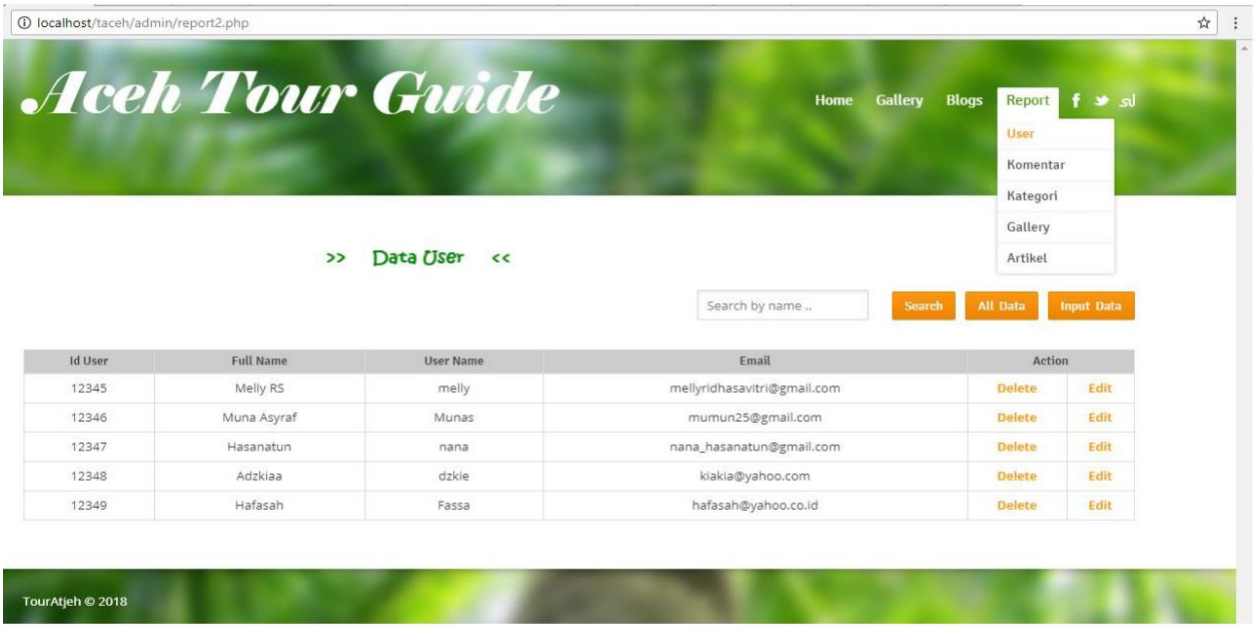
Portal Berita Wisata Menggunakan Database Mongodb

Proses pembuatan website berbasis portal berita tentang wisata di Aceh. Yang di mana portal berita ini di buat menggunakan system dokumentasi berbasis MongoDB. Portal berita ini akan selalu online 7 x 24 jam. Jumlah server yang disimpan lebih dari 3 server yang akan berguna untuk distribusi replikasi data dam sharding data, sekalian membackup data keseluruhan. Berikut adalah langkah langkah pembuatan portal beritanya :

A. Membuat data data utama yang akan di simpan (Collection) pada mongodb menggunakan PHP

1. Data pengguna
   Data pengguna ini berisi tentang ID pengguna, nama pengguna, email dan password

```
spof:PRIMARY> db.user.find()
{ "_id" : ObjectId("5a7081b90e506081a69a1a05"), "id_user" : "12345", "nama" : "Melly RS", "username" : "melly", "password" : "12345", "email" : "mellyridhasavitri@gmail
.com" }
{ "_id" : ObjectId("5a71cb59b11dab1c2300002b"), "id_user" : "12346", "username" : "Munas", "password" : "12346", "nama" : "Muna Asyraf", "email" : "mumun25@gmail.com" }

{ "_id" : ObjectId("5a71d2acb11dab183000002d"), "id_user" : "12347", "username" : "nana", "password" : "12347", "nama" : "Hasanatun", "email" : "nana_hasanatun@gmail.co
m" }
{ "_id" : ObjectId("5a71d2f2b11dab183000002e"), "id_user" : "12348", "username" : "dzkie", "password" : "12348", "nama" : "Adzkiaa", "email" : "kiakia@yahoo.com" }
{ "_id" : ObjectId("5a71d31ab11dab183000002f"), "id_user" : "12349", "username" : "Fassa", "password" : "12349", "nama" : "Hafasah", "email" : "hafasah@yahoo.co.id" }
```

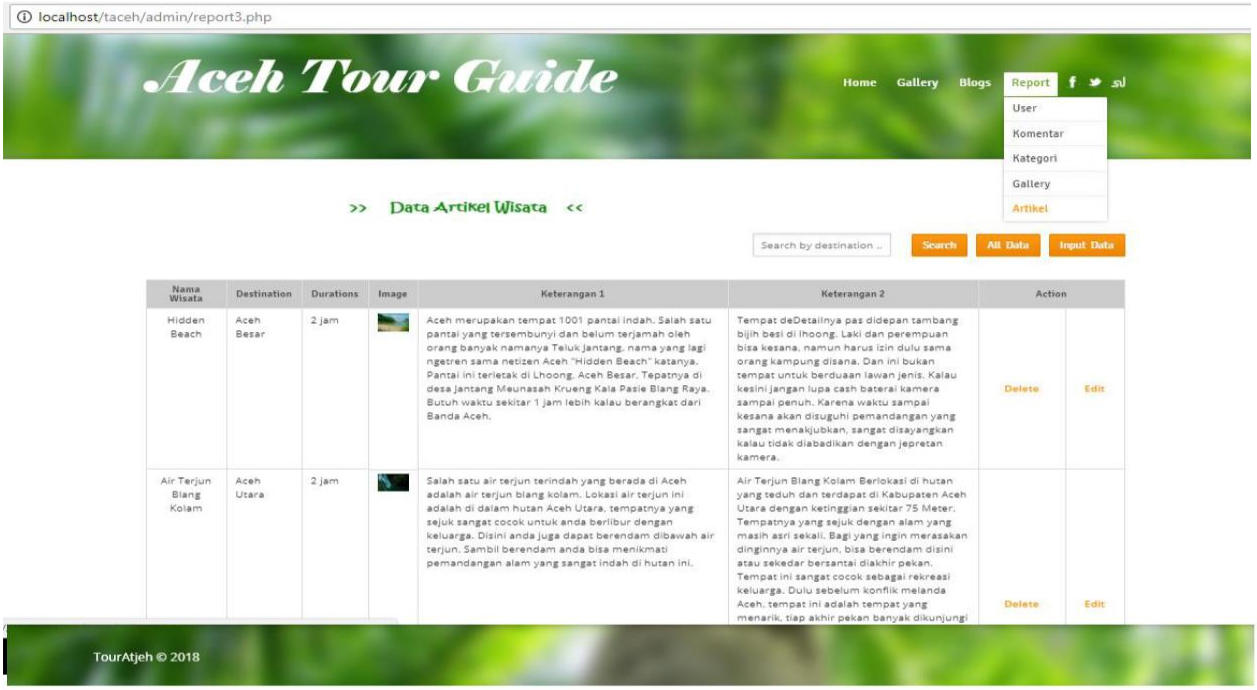



```
131 <?php
132     $collection = $db->user;
133     $cursor = $collection->find();
134     foreach ($cursor as $val) {
135 ?>
136     <tbody>
137      <tr class="nowrap">
138                 <td><center><?php echo $val["id_user"] ?></td>
139                 <td><center><?php echo $val["nama"] ?></td>
140                 <td><center><?php echo $val["username"] ?></td>
141                 <td><center><?php echo $val["email"] ?></center></td>
142                 <td width="81"><div align="center"><strong><a href="<?php echo "action_delete_user.php?id_user=" . $val
143             <td width="48"><div align="center"><strong><a href="<?php echo "edit_user.php?id_user=" . $val["id_user"];?
144             </strong></div>
145 </td>
146      </tr>
147           <?php } ?>
```

2. Data Artikel

Data artikel (wisata) berisi tentang id wisata, nama wisata, destinasi, wisata, gambar, keteranga 1, keterangan 2 .

```
spof:PRIMARY> db.wisata.find().pretty()
{
        "_id" : ObjectId("5a7062aa0e506081a69a1a02"),
        "id_wisata" : "3241",
        "nama_wisata" : "Air Terjun Kutamalaka",
        "destinasi" : "Samahani",
        "durasi" : "2 jam",
        "gambar" : "8bca8e6eb79ec192dc9c45b1042cb071.jpg",
        "ket1" : "Air terjun yang sedang anda lihat sekarang bernama air terjun Kuta Malaka berada di Kuta Malaka, Samahani, Aceh Besar. Air terjun ini selain menjadi o
byek wisata dan tempat rekreasi, juga digunakan untuk kepentingan masyarakat. Seperti Pengairan/Irigasi di sawah-sawah, untuk mandi, dan keperluan masyarakat lainnya. T
idak banyak rakyat Indonesia, khususnya rakyat Aceh yang tahu dimana obyek wisata ini. Oleh karena itu kewajiban Saya sebagai warga kuta malaka untuk memberitahukan keb
eradaan tempat ini. Untuk menuju tempat ini kita harus mendaki bukit dan gunung, tapi saya berani jamin bahwa anda tidak akan rugi dan bosan karena anda akan ditemani o
leh udara yang sejuk dan panorama yang indah. Sehinga rasa lelah dan capek yang anda rasakan akan hilang dan ditebus dengan kesenangan yang tiada tara. Kali ini kita me
ngunjungi salah satu obyek wisata air terjun yang berada di Kuta Malaka, Kecamatan Kuta Malaka, Kabupaten Aceh Besar",
        "ket2" : "Untuk menuju ke lokasi kita harus menempuh perjalanan sejauh 30 km dari pusat kota Banda Aceh. Lokasi wisata ini memang belum dikelola secara profesio
nal baik oleh Pemda Aceh Besar maupun swasta, sehingga tidak banyak para wisata yang berkunjung dan fasilitas tidak tersedia. Lokasi yang dilingkupi oleh hutan perawan
yang sangat sejuk dan memberikan nuansa tersendiri dan jalan menuju ke lokasi harus melalui pegunungan yang sangat indah."
}
```

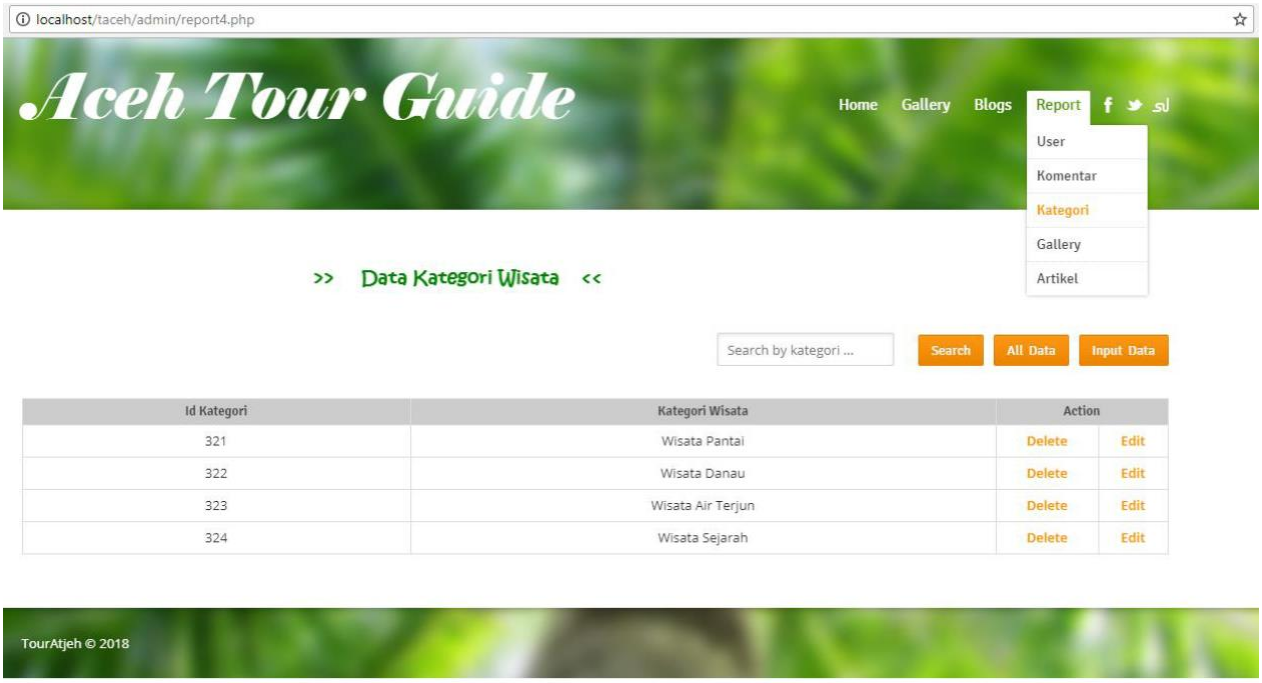



```
<?php
$collection = $db->wisata;
$cursor = $collection->find();
foreach ($cursor as $val) {
?>
    <tbody>
     <tr>
                <td valign="top"><CENTER>
                  <span class="nowrap"><?php echo $val["nama_wisata"] ?></span>
                </center></td>
                <td valign="top" class="nowrap"><?php echo $val["destinasi"] ?></td>
                <td valign="top" class="nowrap"><?php echo $val["durasi"] ?></td>
                <td valign="top" class="nowrap"><img src="../Admin/Uploads/<?php echo $val["gambar"] ?>" alt="" /></td>
                <td valign="top" class="nowrap"><?php echo $val["ket1"] ?></td>
                <td width="81"><div align="center"><center>
                  <strong><a href="<?php echo "action_delete_wisata.php?id_wisata=" . $val["id_wisata"]; ?>" onClick="r
                  </center>
                  </div>
                <td width="48"><div align="center"><center>
                    <strong><a href="edit_wisata.php?id_wisata=<?php echo $val["id_wisata"] ?>" class=" btn="btn" btn-m
                    </center>
                  </div>
                </td>
            </tr>
             <?php } ?>
```

