

BAB V

HASIL PENELITIAN DAN PEMBAHASAN

5.1 Implementasi Sistem

Pada bagian Implementasi Sistem, akan dibahas bagaimana hasil dari perancangan yang dibuat pada tahap sebelumnya. Bagian implementasi ini juga berfungsi untuk menjelaskan fungsi dari hasil yang didapat.

5.1.1 Implementasi Antarmuka Login Portal Ormawa



Gambar 5.1. Antarmuka Login

Saat pertama kali situs Portal Ormawa diakses, halaman yang ditampilkan adalah halaman *login*. Komponen yang ada di halaman *login* adalah sebagai berikut :

1. *Icon dan Heading*
2. *Field Text* untuk mengisi *username* dan *password*
3. *Button login*
4. *Button login google*

Pada proses di halaman *login*, *button login* berfungsi untuk melakukan akses masuk sistem menggunakan *username* dan *password* yang terdaftar pada *database*. Sedangkan *button login*

google berfungsi untuk pengecekan email yang terdaftar pada *database* dan *google*, sehingga mempermudah user untuk masuk.

Jadi sistem Portal Ormawa menggunakan dua metode *login*. *Login* dapat dilakukan secara biasa atau menggunakan email yang sudah terdaftar pada akun.

```
async function login(username: string, password: string) {
  console.log(username, password);
  const resp = await axios.post(API + "/login", { username
, password });
  return resp.data;
}

async function loginGoogle(token: string) {
  console.log(token);
  const resp = await axios.post(API + "/login/google", { t
oken });
  return resp.data;
}
```

Gambar 5.2. Script Kode Untuk Proses Pemanggilan API Login

```
return (
  <Container component="main" maxWidth="xs">
    <CssBaseline />
    <div className={classes.paper}>
      <Avatar className={classes.avatar}>
        <LockOutlinedIcon />
      </Avatar>
      <Typography component="h1" variant="h5" className=
{classes.bold}>
        PORTAL ORMAWA
      </Typography>
      <form className={classes.form} onSubmit={handleFor
mSubmit}>
        <TextField
          variant="outlined"
          margin="normal"
          required
          fullWidth
          id="username"
          label="Username"
          name="username"
          autoComplete="username"
          autoFocus
```

```

        value={usernameInput}
        onChange={e => setUsernameInput(e.target.value
    )}

    />
    <TextField
      variant="outlined"
      margin="normal"
      required
      fullWidth
      name="password"
      label="Password"
      type="password"
      id="password"
      autoComplete="current-password"
      value={passwordInput}
      onChange={e => setPasswordInput(e.target.value
    )}

    />
    <Button
      type="submit"
      fullWidth
      variant="contained"
      color="primary"
      disabled={loading}
      className={classes.submit}
    >
      {
        !loading
          ? "Login"
          : "Logging in"
      }
    </Button>
    <Typography align="center">
      OR
    </Typography>
    <div style={{ display: "flex" }} id="google-
button-container">
      <GoogleLogin
        clientId={process.env.REACT_APP_CLIENT_CODE
    || ""}

        onRequest={onRequestAction}
        disabled={loading}
        isSignedIn={true}

        // @ts-ignore
        onSuccess={e => loginActionGoogle(e.tokenId)
    }

```

```

        style={{
            flexGrow: 1,
            justifyContent: "center",
            alignItems: "center"
        }}
        onFailure={loginActionGoogleFail}
    />
</div>
<div>

</div>

```

Gambar 5.3. Script Kode Untuk Proses Antarmuka Login

5.1.2 Implementasi Antarmuka Menu Home Portal Ormawa

Proses setelah melakukan *login* adalah menampilkan halaman *home*. Home berfungsi sebagai halaman utama setiap *user* yang login. Pada halaman utama terdapat informasi login dengan kategori, juga terdapat agenda tahunan semua organisasi, manfaat agenda itu sendiri adalah berfungsi sebagai informasi setiap acara yang dimiliki oleh organisasi mahasiswa dan dapat dipantau perkembangannya. Setiap organisasi dapat menambahkan agenda yang mereka miliki tetapi admin hanya bertugas melihat data agenda, agenda ini juga menampilkan dari tanggal terdahulu sampai dengan yang terbelakang. Kolom search berfungsi untuk menyaring username yang akan dilihat.

ID	Nama	Username	Tanggal Mulai	Tanggal Selesai	Tempat	Deskripsi	Status	Tanggal Input	Tanggal Status	Tanggal Selesai Acara
3	Acara BEMF 1	bem_fakultas.ikom	2/3/2021, 9:00:11 AM	2/3/2021, 3:00:11 PM	Kampus	Acara keakraban	belum dilaksanakan	2/3/2021, 10:46:07 AM	-	-
1	Acara HM 1	hmpssi	2/3/2021, 10:49:29 AM	2/6/2021, 2:00:00 PM	Kampus	Acara perayaan ulang tahun prodi	sedang dilaksanakan	2/3/2021, 10:43:34 AM	2/3/2021, 10:49:29 AM	-
4	Acara UKM 1	ukm_futsal	2/25/2021, 8:00:00 AM	2/25/2021, 4:00:00 PM	Luar Kampus	Turnamen antar SD se Jateng	belum dilaksanakan	2/3/2021, 10:47:30 AM	-	-
2	Acara HM 2	hmpssi	4/14/2021, 12:00:00 PM	4/14/2021, 4:00:00 PM	Kampus	Seminar Edukasi	belum dilaksanakan	2/3/2021, 10:44:24 AM	-	-

Gambar 5.4. Antarmuka Home

Komponen yang ada di halaman *home* adalah sebagai berikut

:

1. *Heading* yang menampilkan nama dan kode *user*

```
2. const routes: RouteDef[] = [  
3.   {  
4.     path: "/",  
5.     exact: true,  
6.     Component: () => {  
7.       const auth = useStoreState(s => s.auth);  
8.       const theme = useTheme();  
9.       return <div>  
10.        <Typography component="h1" variant="h6">  
11.          Selamat Datang,  
12.        </Typography>  
13.        <Typography component="h1" variant="h4">  
14.          <b>{auth.username}</b>.  
15.        </Typography>  
16.        <Typography style={{ marginTop: theme.spacing  
(1), color: grey[500] }}>  
17.          - {auth.ketua.toUpperCase()}  
18.        </Typography>  
19.      </div>  
20.    }  
  ]
```

Gambar 5.5. Script Kode Untuk Proses Antarmuka *Heading* 1

2. *Sidebar* untuk kumpulan menu Pada *sidebar* atau kumpulan menu diatur dengan fungsi dimana menu sudah ditentukan per user, contoh kode :

```
21. return (  
22.   <div>  
23.     <Tooltip title="Home">  
24.       <ListItem button onClick={e => history.push(""  
25.         /" )}>  
26.         <ListItemIcon>  
27.           <DashboardIcon />  
28.         </ListItemIcon>  
29.         <ListItemText primary="Home" />  
30.       </ListItem>  
31.     </Tooltip>  
32.     {  
33.       auth.ketua === "bau"
```

```

35.     ? <>
36.     <Tooltip title="Barang" >
37.     <ListItem button onClick={e => history.
push("/barang")}>
38.     <ListItemIcon>
39.     <EventSeatIcon />
40.     </ListItemIcon>
41.     <ListItemText primary="Barang" />
42.     </ListItem>
43.     </Tooltip>
44.     <Tooltip title="Peminjaman Barang">
45.     <ListItem button onClick={e => history.
push("/peminjaman-barang")}>
46.     <ListItemIcon>
47.     <DescriptionIcon />
48.     </ListItemIcon>
49.     <ListItemText primary="Peminjaman Bar
ang" />
50.     </ListItem>
51.     </Tooltip>
52.     </>
53.     : null
54.   }

```

Gambar 5.4. Script Kode Untuk Proses Antarmuka Sidebar

Pada contoh kode diatas dapat dilihat bahwa fungsi yang dimulai dengan kurung kurawal bertuliskan `auth.ketua === "bau"` yang berarti mendeteksi informasi *user* yang *login* pada sistem, jika kode *user* adalah "bau" maka menu akan memunculkan barang dan peminjaman barang.

3. Navbar

4. Agenda

```

<div>
  <Typography component="span" variant="h3">
    Agenda Tahunan
  </Typography>
</div>
<div>
  <Grid item xs>
    <Paper className={classes.rootSearch}>
      <InputBase

```

```

placeholder="Cari Agenda"
value={state.search}
className={classes.input}
onChange={e => {
  dispatch({ type: TYPES.SET_SEARCH, payload: e.target.value })
}}
inputProps={{ 'aria-label': 'cari barang' }}
/>
<IconButton
  type="button"
  onClick={e => console.log(e)}
  className={classes.iconButton}
  size="small"
  aria-label="search">
  <SearchIcon />
</IconButton>
</Paper>
</Grid>
</div>
<div>
  <Typography component="span" variant="h6">
    Selamat Datang, &nbsp;  
  </Typography>
  <Typography className={classes.usernameSpan} component="span" variant="h6">
    <b>{auth.username}</b>.
  </Typography>
  <Typography style={{ marginTop: theme.spacing(1), color: grey[500] }}>
    -
  </Typography>
  {auth?.ketua?.toLowerCase() === "fakultas.ukm" ? "UKM" : auth?.ketua?.toUpperCase()}
  </Typography>
</div>
</div>

<Divider className={classes.divider} />

<div className={classes.tableContainer}>
  <Table>
    <TableHead>
      <TableRow>
        <TableCell>ID</TableCell>
        <TableCell>Nama</TableCell>

```

```

        <TableCell>Username</TableCell>
        /* <TableCell>Nama Organisasi</TableCell> *
    /}

    <TableCell>Tanggal Mulai</TableCell>
    <TableCell>Tanggal Selesai</TableCell>
    <TableCell>Tempat</TableCell>
    <TableCell>Deskripsi</TableCell>
    <TableCell>Status</TableCell>
    <TableCell>Tanggal Input</TableCell>
    <TableCell>Tanggal Status</TableCell>
    <TableCell>Tanggal Selesai Acara</TableCell>
  </TableRow>
</TableHead>
<TableBody>
  <BuildRow rows={filtered} selected={state.selected} dispatch={dispatch} />
</TableBody>
</Table>
</div>
<div className={classes.fab}>
  {
    !username.startsWith("admin_")
    && namaKetua !== "bau"
    && namaKetua !== "bak"
    && namaKetua !== "upt"
    ? <Zoom
      in={true}
      timeout={100}
      style={{ transitionDelay: `100ms` }}
      unmountOnExit
    >
      <Fab
        color="primary"
        className={classes.mb1}
        aria-label="Add agenda"
        size="large"
        onClick={() => history.push("/add-
agenda")}
      >
        <AddIcon />
      </Fab>
    </Zoom>
    : null
  }
}

```

```
{
  state.selected
  ? state.selected?.username === auth.username
  ? <Zoom
    in={true}
    timeout={200}
    style={{ transitionDelay: `200ms` }}
    unmountOnExit
  >
    <Fab
      color="secondary"
      className={classes.mb1}
      aria-label="Edit agenda"
      size="small"
      onClick={() => history.push("/edit-
agenda/" + state.selected.id)}
    >
      <EditIcon />
    </Fab>
  </Zoom>
  : null
  : null
}
</div>
```

Gambar 5.5. Script Kode Untuk Agenda

5. Tambah Agenda

Gambar 5.6. Antarmuka Tambah Agenda

5.1.3 Implementasi Antarmuka Menu Acara Admin (Ormawa) Portal Ormawa

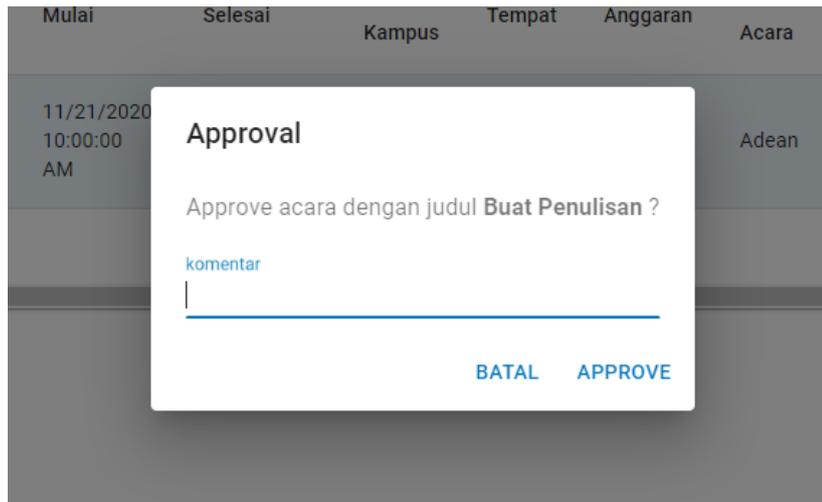
Pada Portal Ormawa terdapat tiga jenis admin yaitu program studi, fakultas, dan universitas. Semua admin memiliki fungsionalitas yang sama dan perbedaan hanya pada universitas karena dapat mengatur daftar *user*, menu acara adalah tempat untuk admin melihat daftar pengajuan dan melakukan aksi untuk menyetujui atau menolak suatu pengajuan.



Gambar 5.7. Antarmuka Menu Acara Admin (Ormawa)

Pada menu acara, tampilan memiliki beberapa komponen sebagai berikut :

1. *Heading 1* judul menu
2. Tombol tab *active* dan *archived* yang berfungsi untuk menampilkan data yang berbeda. *Active* untuk menampilkan data yang statusnya bersifat *pending* dan *archived* untuk menampilkan data yang bersifat *approved* atau *rejected*.
3. Tabel.
4. *Icon* aksi yang berfungsi untuk *approved* (*icon* centang) atau *rejected* (*icon* silang). Pada *icon* centang atau silang akan menampilkan komponen *popup*.



