BAB V

IMPLEMENTASI DAN PENGUJIAN SISTEM

Pada tahap ini akan diuraikan mengenai implementasi dari aplikasi yang dibuat. Implementasi dari perancangan aplikasi ini menggunakan bahasa pemrograman Java untuk logic android, XML untuk layout, MYSQL untuk database, PHP untuk komunikasi android ke MYSQL, dan Eclipse sebagai editor utama.

5.1. Implementasi

Implementasi adalah tahap penerapan sistem yang akan dilakukan jika sistem disetujui termasuk program yang telah dibuat pada tahap perancangan sistem agar siap untuk dioperasikan. Implementasi Aplikasi Pengelolaan Konten Informasi Hasil Penelitian Matahari dan Antariksa dilakukan dengan menggunakan bahasa pemrograman Java untuk logic android, XML untuk layout android, MYSQL untuk database, PHP untuk komunikasi android ke MYSQL, dan Eclipse sebagai editor utama. Aplikasi tersebut dapat dijalankan pada platform android saja, dengan implementasi dan pengujian yang dilakukan pada Galaxy Tab P3100 (Galaxy Tab 2 7.0 inci) dengan sistem operasi Android. Aplikasi ini dapat berjalan dengan baik, apabila memenuhi standar minimal dari perangkat keras *(hardware).*

5.1.1. Batasan Implementasi (Optional)

Dalam mengimplementasikan aplikasi ini ada beberapa hal yang menjadi batasan implementasi, yaitu :

1. Proses menuliskan kode program dengan menggunakan Eclipse Helios (eclipse versi 3.6) sebagai editor utama dengan menggunakan bahasa pemrograman java untuk logic androidnya.
2. Basis data yang digunakan dalam mengimplementasikan aplikasi tersebut adalah MySQL.
3. Implementasi aplikasi ini dilakukan pada Tablet Android dengan resolusi minimum 3.3" WQVGA (240 x 400: ldpi).

5.1.2. Implementasi Perangkat Lunak

Untuk pengembangan perangkat lunak ini menggunakan Eclipse Helios (eclipse versi 3.6), PHP, dan MYSQL. Eclipse Helios dipilih sebagai perangkat lunak untuk *android* yang disertai bahasa pemrograman java untuk logic androidnya, PHP dipilih sebagai komunikasi android ke MYSQL dengan format data JSON, dan MYSQL digunakan sebagai perangkat lunak dalam pembuatan basis data (*database*).

5.1.3. Implementasi Perangkat Keras

Perangkat keras yang digunakan berdasarkan kebutuhan minimal yang harus dipenuhi dalam membangun sebuah aplikasi tersebut antara lain :

1. Personal Komputer
    a. Processor Genuine Intel(R) CPU U7300 @1.30GHz
    b. Memory RAM 2.00 GB
    c. Harddisk 320 GB
    d. Mouse dan keyboard
    e. Modem ESIA AHA

2. Tablet
    a. Processor Dual core 1GHz, PowerVR SGX540 GPU, TI OMAP 4430 chipset
    b. Memory 8/16/32 GB storage, 1GB RAM, eksternal MicroSD hingga 64GB
    c. Layar Sentuh Kapasitif PLS LCD, 600 x 1024 piksel, 7.0 inci (~170 ppi pixel density)
    d. Konektivitas GPRS, EDGE, HSDPA, 21 Mbps, Wi-Fi 802.11, Wi-Fi Direct, dual-band, Wi-Fi hotspot, Bluetoothv3.0 with A2DP, HS, microUSB v3.0 with A2DP

5.1.4. Implementasi Basis Data (Sintaks SQL)

Implementasi basis data dilakukan dengan menggunakan DBMS yang digunakan adalah MySQL. Dan service rss feed dengan format data xml dari web Lapan Bandung. Adapun implementasi basis datanya dan rss feed dengan format data xml adalah sebagai berikut :

a. Implementasi Basis Data Konfigurasi

```
<?php
/*
 * All database connection variables
 */
define('DB_USER', "root"); // db user
```

```
define('DB_PASSWORD', ""); // db password (mention your db password
here)
define('DB_DATABASE', "lapan_matahari"); // database name
define('DB_SERVER', "localhost"); // db server
?>
```

b. Implementasi Basis Data Penyediaan Fungsi untuk membuat Koneksi

```
<?php
/**
 * A class file to connect to database
 */
class DB_CONNECT {
    // constructor
    function __construct() {
        // connecting to database
        $this->connect();
    }
    // destructor
    function __destruct() {
        // closing db connection
        $this->close();
    }
    /**
```

```
     * Function to connect with database
     */
    function connect() {
        // import database connection variables
        require_once 'db_config.php';
        // Connecting to mysql database
        $con = mysql_connect(DB_SERVER, DB_USER,
DB_PASSWORD) or die(mysql_error());
        // Selecing database
        $db = mysql_select_db(DB_DATABASE) or die(mysql_error()) or
die(mysql_error());
        // returing connection cursor
        return $con;
    }
    /**
     * Function to close db connection
     */
    function close() {
        // closing db connection
        mysql_close();
    }
}
```

