

**SISTEM INFORMASI JADWAL SIDANG PKL  
(PRAKTIK KERJA LAPANGAN) *REALTIME***



**Disusun oleh:  
Danar Widi Utomo  
NIM: A1316021**

**PRODI TEKNIK INFORMATIKA  
POLITEKNIK NEGERI TANAH LAUT  
PELAIHARI  
2018**

# DAFTAR ISI

<b>DAFTAR ISI .....</b>	<b><i>ii</i></b>
<b>BAB I PENDAHULUAN.....</b>	<b><i>1</i></b>
1.1    Latar Belakang.....	1
1.2    Rumusan Masalah.....	2
1.3    Tujuan .....	2
1.4    Manfaat .....	3
1.5    Batasan Masalah .....	3
<b>BAB II LANDASAN TEORI.....</b>	<b><i>4</i></b>
2.1    Konsep Dasar Sistem .....	4
2.2    Konsep Dasar Informasi .....	5
2.3    Konsep Dasar Sistem Informasi .....	6
2.4    Konsep Dasar <i>Web</i> .....	11
2.5    Konsep Dasar NodeJS.....	12
2.6    Konsep Dasar Node Package Manager .....	13
2.7    Konsep Dasar Representational State Transfer (REST) .....	14
2.8    Konsep Dasar Application Programming Interface API .....	14
2.9    Konsep Dasar VueJS.....	15
2.10   Konsep Dasar Android.....	15
2.10.1   Sistem Operasi Android.....	15
2.10.2   Android SDK .....	16
2.10.3   Android Studio.....	16
2.11   Konsep Dasar <i>Database</i> .....	17
2.11.1   Konsep Database .....	17
2.12   Konsep Dasar DFD ( <i>Data Flow Diagram</i> ) .....	20
<b>BAB III METODE PENELITIAN.....</b>	<b><i>22</i></b>

<b>3.1</b>	<b>Kerangka Penelitian.....</b>	<b>22</b>
<b>3.2</b>	<b>Jenis Penelitian.....</b>	<b>24</b>
<b>3.3</b>	<b>Metode Penelitian.....</b>	<b>24</b>
3.3.1	Desain Penelitian .....	24
3.3.2	Jenis dan Metode Pengumpulan Data.....	24
3.3.3	Metode Pendekatan dan Pengembangan Sistem .....	25
<b><i>BAB IV PERANCANGAN SISTEM.....</i></b>		<b>28</b>
<b>4.1</b>	<b>Deskripsi Umum Sistem.....</b>	<b>28</b>
<b>4.2</b>	<b>Analisa Kebutuhan Perangkat Lunak.....</b>	<b>28</b>
4.2.1	Kebutuhan fungsional .....	29
4.2.2	Kebutuhan <i>non-fungsional</i> .....	29
<b>4.3</b>	<b>Alat dan Bahan.....</b>	<b>29</b>
4.3.1	Perangkat Keras ( <i>Hardware</i> ).....	29
4.3.2	Perangkat Lunak ( <i>Software</i> ) .....	30
<b>4.4</b>	<b>Desain Sistem .....</b>	<b>30</b>
4.4.1	Work Flow.....	30
4.4.2	Context Diagram.....	30
4.4.3	Data Flow Diagram.....	31
4.4.4	Desain User Interface.....	34
4.4.5	Entity Relationship Diagram .....	37
<b>4.5</b>	<b>Implementasi Perancangan.....</b>	<b>38</b>
<b><i>BAB V KESIMPULAN DAN SARAN.....</i></b>		<b>39</b>
<b>5.1</b>	<b>Kesimpulan.....</b>	<b>39</b>
<b>5.2</b>	<b>Saran.....</b>	<b>39</b>
<b><i>Daftar Isi .....</i></b>		<b>40</b>

# **BAB 1**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Teknologi informasi dan komunikasi saat ini sudah merupakan bagian yang tidak dapat dipisahkan dari kehidupan manusia modern. Pada era globalisasi sekarang ini, teknologi informasi dan komunikasi memegang peranan penting dalam berbagai bidang dalam aspek kehidupan manusia. Perkembangan dunia teknologi membuat sistem komputer ikut berkembang. Perkembangan ini berdampak luas pada semua sektor, termasuk dalam bidang pendidikan.

Dalam bidang pendidikan dewasa ini khususnya tingkat Perguruan Tinggi, sudah membutuhkan proses pengolahan data jadwal yang sangat cepat, efektif, efisien dan akurat untuk meningkatkan kualitas pendidikan pada Perguruan Tinggi tersebut. Salah satunya adalah Program Studi Teknik Informatika pada Politeknik Negeri Tanah Laut.

Pada saat pelaksanaan sidang PKL (Praktik Kerja Lapangan) di Program Studi Teknik Informatika pada Politeknik Negeri Tanah Laut, koordinator sidang akan mengintruksikan untuk membuat jadwal sidang menggunakan aplikasi Microsoft Excel. Setelah dibuat, jadwal tersebut akan di-*share* oleh panitia seperti yang telah diinstruksikan oleh koordinator sidang PKL. Pemberitahuan jadwal ditempelkan lewat papan pengumuman yang ada di Gedung 1 Prodi Teknik Informatika dan *group social chat* seperti Whatsapp dan Line Messenger. Mahasiswa lain yang ingin menghadiri sidang PKL, bisa memperoleh informasi dipapan Pengumuman ataupun *group* tersebut.

Setelah Penulis melakukan pengamatan pada saat pelaksanaan sidang PKL, ternyata Program Studi Teknik Informatika membutuhkan suatu sistem informasi jadwal, karena pengolahan data jadwal masih menggunakan sistem komputerisasi sederhana yaitu dengan menggunakan Microsoft Excel dan untuk pengumuman jadwal sidang masih di letakkan dipapan pengumuman. Sidang PKL ini tidak bisa dipastikan dilaksanakan sesuai dengan jadwal yang sudah ditentukan dalam pengumuman karena adanya beberapa faktor seperti, laporan peserta sidang yang masih belum selesai, perangkat yang akan didemokan masih memiliki kendala atau

*bug*, keadaan peserta sidang yang tidak memungkinkan untuk melaksanakan sidang dan faktor cuaca. Oleh karena itu, jika terjadi perubahan jadwal maupun ruang sidang, maka informasi yang ada pada papan pengumuman menjadi *outdate*. Sehingga para penonton sidang yang ingin menyaksikan menjadi kesulitan untuk mencari informasi terkini.

Pada permasalahan tersebut untuk mengoptimalkan sistem jadwal, Penulis mengusulkan sistem baru untuk jadwal sidang PKL. Dalam sistem ini, pembuatan jadwal tidak akan lagi menggunakan aplikasi Microsoft Excel, tapi dengan aplikasi yang dapat mengelola jadwal seperti menambah, mengedit dan menghapus data. Aplikasi ini akan bersifat *realtime*, dalam artian apabila terjadi perubahan data, perubahan tersebut akan langsung ditampilkan. Sebagai sarana untuk mempermudah mahasiswa memperoleh informasi, aplikasi akan dibuat secara *multiplatform*, yaitu dapat diakses menggunakan *smartphone*, *desktop* dan perangkat lainnya yang mendukung koneksi internet. Kemudian untuk menampilkan informasi jadwal sidang di Gedung 1, papan pengumuman diganti dengan televisi yang akan menampilkan jadwal sidang secara *realtime*.

### **1.2 Rumusan Masalah**

Berdasarkan latar belakang yang ada Penulis menulis beberapa masalah dibawah ini.

1. Bagaimana cara membuat Sistem Informasi Jadwal Sidang PKL di Prodi Teknik Informatika pada Politeknik Negeri Tanah Laut berbasis *multiplatform* yang efektif dan lebih efisien dari sistem komputerisasi sederhana?
2. Bagaimana membuat Sistem Informasi Jadwal Sidang PKL di Prodi Teknik Informatika pada Politeknik Negeri Tanah Laut secara *realtime* menampilkan informasi yang akurat dan selalu *up to date*?

### **1.3 Tujuan**

Sesuai dengan rumusan masalah diatas, maka tujuan dari pengamatan ini adalah sebagai berikut.

1. Mendiskripsikan cara membuat Sistem Informasi Jadwal Sidang PKL di Prodi Teknik Informatika pada Politeknik Negeri Tanah Laut berbasis

*multiplatform* yang efektif dan lebih efisien dari sistem komputerisasi sederhana.

2. Mendiskripsikan cara membuat Sistem Informasi Jadwal Sidang PKL di Prodi Teknik Informatika pada Politeknik Negeri Tanah Laut secara *realtime* menampilkan informasi yang akurat dan selalu *up to date*.

#### **1.4 Manfaat**

Adapun manfaat dari pengamatan ini adalah sebagai berikut.

1. Bagi panitia sidang PKL, Sistem informasi ini dapat diterapkan di Prodi Teknik Informatika pada Politeknik Negeri Tanah Laut guna menampilkan informasi secara *realtime*.
2. Bagi mahasiswa, dengan adanya Sistem Informasi ini pencarian informasi jadwal sidang PKL dapat didapatkan dengan mudah. Sehingga dapat meminimalisir kesalahan informasi yang didapat dari sumber lain.
3. Bagi dosen penguji dan pembimbing, dapat mengurangi kesalahan memasuki ruang sidang PKL.

#### **1.5 Batasan Masalah**

1. Penulis hanya melakukan pengamatan di ruang lingkup Prodi Teknik Informatika pada Politeknik Negeri Tanah Laut. Pengamatan ini dilakukan Penulis hanya pada saat agenda sidang PKL dilaksanakan.
2. Adapun Sistem Informasi Sidang PKL ini dibuat, Penulis membuatnya berbasis *web* dan *mobile*. *Web* digunakan untuk menampilkan data di media proyektor seperti televisi dan *mobile* akan digunakan di *smartphone* yang berbasis Android.

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Konsep Dasar Sistem**

##### **a. Pengertian Sistem**

Suatu sistem adalah suatu entity (kesatuan) yang terdiri dari bagian-bagian yang saling berhubungan (disebut subsistem) yang bertujuan untuk mencapai tujuan-tujuan tertentu (Moscove, 1982: 4). Menurut Murdick (1978: 12), suatu sistem adalah kumpulan elemen-elemen yang dijadikan satu untuk tujuan umum. Ada beberapa konsep yang penting dalam pengembangan sistem untuk memudahkan memahami pengertian sistem, seperti yang dikemukakan oleh Al Fatta (2007: 7-9) yaitu:

##### **1) Dekomposisi**

Dekomposisi adalah pembagian sistem ke dalam komponen-komponen yang lebih kecil yang disebut subsistem.

