

BAB 4

HASIL PENELITIAN

4.1 RINGKASAN HASIL PENELITIAN

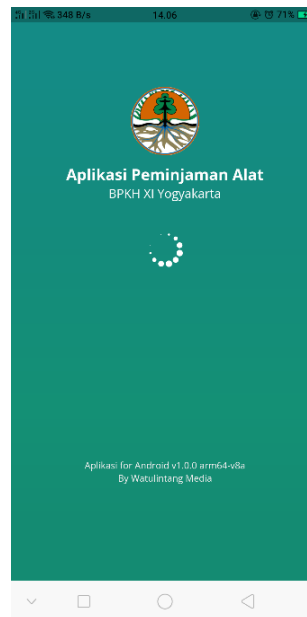
Aplikasi Peminjaman Alat BPKH XI Wilayah Yogyakarta berbasis Android ditujukan untuk proses peminjaman serta persetujuan oleh Pegawai dan Kepala Balai. Aplikasi ini juga dapat digunakan untuk kalangan di luar kantor BPKH wilayah Yogyakarta dengan syarat dan ketentuan tertentu yang telah ditentukan oleh balai.

4.2 IMPLEMENTASI DISAIN INTERFACE PEMINJAM

Tampilan Peminjam tersusun berdasarkan kode program yang dibuat yang direlasikan berdasarkan perancangan aplikasi. Dalam menyusun kode program aplikasi peminjaman alat BPKH XI Wilayah Yogyakarta berbasis Android menggunakan Visual Studio Code dan menggunakan bahasa pemrograman Dart dan *framework* Flutter yang terintegrasi menggunakan data API yang dibuat berdasarkan sistem yang sudah ada. Berikut ini merupakan implementasi tampilan dan kode yang terdapat pada aplikasi.

4.2.1 Tampilan *Splash Screen*

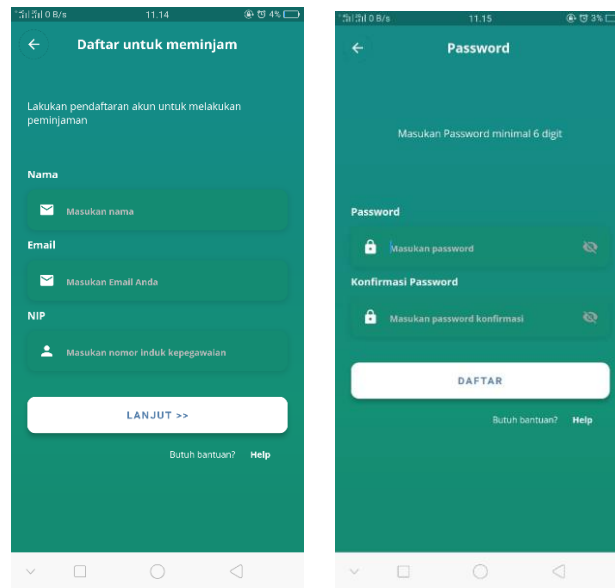
Pada tampilan *splash screen* merupakan tampilan awal aplikasi yang terdapat logo instansi, yaitu logo Kementerian Lingkungan Hidup Republik Indonesia, nama aplikasi, yaitu aplikasi Peminjaman Alat BPKH XI Wilayah Yogyakarta dan versi aplikasi. Tampilan aplikasi ini berfungsi sebagai penanda kepemilikan aplikasi serta identitas aplikasi. Tampilan *splash screen* dapat dilihat pada gambar 4.1.



Gambar 4.1 Tampilan *Splash Screen*

4.2.2 Tampilan Form Register Akun

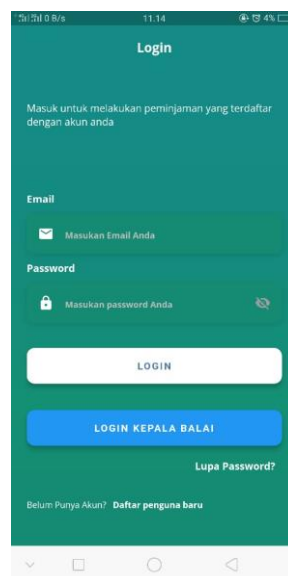
Pada tampilan form register akun, peminjam diharuskan untuk melakukan registrasi akun sebelum melakukan peminjaman. Pada tampilan ini peminjam disediakan form untuk melakukan registrasi yang terdiri dari nama, email, NIP dan *password*. Form login berfungsi untuk melakukan pendaftaran akun yang digunakan dalam melakukan peminjaman oleh peminjam serta validasi terhadap keseriusan dalam melakukan peminjaman. Form register dapat dilihat pada gambar 4.2.



Gambar 4.2 Tampilan Form Register

4.2.3 Tampilan Form Login Peminjam

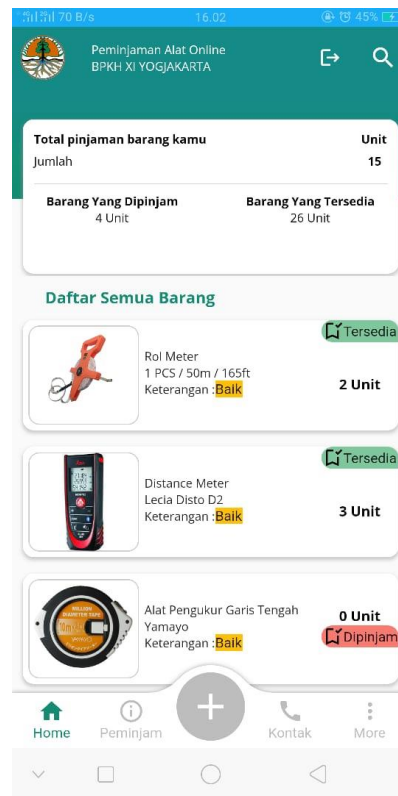
Pada tampilan form login peminjam disediakan *email* dan *password*. Form login berfungsi untuk melakukan login dengan *email* dan *password* yang akan diarahkan ke menu home sesuai hak akses yang diberikan. Tampilan form login peminjam dapat dilihat pada gambar 4.3.



