

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terkait

Penelitian yang dilakukan oleh Yusuf Wahyu Setiya Putra, Muhammad Fadlil Adhim 2022, berjudul “SISTEM INFORMASI PRESENSI ONLINE MENGGUNAKAN TEKNOLOGI FACE RECOGNITION DAN GPS”, Pada penelitian ini membahas Sistem presensi di SMK Muhammadiyah 1 Weleri yang masih menggunakan cara konvensional. Oleh karena itu di buatlah sebuah Perancangan sistem informasi presensi online yang menggunakan GPS dan Face Recognition, dilengkapi dengan web dan Android[3].

Penelitian yang dilakukan oleh Juzack Kristiadhy, Adriyanto Juliastomo Gundo 2022, berjudul Perancangan Aplikasi Presensi Siswa Berbasis Website di SMK Negeri 1 Tenganan Menggunakan Webcam dan GPS Guna Mengurangi Risiko Penularan Virus COVID-19. bertujuan merancang aplikasi absensi untuk mengurangi risiko penularan COVID-19 dengan memperbarui alur sistem presensi. Perancangan aplikasi menggunakan metode perancangan prototype dan pengujian kelayakan menggunakan standar ISO 9126. Aplikasi ini ditujukan untuk digunakan di SMK Negeri 1 Tenganan. Partisipan penelitian meliputi siswa, wali kelas, guru bimbingan konseling, dan software engineer. Hasil uji kelayakan menunjukkan bahwa aplikasi ini sangat layak digunakan sesuai standar ISO 9126[4].

Penelitian yang dilakukan oleh Sofyan Nur Salim, L. M. Rasdi Rere, dan Edi Pranoto yang berjudul “ Aplikasi E- Absensi Dengan Geolokasi Berbasis Web Pada Perusahaan Griya Furniture” penelitian ini membahas tentang absensi yang terdapat pada Perusahaan Griya Furniture, di karenakan absensi itu sangat penting karena tidak hanya untuk kedisiplinan juga untuk mengatur ritme kerja. Saat ini mesin absensi fingerprint yang dipakai perusahaan Griya Furniture, masih menggunakan manual dalam pendataannya oleh karena itu penelitian ini membuat sebuah aplikasi E-absensi. Hasil pengujian Black box testing memperlihatkan bahwa semua fungsi berjalan sesuai dengan yang telah ditentukan[5].

Penelitian yang dilakukan Ahmad Yani dan Susy Rosyida yang berjudul “Penerapan Sistem Informasi Absensi Karyawan Pada CV. Bintang Bangun Persada Bekasi”. Penelitian ini membahas tentang system absensi karyawan yang ada di CV Bintang Bangun Persada Bekasi masih menggunakan manual begitu juga proses pencatannya. Karena dilakukan secara manual kemungkinan kesalahannya besar, oleh karena itu di buatnya sebuah aplikasi . Dengan menggunakan aplikasi ini, sistem informasi absensi akan lebih efektif dan efisien[6].

Penelitian yang dilakukan M. Reza Prayogge , Megawati yang berjudul “RANCANG BANGUN SISTEM INFORMASI ABSENSI KARYAWAN BERBASIS WEB “ Penelitian ini membahas tentang membuat presensi dengan website karena masih menggunakan manual kemungkinan data hilang akan membuat masalah pada masa mendatang,

atau bisa juga di salah gunakan. Oleh karena itu di buatnya sebuah sistem absensi karyawan berbasis web yang semuanya sudah tersimpan dengan rapi datanya dan terupdate di lengkapi fitur pencarian. Dengan adanya sistem ini absensi bisa tertata dan tidak mudah hilang[7].

2.2 Landasan Teori

2.2.1 Sistem Informasi

Sistem merupakan suatu kumpulan komponen - komponen yang membentuk suatu kesatuan. Sebuah organisasi dan sistem informasi adalah sistem fisik dan sosial yang ditata sedemikian rupa untuk mencapai tujuan tertentu. namun sistem informasi bisa diartikan lain seperti data yang saling berhubungan dengan metode yang di dalamnya menggunakan hardware atau software dalam penyampaian sebuah informasi yang bermanfaat[8]

2.2.2 Pengertian Presensi

Presensi merupakan Data kehadiran , pelaporan dari aktivitas atau data data kehadiran yang di susun dan diatur sedemikian rupa agar lebih mudah dalam pencarian dan saat akan dipergunakan apabila sewaktu-waktu diperlukan oleh pihak yang berkepentingan[9].

