

2.3.1 Python

Dengan fitur *dynamic-typing*, sintaks yang singkat serta ekspresif, Python merupakan bahasa pemrograman populer yang telah digunakan pada berbagai sistem untuk mendukung segala bentuk penelitian dalam bidang ilmu pengetahuan. Bahasa ini dapat dipakai untuk berbagai kebutuhan dengan skala yang bervariasi, mulai dari *script* kecil, sampai ke aplikasi berskala besar seperti sebuah sistem ataupun *library*. (Blomqvist, Dulak, Friis, & Hargus, 2017).

Sebagai bahasa pemrograman yang bersifat *interpreted*, Python memudahkan proses diagnosa variabel pada saat *runtime* dengan cepat. (Short, Bayer, & Burns, 2018). Sistem operasi *Windows*, *Linux* dan *Mac OS* mendukung pengembangan dan *runtime* aplikasi berbasis Python (Zhang, Moynihan, Ernest, & Gutenson, 2017).

Dengan koleksi *standard library* yang sangat besar, Python memiliki dukungan *library* untuk berbagai *task* yang bervariasi, termasuk RegEx (Regular Expression), *unit testing*, *database*, *multithreading*, CGI (Common Gateway Interface), email, kriptografi dan manipulasi gambar (Guan, Zhou, & Zhou, 2019).

2.3.2 Pillow

Pillow merupakan *library* dari bahasa pemrograman Python yang mendukung pemrosesan, manipulasi dan pembuatan gambar digital dalam banyak format. Pillow dikembangkan sebagai *fork* dari proyek PIL (Python Imaging Library), di mana Pillow menyediakan dukungan untuk versi Python 3.x (Nair, Athul, & Kartha, 2018).

2.3.3 HTML, CSS dan JavaScript

Teknologi web tidak akan ada tanpa HTML (HyperText Markup Language), CSS (Cascading Style Sheets) dan JavaScript. Ketiga bahasa tersebut digunakan dalam membuat, mendesain dan mempresentasikan sebuah website. HTML adalah bahasa *scripting* yang digunakan untuk mendefinisikan struktur dan konten dari sebuah website dengan menggunakan ‘tag’, ‘attribute’ dan ‘element’. CSS merupakan bahasa yang memodifikasi tampilan dari konten dan struktur web yang telah didefinisikan oleh HTML. Dengan CSS, website dapat didesain dengan

spesifikasi font, warna, posisi, margin dan berbagai atribut lain pada setiap elemen HTML yang ada. Sedangkan JavaScript adalah bahasa *scripting* pada sisi *client* yang menyediakan interaktivitas dan dinamisasi konten dari website (S. A. Garba, 2017).

2.3.4 Electron Framework

Electron adalah *framework open-source*, yang dikembangkan dan di-*maintain* oleh Github. Electron dikembangkan dengan tujuan agar *developer* dapat mengembangkan aplikasi desktop GUI *cross-platform* dengan menggunakan bahasa pemrograman web yaitu HTML, CSS dan JavaScript. Aplikasi yang dikembangkan oleh Electron dapat berjalan pada sistem Windows, Linux dan MacOS (Kredpattanukul & Limpiyakorn, 2019).

2.3.5 Vue.js Framework

Vue.js merupakan *framework front-end* JavaScript yang berbasis model MVVM (Model-View-ViewModel). Vue.js menyediakan implementasi data-binding dua arah di antara data dan DOM. Fitur *components* dari Vue.js menyediakan abstraksi setiap elemen tampilan menjadi sebuah *component* yang dapat dipakai berulang-kali, sehingga mengurangi repetisi penulisan kode untuk mendefinisikan tampilan yang sama (Song, Zhang, & Xie, 2019).

2.3.6 Webpack

Webpack adalah *module bundler*, di mana secara konsep berfungsi mengumpulkan semua *module* dari *source code* yang ada menjadi sebuah file statis. Jenis *source code* dan hasil output file statis yang diperlukan ditentukan oleh empat konfigurasi di Webpack yaitu *entry*, *loader*, *plugin*, dan *output*. *Entry* merupakan titik awal, yang mencakup *source code* serta *dependency* yang digunakan dalam aplikasi JavaScript tersebut. *Loader* merupakan konfigurasi di mana kita dapat menambahkan *static asset* seperti CSS, HTML ataupun gambar. Konfigurasi *Plugin* menyediakan tambahan fungsionalitas kepada Webpack. *Output* meng-*compile* semua yang telah dikonfigurasi di *Entry*, *Output* dan *Loader* menjadi sebuah *bundle* berupa satu file JavaScript (Hoque, 2017).

2.3.7 ZeroRPC

ZeroRPC merupakan sebuah *library* yang dikembangkan dari *framework* komunikasi ZeroMQ. Tersedia untuk bahasa pemrograman Python dan Node.JS, ZeroRPC menyediakan protokol komunikasi antar aplikasi dengan menggunakan *interface* yang berupa *socket*. Komunikasi berbasis protokol TCP ini dapat menghubungkan dua aplikasi baik dalam komputer yang sama, ataupun komputer yang berbeda (Gaspari, 2016). Informasi yang dikirim menggunakan ZeroRPC harus dapat diserialisasikan berdasarkan format JSON (Stipe, Eugen, & Zeljko, 2017).