



PENGEMBANGAN APLIKASI MOBILE TANGGAP BENCANA DI KAB. BENGKALIS MENGGUNAKAN FRAMEWORK FLUTTER

Muhammad Fauzi¹, Agus Tedyyana², Depandi Enda³

^{1,2,3}Program Studi Rekayasa Perangkat Lunak Jurusan Teknik Informatika
Politeknik Negeri Bengkalis

^{1,2,3}Jl. Bathin Alam, Bengkalis, Riau, telp. 0823 8832 0808

e-mail: ¹fauzijuventini@gmail.com, ²agustedyyana@polbeng.ac.id, ³depandi@polbeng.ac.id

Abstrak

Kabupaten Bengkalis merupakan salah satu kabupaten di Indonesia yang terletak pada di pesisir timur Pulau Sumatera. Sepanjang tahun 2019 setidaknya terjadi 209 kejadian bencana kebakaran hutan dan lahan dengan luas areal yang terbakar sekitar 2146,2 Ha. Sebagai bagian upaya mencapai kesiapsiagaan dan ketahanan bencana, peneliti mencoba membangun sebuah aplikasi mobile yang berguna sebagai asisten yang mudah dibawa dan digunakan kapan saja dengan cakupan wilayah sekitar Pulau Bengkalis. Aplikasi mobile tanggap bencana yang akan dikembangkan memiliki fitur utama pengaduan bencana dan fitur konten mitigasi bencana. Flutter merupakan cross-platform framework, alias aplikasi yang dapat digunakan di lebih dari satu platform. Aplikasi yang dibuat dengan menggunakan Flutter dapat dijalankan baik di platform Android maupun iOS. Hal ini akan menghemat waktu sehingga tidak perlu mempelajari bahasa native yang digunakan di masing-masing platform. Dari hasil compatibility testing didapatkan bahwa pada versi Android yang diuji tampaknya sudah sesuai dengan minimum requirement yang ada yaitu dapat diinstal dan berjalan dengan minimum Android SDK 21 Lollipop 5.0 dan mampu berjalan di ukuran dan resolusi layar dengan kerapatan pixel yang berbeda-beda mulai dari 4.7 inch keatas.

Kata Kunci: Flutter, Karhutla, Pengaduan Bencana.

Abstract

Bengkalis Regency is one of the regencies in Indonesia which is located on the east coast of Sumatra Island. Throughout 2019, there were at least 209 incidents of forest and land fire disasters with a burned area of around 2146.2 hectares. As part of efforts to achieve disaster preparedness and resilience, the researchers tried to build a mobile application that is useful as an assistant that is easy to carry and use at any time with the area around Bengkalis Island. The disaster response mobile application that will be developed has the main features of disaster complaints and disaster mitigation content features. Flutter is a cross-platform framework, aka an application that can be used on more than one platform. Applications created using Flutter can be run on both the Android and iOS platforms. This will save you time, so you don't have to learn the native languages used on each platform. From the results of compatibility testing, it was found that the tested Android version appeared to be following the existing minimum requirements, namely that it could be installed and run with a minimum of Android SDK 21 Lollipop 5.0 and was able to run on screen sizes and resolutions with different pixel densities ranging from 4.7 inches and above.

Keywords: Disaster Response, Flutter, Forest and Land Fires

1. PENDAHULUAN

Kabupaten Bengkalis merupakan salah satu kabupaten di Indonesia yang terletak pada di pesisir timur Pulau Sumatera. Sebagian besar wilayah Kabupaten Bengkalis merupakan tanah organosol, yaitu jenis tanah yang banyak mengandung bahan organik atau lazim disebut dengan tanah gambut. Berdasarkan data rekapitulasi pemadaman kebakaran hutan dan lahan milik Dinas Pemadam Kebakaran Kabupaten Bengkalis, sepanjang tahun 2019 setidaknya terjadi 209 kejadian bencana kebakaran hutan dan lahan dengan luas areal yang terbakar sekitar 2146,2 Ha.

Berdasarkan PP No. 21 Tahun 2008 pasal 1 ayat 1, bencana adalah peristiwa atau rangkaian peristiwa yang mengancam dan mengganggu kehidupan dan penghidupan masyarakat yang disebabkan, baik oleh faktor alam dan/atau faktor nonalam maupun faktor manusia sehingga mengakibatkan timbulnya korban jiwa manusia, kerusakan lingkungan, kerugian harta benda, dan dampak psikologis. Dikutip dari Yanuarto dkk 2019, bencana kebakaran hutan dan lahan (karhutla) adalah keadaan pada lahan dan hutan yang dilanda api sehingga mengakibatkan kerusakan serta dampak yang merugikan [1].