##### **2) Modularitas**

Konsep modularitas berhubungan dengan dekomposisi. Modul-modul ini diharapkan dapat membuat beban kerja pengembangan sistem dapat didistribusikan secara merata sehingga pengembangan sistem menjadi lebih sederhana karena hanya terfokus pada satu modul terlebih dahulu, baru kemudian diintegrasikan antar modul.

##### **3) *Coupling***

Modul-modul yang memiliki ketergantungan harus dipasangkan agar dapat diketahui modul yang dapat bekerja secara independen dan modul yang harus diselesaikan sebelum modul lain.

##### **4) Kohesi**

Dari proses coupling akan didapatkan kelompok modul dengan karakteristik yang hampir sama. Kemudian munculah konsep kohesi dimana kelompok modul harus dianalisis bersama kelompok modul yang saling berkohesi.

##### **b. Karakteristik Sistem**

Karakteristik atau sifat-sifat sistem dapat membedakan satu sistem dengan sistem yang lain. Berikut ini adalah karakteristik sistem menurut Al Fatta (2007: 5-6):

1) Batasan (*boundary*)

Penggambaran dari suatu elemen atau unsur mana yang termasuk di dalam sistem dan mana yang di luar sistem.

2) Lingkungan (*environment*)

Segala sesuatu di luar sistem, lingkungan yang menyediakan asumsi, kendala, dan input terhadap suatu sistem.

3) Masukan (*input*)

Sumber daya (data, bahan baku, peralatan dan energi) dari lingkungan yang dikonsumsi dan dimanipulasi oleh suatu sistem.

4) Keluaran (*output*)

Sumber daya atau produk (informasi, laporan, dokumen, tampilan layar komputer, dan barang jadi) yang disediakan untuk lingkungan sistem oleh kegiatan dalam suatu sistem.

5) Komponen (*component*)

Kegiatan-kegiatan atau proses dalam suatu sistem yang mentransformasikan input menjadi bentuk setengah jadi (output). Komponen ini bisa merupakan subsistem dari sebuah sistem.

6) Penghubung (*interface*)

Tempat dimana komponen atau sistem dan lingkungannya bertemu atau berinteraksi.

7) Penyimpanan (*storage*)

Area yang dikuasai dan digunakan untuk penyimpanan sementara. Penyimpanan merupakan media penyangga di antara komponen tersebut bekerja dengan berbagai tingkatan dan memungkinkan komponen yang berbeda dari berbagai data yang sama.

## **2.2 Konsep Dasar Informasi**

### **a. Pengertian Informasi**



Informasi adalah data yang telah diolah kedalam bentuk yang berguna bagi penerimanya dan nyata atau berupa nilai yang dapat dipahami dalam keputusan sekarang maupun di masa depan (Gordon B. Davis, 1974: 32).

Informasi merupakan hal yang sangat penting untuk suatu sistem. Sumber dari informasi adalah data. Data merupakan kenyataan yang menggambarkan kejadian dan kesatuan nyata. Kejadian adalah sesuatu yang terjadi pada saat tertentu. Kesatuan nyata berupa objek nyata seperti tempat, benda dan orang yang benar-benar ada dan terjadi.

b. Kualitas Informasi

Kualitas dari suatu informasi (quality of information) bergantung pada tiga hal, yaitu informasi harus akurat (accurate), tepat waktu (timely base) dan informasi harus relevan (relevance). Menurut John Burch dan Gary Grudnitski (Jogiyanto, H.M., 2005: 10-11), kualitas informasi tergantung dari tiga hal, yaitu:

1) Akurat

Artinya informasi harus bebas dari kesalahan-kesalahan dan tidak menyesatkan. Akurat juga berarti informasi harus jelas menggambarkan maksudnya. Informasi harus akurat karena dari sumber informasi sampai ke penerima informasi banyak kemungkinan terjadi gangguan atau noise yang dapat merusak informasi tersebut.

2) Tepat pada waktunya

Artinya informasi informasi yang datang ke penerima tidak boleh terlambat. Informasi yang sudah usang tidak akan memiliki nilai informasi lagi. Pengambilan keputusan yang terlambat akan berakibat fatal untuk organisasi, karena informasi merupakan landasan dalam mengambil keputusan

3) Relevan

Artinya informasi tersebut harus sesuai dengan kebutuhannya dan mempunyai manfaat untuk pemakainya

## **2.3 Konsep Dasar Sistem Informasi**

a. Pengertian Sistem Informasi

Suatu sistem informasi adalah suatu kegiatan dari prosedur-prosedur yang di organisasikan, bilamana dieksekusi akan menyediakan informasi untuk mendukung pengambilan keputusan dan pengendalian di organisasi (Henry C. Lucas, 1982: 8).

Al Fatta (2007: 12-13) membedakan sistem informasi yang berbasis komputer menjadi beberapa tipe, yaitu:

1) *Transaction Processing Systems* (TPS)

Transaction processing systems adalah sistem informasi terkomputerisasi yang dikembangkan untuk memproses sejumlah besar data untuk transaksi bisnis rutin.

2) *Management Information Systems* (MIS)

Management information systems adalah sebuah sistem informasi pada level manajemen yang berfungsi untuk membantu perencanaan, pengendalian dan pengambilan keputusan dengan menyediakan resume rutin dan laporan-laporan tertentu.

3) *Decision Support Systems* (DSS)

Decision support systems adalah sistem informasi pada level management dari suatu organisasi yang mengkombinasikan data dan model analisis canggih atau peralatan data analisis untuk mengambil yang semi terstruktur dan tidak terstruktur.

4) *Expert Sistem and Artificial Intellegent* (ES & AI)

Expert sistem adalah representasi dari pengetahuan yang menggambarkan cara seorang ahli dalam mendekati suatu masalah.

b. **Komponen Sistem Informasi**

Menurut Stair (Al Fatta, 2007: 9-10) sistem informasi berbasis komputer dalam suatu organisasi terdiri dari beberapa komponen, yaitu:

1) **Perangkat Keras**

Perangkat keras adalah perangkat keras komponen untuk melengkapi kegiatan memasukkan data, memproses data dan keluaran data.

2) **Perangkat Lunak**

Perangkat lunak adalah program dan instruksi yang debrikan ke komputer.

### 3) Basis Data

Basis data adalah kumpulan data dan informasi yang diorganisasikan sedemikian rupa sehingga mudah diakses oleh pengguna sistem informasi.

### 4) Telekomunikasi

Telekomunikasi adalah komunikasi yang menghubungkan antara pengguna sistem dengan komputer secara bersama-sama ke dalam suatu jaringan kerja yang efektif.

### 5) Manusia

Manusia adalah personal dari sistem informasi yang meliputi manajer, analis, programmer dan operator serta teknisi yang bertanggung jawab terhadap perawatan sistem.

John Burch dan Grudnitski (Jogiyanto, 2005: 12) mengatakan sistem informasi terdiri dari komponen-komponen yang disebut dengan blok bangunan (*building block*), yaitu:

#### 1) Blok Masukan (*input block*)

Blok masukan mewakili data masuk ke dalam sistem informasi. Input disini meliputi metode-metode dan media yang digunakan untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumen-dokumen dasar.

#### 2) Blok Model (*model block*)

Blok model terdiri dari kombinasi prosedur, logika, dan model matematik yang akan memanipulasi data input dan data yang tersimpan di basis data dengan cara yang sudah ditentukan untuk menghasilkan keluaran yang diinginkan.

#### 3) Blok Keluaran (*output block*)

Blok keluaran merupakan produk dari sistem informasi yang berupa informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkat manajemen serta pemakai sistem.

#### 4) Blok Teknologi (*technology block*)

Teknologi merupakan tool-box dalam sistem informasi. Teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan akses data, menghasilkan dan mengirimkan keluaran dan membantu pengendalian dari sistem secara menyeluruh.

5) Blok Basis Data (*database block*)

Merupakan kumpulan dari data yang saling berhubungan satu sama lainnya, tersimpan di perangkat keras komputer dan digunakan perangkat lunak 13 untuk memanipulasinya. Basis data diakses dan dimanipulasi dengan menggunakan perangkat lunak paket yang disebut DBMS (Database Management Sistem).

6) Blok Kendali (*control block*)

Beberapa pengendalian dirancang khusus untuk menanggulangi gangguan-gangguan terhadap sistem.

c. Pengembangan Sistem Informasi

Menurut Jogiyanto (2005: 36) Pengembangan sistem dapat berarti penyusunan suatu sistem yang baru untuk menggantikan sistem yang lama secara keseluruhan atau memperbaiki sistem yang telah ada. Pengembangan sistem informasi dapat diartikan sebagai tindakan merubah, mengganti atau menyusun sistem informasi yang telah digunakan baik secara keseluruhan atau sebagian agar menjadi sistem baru yang lebih baik. Beberapa hal yang menyebabkan sebuah organisasi melakukan pergantian sistem lama yaitu:

- 1) Adanya permasalahan (problems) yang muncul pada sistem lama.
- 2) Pertumbuhan organisasi.
- 3) Untuk meraih kesempatan-kesempatan.
- 4) Adanya instruksi dari atasan.

Proses-proses tersebut diaplikasikan ke dalam satu metode yang sering disebut dengan Sistem Development Life Cycle (SDLC) yang merupakan metode umum dalam pengembangan sebuah sistem. Berikut ini adalah tahapan-tahapan SDLC menurut Al Fatta (2007: 26):

- 1) Identifikasi dan seleksi proyek
  - a. Mengidentifikasi proyek-proyek yang potensial

Yaitu mengidentifikasi potensi seperti seberapa besar keuntungan yang bisa diperoleh, durasi yang tersedia, dan sumber daya yang dimiliki untuk menyelesaikan proyek.

b. Melakukan klasifikasi dan merangking proyek

Yaitu melakukan klasifikasi dan merangking proyek dari yang paling layak sampai yang dirasa kurang menguntungkan untuk dikerjakan.

c. Memilih proyek untuk dikembangkan

Yaitu menentukan proyek yang akan dikerjakan.

2) Inisiasi dan perancangan proyek

Sistem informasi yang potensial dijelaskan dalam tahap ini. Argumentasi untuk melanjutkan sistem juga dikemukakan. Pada tahap ini ditentukan secara detail rencana kerja yang harus dikerjakan, durasi yang diperlukan masing-masing tahap, sumber daya manusia yang ada, perangkat keras, perangkat lunak, maupun finansial.

