

Pengembangan Sistem Manajemen Guru Les Privat

Elsa Akhmal Rohmianti¹, Denny Sagita Rusdianto², Faizatul Amalia³

Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya
Email: ¹elsaakhmal01@gmail.com, ²denny.sagita@ub.ac.id, ³faiz_amalia@ub.ac.id

Abstrak

Privatto merupakan website yang digunakan untuk mencari dan memesan guru les privat. Privatto merupakan wadah yang digunakan guru untuk mempromosikan dirinya sebagai guru les privat. Privatto dibuat karena para murid merasa kesulitan memesan guru les karena beberapa faktor seperti tidak tersedianya guru les privat di lingkungannya, ketidakcocokan murid kepada guru karena hal tertentu dan murid tidak memiliki kebebasan dalam memilih guru les privat yang sesuai dengan kebutuhannya. Oleh karena itu dibuatlah sistem manajemen guru les privat yang dapat membantu murid dalam mencari guru les privat sesuai dengan keinginannya. Sistem ini dibangun dengan menggunakan metode *Waterfall*. Kebutuhan fungsional yang didapatkan sebanyak 60 kebutuhan dan kebutuhan non fungsionalnya yaitu *compatibility*. Hasil perancangan dan implementasinya dengan menggunakan pendekatan *object oriented*, dan menggunakan *framework Codeigniter* dan dengan metode *MVC (Model View Controller)*. Sistem diuji dengan pengujian unit, pengujian validasi dan pengujian *compatibility*. Pengujian unit menggunakan metode *white box* serta menggunakan teknik *basis path testing*. Pengujian validasi dengan menggunakan metode *black box* dan menghasilkan 77 kasus uji. Pengujian *compatibility* untuk mengetahui apakah sistem *compatible* digunakan di beberapa perangkat. Hasil pengujian yang didapatkan dengan pengujian validasi menghasilkan 100% valid dan pada pengujian *compatibility* yang telah dilakukan, sistem menghasilkan 90% *compatible* terhadap beberapa *browser*.

Kata kunci: *privatto, guru les privat, website.*

Abstract

Privatto is a website that user to search and ordering private tutors. Privatto is used by the tutor to promote theirselves. Privatto is made for student because students feel have a difficulty to order private tutor because any factors, such as unavailable private tutor in their area, unsuitable because some a reason and students don't have a freedom to choose what their want. Accordingly, private tutor manajemen sistem built to help student for ordering private tutor. Development of this system uses Software Development Life Cycle with Waterfall. There were 60 functional requirements obtained and non-functional requirement. This system is designed and is implementation by using an Object Oriented approach and uses the Codeigniter framework with MVC (Model, View, Controller) model. Sistem is tested by unit test, this research applies some tests covering white box method for the unit test and with basis path testing. Black box for validation test with 77 cases. Compatibility testing to find out how compatible system used in some equipments. The result is obtained by validation testing making 100% valid and compatibility testing that has been done, the system produces 90% compatible with some browsers.

Keywords: *privatto, private tutor, website.*

1. PENDAHULUAN

Les Privat merupakan bantuan yang diberikan oleh pembimbing kepada siswa yang mengalami kesulitan belajar (Sularso, Kokotiasa, & Paramedica, 2015). Les privat biasanya dilakukan oleh guru atau tentor atau

bisa juga dilakukan oleh suatu lembaga yang memberi jasa les. Les privat biasanya dilakukan kepada satu orang murid atau tidak lebih dari 4 orang, dengan ini metode pembelajaran yang dilakukan akan maksimal. Dalam kegiatan belajarnya les privat tetap menggunakan kurikulum dari sekolah sebagai acuan. Les privat biasanya dilakukan di rumah siswa ataupun bisa

dilakukan di rumah guru les. Dalam kegiatan les ini cocok dan bermanfaat untuk menekuni atau mendalami suatu bidang mata pelajaran tertentu.

