

BAB 2

TINJAUAN PUSTAKA

Pada bab ini akan menjelaskan terkait tinjauan pustaka yang digunakan pada penelitian.

2.1 PT RESI SALOKO GADING

PT. RESI SALOKO GADING adalah lembaga riset opini publik yang berdiri sejak 2020. RESI mempunyai kompetensi personal, profesionalisme dan budaya kerja yang positif untuk terwujudnya peran lembaga sebagai pelayan dan pengembangan nilai di masyarakat sehingga tercipta hasil yang optimal. Memiliki solidaritas tim yang kuat dan hubungan kerja sama yang baik dengan masyarakat, instansi dan beberapa pihak. Bagi RESI bukanlah merupakan persoalan untung atau rugi, tapi merupakan bagian yang tak terpisahkan dan upaya bersama dalam memberikan pelayanan dan rekomendasi yang baik. Bersama masyarakat dan atau instansi akan mampu memberikan peranan secara maksimal, konkrit dalam proses pelayanan rekomendasi yang dapat mewujudkan harapan serta cita-cita dalam menghadapi persoalan yang dihadapi masyarakat atau instansi. RESI memberikan pelayanan kepada pemangku kebijakan, instansi, dan masyarakat yang mencakup persoalan politik, hukum, sosial, ekonomi, kesehatan, pendidikan, teknologi dan telekomunikasi (PT. RESI SALOKO GADING, 2020). Adapun visi misi PT. RESI SALOKO GADING adalah sebagai berikut :

A. Visi

RESI berupaya menjadi pusat kajian pelayanan masyarakat mengenai politik, hukum, sosial, ekonomi, kesehatan, pendidikan, teknologi, dan telekomunikasi. Lembaga ini berkomitmen memberikan sumbangan pemikiran dan rekomendasi kebijakan publik serta institusi di Indonesia.

B. Misi

1. Menghasilkan kajian, analisa dan rekomendasi politik, hukum, sosial, ekonomi, kesehatan, pendidikan, teknologi dan

telekomunikasi yang dapat dipertanggungjawabkan untuk membuat kebijakan di masyarakat dan atau instansi

2. Turut serta dalam monitoring opini publik mengenai politik, hukum, sosial, ekonomi, kesehatan, pendidikan, teknologi dan telekomunikasi secara berkala sebagai instrumen masukan bagi pembuatan kebijakan, sekaligus untuk mendekatkan keputusan pemangku kebijakan dengan aspirasi publik yang akurat legitimate konkrit, stabil dan efektif.

2.2 *Quick Count*

Quick Count dapat memberikan estimasi hasil perhitungan suara sementara, hitung cepat memiliki peran penting pada pemilihan umum. Hal ini memberikan hasil perhitungan suara Pemilu sementara yang sedang berlangsung secara real time (Karami, 2018). Metode *Quick Count* merupakan perhitungan suara yang didapatkan dilaksanakan dengan cepat dengan tujuan memantau dan mempercepat perhitungan suara yang didapatkan pada pemilihan umum. Cara kerja pada sistem *Quick Count* yaitu melakukan pemantauan serta mencatat informasi dan juga melakukan perhitungan cepat pada jumlah suara yang didapatkan selanjutnya dapat melaporkan hasil pemilihan umum ke pusat pengumpulan data (Indra, 2013).

2.3 *Laravel*

Framework PHP yang dirilis dibawah lisensi MIT dan dibangun dengan menggunakan konsep *model, view, controller* (MVC) dinamakan dengan *Laravel*. *Laravel* dapat meningkatkan kualitas yang ada pada perangkat lunak dengan mengurangi biaya yang dikeluarkan pada saat pengembangan awal dan pemeliharaan, serta meningkatkan pengalaman bekerja dengan sintaks yang ekspresif jelas dan menghemat waktu. Adapun pendekatan perangkat lunak yang memisahkan aplikasi logika dari presentasi yang dapat disebut dengan MVP. MVC terdiri dari *Model, View* dan *Controller* dimana memiliki arti yakni *model* mewakili struktur data, *view* memiliki pengertian yakni pengaturan tampilan, dan *controller* memiliki arti pengaturan yang dimana guna untuk menjembatani *model* dan *view* (Supardi & Sulaeman, 2019).