3) Analisis

Dalam tahap ini sistem yang berjalan dipelajari dan sistem pengganti diusulkan. Kemudian dideskripsikan sistem yang sedang berjalan, masalah, dan kesempatan. Dan diusulkan rekomendasi umum untuk bagaimana memperbaiki atau mengganti sistem yang sedang berjalan.

4) Desain

Mengubah kebutuhan yang masih dalam berupa konsep menjadi spesifikasi sistem yang nyata. Ada dua tahap dalam tahapan desain, yaitu desain logis (logical design) dan desain fisik (physical design).

a. Desain logis

Merupakan tahapan untuk memilih semua fitur-fitur fungsional dari tahapan analisis sehingga akan menghasilkan deskripsi-deskripsi sebagai berikut:

- (1) Deskripsi fungsional mengenai data dan proses yang ada dalam sistem baru.
- (2) Deskripsi detail dari spesifikasi sistem, meliputi input, output dan proses.

b. Desain fisik

Pada tahap ini spesifikasi dirubah ke dalam detail teknologi dimana pemrograman dan pengembangan sistem dapat diselesaikan. Tahapan ini akan menghasilkan beberapa deskripsi yaitu:

- (1) Deskripsi teknis, tentang pilihan teknologi perangkat lunak dan perangkat keras yang digunakan.
- (2) Deskripsi detail dari spesifikasi sistem meliputi modul-modul program, file-file, sistem perangkat lunak dan sistem jaringan.

Pada tahapan desain ada beberapa aktivitas utama yang dilakukan yaitu:

- (1) Merancang dan mengintegrasikan jaringan.
- (2) Merancang arsitektur aplikasi.
- (3) Mendesain antar muka pengguna.
- (4) Mendesain sistem antar muka.
- (5) Mendesain dan mengintegrasikan database.
- (6) Membuat prototype untuk detail desain.
- (7) Mendesain dan mengintegrasikan kendali sistem.

5) Implementasi

Implementasi sistem merupakan tahap meletakkan sistem agar siap dioperasikan. Ada beberapa hal yang dilakukan yaitu:

- a) Pengkodean (*Coding*)
- b) Pengujian (*Testing*)
- c) Instalasi

6) Pemeliharaan

Merupakan tahapan sistematis memperbaiki dan meningkatkan kinerja sistem.

## 2.4 Konsep Dasar *Web*

*World Wide Web* atau yang biasa disingkat dengan *WWW* adalah suatu ruang informasi yang digunakan oleh *Uniform Resource Identifier* (URL) untuk mengidentifikasi sumber daya yang berguna. *WWW* merupakan suatu kumpulan dari *web server* dari seluruh dunia yang menyediakan data dan

informasi untuk dapat digunakan bersama. *Web* tidak hanya memberikan akses data berupa teks, tetapi juga berupa gambar, suara, video dan animasi.

*Web* merupakan kumpulan dari dokumen-dokumen yang sangat banyak yang tersimpan pada komputer *server* (*web server*). *Web server* tersebar di seluruh benua dan saling terhubung dalam satu jaringan internet. Format yang digunakan untuk menyimpan dokumen-dokumen tersebut adalah *Hypertext Markup Language* (HTML). Suatu halaman *web* dapat terdiri dari teks yang saling terkait dengan teks lain atau dokumen lain. Keterkaitan teks antar halaman di sini disebut juga dengan *hypertext*, sedangkan keterkaitan antar dokumen biasa disebut dengan *hypermedia*.

Dari uraian di atas maka dapat disimpulkan bahwa WWW adalah kumpulan dokumen multimedia yang saling terhubung menggunakan *hypertext link*. Untuk dapat berpindah dari satu dokumen ke dokumen lainnya maka hyperlink harus di klik. Dokumen-dokumen dalam halaman *web* ini dapat diakses melalui *web browser*.

## 2.5 Konsep Dasar NodeJS

Node.js adalah perangkat lunak yang didesain untuk mengembangkan aplikasi berbasis web dan ditulis dalam sintaks bahasa pemrograman JavaScript. Bila selama ini kita mengenal JavaScript sebagai bahasa pemrograman yang berjalan di sisi client / browser saja, maka Node.js ada untuk melengkapi peran JavaScript sehingga bisa juga berlaku sebagai bahasa pemrograman yang berjalan di sisi server, seperti halnya PHP, Ruby, Perl, dan sebagainya. Node.js dapat berjalan di sistem operasi Windows, Mac OS X dan Linux tanpa perlu ada perubahan kode program. Node.js memiliki pustaka server HTTP sendiri sehingga memungkinkan untuk menjalankan server web tanpa menggunakan program server web seperti Apache atau Nginx. (Lutfi, 2017)

Proses mengeksekusi Javascript sebagai bahasa server diperlukan engine yang cepat dan mempunyai performansi yang bagus. Oleh sebab itu Engine Javascript dari Google bernama V8 yang dipakai oleh Node.js yang juga merupakan engine yang dipakai oleh browser Google Chrome. Node.js pertama kali diciptakan dan diperkenalkan oleh Ryan Dahl, pada tahun 2009

sehingga JavaScript bisa digunakan sebagai bahasa pemrograman di sisi server, sekelas dengan PHP, ASP, C#, Ruby dan lain sebagainya. Ryan Dahl adalah seorang pengembang dari Joyent yang merupakan sebuah perusahaan perangkat lunak dan infrastruktur Cloud. Ia memiliki ketertarikan dengan penerapan single-threaded pada bahasa pemrograman sisi server dan akhirnya memilih JavaScript sebagai bahasa untuk Node, setelah sebelumnya mencoba menggunakan Haskell, Lua dan C.

#### Kelebihan Node JS

1. Node.js menggunakan bahasa pemrograman JavaScript yang diklaim sebagai bahasa pemrograman yang paling populer dan banyak dikenal oleh masyarakat luas
2. Node.js mampu menangani ribuan koneksi bersamaan dengan penggunaan resource minimum untuk setiap prosesnya
3. Node.js sangat diandalkan terutama untuk membuat aplikasi real-time
4. Node.js adalah project open source, sehingga siapapun dapat melihat struktur kode dan juga dapat berkontribusi untuk pengembangannya
5. Penggunaan JavaScript di sisi server dan juga client meminimalisir ketidakcocokan antar dua sisi lingkungan pemrograman, seperti terkait komunikasi data yang mana menggunakan struktur JSON yang sama di kedua sisi, validasi form yang sama yang dapat dijalankan di sisi server dan client, dan sebagainya.
6. Database NoSQL seperti MongoDB dan CouchDB mendukung Javascript sehingga interfacing dengan database ini akan jauh lebih mudah.
7. Node.js memakai V8 yang selalu mengikuti perkembangan standar ECMAScript (nama standar resmi dari JavaScript, Namun JavaScript yang lebih dikenal dalam implementasinya), sehingga tidak perlu ada kekhawatiran bahwa browser tidak mendukung fitur-fitur di Node.js.

#### **2.6 Konsep Dasar Node Package Manager**

NPM atau Node Package Manager adalah paket manajer yang digunakan pada bahasa pemrograman JavaScript. NPM merupakan bagian dari NodeJS



yang digunakan untuk mempermudah programmer dalam membuat aplikasi. NPM terdiri dari command line client, dan online database yang dapat diakses secara public maupun premium. Paket manajer ini dibuat oleh Isaac Z. yang ditulis dengan bahasa JavaScript.

## **2.7 Konsep Dasar Representational State Transfer (REST)**

REST (REpresentational State Transfer) adalah suatu arsitektur metode komunikasi yang sering diterapkan dalam pengembangan layanan berbasis web. Arsitektur REST, yang umumnya dijalankan via HTTP (Hypertext Transfer Protocol), melibatkan proses pembacaan laman web tertentu yang memuat sebuah file XML atau JSON. File inilah yang menguraikan dan memuat konten yang hendak disajikan. Setelah melalui sebuah proses definisi tertentu, konsumen akan bisa mengakses antarmuka aplikasi yang dimaksudkan. Kekhasan REST terletak pada interaksi antara klien dan server yang difasilitasi oleh sejumlah tipe operasional (verba) dan Universal Resource Identifiers (URIs) yang unik bagi tiap-tiap sumberdaya. Masing-masing verba – GET, POST, PUT dan DELETE – memiliki makna operasional khusus untuk menghindari ambiguitas. REST kerap dipergunakan dalam mobile application, situs web jejaring sosial, mashup tools, dan automated business processes. Arsitektur REST yang decoupled (terpisah) serta beban komunikasi yang ringan antara produsen dan konsumen membuatnya populer di dunia cloud-based API, seperti yang disajikan oleh Amazon, Microsoft, dan Google. Layanan berbasis web yang menggunakan arsitektur REST semacam itu dinamakan RESTful APIs (Application Programming Interfaces) atau REST APIs. (Saputra, 2015)

## **2.8 Konsep Dasar Application Programming Interface API**

API adalah sekumpulan instruksi program dan protokol yang digunakan untuk membangun aplikasi perangkat lunak. API berperan sebagai pembawa pesan yang menerima permintaan pengguna dan memberitahu sistem apa yang harus dilakukan, lalu memberikan respons yang sesuai untuk permintaan tersebut. Sebagai analogi, dalam sebuah rumah makan anggaplah pelanggan adalah pengguna aplikasi dan dapur adalah sistem. Dapur akan menyiapkan pesanan atau permintaan pelanggan, dalam analogi ini, API

adalah pelayan yang mengkomunikasikan pesanan pelanggan ke dapur atau sistem dan membawa kembali makanan dan minuman yang sesuai dengan permintaan. Dalam contoh dunia nyata, API digunakan dalam aplikasi travel pemesanan penerbangan yang dimana aplikasi tersebut menghubungkan pengguna dengan berbagai macam maskapai. Ketika memesan penerbangan pengguna akan menentukan berbagai macam variabel seperti tujuan bandara, tanggal keberangkatan, jumlah penumpang, dan lainnya. Aplikasi travel akan menerima request tersebut dan berinteraksi dengan API dari setiap web database maskapai penerbangan yang ada dan memilih informasi berdasarkan variabel yang dimasukkan pengguna untuk kemudian memberikan respons untuk ditampilkan kembali ke pengguna. Prinsip yang sama juga berlaku ketika terjadi interaksi antara database aplikasi dengan gawai atau komputer, masing-masing sistem tersebut memiliki API yang memungkinkan untuk saling bertukar informasi. Dengan kata lain API adalah apa yang membawa informasi antara database aplikasi dan device pengguna untuk menyampaikan data dan menciptakan konektivitas antar sistem. (Muhano, 2016)

## **2.9 Konsep Dasar VueJS**

Vue.js biasa disebut sebagai Vue adalah kerangka kerja JavaScript progresif yang bersumber terbuka untuk membangun antarmuka pengguna. Integrasi ke dalam proyek yang menggunakan pustaka JavaScript lainnya mudah dilakukan dengan Vue karena dirancang untuk diadopsi secara bertahap. Vue juga bisa berfungsi sebagai kerangka kerja aplikasi web yang mampu menjalankan aplikasi single-page yang canggih.