Dalam kegiatan les privat tentunya harus memilih calon pengajar atau guru les yang sesuai kriteria dan keahlian pengajar. Dengan memilih guru les dengan keahlian yang tepat maka akan mendapatkan manfaat dan menguasai mata pelajaran yang dianggap sulit sebelumnya (Sularso, Kokotiasa, & Paramedica, 2015). Murid ataupun orang tua harus memeriksa kualitas pengajar dengan melihat sisi akademik dan dari segi pengalaman yang dimiliki serta dari sisi sikap dan tata krama saat mengajar. Peran orang tua dalam kegiatan belajar murid sangat mempunyai pengaruh yang besar dalam keberhasilan anaknya. Maka dari itu orang tua dapat melihat cara belajar anak dan mengontrol waktu belajar mereka.

Untuk menguasai bidang mata pelajaran tertentu atau yang diinginkan les privat sangat mempunyai manfaat yang besar bagi murid. Sedikitnya jumlah murid membuat belajar menjadi lebih intensif. Selain itu waktu yang fleksibel dalam melakukan kegiatan mengajar karena bisa dilakukan kapan saja dan dimana saja di luar jam sekolah. Guru les privat juga dapat membantu murid dalam menetapkan tujuan dan sasaran belajar. Selain itu cara penyampaian materi dalam les privat lebih rinci sehingga mata pelajaran yang ingin dikuasai lebih mudah untuk dipelajari. Guru les harus memiliki tanggung jawab dalam meningkatkan minat belajar murid secara optimal dengan cara seperti menentukan tujuan belajar yang jelas, dapat menyesuaikan kemampuan dengan minat bakat murid, dapat menciptakan suasana belajar yang menantang dan menyenangkan dan memberikan apresiasi kepada murid dengan tepat dan benar (Abidin, 2006).

Berdasarkan wawancara yang telah dilakukan dengan beberapa pelajar di kota Malang, dalam mencari guru les privat biasanya murid hanya mendapatkan info dari kerabat atau teman yang mengetahui info guru les ataupun mereka yang telah melakukan kegiatan les privat sebelumnya. Namun dalam hal ini belum tentu guru tersebut sesuai dengan apa yang diinginkan murid. Selain itu murid harus datang ke lembaga les privat untuk mendaftarkan diri untuk melakukan kegiatan les privat. Di dalam lembaga, murid mendaftarkan diri dengan mengisi formulir dan jadwal namun murid juga

tidak bisa memilih calon guru yang akan mengajar, pihak lembaga yang memilih guru yang akan mengajar murid tersebut. Sehingga memiliki potensi murid tersebut tidak cocok dengan guru pilihan dari lembaga. Serta terdapat pula lingkungan daerah tertentu yang tidak terdapat guru les privat dan membuat para murid kesusahan dalam mencari guru les privat. Sehingga dari uraian tersebut dapat diambil beberapa kesimpulan mengenai faktor-faktor yang memicu permasalahan dalam mencari guru les privat seperti kurangnya informasi mengenai guru les privat dan sempitnya ruang lingkup pencarian guru les privat yang menyebabkan pencari les atau murid tidak menemukan guru les yang sesuai potensi yang diinginkan.

Selain wawancara kepada murid juga telah melakukan wawancara ke salah satu guru les di suatu lembaga les privat di kota Malang. Di dalam lembaga tersebut guru untuk menjadi guru les privat di lembaga tersebut harus melakukan pendaftaran terlebih dahulu dan melakukan tes. Selanjutnya guru menunggu beberapa waktu untuk menerima jadwal bimbingan belajar. Pihak lembaga memilih guru sesuai bidang keahlian guru dan guru yang memiliki jadwal kosong. Pada setiap bimbingan atau kegiatan belajar mengajar dengan murid, guru harus memiliki presensi dan mencatat kehadiran pada setiap pertemuannya sampai satu bulan. Kemudian pada akhir bulan guru harus mendatangi lembaga untuk mengumpulkan presensi tersebut sebagai syarat untuk mendapatkan gaji. Namun apabila presensi tersebut tidak dikumpulkan dalam waktu yang ditentukan maka guru tersebut tidak mendapatkan gaji. Sehingga tidak menutup kemungkinan lembar presensi yang telah dikumpulkan tersebut hilang.

