

Jurnal

Vol.3 No.1 Juli 2018

# PATANI

Pengembangan Teknologi Pertanian dan Informatika

Perancangan Data Tabular UKM Pngolahan Hasil Pertanian  
Kabupaten Sambas berbasis Google Table  
**Heldi Hastriyandi, Junardi**

Analisis Risiko Pengelolaan Keuangan Desa  
**Asbeni, Sunardi**

Pengaruh Suplay Chain Management terhadap Kinerja Usaha Mikro  
Kecil dan Menengah (UMKM) di Kabupaten Sambas  
**Angga Tritisari**

Implementasi Metode Fuzzy Ahp pada Perancangan Penentuan  
Pembangunan Sarana dan Prasarana Penggalangan Dana Sosial  
Dari Sampah Kampus (Studi Kasus Kampus Politeknik Negeri  
Sriwijaya Palembang)  
**Isnainy, Maria Agustin, Ikhathison Mekongga, Ica Admirani**

Implementasi Basisdata Not Only SQL dalam Mendukung Kinerja  
Sistem Informasi Pusat Penelitian dan Pengabdian Kepada  
Masyarakat (SIP3M) Politeknik Negeri Sriwijaya  
**Zulkarnaini, Ridwan Effendi, Robinson, Deri Darfin**

Uji Pendahuluan Perlakuan Irisan Pertikal dan Rekayasa Produksi  
Pada Pohon Gaharu di Kawasan Hutan Paloh  
**Hidayat Asta, Ellys Mei Sundari**



POLITEKNIK NEGERI SAMBAS



9 772356 156007

ISSN 2356-1564

## SUSUNAN DEWAN REDAKSI

- Penanggung Jawab** : Kepala Pusat Penelitian dan Pengabdian  
Kepada Masyarakat (P3M)  
Politeknik Negeri Sambas
- Ketua Dewan Redaktur** : Theresia Widji Astuti, S.Kom., M.Kom
- Penyunting/Editor** : Harmoko, SP., M.Sc  
Heldi Hastriandi, ST., M.Kom  
Rini Fertiasari, S.P., M.Sc
- Desain Grafis** : Eko Pamungkas
- Mitra Bestari** : Denah Suswati (UNTAN)  
Fariastuti (UNTAN)

**Penerbit :**

Politeknik Negeri Sambas  
Jalan Raya Sejangkung – Sambas, 79462 Kalimantan Barat  
Telp (0562) 392701 Fax : (0562) 392723  
Email : [redaksi.patani@gmail.com](mailto:redaksi.patani@gmail.com)

## DAFTAR ISI

Volume 3 Nomor 2, November 2018

ISSN : 23561564

	Halaman
Perancangan Data Tabular UKM Pngolahan Hasil Pertanian Kabupaten Sambas berbasis Google Table <i>Heldi Hastriyandi, Junardi</i>	1 – 5
Analisis Risiko Pengelolaan Keuangan Desa <i>Asbeni, Sunardi</i>	6 – 13
Pengaruh <i>Suplay Chain Management</i> terhadap Kinerja Usaha Mikro Kecil dan Menengah (UMKM) di Kabupaten Sambas <i>Angga Tritisari</i>	14 – 22
Implementasi Metode <i>Fuzzy Ahp</i> pada Perancangan Penentuan Pembangunan Sarana dan Prasarana Penggalangan Dana Sosial Dari Sampah Kampus (Studi Kasus Kampus Politeknik Negeri Sriwijaya Palembang) <i>Isnainy, Maria Agustin, Ikhathison Mekongga, Ica Admirani</i>	23 - 33
Implementasi Basisdata <i>Not Only SQL</i> dalam Mendukung Kinerja Sistem Informasi Pusat Penelitian dan Pengabdian Kepada Masyarakat (SIP3M) Politeknik Negeri Sriwijaya <i>Zulkarnaini, Ridwan Effendi, Robinson, Deri Darfin</i>	34 - 43
Uji Pendahuluan Perlakuan Irisan Pertikal dan Rekayasa Produksi Pada Pohon Gaharu di Kawasan Hutan Paloh <i>Hidayat Asta, Ellys Mei Sundari</i>	44 - 49

# **Implementasi Basisdata Not Only SQL dalam mendukung kinerja Sistem Informasi Pusat Penelitian dan Pengabdian Masyarakat (SIP3M) Politeknik Negeri Sriwijaya.**

