

DAFTAR TABEL

| | |
|---|----|
| Tabel 3.1 Tabel Labelling..... | 11 |
| Tabel 4.1 Pengujian <i>Input Keyword</i> | 15 |
| Tabel 4.1 Pengujian <i>Input Repeat</i> | 15 |
| Tabel 4.1 Pengujian Menampilkan Data..... | 16 |
| Tabel 4.1 Pengujian Halaman Utama..... | 17 |
| Tabel 4.1 Pengujian Halaman Iterasi..... | 17 |
| Tabel 4.1 Pengujian Halaman Hasil..... | 18 |
| Tabel 4.1 Tabel Nama <i>Metadata</i> | 19 |
| Tabel 4.1 Pengujian Dengan <i>Keyword Gundam</i> | 20 |
| Tabel 4.1 Pengujian Dengan <i>Keyword Bandai</i> | 21 |
| Tabel 4.1 Hasil pengujian dengan 27 <i>metadata</i> berbeda..... | 22 |

BAB I PENDAHULUAN

1.1. Latar Belakang Masalah

Dengan berkembangnya jumlah informasi yang terdapat di *web* dan jumlah pengguna internet bertambah secara cepat, muncul juga permintaan untuk mengakses informasi tersebut dengan cepat dan mudah. Pengembangan alat yang memudahkan hal tersebut bekerja pada bidang *web mining*. *Web mining* dapat diartikan sebagai penemuan dan analisa informasi berguna dari *World Wide Web*. [1]

Dengan munculnya *search engine* telah memudahkan pengguna untuk mengakses informasi yang ada di *web*. Dan sekarang *search engine* telah menjadi gerbang untuk pengaksesan informasi yang berasal dari *web*. Oleh karena itu dibutuhkannya desain *user interface* pada *search engine* untuk memudahkan pengguna dalam mengakses informasi terlebih dalam lagi.

Metode desain untuk *user interface* berbasis web sangat berbeda dari *Graphical User Interface* yang sudah ada. *GUI* memiliki metode desain yang sudah lebih maju dan telah terbukti menghasilkan *interface* yang sangat bermanfaat. [2]

User Interface adalah bagian visual dari website yang memastikan bagaimana seorang user berinteraksi dengan website tersebut serta bagaimana informasi ditampilkan di layarnya. *User interface* sendiri menggabungkan konsep desain visual, dan infrastruktur informasi dengan tujuan untuk meningkatkan *usability* dan *user experience*.

Dengan adanya *user interface* yang baik maka akan memudahkan *user* untuk berinteraksi dan menavigasi pada website tersebut. Peningkatan *traffic* website juga akan terpengaruh dari *user interface* yang baik, jika *user* senang dengan tampilan website maka ada kemungkinan besar *user* akan mengunjungi website kembali.

Dan juga dengan berkembangnya perangkat seluler, *user* dapat berinteraksi dengan website melalui perangkat yang berbeda – beda. Karakteristik perangkat seluler dan platform desktop sangat berbeda. Oleh karena itu, *user interface* untuk

sebuah website harus bisa fleksibel dan adaptif untuk mendukung *user experience* yang baik. [3]

1.2. Rumusan Masalah

1. Bagaimana desain dan implementasi *website* yang baik untuk *Google image scraping engine* ?
2. Bagaimana implementasi *API* yang baik untuk menampilkan daftar data pada *website* ?

1.3. Tujuan dan Manfaat

- Terbentuknya website dengan *user interface* yang baik untuk *Google image scraping engine*.
- Daftar data dapat diakses oleh pengguna melalui *website* tanpa ada kehilangan data.

1.4. Batasan Masalah

1. *User Interface* yang dibentuk berbasis web.
2. Bahasa pemrograman utama yang digunakan adalah Python.
3. Website dibentuk menggunakan *Django Web Framework*.
4. *Database* yang digunakan bersifat lokal.

