



Perancangan Aplikasi Unggah Informasi Kecelakaan dari Masyarakat Berbasis Android di PT. Jasa Raharja

Bondan Sasmito Aji ^{1*}, Pratyaksa Ocsa Nugraha Saian ²

^{1,2} Program Studi Teknik Informatika, Fakultas Teknologi Informasi, Universitas Kristen Satya Wacana, Kota Salatiga, Provinsi Jawa Tengah, Indonesia.

article info

Article history:

Received 21 March 2023

Received in revised form

21 May 2023

Accepted 20 August 2023

Available online October 2023

DOI:

<https://doi.org/10.35870/jtik.v7i4.1051>

Keywords:

Android; Java; Information;
Accident.

Kata Kunci:

Android; Java; Informasi;
Kecelakaan.

abstract

PT. Jasa Raharja is a state-owned enterprise (BUMN) operating in the field of accident insurance services and is headquartered in Kuningan, South Jakarta. Currently, this company does not have an accident information application to monitor traffic conditions. In the context of constantly developing technology, PT. Jasa Raharja needed an accident information app that would allow him to track accident locations in real time. This study is based on these problems and aims to find a suitable solution by designing an accident information application. This application allows the public to report accidents through the Android platform built using the Java programming language. This study uses the Waterfall research methodology, which includes requirements, design, implementation, verification, and maintenance phases. Test results from potential users using a Likert scale show that this app received a score of 85.2%, meaning that the majority of users strongly agree with the effectiveness and usefulness of this app. With this value, it can be concluded that the designed application can help the public report accidents that occur and meets the needs of users well.

abstract

PT. Jasa Raharja adalah sebuah perusahaan Badan Usaha Milik Negara (BUMN) yang beroperasi di bidang layanan asuransi kecelakaan, dan berkantor pusat di Kuningan, Jakarta Selatan. Saat ini, perusahaan ini belum memiliki aplikasi informasi kecelakaan untuk memantau kondisi lalu lintas. Dalam konteks perkembangan teknologi yang terus berlanjut, PT. Jasa Raharja perlu untuk memiliki sebuah aplikasi informasi kecelakaan yang memungkinkan mereka untuk memantau lokasi kecelakaan secara real-time. Penelitian ini didasarkan pada permasalahan tersebut dan bertujuan untuk menemukan solusi yang tepat dengan merancang sebuah aplikasi informasi kecelakaan. Aplikasi ini memungkinkan masyarakat untuk melaporkan kecelakaan melalui platform berbasis Android yang dibangun menggunakan bahasa pemrograman Java. Penelitian ini menggunakan metode penelitian Waterfall, yang melibatkan tahapan Requirement (Penyusunan Kebutuhan), Design (Perancangan), Implementation (Pengembangan), Verification (Verifikasi), dan Maintenance (Pemeliharaan). Hasil pengujian kepada calon pengguna dengan menggunakan skala Likert menunjukkan bahwa aplikasi ini mendapatkan nilai sebesar 85.2%, yang berarti bahwa sebagian besar pengguna sangat setuju dengan keefektifan dan kegunaan aplikasi ini. Dengan nilai ini, dapat disimpulkan bahwa aplikasi yang dirancang dapat membantu masyarakat dalam melaporkan kecelakaan yang terjadi dan memenuhi kebutuhan pengguna dengan baik.

*Corresponding Author. 672018184@student.uksw.edu *

1. Latar Belakang

Era globalisasi membuat kemajuan teknologi menjadi bagian dalam kehidupan manusia khususnya pada aktivitas manusia yang memungkinkan berkomunikasi secara *mobile* dan *online*. Hal ini tentu membuat segala aktivitas menjadi lebih efektif dan efisien dalam memperoleh informasi. Perkembangan teknologi pada media informasi menjadi salah satu hal utama untuk memperoleh informasi.

Terbitnya Instruksi presiden Nomor 3 tahun 2003 menjadi titik awal penerapan teknologi pada pemerintah di Indonesia [1]. Instruksi tersebut menekankan pentingnya pemanfaatan teknologi informasi dalam organisasi agar lebih efektif dan efisien. Penerapan awal ini dimulai dari pembuatan web sebagai portal informasi. Dengan adanya informasi secara *online*, masyarakat bisa mengetahui informasi secara mudah serta menambah wawasan tentang keadaan yang sedang terjadi.