**Zulkarnaini<sup>1</sup>, Ridwan Effendi<sup>2</sup>, Robinson<sup>3</sup>, Deri Darfin<sup>4</sup>**

<sup>1,2,3</sup> *Jurusan Manajemen Informatika, Politeknik Negeri Sriwijaya*

<sup>4</sup> *Jurusan Pengembangan Kepribadian Politeknik Negeri Sriwijaya*

*Jl. Sriwijaya Negara, Palembang 30139 Telp.0711-353414 Fax.0711-355918*

*[zulkarnaini\\_mi@polsri.ac.id](mailto:zulkarnaini_mi@polsri.ac.id),*

*[ridwan\\_effendi\\_mi@polsri.ac.id](mailto:ridwan_effendi_mi@polsri.ac.id)*

## **ABSTRAK**

*Berkembangnya aplikasi berbasis web yang memerlukan pengolahan data dalam skala besar melahirkan paradigma baru dalam teknologi basis data. Beberapa website seperti Facebook, Twitter, Digg, Google, Amazon, dan SourceForge menyimpan dan mengolah data puluhan giga setiap harinya, dan total keseluruhan data yang disimpan oleh aplikasi tersebut sudah mencapai ukuran petabyte. Ukuran data yang sangat besar menimbulkan permasalahan dari segi skalabilitas, karena pertambahan data yang terjadi setiap saat. Peningkatan kemampuan server secara vertikal yang dimiliki Relational Database Management System (RDBMS) terbatas pada penambahan prosesor, memori, dan media penyimpanan dalam satu node server yang terbatas. Sedangkan peningkatan kemampuan server secara horizontal yang meliputi penambahan perangkat server baru dalam suatu jaringan memerlukan biaya yang mahal dan sulit dalam pengelolaannya. Salah satu cara yang diterapkan oleh website berskala besar untuk mengatasi permasalahan tersebut adalah dengan menggunakan NoSQL, sebuah paradigma basis data yang merelaksasikan aturan-aturan konsistensi yang terdapat pada basis data relasional. Jika RDBMS menggunakan aturan Atomicity, Consistency, Isolation, dan Durability (ACID) untuk penyimpanan dan pengolahan data, maka NoSQL menggunakan paradigma Basically Available, Soft State, and Eventually consistent (BASE) untuk merelaksasikan aturan tersebut. Hasilnya, NoSQL dapat mengolah data dalam jumlah besar dengan memartisi data ke dalam beberapa server secara lebih mudah. Makalah ini membahas dan menjelaskan latar belakang kemunculan, konsep dasar, dan penggunaan NoSQL.*

**Kata kunci :** *Basisdata, RDBMS, NoSQL dan SQL*

## **1. PENDAHULUAN**

Pusat Penelitian dan Pengabdian kepada Masyarakat Politeknik Negeri Sriwijaya merupakan salah satu unit yang bertugas mengelola kegiatan Tri dharma perguruan tinggi pada politeknik Negeri Sriwijaya. Disamping memberikan pelayanan kepada dosen untuk melakukan penelitian dan pengabdian kepada masyarakat unit ini juga memberikan informasi tentang dana-dana bantuan atau hibah dalam kegiatan penelitian dan pengabdian yang diperlukan oleh dosen. Selain itu data-data tentang semua kegiatan penelitian dan pengabdian kepada masyarakat yang telah dilakukan dosen-dosen Polsri disimpan dan diarsipkan untuk nanti dapat digunakan sebagai informasi dan bahan akreditasi lembaga Politeknik Negeri Sriwijaya.

Untuk menunjang kegiatan tersebut tentunya sistem informasi yang terkomputerisasi sangat diperlukan agar informasi dapat diberikan dengan cepat dan akurat sesuai dengan apa yang diinginkan. Selain aplikasi yang perlu disiapkan, sistem basis data yang mendukungpun perlu diperhatikan agar kemampuan dari sistem informasi yang dimiliki dapat lebih optimal dan efisien. Beberapa teknologi basisdata yang telah digunakan selama ini sudah kita kenal seperti Hierarchical Databases, Network Databases, Relational Databases, Object Oriented Database (OOD), dan lain sebagainya.

Perkembangan basisdata relasional merupakan basisdata yang telah dikenal luas oleh kalangan industri software maupun masyarakat. Model relasional yang telah menggantikan model basisdata sebelumnya dengan menggunakan prinsip-prinsip normalisasi, aljabar relasional dan Structure Query Language (SQL). Dalam banyak kasus penggunaan basisdata relasional merupakan solusi yang dipilih dalam pembuatan sistem informasi dikarenakan faktor konsistensi data sangat terjaga. Selain itu model basisdata relasional lebih mudah dipahami dan mudah untuk melakukan query yang kompleks.

Sejak munculnya Web, volume data yang diolah oleh sistem informasi meningkat dengan signifikan. Selain ketepatan dalam memodelkan kardinalitas suatu relasi dan kemampuan komponen-komponen perangkat keras sangat diperlukan dalam menangani jumlah data yang besar. Untuk meningkatkan kinerja server yang menangani basisdata relasional adalah dengan meningkatkan kapasitas dari server yang biasa disebut dengan *Vertical Server*. Dengan demikian biaya untuk membeli server yang berkapasitas lebih besar tentunya membutuhkan biaya yang sangat mahal.

Salah satu metode yang dapat digunakan untuk mengatasi penanganan volume data yang besar digunakan dengan penambahan perangkat baru pada jaringan dan mempartisi basis data agar dapat membagi penyimpanan datanya pada server yang lama dan yang baru. Metode penambahan perangkat ini akan lebih murah dari pada membeli perangkat server yang lebih besar, namun cara mempartisi database tidak mudah dilakukan oleh basisdata relasional dikarenakan aturan dari RDBMS konsistensi data yang diprioritaskan.