Gambar 4.3 Tampilan Login Peminjam

4.2.4 Tampilan Menu Home

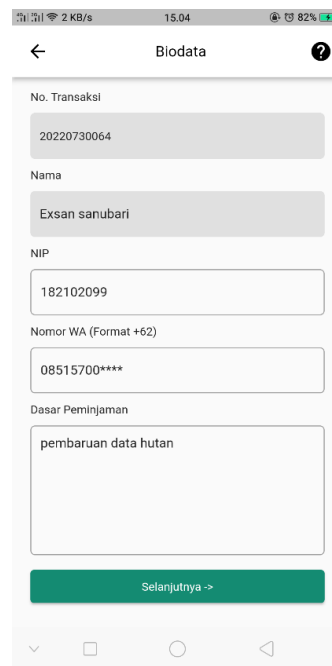
Pada tampilan menu Home digunakan sebagai tampilan utama aplikasi serta menampilkan data barang yang tersedia maupun tidak tersedia. Hal tersebut dapat dilihat pada gambar 4.4.



Gambar 4.4 Tampilan Menu Home

4.2.5 Tampilan Form Pengajuan Peminjaman

Pada tampilan *form* ini pegawai dapat mengajukan peminjaman dengan mengisi beberapa data pada *form* biodata peminjam, yaitu nama peminjam, nomor induk pegawai, nomor telepon, dan dasar peminjaman. Untuk nomor transaksi terisi otomatis oleh sistem yang terdiri dari tanggal peminjam serta nomor urut peminjam. Tampilan form tersebut dapat dilihat pada gambar 4.5.



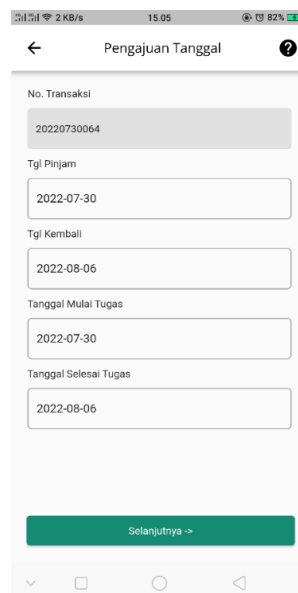
The screenshot shows a mobile application interface for a Biodata form. The title bar at the top reads "Biodata" with a back arrow on the left and a help icon on the right. The form fields are as follows:

- No. Transaksi: 20220730064
- Nama: Exsan sanubari
- NIP: 182102099
- Nomor WA (Format +62): 08515700****
- Dasar Pinjaman: pembaruan data hutan

A green button labeled "Selanjutnya ->" is located at the bottom of the form. The Android navigation bar is visible at the very bottom.

Gambar 4.5 Tampilan *Form* Biodata

Setelah mengisi *form* biodata untuk meminjam pegawai harus mengisi beberapa data pada *form* pengajuan tanggal peminjaman yang berisi tanggal pinjam, tanggal kembali, tanggal mulai tugas, dan tanggal berakhirnya tugas. Form tersebut dapat dilihat pada gambar 4.6.



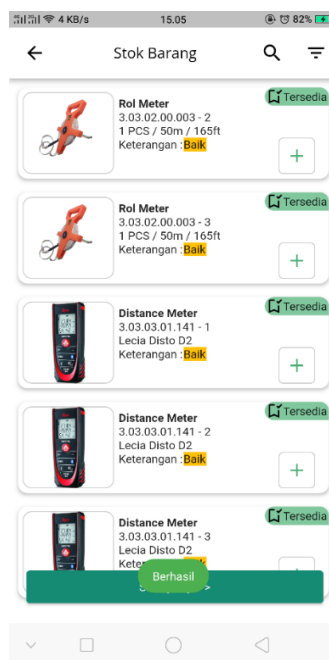
The screenshot shows a mobile application interface for a Pengajuan Tanggal form. The title bar at the top reads "Pengajuan Tanggal" with a back arrow on the left and a help icon on the right. The form fields are as follows:

- No. Transaksi: 20220730064
- Tgl Pinjam: 2022-07-30
- Tgl Kembali: 2022-08-06
- Tanggal Mulai Tugas: 2022-07-30
- Tanggal Selesai Tugas: 2022-08-06

A green button labeled "Selanjutnya ->" is located at the bottom of the form. The Android navigation bar is visible at the very bottom.

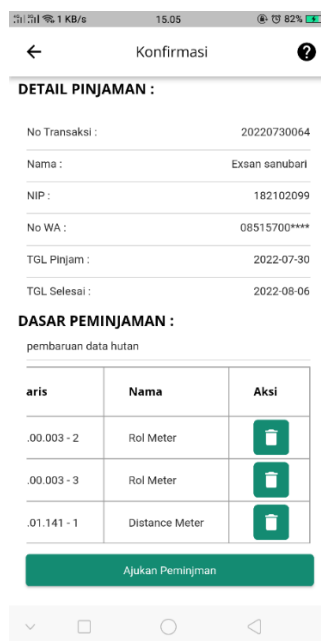
Gambar 4.6 Tampilan *Form* Pengajuan Tanggal Peminjaman

Pada tampilan List Barang pegawai setelah mengisi form biodata peminjam dapat memilih barang yang terdapat di tampilan daftar stok barang yang berisi nama barang, nomor inventaris, merek barang, keterangan barang dan foto barang dengan menekan tanda plus (+) pada barang yang ingin dipinjam. Tampilan tersebut dapat dilihat pada gambar 4.7.



Gambar 4.7 Tampilan *List* Barang

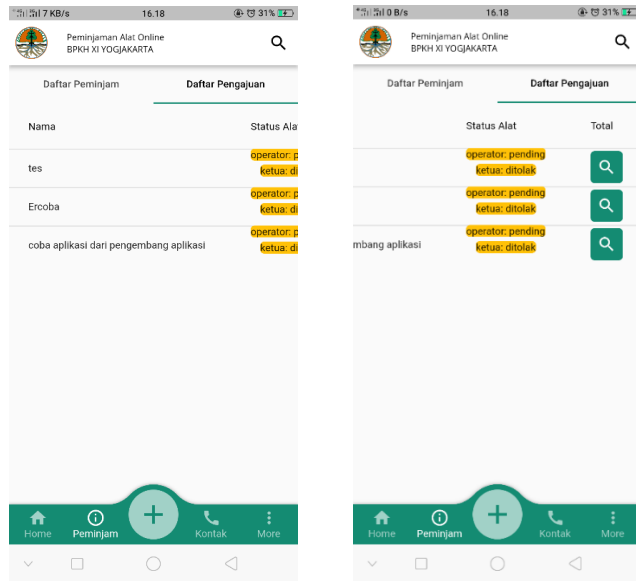
Pada tampilan menu Konfirmasi Pengajuan pegawai dapat meninjau kembali data peminjaman yang berupa nama peminjam, nomor induk pegawai, nomor telepon, tanggal pinjam, tanggal selesai tugas, dasar peminjaman, serta daftar barang yang dipinjam. Peminjam dapat memperbarui data dengan cara kembali ke halaman sebelumnya serta dapat mengurangi barang yang ingin dipinjam sebelum peminjam mengajukan peminjaman. Setelah peminjam meninjau data-data peminjaman peminjam langsung dapat mengajukan peminjaman dengan cara menekan tombol ajukan peminjaman. Tampilan tersebut dapat dilihat pada gambar 4.8.