3. Data kategori

Data kategori wisata ini berisi tentang 4 buah wista yaitu wisata pantai, wisata danau, wisata air terjun, dan wisata sejarah .

```
spof:PRIMARY> db.kategori.find()
{ "_id" : ObjectId("5a7296f367b2d7979ed1cd77"), "id_kategori" : "321", "kategori" : "Wisata Pantai" }
{ "_id" : ObjectId("5a72972467b2d7979ed1cd78"), "id_kategori" : "322", "kategori" : "Wisata Danau" }
{ "_id" : ObjectId("5a72976267b2d7979ed1cd79"), "id_kategori" : "323", "kategori" : "Wisata Air Terjun" }
{ "_id" : ObjectId("5a7297a967b2d7979ed1cd7a"), "id_kategori" : "324", "kategori" : "Wisata Sejarah" }
spof:PRIMARY>
```

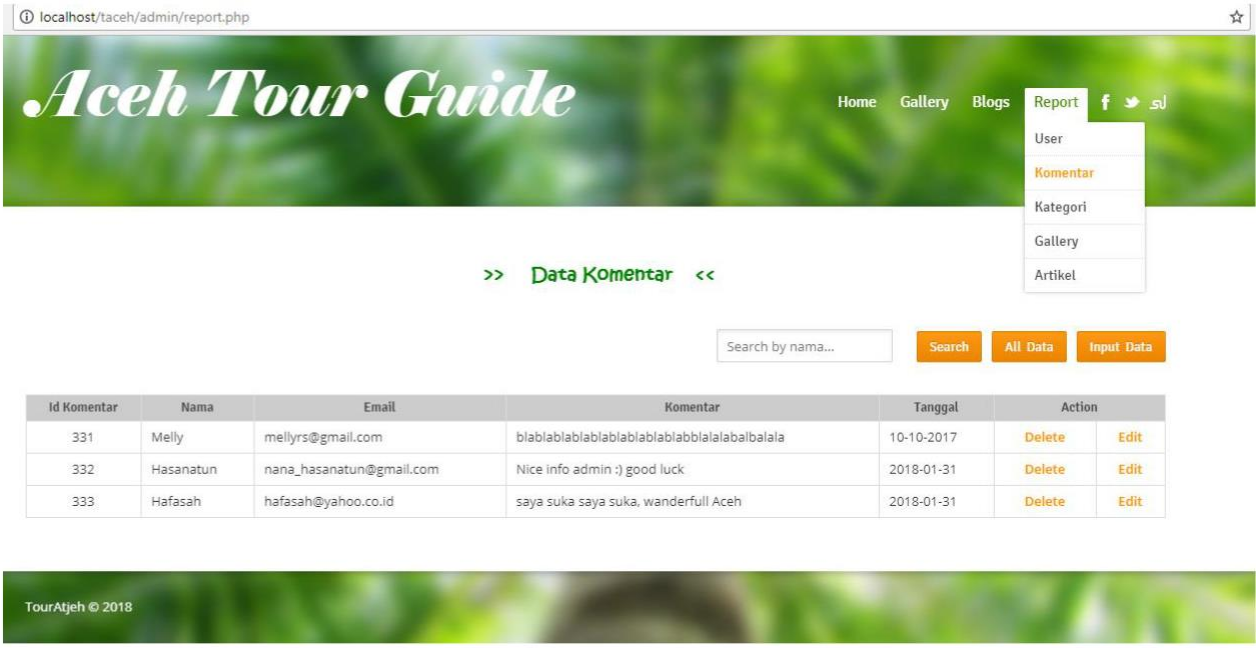



```
<?php
$collection = $db->kategori;
$cursor = $collection->find();
foreach ($cursor as $val) {
?>
    <tbody>
     <tr>
        <td><CENTER>
            <span class="nowrap"><?php echo $val["id_kategori"] ?></span>
        </center></td>
        <td><CENTER>
            <span class="nowrap"><?php echo $val["kategori"] ?></span>
        </center></td>
        <td width="81">
        <div align="center"><center>
        <strong><a href="<?php echo "action_delete_kategori.php?id_kategori=" . $val["id_kategori"]; ?>" onClick="retur
        <span class="nowrap">Delete</span></a></strong>
        </center> </div>
        <td width="48"><div align="center"><center>
        <strong><a href="<?php echo "edit_kategori.php?id_kategori=" . $val["id_kategori"];?> class=" btn="btn" btn-min
        <span class="nowrap">Edit</span></a></strong>
            </center>
         </div>
         </td>
         </tr>
          <?php } ?>
```

4. Data Komentar

Data komentar ini berisi tentang nama, email, isi komentar, dan tanggal yang akan di gunakan apabila pembaca tertarik untuk memberikan suatu pemikiran tentang berita ini.

```
spof:PRIMARY> db.komentar.find()
{ "_id" : ObjectId("5a6d7a9b7d8d226cf84af03a"), "id_komentar" : "331", "nama" : "Melly", "email" : "mellyrs@gmail.com",
 "komentar" : "blablablablablablablablablabblalalabalbalala", "tanggal" : "10-10-2017" }
{ "_id" : ObjectId("5a71d266b11dab183000002c"), "id_komentar" : "332", "komentar" : "Nice info admin :)\r\ngood luck",
"tanggal" : "2018-01-31", "nama" : "Hasanatun", "email" : "nana_hasanatun@gmail.com" }
{ "_id" : ObjectId("5a71d3fdb11dab1830000030"), "id_komentar" : "333", "komentar" : "saya suka saya suka, wanderfull Ac
eh", "tanggal" : "2018-01-31", "nama" : "Hafasah", "email" : "hafasah@yahoo.co.id" }
spof:PRIMARY>
```

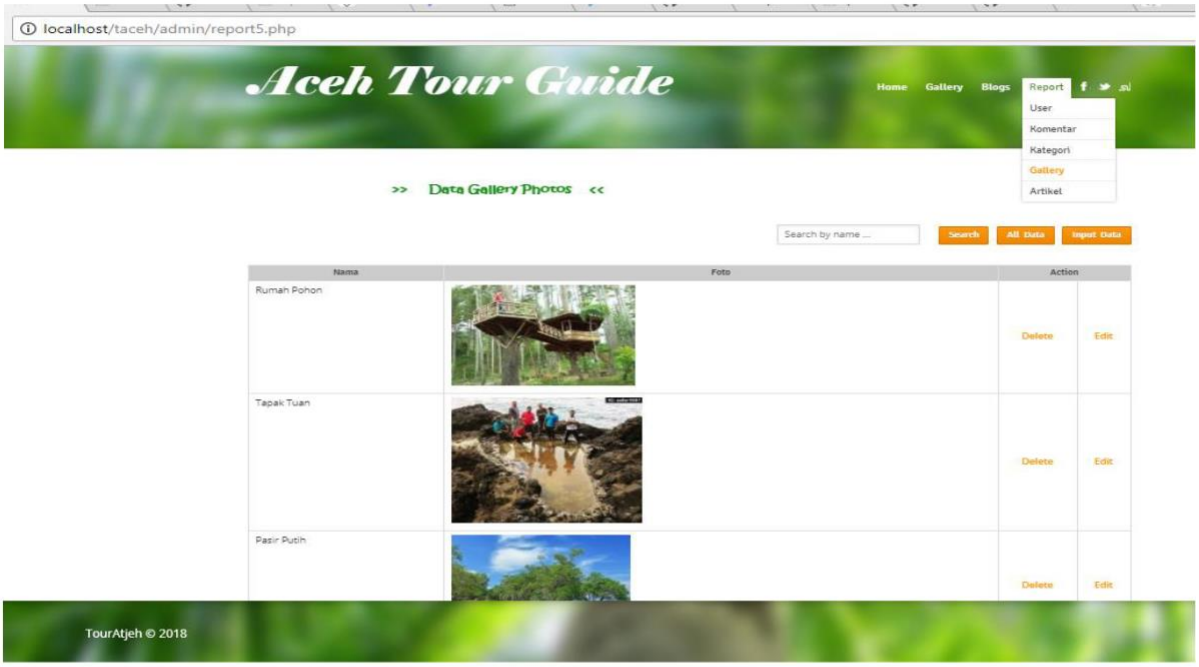



```
130
131  <?php
132  $collection = $db->komentar;
133  $cursor = $collection->find();
134  foreach ($cursor as $val) {
135  ?>
136      <tbody>
137       <tr>
138                 <td><CENTER>
139                   <span class="nowrap"><?php echo $val["id_komentar"] ?></span>
140                 </center></td>
141                 <td class="nowrap"><?php echo $val["nama"] ?></td>
142                 <td class="nowrap"><?php echo $val["email"] ?></td>
143                 <td class="nowrap"><?php echo $val["komentar"] ?></td>
144                 <td class="nowrap"><?php echo $val["tanggal"] ?></td>
145                 <td width="81"><div align="center"><center>
146                   <strong><a href="<?php echo "action_delete_komentar.php?id_komentar=" . $val["id_komentar"]; ?>" onCl
147                   </center>
148                   </div>
149                 <td width="48"><div align="center"><center>
150                    <strong><a href="<?php echo "edit_komentar.php?id_komentar=" . $val["id_komentar"];?> class=" btn="
151                    </center>
152                   </div>
153                 </td>
154          </tr>
155           <?php } ?>
```

5. Data fileattachment

Data file attachment dalam collection galeri, yang berisi ini berisi tentang id galeri, nama, dan foto

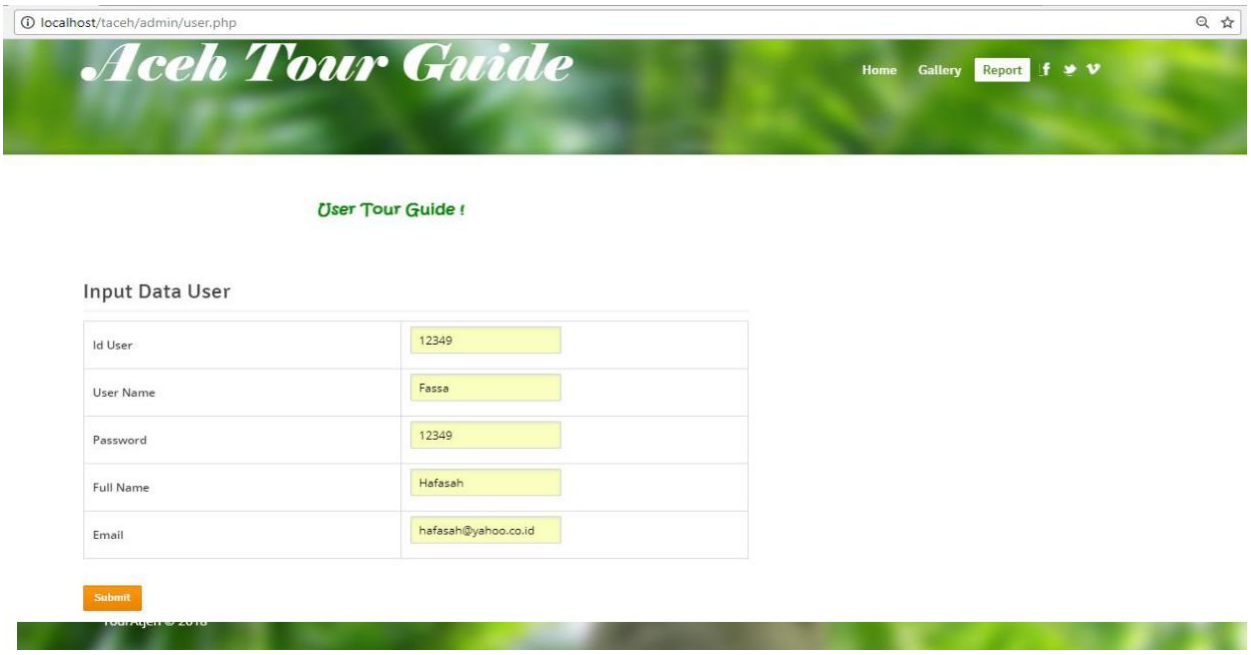
```
spof:PRIMARY> db.galeri.find()
{ "_id" : ObjectId("5a6d5b0a7d8d226cf84af034"), "id_galeri" : "112", "nama" : "Air Terjun Suhom", "foto" : "air_terjun_suhoom.jpg" }
{ "_id" : ObjectId("5a6f465537435ee6719ffe76"), "id_galeri" : "114", "nama" : "Pantang Terong", "foto" : "New Picture (3).bmp" }
{ "_id" : ObjectId("5a72a91467b2d7979ed1cd7b"), "id_galeri" : "113", "nama" : "Lampuuk Beach", "foto" : "images (2).jpg" }
{ "_id" : ObjectId("5a72a97567b2d7979ed1cd7c"), "id_galeri" : "111", "nama" : "Museum Tsunami", "foto" : "images (6).jpg" }
{ "_id" : ObjectId("5a72a9ad67b2d7979ed1cd7d"), "id_galeri" : "115", "nama" : "Pemandian Air Panas", "foto" : "images (7).jpg" }
{ "_id" : ObjectId("5a72a9d167b2d7979ed1cd7e"), "id_galeri" : "116", "nama" : "Pasir Putih", "foto" : "images (8).jpg" }
{ "_id" : ObjectId("5a72aa0067b2d7979ed1cd7f"), "id_galeri" : "117", "nama" : "Tapak Tuan", "foto" : "images (12).jpg" }
{ "_id" : ObjectId("5a72aa7867b2d7979ed1cd80"), "id_galeri" : "118", "nama" : "Rumah Pohon", "foto" : "rmh_pohon.jpg" }
spof:PRIMARY>
```




```
126 <?php
127 $collection = $db->galeri;
128 $cursor = $collection->find()->sort(array("id_galeri" => -1));
129 foreach ($cursor as $val) {
130 ?>
131     <tbody>
132      <tr>
133                 <td valign="top" class="nowrap"><?php echo $val["nama"] ?></td>
134                 <td valign="top" class="nowrap"><img src="../Admin/Uploads/<?php echo $val["foto"] ?>" alt="" /></td>
135                 <td width="81"><div align="center"><center>
136                   <strong><a href="<?php echo "action_delete_galeri.php?id_galeri=" . $val ["id_galeri"]; ?>" onClick="
137                   </center>
138                   </div>
139                 <td width="48"><div align="center"><center>
140                     <strong><a href="<?php echo "edit_galeri.php?id_galeri=".$val ["id_galeri"];?>class="btn="btn" btn-
141                     </center>
142                   </div>
143                 </td>
144         </tr>
145          <?php } ?>
```