Untuk mengatasi kelemahan yang ada dari basisdata relasional tersebut adalah dengan menggunakan basisdata yang semi terstruktur/noSQL. Dengan demikian penelitian ini akan mengimplementasikan model basisdata NoSQL sebagai pendukung sistem informasi Pusat Penelitian dan Pengabdian kepada Masyarakat Politeknik Negeri Sriwijaya.

Disamping itu data-data yang diolah pada unit Penelitian dan Pengabdian Kepada Masyarakat Polri umumnya berbentuk dokumen baik hasil laporan penelitian, pengabdian ataupun tulisan-tulisan ilmiah yang berupa jurnal dan bentuk lain dari laporan tersebut umumnya dalam bentuk gambar ataupun grafik. Dengan demikian sesuai dengan kemampuan yang dimiliki oleh sistem basisdata NoSQL, maka dilakukan penelitian untuk mendukung aplikasi Penelitian dan Pengabdian Kepada Masyarakat dengan menggunakan sistem basisdata Not Only Structure Query Language.

## 1.2. Permasalahan

Pemakaian sistem basisdata umumnya selalu jadi bagian yang pokok dalam membangun suatu aplikasi. Baik aplikasi *Transaction Processing Systems (TPS)*, *Office Automation Systems (OAS)*, *Knowledge Work System (KWS)*, maupun *Informatic Management System* dan masih banyak lagi yang lainnya pasti menggunakan sistem basisdata. Selama ini pemakai sistem basisdata umumnya menggunakan sistem basisdata relasional yang terkenal dengan konsistensi datanya. Namun dengan berkembangnya kebutuhan akan volume data serta metode penanganannya maka sistem basisdata yang tidak berelasi mulai digunakan dan dikembangkan yang disebut dengan NoSQL.

Dengan menggunakan kelebihan-kelebihan yang ada pada sistem basisdata NoSQL maka permasalahan yang ada adalah membangun Sistem Informasi Pusat Penelitian dan Pengabdian Kepada Masyarakat Polri dengan menggunakan sistem basisdata NoSQL.

## 1.3 Referensi Penelitian Sebelumnya

Penelitian di bidang basisdata NoSQL sudah banyak dilakukan. Sejak tahun 2009 sistem database NoSQL sudah dikembangkan. Dalam penelitian ini diantaranya, 1. Tentang basisdata relasional dan tidak relasional; 2. Teknologi basisdata NoSQL; 3. Basisdata NoSQL dalam lingkungan aplikasi berbasis Web.

1. GC, Deepak, "A Critical Comparison of NOSQL Databases in the Context of Acid and Base" (2016). Culminating Projects in Information Assurance. Paper 8. Dalam paper ini membandingkan basisdata model SQL dan NoSQL berdasarkan struktur ACID.
2. Vatika Sharma and Meenu Dave dalam jurnalnya berjudul "SQL and NoSQL Databases" tahun 2012 melakukan penelitian tentang basisdata SQL dan NoSQL.

3. Madison Michael, Mark Barnhill, Cassie Napier, Joy Godin and Georgia College dalam jurnalnya berjudul “*NoSQL Database Technologies*” tahun 2015 yang menuliskan tentang *Big Data* dengan teknologi NoSQL.
4. Moniruzzaman ABM and Syed Akhter Hossain dalam jurnalnya berjudul “*NoSQL Database: New Era of Databases for Big Data Analytics - Classification, Characteristics and Comparison*” tahun 2015 membahas tentang basisdata NoSQL yang berkaitan dengan volume data.

## 2. TINJAUAN PUSTAKA

### 2.1. Basisdata Relasional

Menurut (Edgar F Codd, 1985) memperkenalkan model data *relasional* yang diciptakan berdasarkan teori-teori *relational algebra* dan *relational calculus*. Codd telah menciptakan serangkaian operasi matematika relasional pada model relasional tersebut. Model relasional tersebut sering juga disebut dengan RDBMS (*Relational Database Management Sistem*).

Pada model relasional, basis data disebar dan dipilah-pilah kedalam tabel-tabel yang dalam lajur mendatar disebut *row/record* dan lajur vertikal yang disebut kolom (*colom/field*).

#### 2.1.1 Kelebihan basisdata Relasional (RDBMS)

Model database relasional atau yang dikenal dengan RDBMS adalah suatu sistem yang mengatur organisasi, penyimpanan, akses, keamanan dan integritas data dalam bentuk tabel-tabel. Tabel-tabel tersebut dapat saling berhubungan (relasi) satu sama lain dalam satu kesesuaian aturan pada suatu sistem database relasional. Suatu desain RDBMS dikatakan bagus jika normalisasi data dilakukan dan diterapkan dalam desain tersebut.