Gambar 4.8 Tampilan Menu Konfirmasi Pengajuan

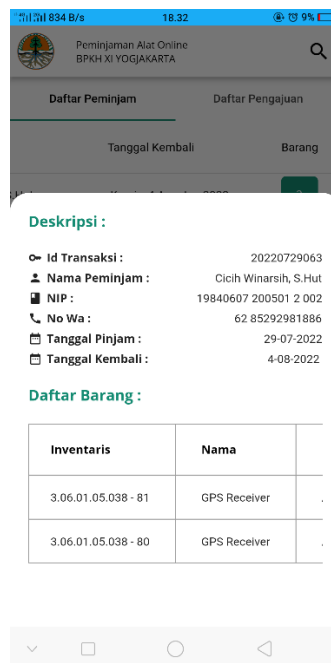
4.2.6 Tampilan Daftar Pengajuan

Pada tampilan ini pegawai dapat melihat daftar nama pengajuan peminjaman yang oleh semua pegawai dalam bentuk tabel. Pada tampilan tabel terdapat nama peminjam serta status alat dan aksi untuk melihat informasi alat yang dipinjam. Status alat berfungsi untuk mengetahui status persetujuan peminjaman oleh kepala balai serta admin inventaris alat dengan barang yang dipinjam. Tampilan Daftar Pengajuan dapat dilihat pada gambar 4.9.



Gambar 4.9 Tampilan Daftar Pengajuan

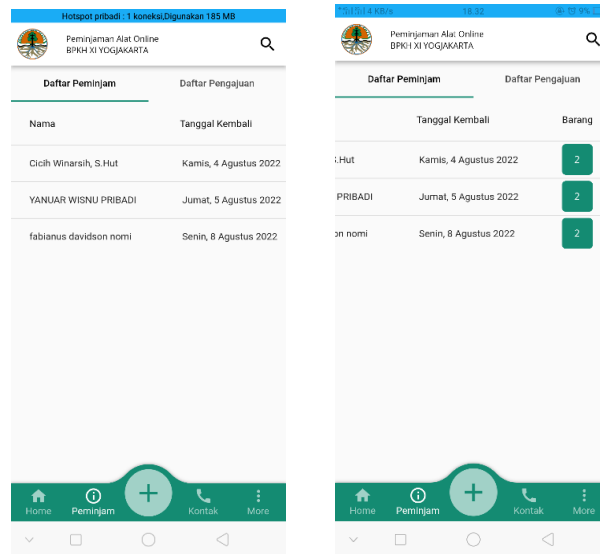
Pada tampilan Tabel Detail Barang pegawai dapat melihat daftar barang yang sedang dipinjam oleh peminjam ditampilkan dalam bentuk tabel. Pada tabel tersebut terdapat biodata peminjam serta daftar barang yang sedang dipinjam, seperti terlihat pada gambar 4.10.



Gambar 4.10 Tampilan Table Detail Barang

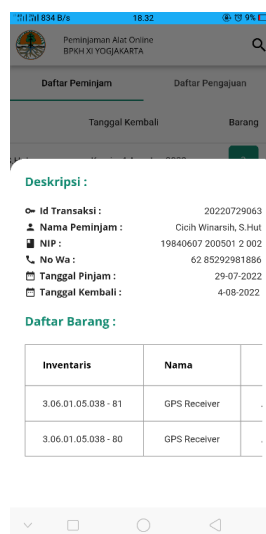
4.2.7 Tampilan Daftar Peminjam

Pada tampilan ini pegawai dapat melihat daftar nama peminjaman barang yang sedang dipinjam oleh semua pegawai dalam bentuk tabel. Pada tampilan tabel terdapat nama pegawai serta tanggal kembali dan aksi untuk melihat informasi barang yang dipinjam, seperti terlihat pada gambar 4.11.



Gambar 4.11 Tampilan Daftar Peminjam

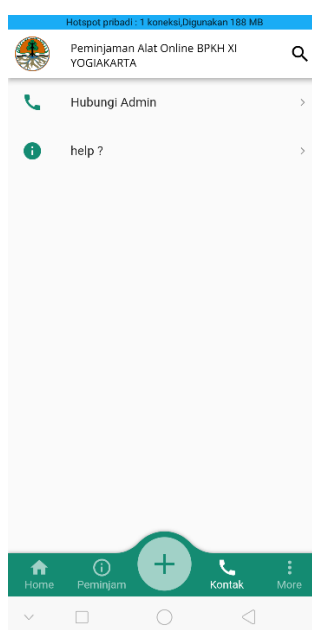
Pada tampilan Tabel Detail Barang pegawai dapat melihat daftar barang yang sedang dipinjam oleh pegawai ditampilkan dalam bentuk tabel. Pada tabel tersebut terdapat biodata pegawai serta daftar barang yang sedang dipinjam, seperti terlihat pada gambar 4.12.



Gambar 4.12 Tampilan Tabel Detail Barang yang Dipinjam

4.2.8 Tampilan Menu Kontak

Pada tampilan ini merupakan tampilan kontak nomor telepon admin barang serta kontak pengembang aplikasi. Kontak berfungsi untuk melaporkan keluhan terhadap sistem dan terhadap keluhan peminjaman. Pegawai dapat menekan menu admin untuk melaporkan keluhan, kemudian dibawa ke aplikasi telepon pada perangkat seluler yang digunakan, sedangkan menu help akan langsung dibawa ke aplikasi Whatsapp untuk melaporkan keluhan terhadap sistem. Hal tersebut dapat dilihat pada gambar 4.13.



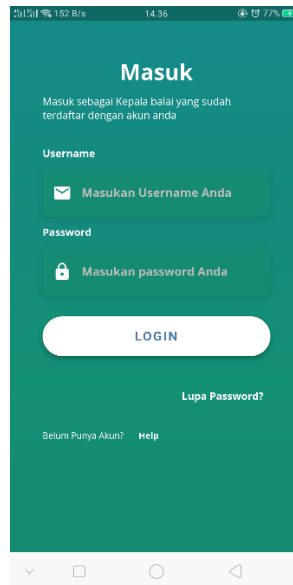
Gambar 4.13 Tampilan Menu Keluhan

4.3 IMPLEMENTASI DISAIN INTERFACE KEPALA BALAI

Tampilan Kepala Balai tersusun berdasarkan kode program yang dibuat yang direlasikan berdasarkan perancangan aplikasi. Dalam Menyusun kode program aplikasi Peminjaman Alat BPKH XI Wilayah Yogyakarta berbasis Android menggunakan Visual Studio Code dan menggunakan bahasa pemrograman Dart dan *framework* Flutter yang terintegrasi menggunakan data API yang dibuat berdasarkan sistem yang sudah ada. Berikut ini merupakan implementasi tampilan dan kode yang terdapat pada aplikasi.