## **2.10 Konsep Dasar Android**

### **2.10.1 Sistem Operasi Android**

Android adalah sistem operasi mobile yang berbasis open source linux kernel yang awalnya dibuat oleh Android Inc. Android termasuk sistem operasi yang dirancang untuk digunakan secara optimal dalam lingkungan mobile yang fleksibel. (Véronique Brossier, 2011: 19).

Android adalah sistem operasi open source berbasis Linux. Pada awalnya, android hanya untuk ponsel, tapi sekarang dapat digunakan pada tablet, TV, komputer, dan stereo mobil (Carlos Sessa, 2013).

#### **2.10.2 Android SDK**

Android SDK adalah tools API (Application Programming Interface) yang diperlukan untuk mulai mengembangkan aplikasi pada platform Android menggunakan bahasa pemrograman Java. (Nazruddin, 2014).

Android SDK menyertakan contoh projects dengan sumber kode, perangkat pengembangan, sebuah emulator, dan library yang dibutuhkan untuk membangun aplikasi Android (TERM, 2016).

Aplikasi ditulis dalam bahasa pemrograman Java dan dijalankan di atas Dalvik, sebuah mesin virtual didesain untuk dapat berjalan di lapisan atas kernel Linux.

#### **2.10.3 Android Studio**

Android Studio adalah sebuah IDE (Integrated Development Environment) yang digunakan untuk pengembangan aplikasi Android (Hafizh Herdi, 2014).

Android Studio merupakan IDE resmi untuk Android (Developers, 2016). Android Studio memiliki fitur editor kode cerdas (Intelligent Code Editor) yang memiliki kemampuan penyelesaian kode, optimalisasi, dan analisis kode yang canggih. Selain itu fitur New Project Wizards membuat proses memulai proyek baru menjadi jauh lebih mudah bahkan dapat mengimpor contoh kode Google dari GitHub. Berbagai modul baru digunakan dalam Android Studio ini, salah satunya adalah pengembangan aplikasi multi layar yang memudahkan pengembangan untuk membangun sebuah aplikasi untuk ponsel dan tablet Android, Android Wear, Android TV, Android Auto, dan Android Google Glass.

Fitur-fitur Android Studio (Developers, 2016) Antara lain sebagai berikut:

- 1) Dibuat dengan IntelliJ IDEA Community Edition, JAVA IDE populer karya JetBrains.

- 2) Sistem pembuatan Gradle yang fleksibel.
- 3) Dapat membangun berbagai generasi dan varian APK.
- 4) Dukungan template bertambah untuk Google Services dan aneka tipe perangkat.
- 5) Editor layout yang lengkap dengan dukungan pengeditan tema.
- 6) Lint Tools untuk solusi kinerja, kegunaan, kompatibilitas versi, dan masalah lain.
- 7) ProGuard dan kemampuan app-signing.
- 8) Dukungan bawaan untuk Google Cloud Platform, mempermudah integrasi Google Cloud Messaging dan App-Engine

## **2.11 Konsep Dasar *Database***

Data adalah bagian penting dari pemrograman modern sehingga keseluruhan bahasa program menyediakan fungsi untuk mengakses database. Standar utama untuk bahasa database adalah *Structured Query Language* (SQL). SQL distandarisasi sebagai bahasa untuk menciptakan database, menyimpan informasi ke dalam database, dan mendapatkan kembali informasi darinya. Aplikasi khusus dan lingkungan pemrograman mengkhususkan diri untuk menginterpretasikan data SQL. Seorang programmer akan memulai dengan menciptakan suatu struktur data di dalam SQL dan kemudian menulis suatu program dalam bahasa (PHP) untuk mengakses data tersebut. Program PHP kemudian bisa memformulasikan permintaan atau memperbaharui data tersebut, yang dilewatkan ke interpreter SQL.

### **2.11.1 Konsep Database**

#### **1) Entitas dan Relationship**

Entitas adalah berbagai hal dalam dunia nyata yang informasinya disimpan dalam database. Sebagai contoh, kita dapat menyimpan informasi pegawai dan bekerja untuk departemen tertentu. Dalam kasus ini, pegawai merupakan suatu entitas dan departemen juga merupakan entitas.

*Relationship* adalah hubungan antar entitas. Sebagai contoh, seorang pegawai bekerja untuk suatu departemen. Bekerja untuk adalah *relationship* antara entitas pegawai dan entitas departemen.

*Relationship* terdiri dari tiga derajat berbeda, yakni *one-to-one*, *one-to-many* (*many-to-one*), dan *many-to-many*.

*One-to-one* menghubungkan secara tepat dua entitas dengan satu kunci (*key*). Misalnya, dalam suatu perusahaan satu orang pegawai memiliki satu komputer saja.

*One-to-many* (*many-to-one*) merupakan hubungan antar entitas dimana kunci (*key*) pada satu tabel muncul berkali-kali dalam tabel lainnya. Misalnya, banyak pegawai bekerja untuk satu perusahaan.

*Many-to-many* merupakan hubungan yang sering menyebabkan permasalahan dalam prakteknya. Dalam hubungan *many-to-many*, kunci utama (*primary key*) dari tabel kedua dapat muncul beberapa kali pada tabel pertama. Misalnya, dalam suatu perusahaan, banyak pegawai bekerja untuk banyak departemen. Untuk mengatasi permasalahan tersebut, dibutuhkan tabel antara.

## 2) Relasi atau Tabel

Database terdiri dari sekumpulan relasi atau tabel. Relasi dan tabel memiliki arti yang sama. Perhatikan contoh tabel pegawai pada tabel berikut.

No Pegawai	Nama	Pekerjaan	Kd_Departmen
123	Rina Erlinda	Developer	14
234	Roni	Programmer	15
345	Nadia	Developer	14
456	Ricco	DBA	16

## 3) Kolom atau *Attribute*

Dalam tabel database, setiap kolom atau *attribute* menjelaskan beberapa bagian record data yang disimpan dalam tabel. Kolom adalah bagian dari tabel, sedangkan suatu *attribute* berkaitan dengan entitas dunia nyata yang merupakan pemodelan tabel. Seperti pada tabel pegawai sebelumnya, dapat dilihat bahwa setiap pegawai memiliki satu

NoPegawai, Nama, Pekerjaan, dan KodeDepartemen yang merupakan kolom, tetapi sering juga disebut attribute dari tabel pegawai.

#### 4) Baris, Record, dan Tuple

Pada tabel pegawai sebelumnya, setiap baris pada tabel mewakili suatu record pegawai. Setiap baris dalam tabel sering juga disebut record atau tuple yang terdiri dari suatu nilai untuk setiap kolom dalam tabel.

#### 5) Kunci (*Key*)

Suatu *superkey* adalah suatu kolom yang dapat digunakan untuk mengidentifikasi suatu baris dalam tabel. Suatu *key* adalah suatu minimal *superkey*. Sebagai contoh, pada tabel pegawai sebelumnya, kita dapat menggunakan NoPegawai dan Nama secara bersama-sama untuk mengidentifikasi baris-baris dalam tabel. Kita juga dapat menggunakan seluruh kolom sebagai *superkey*.

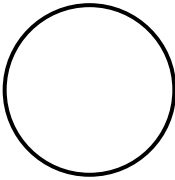

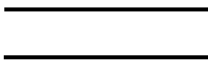

Namun, kita tidak membutuhkan seluruh kolom tersebut untuk mengidentifikasi suatu baris. Kita hanya butuh NoPegawai. Ini adalah suatu minimal *superkey* yang merupakan suatu minimal kolom yang dapat digunakan untuk mengidentifikasi suatu baris tunggal, maka NoPegawai adalah suatu *key*.

Kita dapat mengidentifikasi seorang pegawai dengan Nama atau dengan NoPegawai yang merupakan dua *key* yang disebut *candidate key*. Disebut *candidate key* karena dengan kedua kolom tersebut kita akan memilih salah satu yang menjadi *primary key*. *Primary key* adalah suatu kolom atau sekumpulan kolom yang akan digunakan untuk mengidentifikasi secara tunggal setiap baris dari suatu tabel. Dalam hal ini, kita akan membuat NoPegawai sebagai *primary key* karena secara umum Nama mungkin ada yang sama. *Foreign key* menyatakan hubungan antar tabel. Sebagai contoh, pada tabel pegawai dapat dilihat bahwa KodeDepartemen menyimpan suatu nomor departemen yang akan disimpan dalam suatu tabel terpisah dengan *primary key*-nya adalah KodeDepartemen.

### 2.12 Konsep Dasar DFD (*Data Flow Diagram*)

Data flow diagram adalah sebuah teknik grafis yang menggambarkan aliran informasi dan transformasi yang diaplikasikan pada saat bergerak dari input menjadi output (Pressman, 2002).

DFD dapat digunakan untuk menyajikan sebuah sistem atau perangkat lunak pada setiap tingkat abstraksi. Notasi dasar yang digunakan untuk menciptakan suatu DFD dijelaskan pada table 2.1 dibawah ini:

No	Gambar	Keterangan
1		Lingkaran menunjukkan sistem secara keseluruhan. Penamaan sebuah lingkaran dapat berupa kata, frase atau sebuah kalimat sederhana yang menjelaskan nama sistem itu sendiri.
2		Panah menunjukkan arah aliran data dari sistem ke entity luar atau sistem ke data store atau sebaliknya. Gambar panah diberinama untuk menunjukkan data yang mengalir melalui aliran tersebut.
3		Data store (penyimpanan) digambarkan dengan dua buah garis sejajar mendatar, menunjukkan sekumpulan data yang disimpan. Data store berfungsi untuk tempat penyimpanan data yang dapat digunakan dalam sistem baik sebagai input untuk melakukan suatu proses untuk kemudian digunakan oleh proses-proses lainnya di dalam sistem.
4		Terminator digambarkan dengan sebuah kotak persegi panjang yang berhubungan dengan sistem. Sebuah terminator dapat berupa orang, sekumpulan orang, departement dalam perusahaan atau

		organisasi yang sama tetapi berada di luar sistem yang dimodelkan.
--	--	--

DFD memiliki beberapa level yang merupakan hasil perjejangan dari suatu sistem yang dibuat (level 0, 1, 2, dst).