Berdasarkan permasalahan yang disampaikan, maka dibuat sebuah sistem berbasis *web* yang bisa mengatasi permasalahan dengan cara menyediakan fungsi untuk pencarian dan fitur untuk memesan guru. Sehingga murid dapat melakukan pencarian dengan mengisi form pencarian sesuai apa yang dibutuhkan. Berdasarkan hasil pencarian yang didapat murid juga dapat bebas memilih guru sesuai dengan pilihannya kemudian murid tersebut dapat melakukan pemesanan sesuai prosedur yang ada. Sedangkan guru dapat melihat permintaan belajar dari sistem tersebut, dan dapat menerima permintaan ataupun tidak. Guru juga dapat mengajukan permintaan gaji

secara praktis ke admin sistem yang juga melayani murid dan guru. Sehingga dari uraian diatas diangkat sebuah judul “PENGEMBANGAN SISTEM MANAJEMEN GURU LES PRIVAT ” dengan harapan sistem tersebut dapat membantu dan mempermudah murid dan orang tua dalam mencari guru les privat yang sesuai dengan keinginan. Selain itu sistem tersebut dapat membantu guru les dalam mempromosikan dirinya sebagai pengajar sehingga mendapatkan pekerjaan sebagai guru les yang sesuai dengan potensi yang dimilikinya. Dalam pembangunan sistem tersebut menggunakan *Software Development Life Cycle* yaitu dengan menggunakan metode *Waterfall* dan menggunakan framework *Codeigniter* dan *Bootstrap*. Pengujiannya dengan menggunakan pengujian unit, pengujian validasi dan pengujian *compatibility*. Pengujian unit dilakukan dengan metode *white box*, kemudian pengujian validasi digunakan metode yaitu metode *black box* dan pengujian *compatibility* dengan menggunakan *tool SortSite*.

2. LANDASAN KEPUSTAKAAN

2.1 Kajian Pustaka

Sebelumnya terdapat sebuah penelitian oleh Putra, A. K., Nyoto, R. D., & Pratiwi, H. S. (2017) dengan judul Rancang Bangun Aplikasi *Marketplace* Penyedia Jasa Les Privat Di Kota Pontianak Berbasis Web. Dalam penelitian tersebut bersumber dari beberapa masalah yaitu pencarian yang cukup lama dalam menemukan guru les privat, ketidakcocokan lokasi dan waktu antara guru les privat serta waktu serta tidak bisa mengetahui tentang potensi dan prestasi guru les tersebut. Sehingga tidak dapat menemukan guru les terbaik. Sistem yang dibangun menggunakan metode *waterfall*, dengan menggunakan bahasa pemrograman PHP, *Framework Laravel* dan *database MySQL*. Pada hasil penelitian ini dalam membangun sebuah sistem terdapat tahapan seperti analisis kebutuhan, perancangan, implementasi dan pengujian. Sebelumnya dengan menguraikan beberapa komponen yang menunjang *marketplace* tersebut. Sehingga nantinya akan dapat memudahkan masyarakat untuk menemukan guru les privat dengan kriteria yang diinginkan.

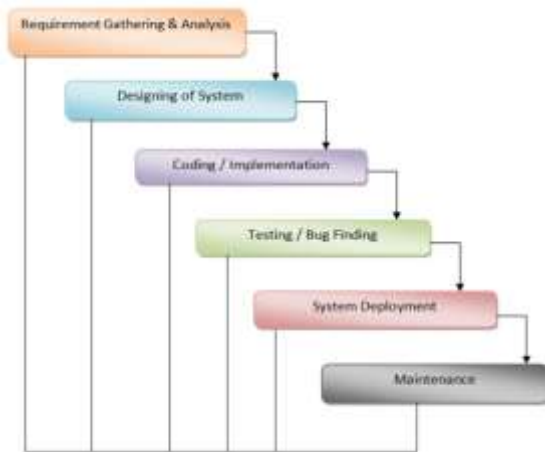
Penelitian sebelumnya juga telah dilakukan oleh Nike Essyana & Anita Qoiriah(2016)

dengan judul “Sistem Pencarian Guru Les Privat Berbasis Web”. Penelitian tersebut didasarkan pada masalah yaitu bergesernya peran orang tua sebagai pengasuh anak karena kesibukan orang tua karena rutinitas sehari-hari. Kebutuhan cara belajar anak yang berbeda-beda sehingga juga memerlukan pengajaran khusus seperti les privat. Namun dari masalah tersebut juga tidak diimbangi dengan kemudahan pencarian guru les privat. Sehingga dari penelitian tersebut menciptakan sebuah sistem pencarian guru les privat bernama ‘Go-Teacher’ yang dapat membantu dalam mencari guru les privat. Dalam penggunaannya menggunakan fitur dari Google Maps API untuk menampilkan guru di area murid. Serta dengan fitur *geocoding* dan *geolocation* dapat menghasilkan sistem yang dapat mengetahui lokasi guru di area rumah murid.

