

Rancang Bangun Sistem Presensi Mahasiswa Berbasis Web Dengan Pendekatan PIECES

Rikie Kartadie¹, Muhammad Agung Nugroho², Yudhi Kusnanto³, Bagus Fahrudi Jamal⁴

¹Teknik Komputer, Fakultas Teknologi Informasi
Yogyakarta, Indonesia

¹rikie@utdi.ac.id

(Corresponding author)

^{2,4}Informatika, Fakultas Teknologi Informasi
Yogyakarta, Indonesia

²m.agung.n@utdi.ac.id

⁴bagus.fahrudin@students.utdi.ac.id

³Teknik Komputer, Fakultas Teknologi Informasi
Yogyakarta, Indonesia

³yudhi@utdi.ac.id

Disubmit: 10-07-23; diterima: 26-07-23; dipublikasikan: 08-08-23

Cara mengutip:

R. Kartadie, et al. "Rancang Bangun Sistem Presensi Mahasiswa Berbasis Web Dengan Pendekatan PIECES", *JuTI "Jurnal Teknologi Informasi"*, Vol. 2, No. 1, pp.23 – 31, Februari 2023, DOI: 10.26798/juti.v2i1.951

Ringkasan

Kondisi saat ini di UTDI, di mana presensi mahasiswa dan RPS digabungkan menjadi satu berkas hardcopy, pada titik tertentu akan membuat lebih mudah bagi bagian akademik untuk memantau dan lebih mudah bagi dosen untuk mengajar. Namun, merujuk pada tagline UTDI, Global, Digital, dan Integritas, tampaknya proses ini harus mendapatkan sentuhan digital. Karena sistem presensi mahasiswa berbasis web dapat membantu manajemen data kehadiran mahasiswa secara lebih efisien dan efektif, serta dapat mengurangi kemungkinan kecurangan dan memudahkan manajemen data kehadiran mahasiswa, penelitian ini sangat relevan untuk industri saat ini. Dengan demikian, sistem ini dapat meningkatkan kualitas pendidikan di kampus.

Kata kunci: Kualitas pendidikan, UTDI, Presensi, RPS, Sistem Informasi

Abstract

The current conditions at UTDI, where student presence and RPS are combined into one hardcopy file, at some point it will be easier for academics to monitor and easier for lecturers to teach. However, referring to the UTDI tagline, Global, Digital and Integrity, it seems that this process must get a digital touch. Because a web-based student attendance system can help hazard data management student administration more efficiently and effectively, and can reduce the possibility of fraud and facilitating the management of student attendance data, this research is very relevant for industry today. Thus, this system can improve the quality of education on campus.

KeyWords: Quality of education, UTDI, Presence, RPS, Information Systems

1. Pendahuluan

Takaran waktu kegiatan belajar yang di bebaskan pada mahasiswa per minggu per semester dalam proses pembelajaran melalui berbagai bentuk pembelajaran atau besarnya pengakuan atas keberhasilan usaha mahasiswa dalam mengikuti kegiatan kurikuler di suatu Program Studi adalah Satuan Kredit Semester [1].

Absensi adalah rutinitas yang dilaksanakan setiap orang untuk menunjukkan bahwa dirinya hadir atau tidak dalam institusi. Absensi diidentifikasi dengan pemanfaatan kehadiran yang dikendalikan oleh masing-masing organisasi atau perusahaan[2].

Kehadiran mahasiswa dalam mengikuti perkuliahan dicatat dan dibukukan dalam presensi. Presensi setiap mahasiswa dan dosen dalam lembaga pembelajaran adalah salah satu bahan yang harus dilakukan karena memiliki peranan penting, di mana presensi merupakan salah satu penunjang yang dapat mendukung atau memotivasi setiap kegiatan di dalamnya. Di sisi lain presensi mahasiswa dapat juga menjadi informasi tentang bagaimana kedisiplinan mahasiswa yang bersangkutan. Penting bagi dosen dalam pembelajaran untuk mengetahui kehadiran mahasiswa [3].