Seiring dengan berkembangnya teknologi web, manusia melakukan pengembangan yang memungkinkan dapat mengakses informasi dimana saja dan kapan saja. Pengembangan tersebut berupa *gadget* atau lebih dikenal secara umum dengan sebutan ponsel. Dahulu, ponsel hanya bisa digunakan sebagai alat komunikasi jarak jauh dengan teknologi bernama *Short Message Service (SMS)* dan telepon. Pada tahun 2010, perkembangan ponsel menjadi lebih canggih dengan layar sentuh serta menjadi masa kejayaan dua sistem operasi yaitu Android dan iOS [2]. Ponsel canggih yang dikembangkan ini dinamakan *smartphone*. Tanpa disadari *smartphone* sudah menjadi bagian dalam kehidupan aktivitas manusia. Melalui pengembangan yang dilakukan oleh perusahaan pengembang, maka *smartphone* dapat dijadikan perangkat untuk mengakses informasi di mana pun dan kapan pun.

Salah satu informasi penting bagi manusia adalah berita. Perusahaan-perusahaan besar mulai melakukan pengembangan berupa aplikasi *mobile* dengan tujuan agar masyarakat dapat mengetahui berita maupun informasi melalui ponsel masing-masing. Di Indonesia, aplikasi berita sudah banyak digunakan oleh masyarakat pengguna ponsel, namun

informasi pada portal informasi yang digunakan oleh masyarakat hanya dapat diunggah oleh perusahaan pengembang portal informasi tersebut, hal ini tentu membatasi keterjangkauan informasi yang terjadi di daerah tertentu. Informasi kecelakaan menjadi hal yang penting pada kehidupan, adanya informasi kecelakaan dapat mengantisipasi masyarakat agar terhindar dari lokasi kecelakaan tersebut sehingga terhindar dari kemacetan, kepanikan, dan menginformasikan masyarakat jika terjadi kecelakaan di daerah tersebut.

Dengan adanya masalah tersebut, maka penelitian ini dilakukan pengembangan portal informasi kecelakaan serta menjadi aplikasi yang dapat mengunggah informasi kecelakaan dari masyarakat berbasis Android. Pembuatan aplikasi menggunakan Java sebagai bahasa pemrograman dan pembuatan sistem menggunakan bahasa PHP untuk menghubungkan *Database Management System (DBMS)* MySQL serta menggunakan *REST API* untuk penyimpanan dan pemanggilan data dari *database*. Aplikasi dirancang dengan basis Android karena Android sangat umum digunakan di masyarakat Indonesia dan sistem operasi tersebut lebih banyak digunakan oleh perusahaan pengembang *smartphone* [3]. Aplikasi yang akan dibuat menggunakan metode *waterfall* dengan tujuan agar pembuatan aplikasi dapat dikembangkan secara sekuensial [4]. Keuntungan dalam mengembangkan sebuah aplikasi menggunakan metode *waterfall* membuat pembuatan aplikasi lebih praktis, serta pengembangan menjadi terstruktur dan kualitas aplikasi terjaga, sehingga metode *waterfall* tepat digunakan untuk penelitian [5].

Penelitian sebelumnya telah menyoroti beberapa kendala yang terjadi dalam implementasi sistem berbasis Android dan website terkait informasi di beberapa instansi dan lembaga. Dalam kaitannya dengan penelitian ini yang berfokus pada perancangan aplikasi unggah informasi kecelakaan dari masyarakat berbasis Android di PT. Jasa Raharja, temuan dari penelitian terdahulu menjadi landasan penting yang mendukung kebutuhan akan pengembangan sistem yang lebih efisien dan responsif. Leo Surya Duha (2018) dalam penelitiannya "Aplikasi Berita Online Berbasis Android: Studi Pada Pemerintah Kabupaten Labuhanbatu" menggarisbawahi tantangan dalam pengelolaan

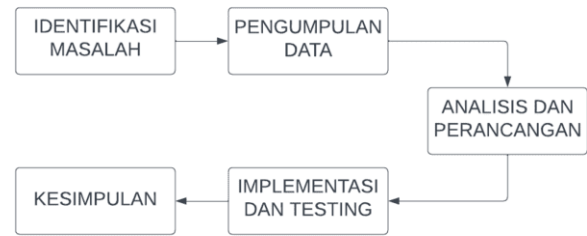
informasi di lingkungan pemerintahan yang masih mengandalkan proses manual. Hal ini memberikan gambaran akan kebutuhan akan pengembangan sistem yang dapat secara otomatis mengelola informasi secara efektif dan efisien di lingkungan instansi pemerintah [6]. Sementara itu, penelitian yang dilakukan oleh Afrinaldi (2015) dengan judul "Aplikasi Mobile Portal Berita Ganto.Or.Id Berbasis Android" menyoroti permasalahan antarmuka pengguna yang kurang ramah dalam aplikasi berbasis Android. Dalam konteks ini, perancangan aplikasi di PT. Jasa Raharja dapat mempertimbangkan pentingnya antarmuka yang user-friendly untuk meningkatkan interaksi pengguna dengan sistem [7]. Demikian pula, dalam penelitian oleh Sarini Vita Dewi (2019) yang membahas "Perancangan Aplikasi Portal Informasi Mahasiswa Pada Universitas Ubudiyah Di Indonesia Berbasis Android," fokus pada keterbatasan akses informasi melalui website. Dengan demikian, perancangan aplikasi di PT. Jasa Raharja harus memperhatikan kemudahan akses informasi bagi masyarakat umum, terutama dalam hal mengunggah informasi kecelakaan secara efektif [8].