c. Implementasi Basis Data Get Approval

```
<?php
/*
 * Following code will list all the products
 */
// array for JSON response
$response = array();
// include db connect class
require_once 'db_connect.php';
// connecting to db
$db = new DB_CONNECT();
// get all products from products table
$result = mysql_query("SELECT *FROM matahari where status = 0") or
die(mysql_error());
// check for empty result
if (mysql_num_rows($result) > 0) {
    // looping through all results
    // products node
    $response["matahari"] = array();
    while ($row = mysql_fetch_array($result)) {
        // temp user array
        $product = array();
```

```
        $product["id"] = $row["id"];
        $product["title"] = $row["title"];
        $product["date"] = $row["date"];
                $product["description"] = $row["description"];
                $product["status"] = $row["status"];
                $product["kategori"] = $row["kategori"];
        // push single product into final response array
        array_push($response["matahari"], $product);
    }
    // success
    $response["success"] = 1;
    // echoing JSON response
    echo json_encode($response);
} else {
    // no products found
    $response["success"] = 0;
    $response["message"] = "No products found";
    // echo no users JSON
    echo json_encode($response);
}
?>
```

d. Implementasi Basis Data Get Matahari

```
<?php
/*
 * Following code will list all the products
 */
// array for JSON response
$response = array();
// include db connect class
require_once 'db_connect.php';
// connecting to db
$db = new DB_CONNECT();
// get all products from products table
$result = mysql_query("SELECT *FROM matahari where status=1 and
kategori=1") or die(mysql_error());
// check for empty result
if (mysql_num_rows($result) > 0) {
    // looping through all results
    // products node
    $response["matahari"] = array();
    while ($row = mysql_fetch_array($result)) {
        // temp user array
        $product = array();
```

```
        $product["id"] = $row["id"];
        $product["title"] = $row["title"];
        $product["date"] = $row["date"];
                $product["description"] = $row["description"];
                $product["status"] = $row["status"];
                $product["kategori"] = $row["kategori"];
                $product["source"] = $row["source"];
                $product["path"] = $row["path"];
        // push single product into final response array
        array_push($response["matahari"], $product);
    }
    // success
    $response["success"] = 1;
    // echoing JSON response
    echo json_encode($response);
} else {
    // no products found
    $response["success"] = 0;
    $response["message"] = "No products found";
    // echo no users JSON
    echo json_encode($response);
} ?>
```

e. Implementasi Basis Data Update Approval

```
<?php
/*
 * Following code will create a new product row
 * All product details are read from HTTP Post Request
 */
// array for JSON response
$response = array();
// check for required fields
if (isset($_GET['id'])) {
    $id = $_GET['id'];
      $kat = $_GET['kat'];
    // include db connect class
    require_once 'db_connect.php';
    // connecting to db
    $db = new DB_CONNECT();
    // mysql inserting a new row
      if($kat==1)
            $result = mysql_query("Update matahari set status = 1 where
id='$id'");
      else
```

```
            $result = mysql_query("Update matahari set status = 2 where
id='$id'");
    // check if row inserted or not
    if ($result) {
        // successfully inserted into database
        $response["success"] = 1;
        $response["message"] = "Product successfully created.";
        // echoing JSON response
        echo json_encode($response);
    } else {
        // failed to insert row
        $response["success"] = 0;
        $response["message"] = "Oops! An error occurred.";
        // echoing JSON response
        echo json_encode($response);
    }
} else {
    // required field is missing
    $response["success"] = 0;
    $response["message"] = "Required field(s) is missing";
    // echoing JSON response
    echo json_encode($response); } ?>
```

f. Implementasi Basis Data Update User

```
<?php
/*
 * Following code will create a new product row
 * All product details are read from HTTP Post Request
 */
// array for JSON response
$response = array();
// check for required fields
if (isset($_POST['id'])) {
    $id = $_POST['id'];
      $user = $_POST['username'];
      $pass = md5($_POST['password']);
      $phone = $_POST['phone'];
      $nama = $_POST['nama'];
    // include db connect class
    require_once 'db_connect.php';
    // connecting to db
    $db = new DB_CONNECT();
    // mysql inserting a new row
```

```
        $result = mysql_query("Update user set
username='$user',password='$pass',phone='$phone',nama='$nama' where
id=$id");
    // check if row inserted or not
    if ($result) {
        // successfully inserted into database
        $response["success"] = 1;
        $response["message"] = "Product successfully created.";
        // echoing JSON response
        echo json_encode($response);
    } else {
        // failed to insert row
        $response["success"] = 0;
        $response["message"] = "Oops! An error occurred.";
        // echoing JSON response
        echo json_encode($response);
    }
} else {
    // required field is missing
    $response["success"] = 0;
    $response["message"] = "Required field(s) is missing";
 // echoing JSON response
```

```
    echo json_encode($response);
}
?>
```

