

RANCANG BANGUN SISTEM INFORMASI PELAYANAN PRA RUMAH SAKIT MENGGUNAKAN SCRUM FRAMEWORK (STUDI KASUS STARTUP G-ONE)

**Muhamad Aji Pandean Mertayasa¹⁾ I Gede Juliana Eka Putra²⁾
Eka Grana Aristyana Dewi³⁾**

Program Studi Teknik Informatika¹⁾²⁾

Program Studi Sistem Informasi Akuntansi³⁾

Sekolah Tinggi Manajemen Informatika dan Komputer (STMIK) Primakara¹⁾²⁾³⁾

Denpasar, Bali

ajikun9@gmail.com¹⁾ gedejep@primakara.ac.id²⁾ aris@primakara.ac.id³⁾

ABSTRACT

Currently, the number of deaths due to traffic accidents in the world continues to increase every year. In 2016, 1.35 million people died due to traffic accidents. In Indonesia, especially Denpasar, cases of death or minor injuries due to traffic accidents are also quite large, namely 221 cases since 2020. The World Health Organization (WHO) in its annual report reveals that one way to reduce the number of deaths due to traffic accidents is to improve post-crash care or pre-hospital services. This research is the design and development of a pre-hospital service information system using the Scrum framework method. This research was conducted in 5 main stages, namely (1) identification of existing problems, (2) literature study related to the issues raised, (3) information system design, (4) information system development using the scrum framework method, and lastly (5) determine conclusions from the development that has been carried out. In the development phase, this research will use Flutter as a mobile framework, Golang and Fiber as an API and Vue.js as a web framework. The results of this study are expected to be a good solution to optimize pre-hospital services in order to reduce the number of deaths due to accidents in Indonesia, especially Denpasar City.

Keywords-Prehospital, Flutter, Vue.js, API, Golang, Scrum

ABSTRAK

Saat ini jumlah kematian akibat kecelakaan lalu lintas di dunia terus bertambah setiap tahunnya. Pada tahun 2016, tercatat 1,35 juta orang mengalami kematian akibat kecelakaan lalu lintas. Di Indonesia, khususnya Denpasar, kasus kematian atau luka ringan akibat kecelakaan lalu lintas juga cukup besar, yaitu 221 kasus sejak tahun 2020. *World Health Organization* (WHO) pada laporan tahunannya mengungkapkan bahwa salah satu cara untuk mengurangi angka kematian akibat kecelakaan lalu lintas adalah dengan meningkatkan pelayanan *post-crash care* atau pra rumah sakit. Penelitian ini merupakan perancangan dan pembangunan sistem informasi pelayanan pra rumah sakit menggunakan *Scrum framework*. Penelitian ini dilakukan dengan 5 tahapan utama, yaitu (1) identifikasi permasalahan yang ada, (2) studi literatur terkait dengan permasalahan yang diangkat, (3) perancangan sistem informasi, (4) pengembangan sistem informasi menggunakan *scrum framework*, terakhir (5) menentukan kesimpulan dari pengembangan yang sudah dilakukan. Pada fase pengembangan, penelitian ini akan menggunakan Flutter sebagai *mobile framework*, Golang dan Fiber sebagai API dan Vue.js sebagai *web framework*. Hasil dari penelitian ini diharapkan dapat menjadi solusi yang baik untuk mengoptimalkan pelayanan pra rumah sakit demi mengurangi angka kematian akibat kecelakaan di Indonesia, utamanya Kota Denpasar.

Kata kunci-Pra rumah sakit, Flutter, Vue.js, API, Golang, Scrum

PENDAHULUAN

Menurut *Global Status Report on Road Safety* tahun 2018 yang dirilis oleh *World Health Organization (WHO)*, kecelakaan lalu lintas di dunia terus bertambah setiap tahunnya. Pada tahun 2016, tercatat sebanyak 1.35 juta orang mengalami kematian akibat dari kecelakaan lalu lintas [1]. Hal ini membuat kecelakaan lalu lintas menempati peringkat ke-8 dalam hal penyebab kematian untuk semua kalangan umur, bahkan lebih tinggi dari HIV/AIDS, tuberkulosis dan penyakit diare. WHO mencatat bahwa tingkat rasio kecelakaan lalu lintas global di dunia saat ini mencapai 18,2 kematian per 100,000 populasi dimana Afrika dan Asia Tenggara menjadi penyumbang terbesar dalam angka tersebut, yakni masing-masing mencapai 26,7 dan 20,7 kematian per 100,000 populasi.

Kondisi kecelakaan lalu lintas di Indonesia juga cukup memprihatinkan. Pasalnya, menurut data dari *World Life Expectancy*, tingkat rasio kecelakaan lalu lintas di Indonesia mencapai 18,01 kematian per 100,000. Angka tersebut terbilang cukup banyak dibandingkan dengan negara-negara tetangga seperti Philipina (11,40 kematian), Korea Selatan (7,98 kematian), Brunei (7,68 kematian), Singapura (2,87 kematian) dan Jepang (2,37 kematian) yang relatif lebih kecil [2]. Presiden IRSP Elly Sinaga mengungkapkan bahwa, kecelakaan lalu lintas di Indonesia menjadi kontributor nomor ketiga kematian terbesar setelah jantung dan *stroke*. Elly seterusnya menjelaskan bahwa dari sekitar 147 ribu kejadian kecelakaan ini mayoritas didominasi oleh sepeda motor sebanyak 80% dan dialami paling banyak oleh anak muda sebanyak 46% rentang usia antara 15-29 tahun [3].

Di daerah Bali sendiri, jumlah kasus kematian ataupun luka ringan dari kecelakaan lalu lintas dapat dikatakan cukup besar, yaitu 221 kasus sejak tahun

2020 [4]. Dari 221 kasus tersebut, tercatat bahwa mayoritas kecelakaan lalu lintas dialami oleh kendaraan bermotor. Hal ini juga didukung oleh data dari hasil wawancara bersama dengan BPBD daerah Renon, yang mana jumlah kecelakaan yang terjadi bisa mencapai 10 kasus perminggu nya.

