LAMPIRAN

Index.html

```
<!DOCTYPE html>
<html>

<head>
  <!-- Required meta tags-->
  <meta charset="utf-8">
  <meta name="viewport"
    content="width=device-width, initial-
scale=1, maximum-scale=1, minimum-scale=1, user-
scalable=no, minimal-ui, viewport-fit=cover">
  <meta name="apple-mobile-web-app-
capable" content="yes">
  <!-- Your app title -->
  <title>My App</title>
  <!-- Path to Framework7 Library Bundle CSS -->
  <link rel="stylesheet" href="css/framework7.bundle.min
.css">
  <link rel="stylesheet" href="css/framework7-
keypad.min.css">
  <!--  -->
  <link type="text/css" rel="stylesheet" href="https://www
.gstatic.com/firebasejs/ui/4.7.1/firebase-ui-auth.css" />
  <!-- Path to your custom app styles-->
  <link rel="stylesheet" href="css/app.css">
  <link rel="stylesheet" href="css/framework7-icons.css">
</head>

<body>
  <!-- App root element -->
  <div id="app">
    <div class="views tabs safe-areas">
      <!-- Additional "tabbar-labels" class -->
      <div id="mainToolbar" class="toolbar tabbar tabba
r-labels toolbar-bottom">
        <div class="toolbar-inner">
          <a href="#view-home" class="tab-link tab-
link-active" id="homeTab">
            <i class="icon f7-icons">house_alt</i>
            <span class="tabbar-label">Home</span>
          </a>
          <a href="#view-chat" class="tab-link">
            <i class="icon f7-
icons" id="chatIconTab">chat_bubble_2</i>
            <span class="tabbar-label">Chat</span>
          </a>
          <a href="#view-jual" class="tab-
link" style="background-color: #2ac9c9; color:white;">
            <i class="icon f7-
icons">camera_viewfinder</i>
            <span class="tabbar-label">Jual</span>
          </a>
          <a href="#view-iklan" class="tab-link">
            <i class="icon f7-
icons">person_2_square_stack</i>
            <span class="tabbar-
label">Iklan Saya</span>
          </a>
          <a href="#view-akun" class="tab-link">
            <i class="icon f7-icons">person</i>
            <span class="tabbar-label">Akun</span>
          </a>
        </div>
      </div>

      <div id="view-home" class="view view-
main tab tab-active"></div>
      <div id="view-chat" class="view tab"></div>
      <div id="view-jual" class="view tab"></div>
      <div id="view-iklan" class="view tab"></div>
      <div id="view-akun" class="view tab"></div>

      <!-- Login Screen -->
      <div class="login-screen" id="my-login-screen">
        <div class="view view-init" data-url="/login/">
        </div>
      </div>
    </div>
  </div>

  <!-- cordova.js required for cordova apps -->
  <script src="cordova.js"></script>
  <!-- Path to Framework7 Library Bundle JS-->
  <script type="text/javascript" src="js/framework7.bundl
e.min.js"></script>
  <script type="text/javascript" src="js/framework7-
keypad.min.js"></script>
  <!-- Firebase-->
  <script src="https://www.gstatic.com/firebasejs/8.1.1/fir
ebase-app.js"></script>
  <script src="https://www.gstatic.com/firebasejs/8.1.1/fir
ebase-firestore.js"></script>
  <!-
- <script src="https://www.gstatic.com/firebasejs/8.1.1/fire
base-auth.js"></script>
  <script src="https://www.gstatic.com/firebasejs/ui/4.7.1/
firebase-ui-auth__id.js"></script> -->
  <!-- App -->
  <script type="text/javascript" src="js/firebase.js"></scri
pt>
  <script type="text/javascript" src="js/moment.min.js"><
/script>
  <script type="text/javascript" src="js/id.js"></script>
  <script type="text/javascript" src="js/template7-
helper.js"></script>
  <script type="text/javascript" src="js/app.js"></script>
</body>

</html>
```

App.js

```
Framework7.use(Framework7Keypad);

let app = new Framework7({
  root: '#app',
  name: 'Mabbalanca',
  id: 'com.skripsi.ecommerce',
  // theme: 'ios',
  panel: {
    swipe: 'left'
  },
  view: {
    stackPages: true
  },
  statusbar: {
    androidBackgroundColor: '#42ddf5',
    androidTextColor: 'white'
  },
  // Add default routes
  routes: [{
    path: '/',
    componentUrl: './pages/home.html',
  }, {
    path: '/chat/',
    componentUrl: './pages/chat/chat.html',
  }, {
    name: 'message',
    path: '/message/:id/',
```

```
    componentUrl: './pages/chat/message.html',
  }, {
    path: '/jual/',
    componentUrl: './pages/jual/kategori.html',
  }, {
    path: '/detail /:kategori/',
    componentUrl: './pages/jual/detail .html'
  }, {
    path: '/iklan/',
    componentUrl: './pages/iklan/iklan.html'
  }, {
    name: 'detail Iklan',
    path: '/detail Iklan/:uid/:id/',
    componentUrl: './pages/iklan/detail Iklan.html'
  }, {
    path: '/ubahIklan/:id/',
    componentUrl: './pages/iklan/ubah.html'
  }, {
    path: '/lokasi/',
    componentUrl: './pages/lokasi/provinsi.html',
  }, {
    path: '/kabupaten/:name/:id_provinsi/',
    componentUrl: './pages/lokasi/kabupaten.html'
  }, {
    path: '/kecamatan/:name/:id_kabupaten/',
    componentUrl: './pages/lokasi/kecamatan.html'
  }, {
    path: '/search/',
    componentUrl: './pages/search/search.html'
  }, {
    path: '/searchLokasi/',
    componentUrl: './pages/lokasi/search.html'
  }, {
    path: '/searchProduk/:keyword/',
    componentUrl: './pages/search/searchProduk.html'
  }, {
    path: '/akun/',
    componentUrl: './pages/akun.html'
  }, {
    path: '/login/',
    componentUrl: './pages/login/login.html'
  }, {
    path: '/loginPhone/',
    componentUrl: './pages/login/loginPhone.html'
  }, {
    path: '/registrasi/',
    componentUrl: './pages/login/registrasi.html'
  }],
});

const ls = app.loginScreen.create({
  el: '#my-login-screen'
});

ls.open()

let $$ = Dom7;

// const URL = 'http://192.168.43.78/ecommerce-server/restapi/';
const URL = 'http://mabbalanca.000webhostapp.com/RestApi/';

// localStorage.clear()
// localStorage.setItem('uid', 1)
var uid = localStorage.getItem('uid')

if (navigator.connection.type == 'none') {
  app.dialog.alert('Anda tidak memiliki koneksi internet', 'Informasi');
}

function createView(newCreate = true) {
  if (!newCreate) {
    console.log(app.views);
    app.views.get('#view-home').router.refreshPage()
    app.views.get('#view-chat').router.refreshPage()
    app.views.get('#view-jual').router.refreshPage()
    app.views.get('#view-iklan').router.refreshPage()
    app.views.get('#view-akun').router.refreshPage()

    app.tab.show('#view-home')
    return
  }
  app.views.create('#view-home', {
    url: '/'
  });
  app.views.create('#view-chat', {
    url: '/chat/'
  });
  app.views.create('#view-jual', {
    url: '/jual/'
  });
  app.views.create('#view-iklan', {
    url: '/iklan/'
  });
  app.views.create('#view-akun', {
    url: '/akun/'
  });
}

function cekLogin(isSignedIn) {
  if (isSignedIn) {
    ls.close()
    createView()
  }
}

function playNotif() {
  var myMedia = new Media(cordova.file.applicationDirectory + 'www/audio/notification_tune.mp3');
  myMedia.play();
  myMedia.setVolume('0.2');
}

function notifSuccess(message) {
  console.log("Message type: " + message.messageType);
  if (message.messageType === "notification") {
    console.log("Notification message received");
    if (message.tap) {
      console.log("Tapped in " + message.tap);
      return
    }
    app.notification.create({
      icon: '<img src="./assets/logo.png" width="20">',
      title: message.title,
      subtitle: message.nama,
      text: message.pesan,
      closeTimeout: 4000,
      on: {
        open: playNotif
      }
    }).open()
  }
  console.dir(message);
}
```

```
document.addEventListener("deviceready", onDeviceRead
y, false);
document.addEventListener("backbutton", onBackKeyDo
wn, false);
document.addEventListener("pause", () => change User Stat
us('Offline'), false);
document.addEventListener("resume", () => change User St
atus('Online'), false);
// cek Login(true);

function onDeviceReady() {
  FirebasePlugin.onMessageReceived(notifSuccess);
  FirebasePlugin.is User SignedIn(cek Login);
  change User Status('Online')
}

function change User Status(status) {
  db.collection(`users`).doc(uid).set({
    status,
    waktu: firebase.firestore.FieldValue.serverTimestamp()
  })
}

let opened = 0;

function exitApp() {
  if (opened > 0) {
    return false;
  } else {
    app.dialog.confirm('Aplikasi akan ditutup?', 'Informasi',
      function () {
        navigator.app.exitApp();
      },
      function () {
        opened = 0;
        return false;
      }
    );
    opened = 1;
  }
}

function onBackKeyDown(e) {
  // Handle the back button
  if (ls.opened) {
    console.log(app.views.current.history.length);
    exitApp();
    e.preventDefault();
    return
  }

  if ($$('.modal-in').length > 0) {
    app.sheet.close('.modal-in')
  } else if (app.views.current.history.length == 1) {
    if (app.views.current.main) {
      exitApp();
      e.preventDefault();
    } else {
      app.tab.show('#view-home')
    }
  } else {
    app.dialog.close();
    app.views.current.router.back();
    return false;
  }
}
```

***Home*.html**

```
<template>
    <div class="page" id="homePage">
        <div class="navbar">
            <div class="navbar-bg"></div>
            <div class="navbar-inner">
                <div class="left"></div>
                <div class="title">
                    <a href="/lokasi/" class="button lokasi"><i cla
ss="f7-icons">placemark</i>{{lokasi.name}}<i
                            class="f7-icons">chevron_down</i></a>
                </div>
                <!-- <div class="right"><a class="button button-
large" @click="changeTheme"><i
                            class="icon f7-icons">bell</i></a>
                </div> -->
                <div class="subnavbar">
                    <!-- Searchbar with auto init -->
                    <a class="searchbar" href="/search/" data-
transition="f7-fade">
                        <div class="searchbar-inner">
                            <div class="searchbar-input-wrap">
                                <input type="search" placeholder="Te
mukan pakaian">
                                <i class="searchbar-icon"></i>
                            </div>
                        </div>
                    </a>
                </div>
            </div>
        </div>
        <div class="page-content hide-navbar-on-
scroll infinite-scroll-content">
            <div class="block block-strong">
                <div class="block-
header">Telusuri Kategori</div>
                <div class="row">
                    {{#each kategori}}
                    <a href="/search/?keyword=Pakaian {{title}}"
 class="link">
                        <div class="item-content">
                            <div class="item-
media"><img src="{{url}}" width="100px"></div>
                            <div class="item-inner title-
kategori">{{title}}</div>
                        </div>
                    </a>
                    {{/each}}
                </div>
            </div>

            <div class="block block-strong">
                <div class="block-
header">Rekomendasi Baru</div>
                <div class="list media-list">
                    <ul>
                        {{listProductHelper listProduct}}
                    </ul>
                </div>
            </div>
            <div class="preloader infinite-scroll-
preloader"></div>
        </div>
    </div>
</template>
<style>
    .lokasi {
        font-size: 14px;
        font-family: 'Alegreya';
    }
```

```
    .lokasi i {
        font-size: 15px;
    }

    .judul-produk {
        display: -webkit-box;
        -webkit-box-orient: vertical;
        -webkit-line-clamp: 2;
        overflow: hidden;
        font-size: 12px;
    }

    .lokasi-produk {
        display: -webkit-box;
        -webkit-box-orient: vertical;
        -webkit-line-clamp: 2;
        overflow: hidden;
        font-size: 12px;
        font-family: 'Alegreya';
    }

    img.demo-lazy {
        width: 100%;
        height: auto;
        max-height: 180px;
        display: block;
    }
</style>
<script>
    // script must return component object
    return {
        data: function () {
            const data = {
                kategori: [{
                    url: './assets/icon-wanita.png',
                    title: 'Wanita'
                }, {
                    url: './assets/icon-pria.png',
                    title: 'Pria'
                }, {
                    url: './assets/icon-anak.png',
                    title: 'Anak-anak'
                }],
                lokasi: {
                    id: null,
                    name: null
                },
                listProduct: [],
                newest: 0,