Berpacu dari penelitian sebelumnya, di kota lainnya belum terdapat suatu sistem yang digunakan dalam mencari dan memesan guru les privat. Maka, pengembangan suatu sistem untuk memesan dan mencari guru les privat sekiranya dibutuhkan dan dapat membantu masyarakat terutama para siswa. Sistem tersebut akan membantu para masyarakat membantu terutama siswa dalam mencari dan memesan guru les privat yang sesuai dengan potensi atau keahlian guru yang diinginkan. Dalam pengembangan aplikasi tersebut maka digunakan suatu metode yaitu metode *Waterfall*. Dalam mengembangkan perangkat lunak dengan memakai metode *Waterfall* bisa dikatakan cepat karena melibatkan pengguna dalam pengembangannya.

2.2 Waterfall

Dalam pengembangan perangkat lunak digunakan suatu metode yaitu metode *Waterfall*. Disebut dengan metode *waterfall* karena dalam pembuatan perangkat lunak dilakukan secara berurutan mulai dari tahap *requirements*, *analysis*, *design*, *coding* dan *testing*. Metode *Waterfall* juga disebut dengan *classic life cycle*, dimana metode ini disusun melalui pendekatan secara sistematis atau secara teratur (Pressman, 2010).



Gambar 1. Tahapan Metode Waterfall

a. *Requirements dan Analysis*

Pada tahap requirements dilakukan penggalian informasi mengenai kebutuhan pengguna, yaitu melakukan penggalian informasi seperti wawancara, observasi dan lain sebagainya. Tahap analisis digunakan untuk menjabarkan kebutuhan-kebutuhan ke dalam kebutuhan fungsional, kebutuhan non fungsional beserta mengidentifikasi aktor atau calon pengguna.

b. *Design*

Tahap design atau perancangan dibuat untuk memperjelas proses pembuatan perangkat lunak setelah tahap analisis. Pada tahap perancangan dibuat untuk merancang algoritma sistem, struktur data dan perancangan *user interface*.

c. *Coding*

Coding atau pengkodean merupakan tahapan pembuatan sistem yang sesungguhnya yang dibuat oleh seorang *programmer*. Pembuatan sistem ini didasarkan oleh tahap analisis dan perancangan yang telah dibuat sebelumnya.

d. *Testing*

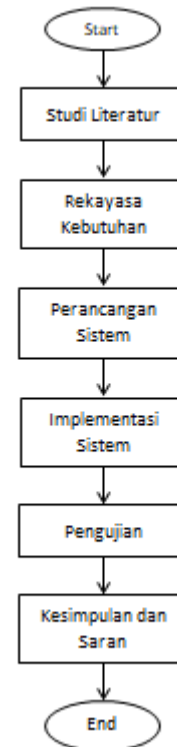
Pengujian sistem sangat penting dilakukan guna untuk mengukur sistem yang dibuat apakah sistem sesuai dengan kebutuhan atau tidak. Testing digunakan untuk mengukur tingkat keberhasilan sistem yang dibuat.

e. *Maintenance*

Merupakan tahap perawatan atau pemeliharaan sistem yang berguna agar sistem dapat berjalan dengan baik.

3. **METODOLOGI PENELITIAN**

Pada bab metodologi dijelaskan mengenai metodologi penelitian yang akan digunakan dalam “Pengembangan Sistem Manajemen Guru Les Privat”. Tahapan yang dilakukan yaitu Studi Literatur, Analisis Kebutuhan, Perancangan, Implementasi, Pengujian serta Pengambilan Kesimpulan dan Saran.



Gambar 2. Diagram Alir Metodologi Penelitian

4. **REKAYASA KEBUTUHAN**

4.1 **Elisitasi Kebutuhan**

Pada tahap ini telah dilakukan proses wawancara di salah satu bimbingan belajar atau les privat di Kota Malang. Wawancara tersebut dilakukan dengan salah satu Guru Les di lembaga tersebut. Wawancara dilakukan untuk mengetahui bagaimana cara memesan guru les, bagaimana bimbingan belajar dilakukan dan bagaimana prosedur pengajiannya.