Gambar 5.7. *Popup approve Admin Ormawa*

```

<Dialog open={approval} onClose={() => setApproval(f
alse)} aria-labelledby="Approve">
  <DialogTitle>
    Approval
  </DialogTitle>
  <DialogContent>
    <DialogContentText>
      Approve acara dengan judul {<Typography
component="span" style={{ fontWeight: "bold" }}>{sel
ectedAcara?.judul}</Typography> } ?
    </DialogContentText>
    <TextField
      color="secondary"
      autoFocus={true}
      margin={"dense"}
      id="komentar"
      label="komentar"
      type="text"
      value={komentarApprove}
      onChange={e => setKomentarApprove(e.targ
et.value)}
      fullWidth={true}
    />
  </DialogContent>
  <DialogActions>
    <Button onClick={() => setApproval(false)}
color="secondary">
      Batal
    </Button>
    <Button onClick={() => {
      if (!selectedAcara) return;

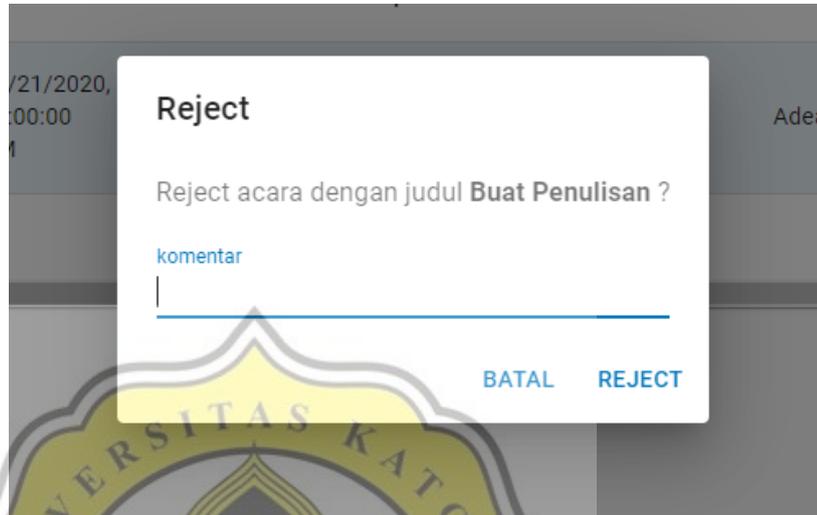
```

```

        handleApprove(selectedAcara?.id, komentarApprove)
      }} color="secondary">
        Approve
      </Button>

```

Gambar 5.8. *Script* Kode Untuk Proses Antarmuka *Approval* Proposal



Gambar 5.9. *Script* Kode Untuk *Popup Reject* Admin Ormawa

```

<Dialog open={rejection} onClose={() => setRejection(
false)} aria-labelledby="Reject">
  <DialogTitle>
    Reject
  </DialogTitle>
  <DialogContent>
    <DialogContentText>
      Reject acara dengan judul {<Typography component="span" style={{
fontWeight: "bold" }}>{selectedAcara?.judul}</Typography> } ?
    </DialogContentText>
    <TextField
      color="secondary"
      autoFocus={true}
      margin={"dense"}
      id="komentar"
      label="komentar"
      type="text"
      value={komentarReject}
      onChange={e => setKomentarReject(e.target.value)}
      fullWidth={true}
    />
  </DialogContent>

```

```

    <DialogActions>
      <Button onClick={() => setRejection(false)}
    } color="secondary">
        Batal
      </Button>
      <Button onClick={() => {
        if (!selectedAcara) return;
        handleReject(selectedAcara?.id, komentar
Reject)
      }} color="secondary">
        Reject
      </Button>
    </DialogActions>
  </Dialog>

```

Gambar 5.10. Script Kode Untuk Proses Antarmuka *Reject* Proposal

Menu *user* terdapat hanya pada admin universitas. Menu ini berfungsi untuk menambah, mengubah, dan menghapus daftar *user* yang ada.

NO	Username	Email	Nama Ketua
1	admin_fakultas_ikom	bilyanos@gmail.com	fakultas_ikom
2	admin_progdi_si	adeanbenedictus@gmail.com	progdi_si
3	admin_universitas	adeanbenedictus@gmail.com	universitas.unika
4	bem_fakultas_ikom	adeanbenedictus@gmail.com	fakultas_ikom
5	bem_universitas	adeanbenedictus@gmail.com	universitas.unika
6	biro_administrasi_umum	adeanbenedictus@gmail.com	bau
7	hmpssi	adeanbenedictus@gmail.com	progdi_si
8	unit_peminjaman_tempat	adeanbenedictus@gmail.com	upt

Gambar 5.11. Menu *user* pada Admin Universitas

Pada menu *user*, tampilan memiliki beberapa komponen sebagai berikut :

1. Tabel daftar *user*.
2. *Icon* minus untuk menghapus *user*

```
{
```

```

<Zoom
  in={!selectedUser}
  style={{
    transitionDelay: `${0}ms`,
  }}
  unmountOnExit
>
  <Fab
    style={{ backgroundColor: theme.pa
lette.error.main, color: "white" }}
    className={classes.mb1} aria-
label="edit" size="small"
    onClick={e => {
      if (window.confirm("Hapus " + se
lectedUser.username)) {
        hapusUser(auth.token, selected
User.username)
          .then(resp => {
            window.location.href = win
dow.location.href;
          })
          .catch(e => alert(e.message)
);
      }
    }}
  </Fab>
</Zoom>
  <RemoveIcon />
</Fab>
</Zoom>

```

Gambar 5.12. *Script* Kode Untuk Proses Antarmuka *Icon Delete User*

3. *Icon* pensil untuk edit user

```

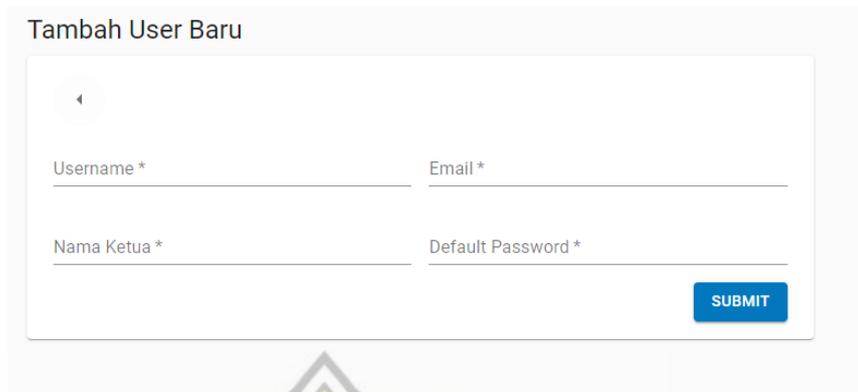
<Zoom
  in={!selectedUser}
  style={{
    transitionDelay: `${0}ms`,
  }}
  unmountOnExit
>
  <Fab color="primary" className={clas
ses.mb1} aria-label="edit" size="small"
    onClick={e => history.push("/user/
edit/" + selectedUser.username)}>
    <EditIcon />
  </Fab>

```

</Zoom>

Gambar 5.13. Script Kode Untuk Proses Antarmuka *Icon Edit User*

4. *Icon* untuk penambahan *user*.



Gambar 5.14. Menu Tambah *User* Pada Admin Universitas

```
<Fab color="secondary" aria-  
label="add" onClick={e => history.push("/user/add"  
)}>  
  <AddIcon />  
</Fab>
```

Gambar 5.15. Script Kode Untuk Proses Antarmuka *Icon Add User*

```
return <Grid container className={classes.gridRoot  
}>  
  <Grid item xs={12} md={10} lg={8} xl={6}>  
    <Typography component={"h1"} variant={"h5"}>  
      Tambah User Baru  
    </Typography>  
    <Paper>  
      <form className={classes.form}>  
        <div className={classes.dense}>  
          <IconButton onClick={() => history.pus  
h("/user")}>  
            <BackIcon />  
          </IconButton>  
        </div>  
        <div className={classes.mb2}>  
          <Grid container spacing={2}>  
            <Grid item xs>  
              <TextField  
                required  
                color="secondary"  
                fullWidth={true}
```

```

        label="Username"
        value={username}
        onChange={e => setUsername(e.target.value)}
    />
</Grid>
<Grid item xs>
    <TextField
        required
        color="secondary"
        fullWidth={true}
        value={email}
        onChange={e => setEmail(e.target.value)}
        label="Email"
    />
</Grid>
</Grid>
</div>
<div className={classes.mb2}>
    <Grid container spacing={2}>
        <Grid item xs>
            <TextField
                required
                color="secondary"
                fullWidth={true}
                label="Nama Ketua"
                value={namaKetua}
                onChange={e => setNamaKetua(e.target.value)}
            />
        </Grid>
        <Grid item xs>
            <TextField
                required
                type="password"
                color="secondary"
                fullWidth={true}
                label="Default Password"
                value={password}
                onChange={e => setPassword(e.target.value)}
            />
        </Grid>
    </Grid>
</div>

```

```

<div>
  <Grid container justify={"flex-end"}>
    <Grid item xs={"auto"}>
      <Button variant="contained" color=
{"secondary"} onClick={() => {
        handleAddUser(username, email, n
amaKetua, password)
      }}>Submit</Button>
    </Grid>
  </Grid>
</div>
</form>
</Paper>
</Grid>
</Grid >
}

```

Gambar 5.16. Script Kode Untuk Proses Antarmuka Add User

5.1.4 Implementasi Antarmuka Menu Pengajuan Peminjaman Admin Unit Peminjaman Tempat Portal Ormawa

Menu peminjaman tempat adalah menu yang ada untuk admin unit peminjaman tempat dimana akan menampilkan pengajuan peminjaman tempat dari semua *user* ormawa yang akan meminjam tempat.

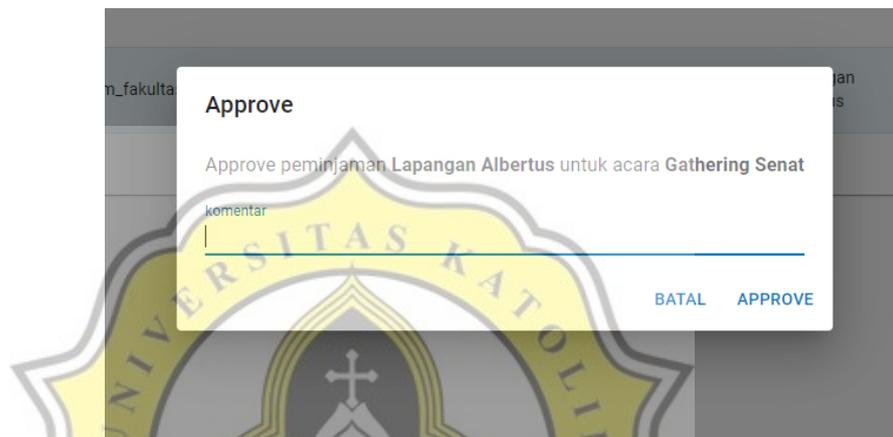
Id	User	Acara	Mulai	Selesai	Nama Tempat	Status	Komentar	Tanggal Pengajuan
5	ben_fakultas.ikom	Gathering Senat	11/16/2020, 6:35:00 PM	11/18/2020, 11:35:00 PM	Lapangan Albertus	pending	-	11/15/2020, 6:36:27 PM

Gambar 5.17. Menu Peminjaman Tempat Admin UPT

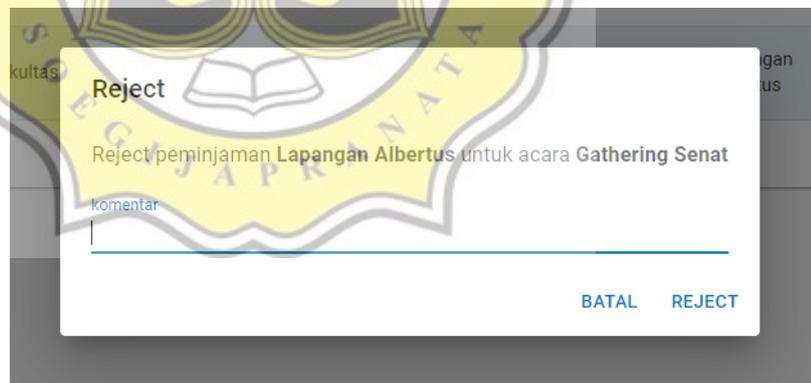
Pada menu peminjaman tempat, tampilan memiliki beberapa komponen sebagai berikut :

1. *Heading* 1 judul menu.

2. Tombol tab *active* dan *archived* yang berfungsi untuk menampilkan data yang berbeda. *Active* untuk menampilkan data yang statusnya bersifat *pending* dan *archived* untuk menampilkan data yang bersifat *approved* atau *rejected*.
3. Tabel.
4. *Icon* aksi yang berfungsi untuk *approved* (*icon* centang) atau *rejected* (*icon* silang). Pada *icon* centang atau silang akan menampilkan komponen *popup*.



Gambar 5.18. *Popup Approve* Peminjaman Tempat



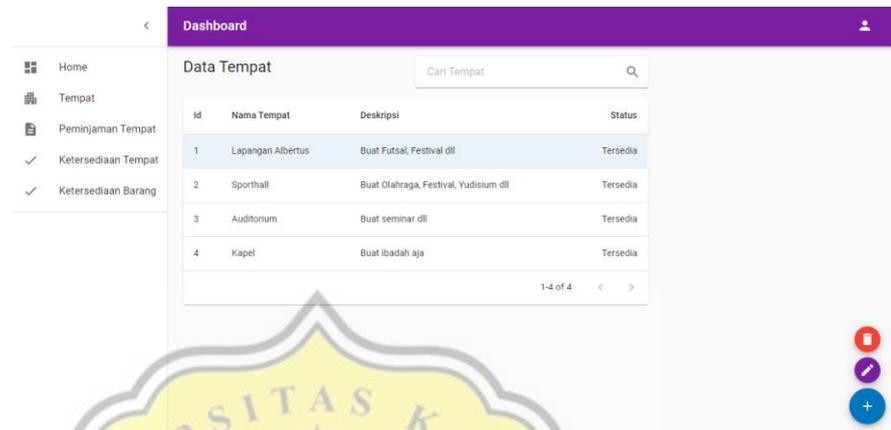
Gambar 5.19. *Popup Reject* Peminjaman Tempat

5.1.5 Implementasi Antarmuka Menu Tempat Admin Unit Peminjaman Tempat Portal Ormawa

Menu tempat adalah menu yang ada untuk admin unit peminjaman tempat untuk melakukan *CRUD* pada tabel tempat.