                allowInfinite: true
            }
            return data;
        },
        on: {
            pageInit: function () {
                const self = this;

                $$('.infinite-scroll-content').on('infinite', () => {
                    if (!this.allowInfinite) return;
                    self.$setState({
                        allowInfinite: false
                    })

                    this.getProduct().then(items => {
                        const newData = [
                            ...this.listProduct,
                            ...items
                        ]
                        self.$setState({
                            listProduct: newData
                        }, () => app.lazy.create('#homePage'))
                    })
                })

                this.getLokasi();
                this.refreshProduk();

                this.$app.on('getLokasi', this.getLokasi);
                this.$app.on('refreshProduk', this.refreshProduk);
            },
            pageBeforeRemove: function () {
                this.$app.off('getLokasi', this.getLokasi);
                this.$app.off('refreshProduk', this.refreshProduk);
            }
        },
        methods: {
            refreshProduk: function () {
                this.$setState({
                    listProduct: [],
                    newest: 0,
                    allowInfinite: false
                }).then(() => {
                    this.getProduct().then(items => {
                        this.$setState({
                            listProduct: items,
                            allowInfinite: true
                        }, () => {
                            app.lazy.create('#homePage')
                            app.infiniteScroll.create('.infinite-scroll-
content')
                        })
                    })
                })
            },
            getLokasi: function () {
                const data = {
                    uid
                }

                app.progressbar.show('aqua')
                app.request.promise.json(`${URL}lokasi`, data)
                    .then(res => {
                        app.progressbar.hide();

                        this.$setState({
                            lokasi: res.data
                        })
                    })
                    .catch(err => {
                        console.log(err);
                        app.progressbar.hide();
                        app.dialog.preloader('Tidak dapat terhubung
 ke server...');
                    })
            },
            getProduct: function () {
                const limit = 20
                let newest = this.newest

                const data = {
                    uid,
                    limit,
                    newest
                }

                app.progressbar.show('aqua')
                return app.request.promise.json(`${URL}searchP
roduk`, data)
```

```
            .then(res => {
              app.progressbar.hide();

              newest += limit;
              let allowInfinite = true;

              if (newest >= res.data.total) {
                allowInfinite = false;
                app.infiniteScroll.destroy('.infinite-scroll-content')
                // Remove preloader
                $$('.infinite-scroll-preloader').remove();
              }
              this.$setState({
                newest,
                allowInfinite
              })
              return res.data.produk
            })
            .catch(err => {
              console.log(err);
              app.progressbar.hide();
              app.dialog.preloader('Tidak dapat terhubung ke server...');
            })
        },
        detail Iklan: function () {
          const {
            id,
            uid,
            hargaformat,
            ...query
          } = $$(event.target).closest('li').dataset()

          query['hargaFormat'] = hargaformat

          const url = app.view.current.router.generateUrl({
            name: 'detail Iklan',
            params: {
              id,
              uid
            },
            query
          });

          app.view.current.router.navigate(url)
        }
      }
    }
</script>
```

Akun.html

```
<template>
  <div class="page">
    <div class="page-content tab tab-active">
      <div class="profil-card-header-pic">
        <div style="background-image:url(./assets/profile-background.jpg)"
          class="card-header display-flex justify-content-center">
          <img src="{{photoUrl}}" width="80px" style="border-radius: 50%;" class="lazy lazy-fade-in">
        </div>
      </div>
      <form id="formProfil" onsubmit="return false;" @submit="submitProfil">
        <div class="list no-hairlines">
          <ul>
            <li class="item-content item-input">
              <div class="item-inner">
                <div class="item-title item-label">Nama</div>
                <div class="item-input-wrap">
                  <input type="text" name="nama" placeholder="Masukkan nama" value="{{name}}">
                  <span class="input-save-button" @click="submitProfil"></span>
                </div>
              </div>
            </li>
            <li class="item-content item-input">
              <div class="item-inner">
                <div class="item-title item-label">Nomor Telpon</div>
                <div class="item-input-wrap">
                  <input type="text" name="telp" placeholder="Masukkan nomor telpon"
                    value="{{phoneNumber}}">
                  <span class="input-save-button" @click="submitProfil"></span>
                </div>
              </div>
            </li>
            <li class="item-content item-input">
              <div class="item-inner">
                <div class="item-title item-label">Email</div>
                <div class="item-input-wrap">
                  <input type="email" name="email" placeholder="Masukkan email" value="{{email}}">
                  <span class="input-save-button" @click="submitProfil"></span>
                </div>
              </div>
            </li>
          </ul>
        </div>
      </form>
      <div class="list simple-list">
        <ul>
          <li>
            <span>Dark Mode</span>
            <label class="toggle" id="toggleDarkMode">
              <input type="checkbox">
              <span class="toggle-icon"></span>
            </label>
          </li>
        </ul>
      </div>
      <div class="list">
        <ul>
          <li>
            <a href="#" class="item-link item-content" @click="logout">
              <div class="item-media"><i class="icon f7-icons">power</i></div>
              <div class="item-inner">
                <div class="item-title">Logout</div>
              </div>
            </a>
          </li>
        </ul>
      </div>

    </div>

  </div>
```

```
</template>
<style>
  .profil-card-header-pic .card-header {
    height: 30vw;
    background-size: cover;
    background-position: center;
    color: white;
    font-size: 24px;
    font-weight: bold;
  }

  .input-save-button {
    opacity: 0;
    pointer-events: none;
    visibility: hidden;
    transition-duration: 100ms;
    position: absolute;
    top: 50%;
    border: none;
    padding: 0;
    margin: 0;
    outline: 0;
    z-index: 1;
    cursor: pointer;
    background: none;
    width: var(--f7-input-clear-button-size);
    height: var(--f7-input-clear-button-size);
    margin-top: calc(-1 * var(--f7-input-clear-button-
size) / 2);
    color: var(--f7-input-clear-button-color);
    right: 0;
  }

  .input-save-button:after {
    font-family: 'Framework7 Icons';
    font-weight: normal;
    font-style: normal;
    line-height: 1;
    letter-spacing: normal;
    text-transform: none;
    white-space: nowrap;
    word-wrap: normal;
    direction: ltr;
    -webkit-font-smoothing: antialiased;
    text-rendering: optimizeLegibility;
    -moz-osx-font-smoothing: grayscale;
    font-feature-settings: "liga";
    text-align: center;
    display: block;
    width: 100%;
    height: 100%;
    font-size: 30px;
  }

  .input-save-button:before {
    position: absolute;
    content: '';
    left: 50%;
    top: 50%;
  }

  .item-input-wrap .input-save-button {
    top: 0.5px;
  }

  .input-save-button.active-state {
    opacity: 0.75 !important;
  }

  .input-with-value~.input-save-button,
  .item-input-with-value .input-save-button,
  .input-with-value .input-save-button {
    opacity: 1;
    pointer-events: auto;
    visibility: visible;
  }

  .input-outline .input-save-button,
  .item-input-outline .input-save-button {
    right: 8px;
  }

  .md .input-save-button:after {
    color: rgb(21, 201, 246);
    font-size: 35px;
    content: 'pencil_circle_fill';
    line-height: 1.2;
    top: 1;
  }

  .md .input-save-button:before {
    width: 48px;
    height: 48px;
    margin-left: -24px;
    margin-top: -24px;
  }
</style>
<script>
  // script must return component object
  return {
    data: function () {
      return null
    },
    on: {
      pageInit: function () {
        const self = this;

        this.toggleDarkMode = app.toggle.create({
          el: '#toggleDarkMode',
          on: {
            change: function () {
              localStorage.setItem('darkMode', this.che
cked)

              if (this.checked) {
                $$('body').addClass('theme-dark')
                app.statusbar.setBackgroundColor('#36
3636')
                return
              }
              $$('body').removeClass('theme-dark')
              app.statusbar.setBackgroundColor('#42dd
f5')
            }
          }
        });

        if (localStorage.getItem('darkMode') === 'true') {
          this.toggleDarkMode.toggle()
          app.statusbar.setBackgroundColor('#363636')
        }

        this.load User()

        this.action = app.actions.create({
          buttons: [{
              text: 'Data akan diubah?',
              label: true
```

```
                },
                {
                    text: 'Simpan',
                    icon: '<i class="icon f7-icons text-color-
blue">checkmark_alt</i>',
                    onClick: () => {
                        this.simpanData()
                    }
                },
                {
                    text: 'Batal',
                    icon: '<i class="icon f7-icons text-color-
red">xmark</i>'
                },
            ]
        })
    }
},
methods: {
    submitProfil: function () {
        this.action.open()
    },
    load User: function () {
        FirebasePlugin.getCurrent User((user) => {
            this.$setState(user)

            app.request.promise.json(`${URL}user`, {
                uid
            }).then(res => {
                const {
                    telp
                } = res.data

                if (this.providerId != 'phone') {
                    this.$setState({
                        phoneNumber: telp
                    })
                }
            })

            if (this.providerId == 'phone') {
                $$('input[name=telp]').attr('disabled', true)
            } else {
                $$('input[name=email]').attr('disabled', true)
            }
        });
    },
    logout: function () {
        app.dialog.confirm('Anda yakin untuk logout?', 'I
nformasi', () => {
            app.preloader.show();
            FirebasePlugin.signOut User(function () {
                FirebasePlugin.unregister();

                ls.open()
                app.preloader.hide();

                app.toast.create({
                    icon: '<i class="icon f7-
icons" style="font-size:22px;">checkmark_alt_circle</i>',
                    text: `Anda berhasil logout`,
                    closeTimeout: 2000,
                }).open()

                localStorage.clear();
                $$('body').removeClass('theme-dark')
            });
        })
    },
    simpanData: function () {

        const data = app.form.convertToData('#formProf
il');

        data['uid'] = this.uid

        app.preloader.show();
        app.request.promise.post(`${URL}user`, data, 'js
on')
            .then(res => {
                console.log(res);

                const data = res.data.data;

                app.preloader.hide();

                if (res.data.success) {
                    FirebasePlugin.update UserProfile({
                        name: data.nama,
                    });
                    if (this.providerId == 'phone') {
                        FirebasePlugin.update UserEmail(data.