Penyelesaian masalah dalam penelitian ini adalah dengan merancang sebuah aplikasi mobile tanggap bencana bagi masyarakat di Kabupaten Bengkalis yang dibangun dengan teknologi aplikasi cross-platform framework yang dapat berjalan di Android dan iOS (dalam kasus ini hanya difokuskan pada Android) dengan menggunakan framework Flutter yang dikembangkan oleh Google. Aplikasi mobile tanggap bencana yang akan dikembangkan memiliki fitur utama pengaduan bencana dan fitur konten mitigasi bencana. Flutter merupakan cross-platform framework, alias aplikasi yang dapat digunakan di lebih dari satu platform. Aplikasi yang dibuat dengan menggunakan Flutter dapat dijalankan baik di platform Android maupun iOS. Hal ini akan menghemat waktu sehingga tidak perlu mempelajari bahasa native yang digunakan di masing-masing platform [2].

Dalam penelitiannya Khristianto dkk (2015) yaitu Pengembangan Aplikasi Pemantau Gempa dan Cuaca di Indonesia Pada Smartphone Android (memanfaatkan data dari BMKG) dimana untuk mendapatkan informasi bencana alam secara cepat dan akurat diperlukan aplikasi yang mendukung mobilitas dan mudah diakses dimana saja seperti aplikasi mobile. Aplikasi yang dirancang menggunakan metode penelitian model siklus hidup pengembangan sistem (System Development Life Cycle) dengan menggunakan pendekatan berorientasi objek dengan memanfaatkan pemodelan sistem menggunakan Unified Modeling Language (UML). Kebutuhan pemantauan gempa dan cuaca dapat diakses dimana saja dan kapan saja secara realtime dengan menerapkan aplikasi monitoring gempa dan cuaca berbasis Android [3].

Penelitian selanjutnya oleh Alkadri dan Insani (2019) yaitu Perancangan Aplikasi Pelaporan Kekerasan Perempuan dan Anak Pada DPPA Prov Kalbar Berbasis Android dimana aplikasi yang dibuat dikembangkan menggunakan metode prototyping. Metode prototyping mampu memberikan evaluasi/feedback dari pengguna secara cepat terhadap aplikasi yang dikembangkan. Untuk melakukan pengaduan pada aplikasi, masyarakat melampirkan bukti kekerasan berupa gambar dan titik lokasi kekerasan terjadi. Aplikasi dibangun menggunakan framework Flutter untuk aplikasi mobile dan Laravel untuk aplikasi berbasis web lalu Restful API digunakan untuk menghubungkan kedua aplikasi tersebut [4].

Kemudian pada penelitian Muhammad dkk (2018) yaitu pengembangan sistem informasi panduan mitigasi bencana alam provinsi Sumatera Barat berbasis Android dimana pengembangan sistem yang dibuat memanfaatkan bahasa pemrograman PHP dengan framework Codeigniter untuk bagian server dan bahasa pemrograman Java pada aplikasi client. Sistem ini membantu dan memberi wawasan kepada masyarakat berupa panduan mitigasi bencana, artikel, berita, kegiatan seputar mitigasi dan layanan interaksi lapor bencana & tanya bencana serta informasi peta petunjuk lokasi BPBD dan jalur evakuasi apabila terjadi bencana. Penggunaan teknologi website di bagian server dan aplikasi platform Android sebagai aplikasi client, penulis mampu menerapkan sebuah paradigma Client-Server dalam pengembangan aplikasi dengan memanfaatkan pemodelan sistem menggunakan Unified Modeling Language (UML) [5].

Penelitian Susanty dkk (2019) yaitu aplikasi GIS menggunakan metode location based service (LBS) berbasis Android dimana berdasarkan studi kasus peneliti yaitu Kota Bandar Lampung merupakan pusat ibukota provinsi Lampung. Sebagai pusat ibukota provinsi, di Kota Bandar Lampung terdapat banyak lokasi objek wisata, rumah sakit, hotel, pusat pendidikan, restoran dan berbagai fasilitas tempat umum lainnya yang belum terjelajahi sehingga banyak wisatawan lokal maupun non-lokal yang tidak mengetahui letak lokasi tersebut. Peneliti merancang aplikasi berbasis Android dengan menerapkan location based service (LBS) yang dapat membantu dan mempermudah masyarakat dalam menemukan lokasi yang diinginkan berdasarkan jarak terdekat dengan pengguna [6].

Dalam penelitian Tofid dkk (2020) yaitu Revamp Aplikasi Teman Bumil Lebih Interaktif Dengan Pendekatan Agile dimana dalam penelitiannya peneliti mengusulkan revamp atau pembaharuan menggunakan framework Flutter yang menghasilkan aplikasi yang dapat digunakan pada platform iOS dan Android atau disebut juga hybrid sehingga lebih interaktif dan sesuai dengan kebutuhan pengguna. Flutter menyediakan tools kepada developer untuk membuat aplikasi yang cantik dan tampak profesional dan dengan kemampuan menyesuaikan segala aspek aplikasi [7].

