



**MAGANG A - EC184918**  
**MAGANG B - EC184919**

**Google Bangkit**  
**(03 Mei 2021 s/d 09 Juni 2021)**

**Deteksi Kekerasan Otomatis Dalam Ruangan**

**Aufa Nabil Amiri      NRP 0721 17 4000 0029**

**Dosen Pembimbing**  
**Reza Fuad Rachmadi ST., MT., Ph.D**

**DEPARTEMEN TEKNIK KOMPUTER**  
**Fakultas Teknologi Elektro dan Informatika Cerdas**  
**Institut Teknologi Sepuluh Nopember**  
**Surabaya 2021**

*Halaman ini sengaja dikosongkan*



**MAGANG A - EC184918**  
**MAGANG B - EC184919**

**Google Bangkit**  
**(03 Mei 2021 s/d 09 Juni 2021)**

## **Deteksi Kekerasan Otomatis Dalam Ruangan**

**Aufa Nabil Amiri      NRP 0721 17 4000 0029**

**Dosen Pembimbing**  
**Reza Fuad Rachmadi ST., MT., Ph.D**

**DEPARTEMEN TEKNIK KOMPUTER**  
**Fakultas Teknologi Elektro dan Informatika Cerdas**  
**Institut Teknologi Sepuluh Nopember**  
**Surabaya 2021**

*Halaman ini sengaja dikosongkan*

# LEMBAR PENGESAHAN

## Deteksi Kekerasan Otomatis Dalam Ruangan

Laporan Magang ini disusun untuk memenuhi persyaratan akademik Departemen Teknik Komputer - Fakultas Teknologi Elektro dan Informatika Cerdas - Institut Teknologi Sepuluh Nopember

Tempat Pengesahan di: Surabaya  
Tanggal: 16 Juni 2021

Menyetujui,  
Dosen Pembimbing,

**Reza Fuad Rachmadi ST., MT., Ph.D**  
NIP. 19850403 201212 1 001

Mengetahui,  
Kepala Departemen Teknik Komputer FTEIC - ITS,

**Dr. Supeno Mardi Susiki Nugroho, ST.,MT.**  
NIP 19700313 199512 1 001

*Halaman ini sengaja dikosongkan*

# LEMBAR PENGESAHAN

## Deteksi Kekerasan Otomatis Dalam Ruangan

Diajukan Guna Memenuhi Sebagian Persyaratan Kurikulum  
Fakultas Teknologi Elektro dan Informatika Cerdas Program  
Sarjana Pada Departemen Teknik Komputer Institut Teknologi  
Sepuluh Nopember

Tanggal: 21 Juni 2021

Mengetahui,  
Kepala Program Google Bangkit

Adrianus Yoza Aprilio

Mengetahui,  
Pembimbing Program *Capstone* Google Bangkit

Dian Ayuningtyas

*Halaman ini sengaja dikosongkan*



# KATA PENGANTAR

Puji syukur atas kehadiran Tuhan yang Maha Esa, karena atas rahmat dan karunia-Nya penulis telah dapat menyelesaikan magang di Google Bangkit yang dilaksanakan tanggal 1 Mei 2021 sampai dengan 09 Juni 2021. Dalam penyelesaian Laporan Magang ini, kami mengucapkan terimakasih kepada berbagai pihak yang telah membantu dalam penyelesaian laporan ini :

- Bapak Dr. Supeno Mardi Susiki Nugroho, ST.,MT. selaku Kepala Departemen Teknik Komputer FTEIC-ITS.
- Dr. Diah Puspito Wulandari, S.T.,M.Sc. selaku Koordinator Magang Departemen Teknik Komputer ITS.
- Bapak Fuad Rachmadi selaku dosen pembimbing dalam magang ini.
- Dian Ayuningtyas sebagai mentor sekaligus pembimbing selama melakukan proses magang.

Magang merupakan kredit mata kuliah opsional yang digunakan sebagai konversi dalam Google Bangkit. Magang juga berfungsi sebagai pendalaman mahasiswa dalam pengaplikasian ilmu yang didapat pada saat perkuliahan ke dalam dunia kerja. Penulis menyampaikan permohonan maaf jika selama pelaksanaan magang terdapat hal yang kurang berkenan dan jikalau ada salah dalam penulisan laporan ini.

Surabaya, Juni 2021

Penulis

*Halaman ini sengaja dikosongkan*

# DAFTAR ISI

<b>LEMBAR PENGESAHAN (DEPARTEMEN)</b>	<b>v</b>
<b>LEMBAR PENGESAHAN (PERUSAHAAN)</b>	<b>vii</b>
<b>KATA PENGANTAR</b>	<b>ix</b>
<b>DAFTAR ISI</b>	<b>xi</b>
<b>DAFTAR GAMBAR</b>	<b>xiii</b>
<b>DAFTAR TABEL</b>	<b>xv</b>
<b>1 PENDAHULUAN</b>	<b>1</b>
1.1 Latar Belakang . . . . .	1
1.2 Tujuan . . . . .	2
1.3 Bentuk Kegiatan . . . . .	2
1.4 Waktu dan Tempat Pelaksanaan . . . . .	3
1.5 Metodologi Magang . . . . .	3
1.6 Sistematika Penulisan . . . . .	4
<b>2 PROFIL PROGRAM GOOGLE BANGKIT</b>	<b>5</b>
2.1 Sejarah Google Bangkit . . . . .	5
2.2 Tujuan Google Bangkit . . . . .	5
<b>3 TINJAUAN PUSTAKA</b>	<b>7</b>
3.1 Kekerasan . . . . .	7
3.2 Mobile Application . . . . .	7
3.3 Figma . . . . .	8
3.4 Firebase . . . . .	8
3.5 App Engine . . . . .	9