4.3.1 Tampilan Form Login

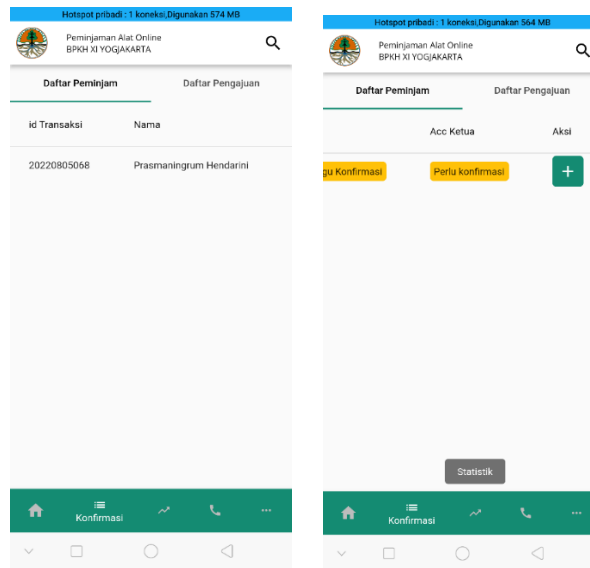
Pada tampilan form login Kepala Balai disediakan *username* dan *password*. Form login berfungsi untuk melakukan login dengan *username* dan *password* yang akan diarahkan ke menu home sesuai hak akses yang diberikan. Tampilan tersebut dapat dilihat pada gambar 4.14.



Gambar 4.14 Tampilan Form Login Approval

4.3.2 Tampilan Daftar Konfirmasi

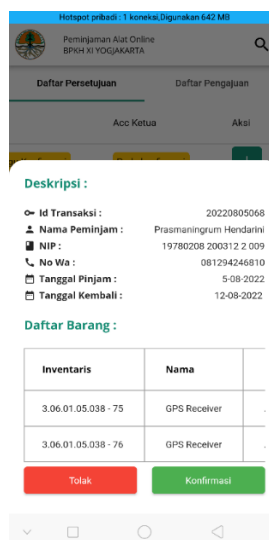
Pada tampilan ini berfungsi untuk menampilkan daftar seluruh peminjam barang yang berupa tabel. Pada tabel ini terdiri dari *id* transaksi, nama pegawai, tanggal pinjam, tanggal kembali, dan status konfirmasi. Tampilan ini hanya dapat diakses oleh kepala balai dengan cara login aplikasi. Tampilan Daftar Konfirmasi dapat dilihat pada gambar 4.15.



Gambar 4.15 Tampilan Daftar Konfirmasi

4.3.3 Tampilan Konfirmasi Peminjaman

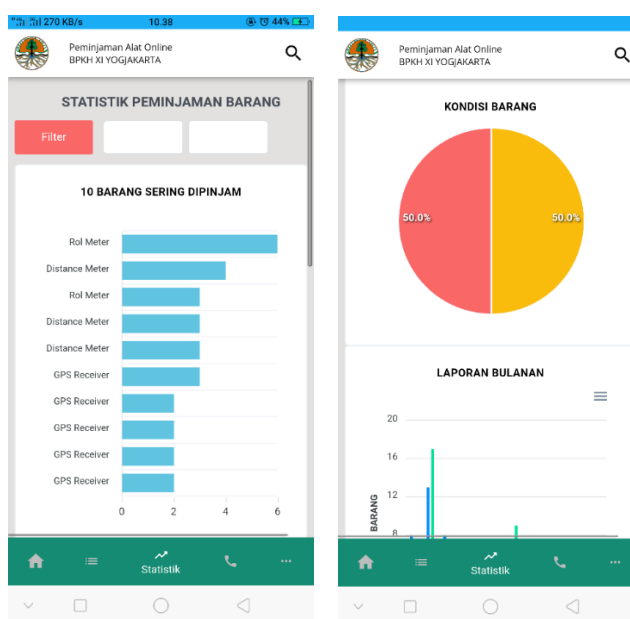
Tampilan ini merupakan tampilan detail daftar konfirmasi. Tampilan ini digunakan untuk persetujuan oleh Kepala Balai terhadap barang yang dipinjam oleh Pegawai. Kepala balai dapat melakukan persetujuan dengan cara menekan pada tombol “Konfirmasi” atau “Tolak”. Tombol Konfirmasi berfungsi sebagai aksi untuk menyetujui peminjaman, sedangkan tombol Tolak berfungsi untuk aksi menolak persetujuan terhadap barang yang dipinjam oleh pegawai. Hal tersebut dapat dilihat pada gambar 4.16..



Gambar 4.16 Tampilan Konfirmasi Peminjaman

4.3.4 Tampilan Statistik Pemijan

Pada tampilan ini berfungsi untuk menampilkan pelaporan bulanan peminjaman alat kepada Kepala Balai. Kepala balai dapat melakukan filtering berdasarkan tanggal yang diajukan peminjam. Pada tampilan terdapat beberapa diagram yang berfungsi untuk mengfvisualisaikan jumlah barang yang dipinjam berdasarkan tanggal peminjam. Tampilan diagram terdiri dari diagram batang, dan diagram pie. Diagram batang difungsikan untuk menampilkan data barang berdasarkan jumlah yang dipinjam serta laporan bulanan berdasarkan data barang yang dipinjam. Diagram *pie* berfungsi untuk menampilkan data seluruh kondisi barang yang ada pada daftar barang. Diagram tersebut dapat dilihat pada gambar 4.17.

