

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terkait

Berdasarkan penelitian yang dilakukan oleh Mursityo, dkk (2020) mengenai Pengembangan Sistem Informasi Penjadwalan Konsultasi Dosen Fakultas Ilmu Komputer Universitas Brawijaya, Mursityo, dkk menyimpulkan bahwa dengan adanya sistem penjadwalan konsultasi dosen maka dapat mempermudah proses konsultasi tanpa harus bertemu secara langsung. Aplikasi yang dibuat Mursityo menggunakan data dosen dan jadwal konsultasi sehingga dapat menyusun jadwal konsultasi dengan teratur.

Berdasarkan penelitian yang dilakukan oleh Puspitarini dan Hidayat (2020) mengenai Sistem Informasi Pendataan Dan Penjadwalan Imunisasi Bayi Pada Klinik Harapan Kita Menggunakan PHP, Puspitarini dan Hidayat menyimpulkan bahwa dengan adanya Sistem Informasi Penjadwalan Imunisasi Bayi ini dapat mempermudah pegawai dalam melakukan pendataan bayi dan penjadwalan imunisasi bayi. Aplikasi yang dibuat Puspitarini dan Hidayat menggunakan data bayi dan jadwal imunisasi sehingga dapat menyusun jadwal imunisasi dengan teratur .

Berdasarkan penelitian yang dilakukan oleh Rahman dan Hidayat (2020) mengenai Sistem Informasi Penjadwalan Dan Pencatatan Kunjungan Serta Perkiraan Ibu Hamil Melahirkan Pada Bidan Praktik Mandiri (BPM), Hidayat

menyimpulkan bahwa dengan adanya aplikasi ini dapat mempermudah Bidan Praktik Mandiri (BPM) dalam memberikan pelayanan terhadap pasien, terutama dalam pengolahan data dan pencarian data pasien serta memudahkan dalam menentukan perkiraan kelahiran pada ibu hamil maupun tanggal kunjungan pasien selanjutnya. Rahman dan Hidayat menggunakan data pasien ANC dan data pasien KB sehingga dapat menyusun penjadwalan dan kunjungan ibu hamil.

Berdasarkan penelitian yang dilakukan oleh Rewita, dkk (2018) mengenai Aplikasi Pendaftaran Berobat *Online* Dan Penjadwalan Dokter Berbasis *Web* (Studi Kasus: Klinik Dan Apotik Bona Mitra Keluarga), Rewita, dkk menyimpulkan bahwa aplikasi ini dapat melakukan pendaftaran berobat secara online tanpa harus datang ke klinik terlebih dahulu. Rewita, dkk menggunakan data dokter, perjanjian, poli dan pendaftaran sehingga dapat menyusun jadwal berobat dengan baik.

Berdasarkan penelitian yang dilakukan oleh Yuliana, dkk (2019) mengenai Pengembangan Sistem Aplikasi Penjadwalan Pendampingan Mahasiswa Disabilitas (Studi Kasus: Pusat Studi dan Layanan Disabilitas Universitas Brawijaya), Yuliana, dkk menyimpulkan bahwa dengan adanya sistem penjadwalan pendamping mahasiswa disabilitas maka dapat mempermudah proses penjadwalan tanpa harus bertemu secara langsung dan dapat menyesuaikan antara jadwal dari *volunteer* dan mahasiswa disabilitas agar dapat melakukan pendampingan terhadap mahasiswa disabilitas. Yuliana, dkk menggunakan data mahasiswa disabilitas, *Volunteer* dan jadwal sehingga dapat menyusun jadwal dengan baik.

II.2. Landasan Teori

Berikut ini adalah landasan teori yang berkaitan dengan penelitian yang di ambil dari beberapa jurnal terkait penelitian:

II.2.1. Perancangan

Perancangan merupakan dasar utama dalam membuat aplikasi, dengan tujuan memberikan gambaran lengkap dengan jelas kepada programmer tentang aplikasi yang akan dibuat. perancangan atau rancang merupakan serangkaian prosedur untuk menerjemahkan hasil analisa dan sebuah sistem ke dalam bahasa pemrograman untuk mendeskripsikan dengan detail bagaimana komponen-komponen sistem diimplementasikan. (Rivanthio, 2020: 188).

Perancangan merupakan sebuah alur atau proses aplikasi macam-macam teknik prinsip dengan tujuan pendefinisian suatu sistem, perangkat, atau proses detail yang mencukupi untuk memungkinkan pengaktualan fisiknya. (Saputri, 2020: 42).

II.2.2. Aplikasi

Aplikasi merupakan program yang secara langsung dapat melakukan proses-proses yang digunakan dalam komputer oleh pengguna. Aplikasi merupakan kumpulan dari file-file tertentu yang berisi kode program yang menghubungkan antara pengguna dan perangkat keras Komputer. (Sihombing dan Yanris, 2020: 13).