Model databases relasional fokus pada konsep *ACID* (*Atomicity, Consistency, Isolation, and Durability*) dalam penyimpanan dan pengelolaan data [Travory Ran 2011].

- **Atomicity** : Yaitu suatu aturan dalam menerapkan atau modifikasi basisdata harus sesuai dengan aturan atau tidak sama sekali (*all or nothing*). Suatu transaksi dikatakan *atomic* jika transaksi mengalami kegagalan maka menyebabkan semua transaksi gagal.
- **Consistency** : Semua transaksi pada basisdata harus memiliki state yang jelas. Manakala proses transaksi pada suatu databases berhasil dijalankan maka dilakukan pengambilan basisdata dari suatu kondisi yang konsisten dengan aturan-aturannya menuju ke kondisi yang lain dengan aturan-aturan yang lain yang berlaku.
- **Isolation** : Semua transaksi tetap dalam koridornya masing-masing, dengan kata lain suatu transaksi tidak boleh mempengaruhi proses atau terpengaruh oleh proses yang lain.
- **Durability** : Suatu transaksi yang dilakukan tidak boleh hilang walaupun komputer dilakukan restart, penyimpanan data harus permanen walaupun kemudian mungkin banyak terjadi kegagalan.

#### 2.1.2 Kelemahan basisdata Relasional (RDBMS)

- **Volume data** : Saat kita membutuhkan penyimpanan data yang “relatif” besar, namun terdapat keterbatasan resources (komputer/server), maka butuh resources yang sangat besar untuk membuat database relasional memiliki performa yang baik bila datanya besar. Peningkatan resources tersebut hanya dapat dilakukan pada *node server*, jika kemampuan suatu node sudah maksimal maka untuk meningkatkan kemampuan server harus membeli perangkat yang lebih besar yang tentunya akan lebih mahal.
- **Data tidak terstruktur** : Misal Dokumen atau biasa disebut sebagai *Schemaless Data Representation*. Bayangkan data yang masuk ke database terkadang hanya memiliki 4 kolom, tapi dilain waktu memiliki jumlah kolom yang berbeda lagi. Di Database relasional kejadian seperti ini akan memaksa kita mengubah struktur DB (*schema*) yang biasanya sangat fatal, terutama apabila datanya besar. Perubahan *schema* adalah bencana besar bagi sistem IT. Karena aplikasi harus direvisi mengikuti perubahan tersebut. Pada terminologi *Big Data*, ini terkait dengan “*Variety*”, yaitu semakin beragamnya tipe/struktur/format data yang masuk ke database.
- **Optimasi Query** : Sebuah query dalam basisdata relasional merupakan penentu dalam mendapatkan data yang diinginkan bukan bagaimana cara mendapatkannya [Ghazal A]. Untuk mempercepat

proses mendapatkan data maka databases relasional menggunakan kunci-kunci primer, kunci skunder dan Index. Jika data bertambah maka akan terdapat permasalahan baru yang mengiringi proses penambahan data tersebut [Arif. N], misal query yang kurang tepat akan menghambat kinerja basisdata yang sudah ada.

- Data Complexity : Data yang disimpan tersebar atau terdistribusi akan merupakan hambatan pada basisdata relasional.

## 2.2. Basisdata NoRelasional (NoSQL)

Konsep basis data *NoSQL* mempunyai fitur utama dalam hal distribusi data dan query dalam beberapa server yang sama. Beberapa produk basis data *NoSQL* dengan berbagai keunggulannya masing-masing telah diimplementasikan pada berbagai aplikasi. Beberapa diantaranya adalah CouchDB yang memiliki nilai konkrusensi dan *RESTfull HTTP request*, *simpleDB* yang memiliki kesederhanaan dan fleksibilitas dalam pemeliharaannya, dan Google BigTable yang memiliki kuota yang terbatas [Bhat Uma].

Eric brewer [2] menyatakan bahwa NoSQL didasarkan pada teori CAP yaitu pemilihan dua dari tiga aspek yang ada yang harus dipenuhi oleh basis data yaitu *Consistency*, *Avaibility*, dan *Partition-Tolerance*. (1) *Consistency Avaibility (CA)* berseberangan dengan *Partition-Tolerance* dan biasanya berhubungan dengan replikasi. (2) *Consistency PartitionTolerance (CP)* berseberangan dengan *Avaibility* dalam penyimpanan data. (3) *Avaibility Partition-Tolerance* sistem mencapai kondisi eventual consistency melalui replikasi dan verifikasi yang konsisten dalam *node* yang telah terbagibagi. Dalam basis data *NoSQL* penerapan konsep tersebut diterjemahkan dalam empat konsep dasar yaitu *Non-Relational*, *MapReduce*, *Schema Free*, dan *Horizontal Scaling*.

