

BAB II

LANDASAN TEORI

2.1 Penelitian Terdahulu

Penelitian terdahulu ini menjadi satu acuan penulis dalam membuat laporan akhir sehingga dapat memperkaya teori yang digunakan dalam mengkaji penelitian yang dilakukan. Berikut merupakan penelitian terdahulu berupa beberapa jurnal yang terkait dengan judul laporan akhir penulis.

Rahmadi dan Yusmiarti (2014) prosiding yang berjudul “Perancangan Sistem Informasi Inventory di Amik Lembah Dempo Pagaram”. Penelitian ini bertujuan untuk membuat sebuah perancangan sistem informasi inventory pada AMIK Lembah Dempo Pagaram yang nantinya hasil dari perancangan ini dapat dipakai ataupun dikembangkan untuk pembangunan sistem informasi inventory di AMIK Lembah Dempo. Dengan menggunakan sistem informasi inventory dapat membantu dan mempermudah proses pengelolaan data inventory pada AMIK Lembah Dempo. Karena beban kerja hanya dilakukan satu kali tanpa perlu melakukan proses pembukuan terlebih dahulu sebelum menginputkan data. Pengambilan data dan informasi jadi lebih cepat dibandingkan dengan sistem manual yang lama. Karena baik data ataupun informasi tersimpan rapi di dalam database yang terjamin keamanannya, tidak bisa diakses oleh siapa saja dan dapat dipanggil kapan saja.

Nugrahanti (2015) prosiding yang berjudul “Perancangan Sistem Informasi Inventory Sparepart Mesin Fotocopy dengan Menggunakan Visual Delphi 7 (Studi Kasus di UD. Eka Taruna Madiun)”. Hasil penelitian ini bertujuan untuk menghasilkan suatu aplikasi program yang dapat melakukan kontrol persediaan sparepart mesin fotocopy dan memberikan laporan update stok dan juga inventory sparepart mesin fotocopy ini memudahkan karyawan bagian administrasi dalam mengolah data sehingga dapat meminimalisir semua kemungkinan dalam manipulasi data dan kesalahan pencatatan.

Tiara dan Arzia, dkk (2015) jurnal yang berjudul “Penerapan Sistem Inventory Labotarium Digital dengan Metode Critical Succes Factor Pada Perguruan Tinggi Raharja”. Hasil Penelitian ini bertujuan untuk pengelolaan data

barang yang ada menjadi lebih mudah karena penggunaan database yang terintegrasikan dengan baik dan diproses secara terkomputerisasi yang dapat memungkinkan hasil pengolahan data akan lebih cepat, tepat, akurat sehingga persediaan barang didalam labotarium digital pada Perguruan Tinggi Raharja dapat termonitoring dengan baik.

Rahmad dan Setiady (2014) jurnal yang berjudul “Perancangan Sistem Informasi Inventory Spare Part Elektronik Berbasis Web PHP (Studi CV. Human Global Service Yogyakarta)”. Hasil Penelitian ini bertujuan untuk membuat perangkat lunak sistem informasi inventory sparepart elektronik berbasis Web PHP dengan kemampuan dapat menyimpan data penjualan dan pembelian barang, mengontrol stok barang masuk dan keluar gudang, dapat mencatat penggunaan biaya inventory yang ada. Melalui sistem informasi inventory ini, pihak manajemen dapat mengambil keputusan berdasarkan rekapitulasi transaksi, sisa stok dan informasi lain.

Yulianti dan Yupianti (2012) jurnal yang berjudul “Sistem Informasi Persediaan Barang Pada PT. Surya Nusa Bhaktindo Bengkulu”. Penelitian ini bertujuan untuk membuat Aplikasi Persediaan Barang pada PT. Surya Nusa Bhaktindo Bengkulu sehingga dapat dijadikan sebagai media pendukung dalam pembuatan informasi kepada pimpinan. Sistem informasi ini akan menampilkan informasi transaksi pemasukan dan pengeluaran material berdasarkan rencana biaya yang telah ditetapkan. Semua informasi ini akan ditampilkan dalam bentuk laporan material dan penginputan data material yang dimulai dari permintaan hingga pemasukan material menjadi cepat serta dapat mengontrol material dilapangan.

Susanto (2010) skripsi yang berjudul “Pengembangan Sistem Informasi Inventory Pada PT. Dwiwarna Inti Sejahtera”. Hasil Penelitian ini bertujuan untuk merancang desain aplikasi yang menarik dan user friendly untuk memudahkan karyawan PT. Dwiwarna Inti Sejahtera dalam mengelola dan mengontrol data barang, menyajikan info barang yang akurat serta menyajikan laporan yang cepat sesuai kebutuhan perusahaan.

Menurut Grace O. Akinola dan Olusegun Timothy Odesola (2018) yang berjudul *“Information and Communications Technology and Inventory Management amongst Breweries in Nigeria”*. Studi telah menunjukkan bahwa hampir semua organisasi baik itu industri, pendidikan, kepemilikan tunggal, kesehatan, dan instansi pemerintah antara lain memiliki alasan untuk melestarikan satu jenis inventaris atau lainnya selama menjalankan operasi perusahaan. Dalam situasi kehidupan nyata, permintaan dan penawaran produk tertentu produk atau bahan mentah tidak selalu sama sehingga ada kebutuhan untuk menyimpan persediaan dalam suatu organisasi semua waktu untuk mengurangi biaya yang berhubungan dengan pengendalian persediaan.

Menurut Putu Arya Diana, Putu Adi Pratama, dan Emmy Febriani Thalib (2019) yang berjudul *“Designing Inventory Information Systems at UD. Miasa Desktop-Based”*. Sistem informasi persediaan telah banyak digunakan atau dikembangkan dalam suatu perusahaan dengan berbagai teknologi dan sistem. Masalah di UD. Miasa bukanlah ketersediaan suatu sistem informasi untuk pengendalian persediaan barang stok yang tersedia di gudang. Sebuah sistem informasi persediaan barang akan sangat membantu UD. Miasa dalam menjalankan bisnisnya.