Penelitian Ki dkk (2019) yaitu Mimic: UI Compatibility Testing System for Android Apps membahas penggunaan model pemrograman mimic sebagai sebuah sistem yang berguna dalam proses *compability testing* secara terotomatisasi untuk menguji aplikasi android di berbagai perangkat yang berbeda. Selain itu, Mimic memungkinkan pengujian paralel dengan beberapa perangkat dan dengan demikian bisa mempercepat waktu pengujian [8].

2. METODE PENELITIAN

Metode Penelitian pada penelitian ini terdiri dari 5 tahap pada metode penelitian, yakni:

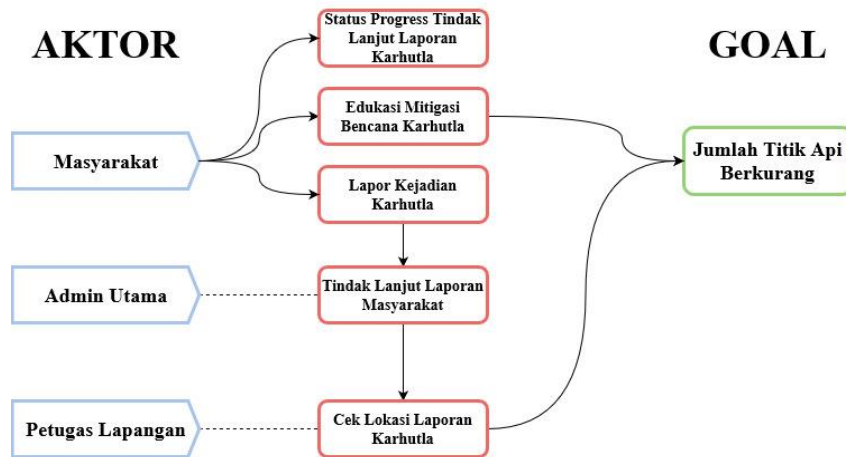
2.1. Analisis Kebutuhan

Berdasarkan hasil studi pustaka diketahui bahwa tingkat siaga dan tanggap bencana dari masyarakat sekitar Kab. Bengkalis belum tergolong tinggi. Dari data rekapitulasi pemadaman kebakaran hutan dan lahan sepanjang tahun 2019 diketahui bahwa tingkat kejadian bencana karhutla masih tergolong tinggi yaitu sebanyak 209 kejadian. Dari data yang diperoleh dari Mobile App UX Survey pada tahun 2018, responden lebih memilih menggunakan aplikasi mobile untuk gaming dan hal yang berkaitan dengan produktivitas seperti surat elektronik, catatan, dan lain-lain, sementara web hanya digunakan untuk kegiatan yang berkaitan dengan membaca. Namun secara umum sebanyak 90,3% responden dari total 2092 orang lebih memilih menggunakan aplikasi mobile daripada mengoptimalkan kegunaan aplikasi web. Atas hal inilah, peneliti mencoba mengembangkan aplikasi yang memiliki fitur pengaduan sekaligus edukasi mitigasi bencana karhutla berbasis aplikasi mobile agar masyarakat sebagai pengguna bisa turut serta tanggap akan bencana karhutla.

2.2. Analisis Kebutuhan Pengguna

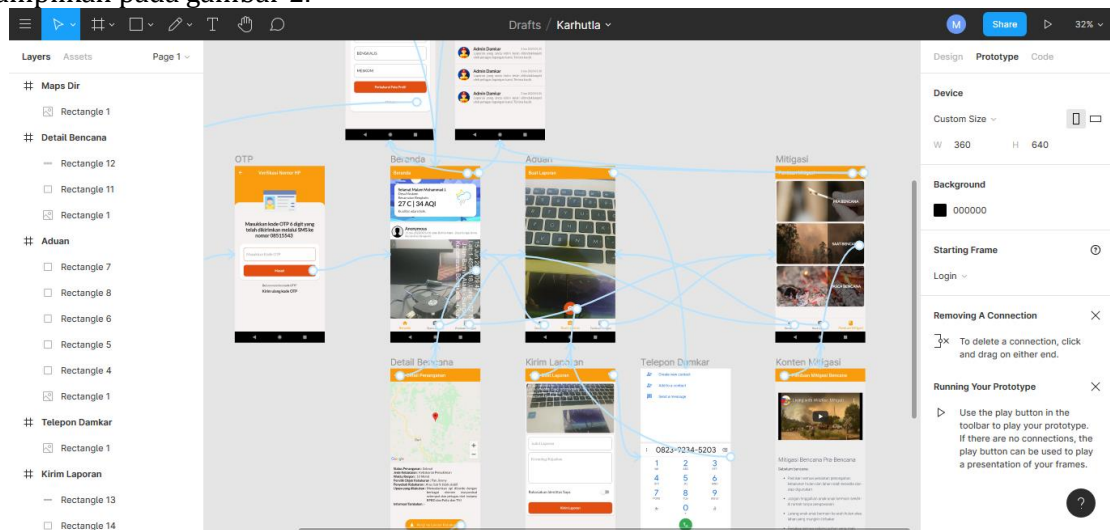
Setelah melakukan tahap analisis kebutuhan sistem selanjutnya dilakukan tahap analisis kebutuhan pengguna yang mana menggunakan metode design sprint. Pada tahap ini dilakukan beberapa fase design sprint yaitu understanding, diverge, decide, prototype, dan validation. Dikutip dari halaman web <https://designsprintkit.withgoogle.com>, Design Sprint adalah metodologi yang telah terbukti dalam proses pemecahan masalah dengan melalui tahap perancangan, pembuatan prototipe, dan menguji ide yang dirancang dengan pengguna [9].

