

BAB IV

HASIL DAN PEMBAHASAN

4.1 Hasil

4.1.1 Hasil proses *web API*

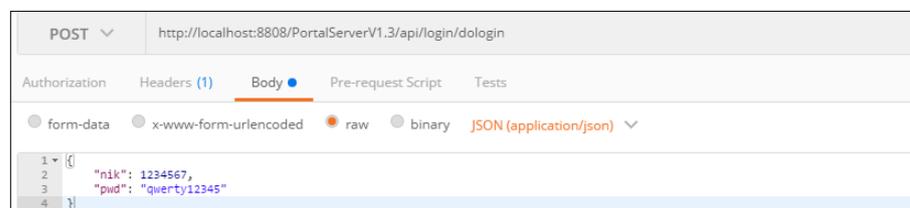
Berdasarkan pada perancangan *sequence* diagram pada bab III sub bab 3.3.3, maka dihasilkan *request* dan *response* yang dibagi menjadi 12 bagian sebagai berikut:

A. Proses *login*

Berdasarkan *sequence* diagram *login* pada gambar 3.15, model data *User* pada *list* 3.14 dan menggunakan fungsi *login* dari tabel *web API* pada tabel 3.1, didapatkan hasil sebagai berikut:

1. *Request*

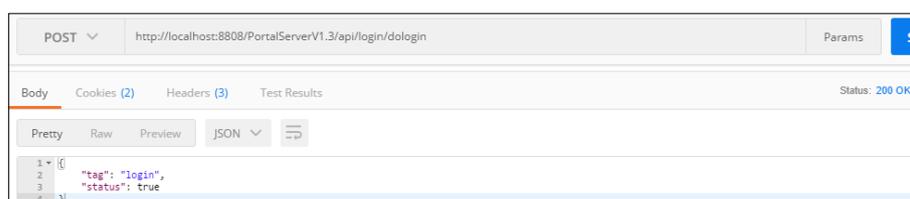
Pada saat aplikasi mengirimkan *request* untuk *login*, aplikasi mengirimkan data *JSON* seperti pada gambar 4.1.



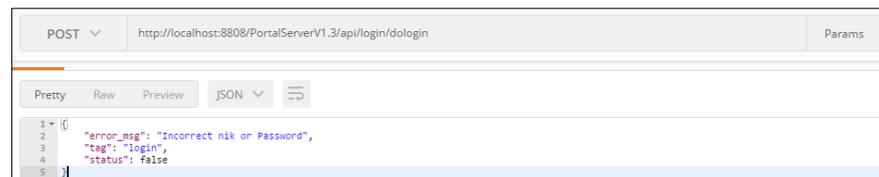
Gambar 4.1 *Request login*

2. *Response*

Setelah *request login* terkirim, *server* akan mengecek data *login* yang dikirimkan. Jika data *login* benar maka *server* akan mengirimkan *response* seperti pada gambar 4.2, namun jika data *login* salah maka *server* akan mengirimkan *response* seperti pada gambar 4.3.



Gambar 4.2 *Response login benar*



Gambar 4.3 Response login salah

B. Proses registrasi

Berdasarkan *sequence* diagram *registrasi* pada gambar 3.14, model data *User* pada list 3.14 dan menggunakan fungsi registrasi dari tabel *web API* pada tabel 3.1, didapatkan hasil sebagai berikut:

1. Request

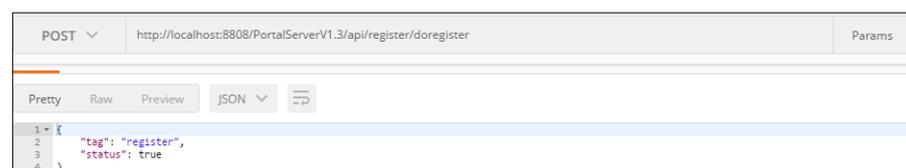
Pada saat aplikasi mengirimkan *request* registrasi, aplikasi mengirimkan data *JSON* seperti pada gambar 4.4.



Gambar 4.4 Request registrasi

2. Response

Setelah *request* registrasi terkirim, *server* akan mengecek data registrasi yang dikirimkan. Jika data registrasi benar maka *server* akan mengirimkan *response* seperti pada gambar 4.5, namun jika data registrasi sudah terdaftar maka *server* akan mengirimkan *response* seperti pada gambar 4.6.



Gambar 4.5 Response registrasi berhasil



Gambar 4.6 Response registrasi gagal

C. Proses informasi beranda

Berdasarkan *sequence* diagram beranda pada gambar 3.15, model data beranda pada list 3.25 dan menggunakan fungsi mengambil data *newsfeed* dari tabel *web API* pada tabel 3.1, didapatkan hasil sebagai berikut:

1. Request

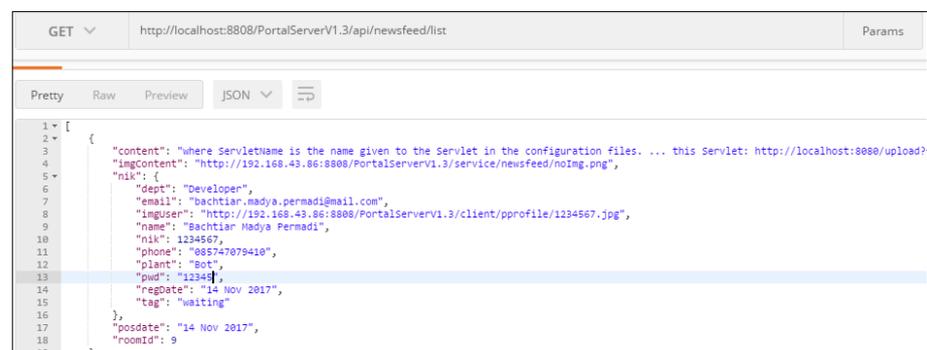
Aplikasi mengirimkan *request* informasi beranda menggunakan fungsi mengambil data *newsfeed* pada *web API* tabel 3.1 seperti pada gambar 4.7



Gambar 4.7 Request informasi beranda

2. Response

Setelah *request* terkirim, *server* akan mengirimkan *response* data informasi beranda seperti pada gambar 4.8.



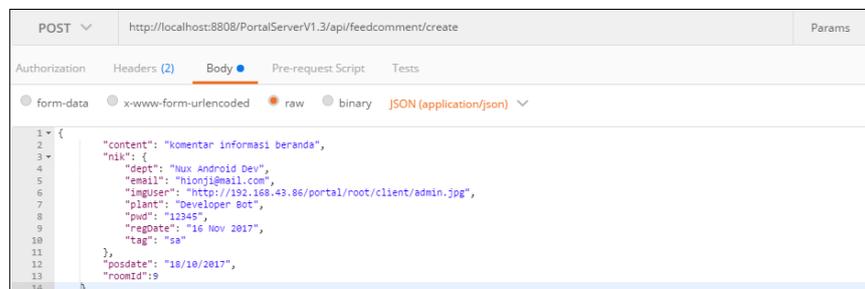
Gambar 4.8 Response informasi beranda

D. Proses komentar beranda

Berdasarkan *sequence* diagram komentar beranda pada gambar 3.16, model data beranda pada *list* 3.26 dan menggunakan fungsi mengambil data komentar pada *newsfeed* dan fungsi mengirim komentar pada *newsfeed* dari tabel *web API* pada tabel 3.1, didapatkan hasil sebagai berikut:

1. Request

Request dikirimkan menggunakan fungsi mengirim komentar beranda pada tabel *web API* tabel 3.1. *Request* komentar beranda dapat dilihat pada gambar 4.9



```

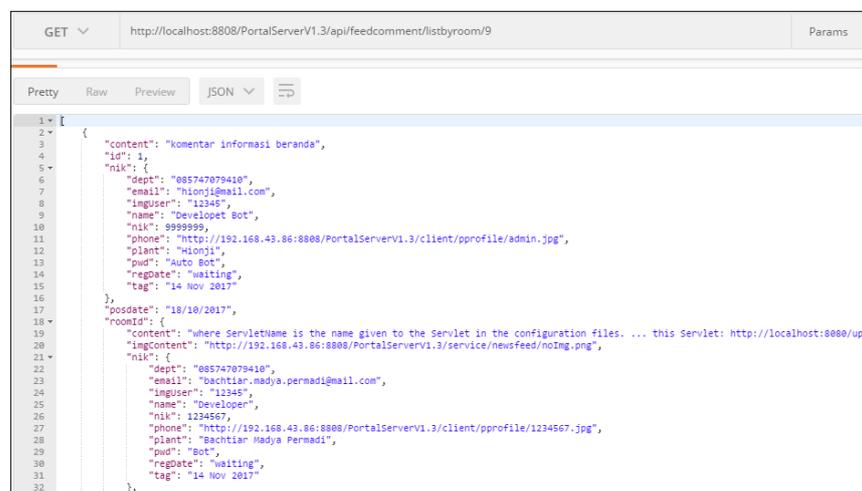
1 - {
2   "content": "komentar informasi beranda",
3   "nik": {
4     "dept": "Nux Android Dev",
5     "email": "hionji@mail.com",
6     "imguser": "http://192.168.43.86/portal/root/client/admin.jpg",
7     "plant": "Developer Bot",
8     "pwd": "12345",
9     "regdate": "16 Nov 2017",
10    "tag": "sa"
11  },
12  "posdate": "18/10/2017",
13  "roomId": 9
14 }