Menurut Ata Allah Taleizadeh (2017) yang berjudul *“International Journal of Inventory Research (IJIR)”*. Edisi IJIR ini mencakup lima makalah penelitian. Temuan dan wawasan yang dihasilkan oleh Semua organisasi ingin meningkatkan jumlah konsumen untuk produk mereka produk. Untuk itu, mereka menawarkan berbagai jenis penawaran promosi, diskon, dan pembiayaan kebijakan. Selain itu harga juga merupakan salah satu faktor kunci dalam pengendalian persediaan dan pengeceran harus mempertimbangkannya ketika membuat keputusan.

Menurut Tsan-Ming Choi and Hau-Ling Chan (2017) yang berjudul *“International Journal of Inventory Research since”*. Dalam 'Dampak berbagi inventaris pada ketersediaan layanan dan tingkat transportasi di distribusi waktu-dibedakan', Jat dan Rafique mengeksplorasi sistem layanan dengan dua pilihan jendela waktu pengiriman. Penulis membandingkan melalui studi simulasi

inventaris skenario berbagi dan tidak berbagi untuk fasilitas layanan. Para penulis memeriksa komposisi permintaan yang bervariasi dan temukan bahwa ada kasus di mana pembagian inventaris tidak selalu mengikuti 'pertukaran inventaris-transportasi' klasik.

Menurut Manfredi Bruccoleri, Salvatore Cannella, dan Giulia La Porta (2014) yang berjudul “*Inventory record inaccuracy in supply chains: the role of workers' behavior*” Dalam makalah ini kami mengusulkan model analitik dari rantai pasokan eselon tunggal dalam inventory di mana pekerja yang terlibat dalam aktivitas perekaman item aliran masuk membuat data kesalahan entri inventory. Kami mengukur bagaimana kesalahan tersebut mempengaruhi efek cambuk. Model analitis yang diusulkan dianalisis secara numerik melalui simulasi, parameter diatur sesuai dengan desain eksperimen faktorial penuh (DOE) dan hasil simulasi dianalisis dengan metode statistik.

Beberapa penelitian mengenai sistem informasi inventory yang telah dilaksanakan oleh peneliti sebelumnya. Maka perbedaan yang dimiliki yang diusulkan penulis yaitu aplikasi inventaris ruang belajar berbasis web pada SMK Negeri 1 Kayuagung yang mana aplikasi inventory ini akan memberikan informasi tentang nama barang, jumlah barang, keadaan barang dan beberapa informasi yang terkait dengan barang perlengkapan, serta pembuatan laporan, yaitu laporan tiap semester, laporan daftar inventaris dan laporan barang rusak yang bisa dilihat informasinya dalam bentuk file yang bisa di print.

2.2 Data

berbagai definisi tentang data yang dikutip dari berbagai sumber yang dapat dilihat pada Tabel 2.1 yang memberikan penjelasan lebih lanjut tentang data.

Tabel 2.1 Definisi Data

Sumber	Definisi
(Sutabri, 2012:2)	Data adalah bahan mentah yang diproses untuk menyajikan informasi.

(Sutanta, 2011:3)	Data dapat didefinisikan sebagai bahan keterangan tentang kejadian-kejadian atau fakta-fakta yang dikelompokkan dalam lambang tertentu yang tidak acak, yang menunjukkan jumlah, tindakan, atau hal.
(Fatta, 2007:9)	Data merupakan nilai, keadaan, atau sifat yang berdiri sendirilepas dari konteks apapun.
(Jogiyanto, 2005:8)	Data adalah kenyataan yang menggambarkan keadaan suatu kejadian-kejadian atau keasatuan nyata. Data merupakan bentuk yang masih mentah yang belum dapat bercerita lebih lanjut, sehingga perlu diolah terlebih dahulu.

Dari berbagai uraian pada Tabel 2.1 dapat disimpulkan bahwa data adalah bahan mentah seperti kejadian atau fakta, nilai, keadaan atau sifat yang belum mempunyai makna atau belum berarti bagi penerima, sehingga perlu diolah atau diproses untuk menyajikan suatu informasi.

2.3 Sistem

Berbagai definisi tentang sistem yang dikutip dari beberapa sumber yang dapat dilihat pada Tabel 2.2 untuk memberikan penjelasan lebih lanjut tentang sistem.

Tabel 2.2 Definisi Sistem

Sumber	Definisi
(Fathansyah, 2012:11)	Sistem adalah sebuah tatanan (keterpaduan) yang terdiri atas jumlah komponen fungsional (dengan satuan fungsi dan tugas khusus) yang saling berhubungan dan bersama sama untuk memenuhi suatu proses tertentu.

(Sutabri, 2012:6)	Sistem adalah sekelompok yang erat hubungannya satu dengan yang lain, yang berfungsi bersama-sama untuk mencapai tujuan tertentu.
(Nugroho, 2010:17)	Sistem dapat didefinisikan sebagai sekelompok elemen yang terintegrasi dengan maksud yang sama untuk mencapai suatu tujuan. Hal yang pertama perlu diperhatikan dalam suatu sistem adalah elemen-elemennya.
(Fatta, 2007:3)	Sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur atau variabel-variabel yang saling terorganisasi, saling berinteraksi, dan saling bergantung sama lain

Dari berbagai uraian pada Tabel 2.2 dapat disimpulkan bahwa pengertian sistem adalah sebuah tatanan (keterpaduan), sekelompok unsur, atau elemen yang berhubungan satu dengan yang lain untuk mencapai suatu tujuan.

2.4 Informasi

Berbagai definisi tentang informasi yang dikutip dari berbagai sumber yang dapat dilihat pada Tabel 2.3 untuk memberikan penjelasan lebih lanjut tentang informasi.

Tabel 2.3 Definisi Informasi

Sumber	Definisi
(Sutabri, 2012:22)	Informasi adalah data yang telah diklasifikasikan atau diolah atau diinterpretasikan untuk digunakan dalam proses pengambilan keputusan.

