**BAB V**

**KESIMPULAN DAN SARAN**

1. **Kajian Pustaka**

Berdasarkan hasil penelitian yang telah dilakukan diperoleh kesimpulan sebagai berikut.

1. Berdasarkan analisis sistem berjalan di BPS ditemukan bahwa Sistem Informasi Geografis yang ada saat ini tidak memiliki fasilitas untuk mengakses data dengan basis kewilayahan dan dalam pembuatannya masih membutuhkan waktu yang lama.
2. Berdasarkan hasil wawancara dengan subjectmatter dibutuhkan sebuah aplikasi SIG berbasis web yang dapat digenerate dengan mudah dan cepat serta dapat memfasilitasi petugas untuk menginput berbagai jenis tipe data dan variabel.
3. Peneliti telah berhasil merancang dan membangun aplikasi SIG berbasis web yang dapat di-*generate* dengan cepat dan mudah, serta *compatible* dengan berbagai jenis data dan variabel.
4. Data suatu wilayah yang ditampilkan menggunakan sistem informasi geografis akan sangat membantu dalam hal pengambilan keputusan atau kebijakan terkait wilayah tersebut
5. Visualisasi data yang dihasilkan layanan ini berupa grafik dan diagram. Melalui grafik dapat diperoleh informasi nilai maksimum dan minimum serta perbedaan dan persamaan tingkat suatu variabel tertentu. Sedangkan melalui diagram dapat diperoleh informasi jumlah persentase atau proporsi dari suatu variabel terhadap nilai totalnya.
6. **Saran**

Adapun saran untuk penelitian ataupun pengembangan layanan visualisasi ini lebih lanjut adalah sebagai berikut.

1. Menambah analisis sederhana terhadap visualisasi peta tematik maupun grafik yang ditampilkan.
2. Menambahkan fitur untuk membandingkan variabel-variabel antara satu wilayah dengan wilayah lainnya.
3. Menambahkan fitur *download* pada peta dalam format bentuk *file jpg* atau *pdf*.
4. Menambahkan fitur *share* ke media sosial atau pun blog pada halaman peta dan visualisasi grafik yang diinginkan.
5. Layanan ini dapat dikembangkan menjadi sebuah layanan visualisasi sistem informasi geografis yang lebih besar dengan membuat modul-modul khusus dari data yang dihasilkan dari kegiatan BPS baik berupa sensus maupun survei.

**DAFTAR PUSTAKA**

Aizatin, Anisa. (2013). *Enhanced GIS : Pengembangan Visualisasi Data ke dalam Peta Tematik* [Skripsi]. Jakarta: Sekolah Tinggi Ilmu Statistik.

Aronoff, Stan. 1989. Geographic Information System a Management Perspective.WDL Publication, Ottawa-Canad

Badiyanto. (2013). Buku Pintar Framework Yii. Yogyakarta : Mediakom.

Bangor, Aaron, Philip Kortum, dan James Miller. (2009). *Determining What Individual Scores Mean : Adding in Adjective Rating Scale, Journal of Usability Studies,* Volume 4, Issue 3, 114-123.

Barus, B. dan Wiradisastra U.S. (2000). Sistem Informasi Geografi; Sarana Manajemen Sumberdaya . Bogor: Laboratorium Pengindraan Jauh dan Kartografi Jurusan Tanah Fakultas Pertanian IPB.

Burrough, P.1986. Principle of Geographical Information System for Land Resources Assesment. Claredon Press : Oxford.

Chaffer, Jonathan., Karl Swedberg. (2011). *Learning JQuery.* Birmingham : Packt Publishing Ltd.

Connolly, Thomas., Carolyn Begg. (2002). *Database systems: A practical approach to design implementation, and management.* England: Addison-Wesley.

Eko Budiyanto. (2002). *Sistem Informasi Geografis Menggunakan ArcView GIS.* Yogyakarta : Andi.

ESRI, ArcView GIS. : The Geographic Information System for Everyone, 2002, http://www.esri.com/software/arcview, Environmental System Research Institute, Inc.

Gosselin, Don., Diana Kokoska & Robert Easterbrooks. (2010). *PHP Programming with MySql : The Web Technologies Series.* US : Cengage Learning.

Iacovella, Stefano. (2014). *Geoserver cookbook.* Birmingham : Packt Publishing Ltd.

Irwansyah, Edi. (2013). *Sistem Informasi Geografis : Prinsip dasar dan pengembangan aplikasi.* Yogyakarta : Digibooks.

Paul Longley et.al. 2001.Geographic Information System and Science. John Wiley and Sons : New York.

Peraturan Presiden Nomor 86 Tahun 2007 tentang Badan Pusat Statistik.

Permata, Disna Haristya. (2012). *Sistem Aplikasi Visualisi Data ke dalam Peta Tematik Berbasis WebGIS* [Skripsi]. Jakarta: Sekolah Tinggi Ilmu Statistik.

Prahasta, Eddy. (2009), Sistem Informasi Geografis Konsep-konsep Dasar (Perspektif Geodesi dan Geomatika),Informatika, Bandung.

Rencana Strategis Badan Pusat Statistik bab II Visi, Misi, dan Tujuan BPS Tahun 2015.

Sekolah Tinggi ilmu Statistik. (2010). *Pedoman Penyusunan Skripsi Jurusan Komputasi Statistik Sekolah Tinggi Ilmu Statistik edisi Keempat*. Jakarta: Sekolah Tinggi Ilmu Statistik.

Shneiderman B. (1998). Designing The User Interface: Strategic For Effective HCI. 3 rd Edition. Addison Wesley.

Sigit, Alysius. (2011). *Website super canggih dengan plugin jQuery terbaik.* Jakarta : Mediakita.