1.5. Metodologi Penelitian

Metodologi penelitian yang digunakan pada Tugas Akhir ini adalah:

1. Studi Literatur

Studi literatur bertujuan untuk mendapatkan informasi mengenai identifikasi, pengumpulan data, pemilihan metode atau gambaran mengenai konsep dan teori mengenai apa yang telah dikerjakan dan bagaimana pekerjaan tersebut dilakukan, kemudian akan dibandingkan dengan

penelitian yang akan dilakukan. Studi literatur akan diperoleh dari berbagai sumber diantaranya buku, jurnal dan dokumentasi proyek sebagai bahan untuk pembuatan Tugas Akhir.

2. Pengumpulan Data

Pengumpulan data dilakukan untuk menganalisis kebutuhan dan pendapat pengguna terhadap sebuah permasalahan dan produk yang akan dibuat sebagai solusi dan analisis penelitian.

3. Analisa Kebutuhan

Analisis kebutuhan dilakukan untuk mengetahui kebutuhan apa saja yang akan mendukung proses perancangan website pada software requirement spesification.

4. Pengujian Aplikasi

Website yang dibuat kemudian di uji dengan pengujian white box

5. Diskusi Ilmiah

Proses diskusi ilmiah dilakukan dengan dosen Pembimbing untuk memperbaiki kesalahan yang didapat pada website yang telah dibuat.

6. Pembuatan Laporan

Pada proses ini dilakukan pembuatan proses dokumentasi laporan Tugas Akhir yang akan digunakan untuk sidang Tugas Akhir.

1.6. Sistematika Penulisan

Penulisan tugas akhir ini dibagi dalam beberapa bagian. Tiap-tiap bagian menjelaskan langkah demi langkah dalam pengerjaan tugas akhir ini. Berikut adalah bagian tersebut:

BAB I PENDAHULUAN

Pada bab ini menjelaskan latar belakang penelitian, rumusan masalah tujuan, dan batasan masalah, dan metodologi penelitian yang dilakukan.

BAB II TINJAUAN PUSTAKA

Bab ini berisi tentang tinjauan teori dan sumber-sumber terkait yang digunakan dalam penelitian yang dilakukan.

BAB III ANALISIS DAN PERANCANGAN

Bab ini menjelaskan tentang proses analisis pada sistem serta desain dan perancangan aplikasi yang akan dibuat pada tugas akhir ini. Perancangan sistem digambarkan dengan menggunakan use case diagram, sequence diagram, dan perancangan mock up.

BAB IV IMPLEMENTASI DAN PENGUJIAN

Bab ini membahas proses pengujian yang dilakukan pada aplikasi, dan hasil pengujian dianalisis agar dapat dilakukannya penarikan kesimpulan dalam penelitian ini.

BAB V KESIMPULAN DAN SARAN

Bab ini membahas kesimpulan dari hasil pengujian aplikasi yang dibuat dan saran yang membuat hasil penelitian lebih baik untuk kedepannya.

BAB II

TINJAUAN PUSTAKA

2.1. User Interface

User Interface pada bidang *Human-Computer Interaction* (HCI), adalah ruang dimana terjadi interaksi antara manusia dan mesin. *UI* terdiri dari lebih dari satu lapisan termasuk juga *Human-Machine Interface* (HMI) seperti monitor computer, keyboard, mouse, dan juga printer. Tujuan dari *Human-Computer Interaction* adalah untuk memenuhi kebutuhan user sebanyak mungkin. [4]

2.2. Django Web Framework

Django adalah *Web Framework* yang gratis dan *open source*. Tujuan utama dari Django adalah untuk memudahkan pembuatan dari website yang rumit dan berbasis database. Django mengikuti prinsip “*Don’t Repeat Yourself*” yaitu mengurangi pengulangan pola perangkat lunak, menggantinya dengan abstraksi atau menggunakan normalisasi data untuk menghindari redundansi. [5] Terlebih lagi, Django memiliki banyak komunitas *developer* dan komponen bawaan yang baik. Django juga bisa mengakses atau membuat contoh data JSON/XML and berurusan dengan sistem pengaturan database seperti Oracle, MySQL, SQLite, and PostgreSQL. [6]

