

## BAB II DASAR TEORI

### 2.1 Sistem Informasi E-Government

Pemanfaatan teknologi untuk menunjang pekerjaan manusia saat ini merupakan salah satu hal yang tidak dapat dipungkiri. Begitu juga dalam pemanfaatan teknologi dalam ranah pemerintahan, baik itu pemerintah pusat maupun pemerintah daerah. E-Government merupakan penerapan penggunaan teknologi pada layanan masyarakat yang ada dalam pemerintahan.

Menurut Indarajit (2002:36) E-Government merupakan suatu mekanisme interaksi baru antara pemerintah dengan masyarakat dan kalangan lain yang berkepentingan, dengan melibatkan penggunaan teknologi informasi (terutama internet) dengan tujuan memperbaiki mutu (kualitas) pelayanan. E-Government adalah penyelenggaraan pemerintahan berbasis elektronik untuk meningkatkan layanan publik secara efisien, efektif dan interaktif. Dimana pada intinya E-Government adalah penggunaan teknologi yang dapat meningkatkan hubungan antara pemerintah dan pihak-pihak lain (penduduk, pengusaha, maupun instansi lain).

Sementara itu, E-Government menurut World Bank Group "*E-Government" refers to the use by government agencies of information technologies (such as Wide Area Networks, the Internet, and mobile computing) that have the ability to transform relations with citizens, businesses, and other arms of government*". Pada intinya E-Government adalah penerapan teknologi informasi pada instansi pemerintah yang memiliki kemampuan untuk melakukan transformasi hubungan antara warganegara, bisnis, dan instansi pemerintah lainnya.

(Tajuddin & Manan, 2015) pada papernya menjelaskan mengenai pembuatan sistem informasi monitoring satuan kerja tujuannya agar dapat memantau pelaksanaan pembangunan di NTB apakah sudah sesuai dengan yang direncanakan sehingga mampu mendukung terwujudnya *Good Governance*. Sistem tersebut dibuat dengan metode *Software Development Life Cycle (SDLC)* yaitu metode pengembangan perangkat lunak yang di mulai dari *planning* sampai dengan *maintenance*. Setelah sistem tersebut dikembangkan, hasil fitur yang selesai dibuat yaitu laporan bulanan, laporan triwulan, laporan tahunan, laporan proyek tahun anggaran, laporan grafik, cetak data dinas, backup data dan restore data.

Sementara itu, (Rahmawati, Tolle, & Rokhmawati, 2019) pada paper Pengembangan Sistem Informasi Monitoring Dan Evaluasi Hasil Kegiatan Pengawas Berbasis Web tentang pengembangan

sistem informasi monitoring dan evaluasi hasil kegiatan pengawas. Kegiatan evaluasi hasil kegiatan pengawas dilakukan untuk mengetahui apakah kegiatan pengawas sudah sesuai dengan apa yang seharusnya. Akan tetapi evaluasi tersebut dilakukan secara manual kemudian dipindah ke *spreadsheet*, kondisi tersebut tidak efisien dan dapat menimbulkan hilangnya data. Sistem Informasi Monitoring dan Evaluasi Hasil Kegiatan Pengawas harapannya dapat membantu dan mempermudah proses monitoring dan evaluasi agar data terorganisir. Setelah selesai dikembangkan hasil keluarannya adalah *sequence diagram*, selain itu dilakukan pengujian fungsional, non-fungsional dan komabilitas. Hasilnya dari pengujian fungsional menyatakan fungsionalitas sistem sudah berjalan sebagaimana mestinya, dari pengujian non-fungsional yaitu usability mendapatkan grade B yang artinya antarmuka sistem baik dan dapat digunakan oleh pengguna dengan mudah, terakhir hasil pengujian komabilitas menunjukkan bahwa *minor issue* yang ada dalam sistem tersebut tidak mempengaruhi fungsionalitas sistem.

## 2.2 Join SQL

Klausa *join* merupakan salah satu klausa yang sering ditemui dalam *query* basis data. Klausa *join* digunakan untuk menggabungkan beberapa tabel yang terkait dalam satu *query*. Penggunaan klausa *join* dapat memudahkan dalam pencarian data yang sama. Penggunaan *join* melibatkan dua kolom dari dua tabel. Selain itu tabel tersebut juga harus memiliki relasi dengan tabel satunya, relasi itu dapat diambil dari kolom *Primary Key* dan kolom *Foreign Key* dari kedua tabel tersebut. Satu hal yang harus diperhatikan ketika membuat relasi untuk kedua tabel adalah kedua kolom pada tabel tersebut memiliki tipe yang sama.