Aplikasi adalah program siap pakai yang direka untuk melaksanakan suatu fungsi bagi pengguna atau aplikasi yang lain dan dapat digunakan oleh sasaran yang

dituju. Berikut beberapa pandangan mengenai aplikasi antara lain Aplikasi merupakan penerapan, menyimpan sesuatu hal, data ,permasalahan, pekerjaan ke dalam suatu sarana atau media yang dapat digunakan untuk diterapkan menjadi sebuah bentuk yang baru. (Ramadhan, dkk, 2021: 26).

II.2.3. Penjadwalan

Pengertian jadwal menurut Kamus Besar Bahasa Indonesia adalah pembagian waktu berdasarkan rencana pengaturan urutan kerja, daftar atau tabel kegiatan atau rencana kegiatan dengan pembagian waktu pelaksanaan yang terperinci. Pengertian menjadwalkan adalah menetapkan jadwal. Sedangkan pengertian dari penjadwalan adalah proses, cara, pembuatan menjadwalkan atau memasukkan dalam jadwal. Jadwal biasa disajikan dalam bentuk tabel. Tabel akan terdiri dari jangka waktu tertentu dan pada setiap jangka waktu akan memiliki jadwal tertentu seperti nama mata kuliah, jam dan ruangan. Semua hal tersebut adalah komponen dari suatu jadwal. (Yuliana, dkk, 2019: 10028).

II.2.4. Konsultasi

Konsultasi tidak ada rumusan resmi tentang pengertian konsultasi, namun konsultasi dapat diartikan sebagai sesuatu hal yang bersifat personal antara satu pihak (klien) dengan pihak lain yang lazim. Selain itu juga Konsultasi adalah aktivitas konsultasi atau perundingan seperti klien dengan penasehat hukumnya. Berdasarkan definisi diatas penulis menyimpulkan konsultasi adalah suatu hal sifat personal antara satu sama yang lain sebagai pertimbangan orang terhadap masalah.

(Putra dan Azhari, 2020: 3).

Pengertian konsultasi menurut Dougherty dalam Sukendro (2007: 51) adalah suatu bentuk hubungan tolong menolong yang dilakukan oleh seorang professional yang disebut konsultan. Selain itu konsultasi diartikan sebagai pertimbangan orang terhadap suatu masalah. (Abdulghani dan Gozali, 2019: 44).

II.2.5. Pengobatan Gigi

Masyarakat mengenal dua jenis pengobatan untuk memperoleh kesehatan yang optimal yaitu, pengobatan modern (medis) dan pengobatan alternatif atau tradisional. Pengobatan medis merupakan jenis pengobatan yang menggunakan alat, cara, dan bahan yang bersifat modern dan berbahan kimia yang termasuk dalam standar pengobatan kedokteran modern. Sedangkan pengobatan alternatif merupakan suatu upaya kesehatan yang berakar pada tradisi dan menggunakan bahan alami yang sistem pengobatannya berbeda jauh dengan sistem pengobatan dalam bidang ilmu kedokteran. Sehingga disimpulkan bahwa pengobatan gigi merupakan jenis pengobatan yang menggunakan alat, cara, dan bahan yang berkaitan dengan gigi. (Nasution, dkk, 2020:).

II.2.6. Web

Website disebut sebagai fasilitas internet, dimana mengkaitkan dokumen dilingkup *local* maupun jarak jauh. Dokumen tersebut dengan *web page* dan *link website* memungkinkan pengguna bisa berpindah *page (hyper text)*, baik diantara *page* yang disimpan *server* yang sama maupun *server* diseluruh dunia. *Browser*

yang digunakan untuk mengakses dan membaca *pages* diantaranya *netscape navigator*, *google chrome*, *internet explorer*, *mozilla firefox*, dan lain sebagainya. (Abdulghani dan Gozali, 2019: 44).

Web adalah suatu sistem yang berkaitan dengan dokumen digunakan sebagai media untuk menampilkan teks, gambar, multimedia, dan lainnya pada jaringan internet. *Web* merupakan salah satu layanan yang didapat oleh pemakai komputer yang terhubung dengan fasilitas *hypertext* untuk menampilkan data berupa teks, gambar, suara, animasi dan multimedia lainnya. (Muhaimin, dkk, 2020: 72).

II.2.7. *Android*

Smartphone atau yang sering disebut dengan *android* adalah sebuah sistem operasi untuk *smartphone* dan *tablet*. Sistem operasi dapat diilustrasikan sebagai jembatan antara peranti (*device*) dan penggunaannya, sehingga pengguna dapat berinteraksi dengan *device*-nya dan menjalankan aplikasi-aplikasi yang tersedia pada *device*. *Android* merupakan generasi baru *platform mobile* berbasis *linux* yang mencakup sistem operasi, *middleware*, dan aplikasi. (Kuswanto, 2020: 80).