2.3. Python

Python adalah bahasa pemrograman tingkat tinggi dan bersifat general. Python juga bahasa Pemrograman Berorientasi Objek yang dapat membantu memecahkan masalah yang kompleks dengan cara membagi masalah tersebut menjadi set yang lebih kecil dengan adanya objek. Python mendukung hampir semua database umum seperti MySQL, Oracle, SQLite, dan lain – lain. [7]

2.4. Web Scraper

Web Scraping adalah teknik untuk mengambil informasi tertentu dari web. [8] Cara kerja *web scraping* adalah mengambil data yang tidak terstruktur lalu mengubahnya menjadi terstruktur dan menyimpannya di database untuk di analisa kembali. *Web Scraping* biasanya menggunakan bahasa pemrograman python karena telah tersedia *library* untuk melakukan *web scrapping* seperti selenium. Secara mendasar, *web scraper* dapat dilihat sebagai sebuah robot yang meniru manusia berinteraksi dengan website dan mengambil data yang tersimpan didalamnya. [9]

2.5. Google Images

Google Images adalah jasa pencarian dari *Google* yang dapat digunakan pengguna untuk melakukan pencarian di *web* untuk mencari gambar. *Google* mencari gambar dengan menggunakan berbagai algoritma untuk melakukan pencarian dan untuk mendapatkan hasil. [10] Pada dasarnya langkah – langkah *Google Images* mencari gambar adalah, pertama gambar akan dianalisa fiturnya seperti warna, titik – titik, garis, dan tekstur lalu dari fitur – fitur tersebut dilakukan pencocokan dengan miliaran gambar yang ada di internet. Algoritma pencarian mencari gambar yang sesuai dengan permintaan.

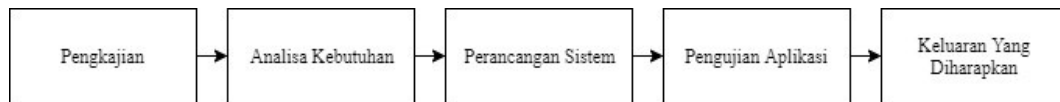
2.6. Application Programming Interface

API (Application Programming Interface) adalah sebuah antarmuka antara kedua bagian berbeda dari sebuah program komputer yang bertujuan untuk menyederhanakan implementasi dan perawatan perangkat lunak. Antarmuka program ini membantu meningkatkan produktifitas dan konektivitas dengan cara memungkinkan developer perangkat lunak untuk menggunakan ulang kode melalui *libraries*, dan aplikasi untuk mengintegrasikan satu sama lain untuk membagi data dan sumber daya.[11]

BAB III

ANALISIS DAN PERANCANGAN

3.1 Tahap Pengerjaan



Gambar 3.1 Tahapan Pengerjaan

3.1.1 Pengkajian

a. Studi Literatur

Studi Literatur adalah tahap yang bertujuan untuk mengumpulkan informasi mengenai identifikasi, pengumpulan data, pemilihan metode atau gambaran mengenai konsep dan teori mengenai apa yang sudah dikerjakan dan bagaimana caranya pekerjaan tersebut dilakukan. Studi yang dilakukan termasuk mencari, mempelajari dan menerapkan teori dan konsep tentang metode pemrograman *web*, *database*, *user interface*, dan *Django Web Framework* melalui artikel, buku, jurnal, dan juga internet. Hasil yang diperoleh akan menjadi bahan dan acuan untuk dasar teori tugas akhir ini.

b. Pengumpulan Data

Pengumpulan data dilakukan untuk menganalisis program sebagai solusi dari permasalahan yang akan dibuat. Pengumpulan data dilakukan dengan cara mengambil data dari *database* melalui website dan juga langsung dari *database*. Untuk mencapai kesimpulan analisis yang akurat, dibutuhkan 30 jenis *metadata* dari semua data yang ada di *database*.

3.1.2 Analisa Kebutuhan

a. Ruang Lingkup

Segala hal yang terdapat pada laporan tugas akhir ini merupakan bagian dari ruang lingkup kebutuhan pembangunan perangkat lunak berupa *website*

yang digunakan untuk memfasilitasi *Google Images Scraping Engine* sehingga dapat menghasilkan informasi yang merupakan hasil dari *Image Scraping* tersebut.

b. Tujuan Pembuatan Aplikasi

Hasil dari penelitian ini adalah sebuah Sistem Informasi Data yang memberikan informasi kepada pengguna untuk mendapatkan bukan hanya gambar tetapi juga informasi *metadata* dari gambar tersebut. Perancangan umum sistem dibangun seperti pada gambar 3.2.