Berfungsi untuk melakukan pemeliharaan tempat yang tersedia atau disediakan oleh pihak kampus.

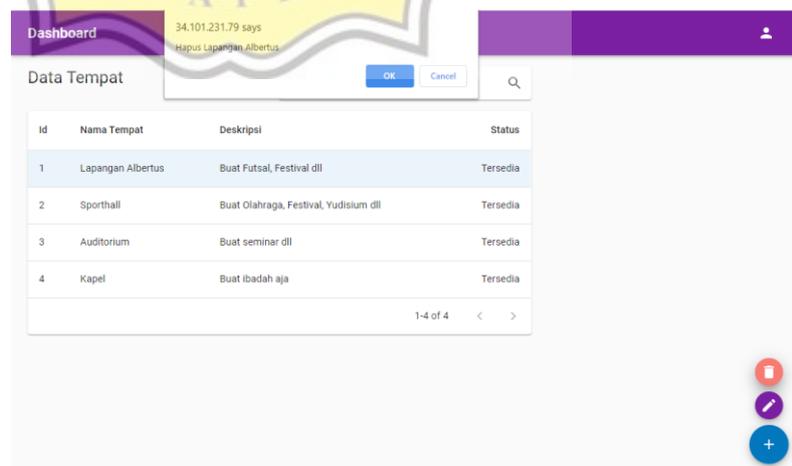
Pada *user* yang lain juga terdapat menu tempat akan tetapi hanya dapat melihat daftar dan tidak dapat melakukan aksi yang lain.



Gambar 5.20. Menu Tempat Admin UPT

Pada menu tempat, tampilan memiliki beberapa komponen sebagai berikut :

1. *Heading* 1 judul menu.
2. Tabel.
3. *Icon* tempat sampah yang berfungsi untuk menghapus daftar tempat.



Gambar 5.21. Aksi *Delete* Tempat

```
<div className={classes.fab}>
```

```

{
  <Zoom
    in={!selectedTempat}
    style={{
      transitionDelay: `${0}ms`,
    }}
    unmountOnExit
  >
    <Fab
      style={{ backgroundColor: theme.palette.error.main, color: "white" }}
      className={classes.mb1} aria-label="edit" size="small"
      onClick={e => {
        if (window.confirm("Hapus " + selectedTempat.namaTempat)) {
          hapusTempat(auth.token, selectedTempat.id)
            .then(resp => {
              window.location.href = window.location.href;
            })
            .catch(e => alert(e.message));
        }
      }}
    >
      <RemoveIcon />
    </Fab>
  </Zoom>
}

```

Gambar 5.22. *Script Kode Untuk Proses Antarmuka Icon Delete Tempat*

4. *Icon pensil untuk mengedit nama dan deskripsi tempat.*

Dashboard

Tambah Tempat Baru

<

Nama Tempat*	Deskripsi*
Lapangan Albertus	Buat Futsal, Festival dll

SUBMIT

Gambar 5.23. Aksi *Edit* Tempat

```
{
  <Zoom
    in={!selectedTempat}
    style={{
      transitionDelay: `${0}ms`,
    }}
    unmountOnExit
  >
    <Fab color="primary" className={classes.mb1} aria-label="edit" size="small"
      onClick={e => history.push("/tempat/edit/" + selectedTempat.id)}>
      <EditIcon />
    </Fab>
  </Zoom>
}
```

Gambar 5.24. *Script* Kode Untuk Proses Antarmuka *Icon Edit Tempat*

```
return <Grid container className={classes.gridRoot}>
  <Grid item xs={12} md={10} lg={8} xl={6}>
    <Typography component="h1" variant="h5">
      Tambah Tempat Baru
    </Typography>
    <Paper>
      <form className={classes.form}>
        <div className={classes.dense}>
          <IconButton onClick={() => history.push("/tempat")}>
            <BackIcon />
          </IconButton>
        </div>
        <div className={classes.mb2}>
          <Grid container spacing={2}>
            <Grid item xs>
              <TextField
                required
                color="secondary"
                fullWidth={true}
                label="Nama Tempat"
                value={nama}
                onChange={e => setNama(e.target.value)}
              />
            </Grid item>
          </Grid container>
        </div>
      </form>
    </Paper>
  </Grid item>
</Grid container>
```

```

</Grid>
<Grid item xs>
  <TextField
    required
    color="secondary"
    fullWidth={true}
    value={deskripsi}
    onChange={e => setDescription(e.target.value)}
    label="Deskripsi"
  />
</Grid>
</Grid>
</div>

<div>
  <Grid container justify="flex-end">
    <Grid item xs="auto">
      <Button variant="contained" color="secondary" onClick={() => {
        handleEditTempat(Number(id), nama, deskripsi, status)
      }}>Submit</Button>
    </Grid>
  </Grid>
</div>
</form>
</Paper>
</Grid>
</Grid >
}

```

Gambar 5.25. *Script* Kode Untuk Proses Antarmuka *Edit Tempat*

5. *Icon* plus untuk menambahkan tempat.

The screenshot shows a web application interface with a purple header bar labeled 'Dashboard'. Below the header, there is a section titled 'Tambah Tempat Baru'. Inside this section, there is a form with a back arrow icon at the top left. The form contains two input fields: 'Nama Tempat *' and 'Deskripsi *'. To the right of the 'Deskripsi *' field is a blue button labeled 'SUBMIT'.

Gambar 5.26. Aksi *Add* Tempat

5.1.6 Implementasi Antarmuka Menu Pengajuan Peminjaman Admin Biro Administrasi Umum Portal Ormawa

Menu peminjaman barang adalah menu yang ada untuk admin biro administrasi umum dimana akan menampilkan permohonan peminjaman barang dari seluruh user *ormawa* yang akan meminjam barang.



Id	User	Acara	Mulai	Selesai	Nama Barang	Qty	Status	Komentar	Tanggal Pengajuan
2	bem_fakultas.ikom	Gathering Senat	11/12/2020, 5:30:00 PM	11/12/2020, 7:00:00 PM	kursi	50	pending	-	11/11/2020, 11:02:41 PM
3	bem_fakultas.ikom	Gathering Senat	11/17/2020, 6:37:00 PM	11/19/2020, 10:37:00 PM	Meja	100	pending	-	11/15/2020, 6:37:50 PM

Gambar 5.27. Menu Peminjaman Barang Admin BAU

Pada menu peminjaman tempat, tampilan memiliki beberapa komponen sebagai berikut :

1. *Heading* 1 judul menu.
2. Tombol tab *active* dan *archived* yang berfungsi untuk menampilkan data yang berbeda. *Active* untuk menampilkan data yang statusnya bersifat *pending* dan *archived* untuk menampilkan data yang bersifat *approved* atau *rejected*.
3. Tabel.
4. *Icon* aksi yang berfungsi untuk *approved* (*icon* centang) atau *rejected* (*icon* silang). Pada *icon* centang atau silang akan menampilkan komponen *popup*.

Approve

Approve peminjaman **Sound system** untuk acara **HUT RI 2021**

komentar

qty

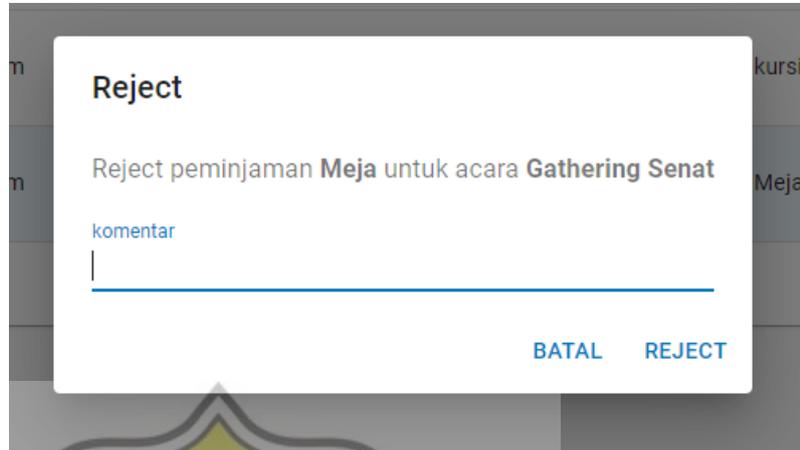
0

BATAL APPROVE

Gambar 5.28. *Popup Approve Peminjaman Barang*

```
<TextField
  color="secondary"
  autoFocus={true}
  margin={"dense"}
  id="komentar"
  label="komentar"
  type="text"
  value={komentar}
  onChange={e => setKomentar(e.target.value
)}
  fullWidth={true}
/>
{
  props.qtyEnabled
  ? <TextField
    style={{ marginTop: theme.spacing(2)
}}
    color="secondary"
    margin={"dense"}
    id="qty"
    label="qty"
    type="number"
    value={qty + ""}
    onChange={e => setQty(Number(e.target
.value))}
    fullWidth={true}
  />
  : null
}
```

Gambar 5.29. *Script* Kode Untuk Proses Antarmuka *Approve* Peminjaman Barang



Gambar 5.30. *Popup Reject* Peminjaman Barang

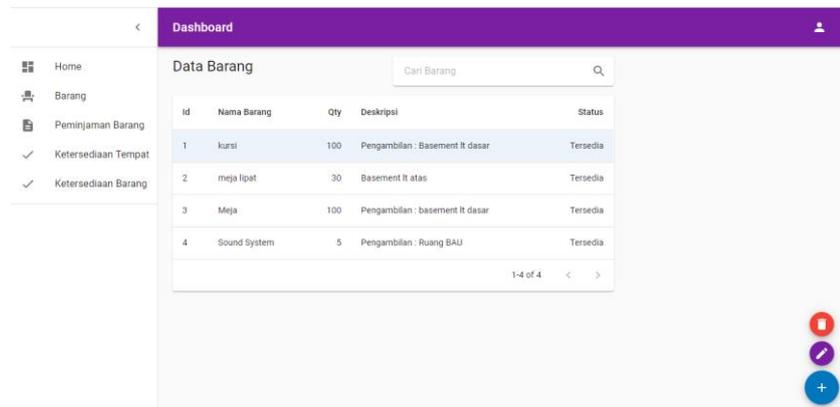
```
<CompletionDialog
  open={rejection}
  close={() => setRejection(false)}
  cancel={() => setRejection(false)}
  ok={(komentar: string) => {
    if (selectedPeminjaman)
      handleReject(selectedPeminjaman?.id, kome
ntar);
    setRejection(false)
  }}
  content={selectedPeminjaman ? <div>Reject pem
injaman <b>{selectedPeminjaman.nama_tempat}</b> unt
uk acara <b>{selectedPeminjaman.acara}</b></div> :
""}
  type="Reject"
/>
```

Gambar 5.31. *Script* Kode Untuk Proses Antarmuka *Reject* Peminjaman Barang

5.1.7 Implementasi Antarmuka Menu Barang Admin Biro Administrasi Umum Portal Ormawa

Menu tempat adalah menu yang ada untuk admin unit peminjaman tempat untuk melakukan *CRUD* pada tabel barang.

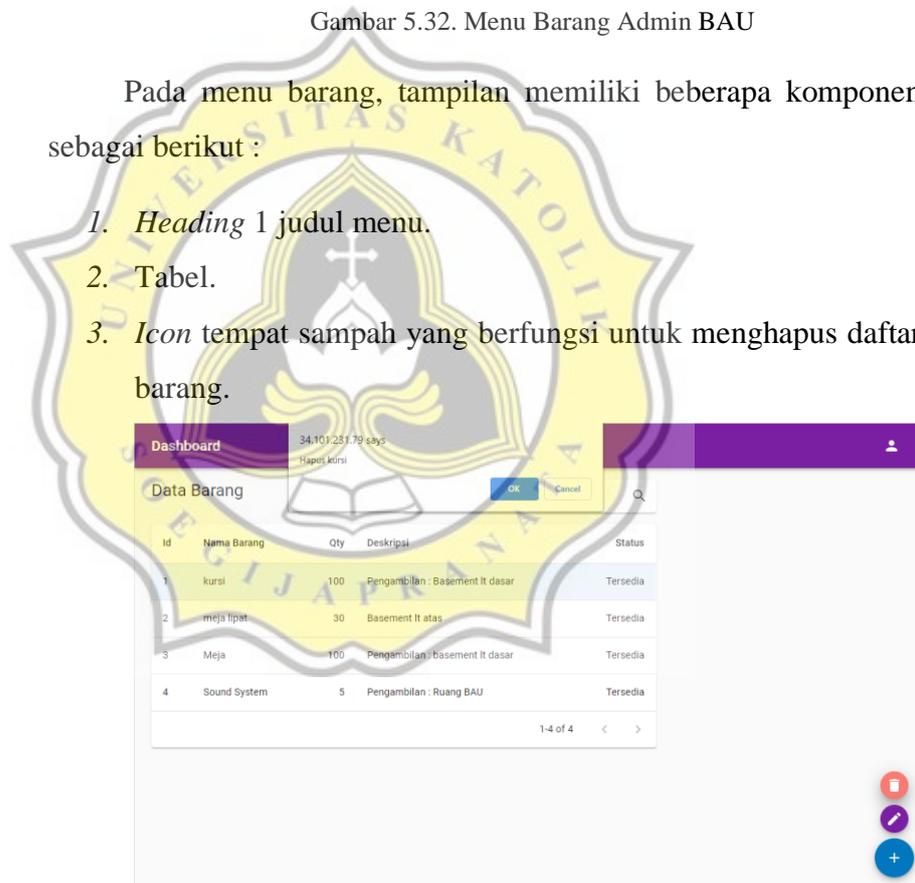
Berfungsi untuk melakukan pemeliharaan tempat yang tersedia atau disediakan oleh pihak kampus.