Oleh karena itu, penelitian ini bertujuan untuk mengatasi hambatan-hambatan yang diidentifikasi dalam penelitian sebelumnya dengan merancang sebuah aplikasi berbasis Android yang memungkinkan masyarakat untuk mengunggah informasi terkait kecelakaan secara efisien. Aplikasi ini akan mencakup fitur-fitur seperti judul kecelakaan, waktu kejadian, lokasi kejadian, informasi detail tentang kecelakaan, dan kemampuan unggah gambar langsung melalui kamera ponsel. Dengan perancangan menggunakan bahasa pemrograman Java, PHP, dan MySQL, aplikasi ini diharapkan dapat memberikan solusi yang efektif bagi PT. Jasa Raharja dalam mengelola informasi kecelakaan dengan lebih efisien dan responsif.

2. Metode Penelitian

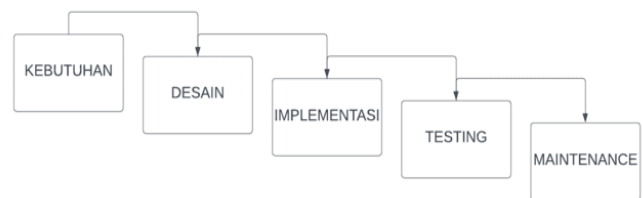
Metode penelitian merupakan tahapan untuk menguraikan penelitian agar menjadi sistematis dengan tujuan untuk memudahkan penelitian yang dilakukan. Tahapan yang penelitian yang dilakukan

seperti pada Gambar 1 [9].



Gambar 1. Tahapan Penelitian [10].

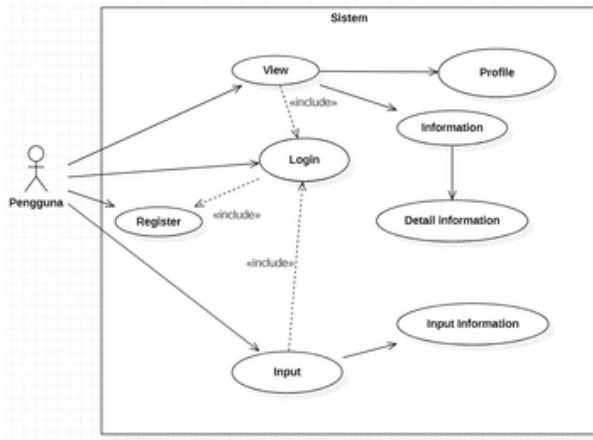
Tahapan awal penelitian diawali dengan identifikasi masalah. Pada identifikasi masalah dilakukan dengan wawancara terhadap pihak PT. Jasa Raharja. Setelah melakukan identifikasi masalah, maka pengumpulan data dilakukan dengan metode wawancara dan studi pustaka [11]. Wawancara yang dilakukan meliputi kebutuhan perusahaan terhadap aplikasi, studi pustaka dilakukan dengan membaca literatur yang berkaitan dengan penelitian untuk mendukung penelitian. Setelah melakukan pengumpulan data, maka analisis data dilakukan. Perancangan yang dilakukan berupa *user interface* aplikasi menggunakan figma dan rancangan sistem menggunakan *Unified Modelling Language (UML)*. Desain sistem dan tampilan yang telah dibuat akan diimplementasikan pada tahap implementasi menggunakan bahasa pemrograman Java dan PHP serta *DBMS* MySQL dan dilakukan *testing* untuk menguji apakah aplikasi dapat berjalan dengan baik. Setelah melakukan implementasi dan *testing* maka dapat diambil kesimpulan terkait perancangan aplikasi yang dibuat.



Gambar 2. Metode Waterfall [12]

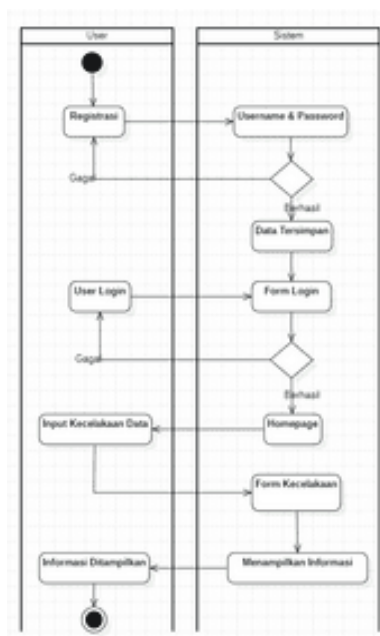
Tahap pertama dalam menggunakan metode *waterfall* adalah kebutuhan. Tahap kebutuhan merupakan proses pengumpulan data untuk penelitian. Tahap desain merupakan tahap perancangan sistem dan tampilan aplikasi serta fungsi fungsi yang akan ditambahkan ke dalam aplikasi. Setelah membuat desain maka akan dilanjutkan ke tahap implementasi, dimana semua desain yang telah dibuat akan