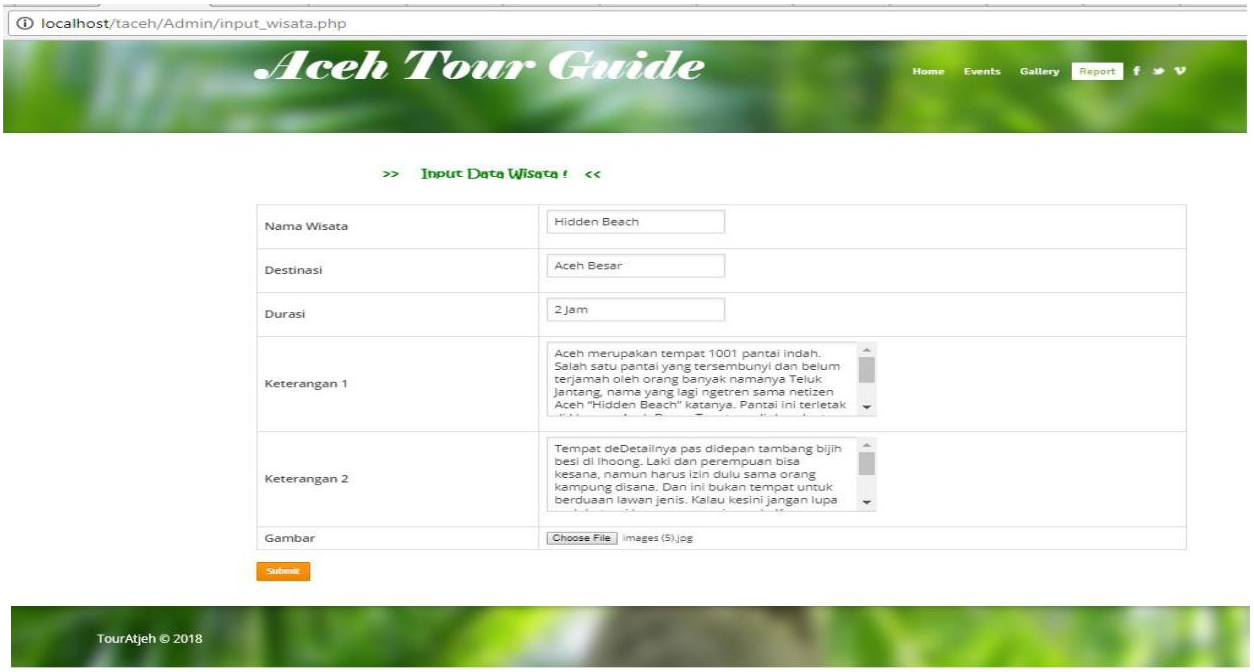
B. Penggisian data data utama nya

1. Input data User



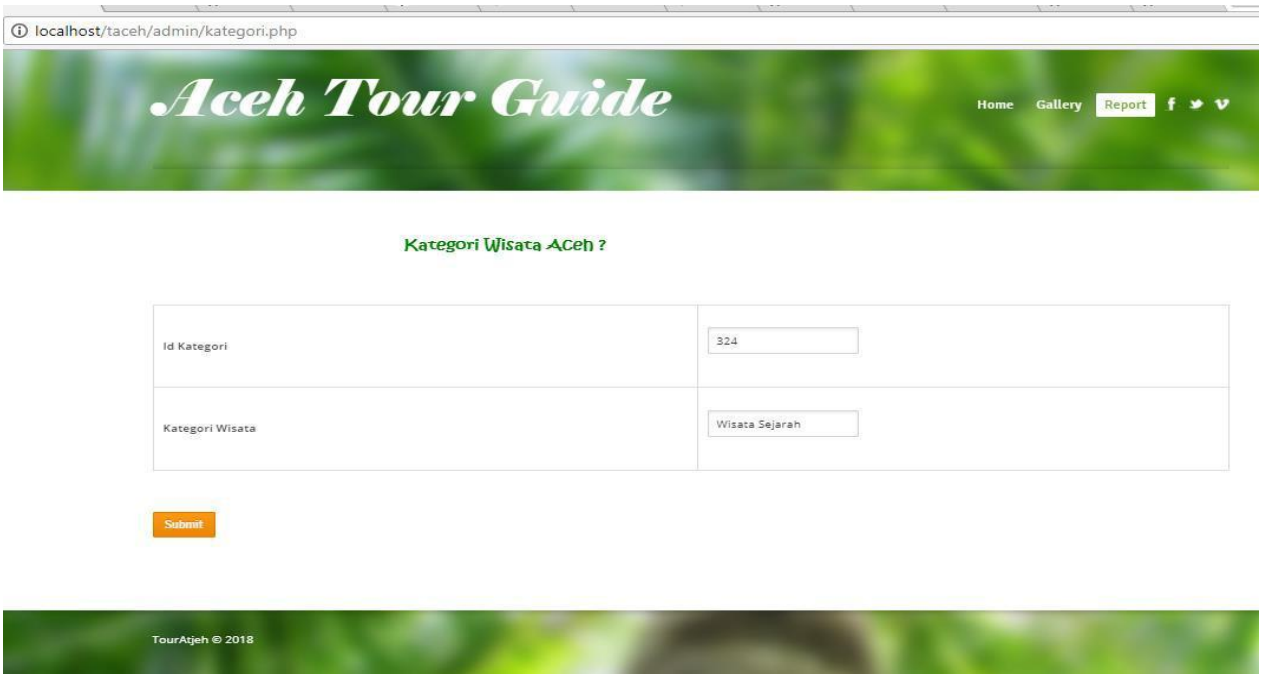

```php
<?php
// connect to mongodb
$m = new MongoClient();
echo "Connection to database successfully";

// select a database
$db = $m->tour;
echo "Database mydb selected";
$collection = $db->user;
echo "Collection selected succsessfully";

$nama = $_POST['name'];
$email = $_POST['email'];
$id_user = $_POST['id_user'];
$username = $_POST['username'];
$password= $_POST['password'];

$document = array(
   "id_user" => "$id_user",
   "username" => "$username",
   "password" => "$password",
   "nama" => "$nama",
   "email" => "$email"
);

$collection->insert($document);
echo "Document inserted successfully";
header("location:report2.php?offset=0");
?>
```

2. Input data artikel wisata



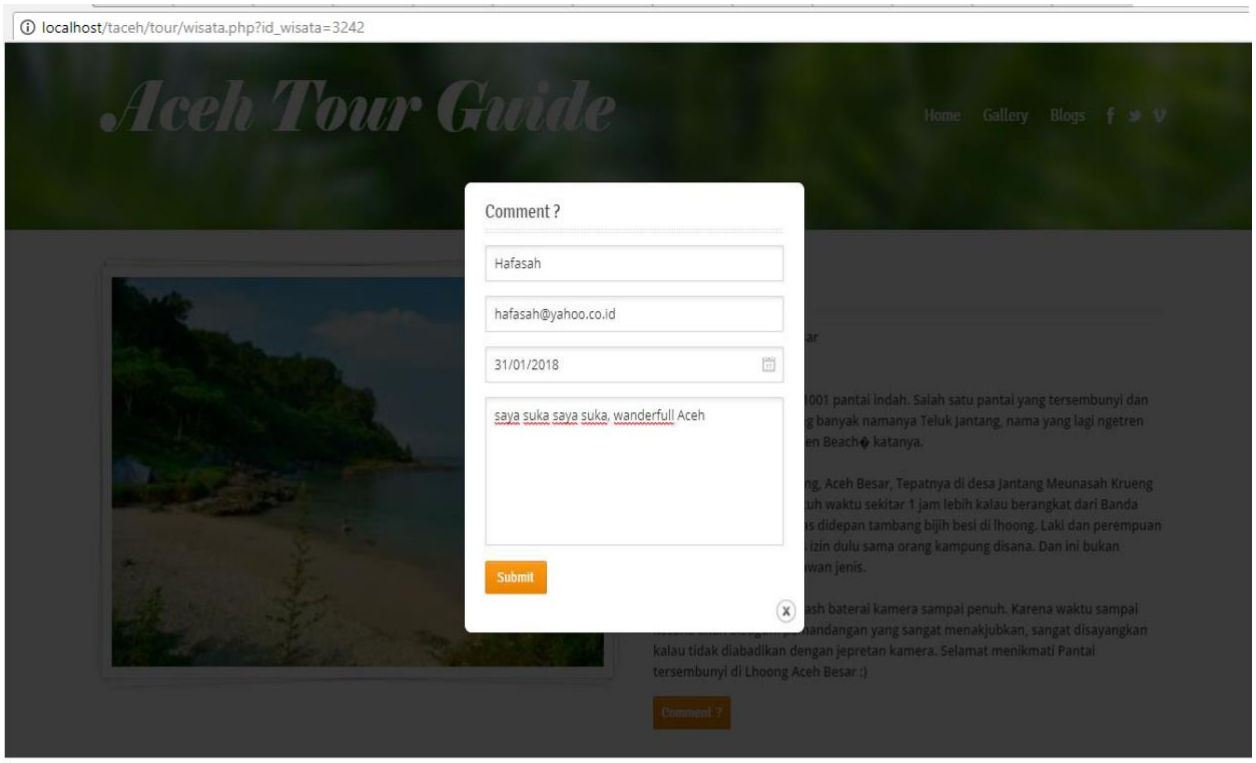

```php
<?php
    $m = new MongoClient();
    $db = $m->tour;
    $collection = $db->galeri;
    $cursor = $collection->find()
    $upload_dir = "Admin/Uploads/";

        $arr = array($_FILES["nama"], $_FILES["destinasi"], $_FILES["durasi"], $_POST['ket1'], $_POST['ket2'],$_FILES['
            foreach ($arr as &$value) {
                if ($value["error"] > 0){
                }else{
                if (file_exists($upload_dir . $value["nama"])){
                  } else {
                  move_uploaded_file($value["tmp_name"], $upload_dir. $value["nama"]);
                    $successful = 1;
                    $url = "/uploads/" . $value["name"];
                    $unique_id = "content".rand();
                    $obj = array( "url" => $url, "unique_id" => $unique_id, "height" => $height, "width" => $width );
                    $collection->insert($obj);
                  }
                }
        }
        unset($value);
         }else{
            }
?>
```