Gambar 5.32. Menu Barang Admin BAU

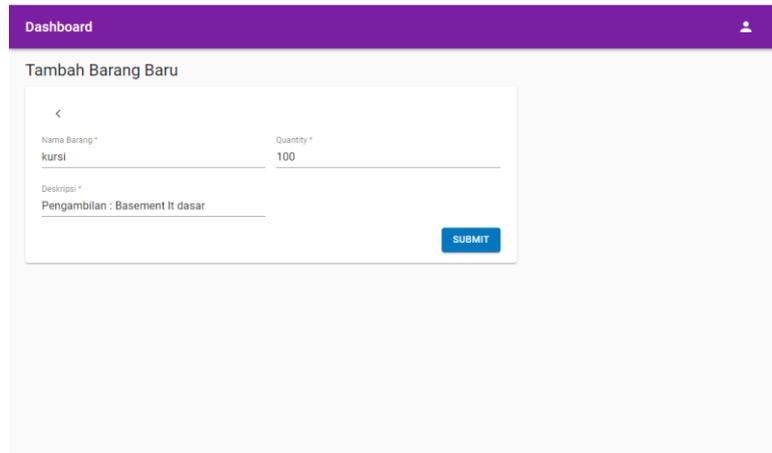
Pada menu barang, tampilan memiliki beberapa komponen sebagai berikut :

1. *Heading* 1 judul menu.
2. Tabel.
3. *Icon* tempat sampah yang berfungsi untuk menghapus daftar barang.



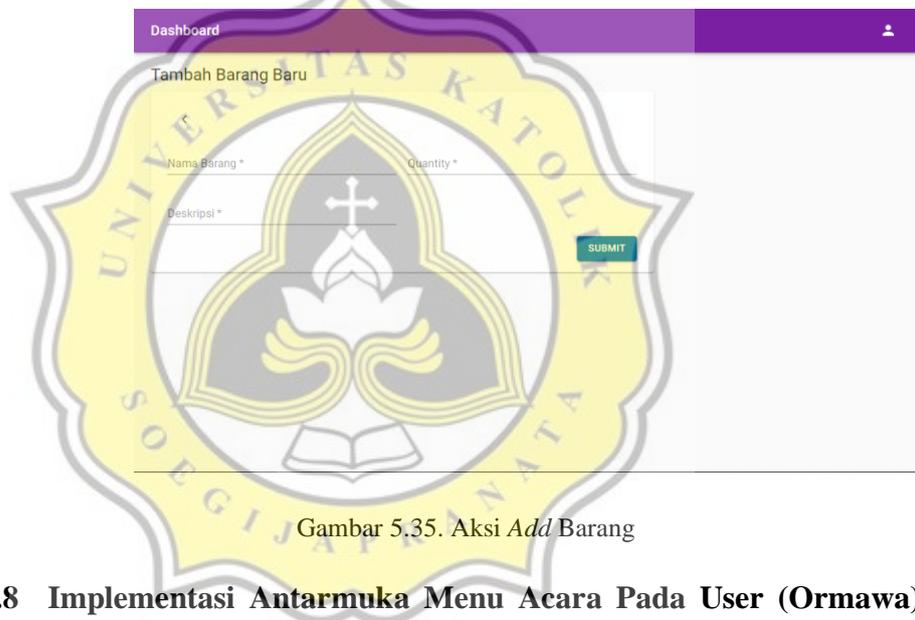
Gambar 5.33. Aksi *Delete* Barang

4. *Icon* pensil untuk mengedit nama dan deskripsi barang.



Gambar 5.34. Aksi *Edit* Barang

5. *Icon* plus untuk menambahkan tempat.



Gambar 5.35. Aksi *Add* Barang

5.1.8 Implementasi Antarmuka Menu Acara Pada User (Ormawa) Portal Ormawa

Pada Portal Ormawa terdapat tiga jenis *user* yaitu program studi, fakultas, dan universitas. Mereka memiliki fungsi dan maksud yang sama, pada menu acara ini berfungsi agar *user* dapat mengajukan proposal dan melihat daftar pengajuan.

Id	Judul	Mulai	Selesai	Dalam Kampus	Tempat	Anggaran	Ketua Acara	File	User	Approval	LPJ
1	Demo 1	11/25/2020, 12:00:00 PM	11/25/2020, 4:00:00 PM	Ya	Lapangan	Rp. 10,000,000	Adean	Download Proposal	hmpssi	approved	-
3	HUT HMPSi	11/11/2020, 12:00:21 PM	11/12/2020, 1:00:00 PM	Ya		Rp. 1,500,000	Kamu	Download Proposal	hmpssi	rejected	-
4	HUT HMPSi	11/11/2020, 11:00:57 PM	11/12/2020, 11:00:00 PM	Ya	Unika Soegijapranata	Rp. 1,500,000	Chris	Download Proposal	hmpssi	approved	-
5	Seminar	11/12/2020, 11:12:00 PM	11/13/2020, 11:12:00 PM	Ya		Rp. 500,000	Chris	Download Proposal	hmpssi	approved	-
6	Geco Fest	11/12/2020, 11:33:47 AM	11/12/2020, 11:33:47 AM	Ya	Kapel	Rp. 10,000	Bima Yang Agung	Download Proposal	hmpssi	approved	-

Gambar 5.36. Menu Acara User Ormawa

Pada menu acara *user* ormawa, tampilan memiliki beberapa komponen sebagai berikut :

1. *Heading* 1.
2. Tombol tab *active* dan *archived* yang berfungsi untuk menampilkan data yang berbeda. *Active* untuk menampilkan data yang statusnya bersifat *pending*, *rejected* dan *approved* sedangkan *archived* untuk menampilkan data acara yang sudah selesai ditandai dengan *user* mengupload LPJ atau lembar pertanggungjawaban.
3. Tabel
4. *Icon printer* yang berfungsi untuk mencetak lembar persetujuan yang dimana dengan syarat proposal sudah disetujui oleh admin yang bersangkutan.
5. *Icon ceklis* untuk menyelesaikan acara dengan cara upload lembar pertanggungjawaban. *Icon* ini memiliki syarat yaitu dapat dilakukan setelah tanggal selesai acara sudah terpenuhi.

Bukti Approval Proposal

proposal/#4/hmpssi

Judul Acara
HUT HMPSI

Ketua Acara
Chris

Tanggal Mulai
11/11/2020, 11:00:57 PM

Tanggal Selesai
11/12/2020, 11:00:00 PM

Tempat
Unika Soegijapranata
dalam kampus.

Anggaran
Rp. 1,500,000

Approval
11/11/2020, 11:02:37 PM

Komentar
laksanakan

Peminjaman Tempat

NO	NAMA	STATUS	Tanggal Mulai	Tanggal Selesai	Komentar	Tanggal Respon
1	Auditorium	approved	11/11/2020, 11:03:23 PM	11/12/2020, 11:03:00 PM	laksanakan	11/11/2020, 11:03:57 PM

Peminjaman Barang

NO	NAMA	QTY	STATUS	Tanggal Mulai	Tanggal Selesai	Komentar	Tanggal Respon
----	------	-----	--------	---------------	-----------------	----------	----------------

Gambar 5.37. Aksi *Print* Permohonan Acara

Kode penampil bukti approval bagi *user* ormawa.

```
!auth.username.startsWith("admin_")
  ? <div className={classes.fab}>
    {
      <Zoom
        in={
          selectedAcara
            && (auth.ketua.startsWit
h("progdi."))
          ? selectedAcara.aprf =
=== "approved" // progdi
          : auth.ketua.startsWit
h("fakultas.")
          ? selectedAcara.aprf
=== "approved" && selectedAcara.aprf === "approve
d" //fakultas
          : selectedAcara.aprf
=== "approved"
        }
      />
    }
  ? true
  : false
}
timeout={transitionDuration}
style={{
```

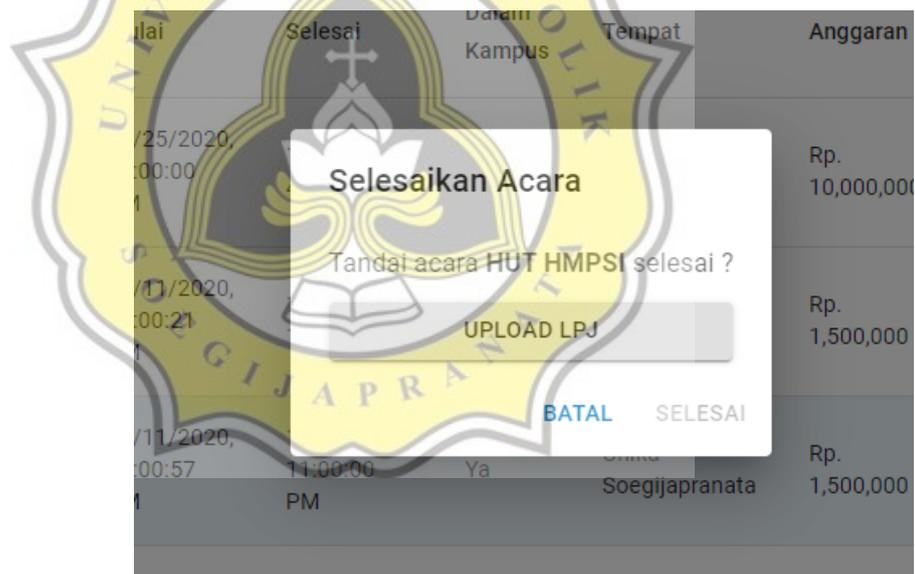
```

        transitionDelay: `${selectedAcara ? transitionDuration.exit : 0}ms`,
      }}
      unmountOnExit
    >
    <Fab color="primary"
      className={classes.mbl}
      aria-
label="print approval"
      size="small"
      onClick={() => history.push("/print/" + selectedAcara?.id)}
    >
      <PrintIcon />
    </Fab>
  </Zoom>

```

Gambar 5.38. Script Kode Untuk Proses Antarmuka Print Approval

6. Icon centang untuk menyelesaikan acara. Tombol Upload LPJ berfungsi untuk meng-upload file LPJ.



Gambar 5.39. Aksi Menyelesaikan Acara

```

<Zoom
  in={
    selectedAcara
    && (
      Number(new Date(selectedAcara.tanggalSelesai)) <= Date.now()
      && auth.ketua.starts
    )
  }
  With("progdi.")

```

```

? selectedAcara.aprf
=== "approved" // progdi
: auth.ketua.startsW
ith("fakultas.")
? selectedAcara.ap
rf === "approved" && selectedAcara.aprp === "appro
ved" //fakultas
: selectedAcara.ap
rp === "approved"
)
? true
: false
}
timeout={transitionDuration}
style={{
transitionDelay: `${select
edAcara ? transitionDuration.exit : 0}ms`,
}}
unmountOnExit
<Fab color="primary"
className={classes.mb1}
aria-label="selesai"
size="small"
onClick={() => setModalCom
plete(true)}
<CheckIcon />
</Fab>
</Zoom>

```

Gambar 5.40. Script Kode Untuk Proses Antarmuka Icon Centang

7. Icon plus untuk menambahkan pengajuan acara/proposal.

Buat Acara Baru

<

Judul Acara *
-

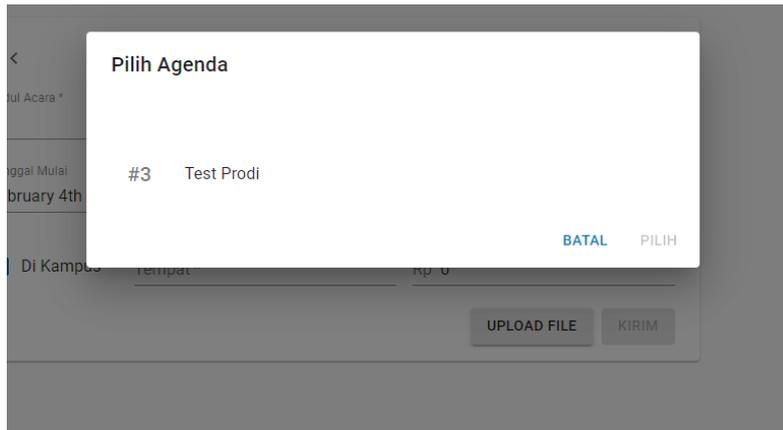
Nama Ketua Acara *

Tanggal Mulai
February 4th 11:48 am

Tanggal Selesai
February 4th 11:48 am

Di Kampus Tempat * Anggaran *
Rp 0

Gambar 5.41. Aksi Menambahkan Permohonan Acara/Proposal



Gambar 5.42. Aksi Menambahkan Permohonan Acara/Proposal (Pilih nama acara)

```

<Fab color="secondary" aria-
label="add" onClick={e => history.push("/acara/add
")}>
  <AddIcon />
</Fab>
</div>

```

Gambar 5.43. Script Kode Untuk Proses Antarmuka Add Acara

```

return <Grid container className={classes.gridRoot
}>
  <Grid item xs={12} md={10} lg={8} xl={6}>
    <Typography component={"h1"} variant={"h5"}>
      Buat Acara Baru
    </Typography>
    <Paper>
      <form className={classes.form}>
        <div className={classes.dense}>
          <IconButton onClick={() => history.pus
h("/acara")}>
            <BackIcon />
          </IconButton>
        </div>
        <div className={classes.mb2}>
          <Grid container spacing={2}>
            <Grid item xs>
              <TextField
                required
                color="secondary"
                fullWidth={true}
                label="Judul Acara"
                value={judul}

```

```

        onChange={e => setJudul(e.target
.value)}}
        helperText="contoh: HUT RI 2020"
      />
    </Grid>
    <Grid item xs>
      <TextField
        required
        color="secondary"
        fullWidth={true}
        value={ketuaAcara}
        onChange={e => setKetuaAcara(e.t
arget.value)}
        label="Nama Ketua Acara"
      />
    </Grid>
  </Grid>
</div>

<div className={classes.mb2}>
  <Grid container spacing={2}>
    <Grid item xs>
      <DateTimePicker
        label={"Tanggal Mulai"}
        fullWidth={true}
        minDate={new Date()}
        value={tanggalMulai}
        onChange={d => setTanggalMulai(n
ew Date(d?.toDate() || Date.now()))}
      />
    </Grid>
    <Grid item xs>
      <DateTimePicker
        label={"Tanggal Selesai"}
        fullWidth={true}
        minDate={tanggalMulai}
        value={tanggalSelesai}
        onChange={d => setTanggalSelesai
(new Date(d?.toDate() || Date.now()))}
      />
    </Grid>
  </Grid>
</div>

<div className={classes.mb2}>
  <Grid container spacing={2}>
    <Grid item xs={"auto"}>

