

# Implementasi *Social Translucence* dan *Collaborative Storage* pada Rancang Bangun Aplikasi BKD Online di Kopertais Wilayah III Daerah Istimewa Yogyakarta

Herwinsyah<sup>1</sup>, Agung Fatwanto<sup>2</sup>

Magister Informatika Fakultas Sains dan Teknologi

UIN Sunan Kalijaga Yogyakarta

Yogyakarta, Indonesia

e-mail: <sup>1</sup>herwinsy@gmail.com, <sup>2</sup>agung.fatwanto@uin-suka.ac.id

Diajukan: 22 Oktober 2022; Direvisi: 21 Juni 2023; Diterima: 24 Agustus 2023

## Abstrak

Dosen pada perguruan tinggi, di setiap semesternya juga diwajibkan untuk membuat pelaporan kegiatan yang telah dilaksanakan di semester sebelumnya dan juga membuat rencana kegiatan untuk semester yang akan datang. Tujuan penelitian ini memberikan solusi melalui aplikasi BKD Online kepada dosen dalam peng-input-an pelaporan kegiatan Beban Kinerja Dosen per-semester menjadi lebih mudah dan transparan dalam penilaian asesor dan menyimpan seluruh berkas dokumen di dalam satu wadah bersama. Metode yang digunakan dalam rancang bangun menggunakan metode *social translucence* (keterbukaan) dimana feedback dari asesor sebagai penilai bisa langsung diketahui oleh dosen dan metode *collaborative storage* (wadah bersama). Hasil dari penelitian ini adalah aplikasi BKD Online yang dapat langsung digunakan oleh para dosen dalam peng-input-an data laporan kinerja, dimana pada aplikasi ini, baik dosen sebagai pelapor dan asesor sebagai penilai dapat langsung berinteraksi melalui fitur message pada setiap laporan yang dilaporkan dosen. Fitur upload yang dimasukkan juga menggunakan metode wadah bersama agar seluruh dokumen terkumpul dalam satu wadah yang dapat diakses langsung oleh kopertais.

**Kata kunci:** Laporan Bkd online, Collaborative storage, Codeigniter, Web.

## Abstract

Lecturers at universities are required to create reports on their activities for each semester, reflecting the activities carried out in the previous semester and planning for the upcoming semester. The objective of this research is to provide a solution through the Online BKD (Lecturer Workload) application, which aims to facilitate and enhance transparency in the inputting of performance reports by lecturers on a per-semester basis, as well as storing all documents in one shared platform. The design methodology employed in this study utilizes the *social translucence* approach, enabling assessors' feedback to be directly accessible to lecturers, and *collaborative storage* method for shared document management. The outcome of this research is the development of the Online BKD application, which can be readily utilized by lecturers for inputting performance report data. Within this application, both the reporting lecturers and the assessing assessors can directly interact through the messaging feature embedded in each report submitted by lecturers. The upload feature also utilizes a shared storage method to centralize all documents in one container, which can be accessed directly by the higher education coordinating agency (Kopertais).

**Keywords:** Online Bkd report, Collaborative storage, Codeigniter, Web.

## 1. Pendahuluan

Pada lembaga Perguruan Tinggi komponen yang sangat esensial adalah dosen. Dalam mewujudkan tujuan utama nasional yaitu mencerdaskan kehidupan bangsa serta meningkatkan kualitas manusia rakyat Indonesia meliputi manusia yang beriman dan bertakwa, memiliki akhlak yang mulia serta menguasai ilmu pengetahuan, teknologi dan seni, juga mewujudkan masyarakat Indonesia yang maju, adil dan makmur lagi beradab, maka peran tugas dan tanggung jawab dosen sangatlah penting. Dalam melaksanakan hal tersebut diperlukan dosen yang berkualitas profesional. Tugas utama seorang dosen adalah melaksanakan Tridharma Perguruan Tinggi yang dibebankan minimal 12 SKS dan maksimal 16 SKS pada setiap semesternya dan disesuaikan dengan kualifikasi akademik [1].

Pelaporan pelaksanaan tugas utama dosen yaitu melakukan aktifitas tridharma ini perlu dilakukan secara periodik sebagai bentuk akuntabilitas kinerja dosen kepada para pemangku kepentingan. Dalam pembuatan laporan BKD ini perlu diberikan arahan dan tata cara penetapan Beban Kerja Dosen dan Evaluasi Pelaksanaan Tridharma Perguruan Tinggi, agar akuntabilitas terlaksana maka diperlukan sebuah aplikasi untuk pelaporan BKD.

Berdasarkan hasil observasi penulis tidak sedikit kondisi perguruan tinggi yang saat ini memiliki dosen dengan usia tidak muda lagi atau sudah mencapai usia senja, dimana banyak kesulitan jika dihadapkan dengan sebuah sistem aplikasi yang terlalu banyak alur (tidak *simple*). Keterbukaan sistem dalam hal data yang di-*input*-kan ke dalam system dapat di-*monitoring* bersama antara dosen yang bersangkutan dengan asesor dan pihak kopertais, agar apabila ada data yang masih kurang ataupun belum dimasukkan, baik asesor maupun pihak kopertais dapat mengingatkan dosen tersebut untuk melengkapi data-datanya. Tujuannya agar tidak adanya keterlambatan data yang di *collecting* pihak kopertais yang akan memicu terhambatnya kompensasi (dana sertifikasi) yang akan diberikan dari kemenag.

Pada penelitian sebelumnya yang dilakukan oleh Fajar Nugraha dkk pada tahun 2019, mereka melakukan penelitian yang serupa yaitu penelitian dengan judul “Rancang Bangun Sistem Informasi Penilaian Beban Kerja Dosen” menitikberatkan pada perancangan secara umum dari sebuah aplikasi, untuk *storage* penyimpanan menggunakan MySql *database*[2]. Pada penelitian yang saya lakukan lebih mengembangkan metode, tidak hanya rancang bangun secara umum tetapi juga melihat dari sisi keterbukaan komunikasi dari asesor terhadap dosen pelapor, dan sebaliknya, dan untuk *storage* penyimpanan berkas menggunakan *drive* penyimpanan dari google drive.