Seluruh negara di dunia telah memberikan upaya maksimal untuk menekan angka kematian akibat kecelakaan lalu lintas begitu juga dengan Indonesia, yaitu dengan menerapkan layanan *post-crash care* atau layanan pre hospital. Hal tersebut sesuai dengan arahan dari WHO yang dalam *Global Status Report on Road Safety* menjelaskan bahwa salah satu cara yang dapat dilakukan untuk mengurangi tingkat kematian akibat kecelakaan lalu lintas adalah dengan meningkatkan *post-crash care* atau dapat juga disebut sebagai pra rumah sakit [1]. Penerapan *post-crash care* atau pra rumah sakit ini dimulai dari penggunaan sistem panggilan darurat, pelayanan pra rumah sakit terpadu seperti pemberian perawatan terhadap korban dan pencatatan kondisi korban selama perjalanan menuju rumah sakit, transportasi dan perawatan kegawatdaruratan di rumah sakit.

Saat ini, Indonesia sudah mengimplementasikan pelayanan pra rumah sakit terpadu dengan cara mengaktifkan panggilan darurat nasional di nomor 112. Nomor darurat 112 adalah sebuah layanan yang diinisiasi oleh Kementerian Komunikasi dan Informasi untuk menyediakan *call center* gratis seperti halnya 911 di Amerika Serikat [5]. Dengan menggunakan nomor darurat 112 ini, kini masyarakat dapat melaporkan kejadian segala kejadian seperti kecelakaan, kriminal, kebakaran dan bencana alam secara cepat, dimanapun dan kapanpun selama masih berada di wilayah Indonesia. Panggilan darurat ini bebas biaya dan dapat dihubungi melalui ponsel, telepon rumah atau bahkan melalui ponsel yang tidak memiliki kartu SIM [5]. Kemudian dalam skala regional, daerah

Bali khususnya Kota Denpasar, sudah memiliki sebuah inovasi berupa aplikasi bernama Denpasar Mantap Kesehatan Masyarakat (DAMAKESMAS) yang dibuat dengan tujuan untuk meningkatkan pelayanan pra rumah sakit terpadu.

DAMAKESMAS merupakan sebuah program inovasi pelayanan kesehatan masyarakat untuk mendukung Padmaksara No. 6 yaitu mewujudkan pelayanan prima berlandaskan Sewaka Dharma dan meningkatkan akses, kapasitas, dan kualitas kesehatan dan Pembangunan Indonesia Sehat dengan Pendekatan Keluarga (PISPK) [6]. DAMAKESMAS diinisiasi oleh Mantan Walikota Denpasar Bapak Ida Bagus Rai Mantra dan diciptakan oleh Dinas Kesehatan Kota Denpasar, dengan tujuan untuk mengatasi kekurangan dan kelemahan pelayanan pra rumah sakit di Denpasar [6]. DAMAKESMAS saat ini sudah terintegrasi dengan Dinas Kesehatan, BPBD dan Dinas Komunikasi dan Informasi untuk meningkatkan akses, kapasitas dan kualitas kesehatan di Denpasar.

Walaupun saat ini Indonesia memiliki panggilan darurat 112 yang ditujukan untuk mempermudah pelaporan kecelakaan dan Bali melalui Kota Denpasar memiliki DAMAKESMAS yang ditujukan untuk mengoptimalkan pelayanan pra rumah sakit kepada masyarakat, tetap saja masih terdapat lubang yang harus ditutupi. Berdasarkan wawancara yang tim G-One lakukan dengan Pusat Pengendalian Operasi (PUSDALOPS), Badan Penanggulangan Bencana Daerah (BPBD) dan rumah sakit di Denpasar, masih banyak pekerjaan yang harus dilakukan untuk mengoptimalkan pelayanan pra rumah sakit kepada masyarakat dari solusi yang sudah ada. Seperti contoh panggilan darurat 112 masih memiliki kekurangan dimana setiap terdapat pelaporan baik itu kecelakaan, kebakaran, kriminalitas ataupun bencana alam, data dari pelapor tidak tercatat ke dalam sistem dan juga kata-kata seperti

“panggilan bebas biaya” itu sendiri menyebabkan masyarakat melakukan tindakan-tindakan tidak pantas seperti melakukan panggilan palsu ke PUSDALOPS. Ini merupakan masalah yang saling terkait, dimana karena tidak tercatatnya data pelapor kejadian kecelakaan, maka PUSDALOPS dan BPBD merasa kesulitan untuk menindaklanjuti panggilan-panggilan palsu yang masuk. Kemudian program inovasi DAMAKESMAS juga masih memiliki kekurangan-kekurangan, yakni sulitnya proses pencatatan rekam kondisi korban dan tidak terintegrasinya program DAMAKESMAS dengan pihak rumah sakit rujukan di Denpasar, sehingga pelayanan terhadap korban kecelakaan menjadi kurang efektif dan efisien dari sisi rumah sakit rujukan karena mereka telat mendapatkan informasi rekam kondisi korban kecelakaan dan tindak keperawatan yang diberikan oleh ambulan BPBD. Rumah sakit baru akan mendapatkan rekam kondisi korban dan tindak keperawatan setelah ambulan BPBD sampai di rumah sakit, keterlambatan informasi ini yang menyebabkan rumah sakit rujukan dapat dikatakan membuang-buang waktu dengan melakukan hal yang sudah sebelumnya dilakukan oleh perawat di ambulan BPBD, yakni memeriksa kondisi korban kecelakaan. Maka dari itu, G-One hadir sebagai improvisasi untuk menutupi lubang atas permasalahan-permasalahan tersebut.

G-One merupakan sebuah usaha rintisan yang menyediakan solusi penuh atas pelayanan manajemen pra rumah sakit yang ditangani oleh BPBD dengan cara menyederhanakan dan mendigitalisasikan proses pelaporan kecelakaan, navigasi ke lokasi kecelakaan atau rumah sakit rujukan dan pencatatan rekam kondisi korban kecelakaan. G-One memberikan solusi berupa 4 aplikasi dan 1 *web* yang saling terintegrasi dengan harapan dapat memaksimalkan efektivitas dan efisiensi pelayanan pra rumah sakit sehingga probabilitas penyelamatan korban

kecelakaan dapat ditingkatkan. Keempat aplikasi yang G-One kembangkan yaitu: **Public App**, merupakan sebuah aplikasi yang digunakan oleh masyarakat umum untuk melakukan pelaporan terhadap kecelakaan. **Nurse App**, merupakan sebuah aplikasi yang digunakan oleh perawat ambulan untuk melakukan pencatatan rekan kondisi korban kecelakaan. **Hospital App**, merupakan sebuah aplikasi yang digunakan oleh dokter jaga rumah sakit untuk menerima catatan kondisi korban. **Driver App**, merupakan sebuah aplikasi yang digunakan oleh supir ambulan untuk menavigasikan ambulan menuju lokasi kecelakaan atau rumah sakit rujukan. **PUSDALOPS Web App**, merupakan sebuah *website* yang digunakan oleh PUSDALOPS untuk menerima dan memverifikasi laporan kecelakaan yang dikirimkan oleh masyarakat umum.