(Sutanta, 2011:13)	Informasi merupakan hasil pengolahan data sehingga menjadi bentuk yang penting bagi penerimanya dan mempunyai kegunaan sebagai dasar dalam pengambilan keputusan yang dapat dirasakan akibatnya secara langsung saat itu juga atau secara tidak langsung pada saat mendatang.
(Nugroho, 2010:17)	Informasi merupakan salah satu elemen dalam manajemen perusahaan. Agar informasi dapat mengalir lancar, para manajer perlu menempatkan informasi dalam suatu kerangka sistem.

Dari berbagai uraian pada Tabel 2.3 dapat disimpulkan bahwa pengertian informasi adalah data yang diolah atau diproses menjadi sebuah bentuk yang berarti bagi penerimanya sehingga dapat dijadikan sebuah keputusan.

2.5 Pengertian Aplikasi *Web*

Aplikasi *web* merupakan sebuah aplikasi yang menggunakan teknologi *browser* untuk menjalankan aplikasi dan diakses melalui jaringan komputer (Remick; 2011). Sedangkan menurut (Rouse; 2011) aplikasi *web* adalah sebuah program yang disimpan di *Server* dan dikirim melalui internet dan diakses melalui antarmuka *browser*.

Dari pengertian diatas dapat disimpulkan aplikasi *web* merupakan aplikasi yang diakses menggunakan *web browser* melalui jaringan internet atau intranet. Aplikasi *web* juga merupakan suatu perangkat lunak komputer yang dikodekan dalam bahasa pemrograman yang mendukung perangkat lunak berbasis *web* seperti *HTML*, *JavaScript*, *CSS*, *Ruby*, *Python*, *Php*, *Java* dan bahasa pemrograman lainnya. Adapun Gambaran rincian aplikasi *web* adalah sebagai berikut :



Gambar 2.1 Contoh Aplikasi *Web*

2.6 Inventory

Menurut Sugiama (2013:173) *inventory* aset adalah serangkaian kegiatan untuk melakukan pendataan, pencatatan, pelaporan hasil pendataan aset, dan mendokumentasikannya baik aset berwujud maupun aset tidak berwujud pada suatu waktu tertentu. Berdasarkan uraian di atas, *inventory* merupakan kumpulan beberapa unsur/item terkait yang dapat mengolah *input* menjadi *output*.

Dimana, *input* yang diproses/dikelola mencakup proses pendataan, pencatatan serta pengecekan mengenai kualitas dan kuantitas aset secara fisik dan yuridis/legal, kemudian dapat menghasilkan keluaran (*output*) berupa laporan atau dokumentasi untuk kepentingan tertentu.

2.7 Sistem *Inventory*

Sistem *inventory* adalah sekumpulan kebijakan dan pengendalian, yang memonitor tingkat *inventory*, dan menentukan tingkat mana yang harus dijaga, bila stok harus diisi kembali dan berapa banyak yang harus dipesan (Assauri, 2016:225).

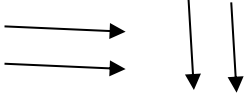
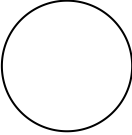
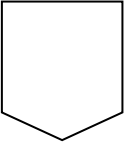

Sistem *inventory* akan memberikan kemungkinan struktur organisasi dan

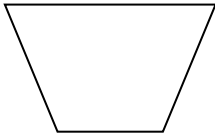
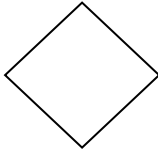
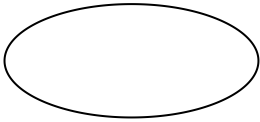
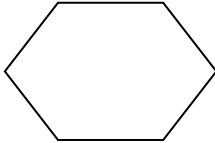
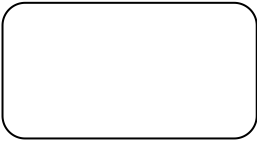
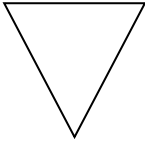
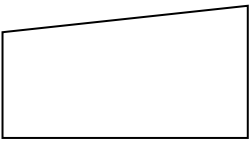

kebijakan operasi produksi, untuk menjaga dan mengawasi barang-barang untuk distok. Dengan sistem *inventory* ini, diharapkan manajemen dapat bertanggung jawab terhadap pemesanan dan penerimaan barang yang dipesan. Hal ini dapat dilakukan dengan mengawasi waktu penempatan pesanan, dan menjaga atau mengawasi jalannya jalur dari apa yang dipesan, serta berapa banyak barang yang dipesan dan dari siapa vendor-nya (Assauri, 2016:229).

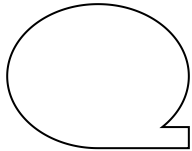
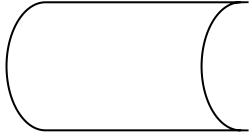


2.8 Flowchart

Bagan alir (*flowchart*) adalah bagan (*chart*) yang menunjukkan alir (*flow*) di dalam program atau prosedur sistem secara logika (Jogiyanto, 2005:795). Bagan alir sistem (*systems flowchart*) merupakan bagan yang menunjukkan arus pekerjaan secara keseluruhan dari sistem. Bagan ini menunjukkan urutan dari prosedur-prosedur dan menunjukkan apa yang dikerjakan sistem (Jogiyanto, 2005:796). Berikut simbol bagan alir sistem (*systems flowchart*) dapat dilihat pada tabel 2.6 :

