

BAB III. LANDASAN TEORI

3.1. Sistem Informasi

Sistem informasi merupakan sebuah sistem yang terdiri dari gabungan berbagai sub-sistem yang saling bergantung satu sama lain untuk mencapai tujuan yang sama, yaitu untuk mengolah suatu informasi sehingga dapat digunakan untuk melakukan pengambilan keputusan. Sistem informasi dapat mencakup beberapa komponen seperti manusia, komputer, teknologi maupun prosedur kerja. Dalam suatu sistem informasi terdapat beberapa tahapan yang dilakukan untuk mengolah data yaitu input, proses, output, penyimpanan, dan control [8].

3.2. Pendaftaran

Pendaftaran adalah pencatatan data atau identitas seseorang seperti nama, alamat, pekerjaan dan sebagainya yang kedepannya dapat digunakan untuk pengambilan keputusan melalui seleksi. Proses pendaftaran dapat digunakan untuk mempermudah dalam mengorganisir data. Dalam prosesnya, pendaftaran dapat dilakukan secara fisik melalui berkas-berkas pendaftaran yang harus dikumpulkan oleh pendaftar maupun secara digital melalui sebuah sistem informasi [9].

3.3. Sistem Informasi Pendaftaran

Sistem informasi pendaftaran adalah sebuah sistem yang mampu menangani kegiatan pendaftaran melalui komputer secara digital atau online. Sistem informasi pendaftaran dapat memiliki kemampuan seperti pencatatan dan pengelolaan data pendaftar, visualisasi data, penyaringan data serta sinkronisasi dan integrasi data secara langsung. Dengan adanya kemampuan tersebut, diharapkan pengguna sistem dapat lebih terbantu dalam mengolah dan menyaring data secara lebih cepat, mudah dan akurat. [9].

3.4. *World Wide Web*

World Wide Web ditemukan oleh seorang ahli komputer berkebangsaan Inggris yaitu Tim Berners Lee pada 12 Maret 1989. Pada saat itu, ia bekerja untuk

lab fisika CERN (Center for Nuclear Research) di Jenewa, Swiss. Hal pertama yang dilakukan oleh Lee dan kedua orang rekannya yaitu Douglas dan Bush adalah merancang *HyperText Transfer Protocol* (HTTP). Bahasa program tersebut mengatur bagaimana sebuah data didistribusikan ke banyak komputer. Setelah merancang HTTP kemudian Lee menciptakan Hyper Text Markup Language (HTML) sebagai pondasi sebuah laman *website*. Secara garis besar, *World Wide Web* ini memiliki fungsi sebagai penyedia data dan informasi yang dibutuhkan oleh pengguna. Selain itu, *World Wide Web* berfungsi juga sebagai penghubung antara pengguna dengan berbagai *web server* agar pengguna dapat mengakses berbagai data dan informasi yang terdapat di situs tertentu [10].

World Wide Web memiliki beberapa layanan yang sangat bermanfaat bagi pengguna diantaranya adalah *web mail service* yang dapat digunakan oleh pengguna untuk berkirim surat elektronik dengan orang lain. Layanan *web mail* ini dapat digunakan secara gratis maupun berbayar. Layanan gratis yang sering dipakai oleh pengguna contohnya seperti Gmail dan Yahoo. Dan untuk layanan email yang berbayar, biasanya digunakan untuk keperluan perusahaan, lembaga pendidikan dengan *domain* milik sendiri. Kemudian layanan berikutnya yaitu search engine yang digunakan untuk mencari informasi berdasarkan kata kunci tertentu. Layanan lainnya yaitu *web hosting* dan portal. *Web hosting* digunakan untuk menyimpan *database* dari sebuah *website*. Sedangkan portal adalah layanan yang didalamnya terdapat berbagai macam layanan sekaligus seperti yahoo.com yang memungkinkan pengguna untuk membaca berita, berkirim surat elektronik dan melakukan tanya jawab [11].