```

Gambar 4.9 Request mengirim komentar beranda

2. Response

Setelah *request* terkirim, *server* akan mengirimkan *response* data komentar beranda menggunakan fungsi mengambil data komentar beranda tabel *web API* tabel 3.1. *Response* komentar beranda dapat dilihat pada gambar 4.10.



```

1 - [
2   {
3     "content": "komentar informasi beranda",
4     "id": 1,
5     "nik": {
6       "dept": "085747079410",
7       "email": "hionji@mail.com",
8       "imguser": "12345",
9       "name": "Developer Bot",
10      "nik": 899999,
11      "phone": "http://192.168.43.86:8808/PortalServerV1.3/client/pprofile/admin.jpg",
12      "plant": "hionji",
13      "pwd": "Auto Bot",
14      "regdate": "waiting",
15      "tag": "14 Nov 2017"
16    },
17    "posdate": "18/10/2017",
18    "roomId": {
19      "content": "where ServletName is the name given to the Servlet in the configuration files. ... this Servlet: http://localhost:8808/upl
20      "imgContent": "http://192.168.43.86:8808/PortalServerV1.3/service/newsfeed/noImg.png",
21      "nik": {
22        "dept": "085747079410",
23        "email": "bachtian.madya.permadi@mail.com",
24        "imgUser": "12345",
25        "name": "Developer",
26        "nik": 1234567,
27        "phone": "http://192.168.43.86:8808/PortalServerV1.3/client/pprofile/1234567.jpg",
28        "plant": "Bachtian Madya Permadi",
29        "pwd": "Bot",
30        "regdate": "waiting",
31        "tag": "14 Nov 2017"
32      }
33    }
34  }
35 ]

```

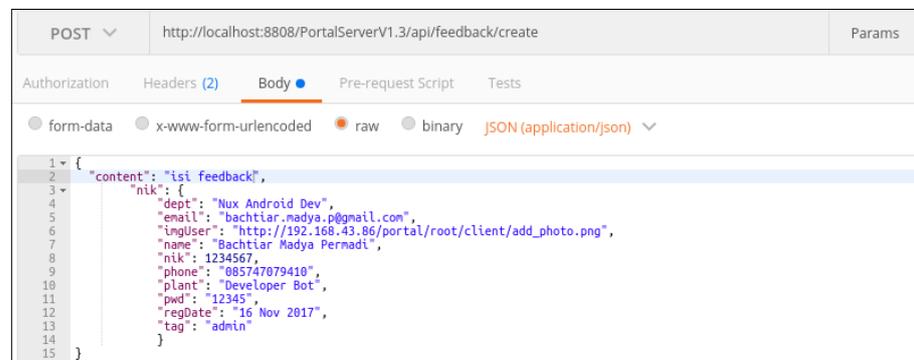
Gambar 4.10 Response menampilkan komentar beranda

E. Proses *feedback*

Berdasarkan *sequence* diagram *feedback* pada gambar 3.17, model data *feedback* pada list 3.27 dan menggunakan fungsi mengirim data *feedback* dan fungsi mengambil data *feedback* dari tabel *web API* pada tabel 3.1, didapatkan hasil sebagai berikut:

1. Request

Request dikirimkan menggunakan fungsi mengirim data *feedback* pada tabel *web API* tabel 3.1. *Request feedback* dapat dilihat pada gambar 4.11



Gambar 4.11 Request data *feedback*

2. Response

Setelah *request* terkirim, *server* akan mengirimkan *response feedback* menggunakan fungsi mengambil data *feedback* tabel *web API* tabel 3.1. *Response feedback* dapat dilihat pada gambar 4.12.

```

GET http://localhost:8808/PortalServerV1.3/api/feedback/list
Body
Pretty Raw Preview JSON
[
  {
    "dept": "Nux Android Dev",
    "email": "bachtiar.madya.p@gmail.com",
    "ingUser": "http://192.168.43.86/portal/root/client/1234567.jpg",
    "name": "Bachtiar Madya Permadi",
    "nik": "1234567",
    "phone": "085747079410",
    "plant": "Developer Bot",
    "pwd": "12345",
    "regDate": "16 Nov 2017",
    "tag": "admin"
  },
  {
    "posdate": "19 Dec 2017"
  }
],
  {
    "content": "test lagi",
    "id": 5,
    "nik": {
      "dept": "Nux Android Dev",
      "email": "hlonji@mail.com",
      "ingUser": "http://192.168.43.86/portal/root/client/admin.jpg",
      "name": "Hlonji",
      "nik": "9999999",
      "phone": "085747079410",
      "plant": "Developer Bot",
      "pwd": "12345",
      "regDate": "16 Nov 2017",
      "tag": "sa"
    },
    "posdate": "09 Dec 2017"
  }
]

```

Gambar 4.12 Response data feedback

F. Proses keluhan absensi

Berdasarkan *sequence* diagram keluhan absensi pada gambar 3.18, model data *feedback* pada list 3.28 dan menggunakan fungsi mengirim data keluhan absensi dan fungsi mengambil data keluhan absensi dari tabel *web API* pada tabel 3.1, didapatkan hasil sebagai berikut:

1. Request

Request dikirimkan menggunakan fungsi mengirim data keluhan absensi pada tabel *web API* tabel 3.1. *Request* keluhan absensi dapat dilihat pada gambar 4.13.

```

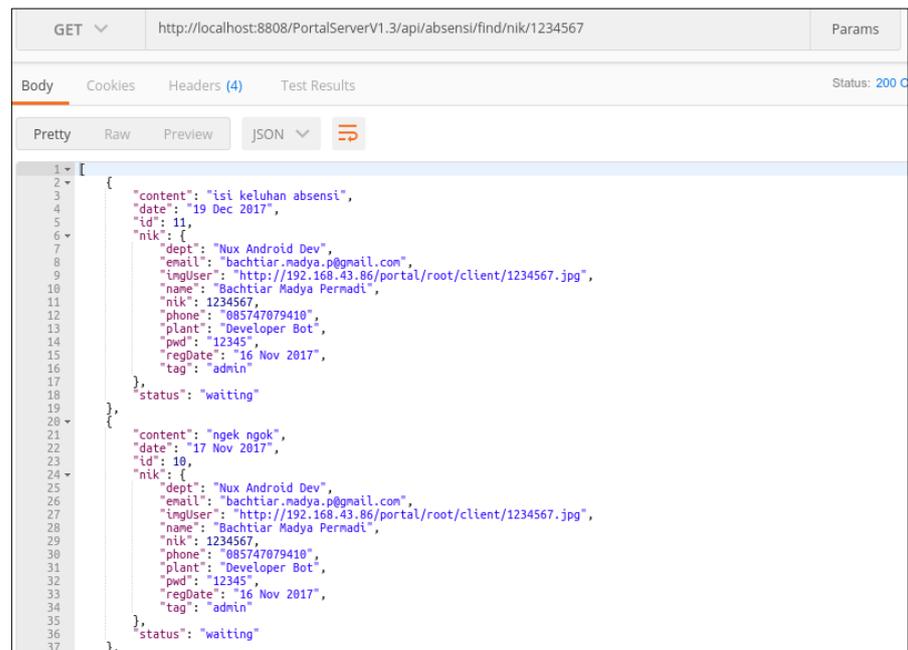
POST http://localhost:8808/PortalServerV1.3/api/absensi/create
Authorization Headers (2) Body Pre-request Script Tests
form-data x-www-form-urlencoded raw binary JSON (application/json)
[
  {
    "content": "isi keluhan absensi",
    "nik": {
      "dept": "Nux Android Dev",
      "email": "bachtiar.madya.p@gmail.com",
      "ingUser": "http://192.168.43.86/portal/root/client/add_photo.png",
      "name": "Bachtiar Madya Permadi",
      "nik": "1234567",
      "phone": "085747079410",
      "plant": "Developer Bot",
      "pwd": "12345",
      "regDate": "16 Nov 2017",
      "tag": "admin"
    }
  }
]

```

Gambar 4.13 Request data keluhan absensi

2. Response

Setelah *request* terkirim, *server* akan mengirimkan *response* keluhan absensi menggunakan fungsi mengambil data keluhan absensi tabel *web API* tabel 3.1. *Response* keluhan absensi dapat dilihat pada gambar 4.14.



```

1  [
2  {
3    "content": "isi keluhan absensi",
4    "date": "19 Dec 2017",
5    "id": 11,
6    "nik": {
7      "dept": "Nux Android Dev",
8      "email": "bachtiar.madya.p@gmail.com",
9      "ingUser": "http://192.168.43.86/portal/root/client/1234567.jpg",
10     "name": "Bachtiar Madya Permadi",
11     "nik": "1234567",
12     "phone": "085747079410",
13     "plant": "Developer Bot",
14     "pwd": "12345",
15     "regDate": "16 Nov 2017",
16     "tag": "admin"
17   },
18   "status": "waiting"
19 },
20 {
21   "content": "ngek ngok",
22   "date": "17 Nov 2017",
23   "id": 10,
24   "nik": {
25     "dept": "Nux Android Dev",
26     "email": "bachtiar.madya.p@gmail.com",
27     "ingUser": "http://192.168.43.86/portal/root/client/1234567.jpg",
28     "name": "Bachtiar Madya Permadi",
29     "nik": "1234567",
30     "phone": "085747079410",
31     "plant": "Developer Bot",
32     "pwd": "12345",
33     "regDate": "16 Nov 2017",
34     "tag": "admin"
35   },
36   "status": "waiting"
37 },
38 ]

```

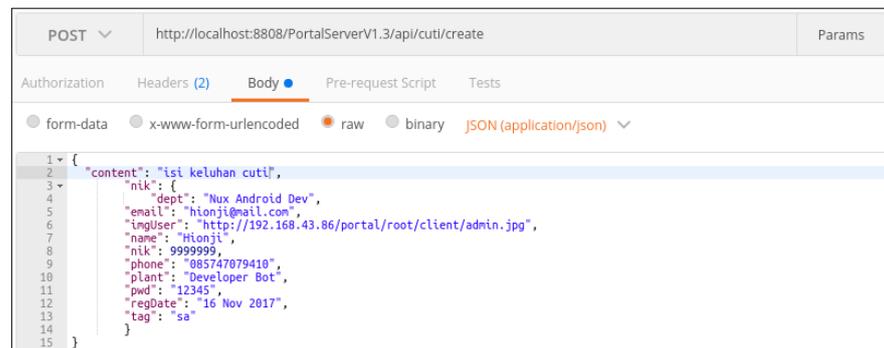
Gambar 4.14 Response data keluhan absensi

G. Proses keluhan cuti

Berdasarkan *sequence* diagram keluhan cuti pada gambar 3.19, model data keluhan cuti pada *list* 3.29 dan menggunakan fungsi mengirim data keluhan cuti dan fungsi mengambil data keluhan cuti dari tabel *web API* pada tabel 3.1, didapatkan hasil sebagai berikut:

1. Request

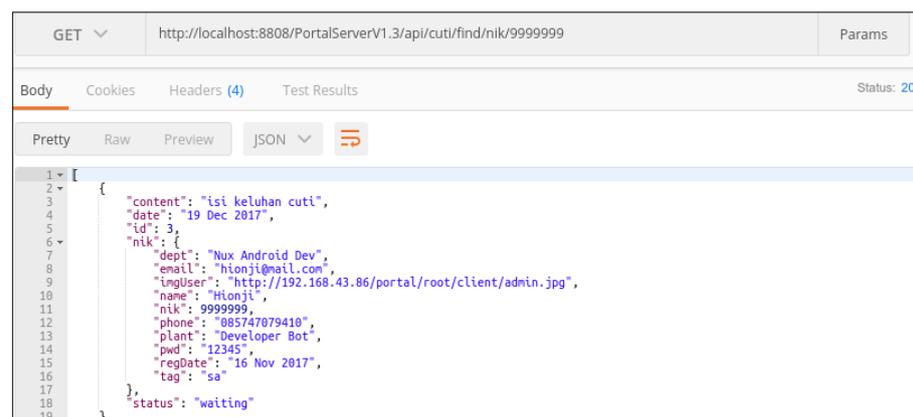
Request dikirimkan menggunakan fungsi mengirim data keluhan cuti pada tabel *web API* tabel 3.1. *Request* keluhan cuti dapat dilihat pada gambar 4.15.