diprogram menggunakan bahasa Java dan PHP serta meliputi pembuatan *database*. Aplikasi yang sudah jadi akan di *testing* sebagai pengecekan bahwa aplikasi dapat berjalan dengan baik. Pada tahap *maintenance* berfungsi untuk pengecekan rutin aplikasi [13].



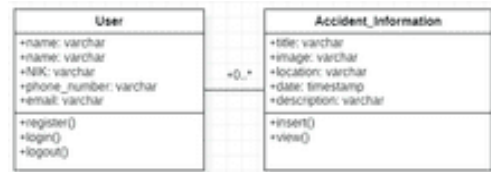
Gambar 3 .Use Case Diagram

Gambar 3. merupakan *Use Case Diagram* yang berfungsi sebagai gambaran ringkas pengguna aplikasi dan apa saja yang dapat dilakukan oleh pengguna [14]. Pada *Use Case Diagram* menjelaskan bahwa pengguna dapat melakukan registrasi, *login*, unggah informasi dan melihat informasi. Pengguna yang ingin *login* harus melakukan registrasi terlebih dahulu dengan *email* dan *password*. Pengguna yang telah berhasil *login* dapat melihat informasi kecelakaan serta unggah informasi kecelakaan.



Gambar 4. Activity Diagram

Gambar 4 merupakan *Activity Diagram* yang berfungsi sebagai alur aktivitas yang dilakukan oleh pengguna aplikasi untuk mengakses fungsi-fungsi yang terdapat pada aplikasi. *Activity Diagram* menjelaskan bahwa pengguna harus melakukan registrasi menggunakan *email* dan *password* agar bisa melakukan *login*, pengguna yang gagal melakukan registrasi tidak akan bisa melakukan *login* dan harus melakukan registrasi lagi. Setelah pengguna berhasil *login*, maka akan dilanjutkan ke halaman utama yang berisikan informasi kecelakaan serta fitur untuk melaporkan informasi kecelakaan.

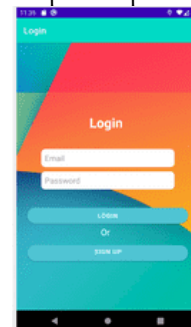


Gambar 5. Class Diagram

Class Diagram berfungsi sebagai gambaran struktur suatu sistem dalam pemrograman. Gambar 5 menjelaskan bahwa tabel *user* memiliki 5 *field* terdiri dari nama, kata sandi, NIK, nomor ponsel, serta *email*. Tabel informasi kecelakaan memuat 5 *field* terdiri dari judul, foto, lokasi, tanggal dan deskripsi. Pengguna yang terdaftar dapat melihat data diri pada bagian *profile* setelah melakukan *login*.

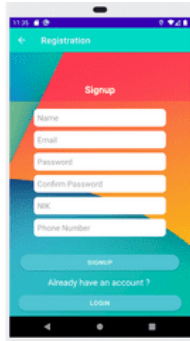
3. Hasil dan Pembahasan

Setelah melakukan rancangan, maka hasil dan pembahasan berisi tentang implementasi berdasarkan rancangan yang telah dibuat. Penelitian yang telah dilakukan menghasilkan sebuah aplikasi informasi kecelakaan serta unggah informasi kecelakaan menggunakan Java untuk tampilan aplikasi dan PHP untuk koneksi antara Java dengan *DBMS* MySQL. Berikut ini adalah tampilan aplikasi.



Gambar 6. Halaman Login

Gambar 6 merupakan tampilan halaman *login*. Halaman *login* terdiri atas dua *field* bernama *email* dan *password*. Pada halaman *login* memiliki dua *button* yang berfungsi sebagai *login* dan *registrasi* akun. *Button login* akan mengarahkan pengguna ke halaman utama, sementara *button sign up* akan mengarahkan pengguna ke halaman registrasi.



Gambar 7. Halaman Sign Up

Gambar 7 merupakan tampilan halaman *sign up*. Halaman *sign up* memiliki enam *field* yang terdiri atas nama, *email*, *password*, *confirm password*, NIK, dan nomor telpon. Halaman *sign up* memiliki dua *button* yaitu *sign up* dan *login*. *Button sign up* berfungsi untuk menambahkan data *user* ke dalam *database*, *button login* berfungsi untuk perpindahan halaman registrasi ke halaman *login*.



