

BAB II

TINJAUAN PUSTAKA

Penerapan teknologi informasi untuk membantu memenuhi kebutuhan dalam hal mengelola data sudah sangat memungkinkan untuk diterapkan. Khususnya pada pelayanan mengenai manajemen di perguruan tinggi sesuai dengan kebutuhan, sehingga pemanfaatan menggunakan teknologi ini mampu menghasilkan suatu informasi yang tepat, akurat dan tentunya tidak menghambat efektifitas kinerja [6]. Seperti halnya penerapan teknologi pada pengelolaan data perguruan tinggi oleh PDDikti yang terdiri dari data dosen, data mahasiswa dan data perkuliahan, dengan menggunakan aplikasi PDDikti Feeder maka pengelolaan bisa dilakukan secara efektif. PDDikti Feeder merupakan penerapan yang dilakukan oleh Direktorat Jenderal Pendidikan Tinggi (Ditjen Dikti) untuk mengelola data-data tersebut. Ditjen Dikti mewajibkan adanya pelaporan data-data akademik di tiap perguruan tinggi setiap semesternya melalui PDDikti supaya data yang berkaitan dengan pendidikan menjadi lebih bermutu di tingkat nasional [7]. PDDikti Feeder merupakan sebuah *software* yang terdapat di tiap Institut atau Universitas yang tersusun berdasarkan struktur basis data salinan dari basis data PDDikti yang berfungsi sebagai sarana pencatatan resmi penyelenggaraan pendidikan tinggi di Perguruan Tinggi yang berisi data berupa data mahasiswa, mata kuliah dengan bobot penilaiannya, data kelas perkuliahan yang terdiri dari kelas kuliah, KRS mahasiswa, dosen, mata kuliah, dan nilai mahasiswa dan data aktivitas kuliah mahasiswa [8]. Data-data yang terdapat pada PDDikti merupakan data yang akurat dikarenakan melalui proses dari keseluruhan pelaporan akademik yang dilakukan secara berkala yang biasanya dilakukan secara dua kali di tiap semester [9].

Sistem informasi berbasis *website* sudah banyak digunakan dalam lingkup perusahaan yang digunakan sebagai sarana dalam sistem komputerisasi yang sudah dilengkapi dengan berbagai macam fitur dan didesain sedemikian rupa yang tentunya disesuaikan dengan kebutuhan yang mampu mempermudah, mempercepat dan membuat data-data yang diolah menjadi akurat [10]. *Website* adalah tampilan

yang terdapat pada internet yang memuat berbagai informasi yang sesuai dengan kebutuhan pengguna [11]. Oleh karena itu, penggunaan *website* ini cukup berpengaruh dalam hal memberikan kemudahan penyampaian informasi yang bisa dikelola dan diakses secara fleksibel [12]. Dalam hal ini, dibutuhkan penerapan teknologi *web service* supaya *website* yang telah dibuat mampu mengakses layanan yang ada di *web service* yang terdiri dari REST dan API dimana REST artinya arsitektur pada *web service* bersifat klien server dengan klien mengirimkan *request* menuju server yang nantinya server akan memproses *request* tersebut kemudian dikembalikan *responsnya*. Ketika menggunakan teknologi REST terdapat pula *tools* berupa *Application Programming Interface (API)* berbasis *website*. Secara umum, API memiliki dua bagian yang terdiri dari server yang digunakan untuk menyediakan berbagai data kemudian klien yang mampu melakukan *request* data. REST API merupakan API berbasis *website* yang menerapkan teknologi REST dan menggunakan format *JavaScript Object Notation (JSON)* [13]. REST API ini mampu melakukan komunikasi dan mengirim atau menerima data-data dengan cara yang sederhana dimana dengan memanfaatkan HTTP dan *URL* untuk melakukan panggilan pada API-nya. Kemudian, ketika cara kerja pada *web service* dilakukan dengan melakukan navigasi menggunakan *link* ke *link* yang sudah disediakan sehingga pergantian *state* dari halaman web begitu juga untuk REST yang memiliki cara kerja yang sama dengan *web service* sehingga perpindahan halaman bisa dilakukan secara efisien [14]. Pembangunan sistem dengan menggunakan *website* sebagai *platformnya* dan menerapkan teknologi REST API tentunya memudahkan karyawan untuk membangun sistem yang pengerjaannya dilakukan secara berkelompok maupun individu.