1) DFD level 0 (*diagram context*)

DFD level 0 menggambarkan seluruh elemen sistem dengan sebuah proses tunggal dengan data input atau output yang ditunjukkan oleh anak panah yang masuk dan keluar secara berurutan.

2) DFD level 1

DFD level 1 menggambarkan proses dan jalur informasi pada DFD level 0 yang telah detail.

3) DFD level 2

DFD level 2 merupakan pengembangan alur data atau pemecahan proses menjadi proses terpisah-pisah, tapi masih berkaitan.

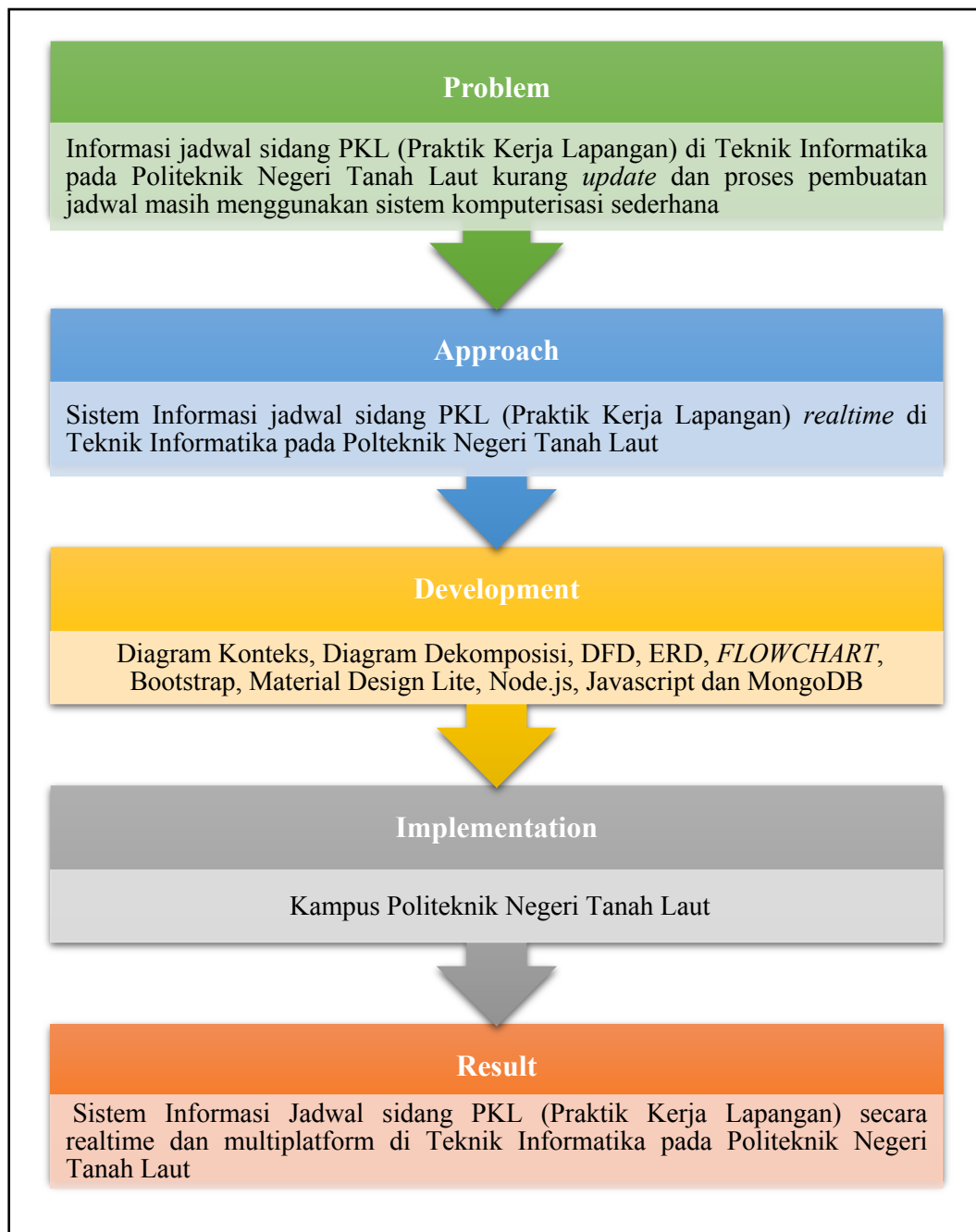


## BAB III

### METODE PENELITIAN

#### 3.1 Kerangka Penelitian

Penulis memiliki rencana atau kerangka penulisan dari latar belakang permasalahan, serta mempunyai pengaruh terhadap penanganan permasalahan. Adapun kerangka penelitian yang penulis buat adalah sebagai berikut.



Keterangan :

Kerangka	Keterangan
<i>Problem</i> (Masalah)	<i>Problem</i> merupakan masalah yang diperoleh sebagai alasan pembuatan Sistem Informasi jadwal sidang PKL (Praktik Kerja Lapangan) di Teknik Informatika pada Politeknik Negeri Tanah Laut. Adapun masalah yang ada di Politeknik Negeri Tanah Laut adalah dalam pelaksanaan sidang PKL, jadwal yang dibuat panitia terkadang masih memerlukan revisi karena faktor yang tidak memungkinkan terlaksananya sidang.
<i>Approach</i> (Pendekatan)	<i>Approach</i> merupakan saran yang dibuat sebagai solusi terhadap masalah yang ada, saran untuk mencapai tujuan atau hasil yang diinginkan, yaitu dengan membuat Sistem Informasi Jadwal sidang PKL secara <i>realtime</i> .
<i>Development</i> (Pengembangan)	<i>Development</i> merupakan sarana yang digunakan dalam pembuatan Sistem Informasi jadwal sidang PKL, Sistem Informasi ini akan di kembangkan dengan pemrograman berbasis objek yaitu menggunakan diagram <i>data flow diagram</i> (DFD), diagram <i>context</i> , diagram dekomposisi, menggunakan bahasa pemograman HTML dan Javascript serta menggunakan <i>framework</i> Bootstrap, Node Js dan <i>database</i> MongoDB.
<i>Implementasi</i> (Penerapan)	<i>Implementation</i> merupakan tempat studi kasus yang disarankan dalam pengimplementasian Sistem Informasi jadwal sidang PKL secara <i>realtime</i> di Teknik Informatika pada Polteknik Negeri Tanah Laut yaitu di Politeknik Negeri Tanah Laut.

<i>Result</i> (hasil)	<i>Result</i> merupakan suatu solusi terhadap masalah yang ada, dimana dengan terbangunnya Sistem Informasi Jadwal sidang PKL secara <i>realtime</i> di Teknik Informatika pada Politeknik Negeri Tanah Laut maka akan mempermudah mahasiswa maupun dosen untuk mengetahui informasi terkini tentang sidang.
-----------------------	--

### 3.2 Jenis Penelitian

Jenis penelitian yang digunakan dalam penelitian adalah model deskriptif. Penelitian deskriptif bertujuan untuk mengimplementasikan aplikasi ini pada Prodi Teknik Informatika di Politeknik Negeri Tanah Laut.

### 3.3 Metode Penelitian

#### 3.3.1 Desain Penelitian

Desain penelitian merupakan kerangka atau perincian prosedur kerja yang akan dilakukan pada waktu meneliti, sehingga diharapkan dapat memberi gambaran dan arah mana yang akan dilakukan dalam melaksanakan penelitian tersebut, desain penelitian yang baik dapat memudahkan kita dalam melakukan penelitian sesuai dengan tujuan yang telah ditetapkan sebelumnya.

Sesuai dengan tujuan yang ingin dicapai dalam penelitian ini, penulis mengumpulkan data sebagai bahan laporan dengan menggunakan metode deskriptif, yaitu suatu metode yang pada tahap pertama penulis melakukan dengan cara mengumpulkan data yang diperlukan terlebih dahulu dan pada tahap berikutnya, penulis mengolah dan membahas sampai pada suatu kesimpulan yang pada akhirnya dapat dibuat suatu laporan.

#### 3.3.2 Jenis dan Metode Pengumpulan Data

Adapun jenis dan pengumpulan data yang digunakan dalam penelitian adalah sebagai berikut:

##### 1. Observasi

Metode yang melakukan pengamatan secara langsung di Teknik Informatika pada Politeknik Negeri Tanah Laut yang diteliti yaitu pada proses kegiatan sidang PKL (Praktik Kerja Lapangan).

## 2. Literatur

Metode yang menggunakan buku dan literatur lainnya dalam penulisan yang dapat membantu proses pembangunan aplikasi.

### 3.3.3 Metode Pendekatan dan Pengembangan Sistem

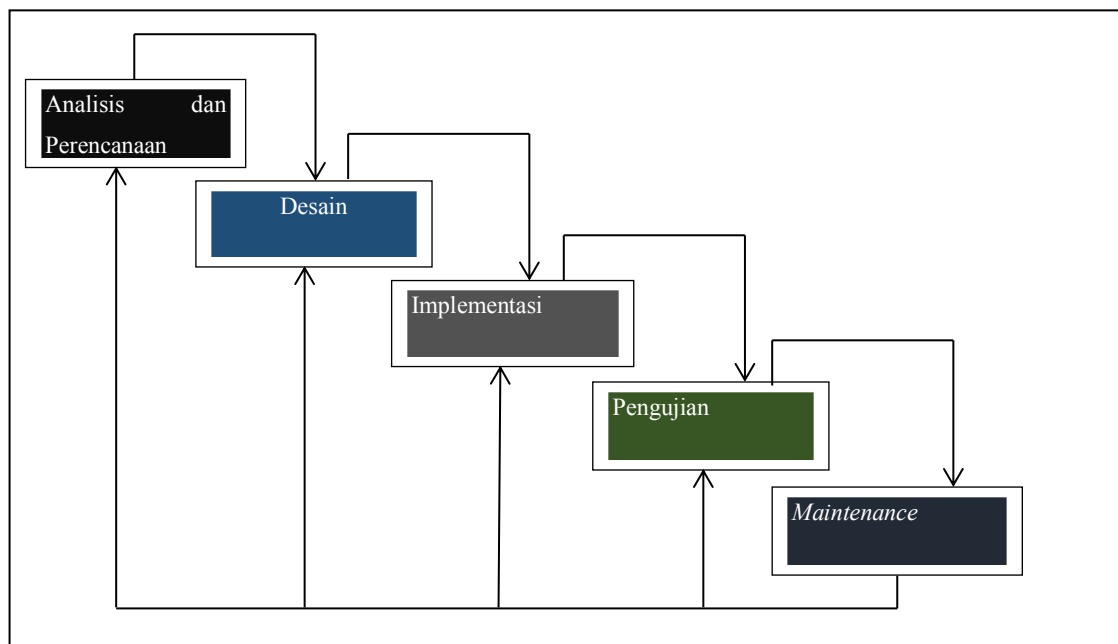
Dalam metode penelitian ada dua macam metode yaitu metode pendekatan dan pengembangan sistem yang akan dirincikan sebagai berikut:

#### 1. Metode Pendekatan Sistem

Metode pendekatan sistem yang digunakan adalah analisis dan perancangan terstruktur, alat yang digunakan untuk menggambarkan model adalah diagram konteks, *flowchart*, DFD, ERD.

