

Pembuatan Aplikasi Pengembangan Sistem Pengawasan Kepatuhan Berbasis Web Pada Perusahaan Kontraktor Pertambangan

Talitha Almira, Rizal Isnanto, Kurniawan Teguh Martono
Program Studi Sistem Komputer Fakultas Teknik Universitas Diponegoro
Jalan Prof. Sudharto, Tembalang, Semarang, Indonesia
talithamir@gmail.com

Abstrak – Perkembangan teknologi informasi yang sangat pesat memungkinkan suatu data atau informasi dapat disimpan dan diperoleh dengan mudah dan cepat. Dari segi ukuran data, juga dapat dilakukan dalam skala yang besar. Bagi perusahaan bisnis, manfaat teknologi informasi ini umumnya di gunakan untuk mencatat, mengolah dan menyajikan data transaksi berdasarkan proses bisnis yang dijalkannya.

Perancangan aplikasi system pengawasan kepatuhan pada penelitian ini bertujuan untuk merancang system pengawasan kepatuhan sebagai sarana untuk mengawasi tingkat kepatuhan perusahaan kontraktor pertambangan terhadap peraturan yang berlaku. Aplikasi ini berbasis web dengan menggunakan bahasa pemrograman C# dan basisdata SQL Server Local Database. Dalam melakukan penelitian ini, aplikasi dibangun dengan metode pengembangan perangkat lunak Rapid Application Development (RAD). Pengujian yang dilakukan pada aplikasi adalah dengan menggunakan metode whitebox.

Berdasarkan hasil pengujian yang dilakukan pada aplikasi sistem pengawasan kepatuhan berbasis website, aplikasi sudah mampu berjalan dengan cukup baik. Hasil pengujian menunjukkan berbagai fitur yang ada pada aplikasi ini dapat berfungsi sebagaimana mestinya.

Kata Kunci : Teknologi Informasi, white box, rapid application development, pengembangan perangkat lunak

I. PENDAHULUAN

Perkembangan teknologi informasi yang sangat pesat memungkinkan suatu data atau informasi dapat disimpan dan diperoleh dengan mudah dan cepat. Dari segi ukuran data, juga dapat dilakukan dalam skala yang besar. Bagi perusahaan bisnis, manfaat teknologi informasi ini umumnya di gunakan untuk mencatat, mengolah dan menyajikan data transaksi berdasarkan proses bisnis yang dijalkannya. Dengan didukung oleh teknologi nirkabel, bahkan review atas dokumen transaksi suatu perusahaan dapat dilakukan dengan tidak mengenal jarak dan waktu. Transaksi yang dilakukan di suatu area di Indonesia misalnya, pada saat yang sama sudah dapat dilihat di daerah lain, diseluruh dunia.

Pada suatu perusahaan, pencatatan data atau informasi diperlukan sebagai sebuah dokumentasi, yang akan ditindaklanjuti jika diperlukan. Dalam hal ini tentu saja adalah kejadian-kejadian penting yang berisiko merugikan bagi perusahaan, seperti terjadinya kesalahan angkutan, kekurangan/kelebihan stok barang dan lain sebagainya, yang dapat menyebabkan kerugian bagi perusahaan tersebut maupun konsumennya.

Dalam pembuatan tugas akhir ini pembahasan masalah memiliki batasan pada permasalahan berikut :

1. Sistem berupa aplikasi berbasis web yang diakses pengguna dengan web browser Internet Explorer dan Mozilla Firefox.
2. Kode perangkat lunak ditulis menggunakan bahasa pemrograman C# dengan kerangka kerja ASP.NET MVC.
3. Basis data dijalankan dengan SQL Server.
4. Lingkup kerja sistem adalah PT. Pamapersada Nusantara sebagai perusahaan kontraktor pertambangan, khususnya jobsite BHPL.
5. Pengguna yang berinteraksi langsung dengan aplikasi sistem pengawasan kepatuhan dibatasi sebanyak 3 aktor diantaranya Admin, Compliance Officer, Manager. Adapun pengguna yang tidak berinteraksi langsung dengan aplikasi diantaranya HCGS Dept Head Site, BHP Compliance, Related Parties bertugas seperti yang digambarkan pada bisnis proses Gambar 2.