Berdasarkan pemaparan latar belakang di atas, penelitian ini ditujukan untuk menjelaskan perancangan dan pengembangan sistem pelayanan pra rumah sakit G-One yang nantinya akan dibangun menggunakan Flutter sebagai *framework* untuk membangun Nurse App, Hospital App, Public App dan Driver App serta VueJS sebagai *framework* untuk membangun PUSDALOPS Web App. Untuk integrasi antar aplikasi, penulis akan menggunakan API yang akan dibangun menggunakan bahasa pemrograman Golang dan juga Firebase untuk kepentingan *real-time*.

I. KAJIAN TEORI

A. Pra Rumah Sakit

Prehospital merupakan sebuah layanan yang menyediakan perawatan dan juga transportasi kepada korban baik kecelakaan ataupun kebakaran sebelum sampai di rumah sakit [7]. Pelayanan pra rumah sakit ini merupakan salah satu bentuk dari Emergency Medical Service (EMS) yang mana EMS sendiri merupakan aspek yang sangat penting untuk dilakukan untuk mengurangi angka kematian akibat kecelakaan ataupun kebakaran. Pelayanan ini mencakup banyak aspek, yaitu mulai dari pelaporan kejadian kecelakaan oleh

masyarakat, perawatan dan pencatatan rekam kondisi korban oleh perawat ambulan dari BPBD, transportasi dari TKP menuju rumah sakit. Di Indonesia sendiri sudah dibuat sebuah program bernama Sistem Penanggulangan Gawat Darurat Terpadu atau SPGDT yang telah melibatkan stakeholder-stakeholder penting seperti masyarakat, rumah sakit dan juga pemerintah melalui PUSDALOPS & BPBD.

B. Sistem Informasi

Sistem informasi didefinisikan sebagai komponen-komponen yang saling terkait, yang digunakan untuk mengumpulkan, memproses dan menyimpan data atau informasi. Sedangkan *computer-based information system* (CBIS) adalah sebuah gabungan *hardware*, *software*, *database*, dan jaringan yang memang dibuat untuk mengumpulkan, memanipulasi, menyimpan dan memproses data menjadi informasi yang bermanfaat [8].

Berikut adalah beberapa tipe yang dapat dikategorikan sebagai sistem informasi [9]:

- 1) *Decision Support System (DSS)*
Decision Support System (DSS) merupakan sebuah sistem informasi yang terkomputerisasi digunakan untuk membantu perusahaan dalam membuat sebuah keputusan bisnis [10]. DSS dapat terkomputerisasi secara utuh atau bahkan tidak terkomputerisasi sama sekali, bahkan banyak perusahaan menggunakan *DSS hybrid* yang mana itu berarti penggabungan antara DSS terkomputerisasi dan tidak terkomputerisasi. Secara umum, DSS digunakan oleh perusahaan untuk memproyeksikan keuntungan di masa depan, target penjualan dari tahun ke tahun, dan data-data lain yang terkait dengan operasi atau inventori. Tujuan perusahaan dalam menggunakan DSS, yaitu DSS dapat diprogram untuk memberikan informasi yang mudah dilihat dan digunakan, kemudian informasi-informasi yang dihasilkan oleh DSS

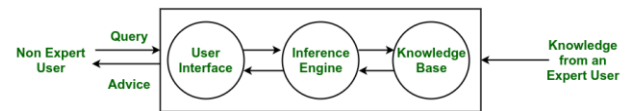
dapat disajikan dalam berbagai bentuk laporan sehingga memudahkan perusahaan dalam meninjau kinerja dari usahanya [11].

- 2) *Enterprise Resource Planning (ERP)* merupakan sebuah sistem manajemen terintegrasi yang digunakan oleh perusahaan pada proses bisnisnya. Penggunaan sistem ERP meliputi pengumpulan data, pemrosesan atau pengolahan data, penyajian data dan penyimpanan data dari aktivitas bisnis yang dilakukan [12]. Perusahaan zaman sekarang tidak dapat lepas dari penggunaan ERP dalam aktivitas bisnis sehari-hari, berikut adalah manfaat-manfaat yang berpotensi didapatkan oleh perusahaan dari penggunaan ERP [13]:

- a) *Insight* bisnis yang berkualitas yang dihasilkan dari laporan-laporan yang dibuat berdasarkan data-data yang didapat dan dikumpulkan secara real-time dari aktivitas bisnis langsung.
- b) Beban operasional yang relatif kecil bila menggunakan ERP dibandingkan menyewa seorang analis.
- c) Kemampuan kolaborasi antar departemen menjadi meningkat dengan adanya sistem yang saling berintegrasi.
- d) Mengurangi resiko-resiko yang tidak diinginkan seperti kehilangan data atau kekeliruan data yang berpotensi menyebabkan kesalahan dalam mengambil keputusan.

- 3) *Expert System (ES)* merupakan sebuah perangkat lunak yang ditenagai oleh *Artificial Intelligence (AI)* yang menggunakan *knowledge* yang disimpan ke dalam *knowledge base* untuk menyelesaikan masalah-masalah kompleks yang hanya dapat

diselesaikan oleh pakar. ES dapat memberikan nasihat-nasihat yang diperlukan dan juga cara-cara untuk menerapkan nasihat-nasihat tersebut. Contoh dari *Expert System* yang ada saat ini adalah MYCIN, DENDRAL, R1/XCON, PXDES, CaDet dan DXplain [14].