Gambar 4.15 Request data keluhan cuti

2. Response

Setelah *request* terkirim, *server* akan mengirimkan *response* keluhan cuti menggunakan fungsi mengambil data keluhan cuti tabel *web API* tabel 3.1. *Response* keluhan cuti dapat dilihat pada gambar 4.16.



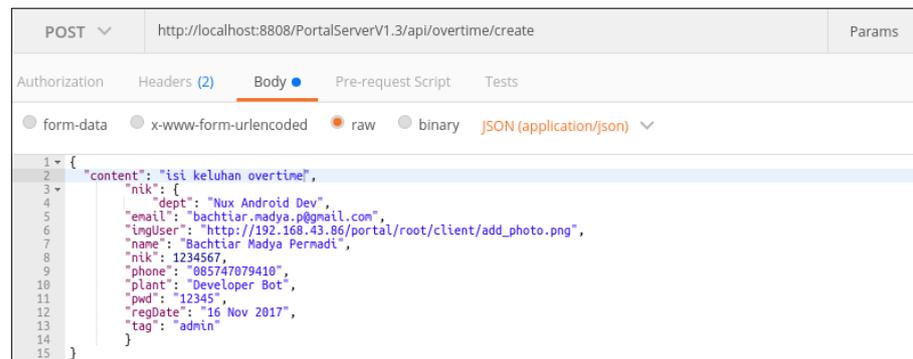
Gambar 4.16 Response data keluhan cuti

H. Proses keluhan overtime

Berdasarkan *sequence* diagram keluhan *overtime* pada gambar 3.20, model data keluhan *overtime* pada *list* 3.30 dan menggunakan fungsi mengirim data keluhan *overtime* dan fungsi mengambil data keluhan *overtime* dari tabel *web API* pada tabel 3.1, didapatkan hasil sebagai berikut:

1. Request

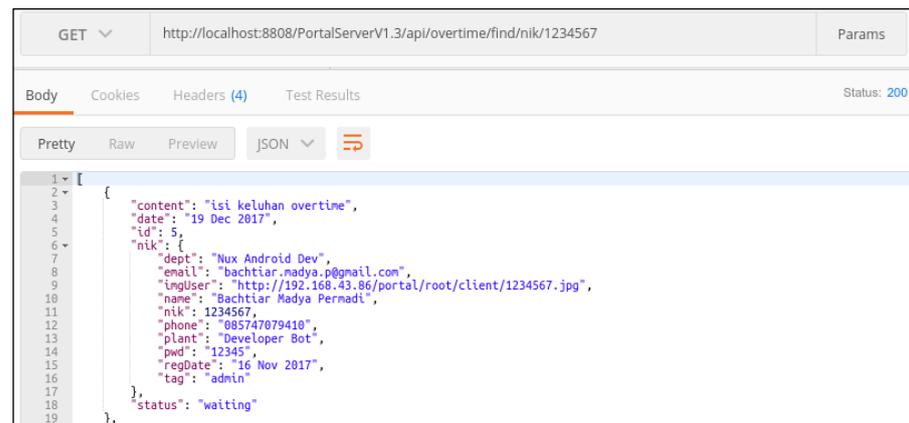
Request dikirimkan menggunakan fungsi mengirim data keluhan *overtime* pada tabel *web API* tabel 3.1. *Request* keluhan *overtime* dapat dilihat pada gambar 4.17



Gambar 4.17 Request data keluhan overtime

2. Response

Setelah *request* terkirim, *server* akan mengirimkan *response* keluhan overtime menggunakan fungsi mengambil data keluhan overtime tabel *web API* tabel 3.1. *Response* keluhan overtime dapat dilihat pada gambar 4.18.



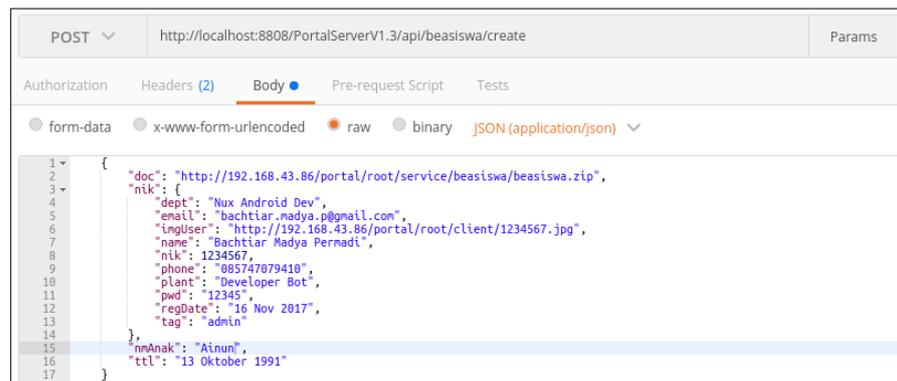
Gambar 4.18 Response data keluhan overtime

I. Proses pendaftaran beasiswa

Berdasarkan *sequence* diagram pendaftaran beasiswa pada gambar 3.21, model data pendaftaran beasiswa pada *list* 3.31 dan menggunakan fungsi mengirim data pendaftaran beasiswa dan fungsi mengambil data pendaftaran beasiswa dari tabel *web API* pada tabel 3.1, didapatkan hasil sebagai berikut:

1. Request

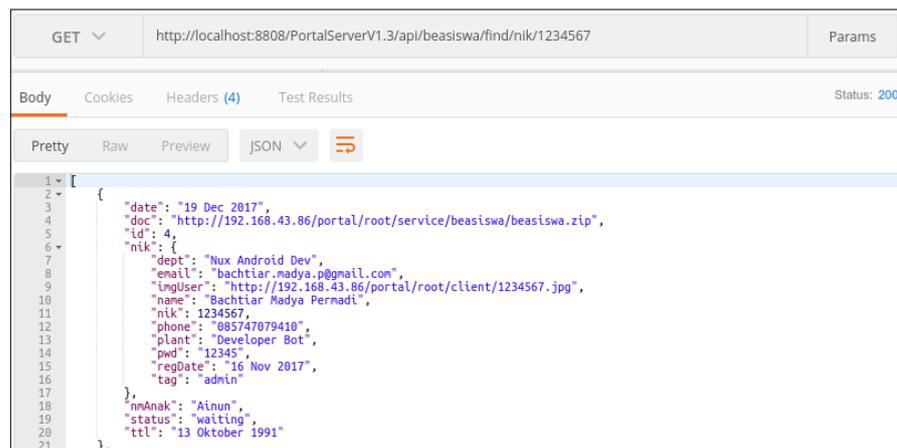
Request dikirimkan menggunakan fungsi mengirim data pendaftaran beasiswa pada tabel *web API* tabel 3.1. *Request* pendaftaran beasiswa dapat dilihat pada gambar 4.19.



Gambar 4.19 Request data pendaftaran beasiswa

2. Response

Setelah *request* terkirim, *server* akan mengirimkan *response* pendaftaran beasiswa menggunakan fungsi mengambil data pendaftaran beasiswa tabel *web API* tabel 3.1. *Response* pendaftaran beasiswa dapat dilihat pada gambar 4.20.



Gambar 4.20 Response data pendaftaran beasiswa

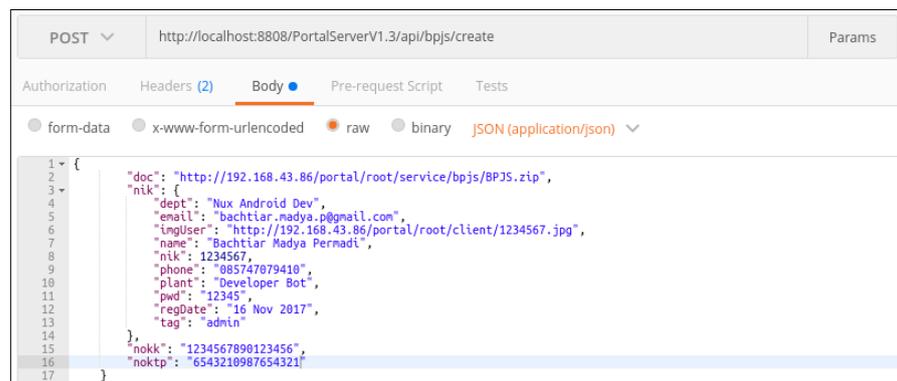
J. Proses pendaftaran BPJS

Berdasarkan *sequence* diagram pendaftaran BPJS pada gambar 3.22, model data pendaftaran BPJS pada *list* 3.32 dan menggunakan fungsi mengirim data

pendaftaran BPJS dan fungsi mengambil data pendaftaran BPJS dari tabel *web API* pada tabel 3.1, didapatkan hasil sebagai berikut:

1. Request

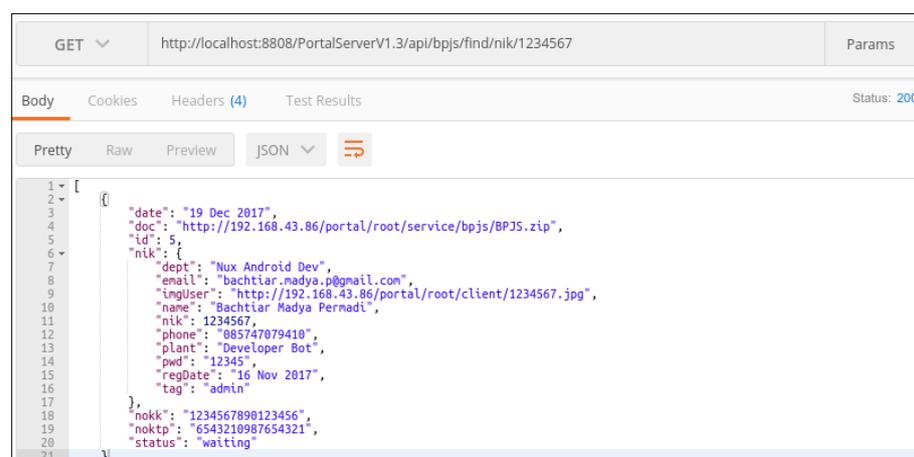
Request dikirimkan menggunakan fungsi mengirim data pendaftaran BPJS pada tabel *web API* tabel 3.1. *Request* pendaftaran BPJS dapat dilihat pada gambar 4.21.