email, function () {
                            console.log("User email successfully
 updated");
                        }, (err) => console.log(err));
                    }

                    app.toast.create({
                        icon: '<i class="icon f7-
icons" style="font-size:22px;">checkmark_alt_circle</i>',
                        text: `Data berhasil diubah`,
                        position: 'center',
                        closeTimeout: 2000,
                    }).open()

                } else {
                    app.toast.create({
                        icon: '<i class="icon f7-
icons" style="font-
size:22px;">exclamationmark_circle</i>',
                        text: `Data gagal diubah`,
                        position: 'center',
                        closeTimeout: 2000,
                    }).open()
                }
            })
        }
    }
}
</script>
```

Chat.html

```
<template>
    <div class="page" id="chatPage">
        <div class="navbar">
            <div class="navbar-bg"></div>
            <div class="navbar-inner">
                <div class="left"></div>
                <div class="title">Chat</div>
                <div class="right"></div>
                <div class="subnavbar">
                    <div class="toolbar tabbar toolbar-top">
                        <div class="toolbar-inner">
                            <!-- Links have "tab-
link" class instead of just "link" to switch tabs -->
                            <a href="#tab-semua" class="tab-
link tab-link-active">Semua</a>
```

```
                    <a href="#tab-membeli" class="tab-
link">Membeli</a>
                    <a href="#tab-menjual" class="tab-
link">Menjual</a>
                  </div>
                </div>
              </div>
            </div>
            <div class="page-content hide-navbar-on-scroll tabs-
swipeable-wrap">
              <div class="tabs">
                <div class="page-content tab tab-active" id="tab-
semua">
                  {{#js_if "this.listChat.length > 0"}}
                  <div class="list media-list">
                    <ul>
                      {{#each listChat}}
                      {{listChatHelper this}}
                      {{/each}}
                    </ul>
                  </div>
                  {{else}}
                  <div class="block">
                    <div style="display: flex;align-
items: center;justify-content: center;">
                      <div class="item-content text-align-
center" style="max-width: 50%;">
                        <div class="item-media">
                          <i class="f7-icons text-color-
deeporange"
                            style="font-
size:80px">exclamationmark_bubble</i>
                        </div>
                        <div class="item-inner title-
kategori" style="font-size:18px;">Anda belum
                          memiliki pesan
                        </div>
                      </div>
                    </div>
                  </div>
                  {{/js_if}}
                </div>
                <!-- Second tab -->
                <div class="page-content tab" id="tab-
membeli">
                  {{#js_if "this.listPembeli.length > 0"}}
                  <div class="list media-list">
                    <ul>
                      {{#each listPembeli}}
                      {{listChatHelper this}}
                      {{/each}}
                    </ul>
                  </div>
                  {{else}}
                  <div class="block">
                    <div style="display: flex;align-
items: center;justify-content: center;">
                      <div class="item-content text-align-
center" style="max-width: 50%;">
                        <div class="item-media">
                          <i class="f7-icons text-color-
deeporange"
                            style="font-
size:80px">exclamationmark_bubble</i>
                        </div>
                        <div class="item-inner title-
kategori" style="font-size:18px;">Anda belum
                          memiliki pesan
                        </div>
                      </div>
                    </div>
                  </div>
                  {{/js_if}}
                </div>
                <!-- Third tab -->
                <div class="page-content tab" id="tab-menjual">
                  {{#js_if "this.listPenjual.length > 0"}}
                  <div class="list media-list">
                    <ul>
                      {{#each listPenjual}}
                      {{listChatHelper this}}
                      {{/each}}
                    </ul>
                  </div>
                  {{else}}
                  <div class="block">
                    <div style="display: flex;align-
items: center;justify-content: center;">
                      <div class="item-content text-align-
center" style="max-width: 50%;">
                        <div class="item-media">
                          <i class="f7-icons text-color-
deeporange"
                            style="font-
size:80px">exclamationmark_bubble</i>
                        </div>
                        <div class="item-inner title-
kategori" style="font-size:18px;">Anda belum
                          memiliki pesan
                        </div>
                      </div>
                    </div>
                  </div>
                  {{/js_if}}
                </div>
              </div>
            </div>
          </div>
        </template>
        <script>
          // script must return component object
          return {
            data: function () {
              return {
                listChat: [],
                listPembeli: [],
                listPenjual: []
              };
            },
            on: {
              pageInit: function () {
                this.firebaseListen()
              },
              pageBeforeRemove: function () {
                this.firebaseUnsubscribe()
              }
            },
            methods: {
              toMessage: function () {
                const {
                  id,
                  ...query
                } = $$(event.target).closest('li').dataset()

                const url = app.views.current.router.generateUrl(
{
                  name: 'message',
```

```
                params: {
                    id
                },
                query
            });

            app.views.current.router.navigate(url)
        },
        firebaseListen: function () {
            const self = this

            this.firebaseUnsubscribe = db.collection(`groups`
)
                .where('members', "array-
contains", uid).where('show', '==', true)
                .orderBy('waktu', 'desc')
                .onSnapshot(function (querySnapshot) {
                    let listChat = querySnapshot.docs.map(chat
=> {
                        const dataChat = {
                            uid,
                            id: chat.id,
                            ...chat.data()
                        }

                        const res = app.request({
                            url: `${URL}chat`,
                            data: dataChat,
                            dataType: 'json',
                            async: false
                        })

                        if (res.status == 204) return

                        return {
                            ...dataChat,
                            ...JSON.parse(res.response)
                        }
                    })

                    listChat = listChat.filter(item => item);

                    let listPembeli = [],
                        listPenjual = [];

                    listChat.forEach(element => {
                        if (element['pembeli'] == uid) {
                            listPembeli.push(element)
                        } else
                        if (element['penjual'] == uid) {
                            listPenjual.push(element)
                        }
                    });

                    if (listChat.length > 0) {
                        $$('#chatIconTab')
                            .html(`chat_bubble_2
                        <span class="badge color-
red">${listChat.length}
                        </span>`);
                    } else {
                        $$('#chatIconTab').html(`chat_bubble_2`)
                    }

                    self.$setState({
                        listChat,
                        listPembeli,
                        listPenjual
                    }, () => app.lazy.create('#chatPage'))
                })
            }

        }
    }
</script>
```

Message.html

```
<template>
    <div class="page" id="messagePage">
        <div class="navbar">
            <div class="navbar-bg"></div>
            <div class="navbar-inner">
                <div class="left">
                    <a class="link back">
                        <i class="icon f7-icons">arrow_left</i>
                    </a>
                </div>
                <div class="title">
                    <div class="list media-list">
                        <a href="#" class="item-content">
                            <div class="item-media">
                                <img src="{{image}}" width="50px" s
tyle="max-height:50px;border-radius: 50%;">
                            </div>
                            <div class="item-inner">
                                <div class="item-title-row">
                                    <div class="item-
title"><b>{{nama*user*}}</b></div>
                                </div>
                                <div class="item-text" style="font-
size: 11px;">
                                    {{status}}</div>
                            </div>
                        </a>
                    </div>
                </div>
                <div class="right">
                    <a class="link popover-open" data-
popover="#popoverMessage">
                        <i class="icon f7-
icons">ellipsis_vertical</i>
                    </a>
                </div>
            </div>
        </div>
        <div class="toolbar messagebar">
            <div class=" toolbar-inner">
                <div class="messagebar-area">
                    <textarea placeholder="Pesan"></textarea>
                </div>
                <a href="#" @click="sendMessage" class="link
button button-fill text-color-white">
                    <i class="icon f7-
icons">arrow_right_circle_fill</i></a>
            </div>
        </div>
        <div class="page-content messages-content">
            <div class="messages">
                <div class="messages-
title"><b>{{judul}}</b></div>
                <div class="messages-
title">{{hargaformat}}</div>
            </div>
        </div>

        <div id="popoverMessage" class="popover">
            <div class="popover-inner">
                <div class="list">
```

```
                <ul>
                  <li><a class="list-button item-
link" href="#" @click="hapusMessage">Hapus Chat</a><
/li>
                </ul>
              </div>
            </div>
          </div>
        </div>
</template>
<script>
  // script must return component object
  return {
    data: function () {
      return {
        listChat: [],
        showMessage: false
      };
    },
    on: {
      pageInit: function () {
        this.$setState({
          ...this.$route.params,
          ...this.$route.query
        });
        // Init Messages
        this.messages = app.messages.create({
          el: '.messages',
          firstMessageRule: function (message, previous
Message, nextMessage) {
            if (message.isTitle) return false;
            if (!previousMessage || previousMessage.typ
e !== message.type || previousMessage
              .name !== message.name)
              return true;
            return false;
          },
          lastMessageRule: function (message, previous
Message, nextMessage) {
            if (message.isTitle) return false;
            if (!nextMessage || nextMessage.type !== m
essage.type || nextMessage.name !==
              message.name) return true;
            return false;
          },
          tailMessageRule: function (message, previous
Message, nextMessage) {
            if (message.isTitle) return false;
            if (!nextMessage || nextMessage.type !== m
essage.type || nextMessage.name !==
              message.name) return true;
            return false;
          }
        });

        // Init Messagebar
        this.messagebar = app.messagebar.create({
          el: '.messagebar'
        });

        this.firebaseStatusListen()

        if (!this.newMessage) {
          this.firebaseChatListen()
          this.firebaseInit()
        }
      },
      pageAfterIn: function (e, page) {
        app.toolbar.hide('#mainToolbar')
      },
      pageBeforeRemove: function () {
        this.messages.destroy()
        this.messagebar.destroy()
        this.unsubscribeChat()
        this.unsubscribe$User$Status()
      }
    },
    methods: {
      sendMessage: function () {
        const self = this

        const text = this.messagebar.getValue();

        if (!text.length) return;

        this.messagebar.clear();

        this.messagebar.focus();

        this.messages.addMessage({
          text: text.replace(/\n/g, '<br>').trim(),
          textFooter: moment(new Date()).format('h.mm
')
        });

        // Add a new document in collection

        db.collection(`chat`).doc(this.id).collection('mess
ages').add({
          sender: uid,
          text,
          waktu: firebase.firestore.FieldValue.serverTim
estamp()
        }).then(() => {
          if (self.newMessage) {
            self.firebaseChatListen()
            self.firebaseInit()
          }

          const to = self.pembeli != uid ? self.pembeli : s
elf.penjual
          app.request.post(`${URL}notifikasi`, {
            uid,
            to,
            text
          })
        })

        db.collection(`groups`).doc(this.id).update({
          last_pesan: text,
          show: true,
          waktu: firebase.firestore.FieldValue.serverTim
estamp()
        })
      },
      getStatusOffline: function (date) {
        const res = 'Terakhir dilihat'
        if (moment(date).isSame(moment(), 'day'))
          return `${res} hari ini pukul ${moment(date).f
ormat('hh.mm')}`

        return `${res} ${moment(date).fromNow()}`
      },
      firebaseInit: function () {
        const self = this

        db.collection(`chat/${this.id}/messages`).orderB
y('waktu', 'asc').get()
```

```
            .then(function (querySnapshot) {
              if (!self.newMessage) {
                const listMessages = self.queryFilter(querySnapshot)
                // self.messages.clear();
                self.messages.addMessages(listMessages)
              }
              self.$setState({
                showMessage: true,
                newMessage: false
              })
            })
        },
        firebaseStatusListen: function () {
          const sender = this.pembeli != uid ? this.pembeli : this.penjual

          this.unsubscribe*User*Status = db.collection(`*user*s`).doc(sender)
            .onSnapshot(querySnapshot => {
              const {
                status,
                waktu
              } = querySnapshot.data()

              const set = status == 'Online' ? status : this.getStatusOffline(waktu.toDate())

              this.$setState({
                status: set
              })
            })
        },
        firebaseChatListen: function () {
          const sender = this.pembeli != uid ? this.pembeli : this.penjual

          this.unsubscribeChat = db.collection(`chat/${this.id}/messages`)
            .where('sender', '==', sender).orderBy("waktu", "desc").limit(1)
            .onSnapshot(querySnapshot => {
              if (!this.showMessage) return
              const listMessages = this.queryFilter(querySnapshot)

              this.messages.addMessage(listMessages[0])
            });
        },
        queryFilter: function (querySnapshot) {
          const self = this

          return querySnapshot.docs.map(messages => {
            const {
              text,
              sender,
              waktu
            } = messages.data()

            const dataMessage = {
              text: text.replace(/\n/g, '<br>').trim(),
              textFooter: moment(waktu.toDate()).format('h.mm')
            }

            if (sender != uid) {
              dataMessage['avatar'] = self.foto*user*
              dataMessage['type'] = 'received'
            }
            return dataMessage
          })
        },
        hapusMessage: function () {
          const self = this

          app.actions.create({
            buttons: [{
                text: 'Anda yakin untuk menghapus chat ini?',
                label: true
              },
              {
                text: 'Hapus',
                icon: '<i class="icon f7-icons text-color-red">trash</i>',
                onClick: function () {
                  app.progressbar.show('aqua')

                  const gagalHapus = app.toast.create({
                    icon: '<i class="icon f7-icons" style="font-size:22px;">exclamationmark_bubble</i>',
                    text: 'Chat gagal dihapus',
                    position: 'center',