Arsitektur yang terdapat pada MVC diadaptasi untuk digunakan pada *Laravel* dalam kerangka kerjanya. MVC sangat berdampak pada pengembangan web

dimana untuk mengurangi kompleksitas dari perancangan serta memberikan fleksibilitas dari penggunaan kembali kode (*re-use code*). Secara garis besar MVC terdiri dari *Model, View, dan Controller*, namun pada *Laravel* ditambah satu elemen lain yaitu *Route* yang membantu menghubungkan permintaan user ke *controller* (Wei Cui, Lin Huang, LiJing Liang, 2009).

Terdapat sebuah kelas yang berfungsi untuk mengelola isi dari basis data, dimana kelas tersebut masuk ke dalam *Model* yang berasal dari tabel pada *database*. Jembatan antara *controller* dengan basis data saat *controller* ingin memanfaatkan atau mengelola data di *database* merupakan fungsi dari *Model* (Wei Cui, Lin Huang, LiJing Liang, 2009).

Controller atau dikenal dengan Pengaturan berisi kelas-kelas yang berfungsi sebagai logika pemrograman dan untuk memproses *user events*. Dapat digunakan menjadi pusat dan algoritma dari sistem dan jembatan antara *model* dan *view* merupakan fungsi *controller* pada MVC. Dapat menerima permintaan dari user lalu memproses permintaan tersebut merupakan fungsi lain dari *controller* (Wei Cui, Lin Huang, LiJing Liang, 2009).

Sedangkan untuk menampilkan tampilan antarmuka sekaligus menampilkan data yang telah dikirimkan oleh *controller* merupakan fungsi dari *view*. *View* umumnya dituliskan menggunakan Bahasa HTML pada MVC dan pengembangan web (Wei Cui, Lin Huang, LiJing Liang, 2009).

Untuk membuat pengembangan web menjadi lebih mudah dapat menggunakan salah satu fitur unggulan yang terdapat pada *Laravel*. Adapun fungsi yang dimiliki oleh fitur unggulan dari *Laravel* ini adalah untuk mengenalkan cara yang berbeda dalam mengakses database relasional. *Eloquent* adalah sebuah syntax yang akan dikonversi sesuai dengan jenis basis data yang digunakan dalam aplikasi tanpa harus membuat perintah untuk setiap basis data, dan *Eloquent* digunakan untuk menghubungkan *Laravel* dengan berbagai jenis data dalam satu aplikasi. Kemudahan yang didapatkan pada *Laravel* yakni dalam hal membuat *Application Programming Interface (API)* sehingga *Laravel* dapat berperan menjadi *Back-end* aplikasi yang dapat dihubungkan dengan berbagai macam aplikasi *front-end* (Gunawan et al., 2020). Pada penelitian ini akan menggunakan fitur API yang

didapatkan dari Laravel untuk menghubungkan antara basis data, *back-end* dan aplikasi *mobile*.

www.itk.ac.id

2.4 *React Native*

Kerangka kerja javascript yang digunakan untuk membangun aplikasi *mobile* Android maupun iOS atau biasa dikenal dengan React Native. React dan library javascript merupakan dasar dari React Native dalam membangun antarmuka. Javascript dan JSX merupakan campuran untuk menulis React Native, lalu antarmuka javascript untuk platform API dipaparkan melalui React Native dimana dapat mengakses fitur-fitur seperti kamera, lokasi, dan lain-lain yang terdapat pada ponsel pada saat pengembang dalam membangun aplikasi (Eisenman, 2016).