Berdasarkan uraian diatas, terdapat kemudahan dalam menerapkan sistem informasi menggunakan *platform website* untuk mendapatkan berbagai macam informasi secara *real-time* [15]. Dengan menggunakan *website*, tentunya juga menggunakan konsep MVC dimana konsep tersebut merupakan salah satu arsitektur yang mampu memisahkan antara tampilan, data dan proses sehingga mampu memudahkan pengembang dalam mengembangkan sebuah sistem. Terlebih lagi, sistem pengelolaan data-data perkuliahan yang memanfaatkan teknologi untuk

mempermudah dan meningkatkan keamanan data di tiap universitas seperti Feeder Universitas Atma Jaya Yogyakarta berbasis *website*. Aplikasi tersebut dibuat menggunakan *framework* .Net Core 6.0 sebagai penghubung antara Siatma Universitas Atma Jaya dengan Neo Feeder PDDikti. Penggunaan *website* menggunakan *framework* ini memudahkan baik dari sisi pengembang dan pengguna karena arsitektur yang mudah untuk dipahami. Kemudian dengan menggunakan Microsoft SQL Server 2018 untuk melakukan pengelolaan basis datanya.

Berdasarkan penelitian, implementasi yang serupa dengan aplikasi untuk mengelola data-data perkuliahan yaitu Sistem Informasi Akademik (SIKA) dengan memanfaatkan teknologi *Single Page Application* (SPA). Penelitian tersebut juga membahas alasan penggunaan teknologi tersebut yaitu untuk mengurangi beban kerja yang berasal dari server, browser dan membangun kinerja aplikasi web yang sifatnya aplikasi desktop melalui proses *Rendering* terhadap keseluruhan data yang berhasil masuk dan akan diterjemahkan dalam bentuk keluaran berupa *output*. SIKA ini juga mewujudkan interoperabilitas di sistem akademik dengan melakukan pertukaran data/informasi melalui suatu protokol *web service* kemudian dikirimkan ke tingkat Universitas kemudian diteruskan ke Feeder PDDikti [16].

Penelitian lain yang serupa dengan aplikasi pengelolaan data-data perkuliahan yaitu SIAKAD. Dengan mengembangkan *web service* kemudian diterapkan pada sistem informasi, maka mampu memberikan kemudahan kepada pengguna ketika melakukan proses penyesuaian data secara *get, insert, update* data melalui aplikasi Feeder PDDikti. Fitur dari aplikasi ini yaitu menu utama yang berisi menu proses, terdapat dashboard header yang berisi informasi sejumlah data yang disesuaikan dengan aplikasi Feeder PDDikti, terdapat dashboard grafik yang berisi grafik dari data yang akan diproses, kemudian terdapat fitur log activity yang berisi aktivitas yang telah dilakukan oleh pengguna [17].

SIAKAT Boyolali merupakan salah satu aplikasi yang serupa dengan aplikasi yang melakukan pengelolaan data-data perkuliahan. Fitur-fitur yang terdapat pada aplikasi ini yaitu menu *dashboard* sebagai menu utama, menu *system*

setting untuk mengatur pengguna, menu master data untuk melakukan *config* data ke Feeder, menu ekspor untuk mengekspor data pada sistem, menu *tool* untuk melakukan pembaharuan dan penghapusan data pada Feeder dan menu admin untuk menampilkan profil admin yang menggunakan sistem tersebut. Dengan adanya aplikasi ini, data berupa tabel mahasiswa, KRS, kurikulum, kelas kuliah, dan nilai yang ada pada Sistem Informasi Akademik Terpadu (SIAKAT) dapat terintegrasi dengan Feeder PDDikti [18].

SIKANTAN ITK merupakan aplikasi yang serupa dengan aplikasi untuk melakukan proses pengelolaan data perkuliahan yang dikirim ke Feeder PDDikti. Aplikasi ini memiliki fitur login yang digunakan admin untuk membuka aplikasi dengan memasukkan nama pengguna dan kata sandi, kemudian terdapat fitur integrasi data ke Feeder PDDikti. Dengan adanya aplikasi ini, perguruan tinggi dapat melakukan migrasi data dengan efisien. Penerapan *web service* juga sangat membantu proses pertukaran data secara otomatis dengan menyediakan kode tokennya, sehingga dengan arsitektur yang dibangun sesuai kebutuhan, informasi terkait sinkronisasi perguruan tinggi bisa sesuai dan lengkap dengan PDDikti [19].

