



Sistem Informasi Bimbingan Belajar Berbasis *Web* dengan Metode *Waterfall*

Daud Situmorang¹, Iskandar Fitri², Benrahman³

^{1,2,3} Fakultas Teknologi Komunikasi dan Informatika, Universitas Nasional

article info

Article history:

Received 13 November 2020

Received in revised form

24 December 2020

Accepted 3 January 2021

Available *online* October 2021

DOI:

<https://doi.org/10.35870/jtik.v5i4.235>

Keywords:

Web Design Application,
Design and Implementation
Application Information
Systems.

Kata Kunci:

Perancangan Sistem Aplikasi
Bimbingan Belajar,
Implementasi Aplikasi Berbasis
Web.

abstract

The creation of a web-based information system at the North Jakarta Sunter branch of Kumon tutoring is intended to improve service quality and quality in the implementation of teaching and learning activities. Based on this, this study will develop a web-based information system that includes an interface design, namely the main page, login, subjects, schedule of subjects, and grades. Making this system an effort to optimize service performance to students.

abstrak

Pembuatan sistem informasi berbasis web pada bimbingan belajar Kumon cabang Sunter Jakarta Utara dimaksudkan untuk meningkatkan kualitas pelayanan dan mutu dalam penyelenggaraan kegiatan belajar mengajar. Berdasarkan hal tersebut dalam penelitian ini akan dikembangkan suatu sistem informasi berbasis web yang meliputi rancangan antar muka yaitu halaman utama, login, mata pelajaran, jadwal mata pelajaran, dan nilai. Pembuatan sistem ini merupakan upaya optimalisasi kinerja pelayanan terhadap siswa didik.

*Corresponding author. Email: daudsitumoranggg@gmail.com¹, tektel2001@yahoo.com², ben.rahman@gmail.com³.

© E-ISSN: 2580-1643.

Copyright © 2021. Published by Lembaga Informasi dan Riset (KITA INFO dan RISET), Lembaga KITA (<http://creativecommons.org/licenses/by/4.0/>).

1. Latar Belakang

Pandemi covid-19 memberi dampak yang sangat signifikan di bidang pendidikan [1], membuat perubahan pola belajar mengajar yang harus menyesuaikan dengan keadaan guna menjaga dan menjalankan protokol kesehatan dalam kehidupan sehari-hari sesuai anjuran Pemerintah [2, 3].

Sebagai lembaga pendidikan informal, Kumon cabang Sunter masih menggunakan cara konvensional dalam menyajikan data-data siswanya. Sehingga perlu dibangun sebuah aplikasi berbasis *web* yang memiliki fitur-fitur untuk memenuhi kebutuhan kegiatan pembelajaran. Sehingga diharapkan dapat menjadikan kegiatan pembelajaran yang lebih efisien serta dapat menghemat waktu, biaya dan tenaga pada saat mengakses informasi belajarnya.

Selain itu pembuatan sistem informasi bimbingan belajar berbasis *web* diharapkan dapat meningkatkan daya saing di tengah semakin banyaknya bimbingan belajar yang muncul. Membantu proses administrasi dalam membuat jadwal belajar siswa, guru, kelas dan nilai siswa serta upaya adaptasi ditengah perkembangan teknologi informasi.

2. Landasan Teori

Perancangan Sistem Informasi

Perancangan sistem merupakan rangkaian aktivitas bagaimana struktur sebuah sistem akan berjalan sesuai dengan kebutuhan [4]. Sedangkan sistem informasi merupakan struktur sebuah kegiatan yang saling berkaitan juga berhubungan untuk mendukung kelancaran suatu kegiatan [5].

Secara konseptual siklus pengembangan sebuah sistem informasi terdiri dari; 1) Analisis Sistem, 2) Perancangan Sistem, 3) Pembangunan dan Testing Sistem, 4) Implementasi Sistem, 5) Operasi dan Perawatan, dan 6) Evaluasi Sistem [6]. Siklus tersebut berlangsung secara berulang-ulang. Siklus di atas merupakan model klasik dari pengembangan sistem informasi [7]. Model-model baru, seperti *prototyping*, spiral, *systems development life cycle*, 4GT dan kombinasi dikembangkan dari model klasik melalui siklus tersebut.