Gambar 1 Arsitektur dari Expert System

Gambar 2.1 menjelaskan arsitektur yang biasa digunakan untuk membuat *Expert System*. Berikut adalah penjelasannya:

- a) *Knowledge base*, merupakan tempat dimana fakta atau pengetahuan dan *rules* untuk menyelesaikan masalah disimpan.
- b) *Inference engine*, merupakan sebuah program yang dibuat untuk mendapatkan pengetahuan atau fakta dan *rules* dari *knowledge base* dan mengolahnya untuk menemukan solusi yang dibutuhkan pengguna dari kasus yang diberikan.
- c) *User interface*, merupakan tampilan yang disajikan kepada pengguna untuk berinteraksi dengan *expert system*.

C. Aplikasi Mobile

Aplikasi *mobile* atau yang biasa disebut aplikasi, merupakan sebuah program yang memang didesain khusus untuk berjalan di platform *mobile* seperti *handphone*, tablet, ataupun jam tangan pintar [15]. Pada awalnya aplikasi *mobile* ini dibuat dengan tujuan untuk meningkatkan produktivitas seperti alarm, kalender, jam, email dan kontak. Akan tetapi sekarang ini sudah banyak sekali

pihak-pihak yang membuat aplikasi *mobile* untuk berbagai macam hal, seperti: *gaming*, *design*, GPS, peta, pesan instan dan masih banyak lagi.

Secara umum, aplikasi *mobile* diklasifikasikan menjadi 3 macam, yaitu *native app*, *hybrid app* dan *web app*.

1) *Native App*

Native app merupakan sebuah aplikasi yang dibuat menggunakan framework atau bahasa pemrograman khusus untuk sistem operasi mobile tertentu [16]. Berikut adalah karakteristik dari *native app*:

- a) Performa *native app* cenderung lebih cepat dibandingkan dengan tipe *app* yang lain.
- b) Pengembangan untuk *native app* cenderung memiliki sedikit kerentanan (*vulnerability*) terhadap sistem.
- c) Cenderung lebih *user-friendly* karena mayoritas tampilan *native app* menggunakan tampilan standar dari sistem operasi.
- d) *Native app* dapat secara langsung memanfaatkan fitur-fitur yang terdapat pada perangkat, seperti kamera, lokasi, penyimpanan file dengan mudah.

Disamping kelebihan-kelebihan yang ditawarkan oleh *native app* terdapat kekurangan yang tidak dapat dihindari, yaitu pengembang harus mengembangkan aplikasi yang sama berulang kali untuk sistem operasi yang berbeda. Hal ini menyebabkan perusahaan harus mengeluarkan biaya lebih untuk merekrut talenta-talenta

khusus demi dapat mengembangkan *native app* untuk setiap sistem operasi.

2) *Web-based App*

Web-based app merupakan sebuah aplikasi yang dikembangkan menggunakan teknologi web standar seperti HTML, CSS dan Javascript. Untuk menjalankan *web-based app* diperlukan sebuah browser dan mungkin juga koneksi internet agar dapat berjalan secara maksimal untuk menggunakan seluruh layanan yang ada [15]. Performa yang ditawarkan oleh *web-based app* ini sama dengan performa dari aplikasi web di browser desktop yang nama dapat dikatakan lebih lambat dibandingkan *native app* dan juga tidak semua fitur yang terdapat pada perangkat mobile dapat digunakan di *web-based app*.

3) *Hybrid App*

Hybrid app merupakan gabungan dari *native* dan *web-based app*. Aplikasi tipe ini biasa dikembangkan menggunakan *framework-framework* yang mendukung *cross-platform* seperti: Flutter, React Native, Xamarin dan lain-lain. *Hybrid app* cenderung lebih cepat dalam pengembangannya dan juga mudah karena *hybrid app* hanya menggunakan 1 *codebase* untuk dapat berjalan di seluruh sistem operasi yang berbeda. Akan tetapi, performa yang dihasilkan tidak secepat *native app* dan hanya sampai *level* medekati.

Setelah pengembangan dari aplikasi selesai, tiba saatnya untuk mempublikasikan aplikasi yang dibuat agar dapat digunakan oleh khalayak umum. Untuk melakukan hal tersebut, dibutuhkan sebuah platform *marketplace* yang menyediakan tempat untuk menghosting aplikasi-aplikasi yang sudah dibuat oleh pengembang. Saat ini terdapat 2 *marketplace* aplikasi populer yang menyediakan fasilitas

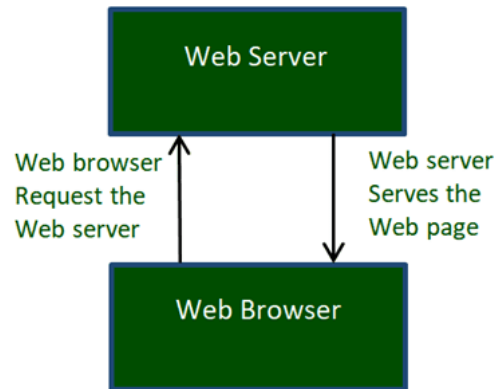
tersebut, yaitu *Google Play Store* dan *Apple App Store*.

D. Flutter

Flutter merupakan *cross-platform UI toolkit* yang didesain untuk mempermudah fase pengembangan dengan menggunakan kembali kode yang sudah dibuat untuk sistem operasi atau lingkungan yang berbeda seperti iOS, Android dan Web dan juga memberikan kemudahan atas penggunaan fitur-fitur dari perangkat yang menjadi tujuan pengembangan [17]. Flutter merupakan salah satu contoh dari *hybrid app* yang mana Flutter menawarkan performa yang mendekati *native app* dan penggunaan fitur-fitur *native* perangkat *mobile*.

E. World Wide Web (WWW)

World wide web (WWW) atau yang biasa disebut sebagai web, merupakan sebuah kumpulan *website-website* di seluruh dunia yang disimpan dalam *web server* yang saling terkoneksi satu sama lain melalui internet [18]. *Website-website* yang tersimpan dapat berupa konten teks, gambar, video ataupun audio. Dikarenakan *website-website* terhubung dengan internet, artinya pengguna internet dapat mengakses *website-website* tersebut melalui perangkat mereka seperti komputer, *smartphone*, tablet dan lain-lain.



Gambar 2 Arsitektur dari World Wide Web

World wide web dasarnya menggunakan arsitektur *client-server* dimana terdapat server yang bertugas sebagai komponen yang memberikan informasi kepada *client* atau user yang ingin mengakses informasi tertentu dari sebuah website.