Kondisi saat ini di lingkungan UTDI (Universitas Teknologi Digital Indonesia), presensi mahasiswa dan RPS (Rencana Pembelajaran Semester) dijadikan 1 berkas secara *hardcopy*. Pada bagian tertentu, hal ini akan mempermudah bagian Akademik melakukan monitoring dan mempermudah dosen dalam mengajar karena RPS sudah jadi satu di dalam berkas. Namun di Era sekarang ini, dan merujuk pada *tagline* UTDI yaitu Global, Digital dan *Integrity*, maka sepertinya proses ini harus mendapatkan perhatian dan sentuhan digital.

Menurut Alter (1992) seperti yang dalam Kadir, sistem informasi adalah kombinasi antar prosedur kerja, informasi, orang, dan teknologi informasi yang diorganisasikan untuk mencapai tujuan sebuah organisasi. Sistem informasi adalah kumpulan perangkat keras dan *software* yang dirancang untuk mentransformasikan data ke dalam bentuk informasi yang berguna. Sistem informasi adalah suatu sistem buatan manusia yang secara umum terdiri atas sekumpulan komponen berbasis komputer dan manual yang dibuat untuk menghimpun, menyimpan, dan mengelola data serta menyediakan informasi keluaran kepada para pemakai [4]. Pada perkembangannya, sistem informasi melibatkan *database*, SQL pertama kali didefinisikan oleh *American National Standards Institute* (ANSI) pada tahun 1986. MySQL adalah sebuah sistem manajemen *database* yang bersifat open source [5].

UML (*Unified Modeling Language*) lahir dari penggabungan banyak Bahasa pemodelan grafis berorientasi objek yang berkembang pesat pada akhir 1980-an dan awal 1990-an. UML terdiri dari 13 jenis diagram, jenis-jenisnya adalah *Diagram Activity*, *Diagram Class*, *Diagram Communication*, *Diagram Component*, *Diagram Composite Structure*, *Diagram Deployment*, *Diagram Interaction Overview*, *Diagram Object*, *Diagram Package*, *Diagram Sequence*, *Diagram State Machine*, *Diagram Timing*, *Diagram Use Case* [6].

Prototyping adalah iterative dalam pengembangan sistem di mana kebutuhan diubah ke dalam sistem yang bekerja (*working system*) yang secara terus menerus diperbaiki melalui kerjasama antara pengguna dan analisis. *Prototyping* juga bisa dibangun melalui beberapa *tool* pengembangan untuk menyederhanakan proses. *Prototyping* merupakan bentuk dari *Rapid Application Development* (RAD)[7]

Rahmat Arief dalam penelitiannya menyatakan telah berhasil membuat sebuah aplikasi presensi dengan menggunakan *google form*, *sheet*, *Sites*, *Awesome Table* dan *Gmail* [8]. Penggunaan *google form* mudah untuk diimplementasikan, namun tidak dapat diintegrasikan dengan *database* yang ada di sistem informasi lain dalam hal ini sistem informasi akademik. Penelitian ini akan membangun aplikasi presensi yang bersifat mandiri, dan ter-integrasi dengan RPS dan berita acara perkuliahan.

Terdapat pula beberapa penelitian yang membuat aplikasi serupa dengan menggunakan android [9, 10] Teknik yang digunakan adalah langsung memasukkan data kehadiran ke dalam *database*, sedangkan Prasetyo membuat aplikasi berbasis *web* dan menyertakan *QRCode* [11] yang juga langsung memasukkan data presensi ke dalam *database*, penelitian ini belum sampai menggunakan *QRCode* namun proses yang penting dalam aplikasi yang akan dibangun adalah adanya fitur sinkronisasi secara periodik ke SIAKAD (Sistem Informasi Akademik) sehingga tidak perlu secara *realtime* menginputkan data ke *database*.