Systems Development Life Cycle (SDLC)

SDLC merupakan langkah-langkah kegiatan agar tercapainya sebuah sistem yang baik sesuai kebutuhan [8]. Dalam studi kasus ini, peneliti menggunakan Model *Waterfall*. *Waterfall* merupakan metode pengembangan yang sistematis juga berurutan yang dimulai dari kebutuhan pengguna, perencanaan, pemodelan, konstruksi dan perawatan sistem [9].

Konsep Basis Data (Database)

Basis data suatu kumpulan data terhubung yang disimpan dengan bersama-sama pada suatu media, tanpa mengatap satu dan yang lainnya [10]. Data disimpan sedemikian rupa sehingga proses pengambilan, penambahan, dan modifikasi data dapat dilakukan dengan sangat mudah dan terkontrol. Basis data bersifat data *oriented* bukan program *oriented* [11], dapat digunakan oleh beberapa program aplikasi tanpa perlu mengubah basis datanya. Hal ini juga dapat dikembangkan dengan mudah baik volume maupun strukturnya sehingga dapat memenuhi kebutuhan sistem-sistem baru dengan mudah. Prinsip utama basis data atau *database* adalah pengaturan data dengan tujuan utama fleksibilitas dan kecepatan akses [12]. Adapun tujuan basis data diantaranya sebagai efisiensi yang meliputi *speed*, *space*, dan *accuracy*, mengenai data dalam jumlah besar, kebersamaan pemakai (*sharebility*), dan meniadakan duplikasi dan inkonsistensi data [13].

Unified Modelling Language (UML)

UML terdiri dari sembilan jenis diagram namun ada yang menyebut delapan karna digabung seperti diagram urutan dan komunikasi [14]. UML merupakan pengganti dari metode analisis berorientasi object dan design berorientasi object (OOAD&D/*object-oriented analysis and design*) [15]. UML menyediakan notasi-notasi yang membantu memodelkan sistem dari berbagai prespektif. UML tidak hanya digunakan dalam pemodelan perangkat lunak, namun hampir dalam semua bidang yang membutuhkan pemodelan.

Pengujian (Testing)

Testing merupakan siklus dan bagian dari proses rekayasa perangkat lunak secara terintegrasi untuk memastikan kualitas perangkat lunak sesuai kebutuhan [16]. *Black-box testing* digunakan untuk menemuka gangguan pada aplikasi tersebut sebelum dirilis secara resmi, beberapa kategori yaitu:

- a) *Functional testing* adalah proses pengujian terhadap fungsi atau fitur spesifik sebuah software.
- b) *Non-functional testing* pengujian yang dilakukan pada aspek-aspek tambahan nonfungsional.
- c) *Regression test*, pengecekan dilakukan pada aspek baik fungsional maupun nonfungsional.

3. Metodologi Penelitian

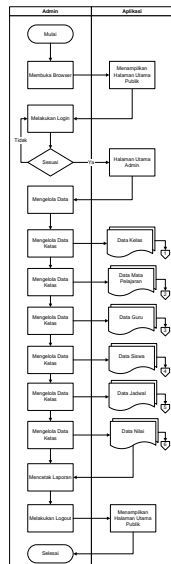
Untuk pengembangan sistem, peneliti menggunakan SDLC metode *Waterfall* yang terdiri dari tahapan;

- 1) Analisa
Dilakukan analisa yang merujuk pada kelebihan dan kekurangan sistem, fungsi sistem dan keterbaruan pada sistem yang dibuat.
- 2) Desain
Perencanaan dan pembuatan sistem yang akan di gunakan oleh *user*.
- 3) Implementasi
Penerapan sebuah desain yang telah di buat dan diintegrasikan menjadi sebuah sistem yang lengkap.
- 4) Pengujian
Merupakan kegiatan terpadu pengendalian standart dan proses sesuai dengan yang di rencanakan.