Facebook mengembangkan Library JavaScript untuk membangun antarmuka pengguna yang menargetkan *platform mobile*. JSX adalah campuran markup JavaScript dan XML-esque yang digunakan untuk menulis aplikasi React Native dimana mirip dengan React.js untuk web. Untuk dapat menjembatani javascript, React Native pada tingkat bawah memanggil API rendering native di Objective-C (untuk iOS) atau Java (untuk Android). Komponen UI mobile yang native akan ter-render pada aplikasi dan akan terlihat seperti aplikasi mobile yang lain. Antarmuka JavaScript untuk API platform akan ditampilkan pada bereaksi bawaan, sehingga kamera ponsel, atau lokasi pengguna dapat mengakses fitur platform pada aplikasi React Native (Saputra, 2019)

2.5 *REST API*

Representational State Transfer API (REST API) merupakan suatu standar dari arsitektur untuk pengembangan sistem berbasis *web* sebagai protokol untuk komunikasi data. Arsitektur *REST* merupakan hubungan antara *client* dan *server*, yang dimana sumber data tersebut tersedia dari REST *server* dan sumber data tersebut kemudian dapat diakses dan ditampilkan pada REST *client*. Sumber data tersebut diidentifikasi berdasarkan URLs (*Universal Resource Identifiers*). Selanjutnya pada sumber data akan ditampilkan dalam format teks, XML, atau JSON. REST mendeskripsikan bagaimana suatu aplikasi dapat berkomunikasi melalui *Hypertext Transfer Protocol* (HTTP) (I Gede Merta Ariantara, 2020).

www.itk.ac.id

2.6 Basis Data Mysql

NoSQL dan SQL adalah 2 jenis dari basis data, pada dua jenis basis data tersebut terdapat perbedaan. Jenis basis data yang tidak mempunyai struktur relasi dan table adalah NoSQL. Penyimpanan data pada NoSQL terdapat pada dokumen, dimana dokumen tersebut memiliki syntax yang mirip dengan JSON. Lebih fleksibel dalam penyimpanan data dan juga unggul kecepatannya saat mencari atau membaca data yang ada di dalam basis data adalah kelebihan dari NoSQL. Namun, tidak mempunyai struktur relasi dan tabel, membuat basis data jenis ini lebih sulit untuk dibaca dan dilihat relasi antar tabelnya adalah kekurangan dari NoSQL.

Kebalikkan dari NoSQL yakni SQL atau *Structured Query Language* memiliki struktur relasi antar tabel yang mempermudah dalam mendefinisikan hubungan antara model dan juga penyimpanan data menjadi lebih rapi (Wiguna et al., 2019). Salah satu contoh perangkat lunak sistem manajemen basis data yang berjenis SQL adalah MySQL. Dengan memiliki beberapa keunggulan dibandingkan dengan SQL lainnya, di dalam MySQL terdapat sistem manajemen basis data yang dapat diakses secara gratis, dapat digunakan dengan bahasa SQL, dan client dapat menggunakan banyak basis data dalam waktu yang bersamaan (Gunawan et al., 2020)