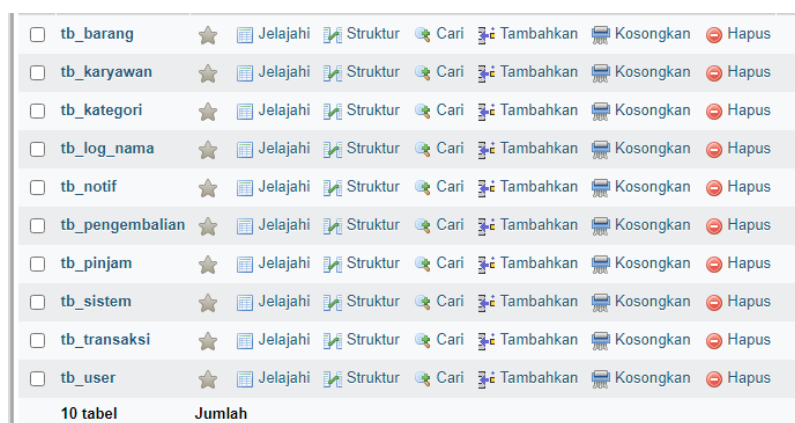


Gambar 4.17 Tampilan Statistik Pemijan

4.4 DATABASE

Dalam pembuatan aplikasi peminjaman alat BPKH wilayah Yogyakarta berbasis Android, menggunakan *database* MySQL yang sudah tersedia pada sistem *web* pada sistem yang ada. Pada bab sebelumnya telah dijelaskan bahwa tidak ada desain database yang akan digunakan. Sistem database yang ada dapat diintegrasikan ke dalam aplikasi yang akan dibangun menggunakan sistem API (*application programming interface*). Berikut ini adalah database MySQL yang

dapat diimplementasikan ke dalam sebuah aplikasi bernama `db_pinjam_barang` yang terdiri dari 5 tabel, yaitu tabel barang, tabel kategori, tabel peminjaman, dan tabel transaksi. Tampilan *database* MySQL dapat dilihat gambar 4.18.



Gambar 4.18 Tampilan *Database* MySQL

4.5 FITUR-FITUR SISTEM

Aplikasi Peminjaman Alat BPKH XI Wilayah Yogyakarta berbasis Android memiliki fitur-fitur sebagai berikut:

1. Menampilkan data barang.
2. Menampilkan data barang yang sudah dipinjam.
3. Menampilkan data peminjam.
4. Menampilkan data pengajuan.
5. Menampilkan data barang yang dipinjam dalam bentuk diagram.
6. Menampilkan data barang yang dipinjam dalam bentuk diagram berdasarkan tanggal peminjaman.
7. Filtering data barang berdasarkan kategori barang.
8. Fitur search barang berdasarkan nama barang, nomor inventaris, dan merk barang.

4.6 PROSES PEMBUATAN API

Proses pembuatan API aplikasi Peminjaman Alat BPKH XI wilayah Yogyakarta menggunakan bahasa pemrograman *framework* CodeIgniter. Penggunaan *framework* ini berfungsi untuk menyamakan bahasa pemrograman sistem *web* yang

sudah ada dan mempermudah dalam proses pengembangan aplikasi lebih lanjut dengan metode API yang dapat digunakan dalam berbagai bahasa pemrograman.

4.6.1 Proses Pembuatan Kode

Proses pembuatan kode digunakan untuk memproses data pada *database web* ke dalam bentuk API khususnya dari sistem aplikasi peminjaman alat BPKH Wilayah Yogyakarta. Pengodean dibuat berdasarkan fungsi aplikasi. Setiap fungsi terdapat beberapa kode terdiri dari fungsi dan aliran data yang ada pada aplikasi.

1. Kode *Connections Database*

Kode *connections database* berfungsi untuk menghubungkan CodeIgniter dengan database MYSQL. Kode ini diberi nama *database.php* yang terdapat pada file *config/database.php*. File ini akan dibikin secara otomatis oleh CodeIgniter yang dapat diubah sesuai pengaturan database. Pada *file config database* CodeIgniter ini kita wajib memasukkan *username database*, *password* serta nama database nya sesuai dengan *database* yang sudah terpasang di *web base* agar *database* bisa dipakai.

Adapun contoh kode fungsi *Connections database* pada aplikasi peminjaman alat BPKH Wilayah Yogyakarta dapat dilihat di bawah ini.

```
$active_group = 'default';
$query_builder = TRUE;

$db[ 'default' ] = array(
    'dsn' => '',
    'hostname' => 'localhost:3306',
    'username' => 'root',
    'password' => 'rootroot',
    'database' => 'db_pinjamalatbpkh11jogja',
    'dbdriver' => 'mysqli',
    'dbprefix' => '',
    'pconnect' => FALSE,
    'db_debug' => (ENVIRONMENT !== 'production'),
    'cache_on' => FALSE,
    'cachedir' => '',
    'char_set' => 'utf8',
    'dbcollat' => 'utf8_general_ci',
    'swap_pre' => '',
    'encrypt' => FALSE,
    'compress' => FALSE,
    'stricton' => FALSE,
    'failover' => array(),
```

```

        'save_queries' => TRUE
    );

```

2. Kode Models Data

Kode model data merupakan suatu file php yang ditempatkan di *directory* `application/models` yang digunakan buat mengambil informasi ataupun data yang terdapat di *database*. Selaku contoh kita mempunyai database daftar barang. Fungsi Kode model yang berisi function buat menjalankan insert, update serta delete terhadap tabel- tabel yang terdapat di database barang tersebut. Dengan demikian, pembuatan API memerlukan sebuah code *model* data hingga di kala membuat aplikasi memakai CodeIgniter kita memakai *model* buat menuntaskan seluruh perihal yang berkaitan dengan pembuatan *query* ataupun proses DML (Database Manipulation Language).

Adapun contoh kode fungsi model data pada aplikasi peminjaman alat BPKH Wilayah Yogyakarta dapat dilihat di bawah ini.

```

class Mod_barang extends CI_Model {

    private $table = "tb_barang";

    function totalRows($table)
    {
        return $this->db->select('')
            ->from($table);
    }
    function barangtersedia(){
        $this->db->select('COUNT(*) AS totalsisa')
            ->from($this->table);
        return $this->db->get();
    }

    function getAll()
    {
        $this->db->order_by('tb_barang.barang_id desc');
        return $this->db->get($this->table);
    }
    function count_by_kolom($kolom, $isi)
    {
        $this->db->select('COUNT(*) AS hasil')
            ->where($kolom, $isi)
            ->from($this->table);
        return $this->db->get();
    }
}

```


3. Kode Response JSON

Kode response JSON berfungsi untuk menampilkan data ke dalam bentuk tampilan data JSON. Data diambil dari penyimpanan data models kedalam bentuk parameter array selanjutnya data ditampilkan kedalam bentuk *string* menggunakan JSON (*array*).