### 2.2.1 Non-Relational

Konsep *Non-Relational* dalam basis data *NoSQL* meliputi hirarki, graf, dan basis data berorientasi obyek yang sudah terlebih dahulu ada sejak tahun 1960 sebelum akhirnya basis data relasional muncul pada tahun 1970 [Leavitt. N]. Penggunaan basis data non-relasional kembali merebak seiring dengan bertambahnya aplikasi berbasis web yang memerlukan banyak penyimpanan data. Meskipun memiliki kelemahan pada konsistensi dan redundansi data, namun basis data non-relasional dapat menyelesaikan beberapa permasalahan terkait dengan *avaibility*, dan *partition-tolerance*. Tugas pengecckan konsistensi dan redundansi data diserahkan pada sisi aplikasi, sedangkan basis data nonrelasional hanya bertugas memanipulasi penyimpanan saja.

### 2.2.2 MongoDB

*MongoDB* merupakan sebuah sistem basis data yang berbasis dokumen (*Document Oriented Database*) dan termasuk sistem basisdata yang menganut paham NoSQL. NoSQL singkatan dari Not Only SQL, artinya sebuah sistem basis data tidak hanya harus menggunakan perintah SQL untuk melakukan proses manipulasi data. *MongoDB* tidak memiliki tabel, kolom, dan baris. Pada *MongoDB* yang ada hanyalah koleksi dan dokumen. Dokumen yang terdapat dalam *mongoDB* dapat memiliki atribut yang berbeda dengan dokumen lain walaupun berada dalam satu koleksi. Hal ini tidak dapat dilakukan dalam RDBMS, dimana sebuah baris dalam tabel tidak mungkin memiliki kolom yang berbeda dengan baris yang lain jika berada dalam satu tabel. *MongoDB* merupakan sistem basis data yang menggunakan konsep *key-value*, artinya setiap dokumen dalam *mongodb* pasti memiliki *key*. Hal ini berbeda dalam RDBMS yang bisa tidak menggunakan *primary key* ketika membuat sebuah tabel. Sehingga walaupun kita membuat sebuah dokumen tanpa menggunakan *primary key*, tapi secara otomatis *mongodb* memberinya sebuah *key*. Penggunaan konsep *key-value* berperan sangat penting, karena hal ini membuat *mongodb* menjadi sistem basis data yang sangat cepat jika dibandingkan dengan non *key-value* seperti RDBMS.

### 2.2.3 Pembangunan perangkat lunak berorientasi objek

Fokus utama metodologi ini pada objek, dengan melihat suatu sistem terdiri dari objek yang saling berhubungan. Objek dapat digambarkan sebagai benda, orang, tempat dan sebagainya yang mempunyai atribut dan metode. Metodologi

terdiri dari pembuatan model dari domain aplikasi, kemudian menambahkan detail implementasi pada saat desain dari suatu sistem. Tahap-tahap metodologi berdasarkan Sistem Development Life Cycle (SDLC) digunakan dengan memperhatikan karakteristik khusus berorientasi objek, dan dapat dijelaskan sebagai berikut :

- Analisa

Analisa berorientasi objek dimulai dengan menyatakan suatu masalah, analisis membuat model situasi dari dunia nyata, menggambarkan sifat yang penting. Analisis harus bekerja dengan pihak yang membutuhkan sistem untuk memahami masalah tersebut. Model analisa adalah abstraksi yang ringkas dan tepat dari apa yang harus dilakukan oleh sistem, dan bagaimana melakukannya. Objek dalam model harus merupakan konsep domain dari aplikasi, dan bukan merupakan implementasi komputer seperti struktur data. Model yang baik harus dipahami dan ditanggapi oleh ahli aplikasi. Empat kesulitan yang merupakan gangguan utama sistem adalah memahami problem domain, komunikasi antara pihak yang berkaitan, perubahan kontinyu, dan reuse (penggunaan kembali).

- Desain

Desain Berorientasi Objek atau Object Oriented Design (OOD) merupakan tahap lanjutan setelah Analisis Berorientasi Objek dimana tujuan sistem diorganisasikan ke dalam sub-sistem berdasar struktur analisis dan arsitektur yang dibutuhkan. System designer menentukan karakteristik penampilan secara optimal, menentukan strategi memecahkan masalah, dan menentukan pilihan alokasi sumberdaya. Sebagai contoh, system designer mungkin menentukan perubahan pada screen untuk workstation yang memerlukan kecepatan serta resolusi lebih tinggi. Desain model berdasarkan model analisa tetapi berisi detail implementasi. Fokus dari object design adalah perencanaan struktur data dan algoritma yang diperlukan untuk implementasi setiap kelas. Objek domain aplikasi dan objek domain komputer dijelaskan dengan menggunakan konsep dan notasi berorientasi objek yang sama.

- Pengkodean

Tahapan pengujian dilakukan untuk melihat kesesuaian kebutuhan perangkat lunak yang telah dibangun. Pengujian perangkat lunak terdiri dari 2 jenis utama yaitu pengujian *White Box* dan pengujian *Black Box*. Pengujian *White Box* dilakukan untuk mengetahui sebagian besar dari tugas modul atau fungsi yang ada sudah berjalan sesuai dengan yang diharapkan atau tidak. Pengujian ini berfokus pada proses jalannya sistem. Sedangkan pengujian *Black Box* dilakukan hanya untuk mengetahui *output* dari satu proses, atau dengan kata lain hanya memamandang hasil dari suatu proses tidak meninjau jalannya proses.

