

# BAB I

## PENDAHULUAN

### 1.1. Latar Belakang

Pengujian perangkat lunak (*software testing*) sangat diperlukan untuk memastikan *software/aplikasi* yang dibuat dapat berjalan sesuai dengan fungsionalitas yang diharapkan. Pengembang atau penguji perangkat lunak harus menyiapkan sesi khusus untuk menguji program yang sudah dibuat agar kesalahan ataupun kekurangan dapat dideteksi sejak awal dan dikoreksi secepatnya. Pengujian atau testing sendiri merupakan elemen kritis dari jaminan kualitas perangkat lunak dan merupakan bagian yang tidak terpisahkan dari siklus hidup pengembangan perangkat lunak seperti halnya analisis, desain, dan pengkodean. [4]

Pengujian perangkat lunak bukan pekerjaan yang mudah. Ada beberapa orang beranggapan bahwa kalau sekedar mengetes perangkat lunak saja semua orang juga bisa dan tidak perlu susah-susah menjadi *programmer/developer*. Tapi *software testing* merupakan disiplin tersendiri dan merupakan salah satu dari bagian proses pengembangan perangkat lunak. Hampir semua metodologi dari pengembangan perangkat lunak (*Software Development*) menempatkan *software testing* sebagai bagian utama dari proses pengembangan.

Kegagalan dalam melakukan pengujian perangkat lunak dapat menyebabkan produk yang dihasilkan tidak berjalan dengan baik dan dapat menimbulkan

kerugian ekonomi yang besar. Sebuah studi yang dilakukan oleh National Institute of Standart and Technologi (NIST) pada tahun 2002 melaporkan bahwa bugs pada perangkat lunak menyebabkan kerugian ekonomi Amerika Serikat sebesar \$59.5 billion tiap tahunnya, dan sepertiga dari kerugian ini bisa dihindari jika dilakukan software testing yang lebih baik.[5]

Tujuan utama dari pengujian perangkat lunak sebenarnya sederhana yaitu untuk memastikan bahwa *software* yang dihasilkan sesuai dengan kebutuhan (*requirement*) yang sebelumnya ditentukan. Ketika *requirement* dari suatu sistem telah disusun maka semestinya sudah ada suatu pengujian perencanaan (*test plan*). Selain itu suatu proses testing membutuhkan tujuan akhir yang dapat dinilai sehingga pihak *tester* bisa berhenti melakukan suatu testing ketika tujuan-tujuan itu tercapai. Pengujian pada perangkat lunak biasanya berarti pengujian yang dilakukan secara otomatis (*automated testing*). Maksudnya adalah pengujian dilakukan menggunakan *software* untuk merekam kegiatan *user* dan mensimulasikan penggunaan komponen internal.

Dalam tahapan akhir pengembangan dan pembuatan suatu produk biasanya ada yang dinamakan tahapan pengetesan (*test ride* misalnya dalam produk *launching* sepeda motor) yaitu proses pengetesan produk atau produk hasil pengembangan untuk pengendalian kualitas untuk memastikan bahwa produk memenuhi atau melampaui persyaratan sesuai kebutuhan (*requirement*). Begitu juga dalam tahap pembuatan dan pengembangan perangkat lunak ada yang namanya tahapan pengetesan yaitu pengujian perangkat lunak (*software testing*).

Dalam pembuatan perangkat lunak tentunya kita sering menghadapi "*bug*" atau kesalahan "*error*" pada proses-proses tertentu. Untuk menghindari banyaknya *bug* maka diperlukan pengujian perangkat lunak, sebelum perangkat lunak yang telah kita buat diberikan ke pelanggan atau selama perangkat lunak masih terus dikembangkan. Pada dasarnya pentingnya pengujian perangkat lunak mengacu pada kualitas perangkat lunak tersebut.

Secara tradisional terdapat dua pendekatan pokok untuk menguji perangkat lunak yaitu *White Box Testing* dan *Black Box Testing*. *White Box Testing* menitik beratkan pada teknik verifikasi yang menggunakan *source code* sebagai pedoman penyeleksian terhadap pengujian data sedangkan *Black Box Testing* menitik beratkan pada teknik validasi yang melakukan pengecekan untuk melihat apakah produk memenuhi kebutuhan *user* (kebutuhan pelanggan). *White Box Testing* memiliki beberapa metode tersendiri dalam pengujian perangkat lunak, begitu juga dengan *Black Box Testing*. Dalam penelitian ini penulis akan mengimplementasikan pengujian perangkat lunak dengan pendekatan *black box testing* yaitu metode *cause effect graphing*.