F. Vue.js

Vue.js merupakan framework Javascript yang ditujukan untuk pengembangan tampilan depan atau *front-end* dari sebuah website [19]. Vue.js diklaim sebagai sebuah framework yang progressive, artinya VueJS didesain sebagai framework yang mampu beradaptasi dengan kebutuhan developer [20]. Tidak seperti framework lain yang terkadang memerlukan kebutuhan tertentu untuk dapat digunakan secara utuh, dengan VueJS, developer hanya perlu mengimport 1 *library* pada tag script dan VueJS sudah dapat terpakai.

```

1 <html>
2 <head>
3   <title>Vue.js Introduction</title>
4   <script type = "text/javascript" src = "js/vue.js"></script>
5 </head>
6 <body>
7   <div id = "intro" style = "text-align:center;">
8     <h1>{{ message }}</h1>
9   </div>
10  <script type = "text/javascript">
11    var vue_det = new Vue({
12      el: "#intro",
13      data: {
14        message: 'My first VueJS Task'
15      }
16    });
17  </script>
18 </body>
19 </html>

```

Gambar 3 Contoh Penggunaan Vue.js

VueJS memiliki beberapa fitur utama sebagai framework Javascript yang mumpuni, yaitu [21] *component*, *reactivity*, *transitions* dan *routing system*.

G. API

API merupakan mekanisme yang memungkinkan dua atau lebih mesin saling berkomunikasi satu sama lain secara peer-to-peer [22]. Sebelum adanya web server, melakukan komunikasi antar mesin sangat sulit untuk dilakukan. Salah satu web service yang banyak digunakan adalah API. API merupakan fungsi yang mengabstraksi fungsi-fungsi lain untuk melakukan tugas tertentu. Fungsi API dapat dipanggil secara langsung di dalam program atau melalui permintaan HTTP. Arsitektur yang digunakan oleh API adalah client-server dimana client menginisiasi panggilan ke API atau yang biasa disebut dengan request, kemudian request diterima oleh API dan API mulai menjalankan fungsi yang sesuai dengan permintaan, setelah fungsi selesai dijalankan, hasil dari fungsi tersebut akan diberikan kepada API untuk selanjutnya diteruskan ke client.

H. Golang

Golang merupakan bahasa pemrograman modern, *general-purpose*, terkompilasi dan *open source* yang diluncurkan pada akhir 2009 oleh Google. Golang berawal dari bahasa pemrograman internal Google, yang mana pada saat itu

masih tahap eksperimen yang terinspirasi dari berbagai bahasa pemrograman lain, yaitu C, Pascal, Alef dan Oberon [23]. Golang menawarkan beberapa keuntungan diantaranya adalah penulisan kode yang simple, performa yang cepat seperti C, pengumpulan sampah otomatis, sintaks yang familiar jadi mudah digunakan dan dipelajari. Saat ini Golang sudah banyak diadopsi oleh perusahaan-perusahaan besar seperti Google, Uber, Dropbox dan Dailymotion. Sedangkan untuk di Indonesia sendiri juga sudah banyak perusahaan yang menggunakan Golang sebagai bagian dari sistem mereka, yaitu Grab, Gojek, Bukalapak, Tokopedia, dan Halodoc.

I. Scrum

Scrum merupakan kerangka kerja dimana sebuah tim bisa mendapatkan nilai dari solusi adaptif yang diberikan untuk permasalahan yang kompleks [24]. *Scrum* dapat juga dikatakan sebagai sebuah panduan yang dapat digunakan untuk menyelesaikan permasalahan yang kompleks dan dinamis dengan memberikan solusi sesuai dengan kebutuhan tersebut.

Scrum menerapkan pendekatan *iterative* dan *incremental* untuk mengoptimalkan dan mengontrol risiko [24]. Keberhasilan proses *scrum* ditentukan oleh 5 *values*, yaitu *commitment*, *focus*, *openness*, *respect* dan *courage*.

Umumnya, tim *scrum* yang dapat dikatakan ideal jika jumlah anggotanya berkisar antara 5-10 atau 10 anggota dan berisi 3 peran, yaitu: *product owner*, *scrum master* dan *developers* [25].

Tabel 2. 1 Komposisi tim dalam scrum

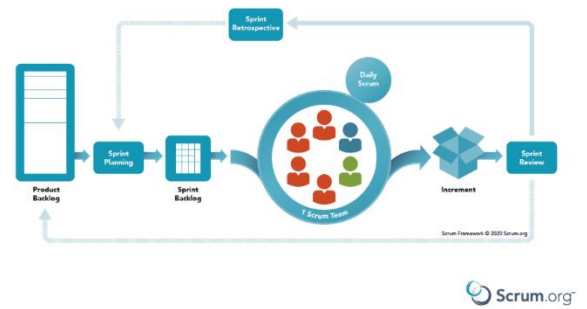
Peran	Jumlah	Tugas
<i>Scrum master</i>	1	Bertugas untuk memimpin jalannya proses <i>scrum</i> dalam sebuah tim.
<i>Product owner</i>	1	Bertugas untuk memberikan visi dan gambaran terkait dengan produk yang akan dikembangkan.
<i>Developers</i>	6	Bertugas untuk mengembangkan produk sesuai dengan aturan dari <i>scrum master</i> dan <i>product owner</i> .
<i>Testers</i>	1	Bertugas untuk menguji produk yang sudah dikembangkan dari sisi teknis.

Komposisi tim dari *scrum* itu sendiri tentu tidak dapat terpaku hanya pada data dari tabel 2.1 di atas, akan tetapi dapat diatur kembali sesuai dengan kebutuhan dan kondisi di lapangan. Berikut ini merupakan contoh komposisi tim *scrum* yang terdapat *UI Designer*, *Technical Writer* dan *Database Specialist* [25], yaitu:

- 1) 1 orang *Scrum Master*.
- 2) 1 orang *Product Owner*.
- 3) 4 orang *Developers* (maksimal).
- 4) 1 atau 2 orang *Testers*.
- 5) 1 orang *Database Specialist*.
- 6) 1 orang *UI Designer*.
- 7) 1 orang *Technical Writer* (tergantung kebutuhan).

Kemudian dalam pelaksanaannya, *scrum framework* dapat dibagi menjadi beberapa *events* dan *artifacts* seperti yang diilustrasikan pada gambar 2.4 berikut.