4.2 **Analisis Kebutuhan**

Tahap yang digunakan untuk mengetahui atau mendapatkan kebutuhan-kebutuhan perangkat lunak merupakan definisi dari analisis kebutuhan. Pada tahap analisis kebutuhan ini terdapat langkah-langkah yang dilakukan yaitu

dengan mengidentifikasi aktor, analisis kebutuhan dan spesifikasi kebutuhan, kebutuhan fungsional murid, kebutuhan fungsional guru les privat dan kebutuhan fungsional admin.

4.3 Identifikasi Aktor

Untuk mengetahui pengguna-pengguna yang terlibat dalam sistem maka ditulis identifikasi aktor. Terdapat empat pengguna sistem, yaitu Pengguna, Murid, Guru Les privat dan Admin. Disini pengguna merupakan aktor yang hanya dapat melihat beranda, daftar dan login. Murid dalam sistem ini merupakan tokoh utama yang berperan dalam mencari dan memesan Guru. Guru juga merupakan tokoh yang penting karena memberikan jasa les privat untuk membantu murid-murid dalam belajarnya. Admin dapat mengontrol kegiatan murid dan guru. Hasil wawancara rata-rata murid membutuhkan sistem yang digunakan untuk mencari guru les privat, sehingga muncul aktor yaitu murid sebagai pengguna sistem. Selanjutnya guru les privat juga diuntungkan oleh adanya sistem ini untuk memperkenalkan dirinya sebagai guru les privat sehingga peluang untuk mengajar besar, maka muncul aktor yaitu guru les privat sebagai pengguna dalam sistem. Kemudian admin yang harus ada untuk mengontrol jalannya sistem.

4.4 Kebutuhan Fungsional

Dalam membangun suatu perangkat lunak maka harus didefinisikan kebutuhan fungsional, yang merupakan kebutuhan yang harus ada pada sistem. Terdapat Kebutuhan Fungsional Pengguna, Kebutuhan Fungsional Murid, Kebutuhan Fungsional Guru Les Privat, Kebutuhan Fungsional Admin. Dari hasil analisis yang dibuat, didapatkan 60 kebutuhan fungsional.

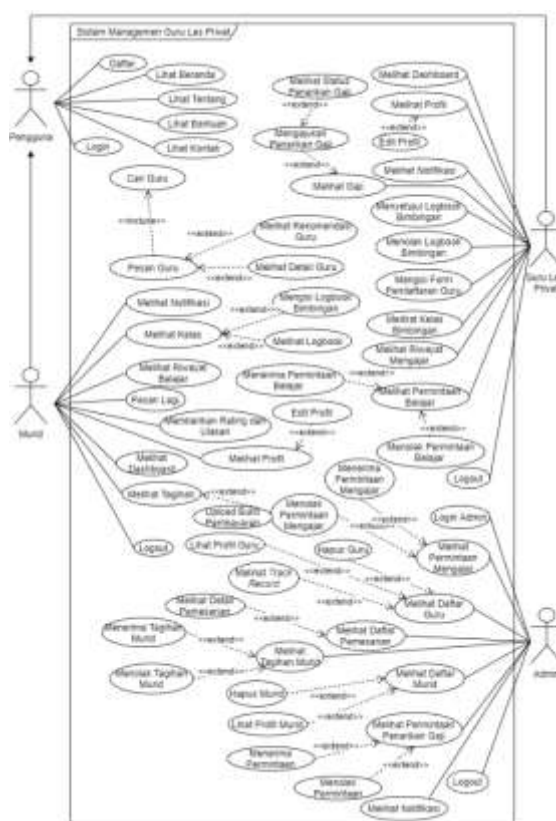
4.5 Kebutuhan Non Fungsional

Kebutuhan non fungsional mengacu pada kinerja atau *performance* sistem atau seberapa baik sistem itu. Untuk mengetahui seberapa baik sistem apabila digunakan di berbagai platform, oleh karena itu kebutuhan non fungsional dipakai untuk mengetahui seberapa *compatible* sistem apabila di buka di berbagai platform browser, hal ini dilakukan karena *software* yang dibangun merupakan sistem berbasis web. Pada tahap ini berharap sistem dapat diakses melalui IE,

Edge, Firefox, Safari, Opera dan Chrome.