Metode *cause effect graphing* adalah teknik pengujian perangkat lunak yang membantu dalam memilih kasus uji yang secara logis menghubungkan *cause* (masukan) dengan *effect* (keluaran) untuk menghasilkan kasus uji (*test case*). Pada beberapa hasil penelitian, metode *cause effect graphing* menghasilkan jumlah kasus uji (*test case*) yang lebih sedikit dibandingkan menggunakan metode lain. Selain dapat mereduksi jumlah kasus uji (*test case*) yang harus di jalankan, metode *cause effect graphing* juga menghasilkan kasus uji yang dapat mencakup 100% kebutuhan

fungsional dalam pengembangan perangkat lunak berdasarkan hasil pengujian yang dilakukan.

Pada penelitian ini penulis akan melakukan pengujian perangkat lunak terhadap sebuah aplikasi bernama Simdes yang di bangun dan dikembangkan di wilayah tempat tinggal penulis yaitu Desa Gunungmanik. Aplikasi ini merupakan aplikasi berbasis web yang dibangun dengan bahasa pemrograman PHP: *Hypertext Preprocessor* (PHP). Aplikasi ini mendukung pengguna untuk mendapatkan informasi mengenai Desa Gunungmanik, kegiatan masyarakat desanya, pembuatan surat untuk urusan kependudukan, akses informasi dan lain sebagainya yang berhubungan dengan Desa Gunungmanik. Aktor yang menggunakan aplikasi ini adalah administrator aplikasi yang dibagi menjadi beberapa tipe administrator dengan hak akses yang berbeda sesuai perannya dalam mengelola aplikasi di Kantor Desa dan masyarakat sebagai *end-user*.

Aplikasi simdes merupakan aplikasi yang mengintegrasikan pengelolaan sistem informasi di pemerintahan Desa Gunungmanik. Pada saat penggunaan aplikasi simdes, masyarakat sering mengalami terjadinya kesalahan cetak data kependudukan pada saat akan mengurus surat-surat untuk administrasi kependudukan maupun untuk keperluan lainnya, sehingga data tidak sesuai dengan data baru yang diharuskan sebagai persyaratan tertentu sehingga masyarakat pun menjadi terkendala dalam pengurusan surat-surat untuk keperluan tertentu atau pun hal yang berkaitan dengan administrasi. Jika terjadi perubahan data kependudukan misalnya, masyarakat harus datang ke kantor desa untuk memperbaikinya sehingga cara ini tidak efisien untuk masyarakat yang memiliki kesibukan atau pun yang

tinggalnya jauh dari kawasan kantor desa. Selain itu kadang pengguna mengalami *error* dan gagal akses pada fitur-fitur tertentu pada web aplikasi, sehingga banyak komplain dari masyarakat kepada admin aplikasi di kantor desa. Selain itu masyarakat juga mengeluhkan bahwa web aplikasi hanya berjalan secara sepihak saja dimana hanya aparat desa saja yang dapat melakukan pengelolaan web tanpa ada nya interaksi dan *feedback* dari masyarakat, sehingga masyarakat sulit menginformasikan kepada aparat desa bila ada kepentingan yang mendesak yang bersifat umum di masyarakat misalnya agenda kegiatan warga atau hal lainnya yang membutuhkan media penyampaian yang cepat dan meluas seperti web. Di sisi lain pengguna, yaitu admin web di kantor desa terkadang menginformasikan berita atau pengumuman tanpa mengecek kebenarannya atau tanpa adanya validasi dari pihak terkait, sehingga berita atau pengumuman yang tersampaikan ke masyarakat terkadang belum tentu kebenarannya, sehingga terjadi kesalahan penyampaian informasi.