                    closeTimeout: 3000,
                  })

                  db.collection("groups").doc(self.id).delete().then(() => {
                    db.collection("chat").doc(self.id).delete()
                      .then(() => {
                        app.progressbar.hide()
                        app.views.current.router.back()
                      }).catch(err => {
                        app.progressbar.hide()
                        gagalHapus.open()
                      })
                  }).catch(err => {
                    app.progressbar.hide()
                    gagalHapus.open()
                  })
                }
              },
              {
                text: 'Batal',
                icon: '<i class="icon f7-icons text-color-blue">xmark</i>'
              },
            ]
          }).open()
        }
      }
    }
</script>
```

Iklan.html

```
<template>
  <div class="page" id="iklanPage">
    <div class="navbar">
      <div class="navbar-bg"></div>
      <div class="navbar-inner">
        <div class="left"></div>
        <div class="title">Iklan Saya</div>
      </div>
    </div>
```

```
      <div class="page-content">
        {{#if loading}}
        <div class="list media-list skeleton-text skeleton-
effect-blink">
          <ul>
            {{#each '12345'}}
            <li class="item-content">
              <a href="#" class="item-link item-content">
                <div class="item-media">
                  <div class="skeleton-
block" style="width: 50px; height: 70px;"></div>
                </div>
                <div class="item-inner">
                  <div class="item-title-row">
                    <div class="item-title">----------
----------</div>
                    <div class="item-after">----------
</div>
                  </div>
                  <div class="item-subtitle">---------
</div>
                </div>
              </a>
            </li>
            {{/each}}
          </ul>
        </div>
        {{else}}
        {{#js_if "this.empty == 0"}}
        <div class="block">
          <div style="display: flex;align-
items: center;justify-content: center;">
            <div class="item-content text-align-
center" style="max-width: 50%;">
              <div class="item-media">
                <i class="f7-icons text-color-
deeporange" style="font-
size:80px">exclamationmark_bubble</i>
              </div>
              <div class="item-inner title-
kategori" style="font-
size:18px;">Anda belum memasang iklan
              </div>
              <div class="item-
inner"><a href="#" @click="pasangIklan"
                  class="button button-fill button-
round">Pasang
                  Iklan</a>
              </div>
            </div>
          </div>
        </div>
        {{/js_if}}
        {{/if}}

        <div id="vlIklan" class="list virtual-list media-
list"></div>
      </div>

    </div>
</template>
<script>
  // script must return component object
  return {
    data: function () {
      return {
        loading: true,
        empty: 0
      };
    },
    on: {
      pageInit: function () {
        this.action = app.actions.create({
          buttons: [{
              text: 'Anda yakin untuk menghapus iklan
ini?',
              label: true
            },
            {
              text: 'Hapus',
              icon: '<i class="icon f7-icons text-color-
red">trash</i>',
              onClick: this.aksiHapus
            },
            {
              text: 'Batal',
              icon: '<i class="icon f7-icons text-color-
blue">xmark</i>'
            },
          ]
        })

        this.$app.on('refreshIklan', this.refreshIklan);
        $$('#vlIklan').on('click', '.swipe-
hapus', () => this.action.open())
      },
      pageTabShow: function () {
        this.getIklan()
      },
      pageBeforeRemove: function () {
        this.vlIklan.destroy();
        this.$app.off('refreshIklan', this.refreshIklan);
      },
    },
    methods: {
      refreshIklan: async function () {
        await this.vlIklan.destroy();

        this.getIklan();
      },
      getIklan: function () {
        const data = {
          uid: localStorage.getItem('uid')
        }

        app.request.promise.json(`${URL}iklan`, data)
          .then(res => {
            this.$setState({
              loading: false,
              empty: res.data.length
            })

            if (!res.data.length > 0) {
              return;
            }

            this.vlIklan = app.virtualList.create({
              el: '#vlIklan',
              items: res.data,
              itemTemplate: `<li class="swipeout delet
ed-callback" data-id="{{id}}">
                <div class="swipeout-content">
                  <a href="#" class="item-link item-
content">
                    <div class="item-media">
```

```
                                    <img data-
src="{{image}}" class="lazy lazy-fade-
in" width="50px" style="max-height:67px;">
                                </div>
                                <div class="item-inner">
                                    <div class="item-title-row">
                                        <div class="item-
title">{{judul}}</div>
                                        <div class="item-
after">{{waktuFormatter ditambahkan_pada}}</div>
                                    </div>
                                    <div class="item-
subtitle">{{hargaFormat}}</div>
                                    <div class="item-text">
                                        <i class="icon f7-
icons" style="font-size: 16px;">eye</i>  
                                        Dilihat:
                                        {{dilihat}}
                                    </div>
                                </div>
                            </a>
                        </div>
                        <div class="swipeout-actions-right">
                            <a href="/ubahIklan/{{id}}/" class=
"color-aqua"><i
                                    class="icon f7-
icons">pencil_ellipsis_rectangle</i>
                                 Ubah</a></a>
                            <a href="#" class="swipe-
hapus color-red swipeout-overswipe"><i class="icon f7-
icons">trash</i>
                                 Hapus</a></a></a>
                        </div>
                    </li>`,
                    height: 95,
                    on: {
                        vlItemsAfterInsert: function () {
                            app.lazy.create('#iklanPage')
                        }
                    }
                })

            })
            .catch(err => {
                console.log(err);
                app.preloader.hide();
                app.dialog.preloader('Tidak dapat terhubung
 ke server...');
            })
    },
    pasangIklan: function () {
        app.tab.show('#view-jual')
    },
    aksiHapus: function () {
        const target = $$(app.swipeout.el)
        const id = target.data('id')

        app.progressbar.show('aqua')

        app.request.promise.post(`${URL}iklanDelete/`,
{
            id
        }, 'json').then(res => {
            app.progressbar.hide()

            console.log(res.data);

            if (!res.data.success) {
                app.toast.create({
                    icon: '<i class="icon f7-
icons" style="font-
size:22px;">exclamationmark_bubble</i>',
                    text: res.data.message,
                    position: 'center',
                    closeTimeout: 3000,
                }).open()
                return;
            }
            app.toast.create({
                icon: '<i class="icon f7-icons" style="font-
size:22px;">checkmark_alt</i>',
                text: res.data.message,
                position: 'center',
                closeTimeout: 3000,
            }).open()

            app.swipeout.delete(target)

            $$('.deleted-
callback').once('swipeout:deleted', () => {
                this.$setState({
                    empty: this.empty - 1
                })
            });

        })
    }
  }
}
</script>
```

***detail* Iklan.html**

```
<template>
  <div class="page" id="detail IklanPage">
    <div class="page-content hide-navbar-on-
scroll container">
      <div class="navbar navbar-transparent navbar-
custom">
        <div class="navbar-bg"></div>
        <div class="navbar-inner">
          <div class="left">
            <a class="link back">
              <i class="icon f7-icons">arrow_left</i>
            </a>
          </div>
          <div class="title"></div>
          <div class="right">
          </div>
        </div>
      </div>

      <div id="swiper-iklan" class="swiper-
container demo-swiper">
        <div class="swiper-wrapper">
          {{#each images}}
          <div class="swiper-slide">
            <img data-
src="{{this}}" @click="openPhotoBrowser" class="swiper
-lazy"
                style="height: 100%;max-
height: 250px;">
            <div class="swiper-lazy-preloader"></div>
          </div>
          {{/each}}
        </div>
        <div class="swiper-pagination"></div>
      </div>
      <div class="block block-strong">
```

```
            <div style="font-
size:18px;"><b>{{dataProduk.hargaFormat}}</b></div>
            <div class="judul-
produk">{{dataProduk.judul}}</div>
            <div class="lokasi-produk">
               <i class="icon f7-icons" style="font-
size:12px;">placemark</i>{{dataProduk.lokasi}}</i>
            </div>
         </div>

         {{#if loading}}
         <div class="block">
            <span class="progressbar-infinite"></span>
         </div>
         {{else}}
         <div class="block block-strong">
            <div style="font-size:14px;"><b>Detail
</b></div>
            {{#each dataDetail }}
            <div class="item-content">
               <div class="item-inner item-cell">
                  <div class="row">
                     <div class="item-cell">{{@key}}</div>
                     <div class="item-cell">{{this}}</div>
                  </div>
               </div>
            </div>
            {{/each}}
         </div>

         <div class="block block-strong">
            <div style="font-
size:14px;"><b>Deskripsi</b></div>
            <div style="overflow-wrap: break-word;">
               {{dataProduk.deskripsi}}
            </div>
         </div>
         {{/if}}
      </div>

      <div id="detail Toolbar" class="toolbar tabbar-
labels toolbar-bottom">
         <div class="toolbar-inner">
            <a @click="createGroupsMessage" class="tab-
link" style="background-color:#2ac9c9">
               <i class="icon f7-icons text-color-
white">chat_bubble_fill</i>
               <span class="tabbar-label text-color-
white">Chat</span>
            </a>

            <a href="tel:+6285255136996" class="tab-
link external" style="background-color: #42ddf5;">
               <i class="icon f7-icons text-color-
white">phone_circle_fill</i>
               <span class="tabbar-label text-color-
white">Telpon</span>
            </a>
         </div>
      </div>
   </div>
</template>
<style>
   .container {
      position: static
   }

   .navbar-custom {
      position: absolute;
      background-color: rgba(0, 0, 0, 0.05);
   }

   .demo-swiper .swiper-slide {
      font-size: 25px;
      font-weight: 300;
      display: flex;
      justify-content: center;
      align-items: center;
      background: #fff;
      color: #000;
   }

   .demo-swiper .swiper-slide {
      box-sizing: border-box;
      border: 1px solid #ddd;
      background: #ddd;
   }

   .demo-swiper {
      margin: 0px 0 35px;
      font-size: 18px;
      height: 250px;
   }

   .demo-swiper.demo-swiper-auto .swiper-slide {
      width: 85%;
   }

   .demo-swiper.demo-swiper-auto .swiper-slide:nth-
child(2n) {
      width: 70%;
   }

   .demo-swiper.demo-swiper-auto .swiper-slide:nth-
child(3n) {
      width: 30%;
   }
</style>
<script>
   // script must return component object
   return {
      data: function () {
         return {
            dataProduk: [],
            detail Produk: [],
            loading: true
         };
      },
      on: {
         pageInit: function () {
            this.getDetail Produk()
            app.toolbar.hide('#detail Toolbar')

            this.$setState({
               dataProduk: {
                  ...this.$route.params,
                  ...this.$route.query
               },
               images: [this.$route.query.image]
            }, () => {
               this.mySwiper = app.swiper.create('#swiper-
iklan', {
                  pagination: {
                     el: '.swiper-pagination',
                     type: 'bullets',
                  },
                  observer: true,
                  speed: 900,
```

```
                lazy: true
              })
            })
        },
        pageBeforeRemove: function () {
          this.mySwiper.destroy()
          this.photoBrowser.destroy()
        },
        pageAfterIn: function () {
          app.toolbar.hide('#mainToolbar')

          if (this.$route.params['uid'] != uid) {
            app.toolbar.show('#detail Toolbar')
          }

          this.klikIklan()
        }
      },
      methods: {
        getDetail Produk: function () {
          const data = {
            ...this.$route.params,
            get: 'All'
          }

          app.request.promise.json(`${URL}produk`, data)
            .then(res => {
              const images = JSON.parse(res.data.images)

              const detail  = ['kondisi', 'dilihat', 'kategori', 'tanggal']
              const dataDetail  = {};

              for (let item in res.data) {
                if (detail .includes(item)) {
                  const key = item.charAt(0).toUpperCase() + item.slice(1);
                  dataDetail [key] = res.data[item]

                  delete res.data[`${item}`]
                }
              }

              moment.locale('id')
              dataDetail ['Tanggal'] = moment(dataDetail ['Tanggal']).format('LL')
              dataDetail ['Dilihat'] = parseInt(dataDetail ['Dilihat']) + 1

              // res.data['deskripsi'] = res.data['deskripsi'].replace(/\n/g, '<br/>')
              delete res.data['images']

              this.$setState({
                dataProduk: res.data,
                dataDetail ,
                images,
                loading: false
              }, () => {
                this.photoBrowser = app.photoBrowser.create({
                  photos: images,
                  popupCloseLinkText: 'Tutup',
                  navbarOfText: 'dari',
                  theme: 'dark'
                })
              })
            })
            .catch(err => {
              console.log(err);
              app.progressbar.hide()
              app.dialog.preloader('Tidak dapat terhubung ke server...');
            })
        },
        openPhotoBrowser: function () {
          this.photoBrowser.open(this.mySwiper.activeIndex)

          this.photoBrowser.once('close', () => {
            this.mySwiper.activeIndex = this.photoBrowser.activeIndex
            this.mySwiper.update()
          });
        },
        klikIklan: function () {
          const data = this.$route.params
          app.request.promise.post(`${URL}iklan`, data)
        },
        createGroupsMessage: function () {
          const id_produk = this.$route.params['id']
          db.collection(`groups`).where('members', "array-contains", uid).where('id_produk', '==', id_produk)
            .get().then(querySnapshot => {
              let newMessage = undefined
              let id = querySnapshot.docs.map(doc => doc.id)[0]
              const penjual = this.$route.params['uid']
              const pembeli = uid

              if (!querySnapshot.size) {
                id = app.utils.id('xxxxxxxxxxxxxxxxxxxx')
                newMessage = true

                db.collection(`groups`).doc(id).set({
                  id_produk,
                  members: [pembeli, penjual],
                  pembeli,
                  penjual,
                  show: false,
                  waktu: firebase.firestore.FieldValue.serverTimestamp()
                })
              }

              const url = app.views.current.router.generateUrl({
                name: 'message',
                params: {
                  id
                },
                query: {
                  namauser: this.dataProduk.nama,
                  fotouser: this.dataProduk.foto,
                  image: this.images[0],
                  judul: this.dataProduk.judul,
                  hargaformat: this.dataProduk.hargaFormat,
                  pembeli,
                  penjual,
                  newMessage
                }
              });

              app.views.current.router.navigate(url)
            })
        }
```

```
    }
  }
</script>
```

Ubah.html

```
<template>
  <div class="page">
    <div class="navbar">
      <div class="navbar-bg"></div>
      <div class="navbar-inner">
        <div class="left">
          <a class="link back">
            <i class="icon f7-icons">arrow_left</i>
          </a>
        </div>
        <div class="title">Ubah *Detail* </div>
      </div>
    </div>

    <div class="page-content">
      <form id="*detail* Form">
        <div class="list no-hairlines-md">
          <ul>
            <li class="item-content">