Subdirektorat Layanan dan Promosi Statistik. (2014). *Prosedur Operasional Standar Peremajaan Tampilan Website Bps 2014*. Jakarta: Badan Pusat Statistik Republik Indonesia.

UU No.14 Tahun 2008 Keterbukan Informasi Publik .

Valade, Janet. (2004). *PHP 5 For Dummies.* Hoboken : John Wiley & Sons.

Whitten, Jeffery L., Bently, Lonnie D., Ditmann, Kevin C. 2004, Systems Analysis and Design Methods, The McGraw-Hill Companies, Inc. International Edition.

## LAMPIRAN

Lampiran 1. Script side controller.php

<?php

**namespace** frontend\controllers;

**use** Yii;

**use** common\models\LoginForm;

**use** frontend\models\PasswordResetRequestForm;

**use** frontend\models\ResetPasswordForm;

**use** frontend\models\SignupForm;

**use** frontend\models\ContactForm;

**use** yii\base\InvalidParamException;

**use** yii\web\BadRequestHttpException;

**use** yii\web\Controller;

**use** yii\filters\VerbFilter;

**use** yii\filters\AccessControl;

**use** yii\db\Query;

**use** yii\helpers\Json;

**use** yii\web\Response;

/\*\*

\* Site controller

\*/

**class** SiteController **extends** Controller

{

/\*\*

\* @inheritdoc

\*/

**public** **function** behaviors()

{

**return** [

'access' => [

'class' => AccessControl::className(),

'only' => ['logout', 'signup'],

'rules' => [

[

'actions' => ['signup'],

'allow' => **true**,

'roles' => ['?'],

],

[

'actions' => ['logout'],

'allow' => **true**,

'roles' => ['@'],

],

],

],

'verbs' => [

'class' => VerbFilter::className(),

'actions' => [

'logout' => ['post'],

],

],

];

}

/\*\*

\* @inheritdoc

\*/

**public** **function** actions()

{

**return** [

'error' => [

'class' => 'yii\web\ErrorAction',

],

'captcha' => [

'class' => 'yii\captcha\CaptchaAction',

'fixedVerifyCode' => YII\_ENV\_TEST ? 'testme' : **null**,

],

];

}

**public** **function** actionIndex()

{

**return** $this->render('index');

}

**public** **function** actionLogin()

{

**if** (!\Yii::$app->user->isGuest) {

**return** $this->goHome();

}

$model = **new** LoginForm();

**if** ($model->load(Yii::$app->request->post()) && $model->login()) {

**return** $this->goBack();

} **else** {

**return** $this->render('login', [

'model' => $model,

]);

}

}

**public** **function** actionLogout()

{

Yii::$app->user->logout();

**return** $this->goHome();

}

**public** **function** actionContact()

{

$model = **new** ContactForm();

**if** ($model->load(Yii::$app->request->post()) && $model->validate()) {

**if** ($model->sendEmail(Yii::$app->params['adminEmail'])) {

Yii::$app->session->setFlash('success', 'Thank you for contacting us. We will respond to you as soon as possible.');

} **else** {

Yii::$app->session->setFlash('error', 'There was an error sending email.');

}

**return** $this->refresh();

} **else** {

**return** $this->render('contact', [

'model' => $model,

]);

}

}

**public** **function** actionAbout()

{

**return** $this->render('about');

}

**public** **function** actionSignup()

{

$model = **new** SignupForm();

**if** ($model->load(Yii::$app->request->post())) {

**if** ($user = $model->signup()) {

**if** (Yii::$app->getUser()->login($user)) {

**return** $this->goHome();

}

}

}

**return** $this->render('signup', [

'model' => $model,

]);

}

**public** **function** actionRequestPasswordReset()

{

$model = **new** PasswordResetRequestForm();

**if** ($model->load(Yii::$app->request->post()) && $model->validate()) {

**if** ($model->sendEmail()) {

Yii::$app->getSession()->setFlash('success', 'Check your email for further instructions.');

**return** $this->goHome();

} **else** {

Yii::$app->getSession()->setFlash('error', 'Sorry, we are unable to reset password for email provided.');

}

}

**return** $this->render('requestPasswordResetToken', [

'model' => $model,

]);

}

**public** **function** actionResetPassword($token)

{

**try** {

$model = **new** ResetPasswordForm($token);

} **catch** (InvalidParamException $e) {

**throw** **new** BadRequestHttpException($e->getMessage());

}

**if** ($model->load(Yii::$app->request->post()) && $model->validate() && $model->resetPassword()) {

Yii::$app->getSession()->setFlash('success', 'New password was saved.');

**return** $this->goHome();

}

**return** $this->render('resetPassword', [

'model' => $model,

]);

}

**public** **function** actionData($wil,$var,$kat){

\Yii::$app->response->format = \yii\web\Response::FORMAT\_JSON;

$namaWilparent = \common\models\Wilayah::findOne(['id' => $wil,]);

$query = **new** Query;

$query->select('fakta.id\_wilayah, wilayah.nama as nama\_wilayah, fakta.nilai, fakta.tahun, variabel.satuan, variabel.nama as nama\_variabel, kategori.nama as nama\_kategori, fakta.id\_bulan, bulan.nama as nama\_bulan')

->from('fakta')

->distinct('wilayah.id ,fakta.nilai,fakta.tahun,variabel.satuan,fakta.id\_bulan,bulan.nama')

->join('INNER JOIN', 'wilayah','fakta.id\_wilayah=wilayah.id')

->join('INNER JOIN', 'variabel','fakta.id\_variabel=variabel.id')

->join('INNER JOIN', 'bulan','fakta.id\_bulan=bulan.id')

