

BAB II KAJIAN PUSTAKA

2.1 Penelitian Relevan

Sebagai pedoman atau landasan, penelitian ini mengacu pada penelitian-penelitian sebelumnya agar dapat mengembangkan seperti yang diharapkan oleh peneliti-peneliti sebelumnya, dan juga sebagai pandangan sistem yang akan dibuat. Berikut beberapa penelitian relevan yang dijadikan acuan untuk penelitian ini.

(Irmayani & Munandar, 2020) Melakukan penelitian yang berjudul “**Sistem Informasi Pengolahan Data Siswa Pada SMA Negeri 02 Bilah Hulu Berbasis Web**”. Latar belakang masalah pada penelitian ini adalah jumlah siswa yang semakin meningkat mendorong SMA Negeri 02 Bilah Hulu untuk terus meningkatkan kualitas layanan dan administrasi baik dalam bidang akademik maupun non-akademik. Pengelolaan data siswa yang dilakukan secara manual tentu sangat tidak efektif jika masih tetap dilakukakan hingga saat ini. Sistem pengolahan data yang dilakukan melalui pembukuan menimbulkan kesulitan bagi petugas pengelola data tersebut. Tujuan dari penelitian ini adalah membuat sistem informasi untuk pengolahan data siswa sehingga menghasilkan proses pengolahan data siswa yang cepat, mudah dan akurat. Metode yang digunakan dalam penelitian ini adalah *System Development Life Cycle (SDLC)* atau *waterfall* yang meliputi analisis, desain, implementasi, uji coba dan pengelolaan. Adapun metode pengumpulan data dilakukan dengan metode observasi, wawancara dan studi kepustakaan. Untuk membangun sistem informasi tersebut peneliti menggunakan PHP dan MySQL. Yang dihasilkan dari penelitian ini adalah sebuah sistem informasi yang dapat meningkatkan kinerja dalam proses pengolahan dan pencarian data siswa yang dibutuhkan kapan saja di SMA Negeri 02 Bilah Hulu berbasis web.

Persamaan penelitian diatas dengan penelitian ini yaitu terletak pada tujuan penelitian yaitu menghasilkan sistem informasi pengolahan data siswa berbasis web dan metode penelitiannya sama-sama menggunakan *waterfall*. Sedangkan perbedaannya terletak pada objek penelitian, yaitu penelitian diatas tentang

pengolahan data siswa yang terdapat pada suatu sekolah sedangkan penelitian ini adalah pengolahan data santri yang mencakup data siswa pada beberapa sekolah yang terdapat pada suatu lembaga pesantren. Selain itu, penelitian di atas menggunakan bahasa pemrograman PHP sedangkan pada penelitian ini menggunakan bahasa pemrograman *python* dengan *framework django*.

(Prabandari, 2019) Melakukan penelitian yang berjudul “**Sistem Informasi Buku Induk Siswa**”, dengan latar belakang saat ini sistem pengolahan data buku induk siswa masih banyak dilakukan secara manual, yaitu dengan menggunakan buku berukuran besar sebagai media pencatatan, sehingga pengisian buku induk membutuhkan banyak waktu dan tenaga, membutuhkan tempat khusus untuk penyimpanan, membutuhkan waktu yang cukup lama untuk mencari dan mencocokkan data serta buku mudah rusak apabila disimpan terlalu lama. Tujuan penelitian ini yaitu membuat sistem informasi buku induk siswa berbasis *web* sehingga dengan demikian dapat meminimalkan pengarsipan berkas-berkas buku induk siswa menjadi berkas-berkas *file* yang disimpan di dalam komputer. Untuk membangun sistem informasi tersebut peneliti menggunakan bahasa pemrograman PHP, *framework bulma* dan MySQL sebagai *database*. Hasil dari penelitian ini berupa sistem informasi buku induk siswa yang dapat mempermudah proses penyimpanan dan pencarian data.

Persamaan penelitian diatas dengan penelitian ini terletak pada tujuan penelitian yaitu membuat sistem informasi buku induk siswa berbasis *web*. Sedangkan perbedaannya terletak pada objek penelitian, yaitu penelitian diatas tentang pengolahan data siswa yang terdapat pada suatu sekolah sedangkan penelitian ini adalah pengolahan data santri yang mencakup data siswa pada beberapa sekolah yang terdapat pada suatu lembaga pesantren. Selain itu, penelitian diatas menggunakan bahasa pemrograman PHP dengan *framework bulma* sedangkan pada penelitian ini menggunakan bahasa pemrograman *python* dengan *framework django*.