3. Input data Kategori



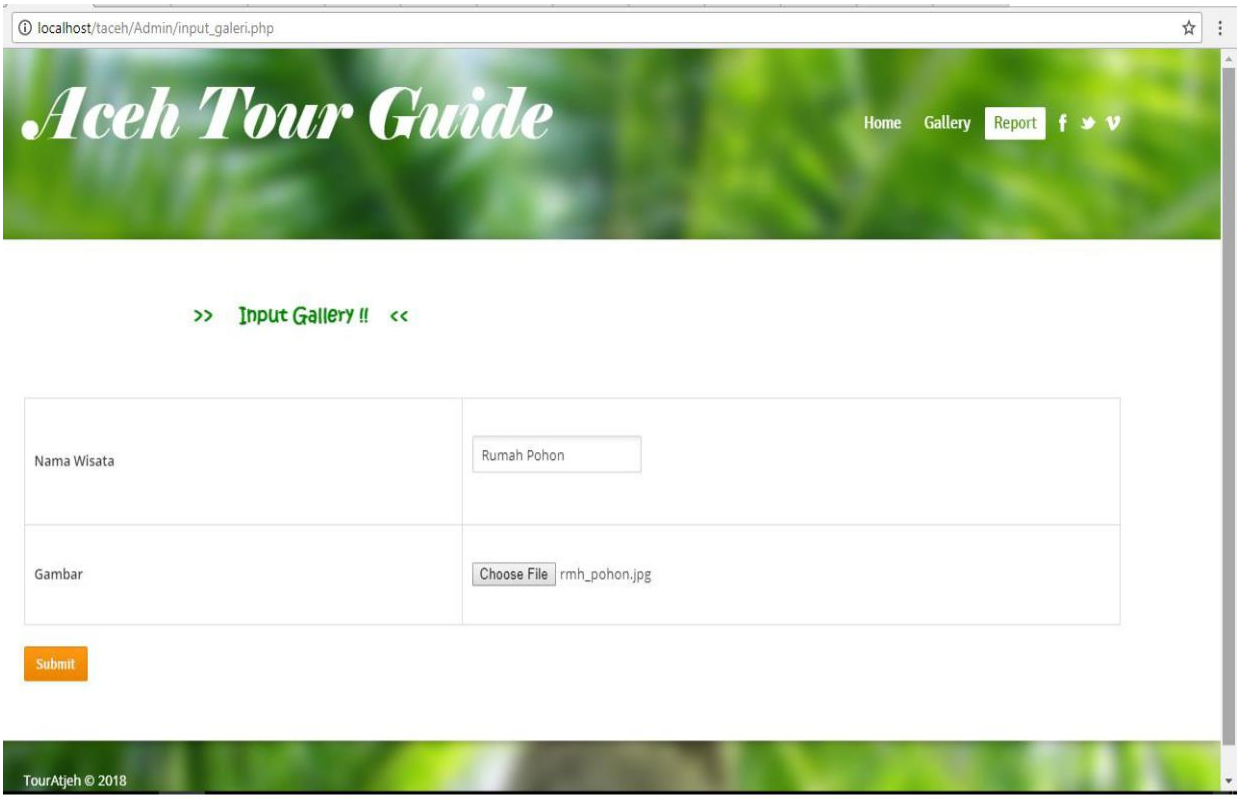

```php
<?php
// connect to mongodb
$m = new MongoClient();
echo "Connection to database successfully";

// select a database
$db = $m->tour;
echo "Database mydb selected";
$collection = $db->kategori;
echo "Collection selected succsessfully";

$id = $_POST['id_kategori'];
$kategori = $_POST['kategori'];

$document = array(
   "id_kategori" => "$id",
   "kategori" => "$kategori",
);

$collection->insert($document);
echo "Document inserted successfully";
header("location:report4.php");
?>
```

4. Input data Komentar




```php
<?php
// connect to mongodb
$m = new MongoClient();
echo "Connection to database successfully";

// select a database
$db = $m->tour;
echo "Database mydb selected";
$collection = $db->komentar;
echo "Collection selected succsessfully";

$id= $_POST['id_komentar'];
$nama = $_POST['nama'];
$email = $_POST['email'];
$komentar = $_POST['komentar'];
$date = $_POST['date'];

$document = array(
    "id_komentar" => "$id",
   "komentar" => "$komentar",
   "tanggal" => "$date",
   "nama" => "$nama",
   "email" => "$email"
);

$collection->insert($document);
?>
```

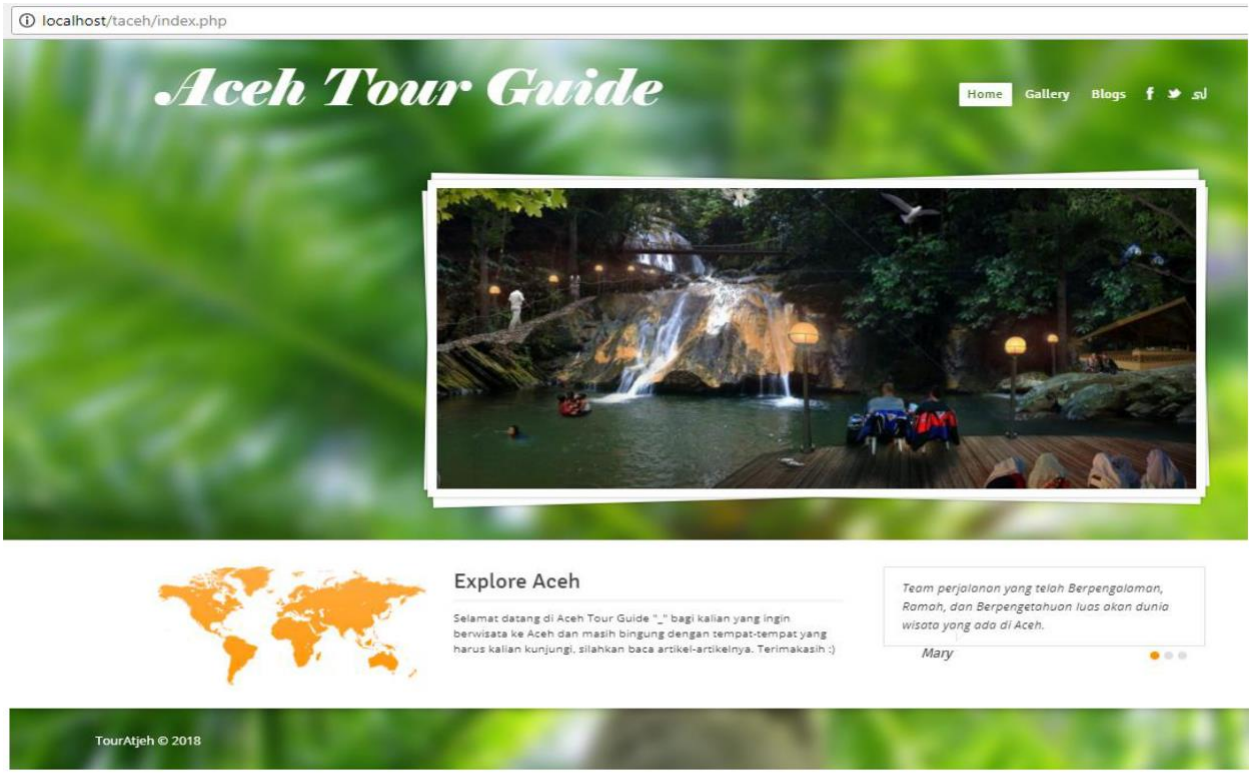
5. Input data attachment




```php
<?php

    $m = new MongoClient();
    $db = $m->tour;
    $collection = $db->galeri; //Change to your collection
    $cursor = $collection->find();

        if(isset($_POST['submit'])){
            $gambar = $_FILES['foto']["name"];
            $gamabarSize = $_FILES ["foto"]["size"];
            $move  = move_uploaded_file($_FILES["foto"]["temp_name"],'Uploads/.$gambar');

            $grid = $db->getGridFS();
            $path = "C:/xampp/htdocs/Taceh/Admin/Uploads/";

            $storefile = $grid->storeFile($path.$gambar, array("metadata" => array("nama"=>$gambar),"nama" => $gambar)
            ?>
            <script language = "javascript">
            alert(" Data Berhasil Ditambah");
            document.location='report3.php'
            </script><?php
        }
?>
```

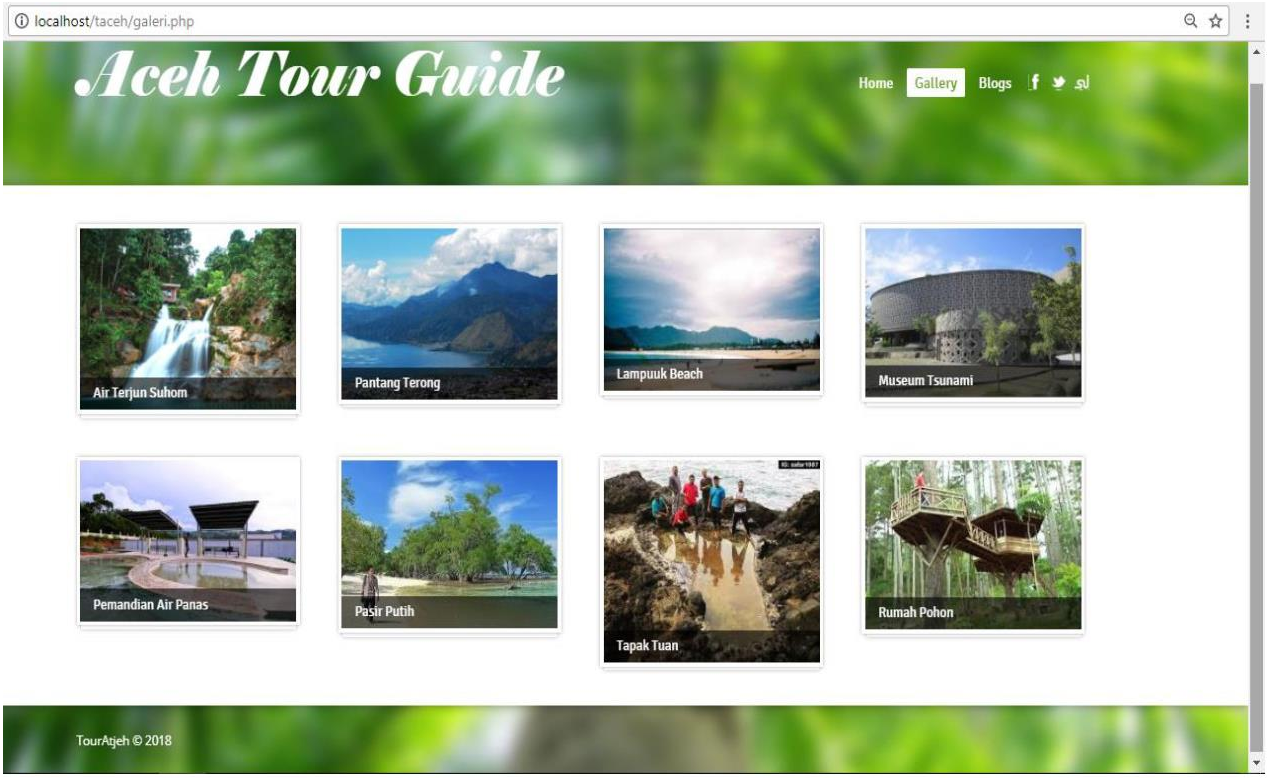
C. Pembuatan portal berita yang di ambil dari data collection mongoDB menggunakan PHP

1. Halaman utama



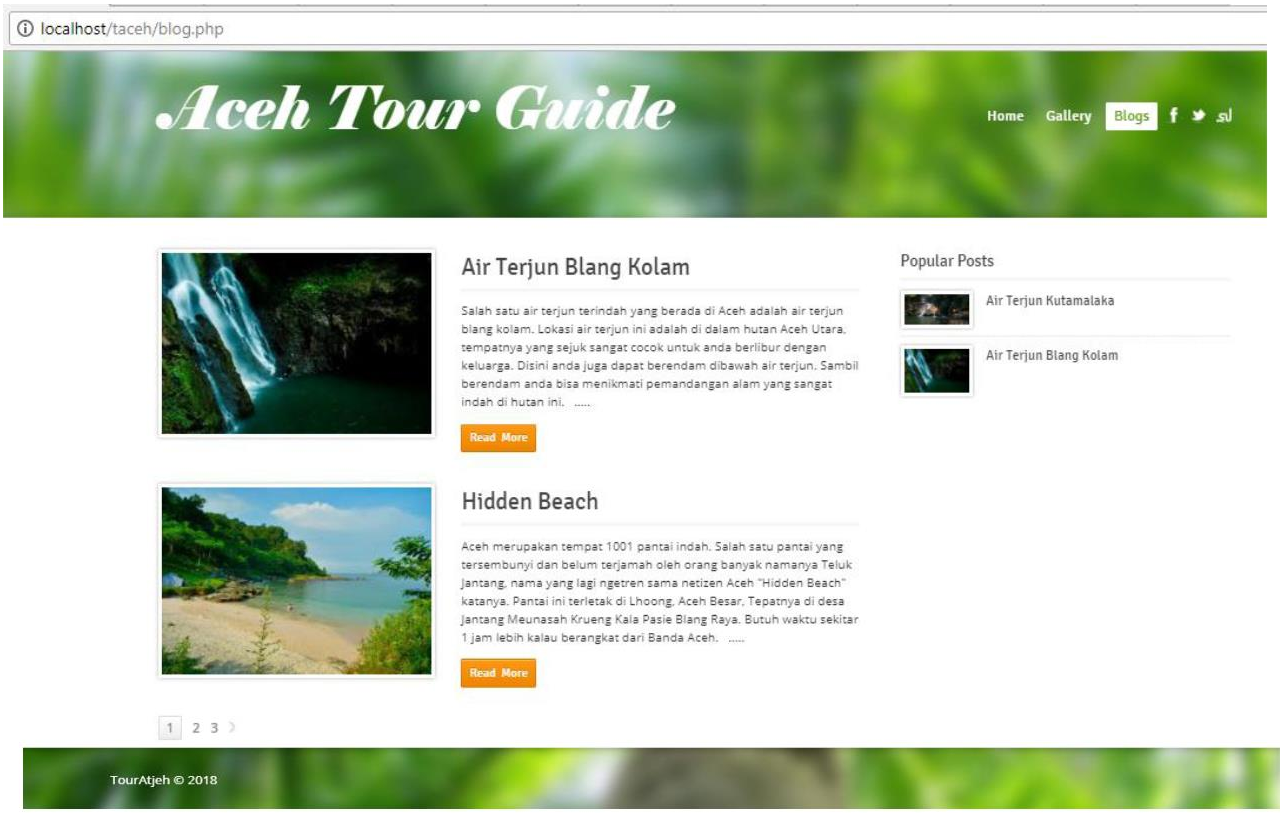

```
    <?php
$collection = $db->wisata;
$cursor = $collection->find()->limit(1);
foreach ($cursor as $val) {

    ?>
        <li>
            <div class="featured-image">
            <a href="<?php echo "tour/wisata.php?id_wisata=" .$val["id_wisata"] ; ?>">
                <img width="824" height="480" src="Admin/Uploads/<?php echo $val["gambar"] ?>" class="attachment-la
            </a>
            </div>
        </li>

    <?php }?>
```