2.7 *Personal Extreme Programming*

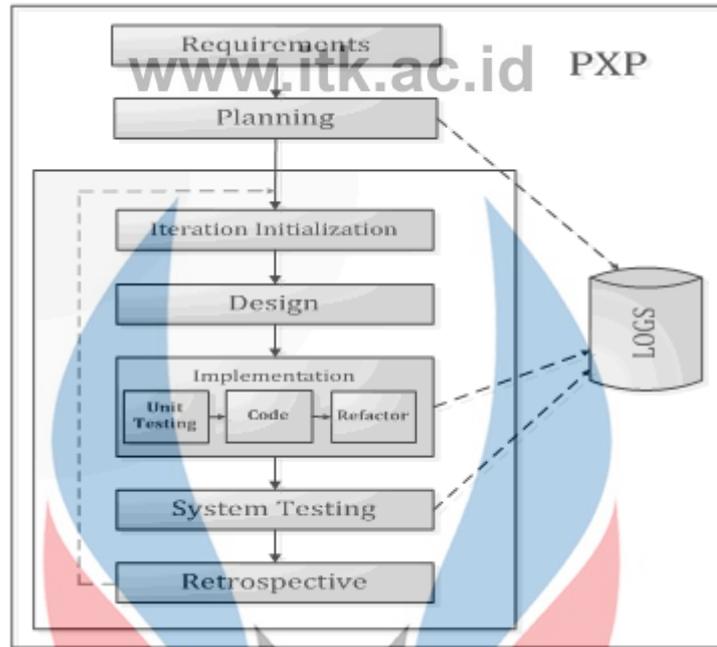
Personal Extreme Programming (PXP) merupakan metodologi pengembangan perangkat lunak yang dirancang untuk diterapkan oleh pengembang perangkat lunak secara individual. Subset dari praktik yang dihasilkan dari pengembangan *Extreme Programming* (XP) yang disesuaikan untuk dilakukan oleh pengembang otonom diperkenalkan dari PXP. Mengurangi upaya dokumentasi dan pemeliharaan namun agar tetap terjaga prinsip-prinsip dasar yang dihasilkan dari pengembangan pada PXP. Berulang dan penerapan praktiknya merupakan sifat dari PXP pada proses pengembangan PXP yang memungkinkan pengembang menjadi lebih fleksibel dan responsif terhadap perubahan. Bagian utama dari pekerjaan pengembang ini menjadi tempat bergantung untuk metodologi yang disarankan khususnya pada otomatisasi dimana untuk meningkatkan kinerja pengembang dan

memperpendek interval pengiriman dan waktu yang dihabiskan pada dukungan sistem perangkat lunak (Dzhurov et al., 2014):

- PXP membutuhkan pendekatan yang disiplin, pengembang bertanggung jawab untuk mengikuti proses dan menerapkan praktik PXP
- Pengembang harus mengukur, melacak, dan menganalisis pekerjaan harian mereka
- Pengembang harus belajar dari variasi kinerja mereka dan bertujuan untuk meningkatkan proses berdasarkan data proyek yang dikumpulkan
- PXP melibatkan pengujian berkelanjutan
- Perbaikan kerusakan harus dilakukan pada tahap pengembangan awal, ketika biayanya lebih rendah
- Pengembang harus mencoba mengotomatiskan pekerjaan sehari-hari mereka sebanyak mungkin

Iteratif adalah sifat dari proses PXP, dan terdiri dari beberapa iterasi dan siklus sebaris. Persyaratan yang ditentukan sebelumnya dan tetap stabil selama pelaksanaan biasanya terdapat persyaratan yang ditentukan sebelumnya dan juga terdapat perencanaan tugas. Persyaratan, perencanaan tugas dapat direvisi biasanya terdapat dalam kasus perubahan. Inisialisasi iterasi adalah awal mula dari pengembangan iterasi dan diakhiri dengan iterasi retrospektif (Dzhurov et al., 2014).

Perencanaan tugas dan durasi aktual, saran perbaikan dan jumlah kerusakan, dan detail merupakan serangkaian proses dalam pengembangan yang berisikan file log dengan informasi (Dzhurov et al., 2014).



Gambar 2.1 Metodologi PXP (Dzhurov et al., 2014)

Gambar 2.2 memberikan gambaran umum tentang proses PXP. Selama fase persyaratan, dokumen dengan persyaratan fungsional dan nonfungsional untuk sistem dibuat. Fase ini opsional, dokumen persyaratan dapat dibuat dalam pertemuan antara klien dan karyawan lain dari organisasi tempat pengembang bekerja, atau dapat diperoleh dari insinyur perangkat lunak lain, dalam hal perpanjangan produk yang sudah ada (Dzhurov et al., 2014).