(Ardian et al., 2019) Melakukan penelitian yang berjudul “**Perancangan Aplikasi Pengolahan Data Siswa Berbasis Android (Studi Kasus: MIS Nurul Huda Labuhan Batu Selatan)**”. Latar belakang masalah pada penelitian ini adalah dalam proses menyajikan pelayanan untuk para siswa mengalami

kesulitan, mulai dari pelayanan dalam penginputan data, penyimpanan data, dan yang paling utama yaitu proses pencarian data para siswa. Hal ini dikarenakan pada saat melakukan pencarian data para siswa tersebut petugas harus mengecek dari berkas-berkas yang ada dilaptop petugas yang tersusun dalam file-file yang terlalu banyak, sehingga memperlambat pekerjaan dari petugas dalam pencarian data siswa, bahkan kehilangan dan kerusakan data akibat serangan *virus* juga dapat terjadi. Tujuan penelitian ini adalah menghasilkan aplikasi yang dapat membantu dan meningkatkan pengolahan data dengan lebih mudah, dalam hal ini pengolahan data siswa. Metode yang digunakan dalam penelitian ini menggunakan *Unified Modeling Language* (UML) untuk perancangan sistemnya. Adapun metode pengumpulan data dilakukan dengan metode observasi, wawancara dan studi kepustakaan. Hasil akhir dari penelitian ini adalah sebuah aplikasi pengolah data yang dapat meningkatkan kinerja dalam proses pengolahan dan pencarian data siswa berbasis *android*.

Persamaan penelitian diatas dengan penelitian ini terletak pada permasalahan penelitian yaitu pengolahan data siswa dan tujuan penelitiannya sama-sama untuk menghasilkan aplikasi pengolahan data siswa. Sedangkan perbedaannya terletak pada objek penelitian, yaitu penelitian diatas tentang pengolahan data siswa yang terdapat pada suatu sekolah sedangkan penelitian ini adalah pengolahan data santri yang mencakup data siswa pada beberapa sekolah yang terdapat pada suatu lembaga pesantren. Perbedaan selanjutnya yaitu bahasa pemrograman yang digunakan, penelitian diatas menggunakan bahasa pemrograman *android* sedangkan penelitian ini menggunakan bahasa pemrograman *python*. Metode penelitian yang digunakan pada penelitian sebelumnya menggunakan menggunakan *Unified Modeling Language* (UML) untuk perancangan sistemnya sedangkan pada penelitian ini menggunakan *waterfall*. Selain itu, penelitian diatas menghasilkan aplikasi berbasis *android* yang fokus untuk meningkatkan kinerja dalam proses pengolahan dan pencarian data siswa, sedangkan pada sistem yang dikembangkan pada penelitian ini menghasilkan sistem untuk pengolahan data santri secara terpusat agar tidak terjadi tidak sinkron data dan data tersimpan dalam satu *database*.

Berdasarkan penelitian-penelitian diatas, dapat disimpulkan bahwa sistem pengolahan data buku induk dapat membantu pengolahan data buku induk yang efektif dan efisien sehingga dapat menghasilkan data yang mudah diakses. Oleh karena itu, dalam penelitian ini akan dikembangkan sebuah aplikasi untuk membantu dalam pengolahan data buku induk santri terpusat di Pondok Pesantren Nurul Qadim sehingga dapat menghasilkan data yang valid dan mudah diakses, serta untuk menghindari terjadinya tidak sinkron data. Aplikasi yang akan dikembangkan dalam penelitian ini menggunakan *framework django* dan MySQL sebagai *database*.

Alasan mengapa penelitian ini menggunakan *django* karena *django* merupakan *web framework* yang berbasis bahasa pemrograman *python*, dimana *python* dikenal memiliki sintaksis yang sederhana dan mudah dimengerti. *Django* juga cocok untuk berbagai macam aplikasi web, karena *django* juga merupakan *web framework* yang sangat cepat dan mudah digunakan sehingga bisa dianjurkan bagi pemula yang hendak membuat *website*.

Satu hal lagi yang membuat *framework* ini populer untuk digunakan. *Django* memiliki fitur admin otomatis, dimana sudah disediakan *dashboard* untuk mengolah data-data aplikasi yang akan dibuat nantinya.