<b>4</b>	<b>DESAIN DAN IMPLEMENTASI</b>	<b>11</b>
4.1	Deskripsi Sistem . . . . .	11
4.2	Implementasi Alat . . . . .	12
4.2.1	Implementasi IoT . . . . .	12
4.2.2	Implementasi <i>Cloud / Server</i> . . . . .	12
4.2.3	Implementasi Aplikasi . . . . .	17
<b>5</b>	<b>KESIMPULAN DAN SARAN</b>	<b>25</b>
5.1	Kesimpulan . . . . .	25
5.2	Saran . . . . .	26
	<b>BIOGRAFI PENULIS</b>	<b>27</b>

## DAFTAR GAMBAR

4.1	Diagram <i>Framework</i> Keseluruhan Proyek . . . . .	13
4.2	Implementasi Basis Data . . . . .	14
4.3	Implementasi Desain <i>Splash Screen</i> . . . . .	18
4.4	Implementasi Desain Halaman <i>Sign In</i> . . . . .	19
4.5	Implementasi Desain Halaman <i>Sign Up</i> . . . . .	20
4.6	Implementasi Autentikasi pada Basis Data . . . . .	21
4.7	Implementasi Desain Halaman Daftar Kekerasan yang Belum Selesai . . . . .	22
4.8	Implementasi Desain Halaman Detail Kekerasan . .	23

*Halaman ini sengaja dikosongkan*

## DAFTAR TABEL

*Halaman ini sengaja dikosongkan*



# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Menurut data yang didapat dari Komisi Nasional (Komnas) Perempuan, terdapat lebih dari 8000 kasus yang sudah ditangani hanya dalam kurun waktu tahun 2020 saja. Dan yang lebih parah lagi, apabila dibandingkan dengan tahun sebelumnya, terjadi kenaikan sebesar 60% kasus. Jumlah tersebut hanya menghitung jumlah kasus yang tercatat dan diketahui, namun, menurut perkiraan Komnas Perempuan, masih banyak sekali kasus yang tidak tercatat dan diketahui karena meningkatnya jumlah kuesioner yang dikembalikan, padahal kuesioner sendiri merupakan alat bantu yang cukup vital untuk mengetahui kondisi lingkungan sekitar pada seorang perempuan atau anak.

Hal yang sama juga terjadi di dalam sekolah. Data dari unicef menunjukkan bahwa pada tahun 2018, hampir 41% anak pada umur 15 tahun pernah mengalami perundungan atau yang biasa disebut *bullying*. Mirisnya, perundungan tidak hanya terbatas dilakukan oleh sesama siswa, namun tidak jarang juga dilakukan oleh guru.

Ditambah lagi dengan kondisi pandemi seperti sekarang ini, banyak sekali orang yang beraktivitas dari rumah, dan karena perubahan yang begitu tiba-tiba, banyak membuat orang menjadi stress dan tertekan. Imbasnya, semakin meningkatkan kemungkinan terjadinya kekerasan di dalam ruangan.

Salah satu hal yang paling menyulitkan dalam menangani kekerasan pada perempuan dan anak adalah karena seringnya, kekerasan dilakukan di area tertutup sehingga tidak diketahui oleh orang lain. Tidak jarang juga korban kekerasan banyak yang tidak mau melapor kepada petugas yang terkait, hal ini dibuktikan dengan banyaknya kuesioner Komnas Perempuan yang dikembalikan.

Sebagai mahasiswa departemen Teknik Komputer, saya berupaya merealisasikan sebuah sistem yang bernama Jaga Bersama.

Jaga Bersama adalah sebuah sistem *monitoring* cerdas yang dapat mendeteksi apabila terjadi suatu kekerasan, terkhusus kekerasan di dalam ruangan. Diharapkan dengan terealisasinya sistem kami, maka dapat membantu untuk mengurangi angka kekerasan pada perempuan dan anak di Indonesia nantinya.

## 1.2 Tujuan

Tujuan dari magang ini dapat dilihat dari dua sudut pandang sebagai berikut:

### 1. *Secara Umum*

- (a) Terciptanya suatu hubungan yang sinergis dan terarah antara dunia perguruan tinggi dan dunia kerja sebagai pengguna outputnya.
- (b) Membuka wawasan mahasiswa agar dapat mengetahui dan memahami aplikasi ilmunya di dunia industri.
- (c) Mahasiswa dapat mampu mengadakan pendekatan masalah secara utuh.
- (d) Menciptakan pola berpikir konstruktif yang lebih berwawasan bagi mahasiswa.

### 2. *Secara Khusus*

- (a) Memperdalam pengetahuan mahasiswa dengan mengenal dan juga mempelajari secara langsung mengenai pengolahan *mobile application development* dan IoT secara umum.
- (b) Mengembangkan pengetahuan, sikap, keterampilan, kemampuan profesi melalui penerapan ilmu, latihan kerja, dan pengamatan teknik yang akan diterapkan.

## 1.3 Bentuk Kegiatan

Dalam proses magang, kami melaksanakan di rumah masing-masing (*Work From Home*) dikarenakan program yang dilakukan di seluruh Indonesia secara serentak sehingga memang harus dilakukan secara jarak jauh melalui media Google Meet.