Gambar 4.21 Request data pendaftaran BPJS

2. Response

Setelah *request* terkirim, *server* akan mengirimkan *response* pendaftaran BPJS menggunakan fungsi mengambil data pendaftaran BPJS tabel *web API* tabel 3.1. *Response* pendaftaran BPJS dapat dilihat pada gambar 4.22.



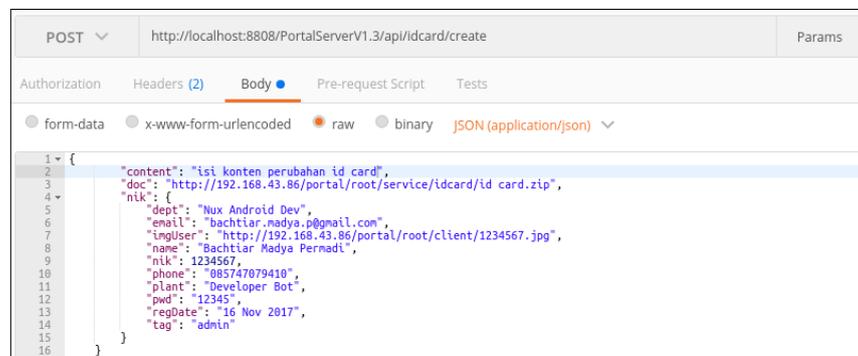
Gambar 4.22 Response data pendaftaran BPJS

K. Proses perubahan *id card*

Berdasarkan *sequence* diagram perubahan *id card* pada gambar 3.23, model data perubahan *id card* pada list 3.33 dan menggunakan fungsi mengirim data perubahan *id card* dan fungsi mengambil data perubahan *id card* dari tabel *web API* pada tabel 3.1, didapatkan hasil sebagai berikut:

1. Request

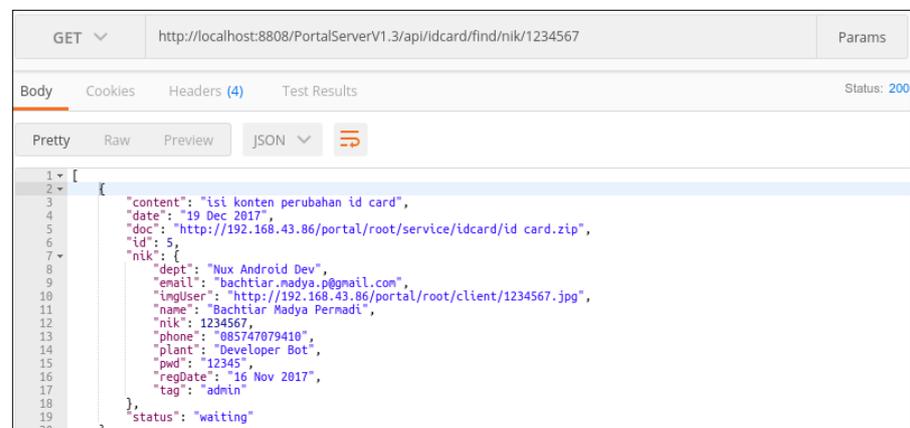
Request dikirimkan menggunakan fungsi mengirim data perubahan *id card* pada tabel *web API* tabel 3.1. *Request* perubahan *id card* dapat dilihat pada gambar 4.24.



Gambar 4.24 Request data perubahan *id card*

2. Response

Setelah *request* terkirim, *server* akan mengirimkan *response* perubahan *id card* menggunakan fungsi mengambil data perubahan *id card* tabel *web API* tabel 3.1. *Response* perubahan *id card* dapat dilihat pada gambar 4.25.



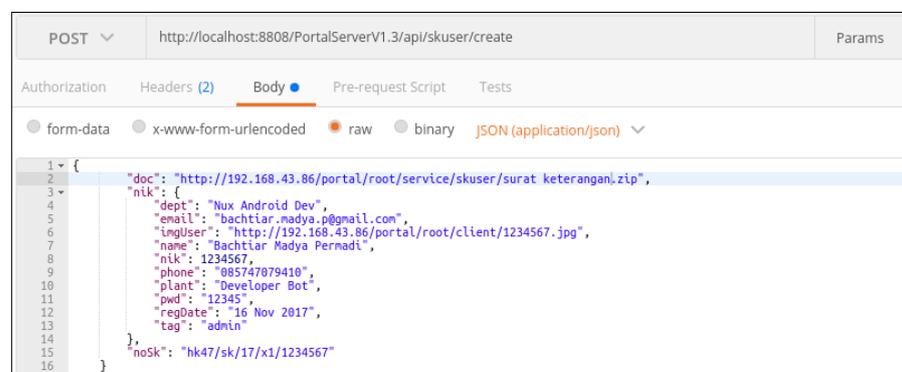
Gambar 4.25 Response data perubahan *id card*

L. Proses surat keterangan

Berdasarkan *sequence* diagram surat keterangan pada gambar 3.24, model data surat keterangan pada *list* 3.34 dan menggunakan fungsi mengirim data surat keterangan dan fungsi mengambil data surat keterangan dari tabel *web API* pada tabel 3.1, didapatkan hasil sebagai berikut:

1. Request

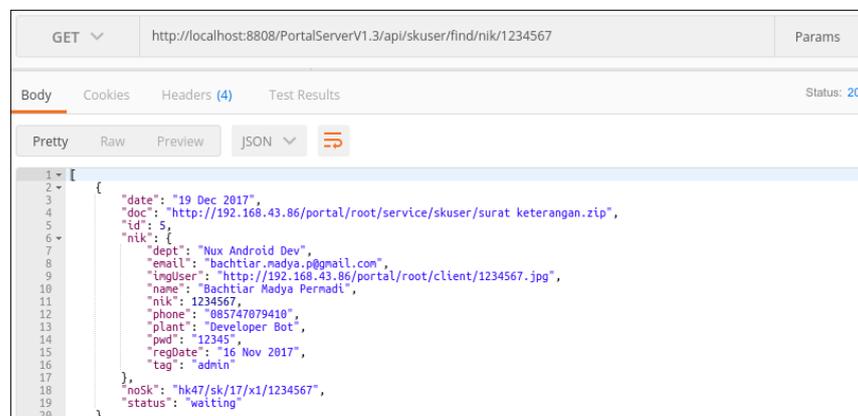
Request dikirimkan menggunakan fungsi mengirim data surat keterangan pada tabel *web API* tabel 3.1. *Request* surat keterangan dapat dilihat pada gambar 4.26.



Gambar 4.26 Request data surat keterangan

2. Response

Setelah *request* terkirim, *server* akan mengirimkan *response* surat keterangan menggunakan fungsi mengambil data surat keterangan tabel *web API* tabel 3.1. *Response* surat keterangan dapat dilihat pada gambar 4.27.



Gambar 4.27 Response data surat keterangan

4.1.2 Hasil implementasi Aplikasi Android

Setelah aplikasi administrasi personalia selesai dibangun, maka kemudian aplikasi diimplementasikan dengan cara instalasi pada smartphone android. Berikut hasil implementasi aplikasi administrasi personalia

A. Halaman registrasi

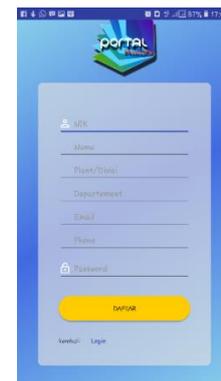
Berdasarkan halaman registrasi gambar 3.22 pada tahap perancangan maka dibuatlah halaman registrasi. Halaman register berfungsi untuk mendaftarkan akun untuk *user* baru, karena jika *user* tidak memiliki akun *user* tidak dapat menggunakan aplikasi administrasi personalia. *user* akan menginputkan data nik, nama, nomor telepon, alamat email, plant, departemen, dan password. Halaman registrasi dapat dilihat pada gambar 4.28.



