

BAB II

TINJAUAN PUSTAKA

2.1. Penelitian Terkait

Menurut Astrianda, N., dan Mohamad, F. S. (2017), banyak penelitian telah dilakukan tentang cara mengklasifikasikan tomat. Dalam hal ini, Komputer dapat digunakan untuk menentukan kematangan tomat. Sebelum penggunaan sistem pemrosesan gambar, identifikasi kematangan tomat sebagian besar dilakukan secara manual. Melakukan identifikasi kematangan tomat secara manual memiliki banyak kelemahan yang disebabkan oleh berbagai faktor, termasuk kelelahan, kurangnya motivasi, pengalaman, kemahiran, dan lain-lain. Untuk mengatasi masalah ini, kematangan buah dapat diidentifikasi dengan komputer dan bahkan jenisnya. Bentuk, warna, ukuran, dan tekstur adalah karakteristik yang dapat membedakan tomat masak dari belum masak. Warna adalah karakteristik yang paling mudah digunakan untuk membedakan tomat masak dari belum masak karena warna dapat dilihat secara langsung tanpa bersentuhan dengan bahan lain[4].

Salah satu buah yang sangat disukai, tomat, menurut Budiarmo et al. pada tahun 2022, memiliki banyak vitamin dan sering digunakan sebagai sayuran. Proses memilih buah tomat juga sering menghasilkan kesalahan karena proses penyortiran untuk mengukur kematangan buah tomat biasanya dilakukan secara manual. Banyak kesalahan terjadi, termasuk data yang salah, karena perbedaan persepsi kematangan dan kecepatan pengolahan yang lambat. Akibatnya, penelitian telah dilakukan tentang sistem penyortir

otomatis yang menggunakan Dengan menggunakan sel beban, sensor berat dan warna TCS3200 dapat digunakan dalam sistem sortir barang berdasarkan warna dan berat[5].

Pada tahun 2020, Thabit et al. menyatakan bahwa petani tomat masih memanen dan menyortir buah tomat dengan tangan kosong. Hal ini menyebabkan banyak kekeliruan penyortiran yang masih terjadi dalam pengklasifikasian ukuran dan tingkat kematangan. Solusi untuk alat penyortir otomatis datang dari pengembang teknologi tepat guna, Penyortir Buah Tomat Berbasis Arduino Uno. Tujuan dari penelitian ini adalah untuk membuat sistem penyortir buah tomat otomatis yang menggunakan mikrokontroler untuk memudahkan petani memilih buah tomat ukuran besar, kecil, besar, dan kecil[6].

2.2. Landasan Teori

2.2.1. Sistem Informasi

Sistem informasi telah menjadi bagian penting dari operasi bisnis di era digital yang berkembang pesat saat ini. Penyortiran buah dan sayuran adalah salah satu bidang yang membutuhkan manajemen yang baik. Manajemen yang baik memastikan stok buah yang cukup, mencegah kelebihan stok, dan mengoptimalkan penggunaan sumber daya. Penggunaan teknologi informasi yang tepat diperlukan untuk meningkatkan efisiensi dan akurasi pengelolaan. Petani dapat

memantau dan mengendalikan lebih baik stok mereka dengan sistem informasi[7].

2.2.2. Website

Sebuah web adalah kumpulan halaman web yang berisi informasi data digital, seperti teks, gambar, animasi, suara, dan video, yang disediakan melalui jalur internet, sehingga setiap orang di seluruh dunia dapat melihat dan mengaksesnya. Halaman web dibuat menggunakan bahasa standar yang disebut HTML. Web browser menerjemahkan skrip HTML ini sehingga informasi dapat dilihat oleh semua orang. Website biasanya terbagi menjadi tiga kategori: statis, dinamis, dan interaktif[8].

2.2.3. PHP

PHP *Preprocessor Hypertext* adalah bahasa pemrograman server side yang sangat populer, terutama untuk membuat website dinamis. Bahasa pemrograman PHP memang diperlukan untuk beberapa fungsi pembuatan website, seperti memproses informasi yang dikirimkan oleh pengguna website[9]. Logo PHP dapat dilihat seperti pada Gambar 2.1.