Dalam fase perencanaan, pengembang merancang tugas berdasarkan dokumen analisa kebutuhan. Tugas yang dirancang dapat berisi tugas kecil yang dikategorikan. Tugas kecil yang dirancang berdasarkan perencanaan dokumen analisa kebutuhan atau dari dokumen proyek sebelumnya dengan jenis yang sama jika data tidak cukup dan pengembang harus membuat asumsi tentang perkiraan waktu proyek. Perkiraan waktu proyek mencakup tugas kecil dan tugas utama. Pada tahap ini juga ditentukan terkait *platform* apa yang akan digunakan, bahasa pemrograman apa yang tepat, dan keputusan teknis lainnya (Dzhurov et al., 2014).

Proses iterasi memiliki tahap awal yakni biasa dikenal dengan *Iteration Initialization*. Memilih tugas mana yang akan dikerjakan dan menjadi tujuan utama dari iterasi merupakan proses yang dilakukan terlebih dahulu dalam Iterasi. Berdasarkan ruang lingkup proyek terdapat waktu terjadinya proses iterasi yang

bervariasi dari 1 hingga 3 minggu lamanya. Hasil akhir dari proyek merupakan hasil dari iterasi yang memungkinkan untuk menjadikan versi produk yang perlu dikembangkan (Dzhurov et al., 2014).

Pada proses pengembang membuat terlebih dahulu modul sistem dan kelas yang nantinya akan dikembangkan selama fase desain berlangsung dan pada proses iterasi. Pemenuhan kebutuhan yang telah ada merupakan keutamaan dalam pengembang dan tanpa menambahkan asumsi terhadap kebutuhan yang akan datang. Penggunaan alat yang sesederhana mungkin dianjurkan dalam menjalankan metode desain dipilih oleh pengembang (Dzhurov et al., 2014).

Fase implementasi adalah tempat pembuatan kode yang sebenarnya terjadi. Pengembang mengimplementasikan modul dan kelas yang telah ditentukan, setelah itu melanjutkan pengujian berdasarkan apa yang telah diimplementasi. Fase ini terdiri dari tiga sub-fase- pengujian unit, pembuatan kode dan pemfaktoran ulang kode, dijalankan dalam urutan berikut (Dzhurov et al., 2014):

- Untuk keluar dari fase implementasi, kode harus dikompilasi tanpa kesalahan dan semua pengujian unit harus berhasil lulus.
- Semua fitur yang dikembangkan sejauh ini diuji selama pengujian sistem
- Pengembang memastikan solusi yang digunakan telah memenuhi persyaratan yang telah ditentukan. Pengembang juga memastikan kesalahan yang terjadi telah diselesaikan dengan baik.

Akhir dari proses iterasi adalah Retrospektif. Data-data yang telah diperoleh pada saat proses iterasi dapat diolah menjadi sebuah analisis. Terdapatnya tenggat waktu yang telah ditentukan sebelumnya sehingga pengembang harus memastikan bahwa proyek tersebut tidak melebihi waktu. Pengembang menganalisis semua Peningkatan sebuah proses dan menyesuaikan salah satu praktik dianalisis pada proses pengembang dengan proses yang menyertai. Pada tahap awal harus terdapat proses dan praktik dalam perbaikan masalah untuk mencegah kegagalan proyek. Kandidat rilis atau rilis produk merupakan keluaran dari fase ini. Apabila terdapat kesalahan dan produk maka fase “retrospektif” dapat memulai kembali fase “inisialisasi iterasi” namun apabila sudah tidak terdapat masalah maka sudah memenuhi kebutuhan maka pengembang dapat menandai sebagai akhir pengembangan (Dzhurov et al., 2014).

2.8 *User Acceptance Test (UAT)*

Klien atau pengguna akhir biasanya melakukan yang namanya *User acceptance test*. *User acceptance test* dilakukan dengan tujuan untuk mengidentifikasi bagaimana sebuah cara kerja dari suatu sistem yang berjalan dan sistem tersebut dapat bermanfaat bagi klien atau pengguna akhir sebelum produk/aplikasi akan diimplementasikan atau dijalankan pada lingkungan klien atau pengguna akhir (Mohd & Shahbodin, 2015). Salah satu hal terpenting pada saat pengujian perangkat lunak adalah *User acceptance test (UAT)* dimana setelah dilakukan *User acceptance test* maka sistem akan langsung diterima oleh klien atau pengguna akhir. Dimana ada tim yang bernama tim pengembangan yang akan mengembangkan produk berdasarkan skenario pengguna dengan tujuan untuk memvalidasi sistem yang telah dikembangkan dengan sistem yang akan dibutuhkan oleh klien atau pengguna akhir, dan juga tim pengembangan ini memiliki tugas lain yakni membantu Klien atau pengguna akhir (Munthe et al., 2015).