Rancangan halaman registrasi yang menunjukkan form input data:

- Logo Aplikasi
- NIK:
- Nama lengkap:
- Plant:
- Departemen:
- Phone:
- Email:
- Password:
- submit pendaftaran
- LOGIN
- Daftar

Rancangan



Implementasi halaman registrasi yang menunjukkan tampilan visual:

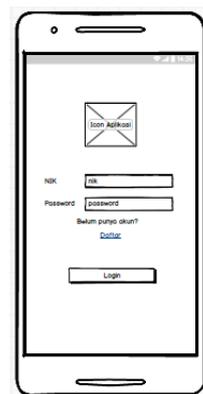
- Logo PORTAL
- NIK:
- Nama:
- Plant/Unit:
- Departemen:
- Email:
- Phone:
- Password:
- Daftar
- Sembul Login

Implementasi

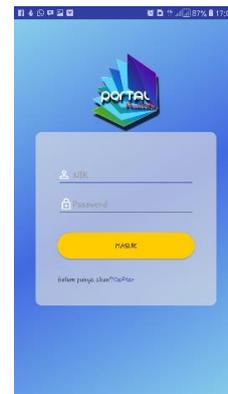
Gambar 4.28 Halaman Registrasi

B. Halaman login

Berdasarkan halaman login gambar 3.21 pada tahap perancangan maka dibuatlah halaman login. Halaman login berfungsi untuk *user* dalam melakukan login. Untuk melakukan login *user* memasukkan nik dan password yang telah dibuat. Jika login salah aplikasi akan memberikan notifikasi jika password salah. Jika password benar maka aplikasi akan menampilkan halaman beranda. Halaman login dapat dilihat pada gambar 4.29



Rancangan

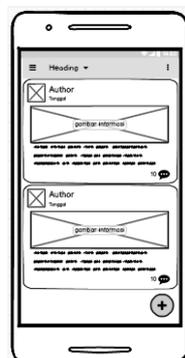


Implementasi

Gambar 4.29 Halaman *login*

C. Halaman beranda

Berdasarkan halaman beranda gambar 3.23 pada tahap perancangan maka dibuatlah halaman beranda. Pada halaman beranda berisi informasi terbaru terkait administrasi. Pada halaman beranda informasi yang ditampilkan dapat berupa gambar dan teks. Halaman beranda dapat dilihat pada gambar 4.30



Rancangan



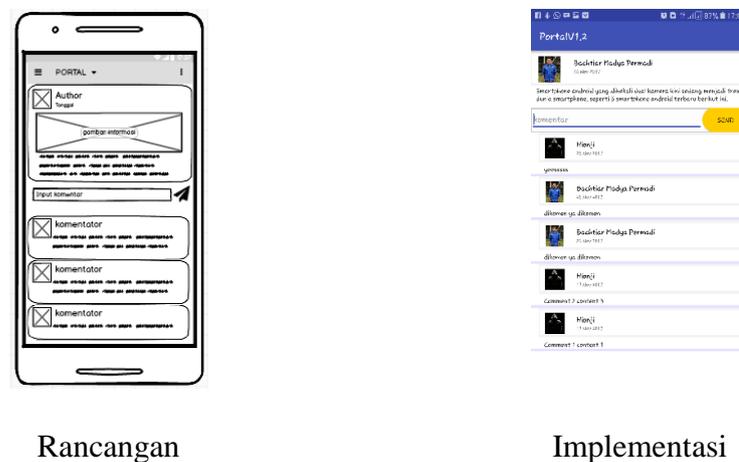
Implementasi

Gambar 4.30 Halaman beranda

D. Halaman komentar beranda

Berdasarkan halaman komentar beranda gambar 3.24 pada tahap perancangan maka dibuatlah halaman komentar beranda. Halaman komentar beranda berisi komentar yang diberikan *user* pada informasi yang ditampilkan pada halaman beranda. Pada halaman komentar beranda *user* dapat menginputkan

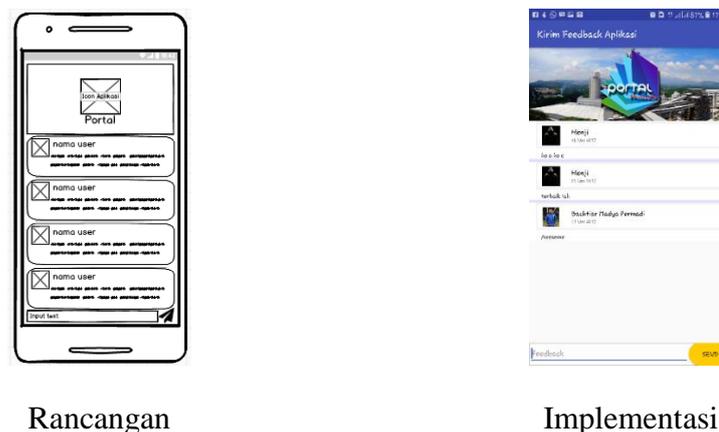
komentar hanya berupa teks. Halaman komentar beranda dapat dilihat pada gambar 4.31.



Gambar 4.31 Halaman komentar beranda

E. Halaman *feedback*

Berdasarkan halaman *feedback* gambar 3.27 pada tahap perancangan maka dibuatlah halaman *feedback*. Halaman *feedback* berisi data *feedback* yang diberikan *user*. Pada halaman *feedback* *user* dapat memberikan *feedback* berupa teks. Halaman *feedback* dapat dilihat pada gambar 4.32

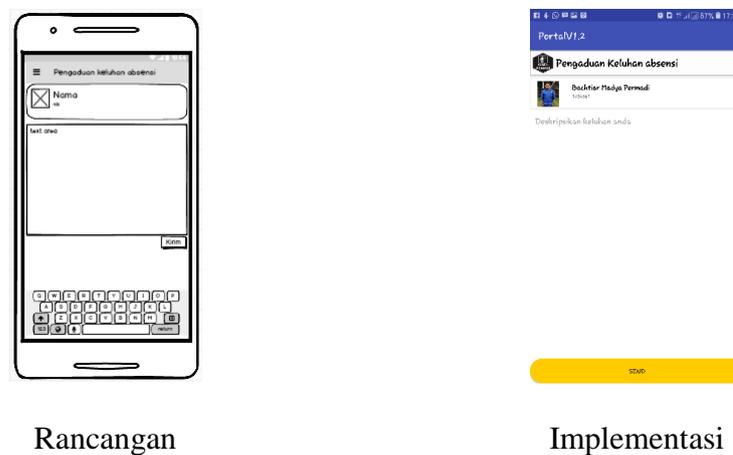


Gambar 4.32 Halaman *feedback*

F. Halaman keluhan absensi

Berdasarkan halaman keluhan absensi gambar 3.30 pada tahap perancangan maka dibuatlah halaman keluhan absensi. Pada halaman keluhan absensi *user* dapat

menginputkan keluhan mengenai absensi berupa teks. Halaman keluhan absensi dapat dilihat pada gambar 4.33.



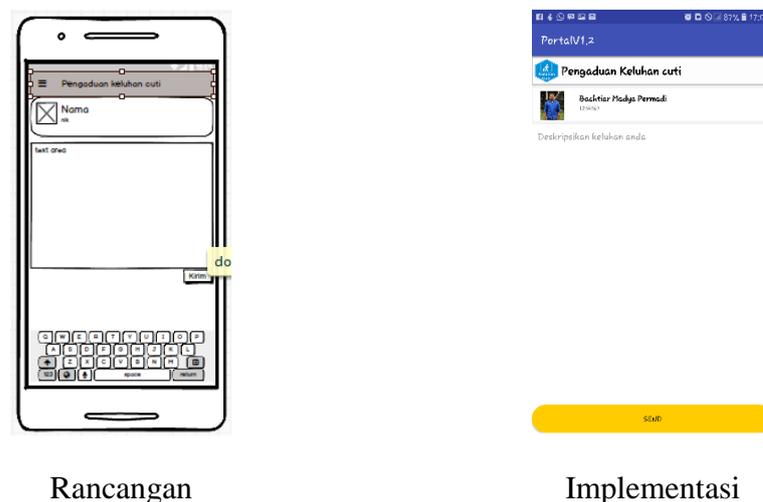
Rancangan

Implementasi

Gambar 4.33 Halaman keluhan absensi

G. Halaman keluhan cuti

Berdasarkan halaman keluhan cuti gambar 3.31 pada tahap perancangan maka dibuatlah halaman keluhan cuti. Pada halaman keluhan cuti *user* dapat menginputkan keluhan mengenai keluhan cuti berupa teks. Halaman keluhan cuti dapat dilihat pada gambar 4.34.



Rancangan

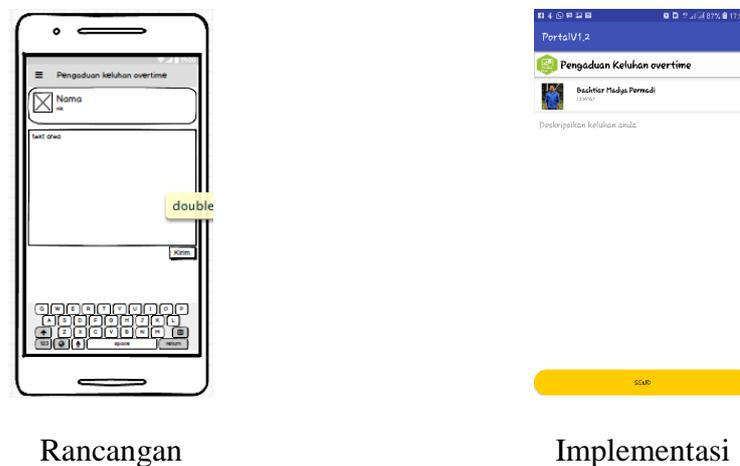
Implementasi

Gambar 4.34 Halaman keluhan cuti

H. Halaman keluhan overtime

Berdasarkan halaman keluhan *overtime* gambar 3.32 pada tahap perancangan maka dibuatlah halaman keluhan *overtime*. Pada halaman keluhan

overtime user dapat menginputkan keluhan mengenai *overtime* berupa teks. Halaman keluhan *overtime* dapat dilihat pada gambar 4.35.



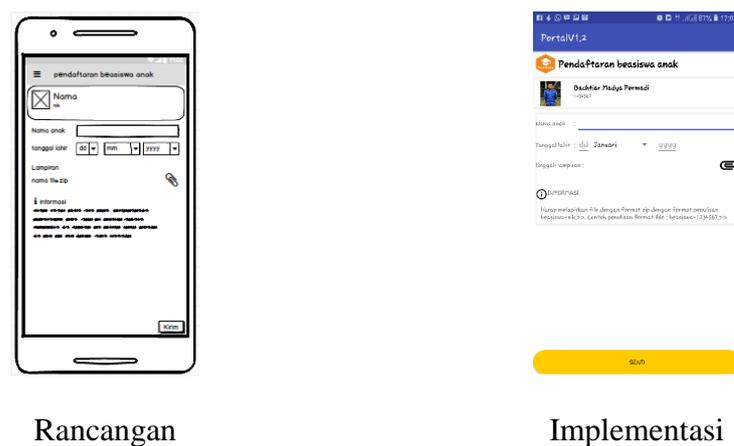
Rancangan

Implementasi

Gambar 4.35 Halaman keluhan *overtime*

I. Halaman pendaftaran beasiswa anak

Berdasarkan halaman pendaftaran beasiswa anak gambar 3.33 pada tahap perancangan maka dibuatlah halaman pendaftaran beasiswa anak. Pada halaman pendaftaran beasiswa anak, *user* dapat menginputkan nama anak dan tanggal lahir anak dan juga melampirkan dokumen yang dibutuhkan dalam pendaftaran beasiswa. Halaman pendaftaran beasiswa dapat dilihat pada gambar 4.36.