3.5. Website

Website adalah kumpulan dari halaman – halaman situs, yang terangkum dalam sebuah *domain* atau *subdomain*, yang berada di *World Wide Web* Internet. Biasanya, sebuah halaman *web* berwujud dokumen yang ditulis dalam format HTML (*Hyper Text Markup Language*) dan dapat diakses melalui HTTP, yaitu sebuah protokol yang bertugas untuk menyampaikan informasi dari *server website* kepada para pengguna melalui *web browser*. Informasi tersebut dapat berupa data

teks, data gambar baik diam atau bergerak, data animasi, suara, video maupun gabungan dari semua jenis data yang ada. Data yang ditampilkan dapat bersifat statis maupun dinamis yang membentuk satu rangkaian yang saling terkait dimana masing-masing dihubungkan dengan *hyperlink* [12].

3.6. Framework Bootstrap

Bootstrap adalah kumpulan dari berbagai fungsi yang terdapat di *framework* CSS yang dibuat secara khusus di bagian pengembangan *front-end website*. Zaman sekarang, sudah banyak *web developer* yang menggunakan *framework* bootstrap untuk pengembangan *front-end* karena lebih mudah dan cepat. Para *web developer* hanya menambahkan *class* tertentu seperti membuat *grid*, navigasi, dan lain sebagainya. Pada *framework* ini juga disediakan berbagai komponen dasar *class interface* sehingga tampilan *website* terlihat lebih menarik. Selain itu, dengan menggunakan bootstrap, *website* akan lebih ringan dan bersih. Bootstrap juga memberikan kebebasan bagi para *web developer* dalam mengembangkan tampilan *website* [13].

Banyak keuntungan yang bisa didapat jika menggunakan *framework* bootstrap, antara lain tampilan *website* mudah menyesuaikan layar dan *browser* dengan *fluid grid layout*. Dengan begitu *website* dapat beradaptasi dengan resolusi serta ukuran layar ponsel maupun PC. Bootstrap sangat mudah digunakan, bahkan bagi yang memiliki pengetahuan dasar mengenai HTML dan CSS pun dapat mengembangkan *website* dengan bootstrap. Kemudian bootstrap memungkinkan bagi *web developer* untuk melakukan pengembangan lebih cepat karena *web developer* juga dapat menggunakan *template* gratis yang ada dalam bootstrap. Keuntungan lainnya yaitu *tools* yang digunakan dalam Bootstrap memiliki kecenderungan tetap sama dan konsisten. Selain itu, bootstrap dapat diakses secara gratis sehingga tidak perlu membeli lisensi. Bootstrap juga memiliki pilihan kustomisasi yang lengkap, sehingga memudahkan *developer* untuk mengkustomisasi bagian per bagian dari *website* [14].

3.7. Waterfall

Metode *Waterfall* adalah salah satu metode pengembangan perangkat

lunak sekuensial di mana proses pengembangannya mengalir semakin ke bawah (mirip dengan air terjun) melalui tahapan-tahapan yang sudah ditentukan dan harus dijalankan untuk berhasil membangun sebuah perangkat lunak. Karena modelnya mirip seperti air terjun, maka pengerjaannya tidak dapat kembali ke fase atau tahap sebelumnya. Awalnya, metode *waterfall* ini dikemukakan pertama kali oleh Winston W. Royce pada tahun 1970 untuk menjelaskan praktik rekayasa perangkat lunak. Metode *waterfall* mendefinisikan beberapa fase atau tahapan berturut-turut yang harus diselesaikan satu per satu lalu pindah ke fase berikutnya jika fase sebelumnya telah selesai sepenuhnya. Oleh karena itu, metode *waterfall* bersifat rekursif pada setiap fasenya karena dapat diulang terus-menerus hingga fase tersebut selesai dengan sempurna. Metode *Waterfall* terdiri dari lima fase di antaranya: analisis, desain, implementasi, pengujian, dan pemeliharaan [15].

Kelebihan dari metode *waterfall* antara lain baik untuk proyek-proyek dengan tujuan sederhana dan dapat dipahami dengan baik, simpel, setiap langkah jelas dan mudah dipahami, serta konsumen dengan mudah mengetahui tentang proyek yang sedang dikerjakan. Sedangkan kerugian menggunakan metode *waterfall* antara lain tidak fleksibel yaitu sampai satu tahap selesai seseorang tidak bisa pergi ke tahap selanjutnya [16].