```

```

        <FormControllabel
          className={classes.mt1}
          control={<Checkbox checked={diKa
mpus} onChange={() => setDiKampus(!diKampus)} name
="checkedA" />}
          label="Di Kampus"
        />
      </Grid>
      <Grid item xs>
        <TextField
          required
          color="secondary"
          fullWidth={true}
          value={tempat}
          onChange={e => setTempat(e.targe
t.value)}
          label="Tempat"
        />
      </Grid>
      <Grid item xs>
        <TextField
          required
          color="secondary"
          InputProps={{
            startAdornment: <InputAdornmen
t position="start">Rp</InputAdornment>,
          }}
          fullWidth={true}
          value={anggaran}
          onChange={(e) => {
            console.log(e.target.value);
            const filtered = e.target.valu
e.split("").filter(c => allowedValue.includes(c)).
join("");
            setAnggaran(filtered);
          }}
          label="Anggaran"
        />
      </Grid>
    </Grid>
  </div>
  <div>
    <Grid container justify={"flex-
end"} spacing={1}>
      <Grid item xs="auto">
        <Button
          variant="contained"

```

```

        component="label"
      >
        {
          file ? file.name : "Upload File"
        }
      </input>
      <input
        onChange={(e: any) => setFile(
          e?.target?.files[0])}
        type="file"
        style={{ display: "none", margin: 0, padding: 0 }}
      />
    </Button>
  </Grid>
  <Grid item xs={"auto"}>
    <Button disabled={!file}
      title="Submit"
      variant="contained"
      onClick={() => handleSubmit(judul, ketuaAcara, tanggalMulai, tanggalSelesai, diKampus, tempat, anggaran, file)}
      color={"secondary"}
    >
      Kirim
    </Button>
  </Grid>
</div>
</form>
</Paper>
</Grid>
</Grid >

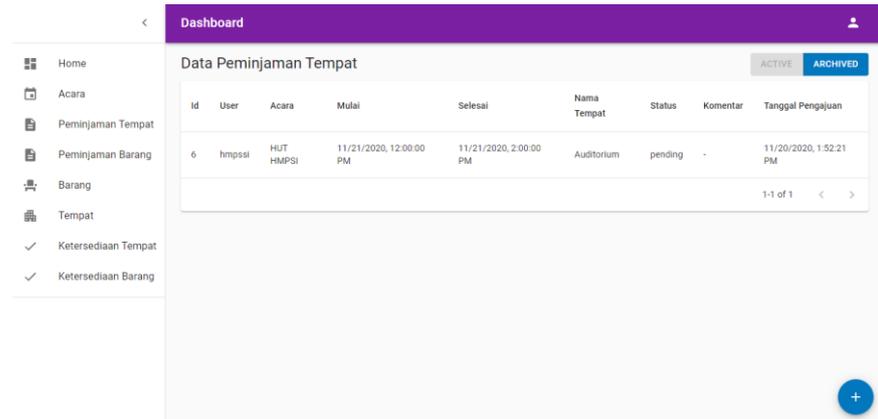
```

Gambar 5.44. Script Kode Untuk Proses Antarmuka Form Acara

5.1.9 Implementasi Antarmuka Menu Peminjaman Tempat Pada User (Ormawa) Portal Ormawa

Menu peminjaman tempat adalah menu yang terdapat juga pada *user* ormawa, menu ini berfungsi untuk melihat daftar peminjaman user terkait dan mengajukan peminjaman tempat dengan syarat proposal acara sudah disetujui dengan admin yang

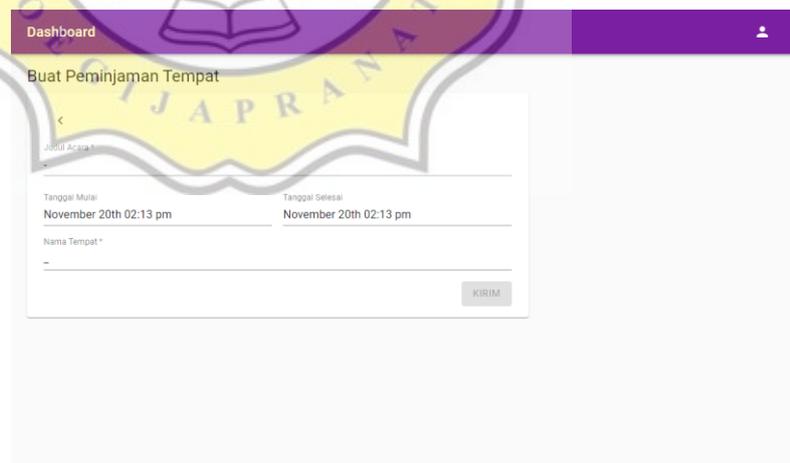
bersangkutan.



Gambar 5.45. Menu Peminjaman Tempat Pada *User* Ormawa

Pada menu peminjaman tempat *user* ormawa, tampilan memiliki beberapa komponen sebagai berikut :

1. *Heading* 1
2. Tombol tab *active* dan *archived* yang berfungsi untuk menampilkan data yang berbeda. *Active* untuk menampilkan data yang statusnya bersifat *pending* dan *archived* untuk menampilkan data yang bersifat *approved* atau *rejected*.
3. *Icon plus* untuk menambahkan peminjaman tempat.



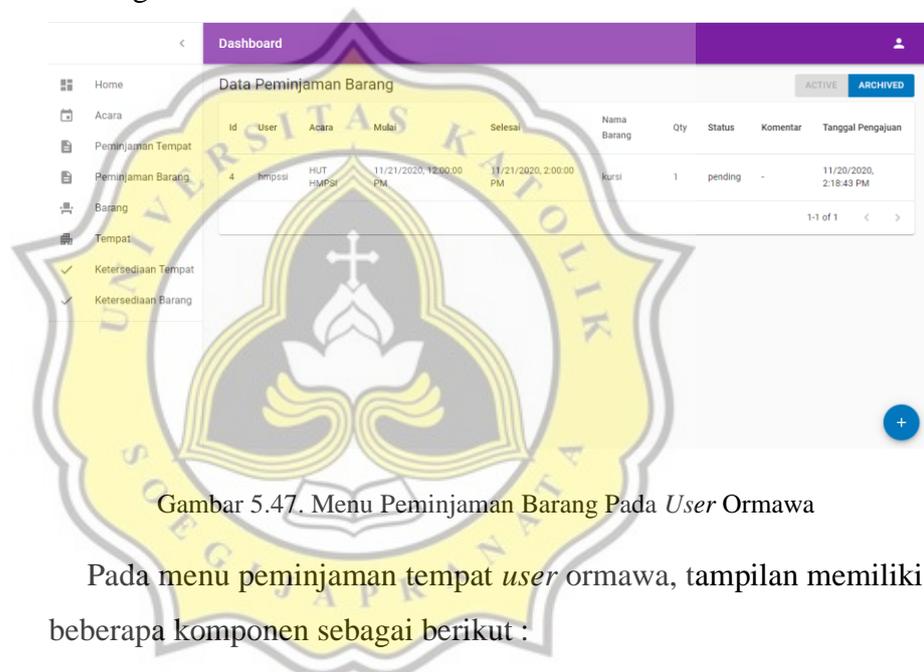
Gambar 5.46. Aksi Menambahkan Peminjaman Tempat

Pada bagian peminjaman, bagian judul acara hanya akan menampilkan acara yang sudah disetujui oleh admin ormawa. Nama tempat akan hanya memunculkan tempat

yang terdaftar oleh UPT, dan akan muncul jika tanggal mulai hingga selesai tidak bertabrakan dengan peminjaman yang sudah disetujui oleh admin UPT.

5.1.10 Implementasi Antarmuka Menu Peminjaman Barang Pada User (Ormawa) Portal Ormawa

Menu peminjaman barang adalah menu yang terdapat juga pada *user* ormawa, menu ini berfungsi untuk melihat daftar peminjaman user terkait dan mengajukan peminjaman barang dengan syarat proposal acara sudah disetujui dengan admin yang bersangkutan.



Gambar 5.47. Menu Peminjaman Barang Pada *User* Ormawa

Pada menu peminjaman tempat *user* ormawa, tampilan memiliki beberapa komponen sebagai berikut :

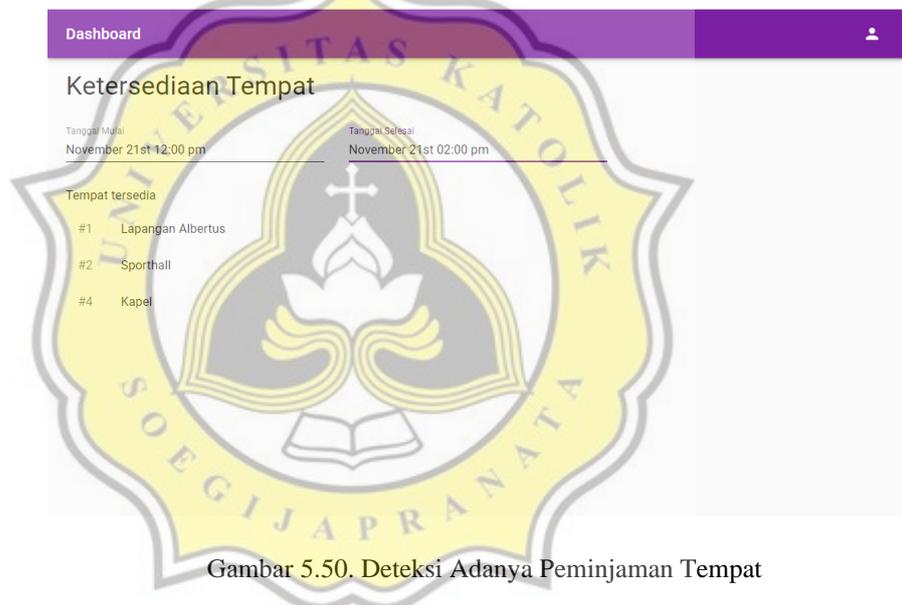
1. *Heading* 1
2. Tombol tab *active* dan *archived* yang berfungsi untuk menampilkan data yang berbeda. *Active* untuk menampilkan data yang statusnya bersifat *pending* dan *archived* untuk menampilkan data yang bersifat *approved* atau *rejected*.
3. *Icon plus* untuk menambahkan peminjaman barang.

Gambar 5.49. Menu Ketersediaan Tempat

Pada menu ketersediaan tempat, tampilan memiliki beberapa komponen sebagai berikut :

1. *Heading* 1.
2. *Form* field pencarian tanggal mulai
3. *Form* field pencarian tanggal selesai.
4. Tabel.

Pada gambar dibawah terlihat hilangnya tempat Auditorium dikarenakan adanya deteksi peminjaman dari tanggal dan jam mulai hingga selesai.



Gambar 5.50. Deteksi Adanya Peminjaman Tempat

```
return <div>
  <Typography component="h1" variant="h4">
    Ketersediaan Tempat
  </Typography>

  <div>
    <Grid container className={classes.gridRoot} spacing={4}>
      <Grid item xs={4}>
        <DateTimePicker
          label={"Tanggal Mulai"}
          fullWidth={true}
          minDate={new Date()}
          value={tanggalMulai}
        />
      </Grid item>
    </Grid container>
  </div>
</div>
```

```

        onChange={d => setTanggalMulai(new Date(d?.toDate() || Date.now()))}
      />
    </Grid>
    <Grid item xs={4}>
      <DateTimePicker
        label={"Tanggal Selesai"}
        fullWidth={true}
        minDate={tanggalMulai}
        value={tanggalSelesai}
        onChange={d => setTanggalSelesai(new Date(d?.toDate() || Date.now()))}
      />
    </Grid>
  </Grid>
</div>

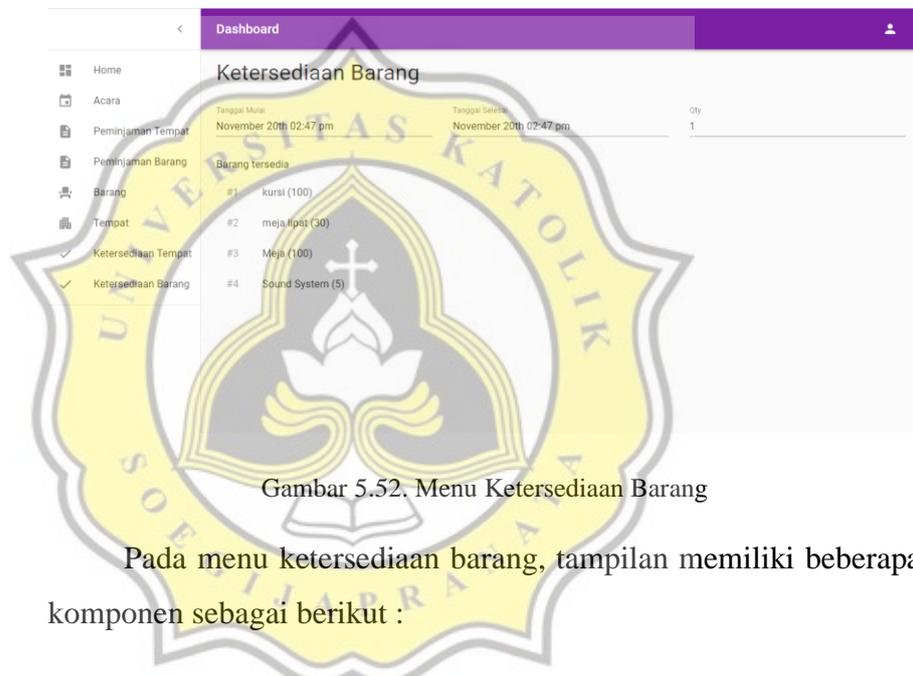
<div className={classes.gridRoot}>
  <Typography>
    Tempat tersedia
  </Typography>
  <Grid container >
    <Grid item xs={8}>
      <List>
        {
          tempat.map(t => <ListItem>
            <ListItemIcon>
              <Typography>
                #{t.id}
              </Typography>
            </ListItemIcon>
            <ListItemText>
              {t.nama_tempat}
            </ListItemText>
          </ListItem>)
        }
      </List>
    </Grid>
  </Grid>
</div>
</div>

```

Gambar 5.51. *Script* Kode Untuk Proses Antarmuka Ketersediaan Tempat

5.1.12 Implementasi Antarmuka Menu Ketersediaan Barang Portal Ormawa

Menu ketersediaan barang adalah menu yang terdapat pada semua *user*. Fungsi dari menu ketersediaan barang adalah mendeteksi adanya peminjaman tempat yang sudah disetujui oleh admin unit peminjaman tempat sehingga tidak menampilkan daftar tempat yang terdeteksi sudah disetujui oleh pihak admin. Perbedaan dengan peminjaman tempat adalah tidak hanya mendeteksi tanggal mulai hingga selesai, juga mendeteksi QTY yang dipinjam.