Rancangan

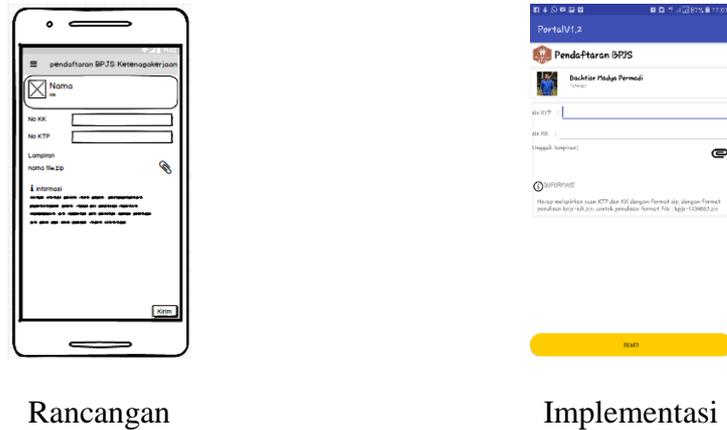
Implementasi

Gambar 4.36 Halaman pendaftaran beasiswa anak

J. Halaman pendaftaran BPJS

Berdasarkan halaman pendaftaran BPJS gambar 3.34 pada tahap perancangan maka dibuatlah halaman pendaftaran BPJS. Pada halaman pendaftaran

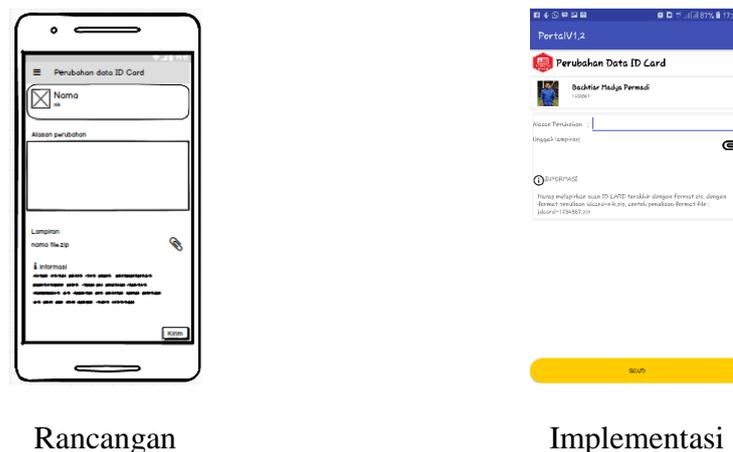
BPJS *user* dapat menginputkan nomor KTP dan nomor KK dan juga melampirkan dokumen yang dibutuhkan dalam pendaftaran BPJS. Halaman pendaftaran BPJS dapat dilihat pada gambar 4.37.



Gambar 4.37 Halaman pendaftaran BPJS

K. Halaman perubahan data *id card*

Berdasarkan halaman perubahan *id card* gambar 3.35 pada tahap perancangan maka dibuatlah halaman perubahan *id card*. Pada halaman perubahan *id card*, *user* dapat menginputkan alasan perubahan *id card* dan melampirkan *id card* lama. Halaman perubahan *id card* dapat dilihat pada gambar 4.38.



Gambar 4.38 Halaman perubahan *id card*

L. Halaman surat keterangan

Berdasarkan halaman surat keterangan gambar 3.36 pada tahap perancangan maka dibuatlah halaman surat keterangan. Pada halaman surat keterangan, *user*

dapat menginputkan jenis surat dan melampirkan surat yang akan di ajukan. Halaman surat keterangan dapat dilihat pada gambar 4.39



Gambar 4.39 Halaman surat keterangan

M. Halaman riwayat pengiriman data

Berdasarkan halaman riwayat gambar 3.37 pada tahap perancangan maka dibuatlah halaman riwayat. Halaman riwayat pengiriman berisi data riwayat pengiriman keluhan absensi, riwayat pengiriman keluhan cuti, riwayat pengiriman keluhan *overtime*, riwayat pendaftaran beasiswa, riwayat pendaftaran BPJS, riwayat perubahan *id card*, dan riwayat pengajuan SK. Halaman riwayat pengiriman dapat dilihat pada gambar 4.40.





halaman riwayat
pendaftaran beasiswa



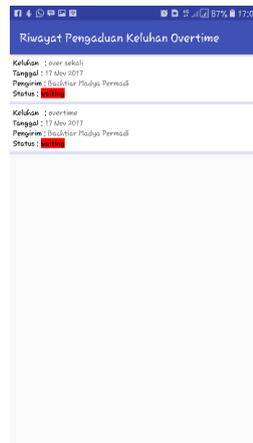
halaman riwayat
pendaftaran BPJS



Halaman riwayat
permintaan perubahan *id*
card



Halaman riwayat
pengajuan SK



Halaman riwayat keluhan
absensi

Gambar 4.40 Halaman riwayat pengiriman

4.1.3 Pengujian Aplikasi Android

Pengujian aplikasi android administrasi personalia dilakukan dengan mencoba semua halaman aplikasi beserta fungsi-fungsi di dalamnya. Pengujian dilakukan untuk mengetahui adanya kesalahan pada aplikasi. Metode pengujian yang digunakan adalah metode *black box* untuk menguji setiap halaman pada aplikasi. Berikut hasil pengujian aplikasi administrasi personalia berbasis android.

A. Pengujian halaman registrasi

Pengujian halaman registrasi bertujuan untuk mengetahui fungsionalitas dari elemen–elemen *interface* yang terdapat pada halaman registrasi. Elemen yang diujikan *textbox*, *spinner* dan *button*, serta rotasi halaman registrasi pada aplikasi. Hasil dari pengujian halaman registrasi dapat dilihat pada tabel 4.1.

Tabel 4.1 Tabel pengujian halaman registrasi

No	Kasus uji	Sebelum	Hasil yang diharapkan	Hasil yang didapat	Sesudah	status
1	Field pada halaman registrasi		Dapat menginputkan data registrasi	user dapat memasukan data untuk registrasi		Berhasil
2	Tombol daftar pada halaman registrasi		Dapat mengirimkan data registrasi	data registrasi user terkirim dan menampilkan pesan		Berhasil
3	Tekan daftar dengan field kosong		Menampilkan pesan field kosong	muncul informasi field masih kosong		Berhasil
4	Daftar dengan akun sudah terdaftar		Menampilkan pesan akun sudah terdaftar	muncul pesan akun sudah terdaftar		Berhasil

5	Rotasi halaman		<i>Field</i> tetap rapi	Setelah dilakukan rotasi, <i>field</i> tetap rapi		Berhasil
---	----------------	-----------------------------------------------------------------------------------	-------------------------	---------------------------------------------------	-------------------------------------------------------------------------------------	----------

B. Pengujian halaman login

Pengujian pada halaman *login* bertujuan untuk mengetahui fungsionalitas dari elemen-elemen *interface* yang terdapat pada halaman *login*. Elemen yang diujikan adalah *textbox*, *button*, *hide and show password*, rotasi halaman *login* pada aplikasi dan validasi data *login*. Hasil pengujian halaman login dapat dilihat pada tabel 4.2.

Tabel 4.2 Tabel pengujian halaman *login*

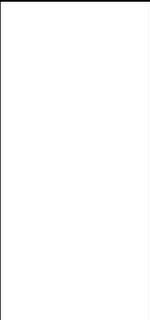
No	Kasus uji	Sebelum	Hasil yang diharapkan	Hasil yang didapat	Sesudah	status
1	<i>Field</i> pada halaman <i>login</i>		Dapat menginputkan <i>nik</i> dan <i>password</i> untuk <i>login</i>	User dapat menginputkan <i>nik</i> dan <i>password</i>		Berhasil
2	Tombol <i>show</i> dan <i>hide password</i>		Dapat menampilkan dan menyembunyikan <i>password</i>	menampilkan dan menyembunyikan <i>password</i>		Berhasil
3	Tombol <i>login</i> pada halaman <i>login</i>		Dapat mengirim data <i>login</i>	data <i>login</i> dapat terkirim dan menuju ke halaman beranda		Berhasil

4	<i>Login dengan password / nik salah</i>		Menampilkan pesan nik/ <i>password</i> salah	muncul peringatan nik atau <i>password</i> salah		Berhasil
5	<i>Login dengan nik belum terdaftar</i>		Menampilkan pesan akun belum terdaftar	muncul peringatan akun belum terdaftar		Berhasil
6	<i>Login dengan nik belum di validasi</i>		Tidak dapat melakukan <i>login</i>	<i>User</i> tidak dapat melakukan <i>login</i> dan aplikasi menampilkan informasi		Berhasil
7	Rotasi halaman <i>login</i>		<i>Field</i> tetap rapi	Setelah dilakukan rotasi <i>field</i> tetap rapi		Berhasil

C. Pengujian halaman beranda

Pengujian halaman beranda bertujuan untuk mengetahui fungsionalitas elemen-elemen *interface* yang terdapat pada halaman beranda. Elemen yang di ujikan yaitu *list* data beranda, *swipe* untuk *merefresh* data dan rotasi halaman beranda pada aplikasi. Hasil pengujian halaman beranda dapat dilihat pada tabel 4.3.

