

I. PENDAHULUAN

1.1 Latar Belakang

PT Teknologi Kode Indonesia adalah salah satu perusahaan dalam bidang *Information and Communication Technology* (ICT). PT Teknologi Kode Indonesia yang mencakup berbagai bidang, yaitu *Web Development*, *Mobile Application*, *Integration System*, dan *Information System*. PT.Teknologi Kode Indonesia memiliki beberapa divisi, yaitu HRD (*Human Resource Departement*), *General Affair*, *Android Developer*, *Mobile Developer*, *Fullstack Developer*, *Backend Developer*, *Frontend Developer*, *Lead Developer*, dan *Office Boy*.

Semakin berkembangnya PT Teknologi Kode Indonesia, perusahaan ini belum memiliki aplikasi untuk mengolah data tugas *Office Boy*, sehingga akan sangat menyulitkan dalam mengolah dan menyimpan daftar tugas *Office Boy* yang sudah dikerjakan. Daftar tugas yang dimiliki *Office Boy* saat ini masih berbentuk dokumen cetak yang diisi manual menggunakan pena dengan cara di-*checklist*. Setelah itu *Office Boy* menyerahkan daftar tugas yang sudah diisi secara langsung ke *General Affair*. Permasalahan yang mungkin terjadi pada skenario diatas adalah daftar kerja yang telah selesai hilang atau rusak, ini dapat merugikan perusahaan karena saat proses dokumentasi perusahaan akan terdapat banyak data yang kosong dikarenakan dokumen daftar tugas hilang atau rusak.

Berdasarkan permasalahan diatas, dibutuhkan suatu aplikasi berbasis web yang dapat memudahkan *Office Boy* dan *General Affair* dalam mengelola data daftar tugas. Aplikasi ini dapat menyimpan data laporan tugas yang telah dikerjakan oleh *Office Boy* sehingga jika ada daftar tugas yang rusak atau hilang dapat dicetak ulang dan *General Affair* dapat dengan mudah mengelola data daftar tugas melalui aplikasi ini.

1.2 Tujuan

Tujuan dari dibangunnya aplikasi berbasis web ini adalah memudahkan pekerjaan baik dari *Office Boy* maupun *General Affair* untuk mengelola dan mengontrol tugas yang sudah dibuat. Hadirnya aplikasi ini, tugas tersebut akan selesai dengan cepat dan tidak perlu *checklist* tugas menggunakan pena.

1.3 Kerangka Pemikiran

Menurut analisis yang ada, proses dalam mengelola data tugas *Office Boy* masih menemukan kendala. Kendala yang didapat berupa, proses pengisian *checklist* masih menggunakan pena dan kertas, sulitnya proses pelaporan kepada *General Affair* apabila tugas sudah diselesaikan. Hal ini membuat proses kerjanya kurang efektif. Dengan hal tersebut, maka perlunya dibangun sebuah aplikasi yang bisa mengatasi masalah tersebut. Aplikasi ini berguna untuk mengelola data tugas *Office Boy*, proses pengisian *checklist* tidak menggunakan lagi kertas dan pena, dan membuat laporan tugas yang selesai dengan aman. Metode pengembangan aplikasi ini menggunakan metode RAD (*Rapid Application Development*). Bagan kerangka pemikiran bisa dilihat pada Gambar 1.



Gambar 1. Kerangka Pemikiran

1.4 Kontribusi

Aplikasi *Checklist System* Tugas OB Pada PT. Teknologi Kode Indonesia Berbasis *Web Framework Laravel* ini diharapkan memberikan kontribusi dari berbagai pihak, diantaranya :

Bagi PT. Teknologi Kode Indonesia :

1. Memudahkan perusahaan mengetahui laporan tugas yang sudah dikerjakan.
2. Mengoptimalkan pengceklisan *Office Boy* dalam tugasnya.
3. Mempermudah *General Affair* mengelola dan menyimpan tugas *office boy* yang telah dibuat.

Bagi Mahasiswa

1. Untuk menerapkan ilmu dari mata kuliah yang telah didapatkan selama kuliah dalam dunia kerja.
2. Untuk menghasilkan dan menyelesaikan tugas akhir mahasiswa

II. TINJAUAN PUSTAKA

2.1 *Aplikasi Web*

Aplikasi *web* adalah bagian dari *client-side* yang dapat dijalankan oleh *browser* web yang mendukung interaksi pengguna melalui antarmuka berbasis *web* (Susanti, 2017).