2. Halaman Gallery




```
<?php
$collection = $db->galeri;
$cursor = $collection->find()->limit(8);

foreach ($cursor as $val) {
}
?>
</div><div class="column gallery-item threecol last">
<div class="featured-image">
<a href= "<?php echo "tour/wisata.php?id_wisata=" .$val["id_galeri"] ?>"  class="colorbox " data-group="gallery-107" t
    <img width="440" height="330" src="Admin/Uploads/<?php echo $val["foto"] ?>" class="attachment-preview wp-post-imag
    <a class="featured-image-caption visible-caption" href="#">
    <h6><?php echo $val["nama"] ?></h6>
</a>
</div>
<div class="block-background"></div>
</div><div class="clear"></div> <div class="clear"></div>
</div>
```

3. Halaman Blog




```
87          <?php
88          $collection = $db->wisata;
89          $cursor = $collection->find()->skip(1);
90          foreach ($cursor as $val) {
91          ?>
92          <article class="post-112 post type-post status-publish format-standard hentry category-guides tag-amet tag-dol
93      <div class="column fivecol post-featured-image">
94          <div class="featured-image">
95          <a href="<?php echo "tour/wisata.php?id_wisata=" . $val ["id_wisata"]; ?>"><img width="440" height="330" src="
96      </div>
97      <div class="column sevencol last">
98              <div class="post-content">
99              <div class="section-title">
100                 <h1><a href="<?php echo "tour/wisata.php?id_wisata=" . $val ["id_wisata"]; ?>" ><?php echo $val["nama_
101             </div>
102             <p><?php echo $val["ket1"] ?>   .....</p>
103             <a href="<?php echo "tour/wisata.php?id_wisata=" . $val ["id_wisata"]; ?>" class="button small"><span>Read
104             </footer>
105         </div>
106     </div>
107 </article>  <?php } ?>
108     </div>
```

4. Halaman Read More

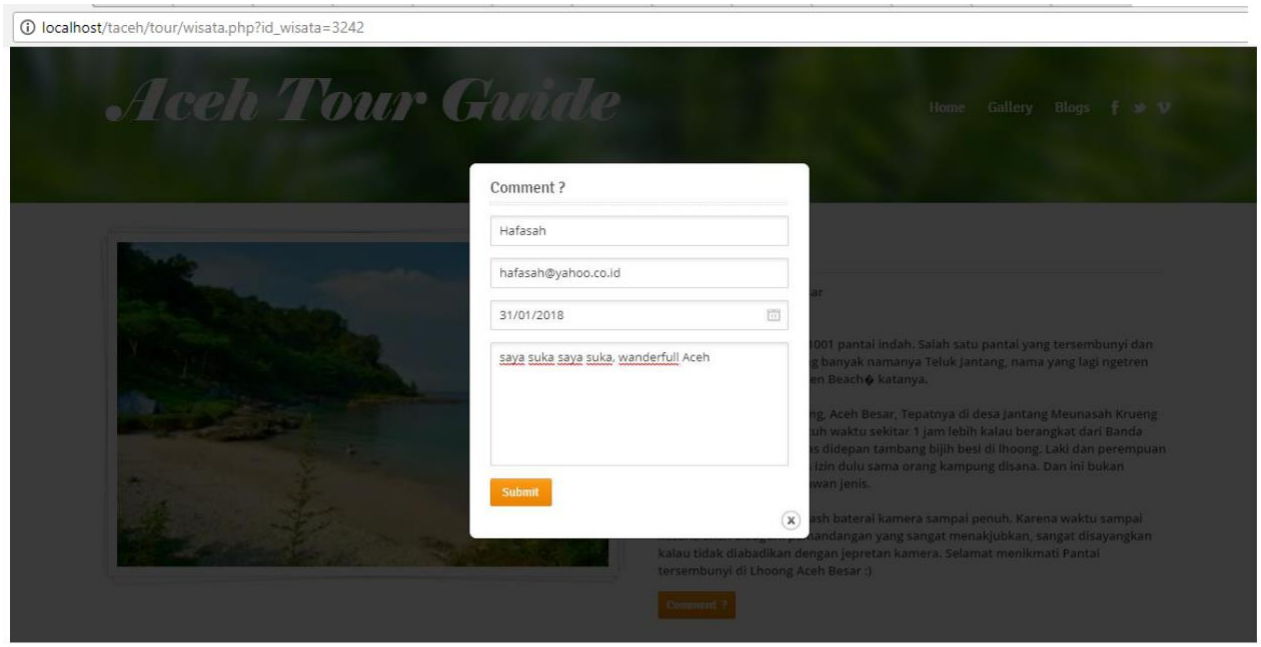
```php
<?php
 error_reporting(E_ALL ^ E_DEPRECATED);
    include "../koneksi.php";
    $collection = $db->wisata;
    $cursor = $collection->find(array("id_wisata"=> ["id_wisata"]));
    foreach ($cursor as $val) {}
    ?>
```

```php
        <ul>
        <?php
            $collection = $db->wisata;
            $cursor = $collection->findOne(array('gambar'=> ["gambar"]),array ('id_wisata'));
            foreach ($cursor as $val) {
        ?>
            <li><input type="hidden" id="id_wisata" name="id_wisata" value= "<?php echo $val["id_wisata"] ?>" />
            <img src="../Admin/Uploads/<?php echo $val["gambar"] ?>" alt="" /></li>
        <?php } ?>
        </ul>
            <div class="arrow arrow-left content-slider-arrow"></div>
            <div class="arrow arrow-right content-slider-arrow"></div>
            <input type="hidden" class="slider-speed" value="400" />
            <input type="hidden" class="slider-pause" value="0" />
        </div>
            <div class="block-background layer-1"></div>
            <div class="block-background layer-2"></div>
        </div>
        </div>
        <div class="sixcol column last">
        <div class="section-title">
        <?php
            $collection = $db->wisata;
            $cursor = $collection->findOne(array('nama_wisata'=> ["nama_wisata"]),array ('id_wisata'));
            foreach ($cursor as $val) {
        ?>
                    <h1><?php echo $val["nama_wisata"] ?></h1>
        <?php } ?>
            </div>
```

```
119         <?php
120             $collection = $db->wisata;
121             $cursor = $collection->findOne(array('nama_wisata'=> ["nama_wisata"]),array ('id_wisata'));
122             foreach ($cursor as $val) {
123         ?>
124             <h1><?php echo $val["nama_wisata"] ?></h1>
125             <?php } ?>
126                 </div>
127         <ul class="tour-meta">
128         <?php
129             $collection = $db->wisata;
130             $cursor = $collection->findOne(array('destinasi'=> ["destinasi"]),array ('id_wisata'));
131             foreach ($cursor as $val) {
132         ?>
133             <li><div class="colored-icon icon-2"></div><strong>Destination:   <?php echo $val["destinasi"] ?> </a>
134             <?php } ?>
135             <?php
136             $collection = $db->wisata;
137             $cursor = $collection->findOne(array('durasi'=> ["durasi"]),array ('id_wisata'));
138             foreach ($cursor as $val) {
139         ?>
140             <li><div class="colored-icon icon-1"><span></span></div><strong>Duration:</strong>   <?php echo $val["
141             <?php } ?>
142                 </ul>
143                 <br>
144             <?php
145             $collection = $db->wisata;
146             $cursor = $collection->findOne(array('ket1'=> ["ket1"]),array ('id_wisata'));
147             foreach ($cursor as $val) {
148         ?>
149             <p><?php echo $val["ket1"] ?></p>
150             <?php } ?>
```

```
152             <?php
153             $collection = $db->wisata;
154             $cursor = $collection->findOne(array('ket2'=> ["ket2"]),array ('id_wisata'));
155             foreach ($cursor as $val) {
156         ?>
157             <p><?php echo $val["ket2"] ?></p>
158             <?php } ?><br><br>
159                     <a href="../komentar.php"><input type="submit" name="button" id="button" value="Your Comment" class
160
```

5. Halaman Komentar



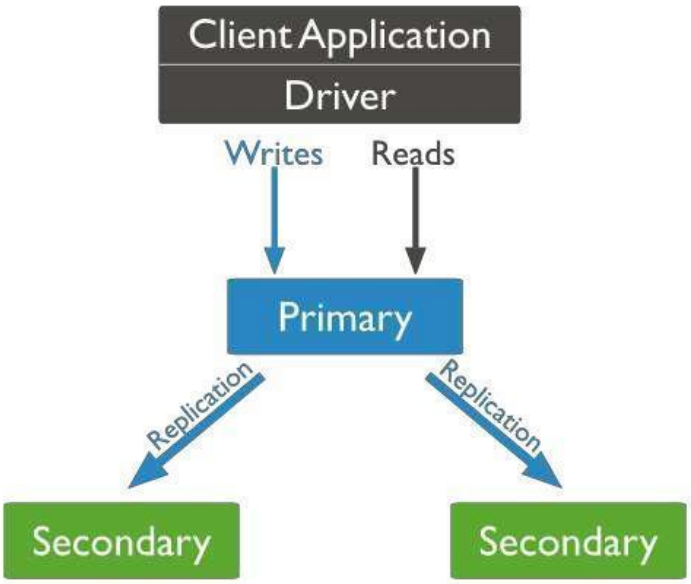

```
    <!-- Commant form -->
    <div class="section-title popup-title"><h4></h4></div>
    <form action="aksi_komentar.php" method="POST" class="formatted-form ajax-form">
        <p></p><div class="message"></div><div class="sixcol column "><div class="field-container">
        <input type="text" id="nama" name="nama" value="" placeholder="Full Name"  /></div></div><div class="sixcol col
        <input type="text" id="email" name="email" value="" placeholder="Email Address"  /></div></div><div class="sixc
        <input type="text" id="date" name="date" value="" class="date-field" placeholder="Date"  /></div></div><div cla
        <textarea id="question" name="question" placeholder="Your Comment" ></textarea></div></div>
        <div class="sixcol column last"><div class="clear"></div><div class="field-container"></div></div>
        <input type="hidden" name="id" value="" class="popup-id" />
        <input type="hidden" name="slug" value="commant" />
        <input type="hidden" class="action" value="themex_form_submit" />
        <a class="submit-button button" href="#">Submit</a>
    </form>
</div>
```

D. Replication dan sharding data pada mongoDB a. Replication

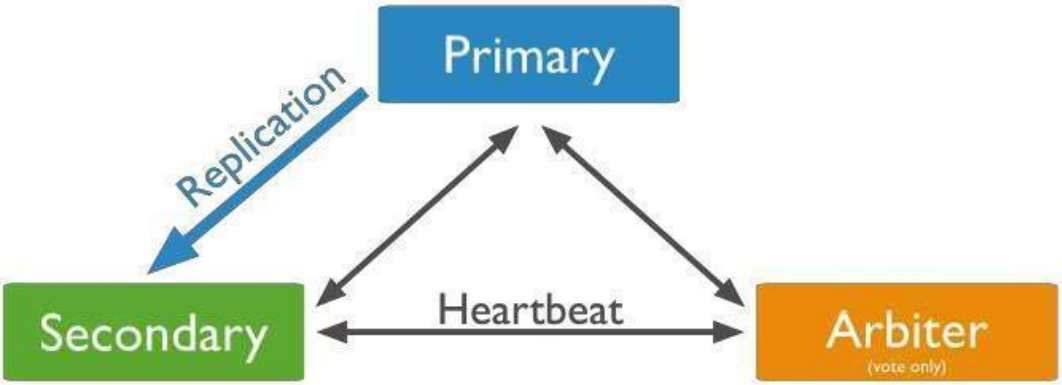
Set replika adalah sekelompok contoh mongod yang menampung kumpulan data yang sama. Satu mongod, primer, menerima semua operasi tulis. Semua contoh lainnya, sekunder, menerapkan operasi dari yang utama sehingga mereka memiliki kumpulan data yang sama.