Tabel 4.3 Tabel pengujian halaman beranda

No	Kasus uji	Sebelum	Hasil yang diharapkan	Hasil yang didapat	Sesudah	status
1	Menampilkan halaman beranda		Menampilkan halaman beranda	berhasil melakukan <i>login</i> dan menampilkan halaman beranda		Berhasil
2	Refresh informasi pada halaman beranda		Data informasi diperbaharui jika mengusap layar	Setelah <i>user</i> mengusap layar, data informasi dapat ditampilkan		Berhasil
3	Rotasi pada halaman		Halaman tetap rapi	Setelah dilakukan rotasi, halaman tetap rapi		Berhasil

D. Pengujian halaman komentar

Pengujian halaman komentar bertujuan untuk mengetahui fungsionalitas elemen-elemen *interface* yang terdapat pada halaman komentar. Elemen yang diujikan yaitu *list* data komentar, *swipe* halaman untuk *refresh* data, *textbox*, *button*, dan rotasi halaman komentar pada aplikasi. Hasil pengujian halaman komentar dapat dilihat pada tabel 4.4.

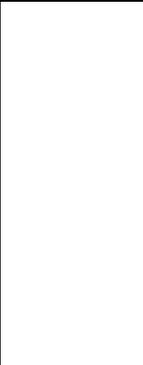
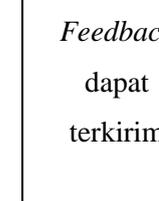
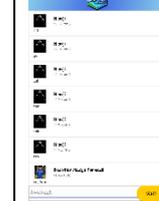
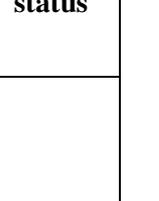
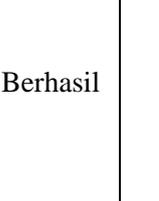
Tabel 4.4 Tabel pengujian halaman komentar

No	Kasus uji	Sebelum	Hasil yang diharapkan	Hasil yang didapat	Sesudah	status
1	Menampilk an halaman komentar		menampilkan komentar dari <i>user</i>	Setelah informasi di klik halaman komentar dapat dimunculkan		Berhasil
2	Tombol kirim komentar pada informasi		Komentar dari <i>user</i> dapat terkirim pada informasi	Komentar terkirim dan komentar diperbaharui		Berhasil
3	<i>Refresh</i> komentar pada halaman komentar		Data komentar diperbaharui jika <i>user</i> mengusap layar / <i>swap</i>	Setelah layar di <i>swap</i> /diusap data komentar dapat diperbaharui		Berhasil
4	Kirim komentar dengan <i>field</i> kosong		Tidak ada komentar terkirim	Setelah tombol kirim di tekan komentar tidak terkirim		Berhasil
5	Rotasi halaman		Halaman tetap rapi	Setelah dilakukan rotasi halaman tetap rapi		Berhasil

E. Pengujian halaman *feedback*

Pengujian halaman *feedback* bertujuan untuk mengetahui fungsionalitas elemen-elemen *interface* yang terdapat pada halaman *feedback*. Elemen yang diujikan yaitu *textbox*, *button*, *swipe*, *list data feedback*, dan rotasi halaman *feedback* pada aplikasi. Hasil pengujian halaman *feedback* dapat dilihat pada tabel 4.5.

Tabel 4.5 Tabel pengujian halaman *feedback*

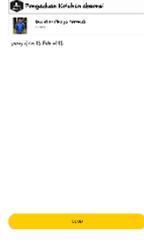
No	Kasus uji	Sebelum	Hasil yang diharapkan	Hasil yang didapat	Sesudah	status
1	Menampilk an data <i>feedback</i>		Data <i>feedback</i> ditampilkan saat aplikasi menampilkan halaman <i>feedback</i>	halaman <i>feedback</i> ditampilkan		Berhasil
2	<i>Field</i> pada halaman <i>feedback</i>		Dapat menginputka n pesan <i>feedback</i>	<i>User</i> dapat menginputka n pesan <i>feedback</i>		Berhasil
3	Tombol kirim pada halaman <i>feedback</i>		<i>feedback</i> dapat terkirim	<i>Feedback</i> dapat terkirim		Berhasil
4	Kirim dengan <i>field</i> kosong		Tidak ada data terkirim	Setelah tombol kirim di klik tidak		Berhasil

				ada data terkirim		
5	Rotasi halaman		Halaman tetap rapi	Setelah dilakukan rotasi, halaman tetap rapi		Berhasil

F. Pengujian halaman pengaduan keluhan absensi

Pengujian halaman pengaduan keluhan absensi uhan absensi bertujuan untuk mengetahui fungsionalitas elemen-elemen *interface* yang terdapat pada halaman pengaduan keluhan absensi. Elemen yang diuji yaitu *textbox*, *button*, dan rotasi halaman pada aplikasi. Hasil pengujian halaman pengaduan keluhan absensi dapat dilihat pada tabel 4.6.

Tabel 4.6 Tabel pengujian halaman pengaduan keluhan absensi

No	Kasus uji	Sebelum	Hasil yang diharapkan	Hasil yang didapat	Sesudah	status
1	Field pada halaman keluhan absensi		dapat menginputkan keluhan absensi	Setelah di klik user dapat menginputkan keluhan absensi		Berhasil
2	Tombol kirim pada halaman keluhan absensi		Keluhan dapat terkirim	Setelah tombol kirim di klik data keluhan absensi user dapat terkirim		Berhasil

3	Tombol kirim dengan <i>field</i> kosong		Tidak ada data terkirim	Setelah tombol kirim di klik tidak ada data terkirim		Berhasil
4	Menampilkan riwayat pengiriman		Menampilkan riwayat pengiriman	Setelah keluhan terkirim, aplikasi menampilkan riwayat pengiriman		Berhasil
5	Rotasi halaman		<i>Field</i> tetap rapi	Setelah dilakukan rotasi, <i>field</i> tetap rapi		Berhasil
6	Rotasi halaman riwayat		<i>Field</i> tetap rapi	Setelah dilakukan rotasi halaman tetap rapi		Berhasil

G. Pengujian halaman pengaduan keluhan cuti

Pengujian halaman pengaduan keluhan cuti bertujuan untuk mengetahui fungsionalitas elemen-elemen *interface* yang terdapat pada halaman pengaduan keluhan cuti. Elemen yang diuji yaitu *textbox*, *button*, dan rotasi halaman pada aplikasi. Hasil pengujian halaman pengaduan keluhan cuti dapat dilihat pada tabel 4.7.

Tabel 4.7 Tabel pengujian halaman pengaduan keluhan cuti

No	Kasus uji	Sebelum	Hasil yang diharapkan	Hasil yang didapat	Sesudah	status
1	Field pada halaman keluhan cuti		dapat menginputkan data keluhan cuti	Setelah di klik <i>User</i> dapat menginputkan data keluhan absensi		Berhasil
2	Tombol kirim pada halaman keluhan cuti		Data dapat terkirim saat tombol kirim di tekan	Setelah tombol kirim di klik data keluhan cuti <i>user</i> dapat terkirim		Berhasil
3	Tombol kirim dengan field kosong		Tidak ada data terkirim	Setelah tombol kirim di klik tidak ada data terkirim		Berhasil
4	Menampilkan riwayat pengiriman		Menampilkan riwayat pengiriman	Setelah keluhan terkirim, aplikasi menampilkan riwayat pengiriman		Berhasil

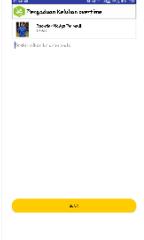
5	Rotasi halaman		Field tetap rapi	Setelah dilakukan rotasi, field tetap rapi		Berhasil
6	Rotasi halaman riwayat		Field tetap rapi	Setelah dilakukan rotasi halaman tetap rapi		Berhasil

H. Pengujian halaman pengaduan keluhan overtime

Pengujian halaman pengaduan keluhan *overtime* bertujuan untuk mengetahui fungsionalitas elemen-elemen *interface* yang terdapat pada halaman pengaduan keluhan *overtime*. Elemen yang diuji yaitu *textbox*, *button*, dan rotasi halaman pada aplikasi. Hasil pengujian halaman pengaduan keluhan *overtime* dapat dilihat pada tabel 4.8.

Tabel 4.8 Tabel pengujian halaman pengaduan keluhan *overtime*

No	Kasus uji	Sebelum	Hasil yang diharapkan	Hasil yang didapat	Sesudah	status
1	Field pada halaman keluhan overtime		dapat menginputkan data keluhan overtime	Setelah di klik user dapat menginputkan data keluhan overtime		Berhasil
2	Tombol kirim pada halaman		Data dapat terkirim saat tombol kirim di tekan	Setelah tombol kirim di klik data keluhan		Berhasil

	keluhan <i>overtime</i>			<i>overtime</i> user dapat terkirim		
3	Tombol kirim dengan <i>field</i> kosong		Tidak ada data terkirim	Setelah tombol kirim di klik tidak ada data terkirim		Berhasil
4	Menampilka an riwayat pengiriman		Menampilkan riwayat pengiriman	Setelah keluhan terkirim, aplikasi menampilka n riwayat pengiriman		Berhasil
5	Rotasi halaman		<i>Field</i> tetap rapi	Setelah dilakukan rotasi, <i>field</i> tetap rapi		Berhasil
6	Rotasi halaman riwayat		<i>Field</i> tetap rapi	Setelah dilakukan rotasi halaman tetap rapi		Berhasil

I. Pengujian halaman pendaftaran beasiswa

Pengujian halaman pendaftaran beasiswa bertujuan untuk mengetahui fungsionalitas elemen-elemen *interface* yang terdapat pada halaman pendaftaran beasiswa. Elemen yang diuji yaitu *textbox*, *spinner*, *button*, dan rotasi halaman pada aplikasi. Hasil pengujian halaman pendaftaran beasiswa dapat dilihat pada tabel 4.9.