3.8. MVC

MVC (*Model-View-Controller*) adalah metode pengembangan *web* yang memisahkan setiap bagian utama dari sebuah *web* mulai dari bagian manipulasi data, bagian *user interface*, dan bagian yang mengontrol sebuah *web*. Saat ini, MVC banyak digunakan oleh berbagai *framework* karena dapat mempermudah perawatan suatu program. Selain itu juga mengingat bagian program yang telah dipetakan dalam suatu struktur yang jelas sehingga tidak mengubah *logic* maupun data [2].

Sesuai dengan namanya, dalam MVC ini terdapat 3 bagian yaitu *Model*, *View*, dan *Controller*. *Model* adalah bagian yang berisi struktur data yang berfungsi untuk menyimpan, mengatur dan memanipulasi data. Bagian model ini adalah bagian yang memiliki tugas untuk mengelola data yang berasal dari bagian

controller. Bagian yang kedua yaitu *View*. *View* adalah bagian *user interface* yang menjadi perantara antara pengguna dengan aplikasi. Bagian *view* bertugas untuk mengatur tampilan aplikasi dan juga menampilkan data dari bagian model. *View* dapat dikatakan sebagai *front-end* dari sebuah *website*. Kemudian bagian terakhir yaitu *Controller*. *Controller* berfungsi untuk melakukan pengolahan sebuah data dari model kemudian dikirim ke halaman *web* di bagian *view*. *Controller* juga akan menerima data yang dikirim oleh pengguna dan bagian *controller* ini merupakan back end dari *website* [2].

Keuntungan menggunakan MVC yaitu dapat memisahkan *business logic* dari proses bisnis dengan *view* tampilan sehingga dapat meningkatkan fleksibilitas. Keuntungan lainnya adalah dapat membantu kerja sama tim dalam mengatur pembagian kerja yang jelas sehingga dapat meminimalkan ketergantungan antar-anggota tim. Kemudian selain yang dipaparkan diatas, dengan menggunakan MVC penyelesaian *error* lebih mudah diselesaikan dan tingkat keamanan *website* juga lebih baik [2].

3.9. *Razor page*

Razor page adalah sintaks pemrograman ASP.NET yang digunakan untuk membuat halaman *web* dinamis dengan bahasa pemrograman C# atau VB.NET. *Razor page* mendukung pengembangan lintas platform dan dapat digunakan untuk sistem operasi Windows, Unix, dan Mac. Framework *razor page* ringan dan fleksibel. *Razor page* direkomendasikan untuk pembuatan HTML sisi *server* lintas platform [17].

3.10. *jQuery*

jQuery adalah *library* javascript yang cukup handal, praktis dan memiliki fitur-fitur yang lengkap. *jQuery* biasanya digunakan untuk pemrosesan di HTML, mulai dari manipulasi dokumen, membuat animasi, dan membuat *event handling*. Untuk penggunaannya, *jQuery* telah didukung dengan API, sehingga dapat bekerja di berbagai macam *browser* [18].

jQuery memiliki berbagai fungsi dan kegunaan, diantaranya yaitu untuk memisahkan javascript dan HTML dan mempersingkat penulisan kode. Selain itu,

jQuery juga mampu mengatasi kompatibilitas antar *browser*, seperti yang telah dipaparkan pada paragraf sebelumnya bahwa jQuery telah didukung API dan dapat bekerja di berbagai macam *browser* sehingga script dapat kompatibel dengan berbagai *browser*. Kemudian kegunaan lainnya yaitu jQuery mampu membantu pengembangan *framework* menjadi lebih mudah [19].

3.11. Database SQL Server

SQL Server adalah *software* RDBMS (*Relational Database Management System*) yang dikembangkan oleh Microsoft. SQL Server mengacu pada standarisasi bahasa SQL (*Structured Query Language*), namun dalam implementasinya, SQL Server menggunakan bahasa T-SQL (*Transact-SQL*), yaitu seperangkat fitur pemrograman tambahan yang bersifat *proprietary* dari Microsoft untuk memperluas fungsionalitas dari bahasa SQL pada SQL Server [20].

