

BAB II

TINJAUAN PUSTAKA DAN DASAR TEORI

2.1 Tinjauan Pustaka

Penelitian ini mengacu pada penelitian terdahulu meliputi:

Wathowuan (2014). Aplikasi Paket Wisata Menggunakan Desain *Responsive* Di Kabupaten Raja Ampat. Penelitian ini bertujuan untuk membuat desain *responsive* pada aplikasi untuk membantu dalam *user interface* atau tampilan berdasarkan layar perangkat seperti *mobile (handphone)*, *tablet* dan *desktop*. Kesimpulan dari penelitian ini bahwa aplikasi telah berhasil dibuat dan berjalan baik sesuai dengan fungsinya.

Susilo (2015) Analisis Perbandingan *Framework* CSS Bootstrap Dan Semantic-UI. Penelitian ini bertujuan untuk membandingkan antara *frameworkBootstrap* dan *Semantic-UI* dengan menganalisis kode, melihat ketersediaan informasi mengenai kedua *framework*, serta menganalisis *user interface* dan *usability*. Kesimpulan dari penelitian ini bahwa web yang dibangun dapat mengimplementasikan kedua *framework* CSS sehingga mampu memberikan gambaran dan detail informasi dan gambaran secara umum dari kedua *framework* CSS dan hasilnya *framework* bootstrap lebih baik dibandingkan dengan Semantic-UI dengan mempertimbangkan implementasi kode, *user interface*, penggunaan *resource template* yang diperoleh dari kuesioner.

Salonga (2017). *Responsive Web Design* Pada Sistem Informasi Pariwisata Kabupaten Ende. Penelitian ini bertujuan untuk membuat website yang dapat diakses di berbagai perangkat, seperti: PC, *tablet*, dan *smartphone*. Kesimpulan dari penelitian ini bahwa aplikasi telah berhasil dibuat dan berjalan dengan baik dengan desain *responsive* yang menyesuaikan layar perangkat *mobile*, *tablet* dan *desktop*.

Widodo (2017). Pengembangan Media Pembelajaran Website Berbasis *Responsive Web Design* (RWD) Mata Pelajaran Ekonomi Untuk Materi Pasar Modal Kelas XI Sekolah Menengah Atas (SMA). Penelitian ini bertujuan untuk menghasilkan website berbasis *Responsive Web Design* (RWD) serta mengetahui kelayakan dari website berbasis *Responsive Web Design* (RWD). Kesimpulan dari penelitian ini bahwa website berbasis *responsive web design* (RWD) telah berhasil dikembangkan dan juga dilakukan penilaian kelayakan website.

Agung (2018). Aplikasi Seleksi Penerimaan Siswa Baru Menggunakan Desain *Responsive* Dengan *Framework Twitter Bootstrap*: Studi Kasus SMK Santu Petrus. Penelitian ini bertujuan untuk menghasilkan aplikasi web *responsive* untuk memberikan tampilan yang dinamis menyesuaikan kebutuhan layar pengguna. Kesimpulan dari penelitian ini bahwa sistem ini telah berhasil dibuat dan berjalan baik sesuai dengan fungsinya.

Pada penelitian ini akan dibuat sebuah antarmuka pengguna (*user interface*) sebuah web yang *responsive* menggunakan mesin *templatingPUG* dan *framework Semantic-UI*.

Tabel 2.1 Perbandingan Penelitian Sebelumnya

Penulis	Judul Penelitian	Teknologi	Keterangan
Wathowuan, 2014	Aplikasi Paket Wisata Menggunakan Desain <i>Responsive</i> Di Kabupaten Raja Ampat	PHP, MySQL	Web <i>responsive</i> untuk informasi wisata di kabupaten Raja Ampat
Susilo, 2015	Analisis Perbandingan <i>Framework</i> CSS Bootstrap Dan Semantic-UI	Framework Bootstrap dan Semantic UI	Menganalisa antara <i>framework Bootstrap</i> dan <i>Semantic-UI</i>
Salonga, 2017	<i>Responsive</i> Web <i>Design</i> Pada Sistem Informasi Pariwisata Kabupaten Ende	PHP, MySQL, Bootstrap	Merancang suatu sistem informasi pariwisata dengan menggunakan metode <i>Responsive Web Design</i>
Widodo 2017	Pengembangan Media Pembelajaran Website Berbasis <i>Responsive Web Design</i> (RWD) Mata Pelajaran Ekonomi Untuk Materi Pasar Modal Kelas XI Sekolah Menengah Atas (SMA)	PHP, MySQL, Wordpress	Menghasilkan website berbasis <i>Responsive Web Design</i> (RWD) sebagai media pembelajaran ekonomi untuk materi pasar modal kelas XI Sekolah Menengah Atas (SMA)
Agung, 2018	Aplikasi Seleksi Penerimaan Siswa Baru Menggunakan Desain <i>Responsive</i> Dengan <i>Framework Twitter Bootstrap</i> : Studi Kasus SMK Santu Petrus	PHP, MySQL, Bootstrap	Menghasilkan aplikasi web <i>responsive</i> yang dapat melakukan pendaftaran siswa
Yanta, 2018	Implementasi Template Engine PUG Dan <i>Framework</i> Semantic-UI Pada Aplikasi Berbasis Web	Nodejs, ExpressJS, PUG, Semantic-UI	Menghasilkan sebuah antarmuka pengguna (<i>user interface</i>) sebuah web yang <i>responsive</i> sesuai dengan perangkat yang digunakan