4. Hasil dan Pembahasan

Flowmap

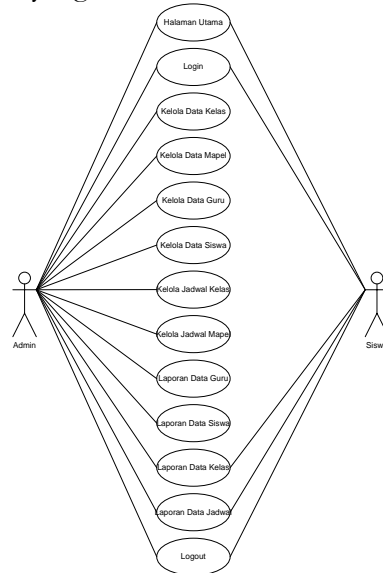
Pada perancangan sistem dibuatkan flowmap yang digunakan sebagai gambaran alur proses dan fitur aplikasi yang diakses oleh admin dan siswa.



Gambar 1. *Flowmap*

Usecase diagram

Pada perancangan sistem dibuatkan *usecase diagram* yang digunakan sebagai gambaran alur proses dan fitur aplikasi yang diakses oleh admin dan siswa.

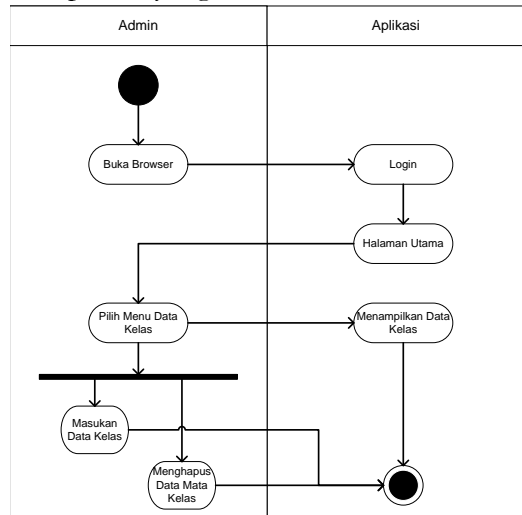


Gambar 2. *Usecase diagram*

Pada *usecase diagram* menunjukkan bahwa sistem hanya memiliki beberapa akses yaitu hanya admin dan siswa. Admin memiliki semua akses pada aplikasi dari login, menu utama, melakukan pengolahan pada data kelas, mapel, data guru, data siswa dan, jadwal kelas, jadwal mapel dan mencetak semua laporan. Sedangkan akses untuk siswa hanya dapat melihat mata pelajaran, melihat jadwal, dan melihat nilai.

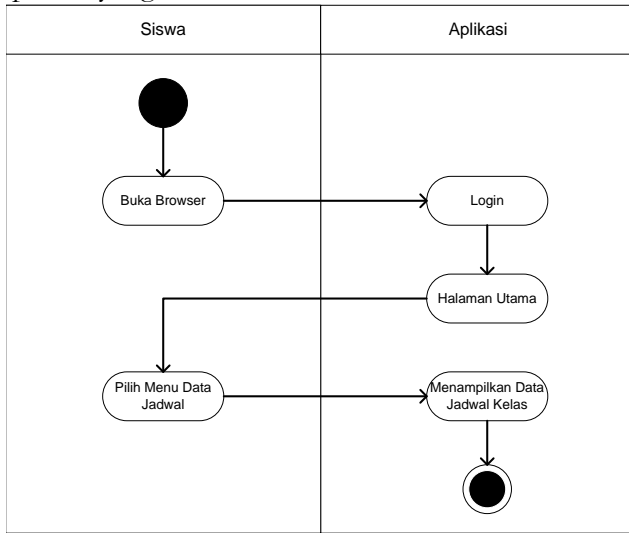
Activity diagram

Pada perancangan sistem selanjutnya dibuatkan *activity diagram* yang digunakan sebagai gambaran alur proses dan fitur aplikasi yang diakses oleh admin dan siswa.



Gambar 3. *Activity diagram Admin*

Pada *activity diagram* menunjukkan proses bisnis pada aplikasi yang dilakukan oleh admin.