II.2.8. *Hypertext Preprocessor (PHP)*

Hypertext Preprocessor (PHP) adalah bahasa *server side scripting* yang menyatu dengan HTML untuk membuat halaman *web* yang dinamis. Karena PHP merupakan *server-side scripting* maka sintaks dan perintah-perintah PHP akan dieksekusi di *server* kemudian hasilnya dikirimkan ke *browser* dalam format

HTML. Dengan demikian kode program yang ditulis dalam PHP tidak akan terlihat oleh *user* sehingga keamanan halaman *web* yang dinamis, yaitu halaman *web* yang dapat membentuk suatu tampilan berdasarkan permintaan terkini, seperti menampilkan isi basis data ke halaman *web*. (Arfianto dan Nugrahanti, 2018: 175).

II.2.9. MySQL

Salah satu contoh *database management system* adalah MySQL. MySQL adalah *database open source* terpopuler di dunia. Dengan kinerja, kehandalan dan kemudahan penggunaan yang terbukti, MySQL telah menjadi pilihan *database* terdepan untuk aplikasi berbasis *web*, yang digunakan oleh properti *web* profil tinggi termasuk Facebook, Twitter, YouTube, Yahoo! dan banyak lagi. Kata “SQL” dari “MySQL” adalah singkatan dari “*Structured Query Language*”. SQL adalah bahasa standar yang paling umum digunakan untuk mengakses *database*. Bergantung pada lingkungan pemrograman yang kita pakai, kita bisa memasukkan SQL secara langsung (misalnya, untuk *men-generate* laporan), memasukkan pernyataan SQL ke dalam kode yang ditulis dalam bahasa lain, atau menggunakan API khusus yang dapat menyembunyikan sintaks SQL. (Arfianto dan Nugrahanti, 2018: 175).

II.2.10. Unified Modelling Language (UML)

Unified Modelling Language (UML) yaitu satu kumpulan konvensi permodelan yang digunakan untuk menentukan atau menggambarkan sebuah sistem perangkat lunak yang terkait dengan objek. UML merupakan suatu

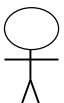
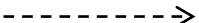
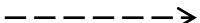
kumpulan teknik terbaik yang telah terbukti sukses dalam memodelkan system yang besar dan kompleks. UML tidak hanya digunakan dalam proses pemodelan perangkat lunak, namun hampir dalam semua bidang yang membutuhkan pemodelan. (Andikos, 2019: 39).

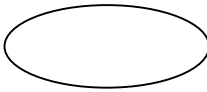
Alat bantu yang digunakan dalam perancangan berorientasi objek berbasis UML adalah sebagai berikut:

1. *Use Case Diagram*

Use case diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”. Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dengan sistem. *Use case diagram* dapat digambarkan dengan sumber-sumber pada Tabel II.1.

Tabel II.1. Simbol *Use Case*

Gambar	Nama	Keterangan
	<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
	<i>Depedency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (<i>independent</i>).
	<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
	<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara eksplisit.
	<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
	<i>Association</i>	Apa yang mnghubungkan antara

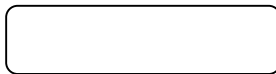
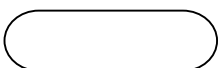
		objek satu dengan objek lainnya.
	<i>System</i>	Menspesifikan paket yang menampilkan sistem secara terbatas.
	<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu <i>actor</i> .
	<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
	<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

(Sumber: Andikos, 2019: 39)

2. Diagram Aktivitas (*Activity Diagram*)

Activity diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. *Activity diagram* dapat digambarkan dengan simbol-simbol seperti pada tabel II.2.

Tabel II.2. Simbol *Activity Diagram*

Gambar	Nama	Keterangan
	<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain.
	<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi.

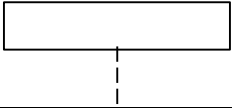
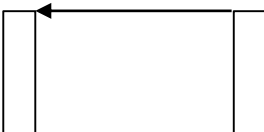
	<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
	<i>Activity Final</i>	Bagaimana objek dibentuk dan dihancurkan
	<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

(Sumber: Andikos, 2019: 39)

3. Diagram Urutan (*Sequence Diagram*)

Sequence diagram menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, *display*, dan sebagainya) berupa *message* yang digambarkan terhadap waktu. *Sequence Diagram* dapat digambarkan dengan simbol-simbol seperti pada Tabel II.3.

Tabel II.3. Simbol *Sequence Diagram*

Gambar	Nama	Keterangan
	<i>Lifeline</i>	Objek entity, antarmuka yang saling berinteraksi.
	<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi.
	<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi.

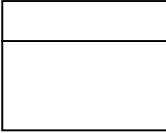
(Sumber: Andikos, 2019: 39)

4. *Class Diagram* (Diagram Kelas)

Class Diagram adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain

berorientasi objek. *Class diagram* menggambarkan struktur dan deskripsi *class*, *package* dan objek beserta hubungan satu sama lain seperti containment, pewarisan, asosiasi, dan lain-lain. *Class diagram* dapat digambarkan dengan simbol-simbol seperti pada Tabel II.4.

Tabel II.4. Class Diagram

Gambar	Nama	Keterangan
	<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
	<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
	<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
	<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.
	<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
	<i>Depedency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempegaruhi elemen yang bergantung padanya elemen yang tidak mandiri.
	<i>Assocation</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya

(Sumber: Andikos, 2019: 39)