

Sistem Informasi Akademik Berbasis Web Dengan Implementasi HMVC (Hierarchical, Model, View, Controller) (Studi Kasus: Smk Muhammadiyah 8 Siliragung Banyuwangi)

Ayu Siska Yuana^{*1}, Aminudin², Ilyas Nuryasin³

^{1,2,3}Teknik Informatika/Universitas Muhammadiyah Malang

siska_437062@webmail.umm.ac.id¹, aminudin2008@umm.ac.id², ilyas@webmail.umm.ac.id³

Abstrak

Sistem informasi akademik ialah sebuah sistem yang mampu memberikan sebuah informasi mengenai segala hal mengenai pelayanan akademik suatu instansi pendidikan. Sistem informasi akademik dapat menjadi solusi dalam upaya meningkatkan sebuah kualitas pendidikan dengan cara memperlancar informasi akademik siswa dengan pihak sekolah sehingga lebih efisien. SMK Muhammadiyah 8 siliragung adalah suatu instansi pendidikan kejuruan dengan matapelajaran adaptif, normative serta produktif sehingga sistem informasi akademik akan sangat menunjang proses pembelajaran di sekolah. Penelitian ini akan membahas sebuah pembangunan sistem informasi akademik SMK Muhammadiyah 8 Siliragung berbasis web dengan mengimplemmentasikan metodel arsitektur HMVC (Hierarchical, Model, View, Controller) yang mampu membuat aplikasi dalam skala besar yang akan dibangun dalam modul-modul yang nantinya setiap modul dapat diimplementasikan kembali ke dalam modul lain ataupun pengembangan. Sistem informasi akademik SMK Muhammadiyah 8 Siliragung dikembangkan dengan pengujian black-box, pengujian UAT (User Acceptance Testing) dan integrasi testing dengan (PHPUnit 8) sebagai sarana untuk menguji model HMVC. Dalam rangka memudahkan dalam pembangunan sistem maka akan digunakannya sebuah metode Waterfall sehingga sistem informasi akan tersusun secara sistematis dan berurutan yang akan meminimalisir sebuah kesalahan.

Kata Kunci: Sistem Informasi Akademik, HMVC (Hierarchical, Model, View, Controller), Pengujian Black-box, UAT (User Acceptance Testing), Integrasi Testing, Metode Waterfall

Abstract

Academic information system is a system that able to provide all the information about everything related to academic services of an educational institution. Academic information system can be a solution (problem solving) in an effort to improve quality of education by accelerating the academic information of students with school side so that, it can be more efisien. Vocational high school of Muhammadiyah 8 Siliragung is a championship educational institution with adaptive, normative and also productive subjects. Therefore, the Academic information system will greatly support the learning process at school. This research will discuss an Academic information system development of Muhammadiyah 8 Siliragung Vocational high school through Web-based by implementing the HMCV (Hierarchical, Model, View, Controller) architectural model that is able to create a large scale application that will be applied in modules. Furthermore, in each module can be implemented back into another module or other development. This research was depeolved with three types of tests as means to testing the HMCV model. They are black-box testing, UAT (User Acceptance Testing) testing and integrity testing with (PHPUnit 8). This research uses the waterfall method in order to make system development easier. Therefore, the information system will be arranged systematically and sequentially which will minimize errors.

Keywords: Academic information system, HMVC (Hierarchical, Model, View, Controller), Black-box testing, UAT (User Acceptance Testing), Integrity Testing, Waterfall method

1. Pendahuluan

SMK Muhammadiyah 8 Siliragung salah satu instansi pendidikan kejuruan swasta yang memiliki kemauan dalam memanfaatkan perkembangan teknologi dengan upaya mengubah sistem manual menjadi sistem yang berbasis komputerisasi, maka sistem yang tepat untuk digunakan dalam instansi pendidikan tersebut ialah "Sistem Informasi Akademik". pembangunan

sistem informasi akademik SMK Muhammadiyah 8 Siliragung akan menjadi solusi besar bagi peningkatan mutu akademik antara lain mutu proses keuangan berjalan, mutu penilaian dan penyampaian nilai, mutu pemberian kehadiran dan penyampaian hasil kehadiran siswa.

Sistem informasi adalah sebuah sarana yang terintegrasi dimana mampu menyediakan informasi yang bermanfaat bagi pengguna sistem tersebut dalam sarana mendukung suatu operasi *management* dalam sebuah instansi pendidikan. Sistem informasi akademik dikembangkan dengan SDLC menggunakan metode *waterfall* dengan ciri pengerjaan per tahap secara urut serta setiap tahapnya yang harus diselesaikan terlebih dahulu sebelum masuk kedalam tahap selanjutnya. Dengan penerapan metode *waterfall* dalam sistem informasi akademik mampu memenuhi syarat sebagai model pengembangan berskala besar yang nantinya menghasilkan pendokumentasian yang baik dengan pengerjaan berkala serta menyeluruh [1].