Untuk menemukan kesalahan pada aplikasi serta memenuhi kebutuhan pengguna maka perlu dilakukan pengujian perangkat lunak pada aplikasi Simdes, sehingga komplain dari *user* dapat diakomodasi dan aplikasi sesuai dengan spesifikasi kebutuhan pengguna. Target pada pengujian ini memastikan fitur dan *behavior* yang disediakan oleh aplikasi sesuai dengan spesifikasi kebutuhan, seperti menguji validasi pada *form* pengguna, konsistensi di *database* ketika melakukan *edit*, *delete*, *create*, *update* dan kegiatan lainnya yang berhubungan dengan *database*. Pada Teknik uji *Black Box* selain dapat menjalankan *Function Test* juga

dapat memastikan *output* sesuai dengan *requirement* tanpa mengetahui struktur kode internal pada aplikasi.

## **1.2. Identifikasi Dan Rumusan Masalah**

Bertitik tolak dari latar belakang masalah yang ada, maka penulis mencoba mengidentifikasi dan merumuskan masalah-masalah berdasarkan hasil observasi dan pengamatan penulis.

### **1.2.1. Identifikasi Masalah**

Permasalahan yang terjadi berdasarkan hasil observasi dan pengamatan penulis adalah sebagai berikut :

1. Masyarakat sering mengalami terjadinya kesalahan cetak data kependudukan.
2. Masyarakat harus datang ke kantor desa untuk memperbaharui data kependudukan sehingga cara ini tidak efisien.
3. Pada saat menggunakan aplikasi, beberapa fitur tertentu kadang tidak dapat diakses dan juga mengalami *error*.
4. Tidak adanya validasi dari pihak terkait dalam penyampain informasi di web desa oleh admin web, sehingga terkadang terjadi *miss*-informasi.
5. Belum adanya dokumentasi hasil pengujian perangkat lunak pada aplikasi simdes, sehingga sulit untuk menentukan spesifikasi pengujian yang akan dilakukan.

### 1.2.2. Rumusan Masalah

Setelah mengetahui masalah-masalah yang ada pada identifikasi di atas, maka penulis dapat merumuskan permasalahan sebagai berikut :

1. Bagaimana mengimplementasikan pengujian perangkat lunak dengan pendekatan Black Testing untuk menjalankan *function test* pada aplikasi simdes?
2. Bagaimana mengimplementasikan metode *cause effect graphing* pada pengujian perangkat lunak pada aplikasi simdes?
3. Bagaimana membuat dan menjalankan test case pada pengujian perangkat lunak pada aplikasi simdes?
4. Bagaimana implementasi *automation testing* menggunakan Katalon Studio pada aplikasi simdes?

### 1.3. Maksud Dan Tujuan Penelitian

Maksud dan tujuan penelitian dapat diuraikan sebagai berikut :

#### 1.3.1. Maksud Penelitian

Maksud dari diadakannya penelitian ini adalah untuk mengimplementasikan pengujian perangkat lunak baik secara manual dengan pendekatan teknik *black box testing* maupun secara otomatis dengan menggunakan *automation testing tool*.

### 1.3.2. Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah :

1. Untuk mengimplementasikan metode *cause effect graphing* pada pengujian perangkat lunak pada aplikasi simdes.
2. Untuk mengimplementasikan *functional test* dengan pendekatan teknik *black box testing* pada aplikasi simdes.
3. Untuk mengimplementasikan salah satu *automation testing tool* yaitu Katalon Studio dalam sebuah pengujian perangkat lunak.
4. Untuk mengetahui kelebihan dan kekurangan salah satu metode pengujian perangkat lunak dibandingkan dengan metode lainnya.

### 1.4. Kegunaan Penelitian

#### 1.4.1. Kegunaan Praktis

Pengujian perangkat lunak merupakan proses mengoperasikan perangkat lunak dalam suatu kondisi yang di kendalikan, untuk verifikasi (mencari sistem yang baik), validasi (membangun sistem yang benar), cek *error* (mencari kesalahan pada sistem yang ditetapkan). Sehingga tujuan akhir dari melakukan pengujian perangkat lunak adalah mendapatkan informasi yang berkualitas. Kegunaan penelitian yang dilakukan oleh peneliti dari hasil penelitian ini adalah :

1. Melakukan cek pada program terhadap spesifikasi
2. Menemukan bug pada program.
3. Menentukan penerimaan dari pengguna.
4. Memastikan suatu sistem siap digunakan.