2.2 Landasan Teori

2.2.1 Buku Induk

Menurut Kamus Besar Bahasa Indonesia (KBBI, 2016) buku induk merupakan suatu buku yang memuat daftar nama beserta jati diri yang tercatat sebagai siswa suatu sekolah. Buku induk siswa juga merupakan dokumen penting sekolah untuk disimpan dan didokumentasikan, sehingga data siswa tersebut tetap ada kapan saja dibutuhkan, bahkan saat siswa berstatus sebagai alumni.

2.2.2 Santri

Santri merupakan istilah atau sebutan atau julukan untuk peserta didik yang sedang mendalami atau menuntut ilmu keagamaan dan tinggal di dalam suatu lembaga pesantren. (Rahmawati, 2015).

2.2.3 Aplikasi

Menurut (Nurcahyo, 2018) aplikasi merupakan sekumpulan *file* komputer yang saling berhubungan dengan tujuan untuk melakukan aktivitas tertentu. Aplikasi dikelompokkan berdasarkan *platform* aplikasi tersebut dijalankan seperti: aplikasi *web* atau *website*, aplikasi *android* dan aplikasi *desktop*.

2.2.4 Website

Menurut (Abdulloh, 2018) *website* merupakan sekumpulan halaman *web* yang saling tergabung dalam satu *domain* atau URL (*Uniform Resource Locator*) dan dapat diakses secara publik melalui *internet*. *Website* atau yang juga dikenal dengan istilah *web* memuat berbagai informasi data *digital* seperti gambar, animasi, *teks*, *video*, *audio* atau bahkan gabungan dari semuanya. HTML (*Hypertext Markup Language*) merupakan bahasa standar yang digunakan dalam pembuatan *website* yang kemudian diterjemahkan oleh *web browser*, sehingga menjadi suatu tampilan informasi yang dapat dilihat dan dibaca oleh semua orang.

Menurut (Abdulloh, 2018) terdapat tiga jenis *website* yaitu *website* statis, *website* dinamis dan *website* interaktif.

1. *Website* statis merupakan *website* yang halamannya tidak berubah atau tidak dapat diperbaharui isinya. *Website* statis biasanya hanya digunakan pada halaman profil pemilik *website* saja. Contoh dari *website* statis yaitu *web company* profil perusahaan.
2. *Website* dinamis merupakan *website* yang halamannya dapat berubah, artinya isi dari *website* dapat diperbaharui oleh pemilik atau pengelola *website*. *Website* dinamis biasanya digunakan oleh perusahaan atau perorangan yang memiliki kegiatan maupun bisnis yang berkaitan dengan *internet*. Contoh dari *website* dinamis yaitu *website* berita seperti *kompas.com*.
3. *Website* interaktif, sama halnya dengan *website* dinamis hanya saja yang berhak memperbaharui isi dari *website* tidak hanya oleh pemilik atau pengelola *website*, melainkan juga oleh pengguna *website*. Contoh dari *website* interaktif yaitu *website marketplace* dan jejaring sosial seperti *shopee*, *bukalapak*, *facebook* dan *instagram*.

2.2.5 Python

Python merupakan salah satu bahasa pemrograman yang populer untuk membuat berbagai macam program, seperti program GUI atau *desktop*, CLI, aplikasi *mobile*, *web*, IoT dan program *hacking*. Bahasa pemrograman ini juga banyak digunakan oleh kalangan *engineer* seperti *Instagram*, *YouTube*, *NASA* dan *Google*. *Python* juga bersifat *open source*, gratis dan *multiplatform*, hal ini yang membuat beberapa *programmer* menyukai *python* (Wardana, 2019).

Menurut (Wardana, 2019) *python* memiliki beberapa kelebihan antara lain:

1. *Python* merupakan bahasa pemrograman terpopuler di dunia di urutan ke tiga, per September 2018.
2. *Python* lebih mudah dipelajari dibandingkan bahasa pemrograman lain. Dengan sintaksisnya yang sederhana, mudah dibaca dan dipahami.
3. Dapat membuat dan mengembangkan berbagai macam program seperti program berbasis *web*, *desktop*, *mobile*, *video game*, robotika, data mining dan kecerdasan buatan.
4. Memiliki banyak dukungan *library* standar dengan berbagai modul dan ekstensi program yang siap dipakai dalam membuat program.