->join('INNER JOIN', 'kategori','fakta.id\_kategori=kategori.id')

->orderBy('fakta.tahun,fakta.id\_bulan')

->where('wilayah.id\_parent='.$wil.' AND fakta.id\_variabel='.$var.' AND fakta.id\_kategori='.$kat);

$rows = $query->all();

$command = $query->createCommand();

$rows = $command->queryAll();

**return** (['data'=>$rows,'namaparent'=>$namaWilparent['nama']]);

}

**public** **function** actionCekdata($wil,$var,$kat){

\Yii::$app->response->format = \yii\web\Response::FORMAT\_JSON;

$query = **new** Query;

$query->select('fakta.id\_wilayah, wilayah.nama as nama\_wilayah, fakta.nilai, fakta.tahun, variabel.satuan, variabel.nama as nama\_variabel, kategori.nama as nama\_kategori, fakta.id\_bulan, bulan.nama as nama\_bulan')

->from('fakta')

->distinct('wilayah.id ,fakta.nilai,fakta.tahun,variabel.satuan,fakta.id\_bulan,bulan.nama')

->join('INNER JOIN', 'wilayah','fakta.id\_wilayah=wilayah.id')

->join('INNER JOIN', 'variabel','fakta.id\_variabel=variabel.id')

->join('INNER JOIN', 'bulan','fakta.id\_bulan=bulan.id')

->join('INNER JOIN', 'kategori','fakta.id\_kategori=kategori.id')

->orderBy('fakta.tahun,fakta.id\_bulan')

->where('wilayah.id\_parent='.$wil.' AND fakta.id\_variabel='.$var.' AND fakta.id\_kategori='.$kat);

$rows = $query->all();

$command = $query->createCommand();

$rows = $command->queryAll();

**if**($rows!=**null**) **return**(['data'=>1,]);

**else** **return**(['data'=>0,]);

}

**public** **function** actionChildTopik() {

\Yii::$app->response->format = \yii\web\Response::FORMAT\_JSON;

$out = [];

**if** (**isset**($\_POST['depdrop\_parents'])&&end($\_POST['depdrop\_parents'])!=**null**) {

$id = end($\_POST['depdrop\_parents']);

$query = **new** Query;

$query->select('topik.id, topik.nama')

->from('fakta')

->distinct('topik.id')

->join('INNER JOIN', 'wilayah','fakta.id\_wilayah=wilayah.id')

->join('INNER JOIN', 'variabel','fakta.id\_variabel=variabel.id')

->join('INNER JOIN', 'topik','variabel.id\_topik=topik.id')

->orderBy('topik.nama')

->where('wilayah.id\_parent='.$id);

$list = $query->all();

$selected = **null**;

**if** ($id != **null** && **count**($list) > 0) {

$selected = '';

**foreach** ($list **as** $i => $topik) {

//$topik[$i]=$variabel['namaTopik'];

//$namaVariabel=\frontend\models\Variabel::find()->andWhere(['kode'=>$account['kode\_variabel']])->asArray()->all();

$out[] = ['id' => $id.$topik['id'], 'name' => $topik['nama']];

**if** ($i == 0) {

$selected = $id.$topik['id'];

}

}

// Shows how you can preselect a value

**return** (['output' => $out, 'selected'=>$selected]);

}

}

**return** (['output' => '', 'selected'=>'']);

}

**public** **function** actionChildVariabel() {

\Yii::$app->response->format = \yii\web\Response::FORMAT\_JSON;

$out = [];

**if** (**isset**($\_POST['depdrop\_parents'])&&end($\_POST['depdrop\_parents'])!=**null**) {

$id = end($\_POST['depdrop\_parents']);

$l=strlen($id)-10;

$idTop=substr($id,10,$l);

$idWil=substr($id,0,10);

$query = **new** Query;

$query->select('variabel.id, variabel.nama')

->from('fakta')

->distinct('variabel.id')

->join('INNER JOIN', 'wilayah','fakta.id\_wilayah=wilayah.id')

->join('INNER JOIN', 'variabel','fakta.id\_variabel=variabel.id')

->join('INNER JOIN', 'topik','variabel.id\_topik=topik.id')

->orderBy('variabel.nama')

->where('topik.id='.$idTop.' AND wilayah.id\_parent='.$idWil);

$list = $query->all();

$selected = **null**;

**if** ($id != **null** && **count**($list) > 0) {

$selected = '';

**foreach** ($list **as** $i => $variabel) {

//$topik[$i]=$variabel['namaTopik'];

//$namaVariabel=\frontend\models\Variabel::find()->andWhere(['kode'=>$account['kode\_variabel']])->asArray()->all();

$out[] = ['id' => $idWil.$variabel['id'], 'name' => $variabel['nama']];

**if** ($i == 0) {

$selected = $idWil.$variabel['id'];

}

}

// Shows how you can preselect a value

**return** (['output' => $out, 'selected'=>$selected]);

}

}

**return** (['output' => '', 'selected'=>'']);

}

**public** **function** actionChildKategori() {

\Yii::$app->response->format = \yii\web\Response::FORMAT\_JSON;

$out = [];

**if** (**isset**($\_POST['depdrop\_parents'])&&end($\_POST['depdrop\_parents'])!=**null**) {

$id = end($\_POST['depdrop\_parents']);

$l=strlen($id)-10;

$idVar=substr($id,10,$l);

$idWil=substr($id,0,10);

$query = **new** Query;

$query->select('kategori.id, kategori.nama')

->from('fakta')

->distinct('kategori.id')

->join('INNER JOIN', 'wilayah','fakta.id\_wilayah=wilayah.id')

->join('INNER JOIN', 'variabel','fakta.id\_variabel=variabel.id')