2.9 **Penelitian Terdahulu**

Berdasarkan hasil penelitian terdahulu yang telah dikumpulkan, dengan adanya penggunaan teknologi informasi pada suatu proses terbukti dapat mengurangi waktu dan sumber daya yang dibutuhkan serta meningkatkan kecepatan penggunaan. Hal ini disebabkan karena adanya suatu proses yang berjalan secara otomatis oleh sistem sehingga proses tersebut menjadi lebih cepat. Dengan adanya hasil tersebut dapat menjadi solusi dalam proses hitung cepat pada pemilihan umum, dimana proses input data *entry* total perhitungan diinputkan secara manual oleh relawan dan kemudian pusat data *entry* akan menginputkan kembali data tersebut hal ini melalui proses yang panjang sehingga membutuhkan waktu yang lama, sumber daya yang lebih banyak dan rentan terhadap kesalahan dalam memasukkan data.

Dalam mengembangkan sistem yang sesuai untuk proses diperlukan pemahaman yang baik terhadap proses tersebut, apa saja fitur yang diperlukan dalam sistem yang dapat menyelesaikan permasalahan yang ada. Target pengguna menjadi salah satu indikator penentu *platform* apa sistem tersebut akan dibangun. Selain itu, perlu juga dilakukan perancangan *user interface* pada sistem dan

dilakukan validasi kepada pengguna yang akan menggunakan sistem tersebut agar sistem yang nantinya dibuat dapat digunakan oleh pengguna dengan mudah dan nyaman.

Pengembangan sistem dibutuhkan metode yang tepat agar dalam proses pembangunan sistem dapat berjalan dengan lancar. Dari penelitian terdahulu dapat disimpulkan bahwa metode yang cocok untuk pembangunan Sistem Hitung Cepat Pada Pemilihan Umum Berbasis *Web* dan *Mobile* yaitu metode *personal extreme programming* (PXP). Metode PXP memiliki kelebihan dapat beradaptasi terhadap perubahan dengan cepat dan tanpa harus mengulang proses pengembangan dari awal. Metode PXP sangat kompatibel dapat digunakan oleh pengembang yang hanya terdiri satu orang saja sehingga sangat cocok untuk digunakan dalam pengembangan Sistem Hitung Cepat ini yang hanya dikembangkan oleh satu pengembang saja.

Tahap pengujian pada sistem yang telah selesai dikembangkan memiliki peran penting untuk dilakukan. Hal ini bertujuan agar sistem yang telah dikembangkan dapat diketahui apakah sistem tersebut sudah memenuhi kebutuhan dari pengguna sistem dan juga dapat mengatasi permasalahan yang didefinisikan sebelumnya. Selain itu perlu adanya proses pembuatan dokumentasi agar Ketika sistem digunakan pengguna dapat menggunakan sistem dengan baik dan jika terdapat kesulitan terkait penggunaan sistem dapat merujuk pada dokumentasi sistem yang telah dibuat untuk mengatasi kesulitan tersebut.