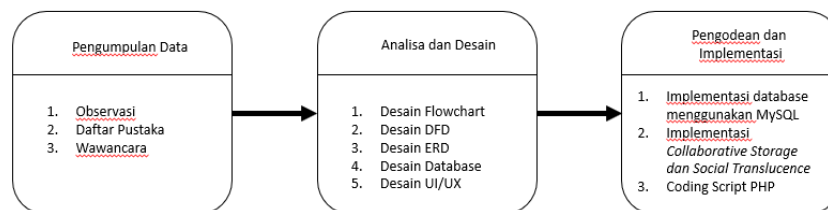
Kekurangan lainnya yang banyak dialami adalah penggunaan sistem yang cukup membingungkan para dosen tersebut sehingga sudah seharusnya pihak pemegang kepentingan menyediakan atau memberikan fasilitas dalam bentuk aplikasi BKD *online* yang *simple* dan mendukung proses yang mudah, dengan tujuan agar para dosen terutama yang berusia lanjut dapat membuat pelaporan BKD nya dengan mudah. Begitupun dengan proses pengecekan dokumen BKD yang selama ini dijalankan harus mengumpulkan dokumen dan harus di cek secara manual oleh para asesor. Berdasarkan observasi peneliti terkadang kesibukan asesor yang sangat tinggi membuat kopertais dalam hal ini sebagai penanggungjawab panitia tim proses review dokumen BKD menjadi kesulitan untuk mengumpulkan asesor dalam satu waktu untuk melakukan proses reviewnya.

Berdasarkan masalah-masalah di atas maka perlu dibuatkan sebuah *system* yang dapat menjadi solusi dari masalah tersebut. Aplikasi BKD *Online* ini mengusung konsep laporan tanpa cetak (Paperless), jadi hasil laporan yang dibuat para dosen dapat langsung dianalisa dan dinilai oleh asesor penilai laporan BKD. Hasil penilaian tersebut juga dapat langsung di lihat oleh dosen yang bersangkutan apabila masih memiliki kekurangan, baik dalam penyajian maupun data lampiran laporan BKD nya. Konsep lainnya yang diaplikasikan ke dalam aplikasi ini adalah konsep *collaborative storage* (konsep penyimpanan data dalam satu *storage cloud*), dalam aplikasi ini pengiriman data baik dari data lampiran laporan dari dosen kepada kopertais (assessor) maupun dari kopertais kepada dosen-dosen, biasanya dalam bentuk surat surat yang berhubungan dengan surat keputusan (sk) seperti sk imparsing dan sk kenaikan pangkat.

## 2. Metode Penelitian

### 2.1. Desain Penelitian

Desain penelitian yang digunakan termasuk dalam kategori penelitian *ex post facto*, dimana penyelidikannya berfokus kejadian telah terjadi sebelumnya. Pengumpulan data dilakukan dengan cara observasi, studi pustaka dan wawancara. Observasi dilakukan dengan menganalisa laporan BKD periode sebelumnya untuk diambil referensi yang berkaitan dengan proses alur sistem yang akan dibuat. Studi pustaka dilakukan untuk mencari referensi dan pembanding dari aplikasi yang akan di bangun dan juga agar tidak terjadi duplikasi penelitian yang sejenis. Aplikasi sebelumnya yang sudah ada juga diobservasi untuk memahami alur transisi bisnis prosesnya. Aplikasi ialah penerapan perangkat lunak (*software*) atau program komputer yang berjalan pada sistem tertentu yang bertujuan untuk melaksanakan perintah atau tugas-tugas yang diberikan oleh pengguna. Sebuah aplikasi berbasis web dapat menjangkau pengguna yang lebih luas, yaitu bisa diakses dari manapun (menggunakan jaringan internet) dengan menggunakan aplikasi web browser [3]. Proses wawancara dilakukan beberapa kali melalui diskusi dengan pimpinan kopertais dan dosen untuk menggali lebih dalam tentang fitur yang akan di masukkan dalam aplikasi BKD *Online* agar aplikasi ini nantinya dapat digunakan oleh para dosen tanpa kesulitan yang berarti. Agar penelitian ini dapat berjalan dengan baik maka perlu dibuat suatu rancangan tahapan penelitian. Tahapan penelitian dapat dilihat pada Gambar 1.



Gambar 1. Tahapan Penelitian

## 2.2. Collaborative Storage

*Collaborative* atau dalam Bahasa Indonesia adalah kolaborasi. Suatu pola kaitan kerjasama yang dilakukan oleh lebih dari satu orang adalah istilah dari kolaborasi. Ada banyak ragam pengertian kolaborasi, beragamnya penafsiran kerja sama didasari oleh prinsip yang sama ialah menimpa kebersamaan, kerja sama, berbagi tugas, kesetaraan, tanggung jawab. Kolaborasi dapat berarti hubungan timbal balik antara bagian-bagian organisasi yang berpartisipasi dan bergantung satu sama lain untuk mencapai tujuan bersama, berbagi informasi, berbagi sumber daya, berbagi manfaat dan mengambil tanggung jawab pengambilan keputusan bersama tanggung jawab untuk memecahkan berbagai masalah.. “*Collaboration is the mutual engagement of participants in a coordinated effort to solve a problem together. Collaborative interactions are characterized by shared goals, symmetry of structure, and a high degree of negotiation, interactivity, and interdependence.*” Definisi tersebut menjelaskan bahwa kolaborasi atau kerjasama adalah partisipasi bersama dalam upaya bersama untuk memecahkan masalah bersama. Interaksi kolaboratif dicirikan oleh tujuan bersama, struktur simetris dengan tingkat negosiasi yang tinggi karena interaktivitas dan saling ketergantungannya [4].