2.2 *Checklist*

Checklist merupakan suatu metode dalam penelitian yang memberikan keterangan yang mengenai muncul atau tidaknya perilaku dengan memberikan tanda cek jika yang diamati muncul (Santi, 2019).

2.3 *Sistem*

Sistem adalah sebuah kesatuan yang memiliki beberapa elemen yang terhubung, sehingga dapat digunakan untuk mencapai tujuan tertentu (Sihombing, 2016).

2.4 *DBMS (Database Management System)*

Database Management System (DBMS) adalah aplikasi komputer dimana aplikasi ini mempunyai fungsi dalam membantu manajemen data seperti proses pemasukkan (Insert), memodifikasi (Update), menghapus (Delete) serta memperoleh data (Select) sesuai dengan kebutuhan (Githa, 2017).

2.5 *Framework*

Framework adalah seperangkat *library* yang terorganisir dalam arsitektur untuk memberikan kecepatan, akurasi, kenyamanan dan konsistensi dalam pengembangan aplikasi (Jaya, 2017).

2.6 GIT

GIT adalah *software* (perangkat lunak) untuk mengontrol versi atau untuk memanajemen sebuah *project* aplikasi yang diciptakan oleh Linus Torvalds (Siregar, 2019).

2.7 Laravel

Laravel merupakan PHP yang sedang populer saat ini. Sejak diciptakan oleh Taylor Otwell, *laravel* kini mendapatkan banyak pujian dari developer seluruh dunia. (Hamid, 2019).

2.8 Hypertext Preprocessing (PHP)

PHP itu bahasa pemrograman berbasis *web*. Jadi, *PHP* itu adalah bahasa pemrograman yang digunakan untuk membuat aplikasi berbasis *web* (Agustina, 2018).

2.9 MySQL

MySQL merupakan *software RDBMS (server database)* yang dapat mengelola *database* dengan sangat cepat, dapat menampung data dalam jumlah sangat besar, dapat diakses oleh banyak *user (multi-user)* (Hesinto, 2018).

2.10 HTML (Hypertext Mail Language)

HTML (Hypertext Markup Language) adalah bahasa dasar untuk web scripting bersifat client side yang memungkinkan untuk menampilkan informasi dalam bentuk teks, grafik, serta multimedia dan juga untuk menghubungkan antartampilan web page (*hyperlink*) (Sovia, 2017)

2.11 Database

Database merupakan penggabungan beberapa tabel *ter-manage* sehingga dapat dipakai ke beberapa pemakai atau program lain untuk diakses dan dimanipulasi sesuai kebutuhan (Umar, 2019).

2.12 *Bootstrap*

Bootstrap merupakan gabungan dari *CSS* dan *Javascript* sebagai alternatif penggunaan *framework* lainnya dengan maksud untuk menghadirkan konsistensi terhadap tahap *interface development* dalam membangun sebuah *website* (Martin, 2018).

2.13 **XAMPP**

XAMPP adalah salah satu paket instalasi *Apache*, *PHP*, dan *MySQL* secara instant yang dapat digunakan untuk membantu proses instalasi ketiga produk tersebut (Sihotang, 2018).

2.14 *Web*

Website atau disingkat web, dapat diartikan sekumpulan halaman yang terdiri atas beberapa laman yang berisi informasi dalam bentuk data digital, baik berupa teks, gambar, video, audio, dan animasi lainnya yang disediakan melalui jalur koneksi internet (Christian, 2018).

2.15 **MVC (Model View Controller)**

MVC merupakan metode untuk pengembangan aplikasi yang memisahkan data (model) dari tampilan atau *frontend(view)* dan *logic* dari aplikasi itu sendiri. MVC memisahkan pengembangan aplikasi berdasarkan komponen utama yang membangun sebuah aplikasi seperti manipulasi data, antarmuka pengguna dan kontrol dalam sebuah aplikasi (Wardani, 2019). Definisi arsitektur MVC dibagi menjadi 3 bagian.

a. *Model*

Model merupakan salah satu dari bagian MVC yang bertugas berhubungan langsung dengan database. Bisa dikatakan juga bahwa *Model* adalah penghubung setiap alur program yang berhubungan dengan data. Cara membuat *model* menggunakan artisan dengan mengetikkan `php artisan make:model nama_model`.

b. View

Blade adalah *template engine* bawaan dari laravel. *Blade* memiliki kode kode yang lebih mudah untuk menghasilkan laravel. Cara membuat file *blade* dilakukan secara manual, dengan membuat nama_file.php.blade di dalam folder *views*.

c. Controller

Controller adalah suatu proses yang bertujuan untuk mengambil permintaan, menginisialisasi, memanggil model untuk dikirimkan ke *view*. Untuk membuat *controller* ada 2 cara yaitu; membuat manual di folder *controller* dengan nama ekstensi *controller*, membuatnya di cmd lokasi folder *controller* berada dengan mengetik php artisan make:controller nama_controller.