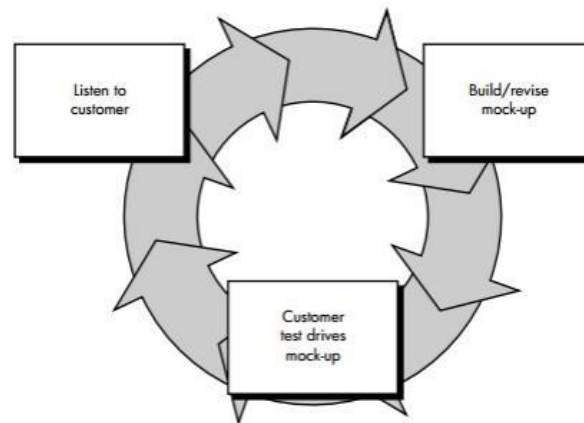
Penelitian ini memiliki relevansi yang tinggi dalam industri saat ini karena sistem presensi mahasiswa berbasis *web* dapat membantu pengelolaan data kehadiran mahasiswa secara lebih efektif dan efisien. Selain itu, sistem ini juga dapat mengurangi kemungkinan terjadinya kecurangan dan memudahkan pengelolaan data kehadiran mahasiswa. Dengan demikian, sistem presensi mahasiswa berbasis *web* dapat meningkatkan kualitas pendidikan di lingkungan kampus.

Diharapkan dari penelitian ini adalah kendala yang dihadapi dosen selama ini dalam presensi akan teratasi. Program *paperless* yang dicanangkan dan berusaha mewujudkan visi universitas menuju universitas berbasis teknologi digital.

2. Metode Penelitian

2.1. Software Development Life Cycle

Prototyping juga bisa dibangun melalui beberapa *tool* pengembangan untuk menyederhanakan proses. *Prototyping* merupakan bentuk dari *Rapid Application Development (RAD)*[7] seperti terlihat pada Gambar 1.



Gambar 1. RAD [7]

RAD memiliki beberapa kelemahan, diantaranya:

1. RAD mungkin mengesampingkan prinsip-prinsip rekayasa perangkat lunak.
2. Menghasilkan inkonsistensi pada modul-modul sistem.
3. Tidak cocok dengan standar.
4. Kekurangan prinsip *reusability* komponen.

Keuntungan RAD diantaranya:

1. *Prototype* melibatkan pengguna dalam analisis dan desain.
2. Punya kemampuan menangkap kebutuhan secara konkret daripada secara abstrak.
3. Untuk digunakan secara *standalone*.
4. Digunakan untuk memperluas SDLC.

Alasan menggunakan model pengembangan *prototype*, karena *prototype* mengutamakan kebutuhan *user* untuk menghasilkan sebuah sistem. Dengan cara peneliti menunjukkan *proto- type* dari sistem yang sedang dikembangkan yang kemudian akan dievaluasi langsung oleh *user*, sehingga peneliti menghasilkan sebuah sistem yang sesuai dengan kebutuhan *user*.

2.2. Analisis PIECES

PIECES framework adalah kerangka yang dipakai untuk mengklasifikasikan suatu problem, *opportunities*, dan *directives* yang terdapat pada bagian *scope definition analisis* dan perancangan sistem. Dengan kerangka ini, dapat dihasilkan hal-hal baru yang dapat menjadi pertimbangan dalam mengembangkan sistem [12].

Pengertian analisis *pieces* adalah suatu system yang di gunakan untuk analisis system kerja pada suatu perusahaan atau organisasi. Ada 6 kriteria analisis *pieces* yaitu kinerja (*Performance*), informasi (*Information*), ekonomi (*Economic*), kontrol (*Control*), efisiensi (*Efficiency*), dan pelayanan (*Services*). Analisis *pieces* juga sangat banyak diterapkan untuk penelitian pada suatu perusahaan atau organisasi. Selain mudah dan dapat dipahami analisis *pieces* juga bersifat ringan tidak membutuhkan data yang banyak [13].

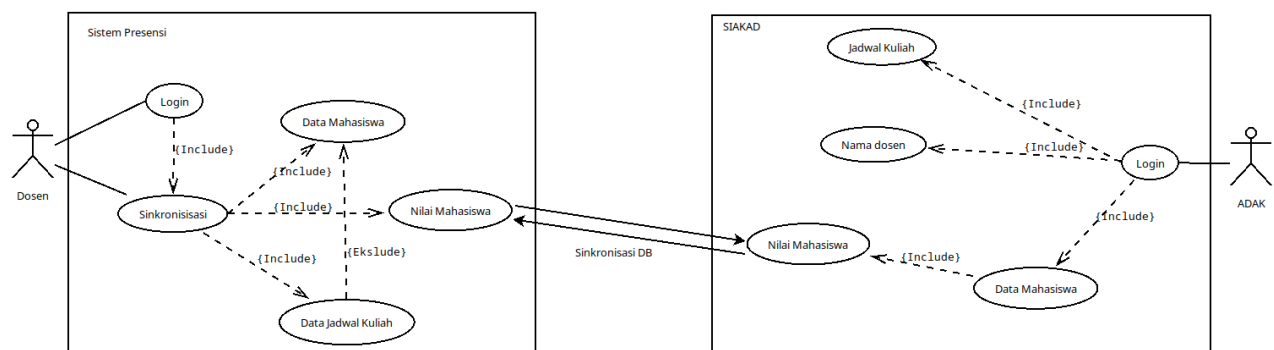
Adapun analisis PIECES dapat dilihat pada Tabel 1 berikut:

Tabel 1. Analisis PIECES

	Lama	Baru
performace	Pada Presensi yang lama, Dosen membawa presensi, mengisi presensi, mengisi berita acara perkuliahan secara manual dan bagian administrasi akademik merekap presensi secara manual.	dosen hanya tinggal melakukan pengiriman/sinkronisasi <i>database</i> kepada SIAKAD langsung dari perangkat dosen.
Information	Presensi yang lama ada kemungkinan untuk terjadinya kesalahan dalam melakukan rekap	kemungkinan salah rekap dapat dihindari.
Economic	Dengan presensi yang lama penggunaan kertas dan sumber daya tinggi, baik sumber daya manusia maupun finansial.	dengan presensi yang baru, penggunaan kertas dan sumber daya dapat diminimalisasi.
Control	Tingkat kontrol pada presensi lama, sangat minim, dan memungkinkan terjadinya <i>human error</i>	Pada presensi yang diusulkan, akan ada rekap secara otomatis dalam bentuk file Excell, atau format lain yang mungkin, yang menginformasikan jumlah pertemuan dosen, jumlah kehadiran tiap mahasiswa dan <i>file pdf</i> berupa berita acara mengajar pada tiap pertemuan.
Efficiency	Tingkat <i>load</i> kerja bagian administrasi akademik selama ini tinggi karena harus secara manual melakukan rekap presensi	Akan memberikan kemudahan pada bagian administrasi akademik dan tentunya dosen dalam melakukan pelaporan presensi dan rekap presensi.
Service	Layanan presensi saat ini belum dapat memberikan informasi yang secara cepat dan akurat kepada mahasiswa, dosen atau bagian administrasi akademik	Dapat langsung didapatkan informasi yang akurat dan cepat tentang presensi.

3. Desain Arsitektur Sistem

Desain Arsitektur Sistem ini akan menjelaskan proses sinkronisasi *database* yang dilakukan oleh sistem presensi ke dan dari SIAKAD. seperti terlihat pada Gambar 2.



Gambar 2. Design Sinkronisasi Database antara SI-Presentasi dan SIAKAD

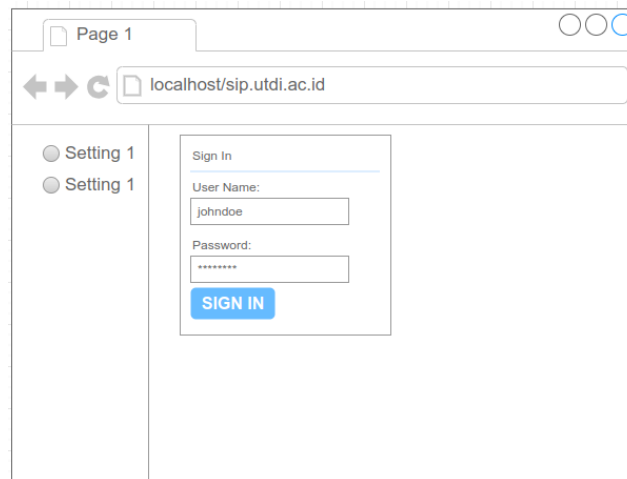
Pada Gambar 2 dapat dilihat bahwa terdapat 2 aplikasi yang berbeda, yaitu Sistem Informasi Perkuliahan Terintegrasi (SIPT) dan SIAKAD yang dijumpai oleh kode *bridging/jembatan* yang menghubungkan dua *database*, *database* dari SIPT dan *database* dari SIAKAD.