*Cloud storage* adalah media penyimpanan file *online* atau digital yang mengandalkan koneksi internet untuk mengakses data. *Cloud storage* akan letakkan di sejumlah server yang dikelola oleh penyedia layanan atau biasa dikenal dengan *hosting*. Beberapa media penyimpanan *cloud* yang sangat populer saat ini adalah Google Drive, One Drive, iCloud, dan Dropbox [5]. *Cloud storage* umumnya digunakan sebagai sarana penyimpanan data digital dengan menggunakan server sebagai media penyimpanannya [6]. Penyimpanan *cloud* berguna untuk penyimpanan yang mudah dan akses yang mudah di mana saja di berbagai perangkat seperti PC, laptop, *tablet*, dan *smartphone*, melalui Internet. Penyimpanan *cloud* dapat berbagi file dengan pengguna lain tanpa harus menukar perangkat penyimpanan seperti flash drive. Teknologi penyimpanan *cloud* memudahkan akses data di mana saja.

## 2.3. Social Translucence

*Social Translucence* atau transparansi sosial adalah istilah sosio-teknis untuk menggambarkan bagaimana sistem dapat memfasilitasi pemahaman berkaitan dengan tindakan orang-orang di lingkungan *online*. Transparansi sosial mencakup tiga atribut desain utama: (a) kesadaran bersama tentang aktivitas, (b) penyebaran kontekstual dari isyarat yang menonjol secara sosial (visibilitas), dan akuntabilitas untuk tindakan seseorang. Melalui dukungan terhadap ketiga karakteristik tersebut anggota dalam suatu komunitas dapat lebih memahami jenis kegiatan yang terjadi, memahami norma-norma komunitas dan konsekuensi dari tindakan yang mungkin mereka ambil. *Social Translucence* dalam dunia digital berarti bahwa terjadinya interaksi antar anggota dalam komunitas *online* yang menjadikan lebih dekat selayaknya dunia nyata untuk meningkatkan keterlibatan yang lebih kompleksakan tetapi juga menjadikan keterbukaan dalam sebuah hubungan sosial [7].

## 2.4. Analisis Kebutuhan Sistem

Analisis kebutuhan sistem dapat dilakukan setelah proses pengumpulan data selesai, proses ini guna mendapatkan spesifikasi kebutuhan yang akan dibuat. Untuk membangun aplikasi BKD *Online* tersebut dibutuhkan data antara lain: data administrator, data dosen wilayah 3 Yogyakarta, data asesori BKD wilayah 3 Yogyakarta, data laporan BKD *Online*. Berdasarkan data-data tersebut maka aplikasi ini nanti nya dapat mengolah antara lain data admin, data *profile* dosen, data *profile* asesori, data laporan BKD *Online* baik laporan rencana kerja maupun laporan kinerja.

## 3. Hasil dan Pembahasan

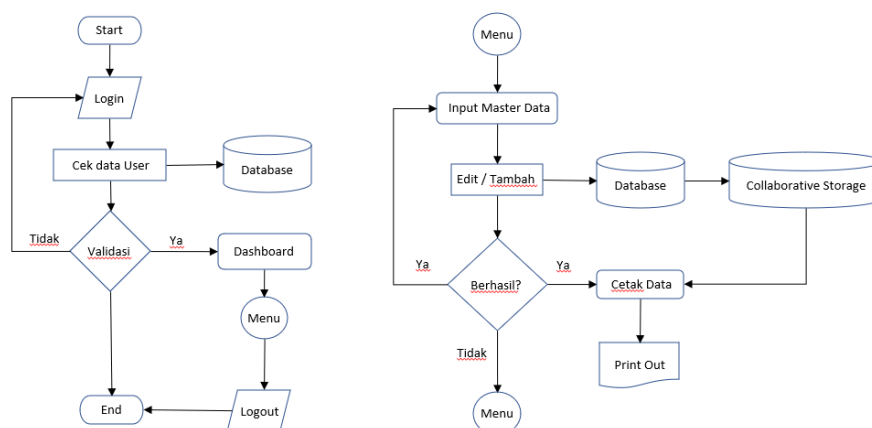
### 3.1. Sistem (Flowchart)

Fungsi utama dari *flowchart* adalah memberi gambaran jalannya sebuah program dari satu proses ke proses lainnya. Sehingga, alur program menjadi mudah dipahami oleh semua orang. Selain itu, fungsi

lain dari *flowchart* adalah untuk menyederhanakan rangkaian prosedur agar memudahkan pemahaman terhadap informasi tersebut. *Flowchart* adalah penggambaran secara grafik dari langkah-langkah dan urutan prosedur suatu program [8]. Bagan alir (*flowchart*) adalah bagan (*chart*) yang menunjukkan alir (*flow*) di dalam program atau prosedur sistem secara logika. Bagan alir digunakan terutama untuk alat bantu komunikasi dan untuk dokumentasi. Ada lima macam bagan alir, di antaranya:

- Bagan Alir Sistem (*system flowchart*) merupakan bagan yang menunjukkan arus pekerjaan secara keseluruhan dari sistem.
- Bagan Alir Dokumen (*document flowchart*) disebut juga bagan alir formulir (*form flowchart*) merupakan bagan alir yang menunjukkan arus dari laporan dan formulir termasuk tembusan-tembusannya.
- Bagan Alir Skematik (*schematic flowchart*) merupakan bagan alir yang menggambarkan prosedur di dalam sistem dengan menggunakan simbol-simbol bagan alir sistem dan gambar-gambar komputer serta peralatan lainnya yang digunakan oleh sistem.
- Bagan Alir Program (*program flowchart*) merupakan bagan yang menjelaskan secara rinci langkah-langkah dari proses program.
- Bagan Alir Proses (*process flowchart*) merupakan bagan alir yang banyak digunakan di teknik industri untuk menggambarkan proses dalam suatu prosedur

Rancangan aplikasi BKD *Online* ini menggambarkan detail transaksi yang dilakukan melalui aplikasi ini. Seperti pada Gambar 2.

