

BAB II

KAJIAN PUSTAKA

A. Kajian Teori

1. Pengembangan Sistem Informasi

Sistem informasi terdiri dari dua kata yaitu "sistem" dan "informasi". Sistem adalah sebuah kumpulan dari komponen yang saling berhubungan, dengan batasan yang jelas, bekerja sama untuk mencapai tujuan tertentu (O'Brien, 2011: 38). Dan Informasi adalah hasil pengelolaan data dalam bentuk yang lebih berguna dan lebih berarti bagi penerimanya yang menggambarkan suatu kejadian-kejadian nyata yang digunakan untuk pengambilan keputusan (Irmawati dan Indrihapsari, 2014: 137).

Secara keseluruhan menurut Kristanto (2008: 12) "Sebuah sistem informasi merupakan kumpulan dari perangkat keras dan perangkat lunak komputer serta perangkat manusia yang akan mengolah data menggunakan perangkat lunak tersebut". Sebuah sistem informasi memiliki beberapa komponen yang memiliki fungsi penting demi kelancaran sebuah sistem informasi, komponen-komponen tersebut adalah: *input*, *output*, teknologi, basis data, dan kendali. (Anwar: 2009: 13). Dalam sebuah organisasi, sistem informasi memiliki manfaat untuk mengolah transaksi, mengurangi biaya operasional dan juga menghasilkan pendapatan dari produk maupun pelayanan yang diberikan organisasi tersebut.

Berdasarkan penjabaran di atas, dapat disimpulkan bahwa sebuah sistem informasi adalah sebuah perangkat yang memiliki peranan penting dalam sebuah organisasi. Sistem informasi dapat menyajikan informasi dan memiliki

fungsi yang dapat membantu proses berjalannya suatu organisasi sehingga dapat mengurangi biaya operasional organisasi tersebut.

Secara keseluruhan, tujuan dari pengembangan sistem adalah untuk mencapai kebutuhan bisnis, bukan teknis, dengan menyampaikan informasi yang tepat kepada orang yang tepat pada waktu yang tepat (Reynolds, 2010: 494). Pengembangan sistem dapat juga disebut sebagai *Software Development Life Cycle* atau *SDLC* karena kegiatan yang terasosiasi dengan prosesnya sedang berjalan (Reynolds, 2010: 496).

SDLC bisa disebut juga model pengembangan *software*. Model pengembangan yang digunakan pada penelitian ini adalah *linear sequential model* atau bisa juga disebut *waterfall* model. Menurut Pressman (2001) model linear terdiri dari lima tahap yang dijabarkan di bawah ini:

a. Analisis

Tahap analisis bisa juga dikatakan sebagai tahap dimana peneliti mencoba untuk menjawab pertanyaan “Apa yang perlu dilakukan oleh sistem informasi agar dapat menyelesaikan masalah?” (Reynolds, 2010: 497). Pada tahap analisis, fungsi dan tujuan dari sistem ditentukan dengan melakukan wawancara pada pengguna. Analisis dilakukan secara mendetail agar sesuai dengan kebutuhan pengguna dan menjadi spesifikasi dari sistem tersebut (Somerville, 2011: 31). Di penelitian ini tahap analisis dilakukan dengan melakukan wawancara dan observasi.

Salah satu luaran dari tahap analisis adalah *use case* yang menggambarkan sekumpulan aktifitas yang dilakukan untuk menghasilkan *output* tertentu (Dennis, 2009: 149). *Use case* yang ada kemudian disederhanakan dalam bentuk diagram *use case* yang dibuat menggunakan

UML. *Unified Modeling Language* atau UML merupakan bahasa model visual yang digunakan untuk menjelaskan, menggambarkan, membangun dan mendokumentasikan struktur dari sistem sebuah perangkat lunak (Rumbaugh, 1998: 3).

b. Desain

Pada tahap desain dialokasikan kebutuhan perangkat keras atau perangkat lunak untuk sistem dengan menetapkan arsitektur sistem secara keseluruhan (Somerville, 2011: 31). Desain yang sukses dibangun dari apa yang dipelajari pada tahap sebelumnya dan dapat menyebabkan lancarnya proses implementasi dengan membuat rencana yang akurat dan jelas mengenai hal yang harus dilakukan (Dennis, 2012: 259). Tahapan desain merupakan proses yang memiliki beberapa tahap yang terfokuskan pada struktur data, arsitektur perangkat lunak, antar muka, dan detil algoritma (Pressman, 2001: 29).