Tabel 2.2 Simbol *Systems Flowchart*

NO	SIMBOL	KETERANGAN
1		Simbol arus/ <i>flow</i> , berfungsi untuk menyatakan jalannya arus suatu proses
2		Simbol <i>connector</i> , berfungsi untuk menyatakan sambungan dari proses ke proses lainnya dalam halaman yang sama
3		Simbol <i>offline connector</i> , berfungsi untuk menyatakan sambungan dari proses ke proses lainnya dalam halaman yang berbeda
4		Simbol <i>process</i> , berfungsi untuk menyatakan suatu tindakan (proses)

		yang dilakukan oleh komputer
5		Simbol <i>manual</i> , berfungsi untuk menyatakan suatu tindakan (proses) yang tidak dilakukan oleh komputer
6		Simbol <i>decision</i> , berfungsi untuk menunjukkan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban : ya/tidak
7		Simbol <i>terminal</i> , berfungsi untuk menyatakan permulaan atau akhir suatu program
8		Simbol <i>predefined process</i> , berfungsi untuk menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal
9		Simbol <i>keying operation</i> , berfungsi untuk menyatakan segala jenis operasi yang diproses dengan menggunakan suatu mesin yang mempunyai <i>keyboard</i>
10		Simbol <i>offline-storage</i> , berfungsi untuk menunjukkan bahwa data dalam simbol ini akan disimpan ke suatu media tertentu
11		Simbol manual input, berfungsi untuk memasukkan data secara manual dengan menggunakan online <i>keyboard</i>
12		Simbol <i>input/output</i> , berfungsi untuk

		menyatakan proses input atau output tanpa tergantung jenis peralatannya
13		Simbol <i>magnetic tape</i> , berfungsi untuk menyatakan input berasal dari pita magnetis atau output disimpan ke pita magnetis
14		Simbol <i>disk storage</i> , berfungsi untuk menyatakan <i>input</i> berasal dari <i>disk</i> atau <i>output</i> disimpan ke <i>disk</i>
15		Simbol <i>document</i> , berfungsi untuk mencetak keluaran dalam bentuk dokumen (melalui printer)
16		Simbol <i>punched card</i> , berfungsi untuk menyatakan <i>input</i> berasal dari kartu atau <i>output</i> ditulis ke kartu

2.9 MySQL (*My Structured Query Language*)

MySQL adalah salah satu jenis *database server* yang sangat terkenal. Kepopulerannya di sebabkan *MySQL* menggunakan *SQL* sebagai bahasa dasar untuk mengakses *databasenya*. *MySQL* bersifat *free* pada berbagai *platform* (kecuali pada Windows, yang bersifat *shareware* atau anda perlu membayar setelah melakukan evaluasi dan memutuskan untuk digunakan untuk keperluan produksi). *MySQL* dilisensi di bawah GNU *General Public License* (GPL).

Dengan adanya keadaan seperti itu maka anda dapat menggunakan *software database* ini dengan bebas tanpa harus takut dengan lisensi yang ada.

(Dalam jurnal Peranan *Egovernment* Dalam Rangka Mewujudkan Potensi dalam Desa Karang Sari, tahun 2009 oleh Anisa Sulitiani).

2.10 DataBase

Basis data terdiri atas 2 kata, yaitu Basis dan Data. Basis kurang lebih dapat diartikan sebagai markas atau gudang, tempat bersarang/berkumpul. Sedangkan Data adalah representasi fakta dunia nyata yang mewakili suatu objek seperti manusia (pegawai, siswa, pembeli, pelanggan), barang, Jurnal TAM (*Technology Acceptance Model*) Volume 3 Desember 2014 3 hewan, peristiwa, konsep, keadaan dan sebagainya, yang direkam dalam bentuk angka, huruf, simbol, teks, gambar, bunyi atau kombinasinya.

2.11 HTML (*Hypertext Markup Language*)

Menurut Fiber Siregar dan Muhammad (2009), *HTML* adalah kepanjangan dari *Hyper Text Markup Language* yang merupakan bahasa yang digunakan untuk membuat suatu situs *web* atau *home page*, setiap dokumen dalam *web* ditulis dengan format *HTML*.

Semua format dokumen *hyperlink* yang dapat di klik gambar, dokument, multimedia, *form* yang dapat diisi di dasarnya atas *HTML*. Setiap menggunakan *HTML* dalam membuat *web* diperlukan juga *syntax PHP*, dengan menggunakan *PHP user* tidak akan melihat kode-kode *PHP* yang telah ditulis tersebut di dalam *browser*.

Umumnya semua dokumen *web* dibagi menjadi dua. *Section* (bagian), yaitu *section head*, dan *section body*. (Dalam Jurnal Aplikasi *egovernment* pada Desa Sinar Harapan Kabupaten Tanggamus, tahun 2013 oleh Yuli Kartika).

2.12 PHP (*Personal Home Page*)

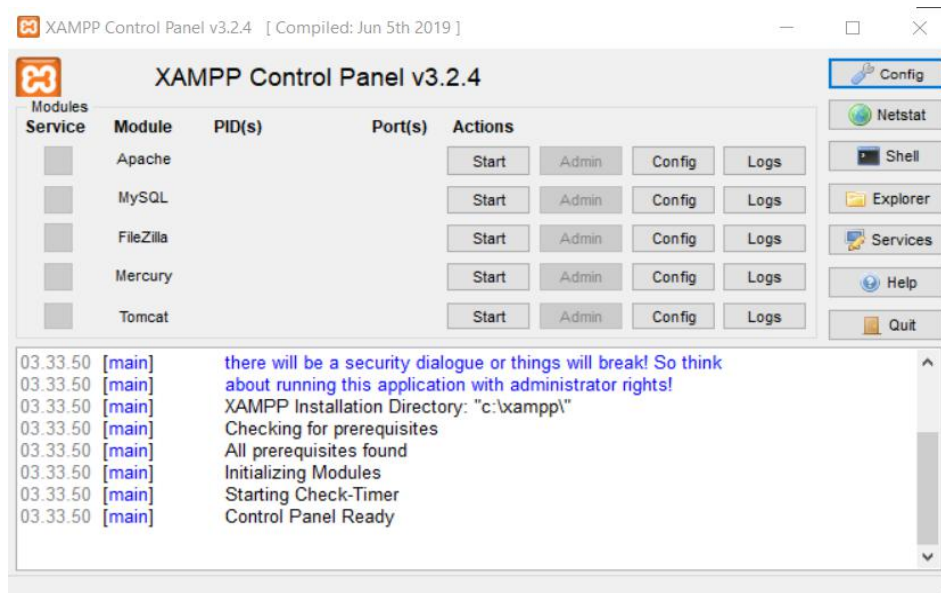
PHP adalah bahasa (*scripting language*) yang dirancang secara khusus untuk penggunaan pada *web*. *PHP* adalah *tool* untuk pembuatan halaman *web* dinamis. Kaya akan fitur yang membuat perancangan *web* dan pemrograman lebih mudah, *PHP* digunakan pada 13 juta domain (menurut survei *Netcraft* pada www.php.net/usage.php). *PHP* kependekan dari *Hyper Text Processor*.