              <div class="item-inner">
                <div class="item-input-wrap">
                  <a class="item-link smart-
select smart-select-init" data-open-in="sheet">
                    <select name="kondisi">
                      <option value="Baru" selected>
Baru</option>
                      <option value="Bekas">Bekas
</option>
                    </select>
                    <div class="item-content item-
input-wrap">
                      <div class="item-inner">
                        <div class="item-
title" style="color:gray">Kondisi</div>
                      </div>
                    </div>
                  </a>
                </div>
              </div>
            </li>
            <li class="item-content item-input item-
input-with-info">
              <div class="item-inner">
                <div class="item-input-wrap">
                  <input type="text" name="judul" pla
ceholder="Judul" required validate
                    data-error-
message="Judul belum diisi">
                  <span class="input-clear-
button"></span>
                  <div class="item-input-
info">Sebutkan fitur utama dari barang anda (misal merek,
                    model, umur, jenis)</div>
                </div>
              </div>
            </li>
            <li class="item-content item-input item-
input-with-info">
              <div class="item-inner">
                <div class="item-input-wrap">
                  <textarea type="text" name="deskri
psi" class="resizable" placeholder="Deskripsi"
                    required validate data-error-
message="Deskripsi belum diisi"></textarea>
                  <span class="input-clear-
button"></span>
                  <div class="item-input-
info">Sertakan kondisi, fitur, dan alasan penjualan</div>
                </div>
              </div>
            </li>
            <li class="item-content item-input item-
input-with-info">
              <div class="item-media">
                <label style="color:#42ddf5;">Rp</lab
el>
              </div>
              <div class="item-inner">
                <div class="item-input-wrap">
                  <input type="text" name="harga" pl
aceholder="Harga" required validate
                    data-error-
message="Harga belum diisi">
                  <!-- <span class="input-clear-
button"></span> -->
                  <div class="item-input-
info">Masukkan harga jual</div>
                </div>
              </div>
            </li>
            <li class="item-content" id="list-lokasi">
              <div class="item-inner">
                <div class="item-input-wrap">
                  <a href="/lokasi/" class="item-
link smart-select smart-select-init">
                    <select name="lokasi">
                      <option value="">Pilih Lokasi<
/option>
                    </select>
                    <div class="item-content item-
input-wrap">
                      <div class="item-inner">
                        <div class="item-
title" style="color:gray">Lokasi</div>
                      </div>
                    </div>
                  </a>
                </div>
              </div>
            </li>
          </ul>
        </div>
        <div class="block">
          <a class="col-100 medium-50 button button-
fill button-
round" href="#" @click="onSubmit">Lanjut</a>
        </div>
      </form>
    </div>

  </div>
</template>
<script>
  // script must return component object
  return {
    data: function () {
      const data = {
        kategori: [{
          url: './assets/icon-wanita.png',
          title: 'Wanita'
        }, {
          url: './assets/icon-pria.png',
          title: 'Pria'
```

```
            }, {
                url: './assets/icon-anak.png',
                title: 'Anak-anak'
            }]
        }
        return data;
    },
    on: {
        pageInit: function () {

            this.loadData()

            this.myKeypad = app.keypad.create({
                inputEl: 'input[name=harga]',
                formatValue: this.formatRupiah,
                toolbar: false,
                dotButton: false,
                on: {
                    open: function () {
                        $$('input, textarea').blur();
                    }
                }
            });

            this.action = app.actions.create({
                buttons: [{
                        text: 'Iklan akan diubah?',
                        label: true
                    },
                    {
                        text: 'Simpan',
                        icon: '<i class="icon f7-icons text-color-
blue">checkmark_alt</i>',
                        onClick: () => {
                            this.sendData()
                        }
                    },
                    {
                        text: 'Batal',
                        icon: '<i class="icon f7-icons text-color-
red">xmark</i>'
                    },
                ]
            })

            this.$app.on('setLokasi', this.setLokasi);
        },
        pageBeforeRemove: function () {
            this.$app.off('setLokasi', this.setLokasi);
        }
    },
    methods: {
        loadData: function () {
            const form = $$('#detail Form');

            const data = {
                uid: localStorage.getItem('uid'),
                id: this.$route.params.id
            }

            app.preloader.show()
            app.request.promise.json(`${URL}produk`, data)
                .then(res => {
                    app.preloader.hide()

                    app.form.fillFromData(form[0], res.data)
                    this.setLokasi(res.data.id_kecamatan, res.dat
a.lokasi)
                })
        },
        setLokasi: function (id, lokasi) {
            $$('#list-
lokasi > div > div > a > div > div > div.item-
after').text(lokasi)
            $$('select[name=lokasi]').html(`<option value="$
{id}" selected>${lokasi}</option>`)
        },
        onSubmit: function () {
            const form = $$('#detail Form');

            if (!form[0].checkValidity()) {
                return;
            }
            const formData = app.form.convertToData(form)
            const {
                lokasi
            } = formData

            if (!lokasi) {
                app.toast.create({
                    icon: '<i class="icon f7-icons" style="font-
size:22px;">exclamationmark_circle</i>',
                    text: 'Silahkan pilih lokasi terlebih dahulu',
                    closeTimeout: 2000,
                }).open()
                return;
            }

            this.action.open()
        },
        sendData: function () {
            const form = $$('#detail Form')
            let data = app.form.convertToData(form)
            data['id_kecamatan'] = data['lokasi']

            data['where'] = {
                id: this.$route.params.id,
                uid: localStorage.getItem('uid')
            }

            delete data['lokasi']

            const updateMethod = {
                url: `${URL}produkUpdate`,
                method: 'POST',
                data: data,
                dataType: 'json'
            }

            app.dialog.progress('Mengirim data...');
            app.request.promise(updateMethod)
                .then(res => {
                    app.dialog.close();
                    const response = res.data;

                    if (!response.success) {
                        app.toast.create({
                            icon: '<i class="icon f7-
icons" style="font-
size:22px;">exclamationmark_bubble</i>',
                            text: response.message,
                            position: 'center',
                            closeTimeout: 3000,
                        }).open()

                        return;
                    }
```

```
                    app.toast.create({
                        icon: '<i class="icon f7-
icons" style="font-size:22px;">checkmark_alt</i>',
                        text: response.message,
                        position: 'center',
                        closeTimeout: 3000,
                    }).open()

                    app.views.current.router.back(app.views.cur
rent.history[0])

                    app.emit('refreshIklan')
                })
                .catch(err => {
                    app.dialog.close();
                    console.log(err);
                    app.dialog.alert('Tidak dapat terhubung ke s
erver', 'Gagal');
                })
            },
            formatRupiah: function (angka) {
                const {
                    ...arrAngka
                } = angka

                if (arrAngka[0] == 0) {
                    this.myKeypad.setValue('');
                    return
                }

                let number_string = angka != null ? angka.replac
e(/[^,\d]/g, '').toString() : '',
                    split = number_string.split(','),
                    sisa = split[0].length % 3,
                    rupiah = split[0].substr(0, sisa),
                    ribuan = split[0].substr(sisa).match(/\d{3}/gi);

                // tambahkan titik jika yang di input sudah menja
di angka ribuan
                if (ribuan) {
                    separator = sisa ? '.' : '';
                    rupiah += separator + ribuan.join('.');
                }

                return rupiah = split[1] != undefined ? rupiah + ','
 + split[1] : rupiah;
            }
        }
    }
</script>
```

Login.html

```
<template>
    <div class="page login-screen-page" data-
name="login">
        <div class="page-content login-screen-content">
            <div class="login-screen-title">
                <p><img src="./assets/logo.png" width="200"><
/p>
            </div>
            <div class="list">
                <ul>
                    <li class="block">
                        <a class="button button-raised color-
orange display-flex" href="#" @click="loginGoogle">
                            <i class="icon f7-
icons">logo_google</i>
                              Masuk dengan Akun Google
                        </a>
                    </li>
                    <li class="block">
                        <a class="button button-raised color-
lightblue display-flex" href="/loginPhone/"><i
                                class="icon f7-
icons">phone</i>  Masuk
                            dengan Nomor Telpon</a>
                    </li>
                    <!-- <li class="block">
                        <a class="button button-raised color-
lime display-flex" href="/registrasi/">
                            <i class="icon f7-icons">envelope</i>
                              Masuk dengan Email
                        </a>
                    </li> -->
                </ul>
            </div>

        </div>
    </div>
</template>
<script>
    // script must return component object
    return {
        on: {
            pageInit: function () {
                this.getLocation()
            }
        },
        methods: {
            getLocation: function () {
                navigator.geolocation.getCurrentPosition((positio
n) => {
                    const data = {
                        latitude: position.coords.latitude,
                        longitude: position.coords.longitude
                    }
                    this.$setState(data)
                }, (err) => {
                    console.log(err)
                    if (err.code == 1) {
                        app.dialog.alert('Lokasi harus diaktifkan', 'In
formasi')
                    }
                }, {
                    enableHighAccuracy: true
                });
            },
            loginGoogle: function () {
                const self = this;
                const clientId = '7162941845-
h2bqfj7kar7a0f56vqgt51a048hrq3js.apps.googleuserconten
t.com'
                FirebasePlugin.authenticateUserWithGoogle(clie
ntId, function (credential) {
                    app.preloader.show();

                    self.signInWithCredential(credential);
                });
            },
            signInWithCredential: function (credential) {
                const self = this;
                FirebasePlugin.signInWithCredential(credential,
function () {
                    console.log("Successfully signed in");
                    FirebasePlugin.getCurrentUser(self.registrasiU
ser);
                }, function (error) {
                    console.error("Failed to sign in", error);
```

```
        });
      },
      registrasi*User*: function (*user*) {
        FirebasePlugin.getToken((token) => {
          console.log(*user*);
          const data*User* = {
            uid: *user*.uid,
            nama: *user*.name,
            email: *user*.email,
            telp: *user*.phoneNumber,
            foto: *user*.photoUrl,
            token
          }

          if (this.latitude && this.longitude) {
            data*User*['latitude'] = this.latitude
            data*User*['longitude'] = this.longitude
          }

          console.log(data*User*);

          app.request.promise.post(`${URL}*user*`, data*User*,
              'json')
            .then(res => {
              if (!res.data.success) {
                app.dialog.alert('Registrasi gagal dilakukan...');
                return
              }

              app.preloader.hide();

              localStorage.setItem('uid', *user*.uid);
              uid = *user*.uid

              createView(app.views.length == 1)
              ls.close()

              app.toast.create({
                icon: '<i class="icon f7-icons" style="font-size:22px;">checkmark_alt_circle</i>',
                text: `Selamat Datang ${*user*.name}`,
                closeTimeout: 2000,
              }).open()
            })
            .catch(err => {
              app.dialog.preloader('Tidak dapat terhubung ke server...');
              console.log(err);
            })
        })
      }
    }
  }
</script>

*login*Phone.html
<template>
  <div class="page">
    <div class="navbar navbar-transparent">
      <div class="navbar-bg"></div>
      <div class="navbar-inner">
        <div class="left">
          <a class="link back">
            <i class="icon f7-icons">arrow_left</i>
          </a>
        </div>
      </div>
    </div>
    <div class="page-content">
      <div class="*login*-screen-title">
        <p><img src="./assets/logo.png" width="200"></p>
      </div>

      <form onsubmit="return false;" @submit="verifyPhoneNumber">
        <div class="firebaseui-card-header">
          <h1 class="firebaseui-title">Masukkan nomor *telephone* Anda</h1>
          <span class="progressbar-infinite"></span>
        </div>
        <div class="firebaseui-card-content">
          <div class="list">
            <ul>
              {{#if beforeVerify}}
              <li class="item-content item-input item-input-with-info">
                <div class="item-media">
                  <span
                    class="firebaseui-flag firebaseui-country-selector-flag firebaseui-id-country-selector-flag firebaseui-flag-ID"></span><span
                    class="firebaseui-id-country-selector-code">&amp;lrm;+62</span>
                </div>
                <div class="item-inner">
                  <div class="item-title item-floating-label">Nomor Telpon</div>
                  <div class="item-input-wrap">
                    <input type="number" name="phoneNumber" required validate>
                    <span class="input-clear-button"></span>
                  </div>
                </div>
              </li>
              {{else}}
              <li class="item-content item-input item-input-with-info">
                <div class="item-inner">
                  <div class="item-title item-floating-label">Masukkan 6 Kode Verifikasi</div>
                  <div class="item-input-wrap">
                    <input type="number" name="kode" required validate>
                    <span class="input-clear-button"></span>
                  </div>
                </div>
              </li>
              {{/if}}
            </ul>
          </div>
        </div>
        <div class="firebaseui-card-actions">
          <div class="firebaseui-form-actions"><button type="submit"
              class="button button-fill">Verifikasi</button></div>
        </div>
        <div class="firebaseui-card-footer">
          <p class="firebaseui-tos firebaseui-phone-tos">Dengan mengetuk Verifikasi, SMS mungkin akan dikirim.
            Mungkin dikenakan biaya pesan &amp;amp; data.</p>
```

```
            </div>
          </form>

      </div>
    </div>
</template>
<style>
    .firebaseui-textfield.mdl-textfield .firebaseui-input {
        border-color: rgba(0, 0, 0, 0.12);
    }
</style>
<script>
    return {
        data: function () {
            return {
                beforeVerify: true
            }
        },
        on: {
            pageInit: function () {
                $$('input[name=phoneNumber]').focus()
                $$('.progressbar-infinite').hide()
            }
        },
        methods: {
            verifyPhoneNumber: function () {
                const self = this

                const timeOutDuration = 60;
                let awaitingSms = false;

                if (!this.beforeVerify) {
                    awaitingSms = false;
                    this.credential.code = $$('input[name=kode]').