Untuk membangun konsep sebuah produk atau jasa, design sprint memiliki lima tahapan yang mempunyai peran serta fungsi masing-masing. Kelima tahapan ini bertujuan untuk mengeluarkan semua ide, inspirasi, kreatifitas, mengatasi masalah-masalah yang ada, lalu memunculkan solusi yang kemudian diwujudkan dalam bentuk prototype yang harus dan di-cross check kepada calon pengguna. Pada gambar 1 berikut ini merupakan sprint map yang digunakan selama tahap analisis kebutuhan pengguna.



Gambar 1. Peta Sprint Proses Perancangan

Adapun hasil dari akhir dari analisis kebutuhan pengguna adalah gambaran prototipe yang telah di ujikan ke pengguna selama 2 iterasi. Berikut adalah hasil prototipe yang ditampilkan pada gambar 2.



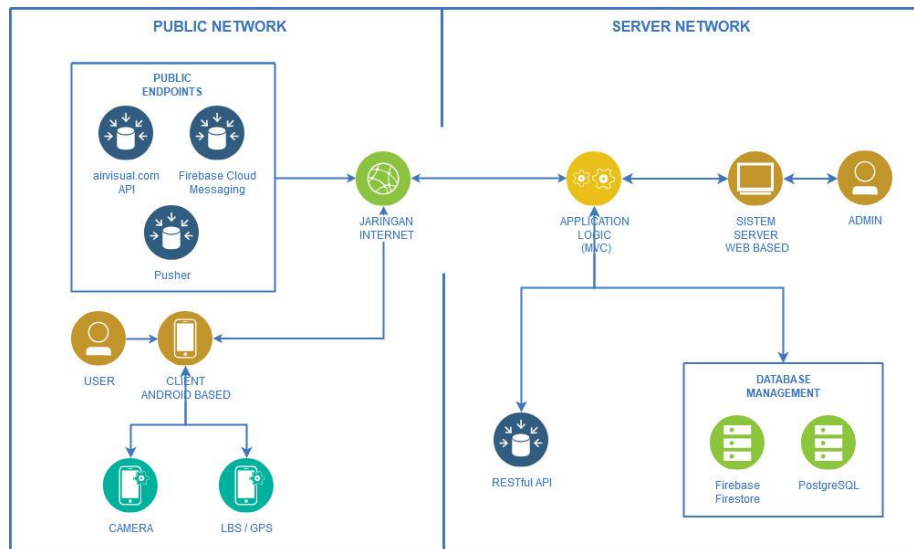
Gambar 2. Prototipe Aplikasi

2.3. Perancangan

Pada tahap ini dilakukan perancangan diagram untuk aplikasi yang akan dibuat nantinya. Rancangan yang dibuat terdiri dari usecase diagram, entity relationship diagram, firebase structure diagram, architectural context diagram dan class diagram.