Sistem informasi akademik akan dibangun dengan mengimplementasikan arsitektur HMVC (*Hierarchical, Model, View, Controller*) ialah sebuah versi dari pengembangan design pater MVC (*Model, View, Control*) ataupun bisa dijelaskan sebagai versi terbaru dari penerapan MVC yang telah terimplementasi dengan baik dan kompleks [2]. HMVC (*Hierarchical Model view controller*) sendiri dapat mendukung pembangunan aplikasi dalam skala besar dengan menawarkan data yang tersusun secara hirarkis dalam setiap foldernya sehingga setiap modul akan tersusun dengan rapi saat dilakukan proses implementasi sistem. Penggunaan arsitektur HMVC dirasa lebih mudah dan lebih fleksibel oleh developer [3].

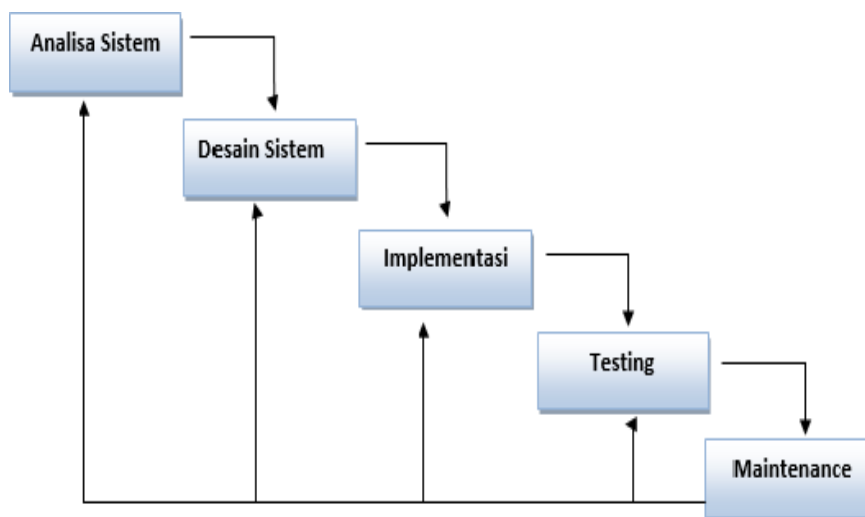
2. Metode Penelitian

2.1 Identifikasi Masalah dan Studi Literatur

Pada tahap identifikasi masalah dan studi literatur dalam menggali permasalahan yang ada pada SMK Muhammadiyah 8 Siliragung Banyuwangi. Saat proses identifikasi masalah dan dengan melibatkan user maka akan di dilakukannya studi literatur seperti halnya dengan penelitian orang lain, jurnal dan dokumen formal sehingga dapat membantu untuk menyelesaikan masalah yang didapat.

2.2 Metode Perancangan

Metode Perancangan yang akan digunakan dalam sistem informasi akademik ini meliputi studi kasus serta literatur yang kemudian akan dilanjutkan dengan proses pengembangan sistem informasi akademik dengan menggunakan metode *waterfall* dengan tahap analisa sistem, desain sistem, implementasi, testing, serta maintenance [4]. seperti ditunjukkan pada Gambar 1.



Gambar 1. Tahapan Metode Waterfall

a. Analisa sistem

Pada Tabel 1, tahap ini dilakukannya analisa sistem dengan cara melakukan interview dengan user yang bersifat terstruktur sehingga pihak pengembang akan menyusun terlebih dahulu pertanyaan-pertanyaan sehingga data yang seharusnya didapat akan tergal dengan baik dan kebutuhan fungsional serta non-fungsional terpenuhi [5].

Tabel 1. Identifikasi Aktor

<i>Actor</i>	<i>Deskripsi</i>
Staff keuangan	Aktor yang dapat mengakses serta melakukan proses (tambah, ubah, hapus dan lihat data) pada data yang ada dalam sistem akademik
Admin	Aktor yang dapat melakukan proses serta memanipulasi data siswa pada (tambah data, edit data, hapus data) terhadap data siswa, data guru, data kelas serta data mata pelajaran pada sistem informasi akademik SMK Muhammadiyah 8 Siliragung. Aktor yang dapat melakukan (view data kehadiran, view data nilai, view data keuangan) pada sistem informasi akademik SMK Muhammadiyah 8 Siliragung
Siswa	Aktor yang dapat mengakses (view keuangan, view nilai dan view kehadiran) terhadap data sistem informasi akademik (siswa) SMK Muhammadiyah 8 Siliragung
Guru	Aktor yang dapat melakukan proses serta manipulasi data seperti halnya (tambah data, edit data, hapus data). terhadap data sistem akademik (kehadiran siswa, nilai siswa)

Menurut Tabel 2, tahapan yang ada pada penelitian dimana bermaksud dalam menggali kebutuhan sistem. Maka digunakannya metode MDI (*Mandatory Desirable Inessential*) untuk proses elisitasi tahap pertama kemudian metode TOE (Teknikal, Operasional, Ekonomi) untuk elisitasi tahap kedua dalam menganalisis kebutuhan sistem [6].