SCRUM FRAMEWORK



Gambar 4 Diagram Alur Kerja Scrum

Scrum events merupakan kegiatan-kegiatan yang harus dilakukan selama proses *scrum* berlangsung. *Scrum events* dipimpin oleh *Scrum Master* dan dihadiri oleh seluruh anggota *scrum*. *Scrum events* dibuat untuk transparansi sehingga tim *scrum* mengetahui seluruh *progress* masing-masing anggota. Terdapat 5 *scrum events* dalam proses *scrum*, adalah sebagai berikut:

- 1) *Sprint planning*
Sprint planning merupakan awal dari sebuah *sprint* yang nanti akan dilakukan oleh tim. Pada *sprint planning* ini, *product owner* harus memastikan bahwa seluruh anggota sudah siap untuk mendiskusikan *product backlog* [24]. Setelah *product backlog* selesai didiskusikan, tim *developers* akan memilah tugas-tugas dalam *product backlog* ke dalam *sprint backlog* untuk dikerjakan pada *sprint* berikutnya.
- 2) *Sprint*
Sprint merupakan jantung dari *scrum*, dimana ide diubah menjadi *values*. Seluruh kerjaan yang dilakukan untuk mencapai *Product Goal* [24], seperti *sprint planning*, *daily scrum*, *sprint review* dan *sprint retrospective* dilakukan pada tahap ini.
Selama tahap *sprint* berlangsung:
 - a) Tidak boleh ada perubahan yang berpotensi merusak *sprint goal*.
 - b) Kualitas tidak boleh berkurang.

- c) *Product backlog* disusun secukupnya.
 - d) Cakupan pekerjaan boleh diklarifikasi dan dinegosiasikan dengan *product owner*.
- 3) *Daily scrum*
Daily scrum merupakan tahap dimana setiap anggota melaporkan *progress* yang telah mereka lakukan terhadap *sprint goal*. Normalnya, *daily scrum* dilakukan selama 15 menit sebelum pekerjaan *sprint* rutin dilakukan. Dengan adanya *daily scrum* ini, dapat meningkatkan komunikasi antar anggota, pengambilan keputusan dengan cepat, dan mengeliminasi pertemuan-pertemuan yang tidak diperlukan.
- 4) *Sprint review*
 Tujuan dari *sprint review* adalah untuk melihat *output* dari *sprint* yang sudah dilakukan dan juga menentukan apakah kerjaan pada *sprint* ini sudah dapat diterima atau belum. Pada tahap ini, *scrum team* mempresentasikan hasil kerjaan pada *sprint* ini kepada *product owner*. *Sprint review* merupakan *event* kedua dari terakhir dari *sprint* dan dibatasi hanya maksimal 4 jam untuk satu bulan *sprint*. Untuk *sprint* yang pendek, normalnya durasi dari *sprint review* juga pendek.
- 5) *Sprint retrospective*
 Tujuan dari *sprint retrospective* adalah untuk menilai kualitas dan keefektifan dari *sprint* yang telah dilakukan [24]. Pada tahap ini, *scrum team* mengevaluasi hasil dari *sprint* yang telah dilakukan untuk meningkatkan efektivitas pada *sprint* selanjutnya. *Sprint retrospective* merupakan kesimpulan dari *sprint*, dan dibatasi kurang lebih hingga 3 jam.
- Scrum artifacts* merupakan hasil representasi dari nilai dalam *scrum* [24]. Setiap artefak yang dibuat mengandung tujuan dari produk atau *sprint* yang dibuat atau lakukan yang dimaksudkan untuk

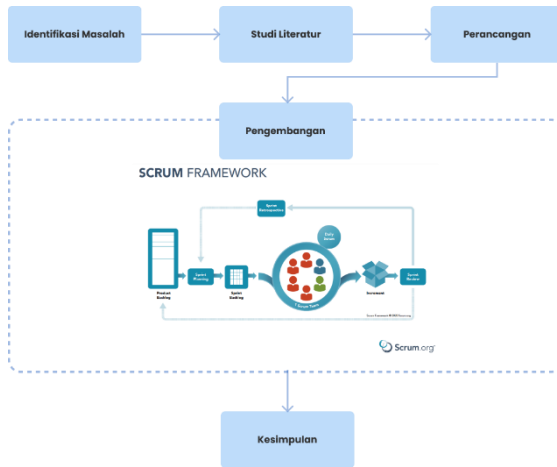
meningkatkan transparansi kepada para anggota tim. Sebagai contoh *product backlog* mengandung *product goal*, *sprint backlog* mengandung *sprint goal* dan *increment* mengandung *definition of done*.

- 1) *Product backlog*
Product backlog merupakan daftar pekerjaan-pekerjaan utama yang dikelola oleh *Product Owner* [26]. Daftar ini mengandung kebutuhan produk, perbaikan produk, dan peningkatan kualitas produk. Daftar ini bisa juga disebut sebagai “*To-Do*” dalam proses *scrum*.
- 2) *Sprint backlog*
Sprint backlog merupakan daftar pekerjaan-pekerjaan yang sudah dipilih oleh tim pengembang dari *product backlog* untuk dikerjakan atau diimplementasikan di *sprint* saat ini [26]. Daftar pekerjaan dalam *sprint backlog* ini bersifat flexible, artinya adalah dapat berubah sesuai dengan kebutuhan pada saat *sprint* berlangsung.
- 3) *Increment*
Increment merupakan daftar pekerjaan dari *sprint backlog* yang sudah selesai dimana ini merepresentasikan produk akhir dari *sprint* yang sudah dilakukan sebelumnya [26].

METODE PENELITIAN

Penelitian ini termasuk ke dalam penelitian terapan, dimana hasil dari penelitian ini merupakan sebuah solusi untuk meningkatkan pelayanan pra rumah sakit di PUSDALOPS, BPBD dan rumah sakit daerah Denpasar.