II. LANDASAN TEORI

A. Penelitian Terdahulu

Dalam melakukan penelitian mengenai pembuatan aplikasi pengembangan sistem pengawasan kepatuhan berbasis web pada perusahaan kontraktor pertambangan mengambil beberapa referensi berupa dokumentasi dari sistem *Corporate Tax Online Report*.

PT. Pamapersada Nusantara (PAMA) selaku kontraktor pertambangan telah memiliki beberapa aplikasi *report service* diantaranya adalah *Corporate Tax Online Report*. Aplikasi tersebut digunakan untuk mencatat laporan tax secara online berbasis web.

Perbedaan dengan penelitian yang akan dilakukan oleh penulis adalah pembuatan aplikasi sistem pengawasan kepatuhan dibuat berdasarkan komitmen anti-korupsi PAMA dan mitra kerja. Aplikasi ini memiliki beberapa fitur utama

yaitu mencatat laporan, mencetak laporan, dan melihat ringkasan data laporan. Adapun persentase tingkat kepatuhan yang ditampilkan dalam bentuk grafik. Penelitian ini menggunakan metodologi pengembangan perangkat lunak *Rapid Application Development* (RAD) dengan kerangka kerja ASP.NET MVC.

B. ASP.NET MVC

ASP.NET MVC adalah kerangka kerja pengembangan *Web* dari Microsoft yang menggunakan arsitektur MVC. ASP.NET MVC merupakan ide dan teknik paling terbaru dalam pengembangan *Agile* dan merupakan kerangka kerja terbaik dari *platform* ASP.NET. Kerangka kerja ini merupakan alternatif lengkap sebagai pengganti ASP.NET *Web Forms* [1]. Microsoft menyediakan ASP.NET yang matang, komponen dan fasilitas yang terbukti untuk mengembangkan aplikasi web yang efektif dan efisien. ASP.NET MVC berdasarkan *platform* .NET yang memiliki fleksibilitas untuk menulis kode didalam semua bahasa .NET dan mengakses fitur API. Fitur-fitur *platform* ASP.NET yang siap pakai seperti otentikasi, keanggotaan, peran, profil, dan internasionalisasi dapat mengurangi sejumlah kode yang dibutuhkan untuk mengembangkan dan memelihara aplikasi web, dan fitur ini hanya efektif bila digunakan dalam kerangka MVC. [2]

C. Entity Framework

Entity framework merupakan sekumpulan teknologi ADO.NET yang membantu memetakan antara pengembangan *object-oriented* dan basis data [9]. Terdapat tiga cara untuk bekerja dengan entity framework, diantaranya adalah: *Database First*, *Model First* dan *Code First*.

D. SQL Server

Microsoft SQL Server 2008 R2 adalah generasi berikutnya dari *platform* Microsoft, dengan fitur baru yang memberikan kinerja yang lebih cepat, dan memberikan wawasan bisnis yang kuat. Microsoft SQL Server 2008 R2 telah membuat dampak pada organisasi di seluruh dunia dengan kemampuan inovatif, peningkatan efisiensi serta kolaborasi antara *database administrator* (DBA) dan pengembang aplikasi, dan skala untuk mengakomodasi beban kerja data.

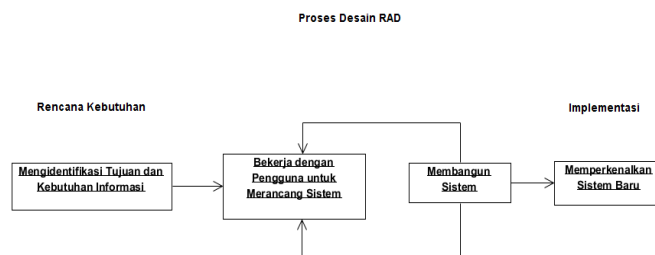
E. Visual Studio

Visual Studio adalah *Integrated Development Environment* (IDE) dari Microsoft yang digunakan untuk membuat, menjalankan, dan memperbaiki program(aplikasi). Visual Studio dapat digunakan untuk membuat aplikasi dengan berbagai bahasa yang didukung .NET seperti C++, C# dan Visual Basic. IDE ini sangat kuat dan memiliki *tool* yang canggih untuk membuat aplikasi bisnis. Visual Studio juga mendukung XML/XSLT,HTML/XHTML, Javascript dan CSS. Visual Studio yang terbaru adalah Visual Studio 2013. [4]