->join('INNER JOIN', 'kategori','fakta.id\_kategori=kategori.id')

->orderBy('kategori.nama')

->where('variabel.id='.$idVar.' AND wilayah.id\_parent='.$idWil);

$list = $query->all();

//$list = \common\models\Fakta::findBySql('SELECT DISTINCT id\_variabel FROM fakta')->asArray()->all();

//SELECT DISTINCT topik.id, topik.nama FROM fakta

//RIGHT JOIN topik ON fakta.id\_variabel=topik.id

$selected = **null**;

**if** ($id != **null** && **count**($list) > 0) {

$selected = '';

**foreach** ($list **as** $i => $variabel) {

//$topik[$i]=$variabel['namaTopik'];

//$namaVariabel=\frontend\models\Variabel::find()->andWhere(['kode'=>$account['kode\_variabel']])->asArray()->all();

$out[] = ['id' => $variabel['id'], 'name' => $variabel['nama']];

**if** ($i == 0) {

$selected = $variabel['id'];

}

}

// Shows how you can preselect a value

**return** (['output' => $out, 'selected'=>$selected]);

}

}

**return** (['output' => '', 'selected'=>'']);

}

**public** **function** actionChildWilayah() {

\Yii::$app->response->format = \yii\web\Response::FORMAT\_JSON;

$out = [];

**if** (**isset**($\_POST['depdrop\_parents'])&&end($\_POST['depdrop\_parents'])!=**null**) {

$id = end($\_POST['depdrop\_parents']);

$query = **new** Query;

$query->select('id,nama')

->from('wilayah')

->where('tipe='.$id);

$list = $query->all();

$selected = **null**;

**if** ($id != **null** && **count**($list) > 0) {

$selected = '';

**foreach** ($list **as** $i => $account) {

//$namaVariabel=\frontend\models\Variabel::find()->andWhere(['kode'=>$account['kode\_variabel']])->asArray()->all();

//$wilayah = \common\models\Wilayah::findOne(['id' => $account['id\_wilayah'],]);

$out[] = ['id' => $account['id'], 'name' => $account['nama']];

**if** ($i == 0) {

$selected = $account['id'];

}

}

// Shows how you can preselect a value

**return** (['output' => $out, 'selected'=>$selected]);

}

}

**return** (['output' => '', 'selected'=>'']);

}

// controlller sapu

**public** **function** actionDatasum($twil,$nwil,$namawil){

\Yii::$app->response->format = \yii\web\Response::FORMAT\_JSON;

//Yii::$app->response->format = 'jsongis';

**switch** ($twil) {

**case** 1 : {$query = **new** Query;

$query->select("COUNT(DISTINCT `R102N`) AS jkota, COUNT(DISTINCT `R103N`)as jkec, COUNT(DISTINCT `R104N`) as jdesa")

->from('kepri')

->where('R101N='."'".$namawil."'");} **break**;

**case** 2 : {$query = **new** Query;

$query->select("COUNT(DISTINCT `R102N`) AS jkota, COUNT(DISTINCT `R103N`)as jkec, COUNT(DISTINCT `R104N`) as jdesa")

->from('kepri')

->where('R102N='."'".$namawil."'");} **break**;

**case** 3 : {$query = **new** Query;

$query->select("COUNT(DISTINCT `R102N`) AS jkota, COUNT(DISTINCT `R103N`)as jkec, COUNT(DISTINCT `R104N`) as jdesa")

->from('kepri')

->where('R103N='."'".$namawil."'");} **break**;

}

$rows = $query->all();

$command = $query->createCommand();

$rows = $command->queryAll();

**return** (['data'=>$rows]);

}

**public** **function** actionDatavis($nwil,$twil){

\Yii::$app->response->format = \yii\web\Response::FORMAT\_JSON;

//Yii::$app->response->format = 'jsongis';

$query = **new** Query;

$query->select('LEVEL\_WILA, ID\_WIL, R101, R101N, R102N, R103N, R104N, R401, R404A, R501A1, R501A2, R501B,

R503, R507A,R701A\_K2,R701A\_K3,R701B\_K2, R701B\_K3, R701C\_K2, R701C\_K3, R701D\_K2,

R701D\_K3, R701E\_K2, R701E\_K3, R701F\_K2, R701F\_K3, R701G\_K2, R701G\_K3, R701H\_K3,

R701I\_K3, R704A\_K3, R704B\_K3, R704C\_K3, R704D\_K3, R704E\_K3, R704F\_K3, R704G\_K3,

R704H\_K3, R704I\_K3, R704J\_K3, R704K\_K3, R704L\_K3,R706A1, R706A2, R706B, R706C,

R706D,R1201A, R1201B, R1201C, R1201D, R1201E, R1201F, R1201G, R1201H, R1204A,

R1204B, R1205, R1206, R1209, R1210, R1211, R1215A\_K3, R1215B\_K3, R802, R803A,

R803B, R803C, R803D, R803E, R803F, R803G, R803H, R1305, R1303B, R601A\_K2, R601B\_K2,

R601C\_K2, R601D\_K2, R601E\_K2, R601F\_K2, R601G\_K2, R601H\_K2, R601I\_K2, R601J\_K2')

->from('kepri')

->where('ID\_WIL='.$nwil);

$rows = $query->all();

$command = $query->createCommand();

$rows = $command->queryAll();

**return** (['data'=>$rows]);

}

**public** **function** beforeAction($action) {

$this->enableCsrfValidation = **false**; // <-- here

**return** **parent**::beforeAction($action);

}

}

“… sengaja dikosongkan …”