4.6 Pemodelan Kebutuhan

Pemodelan kebutuhan ini digunakan untuk lebih memperjelas mengenai fungsionalitas sistem yang sedang dianalisis. Dalam pemodelan kebutuhan dibagi menjadi dua pendekatan yaitu pendekatan terstruktur dan pendekatan object oriented. Dalam penelitian ini dibuat dengan pendekatan *object oriented*. Pada pendekatan *object oriented* objek-objek berinteraksi dan bekerja sama. Pemodelan ini akan dibuat *use case diagram* dan *use case* skenario.



Gambar 3. Use Case Diagram

5. PERANCANGAN

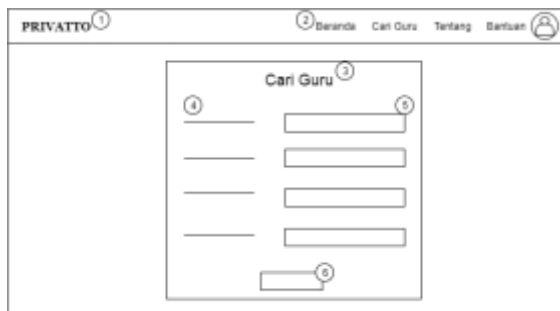
5.1 Perancangan Basis Data

Dalam perancangan basis data divisualisasikan dengan *Entity Relationship Diagram(ERD)* yang terdapat beberapa entiti beserta atributnya dan entity tersebut saling terhubung satu sama lain.

5.2 Perancangan Antarmuka

Perancangan antarmuka merupakan sebuah gambaran awal antarmuka yang nantinya akan

dibuat pada tahap implementasi. Pada perancangan antarmuka ini juga mempermudah dan cepat dalam membuat antarmuka sistem di tahap implementasi.



Gambar 4. Perancangan Antarmuka Cari Guru

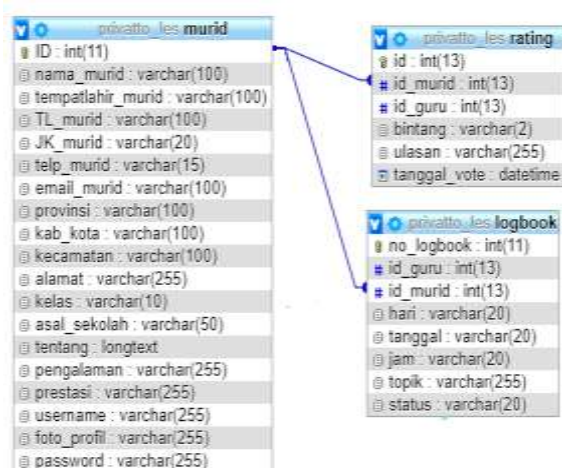
Keterangan Gambar 4:

1. Logo
2. Menu Bar
3. Judul Form
4. Label
5. Isian
6. Cari

6. IMPLEMENTASI

6.1 Implementasi Basis Data

Dalam mengimplementasikan program digunakan MySQL, dan pada Gambar merupakan hasil implementasi databasenya.



Gambar 5 Implementasi Basis Data

6.2 Implementasi Antarmuka

Pada halaman cari guru sangat penting bagi murid untuk mencari guru yang mereka inginkan. Pada Gambar 6 merupakan halaman yang digunakan untuk mencari guru les privat.



Gambar 6 Implementasi Halaman Antarmuka Cari Guru

7. PENGUJIAN

7.1 Pengujian Unit

Untuk menguji algoritme pada suatu method maka dilakukan pengujian unit. Pengujian Unit ini menggunakan metode *white box*. Pengujian *White Box* berguna untuk mengetahui cara kerja suatu perangkat lunak secara internal. Dalam pengujian *White Box* menggunakan pengujian *basis path testing* atau pengujian jalur dasar. Algoritme yang telah diuji yaitu algoritme cari guru, tambah rating dan terima tagihan murid.