III. PERANCANGAN SISTEM

A. Tahap Pengembangan Sistem

Perancangan sistem dalam penelitian ini akan menggunakan metodologi pengembangan perangkat lunak *Rapid Application Development* (RAD). Bab ini akan membahas langkah-langkah perancangan aplikasi satu per satu sesuai dengan tahapan yang ada pada *Rapid Application Development*. Pada Gambar 3.1 ditunjukkan bahwa RAD memiliki tiga tahapan utama, yaitu Rencana Kebutuhan (*Requirement Planning*), Proses Desain (*Design Workshop*), dan Implementasi (*Implementation*). Sedangkan pada tahap Proses Desain dibagi lagi menjadi dua hal yaitu Desain Sistem (*Design System*) dan Membangun Sistem (*Build System*).



Gambar 1. Tahapan metode *Rapid Application Development*

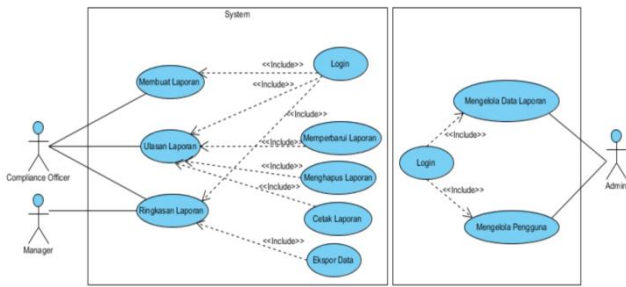
B. Rencana Kebutuhan (Requirement Planning)

Rencana kebutuhan merupakan tahapan awal dan penting karena pada tahap ini dilakukan perencanaan terhadap sistem, tujuan pembuatan sistem, fungsi-fungsi yang diinginkan serta pengelompokkan terhadap fitur-fitur yang harus ada dan fitur-fitur tambahan.

Aplikasi sistem pengawasan kepatuhan ini berfungsi untuk mencatat laporan untuk memastikan semua kegiatan tercatat secara *realtime*. Secara umum aplikasi ini memiliki beberapa komponen penting, yaitu pencatatan laporan, dan penyimpanan laporan ke dalam basisdata.

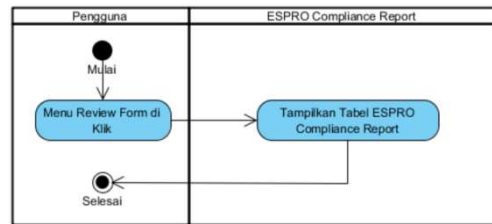
Sasaran pengguna aplikasi ini adalah PT. Pamapersada Nusantara khususnya *compliance officer* dan *manager*. Sasaran lain dari aplikasi ini adalah memudahkan pengguna dalam pencatatan dan laporan kepatuhan secara *online*.

Dari perencanaan kebutuhan tersebut dapat digambarkan secara umum sistem yang akan dibuat dengan menggunakan UML, UML yang digunakan adalah UML versi 1.1 dimana terdapat delapan diagram (*Use Case*, *Activity*, *Sequence*, *Collaboration*, *Class*, *Statechart*, *Component*, *Deployment*) yang memiliki fungsi-fungsi yang berbeda dalam menggambarkan suatu sistem. Dalam menggambarkan kebutuhan sistem digunakan *Use Case Diagram*, untuk kebutuhan dari aplikasi ini dapat dilihat pada Gambar 2.



Gambar 2. Diagram Sistem Use Case

Pada Gambar 4. digambarkan bagaimana proses pengguna melakukan pencatatan laporan.