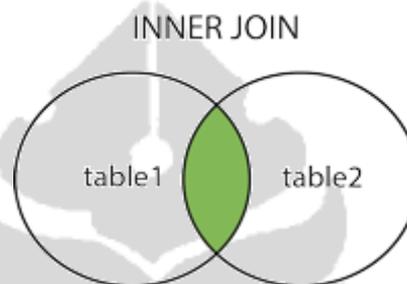
(Lafler, 2011) menerbitkan sebuah paper yang membahas tentang bagaimana cara melakukan *join* table. Dalam paper tersebut dijelaskan bahwa jarang ditemui dalam sebuah sistem dimana semua data berada dalam satu tabel. Hal tersebut karena jika berada dalam satu tabel maka kemungkinan terjadi duplikasi data akan tinggi dan perlu pengelolaan khusus. Maka dari itu, pemisahan data dalam beberapa tabel perlu dilakukan, atas dasar hal tersebut dibutuhkan sebuah metode untuk mengakses dua tabel secara bersamaan. Tujuan dari paper ini memberikan pengetahuan kepada para pembaca tentang *join* dan bagaimana cara melakukannya. Hasil pembahasan pada paper tersebut menjelaskan apa itu *join*, *join* dengan *where*, *join* tiga tabel dan menggunakan *table aliases*.

### 2.2.1 Klasifikasi *Join*

Terdapat berbagai klasifikasi *join* yang digunakan untuk menggabungkan tabel, masing-masing *join* memiliki karakteristik yang berbeda-beda. Untuk itu, pemilihan *join* apa yang akan digunakan dapat berpengaruh pada hasil *query*. Beberapa macam *join* tersebut antara lain

#### 1. *Inner Join*

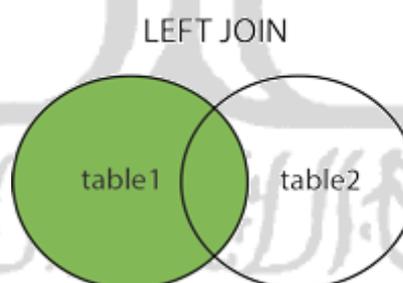
*Inner Join* atau biasa disebut *join* digunakan untuk mencari data yang sama dari dua tabel dengan cara mencocokkan antara *primary key* pada tabel pertama dengan *foreign key* pada tabel kedua. Jika antara *primary key* dan *foreign key* tidak ada kecocokan maka data tidak akan ditampilkan. Irisan tabel yang akan ditampilkan Gambar 2.1



Gambar 2.1 Irisan Inner Join

#### 2. *Left Join*

*Left Join* digunakan untuk menampilkan semua data pada tabel pertama walaupun tidak memiliki relasi dengan tabel kedua. Data yang tidak memiliki relasi akan bernilai *null*. Hasil irisan tabel Gambar 2.2

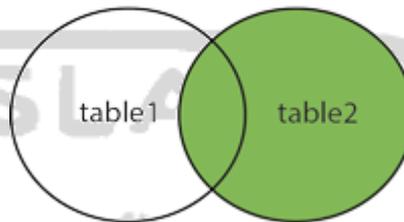


Gambar 2.2 Irisan Left Join

### 3. *Right Join*

*Right Join* pada prinsipnya sama dengan *Left Join* yaitu akan menampilkan data walaupun tidak memiliki relasi. Akan tetapi data yang akan ditampilkan adalah data pada tabel kedua yang tidak memiliki relasi dengan tabel pertama. Hasil dari irisan untuk *right join* Gambar 2.3

RIGHT JOIN



Gambar 2.3 Irisan Right Join

### 4. *Subquery*

Selain dengan menggunakan klausa *join* untuk menggabungkan table, cara lain yang dapat dilakukan adalah dengan menggunakan klausa *subquery*. Hasil yang didapatkan akan sama, namun metode untuk menggabungkan tabel saja yang berbeda.

#### 2.2.2 Cara Melakukan *Join*

*Join* dengan *inner join*, *left join*, *right join* dan *subquery* memiliki cara yang berbeda. Utamanya pada *subquery* dimana tidak hanya klausa saja yang berbeda, namun cara untuk mendapatkan id pada tabel lain juga berbeda.

##### 1. *Join* dengan klausa *join*

Sebuah *query* dengan melibatkan lebih dari satu tabel dapat dilakukan dengan cara menggabungkan tabel pertama dan tabel kedua. Klausa *join* ditambahkan setelah klausa *from* diikuti dengan menambahkan *primary key* dan *foreign key* terkait. Untuk membuatnya bisa dilakukan dengan menggunakan tiga klausa *join* yaitu *inner join*, *left join* dan *right join*.

```

SELECT *
  FROM table1
  INNER JOIN table2 on table1.id = table2.id

SELECT *
  FROM table1
  LEFT JOIN table2 on table1.id = table2.id

SELECT *
  FROM table1
  RIGHT JOIN table2 on table1.id = table2.id

```

## 2. Join dengan Subquery

*Join* dengan menggunakan *subquery* bisa dilakukan dengan menambahkan *subquery* setelah klausa *in*. Dalam kasus ini terdapat dua buah *query*, yang pertama dijalankan adalah *subquery* kemudian akan menjalankan *query* utama. Contohnya seperti pada

```
SELECT * FROM table1 WHERE id NOT IN (SELECT id FROM table2)
```

Query di atas akan menghasilkan hasil yang sama dengan menggunakan kalusa *left join* seperti pada

```

SELECT table1.*
  FROM table1
  LEFT JOIN table2 ON table1.id = table2.id
  WHERE table2.id IS NULL;

```