Tahap desain yang dilakukan pada penelitian ini hanya berupa tahap desain antar muka, dan struktur data, karena proses pengembangan sistem ini menggunakan Laravel, yang merupakan *web application framework* dengan sintaks yang ekspresif dan elegan (www.laravel.com). Laravel memiliki arsitektur berupa MVC atau *Model view Controller*. MVC yang merupakan sebuah cara dimana objek-objek yang berbeda mengambil alih tugas yang berhubungan dengan *domain* aplikasi (*model*), tampilan dari aplikasi tersebut (*view*), dan interaksi pengguna dengan model dan tampilan (*controller*). Selain itu Laravel juga sudah mempermudah penelitian dengan menetapkan detil algoritma, sehingga tahapan desain yang harus dilakukan hanyalah desain antar muka dan desain struktur data.

Desain antar muka dilakukan dengan menggunakan wireframe yang merupakan *blueprint* sederhana yang menggambarkan form utama dan fungsi yang ditemukan pada sebuah halaman dari web atau aplikasi yang dikembangkan (Hamm, 2014: 15). Sedangkan untuk struktur data didesain menggunakan ERD atau *Entity Relationship Diagram* merupakan salah satu cara untuk menampilkan data secara agar mudah dipahami. Menurut Kristanto (2008: 91) ERD digunakan untuk mengembangkan inisial dari desain basis data. Dengan ERD data yang disimpan dapat terlihat dengan jelas dan akurat (Dennis, 2012: 419).

c. Implementasi

Saat memasuki tahapan ini, desain yang telah dibuat direalisasikan menjadi sekumpulan program dan fungsi (Somerville, 2011: 31). Penelitian ini diimplementasikan pada platform *web*, karena dengan itu pengguna dapat berbagi informasi pada platform yang independen yang dapat diakses dari banyak tempat (Fong, 2007). Implementasi dapat dikatakan berjalan dengan baik jika memenuhi semua kebutuhan yang sudah diutarakan dalam tahapan desain (Pressman, 2001: 31). Pada penelitian ini implementasi dilakukan dengan menerapkan apa yang didapat dari tahap desain pada kode program yang ditulis pada *framework* Laravel.

d. Pengujian

Pada tahap pengujian, sistem yang telah diimplementasikan diuji secara menyeluruh untuk memastikan bahwa sistem yang dibuat telah sesuai dengan kebutuhan. Setelah diuji, perangkat lunak yang dikembangkan diberikan kepada pengguna yang membutuhkan (Somerville, 2011: 31). Pada penelitian

ini tidak dilakukan pengujian, melainkan diganti dengan pengecekan kualitas perangkat lunak menggunakan ISO 25010.

e. Pemeliharaan

Tahap pemeliharaan merupakan tahap yang paling panjang, walaupun terkadang tidak selalu begitu. Setelah diberikan kepada pengguna, perangkat lunak mengalami proses perawatan dimana jika terjadi error selama penggunaan maka perlu dilakukan perbaikan. Tahapan ini juga menjadi tahapan dimana jika ingin menambahkan fitur baru maka perubahan pada sistem pun perlu dilakukan (Somerville, 2011: 31).

Dalam model pengembangan *waterfall* secara teori proses berjalan setelah tahapan sebelumnya selesai dan disetujui, namun realitanya setiap tahapan berjalan dengan bersentuhan dengan tahapan yang lain. Desain yang telah disetujui bisa mengalami perubahan ketika diperlukan (Somerville, 2011).