### 3. METODE PENELITIAN

Dalam melakukan penelitian ini, dilakukan dengan beberapa tahap :

#### 3.1 Pengumpulan Data dan Kebutuhan

##### a. Studi Literatur

Metode pengumpulan data dengan studi literatur dilakukan dengan mengkaji literatur-literatur yang terkait dengan masalah penelitian yaitu pembangunan Sistem Informasi berbasis objek dan menggunakan manajemen databases Not Only SQL (NoSQL). Adapun literatur-literatur tersebut didapat dari :



- Jurnal-jurnal penelitian dan buku-buku tentang konsep pembangunan perangkat lunak berbasis objek dan basisdata NoSql baik dari dalam negeri maupun dari luar negeri.
- Data hasil penelitian Perguruan Tinggi Negeri
- Informasi dari media masa, seperti surat kabar dan internet.

b. Survei

Pelaksanaan metode ini dengan melakukan observasi, kuesioner dan wawancara langsung pada ketua Unit Pusat Penelitian dan Pengabdian Kepada Masyarakat Politeknik Negeri Sriwijaya Palembang beserta stafnya. Peneliti mendatangi langsung dan meminta masukan tentang kebutuhan dan proses bisnis dari sistem yang berjalan serta luaran yang diharapkan oleh sistem yang baru dengan berbasis web agar dapat dengan mudah digunakan oleh seluruh sivitas akademika polsri dalam melaksanakan Tri dharma perguruan tinggi dibidang penelitian dan pengabdian kepada masyarakat secara *on line*.

### 3.2. Pembangunan perangkat lunak Sistem Informasi Pusat Penelitian dan Pengabdian Kepada Masyarakat (SIP3M) berorientasi objek

Metode pengembangan perangkat lunak SIP3M yang akan digunakan pada penelitian ini adalah metode spiral. Secara umum metode spiral memiliki lima fase yaitu : (1) Analisis, (2) Desain, (3) Pengkodean, (4) Pengujian, (5) Pemeliharaan. Metode spiral merupakan pengembangan dari metode spiral. Metode ini dipilih dalam pengembangan perangkat lunak karena metode ini mengizinkan pengembang kembali ke fase sebelumnya ketika ditemukan sebuah kesalahan. Sebagai contoh jika saat ini pengembang berada di fase pengkodean dan pengembang menemukan kesalahan terjadi di fase analisis maka pengembang dapat kembali ke fase analisis. Pada fase analisis tersebut pengembang dapat memperbaiki kesalahan tersebut sampai benar. Perangkat lunak ini nantinya akan dibuat dengan menggunakan bahasa pemrograman PHP menggunakan IDE (*Integrated Development Environment*) Notpad++ dan DBMS (*Database Management System*) *mongodb* di atas Sistem Operasi Windows 7 (*seven*). Pemilihan bahasa pemrograman PHP, Notpad++ dan *mongodb* dilakukan karena bersifat *freeware* kecuali untuk sistem operasi Windows 7 (*seven*).

Berikut ini penjelasan tahapan pengembangan perangkat lunak menggunakan metode spiral

a. Analisis

1. *Pengumpulan kebutuhan*: data yang dikumpulkan adalah kebutuhan primer dan sekunder. Kebutuhan primer berasal dari data yang diambil langsung dari para *user* dan *stakeholder* pengguna perangkat lunak SIP3M seperti dosen, mahasiswa, dan administrator pengelola unit Pusat Penelitian dan Pengabdian Kepada Masyarakat Politeknik Negeri Sriwijaya.. Tahap ini sering disebut juga dengan tahapan akuisisi kebutuhan perangkat lunak. Teknik yang digunakan untuk mengumpulkan kebutuhan adalah wawancara, observasi, kuesioner, dan dokumentasi.
2. *Analisis kebutuhan*: Kebutuhan yang telah terkumpul diinterpretasi maksud dari kebutuhan yang didapat. Kebutuhan-kebutuhan yang berasal dari *user* dan *stakeholder* dianalisis untuk menentukan prioritas implementasinya. Interpretasi sebuah kebutuhan dapat berupa data, batasan, aturan, skenario, dan fungsionalitas perangkat lunak.
3. *Analisis basis data*: kasus yang telah terkumpul kemudian dibentuk sesuai dengan format basis data supaya bisa dimasukkan ke dalam basis data seperti pembuatan struktur data dengan format JSON.
4. *Analisis model aplikasi*: model aplikasi yang akan digunakan berbasis *web* yang nantinya dapat didistribusikan secara melalui internet.
5. *Analisis perangkat lunak*: perangkat lunak yang digunakan adalah bahasa pemrograman PHP, IDE Notpad++, *mongodb* serta Sistem Operasi Windows 7 (*seven*).
6. *Analisis perangkat keras*: Analisa kebutuhan perangkat keras terhadap sistem seperti kecepatan *processor*, kapasitas memori utama dan memori sekunder.

b. Perancangan

1. *Rancangan basis data*: Rancangan basis data merupakan lanjutan dari analisa basis data. Perancangan basis data dilakukan dengan menggunakan DBMS NoSQL serta melakukan pembuatan struktur data yang nanti akan diimplementasikan oleh DBMS *mongodb*.