Dapat dilihat bahwa Sistem presensi akan melakukan sinkronisasi data secara terbatas, dan akan meng-*update database* pada SIAKAD pun secara terbatas. Bila pada awal perkuliahan, di mana admin SIAKAD telah menginput nama dosen, jadwal perkuliahan, daftar nama mahasiswa per kelas, maka sistem presensi yang belum memiliki data terbaru tersebut akan melakukan sinkronisasi dan *update database*-nya sehingga sesuai dengan data yang ada pada SIAKAD.

Setelah data yang dibutuhkan pada perkuliahan telah sinkron dengan SIAKAD, sistem ini dapat berjalan secara *stand alone*, untuk kemudian melakukan sinkronisasi lagi pada saat akan diadakan Ujian Tengah Semester (UTS) dan meng-*update database* yang ada di SIAKAD sehingga admin akan memperoleh data kehadiran mahasiswa, materi yang disampaikan dosen, kehadiran dosen dan durasi mengajar dosen (tergantung *database* yang diizinkan di *update*). Sistem presensi akan berjalan kembali secara *stand alone*, dan melakukan sinkronisasi kembali pada saat akan diadakan Ujian Akhir semester (UAS) sehingga admin SIAKAD dapat memperoleh data ter-*update*.

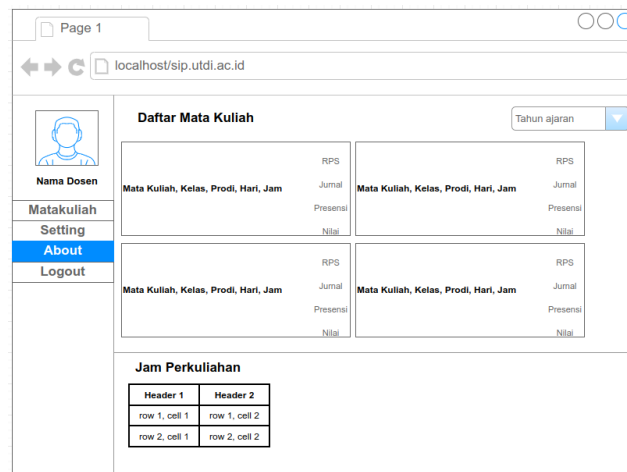
3.1. Desain Halaman Sistem

Halaman sistem di desain semudah dan sebaik mungkin dengan mengedepankan bagian utama dari sistem ini yaitu nama mata kuliah dan jadwal. Desain halaman *login* dapat dilihat pada Gambar 3 berikut;



Gambar 3. Design halaman login

Halaman yang memuat daftar mata kuliah, jadwal dan tahun ajaran, halaman setelah login, dapat dilihat pada Gambar 4 berikut:



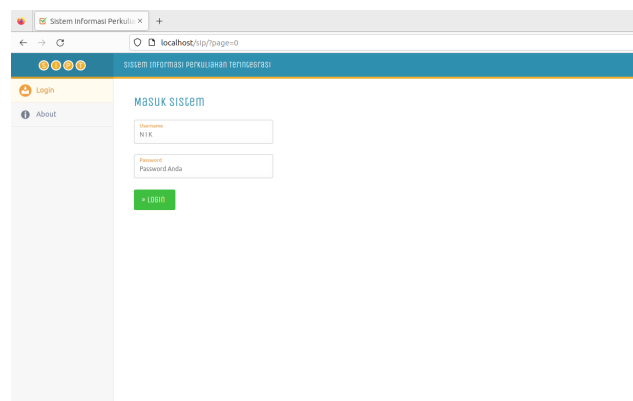
Gambar 4. Design halaman utama

4. Hasil dan Pembahasan

Sistem presensi yang diberi nama "Sistem Informasi Perkuliahan Terintegrasi (SIPT)" merupakan aplikasi yang digunakan untuk mengelola data perkuliahan yang meliputi Rencana Pembelajaran Semester (RPS), Berita Acara Perkuliahan, Daftar Presensi, dan juga *Upload* Nilai. Aplikasi ini dibangun menggunakan bahasa pemrograman PHP dan SQLite sebagai basis datanya. Aplikasi ini di-install pada PC *client* dan menggunakan koneksi internet untuk sinkronisasi data dengan *server*.