g. Implementasi Basis Data Validasi Login

```
<?php
/*
 * Following code will list all the products
 */
// array for JSON response
$response = array();
// include db connect class
require_once 'db_connect.php';
// connecting to db
$db = new DB_CONNECT();
// get all products from products table
$user = $_GET['user'];
$pass = md5($_GET['pass']);
$result = mysql_query("SELECT *FROM user where username = '$user'
and password='$pass'") or die(mysql_error());
// check for empty result
if (mysql_num_rows($result) > 0) {
```

```
    $response["user"] = array();
      while ($row = mysql_fetch_array($result)) {
        // temp user array
        $product = array();
        $product["id"] = $row["id"];
                $product["phone"] = $row["phone"];
                $product["nama"] = $row["nama"];
        // push single product into final response array
        array_push($response["user"], $product);
    }
    // success
    $response["success"] = 1;
    // echoing JSON response
    echo json_encode($response);
} else {
    // no products found
    $response["success"] = 0;
    $response["message"] = "No products found";
    // echo no users JSON
    echo json_encode($response);
}
?>
```

h. Implementasi Basis Data Get Dampak Aktivitas Matahari

```
<?php
/*
 * Following code will list all the products
 */
// array for JSON response
$response = array();
// include db connect class
require_once 'db_connect.php';
// connecting to db
$db = new DB_CONNECT();
// get all products from products table
$result = mysql_query("SELECT *FROM matahari where status=1 and
kategori=2") or die(mysql_error());
// check for empty result
if (mysql_num_rows($result) > 0) {
    // looping through all results
    // products node
    $response["matahari"] = array();
    while ($row = mysql_fetch_array($result)) {
        // temp user array
        $product = array();
```

```
        $product["id"] = $row["id"];
        $product["title"] = $row["title"];
        $product["date"] = $row["date"];
                $product["description"] = $row["description"];
                $product["status"] = $row["status"];
                $product["kategori"] = $row["kategori"];
                $product["source"] = $row["source"];
                $product["path"] = $row["path"];
        // push single product into final response array
        array_push($response["matahari"], $product);
    }
    // success
    $response["success"] = 1;
    // echoing JSON response
    echo json_encode($response);
} else {
    // no products found
    $response["success"] = 0;
    $response["message"] = "No products found";
    // echo no users JSON
    echo json_encode($response);
} ?>
```

i. Implementasi Data Base Get Gallery

```
<?php
/*
 * Following code will list all the products
 */
// array for JSON response
$response = array();
// include db connect class
require_once 'db_connect.php';
// connecting to db
$db = new DB_CONNECT();
// get all products from products table
$result = mysql_query("SELECT *FROM galery") or die(mysql_error());
// check for empty result
if (mysql_num_rows($result) > 0) {
    // looping through all results
    // products node
    $response["galery"] = array();
    while ($row = mysql_fetch_array($result)) {
        // temp user array
        $product = array();
        $product["id"] = $row["id"];
```

```
        $product["path"] = $row["path"];
                $product["desc"] = $row["desc"];
        // push single product into final response array
        array_push($response["galery"], $product);
    }
    // success
    $response["success"] = 1;
    // echoing JSON response
    echo json_encode($response);
} else {
    // no products found
    $response["success"] = 0;
    $response["message"] = "No products found";
    // echo no users JSON
    echo json_encode($response);
}
?>
```

j. Implementasi Data Base Insert Coment

```
<?php
/*
 * Following code will create a new product row
```

```
 * All product details are read from HTTP Post Request
 */
// array for JSON response
$response = array();
// check for required fields
if (isset($_GET['email']) && isset($_GET['comment'])) {
    $email = $_GET['email'];
    $comment = $_GET['comment'];
    $id_matahari = $_GET['id_matahari'];
    // include db connect class
    require_once 'db_connect.php';
    // connecting to db
    $db = new DB_CONNECT();
    // mysql inserting a new row
    $result = mysql_query("Insert into
comment(email,comment,id_matahari)
Values('$email','$comment',$id_matahari)");
    // check if row inserted or not
    if ($result) {
        // successfully inserted into database
        $response["success"] = 1;
        $response["message"] = "Product successfully created.";
```

```
        // echoing JSON response
        echo json_encode($response);
    } else {
        // failed to insert row
        $response["success"] = 0;
        $response["message"] = "Oops! An error occurred.";
        // echoing JSON response
        echo json_encode($response);
    }
} else {
    // required field is missing
    $response["success"] = 0;
    $response["message"] = "Required field(s) is missing";
    // echoing JSON response
    echo json_encode($response);
}
?>
```

k. Implementasi Data Base Get Coment

```
<?php
/*
 * Following code will list all the products
```

```
 */
// array for JSON response
$response = array();
// include db connect class
require_once 'db_connect.php';
// connecting to db
$db = new DB_CONNECT();
// get all products from products table
$id = $_GET['id'];
$result = mysql_query("SELECT *FROM comment where id_matahari =
$id") or die(mysql_error());
// check for empty result
if (mysql_num_rows($result) > 0) {
    // looping through all results
    // products node
    $response["matahari"] = array();
    while ($row = mysql_fetch_array($result)) {
        // temp user array
        $product = array();
        $product["id"] = $row["id"];
        $product["email"] = $row["email"];
        $product["comment"] = $row["comment"];
```

```
        // push single product into final response array
        array_push($response["matahari"], $product);
    }
    // success
    $response["success"] = 1;
    // echoing JSON response
    echo json_encode($response);
} else {
    // no products found
    $response["success"] = 0;
    $response["message"] = "No products found";
    // echo no users JSON
    echo json_encode($response);
}
?>
```