Gambar 2. 1. Logo PHP

2.2.4. Arduino IDE

Arduino telah dikenal dan dimanfaatkan serta terimplementasi dan diterima baik dikalangan pembelajar maupun profesional. Sebuah *Rotary encoder* sebetulnya adalah jenis sensor posisi yang berfungsi untuk menentukan posisi sudut poros yang berputar yang outputnya berupa jarak yang menghasilkan sinyal listrik analog maupun digital berdasarkan gerakan rotasinya. Kegiatan pengabdian ini dilakukan dengan melaksanakan pelatihan pemrograman robot dengan Arduino yang disinkronkan dengan *rotary encoder* [10]. Logo Arduino Ide dapat dilihat seperti pada Gambar 2.2.



Gambar 2. 2. Logo Arduino IDE

2.2.5. MySql.

MySQL, yang dikembangkan, didistribusikan, dan didukung oleh Oracle Corporation, adalah sistem manajemen database yang paling populer dan bersifat open source. Informasi ini dapat ditemukan di situs resmi MySQL. MySQL termasuk bahasa pemrograman yang paling dikenal karena hampir semua server basis

data yang ada mendukung fungsi manajemen datanya[11]. Logo MySQL dapat dilihat seperti pada Gambar 2.3.



Gambar 2. 3. Logo MySql

2.2.6. CSS

Cascading Style Sheet (CSS) adalah singkatan dari sebuah konsep yang memiliki struktur dan terorganisir dalam paket atau *library*. Dengan konsep ini, *desainer* dan *developer* lebih fokus pada proyek yang sedang dikerjakannya tanpa harus memulainya dari awal. *CSS Framework* itu sendiri merupakan suatu kumpulan sintak dari bahasa CSS yang telah diorganisir secara tersusun[12].

2.2.7. *CodeIgniter*.

CodeIgniter adalah framework pengembangan aplikasi yang menggunakan PHP dan mempermudah pembuatan web. Dengan *CodeIgniter*, Anda dapat mengakses librari dan fungsi yang *diperlukan* untuk menyelesaikan tugas dengan menggunakan antarmuka dan struktur logika yang sederhana[13]. Logo *CodeIgniter* dapat dilihat seperti pada Gambar 2.4.



Gambar 2. 4. Logo CodeIgniter

2.2.8. *Visual Studio Code*

Teks editor *Microsoft Visual Studio Code* mendukung *JavaScript*, *Typescript*, dan *Node* secara langsung, dan tersedia untuk Linux, Mac, dan Windows versi sistem operasi[12]. Logo *Visual Studio Code* dapat dilihat seperti pada Gambar 2.5.



Gambar 2. 5. Logo *Visual Studio Code*

2.2.9. **Xampp**

Dengan menggunakan XAMPP di localhost, kita dapat membuat sebuah proyek website mudah dan efisien. Studi kasus kali ini akan fokus pada implementasi pembuatan website Karena kemudahan instalasinya, Xampp adalah bundel web server yang populer untuk coba-coba di Windows. *Server web Apache*, *interpreter PHP*, dan basis data MySQL termasuk dalam kumpulan program open source tersebut. Kami dapat mencoba menginstal aplikasi web dan

memulai pemrograman PHP di komputer kami sendiri setelah menginstal XAMPP. Berikut adalah gambar logo dari Xampp[14]. Logo Xampp dapat dilihat seperti pada gambar 2.6.



Gambar 2. 6. Logo Xampp

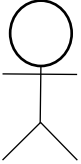
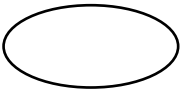
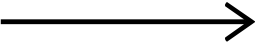


2.2.10. Unified Modelling Language

Unified Modelling Language (UML)[10] merupakan metode permodelan yang divisualisasikan dalam berbagai bentuk diagram dengan tujuan menggambarkan rancangan sistem dengan pendekatan berorientasi objek. Diagram UML yang digunakan untuk membuat model rancangan aplikasi media pembelajaran terdiri dari: diagram use case, diagram activity, diagram sequence dan diagram state chart[15].