Kelebihan menggunakan SQL Server antara lain dapat beroperasi dengan baik pada *Windows Operating System*, dapat mendukung berbagai *software database*, mampu membuat *cluster-cluster* data, dapat digunakan dengan mudah baik oleh *client* maupun *user*, dapat melakukan *recovery* dan *restore* data, mampu mengelola keamanan *password* dengan baik dan dapat melakukan *back-up* data [21].

3.12. C#

C# (dibaca "*See Sharp*") adalah bahasa pemrograman modern, berorientasi objek, dan merupakan bahasa pemrograman yang bersifat *type-safe*. C# berakar pada keluarga bahasa pemrograman C, oleh karena itu C# akan serupa dengan bahasa C, C ++, Java, dan javascript. C# adalah bahasa pemrograman berorientasi objek dan berorientasi komponen. C# menyediakan konstruksi bahasa untuk secara langsung mendukung konsep-konsep ini, menjadikan C# sebagai bahasa alami untuk mengembangkan dan menggunakan komponen perangkat lunak [22].

Beberapa fitur C # membantu konstruksi aplikasi yang kokoh dan tahan lama. *Garbage collection* secara otomatis mengambil kembali memori yang ditempati oleh objek tak terpakai yang tidak terjangkau. *Exception handling* menyediakan pendekatan terstruktur dan dapat diperluas untuk deteksi dan

pemulihan kesalahan. *Lambda expressions* mendukung teknik pemrograman fungsional. *Query syntax* membuat pola umum untuk bekerja dengan data dari sumber mana pun. Dukungan bahasa untuk operasi asinkron menyediakan sintak untuk membangun sistem terdistribusi. *Pattern matching* menyediakan sintak untuk memisahkan data dengan mudah dari algoritma dalam sistem terdistribusi modern. C# memiliki sistem dengan tipe yang terpadu. Semua tipe C#, termasuk tipe primitif seperti *int* dan *double*, mewarisi dari tipe objek *root* tunggal. Semua jenis tipe dapat berbagi satu set operasi yang umum. Nilai dari tipe apa pun dapat disimpan, ditransportasi, dan dioperasikan secara konsisten. Lebih lanjut, C# mendukung tipe referensi dan tipe value yang ditentukan pengguna. C # memungkinkan alokasi dinamis sebuah objek pada memori dan penyimpanan *in-line* dari struktur yang ringan [22].

3.13. ASP.net Core 3.1

ASP.net Core adalah *framework web* gratis dan open source yang merupakan penerus ASP.net yang dikembangkan oleh Microsoft. Namun ASP.net Core versi 3 hanya bekerja pada .net Core. ASP.net Core 3.1 cocok untuk kebutuhan lintas platform karena didukung di Windows, Linux, dan macOS. Selain itu, arsitektur layanan mikro juga didukung dalam versi ini sehingga sangat memungkinkan layanan lintas platform untuk bekerja dengan .net Core. Hal ini juga memudahkan *developer* untuk menjalankan beberapa versi dan aplikasi secara berdampingan di mesin yang sama. Di dalam ASP.net Core 3.1, kebutuhan akan kinerja yang tinggi dan sistem dapat diskalakan karena kinerja dan skalabilitas merupakan dua area utama yang dikembangkan dalam versi ini [23].

Berbeda dengan versi-versi sebelumnya, dalam ASP.net Core 3.1, ada banyak fitur baru yang menarik. Diantaranya yaitu sudah mendukung C# 8 yang dapat menggabungkan beberapa referensi seperti *Async streams*, *Ranges and files*, *Switch Expressions*, dan *Nullable*. Kemudian ada juga fitur Utilize Declarations yang dapat memastikan suatu objek aktif dalam ruang lingkup tertentu. Selain itu, ASP.net Core 3.1 secara *default* dapat memberdayakan *runtime* secara efektif menggunakan *Just-In-Time* (JIT) sehingga kinerja dapat lebih efektif dan efisien.

Fitur lainnya yaitu Windows Desktop Apps, ASP.net Core 3.1 sudah mendukung penggunaan kontrol modern dari Windows UI XAML Library (WinUI) [24].