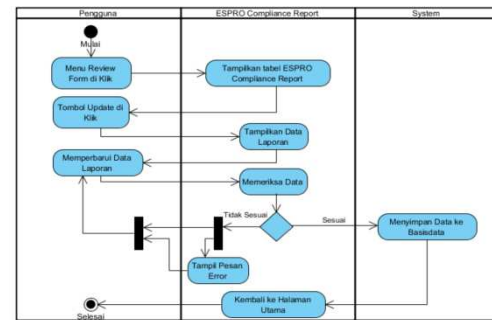
Gambar 5. Diagram Aktivitas Ulasan Laporan

Pada Gambar 5. digambarkan bagaimana proses pengguna melakukan ulasan laporan.

C. Proses Desain

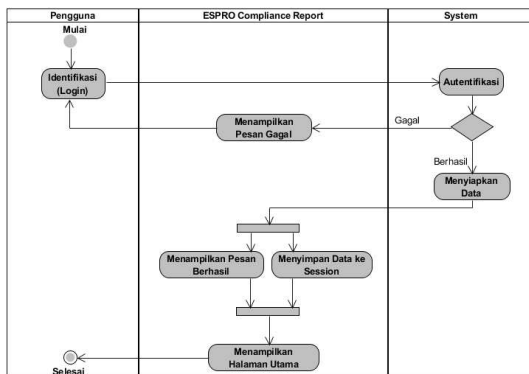
Proses desain merupakan proses yang dilakukan setelah perencanaan kebutuhan dilakukan, ini dikarenakan dalam melakukan desain terhadap suatu sistem pasti seorang analis harus mengetahui spesifikasi atau kebutuhan dari sistem itu sendiri. Dalam melakukan desain sistem terdapat beberapa hal yang harus dibuat antara lain desain proses kerja (*bussiness process*), desain penyebaran sistem (*Deployment Design*), desain basisdata (*Database Design*), serta desain tampilan (*Layout Design*).

Desain proses kerja merupakan desain mendasar mengenai perilaku sistem dan aktivitas yang terjadi ketika aplikasi dijalankan. Aktivitas serta perilaku aplikasi ini digambarkan dalam beberapa Aktivitas Diagram.



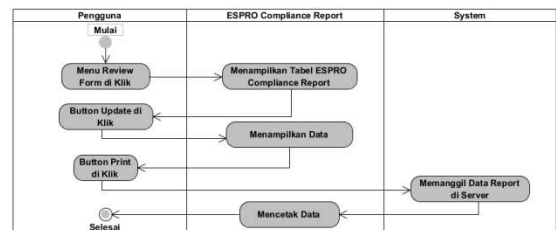
Gambar 6. Diagram Aktivitas Memperbarui Laporan

Pada Gambar 6. digambarkan bagaimana proses pengguna memperbarui laporan.



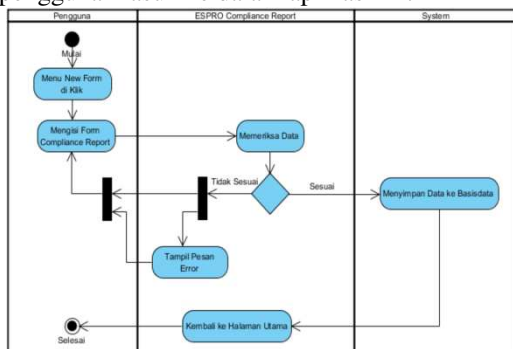
Gambar 3. Diagram Aktivitas Login

Diagram Aktivitas Login yang terdapat pada Gambar 3. menggambarkan tentang bagaimana proses yang terjadi ketika seorang pengguna masuk ke dalam aplikasi ini.

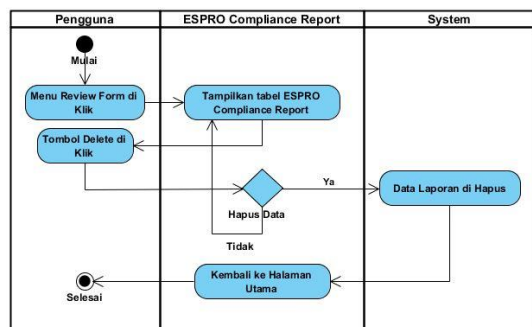


Gambar 7. Diagram Aktivitas Mencetak Laporan

Pada Gambar 7. digambarkan bagaimana proses pengguna mencetak laporan dimulai dari pengguna menekan menu Review Form kemudian aplikasi akan mengarahkan pengguna ke halaman Review Form.