Tabel 2. 1 Penelitian Terdahulu

No	Nama Penulis dan Tahun Publikasi	Hasil
1	(Wagearto, 2009)	<p>Metode : <i>COMPREHENSIVE PARALEL VOTE TABULATION</i></p> <p>Hasil : Dengan Aplikasi <i>Real Quick Count</i> perhitungan dilakukan secara otomatis oleh sistem serta proses pengiriman data lebih cepat dari proses manual. Penerapan CPVT pada Aplikasi <i>Quick Count</i> menghasilkan tingkat akurat yang tinggi dan ketelitian sebesar 99%.</p>

		<p>Pengembangan aplikasi real quick count sangat membantu KPU dalam menentukan Pemenang Pemilu di setiap daerah tanpa harus menunggu waktu yang lama.</p>
2	(Indra, 2013)	<p>Metode : <i>Metode Systematic Random Sampling</i></p> <p>Hasil : Kesimpulan yang didapatkan adalah dengan adanya aplikasi <i>quick count</i> berbasis sms dapat mengurangi indikasi kecurangan, proses perhitungan lebih cepat & akurat dan semoga dapat membantu KPU dalam hal perbandingan data hasil perhitungan suara</p>
3	(Suharhana & Wedasari, 2014)	<p>Metode : Mengumpulkan informasi dan mempelajari sistem kerja pemungutan suara di TPS serta melakukan observasi & wawancara.</p> <p>Hasil : dengan sistem <i>quick count</i> proses perhitungan suara pemilihan Kepala Desa berlangsung cepat dan efisien karena pengiriman menggunakan sms membutuhkan waktu yang sangat singkat.</p>
4	(RIZANI, 2017)	<p>Metode : Perancangan Sistem Menggunakan ERD</p> <p>Hasil : perancangan sistem quick count bertujuan untuk membantu perhitungan suara saat pemilu dengan hasil yang lebih cepat, mudah dan akurat</p>
5	(Yunianto, 2017)	<p>Metode : Metode Perancangan Sistem Informasi</p> <p>Hasil : Dengan Sistem informasi <i>Quick Count</i> Lembaga survei Radiopanjakarta dimudahkan dalam hal menampilkan hasil surveinya menjadi lebih cepat dan akurat. Kemudian dapat</p>

		mengurangi jumlah tenaga data <i>entry</i> yang bertugas mencatat hasil pada TPS.
6	(Pradipta et al., 2017)	<p>Metode : <i>Waterfall</i></p> <p>Hasil : Pengembangan aplikasi dapat membuat proses menjadi lebih efisien. Aplikasi yang dihasilkan dapat benar-benar membantu pengguna.</p>
7	(Luthfi, 2017)	<p>Metode : <i>Metode Prototyping</i></p> <p>Hasil : <i>Framework</i> Laravel cocok untuk digunakan sebagai <i>Back-end</i> yang kemudian dapat dihubungkan kepada aplikasi lain berbasis <i>mobile</i> maupun <i>website</i>.</p>
8	(Chen et al., 2017)	<p>Metode : <i>RESTFUL API</i></p> <p>Hasil : <i>Restful web service</i> yang canggih pada Laravel terdiri dari beberapa komponen yaitu <i>routing engine, middleware, MVC Framework, ORM (Object Relational Mapping)</i>, dan komponen Autentikasi. Arsitektur <i>Microservice</i> menjadi standar industri, model teknologi pada <i>web service</i> dan <i>Restful web</i> dikarenakan <i>microservice</i> bersifat ringan, skalabilitas, dan kompatibilitas <i>protocol HTTP</i>.</p>
9	(Karami, 2018)	<p>Metode : <i>Parallel Vote Tabulation</i></p> <p>Hasil : Pada Aplikasi <i>SMS Gateway Quick Count</i>, Sistem dapat membantu untuk melihat hasil perolehan suara dengan cepat, dapat</p>