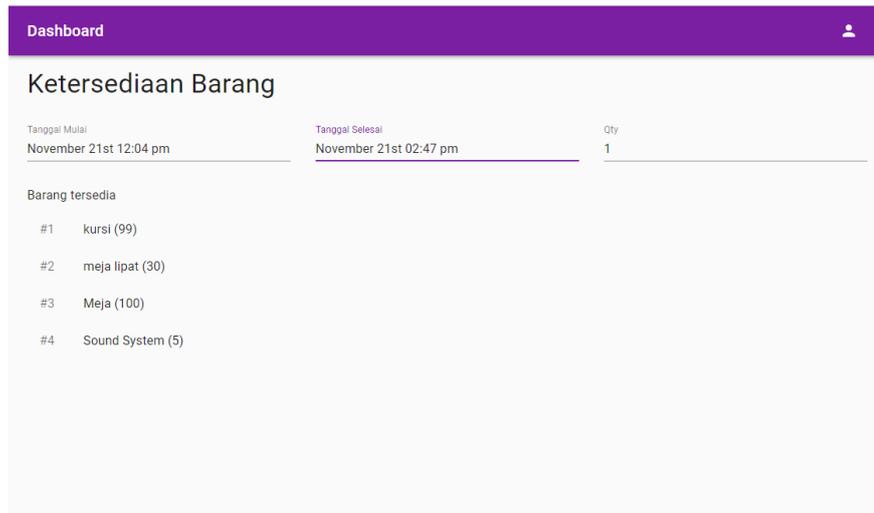


Gambar 5.52. Menu Ketersediaan Barang

Pada menu ketersediaan barang, tampilan memiliki beberapa komponen sebagai berikut :

1. *Heading* 1.
2. *Form* field pencarian tanggal mulai.
3. *Form* field pencarian tanggal selesai.
4. *Form* field pencarian QTY.
5. Tabel.

Pada gambar dibawah terlihat semua barang masih lengkap akan tetapi dikarenakan tidak hanya mendeteksi tanggal tetapi juga QTY. Maka terlihat ada QTY yang berkurang dan adapun jika QTY yang tidak mencukupi juga sistem tidak akan menampilkan.



Gambar 5.53 Menu Ketersediaan Barang

```
return <div>
  <Typography component="h1" variant="h4">
    Ketersediaan Barang
  </Typography>
  <div>
    <Grid container className={classes.gridRoot} spacing={4}>
      <Grid item xs={4}>
        <DateTimePicker
          label={"Tanggal Mulai"}
          fullWidth={true}
          minDate={new Date()}
          value={tanggalMulai}
          onChange={d => setTanggalMulai(new Date(d?.toDate() || Date.now()))}
        />
      </Grid>
      <Grid item xs={4}>
        <DateTimePicker
          label={"Tanggal Selesai"}
          fullWidth={true}
          minDate={tanggalMulai}
          value={tanggalSelesai}
          onChange={d => setTanggalSelesai(new Date(d?.toDate() || Date.now()))}
        />
      </Grid>
    </Grid>
  </div>
</div>
```

```

<Grid item xs={4}>
  <TextField
    label="Qty"
    fullWidth
    type="number"
    value={qty}
    onChange={e => setQty(Number(e.target.value))}
  >>
  />
</Grid>
</Grid>
</div>

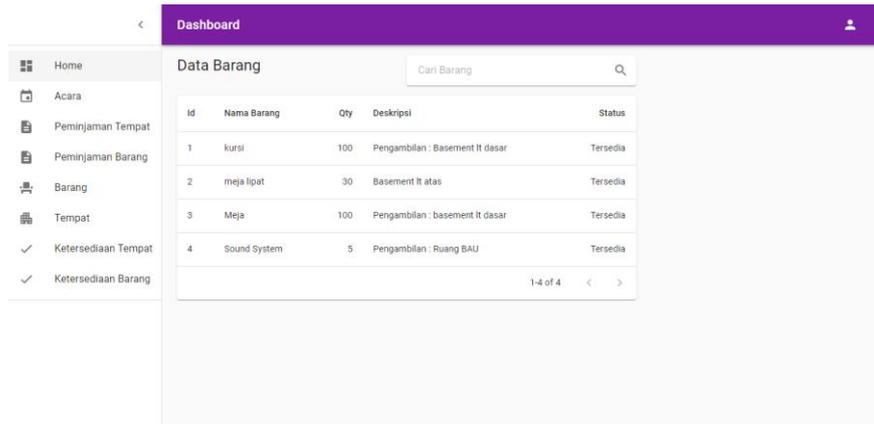
<div className={classes.gridRoot}>
  <Typography>
    Barang tersedia
  </Typography>
  <Grid container >
    <Grid item xs={8}>
      <List>
        {
          barang.map(t => <ListItem>
            <ListItemIcon>
              <Typography>
                #{t.id}
              </Typography>
            </ListItemIcon>
            <ListItemText>
              {t.namaBarang} ({t.qty})
            </ListItemText>
          </ListItem>
        )
      </List>
    </Grid>
  </Grid>
</div>
</div>

```

Gambar 5.54. *Script* Kode Untuk Proses Antarmuka Ketersediaan
Barang

5.1.13 Implementasi Antarmuka Menu Barang Portal Ormawa

Menu barang juga tampil di setiap user dan yang berbeda hanya pada admin terkait. Menu barang ini berfungsi untuk *user* mengetahui daftar barang apa saja yang disediakan oleh pihak kampus.



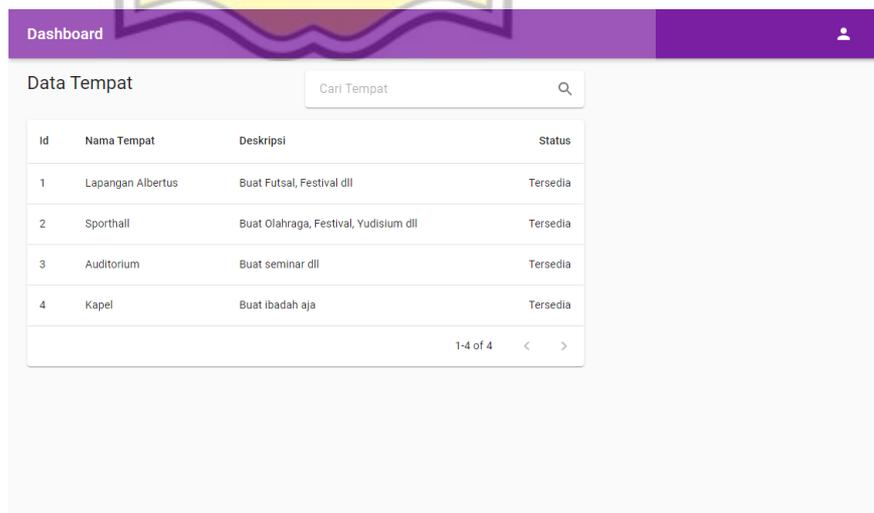
Gambar 5.55. Menu Barang

Pada menu barang, tampilan memiliki beberapa komponen sebagai berikut :

1. *Heading* 1.
2. *Form Field* pencarian nama barang.
3. Tabel.

5.1.14 Implementasi Antarmuka Menu Tempat Portal Ormawa

Menu tempat juga tampil di setiap user dan yang berbeda hanya pada admin terkait. Menu tempat ini berfungsi untuk setiap *user* mengetahui daftar tempat dan kegunaannya yang disediakan kampus.



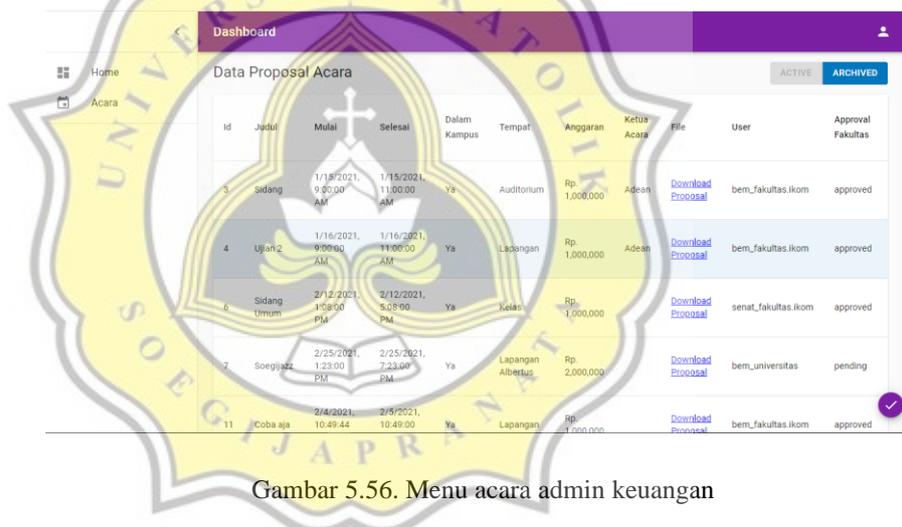
Gambar 5.56. Data Tempat

Pada menu tempat, tampilan memiliki beberapa komponen sebagai berikut :

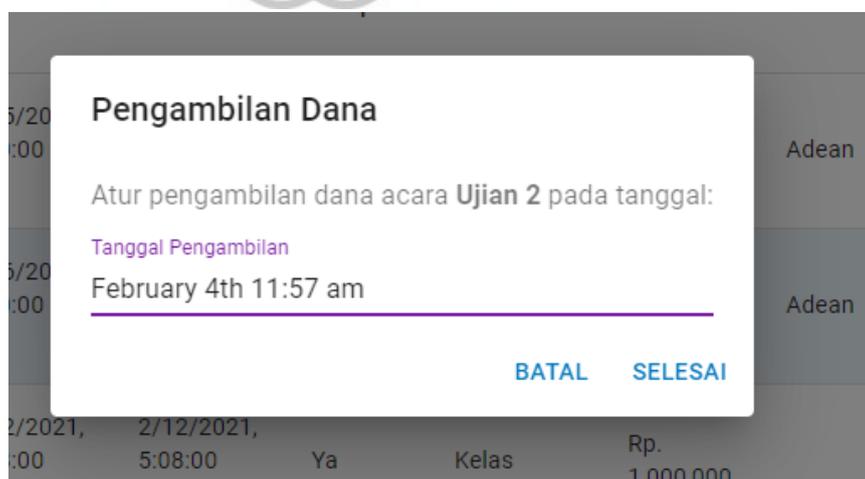
1. *Heading* 1.
2. *Form Field* pencarian nama tempat.
3. Tabel.

5.1.15 Implementasi Antarmuka Menu Admin Biro Administrasi Keuangan

Pada admin biro administrasi keuangan, terdapat satu menu yaitu acara. Biro administrasi keuangan ini berfungsi untuk menentukan tanggal pengambilan dana pada organisasi pengaju dari kalangan fakultas maupun universitas yang sudah disetujui sebelumnya oleh admin terkait.



Gambar 5.56. Menu acara admin keuangan



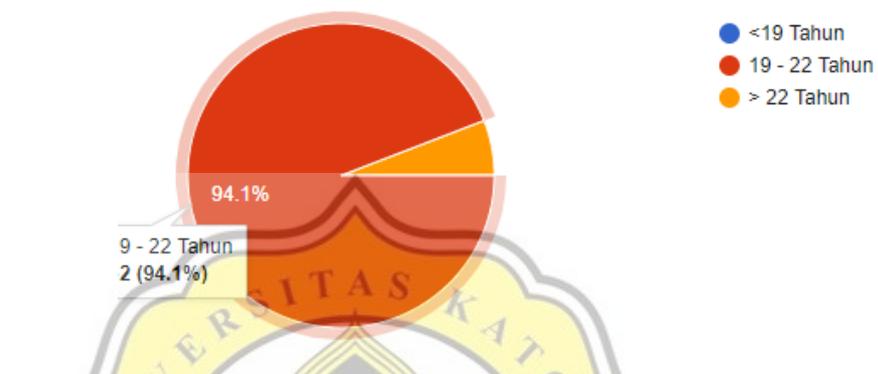
Gambar 5.57. Set tanggal jam pengambilan dana

5.2 Analisa data responden

5.2.1 Profil Responden

1. Usia

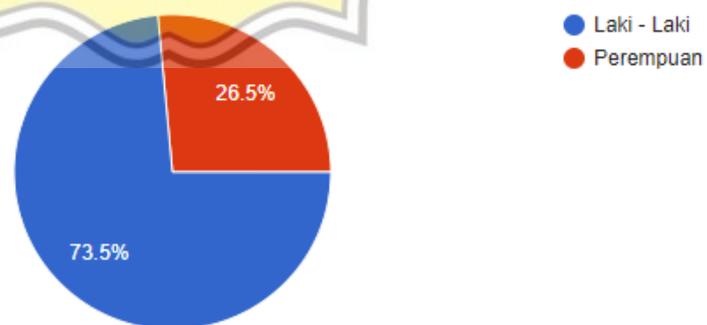
Responden terbanyak adalah berusia 19 hingga 22 tahun dengan jumlah responden 32 orang, sangat wajar karena responden adalah bagian dari mahasiswa.



Gambar 5.58. Diagram Umur Responden

2. Jenis Kelamin

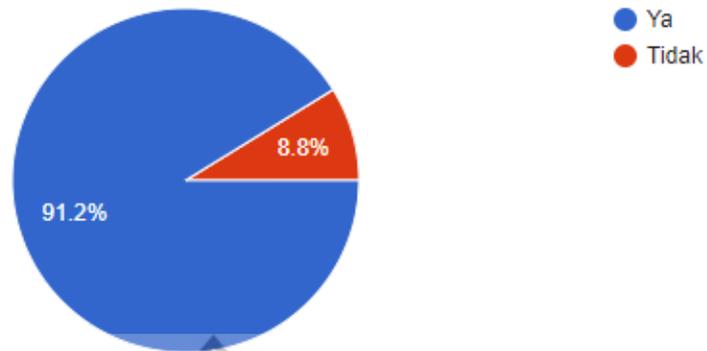
Responden terbanyak adalah berjenis kelamin Laki-Laki dengan jumlah 25 orang dan 9 orang lainnya adalah perempuan.