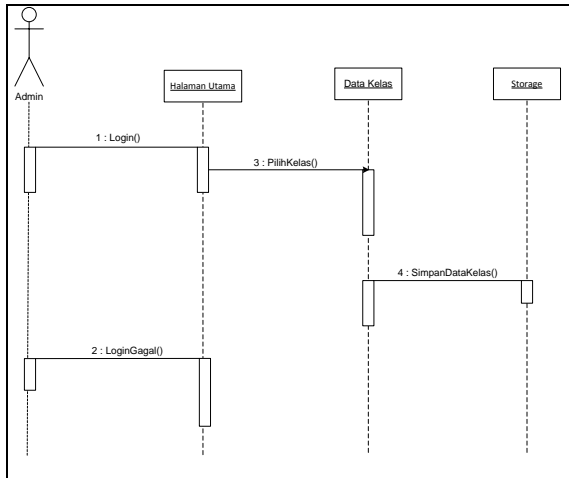


Gambar 4 *Activity diagram* Siswa

Pada *activity diagram* menunjukkan proses bisnis pada aplikasi yang dilakukan oleh siswa.

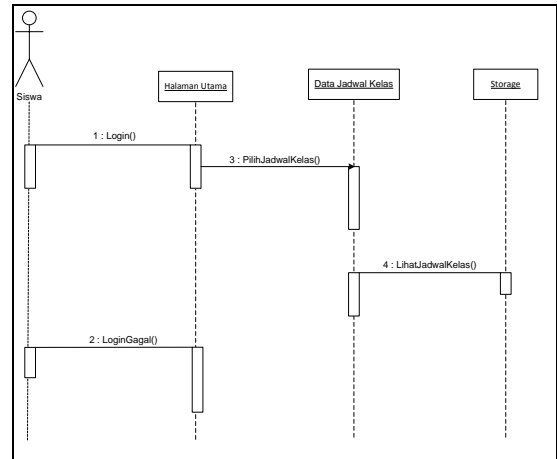
Sequence Diagram

Pada perancangan sistem selanjutnya dibuatkan *activity diagram* yang digunakan sebagai gambaran alur proses dan fitur aplikasi yang diakses oleh admin dan siswa.



Gambar 5. *Sequence Diagram* Admin

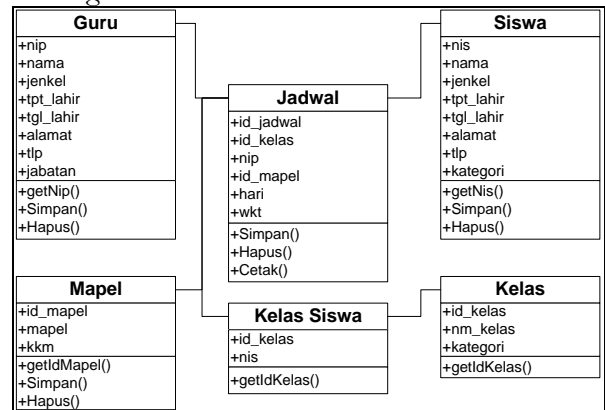
Pada *Sequence Diagram* menunjukkan interaksi yang menekankan pada pengiriman pesan yang dilakukan oleh admin.



Gambar 6. *Sequence Diagram* Siswa

Pada *Sequence Diagram* menunjukkan interaksi yang menekankan pada pengiriman pesan yang dilakukan oleh siswa.