Alur Penelitian



Gambar 3 Alur Penelitian

Alur yang digunakan dalam penelitian ini ditunjukkan pada gambar 3.1 sebagai berikut:

Berdasarkan gambar diatas dapat dijelaskan tahapan-tahapan yang akan dilakukan sebagai berikut:

- 1) Identifikasi Masalah
Langkah pertama yang akan dilakukan adalah identifikasi masalah terhadap penelitian yang akan dilakukan. Masalah yang ditemukan pada penelitian ini bersumber pada data-data statistic yang menunjukkan tingginya angka korban kecelakaan yang mengalami kematian akibat kurang terpadunya sistem pra rumah sakit di Indonesia khususnya Denpasar.
- 2) Studi Literatur
Setelah penulis mengidentifikasi permasalahan, selanjutnya adalah melakukan studi literatur untuk memperkuat argumen penulis atas permasalahan yang diangkat. Sumber studi literatur berasal dari buku, jurnal maupun artikel.
- 3) Perancangan
Pada fase perancangan ini, penulis melakukan riset terkait dengan teknologi yang cocok untuk diimplementasikan, merancang use case diagram sistem yang nantinya akan digunakan untuk membuat

backlog. Kemudian, merancang struktur data yang akan disimpan dan diproses pada saat implementasi.

- 4) Pengembangan
Setelah proses perancangan selesai, penulis melakukan pengembangan sesuai dengan perancangan yang sudah dibuat pada tahap sebelumnya. Pada proses pengembangan ini, penulis akan mengikuti kaidah-kaidah dari *scrum framework*. Untuk lebih detailnya adalah sebagai berikut:

- a) Sprint planning
Pada sprint planning, penulis

bersama dengan anggota tim G-One, melakukan diskusi terkait dengan penentuan prioritas pekerjaan di Product Backlog yang akan dilakukan dan memasukkan pekerjaan tersebut ke dalam Sprint Backlog.

- b) Sprint
Setelah *sprint planning* selesai dilakukan, selanjutnya adalah melakukan *sprint* dengan rentang waktu antara 1-4 minggu per *sprint*.

- c) Daily scrum
Ketika sedang melakukan sprint, disaat yang bersamaan, penulis dan anggota tim G-One melakukan Daily Scrum yang diselenggarakan di pagi hari setiap sebelum mulai bekerja. Pada daily scrum, peneliti dan anggota tim G-One, menyampaikan progress terkait dengan pekerjaan yang dilakukannya. Progress yang disampaikan adalah pekerjaan yang kemarin dilakukan, pekerjaan apa yang akan dilakukan hari ini dan masalah-masalah yang ditemukan selama melakukan pekerjaan.

- d) Sprint review
Setelah selesai melakukan sprint, selanjutnya adalah

melakukan sprint review. Dimana penulis dan anggota tim G-One, mempresentasikan hasil pekerjaan selama sprint dan meminta feedback kepada Product Owner. Jika ada yang harus diperbaiki, maka catatan perbaikan dari Product Owner akan dimuat kedalam product backlog untuk nanti dikerjakan pada sprint berikutnya.

- e) Sprint retrospective
- Ketika pekerjaan-pekerjaan pada sprint backlog sudah selesai, selanjutnya adalah melakukan sprint retrospective. Pada sprint retrospective, penulis dan anggota tim G-One, melakukan evaluasi terkait dengan sprint yang telah dilakukan. Terdapat beberapa pertanyaan yang nantinya akan terjawab pada fase ini, yaitu adalah tentang bagaimana *sprint* minggu ini berlangsung, apa saja yang perlu diperbaiki dan diterapkan pada *sprint* selanjutnya dan apa saja yang sudah terselesaikan pada *sprint* kali ini.

A. Sumber Data

Penelitian ini memiliki 2 jenis sumber data, yaitu adalah sebagai berikut:

- 1) Sumber data primer yang digunakan dalam penelitian ini merupakan data hasil dari penelitian yang sebelumnya sudah dilakukan oleh tim G-One, yaitu adalah prototipe *low fidelity* dan *high fidelity*.
- 2) Sumber data sekunder pada penelitian ini berasal dari media-media ilmu pengetahuan seperti buku, jurnal, video edukasi dan lain-lain yang memiliki keterkaitan dengan topik dan juga teknologi yang digunakan dalam penelitian ini.

HASIL DAN PEMBAHASAN

Pengujian sistem merupakan sebuah tahapan untuk memastikan bahwa seluruh fungsionalitas sistem yang telah dibuat sudah berjalan dengan baik. Pengujian sistem dalam penelitian ini dilakukan pada *sprint review* di setiap *sprint*. Pengujian ini menggunakan *black-box testing* dimana penguji atau *product owner* tidak mengetahui implementasi internal dari sistem yang dibuat. Berikut merupakan hasil pengujian pada setiap *use case* yang dikelompokkan berdasarkan sprint dari sistem G-One.

SIMPULAN

Berdasarkan hasil pembahasan yang telah diuraikan, maka dapat ditarik kesimpulan bahwa metode pengembangan perangkat lunak dengan *scrum framework* dapat digunakan untuk mengembangkan sistem informasi pelayanan pra rumah sakit. Penggunaan framework *scrum* dalam penelitian ini menghasilkan 20 product backlog yang dibagi ke dalam 4 sprint dan diselesaikan dalam kurun waktu kurang lebih sekitar 4 bulan, kemudian output dari proses *scrum* ini akan menghasilkan sebuah sistem informasi pelayanan pra rumah sakit yang terdiri dari tiga aplikasi *mobile* dan satu web, yaitu Public App, PUSDALOPS Web App, Nurse App dan Hospital App yang dikembangkan menggunakan teknologi-teknologi diantaranya adalah *Flutter* untuk aplikasi *mobile*, *Vue.js* untuk *web*, *Golang* untuk API, *Firebase* untuk *realtime communication* dan *realtime database* dan *MySQL* untuk *database* utama.

Berdasarkan kesimpulan yang sudah diuraikan, maka saran yang dapat penulis berikan yaitu adalah rancangan dari *database* utama yang penulis merasa masih harus diperbaiki di bagian peran pengguna seperti *nurses*, *doctors*, *publics* dan *operators*. Pada penelitian ini, rancangan data untuk setiap pengguna dibuat dalam tabel terpisah sehingga mendukung adanya duplikasi data ketika

mendaftar. Kedepannya, dapat dipertimbangkan untuk menggunakan konsep *polymorphic association* untuk memusatkan data pengguna ke dalam satu tabel yang berelasi dengan tabel peran di tabel terpisah. Dengan diterapkannya hal tersebut, harapannya pengguna yang sudah mendaftar di aplikasi Nurse App tidak dapat lagi mendaftar di aplikasi lainnya, sehingga tidak ada duplikasi data.