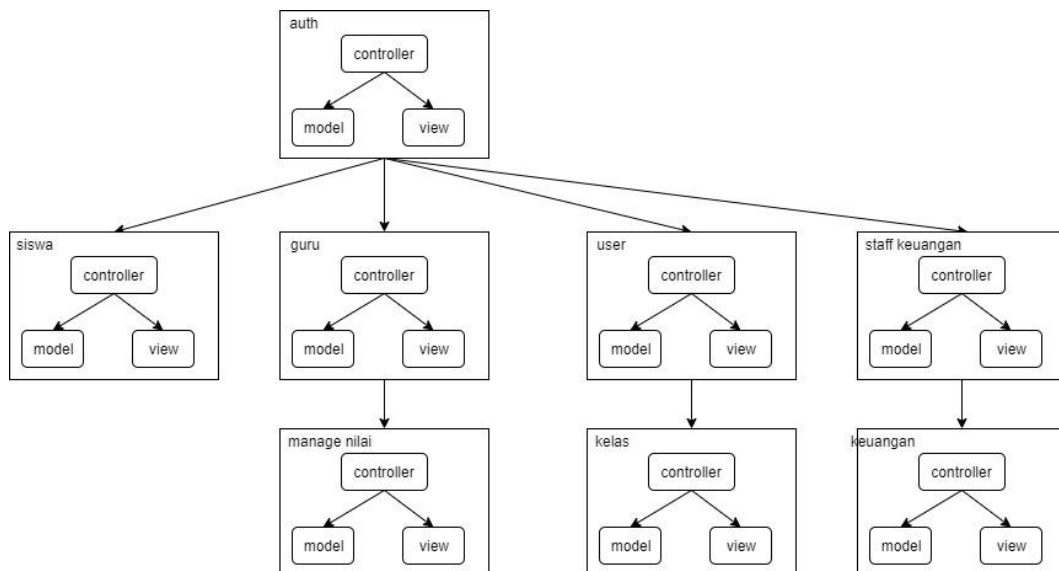
Tabel 2. Kebutuhan Fungsional

No.	Kebutuhan	Aktor
1	Sistem mampu menerima inputan <i>login</i>	admin, staff keuangan, siswa, guru
2	Sistem mampu menampilkan data keuangan berdasarkan jenis pembayaran	admin, staff keuangan, siswa
3	Sistem mampu menampilkan <i>form</i> pengisian data keuangan	staff keuangan
4	Sistem mampu menambah, merubah dan menghapus data keuangan	staff keuangan
5	Sistem mampu menampilkan data siswa berdasarkan kelas	Admin, staff keuangan, siswa, guru
6	Sistem mampu menampilkan <i>form</i> pengisian data siswa	Admin
7	Sistem mampu menambah, merubah dan menghapus data siswa berdasarkan kelas	Admin
8	Sistem mampu menampilkan data guru	Admin, guru
9	Sistem mampu menampilkan <i>form</i> pengisian data guru	Admin, guru
10	Sistem mampu menambah, mengupdate, dan menghapus data guru	Admin, guru
11	Sistem mampu menampilkan data kelas sesuai dengan jurusan	Admin, siswa, guru
12	Sistem mampu menampilkan <i>form</i> pengisian data kelas	Admin
13	Sistem mampu menambah, merubah dan menghapus data kelas sesuai dengan jurusan	Admin

14	Sistem mampu menampilkan data nilai siswa sesuai mata pelajaran	Admin, siswa, guru
12	Sistem mampu menampilkan <i>form</i> pengisian data kelas	Admin
13	Sistem mampu menambah, merubah dan menghapus data kelas sesuai dengan jurusan	Admin
14	Sistem mampu menampilkan data nilai siswa sesuai mata pelajaran	Admin, siswa, guru
15	Sistem mampu menampilkan <i>form</i> pengisian data nilai	Admin, guru
16	Sistem mampu menambah, merubah dan menghapus data nilai siswa sesuai matapelajaran	Admin, guru
17	Sistem mampu menampilkan kehadiran siswa sesuai mata pelajaran	Admin, siswa, guru
18	Sistem mampu menampilkan <i>form</i> pengisian data kehadiran siswa	Admin, guru
19	Sistem mampu menambah, merubah, dan menghapus data kehadiran siswa sesuai matapelajaran	Admin, guru
20	Sistem mampu menerima inputan <i>logout</i>	Admin, staff keuangan, siswa, guru

b. Desain Arsitektur Sistem

Desain arsitektur sistem ialah tahap dimana sistem akan dilakukannya perancangan struktur pada sistem informasi akademik dengan menggunakan UML (*Unified Modelling Language*) dengan menggambar kan arsitektur hmvc sistem, serta *usecase Diagram* [7]. Model Gambar 2 pada HMVC dapat dijabarkan menjadi struktur data dari sebuah sistem yang akan dikerjakan, didalamnya terdapat akses file teks atau sejenis xml, selain hal itu model berisikan class dan fungsi yang akan memiliki hubungan dengan akses ke dalam database [8].