Gambar 8. Halaman Home

Gambar 8 merupakan tampilan halaman utama. Pada tampilan halaman utama berisi informasi kecelakaan, *profile*, dan fitur unggah informasi. Fitur unggah informasi berfungsi untuk perpindahan dari halaman utama ke halaman unggah informasi, *profile* berfungsi sebagai informasi data diri pengguna yang telah berhasil *login*.



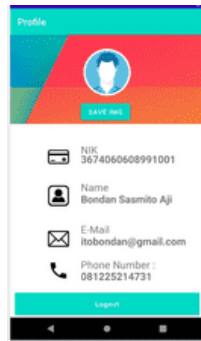
Gambar 9. Halaman Input

Gambar 9 merupakan halaman unggah informasi yang berisi unggah gambar, judul kecelakaan, lokasi berdasarkan *GPS*, deskripsi gambar dan *button* unggah informasi. Fitur unggah gambar hanya bisa digunakan oleh kamera dan akan di unggah ke halaman unggah informasi setelah melakukan pengambilan foto. Fitur lokasi menggunakan *GPS* dari pengguna ponsel, deskripsi berisi keterangan terjadinya kecelakaan, *button* unggah berfungsi untuk menyimpan informasi ke *database*. Apabila terdapat satu kolom yang tidak terisi maka *button* unggah informasi tidak dapat digunakan.



Gambar 10. Halaman Detail

Gambar 10 merupakan halaman detail dari informasi kecelakaan yang dipilih. Pada halaman detail terdapat foto, judul kecelakaan, lokasi kecelakaan, deskripsi kecelakaan dan jam kecelakaan secara *real time* setelah berhasil melakukan unggah informasi kecelakaan.



Gambar 11. Halaman Profile

Gambar 11 merupakan tampilan dari halaman *profile*. Halaman *profile* berisi foto *profile*, NIK, nama, *email*, serta nomor telepon. Pada halaman *profile* terdapat dua fitur yaitu fitur untuk mengganti foto *profile* dan fitur *log out* untuk keluar dari aplikasi menuju halaman *login*. Informasi pengguna yang tertera pada halaman *profile* merupakan informasi yang telah di daftarkan pada halaman registrasi dan informasi tersebut akan tampil apabila pengguna berhasil *login*.

Kode Program 1 Pengambilan Data Menggunakan PHP

```
$stmt = $conn->prepare("SELECT id,
image_path, image_name, location,
description, datet FROM tbl_image");
```

Kode Program 1 merupakan program pemanggilan data kecelakaan dari MySQL yang nantinya akan dihubungkan ke bahasa pemrograman Java.

Kode Program 2 Adapter Jawa

```
public void onBindViewHolder(@NonNull
ViewHolder holder, int position) {

    final ModelData modelData =
mListItem.get(position);

    Glide.with(context)
    .load(modelData.getImage_path())
    .diskCacheStrategy(DiskCacheStrategy.ALL)
    .into(holder.thumbnail);

    holder.imageName.setText(modelData.getIm
age_name());

    holder.location.setText(modelData.getLoc
ation());

    holder.description.setText(modelData.get
Description());
```

```
holder.datet.setText(modelData.getDatet()
);

holder.ReadMore.setOnClickListener(new
View.OnClickListener() {
    @Override
    public void onClick(View view)
    {
        Intent intent = new
Intent(context, DetailActivity.class);

        intent.putExtra("model",mListItem.get(pos
ition));

        context.startActivity(intent);
    }
});
```

Kode Program 2 merupakan program *adapter*. *Adapter* digunakan sebagai penghubung antara data dengan halaman utama. *Adapter* memiliki *ViewHolder* yang berfungsi untuk memberikan tata letak data di tempat yang telah ditentukan. *Library glide* pada *adapter* berfungsi untuk pemanggilan foto yang tersimpan di *database*.

Kode Program 3 Halaman Utama

```
JSONArrayRequest requestImage = new
JSONArrayRequest(Request.Method.POST,
url, null,
new Response.Listener<JSONArray>() {
    @Override
    public void
onResponse(JSONArray response) {
        Log.d("JSONResponse",
response.toString());

        //mRecyclerview.setVisibility(View.VISIBL
E);
        for (int i = 0; i < response.length(); i++)
        {
            try {

                JSONObject data =
response.getJSONObject(i);
                ModelData model = new ModelData();

                model.setId(data.getInt("id"));
                model.setImage_name(data.getString("image
_name"));
                model.setLocation(data.getString("locatio
n"));
                model.setDescription(data.getString("desc
ription"));
                model.setImage_path(data.getString("image
```

```

_path"));
model.setDatet(data.getString("datet"));
mListItem.add(model);
} catch (JSONException e) {
e.printStackTrace();}
mAdapter.notifyDataSetChanged();
}
},
new
Response.ErrorListener() {
@Override
public void
onErrorResponse(VolleyError error) {
Log.d("ERRORRequest", "Error : " +
error.getMessage());
}
});

mRequest.add(requestImage);

```

Kode Program 3 merupakan program pemanggilan pada halaman utama. Pada kode program 3 memiliki keterkaitan dengan *adapter* dan *model data* yang sudah dibuat. Apabila *adapter* dan *model data* sudah ada, maka kode program 3 berfungsi untuk pemanggilan data dengan menyesuaikan tata letak pada halaman utama.