Pembuatan sistem kami kerjakan sesuai dengan *timeline* yang sudah kami sepakati dan kami mengerjakan sesuai peran masing-

masing. Adapun bentuk teknis kegiatannya adalah sebagai berikut :

1. Penyusunan laporan sementara yang dilakukan pada minggu terakhir kegiatan magang dilaksanakan. Selanjutnya laporan sementara ini dimintakan persetujuan kepada pembimbing magang yang bersangkutan. Dalam memberikan persetujuan terhadap laporan sementara ini maka pembimbing lapangan memberikan penilaian terhadap pelaksanaan magang yang dilaksanakan oleh mahasiswa. Laporan sementara ini selanjutnya dijadikan dasar untuk pembuatan laporan resminya. Penilaian dari pembimbing lapangan akan diserahkan pada dosen pembimbing magang di Departemen Teknik Komputer - ITS Surabaya.
2. Penilaian kedua diberikan berdasarkan hasil penyusunan laporan resmi yang dibuat oleh mahasiswa selesai dilaksanakannya magang.
3. Penilaian ketiga diberikan berdasarkan hasil presentasi laporan resmi magang.

## 1.4 Waktu dan Tempat Pelaksanaan

Berikut tempat dan waktu pelaksanaan magang yang dilakukan pada :

Tempat : Rumah Masing - Masing (WFH)  
Waktu : 03 Mei 2021 s/d 09 Juni 2021

## 1.5 Metodologi Magang

Metodologi magang yang digunakan dalam pelaksanaan magang ini adalah sebagai berikut :

### 1. Tahap Persiapan

Membuat *timeline* beserta proposal untuk kemudian akan dicek dan diberi kritik atau saran oleh tim Bankgit.

### 2. Pengembangan Sistem

Sistem dikembangkan sesuai ide awal yang sudah tertulis di dalam proposal yang sudah disetujui. Selain itu juga dilakukan *mentoring* dengan orang - orang dari industri.

### 3. Pembuatan Laporan

Pembuatan laporan yang dibutuhkan sebagai prasyarat kelulusan dalam mata kuliah magang.

## 1.6 Sistematika Penulisan

Laporan Kerja Praktek ini terdiri atas lima bab dengan rincian sebagai berikut :

1. **Bab I Pendahuluan**

Pada BAB I dibahas mengenai latar belakang, tujuan, waktu dan tempat pelaksanaan Kerja Praktek, metode penulisan, serta sistematika penulisan.

2. **Bab II Profil Google Bangkit**

Pada BAB II dibahas mengenai profil singkat dari program Google Bangkit.

3. **Bab III Tinjauan Pustaka**

Pada BAB III dibahas mengenai teori-teori penunjang dalam pembuatan produk aplikasi, seperti *mobile application*, *figma*, *firebase* dan *App Engine*.

4. **Bab IV Desain dan Implementasi**

Pada BAB IV dibahas mengenai pembuatan produk.

5. **Bab V PENUTUP**

Pada BAB V dibahas mengenai kesimpulan dan saran.

## BAB II

# PROFIL PROGRAM GOOGLE BANGKIT

### 2.1 Sejarah Google Bangkit

Google Bangkit adalah sebuah program yang digagas oleh google. Diawali pada tahun 2020 dengan berisi 300 peserta, Google Bangkit berhasil meraih perhatian banyak orang termasuk, menteri pendidikan saat ini, yaitu Nadiem Makarim. Banyak sekali alumni - alumninya yang mengatakan bahwa dengan mengikuti program Google Bangkit ini, mereka mendapatkan prospek kerja yang lebih baik sehingga meningkatkan kemungkinan untuk mendapatkan pekerjaan.

Pada tahun 2021 ini, Google Bangkit menerima peserta dengan jumlah yang jauh lebih banyak, yaitu sebesar 3000 peserta. Berbeda dengan Google Bangkit pada tahun 2020, tahun ini terdapat 3 fokus-an yang dapat dipilih oleh peserta, terdapat *Google Cloud Engineer*, *Android Engineer* dan *Machine Learning Engineer*. Tidak berhenti sampai disana, saat ini Google Bangkit berkerja sama dengan 15 partner universitas dan bahkan berkolaborasi dengan perusahaan - perusahaan *unicorn* seperti Gojek, Tokopedia, dan Traveloka.

Tentu saja disini sebagai suatu perusahaan yang sangat ternama, Google tidak hanya mengambil peserta dari kota - kota besar. Menurut William Florence, Pemimpin *Asia Pacific Education Program*, bahwa terdapat 3000 peserta yang berasal dari 250 universitas dari segala penjuru Indonesia. Selain itu, terdapat 30% peserta perempuan, yang apabila dibandingkan dengan tahun lalu, terjadi peningkatan sebesar 4%.

### 2.2 Tujuan Google Bangkit

Tujuan utama dari program Google Bangkit yang utama adalah meningkatkan kualitas manusia Indonesia sehingga menjadi manu-

sia yang kreatif, terampil dan ahli dalam bidangnya masing - masing.

Selain itu, melalui program ini, diharapkan mahasiswa dapat membangun kepercayaan diri dalam berekreasi. Hal ini termasuk dengan kemampuan bertindak dan kerangka pikir kewirausahaan agar dapat mengatasi tantangan global dan membangun masa depan yang lebih baik.

Dan apabila kita melihat lebih dalam apa saja yang disuguhkan oleh Google Bangkit, dapat dilihat bahwa mahasiswa juga diharapkan untuk mengasah ketrampilan penting yang bermanfaat dalam dunia kerja, seperti *design thinking*, kepemimpinan, serta kemampuan komunikasi dan presentasi.

## BAB III

### TINJAUAN PUSTAKA

#### 3.1 Kekerasan

Kekerasan merupakan kata yang berasal dari bahasa Latin *violentus* yang berarti berkuasa atau kekuasaan. Secara arti, kekerasan merupakan sebuah ekspresi baik yang dilakukan secara fisik ataupun secara verbal yang mencerminkan pada tindakan agresi dan penyerangan pada kebebasan seseorang.

Terdapat beberapa kekerasan yang dapat dibagi berdasarkan kategori - kategori tertentu, salah satu kategori - kategori tersebut adalah kekerasan yang dilakukan perorangan, atau kekerasan yang dilakukan oleh kelompok.