val();
                    this.signInWithCredential(this.credential);
                    return
                }

                const inputValue = $$('input[name=phoneNumbe
r]').val()

                let number

                if (inputValue[0] == 0) {
                    number = inputValue.substring(1)
                } else {
                    number = inputValue
                }

                number = `+62${number}`

                $$('.progressbar-infinite').show()

                FirebasePlugin.verifyPhoneNumber(function (cre
dential) {
                    $$('.progressbar-infinite').hide()
                    self.$setState({
                        beforeVerify: false,
                        credential
                    })

                    if (credential.instantVerification) {
                        if (awaitingSms) {
                            awaitingSms = false;
                            $$('.progressbar-infinite').show()
                        }
                        console.log('instant ', credential);
                        self.signInWithCredential(credential);
                    } else {
                        awaitingSms = true;

                        $$('input[name=kode]').focus()
                        console.log(credential);

                    }
                }, function (error) {
                    console.error("Failed to verify phone number:
" + JSON.stringify(error));
                }, number, timeOutDuration);
            },
            signInWithCredential: function (credential) {
                const self = this;
                FirebasePlugin.signInWithCredential(credential,
function () {
                    console.log("Successfully signed in");
                    FirebasePlugin.getCurrent User(self.registrasiU
ser);
                }, function (error) {
                    console.error("Failed to sign in", error);
                });
            },
            registrasi User: function (user) {
                FirebasePlugin.getToken((token) => {
                    const data User = {
                        uid: user.uid,
                        nama: user.name,
                        email: user.email,
                        telp: user.phoneNumber,
                        foto: user.photoUrl,
                        latitude: this.latitude,
                        longitude: this.longitude,
                        token
                    }

                    console.log(data User);

                    app.request.promise.post(`${URL}user`, data
User,
                        'json')
                        .then(res => {
                            if (!res.data.success) {
                                app.dialog.alert('Registrasi gagal dilaku
kan...');
                                return
                            }

                            app.preloader.hide();

                            localStorage.setItem('uid', user.uid);
                            uid = user.uid

                            createView(app.views.length == 1)
                            ls.close()

                            app.toast.create({
                                icon: '<i class="icon f7-
icons" style="font-size:22px;">checkmark_alt_circle</i>',
                                text: `Selamat Datang ${user.name}`,
                                closeTimeout: 2000,
                            }).open()
                        })
                        .catch(err => {
                            app.dialog.preloader('Tidak dapat terhubu
ng ke server...');
                            console.log(err);
                        })
                })
```

```
        }
      }
    }
</script>

Provinsi.html
<template>
    <div class="page">
        <div class="navbar">
            <div class="navbar-bg"></div>
            <div class="navbar-inner">
                <div class="left">
                    <a class="link back">
                        <i class="icon f7-icons">arrow_left</i>
                    </a>
                </div>
                <div class="title">Lokasi</div>
                <div class="subnavbar">
                    <!-- Searchbar with auto init -->
                    <a href="/searchLokasi/" data-transition="f7-
fade" class="searchbar">
                        <div class="searchbar-inner">
                            <div class="searchbar-input-wrap">
                                <input type="search" placeholder="Car
i lokasi">
                                <i class="searchbar-icon"></i>
                                <span class="input-clear-
button"></span>
                            </div>
                            <span class="searchbar-disable-button if-
not-aurora">Cancel</span>
                        </div>
                    </a>
                </div>
            </div>
        </div>

        <div class="page-content">
            {{#if search}}
            <div class="list media-list">
                <ul>
                    {{#each dataSearch}}
                    <li>
                        <a href="/kabupaten/{{name}}/{{id}}/" dat
a-transition="f7-parallax"
                            class="item-link item-content">
                            <div class="item-inner">
                                <div class="item-title-row">
                                    <div class="item-
title">{{alt_name}}</div>
                                </div>
                            </div>
                        </a>
                    </li>
                    {{/each}}
                </ul>
            </div>
            {{else}}
            <!-- <div class="block">
                <a href="/lokasi/" class="button lokasi"><i class
="icon f7-icons">placemark_fill</i>Gunakan lokasi saat
                    ini</a>
            </div> -->
            <div class="block block-strong">
                <div class="block-header">Pilih Provinsi</div>
                <span class="progressbar-infinite"></span>
                <div id="vlProvinsi" class="list virtual-
list media-list"></div>
            </div>
            {{/if}}
        </div>

    </div>
</template>
<script>
    // script must return component object
    return {
        data: function () {
            return {
                search: false
            }
        },
        on: {
            pageInit: function () {
                this.getProvinsi()

            },
            pageBeforeRemove: function () {
                this.vlProvinsi.destroy();
            }
        },
        methods: {
            getProvinsi() {
                app.request.promise.json(`${URL}provinsi`)
                    .then(res => {
                        this.vlProvinsi = app.virtualList.create({
                            el: '#vlProvinsi',
                            items: res.data,
                            itemTemplate: '<li>' +
                                '<a href="/kabupaten/{{name}}/{{id}}
/" data-transition="f7-parallax" class="item-link item-
content">' +
                                '<div class="item-inner">' +
                                '<div class="item-title-row">' +
                                '<div class="item-
title">{{name}}</div>' +
                                '</div>' +
                                '</div>' +
                                '</a>' +
                                '</li>',
                        });

                        // app.progressbar.hide('.progressbar-
infinite');
                        $$('.progressbar-infinite').hide()
                    })
                    .catch(err => {
                        console.log(err);
                        app.progressbar.hide();
                        app.dialog.preloader('Tidak dapat terhubung
 ke server...');
                    })
            },
            aksiSearch: function () {
                const keyword = event.target.value

                this.$setState({
                    search: false,
                })

                if (keyword.length < 4) return

                app.request.promise.json(`${URL}searchLokasi`
, {
                    keyword
                }).then(res => {
                    if (res.data.error) return
```

```
                this.$setState({
                    search: true,
                    dataSearch: res.data
                })
            })
        }
    }
</script>
```

Kabupaten.html

```
<template>
    <div class="page">
        <div class="navbar">
            <div class="navbar-bg"></div>
            <div class="navbar-inner">
                <div class="left">
                    <a class="link back">
                        <i class="icon f7-icons">arrow_left</i>
                    </a>
                </div>
                <div class="title">{{name}}</div>
                <div class="subnavbar">
                    <!-- Searchbar with auto init -->
                    <a href="/searchLokasi/" data-transition="f7-
fade" class="searchbar">
                        <div class="searchbar-inner">
                            <div class="searchbar-input-wrap">
                                <input type="search" placeholder="Car
i lokasi">
                                <i class="searchbar-icon"></i>
                                <span class="input-clear-
button"></span>
                            </div>
                            <span class="searchbar-disable-button if-
not-aurora">Cancel</span>
                        </div>
                    </a>
                </div>
            </div>
        </div>

        <div class="page-content">
            <div class="block block-strong">
                <div class="block-
header">Pilih Kabupaten/Kota</div>
                <span class="progressbar-infinite"></span>
                <div id="vlKabupaten" class="list virtual-
list media-list"></div>
            </div>
        </div>

    </div>
</template>
<script>
    // script must return component object
    return {
        data: function () {
            return {
                loading: true
            }
        },
        on: {
            pageInit: function () {
                let self = this;
                this.$setState(this.$route.params);
                // app.progressbar.show('aqua');
                this.getKabupaten()
            },
            pageBeforeRemove: function () {
                this.vlKabupaten.destroy();
            },
        },
        methods: {
            getKabupaten() {
                app.request.promise.json(`${URL}kabupaten`, {
                        id_provinsi: this.id_provinsi
                    })
                    .then(res => {
                        this.vlKabupaten = app.virtualList.create({
                            el: '#vlKabupaten',
                            items: res.data,
                            itemTemplate: '<li>' +
                                '<a href="/kecamatan/{{name}}/{{id}
}/" data-transition="f7-parallax" class="item-link item-
content">' +
                                '<div class="item-inner">' +
                                '<div class="item-title-row">' +
                                '<div class="item-
title">{{name}}</div>' +
                                '</div>' +
                                '</div>' +
                                '</a>' +
                                '</li>',
                        });

                        // app.progressbar.hide()
                        $$('.progressbar-infinite').hide()
                    })
                    .catch(err => {
                        console.log(err);
                        app.progressbar.hide();
                        app.dialog.preloader('Tidak dapat terhubung
 ke server...');
                    })
            }
        }
    }
</script>
```

Kecamatan.html

```
<template>
    <div class="page">
        <div class="navbar">
            <div class="navbar-bg"></div>
            <div class="navbar-inner">
                <div class="left">
                    <a class="link back">
                        <i class="icon f7-icons">arrow_left</i>
                    </a>
                </div>
                <div class="title">{{name}}</div>
                <div class="subnavbar">
                    <!-- Searchbar with auto init -->
                    <a href="/searchLokasi/" data-transition="f7-
fade" class="searchbar">
                        <div class="searchbar-inner">
                            <div class="searchbar-input-wrap">
                                <input type="search" placeholder="Car
i lokasi">
                                <i class="searchbar-icon"></i>
                                <span class="input-clear-
button"></span>
                            </div>
                            <span class="searchbar-disable-button if-
not-aurora">Cancel</span>
                        </div>
                    </a>
```

```
            </div>
          </div>
        </div>

        <div class="page-content">
          <div class="block block-strong">
            <div class="block-
header">Pilih Kecamatan</div>
            <span class="progressbar-infinite"></span>
            <div id="vlKecamatan" class="list virtual-
list media-list"></div>
          </div>
        </div>

    </div>
</template>
<script>
    // script must return component object
    return {
        data: function () {
            return {
                loading: true
            }
        },
        on: {
            pageInit: function () {
                const self = this;
                this.$setState(this.$route.params);
                // app.progressbar.show('aqua');
                this.getKecamatan()

                $$('#vlKecamatan').on('click', 'li', function () {
                    const id_lokasi = $$(this).attr('data-id');

                    if (app.views.current.main) {
                        const data = {
                            uid,
                            id_lokasi
                        }

                        app.preloader.show();

                        app.request.promise.post(`${URL}lokasi`, d
ata)
                            .then(res => {
                                const response = JSON.parse(res.data)

                                app.preloader.hide();

                                if (response.success) {
                                    app.emit('getLokasi');
                                    app.emit('refreshProduk');

                                    app.views.current.router.back('/', {
                                        force: true
                                    });
                                } else {
                                    app.toast.create({
                                        icon: '<i class="icon f7-
icons" style="font-
size:22px;">exclamationmark_bubble</i>',
                                        text: response.message,
                                        position: 'center',
                                        closeTimeout: 3000,
                                    }).open()
                                }
                            }).catch(err => {
                                app.preloader.hide();
                                console.log(err);
                                app.dialog.alert('Tidak dapat terhubung
 ke server', 'Gagal');
                            })
                    } else {
                        const kec = $$(this).attr('data-name');
                        const kab = self.name.replace('Kabupaten', '
Kab.');
                        const title = `${kec}, ${kab}`;

                        app.emit('setLokasi', id_lokasi, title)

                        app.views.current.router.back(app.views.cur
rent.history[1], {
                            force: true
                        });
                    }

                })
            },
            pageBeforeRemove: function () {
                this.vlKecamatan.destroy();
            },
        },
        methods: {
            getKecamatan() {
                app.request.promise.json(`${URL}kecamatan`, {
                        id_kabupaten: this.id_kabupaten
                    })
                    .then(res => {
                        this.vlKecamatan = app.virtualList.create({
                            el: '#vlKecamatan',
                            items: res.data,
                            itemTemplate: '<li data-id="{{id}}" data-
name="{{name}}">' +
                                '<a href="#" class="item-link item-
content">' +
                                '<div class="item-inner">' +
                                '<div class="item-title-row">' +
                                '<div class="item-
title">{{name}}</div>' +
                                '</div>' +
                                '</div>' +
                                '</a>' +
                                '</li>',
                        });

                        // app.progressbar.hide()
                        $$('.progressbar-infinite').hide()
                    })
                    .catch(err => {
                        console.log(err);
                        app.progressbar.hide();
                        app.dialog.preloader('Tidak dapat terhubung
 ke server...');
                    })
            }
        }
    }
</script>

Search.html
<template>
    <div class="page">
        <div class="navbar">
            <div class="navbar-bg"></div>
            <div class="navbar-inner">
                <div class="left">
                    <a class="link back">
                        <i class="icon f7-icons">arrow_left</i>
```

```
            </a>
          </div>
          <div class="title">Lokasi</div>
          <div class="subnavbar">
            <!-- Searchbar with auto init -->
            <div class="searchbar">
              <div class="searchbar-inner">
                <div class="searchbar-input-wrap">
                  <input type="search" id="inputSearch"
@keyUp="aksiSearch" placeholder="Cari lokasi">
                  <i class="searchbar-icon"></i>
                  <span class="input-clear-
button"></span>
                </div>
                <span class="searchbar-disable-button if-
not-aurora">Cancel</span>
              </div>
            </div>
          </div>
        </div>

        <div class="page-content">
          <div class="block">
            {{#if loading}}
            <span class="progressbar-infinite"></span>
            {{else}}
            <div class="list media-list">
              <ul>
                {{#each dataSearch}}
                <li data-id="{{id}}" data-
name="{{alt_name}}" @click="submitLokasi">
                  <div class="item-link item-content">
                    <div class="item-inner">
                      <div class="item-title-row">
                        <div class="item-
title">{{name}}</div>
                      </div>
                    </div>
                  </div>
                </li>
                {{/each}}
              </ul>
            </div>
            {{#if empty}}
            <div class="block">
              <div style="display: flex;align-
items: center;justify-content: center;">
                <div class="item-content text-align-
center" style="max-width: 50%;">