2. ISO 25010

ISO 25010:2011 merupakan standar internasional untuk kualitas sebuah produk dalam cakupan pengembangan perangkat lunak. ISO 25010:2011 merupakan pengganti dari ISO 9126:1991. ISO 25010 digunakan sebagai standar untuk pengujian aplikasi ini. Sean Mee dalam Nilamsari (2014: 18) menambahkan bahwa penggunaan model ISO dalam suatu pengembangan perangkat lunak disesuaikan dengan karakteristik aplikasi yang akan dikembangkan. Menurut ISO 2510:2011 standar kualitas yang harus diperhatikan antara lain:

a. *Functional suitability*

Functional suitability merupakan karakteristik yang menyatakan bahwa produk atau sistem yang dikembangkan menyediakan fungsi yang diperlukan ketika digunakan dalam kondisi tertentu (ISO 25010). Pengujian fungsionalitas dilakukan dengan menggunakan kuisioner dengan skala Guttman. Skala Guttman dipakai untuk mendapatkan hasil yang tegas yaitu ya dan tidak (Sugiyono, 2011: 139). Pengukuran aspek *functional suitability* dilakukan dengan mengukur *feature completeness* atau kelengkapan fitur dengan mengukur jumlah fitur yang diterapkan (Acharya, 2013). Menurut Acharya sistem dikatakan berfungsi jika semua fitur yang diajukan dapat diimplementasikan seluruhnya.

b. *Performance efficiency*

Performance efficiency merupakan kriteria yang menggambarkan performa sistem yang relatif terhadap jumlah kebutuhan yang digunakan pada kondisi tertentu (ISO 25010). Dalam pengujian performa, digunakan *software* GTMetrix untuk mengukur performa sistem. Sistem dianggap memiliki performa yang baik jika berada diatas nilai rata-rata GTMetrix (GTMetrix.com).

c. *Compatibility*

Compatibility adalah tingkat dimana produk atau sistem dapat bertukar informasi dengan produk atau sistem yang lain, dan juga dapat melakukan fungsi yang dibutuhkan ketika menggunakan perangkat keras atau perangkat lunak yang sama (ISO 25010). Dalam penelitian ini *compatibility* tidak diuji karena sistem tidak membutuhkan integrasi dengan sistem lain.

d. *Usability*

Usability adalah tingkat dimana produk atau sistem dapat digunakan oleh pengguna tertentu untuk mencapai tujuan tertentu dengan efektif, efisien, dan kepuasan dalam menjalankan perintah tertentu (ISO 25010). Penelitian ini menguji *usability* dengan menggunakan kuisisioner yang diberikan oleh John Brooke (1996) yang disebut *System Usability Scale* yang terdiri dari 10 pertanyaan yang diukur menggunakan skala Likert. Menurut Bangor (2009) jika hasil kuisisioner nilainya diatas 50, maka sistem dianggap cukup memenuhi kriteria *usability*.

e. *Reliability*

Reliability adalah tingkat dimana produk atau sistem dapat menjalankan fungsi tertentu dengan kondisi yang sudah ditentukan dalam waktu yang ditentukan (ISO 25010). Menurut Scott Speaks (2005) *Reliability* diuji dengan menghitung *Mean Time Between Failure (MTBF)*, dan jika nilai MTBF rendah maka sistem semakin *reliable*.

f. *Security*

Security adalah tingkat dimana produk atau sistem dalam melindungi informasi dan data sehingga ketika pengguna atau sistem yang lain dapat mengakses data yang sesuai dengan tipe dan tingkat otorisasi yang sesuai (ISO 25010). *Security* diuji menggunakan *software* Acunetix dan diukur tingkat keamanannya, jika *level*/ancamannya rendah maka perangkat lunak dianggap aman (Accunetix FAQ).