#### 3.2 Mobile Application

*Mobile Application* adalah program atau perangkat lunak yang dibuat untuk perangkat-perangkat bergerak seperti : *Smartphone*, *SmartWatch*, *Tablet*, dan lainnya. Perangkat lunak atau disebut juga *software* aplikasi merupakan hasil dari pemrograman *mobile* yang dirancang menggunakan bahasa pemrograman tertentu. Banyak sekali keunggulan yang bisa didapatkan saat menggunakan aplikasi *mobile* dibandingkan dengan aplikasi *web* dan *desktop*, di antaranya yaitu:

1. *User Interface* dan *User Experience* (*UI/UX*) aplikasi seluler biasanya cukup menarik dan sangat mudah digunakan.
2. Ada beberapa aplikasi yang bisa digunakan tanpa harus terkoneksi internet.
3. Pengguna dapat mengakses aplikasi dimana saja melalui perangkat yang dimiliki.

Aplikasi yang tidak *terinstall* biasanya tersedia melalui *platform* distribusi yang disebut "toko aplikasi". Mereka mulai muncul pada tahun 2008 dan biasanya dioperasikan oleh pemilik sistem ope-

rasi *mobile*, seperti Apple App Store, Google Play, Windows Phone Store, dan BlackBerry App World. Namun, ada juga toko-toko aplikasi independen, seperti Cydia, GetJar dan F-Droid. Beberapa aplikasi sifatnya gratis, sementara yang lain harus dibeli. Biasanya, mereka diunduh dari *platform* tersebut ke perangkat target, tetapi terkadang mereka dapat diunduh ke laptop atau komputer *desktop*. Untuk aplikasi berbayar, umumnya sebesar 20-30% untuk penyedia distribusi (seperti iTunes), dan sisanya masuk ke produsen aplikasi. Oleh karena itu, aplikasi yang sama dapat dikenakan biaya yang berbeda tergantung pada *desktopnya*.

Penggunaan aplikasi seluler menjadi semakin lazim bagi para pengguna ponsel. Sebuah studi comScore pada Mei 2012 melaporkan bahwa selama kuartal sebelumnya, pelanggan perangkat *mobile* lebih banyak menggunakan aplikasi dibanding menjelajahi *web* di perangkat mereka: masing-masing 51,1% vs 49,8%. Para peneliti menemukan bahwa penggunaan aplikasi *mobile* sangat berkorelasi dengan konteks pengguna dan bergantung pada lokasi dan waktu pengguna.

### 3.3 Figma

Figma adalah perangkat lunak yang digunakan oleh para desainer aplikasi *mobile* maupun desainer *web* untuk mendesain suatu UI / UX dan bersifat *web-based* dimana maksudnya adalah Figma dapat diakses dimana saja selama terdapat *browser* dan koneksi internet. Fitur - fitur utama dari Figma adalah kemudahan dalam melakukan desain UI maupun UX, dan memiliki kemampuan untuk melakukan kolaborasi secara *real-time* sehingga dapat mempermudah tim yang berkolaborasi secara *remote*. Selain itu, sudah disediakan banyak sekali *template* sehingga sangat memudahkan dalam membuat tampilan yang bagus dan menarik dari awal.

### 3.4 Firebase

Firebase adalah *platform* yang dibuat oleh Google yang bertujuan untuk memudahkan pengembang perangkat lunak dalam membuat aplikasi gawai dan *web*. Terdapat banyak sekali fitur - fitur yang ditawarkan seperti *Cloud Firestore* yang merupakan basis data



NoSQL yang ditawarkan oleh Google, *Authentication* yang berguna untuk melakukan autentikasi secara mudah dan sudah mendukung beberapa *platform* seperti Email dan Password, Google SignIn, Facebook SignIn, dan masih banyak lagi, kemudian terdapat *Cloud Messaging* yang digunakan untuk mengirimkan notifikasi kepada *device* seperti Android, maupun IOS. Dan masih banyak lagi fitur - fitur yang ditawarkan pada Firebase ini.

### 3.5 App Engine

App Engine adalah *platform* dari Google yang berada dibalik bendera *Google Cloud Platform (GCP)*. *Google Cloud Platform* sendiri adalah gabungan dari beberapa *cloud service* yang dimiliki oleh Google dan *server* nya sudah tersebar di berbagai negara di dunia.

App Engine sendiri adalah sebuah *serverless platform* yang bersifat *fully managed*. Yang dimaksud sebagai *fully managed* adalah ketika *server* atau *service* sepenuhnya diurus oleh pihak Google sehingga kita sebagai pengembang perangkat lunak hanya perlu fokus pada perangkat lunak yang kita buat dan tidak terganggu oleh hal - hal lain seperti konfigurasi *server*, memilih *database*, dan hal - hal yang menyangkut infrastruktur lainnya. Selain itu, fiturnya sendiri sudah cukup lengkap, seperti *auto-scaling* yaitu fitur yang berguna apabila pengguna perangkat lunak kita meningkat maka akan secara otomatis App Engine akan menaikkan model *server* yang digunakan, begitu juga sebaliknya, saat pengguna perangkat lunak kita turun maka akan secara otomatis kita akan 'dipilihkan' konfigurasi yang paling kecil dan bahkan bisa saja *service* kita dimatikan sementara sampai ada yang mengakses perangkat lunak kita.