DAFTAR PUSTAKA

- [1] World Health Organization (WHO), "Global Status Report on Road Safety," World Health Organization (WHO), 2018.
- [2] "WORLD ROAD TRAFFIC ACCIDENT REPORT!," [Online]. Available: <https://www.worldlifeexpectancy.com/world-road-traffic-accidents-report>. [Accessed 19 Oktober 2021].
- [3] "Kecelakaan Lalu Lintas Jadi Kontributor Ketiga Kematian Terbesar di Indonesia," [Online]. Available: <https://otomotif.kompas.com/read/2021/09/23/074200615/kecelakaan-lalu-lintas-jadi-kontributor-ketiga-kematian-di-indonesia>. [Accessed 19 Oktober 2021].
- [4] R. Y. Wardana, "Ratusan Kasus Kecelakaan Terjadi di Denpasar Dalam 5 Bulan Terakhir, Didominasi Kendaraan Ini, Begini Kata Polisi," 9 Juni 2020. [Online]. Available: <https://www.gridoto.com/read/222187573/ratusan-kasus-kecelakaan-terjadi-di-denpasar-dalam-5-bulan-terakhir-didominasi-kendaraan-ini-begini-kata-polisi>. [Accessed 29 Oktober 2021].
- [5] "Layanan Darurat 112," [Online]. Available: <https://www.indonesia.go.id/layanan/kependudukan/sosial/layanan-darurat-112>. [Accessed 21 Oktober 2021].
- [6] "DAMAKESMAS, menjawab kebutuhan kesehatan masyarakat Denpasar," [Online]. Available: <https://www.menpan.go.id/site/berita-terkini/damakesmas-menjawab-kebutuhan-kesehatan-masyarakat-denpasar>. [Accessed 21 Oktober 2021].
- [7] "Prehospital Emergency Medicine," Wikipedia, [Online]. Available: https://en.wikipedia.org/wiki/Pre-hospital_emergency_medicine. [Accessed 27 Oktober 2021].
- [8] G. W. S. R. M. Reynolds, Principles of Information Systems, 2018.
- [9] "Information System," Wikipedia, [Online]. Available: https://en.wikipedia.org/wiki/Information_system. [Accessed 25 10 2021].
- [10] "Decision Support System," Wikipedia, [Online]. Available: https://en.wikipedia.org/wiki/Decision_support_system. [Accessed 25 10 2021].
- [11] "Decision Support System - DSS," Investopedia, [Online]. Available: <https://www.investopedia.com/terms/d/decision-support-system.asp>. [Accessed 25 10 2021].
- [12] "Enterprise Resource Planning," Wikipedia, [Online]. Available: https://en.wikipedia.org/wiki/Enterprise_resource_planning. [Accessed 25 10 2021].
- [13] "What is ERP?," Oracle, [Online]. Available: <https://www.oracle.com/erp/what-is-erp/>. [Accessed 25 10 2021].
- [14] "Expert System," geeksforgeeks, [Online]. Available: <https://www.geeksforgeeks.org/expert-systems/>. [Accessed 25 10 2021].
- [15] "Mobile App," Wikipedia, [Online]. Available: https://en.wikipedia.org/wiki/Mobile_app. [Accessed 25 10 2021].
- [16] "What is A Native App?," [Online]. Available: <https://decemberlabs.com/blog/what-is-a-native-app/>. [Accessed 26 10 2021].
- [17] "Flutter Architectural Overview," Flutter, [Online]. Available: <https://flutter.dev/docs/resources/architectural-overview>. [Accessed 26 10 2021].
- [18] "What is World Wide Web?," [Online]. Available: <https://www.javatpoint.com/what-is-world-wide-web>. [Accessed 26 10 2021].
- [19] "Introduction - Vue.js," Vue.js, [Online]. Available: <https://vuejs.org/v2/guide/>. [Accessed 26 10 2021].
- [20] "Introduction to Vue.js," [Online]. Available: https://vuecourse.com/_course/1/1-intro/. [Accessed 26 10 2021].
- [21] "Vue.js," Wikipedia, [Online]. Available: <https://en.wikipedia.org/wiki/Vue.js>. [Accessed 26 10 2021].
- [22] N. Yellavula, Building RESTful Web Service with Go, Packt Publishing, 2017.
- [23] M. Tsoukalos, Mastering Go: Create Golang production applications using network libraries,

- concurrency and advanced Go data structures, Packt Publishing, 2019.
- [24] K. S. & J. Sutherland, *The Scrum Guide: The Definitive Guide to Scrum: The Rules of the Game*, scrum.org, 2020.
- [25] S. M. d. C. A. D. Ashish Mundra, *Practical Scrum-Scrum Team: Way to Produce Successful and Quality Software*, p. 5, 2013.
- [26] "Scrum: Learn how to scrum with the best of 'em," Atlassian Agile Coach, [Online]. Available: <https://www.atlassian.com/agile/scrum>. [Accessed 25 12 2021].
- [27] M. N. Novri Hadinata, "Implementasi Metode Scrum dalam Rancang Bangun Sistem Informasi Penjualan (Studi Kasus: Penjualan Sperpart Kendaraan)," 2017.
- [28] A. D. P. M. H. D. S. Gigih Forda Nama, "Rancang Bangun Aplikasi Game Edukasi Koleksi Permainan Aksara Lampung (Koper Apung) Berbasis Android Menggunakan Metode Scrum," *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK)*, vol. 6, pp. 421-430, 2019.
- [29] T. N. S. K. T. Shimazui, "Speech recognition shortens the recording time of prehospital medical documentation," *American Journal of Emergency Medicine*, 2021.
- [30] H. K. S. K. K. I. Takuro Nakae, "Smartphone-Assisted Prehospital Medical Information System for Analyzing Data on Prehospital Stroke Care," 2014.
- [31] S. A. A. G. Eko Setiadana, "Pengembangan Sistem Penagihan Biaya Kuliah Dengan Fitur WhatsApp Menggunakan Metode Scrum," *Jurnal Teknik Informatika dan Sistem Informasi*, vol. 8, pp. 1252-1264, 2021.
- [32] S. Mitropoulos, "An online emergency medical management information system using mobile computing," 2021.