#### 2. Metode Pengembangan Sistem

Model yang digunakan adalah model *waterfall*, atau sering disebut juga model Sekuensial Linear. Pada model ini, sistem akan dibuat secara sistematis dan sekuensial yang mulai pada tingkat dan kemajuan sistem pada seluruh analisis, desain, kode dan pengujian. Beberapa kelebihan model *waterfall* adalah menyediakan dokumentasi untuk tiap tahapan, tahapan sistematis, dan melakukan pemeriksaan keluaran setiap tahapan. Model *waterfall* dapat dilihat pada gambar dibawah ini.



**Keterangan:**

#### 1. Analisis dan Perencanaan

Langkah ini bertujuan untuk menganalisa semua komponen-komponen yang dibutuhkan oleh sistem yang akan dibangun, termasuk didalamnya analisa kebutuhan data. Data-data yang dibutuhkan diperoleh melalui observasi, studi pustaka, dan wawancara.

## **2. Desain**

Pada tahapan ini, pengembangan sistem mendefinisikan proses-proses dan kebutuhan-kebutuhan sistem yang berkaitan dengan pengembangan aplikasi (*software development process*). Dalam fase ini ditentukan pemilihan teknologi yang akan diterapkan baik berupa *Website integrated technology*, rancangan database, maupun beragam aspek lainnya yang berkaitan dengan kegiatan analisis dan perancangan ini.

## **3. Implementasi**

Pada tahapan ini hasil dari fase-fase sebelumnya dituangkan kedalam penulisan kode-kode dengan menggunakan bahasa pemrograman komputer yang telah ditentukan dalam tahap sebelumnya. Untuk melakukan pemrograman ini dibutuhkan perangkat-perangkat pemrograman seperti *PHPStorm*, *Browser*, *Node.js* dan *MongoDB* serta aneka perangkat lunak berkaitan lainnya sesuai dengan kebutuhan pemrograman bersangkutan.

## **4. Pengujian**

Setelah proses penulisan kode pemrograman langkah berikutnya berupa proses pengujian terhadap hasil pemrograman tersebut . Pengujian mencakup beragam aspek yang berkaitan dengan *Sistem and Performance* dari fase *Code Generation*. Pengujian-pengujian tersebut berupa Pengujian *Database*, Pengujian Validitas Data, Pengujian *Logic* Aplikasi, Pengujian Antar Muka Aplikasi *General User Interface* (GUI), Pengujian *User Administration*. Hasil pengujian ini merupakan Umpan balik perbaikan *Sistem and Performance* yang akan digunakan dalam proses perbaikan sistem hingga mencapai hasil yang diharapkan dan telah ditentukan sebelumnya.

## **5. Maintenance**

Tahapan ini melengkapi keseluruhan dari siklus SDLC (*Sistem Development Life Cycle*), dalam pemeliharaan sistem ini dilakukan penyesuaian dan

peningkatan sistem yang dilakukan secara audit dan periodic dengan membuat perubahan-perubahan berdasarkan dengan kondisi terbaru.

## **BAB IV**

### **PERANCANGAN SISTEM**

Pada bab ini akan diuraikan tentang desain perancangan serta implementasi dari perancangan Sistem Informasi Jadwal Sidang PKL di Prodi Teknik Informatika pada Politeknik Negeri Tanah Laut

#### **4.1 Deskripsi Umum Sistem**

Deskripsi umum sistem yang akan dibangun dalam penelitian ini adalah sistem jadwal sidang PKL yang dapat membantu mempermudah mahasiswa Politeknik Negeri Tanah Laut khususnya Prodi Teknik Informatika dalam mengetahui informasi terkini tentang sidang PKL yang sedang berlangsung pada saat itu. Selain itu aplikasi ini dibuat untuk mempermudah kinerja panitia PKL dalam melaksanakan tugasnya sebagaimana mestinya.

Sistem yang dibangun ini berbasis *web* dan *mobile*. Aplikasi berbasis web memudahkan operator PKL dalam mengelola semua tentang PKL. Dengan adanya aplikasi tersebut, operator tidak akan lagi mengelola jadwal pada aplikasi *office*, dan menyebarkan informasi-informasi pada media social, cukup dengan memasukkannya pada aplikasi, informasi akan secara otomatis terupdate. Sedangkan aplikasi *mobile*, akan memudahkan para mahasiswa dalam mengetahui informasi terkini mengenai sidang PKL.

#### **4.2 Analisa Kebutuhan Perangkat Lunak**

Analisis kebutuhan perangkat lunak dalam penelitian ini dilakukan dengan mengidentifikasi permasalahan yang ada untuk kemudian dicatat dan dijadikan bahan untuk mulai membangun aplikasi jadwal sidang PKL. Analisa kebutuhan yang dimaksud diantaranya adalah kebutuhan fungsional dan *non-fungsional* dari sistem jadwal sidang PKL. Kebutuhan fungsional adalah jenis kebutuhan yang berisikan proses – proses apa saja yang diberikan oleh sistem informasi tersebut. Sedangkan kebutuhan *non-fungsional* adalah batasan layanan atau fungsi yang ditawarkan sistem.

#### 4.2.1 Kebutuhan fungsional

Kebutuhan fungsional merupakan kebutuhan atau proses yang harus dikerjakan atau informasi yang harus dimuat sistem yang berkaitan dengan fungsi sistem. Kebutuhan fungsional dalam sistem jadwal PKL ini antara lain:

1. Sistem dibangun menggunakan aplikasi berbasis *web* dan *mobile*
2. Sistem dapat melakukan CRUD (create, review, edit, delete) data sidang PKL
3. Sistem dapat menampilkan daftar calon peserta selama sidang PKL berlangsung
4. Sistem dapat menampilkan *history* peserta sidang
5. Sistem dapat menampilkan informasi sidang PKL secara realtime
6. Sistem dapat melakukan CRUD pada status peserta sidang

#### 4.2.2 Kebutuhan *non-fungsional*

Kebutuhan non fungsional yaitu yang mengacu pada kinerja pada sebuah sistem maupun yang berhubungan dengan kemudahan penggunaannya dilihat dari segi kualitas. Kebutuhan non-fungsional ini dikerjakan apabila kebutuhan fungsional sudah terpenuhi. Adapun kebutuhan non-fungsional yang dimiliki oleh sistem antara lain:

1. Sistem menggunakan tampilan yang user friendly untuk mempermudah pengguna dalam pengoperasiannya.
2. Respon time untuk menampilkan tiap fitur maksimal 5 detik
3. Sistem dapat di operasikan pada sistem operasi apapun selama perangkat mempunyai browser dan koneksi internet.
4. Sistem untuk perangkat mobile khusus hanya untuk perangkat yang bersistem operasi Android.

#### 4.3 Alat dan Bahan

Alat penelitian yang digunakan dalam penelitian adalah *software* dan *hardware*. Adapun *software* dan *hardware* yang digunakan adalah sebagai berikut:

##### 4.3.1 Perangkat Keras (*Hardware*)

Perangkat keras yang digunakan dalam pembuatan aplikasi ini antara lain:

1. Laptop Asus X455LAB dengan spesifikasi  
Prosesor i3-4005U, RAM 6GB, SSD 128GB, HDD 500GB.



2. Mouse Logitech
3. Keyboard Fantech

#### **4.3.2 Perangkat Lunak (*Software*)**

Perangkat lunak yang digunakan dalam pembuatan aplikasi ini antara lain:

1. MacOS Yosemite 10.10.5
2. PHPStorm 2017.2
3. Sublime Text 3
4. WebStorm 2017.2
5. XAMPP
6. MYSQL

#### **4.4 Desain Sistem**

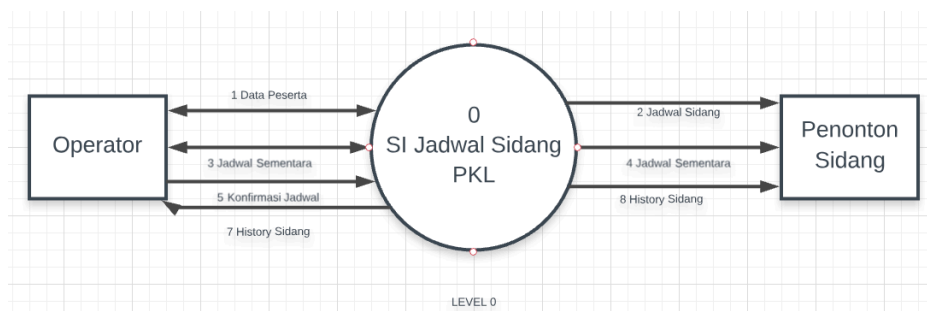
Desain sistem merupakan rancangan yang menggambarkan bagaimana suatu sistem akan dibentuk. Desain sistem yang akan dibuat meliputi *WorkFlow*, *Data Flow Diagram*(DFD), *Event List*, Kamus Data, dan *Entity Relationship Diagrams*(ERD).

##### **4.4.1 Work Flow**

Workflow merupakan suatu aliran kerja atau suatu informasi dari proses kerja maupun bisnis yang sistematis dimana dokumen atau informasi yang di buat dialirkan dari satu pihak ke pihak yang lain sesuai aturan atau prosedur tertentu yang telah disepakati bersama dalam sebuah organisasi atau perusahaan. Workflow pada sistem jadwal sidang PKL dapat dilihat pada gambar berikut.