*Halaman ini sengaja dikosongkan*

## BAB IV

# DESAIN DAN IMPLEMENTASI

### 4.1 Deskripsi Sistem

Sistem kami bernama JagaBersama, suatu sistem yang dapat digunakan untuk mendeteksi kekerasan baik itu merupakan kekerasan dalam rumah tangga maupun *bullying* atau perundungan yang biasa terjadi di sekolah - sekolah maupun di tempat umum. Dengan memanfaatkan salah satu cabang dalam pembelajaran mesin yaitu *Deep Learning* ditambah dengan pengolahan citra modern, kita dapat melakukan pendeteksian secara otomatis dengan cepat dan murah. Diharapkan dengan adanya sistem kami maka dapat mengurangi jumlah kasus kekerasan di Indonesia dan dapat memberikan suatu ide pendekatan yang baru dalam menanggulangi kekerasan di Indonesia.

Terdapat 2 target pengguna dalam sistem kami, yang pertama adalah Pengguna Umum yang akan memasang sistem kami (kamera + *raspberrypi*) yang akan dapat secara otomatis mendeteksi kekerasan. Kemudian target pengguna yang pertama adalah Agen, agen disini yang dimaksud adalah agen yang dapat berasal dari Komnas Perempuan yang bertugas untuk menyelesaikan apabila terjadi suatu kasus kekerasan maupun bisa juga agen lain seperti polisi atau guru apabila kekerasan terjadi di sekolah.

Namun, sebagai catatan, tanggung jawab saya kebanyakan merupakan pada bagian membuat aplikasi Android yang digunakan sebagai antar-muka antar agen dengan sistem basis data secara keseluruhan.

Seluruh kode yang digunakan untuk melakukan implementasi pada proyek ini dapat diakses pada tautan [https://github.com/A0040325/capstone\\_app](https://github.com/A0040325/capstone_app) untuk implementasi pada bagian aplikasi Android, tautan [https://github.com/A0040325/capstone\\_cloud](https://github.com/A0040325/capstone_cloud) untuk implementasi pada bagian *Cloud* dan *Server*, dan tautan [https://github.com/A0040325/capstone\\_ml](https://github.com/A0040325/capstone_ml) untuk implementasi

pada bagian pembelajaran mesin. Keseluruhan projek dapat diakses oleh publik secara gratis dan tanpa batasan apapun.

## 4.2 Implementasi Alat

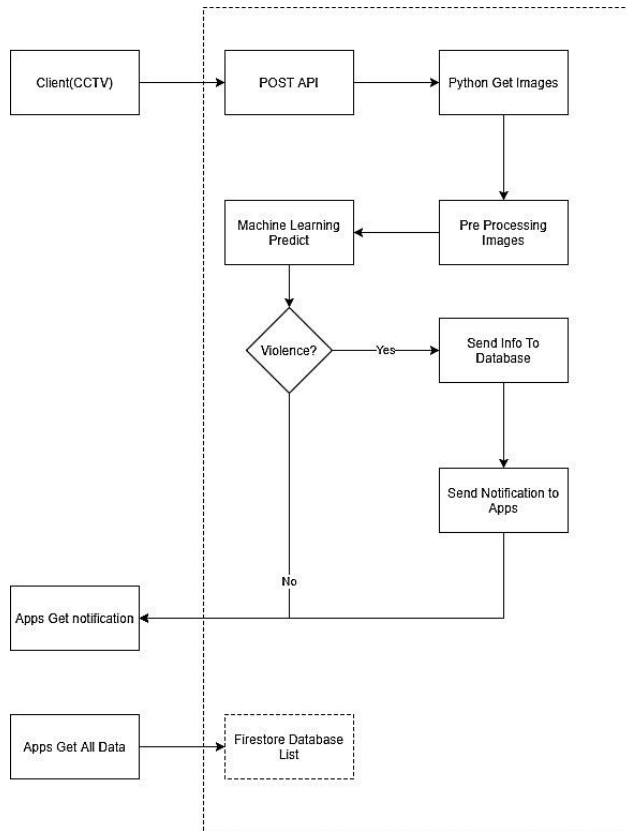
Terdapat 3 bagian terpisah dari sistem kami yang semuanya bekerja secara bersamaan sehingga menciptakan suatu sistem yang dapat berjalan dengan lancar dan *reliable*. Bagian pertama adalah bagian *Internet of Things* (IoT), selanjutnya adalah *server* sebagai bagian paling penting dalam keseluruhan sistem dan terakhir adalah aplikasi Android sebagai antar-muka dengan agen. Gambar 4.1 adalah gambaran secara luas bagaimana sistem kami bekerja secara keseluruhan.

### 4.2.1 Implementasi IoT

Dalam projek ini kami berencana untuk menggunakan 2 tipe alat, yang pertama adalah menggunakan CCTV ditambah dengan Raspberry Pi, yang kedua adalah menggunakan CCTV ditambah dengan Jetson Nano yang notabene lebih mahal dibanding alat pertama. Alasan kami memiliki 2 alat adalah kami ingin memberikan pengguna suatu pilihan. Apabila menggunakan Raspberry Pi maka nantinya Raspberry Pi akan mengirimkan foto ke *server* kami setiap beberapa detik sekali sehingga dari sisi *privacy*, hal ini cukup mengkhawatirkan karena walaupun *server* kami sama sekali tidak menyimpan foto dalam jangka waktu yang lama kecuali apabila dalam foto tersebut terdeteksi adanya kekerasan, namun seseorang dapat meluncurkan *Man in The Middle Attack* (MITM), yang berarti seseorang 'menguping' seluruh permintaan koneksi dari *server* ke Raspberry Pi. Alat kedua memiliki tingkat *privacy* yang lebih baik, karena Jetson Nano memiliki kemampuan untuk melakukan deteksi secara *on-device* sehingga hanya mengirimkan data atau foto apabila terdapat kekerasan terdeteksi, namun memerlukan biaya pemasangan yang lebih tinggi.