2. *Rancangan antarmuka*: Merancang tampilan masukan dan keluaran yang berbasis GUI (*Graphical User Interface*) menggunakan IDE Notepad++.
  3. *Rancangan Objek* : Merancang modul-modul program dalam bentuk objek yang nantinya akan digunakan pada saat pengkodean sistem. Rancangan modul dapat berbentuk algoritma, notasi UML dan *pseudo-code*.
- c. Pengkodean
1. *Pembuatan kode modul basis data*: kode modul basis data dibuat terpisah dengan kode sistem sehingga lebih bersifat *reusable*. Kode modul basis data berisi operasi basis data seperti membuat koneksi ke basis data, *insert*, *update*, *delete* dan *query*.
  2. *Pembuatan kode komponen program*: kode program dibuat dan dimodularisasi menjadi objek berdasarkan spesifikasi yang telah ditetapkan dari hasil analisis.
- d. Pengujian
1. *Pengujian basis data*: pengujian koneksi basis data dan akurasi *query* basis data.
  2. *Pengujian sistem*: pengujian secara keseluruhan dari sistem baik dari masukan, proses dan keluaran sistem.
  3. *Pengujian program*: pengujian ini dilakukan untuk mengetahui kualitas objek perangkat lunak.
- e. Pemeliharaan
- Dilakukan dengan dua cara yaitu ketika proses pengembangan berlangsung melakukan *backup* kode-kode program yang dibuat jika melakukan revisi program dan setelah proses pengembangan dengan melihat kinerja sistem apakah masih menghasilkan akurasi yang baik selama sistem berjalan.

### 3.3. Lokasi Penelitian

Lokasi penelitian dilakukan di lingkungan Kampus Politeknik Negeri Sriwijaya pada Unit Penelitian dan Pengabdian Kepada Masyarakat, Politeknik Negeri Sriwijaya, Palembang – Sumatera Selatan.

## 4. HASILDAN PEMBAHASAN

Dalam implementasi difokuskan pada pembangunan database dengan menggunakan MongoDB yang merupakan system basis data berbasis dokumen (*Document Oriented Database*) dengan menganut paham NoSQL. Pada system informasi ini dokumen data yang diperlukan terdiri dari beberapa dokumen yaitu dokumen Admin, Dosen, Berita, dan Usulan. Dokumen Admin berfungsi untuk menyimpan data administrator dari sistem yang mengatur kebutuhan data-data yang akan ditampilkan atau diunggah atau diunduh sebagai informasi yang akan diberikan kepada masyarakat yang dalam hal ini dosen-dosen peneliti.

Dokumen admin disusun dengan komposisi sebagai berikut :

Tabel 1. Dokumen Admin

Key	Type	Deskripsi
<u>_id</u>	ObjectId	Collection Primary Key
nama	String	Nama Operator /Admin
username	String	User Name Admin
password	String	Password Admin
Icon_profil	String	Icon Profil Admin

Pada dokumen admin akan disimpan data-data administrator dari sistem informasi P3M Polstri. Data administrator dapat ditambah, diperbaiki, atau di hapus sesuai dengan criteria yang diinginkan.

Tabel 2. Dokumen Dosen/Anggota

Key	Type	Deskripsi
_id	ObjectId	Collection Primary Key
nip	String	Nomor Induk Pegawai
nama	String	Nama Dosen/Anggota
username	String	User Name Dosen/Anggota
password	String	Password Anggota/Dosen
kode_jurusan	String	Kode Jurusan Dosen
nama_jurusan	String	Nama Jurusan Dosen
penelitian	Object	Penelitian yang sudah dilakukan
tanggal_penelitian	Date	Tanggal dilakukan penelitian
judul_penelitian	String	Judul penelitian
sebagai_ketua	Boolean	Posisi dalam struktur peneliti
didanai_oleh	String	Sumber dana
biaya	Double	Besar biaya penelitian
anggota	Object	Anggota peneliti
nip_anggota	String	Nip anggota peneliti
nama_anggota	String	Nama Anggota peneliti
jurusan	String	Jurusan Peneliti
pengabdian	String	Pengabdian yang dilakukan
tanggal_pengabdian	Date	Tanggal dilakukan
judul_pengabdian	String	Judul pengabdian
sebagai_ketua	Boolean	posisi dalam struktur
didanai_oleh	String	Sumber dana
biaya	Double	Besar biaya pengabdian
anggota	Object	Anggota pengabdian
nip_anggota	String	Nip anggota pengabdian
nama_anggota	String	Nama anggota pengabdian
jurusan	String	Jurusan
pendidikan	String	bidang pendidikan
tanggal_buku	Date	Tanggal pembuatan buku
judul_buku	String	Judul buku ajar
sebagai_ketua	Boolean	posisi dalam struktur buku
didanai_oleh	String	Sumber dana
biaya	Double	Besar biaya
anggota	Object	Anggota tim buku ajar
nip_anggota	String	Nip anggota tim buku ajar
nama_anggota	String	Nama anggota tim buku ajar
jurusan	String	Jurusan anggota

Tabel 2. Menggambarkan Dokumen anggota/dosen yang akan didaftarkan sebagai anggota peneliti. Untuk setiap dosen melakukan registrasi agar mendapatkan username dan password pribadi yang tujuannya menjaga data-data masing-masing dosen/anggota dengan aman.