Adapun contoh kode *response* JSON pada proses pembuatan data API dapat dilihat di bawah ini.

```
public function api_daftar_peminjam(){
    $data['status'] = '';
    $queryBarang = $this->Mod_pengembalian-
>getPengembaliandepan();
    $table = array();

    if($queryBarang->num_rows() > 0){
        foreach($queryBarang->result() as $transaksi){
            $sekarang = new DateTime();
            $tglselesai = new DateTime($transaksi-
>tanggal_kembali);
            if ($sekarang > $tglselesai) {
                $status = 'Terlambat Dikembalikan';
            } else {
                $kurangtiga = date_create($transaksi-
>tanggal_kembali)->modify('-3 days');
                if ($sekarang > $kurangtiga) {
                    $status = 'Masa pinjam kurang 3 kurang
                    hari';
                } else {
                    $status = 'Barang Masih di Dipinjam';
                }
            }
        }
        $table[] = array(
            'id' => (int)$transaksi->transaksi_id,
            'nama' => $transaksi->nama,
            'keterangan' => $status,
            'total_barang' => (int)$transaksi->totalbarang,
            'tanggal_pinjam'=>$transaksi->tanggal_pinjam,
            'tanggal_kembali'=>$transaksitanggal_kembali,
        );
    }
    $this->response->json(array(
        'data' => $table,
        'pesan' => 'Barang berhasil di ambil',
        'status' => true
    ),200);
}
}
else{
```

```

        $this->response->json(array(
            'data' => $table,
            'pesan' => 'Barang berhasil di ambil',
            'status' => false
        ),200);
    }
}

```

4.7 PROSES PEMBUATAN APLIKASI

Proses pembuatan aplikasi peminjaman alat BPKH wilayah Yogyakarta menggunakan bahasa pemrograman Dart dan *framework* Flutter. Dengan menggunakan bahasa pemrograman Dart dan Flutter mempermudah dalam proses pengembangan aplikasi lebih lanjut dengan teknologi Mobile Cross Platform yang dapat di gunakan dalam berbagai sistem operasi.

4.7.1 Proses Pembuatan Kode

Proses pengodean digunakan untuk memproses data *web* ke dalam bentuk platform Android interface khususnya dari sistem aplikasi peminjaman alat BPKH Wilayah Yogyakarta. Pengodean di buat berdasarkan fungsi aplikasi. Setiap fungsi terdapat beberapa kode terdiri dari fungsi dan aliran data yang ada pada aplikasi.

1. Kode Connections

Pengodean *Connections* berfungsi sebagai jalur untuk pengambilan data dari internet. Proses pengambilan data internet memerlukan sebuah paket yang ada di bahasa pemrograman Dart dan *framework* Flutter yang disebut dengan *http package*. Paket ini berfungsi untuk mengambil data pada internet yang berupa data JSON Map.

Adapun contoh kode fungsi *Connections* pada aplikasi peminjaman alat BPKH Wilayah Yogyakarta dapat di lihat di bawah ini.

```

import 'package:http/http.dart' as http;

RxInt limit = 5.obs;
RefreshController onRefres = RefreshController();
List<Datum> list = [];
List<Datum> listSearch = [];

Stream<InfoAlat> infoAlat() async* {
  try {
    final response = await http.get(Uri.parse(

```

```

        "https://pinjamalat.bpkh11jogja.net/api/pinjam/infobara
ng?limit=1000");
    if (response.statusCode == 200) {
        final data = infoAlatFromJson(response.body);
        yield data;
    }
  } catch (e) {
    print(e.toString());
  }
}

```

2. Kode Models Data

Kode *model data* berfungsi sebagai kerangka untuk menyimpan data sementara yang telah diambil dari data JSON yang berupa data mapping dengan di ambil melalui data API. Dalam proses ini memerlukan paket yang disebut Dart:convert. Paket ini berfungsi untuk mengubah data JSON ke dalam bentuk data yang bisa di gunakan dan di tampilkan pada aplikasi.

Adapun contoh kode fungsi *model data* pada aplikasi peminjaman alat BPKH Wilayah Yogyakarta dapat di lihat di bawah ini.

```

import 'dart:convert';

InfoAlat infoAlatFromJson(String str) =>
InfoAlat.fromJson(json.decode(str));

String infoAlatToJson(InfoAlat data) =>
json.encode(data.toJson());

class InfoAlat {
  InfoAlat({
    required this.data,
    required this.pesan,
    required this.status,
  });

  List<Datum> data;
  String pesan;
  bool status;

  factory InfoAlat.fromJson(Map<String, dynamic> json) =>
InfoAlat(
  data: List<Datum>.from(json["data"].map((x) =>
Datum.fromJson(x))),
  pesan: json["pesan"],
  status: json["status"],
);

  Map<String, dynamic> toJson() => {

```

```

        "data": List<dynamic>.from(data.map((x) => x.toJson())),
        "pesan": pesan,
        "status": status,
    };
}

class Datum {
    Datum({
        required this.id,
        required this.status,
        required this.kondisi,
        required this.namabarang,
        required this.merk,
        required this.kategori,
        required this.inventaris,
        required this.gambar,
        required this.keterangan,
    });

    int id;
    bool status;
    String kondisi;
    String namabarang;
    String merk;
    String kategori;
    String inventaris;
    String gambar;
    String keterangan;

    factory Datum.fromJson(Map<String, dynamic> json) => Datum(
        id: json["id"],
        status: json["status"],
        kondisi: json["kondisi"],
        namabarang: json["nama"],
        merk: json["merk"],
        kategori: json["kategori"],
        inventaris: json["inventaris"],
        gambar: json["gambar"],
        keterangan: json["keterangan"],
    );

    Map<String, dynamic> toJson() => {
        "id": id,
        "status": status,
        "kondisi": kondisi,
        "nama": namabarang,
        "merk": merk,
        "kategori": kategori,
        "inventaris": inventaris,
        "gambar": gambar,
        "keterangan": keterangan,
    };
}

```

3. Kode Views

Kode *views* berfungsi untuk menampilkan data kedalam bentuk tampilan Ui (*User Interface*) aplikasi. Data diambil dari penyimpanan data models ke dalam bentuk parameter *List<string, dynamic>* selanjutnya data di tampilkan kedalam bentuk string menggunakan *listview widget*.