                  <div class="item-media">
                    <i class="f7-icons text-color-
deeporange"
                      style="font-
size:80px">exclamationmark_bubble</i>
                  </div>
                  <div class="item-inner title-
kategori" style="font-size:18px;">Lokasi tidak ditemukan
                  </div>
                </div>
              </div>
            </div>
            {{/if}}
            {{/if}}
          </div>
        </div>

    </div>
</template>

<script>
    // script must return component object
    return {
        data: function () {
            return {
                loading: false,
                empty: false,
                dataSearch: []
            }
        },
        on: {
            pageInit: function () {
                $$('#inputSearch').focus()
            },
            pageBeforeRemove: function () {}
        },
        methods: {
            aksiSearch: function () {
                event.preventDefault()

                const self = this
                const keyword = event.target.value

                if (keyword.length < 3) {
                    self.$setState({
                        dataSearch: []
                    })
                    return
                }

                this.$setState({
                    loading: true,
                })
                app.request.promise.json(`${URL}searchLokasi`
, {
                    keyword
                }).then(res => {
                    if (res.data.error) return

                    const reg = new RegExp(keyword, "gi");

                    const dataSearch = res.data.map(v => {
                        v.name = v.alt_name.replace(reg, (str) => `<
b>${str}</b>`)
                        return v
                    })

                    self.$setState({
                        loading: false,
                        empty: res.data.length == 0,
                        dataSearch
                    })
                })
            },
            submitLokasi: function () {
                const target = $$(event.target).closest('li')
                const id_lokasi = target.data('id');

                if (app.views.current.main) {
                    const data = {
                        uid,
                        id_lokasi
                    }

                    app.preloader.show();

                    app.request.promise.post(`${URL}lokasi`, dat
a)
                        .then(res => {
```

```
                    const response = JSON.parse(res.data)

                    app.preloader.hide();

                    if (response.success) {
                        app.emit('getLokasi');
                        app.emit('refreshProduk');

                        app.views.current.router.back('/', {
                            force: true
                        });
                    } else {
                        app.toast.create({
                            icon: '<i class="icon f7-
icons" style="font-
size:22px;">exclamationmark_bubble</i>',
                            text: response.message,
                            position: 'center',
                            closeTimeout: 3000,
                        }).open()
                    }
                }).catch(err => {
                    app.preloader.hide();
                    console.log(err);
                    app.dialog.alert('Tidak dapat terhubung k
e server', 'Gagal');
                })
            } else {
                const title = target.data('name');

                app.emit('setLokasi', id_lokasi, title)

                app.views.current.router.back(app.views.curre
nt.history[1], {
                    force: true
                });
            }
        }
    }
</script>

searchProduk.html
<template>
    <div class="page" id="searchProdukPage">
        <div class="navbar">
            <div class="navbar-bg"></div>
            <div class="navbar-inner">
                <form class="searchbar" id="searchProduk">
                    <div class="searchbar-inner">
                        <div class="searchbar-input-wrap">
                            <input type="search" id="inputSearchPro
duk" value="{{keyword}}" placeholder="Cari Pakaian">
                            <i class="searchbar-icon"></i>
                            <span class="input-clear-
button"></span>
                        </div>
                        <span class="searchbar-disable-button if-
not-aurora">Cancel</span>
                    </div>
                </form>
            </div>
        </div>
        <div class="page-content hide-navbar-on-
scroll infinite-scroll-content">
            {{#if loading}}
            <div class="block">
                <span class="progressbar-infinite"></span>
            </div>
            {{else}}
            {{#js_if "this.totalIklan > 0"}}
            <div class="block block-strong">
                <div class="block-
header">Total Iklan {{totalIklan}}</div>
                <div class="list media-list">
                    <ul>
                        {{listProductHelper listProduct}}
                    </ul>
                </div>
            </div>

            <div class="preloader infinite-scroll-
preloader" id="preloaderSearch"></div>
            {{else}}
            <div class="block">
                <div style="display: flex;align-
items: center;justify-content: center;">
                    <div class="item-content text-align-
center" style="max-width: 50%;">
                        <div class="item-media">
                            <i class="f7-icons text-color-
deeporange" style="font-
size:80px">exclamationmark_bubble</i>
                        </div>
                        <div class="item-inner title-
kategori" style="font-size:18px;">{{errMsg}}
                        </div>
                    </div>
                </div>
            </div>
            {{/js_if}}
            {{/if}}
        </div>
    </div>
</template>
<script>
    // script must return component object
    return {
        data: function () {
            return {
                totalIklan: 0,
                listProduct: [],
                newest: 0,
                allowInfinite: true,
                loading: true,
                errMsg: 'Tidak ada iklan ditemukan'
            }
        },
        on: {
            pageInit: function () {
                const self = this;

                this.$setState(this.$route.params)

                this.searchbar = app.searchbar.create({
                    el: '#searchProduk',
                    searchContainer: '.list',
                    searchIn: '.item-title',
                    backdrop: false,
                    on: {
                        disable: this.backTo*Home*
                    }
                });

                this.searchbar.enable()

                this.getProduct().then(items => {
                    if (!items) return
```

```
                this.$setState({
                    listProduct: items,
                    allowInfinite: true
                }, () => {
                    app.lazy.create('#searchProdukPage')
                    app.infiniteScroll.create('.infinite-scroll-
content')
                })
            })

            $$('.infinite-scroll-content').on('infinite', () => {
                if (!this.allowInfinite) return;
                self.$setState({
                    allowInfinite: false
                })

                this.getProduct().then(items => {
                    if (items.length <= 0) return

                    const newData = [
                        ...this.listProduct,
                        ...items
                    ]
                    self.$setState({
                        listProduct: newData
                    }, () => app.lazy.create('#searchProdukPage'
))
                })
            })

            $$('#homeTab').on('touchstart', this.backToHome
)

            $$('#inputSearchProduk').on('focusin', this.backT
oSearch)
        },
        pageBeforeRemove: function () {
            this.searchbar.destroy()
            app.infiniteScroll.destroy('.infinite-scroll-
content')
            app.lazy.destroy('#searchProdukPage')
            $$('#preloaderSearch').remove();
            $$('#homeTab').off('touchstart', this.backToHome
)
        }
    },
    methods: {
        getProduct: function () {
            const limit = 20
            let newest = this.newest

            const data = {
                uid,
                limit,
                newest,
                keyword: this.keyword
            }

            return app.request.promise.json(`${URL}searchP
roduk`, data)
                .then(res => {
                    app.progressbar.hide();

                    if (res.data.error) {
                        this.$setState({
                            loading: false,
                            errMsg: res.data.message
                        })
                        return
                    }

                    newest += limit;
                    let allowInfinite = true;

                    if (newest >= res.data.total || res.data.total =
= 0) {
                        allowInfinite = false;
                        app.infiniteScroll.destroy('.infinite-scroll-
content')
                        // Remove preloader
                        $$('#preloaderSearch').remove();
                    }
                    this.$setState({
                        loading: false,
                        newest,
                        allowInfinite,
                        totalIklan: res.data.total
                    })

                    return res.data.produk
                })
                .catch(err => {
                    console.log(err);
                    app.progressbar.hide();
                    app.dialog.preloader('Tidak dapat terhubung
 ke server...');
                })
        },
        detail Iklan: function () {
            const {
                id,
                uid,
                hargaformat,
                ...query
            } = $$(event.target).closest('li').dataset()

            query['hargaFormat'] = hargaformat

            const url = app.views.current.router.generateUrl(
{
                name: 'detail Iklan',
                params: {
                    id,
                    uid
                },
                query
            });

            app.views.current.router.navigate(url)
        },
        backToSearch: function () {
            app.emit('setKeyword', this.keyword)

            app.views.current.router.back({
                transition: 'f7-fade',
            })
        },
        backToHome: function () {
            if (!app.views.current.main) return

            app.emit('refreshProduk')

            app.views.current.router.back('/', {
                force: true
            })
        }
    }
```

```
    }
</script>
```

RestApi.php

```
<?php
header('Access-Control-Allow-Origin: *');
header('Access-Control-Allow-
Methods: GET, POST, DELETE, PUT');
header('Access-Control-Allow-Headers: Origin, X-
Requested-With, Content-Type, Accept, Authorization');
header('Access-Control-Allow-Credentials: true');
header('Content-Type: application/json');

defined('BASEPATH') or exit('No direct script access allo
wed');

use chriskacerguis\RestServer\RestController;

class RestApi extends RestController
{
    function __construct()
    {
        parent::__construct();
        $this->load->model('Model');
        $this->db->query("SET sql_mode = " ");
    }

    public function provinsi_get()
    {
        $id = $this->get('id');
        $tbl = 'tbl_provinsi';

        $this->db->select('id, name');

        if ($id !== null) {
            $this->db->where('id', $id);
        }

        $this->db->order_by('name', 'ASC');
        $data = $this->db->get($tbl)->result();

        $this->response($data, 200);
    }
    // ------------------------------------------------------------------
------------------------------------------------------
    public function kabupaten_get()
    {
        $id = $this->get('id');
        $id_provinsi = $this->get('id_provinsi');

        $this->db->select('id, name');

        if ($id !== null) {
            $this->db->where('id', $id);
        }

        if ($id_provinsi !== null) {
            $this->db->where('province_id', $id_provinsi);
        }

        $this->db->order_by('name', 'ASC');
        $data = $this->db->get('tbl_kabupaten')->result();

        $this->response($data, 200);
    }
    // ------------------------------------------------------------------
------------------------------------------------------
    public function kecamatan_get()
    {
        $id = $this->get('id');
        $id_kabupaten = $this->get('id_kabupaten');

        $this->db->select('id, regency_id, name');

        if ($id !== null) {
            $this->db->where('id', $id);
        }

        if ($id_kabupaten !== null) {
            $this->db->where('regency_id', $id_kabupaten);
        }

        $this->db->order_by('name', 'ASC');
        $data = $this->db->get('tbl_kecamatan')->result();

        $this->response($data, 200);
    }
    // ------------------------------------------------------------------
------------------------------------------------------
    public function produk_get()
    {
        $uid = $this->get('uid');
        $id = $this->get('id');
        $get = $this->get('get');

        if ($uid === null || $id === null) {
            $this-
>response(['message' => 'Akses ditolak bosku'], 502);
        }

        $select = '';
        if ($get == 'All') {
            $select = ', a.kategori, a.images, a.dilihat, a.ditamb
ahkan_pada as tanggal';
        }

        $data = $this->Model-
>query("SELECT a.judul, a.deskripsi, a.kondisi, a.harga, C
ONCAT('Rp ', REPLACE(FORMAT(a.harga, 0), ',', '.')) as
hargaFormat, a.id_kecamatan, CONCAT(b.name, ', ', c.na
me) as lokasi $select FROM tbl_produk a LEFT JOIN tbl_
kecamatan b ON a.id_kecamatan = b.id LEFT JOIN tbl_ka
bupaten c ON b.regency_id = c.id WHERE a.id = '$id' AN
D a.uid = '$uid'")[0];

        if ($get == 'All') {
            $*user* = $this->Model-
>query("SELECT nama, foto FROM tbl_*user* WHERE uid
 = '$uid'")[0];
            $data['nama'] = $*user*['nama'];
            $data['foto'] = $*user*['foto'];
            $data['deskripsi'] = nl2br(htmlspecialchars($data['d
eskripsi']));
        }

        $this->response($data, 200);
    }

    public function produk_post()
    {
        $images = $this->post('images');

        $data = array(
            'uid' => $this->post('uid'),
            'judul' => $this->post('judul'),
            'deskripsi' => $this->post('deskripsi'),
            'kondisi' => $this->post('kondisi'),
            'harga' => $this->post('harga'),
```

```
            'kategori' => $this->post('kategori'),
            'id_kecamatan' => $this->post('id_kecamatan'),
        );

        $data['id'] = $this->Model->getRandom('18');

        // ----------------Convert Base64 to Images----------------
        $filePath = APPPATH . "/../assets/images/produk/$data[id]/";

        if (!file_exists($filePath)) {
            mkdir($filePath, 0777, true);
        }

        foreach ($images as $base64String) {
            $fileName = uniqid() . '.jpeg';
            $data['images'][] = base_url("assets/images/produk/$data[id]/") . $fileName;
            file_put_contents($filePath . $fileName, base64_decode($base64String));
        }
        $data['images'] = json_encode($data['images']);
        // ------------------------------------------------------------
        $data['harga'] = str_replace('.', '', $data['harga']);
        $data['ditambahkan_pada'] = $this->Model->getWaktu();
        $data['diubah_pada'] = $this->Model->getWaktu();

        $simpan = $this->Model->setTambah('tbl_produk', $data);

        if ($simpan) {
            $this->response(['success' => true, 'message' => 'Iklan anda telah berhasil dikirim'], 200);
        } else {
            $this->response(['success' => false, 'message' => 'Iklan gagal dikirim'], 200);
        }
    }

    public function produkUpdate_post()
    {
        $data = array(
            'kondisi' => $this->post('kondisi'),
            'judul' => $this->post('judul'),
            'deskripsi' => $this->post('deskripsi'),
            'harga' => $this->post('harga'),
            'id_kecamatan' => $this->post('id_kecamatan'),
        );

        $where = $this->post('where');

        $data['harga'] = str_replace('.', '', $data['harga']);
        $data['diubah_pada'] = $this->Model->getWaktu();

        $update = $this->Model->setUpdate('tbl_produk', $data, $where);

        if ($update) {
            $this->response(['success' => true, 'message' => 'Iklan anda berhasil diubah'], 200);
        } else {
            $this->response(['success' => false, 'message' => 'Iklan anda gagal diubah'], 200);
        }
    }
    // ------------------------------------------------------------
    public function searchProduk_get()
    {
        $uid = $this->get('uid');
        $location = $this->get('location');
        $keyword = $this->get('keyword');
        $limit = $this->get('limit');
        $newest = $this->get('newest');

        if ($uid === null || $limit === null || $newest === null) {
            $this->response(['message' => 'Akses ditolak bosku'], 502);
        }

        if (!$location) {
            $data_user = $this->Model->query("SELECT latitude, longitude FROM tbl_user WHERE uid = '$uid'")[0];
            $latitude = $data_user['latitude'];
            $longitude = $data_user['longitude'];
        } else {
            $data_kec = $this->Model->query("SELECT latitude, longitude FROM tbl_kecamatan WHERE id = '$location'")[0];
            $latitude = $data_kec['latitude'];
            $longitude = $data_kec['longitude'];
        }

        $latitude = $latitude ? $latitude : -4.03333;
        $longitude = $longitude ? $longitude : 119.65;

        if ($keyword) {
            $search = explode(' ', strtolower($keyword));

            $exclude = ['pakaian']; //Abaikan keyword

            $search = array_filter($search, function ($v) use ($exclude) {
                return !in_array($v, $exclude);
            });

            if (empty($search)) {
                $this->response(['error' => true, 'message' => 'Keyword tidak spesifik'], 200);
            }

            $field = ['a.judul', 'a.deskripsi', 'a.kondisi'];

            foreach ($field as $col) {
                foreach ($search as  $que) {
                    $text[] = "$col LIKE '%$que%'";
                }
            }

            // Filter Berdasarkan Kategori

            $kategori = [
                'wanita' => 'Wanita',
                'perempuan' => 'Wanita',
                'cewek' => 'Wanita',
                'pria' => 'Pria',
                'laki' => 'Pria',
                'laki-laki' => 'Pria',
                'anak' => 'Anak',
```

```
            'anak-anak' => 'Anak'
        ];

        $filterKategori = array_map(function ($v) use ($kategori) {
            if (!array_key_exists($v, $kategori)) return;

            return "a.kategori LIKE '%$kategori[$v]%'";
        }, $search);

        $filterKategori = array_filter($filterKategori); // menghilangkan nilai null

        $keyword = "AND (" . implode(' OR ', $text) .