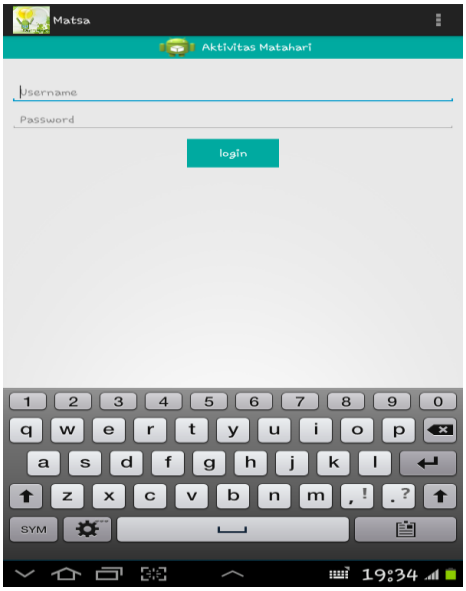
5.1.5. Implementasi Antar Muka

Implementasi antar muka merupakan tahap dimana aplikasi siap untuk dioperasikan, sehingga akan diketahui apakah aplikasi yang telah dibuat sesuai dengan apa yang telah direncanakan atau tidak. Fasilitas antar muka yang baik sangat membantu pemakai dalam memahami proses yang sedang dilakukan oleh aplikasi

tersebut dan dapat meningkatkan kinerja sistem. Berikut beberapa penjelasan dari bagaimana aplikasi ini bekerja dengan tampilan aplikasi yang dibuat.

1. Tampilan *Form Login*

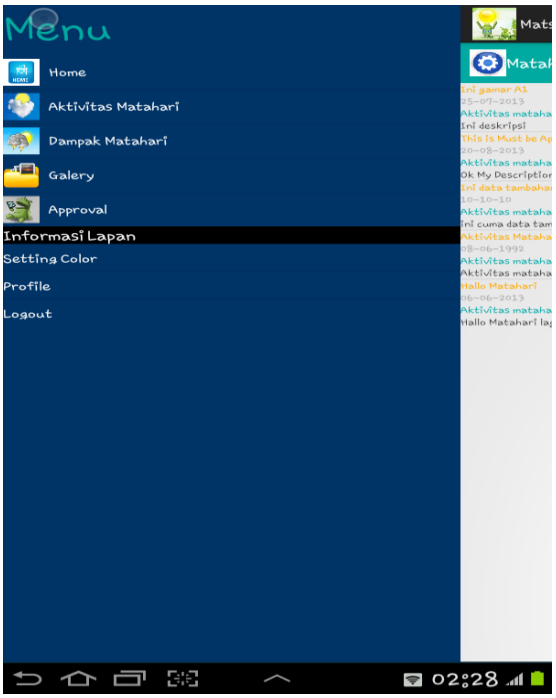
   Form Login ini adalah tampilan saat pertama kali aplikasi dijalankan untuk masuk menu utama admin. Untuk masuk menu utama, admin harus mengisi *username* dan *password* kemudian klik button *login.* Berikut tampilan *Form Login :*



Gambar 5.1 *Form Login* Admin

2. Tampilan Menu Utama Admin

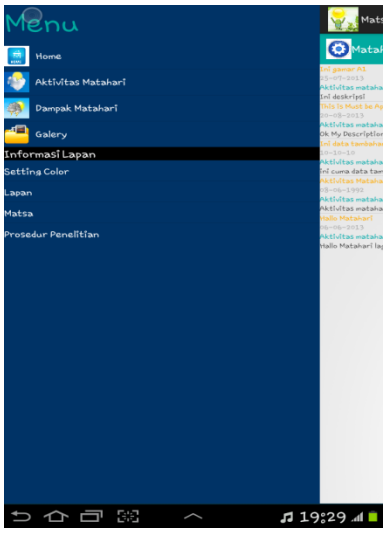
   Tampilan Menu Utama Admin adalah tampilan yang memperlihatkan menu apa saja yang dapat diakses oleh Admin dalam aplikasi ini. Berikut tampilan Menu Utama Admin :



Gambar 5.2 Tampilan Menu Utama Admin

3. Tampilan Menu Utama User

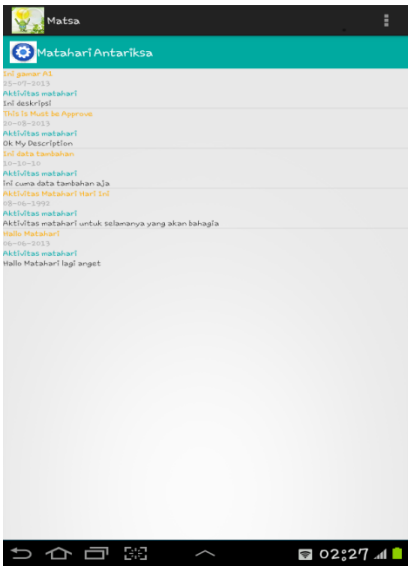
   Tampilan Menu Utama User adalah tampilan yang memperlihatkan menu apa saja yang dapat diakses oleh User dalam aplikasi ini. Berikut tampilan Menu Utama User :