Replicate utama menerima semua operasi penulisan dari klien. Set replika hanya bisa memiliki satu primer. Untuk mendukung replikasi, catatan utama semua perubahan pada set datanya di oplog-nya. Untuk informasi lebih lanjut tentang operasi simpul primer, lihat Replica Set Primary.

Diagram default routing membaca dan menulis ke primer.



Replicate yang kedua yang primary utama dan menerapkan operasi ke kumpulan data sehingga kumpulan data orang tua mencerminkan kumpulan data primer. Jika primary tidak tersedia, set replika akan memilih secondary untuk menjadi yang utama. Berikut adalah diagram set replika anggota 3 yang terdiri dari primary dan two secondaryaries.



Prosedure dengan memakai portal berita

Prosedur berikut menguraikan langkah-langkah untuk menerapkan set replika saat kontrol akses dinonaktifkan.

1) Mulailah setiap anggota replika set dengan pilihan yang sesuai.

Untuk setiap anggota, mulailah mongod dan tentukan nama replika setel melalui pilihan replset. Tentukan parameter lain yang spesifik untuk penempatan Anda. Jika aplikasi Anda terhubung ke lebih dari satu set replika, setiap rangkaian harus memiliki nama yang berbeda. Beberapa driver group replika mengatur koneksi dengan nama replika set.

Contoh berikut menentukan nama set replika melalui opsi baris perintah --replSet:

mongod --replSet "rs0"

mongod - dbpath D: \ data --replSet rs0 --port 27017

2) Hubungkan shell mongo ke anggota replika. Misalnya, untuk terhubung ke mongod yang berjalan di localhost di port default 27017, cukup mengeluarkan: mongo

3) Memulai replika set. Gunakan rs.initiate () pada anggota replika:

MongoDB memulai sebuah set yang terdiri dari anggota saat ini dan yang menggunakan konfigurasi default replika set.

4) Verifikasi konfigurasi set replika awal.

Gunakan rs.conf () untuk menampilkan objek konfigurasi replika set: rs.conf ()

Objek konfigurasi replika set menyerupai berikut ini:

{"_id": "rs0", "version": 1, "members": [{"_id": 1, "host": "mongodb0.example.net Catatan7017"}]

5) Tambahkan anggota yang tersisa ke set replika.

Tambahkan anggota yang tersisa dengan metode rs.add (). Contoh berikut menambahkan dua anggota:

rs.add ("localhost: 27017")

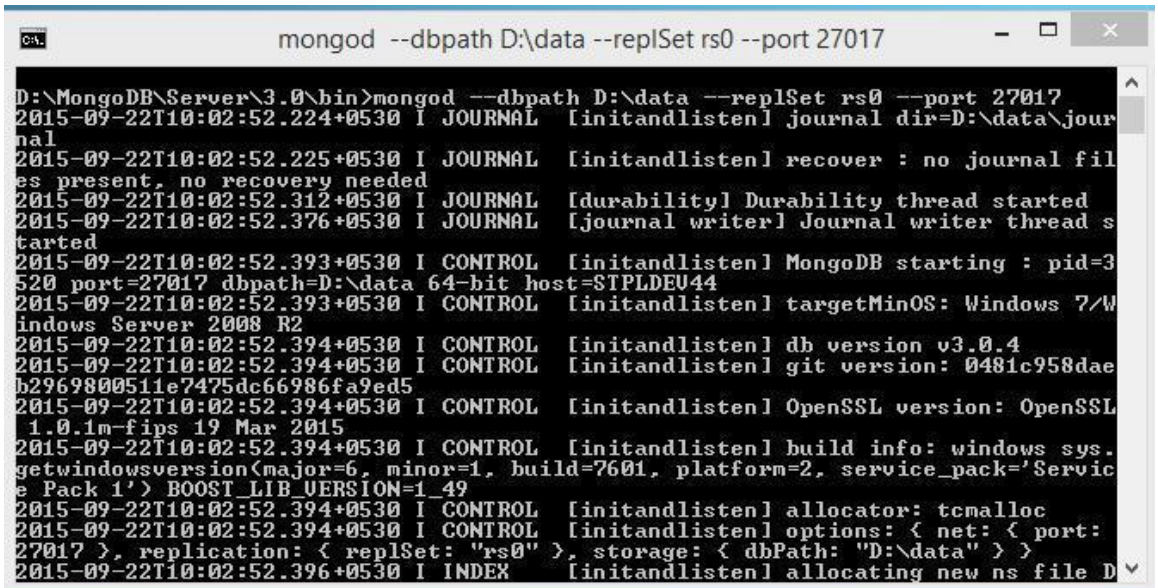
rs.add ("10.0.1.128 perkataan7017")

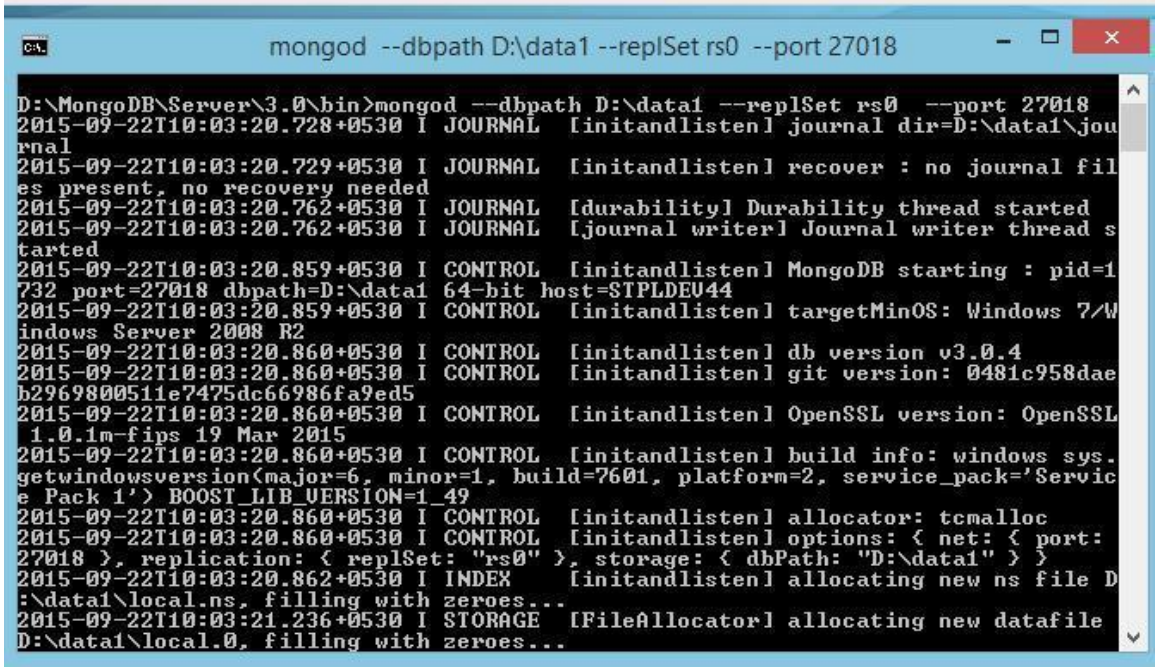
Setelah selesai, maka memiliki replika set lengkap. Set replika baru akan memilih primer.

6) Gunakan operasi rs.status (): rs.status ()

Screen Shoot

1) Pada mesin yang sama mulai dua layanan mongod dengan port yang berbeda





2. Pada mesin lain mulai layanan mongod dengan nomor port yang berbeda

```
D:\MongoDB\Server\3.0\bin>mongod --dbpath D:\data2\ --replSet rs0 --port 27019
2015-09-22T14:59:35.478+0530 I JOURNAL  [initandlisten] journal dir=D:\data2\jou
rnal
2015-09-22T14:59:35.505+0530 I JOURNAL  [initandlisten] recover : no journal fil
es present, no recovery needed
2015-09-22T14:59:35.543+0530 I JOURNAL  [durability] Durability thread started
2015-09-22T14:59:35.693+0530 I CONTROL  [initandlisten] MongoDB starting : pid=6
856 port=27019 dbpath=D:\data2\ 64-bit host=STPLDEV44
2015-09-22T14:59:35.693+0530 I CONTROL  [initandlisten] targetMinOS: Windows 7/W
indows Server 2008 R2
2015-09-22T14:59:35.693+0530 I CONTROL  [initandlisten] db version v3.0.4
2015-09-22T14:59:35.693+0530 I CONTROL  [initandlisten] git version: 0481c958dae
b2969800511e7475dc66986fa9ed5
2015-09-22T14:59:35.693+0530 I CONTROL  [initandlisten] OpenSSL version: OpenSSL
 1.0.1m-fips 19 Mar 2015
2015-09-22T14:59:35.694+0530 I CONTROL  [initandlisten] build info: windows sys.
getwindowsversion(major=6, minor=1, build=7601, platform=2, service_pack='Servic
e Pack 1') BOOST_LIB_VERSION=1_49
2015-09-22T14:59:35.694+0530 I CONTROL  [initandlisten] allocator: tcmalloc
2015-09-22T14:59:35.694+0530 I CONTROL  [initandlisten] options: { net: { port:
27019 }, replication: { replSet: "rs0" }, storage: { dbPath: "D:\data2\" } }
2015-09-22T14:59:35.695+0530 I INDEX    [initandlisten] allocating new ns file D
:\data2\local.ns, filling with zeroes...
2015-09-22T14:59:35.711+0530 I JOURNAL  [journal writer] Journal writer thread s
```

3. Hubungkan ke instance mongod yang sedang berjalan dan tambahkan alamat host dan ip dari anggota replika ke variabel konfigurasi seperti yang ditunjukkan di bawah ini dan lakukan set replika dengan menggunakan metode rs.initiate ().

```
mongo 10.0.1.128:27017

D:\MongoDB\Server\3.0\bin>mongo 10.0.1.128:27017
MongoDB shell version: 3.0.4
connecting to: 10.0.1.128:27017/test
> config = {_id:'rs0', members:[{_id: 0 , host: 'STPLDEV44:27017'},{_id: 1 , hos
t:'STPLDEV44:27018'},{_id: 2 , host:'STPLDEVSQ74:27019'}]}
{
        "_id" : "rs0",
        "members" : [
                {
                        "_id" : 0,
                        "host" : "STPLDEV44:27017"
                },
                {
                        "_id" : 1,
                        "host" : "STPLDEV44:27018"
                },
                {
                        "_id" : 2,
                        "host" : "STPLDEVSQ74:27019"
                }
        ]
}
> rs.initiate(config)
{ "ok" : 1 }
```

4. Periksa status replika set dengan menggunakan perintah rs.status berikut ()

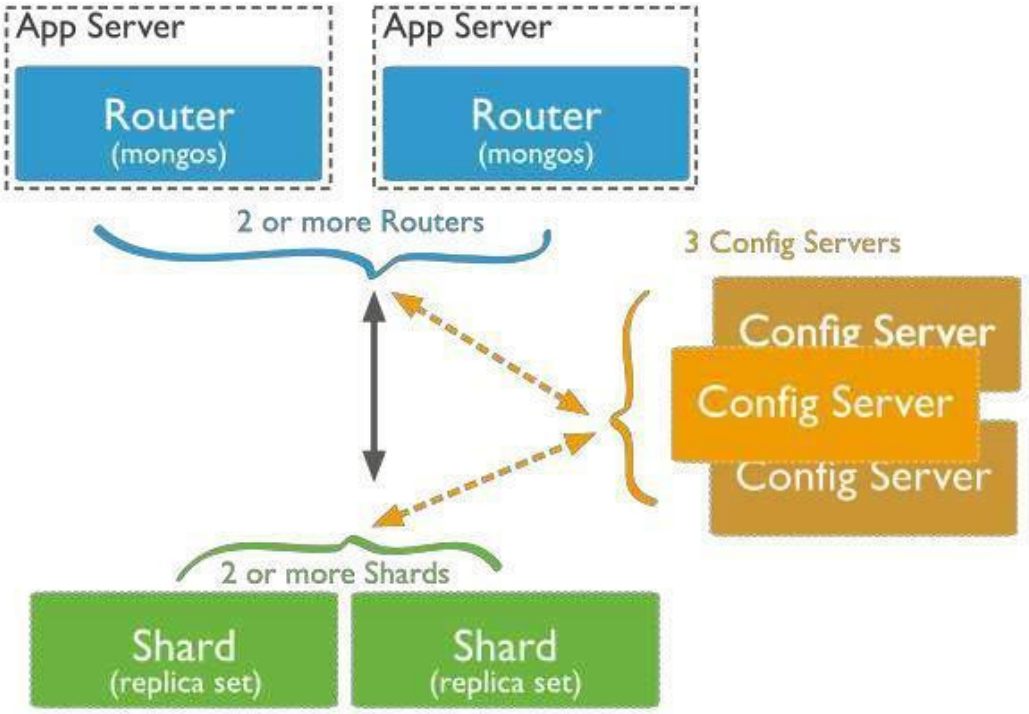
```
rs0:SECONDARY> rs.status()
{
        "set" : "rs0",
        "date" : ISODate("2015-09-22T04:35:39.596Z"),
        "myState" : 1,
        "members" : [
                {
                        "_id" : 0,
                        "name" : "STPLDEV44:27017",
                        "health" : 1,
                        "state" : 1,
                        "stateStr" : "PRIMARY",
                        "uptime" : 167,
                        "optime" : Timestamp(1442896531, 1),
                        "optimeDate" : ISODate("2015-09-22T04:35:31Z"),
                        "electionTime" : Timestamp(1442896533, 1),
                        "electionDate" : ISODate("2015-09-22T04:35:33Z"),
                        "configVersion" : 1,
                        "self" : true
                },
                {
                        "_id" : 1,
                        "name" : "STPLDEV44:27018",
                        "health" : 1,
                        "state" : 2,
                        "stateStr" : "SECONDARY",
                        "uptime" : 8,
                        "optime" : Timestamp(1442896531, 1),
                        "optimeDate" : ISODate("2015-09-22T04:35:31Z"),
                        "lastHeartbeat" : ISODate("2015-09-22T04:35:39.202Z"),
                        "lastHeartbeatRecv" : ISODate("2015-09-22T04:35:39.240Z"
),
                        "pingMs" : 1,
                        "configVersion" : 1
                },
                {
                        "_id" : 2,
                        "name" : "STPLDEVSQ74:27019",
                        "health" : 1,
                        "state" : 2,
                        "stateStr" : "SECONDARY",
                        "uptime" : 8,
                        "optime" : Timestamp(1442896531, 1),
                        "optimeDate" : ISODate("2015-09-22T04:35:31Z"),
                        "lastHeartbeat" : ISODate("2015-09-22T04:35:39.201Z"),
                        "lastHeartbeatRecv" : ISODate("2015-09-22T04:35:39.235Z"
),
                        "pingMs" : 0,
                        "configVersion" : 1
                }
        ],
        "ok" : 1
}
rs0:PRIMARY>
```