            (empty(!$filterKategori) ? " OR " . implode(' OR ', $filterKategori) . ")" : ")");
    }

    $produk = $this->Model->query("SELECT a.id, a.uid, a.judul, a.harga, JSON_UNQUOTE(JSON_EXTRACT(a.images, '$[0]')) as image, FN_GET_DISTANCE(b.latitude,b.longitude,$latitude,$longitude) as jarak, CONCAT(b.name,', ',c.name) as lokasi FROM tbl_produk a LEFT JOIN tbl_kecamatan b ON a.id_kecamatan = b.id LEFT JOIN tbl_kabupaten c ON b.regency_id = c.id WHERE FN_GET_DISTANCE(b.latitude,b.longitude,$latitude,$longitude) < 50 $keyword ORDER BY jarak, a.ditambahkan_pada DESC LIMIT $limit OFFSET $newest");

    $produk = array_map(function ($v) {
        $v['judul'] = htmlspecialchars($v['judul']);
        $v['hargaFormat'] = $this->Model->rupiah($v['harga']);
        return $v;
    }, $produk);

    $total = $this->Model->query("SELECT COUNT(a.id) as total FROM tbl_produk a LEFT JOIN tbl_kecamatan b ON a.id_kecamatan = b.id WHERE FN_GET_DISTANCE(b.latitude,b.longitude,$latitude,$longitude) < 50 $keyword")[0]['total'];

    $data['total'] = $total;
    $data['produk'] = $produk;

    $this->response($data, 200);
  }
  // ------------------------------------------------------------------------
  public function iklan_get()
  {
    $uid = $this->get('uid');

    if ($uid === null) {
        $this->response(['message' => 'Akses ditolak bosku'], 502);
    }

    $data = $this->Model->query("SELECT id, judul, harga, CONCAT('Rp ',REPLACE(FORMAT(harga,0), ',', '.')) as hargaFormat, dilihat, JSON_UNQUOTE(JSON_EXTRACT(images, '$[0]')) as image, ditambahkan_pada FROM tbl_produk WHERE uid = '$uid' ORDER BY ditambahkan_pada DESC");

    $this->response($data, 200);
  }

  public function iklan_post()
  {
    $uid = $this->post('uid');
    $id = $this->post('id');

    if ($uid === null || $id === null) {
        $this->response(['message' => 'Akses ditolak bosku'], 502);
    }

    $data = $this->db->query("UPDATE tbl_produk SET dilihat = dilihat + 1 WHERE uid = '$uid' AND id = '$id'");

    $this->response($data, 200);
  }

  public function iklanDelete_post()
  {
    $id = $this->post('id');

    if ($id === null) {
        $this->response(['message' => 'Akses ditolak bosku'], 404);
    }

    $filePath = APPPATH . "/../assets/images/produk/$id/";
    $this->Model->delete_files($filePath);

    $hapus = $this->Model->setHapus('tbl_produk', ['id' => $id]);

    if ($hapus) {
        $this->response(['success' => true, 'message' => 'Iklan berhasil dihapus'], 200);
    } else {
        $this->response(['success' => false, 'message' => 'Iklan gagal dihapus'], 200);
    }
  }
  // ------------------------------------------------------------------------
  public function lokasi_get()
  {
    $uid = $this->get('uid');

    if ($uid === null) {
        $this->response(['message' => 'Akses ditolak bosku'], 502);
    }

    $dataLokasi = $this->Model->query("SELECT latitude, longitude FROM tbl_*user* WHERE uid = '$uid'")[0];

    ['latitude' => $latitude, 'longitude' => $longitude] = $dataLokasi;

    if ($latitude === null || $longitude === null) {
        $this->response(['name' => 'Pilih Lokasi', 'message' => 'Koordinat belum ditentukan'], 200);
    }

    $data = $this->Model->query("SELECT CONCAT(a.name, ', ', REPLACE(b.na
```

```
me, 'Kabupaten', 'Kab.')) as name, FN_GET_DISTANCE(a
.latitude,a.longitude,$latitude,$longitude) as jarak FROM t
bl_kecamatan a LEFT JOIN tbl_kabupaten b ON a.regency
_id = b.id ORDER BY jarak LIMIT 1")[0];

        $this->response($data, 200);

        // $dataLokasi = json_decode($this->Model-
>query("SELECT lokasi FROM tbl_user WHERE uid = '$
uid'")[0]['lokasi']);

        // $key = key($dataLokasi);
        // $id = $dataLokasi->$key;

        // switch (key($dataLokasi)) {
        //     case 'id_provinsi':
        //         $query = "SELECT name FROM tbl_provinsi
WHERE id = '$id'";
        //         break;
        //     case 'id_kabupaten':
        //         $query = "SELECT name FROM tbl_kabupate
n WHERE id = '$id'";
        //         break;
        //     case 'id_kecamatan':
        //         $query = "SELECT CONCAT(a.name, ', ', RE
PLACE(b.name, 'Kabupaten', 'Kab.')) as name FROM tbl_
kecamatan a LEFT JOIN tbl_kabupaten b ON a.regency_id
 = b.id WHERE a.id = '$id'";
        //         break;
        // }

        // $data = $this->Model->query($query)[0];

        // $this->response($data, 200);
    }

    public function lokasi_post()
    {
        $uid = $this->input->post('uid');
        $id_lokasi = $this->input->post('id_lokasi');

        // switch (strlen($id_lokasi)) {
        //     case 2:
        //         $lokasi = json_encode(['id_provinsi' => $id_lo
kasi]);
        //         $tbl = 'tbl_provinsi';
        //         break;
        //     case 4:
        //         $lokasi = json_encode(['id_kabupaten' => $id_l
okasi]);
        //         $tbl = 'tbl_kabupaten';
        //         break;
        //     case 7:
        //         $lokasi = json_encode(['id_kecamatan' => $id_
lokasi]);
        //         $tbl = 'tbl_kecamatan';
        //         break;
        // }
        $data_latlong = $this->Model-
>query("SELECT latitude, longitude FROM tbl_kecamata
n WHERE id ='$id_lokasi'")[0];
        $latitude = $data_latlong['latitude'];
        $longitude = $data_latlong['longitude'];

        $data = ['latitude' => $latitude, 'longitude' => $longitu
de];

        $simpan = $this->Model-
>setUpdate('tbl_user', $data, ['uid' => $uid]);

        if ($simpan) {
            $this-
>response(['success' => true, 'message' => 'Lokasi berhasil
diubah'], 200);
        } else {
            $this-
>response(['success' => false, 'message' => 'Lokasi gagal di
ubah'], 200);
        }
    }

    public function searchLokasi_get()
    {
        $keyword = $this->get('keyword');

        if ($keyword === null) {
            $this-
>response(['message' => 'Keyword tidak ditemukan'], 502);
        }

        $data = $this->Model-
>query("SELECT id, alt_name FROM tbl_kecamatan WH
ERE name LIKE '$keyword%' OR alt_name LIKE '%$key
word%' ORDER BY name LIMIT 20");

        $this->response($data, 200);
    }
    // ----------------------------------------------------------------
-------------------------------------------------------
    public function chat_get()
    {
        $uid = $this->get('uid');
        $id_produk = $this->get('id_produk');
        $id_pembeli = $this->get('pembeli');
        $id_penjual = $this->get('penjual');

        if ($uid === null || $id_produk === null || $id_pembeli
 === null || $id_penjual === null) {
            $this-
>response(['message' => 'Akses ditolak bosku'], 502);
        }

        $getProduk = $this->Model-
>query("SELECT judul, CONCAT('Rp ', REPLACE(FOR
MAT(harga, 0), ',', '.')) as hargaFormat, JSON_UNQUOTE
(JSON_EXTRACT(images, '$[0]')) as image FROM tbl_pr
oduk WHERE id = '$id_produk'");

        if (empty($getProduk)) {
            $this-
>response(['error' => true, 'message' => 'Produk tidak dite
mukan'], 204);
        }

        $data_produk = $getProduk[0];

        $data_produk['judul'] = htmlspecialchars($data_produ
k['judul']);
        $response['dataProduk'] = $data_produk;

        $uid = $uid != $id_pembeli ? $id_pembeli : $id_penju
al;
        $data*User* = $this->Model-
>query("SELECT nama, foto FROM tbl_user WHERE uid
 = '$uid'")[0];
        $response['data*User*'] = $data*User*;

        $this->response($response, 200);
```

```
    }
    // ------------------------------------------------------------------
-------------------------------------------------------
    public function user_get()
    {
        $uid = $this->get('uid');

        if ($uid === null) {
            $this-
>response(['message' => 'Data tidak spesifik'], 502);
        }

        $data = $this->Model-
>query("SELECT * FROM tbl_user WHERE uid = '$uid'")
[0];

        $this->response($data, 200);
    }
    public function user_post()
    {
        // $data = array(
        //     'uid' => $this->post('uid'),
        //     'nama' => $this->post('nama'),
        //     'email' => $this->post('email'),
        //     'telp' => $this->post('telp'),
        //     'foto' => $this->post('foto'),
        //     'token' => $this->post('token'),
        //     'latitude' => $this->post('latitude'),
        //     'longitude' => $this->post('longitude')
        // );

        foreach ($_POST as $key => $val) {
            $data[$key] = $val;
        }

        $cek = $this->Model-
>query("SELECT COUNT(uid) as total FROM tbl_user W
HERE uid = '$data[uid]'")[0]['total'];

        if ($cek > 0) {
            $query = $this->Model-
>setUpdate('tbl_user', $data, ['uid' => $data['uid']]);
            $data = $this->Model-
>query("SELECT * FROM tbl_user WHERE uid = '$data[
uid]'")[0];
        } else {
            $query = $this->Model-
>setTambah('tbl_user', $data);
        }

        if ($query) {
            $this-
>response(['success' => true, 'data' => $data], 200);
        } else {
            $this->response(['success' => false], 200);
        }
    }
    // ------------------------------------------------------------------
-------------------------------------------------------
    public function notifikasi_post()
    {
        $uid = $this->post('uid');
        $to = $this->post('to');
        $text = $this->post('text');

        // -- Data Sender ---
        $dataSender = $this->Model-
>query("SELECT nama FROM tbl_user WHERE uid = '$u
id'")[0];

        // -- Data Receiver ---
        $dataReceiver = $this->Model-
>query("SELECT token FROM tbl_user WHERE uid = '$t
o'")[0];

        $data = ['nama' => $dataSender['nama'], 'pesan' => $te
xt];

        $this->Model-
>sendPushNotification(['title' => 'Mabbalanca', 'text' => 'A
nda memiliki pesan baru', 'data' => $data, 'token' => $dataR
eceiver['token']]);
    }
    // ------------------------------------------------------------------
-------------------------------------------------------
}
```