Pada awal pengembangannya oleh Rumus Laddrof, dia menyebutnya sebagai *tools Personal Home Page*. (Dalam Jurnal Peranan *E-government* Dalam Rangka Mewujudkan *Good Governance* Bagi Masyarakat Sekolah Tinggi Teknologi Adisutjipto Yogyakarta, tahun 2008 oleh Nurcahyani Dewi Retnowati).

2.13 *Xampp*

Xampp merupakan paket *PHP* yang berbasis *Open Source* yang dikembangkan oleh sebuah komunitas *Open Source* (Nugroho, 2008:74). Penggunaan perangkat lunak *XAMPP* diawali dengan install paket *Xampp* pada halaman resmi. Tersedia beberapa *update* yang dapat di *download* sesuai dengan *platform* komputer pengguna.

Setelah penginstalan selesai maka pengguna dapat memulai pemrograman dengan membuka *XAMPP Control Panel* terlebih dahulu untuk mengaktifkan *service* yang disediakan seperti : *Apache, MySQL, FileZilla, Mercury dan Tomcat* dengan mengklik *Action : Start*.



Gambar 2.2 XAMPP Control Panel

2.14 *Framework Bootstrap*

Menurut (Eko, 2016:11), “bootstrap merupakan salah satu framework HTML, CSS, Dan JS yang digunakan untuk membuat website yang bersifat

responsive atau bisa menyesuaikan tampilan layout nya berdasarkan ukuran viewport dari device pengaksesnya, mulai dari smartphone, tablet, maupun layar PC”.

Bootstrap adalah sebuah framework yang dibuat dengan menggunakan bahasa dari HTML dan CSS, namun juga menyediakan efek javascript yang dibangun dengan menggunakan jquery. Bootstrap telah menyediakan kumpulan komponen class interface dasar yang telah dirancang sedemikian rupa untuk menciptakan tampilan yang menarik, bersih dan ringan.

Selain itu, bootstrap juga memiliki fitur grid yang berfungsi untuk mengatur layout yang bisa digunakan dengan sangat mudah dan cepat. Kita juga diberi keleluasaan dalam mengembangkan tampilan website yang menggunakan bootstrap yaitu dengan mengubah tampilan bootstrap dengan menambahkan class dan CSS sendiri.

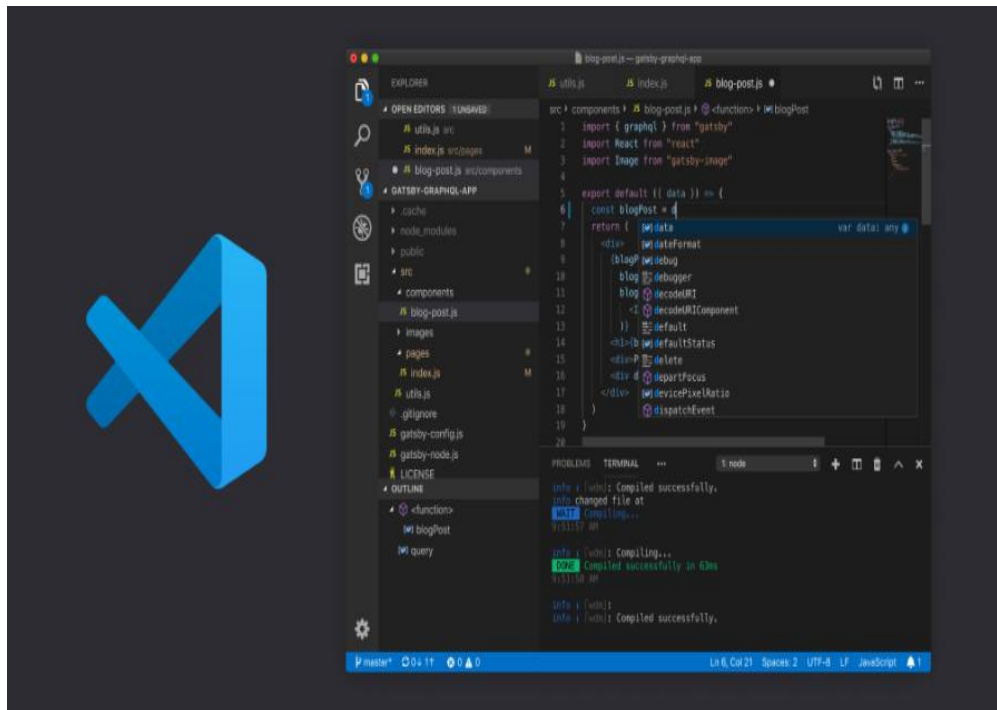
2.15 *Visual Studio Code*

Visual Studio Code (VS Code) ini adalah sebuah *teks editor* ringan dan handal yang dibuat oleh Microsoft untuk *sistem operasi multiplatform*, artinya tersedia juga untuk versi Linux, Mac, dan Windows. *Teks editor* ini secara langsung mendukung bahasa pemrograman *JavaScript*, *Typescript*, dan *Node.js*, serta bahasa pemrograman lainnya dengan bantuan *plugin* yang dapat dipasang via *marketplace Visual Studio Code* (seperti *C++*, *C#*, *Python*, *Go*, *Java*, dst).