Lampiran 2. *Script* map.js

**var** layerPopup**;** //popup

**var** map**;** //variabel untuk menampung peta

**var** kodewilayah**;**

**var** serie**;**

**var** indexTahun**=**0**;**

**var** jlhKelas **=** 4**;**

**var** metode **=** **[**'natural breaks'**,**'equal intervals'**,**'standard deviation'**,**'arithmetic progression'**,**'geometric progression'**,**'quantiles'**];**

**var** noMetode **=** 0**;**

//untuk posisi back dan forward

**var** lokasi **=** **new** Array**();**

**var** posisi **=** **-**1**;**

//warna

**var** noWarna **=** 0**;**

**var** warnaa **=** **new** Array**();**

**var** warnab **=** **new** Array**();**

**var** kelasWarna **=** **new** Array**();**

kelasWarna**[**9**]=[**8**,**7**,**6**,**5**,**4**,**3**,**2**,**1**,**0**];**

kelasWarna**[**8**]=[**8**,**7**,**6**,**5**,**4**,**3**,**2**,**1**];**

kelasWarna**[**7**]=[**7**,**6**,**5**,**4**,**3**,**2**,**1**];**

kelasWarna**[**6**]=[**7**,**6**,**5**,**4**,**3**,**1**];**

kelasWarna**[**5**]=[**7**,**6**,**4**,**3**,**1**];**

kelasWarna**[**4**]=[**7**,**6**,**4**,**2**];**

kelasWarna**[**3**]=[**7**,**4**,**2**];**

warnaa**[**0**]=[**'#67000D'**,**'#A50F15'**,**'#CB181D'**,**'#EF3B2C'**,**'#FB6A4A'**,**'#FC9272'**,**'#FCBBA1'**,**'#FEE0D2'**,**'#FFF5F0'**];**//merah

warnaa**[**1**]=[**'#3F007D'**,**'#54278F'**,**'#6A51A3'**,**'#807DBA'**,**'#9E9AC8'**,**'#BCBDDC'**,**'#DADAEB'**,**'#EFEDF5'**,**'#FCFBFD'**];**//ungu

warnaa**[**2**]=[**'#08306B'**,**'#08519C'**,**'#2171B5'**,**'#4292C6'**,**'#6BAED6'**,**'#9ECAE1'**,**'#C6DBEF'**,**'#DEEBF7'**,**'#F7FBFF'**];**//biru

warnaa**[**3**]=[**'#00441B'**,**'#006D2C'**,**'#238B45'**,**'#41AB5D'**,**'#74C476'**,**'#A1D99B'**,**'#C7E9C0'**,**'#E5F5E0'**,**'#F7FCF5'**];**//hijau

warnaa**[**4**]=[**'#662506'**,**'#993404'**,**'#CC4C02'**,**'#EC7014'**,**'#FE9929'**,**'#FEC44F'**,**'#FEE391'**,**'#FFF7BC'**,**'#FFFFE5'**];**//orange

warnaa**[**5**]=[**'#7F0000'**,**'#B30000'**,**'#D7301F'**,**'#EF6548'**,**'#FC8D59'**,**'#FDBB84'**,**'#FDD49E'**,**'#FEE8C8'**,**'#FFF7EC'**];**//merah tua

warnaa**[**6**]=[**'#49006A'**,**'#7A0177'**,**'#AE017E'**,**'#DD3497'**,**'#F768A1'**,**'#FA9FB5'**,**'#FCC5C0'**,**'#FDE0DD'**,**'#FFF7F3'**];**//merah ungu

warnaa**[**7**]=[**'#081D58'**,**'#253494'**,**'#225EA8'**,**'#1D91C0'**,**'#41B6C4'**,**'#7FCDBB'**,**'#C7E9B4'**,**'#EDF8B1'**,**'#FFFFD9'**];**//hijau biru

warnaa**[**8**]=[**'#004529'**,**'#006837'**,**'#238443'**,**'#41AB5D'**,**'#78C679'**,**'#ADDD8E'**,**'#D9F0A3'**,**'#F7FCB9'**,**'#FFFFE5'**];**//hijau kuning

warnab**[**0**]=[**'#01665E'**,**'#35978F'**,**'#80CDC1'**,**'#C7EAE5'**,**'#F5F5F5'**,**'#F6E8C3'**,**'#DFC27D'**,**'#BF812D'**,**'#8C510A'**];**//biru coklat

warnab**[**1**]=[**'#4D9221'**,**'#7FBC41'**,**'#B8E186'**,**'#E6F5D0'**,**'#F7F7F7'**,**'#FDE0EF'**,**'#F1B6DA'**,**'#DE77AE'**,**'#C51B7D'**];**//hijau ungu

warnab**[**2**]=[**'#1B7837'**,**'#5AAE61'**,**'#A6DBA0'**,**'#D9F0D3'**,**'#F7F7F7'**,**'#E7D4E8'**,**'#C2A5CF'**,**'#9970AB'**,**'#762A83'**];**//hijau ungu tua

warnab**[**3**]=[**'#542788'**,**'#8073AC'**,**'#B2ABD2'**,**'#D8DAEB'**,**'#F7F7F7'**,**'#FEE0B6'**,**'#FDB863'**,**'#E08214'**,**'#B35806'**];**//ungu oranye

warnab**[**4**]=[**'#2166AC'**,**'#4393C3'**,**'#92C5DE'**,**'#D1E5F0'**,**'#F7F7F7'**,**'#FDDBC7'**,**'#F4A582'**,**'#D6604D'**,**'#B2182B'**];**//biru merah

warnab**[**5**]=[**'#4D4D4D'**,**'#999999'**,**'#E0E0E0'**,**'#FFFFFF'**,**'#FFFFFF'**,**'#FDDBC7'**,**'#F4A582'**,**'#EF8A62'**,**'#B2182B'**];**//hitam merah

warnab**[**6**]=[**'#1A9850'**,**'#66BD63'**,**'#A6D96A'**,**'#D9EF8B'**,**'#FFFFBF'**,**'#FEE08B'**,**'#FDAE61'**,**'#F46D43'**,**'#D73027'**];**//hijau merah

warnab**[**7**]=[**'#3288BD'**,**'#66C2A5'**,**'#ABDDA4'**,**'#E6F598'**,**'#FFFFBF'**,**'#FEE08B'**,**'#FDAE61'**,**'#F46D43'**,**'#D53E4F'**];**//hijau biru merah