4.1. Halaman Login

User tidak dapat membuat *username* sendiri, *username* dan *password* yang sama dengan yang ada di server SIAKAD. Sistem akan mencari data *username* dan *password* di server SIAKAD untuk pertama sekali, setelah berhasil *login*, *user* dapat melakukan sinkronisasi dan mendapatkan data-data dari *server* SIAKAD. Selanjutnya proses login akan mencari *username* dan *password* yang ada pada database lokal, tidak lagi dari server SIAKAD. Halaman Login dapat dilihat pada Gambar 5 berikut.



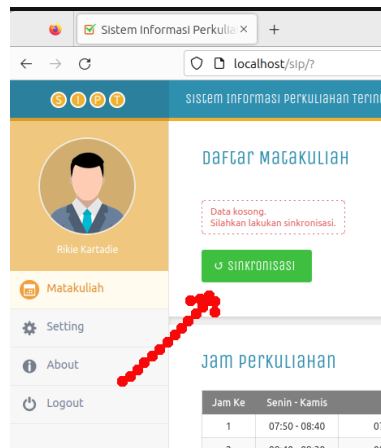
Gambar 5. Halaman Login

Begitu pula bila terjadi perubahan *username* dan *password* di SIAKAD, maka untuk pertama kali user harus melakukan login sesuai dengan login yang ada di SIAKAD, dan database local akan melakukan update *username* dan *password* kembali.

4.2. Sinkronisasi

Fitur ini digunakan untuk menyelaraskan data perkuliahan yang ada di *client* dan juga *server*. Terdapat 2 (dua) proses utama pada fitur ini yaitu *Sync Back* dan *New Sync*. *Sync Back* digunakan untuk mengirim semua data pada *client* (jika ada) ke dalam *server*. *New Sync* digunakan untuk menarik data pada *server* (jika ada) ke dalam *client*. Kedua proses di atas menggunakan beberapa parameter sebagai kunci utama, diantaranya: tahun akademik, semester, dan kode dosen. Tahun akademik dan Semester didapat dari perhitungan tahun dan bulan yang berjalan saat ini. Semester genap diasumsikan berkisar dari bulan Februari hingga Agustus, selain itu diasumsikan semester gasal.

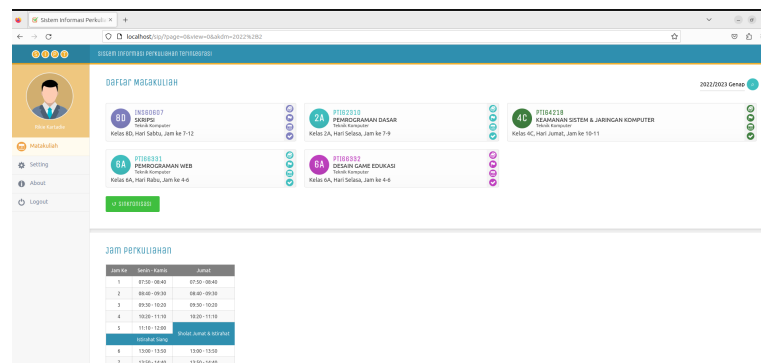
User akan melakukan sinkronisasi dengan menekan tombol sinkron seperti terlihat pada Gambar 6, proses ini akan melakukan request data dari aplikasi menuju ke server dan akan dihasilkan jadwal perkuliahan, ruang kuliah dan nama mahasiswa yang mengambil mata kuliah tersebut yang telah di *input* oleh bagian akademik. Hasil sinkronisasi dapat dilihat pada Gambar 7.



Gambar 6. Tombol Sinkron

Proses *Sync Back* akan mengumpulkan semua data pada database dan file txt berdasarkan parameter tahun, semester, dan kode dosen, kemudian mengemasnya ke dalam sebuah *array*. Data dalam jumlah besar ini dikirim ke server dalam bentuk *JSON* yang dienkripsi menggunakan *cURL*.