Banyak sekali fitur-fitur yang disediakan oleh *Visual Studio Code*, diantaranya *Intellisense*, *Git Integration*, *Debugging*, dan fitur ekstensi yang menambah kemampuan *teks editor*. Fitur-fitur tersebut akan terus bertambah seiring dengan bertambahnya versi *Visual Studio Code*. Pembaruan versi *Visual Studio Code* ini juga dilakukan berkala setiap bulan, dan inilah yang membedakan *VS Code* dengan *teks editor-teks editor* yang lain.

Teks editor VS Code juga bersifat *open source*, yang mana kode sumbernya dapat kalian lihat dan kalian dapat berkontribusi untuk pengembangannya. Kode sumber dari *VS Code* ini pun dapat dilihat di *link Github*. Hal ini juga yang membuat *VS Code* menjadi favorit para pengembang aplikasi,

karena para pengembang aplikasi bisa ikut serta dalam proses pengembangan *VS Code* ke depannya.



Gambar 2.3 Tampilan Aplikasi Visual Studio Code

Microsoft Visual Studio Code adalah one-stop shop yang memungkinkan kita fokus pada proses pengembangan dan melupakan tools baru. Berikut beberapa keunggulan fitur Visual Studio Code:

1. **Cross platform** – tersedia di macOS, Linux dan Windows artinya Anda dapat bekerja pada sistem operasi manapun tanpa khawatir belajar coding tools yang sama untuk sistem yang berbeda-beda.
2. **Lightweight** – tak perlu menunggu lama untuk memulai. Anda mengontrol sepenuhnya bahasa, tema, debugger, commands dan lain-lainnya sesuai keinginan. Ini dapat dilakukan melalui *extensions* untuk bahasa populer seperti python, node.js, java dan lain-lainnya di Visual Studio Code Marketplace.
3. **Powerful editor** – memfungsikan fitur untuk source code editing yang sangat produktif, seperti membuat code snippets, IntelliSense, auto correct, dan formatting.

4. **Code Debugging** – salah satu fitur terkeren yang ditawarkan Visual Studio Code adalah membantu Anda melakukan *debug* pada kode dengan cara mengawasi kode, variabel, call stack dan expression yang mana saja.
5. **Source control** – Visual Studio Code memiliki integrated source control termasuk *Git support in-the-box* dan penyedia source code control lainnya di pasaran. Ini meningkatkan siklus rilis proyek Anda secara signifikan.
6. **Integrated terminal** – Tiada lagi *multiple windows* dan alt-tabs. Anda dapat melakukan command-line task sekejap dan membuat banyak terminal di dalam editor.

2.16 Laravel

Laravel adalah *framework PHP* yang *expressif*, artinya ketika melihat suatu *sintaks* Laravel, seorang programmer diharapkan akan langsung tahu kegunaan dari *sintaks* tersebut meskipun belum pernah mempelajarinya apalagi menggunakannya. Sekarang coba bandingkan dua buah kode berikut yang memiliki tujuan yang sama, tetapi dengan gaya penulisan yang berbeda.

```
// kode 1, framework xxx
$uri = Uri::create('some/uri',array(),array(),true);

// kode 2, framework laravel
$url = URL::to\_secure\('some/uri'\);
```

Untuk kode pertama, Anda pasti bertanya-tanya apa sih maksud dari parameter kedua, ketiga dan keempat, singkatnya parameter ke empat berfungsi sebagai *lagging*, *true* untuk *https* dan *false* untuk *http*. Laravel melakukan pendekatan yang berbeda dengan membuang parameter yang sifatnya *lagging* dan memilih untuk membuat dua fungsi yang berbeda. Menurut saya, pendekatan semacam ini lebih jelas karena minimal satu parameter fungsi telah berkurang.

2.16.1 Fitur-Fitur Laravel

Beberapa fitur yang dimiliki oleh framework Laravel adalah sebagai berikut:

- 1) *Bundles* yaitu sebuah fitur dengan *system* pengemasan modular dan berbagai *bundle* telah tersedia untuk di gunakan dalam aplikasi Anda.
- 2) *Eloquent ORM* merupakan penerapan *PHP* lanjutan dari pola “*active record*” menyediakan metode internal untuk mengatasi kendala hubungan antara objek *database*. Pembangun query Laravel *Fluent* didukung *Eloquent*.
- 3) *Application Logic* merupakan bagian dari aplikasi yang dikembangkan, baik menggunakan *Controllers* maupun sebagai bagian dari deklarasi *Route*. *Sintaks* yang digunakan untuk mendefinisikannya mirip dengan yang digunakan oleh *framework Sinatra*.
- 4) *Reverse Routing*, mendefinisikan hubungan antara *Link* dan *Route*, sehingga jika suatu saat ada perubahan pada *route* secara otomatis akan tersambung dengan link yang relevan. Ketika Link yang dibuat dengan menggunakan nama -nama dari *Route* yang ada, secara otomatis Laravel akan membuat URI yang sesuai.
- 5) *Restful Controllers*, memberikan sebuah *option* (pilihan) untuk memisahkan logika dalam melayani *HTTP GET* dan permintaan *POST*.
- 6) *Class Auto Loading*, menyediakan otomatis *loading* untuk *class-class PHP*, tanpa membutuhkan pemeriksaan manual terhadap jalur masuknya. Fitur ini mencegah *loading* yang tidak perlu.
- 7) *View Composers* adalah kode unit *logical* yang dapat dijalankan ketika sebuah *View* di *load*.
- 8) *IoC Container* memungkinkan untuk objek baru yang dihasilkan dengan mengikuti prinsip *control* pembalik, dengan pilhan contoh dan referensi dari objek baru sebagai *Singletons*.
- 9) *Migrations* menyediakan versi sistem *control* untuk skema *database*, sehingga memungkinkan untuk menghubungkan perubahan adalah basis kode aplikasi dan keperluan yang dibutuhkan dalam merubah tata letak *database*, mempermudah dalam penempatan dan memperbarui aplikasi.