Replicate sukses di jalankan

b. Sharding

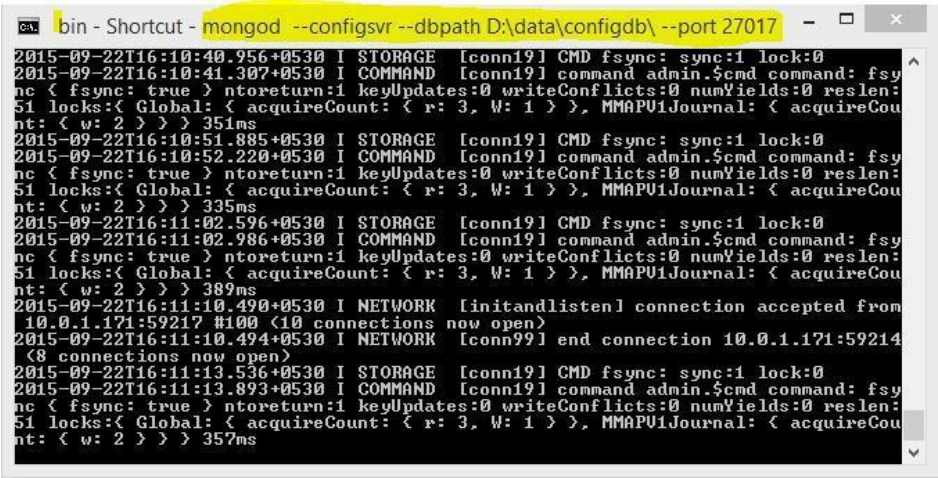
MongoDB mendukung sharding melalui konfigurasi cluster .

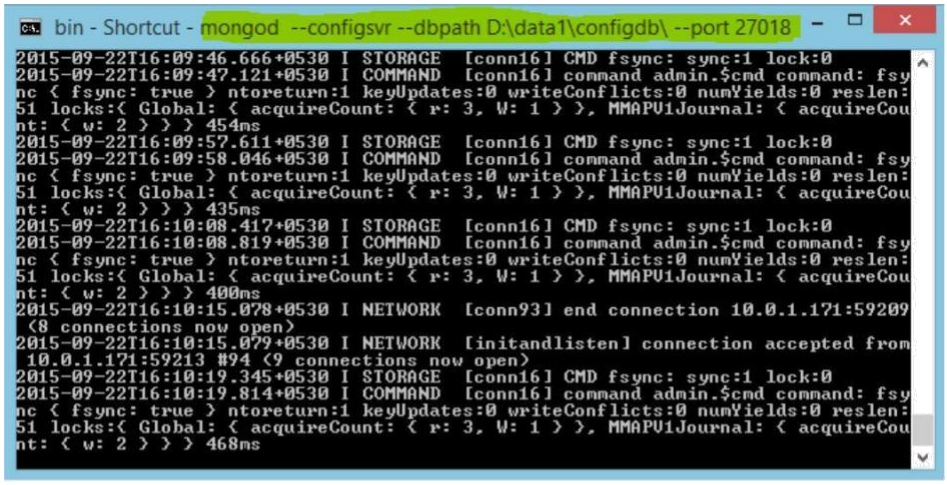
Diagram klaster sharded sampel untuk keperluan produksi. Berisi persis 3 server konfigurasi, 2 atau lebih `` mongos`` query router, dan setidaknya 2 shards. Pecahannya adalah set replika.

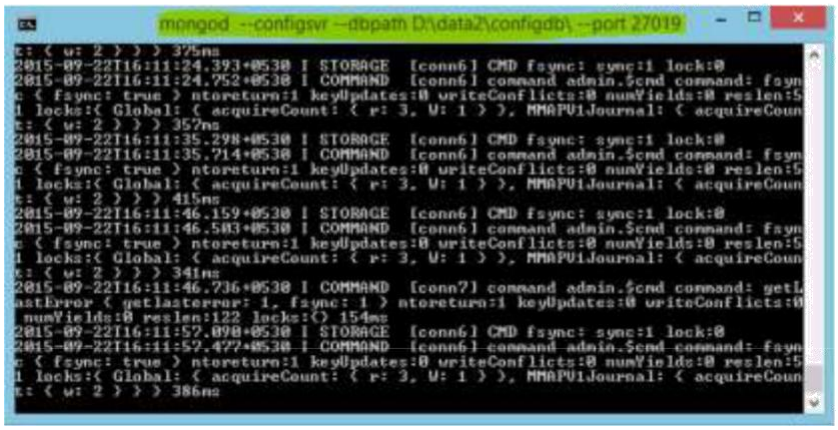


Screen Shoot procedure

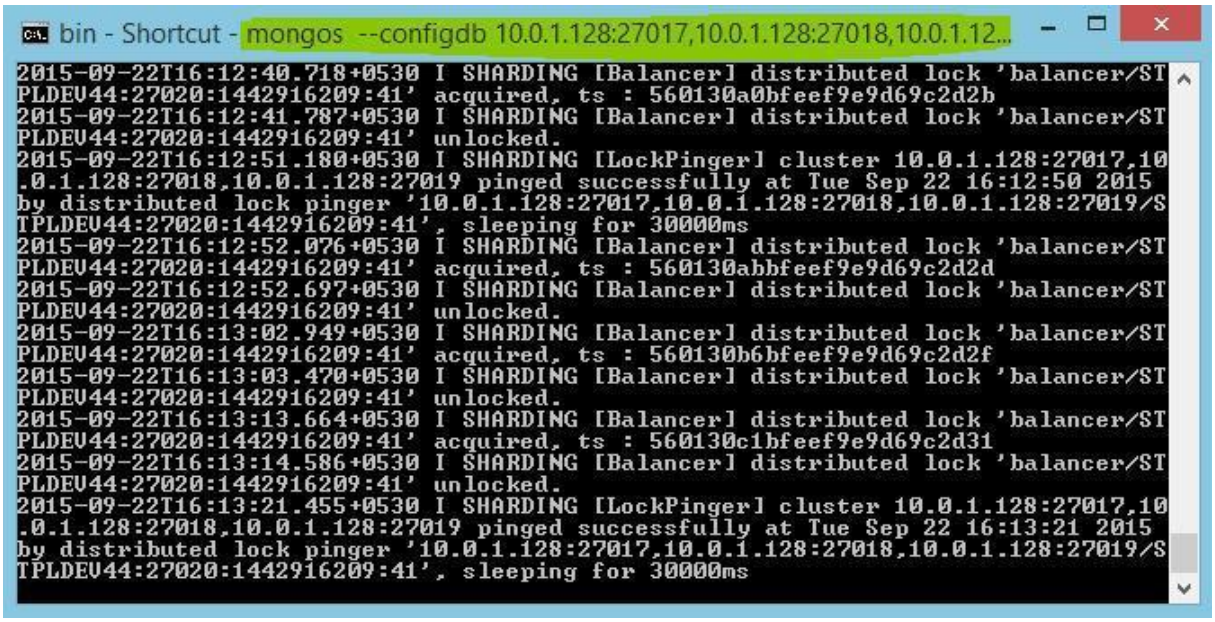
1) jalankan 3 mongodb dengan configsvr option dan ports yang berbeda







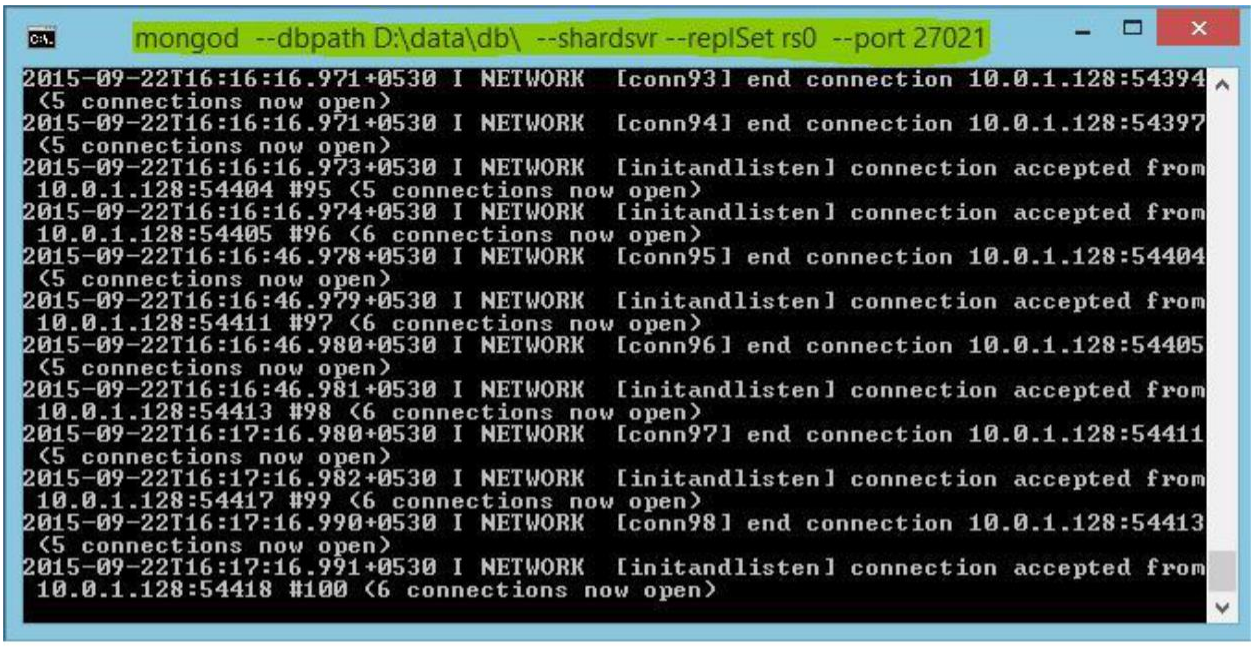
2) Setelah memulai tiga contoh mongod, mereka harus ditambahkan ke pecahan sebagai berikut

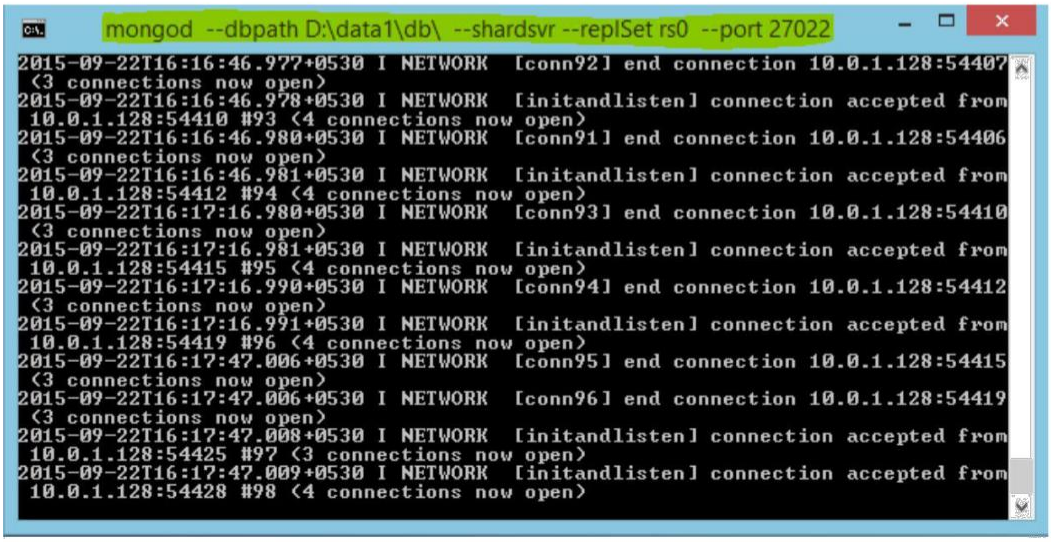


Command yang di gunakan

mongos --configdb 10.0.1.128:27017, 10.0.1.128:27018, 10.0.1.128:27019 --port 27020 Sekarang saatnya menampilkan replika set. Prosedur penampilan dengan layar yang ada di atas