2.2 Dasar Teori

2.2.1 Aplikasi

Aplikasi adalah program yang memiliki aktifitas pemrosesan perintah yang diperlukan untuk melaksanakan permintaan pengguna dengan tujuan tertentu. (Supriyanto, 2005).

2.2.2 Website

Website merupakan sekumpulan halaman-halaman yang digunakan untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara dan atau gabungan dari semuanya, dapat bersifat statis maupun dinamis yang membentuk satu rangkaian bangun saling terkait, yang masing-masing dihubungkan dengan jaringan-jaringan halaman dan jika hubungan satu halaman dengan halaman web lain disebut dengan hiperlink, lalu teks yang dijadikan dijadikan media pendukung disebut hipertext. (Rahmat Hidayat, 2010).

2.2.3 Node.js

Node.js adalah perangkat lunak yang didesain untuk mengembangkan aplikasi berbasis web dan ditulis dalam sintaks bahasa pemrograman JavaScript. Selama ini hanya mengenal JavaScript sebagai bahasa pemrograman yang berjalan di sisi *client/browser* saja, maka Node.js ada untuk melengkapi peran JavaScript sehingga bisa juga berlaku sebagai bahasa pemrograman yang berjalan di sisi server, seperti halnya PHP, Ruby, Perl, dan sebagainya. Node.js dapat berjalan di sistem operasi Windows, Mac OS X dan Linux tanpa perlu ada

perubahan kode program. Node.js memiliki pustaka server HTTP sendiri sehingga memungkinkan untuk menjalankan server web tanpa menggunakan program server web seperti *Apache* atau *Nginx*. (Anditya, 2015).

2.2.4 ExpressJS

ExpressJS merupakan framework aplikasi web Node.js yang fleksibel dan minimal, serta menyediakan fitur yang kuat bagi web dan *mobile*. Express memiliki fungsi utama pada aplikasi server ini, karena menggantikan modul HTTP yang umumnya digunakan pada Node.js. ExpressJS bekerja sebagai *framework view engine* yang juga mengatur folder untuk file tampilan halaman, file *framework* halaman HTML yang digunakan, dan *passing* data menuju halaman pada saat diunduh.

2.2.5 PUG

PUG merupakan mesin *templating* berkinerja tinggi yang diimplementasikan dengan JavaScript untuk Node.js dan browser. Dengan *mesin templating* ini *developer* dapat membuat markup HTML dengan kode yang minimalis. Selain itu, PUG dapat memasukkan data ke dalam *template* HTML dan menghasilkan markup HTML pada browser di sisi klien.

2.2.6 Semantic-UI

Semantic-UI merupakan *framework* untuk pengembangan *front-end* modern yang didukung oleh LESS dan jQuery. Semantic-UI memungkinkan *developer* untuk membangun situs web dengan cepat dan menggunakan HTML

yang singkat, javascript yang intuitif, dan *debugging* sederhana, yang membantu *front-end developer* memiliki pengalaman yang menyenangkan dalam sebuah pengembangan web. Semantic-UI dirancang secara *responsive* yang memungkinkan situs web dapat mengukur layar pada beberapa perangkat. Semantic-UI dapat diintegrasikan dengan *framework* seperti React, Angular, Meteor, dan Ember, yang berarti *framework* ini mengatur lapisan *user interface* di samping logika aplikasi. Semantic-UI menggunakan konsep *grid* untuk menyusun tata letak *layout* dengan jumlah 16 *grid* serta lebih dari 50 elemen untuk *user interface*.