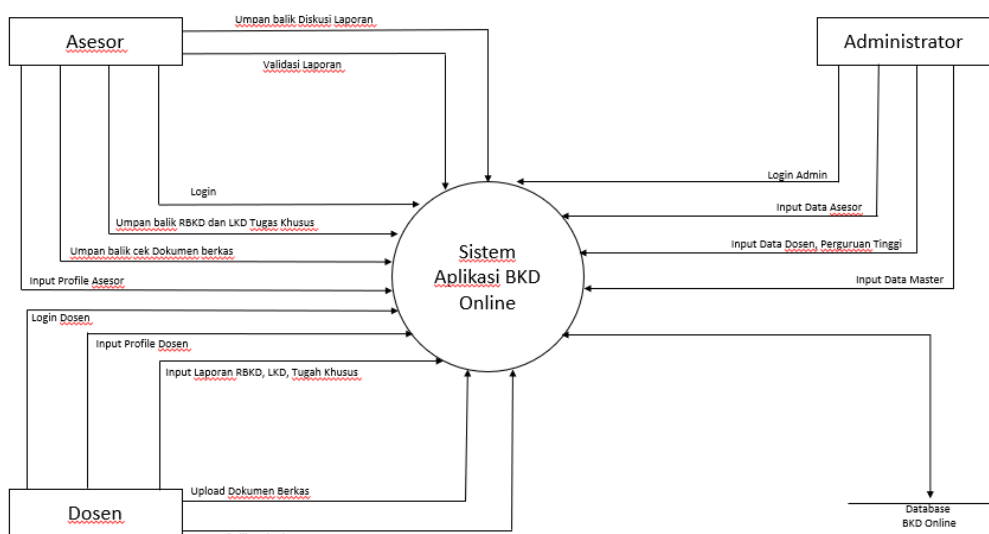


Gambar 2. *Flowchart* Aplikasi BKD *Online*

Dari urutan *flowchart* di atas terlihat alur proses dari aplikasi BKD *online* ini, proses *input* data dapat langsung di masukkan ke dalam *database* yang juga terhubung ke *storage* penyimpanan di luar *database system* utama, penggunaan *storage* dalam aplikasi ini menggunakan *link active* dari *storage* yang langsung dimasukkan pada *database*, jadi dalam aplikasi ini yang di-*input*-kan ke dalam *database* adalah *link* aktifnya saja untuk file lampirannya di masukkan ke dalam *storage*.

### 3.2. Data Flow Diagram (DFD)

*DFD* (*Data Flow Diagram*) adalah sebuah metode grafis untuk menggambarkan aliran informasi di dalam sistem. *DFD* menyediakan pemodelan yang jelas dan terstruktur untuk menggambarkan bagaimana data diproses dalam suatu sistem, mulai dari *input*, pemrosesan, hingga *output*. Pada perancangan ini *DFD* aplikasi BKD *Online* yang di tampilkan menunjukkan beberapa aktifitas aliran informasi dari dalam sistem seperti dari entity administrator ada aliran data yang di *input*-kan ke dalam sisten yaitu *input* data dosen, *input* data asesor dan *input* data master. Begitu juga dari entity dosen dan asesor ada proses *input* ke dalam sistem seperti *login*, *input* data laporan dan umpan balik dari laporan. Gambar untuk *DFD* gambar 3



Gambar 3. Data Flow Diagram level 0 pada Aplikasi BKD Online

Komponen penyusun dari DFD antara lain:

- 1) Proses (*Process*): Mewakili aktivitas atau operasi yang dilakukan pada data. Proses ini dapat berupa pemrosesan data, transformasi, perhitungan, penggabungan, atau pengiriman informasi. Pada aplikasi terlihat alur proses data informasi dari luar (*entity* terhadap sistem).
- 2) Aliran Data (*Data Flow*): Mewakili aliran data antara proses, entitas, dan luar sistem. Aliran data menunjukkan bagaimana data dikirim, diterima, atau diproses di dalam sistem.
- 3) Entitas Luar (*External Entity*): Mewakili entitas luar sistem yang berinteraksi dengan sistem yang sedang dianalisis. Entitas luar bisa berupa pengguna, sistem eksternal, atau perangkat keras/teknologi lainnya yang terhubung dengan sistem. *Entity* pada sistem adalah administrator, dosen dan asesor.
- 4) Penyimpanan Data (*Data Store*): Mewakili tempat penyimpanan data dalam sistem. Penyimpanan data bisa berupa basis data, file, atau memori yang digunakan untuk menyimpan dan mengambil informasi. Penyimpanan data pada sistem adalah *database* menggunakan MySQL dan *storage* berkas dengan Google drive.
- 5) Garis Panah (*Arrow*): Menghubungkan elemen-elemen DFD dan menunjukkan arah aliran data. Garis panah menunjukkan bagaimana data bergerak dari satu komponen ke komponen lainnya.

### 3.3. Entity Relationship Diagram (ERD)

#### a. Definisi ERD

Berdasarkan pengertian yang disampaikan Al-Bahra [9] “*Entity Relationship Diagram (ERD)* adalah diagram yang menunjukkan informasi dibuat, disimpan, dan digunakan dalam sistem bisnis”. *Entity Relationship Diagram (ERD)* merupakan gambaran data yang dimodelkan dalam suatu diagram yang digunakan untuk mendokumentasikan data dengan cara menentukan apa saja yang terdapat tiap *entity* dan bagaimana hubungan antara *entity* satu dengan lainnya.

#### b. Komponen Penyusun ERD

##### 1) Entitas (*Entity*)

Suatu kumpulan objek atau sesuatu yang dapat dibedakan atau dapat didefinisikan. Pada ERD, Entitas digambarkan dengan sebuah bentuk persegi panjang.

##### 2) Relasi (*Relationship*)

Hubungan yang terjadi antara satu entitas atau lebih. Relasi diberi nama dengan kata kerja dasar. Sehingga memudahkan untuk melakukan pembacaan relasinya (bisa dengan kalimat aktif atau kalimat pasif). Pada ERD, Relasi digambarkan dengan sebuah bentuk belah ketupat.