Tabel 2.1. Perbandingan dengan Penelitian Terdahulu

Pembanding	Poetri Lestari Lokapitasari (2018)	Slamet Widodo, dll. (2018)	Erni Widarti, Saiful Bahri, dll. (2018)	M. Gilvy Langgawa, dll. (2021)	Hesti Probodinanti (2022)
Aplikasi	SIAKA	SIAKAD STMIK	SIAKAT Boyolali	SIKANTAN	FEEDER UAJY
Bahasa Pemrograman	PHP 7.2, HTML 5, CSS 4	PHP	PHP	PHP dan HTML	C#, HTML, CSS, JS
Target	Civitas Universitas Muslim Indonesia	Civitas STMIK Nusa Mandiri Jakarta	Civitas Universitas Boyolali	Pegawai UPT TIK	Karyawan KAA
Subjek	Universitas	Sekolah Tinggi	Universitas	Universitas	Universitas
Platfrom	<i>Website</i>	<i>Website</i>	<i>Website</i>	<i>Website</i>	<i>Website</i>
Tujuan	Pengelolaan data perkuliahan PDDikti secara realtime.	Layanan data dan informasi pelaporan data PDDikti.	Akses pengelolaan data perkuliahan dengan PDDikti.	Migrasi data perguruan tinggi ke PDDikti.	Pengelolaan data akademik, melakukan sinkronisasi data, ekspor data-data

					akademik dari <i>database</i> dan excel ke PDDikti.
Menyediakan fitur menampilkan detail data profil perguruan tinggi dengan mengambil data dari <i>Web Service</i>.	Tidak	Tidak	Tidak	Tidak	Ya
Menyediakan fitur sinkronisasi identitas mahasiswa, mata kuliah, kelas, prodi.	Tidak	Tidak	Tidak	Tidak	Ya
Menyediakan fitur ekspor data akademik dengan menyimpan dan menampilkan identitas berupa serangkaian huruf dan	Tidak	Tidak	Tidak	Tidak	Ya

angka secara acak melalui basis data atau excel yang bisa dilakukan <i>tracking</i> supaya dapat mengecek hasil pengelolaannya.					
Menyediakan fitur pengelolaan data referensi mahasiswa yang bisa dilakukan <i>tracking</i> supaya dapat mengecek hasil pengelolaannya.	Tidak	Tidak	Tidak	Tidak	Ya
Menampilkan data transaksi <i>web service</i> yang berhasil dan gagal dalam bentuk grafik dan tabel	Tidak	Tidak	Tidak	Tidak	Ya

Menampilkan data riwayat aktivitas pengguna	Tidak	Ya	Tidak	Tidak	Ya
--	-------	----	-------	-------	----



Berdasarkan perbandingan penelitian yang sudah dilakukan penulis seperti Tabel 2.1. yang telah dipaparkan diatas, penelitian yang sudah dilakukan penulis sebelumnya telah dilakukan oleh penulis yang lain. Proyek yang dikerjakan penulis tentunya terdapat perbedaan dari segi aplikasi, bahasa pemrograman yang digunakan, target, subjek dan tujuan. Penelitian yang dilakukan penulis merupakan aplikasi Feeder UAJY yang memiliki target untuk KAA yang memiliki fitur sinkronisasi data antara database KSI dengan Feeder PDDikti, memasukkan data yang berasal dari database dan dari file excel ke dalam API dari Feeder Dikti. Aplikasi tersebut dibuat dengan menggunakan bahasa pemrograman C#, HTML, CSS dan Java Script dengan menggunakan aplikasi Visual Studio 2022 dalam platform .NET 6.0. Tujuan dari penelitian ini yaitu dengan adanya sistem yang telah dibuat, harapannya mampu membantu karyawan pada perusahaan dalam hal mengelola data-data akademik seperti melakukan sinkronisasi data, ekspor data melalui basis data atau excel ke *web service*. Sesuai dengan tujuan tersebut, penulis juga akan menambahkan fitur yaitu menampilkan detail data profil perguruan tinggi dengan mengambil data dari *web Service*, fitur sinkronisasi identitas mahasiswa, mata kuliah, kelas, prodi. Kemudian fitur ekspor data akademik melalui basis data atau excel yang disesuaikan dengan kebutuhan kemudian menyimpan dan menampilkan identitasnya, fitur pengelolaan data referensi mahasiswa untuk membentuk satu data mahasiswa. Kemudian menampilkan data yang berhasil dan gagal ketika melakukan sinkronisasi dan tambah data dalam bentuk grafik dan tabel, setelah itu menampilkan aktivitas pengguna sebagai pengingat bagi pengguna lain sudah mengakses menu mana saja.