2.2.3 Web

Internet atau dapat disingkat web adalah suatu layanan sajian informasi yang menggunakan konsep hyperlink (tautan), yang memudahkan surfer (sebutan para pemakai komputer yang melakukan

perusing atau bahasa yang telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak. UML mendefinisikan graph chart seperti : use case outline, class graph, statechart graph, movement outline, grouping chart, joint effort chart, part diagram dan arrangement graph[10].

2.2.4 Hypertext Markup Language (HTML)

Bahasa pemrograman dasar untuk web programming bersifat client side yang memungkinkan untuk menampilkan informasi dalam bentuk teks, grafik, serta interactive media dan juga untuk menghubungkan antar tampilan site page (hyperlink)[11].

2.2.5 HyperText Preprocessor (PHP)

Merupakan bahasa pemrograman yang di compositions di server, Fungsi utama PHP dalam membangun site adalah untuk melakukan pengelolaan information dalam data set. "PHP (*HyperText PreProcessor*) merupakan suatu bahasa pemrograman yang digunakan untuk menerjemahkan baris kode program menjadi kode mesin yang dapat di mengerti oleh komputer yang bersifat serverside yang dapat di tambahkan ke dalam HTML[12] .

2.2.6 JavaScript

JavaScript merupakan bahasa pemrograman tingkat dinamis yang tinggi dan serta merupakan teknologi inti Internet selain HTML

dan CSS. *JavaScript* sangat membantu dalam pembuatan halaman web interaktif dan juga merupakan bagian aplikasi web yang esensial. *JavaScript* itu sendiri dikembangkan oleh Brendan Eich yang berasal dari Netscape dengan nama Mocha, yang kemudian dikembangkan kembali menjadi *JavaScript*[13].

2.2.7 CodeIgniter

CodeIgniter merupakan sebuah *structure php* yang sifatnya *open source* dan menggunakan metode MVC (Model, View, Regulator) yang bertujuan untuk membantu mempermudah *designer* atau *software engineer* dalam membangun sebuah aplikasi framework yang berbasis web tanpa harus membuatnya dari nol". *Structure Codeigniter* itu sendiri dikembangkan oleh Rick Ellis, sebagai Chief Ellislab, Inc[dick[14].

2.2.8 MySql

Sebuah implementasi dari sistem manajemen basisdata relasional (RDBMS) yang didistribusikan secara free dibawah lisensi GPL (Overall population Permit). Setiap pengguna dapat secara bebas menggunakan MySQL, namun dengan batasan perangkat lunak tersebut tidak boleh dijadikan produk turunan yang bersifat komersial. MySQL sebenarnya merupakan turunan salah satu konsep utama dalam basisdata yang telah ada sebelumnya; SQL (Organized Inquiry Language). SQL adalah sebuah konsep pengoperasian basisdata,

terutama untuk pemilihan atau seleksi dan pemasukan information, yang memungkinkan pengoperasian information dikerjakan dengan mudah secara otomatis[15].

2.2.9 Visual Studio Code

Merupakan aplikasi korektor kode sumber terbuka yang dibuat oleh Microsoft untuk kerangka kerja Windows, Linux, dan MacOS. Visual Studio Code memudahkan pembuatan kode yang menampung beberapa jenis pemrograman, seperti *C++*, *C#*, *Java*, *Python*, *PHP*, *GO*. Kode Visual dapat mengenali jenis bahasa pemrograman yang terlibat dan menyediakan berbagai variasi sesuai kemampuan suksesi kode. Visual Studio Code juga telah dikoordinasikan ke dalam Github. Selain itu, satu komponen lagi adalah kemampuan untuk menambahkan augmentasi dimana desainer dapat menambahkan perluasan untuk menambahkan penyertaan yang tidak ada dalam Visual Studio Code[16].

2.2.10 Xampp

Software yang terdapat server *MySQL* dan digunakan untuk membantu mengelola data. Di dukung oleh bahasa pemrograman. Dapat juga digunakan untuk membuat web dinamis karena di dalamnya terdapat server apache yang dapat dijalankan di beberapa platform seperti OS X, Windows, Linux, Mac, dan Solaris[17].

2.2.11 UML



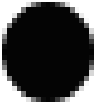


Unified Modeling Language merupakan salah satu metode pemodelan visual yang digunakan dalam perancangan dan pembuatan sebuah software yang berorientasikan pada objek. UML merupakan sebuah standar penulisan atau semacam blue print dimana didalamnya termasuk sebuah bisnis proses, penulisan kelas-kelas dalam sebuah bahasa yang spesifik. Terdapat beberapa diagram UML yang sering digunakan dalam pengembangan sebuah sistem[18], yaitu.

1) Activity Diagram

Activity Diagram merupakan *diagram* yang menjelaskan alur kerja aktivitas pengguna atau sistem yang berbeda, individu yang melakukan tindakan dan urutan terjadinya aktivitas.