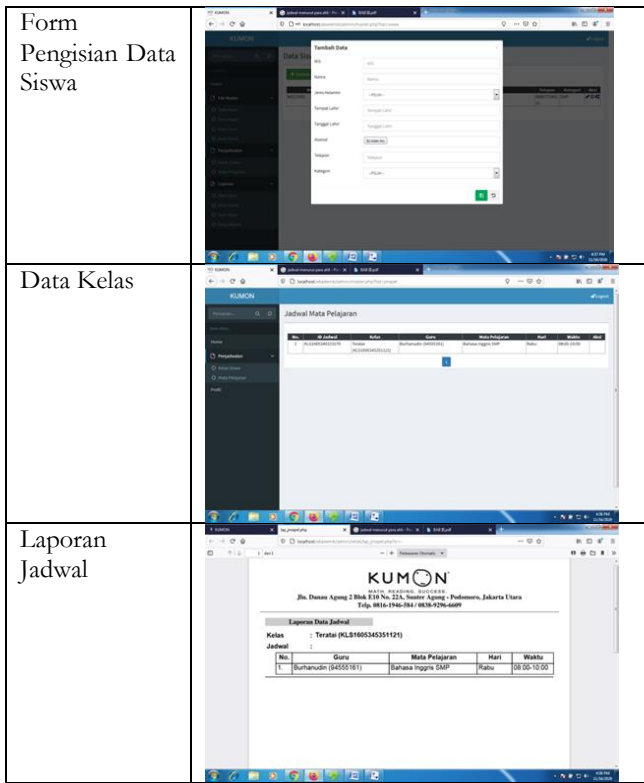
Class Diagram



Gambar 7. *Class Diagram*

Tabel 1. Rancangan Antar Muka

Keterangan	Gambar
Tampilan Menu Utama	
Halaman Login	



		kembali halaman <i>log in</i> .	halaman <i>log in</i> .
Tambah Kelas	Menambahkan kelas.	Sistem dapat menambahkan kelas.	Sistem berhasil menambahkan data kelas.
Hapus kelas	Menghapus kelas.	Sistem dapat menghapus kelas.	Sistem berhasil menghapus data kelas.
Seleksi kelas	Menyeleksi kelas yang dipilih.	Sistem dapat menentukan kelas yang dipilih berdasarkan semester kebutuhan peserta.	Sistem berhasil menentukan kelas yang dipilih berdasarkan semester kebutuhan peserta.
Cetak jadwal	Mencetak data jadwal.	Sistem dapat mencetak data jadwal.	Sistem berhasil mencetak data jadwal.

Pengujian kegunaan siswa sebagai berikut:

Tabel 3. kegunaan Siswa

Nama Kegunaan	Aktivitas Yang Diuji	Produk Yang Diharapkan	Produk Pengetesan
Log in	a. Nama identitas dan kata kunci tidak sesuai. b. Nama identitas dan kata kunci sesuai.	a. Menyajikan pesan nama identitas dan kata kunci tidak sesuai. b. Sistem menampilkan halaman utama dan <i>log in</i> .	a. Tidak tervalidasi, akan muncul pemberitahuan nama identitas dan kata kunci tidak sesuai. b. Tervalidasi, sistem menampilkan halaman utama dan <i>log in</i> .
Log out	Memilih menu <i>Log out</i> .	Sistem memutus akses user dan menampilkan kembali halaman <i>log in</i> .	sistem sukses memutus akses dan kembali pada halaman <i>log in</i> .
Lihat Data Jadwal	Melihat Jadwal Kelas.	Sistem dapat memberikan informasi jadwal kelas.	Sistem berhasil memberikan informasi jadwal kelas.
Cetak jadwal kelas	Mencetak data jadwal kelas berdasarkan nis siswa.	Sistem dapat mencetak data peserta berdasarkan nis siswa.	Sistem berhasil mencetak data kelas berdasarkan nis siswa.

Pengujian Sistem

Pengujian sistem informasi bimbingan belajar berbasis *web* menggunakan *black-box testing*. Hasilnya berupa ketepatan sistem yang dibuat dalam mengolah masukan dan keluaran. Pengujian kegunaan *web* Bimbingan belajar ada dua aktor yaitu admin dan siswa. Pengujian kegunaan admin sebagai berikut:

Tabel 2. Pengujian Kegunaan Admin

Nama Kegunaan	Aktivitas Yang Diuji	Produk Yang Diharapkan	Produk Pengetesan
Log in	a. Nama identitas dan kata kunci tidak sesuai. b. Nama identitas dan kata kunci sesuai.	a. Menyajikan pesan nama identitas dan kata kunci tidak sesuai. b. Sistem menampilkan halaman utama dan <i>log in</i> .	a. Tidak tervalidasi, akan muncul pemberitahuan nama identitas dan kata kunci tidak sesuai. b. Tervalidasi, sistem menampilkan halaman utama dan <i>log in</i> .
Log out	Memilih menu <i>Logout</i> .	Sistem memutus akses user dan menampilkan	sistem sukses memutus akses dan kembali pada

5. Kesimpulan dan Saran

Berikut kesimpulan dari perancangan aplikasi bimbingan belajar yang dibangun:

- 1) Diperlukan *hardware* yang mumpuni agar ketika banyak user memproses data tidak memakan waktu yang lama.
- 2) Aplikasi Bimbingan belajar ini akan mempermudah para siswa dalam mengakses jadwal dan nilai dimana guru tidak perlu lagi membagikan lembaran nilai ke masing-masing siswa, selain itu aplikasi ini juga mengurangi kerja administrasi.