		memprediksi hasil perhitungan suara yang dilakukan tanpa menunggu hasil dari KPU.
10	(Nurdin et al., 2018)	<p>Metode : Metode <i>Random Sampling</i> .</p> <p>Hasil : Aplikasi <i>quick count</i> dapat membantu Ketika hasil yang telah dikirimkan oleh relawan langsung diproses oleh sistem dan relawan dapat langsung mengirimkan hasil perhitungan suara langsung menggunakan <i>smartphone</i>.</p>
10	(Marthasari et al., 2018)	<p>Metode : <i>Personal Extreme Programming</i></p> <p>Hasil : Dari penelitian ini dapat disimpulkan bahwa PXP cocok untuk studi kasus yang diambil karena membutuhkan waktu yang cepat, hanya diperlukan 1 programmer dan diharuskan memenuhi kebutuhan pengguna dengan baik, karena PXP berfokus pada pemenuhan kebutuhan utama.</p>
11	(Asri et al., 2018)	<p>Metode : <i>Personal Extreme Programming</i></p> <p>Hasil : Penerapan metode PXP dalam <i>Job Training</i>, membantu proses pengembangan sehingga lebih cepat namun masih tetap dapat memenuhi kebutuhan pengguna, dibuktikan dari hasil kuesioner yang sangat bagus dapat diartikan bahwa penerapan PXP memberikan kemudahan.</p>
12	(Wijonarko & Aji, 2018)	<p>Metode : <i>Phonegap dan React Native</i></p> <p>Hasil : React native cocok untuk pengembangan aplikasi berbasis mobile dikarenakan dapat membuat aplikasi Android dan IOS hanya dengan satu bahasa yaitu Javascript.</p>

13	(Wiguna et al., 2019)	<p>Metode : <i>Classic Life Cycle</i></p> <p>Hasil : Pengembangan aplikasi <i>Distro Management Sistem</i> terbukti dapat memudahkan pengguna dalam melakukan transaksi dan manajemen produk. Penentuan platform aplikasi yang akan dibuat ditentukan dari kebutuhan pengguna yang menjadi target.</p>
14	(Nugraha et al., 2019)	<p>Metode : <i>SAINTE LAGUE</i></p> <p>Hasil : Berdasarkan hasil penelitian, dibandingkan dengan cara konvensional, dapat disimpulkan Sistem Hitung Cepat dapat menyampaikan hasil perhitungan suara tanpa menunggu lama. Sistem hitung cepat berbasis <i>mobile</i> dapat digunakan untuk proses pengiriman hasil perhitungan suara.</p>
15	(Bagwan & Ghule, 2019)	<p>Metode : <i>Framework Laravel</i></p> <p>Hasil : Laravel merupakan salah satu solusi terbaik untuk mencegah masalah keamanan dan untuk mengembangkan berbasis <i>web</i> terbaik pada sebuah produk. Laravel menyediakan dokumentasi yang mudah untuk dipelajari, mudah dalam mengerjakan proyek secara bersama dengan pengembang lain, dan menyediakan fitur terbaik serta paket bawaan pada <i>framework</i>.</p>
17	(Laaziri et al., 2019)	<p>Metode : <i>PHP Frameworks</i></p> <p>Hasil : Laravel lebih baik ketika digunakan untuk pengembangan proyek <i>web</i> berskala besar yang membutuhkan pengiriman lebih cepat dengan lebih sedikit sumber daya pada saat yang sama.</p>

18	(Rofiq, 2019)	<p>Sedangkan Symfony lebih baik untuk aplikasi kompleks di tingkat perusahaan, dan juga membutuhkan pengembang dengan keterampilan yang unggul. Memilih Kerangka kerja PHP harus bergantung pada sumber daya dan rencana pembangunan jangka panjang dan untuk membantu pengembang dalam menentukan pilihan terbaik.</p> <p>Metode : <i>RESTFul Web Service</i></p> <p>Hasil : Pengumpulan data suara lebih cepat dikarenakan data input dapat dilakukan pada tps melalui aplikasi, Sistem ini dapat mengurangi terjadinya kesalahan data karena telah melewati beberapa tahap validasi.</p>
19	(Sudarsono et al., 2020)	<p>Metode : <i>Personal Extreme Programming</i></p> <p>Hasil : Hasil penelitian ini mendukung kesimpulan dari beberapa penelitian sebelumnya tentang PXP, disebutkan penerapan PXP di bidang perhotelan sangat responsive, cepat, dan cocok untuk pengembangan sistem yang kebutuhannya sering berubah</p>