##### 3) Derajat Relasi (*Relationship Degree*)

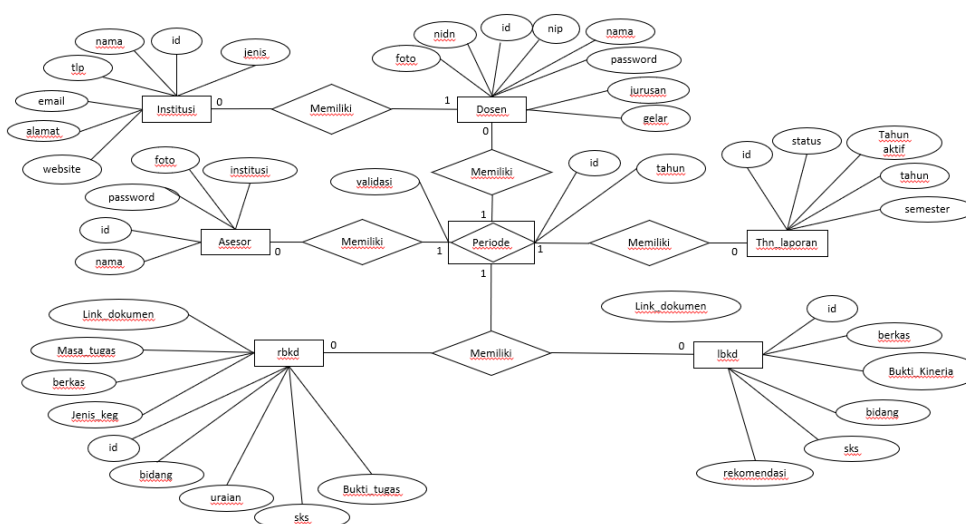
*Relationship degree* atau derajat relasi adalah jumlah entitas yang berpartisipasi dalam satu relasi. Derajat relasi yang sering dipakai di dalam ERD:

- a) *Unary Relationship* adalah model relasi yang terjadi di antara *entity* yang berasal dari

entity set yang sama. Sering juga disebut sebagai *recursive relationship* atau *relective relationship*.

- b) *Binary Relationship* adalah model relasi antar instansi-instansi (*instances*) dari suatu tipe entitas (dua *entity* yang berasal dari *entity* yang sama). *Relationship* ini paling umum digunakan dalam pembuatan model data.
- c) *Ternary Relationship* merupakan *relationship* antara instansi-instansi dari tiga tipe entitas secara sepihak. Masing-masing entitas mungkin berpartisipasi satu atau banyak dalam suatu *relationship ternary*. Perlu dicatat bahwa *relationship ternary* tidak sama dengan tiga *relationship binary*.
- d) Atribut (*Attribute*) Atribut merupakan karakteristik dari entitas atau *Relationship* yang menyediakan penjelasan detail entitas atau relation. Ada dua jenis atribut:
  - *Identifier (key)*, digunakan untuk menentukan suatu entitas secara unik (*primary key*).
- e) *Descriptor (nonkey attribute)* digunakan untuk menspesifikasikan karakteristik dari suatu entitas yang tidak unik. Pada umumnya atribut merupakan karakteristik dari entitas untuk relasi yang menyediakan penjelasan detail tentang entitas atau relasi tersebut. Kardinalitas (*Cardinality*) Menurut Simarmata [10] “Kardinalitas suatu hubungan menyatakan sejumlah kejadian terkait untuk masing-masing dua entitas”. Kardinalitas mendefinisikan jumlah kemunculan baik minimum maupun maksimum satu entitas yang dapat dihubungkan dengan kemunculan tunggal entitas lain. Jenis Derajat Kardinalitas:
  - *One to One(1:1)* Hubungan satu-ke-satu (1:1) terjadi jika sebanyak satu kejadian dari suatu entitas A dihubungkan dengan satu kejadian entitas B.
  - *One to Many / Many to One (1 : N / N : 1)* Tingkat hubungan satu ke banyak (1:N) adalah sama dengan banyak-ke-satu (N:1). Tergantung dari arah mana hubungan tersebut dilihat. Untuk satu kejadian pada entitas A dapat mempunyai banyak hubungan dengan kejadian pada entitas B. Sebaliknya satu kejadian pada entitas B hanya dapat mempunyai satu hubungan.
  - *Many to Many (M : N)* Hubungan banyak-ke-banyak (M:N) mempunyai arti setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, dan demikian juga sebaliknya, dimana setiap entitas pada himpunan entitas B berhubungan banyak entitas pada himpunan entitas A.

Dalam aplikasi BKD *online* ini ERD yang disusun dapat dilihat pada Gambar 4.



Gambar 4. Entity Relationship Diagram Aplikasi BKD Online

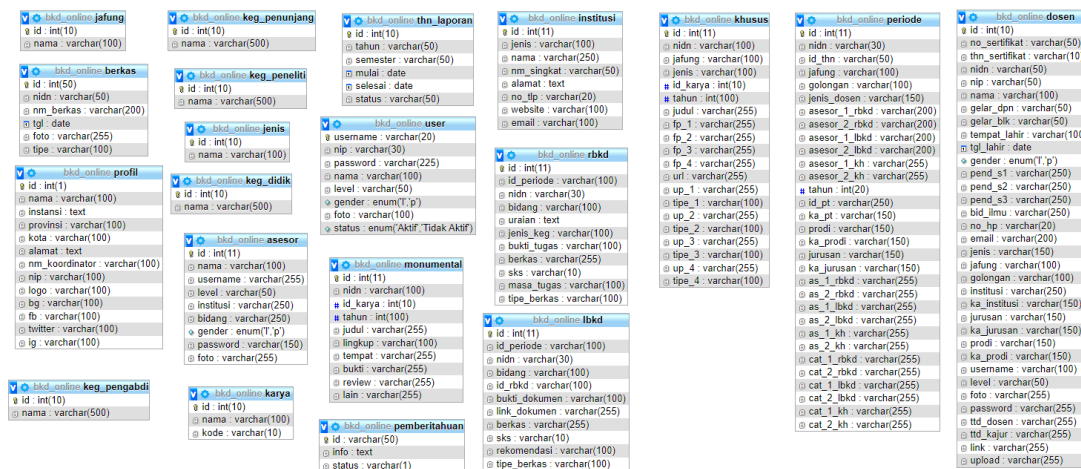


### 3.4. Basis Data (Database)

Basis data (*database*) adalah kumpulan terstruktur dari data yang saling terkait yang disimpan secara elektronik dalam suatu sistem komputer. Basis data menyediakan metode untuk menyimpan, mengatur, mengelola, dan mengambil data secara efisien.