### 4.2.2 Implementasi *Cloud* / *Server*

*Cloud* atau *Server* merupakan salah satu komponen yang paling penting dalam projek kami, selain sebagai tempat menyimpan data, *server* juga digunakan sebagai media pemroses foto yang di-



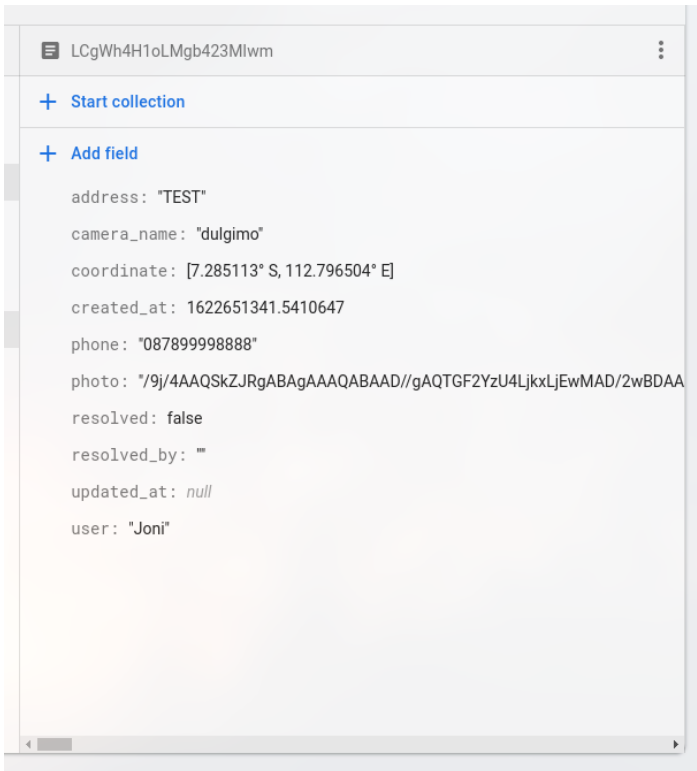
Gambar 4.1: Diagram *Framework* Keseluruhan Proyek

kiriman oleh Raspberry Pi setiap beberapa detik sekali. Sebagai media pemrosesan kami memilih untuk menggunakan App Engine, karena salah satu fitur utamanya yaitu *serverless setup* sehingga kami hanya perlu menyiapkan kode untuk pendeteksi fotonya saja menggunakan teknik pembelajaran mesin.

Untuk teknik pembelajaran mesinnya kami menggunakan teknik *3D Convolutional Neural Network* (3D-CNN), sebuah teknik yang merupakan pengembangan dari CNN biasa dengan cara me-

nambahkan kemampuan untuk melihat tidak secara 1 *frame* atau gambar, melainkan beberapa *frame* sekaligus sehingga sangat cocok digunakan untuk tipe data video yang bersifat sekuensial seperti misalnya untuk mendeteksi orang terjatuh, mendeteksi orang berkelahi sampai mendeteksi kekerasan.

Semua itu kami gabungkan dengan Flask sebagai *webserver* dengan bahasa Python sehingga memudahkan saat implementasi karena hanya memerlukan 1 bahasa saja. Flask kami gunakan sebagai *endpoint* yang dapat diakses oleh Raspberry Pi untuk mengirimkan potongan video setiap beberapa detik sekali.



Gambar 4.2: Implementasi Basis Data

Sebagai media penyimpanan, kami memutuskan untuk menggunakan Firebase, selain karena fiturnya yang lengkap, Firebase sudah menyediakan SDK baik untuk perangkat Android maupun untuk Python sehingga sangat mempermudah dalam melakukan pengembangan sistem. Gambar 4.2 adalah struktur basis data yang kami gunakan. Terdapat beberapa isian dengan rincian sebagai berikut :

- *address*

Menyimpan data alamat pengguna tempat sistem kami dipasang dan dijalankan. Nilai ini didapat pada saat pemasangan karena setiap alat dari kami sudah memiliki nilai identifikasi masing - masing sehingga memudahkan untuk melakukan pengecekan maupun untuk mengetahui dari kamera mana kekerasan terdeteksi.

- *camera\_name*

Menyimpan nomor identifikasi kamera, berguna untuk mengetahui dari kamera mana foto yang terdeteksi tersebut berasal.

- *coordinate*

Menyimpan data koordinat tempat kekerasan terjadi. Nilai ini didapat dari sistem melalui pengaturan pada saat pemasangan. Fungsi koordinat ini adalah sebagai informasi tambahan dari isian *address* yang sudah disebutkan sebelumnya. Selain itu juga sebagai sumber data yang digunakan untuk menampilkan peta.

- *created\_at*

Menyimpan waktu kapan data dimasukkan ke dalam basis data.

- *phone*

Menyimpan informasi berupa nomor telepon pengguna yang memasang sistem kami. Seperti nilai koordinat, *camera\_name* dan *address*, nilai dari *phone* ini nantinya akan diatur pada saat memasang sistem kami pertama kali.

- *photo*

Adalah isian yang digunakan untuk menyimpan foto yang menunjukkan gambar ketika terjadinya kekerasan. Berformat *base64* untuk memudahkan dalam menyimpan data sekaligus mengurangi penggunaan media penyimpanan.

- *resolved*

Merupakan penanda apakah suatu kasus statusnya sudah selesai atau masih dalam pengerjaan. Apabila nilai dari isian ini sudah berisi *True*, maka berarti kasus ini sudah selesai dan dimasukkan ke dalam *history*, begitu juga sebaliknya, apabila nilai masih berisi *False*, maka kasus belum selesai dan masih menunggu untuk ditangani.