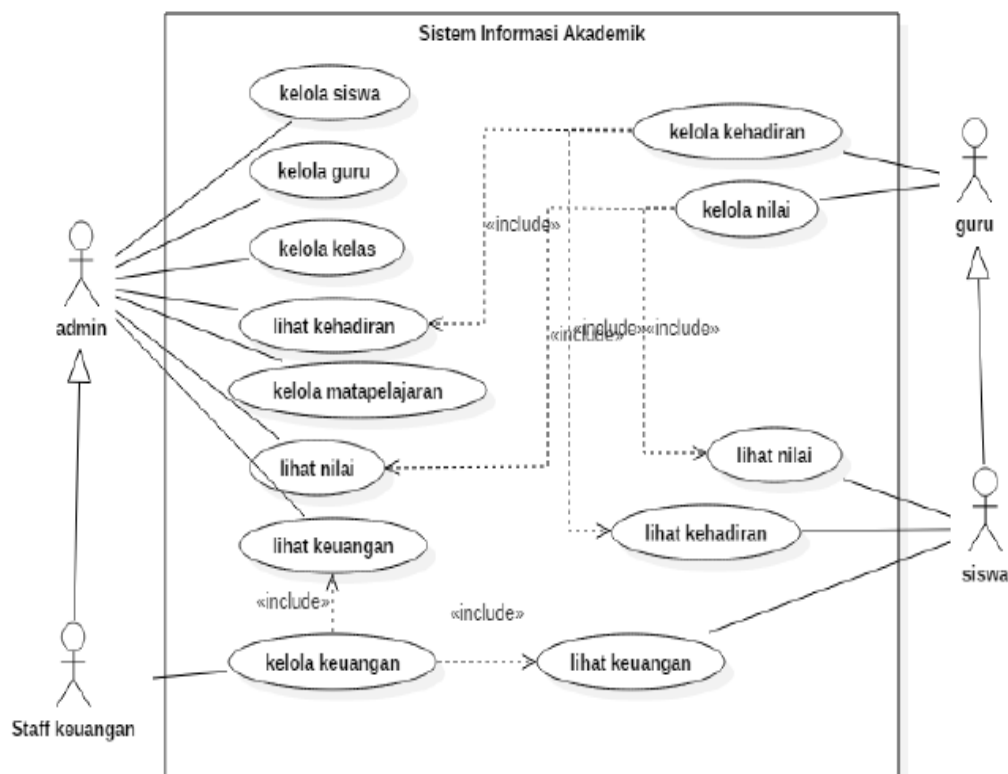
Gambar 2. Arsitektur HMVC pada Sistem

View pada pembuatan sistem ialah sebuah wadah yang berisikan informasi yang nantinya bisa ditampilkan kepada user, view dalam HMVC disini tidak berisikan code-code seperti halnya *controller* dan model melainkan hanya berisi variabel-variabel data yang nantinya akan

ditampilkan. Controller pada HMVC merupakan bagian paling krusial dari sistem karena dengan adanya controller maka model dan view akan dihubungkan satu sama lain (penghubung) maka dari itu didalam controller terdapat class serta fungsi-fungsi yang nantinya akan memproses dari view HMVC ke dalam struktur data yang telah ada pada bagan model.

c. Usecase Diagram

Untuk mempermudah pemahaman terhadap kebutuhan fungsional yang ada, digunakan *usecase diagram* pada Gambar 3 untuk menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Tujuan utama dari *usecase diagram* adalah untuk membantu penulis memvisualisasikan persyaratan fungsional suatu sistem, termasuk hubungan "aktor" (manusia yang akan berinteraksi dengan sistem) dengan proses-proses penting, serta hubungan di antara *usecase* yang berbeda-beda.



Gambar 3. Usecase Diagram

d. Pengujian

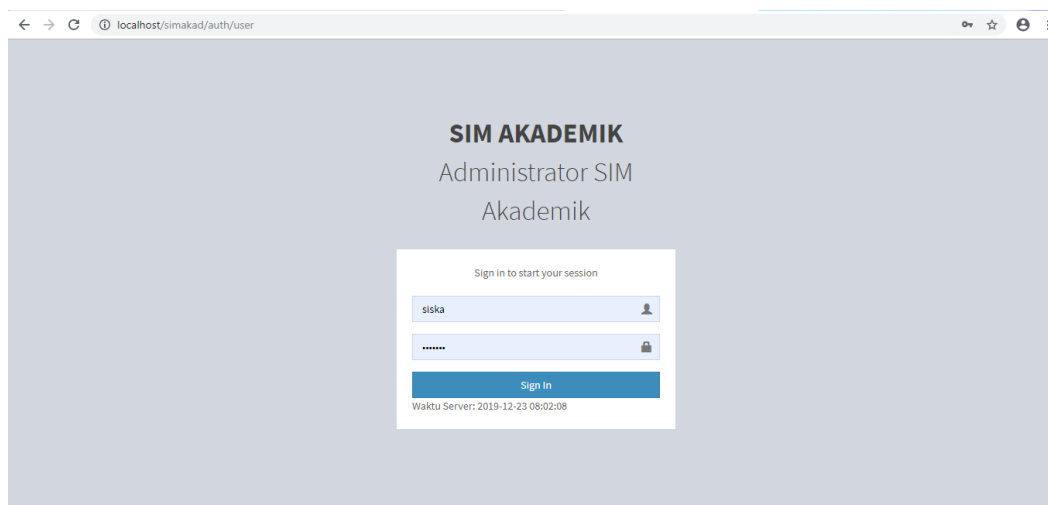
Pada ini ialah tahap implementasi yang telah terselesaikan, kemudian tahap berikutnya akan dilakukannya sebuah implementasi pengujian yang dilakukan pada fungsi dari sistem dan mencari kekurangan serta memeriksa sistem, sehingga meminimalisir kesalahan yang tidak diinginkan. Berikut ialah tahap pengujian Sistem Informasi Akademik SMK Muhammadiyah 8 Siliragung banyuwangi dengan menggunakan 3 model pengujian :