Tabel 4.9 Tabel pengujian halaman pendaftaran beasiswa

No	Kasus uji	Sebelum	Hasil yang diharapkan	Hasil yang didapat	Sesudah	status
1	Field pada halaman pendaftaran beasiswa		Dapat menginputkan data pendaftaran beasiswa	User dapat menginputkan data pendaftaran beasiswa		Berhasil
2	Tombol lampiran pada halaman pendaftaran beasiswa		Dapat membuka halaman memory smartphone	menampilkan halaman data memory pada smartphone		Berhasil
3	Tombol kirim dengan field kosong		Tidak ada data terkirim	Setelah tombol kirim di klik tidak ada data terkirim		Berhasil
4	Tombol kirim pada halaman pendaftaran beasiswa		Dapat mengirim data pendaftaran beasiswa	Setelah tombol kirim di klik, data pendaftaran dapat terkirim		Berhasil
5	Menampilkan riwayat pengiriman		Menampilkan riwayat pengiriman	Setelah keluhan terkirim, aplikasi menampilkan		Berhasil

				n riwayat pengiriman		
6	Rotasi halaman		<i>Field</i> tetap rapi	Setelah dilakukan rotasi, <i>field</i> tetap rapi		Berhasil
7	Rotasi halaman riwayat		<i>Field</i> tetap rapi	Setelah dilakukan rotasi halaman tetap rapi		Berhasil
8	Pengiriman lampiran		Lampiran terkirim	File lampiran terkirim		Berhasil

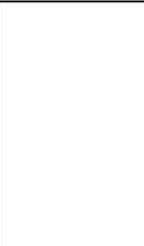
J. Pengujian halaman pendaftaran BPJS

Pengujian halaman pendaftaran BPJS bertujuan untuk mengetahui fungsionalitas elemen-elemen *interface* yang terdapat pada halaman pendaftaran BPJS. Elemen yang diuji yaitu *textbox*, *button*, dan rotasi halaman pada aplikasi. Hasil pengujian halaman pendaftaran BPJS dapat dilihat pada tabel 4.10.

Tabel 4.10 Tabel pengujian halaman pendaftaran BPJS

No	Kasus uji	Sebelum	Hasil yang diharapkan	Hasil yang didapat	Sesudah	status
1	<i>Field</i> pada halaman pendaftaran BPJS		Dapat menginputkan data pendaftaran BPJS	Setelah di klik <i>user</i> dapat menginputkan data		Berhasil

				pendaftaran BPJS		
2	Tombol lampiran pada halaman pendaftaran BPJS		Dapat membuka halaman <i>memory smartphone</i>	Setelah di klik aplikasi menampilkan halaman data <i>memory smartphone</i>		Berhasil
3	Tombol kirim pada halaman pendaftaran BPJS		Dapat mengirim data pendaftaran BPJS	Setelah tombol kirim di klik, data pendaftaran dapat terkirim		Berhasil
4	Tombol kirim dengan <i>field</i> kosong		Tidak ada data terkirim	Setelah tombol kirim di klik tidak ada data terkirim		Berhasil
5	Menampilkan riwayat pengiriman		Menampilkan riwayat pengiriman	Setelah data terkirim, aplikasi menampilkan riwayat pengiriman		Berhasil
6	Rotasi halaman		<i>Field</i> tetap rapi	Setelah dilakukan rotasi, <i>field</i> tetap rapi		Berhasil

7	Rotasi halaman riwayat		Field tetap rapi	Setelah dilakukan rotasi halaman tetap rapi		Berhasil
8	Pengiriman lampiran		Lampiran terkirim	File lampiran terkirim		Berhasil

K. Pengujian halaman perubahan Id card

Pengujian halaman perubahan *id card* bertujuan untuk mengetahui fungsionalitas elemen-elemen *interface* yang terdapat pada halaman perubahan *id card*. Elemen yang diuji yaitu *textbox*, *button*, dan rotasi halaman pada aplikasi. Hasil pengujian halaman perubahan *id card* dapat dilihat pada tabel 4.11.

Tabel 4.11 Tabel pengujian halaman perubahan *id card*

No	Kasus uji	Sebelum	Hasil yang diharapkan	Hasil yang didapat	Sesudah	status
1	Field pada halaman perubahan <i>id card</i>		Dapat menginputkan data perubahan <i>id card</i>	Setelah di klik <i>User</i> dapat menginputkan data perubahan <i>id card</i>		Berhasil
2	Tombol lampiran pada halaman		Dapat membuka halaman <i>memory smartphone</i>	Setelah di klik aplikasi menampilkan halaman data <i>memory</i>		Berhasil

	perubahan <i>id card</i>			pada <i>smartphone</i>		
3	Tombol kirim pada halaman perubahan <i>id card</i>		Dapat mengirim data perubahan <i>id card</i>	Setelah tombol kirim di klik, data permintaan dapat terkirim		Berhasil
4	Tombol kirim dengan <i>field</i> kosong		Tidak ada data terkirim	Setelah tombol kirim di klik tidak ada data terkirim		Berhasil
5	Menampilkan riwayat pengiriman		Menampilkan riwayat pengiriman	Setelah keluhan terkirim, aplikasi menampilkan riwayat pengiriman		Berhasil
6	Rotasi halaman		<i>Field</i> tetap rapi	Setelah dilakukan rotasi, <i>field</i> tetap rapi		Berhasil
7	Rotasi halaman riwayat		<i>Field</i> tetap rapi	Setelah dilakukan rotasi halaman tetap rapi		Berhasil

8	Pengiriman lampiran		Lampiran terkirim	File lampiran terkirim		Berhasil
---	---------------------	--	-------------------	------------------------	-------------------------------------------------------------------------------------	----------

L. Pengujian halaman surat keterangan

Pengujian halaman surat keterangan bertujuan untuk mengetahui fungsionalitas elemen-elemen *interface* yang terdapat pada halaman surat keterangan. Elemen yang diuji yaitu *textbox*, *button*, dan rotasi halaman pada aplikasi. Hasil pengujian halaman surat keterangan dapat dilihat pada tabel 4.12.

Tabel 4.12 Tabel pengujian halaman surat keterangan

No	Kasus uji	Sebelum	Hasil yang diharapkan	Hasil yang didapat	Sesudah	status
1	Field pada halaman surat keterangan		Dapat menginputkan data surat keterangan	Setelah di klik <i>user</i> dapat menginputkan surat keterangan		Berhasil
2	Tombol lampiran pada halaman surat keterangan		Dapat membuka halaman <i>memory smartphone</i>	Setelah di klik aplikasi menampilkan halaman data <i>memory smartphone</i>		Berhasil
3	Tombol kirim pada halaman		Dapat mengirim data surat keterangan	Setelah tombol kirim di klik, data surat		Berhasil

	surat keterangan			keterangan dapat terkirim		
4	Tombol kirim dengan <i>field</i> kosong		Tidak ada data terkirim	Setelah tombol kirim di klik tidak ada data terkirim		Berhasil
5	Menampilka an riwayat pengiriman		Menampilkan riwayat pengiriman	Setelah keluhan terkirim, aplikasi menampilka n riwayat pengiriman		Berhasil
6	Rotasi halaman		<i>Field</i> tetap rapi	Setelah dilakukan rotasi, <i>field</i> tetap rapi		Berhasil
7	Rotasi halaman riwayat		<i>Field</i> tetap rapi	Setelah dilakukan rotasi halaman tetap rapi		Berhasil
8	Pengiriman lampiran		Lampiran terkirim	File lampiran terkirim		Berhasil

4.1.4 Pengujian aplikasi berdasarkan spesifikasi android

Pengujian aplikasi berdasarkan spesifikasi android bertujuan untuk mengetahui aplikasi dapat berjalan pada *smartphone* dengan spesifikasi yang berbeda. Hasil pengujian aplikasi berdasarkan spesifikasi android dapat dilihat pada tabel 4.13.

Tabel 4.13 Pengujian aplikasi berdasarkan spesifikasi android

Sistem operasi	Memory	Ram	Status
Android Jelly Bean	8 GB	1 GB	Berhasil
Android Kitkat	8 GB	1 GB	Berhasil
Android Lolipop	8 GB	2 GB	Berhasil
Android Marsmellow	16 GB	3 GB	Berhasil
Android Nougat	16 GB	3 GB	Berhasil

4.2 Pembahasan

Berdasarkan hasil pengujian aplikasi administrasi personalia berbasis android yang telah dilakukan, berikut hasil yang didapatkan:

1. Berdasarkan hasil pengujian pada halaman registrasi, *user* bukan pegawai tidak bisa mendaftarkan akun karena setiap data pendaftaran akan dilakukan validasi oleh admin. *User* juga tidak dapat membuat akun ganda karena setiap 1 nik hanya dapat digunakan untuk 1 akun.
2. Berdasarkan hasil pengujian halaman *login*, *user* tidak dapat melakukan *login* jika nik atau *password* yang dimasukkan salah.
3. Berdasarkan pengujian halaman beranda. *User* dapat menerima informasi dengan cepat.
4. Berdasarkan pengujian halaman komentar. *User* dapat memberikan komentar untuk informasi.
5. Berdasarkan pengujian halaman *feedback*. *User* dapat mengirimkan *feedback* untuk aplikasi administrasi personalia.

6. Berdasarkan pengujian halaman keluhan absensi. *User* dapat mengirimkan keluhan absensi dari lokasi user tersebut ditempatkan.
7. Berdasarkan pengujian halaman keluhan cuti. *User* dapat mengirimkan keluhan cuti dari lokasi user tersebut ditempatkan.
8. Berdasarkan pengujian halaman keluhan *overtime*. *User* dapat mengirimkan keluhan *overtime* dari lokasi user tersebut ditempatkan.
9. Berdasarkan pengujian halaman pendaftaran beasiswa. *User* dapat melakukan pendaftaran beasiswa dari lokasi *user* ditempatkan dan juga dapat mengirimkan lampiran yang dibutuhkan.
10. Berdasarkan pengujian halaman pendaftaran BPJS. *User* dapat melakukan pendaftaran BPJS dari lokasi *user* ditempatkan dan juga dapat mengirimkan lampiran yang dibutuhkan.
11. Berdasarkan pengujian halaman perubahan *id card*. *User* dapat melakukan permintaan perubahan *id card* dari lokasi *user* ditempatkan dan juga dapat mengirimkan lampiran yang dibutuhkan.
12. Berdasarkan pengujian halaman surat keterangan. *User* dapat melakukan pengiriman surat keterangan dari lokasi user ditempatkan dan juga dapat mengirimkan lampiran yang dibutuhkan.