7.2 Pengujian Validasi

Pengujian *Black Box*, untuk mengetahui dan memeriksa fungsional dari perangkat lunak serta kinerja sistem. Untuk pengujian *Black Box* menggunakan pengujian validasi. Dari pengujian yang dilakukan didapatkan 77 kasus uji.

7.3 Pengujian Compatibility

Pengujian *compatibility* digunakan sebagai sarana dalam pengujian sistem sehingga dapat diketahui seberapa *compatible* sistem yang telah dibuat maka digunakan. Pengujian ini digunakan untuk melihat kemampuan sistem dapat berjalan di berbagai *browser* yang berbeda. Pengujian *compatibility* diuji menggunakan tool SortSite.

7.4 Analisis dan Hasil Pengujian

Pengujian unit yang telah dilakukan sebelumnya, yaitu dengan menguji tiga algoritme yaitu pengujian algoritme cari guru, tambah rating dan terima tagihan murid. Pengujian algoritme cari guru didapatkan dua jalur independen yang di masing-masing jalurnya berstatus valid. Selanjutnya dalam pengujian algoritme tambah rating juga didapatkan 2 jalur independen yang masing-masing jalur berstatus valid. Yang terakhir yaitu

pengujian algoritme terima tagihan murid yang memiliki 2 jalur independen dan di masing-masing jalurnya berstatus valid.

Dalam pengujian validasi yang telah dilakukan, telah didapatkan sebanyak 77 kasus uji. Dalam setiap kasus uji didapatkan status valid. Dari hasil persentase tersebut menunjukkan hasil persentase valid 100% berdasarkan banyaknya jumlah kasus uji yang ber status valid yang dibagi dengan jumlah total kasus uji dan kemudian dikalikan 100%. Maka hal ini menunjukkan bahwa hasil pengujian validasi yang dilakukan telah sesuai dengan kebutuhan fungsional.

Setelah dilakukan pengujian *compatibility* dengan menggunakan *tool* SortSite, maka didapatkan sebuah hasil yang menunjukkan bahwa sistem manajemen guru les privat dapat digunakan di berbagai platform *browser* seperti IE, Edge, Firefox, Safari, Opera dan Chrome. Serta pada platform IOS dan android. Namun privatto tidak dapat digunakan pada android dengan versi di bawah 3.

8. KESIMPULAN DAN SARAN

Setelah Sistem Manajemen Guru les privat dibuat yang dimulai dari tahap rekayasa kebutuhan, perancangan, implementasi dan pengujian. Dalam rekayasa kebutuhan yang telah diuraikan di bab sebelumnya, maka dapat disimpulkan bahwa Sistem Manajemen Guru Les Privat memiliki sebanyak 60 kebutuhan fungsional dan digunakan oleh 3 pengguna yaitu murid, guru dan admin. Dalam merancang Sistem Manajemen Guru Les Privat dilakukan enam tahap, yaitu perancangan arsitektur, perancangan *class diagram*, perancangan *sequence diagram*, perancangan algoritme dan perancangan antarmuka. Implementasi Sistem Manajemen Guru Les Privat dilakukan dengan memakai *framework Codeigniter* dan dengan metode *MVC(Model View Controller)* dan menggunakan *MySQL* untuk implementasi *database*. Pengujian dilakukan dengan pengujian unit, pengujian validasi dan pengujian *compatibility*. Kemudian setelah dilakukan pengujian maka dihasilkan persentase valid sebanyak 100%.

Menurut hasil penelitian yang telah selesai dikerjakan maka ditarik beberapa saran, yang mungkin nantinya dapat dibuat untuk memperbaiki sistem lebih lanjut. Saran ini diharapkan dapat membuat sistem yang telah

dibuat menjadi lebih baik lagi. Dalam pengembangan lebih lanjut diharapkan Sistem Manajemen Guru Les Privat memiliki fitur untuk menyeleksi guru yang ingin bergabung di Privatto. Penambahan fitur *payment gateway* agar dapat melakukan pembayaran online sehingga lebih sistem bisa kompleks lagi dan lebih mudah penggunaannya. Penambahan fitur untuk mengganti jadwal bimbingan belajar, agar murid dapat mengganti jadwal les ketika ingin menggantinya. Penambahan fitur untuk *edit* jadwal guru apabila guru ingin mengganti atau menambah jadwal yang ia inginkan. Saat ini dalam penggunaannya, Privatto masih digunakan dalam bentuk *web* dan dapat dikembangkan lagi lebih lanjut dengan berbasis *mobile*.