2.3.1. Architectural Context Diagram

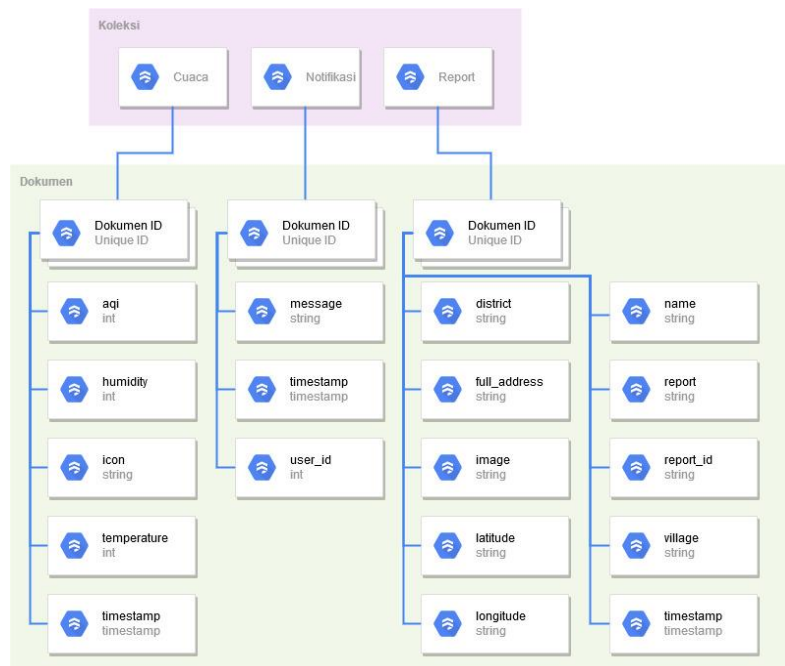
Architectural Context Diagram (ACD) digunakan untuk memodelkan cara bagaimana perangkat lunak berinteraksi dengan entitas-entitas eksternal. Architectural context diagram pada aplikasi ditunjukkan pada gambar 3 dibawah ini. Jaringan terbagi atas dua, yaitu public network yang bisa diakses oleh client dan server network yang hanya bisa diakses oleh admin.



Gambar 3. Architectural Context Diagram

2.3.2. Firebase Firestore Model Design Diagram

Firestore Model Design Diagram digunakan untuk menggambarkan rancangan model database Firebase Firestore yang akan diimplementasikan guna kebutuhan data di aplikasi client. Firestore adalah sebuah fitur Backend-as-a-service (BaaS) yang membuat proyek besar menjadi lebih mudah. Firestore menghilangkan banyak tugas yang membosankan dan membutuhkan banyak tenaga serta mempermudah membuat kode di sisi server dengan platform yang lebih aman, sederhana dan bersifat skalabilitas [10]. Rancangan ditunjukkan pada gambar 4 yang mana terdiri atas koleksi-koleksi dan dokumen. Koleksi pada cuaca berguna untuk menampilkan data cuaca yang diperbaharui secara berkala. Koleksi notifikasi untuk menyimpan notifikasi yang dikirim oleh admin ke pengguna. Dan koleksi report digunakan untuk menyimpan data pengaduan yang telah divalidasi oleh admin untuk dikirimkan ke pengguna.



Gambar 4. Firestore Model Design Diagram

2.4. Implementasi

Di tahap implementasi dilakukan proses konstruksi sebuah aplikasi berdasarkan dengan rancangan prototipe yang ada. Proses konstruksi aplikasi menggunakan bahasa pemrograman Dart dengan framework Flutter untuk membuat tampilan *frontend* yang dilakukan menggunakan *integrated development environment* Android Studio. Tampilan *frontend* nantinya akan terkoneksi dengan *backend* yang menggunakan Laravel sebagai frameworknya dan juga Firebase untuk mengirimkan notifikasi dengan memanfaatkan Firebase Cloud Messaging. Potongan kode pada gambar 3 berikut ini merupakan kode untuk proses mengirimkan pengaduan menggunakan package HTTP Dio pada flutter. Data yang dikirimkan pada proses pengaduan adalah data posisi pengguna, data alamat IP, data alamat pengguna, dan data pengaduan.