Gambar 5.59. Diagram Jenis Kelamin Responden

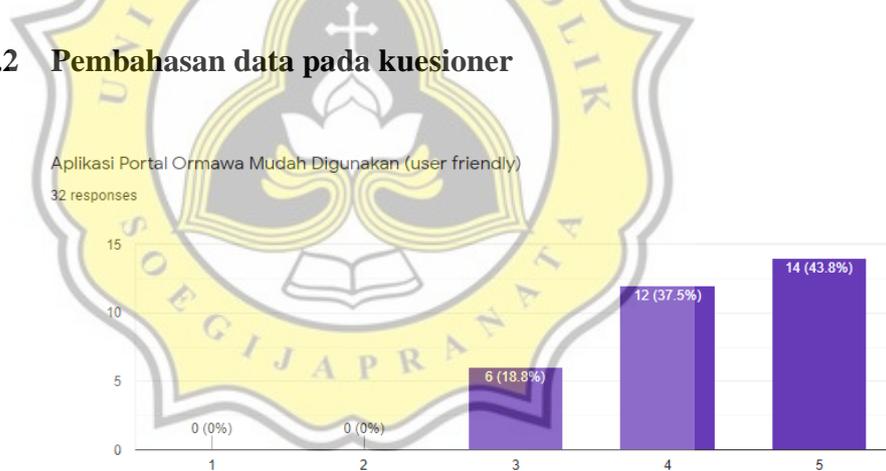
3. Anggota Organisasi Mahasiswa atau UKM

Responden rata – rata adalah anggota ormawa atau ukm sebanyak 32 Orang dan sisanya buka anggota sehingga data yang sah adalah 32 orang tersebut.



Gambar 5.60. Diagram Keanggotaan Ormawa Atau UKM

5.2.2 Pembahasan data pada kuesioner

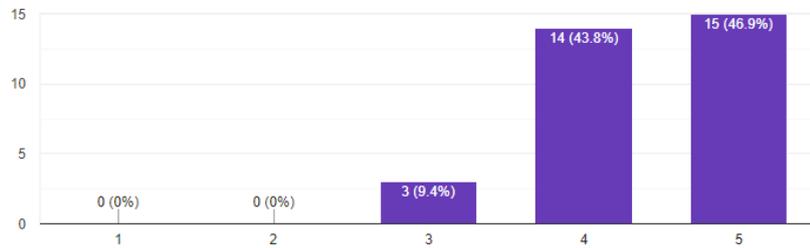


Gambar 4.87. Grafik Aplikasi Portal Ormawa Mudah Digunakan

Pada grafik di atas menunjukkan bahwa 14 responden sangat setuju aplikasi mudah digunakan, 12 responden setuju aplikasi mudah digunakan, dan 6 responden netral aplikasi mudah digunakan

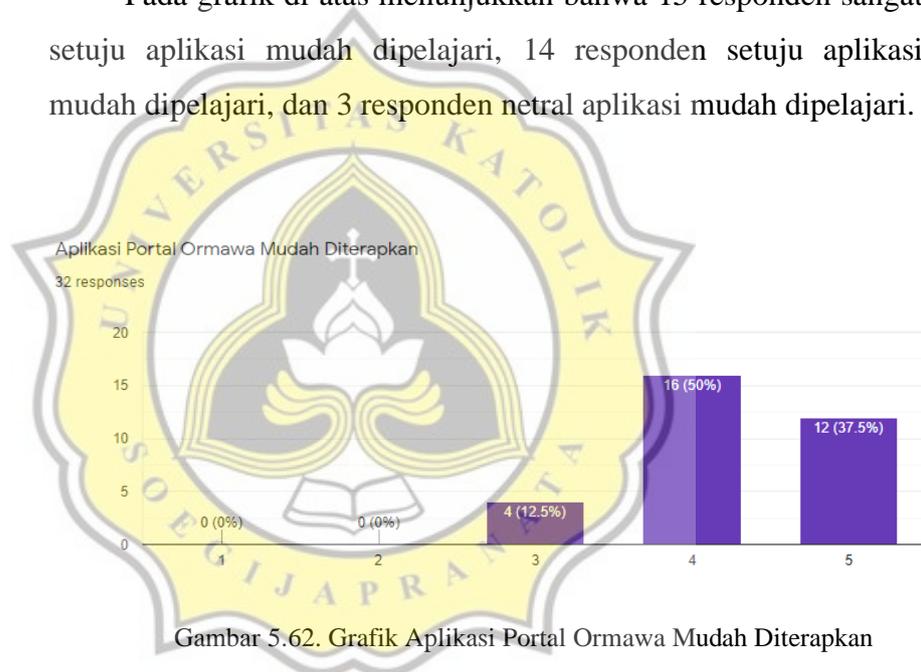
Aplikasi Portal Ormawa Mudah Dipelajari

32 responses



Gambar 5.61. Grafik Aplikasi Portal Ormawa Mudah Dipelajari

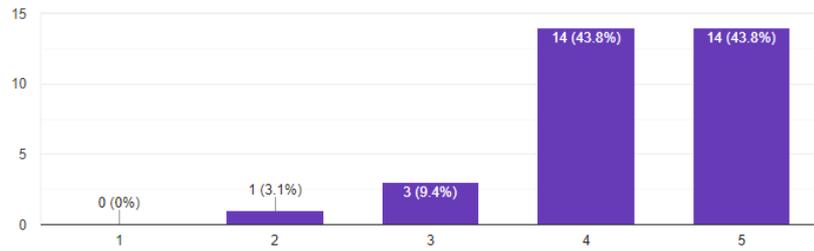
Pada grafik di atas menunjukkan bahwa 15 responden sangat setuju aplikasi mudah dipelajari, 14 responden setuju aplikasi mudah dipelajari, dan 3 responden netral aplikasi mudah dipelajari.



Gambar 5.62. Grafik Aplikasi Portal Ormawa Mudah Diterapkan

Pada grafik di atas menunjukkan bahwa 12 responden sangat setuju aplikasi mudah diterapkan, 16 responden setuju aplikasi mudah diterapkan, dan 4 responden netral aplikasi mudah diterapkan.

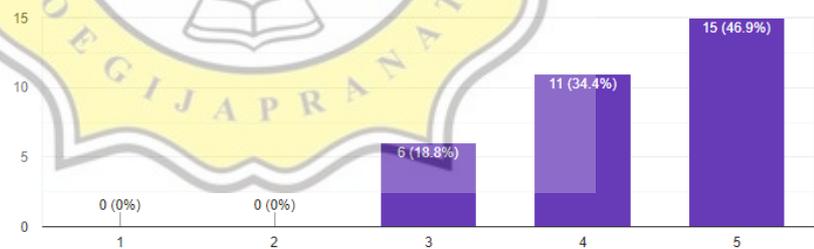
Aplikasi Portal Ormawa membantu saya untuk mengajukan proposal dan peminjaman
32 responses



Gambar 5.63 Grafik Aplikasi Portal Ormawa Membantu Untuk Mengajukan Proposal dan Peminjaman

Pada grafik diatas menunjukkan bahwa 14 responden sangat setuju aplikasi membantu untuk mengajukan proposal dan peminjaman, 14 responden setuju aplikasi membantu untuk mengajukan proposal dan peminjaman, 3 responden netral setuju aplikasi membantu untuk mengajukan proposal dan peminjaman, dan 1 responden tidak setuju setuju aplikasi membantu untuk mengajukan proposal dan peminjaman.

Aplikasi Portal Ormawa dapat bermanfaat untuk mempercepat pengajuan acara dan peminjaman
32 responses



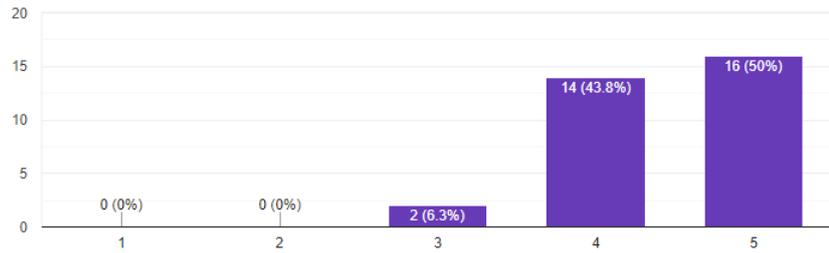
Gambar 5.64. Grafik Aplikasi Portal Ormawa Bermanfaat Untuk Mempercepat Pengajuan dan Peminjaman

Pada grafik di atas menunjukkan bahwa 15 responden sangat setuju aplikasi bermanfaat untuk mempercepat pengajuan dan peminjaman, 11 responden setuju aplikasi bermanfaat untuk mempercepat pengajuan dan peminjaman, dan 6 responden netral aplikasi bermanfaat untuk mempercepat pengajuan dan peminjaman.

Aplikasi Portal Ormawa dapat membantu saya dalam dokumentasi pengajuan acara dan peminjaman



32 responses

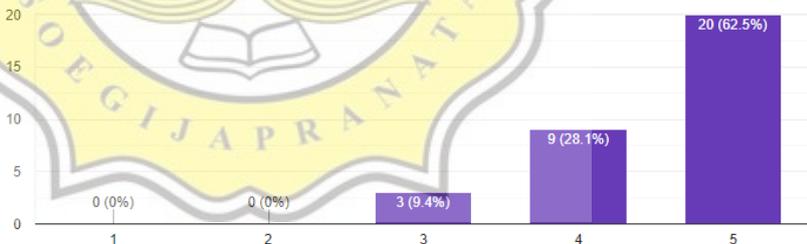


Gambar 5.65. Grafik Aplikasi Portal Ormawa Bermanfaat Untuk Mempercepat Pengajuan dan Peminjaman

Pada grafik di atas menunjukkan bahwa 16 responden sangat setuju aplikasi dapat membantu dalam dokumentasi pengajuan acara dan peminjaman, 14 responden setuju aplikasi dapat membantu dalam dokumentasi pengajuan acara dan peminjaman, dan 2 responden netral aplikasi dapat membantu dalam dokumentasi pengajuan acara dan peminjaman.

Saya memiliki jaringan internet untuk menggunakan Aplikasi Portal Ormawa

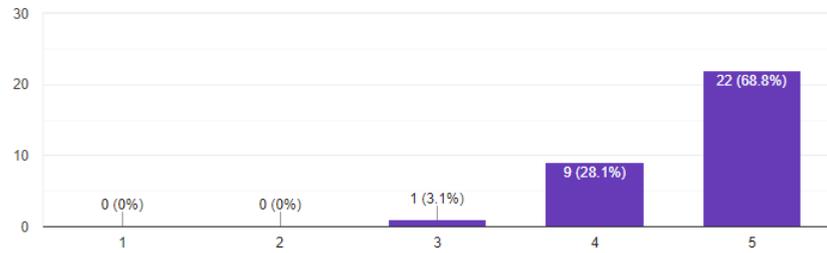
32 responses



Gambar 5.66. Grafik Memiliki Jaringan Internet Untuk Menggunakan Aplikasi Portal Ormawa

Pada grafik di atas menunjukkan bahwa 20 responden sangat setuju memiliki jaringan internet untuk menggunakan aplikasi, 9 responden setuju memiliki jaringan internet untuk menggunakan aplikasi, dan 3 responden netral memiliki jaringan internet untuk menggunakan aplikasi.

Saya memilik gadget (Laptop, PC, Handphone) untuk menggunakan Aplikasi Portal Ormawa
32 responses



Gambar 5.67. Grafik Memiliki Gadget Untuk Menggunakan Aplikasi Portal Ormawa

Pada grafik di atas menunjukkan bahwa 22 responden sangat setuju memiliki gadget untuk menggunakan aplikasi, 9 responden setuju memiliki gadget untuk menggunakan aplikasi, dan 1 responden netral memiliki gadget untuk menggunakan aplikasi.

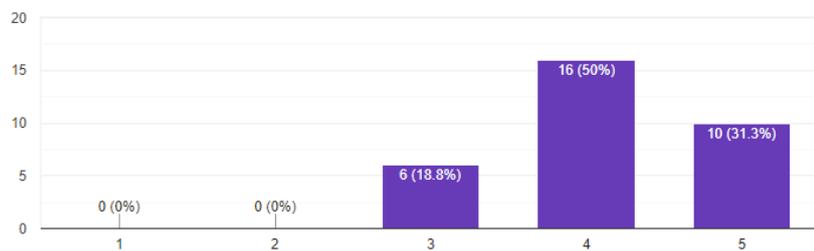


Gambar 5.68. Grafik Mampu Menggunakan Gadget Dalam Pengoperasian Aplikasi

Pada grafik di atas menunjukkan 18 responden sangat setuju mampu menggunakan gadget yang diperlukan dalam pengoperasian aplikasi, 12 responden setuju mampu menggunakan gadget yang diperlukan dalam pengoperasian aplikasi, dan 2 responden netral mampu menggunakan gadget yang diperlukan dalam pengoperasian aplikasi.

Saya akan dapat bantuan dari pengguna lain jika saya kesulitan menggunakan aplikasi Portal Ormawa

32 responses

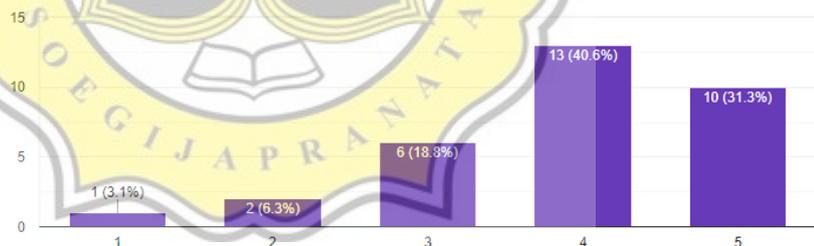


Gambar 5.69. Grafik Saya Akan Dapat Bantuan Dari Pengguna Lain Jika Saya Kesulitan Menggunakan Aplikasi

Pada grafik di atas menunjukkan bahwa 10 responden sangat setuju akan dapat bantuan dari pengguna lain jika kesulitan menggunakan aplikasi, 16 responden setuju akan dapat bantuan dari pengguna lain jika kesulitan menggunakan aplikasi, 6 responden netral akan dapat bantuan dari pengguna lain jika kesulitan menggunakan aplikasi.