5. Meningkatkan kepercayaan terhadap kinerja program.
6. Memperlihatkan bahwa program berkerja dengan benar.
7. Membuktikan bahwa error tidak terjadi.

#### **1.4.2. Kegunaan Akademis**

Hasil dari penelitian ini diharapkan dapat digunakan untuk :

1. Bagi pengembangan ilmu pengetahuan, hasil penelitian ini dapat berguna bagi pengembangan ilmu pengetahuan dibidang teknologi informasi.
2. Bagi penelitian, sebagai literatur acuan yang berguna bagi pendidikan dan penelitian selanjutnya, terhadap permasalahan tentang pengujian perangkat lunak.
3. Bagi Peneliti, diharapkan berguna bagi peneliti sendiri khususnya dalam menambah dan memperkaya wawasan pengetahuan baik teori maupun praktek.

#### **1.5. Batasan Masalah**

Ada beberapa batasan masalah dalam penelitian ini agar pembahasan lebih terfokus sesuai dengan tujuan yang akan dicapai. Batasan masalah ini adalah sebagai berikut :

1. Dalam penelitian ini penulis hanya berfokus pada pengujian perangkat lunak terhadap aplikasi simdes yang telah dibangun dan dikembangkan sebelumnya.

2. Teknik pengujian perangkat lunak yang digunakan penulis dalam penelitian ini adalah teknik uji *black box* dengan metode *cause effect graphing*.
3. *Automation testing tool* yang penulis gunakan adalah *Katalon Studio*
4. Pengujian perangkat lunak dilakukan pada aplikasi Simdes Gunungmanik.
5. Implementasi pengujian perangkat lunak hanya pada beberapa fitur yang terdapat pada aplikasi simdes yaitu fitur login administrator, login penduduk, cetak biodata penduduk, berita, tambah artikel, cetak surat dan profil penduduk.

#### 1.6. Lokasi dan Waktu Penelitian

Adapun lokasi penelitian adalah di wilayah tempat tinggal penulis tepatnya di kantor Desa Gunungmanik yang beralamatkan di Jl. Pamager Sari, Dusun Pasantren, Desa Gunungmanik, Kecamatan Tanjungsari, Kabupaten Sumedang. Untuk mempermudah kegiatan penelitian hingga pembuatan laporan, penulis membuat *time schedule* yaitu sebagai berikut :

**Tabel 1.1 Waktu Penelitian**

Keterangan	2019																
	April				Mei					Juni				Juli			
	1	2	3	4	1	2	3	4	5	1	2	3	4	1	2	3	4
Penyusunan proposal																	
Survey objek penelitian																	
Study literatur																	
Observasi																	
Planning testing																	
Software testing																	
Evaluasi & dokumentasi																	

### **1.7. Sistematika Penulisan**

Berdasarkan struktur penulisan yang dibuat, terdapat penjelasan mengenai isi tiap bagian bab yang ditulis.

#### **BAB I : PENDAHULUAN**

Bab ini membahas ruang lingkup permasalahan secara umum dan singkat yang terdiri dari latar belakang masalah, identifikasi dan rumusan masalah, maksud dan tujuan penelitian, kegunaan penelitian, batasan masalah, lokasi dan waktu penelitian serta sistematika penulisan.

#### **BAB II : LANDASAN TEORI**

Bab ini diuraikan mengenai teori-teori yang berkaitan dengan pengujian perangkat lunak. Teori-teori tersebut diperoleh dari buku-buku dan referensi lain.

#### **BAB III : OBJEK DAN METODE PENELITIAN**

Bab ini berisi mengenai objek penelitian, yang selanjutnya akan dilakukan pengujian perangkat lunak dengan beberapa tahapan meliputi perancangan strategi, taktik dan pemilihan metode yang akan dilaksanakan dalam pengujian perangkat lunak.

#### **BAB IV : HASIL DAN PEMBAHASAN**

Bab ini berisi pelaksanaan, laporan dan hasil analisis pengujian perangkat lunak.

#### **BAB V : KESIMPULAN DAN SARAN**

Bab ini menguraikan kesimpulan yang diperoleh setelah melakukan implementasi metode cause effect graphing pada pengujian perangkat lunak pada aplikasi yang telah penulis bangun.

Bagian akhir terdiri dari daftar pustaka dan lampiran-lampiran.