##### **4.4.2 Context Diagram**

Context Diagram (CD) disebut juga diagram level nol atau diagram tingkat tertinggi yang belum dikerjakan secara rinci. Context Diagram (CD) menggambarkan konteks keseluruhan sistem secara umum. Pada Context Diagram (CD) hanya terdapat satu proses dengan aliran data yang masuk dan keluar dari sistem tanpa adanya penyimpanan (storage). Context Diagram pada sistem informasi jadwal sidang PKL dapat di lihat pada gambar sebagai berikut.



Context Diagram pada gambar ini juga bisa disebut sebagai DFD level 0. Context Diagram merupakan langkah awal dalam pembuatan Data Flow Diagram. Context Diagram tersebut menjelaskan ada dua aktor yang berperan dalam penerapan sistem jadwal keberangkatan ini, diantaranya operator dan penonton sidang. Operator bertugas menjalankan aplikasi berbasis web untuk manajemen jadwal sidang PKL pada hari yang bersangkutan. Pada Context Diagram terlihat bahwa operator bisa melakukan input dan view data Peserta, konfirmasi jadwal sementara, dan view data sementara dan view data history sidang. Penonton sidang pada Context Diagram ini merupakan user dari aplikasi berbasis mobile, aplikasi berbasis mobile ini dibuat bertujuan untuk mempermudah penonton sidang dalam mengetahui jadwal sidang yang berlangsung pada hari tersebut. Dalam Context Diagram tersebut dijelaskan bahwa penonton hanya bisa melihat data jadwal. Jadwal ini meliputi jadwal sidang secara keseluruhan, jadwal sidang pada hari tersebut dan history sidang.

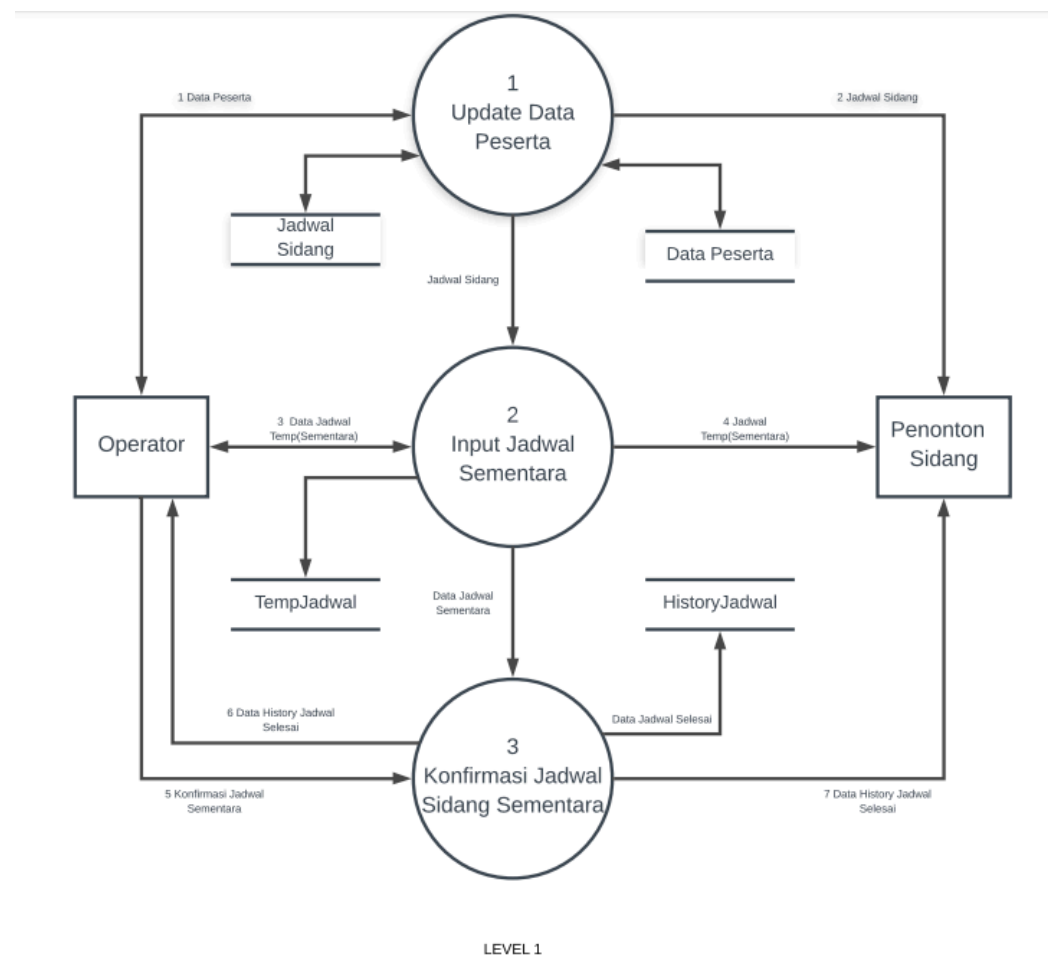
#### 4.4.3 Data Flow Diagram

Menurut Kristanto (2003), Data Flow Diagram (DFD) adalah suatu model logika data atau proses yang dibuat untuk menggambarkan dari mana asal data dan kamana tujuan data yang keluaran dari sistem, dimana data disimpan, proses apa yang menghasilkan data tersebut, dan interaksi antara data yang tersimpan dan proses yang dikenakan pada data tersebut.

##### 4.4.3.1 Data Flow Diagram Level 1

DFD level 1 menggambarkan tentang proses dan aliran data yang dijalankan oleh masing-masing *user* secara keseluruhan. DFD level 1 merupakan hasil *breakdown* dari *Context Diagram* atau DFD Level 0 yang sudah dibuat sebelumnya. DFD level 1 ditandai dengan penomoran satu digit pada setiap

proses nya. Data Flow Diagram level 1 pada sistem informasi jadwal sidang PKL dapat dilihat pada gambar berikut.



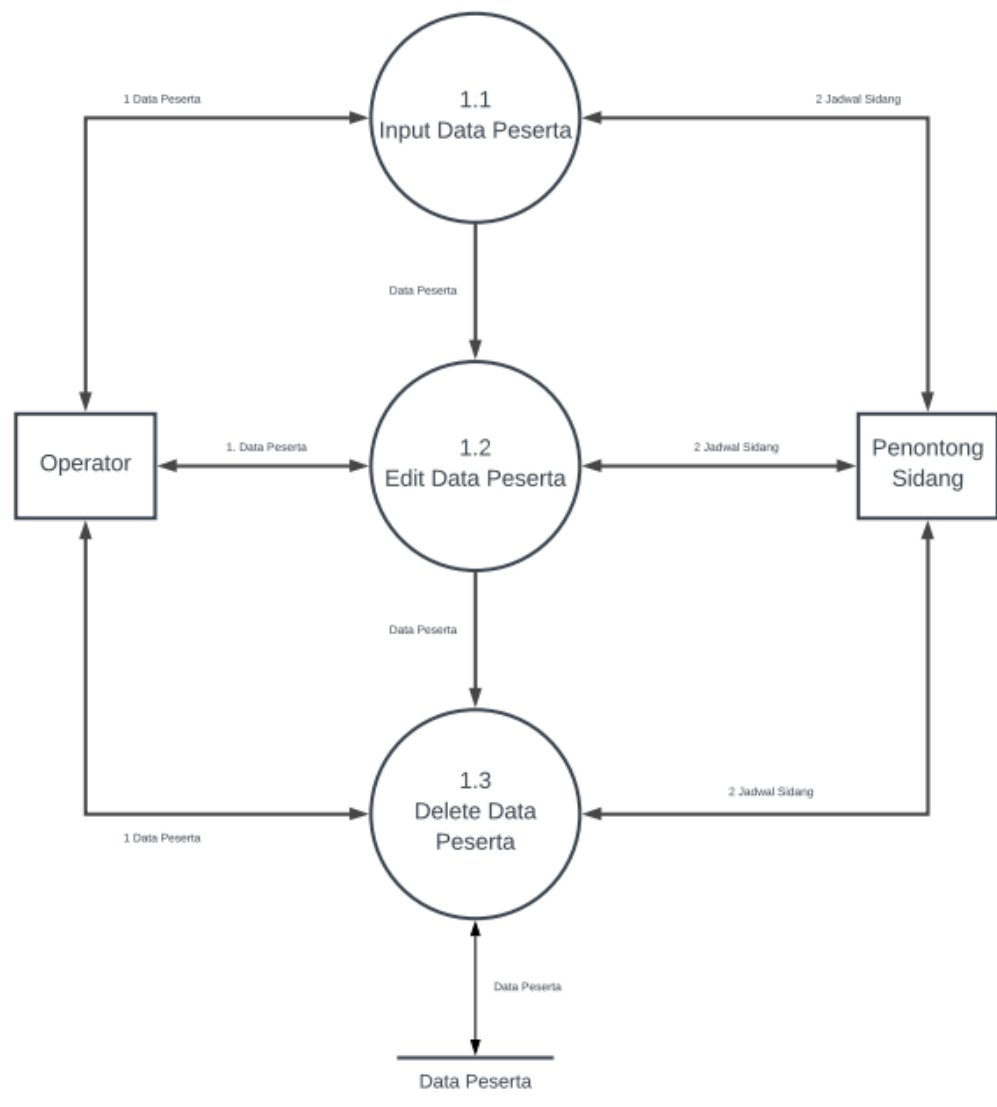
Data Flow Diagram Level 1 memiliki tiga proses seperti dijelaskan pada gambar diatas. Proses pertama merupakan proses update data peserta, pada proses ini operator bisa melakukan input dan view data peserta. Data peserta yang diinputkan operator akan disimpan oleh sistem kedalam storage data\_peserta, selanjutnya data tersebut dapat dilihat oleh penumpang sebagai data jadwal sidang PKL.

Proses yang kedua pada DFD level 1 ini adalah proses input data jadwal sementara. Penginisialisasi data jadwal sementara berarti data peserta yang melakukan sidang pada hari itu. Pada proses input data jadwal sementara operator dan penonton sidang dapat melihat data jadwal sementara tetepi hanya operator yang dapat mengedit status dari peserta ketika peserta mengalami kendala prasidang. Jadwal sementara selanjutnya disimpan pada storage TempJadwal.

Proses yang ketiga adalah proses konfirmasi data jadwal sidang sementara. Proses ini dilakukan oleh operator ketika jadwal sidang hari itu sudah dipastikan bahwa tidak ada mahasiswa yang akan menunda sidangnya. Setelah peserta sidang melakukan sidang, jadwal sementara akan di konfirmasi oleh operator selanjutnya sistem akan menampilkan history sidang yang dapat dilihat oleh operator dan penonton. Data history sidang tersebut disimpan kedalam storage historyJadwal.