Seseorang yang mempengaruhi saya berpikir aplikasi ini baik untuk kegiatan berorganisasi

32 responses

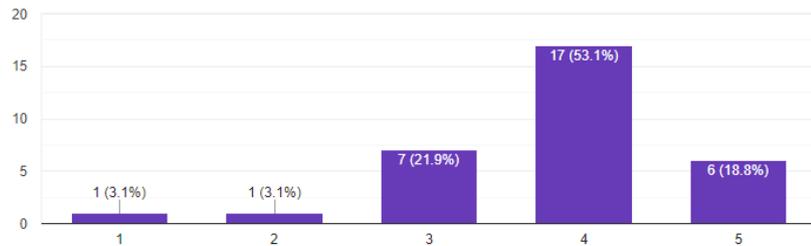


Gambar 5.70. Grafik Seseorang Yang Mempengaruhi Saya Berpikir Aplikasi Ini Baik Untuk Kegiatan Berorganisasi

Pada grafik di atas menunjukkan bahwa 10 responden sangat setuju seseorang yang mempengaruhi saya berpikir aplikasi ini baik untuk kegiatan berorganisasi, 13 responden setuju seseorang yang mempengaruhi saya berpikir aplikasi ini baik untuk kegiatan berorganisasi, 6 responden netral seseorang yang mempengaruhi saya berpikir aplikasi ini baik untuk kegiatan berorganisasi, 2 responden tidak setuju seseorang yang mempengaruhi saya berpikir

aplikasi ini baik untuk kegiatan berorganisasi, dan 1 responden seseorang yang mempengaruhi saya berpikir aplikasi ini baik untuk kegiatan berorganisasi.

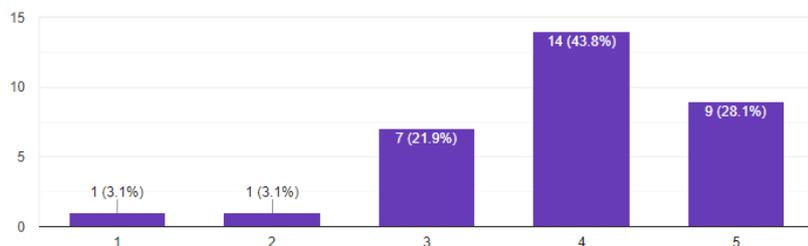
Seseorang yang penting berpikir saya harus menggunakan aplikasi ini
32 responses



Gambar 5.71. Grafik Seseorang Yang Penting Berpikir Saya Harus Menggunakan Aplikasi Ini

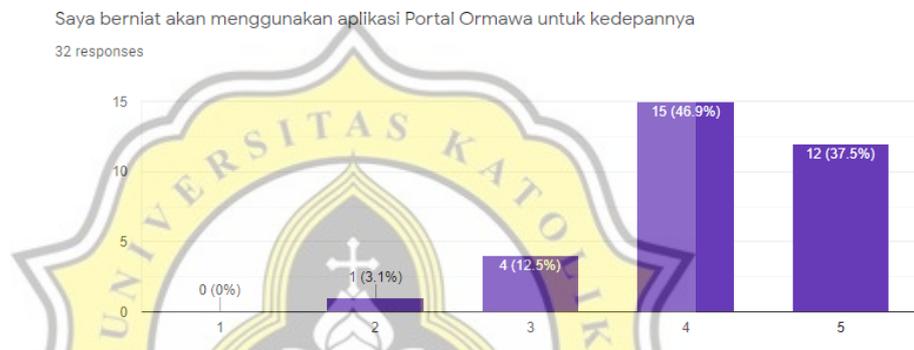
Pada grafik diatas menunjukkan bahwa 6 responden sangat setuju seseorang yang penting berpikir saya harus menggunakan aplikasi ini, 17 responden setuju seseorang yang penting berpikir saya harus menggunakan aplikasi ini, 7 responden netral seseorang yang penting berpikir saya harus menggunakan aplikasi ini, 1 responden tidak setuju seseorang yang penting berpikir saya harus menggunakan aplikasi ini, dan 1 responden sangat tidak setuju seseorang yang penting berpikir saya harus menggunakan aplikasi ini.

Teman/Kolega saya berfikir menggunakan aplikasi ini baik untuk saya
32 responses



Gambar 5.72. Grafik Teman/Kolega Saya Berpikir Menggunakan Aplikasi Ini Baik Untuk Saya

Pada grafik diatas menunjukkan bahwa 9 responden sangat setuju teman/kolega saya berpikir menggunakan aplikasi ini baik untuk saya, 14 responden setuju teman/kolega saya berpikir menggunakan aplikasi ini baik untuk saya, 7 responden netral teman/kolega saya berpikir menggunakan aplikasi ini baik untuk saya, 1 responden tidak setuju teman/kolega saya berpikir menggunakan aplikasi ini baik untuk saya, dan 1 responden sangat tidak setuju teman/kolega saya berpikir menggunakan aplikasi ini baik untuk saya.

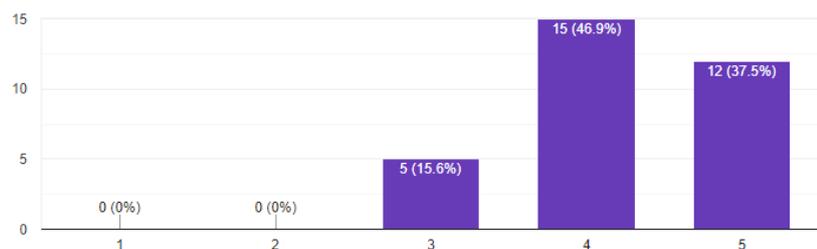


Gambar 5.73. Grafik Saya Berniat Akan Menggunakan Aplikasi Untuk Masa depan

Pada grafik diatas menunjukkan bahwa 12 responden sangat setuju saya berniat akan menggunakan aplikasi untuk masa depan, 15 responden setuju saya berniat akan menggunakan aplikasi untuk masa depan, 4 responden netral saya berniat akan menggunakan aplikasi untuk masa depan, dan 1 responden tidak setuju saya berniat akan menggunakan aplikasi untuk masa depan

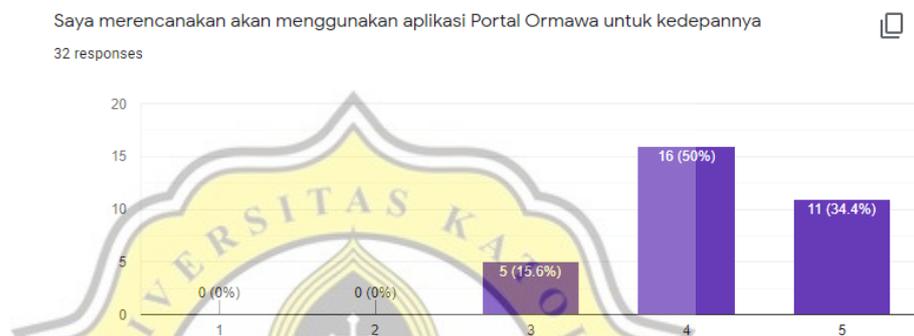
Saya akan terus menggunakan aplikasi Portal Ormawa untuk kedepannya

32 responses



Gambar 5.74. Grafik Saya Akan Terus Menggunakan Aplikasi Untuk Masa Depan

Pada grafik diatas menunjukkan bahwa 12 responden sangat setuju saya akan terus menggunakan aplikasi untuk masa depan, 15 responden setuju saya akan terus menggunakan aplikasi untuk masa depan, dan 5 responden netral saya akan terus menggunakan aplikasi untuk masa depan.



Gambar 5.75. Grafik Saya Merencanakan Akan Menggunakan Aplikasi Untuk Masa Depan

Pada grafik diatas menunjukkan bahwa 11 responden sangat setuju saya akan terus menggunakan aplikasi untuk masa depan, 16 responden setuju saya akan terus menggunakan aplikasi untuk masa depan, dan 5 responden netral saya akan terus menggunakan aplikasi untuk masa depan.

5.3 Uji Validitas

Validitas adalah suatu ukuran yang menunjukkan kesahihan suatu instrumen. Maka uji validitas memiliki fungsi untuk mengacu pada sejauh mana suatu instrumen menjalankan fungsi. Instrumen disini adalah variabel yang diuji pada kuesioner, variabel yang uji adalah EE, PE, FC,SI terhadap BI.

Rotated Component Matrix^a

	Component			
	1	2	3	4
EE1	,610	,443	,234	,317
EE2	,660	,336	,221	,079
EE3	,805	,001	,389	,202
PE1	,621	,545	,386	,135
PE2	,229	,589	,407	,361
PE3	,724	,369	,255	-,130
FC1	,237	,048	,860	,219
FC2	,096	,335	,791	-,030
FC3	,277	,171	,908	,074
SI1	,054	,127	,267	,718
SI2	,255	-,224	-,062	,773
SI3	-,004	,331	,128	,796
SI4	,007	,276	,006	,894
BI1	,651	,586	-,063	,123
BI2	,216	,804	,332	,245
BI3	,436	,827	,180	,087

Tabel 5.1. Tabel Hasil Uji Validitas Pertama

Uji validitas pertama, ditemukan bahwa variabel PE khususnya PE2 tidak mencapai angka validitas yaitu 0,5. Maka akan diuji kembali dengan menghilangkan variabel PE2.

Rotated Component Matrix^a

	Component		
	1	2	3
EE1	,719	,300	,339
EE2	,689	,286	,081
EE3	,554	,480	,174
PE1	,788	,442	,158
PE3	,763	,324	-,127
FC1	,134	,881	,210
FC2	,245	,795	,002
FC3	,252	,934	,081
SI1	,078	,272	,719
SI2	,003	-,034	,725
SI3	,190	,139	,839
SI4	,160	,012	,919
BI1	,875	,007	,160
BI2	,671	,345	,312
BI3	,866	,219	,152

Tabel 5.2. Tabel Hasil Uji Validitas Kedua

Pada tabel diatas terlihat bahwa setiap variabel mengelompok dan bernilai diatas 0,5.

5.4 Uji Reliabilitas

Tujuan pengujian reliabilitas adalah mengukur konsistensi sebuah kuesioner atau data pengujian. Berikut rentang nilai pengujian reliabilitas,

Cronbach's alpha	Internal consistency
$\alpha \geq 0.9$	Excellent
$0.9 > \alpha \geq 0.8$	Good
$0.8 > \alpha \geq 0.7$	Acceptable
$0.7 > \alpha \geq 0.6$	Questionable
$0.6 > \alpha \geq 0.5$	Poor
$0.5 > \alpha$	Unacceptable

Tabel 5.3. Tabel Rentang Nilai Reliabilitas

VARIABEL	CRONBACH'S ALPHA	INTERNAL CONSISTENCY
EE	,810	GOOD
PE	,809	GOOD
FC	,896	GOOD
SI	,837	GOOD
BI	,889	GOOD

Tabel 5.4. Tabel Hasil Uji Reliabilitas

Pada tabel diatas dapat dilihat bahwa semua variabel yang saya uji menunjukkan nilai *GOOD* yang berarti semua variabel yang saya uji dapat dipertanggungjawabkan dan bersifat konsisten.

5.5 Uji Korelasi

Uji korelasi berfungsi untuk menguji atau mencari hubungan antar variabel yang bersifat kualitatif atau berbentuk kuesioner seperti yang saya lakukan. Berikut tabel hasil uji korelasi.

		EE	PE	FC	SI	BI
EE	Pearson Correlation	1	,716**	,569**	,377*	,718**
	Sig. (2-tailed)		,000	,001	,037	,000
	N	31	31	31	31	31
PE	Pearson Correlation	,716**	1	,635**	,424*	,803**
	Sig. (2-tailed)	,000		,000	,017	,000
	N	31	31	31	31	31
FC	Pearson Correlation	,569**	,635**	1	,236	,449*
	Sig. (2-tailed)	,001	,000		,202	,011
	N	31	31	31	31	31
SI	Pearson Correlation	,377*	,424*	,236	1	,343
	Sig. (2-tailed)	,037	,017	,202		,059
	N	31	31	31	31	31
BI	Pearson Correlation	,718**	,803**	,449*	,343	1
	Sig. (2-tailed)	,000	,000	,011	,059	

N	31	31	31	31	31
---	----	----	----	----	----

** . Correlation is significant at the 0.01 level (2-tailed).

* . Correlation is significant at the 0.05 level (2-tailed).

Tabel 5.5. Tabel Hasil Uji Korelasi

Data yang berkorelasi ditandai dengan nilai yang lebih dari 0,4, maka dapat disimpulkan bahwa :

- a. Dari tabel diatas dapat dilihat bahwa nilai arah hipotesis (Sig. (2-tailed)) menunjukkan hasil $<0,05$ dan nilai sebesar 0,718 EE memiliki korelasi yang signifikan dan positif dengan BI.
- b. Dari tabel diatas dapat dilihat bahwa nilai arah hipotesis (Sig. (2-tailed)) menunjukkan hasil $<0,05$ dan nilai sebesar 0,803 PE memiliki korelasi yang signifikan dan positif dengan BI.
- c. Dari tabel diatas dapat dilihat bahwa nilai arah hipotesis (Sig. (2-tailed)) menunjukkan hasil $<0,05$ dan nilai sebesar 0,449 FC tidak memiliki korelasi yang signifikan dan positif dengan BI.
- d. Dari tabel diatas dapat dilihat bahwa nilai arah hipotesis (Sig. (2-tailed)) menunjukkan hasil $>0,05$ dan nilai sebesar 0,343 SI memiliki korelasi yang signifikan dan positif dengan BI. Dengan pendapat penulis sebagai peneliti, kemungkinan SI tidak berkorelasi karena *Social Influence* atau pengaruh dari luar tidak dapat mempengaruhi *Behavioral Intention* atau keinginan untuk terus menggunakan dikarenakan aplikasi ini baru diciptakan.