g. *Maintainability*

Maintainability adalah karakteristik yang menampilkan tingkat efektifitas dan efisiensi dari produk atau sistem ketika akan diubah, dikembangkan,

diperbaiki, atau disesuaikan dengan kebutuhan (ISO 25010). Aspek *Maintainability* diukur dengan menggunakan PHPMetric dan dilihat hasil perhitungan *Maintainability Index*-nya. Menurut Coleman (1994: 44), software dikatakan baik jika nilai *maintainability index* ≥ 65 , kategori sedang karena tidak memiliki masalah yang serius.

h. *Portability*

Portability adalah tingkat efektifitas dan efisiensi sistem atau produk ketika digunakan di lingkungan, perangkat keras, atau perangkat lunak yang berbeda (ISO 25010). Pengujian portabilitas sistem diukur dengan menguji sistem diberbagai lingkungan yang berbeda, pada penelitian ini sistem diuji pada lima *browser* dan 2 sistem operasi. Sebuah sistem dikatakan baik, ketika sistem tersebut dapat berjalan di browser yang berbeda, perangkat keras berbeda dan sistem operasi yang berbeda (Weinmaan, 2001: 7)

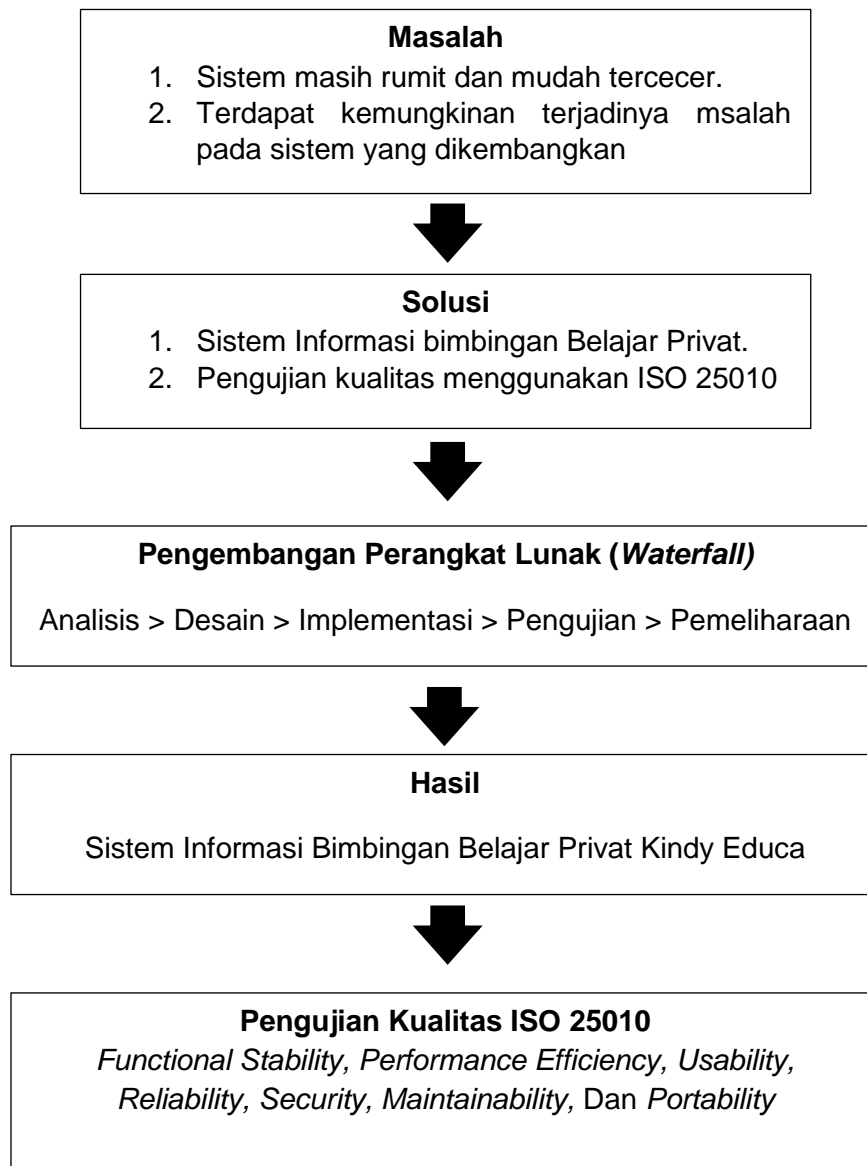
B. Kerangka Berfikir

Permasalahan yang ada pada Lembaga Bimbingan Belajar Kindy Educa adalah tidak adanya sistem administrasi pendaftaran yang *reliable* dan juga tidak memakan banyak biaya. Saat ini proses pendaftaran dan administrasi masih dilakukan secara manual sehingga membutuhkan banyak biaya untuk mencetak formulir pendaftaran dan kemungkinan tercecernya berkas-berkas administrasi. Untuk mengatasi permasalahan tersebut, maka dibuatlah sistem informasi bimbingan belajar, yang dapat memudahkan proses pendaftaran dan administrasi.

Pengembangan sistem dilakukan melalui tahapan analisis, desain, implementasi, dan pengujian. Pengujian kualitas sistem dilakukan agar mencegah terjadinya error dan munculnya *bug* pada sistem yang

dikembangkan, pengujian dilakukan menggunakan ISO 25010 yang terdiri dari beberapa aspek yaitu *functional stability*, *performance efficiency*, *compatibility*, *usability*, *reliability*, *security*, *maintainability*, dan *portability*.

Kerangka penelitian ini dijelaskan pada Gambar 2 di bawah ini:



Gambar 1. Diagram Kerangka Pikir