2) Menggunakan perintah dengan -shardsvr opsi mongod dimulai.





```
mongod  --dbpath D:\data2\db\  --shardsvr --replSet rs0  --port 27023
2015-09-22T16:17:47.007+0530 I NETWORK  [conn96] end connection 10.0.1.128:54416
 (3 connections now open)
2015-09-22T16:17:47.010+0530 I NETWORK  [initandlisten] connection accepted from
 10.0.1.128:54429 #98 (4 connections now open)
2015-09-22T16:17:50.972+0530 I NETWORK  [conn97] end connection 10.0.1.128:54420
 (3 connections now open)
2015-09-22T16:17:50.972+0530 I NETWORK  [initandlisten] connection accepted from
 10.0.1.128:54430 #99 (4 connections now open)
2015-09-22T16:18:17.014+0530 I NETWORK  [conn98] end connection 10.0.1.128:54429
 (3 connections now open)
2015-09-22T16:18:17.017+0530 I NETWORK  [initandlisten] connection accepted from
 10.0.1.128:54435 #100 (4 connections now open)
2015-09-22T16:18:20.978+0530 I NETWORK  [conn99] end connection 10.0.1.128:54430
 (3 connections now open)
2015-09-22T16:18:20.978+0530 I NETWORK  [initandlisten] connection accepted from
 10.0.1.128:54437 #101 (4 connections now open)
2015-09-22T16:18:47.023+0530 I NETWORK  [conn100] end connection 10.0.1.128:5443
5 (3 connections now open)
2015-09-22T16:18:47.029+0530 I NETWORK  [initandlisten] connection accepted from
 10.0.1.128:54441 #102 (4 connections now open)
2015-09-22T16:18:50.983+0530 I NETWORK  [conn101] end connection 10.0.1.128:5443
7 (3 connections now open)
2015-09-22T16:18:50.984+0530 I NETWORK  [initandlisten] connection accepted from
 10.0.1.128:54443 #103 (4 connections now open)
```

```
mongo  10.0.1.128:27021
D:\MongoDB\Server\3.0\bin>mongo 10.0.1.128:27021
MongoDB shell version: 3.0.4
connecting to: 10.0.1.128:27021/test
> config = {_id:'rs0', members:[{_id: 0 , host: '10.0.1.128:27021'},{_id: 1 , ho
st:'10.0.1.128:27022'},{_id: 2 , host:'10.0.1.128:27023'}]}
{
        "_id" : "rs0",
        "members" : [
                {
                        "_id" : 0,
                        "host" : "10.0.1.128:27021"
                },
                {
                        "_id" : 1,
                        "host" : "10.0.1.128:27022"
                },
                {
                        "_id" : 2,
                        "host" : "10.0.1.128:27023"
                }
        ]
}
> rs.initiate(config)
{ "ok" : 1 }
```

Konfigurasi untuk menambahkan contoh mongod didefinisikan ke variabel yang disebut config dan replika set dikerahkan dengan perintah rs.initiate (config). Periksa rs.status () untuk status set replika.

```
rs0:OTHER> rs.status()
{
        "set" : "rs0",
        "date" : ISODate("2015-09-22T10:24:51.964Z"),
        "myState" : 1,
        "members" : [
                {
                        "_id" : 0,
                        "name" : "10.0.1.128:27021",
                        "health" : 1,
                        "state" : 1,
                        "stateStr" : "PRIMARY",
                        "uptime" : 201,
                        "optime" : Timestamp(1442917486, 1),
                        "optimeDate" : ISODate("2015-09-22T10:24:46Z"),
                        "electionTime" : Timestamp(1442917490, 1),
                        "electionDate" : ISODate("2015-09-22T10:24:50Z"),
                        "configVersion" : 1,
                        "self" : true
                },
                {
                        "_id" : 1,
                        "name" : "10.0.1.128:27022",
                        "health" : 1,
                        "state" : 2,
                        "stateStr" : "SECONDARY",
                        "uptime" : 5,
                        "optime" : Timestamp(1442917486, 1),
                        "optimeDate" : ISODate("2015-09-22T10:24:46Z"),
                        "lastHeartbeat" : ISODate("2015-09-22T10:24:50.642Z"),
                        "lastHeartbeatRecv" : ISODate("2015-09-22T10:24:50.677Z"
),
                        "pingMs" : 0,
                        "configVersion" : 1
                },
                {
                        "_id" : 2,
                        "name" : "10.0.1.128:27023",
                        "health" : 1,
                        "state" : 2,
                        "stateStr" : "SECONDARY",
                        "uptime" : 5,
                        "optime" : Timestamp(1442917486, 1),
                        "optimeDate" : ISODate("2015-09-22T10:24:46Z"),
                        "lastHeartbeat" : ISODate("2015-09-22T10:24:50.649Z"),
                        "lastHeartbeatRecv" : ISODate("2015-09-22T10:24:50.666Z"
),
                        "pingMs" : 0,
                        "configVersion" : 1
                }
        ],
        "ok" : 1
}
rs0:PRIMARY>
```

```
mongo 10.0.1.128:27020

D:\MongoDB\Server\3.0\bin>mongo 10.0.1.128:27020
MongoDB shell version: 3.0.4
connecting to: 10.0.1.128:27020/test
mongos> sh.status()
--- Sharding Status ---
  sharding version: {
        "_id" : 1,
        "minCompatibleVersion" : 5,
        "currentVersion" : 6,
        "clusterId" : ObjectId("56012774bfeef9e9d69c2b79")
}
  shards:
  balancer:
        Currently enabled:  yes
        Currently running:  no
        Failed balancer rounds in last 5 attempts:  0
        Migration Results for the last 24 hours:
                No recent migrations
  databases:
        {  "_id" : "admin",  "partitioned" : false,  "primary" : "config" }
```

Untuk mengubah contoh mongod ke variabel yang disebut config dan replika set dikerahkan dengan perintah rs.initiate (config). Periksa rs.status () untuk status set replika.

Sekarang mongo terhubung ke server shard yang berjalan pada port 27020 dan dicek untuk sh.status () untuk status shard.

Dengan sh.addShard () perintah replika set rs0 ditambahkan ke shard seperti yang ditunjukkan pada gambar di bawah ini

```
mongos> sh.addShard("rs0/10.0.1.128:27021")
{ "shardAdded" : "rs0", "ok" : 1 }
```

Jadi semua anggota replika set rs0 ditambahkan ke dalam pecahan. Demikian pula menambahkan beberapa set replika untuk mulai melakukan sharding ke beberapa server. Sekarang Anda dapat mengaktifkan sharding ke database dan koleksi apapun dengan menggunakan perintah berikut

1) sh.enableSharding (database)

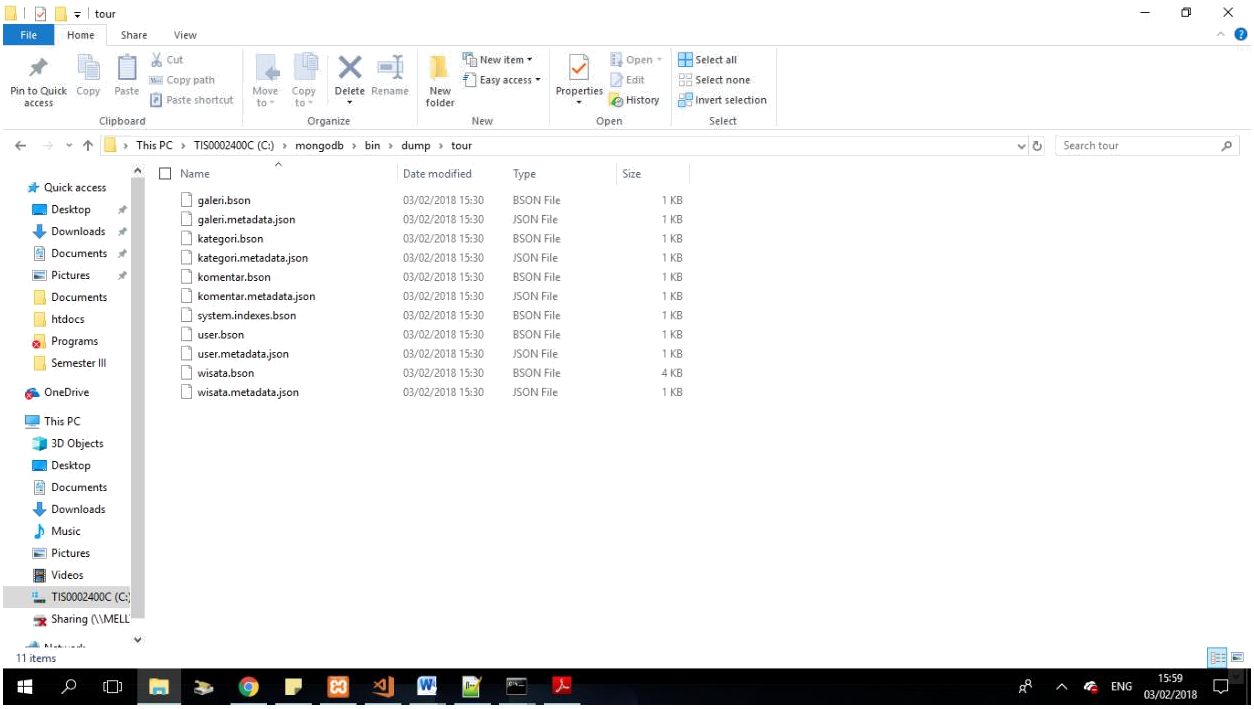
2) sh.shardCollection ("nama koleksi", {key: 1})

E. Backup Data

Kami disini untuk backup data menggunakan mongodump dengan perintah :

Ketikkan pada cmd : C:\Program Files\MongoDB\Server\3.4\bin>mongodump –db tour – host=127.0.0.1 –port=27017 –out=C:\mongodb\bin\dump\tour

```
C:\Program Files\MongoDB\Server\3.4\bin>mongodump --db tour --host=127.0.0.1 --port=27017 --out=C:\mongodb\bin\dump\tour
2018-02-03T15:41:39.915+0700    writing tour.system.indexes to
2018-02-03T15:41:39.931+0700    done dumping tour.system.indexes (5 documents)
2018-02-03T15:41:39.931+0700    writing tour.galeri to
2018-02-03T15:41:39.931+0700    writing tour.komentar to
2018-02-03T15:41:39.931+0700    writing tour.user to
2018-02-03T15:41:39.931+0700    writing tour.kategori to
2018-02-03T15:41:39.946+0700    done dumping tour.galeri (8 documents)
2018-02-03T15:41:39.947+0700    writing tour.wisata to
2018-02-03T15:41:39.947+0700    done dumping tour.komentar (6 documents)
2018-02-03T15:41:39.951+0700    done dumping tour.kategori (4 documents)
2018-02-03T15:41:39.951+0700    done dumping tour.wisata (3 documents)
2018-02-03T15:41:39.965+0700    done dumping tour.user (5 documents)
```


tour
File
Home
Share
View
Pin to Quick access
Copy
Paste
Cut
Copy path
Paste shortcut
Clipboard
Move to
Copy to
Delete
Rename
Organize
New folder
New item
Easy access
New
Properties
Open
Edit
History
Select all
Select none
Invert selection
Select
This PC › TIS0002400C (C:) › mongodb › bin › dump › tour
Search tour
Quick access
Desktop
Downloads
Documents
Pictures
Documents
htdocs
Programs
Semester III
OneDrive
This PC
3D Objects
Desktop
Documents
Downloads
Music
Pictures
Videos
TIS0002400C (C:)
Sharing (\\MELL
Name
Date modified
Type
Size
galeri.bson 03/02/2018 15:30 BSON File 1 KB
galeri.metadata.json 03/02/2018 15:30 JSON File 1 KB
kategori.bson 03/02/2018 15:30 BSON File 1 KB
kategori.metadata.json 03/02/2018 15:30 JSON File 1 KB
komentar.bson 03/02/2018 15:30 BSON File 1 KB
komentar.metadata.json 03/02/2018 15:30 JSON File 1 KB
system.indexes.bson 03/02/2018 15:30 BSON File 1 KB
user.bson 03/02/2018 15:30 BSON File 1 KB
user.metadata.json 03/02/2018 15:30 JSON File 1 KB
wisata.bson 03/02/2018 15:30 BSON File 4 KB
wisata.metadata.json 03/02/2018 15:30 JSON File 1 KB
11 items
ENG
15:59
03/02/2018