Gambar 5.3 Tampilan Menu Utama User

4. Tampilan Menu Aktivitas Matahari

   Tampilan ini akan menampilkan list informasi yang telah di publish / di upload oleh admin berdasarkan kategori. Berikut adalah tampilannya :



Gambar 5.4 Tampilan Menu Aktivitas Matahari

5. Tampilan Menu Dampak Aktivitas Matahari

   Tampilan dari menu ini sama seperti tampilan pada menu Aktivitas Matahari. Hanya menampilkan list informasi yang telah di publish / di upload oleh admin berdasarkan kategori. Berikut tampilannya :



Gambar 5.5 Tampilan Menu Dampak Aktivitas Matahari

6. Tampilan Menu Approve

   Pada tampilan menu ini admin dapat *mem-publish* informasi apa saja yang akan dipublish berdasarkan kategori. Berikut tampilannya :



Gambar 5.6 Tampilan Menu Approve

7. Tampilan Menu Galery

   Menu ini akan menampilkan gambar-gambar atau foto-foto dari aktivitas dan dampak aktivitas matahari. Berikut tampilannya :



Gambar 5.7 Tampilan Menu Gallery

8. Tampilan Menu Setting Color

   Menu ini akan menampilkan beberapa warna background color, menu font color, dan header font color. Dimana admin dapat mengubah warna tampilan dari aplikasi ini dan juga dapat mengembalikan warna ke semula. Berikut tampilannya :



Gambar 5.8 Tampilan Menu Setting Color

9. Tampilan Menu Profile

   Menu ini akan menampilkan nama, *username*, *password* dan phone. Dimana admin dapat mengisi dan mengubah data tersebut yang akan langsung tersimpan ke database. Berikut tampilannya :



Gambar 5.9 Tampilan Menu Profile

10. Tampilan Menu Prosedur Penelitian User

    Menu ini adalah menu untuk menampilkan prosedur penelitian yang tersiri dari apa yang diteliti, cara meneliti dan alat-alat untuk meneliti, berikut tampilannya :



Gambar 5.10 Tampilan Menu Prosedur Penelitian User

11. Tampilan Menu Alat Penelitian User

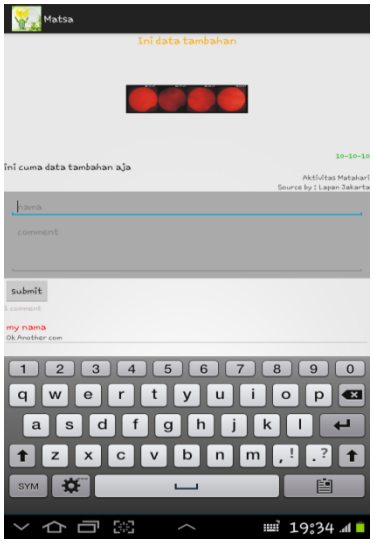
Menampilkan alat-alat yang digunakan untuk membantu peneliti dalam meneliti aktivitas matahari dan dampak aktivitas matahari, dibawah ini tampilannya :



Gambar 5.11 Tampilan Menu Alat Penelitian User

12. Tampilan Coment

Tampilan untuk menampilkan kolom komentar, berikut tampilannya :

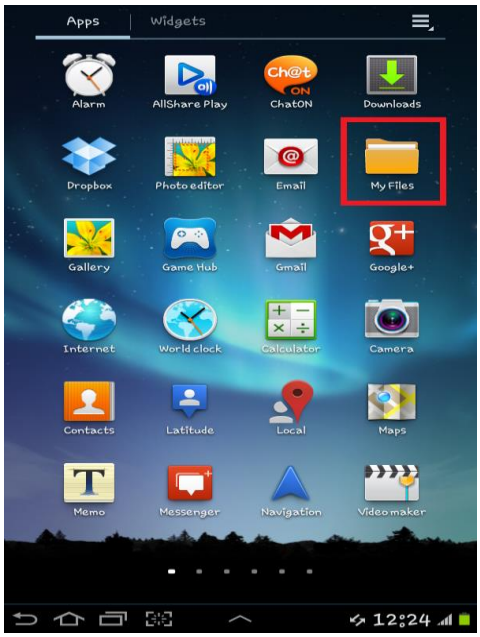


Gambar 5.12 Tampilan Coment

5.1.6. Implementasi Instalasi Program

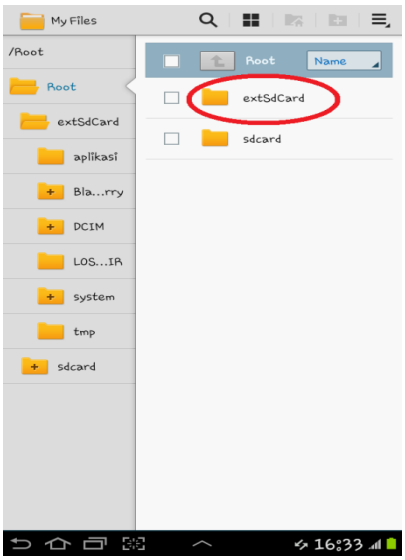
Untuk mengimplementasikan aplikasi ini, perlu penginstalan program pada *smartphone* Samsung Galaxy Tab 2 7” P3100. Seperti program pada umumnya tidak ada perbedaan pada penginstalan program. Berikut langkah-langkah instalasi program:

1. Buka folder My Files



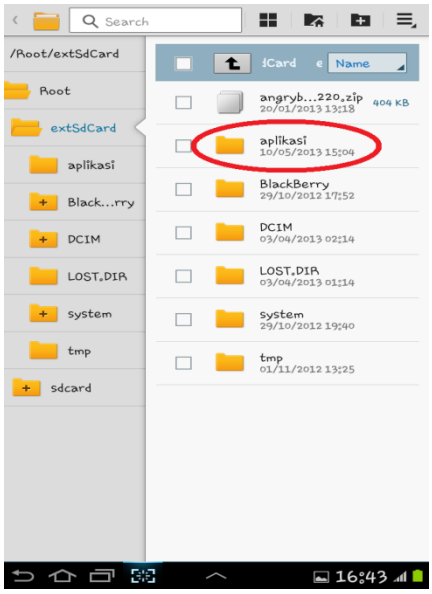
Gambar 5.13 Lokasi Folder My Files

2. Klik My files kemudian akan muncul tampilan seperti di bawah ini, kemudian pilih extSdCard



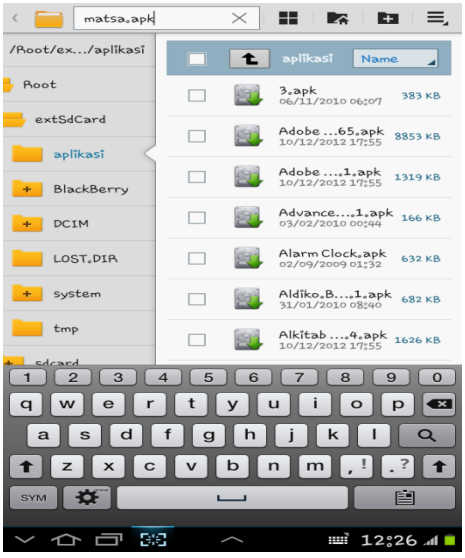
Gambar 5.14 Tapilan setelah klik folder My Files

3. Setelah memilih extSdCard, maka akan muncul tampilan seperti dibawah ini, kemudian pilih aplikasi



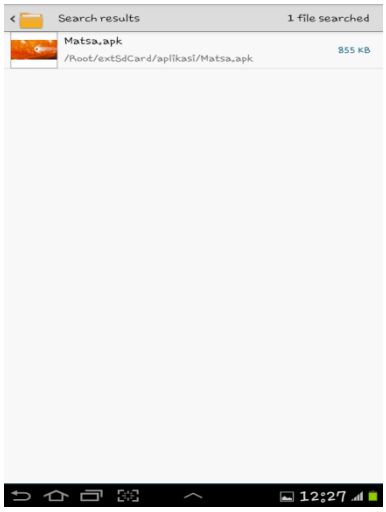
Gambar 5.15 Tampilan setelah memilih extSdCard

4. Kemudian cari file *matsa.apk*



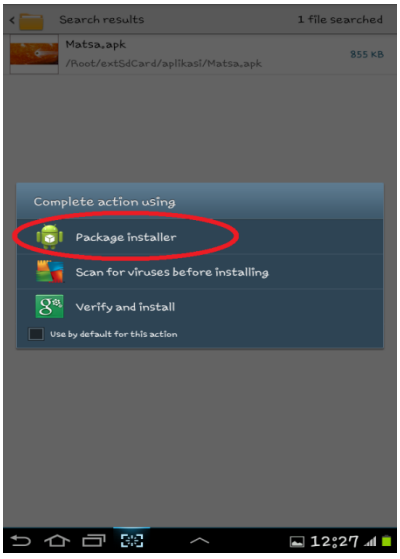
Gambar 5.16 Proses mencari file *Matsa.apk*

Jika sudah selesai mencari file *Matsa.apk*, maka akan muncul tampilan seperti dibawah, kemudian klik file *Matsa.apk* tsb.



Gambar 5.17 Hasil pencarian file *Matsa.apk*

5. Setelah di klik akan muncul tampilan seperti dibawah, dan pilih *Package Installer*



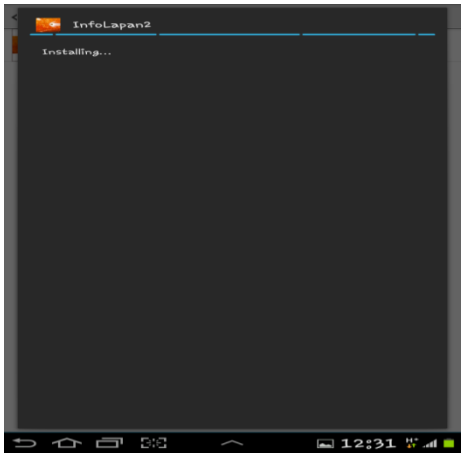
Gambar 5.18 Tampilan setelah pilih *Matsa.apk*

Setelah itu maka akan muncul message apakah aplikasi ini akan di install atau tidak. Karena kita akan menginstal aplikasi tersebut, maka pilih Install. Berikut tampilannya.