2.2.6 Framework

Framework atau kerangka kerja merupakan kumpulan dari *script* seperti *class* dan *function* yang digunakan oleh para *programmer* untuk membantu dalam membuat program. *Framework* dapat mempermudah dan mempercepat para *programmer* dalam membangun aplikasi, tanpa harus membuat fungsi atau *class* dari awal lagi (Mualim & Putra, 2017).

2.2.7 Django

Django merupakan *web framework* tingkat tinggi bersifat *open source* yang menggunakan bahasa pemrograman *python* (Django, 2020). *Django* merupakan *web framework python* yang didesain untuk membuat aplikasi *web* yang dinamis, kaya fitur dan aman. *Django* didukung dan dikembangkan oleh the *Django Software Foundation* yaitu sebuah organisasi yang memelihara dan mengembangkan *django*, sehingga membuat *web framework* ini disukai dan

menjadi pilihan utama bagi banyak developer atau pengembang (Saputra & Aji, 2018). Saat ini banyak perusahaan-perusahaan besar yang menggunakan *django* seperti *Instagram* dan *Mozilla* (Dauzon et al., 2016).

Disebutkan juga dalam situsnya <https://www.djangoproject.com/> bahwa *django* memiliki beberapa keunggulan seperti:

1. Sangat cepat, *django* dirancang untuk membantu *developer* atau pengembang dalam membuat aplikasi secepat mungkin.
2. Sangat aman, *django* memiliki keamanan yang kuat dan membantu *developer* atau pengembang menghindari banyak kesalahan keamanan umum, seperti *injeksi SQL*, pembuatan skrip lintas situs, pemalsuan permintaan lintas situs dan pembajakan klik. Sistem otentikasi penggunaannya menyediakan cara yang aman untuk mengelola akun pengguna dan kata sandi.
3. Serbaguna, *django* bisa membangun berbagai macam hal mulai dari sistem manajemen konten, jaringan sosial, hingga *platform* komputasi ilmiah.

2.2.8 MySQL

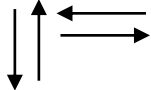
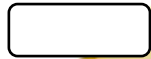



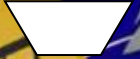




MySQL merupakan suatu perangkat lunak sistem manajemen basis data yang menggunakan perintah dasar SQL (*Structured Query Language*). SQL merupakan bahasa standar *database* yang memudahkan dalam penyimpanan dan akses informasi. Terdapat beberapa kelebihan MySQL seperti *database* ini bersifat gratis, fleksibel dengan berbagai pemrograman dan keamanan yang baik. Dengan berbagai keunggulan tersebut, membuat perangkat lunak database ini banyak digunakan (Solichin, 2016).

2.2.9 Flowchart

Flowchart adalah suatu bagan yang mendeskripsikan sebuah proses, tahapan, dan urutan dengan simbol-simbol tertentu untuk menyelesaikan suatu masalah. Setiap simbol *flowchart* melambangkan suatu pekerjaan. Disebutkan (Astuti & Iftadi, 2016) *flowchart* yang baik dapat menggambarkan semua tahapan proses, dapat menemukan titik proses yang baik, dapat menjelaskan dan

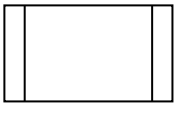
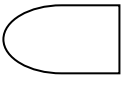
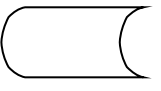



membantu dalam pemecahan permasalahan, serta dapat menggambarkan kesempatan perbaikan yang bisa dilakukan. Simbol-simbol yang digunakan dalam *flowchart* adalah sebagai berikut:

Tabel 2.1. Simbol *Flowchart*

NO	SIMBOL	NAMA	KETERANGAN
1		<i>Flow Direction Symbol</i>	Penghubung antara simbol satu dengan simbol lainnya
2		<i>Terminator Symbol</i>	Mendefinisikan sebagai permulaan atau akhir suatu kegiatan
3		<i>Connector Symbol</i>	Penghubung proses dalam halaman yang sama
4		<i>Connector Symbol</i>	Penghubung proses dalam halaman yang berbeda
5		<i>Processing Symbol</i>	Mendefinisikan proses pengolahan data yang dilakukan komputer
6		<i>Manual Operation</i>	Mendefinisikan proses pengolahan data yang dilakukan manual
7		<i>Decision</i>	Pemilihan proses pada suatu kegiatan
8		<i>Input-Output</i>	Mendefinisikan proses masukan dan keluaran
9		<i>Manual Input</i>	Pemasukan data dengan manual
10		<i>Preperation</i>	Manyiapkan penyimpanan yang diperlukan dalam <i>storage</i>