**var** aVar**;** //tampung kode variabel

**var** aKat**;** //tampung kode Kategori

**var** layerprovinsi**=**0**;**

**var** peta**;**

**var** judul**;**//judul peta

//array untuk menampung idprovinsi, tahun, bulan, satuan nama provinsi, dan nilai

**var** dataTabel **=** **new** Array**();**

**var** jumlah**=new** Array**();**

**var** tahun**=new** Array**();**

**var** idProvinsi**=new** Array**();**

**var** namaProvinsi**=new** Array**();**

**var** bulan**=new** Array**();**

**var** satuan**=new** Array**();**

// add compas

**var** compas **=** L**.**control**({**position**:** 'topleft'**});**

compas**.**onAdd **=** **function** **(**map**)** **{**

**var** div **=** L**.**DomUtil**.**create**(**'div'**,** 'compas'**);**

div**.**innerHTML **=**'<img src="logo/compass2.png" width="25px">'**;**

**return** div**;**

**}**

// menampilkan info wilayah ketika di hover

**var** info **=** L**.**control**();**

info**.**onAdd **=** **function** **(**map**)** **{**

**this.**\_div **=** L**.**DomUtil**.**create**(**'div'**,** 'info'**);**

**this.**update**();**

**return** **this.**\_div**;**

**};**

info**.**update **=** **function** **(**props**)** **{**

**this.**\_div**.**innerHTML **=(**props **?**

'<b>' **+** namaProvinsi**[**props**.**ID**]** **+** '</b><br />' **+** dataTabel**[**vVal**][**props**.**ID**]**

**:** 'Hover over a state'**);**

**};**

//membuat Lagenda

**var** grades**=new** Array**();**

**function** legenda**(){**

$**(**'#legend'**).**html**(**''**);**

$**(**'#legend'**).**append**(**'<div class="row"><div class="col-md-6"><h4>Legenda</h4></div><div class="col-md-6"><a class="pull-right" style="cursor: pointer" data-toggle="modal" data-target="#myModal"><span class="glyphicon glyphicon-list" data-toggle="tooltip" data-placement="top" title="Ubah Legenda"></span></a></div></div>'**);**

**for** **(var** i **=** 0**;** i **<** grades**.**length**;** i**++)** **{**

$**(**'#legend'**).**append**(**'<i style="background:' **+** getColor**(**i**)** **+** '"></i><span> ' **+** grades**[**i**]** **+** '</span><br>'**);**

**}**

**}**

**var** vVal**;**

//style untuk polygon

**var** styleProvinsi **={**

'fillColor'**:**'#ff0000'**,**

'fillOpacity'**:**0.6**,**

'color'**:**'#ffffff'**,**

'opacity'**:**1**,**

'weight'**:**2**,**

'dashArray'**:**3**,**

**}**

**var** styleSelected **={**

'weight'**:** 5**,**

'color'**:** '#666'**,**

'dashArray'**:** ''**,**

'fillOpacity'**:** 0.7

**}**

**function** initializez**()**

**{**

//panggil base map

map **=** L**.**map**(**'map'**).**setView**([-**1.889306**,**114.697266**],** 4**);**

**var** osmUrl**=**'http://{s}.tile.openstreetmap.org/{z}/{x}/{y}.png'**;**

**var** osmAttrib**=**'BPS RI'**;**

**var** osm2 **=** **new** L**.**TileLayer**(**osmUrl**,** **{**minZoom**:** 0**,** maxZoom**:** 13**,** attribution**:** osmAttrib **}).**addTo**(**map**);**

**var** defaultLayer **=** L**.**tileLayer**.**provider**(**'Esri.WorldTopoMap'**).**addTo**(**map**);**

**var** baseLayers **=** **{**

'Thunderforest Landscape'**:** L**.**tileLayer**.**provider**(**'Thunderforest.Landscape'**),**

'MapQuest Aerial'**:** L**.**tileLayer**.**provider**(**'MapQuestOpen.Aerial'**),**

'Stamen Watercolor'**:** L**.**tileLayer**.**provider**(**'Stamen.Watercolor'**),**

'Esri WorldStreetMap'**:** L**.**tileLayer**.**provider**(**'Esri.WorldStreetMap'**),**

'Esri DeLorme'**:** L**.**tileLayer**.**provider**(**'Esri.DeLorme'**),**

'Esri WorldImagery'**:** L**.**tileLayer**.**provider**(**'Esri.WorldImagery'**),**

'Esri WorldTerrain'**:** L**.**tileLayer**.**provider**(**'Esri.WorldTerrain'**),**

'Esri WorldShadedRelief'**:** L**.**tileLayer**.**provider**(**'Esri.WorldShadedRelief'**),**

'Esri WorldPhysical'**:** L**.**tileLayer**.**provider**(**'Esri.WorldPhysical'**),**

'Esri OceanBasemap'**:** L**.**tileLayer**.**provider**(**'Esri.OceanBasemap'**),**

'Esri NatGeoWorldMap'**:** L**.**tileLayer**.**provider**(**'Esri.NatGeoWorldMap'**),**