Gambar 5.19 Tampilan untuk menginstall

6. Setelah itu klik install lalu akan dilakukan proses penginstalan seperti pada tampilan dibawah ini.



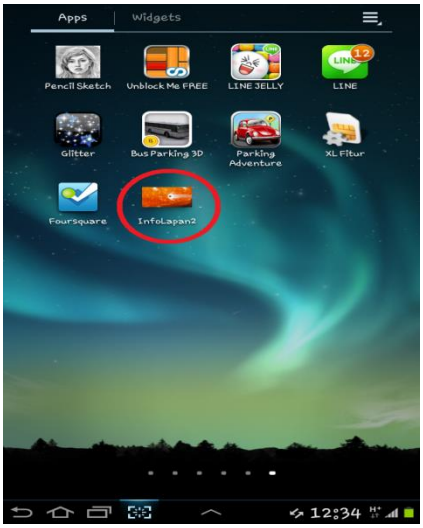
Gambar 5.20 Tampilan Proses Instalasi Aplikasi

7. Setelah Proses Instalasi selesai maka akan muncul tampilan apakah aplikasi akan langsung di buka atau tidak seperti tampilan berikut.



Gambar 5.21 Tampilan setelah selesai menginstal *Matsa.apk*

8. Jika sudah terinstal, maka aplikasi akan ada di *smartphone*, seperti pada gambar yang terlihat dibawah ini.



Gambar 5.22 Tampilan Aplikasi *Matsa.apk* di *Smartphone Android*

5.1.7. Penggunaan Program

Pada bagian ini akan dijelaskan bagaimana cara-cara mengaplikasikan aplikasi pengelolaan konten informasi hasil penelitian matahari dan antariksa. Di bawah ini terdapat beberapa *interface* yang dapat digunakan oleh admin.

5.1.7.1. Halaman *Form Login*

Halaman ini adalah halaman utama yang merupakan tampilan awal ketika admin menjalankan aplikasi pengelolaan konten informasi hasil penelitian matahari dan antariksa. Dalam halaman ini terdapat kolom *username* dan *password* yang akan diisi oleh admin untuk bias mengakses semua menu yang ada pada aplikasi.



Gambar 5.23 Halaman *Form Login*



Gambar 5.24 Proses Validasi *Login*

5.1.7.2. Halaman Utama Admin

Setelah mengisi *Form Login*, halaman utama setelah admin mengisi *username* dan *password* adalah menampilkan list informasi aktivitas matahari yang telah di publish / di upload.



Gambar 5.25 Halaman Utama Admin

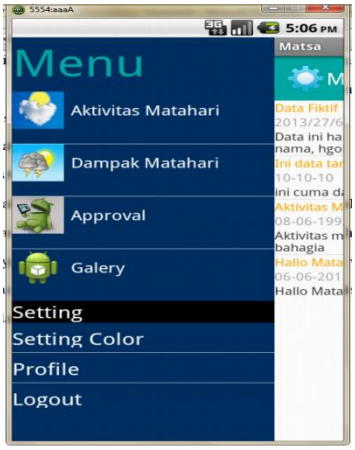
5.1.7.3. Halaman Menu Utama Admin

Setelah masuk halaman utama admin, admin akan masuk ke halaman menu utama admin untuk mengakses menu-menu yang disediakan oleh sistem. Untuk mengakses menu utama admin, admin harus mengklik tombol yang ada di header menu aktivitas matahari.



Gambar 5.26 Admin mengklik tombol menu utama

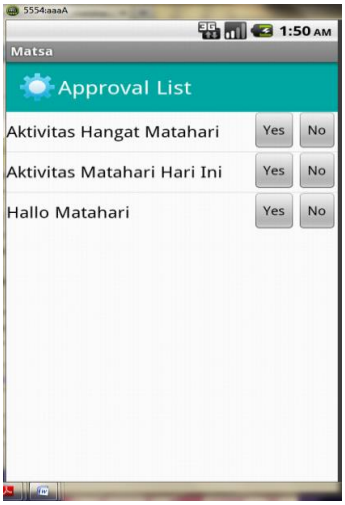
Setelah klik tombol tersebut, maka akan muncul halaman menu utama admin :



Gambar 5.27 Halaman Menu Utama Admin

5.1.7.4. Halaman Menu Approve

Halaman ini adalah halaman dimana admin akan mem-publish informasi yang telah diolah oleh admin bank data. Jika admin memilih “yes” maka informasi berhasil di publish, dan jika admin memilih “no” maka informasi tidak di publish. Maka status pada database akan berubah, misalkan dari “0” menjadi “1” artinya informasi berhasil di publish dan jika dari “0” menjadi “2” artinya informasi ditolak atau tidak di publish.