Selain itu dalam dokumen ini juga menyimpan data-data penelitian, pengabdian dan kegiatan pendidikan seperti membuat buku ajar atau bahan ajar lainnya. Dengan demikian maka data-data kegiatan tridharma perguruan tinggi seorang dosen akan tersimpan didalam penyimpan elektronik/server polsri.

Tabel 3. Dokumen Berita

Key	Type	Deskripsi
_id	ObjectId	Collection Primary key

judul	String	Judul berita
tanggal	Date	Tanggal terbit berita
sumber berita	Sumber String	Sumber berita Isi

Tabel 3 fungsinya untuk menyampaikan berita dari administrator ke anggota/dosen peneliti. Jenis berita dapat berupa informasi, pengumuman atau berita yang merupakan informasi bagi peneliti baik mengenai sumber dana ataupun besarnya dana yang disiapkan bagi peneliti. Selain penelitian table 3 juga digunakan untuk menyimpan data informasi tentang pengabdian kepada masyarakat dan juga tentang buku ajar.

Tabel 4. Dokumen Usulan

Key	Type	Deskripsi
_id	ObjectId	Collection Primary key
nip	String	Nomor Induk Pegawai
nama	String	Nama pengusul proposal
tanggal	Date	Tanggal usulan
jenis	Object	jenis kegiatan
nama_kegiatan	String	Nama kegiatan
sumber_kegiatan	String	Sumber kegiatan
sumber_dana	String	Sumber dana kegiatan
biaya	Double	Biaya kegiatan
anggota	Object	Anggota tim buku ajar
nip_anggota	String	Nip anggota tim buku ajar
nama_anggota	String	Nama anggota tim buku ajar
jurusan	String	Jurusan anggota
status_usulan	String	Status usulan diterima atau tolak

Tabel 4. Berfungsi untuk menyimpan data usulan proposal dari anggota/dosen yang akan ikut dalam kegiatan dengan mengusulkan proposal. Usulan yang dilakukan oleh dosen merupakan masukan bagi administrator dan selanjutnya akan dilakukan seleksi dan memberikan status usulan apakah diterima dan didanai atau ditolak

## 5. KESIMPULAN DAN SARAN

### 5.1. Kesimpulan

- Berdasarkan hasil yang didapat Sistem Informasi Pusat Penelitian dan Pengabdian kepada masyarakat Polstri dapat dibangun dan dikembangkan dengan menggunakan basis data yang bukan relasional atau NoSQL dengan sistem manajemen basisdata *mongodb*.
- Karna sistem basis data *mongodb* mempunyai kelemahan dalam mengolah data yang bersifat transaksi maka untuk sistem informasi Pusat Penelitian dan Pengabdian kepada Masyarakat polri dapat dibangun karna data yang diolah semuanya bersifat berita bukan transaksi.

### 5.2. Saran

Perlu diadakan pelatihan kepada semua anggota/dosen dalam hal cara pemakaian sistem informasi yang telah dibangun agar proses kegiatan pengabdian dan penelitian akan lebih lancar.

## DAFTAR PUSTAKA

- [1 A. Ghazal and R. Bhashyam, "Dynamic Constraints Derivation and Maintenance in the Teradata RDBMS," International Conference on Database and Expert Systems Applications, 2001, pp. 390-399
- Eben Hewitt, *Cassandra: The Definitive Guide*, United States of America, O'Reilly Media, Inc 2010
- Fowler, M.; Beck, K.; Brant, J.; Opdyke, W. F.; Roberts, D. 2002. *Refactoring: Improving the Design of Existing Code*. Addison-Wesley.
- Jacobson dan Pan Wei Ng. 2004. *Aspect-Oriented Software Development with Use Cases*. Addison Wesley. USA.
- Jeffrey Dean and Sanjay Ghemawat, *MapReduce: Simplified Data Processing on Large Clusters*, Proceeding of OSDI, 2004
- Joe Lennon, *Beginning CouchDB*, New York, Apress 2009
- N. Arif, *NoSQL: the End of RDBMS?* <http://arifn.web.id/blog/2010/05/05/nosql-the-end-of-RDBMS.html>
- N. Leavitt. *Will NO-SQL Basis datas Live Up to Their Promise?* IEEE Computer Society, Volume 43, Issue 2, 2010.