'Esri WorldGrayCanvas'**:** L**.**tileLayer**.**provider**(**'Esri.WorldGrayCanvas'**),**

'Acetate'**:** L**.**tileLayer**.**provider**(**'Acetate'**)**

**};**

L**.**control**.**layers**(**baseLayers**,{**collapsed**:** **true}).**addTo**(**map**);**

**var** skala **=** L**.**control**.**scale**({**position**:**'bottomright'**}).**addTo**(**map**);**

compas**.**addTo**(**map**);**

info**.**addTo**(**map**);**

warnalegenda**();**

**}**

//highlight ketika mouse diarahkan

**function** highlightFeature**(**e**)**

**{**

**var** layer**=**e**.**target**;**

layer**.**setStyle**({**

color**:** '#666'**,**

dashArray**:** ''**,**

fillOpacity**:** 0.9**,**

**});**

**if** **(!**L**.**Browser**.**ie **&&** **!**L**.**Browser**.**opera**)** **{**

layer**.**bringToFront**();**

**}**

info**.**update**(**layer**.**feature**.**properties**);**

**}**

//kembalikan style ketika tidak ditunjuk oleh mouse

**function** resetHighlight**(**e**)** **{**

**var** layer**=**e**.**target**;**

layer**.**setStyle**({**

color**:** 'white'**,**

dashArray**:** ''**,**

fillOpacity**:** 0.9**,**

**});**

info**.**update**();**

**}**

**function** zoomToFeature**(**e**)** **{**

map**.**fitBounds**(**e**.**target**.**getBounds**());**

calldata**(**e**.**target**.**feature**.**properties**.**ID**);**

**}**

**function** callProvMap**(**aWil**)**

**{**

//ajax untuk memanggil polygon provinsi dari geoserver

//loadingt(20);

$**.**ajax**({**

url**:** '?r=geoserver-url/load-peta&idWil='**+**aWil**,**

type **:** 'POST'**,**

dataType **:** 'json'**,**

success**:** **function(**data**)**

**{**

kodewilayah**=**aWil**;**

**if(**aWil**!=**lokasi**[**posisi**])**

**{**lokasi**[(**posisi**+**1**)]** **=** aWil**;**

posisi**++;**

**if(**posisi**>**0**)** $**(**'#untukBack'**).**html**(**'<span class="glyphicon glyphicon-arrow-left" style="cursor: pointer" aria-hidden="true" data-toggle="tooltip" data-placement="top" title="Ke Peta Level Sebelumnya" onclick="petaBack()"></span>'**);**

**if(**posisi**==(**lokasi**.**length**-**1**))** $**(**'#untukForward'**).**html**(**'<span class="glyphicon glyphicon-arrow-right" style="color: #d9d9d9;" aria-hidden="true"></span>'**);**

**else** $**(**'#untukForward'**).**html**(**'<span class="glyphicon glyphicon-arrow-right" style="cursor: pointer" aria-hidden="true"data-toggle="tooltip" data-placement="top" title="Ke Peta Selanjutnya (forward)" onclick="petaNext()"></span>'**);**

**}**

map**.**removeLayer**(**layerprovinsi**);** //hapus layer sebelumnya

//kasih layer geoJson

layerprovinsi**=**L**.**geoJson**(**data**,{**style**:**styleProvinsi**}).**addTo**(**map**);**

map**.**fitBounds**(**layerprovinsi**.**getBounds**());**

//kasih warna

setVariableTahun**(**tahun**[**indexTahun**]);**

loadingt**(**100**);**

**}**

**});**

**}**

**function** setVariableTahun**(**valTahun**)** **{**

vVal**=**valTahun**;**

**var** jp**=new** Array**();**

**var** npr**=new** Array**();**

**var** i**=**0**;**

//tampung data yang akan diwarnai

idProvinsi**.**forEach**(function(**untukSeries**)** **{**

npr**[**i**]=**namaProvinsi**[**untukSeries**];**

jp**[**i**]=**dataTabel**[**valTahun**][**untukSeries**];**

i**++;**

**});**

//menutup popup

**if** **(**layerPopup **&&** map**)** **{**

map**.**closePopup**(**layerPopup**);**

layerPopup **=** null**;**

**}**

serie **=** **new** geostats**(**jp**);**

$**(**'#rata-rata'**).**html**(**'<b>'**+**serie**.**mean**().**toFixed**(**2**)+**'</b>'**);**

$**(**'#nilai-min'**).**html**(**'<b>'**+**serie**.**min**()+**'</b>'**);**

$**(**'#nilai-max'**).**html**(**'<b>'**+**serie**.**max**()+**'</b>'**);**

$**(**'#varian'**).**html**(**'<b>'**+**serie**.**variance**().**toFixed**(**2**)+**'</b>'**);**

$**(**'#s-d'**).**html**(**'<b>'**+**serie**.**stddev**().**toFixed**(**2**)+**'</b>'**);**

$**(**'#kov'**).**html**(**'<b>'**+**serie**.**cov**().**toFixed**(**2**)+**'</b>'**);**

//metode

**if(**noMetode**==**0**)** serie**.**getClassJenks**(**jlhKelas**);**

**else** **if(**noMetode**==**1**)** serie**.**getClassEqInterval**(**jlhKelas**);**

**else** **if(**noMetode**==**2**)** serie**.**getClassStdDeviation**(**jlhKelas**);**

**else** **if(**noMetode**==**3**)** serie**.**getClassArithmeticProgression**(**jlhKelas**);**

**else** **if(**noMetode**==**4**)** serie**.**getClassGeometricProgression**(**jlhKelas**);**

**else** serie**.**getClassQuantile**(**jlhKelas**);**

grades **=** serie**.**ranges**;**

layerprovinsi**.**eachLayer**(function(**layer**)** **{**

**var** kodeprovinsi**=**layer**.**feature**.**properties**.**ID**;**

layer**.**setStyle**(**style**(**dataTabel**[**vVal**][**kodeprovinsi**]));**

layer**.**on**({**

mouseover**:** highlightFeature**,**

mouseout**:** resetHighlight**,**

dblclick **:** zoomToFeature**,**

click **:** grafikPetaline**,**

**});**

// layer.bindPopup('<div id="lineChart" style="width:300px; height:300px"></div>');