- 10) *Unit Testing* mempunyai peran penting dalam *framework* Laravel, dimana *unit testing* ini mempunyai banyak tes untuk mendeteksi dan mencegah *regresi*. *Unit testing* dapat dijalankan melalui itur “*artisan command -line*”.
- 11) *Automatic pagination* menyederhanakan tugas dari penerapan halaman, menggantikan penerapan yang manual dengan metode otomatis yang terintegrasi ke Laravel.

2.16.2 Composer

Seperti yang sudah dijelaskan sebelumnya bahwa Laravel berbeda dengan *framework* pada umumnya, jika *framework* pada umumnya itu download *framework* kemudian taruh didalam *web server*, setelah itu dijalankan, tetapi Laravel ini seperti pada *Linux*, jadi untuk *update download* dan sebagainya menggunakan *command* atau perintah didalam *command prompt*, Nah perintah tersebut yang dinamakan dengan *composer*, maka langkah pertama untuk menjalankan Laravel adalah *mendownload composer*, lalu menginstallnya.

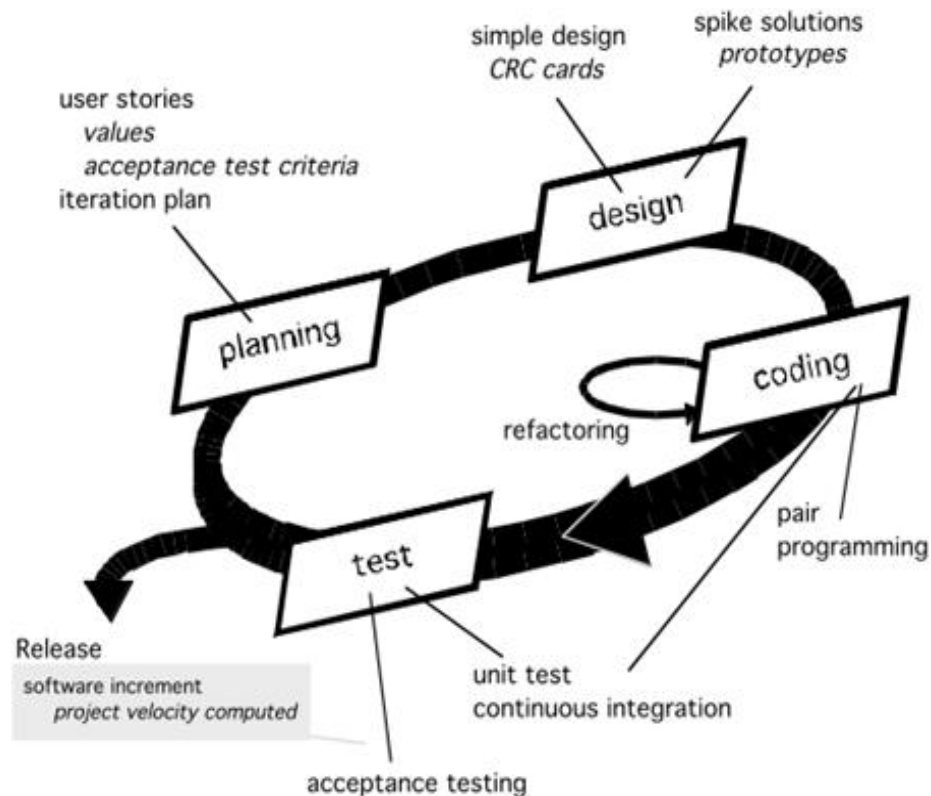
Composer sendiri adalah fitur (*dependency*) tambahan untuk *PHP* yang memiliki basis layaknya *Command Line* dan berfungsi sebagai penginstall *third-party plugin* untuk aplikasi *web* secara cepat.

2.17 Metode Pengembangan Software

Dalam pembangunan sistem informasi atau aplikasi terdapat beberapa pendekatan atau metodologi yang dapat digunakan antara lain *System Development Life Cycle (SDLC)*, dan *Extreme Programming (XP)*. Pada penelitian ini metodologi yang digunakan dalam pembangunan aplikasi inventory ruang belajar berbasis web pada smk negeri 1 kayuagung adalah metode *Extreme Programming(XP)*.

Menurut Prabowo dalam (Supriyatna, 2018) *Extreme Programming (XP)* merupakan sebuah proses rekayasa perangkat lunak yang cenderung menggunakan pendekatan berorientasi objek dan sasaran dari metode ini adalah tim yang dibentuk dalam skala kecil sampai medium serta metode ini juga sesuai jika tim dihadapkan dengan *requirement* yang tidak jelas maupun terjadi perubahan-perubahan *requirement* yang sangat cepat.

Sedangkan menurut Ferdiana dalam (Lubis, 2016) *Extreme Programming (XP)* dikenal dengan metode atau “*technical how to*” bagaimana suatu tim teknis mengembangkan perangkat lunak secara efisien melalui berbagai prinsip dan teknik praktis pengembangan perangkat lunak. XP menjadi dasar bagaimana tim bekerja sehari-hari.



Gambar 2.4 Metode *Extreme Programming*.

Terdapat empat tahapan yang harus dikerjakan pada metode *extreme programmin(xp)* yaitu:

1. *Planning* (Perencanaan).

Tahapan ini merupakan langkah awal dalam pembangunan sistem dimana dalam tahapan ini dilakukan beberapa kegiatan perencanaan yaitu, identifikasi permasalahan, menganalisa kebutuhan sampai dengan penetapan jadwal pelaksanaan pembangunan sistem.

2. *Design* (Perancangan).

Tahapan berikutnya adalah perancangan dimana pada tahapan ini dilakukan kegiatan pemodelan yang dimulai dari pemodelan sistem, pemodelan arsitektur sampai dengan pemodelan basis data. Pemodelan sistem dan arsitektur menggunakan diagram *Unified Modelling Language (UML)* sedangkan pemodelan basis data menggunakan *Entity Relationship Diagram (ERD)*.