- *resolved\_by*

Digunakan untuk menyimpan nomor identifikasi agen yang menyelesaikan kasus tersebut, aplikasi menggunakan nilai dari *resolved\_by* dan status *resolved* sebelum memutuskan bahwa kasus sudah selesai dan dimasukkan ke dalam *history*. Apabila status *resolved* masih berisi *False* namun *resolved\_by* sudah memiliki isi, berarti kasus sedang dalam tahap pengerjaan oleh agen yang nomor identifikasinya tersimpan di isian ini.

- *updated\_at*

Menyimpan data tentang kapan suatu kasus mengalami pembaruan, bisa berarti perubahan status kasus yang awalnya *False* menjadi *True* atau perubahan agen yang mengerjakan kasus tersebut. Selain itu dapat juga digunakan untuk mengetahui kapan kasus diselesaikan.

- *user*

Berguna untuk menyimpan nama pengguna yang memasang sistem kami. Seperti data nomor telepon dan beberapa data lainnya, nilai ini diatur pada saat melakukan pemasangan pertama kali.



### 4.2.3 Implementasi Aplikasi

Aplikasi disini digunakan sebagai antar-muka antara agen dengan sistem basis data yang digunakan pada sistem kami. Hal yang pertama kami lakukan adalah melakukan implementasi pada bagian tampilan, disini target utama kami adalah membuat tampilan yang mudah dipahami, serta gampang untuk digunakan.

Gambar 4.3 adalah tampilan pada halaman *splash*. *Splash* disini berguna selain untuk melakukan inisialisasi basis data untuk melihat apakah pengguna sudah melakukan *login* sebelumnya, juga digunakan untuk membuat aplikasi tampak lebih cepat dalam dijalankan.

Gambar 4.4 adalah tampilan pada halaman *sign in*. Tampilan ini adalah yang kedua dilihat setelah memasuki halaman *splash*. Dimana pengguna dapat melakukan *login* untuk mengakses fitur aplikasi lebih lanjut lagi.

Gambar 4.5 adalah tampilan pada halaman *sign up*. Tampilan ini adalah halaman yang akan dilihat saat suatu agen akan membuat akun baru. Terdapat beberapa *field* yang harus diisi apabila akan mendaftar akun baru, yaitu *username*, *email*, nomor telepon dan *password*. *Username*, surel dan *password* digunakan sebagai cara untuk melakukan *login*.

Untuk membantu dalam hal melakukan autentikasi, kami menggunakan Firebase Authentication. Firebase Authentication ini dipilih karena selain lebih mudah digunakan dan sudah memiliki fitur keamanan yang cukup. Seperti bisa dilihat pada gambar 4.6, kode sandi yang dimasukkan ke dalam basis data akan secara otomatis dilakukan proses enkripsi sehingga data yang dimasukkan menjadi lebih aman dan hanya pengguna saja yang tahu apa kode sandi yang digunakan.

Gambar 4.7 adalah tampilan yang akan paling sering dilihat oleh agen pada saat menggunakan aplikasi yang kami buat, dengan kata lain, halaman ini adalah halaman utama aplikasi kami. Halaman utama ini digunakan untuk melakukan daftar berbagai kasus kekerasan yang belum selesai atau belum ditangani. Nantinya, apabila agen sudah setuju untuk menanganinya, maka agen hanya perlu menekan tombol 'TERIMA', sistem akan otomatis mencatat nama



Gambar 4.3: Implementasi Desain *Splash Screen*

agen tersebut sebagai agen yang datang ke lokasi dan menyelesaikan kasus kekerasan yang terjadi. Saat kasus sudah selesai, maka agen harus menekan tombol 'SELESAI' pada daftar kasus di aplikasi, dengan menekan tombol tersebut, maka kasus akan dinyatakan selesai dan masuk ke dalam *history* kasus.

Apabila agen menekan pada bagian tengah pada daftar kekerasan, maka halaman akan berpindah ke halaman detail. Gambar 4.8 adalah tampilan dari halaman detail. Halaman detail ini digu-

# Welcome Back

**Sign In**



**SIGN UP**

Gambar 4.4: Implementasi Desain Halaman *Sign In*

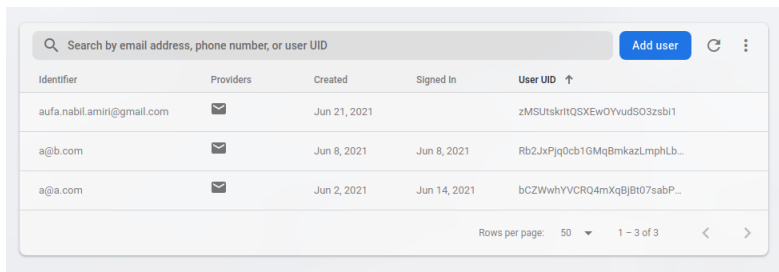
nakan untuk menunjukkan detail dari kekerasan, terdapat beberapa data diantaranya adalah foto kekerasan yang terjadi, data lengkap pengguna, tampilan peta tempat lokasi kejadian dan tombol untuk melakukan panggilan dengan mudah karena hanya memerlukan 1 kali menekan tombol tersebut saja.