1. Model pengujian dengan menggunakan PHPUnit, yaitu sebuah pengujian *framework* dengan bahasa pemrograman PHP memanfaatkan *library* PHPUnit, PHPUnit tetsing digunakan untuk menguji hasil dari integrasi antara komponen-komponen pada sistem HMVC telah memenuhi syarat atau tidak
2. Model pengujian *black-box*, terfokus pada sebuah persyaratan fungsional dari hasil eksekusi sistem melalui data pengujian, dengan demikian pengujian blackbox memungkinkan sistem mendapatkan serangkaian dari fungsi masukan dan keluarannya
3. Model pengujian UAT (*User Accepted Test*),

4. Hasil dan Implementasi

4.1 Implementasi Sistem

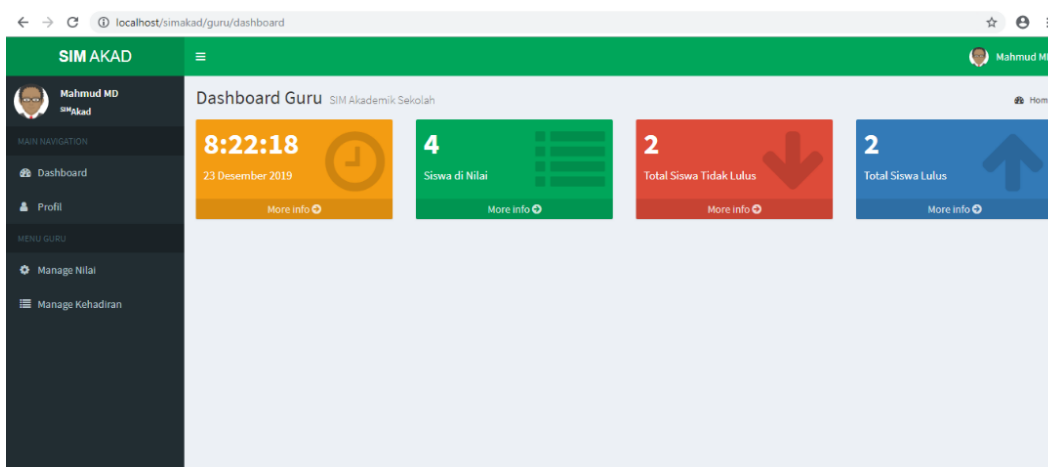
a. Halaman login sistem



Gambar 4. Implementasi Login Sistem

Pada Gambar 4 ialah implementasi sistem login, sistem login diterapkan oleh user admin, staff keuangan, siswa dan guru. Untuk login ke dalam sistem maka user harus memasukkan *username* serta *password*.

b. Dashboard Admin

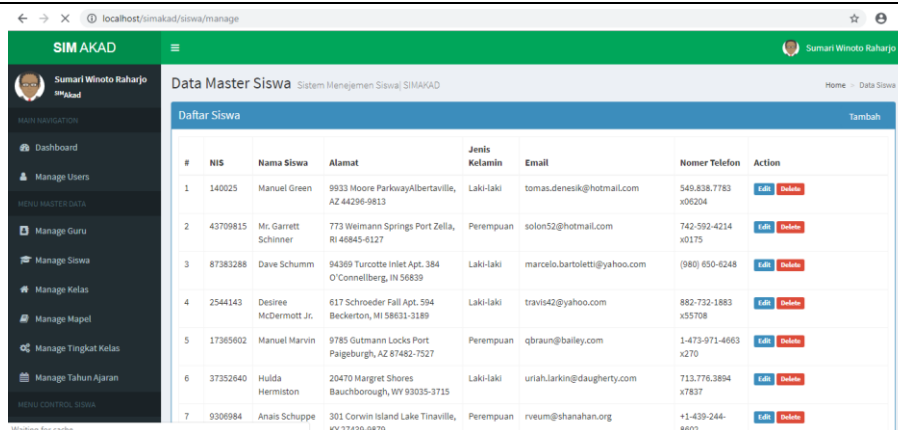


Gambar 5. Dashboard Admin

Pada Gambar 5 ialah implementasi sistem dashboard pada guru, yang terlihat dari implementasi tersebut guru hanya melakukan kelola nilai serta kelola kehadiran. Guru hanya menilai berdasarkan matapelajaran yang diampu dan memberikan kehadiran untuk matapelajaran yang diampu.

c. Halaman Data Master Pada Admin

Gambar 6 ialah implementasi sistem kelola data siswa pada proses ini data siswa mampu dikelola untuk proses tambah data, hapus data, edit data serta lihat data itu sendiri. Pada proses kelola data siswa user yang memiliki wewenang adalah admin.



Gambar 6. Implementasi Halaman Data Master Pada Admin

4.2 Pengujian HMVC dengan PHPUnit CodeIgniter

PHPUnit testing digunakan untuk menguji hasil dari integrasi antara komponen-komponen pada sistem HMVC telah memenuhi syarat atau tidak [9]. Berikut ialah pengujian integrasi testing dengan menggunakan PHPUnit seperti pada Tabel 3.