Sumber (Astuti & Iftadi, 2016)

Tabel 2.2. Simbol *Flowchart* (Lanjutan)

11		<i>Predefine Proses</i>	Pelaksanaan pada suatu prosedur
12		<i>Display</i>	Alat <i>output</i> yang dipakai seperti <i>printer</i> dan layar
13		<i>Disk and On-Line Storage</i>	Mendefinisikan masukan dari disk atau disimpan ke <i>disk</i>
14		<i>Magnetik Tape Unit</i>	Mendefinisikan masukan dari penyimpanan magnetik atau keluaran ke penyimpanan magnetik
15		<i>Puch Card</i>	Mendefinisikan masukan atau keluaran berasal dan ditulis ke kartu
16		<i>Document</i>	Mendefinisikan masukan dan keluaran berbentuk dokumen kertas

Sumber (Astuti & Irtadi, 2016)

2.2.10 Data Flow Diagram (DFD)

Data Flow Diagram adalah gambaran suatu sistem yang sudah ada maupun sistem baru yang dikembangkan secara logika. *Data Flow Diagram* biasa digunakan untuk menggambarkan aliran data dari proses awal sampai sistem selesai. Dengan *Data Flow Diagram* dapat mempermudah pemakai sistem dalam memahami bagaimana sistem tersebut berjalan (Muslihudin & Oktafianto, 2016).

Menurut (Muslihudin & Oktafianto, 2016) simbol-simbol dalam model struktur *Data Flow Diagram* terdapat empat komponen yaitu:

- a. *External Entity*, untuk menggambarkannya sebagai elemen sistem seperti perangkat keras atau *hardware* dan manusia atau *user*.
- b. *Process*, untuk menggambarkan suatu proses aliran data yang dapat dilakukan manusia atau komputer

c. *Data flow*, suatu panah yang mendefinisikan arus data, seperti masuk dan keluarnya data dari suatu proses.

d. *Data store*, mendefinisikan penyimpanan data dalam suatu *database*.

Tabel 2.3. Simbol *Data Flow Diagram*


NO	GANE/ SARSON	YOURDON/ DE MARCO	KETERANGAN
1			<i>External Entity</i> (Entitas Eksternal)
2			<i>Process</i> (Proses)
3			<i>Data Flow</i> (Aliran Data)
4			<i>Data Store</i> (Penyimpanan)

Sumber (Muslihudin & Oktafianto, 2016)

2.2.11 Entity Relationship Diagram (ERD)

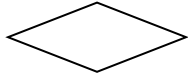


Entity Relationship Diagram adalah model perancangan sistem dengan teknik pendekatan yang menggambarkan hubungan pada suatu model. Dalam hubungan tersebut menunjukkan entitas dan relasi, yang ada pada entitas berikutnya (Fridayanthie & Mahdiati, 2016).

Tabel 2.4. Simbol *Entity Relationship Diagram*

NO	GANE/ SARSON	YOURDON/ DE MARCO	KETERANGAN
1		<i>Entitas</i>	Sesuatu nyata yang mempunyai karakteristik untuk menyimpan data

Sumber(Fridayanthie & Mahdiati, 2016)

Tabel 2.5. Simbol *Entity Relationship Diagram* (Lanjutan)

2		Relasi	Hubungan yang terjadi antara satu atau lebih entitas
3		Atribut	Karakteristik atau ciri umum pada entitas tertentu.
4.		<i>Link</i>	Penghubung antara atribut dengan kumpulan entitas atau kumpulan entitas dengan relasi

Sumber(Fridayanthie & Mahdiati, 2016)

2.2.12 Black Box Testing

Black box testing adalah suatu metode yang digunakan dalam menguji suatu *software* tanpa memperhatikan detail dari *software* tersebut. Metode pengujian ini berfokus terhadap fungsional perangkat lunak tanpa perlu mengetahui kode program yang dipakai (Ningrum et al., 2019). Pengujian *black box* dilakukan dengan percobaan pada perangkat lunak dengan cara memasukkan data ke dalam *form* yang sudah tersedia (Wahyudi et al., 2016).