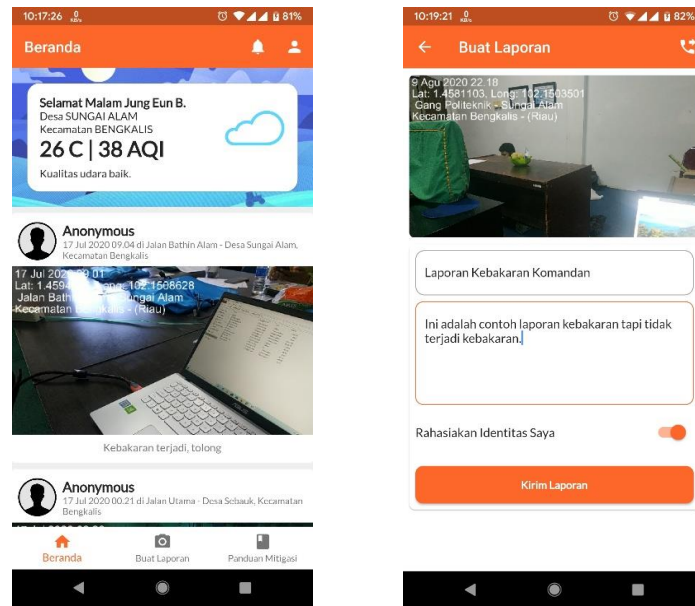
```
Future<void> postReport(context) async {
  Response response;
  Dio dio = new Dio();
  try{
    var baseUrl = Konstanta().getServerApi();
    response = await dio.post('${baseUrl}report', data: await FormDataReport());
    _showResultDialog(context, "Proses Berhasil", "Laporan telah terkirim dan akan segera ditindak
lanjuti oleh admin.", "success_report.png");
  } on DioError catch (e) {
    _showResultDialog(context, "Proses Gagal", "Terdapat kesalahan : error -
${e.response.data['error']}", "failed_report.png");
  }
}

Future<FormData> FormDataReport() async {
  Position position = await Geolocator().getCurrentPosition(desiredAccuracy:
LocationAccuracy.bestForNavigation);
  List<Placemark> p = await Geolocator().placemarkFromCoordinates(position.latitude,
position.longitude);
  Placemark place = p[0];
  var reFormat = new NumberFormat("###.0#####", "en_US");
  String latitude = reFormat.format(position.latitude).toString();
  String longitude = reFormat.format(position.longitude).toString();
  String fullAddress = '${place.subThoroughfare} ${place.thoroughfare} - Desa ${place.subLocality},
${place.locality}';
  String ipAddress = await GetIp.ipAddress;
  SharedPreferences prefs = await SharedPreferences.getInstance();
  final user_id = prefs.getInt('id_user');
  final kecamatan = prefs.getString('kecamatan');
  final desa = prefs.getString('desa');
  final token = prefs.getString('token');
  print("${_anonim.toString()} - ${latitude} - ${longitude} - ${ipAddress} - ${_kronologi} -
${_judul}");
}
```

Gambar 5. Potongan Kode Proses Kirim Pengaduan

3. HASIL DAN PEMBAHASAN

Tahap ini menampilkan hasil dari proses konstruksi yang telah dilakukan. Hasil yang ditampilkan disini merupakan fitur-fitur utama pada aplikasi. Pada gambar 4 berikut ini merupakan halaman beranda dan halaman form pengaduan. Halaman beranda merupakan halaman utama dari aplikasi Tanggap Api yang menampilkan berbagai aduan atau laporan dari masyarakat tentang kejadian bencana kebakaran hutan dan lahan. Setiap laporan dari pelapor yang masuk akan divalidasi terlebih dahulu oleh admin utama di aplikasi website. Hanya laporan yang dinyatakan valid saja yang akan tampil di screen beranda. Halaman form pengaduan muncul setelah pengguna mengambil foto laporan pada screen buat laporan. Komponen yang ditampilkan berupa pratinjau dari foto yang diambil pengguna, kolom judul laporan, kolom detail kronologi kejadian, dan switch opsi untuk merahasiakan identitas. Jika pengguna memilih untuk merahasiakan identitasnya maka daftar aduan yang muncul di beranda tidak akan menampilkan nama dari pelapor



Gambar 6. Halaman Beranda dan Halaman Form Pengaduan

Penggunaan framework Flutter sangat memudahkan peneliti dalam mengkonversi rancangan prototipe ke dalam kode. Hal ini dikarenakan widget-widget yang disediakan oleh Flutter sangat membantu dalam membuat tampilan pengguna yang cantik. Ditambah dengan dukungan komunitas yang luar biasa dengan adanya package-package yang sangat membantu dalam proses pengembangan.

Setelah melakukan pengembangan, kemudian dilakukan pengujian oleh peneliti menggunakan berbagai platform. Untuk pengujian aplikasi, peneliti menggunakan perangkat mobile Android serta Android Virtual Devices (AVD) pada Android Studio. Perangkat mobile digunakan untuk menguji apakah aplikasi sudah berjalan di berbagai lingkungan versi Android sesuai dengan rancangan dan fungsi-fungsi utama aplikasi. Sedangkan AVD digunakan untuk menguji apakah tampilan aplikasi telah responsif untuk dijalankan di berbagai perangkat Android dengan resolusi yang berbeda-beda.

Pengujian yang dilakukan menggunakan teknik compatibility testing. Compatibility testing akan difokuskan terhadap kemampuan aplikasi untuk berjalan sesuai fungsional di berbagai jenis versi sistem operasi Android dengan minimal requirement yang telah ditetapkan sebelumnya. Dalam melakukan compatibility testing peneliti melakukan langkah-langkah berikut:

1. Menyiapkan lingkungan pengujian dengan minimal requirement yang telah peneliti tetapkan. Dalam hal ini, meminjam perangkat dengan versi Android yang berbeda-beda dan membuat AVD dengan ukuran layar yang berbeda-beda di Android Studio.