*Database* (basis data) disusun menurut hasil analisa dari aplikasi sebelumnya, dan ditambahkan berdasarkan kebutuhan yang dirangkum dari hasil wawancara kepada tim kopertais dan beberapa dosen perguruan tinggi swasta di Yogyakarta. *Database* yang terdiri dari data dosen, data asesor, data perguruan tinggi, data master yang terdiri dari data jabatan fungsional, jenis dosen, jenis kegiatan Pendidikan, jenis kegiatan penelitian, jenis kegiatan pengabdian, kegiatan penunjang dan jenis karya, data. Untuk data transaksi terdiri dari *table* data periode pelaporan, *table* data rencana beban kerja dosen (rbkd), *table* laporan kinerja dosen (lkd), data kewajiban khusus, *table* profile dan pengaturan, juga tabel *user*.

Hasil perancangan pada *ERD* ditransformasikan dengan memilih DBMS MySQL sebagai basis data. Basis data ini digunakan untuk menyimpan data dosen, Asesor, tahun laporan, periode, rbkd, lbkd, dan data master lainnya. Gambar 5 adalah gambar implementasi *database* dari sistem yang dibangun menggunakan phpMyAdmin.

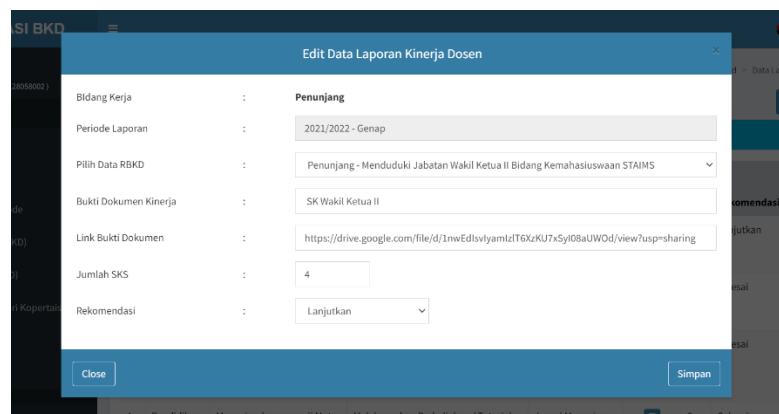


Gambar 5. Implementasi Basis Data Pada phpMyAdmin

### 3.5. Implementasi

Hasil perancangan pada DFD dan *ERD* kemudian diimplementasikan ke dalam basis data serta pengodean sistem menggunakan bahasa pemrograman PHP, Javascript, dan Bootstrap dengan menggunakan framework CodeIgniter versi 3. Codeigniter adalah sebuah pemograman dengan *framework* web yang menggunakan bahasa pemograman PHP, aplikasi ini dibuat pada tahun 2006, pembuatnya bernama Rick Ellis. EllisLab yaitu sistem kerja yang dimulai pada tahun 2002 dan bergerak dibidang pembuatan *software* dan *tool* yang banyak digunakan para *developer* web [11].

#### 3.5.1. Implementasi collaborative storage

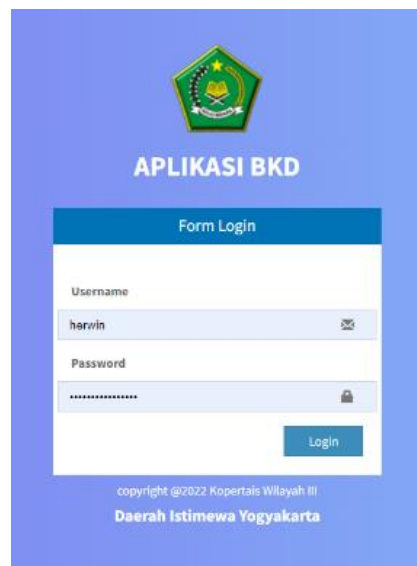


Gambar 6. Implementasi Collaborative Storage

Metode *collaborative storage* diterapkan dari aplikasi ini pada proses pengumpulan data lampiran untuk laporan BKD yang dilakukan pada setiap semester. Biasanya berkas lampiran yang cukup banyak dilampirkan dalam bentuk bundel laporan secara *hard copy*. Kondisi pandemi covid kemarin memaksa kita secara tidak langsung untuk meminimalisir penggunaan kertas dan juga meminimalisir interaksi antara seseorang dengan orang lain secara langsung. Untuk mendukung konsep ini maka dibuatlah sebuah konsep aplikasi di mana aplikasi ini mendukung untuk proses pengumpulan berkas dilakukan secara *soft file*. Mengingat berkas yang dikumpulkan cukup banyak maka di dalam aplikasi ini dibuat konsep penyimpanan dalam satu *storage* yang besar agar data file berkas tersebut tidak membebani server utama dari aplikasi ini. Fungsi lain dari *collaborative storage* yang digunakan dalam aplikasi ini adalah agar *file* data yang dikumpulkan bisa tersusun secara baik dan dapat diakses dari manapun saat dibutuhkan. Aplikasi *collaborative storage* yang digunakan dalam aplikasi ini adalah google drive cloud *storage*, dimana penggunaannya cukup mudah dan *simple*.

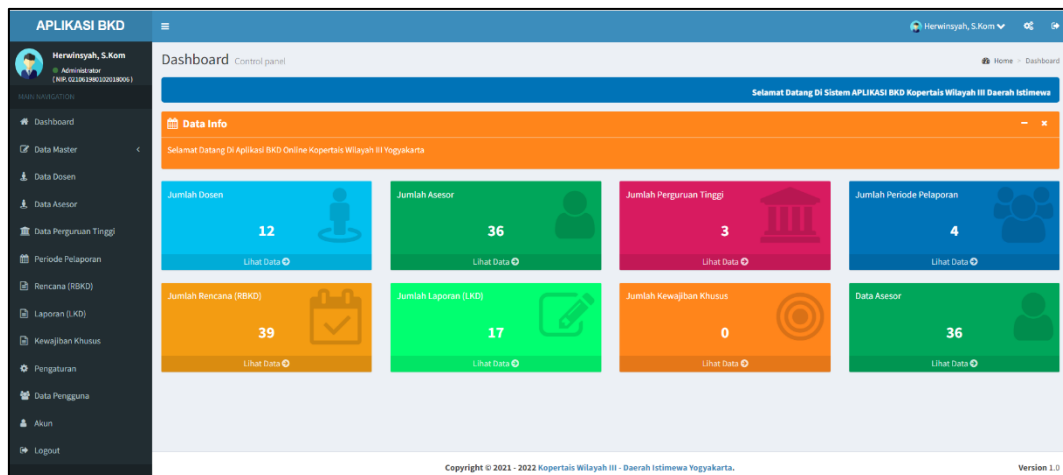
### 3.5.2. Implementasi UI/UX Tampilan Aplikasi