# Create Your Account

**Sign Up**



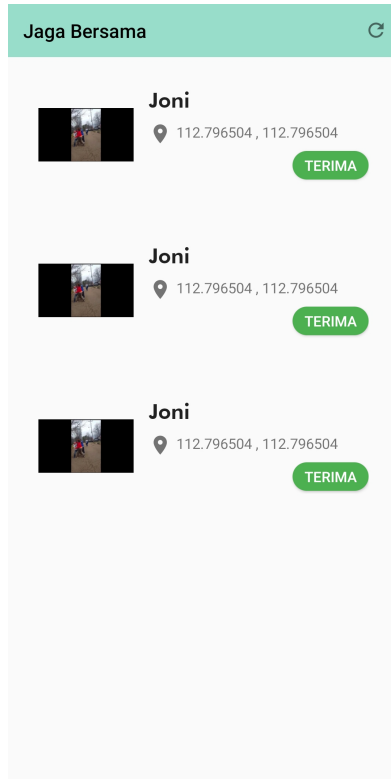
Gambar 4.5: Implementasi Desain Halaman *Sign Up*



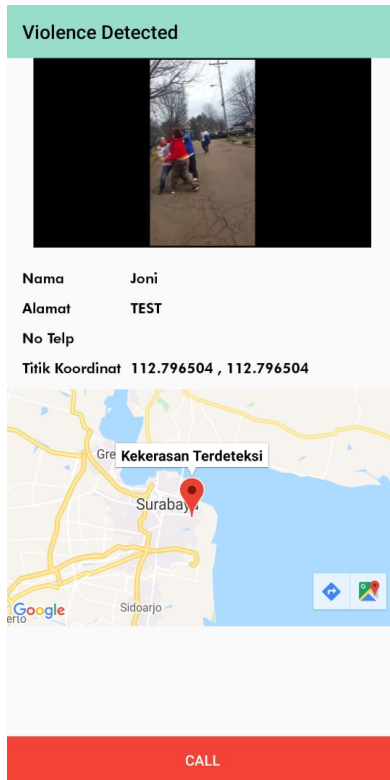
<input type="text" value="Search by email address, phone number, or user UID"/> <span>Add user</span> <span>↺</span> <span>⋮</span>				
Identifier	Providers	Created	Signed In	User UID ↑
aufa.nabil.amiri@gmail.com	✉	Jun 21, 2021		zMSUtskrttQsXEwOYvudSO3zsb1
a@b.com	✉	Jun 8, 2021	Jun 8, 2021	RbZJxFjqQcb1GMqBmkazLmphLb...
a@a.com	✉	Jun 2, 2021	Jun 14, 2021	bCZWwhYVCRQ4mXqBjBt07sabP...

Rows per page: 50 1 – 3 of 3 < >

Gambar 4.6: Implementasi Autentikasi pada Basis Data



Gambar 4.7: Implementasi Desain Halaman Daftar Kekerasan yang Belum Selesai



Gambar 4.8: Implementasi Desain Halaman Detail Kekerasan

*Halaman ini sengaja dikosongkan*



## BAB V

### KESIMPULAN DAN SARAN

Pelaksanaan Magang dalam program Google Bangkit telah kami lakukan. Kami mendapatkan ilmu dan pengalaman bekerja dalam suatu proyek kolaborasi yang mengedepankan teknologi, kreativitas dan ketekunan dalam melakukan suatu hal. Dan juga dengan terlaksananya magang ini, kami mendapatkan pengetahuan baru tentang cara pengembangan suatu sistem dari awal sampai jadi dan dapat digunakan. Terkhusus, kami mendapatkan pengetahuan baru tentang bidang pembelajaran mesin, bidang pengembangan aplikasi Android, serta dalam bidang *Cloud*. Tidak hanya bidang keilmuan yang sejalan dengan departemen kami saja, namun softskill seperti cara berpresentasi dan alur bekerja yang baik kami dapatkan saat magang kami berlangsung.

#### 5.1 Kesimpulan

Berdasarkan kegiatan Magang yang sudah kami lakukan, diperoleh beberapa kesimpulan sebagai berikut :

1. Pembuatan sistem pendeteksi kekerasan ini dapat dilakukan dengan memanfaatkan beberapa teknologi seperti App Engine dan Firebase. Selain itu, untuk pembuatan aplikasi kami menggunakan bahasa Kotlin karena lebih mudah dalam proses membuat aplikasi dan memiliki hasil *build* yang kecil, cepat dan ringan.
2. Dengan menggunakan sistem Jaga Bersama, suatu tindak kejahatan berupa kekerasan dapat langsung terdeteksi secara otomatis dan seorang agen akan dapat dengan cepat mengetahui lokasi tempat kekerasan terjadi sehingga penanganan dapat lebih cepat dilakukan.

## 5.2 Saran

Berdasarkan kegiatan Magang yang sudah kami lakukan, terdapat beberapa saran dari penulis yang dapat dijadikan masukan :

1. Adanya *maintenance* yang berkala pada aplikasi untuk memastikan keseluruhan sistem dapat berjalan dengan semestinya.
2. Memperbaiki tampilan agar aplikasi dapat lebih menarik bagi agen dan lebih mudah digunakan.
3. Basis data yang digunakan sebaiknya menggunakan sistem yang lebih baik dan memiliki tingkat keamanan yang terjaga.

## BIOGRAFI PENULIS



Aufa Nabil Amiri, lahir pada tanggal 5 Maret 2000, Surabaya. Merupakan seseorang mahasiswa yang berasal dari Institut Teknologi Sepuluh Nopember departemen Teknik Komputer. Penulis merupakan lulusan SMP Muhammadiyah 5 Surabaya dan dilanjutkan dengan SMA Negeri 2 Surabaya. Dalam masa kuliah, penulis tertarik pada bidang pengembangan *Software Development* dan Pembelajaran Mesin. Selain itu, penulis juga aktif dalam organisasi Lab B201 selama kurang 2 tahun. Penulis juga aktif dalam mengikuti kompetisi pengembangan perangkat lunak dan berhasil meraih penghargaan di ajang

GEMASTIK XII 2019. Bagi pembaca yang memiliki kritik, saran, atau pertanyaan mengenai laporan magang ini dapat menghubungi penulis melalui surel [aufa.nabil.amiri@gmail.com](mailto:aufa.nabil.amiri@gmail.com)

*Halaman ini sengaja dikosongkan*