2. Melakukan instalasi di perangkat yang akan di uji. Pada perangkat fisik dilakukan instalasi dari playstore, sedangkan untuk AVD dilakukan build langsung dari IDE Android Studio.

3. Melakukan pengamatan dari tiap-tiap screen atau halaman lalu mencatat hasil pengamatan

TABEL 1. Data Hasil Pengujian Kompabilitas Versi Android

No	Versi Android	Hasil Instalasi	Kesimpulan
1	SDK 18 JELLYBEAN 4.3	Gagal	Instalasi gagal dilakukan karena requirement dari aplikasi minimal SDK 21
2	SDK 19 KITKAT 4.4	Gagal	Instalasi gagal dilakukan karena requirement dari aplikasi minimal SDK 21
3	SDK 21 LOLIPOP 5.0	Berhasil	Beberapa fitur tidak berjalan sesuai fungsi, pada halaman untuk mengambil gambar dan halaman web view hanya menampilkan layar putih. Pada halaman detail laporan map tidak tampil. Kondisi image yang digunakan untuk emulator tidak terpasang Google Apps Service.
4	SDK 22 LOLIPOP 5.1	Berhasil	Semua fitur berhasil berjalan di AVD dengan Google Apps Service, namun saat dicoba di beberapa perangkat fisik dengan tipe yang sama yaitu Oppo A37, tidak bisa masuk ke halaman utama karena dideteksi sistem operasi yang digunakan telah dimodifikasi. Meski sebenarnya tidak dimodifikasi.
5	SDK 23 MARSHMALLOW 6.0	Berhasil	Semua fitur berhasil berjalan di AVD dengan Google Apps Service.
6	SDK 24 NOUGAT 7.0	Berhasil	Semua fitur berhasil berjalan di AVD dengan Google Apps Service.
7	SDK 25 NOUGAT 7.1.1	Berhasil	Semua fitur berhasil berjalan di perangkat fisik Xiaomi Redmi 5 dan Oppo A71.
8	SDK 26 OREO 8.0	Berhasil	Semua fitur berhasil berjalan di perangkat fisik Xiaomi Redmi 6A.
9	SDK 28 PIE 9.0	Berhasil	Semua fitur berhasil berjalan di perangkat fisik Xiaomi Redmi Note 7 dan Oppo A52 2020.
10	SDK 29 Q 10.0	Berhasil	Semua fitur berhasil berjalan di perangkat fisik Xiaomi Mi A2.
11	SDK 30 API 30	Berhasil	Semua fitur berhasil berjalan di AVD dengan Google Apps Service.

TABEL 2. Data Hasil Pengujian Kompabilitas Resolusi Layar Perangkat

No	Resolusi Layar	API	DPI	Kesimpulan
1	QVGA 240X320 2.7"	22	LDPI ~120dpi	Terdapat error widget overflow pada tampilan login, widget cuaca, dan tombol update dan logout pada halaman profil.
2	HVGA 320X480 3.2"	22	MDPI ~160dpi	Terdapat error widget overflow pada tampilan login, widget cuaca, dan tombol update dan logout pada halaman profil.
3	WQVGA 240X400 3.3"	22	LDPI ~120dpi	Terdapat error widget overflow pada tampilan login, widget cuaca, dan tombol update dan logout pada halaman profil.
4	FWVGA 480X854 3.7"	22	HDPI ~240dpi	Terdapat error widget overflow pada tampilan widget cuaca, dan tombol update dan logout pada halaman profil.
5	WVGA 480X800 4"	24	HDPI ~240dpi	Terdapat error widget overflow pada tampilan widget cuaca, dan tombol update dan logout pada halaman profil.
6	WXGA 720X1280 4.7"	22	XHDPI ~320dpi	Semua halaman tampak responsive dan bebas dari error widget overflow.

7	WVGA 480X800 5.1"	22	MDPI ~160dpi	Semua halaman tampak responsive dan bebas dari error widget overflow.
8	FWVGA 480X854 5.4"	22	MDPI ~160dpi	Semua halaman tampak responsive dan bebas dari error widget overflow.
9	NEXUS 6P 1440X2560 5.7"	24	560dpi	Semua halaman tampak responsive dan bebas dari error widget overflow.
10	PIXEL 2 1080X1920 5"	22	420dpi	Semua halaman tampak responsive dan bebas dari error widget overflow.
11	PIXEL 2 XL 1440X2880 5.99"	22	420dpi	Semua halaman tampak responsive dan bebas dari error widget overflow.
12	PIXEL 3a XL 1080X2160 6"	22	400dpi	Semua halaman tampak responsive dan bebas dari error widget overflow.
13	Foldable 1536X2152 7.3"	30	420dpi	Semua halaman tampak responsive dan bebas dari error widget overflow.