Adapun contoh kode fungsi *views* pada aplikasi peminjaman alat BPKH Wilayah Yogyakarta dapat di lihat di bawah ini.

```
body: Obx(
  () => SmartRefresher(
    controller: controller.onRefres,
    onRefresh: controller.toRefres,
    enablePullUp: true,
    onLoading: controller.toLoading,
    child: SingleChildScrollView(
      child: Column(
        children: [
          Container(
            margin: EdgeInsets.only(
              left: 30,
              right: 30,
              top: 10),
            child: Row(
              mainAxisAlignment:
                MainAxisAlignment.spaceBetween,
              children: <Widget>[
                Text(
                  'Daftar Barang Sedang di Pinjam Hari Ini',
                  style: GoogleFonts.openSans(
                    fontSize: 16,
                    fontWeight: FontWeight.bold,
                    color:
                      Color.fromARGB(255, 20, 139, 114))),
              ],
            ),
          ),
        StreamBuilder<InfoAlat>(
          stream: controller.infoAlat(),
          builder: (context, snapshot) {
            if (snapshot.connectionState ==
              ConnectionState.waiting) {
              return LoadingWidget();
            } else {
              if (snapshot.hasData) {
                final infoAlat = snapshot.data!.data;
                return DayItemsWidget(
                  data: infoAlat,
                );
              }
            }
          }
        )
      )
    )
  )
)
```


4.8.1 Pengujian Forms Register Akun Peminjaman

Pengujian forms register akun peminjaman aplikasi merupakan sebuah proses pengujian terhadap proses halaman pada aplikasi Android yang dibuat. Hasil pengujian pada halaman akses aplikasi terdiri dari skenario yang diuji, cara pengujian, hasil yang dicapai, dan keterangan. Hasil pengujian pada halaman tombol forms register akun peminjaman dapat dilihat pada tabel 4.1

Tabel 4.1 Pengujian Forms Register Akun Peminjaman

No	Skenario yang Diuji	Cara Pengujian	Hasil yang dicapai	Keterangan
1.	Register	Mengklik Pendaftaran PenggunaDaru	Berhasil masuk ke menu register Aplikasi	Ok berhasil masuk
2.	Forms Register Akun Peminjaman dengan akun yang belum terdaftar	Melakukan register dengan nama, <i>email</i> , NIP (Nomor Induk Pegawai), dan password yang belum terdaftar dalam sistem	Berhasil melakukan register	Ok berhasil tersimpan
3.	Forms Register Akun Peminjaman dengan akun yang sudah terdaftar	Melakukan register dengan nama, <i>email</i> , NIP (Nomor Induk Pegawai), dan password yang sudah terdaftar dalam sistem	Menampilkan Akun sudah terdaftar	Ok berhasil menampilkan pesan

4.8.2 Pengujian Form Login Kepala Balai dan Peminjam

Pengujian *form* ini merupakan proses pengujian halaman di aplikasi Android yang dibuat. Hasil pengujian pada form login terdiri dari skenario yang

diuji, cara pengujian, hasil yang dicapai, dan keterangan. Hasil pengujian pada halaman form login dapat kita lihat pada tabel 4.2.

Tabel 4.2 Pengujian halaman Form Login Kepala Balai dan Peminjam

No	Skenario yang Diuji	Cara Pengujian	Hasil yang dicapai	Keterangan
1.	<i>Form login</i> Aplikasi	Melakukan <i>login</i> dengan <i>username</i> dan <i>password</i> yang valid	Berhasil masuk ke menu home Aplikasi	Ok berhasil masuk
2.	<i>Logout</i>	Mengklik Logout	Berhasil melakukan Logout	OK berhasil Logout
3.	<i>Form login</i> Aplikasi untuk Kepala Balai	Melakukan <i>login</i> dengan <i>username</i> dan <i>password</i> yang tidak valid	Menampilkan gagal Login	Ok berhasil menampilkan pesan

4.8.3 Pengujian Pengajuan Peminjaman

Pengujian pengajuan peminjaman merupakan proses pengujian halaman di aplikasi Android yang dibuat. Hasil pengujian pada halaman pengajuan peminjaman terdiri dari skenario yang diuji, cara pengujian, hasil yang dicapai, dan keterangan. Hasil pengujian pada halaman pengujian pengajuan dapat kita lihat pada tabel 4.3.

Tabel 4.3 Pengujian halaman Pengajuan Peminjaman

No	Skenario yang Diuji	Cara Pengujian	Hasil yang dicapai	Keterangan
1.	Form biodata peminjaman	Klik tombol Lanjutkan	Berhasil menambahkan biodata	Ok berhasil masuk ke <i>form</i> pengajuan tanggal peminjaman

2.	Form pengajuan tanggal peminjaman	Klik tombol Lanjutkan	Berhasi menambahkan tanggal peminjaman	Ok berhasil masuk ke list stok barang
3.	Tambah barang peminjaman	Klik tombol Tambah	Menampilkan pesan barang berhasil di tambahkan	Ok berhasil menampilkan pesan
4.	Tambah barang peminjaman	Klik tombol Lanjutkan	Berhasi menambahkan barang	Ok berhasil masuk ke masuk ke halaman konfirmasi
5.	Konfirmasi pengajuan peminjaman	Klik tombol Ajukan Peminjaman	Menampilkan pesan Berhasil	Ok berhasil menampilkan pesan

4.8.4 Pengujian Daftar Pengajuan Peminjaman

Pengujian daftar pengajuan peminjaman merupakan proses pengujian halaman di aplikasi Android yang dibuat. Hasil pengujian pada halaman daftar pengajuan peminjaman terdiri dari skenario yang diuji, cara pengujian, hasil yang dicapai, dan keterangan. Hasil pengujian pada halaman daftar pengajuan peminjaman dapat kita lihat pada tabel 4.4.

Tabel 4.4 Pengujian Halaan Daftar Pengajuan Peminjaman

No	Skenario yang Diuji	Cara Pengujian	Hasil yang dicapai	Keterangan
1.	Daftar Pengajuan Peminjan	Mengeklik Menu Peminjam selanjtnya Mengklik Daftar pengajuan	Menampilkan table Pengajuan peminjaman	OK

2.	Daftar Peminjam	Mengeklik Menu Peminjam selanjtnya Mengklik Daftar Peminjam	Menampilkan table Peminjaman	OK
----	-----------------	---	------------------------------	----

4.8.5 Pengujian Approval Peminjaman

Pengujian approval peminjaman merupakan proses pengujian halaman di aplikasi Android yang dibuat. Hasil pengujian pada approval peminjaman terdiri dari sekenario yang diuji, cara pengujian, hasil yang diharapkan, dan keterangan. Hasil pengujian pada halaman approval peminjaman dapat kita lihat pada tabel 4.5.

