   | 0       | R        | bersin   | 15         | 0            | 15       |
| Putih | 4       | S        | diare    | 30         | 15           | 45       |
| Hitam | 20      | R        | bersin   | 15         | 45           | 60       |
| Merah | 21      | B        | kuning   | 45         | 60           | 105      |

5. Data diurutkan berdasarkan prioritas karena ada pasien yang harus di prioritaskan

| Nama  | WDatang | Kategori | Penyakit | WPelayanan | WPemeriksaan | WSelesai |
|-------|---------|----------|----------|------------|--------------|----------|
| Abu   | 0       | R        | bersin   | 15         | 0            | 15       |
| Merah | 21      | B        | kuning   | 45         | 21           | 66       |
| Putih | 4       | S        | diare    | 30         | 66           | 96       |
| Hitam | 20      | R        | bersin   | 15         | 96           | 111      |

6. Memasukkan data kelima kucing bernama Hijau

| Nama  | WDatang | Kategori | Penyakit | WPelayanan | WPemeriksaan | WSelesai |
|-------|---------|----------|----------|------------|--------------|----------|
| Abu   | 0       | R        | bersin   | 15         | 0            | 15       |
| Putih | 4       | S        | diare    | 30         | 21           | 66       |
| Merah | 21      | B        | kuning   | 45         | 66           | 96       |
| Hitam | 20      | R        | bersin   | 15         | 96           | 111      |
| Hijau | 30      | S        | diare    | 30         | 111          | 141      |

7. Pemanggilan kucing bernama Abu

| Nama  | WDatang | Kategori | Penyakit | WPelayanan | WPemeriksaan | WSelesai |
|-------|---------|----------|----------|------------|--------------|----------|
| Putih | 4       | S        | diare    | 30         | 21           | 66       |
| Merah | 21      | B        | kuning   | 45         | 66           | 96       |
| Hitam | 20      | R        | bersin   | 15         | 96           | 111      |
| Hijau | 30      | S        | diare    | 15         | 111          | 141      |

### 3.8.3. Menjalankan program sesuai dengan data simulasi

- Memasukkan data pertama kucing bernama Abu

```
|~|~|~|~QUEENPET~|~|~|~|
~|Tampil Pendaftar|~

>>>-----<<<
1. Bentuk Tabel
2. Bentuk List
3. Kembali ke Main Menu

>>>-----<<<
Masukkan Pilihan: 1

Data Pendaftar Layanan Dokter Hewan - Queue
=====
No Nama      Waktu Datang   Kategori   Penyakit   Wpelayanan   Wpemerikasaan   Wselesai   Prioritas
1. Abu Bakar    0           Ringan     bersin       15            0              15          1

Press any key to continue..
```

- Memasukkan data kedua kucing bernama Putih

```
|~|~|~|~QUEENPET~|~|~|~|
~|Tampil Pendaftar|~

>>>-----<<<
1. Bentuk Tabel
2. Bentuk List
3. Kembali ke Main Menu

>>>-----<<<
Masukkan Pilihan: 1

Data Pendaftar Layanan Dokter Hewan - Queue
=====
No Nama      Waktu Datang   Kategori   Penyakit   Wpelayanan   Wpemerikasaan   Wselesai   Prioritas
1. Abu Bakar    0           Ringan     bersin       15            0              15          1
2. Putih Mawar    4           Sedang    diare        30            15             45          1

Press any key to continue..
```

- Memasukkan data ketiga kucing bernama Hitam

```
|~|~|~|~QUEENPET~|~|~|~|
~|Tampil Pendaftar|~

>>>-----<<<
1. Bentuk Tabel
2. Bentuk List
3. Kembali ke Main Menu

>>>-----<<<
Masukkan Pilihan: 1

Data Pendaftar Layanan Dokter Hewan - Queue
=====
No Nama      Waktu Datang   Kategori   Penyakit   Wpelayanan   Wpemerikasaan   Wselesai   Prioritas
1. Abu Bakar    0           Ringan     bersin       15            0              15          1
2. Putih Mawar    4           Sedang    diare        30            15             45          1
3. Hitam Manis   20          Ringan     bersin       15            45             60          1

Press any key to continue..
```

- Memasukkan data keempat kucing bernama Merah

| Nama  | WDatang | Kategori | Penyakit | WPelayanan | WPemeriksaan | WSelesai |
|-------|---------|----------|----------|------------|--------------|----------|
| Abu   | 0       | R        | bersin   | 15         | 0            | 15       |
| Putih | 4       | S        | diare    | 30         | 15           | 45       |
| Hitam | 20      | R        | cacingan | 15         | 45           | 60       |
| Merah | 21      | B        | kuning   | 45         | 60           | 105      |

- Data diurutkan berdasarkan prioritas karena ada pasien yang harus di prioritaskan

```
|~|~|~|~QUEENPET~|~|~|~|
~|Tampil Pendaftar|~

>>>-----<<<
1. Bentuk Tabel
2. Bentuk List
3. Kembali ke Main Menu

>>>-----<<<
Masukkan Pilihan: 1

Data Pendaftar Layanan Dokter Hewan - Queue
=====
No Nama      Waktu Datang   Kategori    Penyakit   Wpelayanan   Wpemeriksaan   Wselesai   Prioritas
1. Abu Bakar     0           Ringan      bersin       15            0             15          1
2. Merah Api     21          Berat       kuning       45            21            66          4
3. Putih Mawar    4           Sedang      diare        30            66            96          1
4. Hitam Manis   20          Ringan      bersin       15            96            111         1

Press any key to continue..
```