Tabel 3. Hasil Pengujian Menggunakan PHPUnit CodeIgniter

ID Pengujian	Deskripsi Pengujian	Prosedur pengujian sistem	Hasil Pengujian
A.1	Pengujian modul admin		
A.1.1	Modul admin Login dan Logout	testLoginAsAdmin() testLogoutAdmin()	Sukses Sukses
A.1.2	Modul admin kelola data siswa	testInsertSiswa() testUpdateSiswa() testViewSiswa() testDeleteSiswa()	Sukses Sukses Sukses Sukses
A.1.3	Modul admin kelola data guru	testInsertGuru() testUpdateGuru() testDeleteGuru() testViewGuru()	Sukses Sukses Sukses Sukses
A.1.4	Modul admin kelola data kelas	testInsertKelas() testEditKelas() testHapusKelas() testViewKelas()	Sukses Sukses Sukses Sukses
A.1.5	Modul admin view data keuangan	testViewKeuangan()	Sukses
A.2	Pengujian modul guru		
A.2.1	Modul guru login dan logout	testLoginAsGuru() testLogoutGuru()	Sukses Sukses
A.2.2	Modul pengujian guru kelola data nilai	testInputNilai() testEditNilai() testHapusNilai() testViewNilai()	Sukses Sukses Sukses Sukses
A.2.3	Modul pengujian guru kelola data kehadiran	testInputKehadiran() testeditkehadiran() testViewKehadiran() testKehadiran()	Sukses Sukses Sukses Sukses
A.3	Pengujian modul siswa		
A.3.1	Modul siswa login dan logout	testLoginAsSiswa() testLogoutSiswa()	Sukses Sukses

A.3.2	Modul siswa view data nilai	testViewNilai()	Sukses
A.3.3	Modul siswa view data keuangan	testViewKeuangan()	Sukses
A.3.4	Modul siswa view data kehadiran	testViewKehadiran()	Sukses
A.4	Pengujian Modul Keuangan		
A.4.1	Modul keuangan login dan logout	testLoginKeuangan() testLogoutKeuangan()	Sukses Sukses
A.4.2	Modul keuangan kelola data keuangan	testLogoutKeuangan() testInputkeuangan()	Sukses Sukses

4.3 Pengujian Blackbox

Dalam Tabel 4, pengujian blackbox akan terfokus pada sebuah persyaratan fungsional dari hasil eksekusi sistem melalui data pengujian, dengan demikian pengujian blackbox memungkinkan sistem mendapatkan serangkaian dari fungsi masukan dan keluarannya telah sesuai dengan spesifikasi dari *requirement* data atau yang sepenuhnya menggunakan semua persyaratan fungsional untuk sistem yang telah dibangun [10].

Tabel 4. Pengujian Blackbox

No	Pengujian	Fungsi	Data	Hasil Pengujian
1	Login	Dapat memvalidasi username dan password	Pengguna (admin, siswa, guru staff keuangan)	Sukses
2	Halaman utama admin	Menampilkan dashboard admin	Admin	Sukses
3	Halaman utama siswa	Menampilkan dashboard siswa	Siswa	Sukses
4	Halaman utama guru	Menampilkan dashboard guru	Guru	Sukses
5	Halaman utama staff keuangan	Menampilkan dashboard taff keuangan	Staff keuangan	Sukses
6	Memilih menu data keuangan	Menampilkan data keuangan	Data keuangan	Sukses
7	Memilih button tambah data keuangan	Menampilkan form tambah data	Hasil transaksi keuangan	Sukses
8	Memilih button hapus data	Hapus data keuangan	Data keuangan	Sukses
10	Memilih button edit data	Edit data keuangan	Data keuangan	Sukses
11	Memilih data siswa	Lihat data siswa	Data keuangan	Sukses
		Hapus data		
		Edit data siswa		
		Tambah data siswa		
12	Memilih data guru	Lihat data guru	Data guru	Sukses
		Hapus data guru		
		Edit data guru		

		Tambah data guru		
		Lihat data kelas		
13	Memilih data kelas	Hapus data kelas	Data kelas	Sukses
		Edit data kelas		
		Tambah data kelas		
		Lihat data guru		
12	Memilih data guru	Hapus data guru	Data guru	Sukses
		Edit data guru		
		Tambah data guru		
		Lihat data guru		
13	Memilih data nilai	hapus data nilai	Data nilai	Sukses
		edit data nilai		
		Tambahd ata nilai		

4.4 UAT (*User Acceptation Test*)

Pengujian UAT (*User Accepted Test*) ialah sebuah pengujian sistem yang saat implementasi telah usai dan dilakukan oleh user admin, siswa, guru staff keuangan. pengujian dilakukan dalam sarana mengamati sistem yang telah di implementasi melalui data dan memeriksa data fungsional [11]. Berikut ialah asil dari pengujian UAT (*User Accepted Test*) yang ditunjukkan pada Tabel 5.