**});**

legenda**();** //tampilkan lagenda

grafikPeta**(**npr**,**jp**);**//tampilkan grafik dibawah peta

**}**

**function** style**(**nilaiData**)** **{**

**return** **{**

fillColor**:** getColor**(**serie**.**getRangeNum**(**nilaiData**)),**

weight**:** 2**,**

opacity**:** 1**,**

color**:** 'white'**,**

dashArray**:** ''**,**

fillOpacity**:** 0.9**,**

**};**

**}**

**function** calldatabaru**(){**

**var** aWil **=** $**(**'#wilayah-nama'**).**val**();**

aKat **=** $**(**'#kategori-nama'**).**val**();**

**var** aVarn **=** $**(**'#variabel-nama'**).**val**();**

aVar **=** aVarn**.**substring**(**10**);**

//sapu

**var** twil **=** $**(**'#tipewilayah-nama'**).**val**();**

**var** nwil **=** $**(**'#wilayah-nama'**).**val**();**

**var** namawil **=** $**(**'#wilayah-nama option:selected'**).**text**();**

calldatasum**(**twil**,**nwil**,**namawil**);**calldatavis**(**twil**,**nwil**);**

//sapu

calldata**(**aWil**);**

**}**

**function** getColor**(**indexWarna**)** **{**

**if(**noWarna**<**9**)** **return** warnaa**[**noWarna**][**kelasWarna**[**jlhKelas**][**indexWarna**]];**

**else** **return** warnab**[**noWarna**-**9**][**kelasWarna**[**jlhKelas**][**indexWarna**]];**

**}**

**function** loadingt**(**t**){**

**if(**t**==**0**){**

$**(**'#loadingmap'**).**html**(**'<img src="logo/ajax-loader.gif" style="margin-Left:46%">'**);**

**}**

**else** **{**

$**(**'#loadingmap'**).**html**(**''**);**

**}**

**}**

**function** zoomdefault**(){**

**if(**layerprovinsi**!=**0**)** map**.**fitBounds**(**layerprovinsi**.**getBounds**());**

**else** map**.**setView**(new** L**.**LatLng**(-**1.889306**,**114.697266**),** 4**);**

**}**

**function** petaBack**(){**

calldata**(**lokasi**[**posisi**-**1**]);**

posisi**=**posisi**-**2**;**

**if(**posisi**==-**1**){**$**(**'#untukBack'**).**html**(**'<span class="glyphicon glyphicon-arrow-left" style="color: #d9d9d9;" aria-hidden="true"></span>'**);}**

**}**

**function** petaNext**(){**

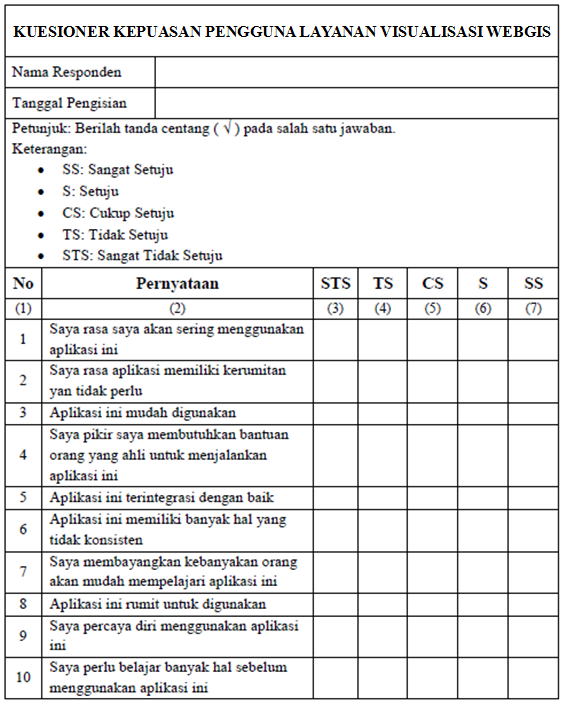
console**.**log**(**lokasi**[(**posisi**+**1**)]);**

calldata**(**lokasi**[(**posisi**+**1**)]);**

**}**

“… sengaja dikosongkan …”

Lampiran 3. Kuesioner System Usability Scale



## RIWAYAT HIDUP

Penulis dilahirkan di Anggana Kabupaten Kutai Kartanegara pada tanggal 1 Otober 1993. Penulis merupakan anak kedua dari 7 bersaudara keluarga Bapak Busrani dan Ibu Sumiyati.

Pada Tahun 1999 penulis memulai pendidikan dasar di SDN 007 Anggana dan menamatkannya pada tahun 2005. Setelah itu penulis melanjutkan pendidikannya di SMP Negeri 1 Anggana dan lulus pada tahun 2008 dan diteruskan ke jenjang menengah atas di SMAN 3 Unggulan Tenggarong. Pada Tahun 2011, setelah lulus SMA penulis melanjutkan studinya di Sekolah Tinggi Ilmu Statistik dan mengambil jurusan komputasi statistik pada tahun kedua pendidikannya disana. Pada penghujung tahun 2015, penulis berhasil menyelesaikan program pendidikannya D-IVnya di Sekolah Tinggi Ilmu Statistik Jakarta.