Tabel 4.5 Pengujian Halaman Approval Peminjaman

No	Skenario yang Diuji	Cara Pengujian	Hasil Capaian	Keterangan
1.	Daftar konfirmasi	Mengeklik tombol Menu Konfirmasi selanjutnya mengeklik Daftar Konfirmasi	Menampilkan table Konfirmasi Peminjaman	OK
2.	Deskripsi Konfirmasi	Mengeklik tombol Aksi	Menampilkan Deskripsi Peminjaman	Ok
3.	Konfirmasi Peminjaman	Mengeklik tombol Konfirmasi	Menampilkan pesan Berhasil	Ok berhasil menampilkan pesan

4.8.6 Pengujian Data API Metode Get

Pengujian data API metode *get* merupakan proses pengujian aliran database yang di buat dalam bentuk data json. Hasil pengujian pada data API metode *get* terdiri dari data yang diuji, cara pengujian, hasil yang dicapai, dan keterangan. Hasil pengujian pada halaman pengujian data API metode *get* dapat kita lihat pada tabel 4.6.

Tabel 4.6 Pengujian Data API Metode Get

No	Data yang Diuji	Cara Pengujian	Hasil yang dicapai	Keterangan
1.	Get Info Alat	Di jalankan menggunakan postman menggunakan metode Get	Data Tampil dalam bentuk JSON	Ok berhasil tampil
2.	Get Kategori Alat	Di jalankan menggunakan postman menggunakan metode Get	Data Tampil dalam bentuk JSON	Ok berhasil tampil
3.	Get Detail Transaksi	Di jalankan menggunakan postman menggunakan metode Get	Data Tampil dalam bentuk JSON	Ok berhasil tampil
4.	Get Daftar Barang Pengajuan	Di jalankan menggunakan postman menggunakan metode Get	Data Tampil dalam bentuk JSON	Ok berhasil tampil
5.	Get Info Peminjaman	Di jalankan menggunakan postman menggunakan metode Get	Data Tampil dalam bentuk JSON	Ok berhasil tampil
6.	Get Daftar Barang yang di Pinjam	Di jalankan menggunakan postman menggunakan metode Get	Data Tampil dalam bentuk JSON	Ok berhasil tampil
7.	Get Daftar Konfirmasi Peminjaman	Di jalankan menggunakan postman menggunakan metode Get	Data Tampil dalam bentuk JSON	Ok berhasil tampil

8.	Get Detail Konfirmasi Peminjaman	Di jalankan menggunakan postman menguankan metode Get	Data Tampil dalam bentuk JSON	Ok berhasil tampil
----	----------------------------------	---	-------------------------------	--------------------

4.8.1 Pengujian Data API Metode Post

Pengujian data API metode *post* merupakan proses pengujian aliran database yang berfungsi untuk mengirimkan data ke datanbase. Hasil pengujian pada data API metode *post* terdiri dari data yang diuji, cara pengujian, hasil yang dicapai, dan keterangan. Hasil pengujian pada halaman pengujian pengajuan dapat kita lihat pada tabel 4.7.

Tabel 4.7 Pengujian Data API Metode Post

No	Data yang Diuji	Cara Pengujian	Hasil yang dicapai	Keterangan
1.	Post Login	Di jalankan menggunakan postman menguankan metode post mengirimkan <i>password</i> dan <i>username</i>	Data Tampil dalam bentuk JSON yang berupa token	Ok berhasil tampil
2.	Post Ajukan Peminjaman	Di jalankan menggunakan postman menguankan metode post dan mengirimkan data peminjam	Data Terkirim ke database dan menampilkan response	Ok data tersimpan
3.	Post Konfirmasi pengawas	Di jalankan menggunakan postman menguankan metode post mengirimkan Data <i>boolean</i>	Data Terkirim ke database dan menampilkan response	Ok data tersimpan

4.9 PEMBAHASAN

Aplikasi Peminjaman Alat BPKH wilayah Yogyakarta berbasis Android yang terintegrasi dengan halaman *web* situs peminjaman alat BPKH Wilayah Yogyakarta dibangun menggunakan *framework* Dart, yaitu Flutter dalam Android. Aplikasi ini dapat membantu pegawai dan kepala balai khususnya di bagian pengelolaan peminjaman alat barang milik negara dengan pelaporan kepada pemerintah terhadap data ketersediaan barang. Selain membantu dalam proses pelaporan ketersediaan barang, aplikasi ini juga dapat mempermudah dalam proses peminjaman alat inventaris oleh pegawai tanpa adanya batasan hak akses aplikasi, serta dapat diakses di manapun melalui perangkat seluler oleh penggunanya.

Secara garis besar, bab ini akan memberikan gambaran penelitian, yaitu perbandingan proses pengelolaan dan pelaporan yang dicoba sebelumnya dengan yang baru. Tidak hanya itu, bab ini juga akan memaparkan kesimpulan dari pembuatan aplikasi baru berupa kelebihan dan kekurangan dari aplikasi baru tersebut.

4.9.1 Proses Bisnis Lama

Setelah melakukan analisis terhadap proses bisnis yang digunakan dalam peminjaman alat dengan metode *web base* peminjam harus mengakses melalui *web browser* peminjam harus mengingat nama domain aplikasi dan memasukkannya ke dalam *forms* url pada *web browser*, sedangkan peminjam banyak yang lupa nama domain dengan sistem *web base* yang digunakan. Setelah masuk ke sistem peminjam dapat mengajukan peminjaman aplikasi dengan memasukkan biodata peminjam dan memilih alat yang ingin dipinjam. Sistem akan mengirimkan data peminjam kepada kepala balai dan kemudian disetujui kepala balai, sedangkan kepala balai tidak ada notifikasi pengajuan peminjaman oleh peminjam. Peminjam diharuskan untuk mengirimkan pesan WhatsApp ketua balai. Cara ini dinilai tidak efektif karena pengelola admin akan mendapatkan banyak permintaan atau pesan masuk di WhatsApp dari banyak peminjam, proses ini mengakibatkan penumpukan

pinjaman, hal ini mengakibatkan admin alat peminjaman kewalahan dalam memproses pinjaman.

4.9.2 Proses Bisnis Baru

Setelah membangun aplikasi baru maka proses bisnis dalam pengelolaan peminjaman alat kantor BPKH Wilayah Yogyakarta akan berlangsung sebagai berikut:

1. Peminjam dapat mengajukan peminjaman melalui aplikasi yang mudah di-download melalui Google Playstore tanpa mengingat nama domain sebuah *web base*.
2. Peminjam dapat memonitor secara *real time* dengan aplikasi tanpa mengingat nama domain sebuah *web base*.
3. Kepala Balai dapat mengajukan persetujuan melalui aplikasi yang mudah di-download melalui Google Playstore tanpa mengingat nama domain sebuah *web base*.
4. Kepala Balai dapat menerima notifikasi persetujuan tanpa melalui aplikasi WhatsApp dan dapat diterima langsung melalui aplikasi peminjaman.