Tabel 5. Pengujian UAT

Id pengujian	Deskripsi pengujian	Prosedur Pengujian	Keluaran yang diharapkan	Hasi Pengujian		
				Di terima	Diterima dengan catatan	Tidak diteri ma
A.1			Kelas pengujian modul akses			
A.1.1			Buat pengujian halaman login			
	Pengujian login	<ul style="list-style-type: none"> Masukkan username dan password Klik button login 	Halam utama user	Admin guru siswa		
A.2			Kelas pengujian modul akses			
A.2.1			Halaman utama admin			
	Pengujian menampilkan halam utama admin		Halaman utama admin	Admin		
A.3			Kelas pengujian modul akses dashboard			
A.3.1			Halaman utama siswa			
	Pengujian menampilkan halaman utama siswa		Halaman admin	Admin		
A.4			Kelas pengujian modul akses dashboard			
A.4.1			Halaman utama guru			
	Pengujian menampilkan		Halaman utama staff keuangan	Guru		

	halaman utama guru			
A.5		Kelas pengujian modul akses dashboard		
A.5.1		Halamn utama staff keuangan		
	Pengujian menampilkan halaman utama staff keuangan		Halaman utama staff keuangan	Staff keuangan
A.6		Kelas pengujian modul siswa		
A.6.1		Butir pengujian menambahkan data siswa		
	Pengujian menambahkan data siswa	<ul style="list-style-type: none"> Pilih button tambah Masukkan data siswa Pilih simpan 	Tambah data siswa	Admin
A.6.2		Butir pengujian melihat data siswa		
	Pengujian melihat data siswa	Pilih list data siswa	Manage siswa	Admin
A.6.3		Butir pengujian mengedit data siswa		
	Pengujian mengedit data siswa	<ul style="list-style-type: none"> Pilih salah satu data Pilih button edit Masukkan edit data siswa 	Manage siswa	Admin
A.6.4		Butir pengujian menghapus data siswa		
	Pengujian menghapus data siswa	<ul style="list-style-type: none"> Pilih salah satu data Pilih utton hapus Klik OK 	Manage siswa	Admin
A.7		Kelas pengujian modul		
A.7.1		Butir pengujian lihat data guru		
	Pengujian melihat data guru	Pilih list data guru	Manage guru	Admin
A.7.2		Butir pengujian tambah data guru		
	Pengujian menambah data guru	<ul style="list-style-type: none"> Pilih button tambah Masukkan data guru Pilih simpan 	Manage guru	Admin
A.7.3		Butir pengujian mengedit data guru		
	Butir pengujian mengedit data guru	<ul style="list-style-type: none"> Pilih salah satu list data guru Pilih button edit Masukkan data edit Pilih simpan 	Manage guru	Admin
A.7.4		Butir pengujian hapus data guru		
	Butir pengujian hapus data guru	<ul style="list-style-type: none"> Pilih salah satu list data guru Pilih button hapus Pilih OK 	Manage guru	Admin
A.8		Kelas pengujian modul data kelas		
A.8.1		Butir pengujian tambah data kelas		
	Butir pengujian menambah data kelas	<ul style="list-style-type: none"> Pilih button tambah Masukkan data kelas Pilih simpan 	Manage kelas	Admin
A.8.2		Butir pengujian melihat data kelas		

	Butir pengujian melihat data kelas	Pilih salah satu list data kelas	Manage kelas	Admin
A.8.3	Butir pengujian mengedit data kelas			
	Butir pengujian edit data kelas	<ul style="list-style-type: none"> • Pilih salah satu data kelas • Pilih button edit • Masukkan data edit • Pilih simpan 	Manage kelas	Admin
A.8.4	Butir pengujian menghapus data kelas			
	Butir pengujian hapus data kelas	<ul style="list-style-type: none"> • Pilih salah satu data kelas • Pilih button hapus • Klik OK 	Manage kelas	Admin
A.9	Kelas pengujian modul data nilai			
A.9.1	Butir pengujian melihat data nilai			
	Butir pengujian lihat data nilai	Pilih salah satu list data kelas	Manage data nilai	guru
A.9.2	Butir pengujian menambah data nilai			
	Butir pengujian tambah data nilai	<ul style="list-style-type: none"> • Pilih button tambah • Masukkan data nilai • Pilih simpan 	Manage data nilai	guru
A.9.3	Butir pengujian mengedit data nilai	<ul style="list-style-type: none"> • Pilih salah satu data nilai • Pilih button edit • Masukkan data edit nilai • Pilih button simpan 	Manage data nilai	Guru

5. Kesimpulan

Adapun kesimpulan dari sistem yang telah diimplementasikan serta di lakukan pengujian sistem pada sistem informasi akademik, yaitu:

1. User yang menggunakan aplikasi akademik SMK Muhammadiyah 8 Siliragung ini terdiri dari 4 aktor, dimana aktor tersebut adalah admin, siswa, guru dan staff keuangan.
2. Pengembangan sistem informasi akademik berbasis web dilakuka dengan menggunakan Framework Codeigniter dengan mengimplemmentasikan arsitektur HMVC.
3. Dari segi fitur pengembangan yang dilakukan adalah fitur kelola data siswa, kelola data guru, kelola data nilai, kelola data keuangan, kelola data kehadiran.
4. Semua fitur aplikasi berjalan dengan sukses sesuai dengan fungsinya.