2.16 RAD (Rapid Development Application)

RAD merupakan metode proses pembangunan perangkat lunak yang tergolong teknik yang dimana proses pengerjaan dalam sebuah sistem sangat lebih efisien dan cukup singkat dalam proses pengerjaannya. Metode ini sangat berguna untuk proses memahami kebutuhan kebutuhan apa saja yang di gunakan untuk proses pengembangan perangkat lunak (Sagala, 2018).

Langkah – langkah Metode RAD :

- Rencana Kebutuhan (*Requirement/Planning*)

Pada tahapan pertama berfungsi untuk mengidentifikasi dan analisis layanan, batasan, dan obyektifitas dari sistem pengumpulan data.

- Proses Desain Sistem (*Design System*)

Pada tahapan kedua berfungsi untuk melakukan perancangan sistem berdasarkan analisis yang telah di lakukan pada tahapan sebelum nya. Tahapan *Planning* dan *Design* sangat cukup berkaitan hingga memperoleh rancangan sistem yang benar–benar memenuhi kebutuhan yang diinginkan.

- Implementasi (*Implementation*)

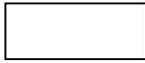
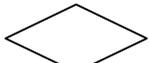
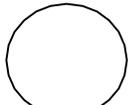
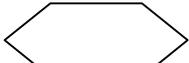
Pada tahapan ini berfungsi untuk melakukan pengujian terhadap program tersebut apakah terdapat kesalahan atau tidak. Jika proses pengujian telah

berhasil maka akan menghasilkan sistem yang lengkap.

2.17 Flowchart

Flowchart adalah representasi secara simbolik dari suatu algoritma untuk menyelesaikan suatu masalah, dengan menggunakan *Flowchart* akan memudahkan pengguna melakukan pengecekan bagian-bagian yang terlupakan dalam analisis masalah. *Flowchart* membantu memahami urutan-urutan logika yang rumit dan panjang (Magawan, 2017). Simbol-simbol flowchart terlihat pada Tabel 1.

Tabel 1. Simbol-simbol dalam *Flowchart*

Simbol	Nama	Keterangan
	<i>Input/Output</i>	Menunjukkan proses <i>input</i> <i>output</i>
	Terminal	Menyatakan awal dan akhir pada suatu program
	Proses	Menunjukkan proses yang dilakukan
	<i>Decision</i>	Menyatakan suatu kondisi yang menghasilkan 2 pilihan ya/tidak
	<i>Offline connector</i>	Menghubungkan dari satu proses ke proses lainnya dalam halaman yang berbeda
	<i>Connector</i>	Menghubungkan dari proses ke proses lainnya dalam halaman yang sama
	<i>Pradefined Process</i>	Menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk member mengenai harga awal.
	<i>Punched Card</i>	Menyatakan <i>input</i> berasal dari kartu atau output yang berasal dari kartu

Tabel 1. (Lanjutan)

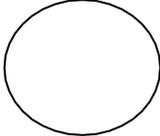
	<i>Flow</i>	Menyatakan aliran arus suatu proses
	<i>Punch Tape</i>	Menyatakan <i>input</i> atau <i>output</i> yang menggunakan kertas berlubang
	Document	Digunakan untuk mencetak output dalam bentuk document

Sumber : Sitorus (2015)

2.18 DFD (*Data Flow Diagram*)

DFD (*Data Flow Diagram*) atau dalam bahasa Indonesia menjadi Diagram Alir Data adalah representasi grafik yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengatur dari masukan (*input*) dan keluaran (*output*) (Yuliantikno, 2019). Simbol-simbol DFD terlihat pada Tabel 2.

Tabel 2. Simbol-simbol DFD

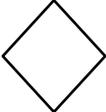
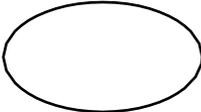
Simbol	Nama	Keterangan
	Input-output	Menyatakan proses <i>input</i> atau <i>output</i> tanpa tergantung jenis peralatannya
	Storage	Menyatakan <i>input</i> berasal dari <i>disk</i> atau <i>output</i> disimpan ke <i>disk</i> .
	Document	Menunjukkan simpanan dari data yang dapat berupa file <i>database</i> ke sistem komputer
	Display	Menunjukkan arus data yang dapat berupa masukan untuk sistem atau hasil dari proses sistem

Sumber : Yuliantikno (2019)

2.19 ERD (*Entity-Relationship Diagram*)

Entity-relationship diagram (ERD) adalah pemodelan awal basis data yang dikembangkan berdasarkan teori himpunan dalam bidang matematika. ERD digunakan untuk pemodelan basis data relasional (Andini, 2016). Simbol-simbol ERD terlihat pada Tabel 3.