Kode Program 4 Permission Android Manifest

```

<uses-permission
android:name="android.permission.CAMERA"
/>
<uses-permission
android:name="android.permission.ACCESS_
FINE_LOCATION" />
<uses-permission
android:name="android.permission.ACCESS_
COARSE_LOCATION"/>

```

Kode Program 4 merupakan bagian dari *Android Manifest* yang berisi perizinan untuk menggunakan fitur pada *smartphone*. Perizinan yang digunakan adalah penggunaan kamera untuk kebutuhan aplikasi agar dapat melakukan pengambilan gambar.

Kode Program 5 Program Kamera

```

this.viewImage
=(ImageView) this.findViewById(R.id.image
View1);
Button photoButton =
(Button) this.findViewById(R.id.button1);

photoButton.setOnClickListener(new

```

```

View.OnClickListener() {
@Override
public void onClick(View view)
{
if
(checkSelfPermission(Manifest.permission.
CAMERA)
!=
PackageManager.PERMISSION_GRANTED)
{
requestPermissions(new
String[]{Manifest.permission.CAMERA},
MY_CAMERA_PERMISSION_CODE);
}
else
{
Intent cameraIntent =
new
Intent(android.provider.MediaStore.
ACTION_IMAGE_CAPTURE);
startActivityForResult(cameraIntent,
CAMERA_REQUEST);
}
}
});

```

Kode Program 5 merupakan program yang digunakan untuk menjalankan fungsi kamera pada *smartphone*. Setelah memberikan perizinan untuk menggunakan kamera pada *smartphone*, maka perlu dilakukan pemrograman untuk menjalankan fitur kamera. Setelah dilakukan pengambilan foto, maka foto akan langsung ditempatkan di *PlaceHolder* yang telah disiapkan pada file XML.

Kode Program 6 Program Unggah

```

UploadImageServer =
(Button) findViewById(R.id.buttonUpload);

UploadImageServer.setOnClickListener(new
View.OnClickListener() {
@Override
public void onClick(View
view) {

GetImageNameEditText =
imageName.getText().toString();

GetLocationEditText =
alocation.getText().toString();

GetDescriptionEditText =
description.getText().toString();

if (GetDescriptionEditText.length() <
20) {

```

```

        Toast.makeText(context, "can't
upload",
        Toast.LENGTH_SHORT).show();

        } else if (viewImage
!=null){
ImageUploadToServerFunction();

Intent intent = new
Intent(InputActivity.this,RecyclerViewAc
tivity.class);
        startActivity(intent);
        return;
        }
    }
});

```

Kode Program 6 merupakan program penyimpanan informasi ke dalam database. *getDescriptionEditText.length() < 20* berfungsi untuk

mencegah deskripsi yang singkat. Setelah berhasil melakukan unggah informasi, maka fungsi *intent* akan melakukan perpindahan ke halaman utama.

Setelah melalui tahap implementasi maka dilanjutkan ke tahap pengujian sistem. Pengujian dilakukan dengan melibatkan orang-orang internal pada divisi *Information Technology (IT)* di PT. Jasa Raharja untuk mengetahui apakah aplikasi dapat berjalan dengan lancar.

Setelah melakukan pengujian maka selanjutnya akan dilakukan presentasi terakhir kepada divisi *Information Technology (IT)* PT. Jasa Raharja dengan tujuan untuk memastikan bahwa aplikasi dapat berjalan dengan lancar dan dapat dipahami.

Tabel 1. Pengujian Sistem

Pengujian	Yang Diharapkan	Pengamatan	Kesimpulan
Registrasi dan <i>login</i> yang dilakukan oleh pengguna aplikasi	Registrasi berhasil. Data tersimpan di <i>database</i>	Registrasi berhasil. Data tersimpan di <i>database</i>	[V] Diterima [] Ditolak
Unggah informasi kecelakaan melalui halaman <i>input</i>	Pengambilan gambar dari kamera dan pengambilan lokasi melalui GPS berhasil. Informasi tersimpan	Pengambilan gambar dari kamera dan pengambilan lokasi melalui GPS berhasil. Informasi tersimpan	[V] Diterima [] Ditolak
Informasi kecelakaan yang tersimpan di <i>database</i> dapat ditampilkan menggunakan <i>RecyclerView</i>	Informasi dapat ditampilkan di halaman utama menggunakan <i>RecyclerView</i> pada <i>CardView</i>	Informasi dapat ditampilkan di halaman utama menggunakan <i>RecyclerView</i> pada <i>CardView</i>	[V] Diterima [] Ditolak
Data pengguna yang telah <i>login</i> ditampilkan pada halaman <i>profile</i> menggunakan <i>SharedPreferences</i>	Data pengguna yang telah <i>login</i> berhasil ditampilkan pada halaman <i>profile</i> menggunakan <i>SharedPreferences</i>	Data pengguna yang telah <i>login</i> berhasil ditampilkan pada halaman <i>profile</i> menggunakan <i>SharedPreferences</i>	[V] Diterima [] Ditolak