Peneliti memberikan beberapa saran agar aplikasi ini semakin baik jika nantinya akan dikembangkan:

- 1) Pada aplikasi ini sebaiknya di tambahkan fasilitas materi mata pelajaran, kuis dan materi ujian.
- 2) Kedepannya diharap kepada pengembang agar membuat fasilitas chat yang saling terhubung sehingga siswa dapat saling berinteraksi.

6. Daftar Pustaka

- [1] Astini, N.K.S., 2020. Tantangan Dan Peluang Pemanfaatan Teknologi Informasi Dalam Pembelajaran Online Masa Covid-19. *Cetta: Jurnal Ilmu Pendidikan*, 3(2), pp.241-255.
- [2] Mahardhani, A.J. and KP, M., 2020. Menjadi Warga Negara yang Baik pada Masa Pandemi Covid-19: Perspektif Kenormalan Baru. *JPK (Jurnal Pancasila dan Kewarganegaraan)*, 5(2), pp.65-76.
- [3] Gusty, S., Nurmiati, N., Muliana, M., Sulaiman, O.K., Ginantra, N.L.W.S.R., Manuhutu, M.A., Sudarso, A., Leuwol, N.V., Apriza, A., Sahabuddin, A.A. and Hastuti, P., 2020. Belajar Mandiri: Pembelajaran Daring di Tengah Pandemi Covid-19. Yayasan Kita Menulis.
- [4] Purnomo, D., 2017. Model Prototyping Pada Pengembangan Sistem Informasi. *JIMP-Jurnal Informatika Merdeka Pasuruan*, 2(2).
- [5] Sutabri, T., 2012. Analisis sistem informasi. Penerbit Andi.
- [6] Tvedt, J.D. and Gollofello, J.S., 1995, August. Evaluating the effectiveness of process improvements on software development cycle time via system dynamics modelling. In *Proceedings Nineteenth Annual International Computer Software and Applications Conference (COMPSAC'95)* (pp. 318-325). IEEE.
- [7] Smith, P.G. and Reinertsen, D.G., 1992. Shortening the product development cycle. *Research-Technology Management*, 35(3), pp.44-49.
- [8] Dwanoko, Y.S., 2016. Implementasi Software Development Life Cycle (SDLC) Dalam Penerapan Pembangunan Aplikasi Perangkat Lunak. *Jurnal Teknologi Informasi: Teori, Konsep, dan Implementasi*, 7(2), p.143003.
- [9] Bassil, Y., 2012. A simulation model for the waterfall software development life cycle. *arXiv preprint arXiv:1205.6904*.
- [10] Sutanta, E., 2004. Evaluasi Rancangan Struktur Database. *Dinamik*, 9(2).
- [11] Lubis, A., 2016. *Basis Data Dasar*. Deepublish.
- [12] Yanto, R., 2016. *Manajemen Basis Data Menggunakan MySQL*. Deepublish.
- [13] Jayanti, N.K.D.A. and Sumiari, N.K., 2018. *Teori Basis Data*. Penerbit Andi.
- [14] Herlawati, P.P.W., 2011. *Menggunakan UML*. Bandung: Informatika.
- [15] Booch, G., Maksimchuk, R.A., Engle, M.W., Young, B.J., Connallen, J. and Houston, K.A., 2008. Object-oriented analysis and design with applications. *ACM SIGSOFT software engineering notes*, 33(5), pp.29-29.
- [16] Rizky, S., 2011. *Konsep dasar rekayasa perangkat lunak*. Jakarta: Prestasi Pustaka.