Gambar 3.2 Gambaran Umum Sistem

A. Fungsi Aplikasi

Sistem yang telah dibentuk ini diimplementasikan untuk membuat sebuah sistem informasi data. Sistem informasi data ini akan digunakan untuk membantu pengguna yang ingin menggunakan *Google Images Scraping Engine* dengan mudah. Sistem Informasi Data ini memiliki beberapa fitur, yaitu:

1. Melihat besar kapasitas *storage* yang dibutuhkan untuk melakukan proses.
2. Mengatur berapa banyak kategori yang dipilih untuk diproses.
3. Mencari gambar berdasarkan *metadata*.

B. Karakteristik Pengguna

Sistem ini hanya memiliki satu jenis pengguna, yaitu user saja. User hanya dapat hak untuk melihat data dan mengunduh data.

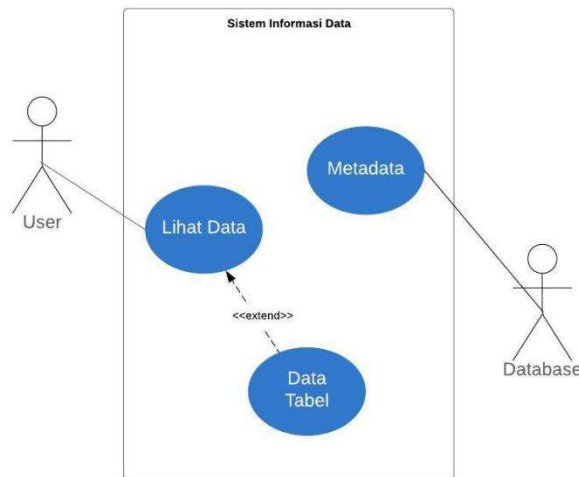
1. User

- Hak pengguna: User hanya memiliki hak untuk melihat data yang disajikan oleh sistem, data yang disajikan berupa tabel.
- Level Pendidikan: Memiliki dan mengerti penggunaan internet dan *web browser* pada komputer maupun *smartphone*.

3.2 Perancangan Sistem

3.2.1 Perancangan UML

a. Use Case Diagram

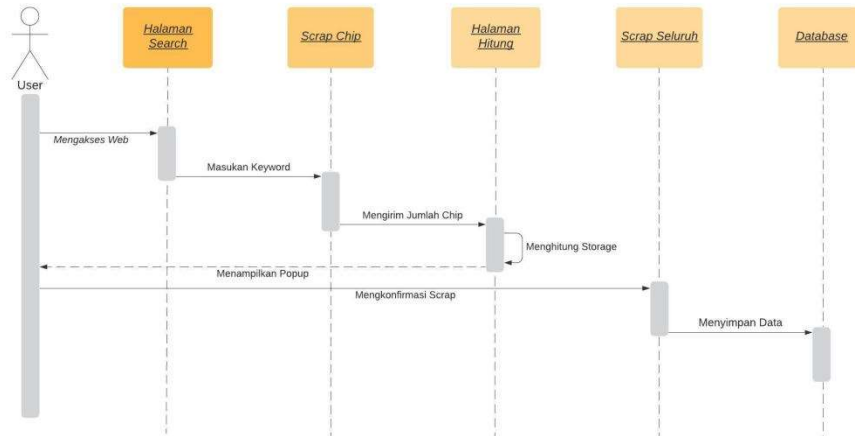


Gambar 3.3 Use Case Diagram

Website yang akan dibuat memiliki satu jenis pengguna, yaitu *user*. *User* hanya dapat mengakses data untuk melihat datanya saja. Data yang disajikan merupakan sebuah data tabel.