Tabel 3. Simbol-simbol ERD

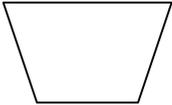
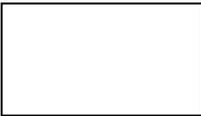
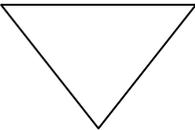
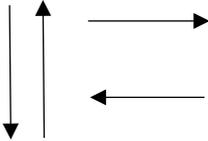
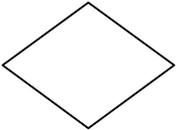
Simbol	Nama	Keterangan
	Entitas	Berupa orang, kejadian, atau benda di mana data akan dikumpulkan
	Relationship	Menunjukkan hubungan antar 2 entitas,
	Atribut	Merupakan properti dari entitas, dan dideskripsikan dengan kata kerja
	Link	Sebagai penghubung entitas dan relationship, serta entitas dan atribut

Sumber : Andini (2016)

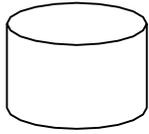
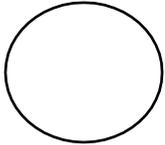
2.20 Mapping Chart

Mapping Chart atau *Flow Map* merupakan bagan yang menunjukkan aliran atau komunikasi dokumen di dalam program atau prosedursistem secara logika (Marliani, 2018). Simbol-simbol *mapping chart* terlihat pada Tabel 4.

Tabel 4. Simbol-simbol *mapping chart*

Simbol	Nama	Keterangan
	Dokumen	Menunjukkan dokumen input dan output baik proses manual, mekanik atau komputer
	Proses Manual	Menunjukkan pekerjaan yang dilakukan secara manual
	Proses Komputerisasi	Menunjukkan kegiatan proses dari operasi program komputer
	File Arsip	File non komputer yang diurutkan dan diarsipkan berdasarkan huruf, angka maupun tanggal
	Arus Proses	Menunjukkan arus dari proses
	Keputusan	Menunjukkan pemilihan kondisi dan pemilihan keputusan

Tabel 4. Lanjutan

	File	Menunjukkan input atau Output menggunakan hardisk/komputer
	Konektor	Menunjukkan penghubung dalam satu halaman
	Entry Data	Menunjukkan pengisian data

Sumber : Marliani (2018).

2.21 Blackbox Testing

BlackBox Testing adalah suatu metode yang berfungsi untuk menguji semua fungsi kebutuhan software dari menu satu dengan menu yang lain (Andani, 2019).

2.22 Visual Studio Code

Visual Studio Code adalah *software* yang sangat ringan, namun kuat editor kode sumbernya yang berjalan dari desktop. Muncul dengan *built-in* dukungan untuk JavaScript, naskah dan Node.js dan memiliki *array* beragam ekstensi yang tersedia untuk bahasa lain, termasuk C ++, C # , *Python*, dan PHP (Fajriani, 2020).

2.23 Artikel Terkait

Artikel terkait merupakan teori yang diambil dari beberapa referensi jurnal penelitian yang digunakan sebagai dasar penelitian. Referensi yang diambil dapat berupa kesamaan dalam studi kasus. Berikut ini merupakan beberapa referensi jurnal yang digunakan oleh penulis :

1. Sukirman, (2013) dalam jurnal yang berjudul “Aplikasi *Targeting Checklist* Dengan Menggunakan *Website* di PT. Telview Technology Bandung” dengan tujuan menangani permasalahan target karyawan dan penugasan setiap karyawan. Penelitian ini diuji dengan menggunakan metode *blackbox testing*
2. Yoga Hanggara, (2012) dalam jurnal yang berjudul “Analisis Sistem Informasi Pengelolaan Data Alumni Sekolah Berbasis *Codeigniter PHP Framework*” dengan tujuan membuat aplikasi mengelola data alumni SMA (Sekolah Menengah Atas) dalam segi penambahan, perubahan, pencarian data, dan menampilkan inoformasi. Penelitian ini memiliki tujuan untuk mengembangkan sistem pengelolaan data alumni sekolah yang masih menggunakan dokumen tertulis.