Penjelasan lengkapnya dapat di lihat pada Tabel 2.2.10.1 Activity Diagram dibawah ini.

Tabel 2.2.11 1 *Activity Diagram*




NO	GAMBAR	NAMA	KETERANGAN
1.		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2.		<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3.		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
4.		<i>Activity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan
5.		<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran






2) Usecase Diagram

Diagram *use case* adalah gambar yang menggunakan simbol-simbol tertentu untuk menjelaskan bagaimana pengguna menggunakan suatu sistem atau program komputer, sehingga alurnya menjadi jelas.

Penjelasan lengkapnya dapat di lihat pada Tabel 2.2.10.2 *Usecase Diagram* dibawah ini.

Tabel 2.2.11 2 *Usecase Diagram*






NO	GAMBAR	NAMA	KETERANGAN
1.		<i>Actor</i>	Menspesifikasikan himpunan peran yang dimainkan ketika berinteraksi dengan <i>use case</i> .
2.		<i>Dependency</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>).
3.		<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>).
4.		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara eksplisit.
5.		<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu



NO	GAMBAR	NAMA	KETERANGAN
			titik yang diberikan.
6.		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7.		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
8.		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
9.		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
10.		Note	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

3) *Class Diagram*

Class diagram merupakan gambaran diagram yang menjelaskan hubungan antar *class* dalam sebuah sistem yang sedang dibuat dan menjelaskan bagaimana caranya agar mereka saling berkolaborasi. Penjelasan lengkapnya dapat di lihat pada Tabel 2.2.10.3 *Class diagram* dibawah ini.

Tabel 2.2.11 3 *Class Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1.		<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>).
2.		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3.		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
4.		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
5.		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.


NO	GAMBAR	NAMA	KETERANGAN
6.		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan memengaruhi elemen yang bergantung padanya elemen yang tidak mandiri
7.		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya

4) Sequence Diagram

Sequence Diagram merupakan gabungan dari diagram *class* dan diagram *object* yang memiliki suatu gambaran model statis, namun juga bersifat dinamis, seperti diagram *interaction*. *Sequence* Diagram menggambarkan kolaborasi dinamis antara sejumlah objek untuk menunjukkan rangkaian pesan yang dikirim antara objek juga interaksi antar objek, sesuatu yang terjadi pada titik tertentu dalam eksekusi system.

Penjelasan lengkapnya dapat di lihat pada Tabel 7.4 *Sequence* Diagram dibawah ini.

Tabel 2.2.11 4 *Sequence* Diagram

NO	GAMBAR	NAMA	KETERANGAN
1.		<i>LifeLine</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.

NO	GAMBAR	NAMA	KETERANGAN
2.		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi
3.		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi

2.2.12 BOOTSTRAP

Paket aplikasi yang siap pakai untuk membuat halaman front end maupun back end dari sebuah situs web. Dapat dikatakan, Bootstrap adalah template desain web dengan fitur plus (framework css). Bootstrap diciptakan untuk mempermudah proses desain web bagi berbagai tingkat pengguna, mulai dari level pemula hingga yang sudah berpengalaman. Cukup bermodalkan pengetahuan dasar mengenai HTML dan CSS, anda pun siap menggunakan Bootstrap[19].

2.2.13 CSS (Cascading Style Sheet)

Merupakan Bahasa pemrograman untuk memberikan tampilan desain yang akan digunakan pada web seperti warna, font, outline, background, menyesuaikan tampilan website dengan ukuran layar, dsb. CSS digunakan pada pembuatan website ini adalah untuk berkolaborasi dengan HTML agar dapat menghasilkan tampilan

website yang menarik[20].

2.2.14 White Box

White box testing salah satu metode pengujian perangkat lunak yang dilakukan dengan pemahaman penuh tentang struktur dan logika internalnya. Penguji mengidentifikasi keralahan logis dan mengoptimalkan kode dengan memeriksa semua bagian yang dapat diuji. Pengujian ini dapat dimulai bahkan jika antarmuka grafis (GUI) masih dalam pengembangan. Metode ini efektif karena memungkinkan pengujian yang komperhensif dari seluruh sruktur perangkat lunak dan sering disebut sebagai *clearbox*, *glass box*, atau *open box* testing[21].

2.2.15 Black Box

Black box merupakan metode pengujian perangkat lunak tanpa pengetahuan tentang struktur atau logika internal,yang berfokus pada spesifikasi kebutuhan dari prefektif pengguna akhir. Metode ini memiliki kelebihan dalam menentukan aspek yang tidak sesuai dengan spesifikasi, namun terbatas oleh kekurangannya pemahaman penguji terhadap perangkat lunak secara keseluruhan[22].