Tabel 1 merupakan pengujian yang dilakukan oleh peneliti dengan divisi *Information Technology (IT)* PT. Jasa Raharja. Hasil pengujian yang dilakukan

menghasilkan bahwa aplikasi dapat berjalan dengan baik.

Tabel 2. Hasil Uji Coba Skala Likert

Kode Pertanyaan	Pertanyaan	Jawaban					Jumlah
		A	B	C	D	E	
1	Apakah aplikasi informasi kecelakaan berbasis Android membantu dalam melihat informasi kecelakaan ?	-	-	4	11	9	84
2	Apakah aplikasi informasi kecelakaan berbasis Android membantu dalam melaporkan kecelakaan ?	-	-	3	10	11	86
3	Apakah aplikasi informasi kecelakaan berbasis Android mempermudah pengguna untuk melihat informasi kecelakaan ?	-	-	2	12	10	86
4	Apakah aplikasi informasi kecelakaan berbasis Android mempermudah pengguna untuk melaporkan kecelakaan ?	-	-	3	12	9	85
5	Apakah anda merasa puas dengan menggunakan aplikasi informasi kecelakaan berbasis Android ?	-	-	2	14	8	85

Tabel 2 merupakan uji coba yang dilakukan kepada calon pengguna aplikasi menggunakan perhitungan skala Likert. Pada pengujian menggunakan skala Likert bertujuan untuk memperoleh kelayakan dan kepuasan aplikasi terhadap calon pengguna. Pengujian dilakukan dengan memberikan pertanyaan yang berkaitan dengan kepuasan pengguna terhadap aplikasi. Hasil dari responden akan ditambahkan ke dalam tabel uji coba dan akan digunakan untuk mengambil kesimpulan [15].

$$\sum(T \times P_n)$$

Keterangan :

T = total jumlah responden

P_n = pilihan angka skor Likert

Tabel 3. Keterangan Skala Likert

Jawaban	Skor
Sangat Tidak Setuju	1
Tidak Setuju	2
Cukup	3
Setuju	4
Sangat Setuju	5

Berdasarkan hasil dari pengujian menggunakan skala Likert, maka dapat diperoleh jumlah nilai pada tabel uji coba dengan mengalikan total jumlah responden dengan angka skor Likert diperoleh hasil $24 \times 5 = 120$

Tabel 4. Keterangan Nilai Skor Likert

No Pertanyaan	Skor
1	$(101/120) \times 100\% = 84\%$

$$2 \quad (104/120) \times 100\% = 86\%$$

$$3 \quad (104/120) \times 100\% = 86\%$$

$$4 \quad (102/120) \times 100\% = 85\%$$

$$5 \quad (102/120) \times 100\% = 85\%$$

Setelah mendapatkan hasil persentase, maka perlu dilakukan perhitungan rata-rata persentase dari skor yang telah diperoleh. Hasil yang diperoleh adalah $(84+86+86+85+85) : 5 = 85.2\%$. Apabila dilihat dari *index range* skala Likert 85.2% memiliki keterangan Sangat Setuju.

Tabel 5. Keterangan Index Range [15]

Index Range	Keterangan
0% - 19,99%	Sangat Tidak Setuju
20% - 39,99%	Tidak Setuju
40% - 59,99%	Kurang Setuju
60% - 79,99%	Setuju
80 - 100%	Sangat Setuju

4. Kesimpulan

Berdasarkan penelitian yang dilakukan menghasilkan sebuah solusi bagi PT. Jasa Raharja serta calon pengguna aplikasi dan berpotensi untuk dilakukan pengembangan lebih lanjut. Melalui hasil yang diperoleh pada pengujian menggunakan skala Likert, dapat diambil kesimpulan bahwa aplikasi yang dirancang berguna bagi pengguna dalam melaporkan kecelakaan. Perolehan nilai skala Likert sebesar 85.2% memiliki arti bahwa informasi kecelakaan dari masyarakat dapat tersalurkan melalui aplikasi

penelitian serta memuaskan calon pengguna. Selain itu saran untuk penelitian selanjutnya diharapkan bahwa sistem yang dirancang dapat terhubung ke *database* rumah sakit dan kepolisian agar informasi kecelakaan tersebut dapat ditangani secara langsung oleh pihak yang bersangkutan.