6. Saran

Adapun saran yang dapat diberikan untuk penelitian selanjutnya guna menjadikan aplikasi sistem informasi akademik ini menadi lebih baik, diantaranya ialah.

1. Bisa ditambahkan notifikasi ketika sudah di updatenya nilai siswa, data kehadiran ataupun data keuangan siswa maka siswa akan bias mengetahui dengan cepat.
2. Bisa diimplementasikan dengan arsitektur sistem selain HMVC dengan *Framework CodeIgniter*
3. Pengujian sistem dapat dilakukan dengan pengujian yang lain selain PHPunit, Blax-box dan UAT (*User Accepted Test*).

Referensi

- [1] W. S. Dharmawan, D. Purwaningtias, and D. Risdiansyah, "Penerapan Metode SDLC Waterfall Dalam Perancangan Sistem Informasi Administrasi Keuangan Berbasis

- Desktop,” *Penerapan Metod. SDLC Waterfall Dalam Peranc. Sist. Inf. Adm. Keuang. Berbas. Deskt.*, vol. 6, no. 2, pp. 159–167, 2018.
- [2] R. F. Komputer, “Informasi Rumah Sakit Menggunakan Arsitektur Hierarchical Model , View , Controller Berbasis Web,” *Peranc. dan Implementasi Sist. Inf. Rumah Sakit Menggunakan Arsit. HMVC Berbas. Web*, pp. 2–12, 2019.
- [3] P. M. Kepegawaian *et al.*, “Kota Yogyakarta Menggunakan Konsep Hierarchical Model View Controller Kota Yogyakarta Menggunakan View Controller,” *Kota Yogyakarta Menggunakan Konsep Hierarchical Model View Controll. Kota Yogyakarta Menggunakan View Controll.*, vol. 6, 2019.
- [4] Y. Andrika and J. Jend Sudirman Selindung Lama Pangkalpinang Kepulauan Bangka Belitung, “Penerapan Model Waterfall Pada Sistem Informasi Rawat Jalan Dengan Kartu Pasien Ber-Barcode Studi Kasus : Puskesmas Kenanga Sungailiat,” *Penerapan Model Waterfall Pada Sist. Inf. Rawat Jalan Dengan Kartu Pasien Ber-Barcode Stud. Kasus Puskesmas Kenanga Sungailiat*, vol. 1, no. 2, pp. 195–204, 2017.
- [5] “Pengembangan Dan Analisis Kualitas Sistem Informasi Manajemen Dan Bimbingan Skripsi Mahasiswa Jurusan Pendidikan Teknik Elektronika Fakultas Teknik UNY,” *Pengemb. DAN Anal. KUALITAS Sist. Inf. Manaj. DAN Bimbing. SKRIPSI Mhs. Jur. Pendidik. Tek. Elektron. Fak. Tek. UNY*, 2016.
- [6] S. Kurniasari, M. C. Saputra, and A. D. Herlambang, “Evaluasi Kebutuhan Fungsional Sistem Informasi Pelaporan Kegiatan Menggunakan Metode Quality Function Deployment di BNN Kota Malang,” *J. Pengemb. Teknol. Inf. dan Ilmu Komput. Univ. Brawijaya*, vol. 2, no. 12, 2018.
- [7] Yunahar Heriyanto, “Perancangan Sistem Informasi Rental Mobil Berbasis Web Pada PT.APM Rent Car,” *J. Intra-Tech*, vol. 2, no. 2, pp. 64–77, 2018.
- [8] B R Definiujemy, “Perancangan Aplikasi Clickost Berbasis Web Menggunakan Metode HMVC (Hierarchical Model View Controller) Sebagai Sistem Pencarian dan Promosi Indekos Di Kota Padang,” *Peranc. Apl. Clickost Berbasis Web Menggunakan Metode HMVC (Hierarchical Model View Control. Sebagai Sist. Pencarian Dan Promosi Indekos Di Kota Padang*, vol. 4, no. 1, pp. 1–12, 2017.
- [9] Unjani, “Pengembangan dan analisis kualitas sistem manajemen dan bimbingan skripsi mahasiswa jurusan pendidikan teknik elektro fakultas teknik UNV,” *Pengayakan*, no. 37, pp. 1–4, 2017.
- [10] F. G. john Rupilele, “Perancangan Sistem Informasi Manajemen Pelayanan Anggota Jemaat, Baptisan, dan Pernikahan Berbasis Web (Studi Kasus: Gekari Lembah Pujian Kota Sorong),” *Peranc. Sist. Inf. Manaj. Pelayanan Anggota Jemaat, Baptisan, Dan Pernikahan Berbas. Web (Studi Kasus Gekari Lembah Pujian Kota Sorong)*, vol. 5, no. 2, p. 147, 2018.
- [11] W. S. Wardhono, L. P. Kusuma, W. S. Wardhono, and K. Malang, “Evaluasi User Acceptance Augmented Reality Triage Mobile Pada Sistem Kedaruratan Medis,” *Eval. USER Accept. Augment. Real. Triage Mob. Pada Sist. Kedaruratan Medis*, vol. 6, no. 238, pp. 978–979, 2015.