C. Penelitian yang Relevan

1. Aplikasi Reservasi Les Private Di *Easyspeak* Denpasar Berbasis Web Dan Android oleh Fery Yudhitama Putra, dkk pada tahun 2016. Hasil penelitian ini terdiri dari web dan Android. Pembangunan aplikasi reservasi les privat dibangun dengan bahasa pemrograman PHP dan Java dengan menggunakan *framework* Codeigniter pada sisi web, sedangkan pada sisi Android menggunakan *tool* Eclipse, serta MySQL sebagai media penyimpanan basis data. Aplikasi reservasi les privat ini memiliki beberapa fungsi untuk melakukan pemesanan waktu dan tutor yang dapat dilakukan sendiri oleh student dari *Easyspeak* dan pada sisi tutor aplikasi dapat memberikan informasi student yang akan diajar, serta pada sisi operator dapat memberikan kemudahan dalam melakukan pengaturan booking les private karena sudah terkomputerisasi tidak manual seperti dulu. Aplikasi reservasi les privat juga dilengkapi dengan sistem *reminder* atau pengingat yang dibuat pada sisi aplikasi Android, dengan menggunakan sistem *alarm manager*.
2. Sistem Reservasi Tempat *Meeting* Berbasis Android oleh Desta Aditya Kusumajati pada tahun 2017. Hasil penelitian ini adalah suatu sistem informasi reservasi tempat *meeting* berbasis android untuk user/pengguna dan berbasis website untuk admin di lokasi tempat *meeting*. Fitur yang diberikan pengguna adalah pengguna dapat memesan tempat meeting berdasar waktu, tempat yang diinginkan dan jumlah kapasitas peserta *meeting* yang bisa dilakukan tanpa pengguna harus reservasi langsung kelokasi dan mendapatkan informasi

pembayaran setelah melakukan pemesanan. Untuk admin diberi website admin yang digunakan untuk memantau pesanan yang masuk. Dengan dilakukannya penelitian ini dapat mempermudah sejumlah orang atau instansi dan perusahaan yang ingin memesan tempat *meeting* secara cepat disaat waktu yang tidak terduga atau dari jauh-jauh hari tanpa harus *review* lokasi meeting dengan cara mengunjungi satu persatu lokasi tetapi hanya dengan melihat *review* tempat melalui aplikasi. Mereka langsung dapat menentukan tempat seperti apa yang diinginkan serta menentukan ruangan yang memadai dengan jumlah peserta *meeting*.