Tampilan awal dalam aplikasi ini implementasi fungsionalitas dari sistem yang diperoleh pada tahap analisis kebutuhan awal. Seluruh pengguna baik dari level admin sampai dengan level pengguna (dosen) diwajibkan *login* terlebih dahulu. Gambar 5 di bawah ini merupakan tampilan awal saat pengguna akan *login* ke dalam aplikasi. Tampilan antar muka untuk halaman *dashboard* admin terlihat rekapitulasi dari data yang ada di dalam aplikasi, seperti jumlah dosen, jumlah asesor, jumlah perguruan tinggi, jumlah periode laporan jumlah data RBKD, jumlah data LKD dan lainnya.



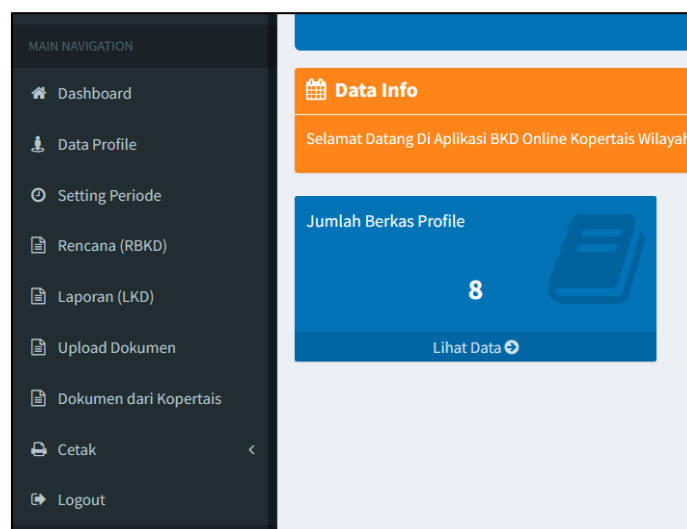
Gambar 7. Login Page





Gambar 8. Halaman *Dashboard Admin*

Implementasi *collaborative storage* pada aplikasi BKD *Online* diterapkan pada saat pengguna (dosen) melampirkan data berkas sebagai bukti tugas atau bukti kinerja yang telah dilakukan. Dokumen bukti tersebut wajib diuploadkan oleh dosen ke dalam penyimpanan data google drive yang telah disediakan oleh pihak kopertais, yang selanjutnya, dosen tinggal mengambil *link* dari dokumen tersebut dan disematkan dalam *input*-an yang ada di aplikasi. Google drive masing-masing dosen bisa diakses pada menu *upload* dokumen di menu dosen seperti gambar 9 di bawah ini.



Gambar 9. Halaman *Dashboard Dosen*

Implementasi *Social Translucence* pada aplikasi BKD *Online* ini bisa terlihat pada proses validasi dari asesor terhadap pelaporan yang dilakukan oleh dosen. Antara asesor satu dan asesor dua masing-masing dapat mengetahui apakah pelaporan yang dilakukan dosen telah disetujui atau belum disetujui. Tanda telah disetujui terlihat pada kolom validasi dari masing-masing asesor. Jika pelaporan yang dilakukan oleh dosen terdapat catatan dari asesor satu, maka asesor dua dapat melihat catatan apa yang diberikan oleh asesor satu, begitu pula sebaliknya. Tujuan dari konsep ini agar masing-masing asesor dapat saling mengetahui dan merekomendasikan hal yang sejalan terhadap laporan yang diberikan oleh dosen pelapor. Tampilan konsep ini dapat terlihat pada gambar 10 di bawah ini.

No	Nama Dosen	Jafung	Perguruan Tinggi	Kinerja	Asesor 1		Asesor 2		Data (LKD)	Kesimpulan
					Validasi	Catatan	Validasi	Catatan		
1	Dr. Farid Setiawan, S.Pd., M.Pd.I.	Lektor	FAI UAD	4	Di Setujui	Catatan	Di Setujui		<a href="#">Detail</a>	<a href="#">Lihat</a>
2	Drs. Nanang Nuryanta, M.Pd.	Lektor	FAI UII	4	Di Setujui	Catatan	Di Setujui		<a href="#">Detail</a>	<a href="#">Lihat</a>
3	Supriyanto, S.Ag., MCAA	Asisten Ahli	FAI UII	6	Di Setujui	Catatan	Di Setujui		<a href="#">Detail</a>	<a href="#">Lihat</a>
4	Dr. H. Munjahid, M.Ag	Lektor	IIQ AN NUR	5	Di Setujui	Catatan	Di Setujui	Catatan	<a href="#">Detail</a>	<a href="#">Lihat</a>
5	Dra. Hj. Sutinah, M.Pd	Lektor	STAIMS	6	Di Setujui	Catatan	Di Setujui	Catatan	<a href="#">Detail</a>	<a href="#">Lihat</a>
6	Sarwadi, M.Pd.I	Lektor	STITMA	4	Di Setujui	Catatan	Di Setujui		<a href="#">Detail</a>	<a href="#">Lihat</a>
7	Dr. Rubini, S.Pd.I., M.Pd.I.	Lektor	STAIMS	4	Di Setujui	Catatan	Di Setujui		<a href="#">Detail</a>	<a href="#">Lihat</a>
8	Ana Dwi Wahyuni, S.Pd.I., M.Pd.I	Asisten Ahli	STAIYO	3	Di Setujui	Catatan	Di Setujui	Catatan	<a href="#">Detail</a>	<a href="#">Lihat</a>