3. *Coding* (Pengkodean).

Tahapan ini merupakan kegiatan penerapan pemodelan yang sudah dibuat kedalam bentuk *user interface* dengan menggunakan bahasa pemrograman. Adapun bahasa pemrograman yang digunakan adalah PHP dengan metode terstruktur. Untuk sistem manajemen basis data menggunakan piranti lunak MySQL.

4. *Testing* (Pengujian).

Setelah tahapan pengkodean selesai, kemudian dilakukan tahapan pengujian sistem untuk mengetahui kesalahan apa saja yang timbul saat aplikasi sedang berjalan serta mengetahui apakah sistem yang dibangun sudah sesuai dengan kebutuhan pengguna. Metode pengujian yang digunakan pada tahapan ini adalah metode *blackbox testing*, dimana pengujian yang dilakukan terhadap form beberapa masukkan apakah sudah berjalan sesuai dengan fungsinya masing- masing.

2.18 Entity Relationship Diagram (ERD)

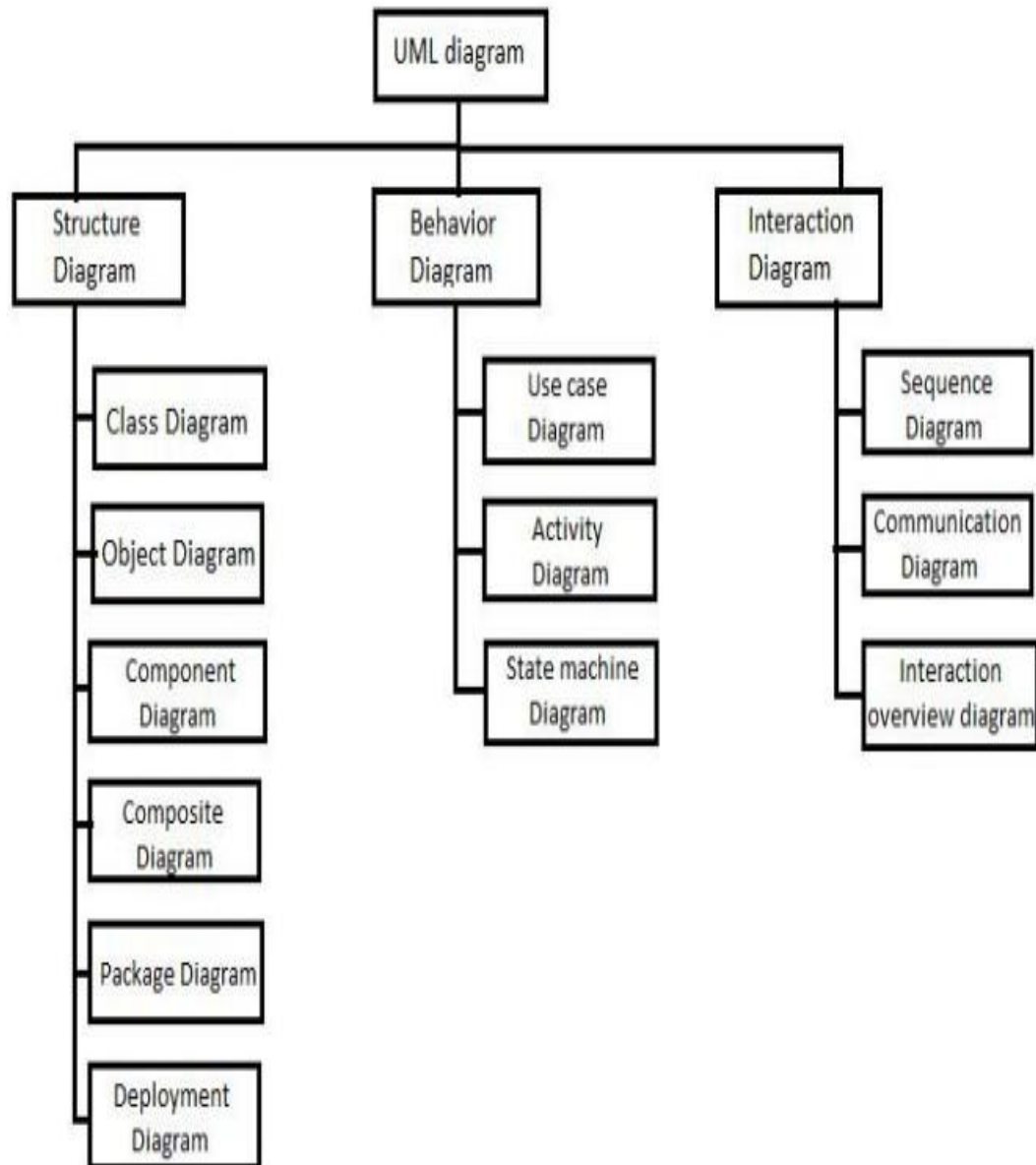
ERD dikembangkan berdasarkan teori himpunan dalam bidang matematika. ERD digunakan untuk pemodelan basis data relasional. Sehingga jika penyimpanan basis data menggunakan *OODBMS* maka perancangan data tidak perlu menggunakan ERD (Lubis, 2016).

2.19 Unified Modelling Language(UML)

UML (Unified Modeling Language) adalah bahasa pemodelan untuk sistem atau perangkat lunak yang berparadigma berorientasi objek. Pemodelan (*modeling*) sesungguhnya digunakan untuk penyederhanaan permasalahan-

permasalahan yang kompleks sedemikian rupa sehingga lebih mudah dipelajari dan dipahami (Nugroho, 2010).

Menurut (Sukamto & Shalahuddin, 2014) *UML* terdiri dari 13 macam diagram yang dikelompokkan dalam 3 kategori. Pembagian kategori dan macam-macam diagram tersebut dapat dilihat pada gambar dibawah ini:



Gambar 2.5 Diagram UML