5. Daftar Pustaka

- [1] Sinaga, E., Suwitri, S. and Mustam, M., 2014. Implementasi Instruksi Presiden No. 3 tahun 2003 Tentang Kebijakan dan Strategi Nasional Pengembangan Electronic Government Di Kabupaten Semarang. *Journal of Public Policy and Management Review*, 3(4), pp.202-210. DOI: <https://doi.org/10.14710/jppmr.v3i4.6568>.
- [2] Sobry, M.G., 2017. Peran Smartphone Terhadap Pertumbuhan Dan Perkembangan Anak. *JPGI (Jurnal Penelitian Guru Indonesia)*, 2(2), pp.24-29. DOI: <https://doi.org/10.29210/02222jpgi0005>.
- [3] Gyorödi, R., Zmaranda, D., Adrian, V.G. and Gyorödi, C., 2017. A comparative study between applications developed for Android and iOS. *International Journal of Advanced Computer Science and Applications*, 8(11).
- [4] Handrianto, Y. and Sanjaya, B., 2020. Model Waterfall Dalam Rancang Bangun Sistem Informasi Pemesanan Produk Dan Outlet Berbasis Web. *Jurnal Inovasi Informatika*, 5(2), pp.153-160.
- [5] Wahid, A.A., 2020. Analisis metode waterfall untuk pengembangan sistem informasi. *J. Ilmu-ilmu Inform. dan Manaj. STMIK*, no. November, pp.1-5.
- [6] Duha, D.L.S., Sarkum, S., Munthe, I.R. and Purnama, I., 2018. Aplikasi Berita Online Berbasis Android: Studi pada Pemerintah Kabupaten Labuhanbatu. *Jurnal Informatika: Jurnal Pengembangan IT*, 3(3), pp.380-386. DOI: <https://doi.org/10.30591/jpit.v3i3.963>.
- [7] Afrinaldi, A., Huda, A. and Kurniadi, D., 2015. Aplikasi Mobile Portal Berita Ganto. or. id Berbasis Android. *Voteteknika (Vocational Teknik Elektronika dan Informatika)*, 3(1). DOI : <https://doi.org/10.24036/voteteknika.v3i1.5167>.
- [8] Dewi, S.V., Mayuska, E.N. and Nunandar, A., 2019. Perancangan Aplikasi Portal Informasi Mahasiswa Pada Universitas Ubudiyah Indonesia Berbasis Android. *Journal of Informatics and Computer Science*, 5(2), pp.117-122.
- [9] Alfianto, T. and Anu, B., 2018. Aplikasi Diagnosa Dini Penyakit Tuberculosis Dengan Menggunakan Metode Certainty Factor. *AITI*, 15(2), pp.121-127. DOI: <https://doi.org/10.24246/aiti.v15i2.121-127>.
- [10] Fadli, M.R., 2021. Memahami desain metode penelitian kualitatif. *Humanika, Kajian Ilmiah Mata Kuliah Umum*, 21(1), pp.33-54.
- [11] Sasmito, G.W., 2017. Penerapan metode Waterfall pada desain sistem informasi geografis industri kabupaten Tegal. *Jurnal Informatika: Jurnal Pengembangan IT*, 2(1), pp.6-12. DOI: <https://doi.org/10.30591/jpit.v2i1.435>.
- [12] Wahyudin, Y. and Rahayu, D.N., 2020. Analisis metode pengembangan sistem informasi berbasis website: a literatur review. *Jurnal Interkom: Jurnal Publikasi Ilmiah Bidang Teknologi Informasi Dan Komunikasi*, 15(3), pp.119-133. DOI: <https://doi.org/10.35969/interkom.v15i3.74>.
- [13] Badrul, M., 2021. Penerapan Metode Waterfall Untuk Perancangan Sistem Informasi Inventory Pada Toko Keramik Bintang Terang. *PROSISKO: Jurnal Pengembangan Riset dan Observasi Sistem Komputer*, 8(2), pp.57-52. DOI: <https://doi.org/10.30656/prosisko.v8i2.3852>.

- [14] Kurniawan, T.B., 2020. Perancangan sistem aplikasi pemesanan makanan dan minuman pada cafetaria no caffe di Tanjung Balai Karimun menggunakan bahasa pemograman PHP Dan MySQL. *Jurnal Tikar*, 1(2), pp.192-206. DOI: https://doi.org/10.51742/teknik_informatika.v1i2.153.
- [15] Kho, D., 2018. Pengertian skala likert (likert scale) dan menggunakannya. *Diambil kembali dari Teknik Elektro: <https://teknikelektronika.com/pengertian-skala-likert-likertscale-menggunakan-skala-likert/> diakses tgl, 9.*