- Memasukkan data kelima kucing bernama Hijau

```
|~|~|~|~QUEENPET~|~|~|~|
~|Tampil Pendaftar|~

>>>-----<<<
1. Bentuk Tabel
2. Bentuk List
3. Kembali ke Main Menu

>>>-----<<<
Masukkan Pilihan: 1

Data Pendaftar Layanan Dokter Hewan - Queue
=====
No Nama      Waktu Datang   Kategori    Penyakit   Wpelayanan   Wpemeriksaan   Wselesai   Prioritas
1. Abu Bakar     0           Ringan      bersin       15            0             15          1
2. Merah Api     21          Berat       kuning       45            21            66          4
3. Putih Mawar    4           Sedang      diare        30            66            96          1
4. Hitam Manis   20          Ringan      bersin       15            96            111         1
5. Hijau Daun    30          Sedang      diare        30            111           141         1

Press any key to continue..
```

- Pemanggilan kucing bernama Abu

```

Selamat Datang pada Layanan Dokter Hewan
QUEENPET

Program Layanan Dokter Hewan

1. Tambah Pendaftar
2. Tampilkan Pendaftar
3. Panggil Pendaftar
4. Help
5. Credit
6. Keluar

Masukkan Pilihan: 3
~~Memanggil Antrian~~
Nama : Abu Bakar
Nama Penyakit : bersin

Press any key to continue..

```

```

|~|~|~|~QUEENPET~|~|~|~|
~|Tampil Pendaftar|~

>>>-----<<<

1. Bentuk Tabel
2. Bentuk List
3. Kembali ke Main Menu

>>>-----<<<

Masukkan Pilihan: 1

Data Pendaftar Layanan Dokter Hewan - Queue
=====
No Nama      Waktu Datang   Kategori   Penyakit   Wpelayanan   Wpemerikasaan   Wselesai   Prioritas
1. Merah Api    21        Berat       kuning      45           21          66            4
2. Putih Mawar    4         Sedang      diare       30           66          96            1
3. Hitam Manis   20        Ringan     bersin      15           96          111           1
4. Hijau Daun    30        Sedang      diare       30           111         141           1

Press any key to continue.. -

```

## **Bab III**

### **Kesimpulan**

#### **1. Ringkasan Kontribusi Setiap Kelompok**

##### **1. Anna Kurniaty**

Pada tugas ini saya berperan dalam segala aspek contohnya saat membuat laporan saya mengerjakan 30%. Sebanyak 30% modul merupakan karya saya. Saya juga bertugas dalam membuat sketsa dan juga memperbaiki tampilan saat ada yang tidak sesuai.

##### **2. Dimas Wisnu Saputro**

Setiap anggota pada kelompok kami berkontribusi dalam segala aspek. Mulai dari ide, gagasan, tampilan, codingan hingga laporan. Oleh karena itu, saya berperan di semua aspek. Mengapa pembagian tugas kelompok kami seperti itu disebabkan agar kami siap ditempatkan dibagian mana saja suatu saat nanti.

##### **3. Hasanah**

Pada tugas ini saya berkontribusi dalam hal segala aspek contohnya membuat laporan dan aspek lainnya. Saya juga berperan membuat sketsa interface tampilan.

#### **2. Lesson Learned**

##### **1. Anna Kurniaty**

Pada tugas besar ini saya menjadi lebih memahami tentang ADT Queue, selain itu saya juga menjadi lebih mengerti perihal prioritas. Pada tugas besar ini juga saya belajar bagaimana membagi waktu untuk mengerjakan tubes ini dengan tugas yang lain. Dan juga saya belajar untuk menghargai pendapat orang lian untuk mencapai kesepakatan bersama.

##### **2. Dimas Wisnu Saputro**

Tugas besar ini mengajarkan saya banyak hal. Awal - awal saya mengalami kesulitan dalam menentukan prioritas, tetapi dengan pantang menyerah dan kesabaran permasalahan itu dapat dilalui. Dengan adanya tugas besar ini, saya bisa lebih memahami ADT Queue karena teori yang telah dipelajari langsung diimplementasikan. Selain itu, Tugas besar ini juga mengajarkan saya dalam hal bekerja sama. Menyatukan beberapa kepala agar

menjadi satu tujuan bukan hal yang mudah. Oleh karena itu, Menghargai pendapat, sabar, dan manajemen waktu adalah hal wajib yang harus dimiliki.

### **3. Hasanah**

Pada tugas besar yang kedua ini, bagi saya mengajarkan banyak hal mulai dari teori ADT Queue dan permasalahan pada kasus yang dikerjakan. Dalam hal lain dalam mengerjakan tugas besar ini mengajarkan saya memanajemen waktu, bekerja sama dalam tim, dan menghargai satu dengan lainnya.