Dari hasil data pengujian yang telah dipaparkan, dapat peneliti rangkum hasil evaluasi berikut. Dimulai dari hasil compatibility testing dapat peneliti tarik kesimpulan bahwa pada versi Android yang diuji tampaknya sudah sesuai dengan minimum requirement yang ada yaitu dapat diinstal dan berjalan dengan minimum Android 5.0 Lollipop. Selain itu yang perlu diperhatikan adalah memastikan di perangkat Android itu sudah tersedia layanan Google Apps agar komponen didalam aplikasi seperti peta, kamera, dan webview berjalan dengan baik.

Selain itu yang perlu diperhatikan dari compability testing adalah seberapa responsive aplikasi ini untuk berjalan di berbagai ukuran dan resolusi layar beserta kerapatan pixel yang berbeda-beda juga. Dari data hasil pengujian yang telah dilakukan, peneliti mendapati bahwa ukuran layar yang diperlukan agar tampilan yang ditampilkan tidak mengalami kecacatan berupa overflow setidaknya harus berukuran 4.7 inch keatas. Beberapa widget yang mengalami overflow semuanya berada di halaman yang sama yaitu di halaman login, widget cuaca di halaman beranda, serta halaman update profil. Pada perbaikan selanjutnya halaman ini akan menjadi titik fokus perbaikan tampilan agar menjadi lebih responsif

4. KESIMPULAN

Berdasarkan hasil evaluasi dari keseluruhan tahap yang telah peneliti lakukan, maka dapat disimpulkan bahwa aplikasi yang dikembangkan menggunakan Flutter pada penelitian ini dapat dibuat dengan waktu yang relatif singkat, apalagi aplikasi dapat dijalankan secara hybrid yaitu menggunakan sistem operasi Android dan iOS. Dari sisi arsitektur Android dalam pengembangan belum diterapkan state management dan juga design patern, sehingga logic dan view masih bercampur di satu berkas dan mungkin akan menyulitkan pengembangan di kemudian hari. Kedepannya semoga bisa dikembangkan dengan menerapkan state management yang ada di flutter. Untuk saat ini state management yang lazim digunakan adalah Provider, Redux, Bloc / Rx, dan MobX.

DAFTAR PUSTAKA

- [1] S. P. Andri Cipto Utomo, Ignatius Toto, Satrio Theophilus Yanuarto, *Buku Saku Tanggap Tangkas Tangguh Menghadapi Bencana*, no. 48. Jakarta Timur: Pusat Data Informasi dan Humas BNPB, 2019.
- [2] Dicoding, "Menjadi Flutter Developer Expert : Deskripsi," 2019. <https://www.dicoding.com/academies/14/tutorials/1123> (accessed May 12, 2020).
- [3] T. Khristianto, S. N. Anwar, and H. Murti, "Pengembangan Aplikasi Pemantau Gempa dan Cuaca di Indonesia Pada Smartphone Android (Memanfaatkan Data Dari BMKG)," vol. 7, no. 1, 2015.
- [4] S. P. A. Alkadri and R. S. W. Insani, "Perancangan Aplikasi Pelaporan Kekerasan Perempuan Dan Anak Pada DPPA Prov Kalbar Berbasis Android," no. September, pp.

- 277–291, 2019.
- [5] F. Muhammad, A. Hadi, and D. Irfan, “Pengembangan Sistem Informasi Panduan Mitigasi Bencana Alam Provinsi Sumatera Barat Berbasis Android,” *J. Teknol. Inf. Pendidik.*, vol. 11, no. 1, pp. 27–42, 2018.
 - [6] W. Susanty, I. N. Astari, and T. Thamrin, “Aplikasi GIS Menggunakan Metode Location Based Service (LBS) Berbasis Android,” vol. 10, no. 1, pp. 53–58, 2019.
 - [7] Tofid, E. Julianto, and Y. Harjoseputro, “Revamp Aplikasi Teman Bumil Lebih Interaktif Dengan Pendekatan Agile,” vol. 4, no. 5, pp. 923–929, 2020.
 - [8] T. Ki, C. M. Park, K. Dantu, S. Y. Ko, and L. Ziarek, “Mimic: UI Compatibility Testing System for Android Apps,” *Proc. - Int. Conf. Softw. Eng.*, vol. 2019-May, pp. 246–256, 2019, doi: 10.1109/ICSE.2019.00040.
 - [9] Designsprintkit, “Share and engage with the Design Sprint Community,” 2019. <https://designsprintkit.withgoogle.com/> (accessed May 10, 2020).
 - [10] Yahiaoui H, *Firebase Cookbook*. Birmingham: Packt Publishing, 2017.



ZONasi: Jurnal Sistem Informasi

is licensed under a [Creative Commons Attribution International \(CC BY-SA 4.0\)](https://creativecommons.org/licenses/by-sa/4.0/)