1. Use Case Diagram

Use Case Diagram adalah model yang digunakan untuk menggambarkan perilaku sistem informasi yang akan dikembangkan. Diagram ini digunakan untuk mengidentifikasi fungsi-fungsi yang ada dalam sistem informasi serta menentukan siapa saja yang berhak menggunakannya. Simbol-simbol *Use Case* diagram dapat dilihat seperti pada Tabel 2.1.


Tabel 2.1. *Use Case*




Simbol	Nama	Keterangan
	<i>Actor</i>	Pelaku, seseorang atau apa saja yang berhubungan dengan sistem yang sedang dibangun.
	<i>Use case</i>	Menggambarkan bagaimana seseorang menggunakan sistem.
	<i>Relasi</i>	Hubungan dengan <i>use case</i> lain, maupun relasi antara aktor dan <i>use case</i> .
	<i>Batasan Sistem</i>	Untuk memperlihatkan batasan sistem dalam diagram <i>use case</i> .
	<i>Dependency</i>	Suatu hubungan semantic antara dua <i>things</i> dimana perubahan pada suatu <i>things</i> (<i>independent</i>)

2. Diagram Aktivitas (*Activity Diagram*)

Diagram Activity dapat menggambarkan alur kerja atau aktivitas dari suatu sistem atau proses bisnis. Simbol-simbol *Activity Diagram* dapat dilihat seperti pada Tabel 2.2.

Tabel 2.2. *Activity Diagram*

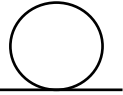
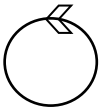
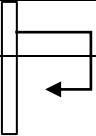
Simbol	Nama	Keterangan
	<i>Start State</i>	Menunjukkan dimulainya suatu <i>workflow</i> .




Simbol	Nama	Keterangan
	<i>End State</i>	Menggambarkan akhir dari pada sebuah <i>activity diagram</i> .
	<i>Activities</i>	Menggambarkan sebuah pekerjaan atau tugas dalam <i>workflow</i> . Siapa yang bertanggung jawab terhadap aktivitas.
	<i>Note</i>	Suatu <i>symbol</i> yang memberikan batasan dan komentar yang dikaitkan pada suatu elemen atau kumpulan elemen.

3. Diagram Urutan (*Sequence Diagram*)

Sequence Diagram menggambarkan interaksi antara objek-objek dalam urutan waktu. Diagram ini digunakan sebagai gambaran kolaborasi dinamis antara sejumlah objek. Simbol-simbol *Sequence Diagram* dapat dilihat seperti pada Tabel 2.3.

Tabel 2.3. *Sequence Diagram*

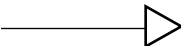

Simbol	Nama	Keterangan
	<i>Entity Class</i>	Gambaran Sistem sebagai landasan dalam menyusun basis data.
	<i>Control Class</i>	Bertanggung jawab terhadap kelas-kelas, dan objek yang berisi logika
	<i>Recursive</i>	Pesan untuk dirinya

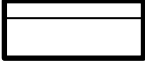
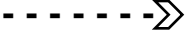


Simbol	Nama	Keterangan
		
	<i>Activation</i>	Mewakili proses durasi aktivasi sebuah operasi
	<i>Life Line</i>	Komponen yang digambarkan garis putus terhubung dengan objek

4. Class Diagram

Class Diagram merupakan penghubung antara kelas dan penjelasan rinci setiap kelas dalam model perancangan suatu sistem. Hal ini juga menunjukkan karakteristik yang melekat pada sistem dan tanggung jawab entitas. Simbol-simbol *Class Diagram* dapat dilihat seperti pada Tabel 2.4.

Tabel 2.4. Simbol *Class Diagram*

Simbol	Nama	Keterangan
	Generalization	Hubungan antara objek (<i>descendent</i>) dan struktur data objek induk (<i>ancestor</i>)
	<i>Nary Association</i>	Suatu Aksi untuk terhindar dari asosiasi yang memiliki objek lebih dari 2

Simbol	Nama	Keterangan
	<i>Class</i>	Kumpulan objek atribut dan operasi yang sama
	<i>Realization</i>	Tindakan dari suatu objek
	<i>Dependency</i>	Suatu perubahan pada elemen yang mempengaruhi elemen yang tidak mandiri.
	<i>Association</i>	hubungan objek 1 dengan yang lainnya