#### 4.4.3.2 Data Flow Diagram Level 2

Data Flow Diagram level 2 merupakan hasil *breakdown* dari proses 1 pada Data Flow Diagram level 1. Pada Data Flow Diagram level 2 memaparkan proses yang tidak dijelaskan pada Data Flow Diagram level 1. Proses pada Data Flow Diagram level 2 dapat dilihat pada gambar dibawah ini.

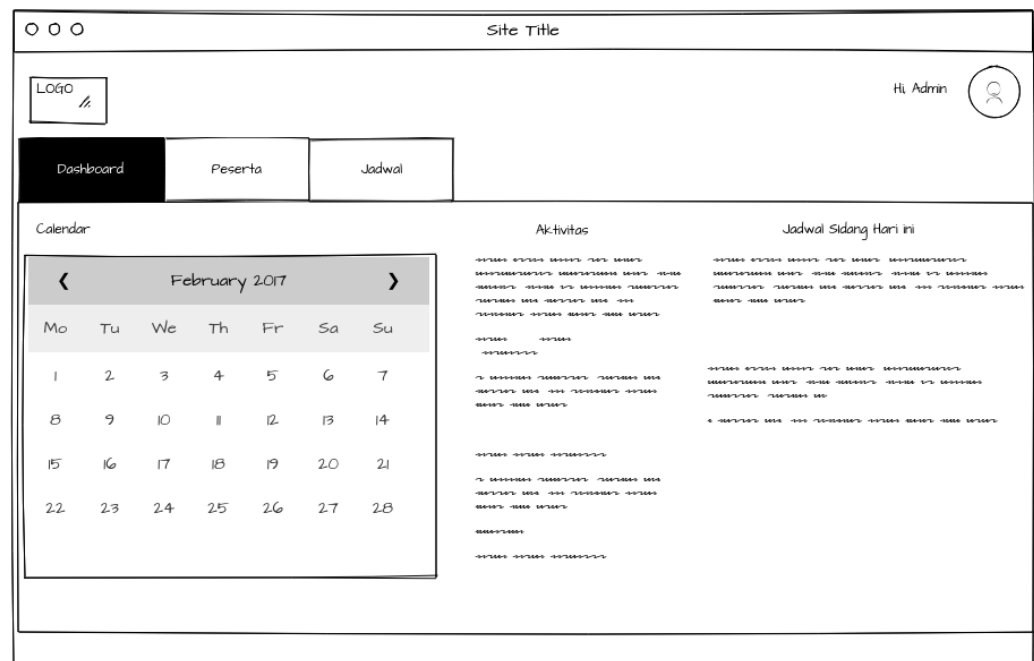


Proses yang terdapat pada Data Flow Diagram level 2 adalah proses input, edit dan delete data peserta. Proses ini hanya dapat dilakukan oleh operator. Selanjutnya data yang dikelola operator tersebut dapat dilihat oleh penonton sidang sebagai data jadwal sementara pada hari dimana sidang berlangsung.

#### 4.4.4 Desain User Interface

Berikut adalah desain User Interface yang saya buat.

##### 1. Desain Dashboard



Gambar diatas merupakan tampilan utama ketika aplikasi dijalankan oleh admin. Tersedia kalender untuk memonitoring jalanya sidang. Disebelahnya merupakan aktivitas yang nantinya akan selalu realtime menampilkan informasi sidang. Kemudian dibagian paling kanan akan menampilkan jadwal sidang yang terjadi pada hari tersebut.

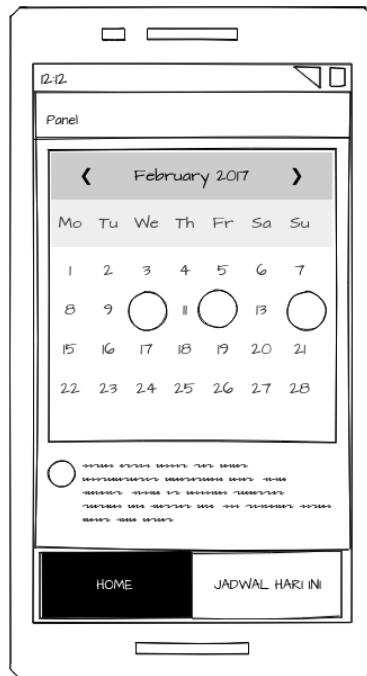
##### 2. Desain pengelola peserta

Pada tab peserta, berisi 2 komponen utama yaitu tabel untuk menampilkan peserta sidang dan input teks yang digunakan untuk memasukkan data dari peserta yang akan mengikuti sidang. Pada input bagian atas, nantinya akan bisa mencari NIM yang apabila dipilih, data akan langsung masuk pada input secara otomatis.

### 3. Desain pengelola jadwal sidang

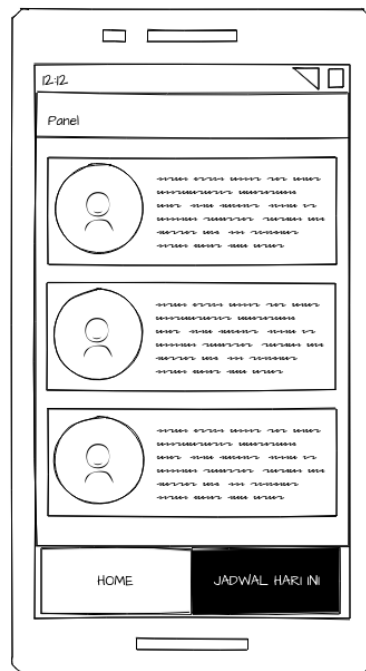
Jadwal sidang komponen utamanya kurang lebih sama dengan pada tab peserta, yang membedakannya adalah data dari tab ini berkaitan tentang jadwal.

#### 4. Desain untuk user layer utama



Pengguna android akan disuguhkan tampilan utama berupa kalender yang secara otomatis menandai hari-hari yang digunakan untuk sidang PKL

#### 5. Desain untuk user layer ke dua



Pada tab ini, berdisi data-data dari peserta sidang yang diadakan pada hari tersebut. Ketika ada kendala, secara otomatis akan ada pemberitahuan bahwa peserta tersebut tidak melaksanakan sidang.

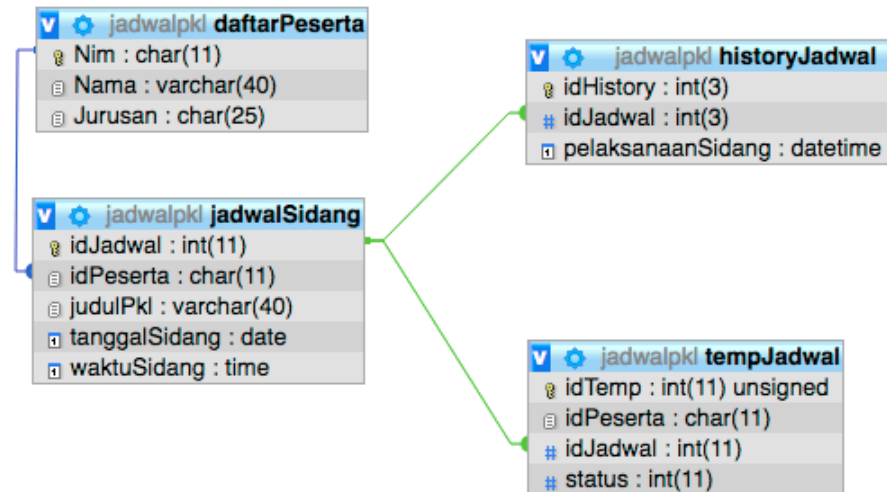
#### 4.4.5 Entity Relationship Diagram

Entity Relationship Diagram (ERD) merupakan model untuk menjelaskan hubungan antara data dalam database berdasarkan objek-objek yang mempunyai hubungan antar relasi pada sistem yang dibangun. Terdapat 4 entitas pada ERD ini, diantaranya tabel daftarPeserta, jadwalSidang, historyJadwal dan tempJadwal.

Tabel daftarPeserta berfungsi untuk menyimpan data peserta yang mengikuti sidang PKL. Data ini akan diinputkan oleh operator pertama kali kedalam sistem. Selain tabel daftarPeserta, terdapat tabel jadwalSidang yang berfungsi untuk menyimpan jadwalSidang secara keseluruhan. jadwalSidang ini nanti akan diambil oleh tempJadwal yang merupakan jadwal sementara yang dibuat untuk menyimpan jadwal setiap harinya. Kemudian untuk melihat daftar peserta yang sudah melaksanakan sidang, dari data jadwal sementara akan dikirim ke tabel historyJadwal setiap



harinya.



#### 4.5 Implementasi Perancangan

Implementasi perancangan merupakan tahap penulisan kode program dari desain sistem yang telah di buat. Penulisan kode program yang dibuat menggunakan bahasa pemrograman JavaScript dan Java. Bahasa pemrograman Java untuk pengaplikasian sistem berbasis mobile yang ditujukan untuk penonton sidang agar dapat diakses pada perangkat *smartphone*, sedangkan bahasa pemrograman JavaScript untuk proses pengaplikasian web supaya dapat diakses secara online oleh operator. Database yang digunakan menggunakan Firebase dengan tool yang digunakan adalah WebStrom.

## **BAB V**

### **KESIMPULAN DAN SARAN**

#### **5.1 Kesimpulan**

Berdasarkan analisis yang telah dilakukan oleh peneliti, dapat diambil kesimpulan sebagai berikut:

1. Peneliti masih melakukan pengamatan, belum melakukan pengujian pada sistem, jadi dapat dikatakan sistem belum dilakukan uji coba lapangan.
2. Proses pembuatan aplikasi masih belum sepenuhnya selesai, masih perlu perbaikan fitur dan bug-bug yang ada.

#### **5.2 Saran**

Saran dan masukan untuk pengembangan lebih lanjut dalam penelitian selanjutnya adalah:

1. Perlu adanya fitur baru pada dashboard admin
2. Penyegaran desain User Interface

## Daftar Isi

- Lutfi, F. (2017, January 19). *https://www.codepolitan.com/mengenal-nodejs-5880234fe9ae3*. Retrieved from Code Politan: [www.codepolitan.com](http://www.codepolitan.com)
- Muhano, G. (2016, October 31). *Pengertian API*. Retrieved from Erabelajar.com: <http://developer.erabelajar.com/api-application-programming-interface/>
- Saputra, E. Y. (2015, July 30). *Pengertian REST*. Retrieved from Ekajogja.com: <https://ekajogja.com/arsip/definisi/rest-representational-state-transfer/>