9. DAFTAR PUSTAKA

- Abidin, Z. (2006). Layanan Bimbingan Belajar sebagai Upaya Peningkatan Kualitas Proses Belajar Mengajar. *Jurnal Pemikiran Alternatif Kependidikan*, 34-48.
- Asghar, D., & Umar, M. (2010). Requirement Engineering Challenges in Development of Software Applications and Selection of Customer-off-the-Shelf (COTS) Components. *International Journal of Software Engineering (IJSE)*, 32-50.
- Dwanoko, Y. S. (2016). Implementasi Software Development Life Cycle(SDLC) Dalam Penerapan Pembangunan Aplikasi Perangkat Lunak. *Jurnal Teknologi Informasi*.
- Fahurrozi, I., & SN, A. (2012). Poses Pemodelan Software Dengan Metode Waterfall dan Extremme Programming: Studi Perbandingan. *Jurnal ONLINE STMIK EL- Rahma*.
- Gandy, E., & Stobart, S. (2005). *JavaScript: Creating Dynamic Web Pages*. Colchester: Lexden Publishing.
- Haryana, K. S. (2008). Pengembangan Perangkat Lunak Dengan Menggunakan PHP. *Jurnal Computech & Bisnis*.
- Hustinawati, Himawan, A. K., & Latifah. (2014). Performance Analysis Framework Codeigniter and CakePHP in Website Creation. *International Journal of Computer Application*.

- Kumar, M., Kumar, V., & Yogi, M. K. (2015). Text Data Generation Technique for Object Oriented Software with Comparison among Black Box Testing and White Box Testing Techniques . *International Journal of Modern Computer Science and Applications (IJMCSA)* , 10-14.
- Meyer, E. A. (2004). *Cascading Style Sheets: The Definitive Guide*. USA: O'Reilly Media.
- Nidhra, S., & Dondeti, J. (2012). Black Box And White Box Testing Techniques - A Literature Review. *International Journal of Embedded Systems and Application(IJESA)*.
- Nugroho, A. (2009). *Rekayasa Perangkat Lunak Menggunakan UML dan JAVA*. Yogyakarta: ANDI Yogyakarta.
- Oktavian, D. P. (2010). *Menjadi Programmer Jempolan Menggunakan PHP*. Yogyakarta: MediaKom.
- Pressman, R. S. (2010). *Software Engineering: A Practitioner's Approach Seventh Edition*. New York: McGraw-Hill.
- Sari, F. (2018). *Metode dalam Pengambilan Keputusan*. Sleman: Deepublish.
- Shah, M. D., & Dhiman, C. (2016). Software Process Models Outline. *International Journal of Computer Science and Information Technology & Security (IJCSITS)*.
- Simarmata, J. (2010). *Rekayasa Perangkat Lunak*. Yogyakarta: Penerbit ANDI.
- Sularso, P., Kokotiasa, W., & Paramedica, D. A. (2015). Pengaruh Bimbingan Belajar Di Luar Sekolah Terhadap Minat Untuk Berprestasi Siswa Kelas VIII MTsN Sidorejo Kabupaten Madiun Tahun Pelajaran 2014/2015. *Citizenship Jurnal Pancasila dan Kewarganegaraan*.
- Sulistiono, H. (2018). *Coding Mudah dengan Codeigniter, JQuery, Bootstrap dan Datatable*. Jakarta: Elex Media Komputindo.
- White, S. A. (2004). Introduction to BPMN. *BPTrends*, 1-6.
- Whitten, J. L., Bentley, L. D., & Dittman, K. C. (2001). *Systems Analysis And Design Methods*. New York: David Kendric Brake.
- Whitten, J. L., Bentley, L. D., & Dittman, K. C. (2004). *Metode Desain & Analisis Sistem Edisi 6*. Yogyakarta: Penerbit ANDI.
- Whitten, J. L., Bentley, L. D., & Dittman, K. C. (2004). *Metode Desain & Analisis Sistem Edisi 6*. Yogyakarta: Penerbit ANDI.