Gambar 10. Menu Laporan LKD pada Level Asesor

### 3.5.3. Pengujian Program

Sebuah aplikasi untuk mengetahui kekurangan maka pada proses terakhir biasanya dilakukan pengujian, agar saat aplikasi di gunakan tidak lagi ditemukan kesalahan ataupun *error* yang signifikan pada sistem. Metode yang biasanya digunakan dalam pengujian sebuah aplikasi menggunakan metode *black box*. Apabila dengan proses pengujian masih ada ditemukan kekurangan atau kelemahan pada *system*, maka aplikasi masih bisa dikembangkan atau diperbaiki pada bagian tersebut. Pengujian yang dilakukan akan dicatat pada tabel *case test*, dari hasil catatan tersebut nantinya akan diambil sebuah kesimpulan, apakah aplikasi telah berjalan dengan baik sesuai keinginan atau masih banyak kesalahan atau *error*. Setelah aplikasi BKD *Online* dilakukan pengujian, hasilnya tercatat seperti pada tabel *case test* di bawah ini.

Tabel 1. *Test Case* Aplikasi BKD *Online* dengan Pengujian *Blackbox*

No.	Pengujian	Deskripsi Pengujian	Hasil Yang diharapkan	Hasil Pengujian	Kesimpulan
1	Login	Memasukkan username dan password	Masuk Ke dalam aplikasi	Sesuai	Berhasil
2	Memasukkan <i>Input</i> kosong	Memasukkan username kosong atau password kosong	Tampil Popup "Data tidak boleh kosong"	Sesuai	Berhasil
3	<i>Input</i> Master data	Menambah dan Mengedit data	Data Tersimpan dan tampil	Sesuai	Berhasil
4	Menyimpan Link Aktif dari Google Drive <i>Cloud</i>	Link di masukkan pada Aplikasi	Link aktif dapat langsung digunakan	Sesuai	Berhasil
5	Mengganti Link Aktif	Mengedit Link Aktif	Setelah diganti link langsung berganti pada link yang baru	Sesuai	Berhasil
6	<i>Input</i> data Kegiatan RBKD	Menginput data kegiatan RBKD	Data Kegiatan RBKD tersimpat di <i>database</i>	Sesuai	Berhasil
7	Lagout	User yang telah logout tidak dapat kembali ke aplikasi tanpa melalui login	Jika telah Logout maka User harus login kembali untuk masuk ke dalam aplikasi	Sesuai	Berhasil

## 4. Kesimpulan

Perancangan aplikasi BKD *Online* secara tahapannya telah sesuai dengan yang diharapkan. Berdasarkan hasil pengujian aplikasi BKD *Online* telah menunjukkan berhasil. Menu-menu yang ditampilkan sudah sesuai dengan kebutuhan yang diinginkan oleh penyelenggara pelaporan BKD.

Penggunaan *collaborative storage* yang diinginkan juga telah sesuai dengan keinginan, dimana agar penggunaan data kapasitas pada server dapat diminimalisir, dan tujuan utama dari dibuatnya aplikasi ini agar dosen-dosen pada PTKIS di wilayah Yogyakarta dapat menggunakannya dengan mudah serta transparansi data yang di-*upload*-kan ke dalam aplikasi dapat digunakan oleh pihak kopertais untuk kelengkapan data lainnya. *System* penilaian yang dilakukan asesor juga dapat dengan mudah dipahami oleh dosen yang bersangkutan terkait dengan kekurangan data atau salah dalam memasukkan data, jadi baik dari dosen maupun dari asesor, sama-sama dapat memantau data yang dimasukkan di dalam aplikasi ini.

#### Daftar Pustaka

- [1] Dikti, *Pedoman Beban Kerja Dosen Dan Evaluasi Pelaksanaan Tridharma Perguruan Tinggi*. Jakarta: DIREKTORAT JENDERAL PENDIDIKAN TINGGI DEPARTEMEN PENDIDIKAN NASIONAL, 2010.
- [2] F. Nugraha, W. A. Triyanto, M. Arifin, and Y. Rahayu, "Rancang Bangun Sistem Informasi Penilaian Beban Kerja Dosen," *Simetris J. Tek. Mesin, Elektro dan Ilmu Komput.*, vol. 10, no. 1, pp. 405–412, 2019, doi: 10.24176/simet.v10i1.3312.
- [3] F. Adiputra and K. Mustofa, "Purwarupa Framework Aplikasi Desktop Menggunakan Teknologi Web," *IJCCS (Indonesian J. Comput. Cybern. Syst.*, vol. 9, no. 1, p. 23, 2015, doi: 10.22146/ijccs.6637.
- [4] E. R. Lai, "Collaboration : A Literature Review Research Report," *Res. Reports*, no. April, p. 41, 2011, [Online]. Available: <http://www.datec.org.uk/CHAT/chatmeta1.htm>.
- [5] F. Kurniawan, "Apa Itu *Cloud Storage* dan Mengapa Kita Harus Memakainya?," 2021. <https://tekno.sindonews.com/read/428764/207/apa-itu-cloud-storage-dan-mengapa-kita-harus-memakainya-di-2021-1621184750>.
- [6] C. L. Alali, F. A., & Yeh, "Cloud computing: Overview and risk analysis," *J. Inf. Syst.*, 2012.
- [7] T. Erickson and W. A. Kellogg, "Social Translucence: An Approach to Designing Systems that Support Social Processes," *ACM Trans. Comput. Interact.*, vol. 7, no. 1, pp. 59–83, 2000, doi: 10.1145/344949.345004.
- [8] Indrajani, *Database Design (Case Study All in One)*. Jakarta: PT. Elex Media Komputindo, 2015.
- [9] A.-B. Bin Ladjamudin, *Analisis dan Desain Sistem Informasi*. Yogyakarta: Graha Ilmu, 2005.
- [10] J. Simarmata, *Perancangan Basis data*. Yogyakarta: Andi, 2007.
- [11] Budi Raharjo, *Belajar Otodidak Framework CodeIgniter*. Bandung: Informatika Bandung., 2015.