Proses *New Sync* secara umum adalah kebalikan dari proses *Sync Back*. Pertama-tama sistem akan melakukan *request* data ke server dengan parameter tahun, semester, dan kode dosen. *Request* dikirim dalam bentuk *JSON* yang dienkripsi menggunakan *cURL*. Apabila *request* berhasil, server akan mengirim balik data dalam jumlah yang besar. Data tersebut berupa *JSON* yang dienkripsi. Sistem terlebih dahulu akan melakukan dekripsi sebelum memecah data tersebut ke dalam *array*. Setelah data berhasil di pecah ke dalam *array*, selanjutnya sistem akan melakukan distribusi data. Beberapa data dimasukkan ke dalam *database*, beberapa data dimasukkan ke dalam txt.



Gambar 7. Halaman dashboard setelah sinkronisasi

5. Pembahasan

5.1. Uji Bridge SIP dan Dummy SIAKAD

Bridge yang dibuat khusus untuk melakukan pengujian ke *dummy* SIAKAD, proses pengujian ini dilakukan untuk mencoba fungsi sinkronisasi yang dilakukan sistem ke dan dari *database* SIAKAD, kode *bridge* dapat dilihat pada algoritma 1 berikut:

Hasil uji dilakukan secara *blackbox testing* dan menguji keberhasilan proses sinkronisasi dari sistem. Proses sinkronisasi ini merupakan proses penting dalam sistem yang akan memastikan data dapat terinput dengan baik, hasil uji *Black box* ini dapat dilihat pada Tabel 2 berikut:

Algorithm 1 Algoritma Brigde SIPT - SIAKAD

```

Data pada SIPT kosong.
while true do
    Lakukan penarikan data dari server SIAKAD.
    Cek kondisi data.
    if Data SIP = data SIAKAD then
        Tidak ada data yang sinkron.
    end if
    if Data SIAKAD < data SIP then
        update data SIAKAD.
    end if
    Menampilkan Pesan data tersimpan.
end while
Ulangi langkah pada saat sinkronisasi dilakukan.
    
```

Tabel 2. hasil uji Bridge

Kondisi	Tindakan Uji	Hasil Yang diharapkan	Hasil Uji	Status Uji
Database Sistem Informasi Presensi kosong	memasukkan <i>username</i> dan <i>password</i>	berhasil <i>request username</i> dan <i>password</i> ke SIAKAD	Hasil berhasil melakukan <i>request</i>	Berhasil
Data sistem kosong	sinkronisasi ke SIAKAD	Berhasil mendapatkan data Jadwal mengajar, Mahasiswa per kelas	Memperoleh sinkronisasi dari SIAKAD untuk data yang dibutuhkan	Berhasil
Pengisian Sistem	<i>Input</i> RPS per mata kuliah	Proses <i>input</i> RPS berhasil dilaksanakan	Proses <i>input</i> berhasil	Berhasil namun hasil cetak RPS di SIAKAD tidak diujikan
Memulai perkuliahan	Menambah pertemuan dan melakukan presensi	Berhasil memulai kelas, timer berjalan, presensi terekam	Memulai kelas dapat dilakukan, <i>timer</i> berjalan dengan baik dan presensi mahasiswa berhasil terekam	Berhasil
Sinkronisasi ke Database SIAKAD	Melakukan sinkronisasi ke-2 utk proses UTS	Data berhasil tersinkronisasi	Terdapat beberapa Data yang tidak masuk ke dalam database SIAKAD	Berhasil sebagian

6. Simpulan

Berdasarkan hasil uji yang diperoleh, terdapat hasil uji sinkronisasi yang belum sempurna, hal ini dapat dikarenakan proses pengkodean *dummy* siakad belum sempurna atau dapat pula dikarenakan kode *bridge* yang masih ada kesalahan. Namun secara garis besar, sistem yang di rancang telah dapat berjalan dengan baik. Proses pada sistem telah dapat berjalan sesuai dengan yang diharapkan.

Sistem Informasi Perkuliahan Terintegrasi ini, akhirnya bukan hanya sebuah sistem presensi mahasiswa saja namun dapat menjangkau pada RPS, Penjadwalan, durasi mengajar Dosen, tingkat kesesuaian mengajar dengan RPS yang telah direncanakan sebelumnya, dan sistem penilaian yang akan sangat membantu bagian akademik melakukan rekapitulasi data akademik.