b. *Sequence Diagram*

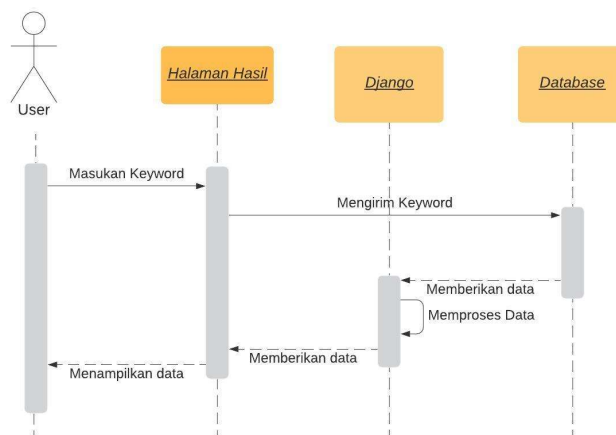
1) *Scraping Data*



Gambar 3.4 *Sequence Diagram Scraping Data*

Gambar 3.4 adalah *sequence diagram* untuk fitur *scraping* data. Pada diagram diatas dapat dilihat respon yang diberikan entitas masing-masing pada saat fitur ini berinteraksi. Terlihat dari diagram di atas proses pada saat *scraping* data dijalankan, mulai dari *input* hingga masuknya data di *database*.

2) *Menampilkan Data*



Gambar 3.5 *Sequence Diagram Menampilkan Data*

Gambar 3.5 adalah *sequence diagram* untuk fitur *scraping* data. Pada diagram diatas dapat dilihat respon yang diberikan entitas masing-masing pada saat fitur ini berinteraksi. Pada proses penampilan data pengguna pertama memasukan keyword yang akan dikirim ke *Database* yang nantinya akan mengirim data untuk diproses dalam bentuk JSON, dan akhirnya akan ditampilkan pada halaman hasil.

3.2.2 Pemodelan Data

a. Kamus Data

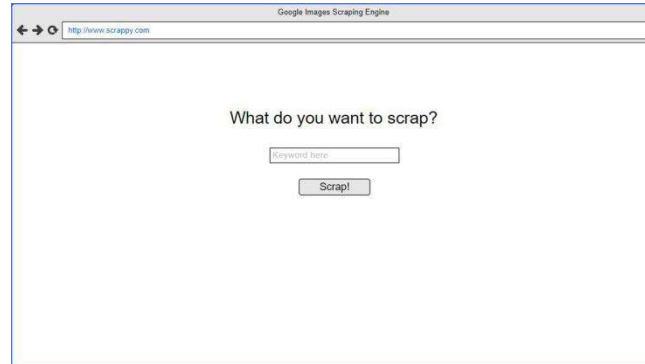
1. Tabel Labelling

Tabel labelling memiliki 4 kolom, yaitu *index*, *MetadataName*, *Metadata* dan URL. Kolom *index* mengacu pada *index* didalam file csv sebelum disimpan kedalam *database*. Kolom *MetadataName* adalah nama dari *metadata* yang terdapat pada gambar seperti *Orientation* dan *ColorSpace*. Kolom *metadata* mengacu pada nilai dari *metadata* itu sendiri. Kolom URL mengacu pada URL dimana gambar tersebut didapatkan. Dari tabel berikut user dapat mencari URL berdasarkan *metadata* yang diinginkan.

Table 3.1 Tabel Labelling

| Column | Type | Comment |
|--------------|--------------|---------|
| Index | varchar(999) | |
| MetadataName | varchar(999) | |
| Metadata | LONGTEXT | |
| URL | varchar(999) | |

3.2.3 Perancangan Antarmuka



Gambar 3.6 *Mockup* Halaman Awal *Website*

Pada saat pengguna pertama kali berkunjung pada halaman *website*, pengguna akan disajikan dengan tampilan seperti pada gambar 3.6. Pengguna dapat melanjutkan proses dengan cara memasukan kata kunci dari gambar yang ingin di *scrap*.



Gambar 3.7 *Mockup* Halaman Hitung *Website*

Halaman tampilan berikutnya adalah merupakan pemilihan jumlah *repeat* atau jumlah kategori yang ingin dimasukan kedalam proses *scraping*. Pengguna dapat mengatur jumlah *repeat* tersebut.