Gambar 5.28 Halaman Menu Approve

5.1.7.5. Halaman Menu Profile

Halaman menu profile adalah menu dimana admin dapat mengubah *username* dan *password***,** nama, dan nomor *telephone* jika admin tersebut mengganti nomor *telephone*-nya. Jika semua kolom telah diisi, maka admin akan mengklik button “*edit*” dan jika tidak akan ada yang diganti maka admin hanya mengklik tanda kembali yang disediakan *device*.



Gambar 5.29 Halaman Menu Profile

Jika admin klik "*edit*" maka data akan tersimpan di database.



Gambar 5.30 Proses Penyimpanan Data Baru ke Database

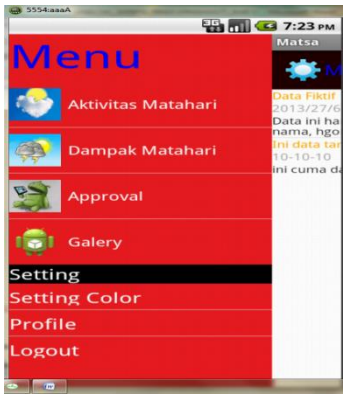
5.1.7.6. Halaman Menu Setting Color

Halaman ini digunakan admin untuk mengubah warna background, menu font, dan header font. Admin dapat memilih beberapa warna yang telah disediakan di aplikasi ini. Jika admin ingin merubah warna dari tampilan aplikasi ini, admin tinggal memilih warna dan klik "*save"*, maka warna tampilan akan berubah. Dan jika admin menginginkan warna semula, maka admin tinggal memilih "*setdefault*".



Gambar 5.31 Halaman Setting Color

Maka warna akan berubah seperti berikut :



Gambar 5.32 Tampilan setelah warna dirubah

5.2. Pengujian

Pengujian adalah bagian yang penting dalam siklus pembangunan perangkat lunak. Pengujian dilakukan untuk menjamin kualitas dan juga mengetahui kelemahan dari perangkat lunak. Dalam pengujian ini digunakan metode pengujian *black box*. Dimana pengujian *black box* itu sendiri adalah pengujian yang hanya menggunakan aspek fundamental system tanpa memperhatikan struktur logika internal perangkat lunak. Tujuannya adalah untuk mengetahui apakah perangkat lunak sudah berfungsi dengan benar. Perangkat lunak ini dieksekusi dan dicek apakah perangkat lunak yang dibuat sesuai dengan apa yang diharapkan atau tidak.

5.2.1. Rencana Pengujian

Dalam metode pengujian *black box* terdapat beberapa faktor pengujian untuk mengetahui tingkat kelayakan penggunaan perangkat lunak. Rencana pengujian pada perangkat lunak yang telah dibuat, akan dilakukan dengan menggunakan tiga faktor pengujian saja, yaitu :

1. Correctness

   Menjamin pada data yang dimasukan, proses dan output yang dihasilkan dari aplikasi harus akurat dan lengkap.

2. File Integrity

   Menekankan pada data yang dimasukan melalui aplikasi akan tidak bisa diubah. Procedure yang akan memastikan bahwa file yang digunakan benar dan data dalam file tersebut akan disimpan sekuensial dan benar.

3. Audit Trail

Menekankan pada kemampuan untuk mendukung proses yang terjadi, pemrosesan data secara keseluruhan berdasarkan retensi dari kejadian yang cukup mendukung keakuratannya, kelengkapannya, batas waktu dan otoritas data.

Berikut adalah table rencana pengujian pada Aplikasi Pengelolaan Konten Informasi Hasil Penelitian :

Tabel 5.1 *Rencana Pengujian*

Kelas Uji	Detail Pengujian	Jenis Pengujian
Login	Menampilkan form login untuk admin	Black Box

5.2.2. Kasus dan Hasil Pengujian

Berdasarkan pengujian yang telah disusun, maka dapat dilakukan pengujian sebagai berikut :

Tabel 5.2 *Pengujian Login Admin*

Kasus dan Hasil Uji Login Admin (Data Normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Username : rindy *Password* : rindy	Dapat masuk ke form selanjutnya.	Dapat melakukan pengisian data secara benar sesuai yang diharapkan.	[√] Diterima [ ] Ditolak

Klik tombol ENTER	Tampil ke form selanjutnya / form menu_admin	Tombol ENTER dapat berfungsi sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Login Admin (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
User ID : rindy *Password* : rindy	Tidak dapat login dan isian kembali kosong, tampilan tidak berubah	*Admin* tidak dapat login sebagai admin sehingga tidak sesuai dengan yang diharapkan.	[√] Diterima [ ] Ditolak

5.2.3. Kesimpulan Hasil Pengujian

Berdasarkan hasil pengujian dengan kasus uji sampel diatas, dapat ditarik kesimpulan bahwa perangkat lunak bebas dari kesalahan sintaks dan secara fungsional mengeluarkan hasil yang sesuai dengan yang diharapkan.