Untuk pengujian akan terus dilakukan hingga seluruh fungsi yang diharapkan berhasil dilakukan. Sistem belum memiliki sistem keamanan *database*, bila terjadi perubahan database pada sisi sistem (*client*) atau

perubahan secara manual proses bisnis sistem langsung ke *database* tidak/belum tercatat.

Sistem informasi ini akan lebih baik jika adanya sistem keamanan *database* dan log *database* yang dapat merekam perubahan *database* secara ilegal dari *user* sehingga tidak terjadi manipulasi data yang akan terekam di SIAKAD. Adanya pembatasan akses aplikasi dengan menambahkan cek keberadaan laptop *user*, dengan IP kampus atau menggunakan API *google location*, sehingga aplikasi tidak dapat diakses kecuali dalam lingkungan kampus yang pada saat penelitian ini dikerjakan belum ada dalam sistem.

Pustaka

- [1] Permendikbud, "PERATURAN MENTERI PENDIDIKAN DAN KEBUDAYAAN REPUBLIK INDONESIA TENTANG STANDAR NASIONAL PENDIDIKAN TINGGI", No. 3, 2020.
- [2] Lengkong, O. H., Fiden, D. H., & Masriat, A. (2016). Sistem Informasi Absensi Real-Time di Universitas Klaten. *Cogito Smart Journal*. <https://doi.org/10.31154/cogito.v2i2.31.216-228>
- [3] Khoiriyah NL, Marisa F, Wijaya ID. Rancang bangun sistem presensi online berbasis granted validitas data. *JIMP (Jurnal Informatika Merdeka Pasuruan)*. 2018 Mar 23;3(1).
- [4] Kadir, A. (2014). *Pengertian Sistem Informasi* (2 ed.; Dewi, Ed.). Yogyakarta: ANDI.
- [5] Novendri, M. S., Saputra, A., & Firman, C. E. (2019). Aplikasi Inventaris Barang Pada Mts Nurul Islam Dumai Menggunakan PHP Dan MYSQL. *Lentera Dumai*, 10(2), 46–57.
- [6] Fowler, M. (2005). *UML Distilled Edisi 3 Panduan Singkat Bahasa Pemodelan Objek Standar* (3 ed.). Yogyakarta: ANDI.
- [7] Fatta, H. Al. (2007). *Analisis dan Perancangan Sistem Informasi untuk Keunggulan Bersaing Perusahaan dan Organisasi Modern* (1 ed.). Yogyakarta: ANDI.
- [8] Arief R. Aplikasi Presensi Siswa Online Menggunakan Google Forms, Sheet, Sites, Awesome Table dan Gmail. *Sntekpan V, Itats, Surabaya*. 2017 Oct 17:137-43.
- [9] Ardiansyah A, Fitri I, Iskandar A. Aplikasi Manajemen Perkantoran dan Absensi Online Berbasis Android. *Jurnal JTIK (Jurnal Teknologi Informasi dan Komunikasi)*. 2021 Jun 30;5(2):126-34.
- [10] Hermanto N, Riyanto NR. Aplikasi sistem presensi mahasiswa berbasis android. *Simetris: Jurnal Teknik Mesin, Elektro dan Ilmu Komputer*. 2019 Apr 1;10(1):107-16.
- [11] Prasetyo D, Fitri I, Rubhary A. Sistem Absensi Online Berbasis Web Dengan QR Code Secara Real Time Menggunakan Algoritma Vigenere Cipher. *INTECOMS: Journal of Information Technology and Computer Science*. 2021 Jun 14;4(1):88-96.
- [12] Supriyatna, Adi. Maria, Vivi. (2017). Analisis Tingkat Kepuasan Pengguna dan Tingkat Kepentingan Penerapan Sistem Informasi DJP Online dengan Kerangka PIECES. ISSN: 2477-698X. *KHAZANAH INFORMATIKA* Vol. 3 No. 2 Desember 2017
- [13] Wetherbe, James. (2012). *PIECES Analysis*
- [14] S. Ramadhani & W. A. Kusuma, PIECES Framework untuk Analisa Tingkat Kepuasan Pengguna dan Kepentingan Sistem Informasi, *Jurnal Teknologi & Manajemen Informatika - Vol.4 No.2 2018*, DOI:10.26905/jtmi.v4i2.2101