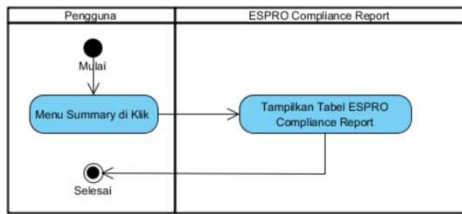


Gambar 4. Diagram Aktivitas Membuat Laporan



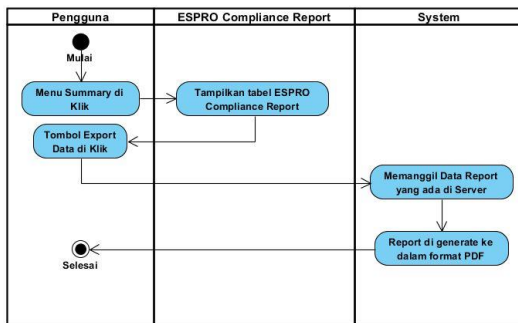
Gambar 8. Diagram Aktivitas Menghapus Laporan

Pada Gambar 8. digambarkan bagaimana proses pengguna menghapus data laporan.



Gambar 9. Diagram Aktivitas Ringkasan Laporan

Diagram Aktivitas Ringkasan Laporan yang terdapat pada Gambar 9. menggambarkan tentang bagaimana proses yang terjadi ketika mengakses ringkasan laporan.

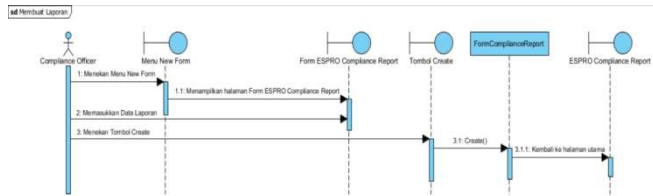


Gambar 10. Diagram Aktivitas Ekspor Data

Pada Gambar 10. digambarkan bagaimana proses pengguna melakukan ekspor data.

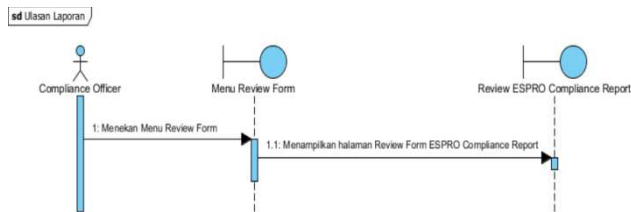
Selanjutnya untuk mendeskripsikan bagaimana entitas dalam sistem berinteraksi akan digambarkan dengan *Sequence Diagram*. Pada *Sequence Diagram* akan menggambarkan tiap satu Diagram Sistem Use Case.

Pada Gambar 11. menggambarkan mengenai proses yang terjadi pada saat Membuat Laporan.



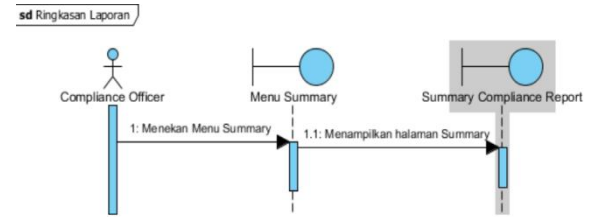
Gambar 11. Diagram Urutan Membuat Laporan

Pada Gambar 12. menggambarkan mengenai proses Ulasan Laporan Laporan.



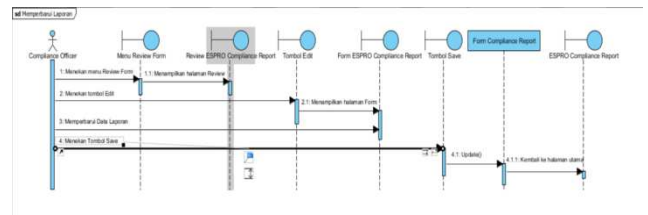
Gambar 12. Diagram Urutan Ulasan Laporan

Pada Gambar 13. menggambarkan Ringkasan Laporan.



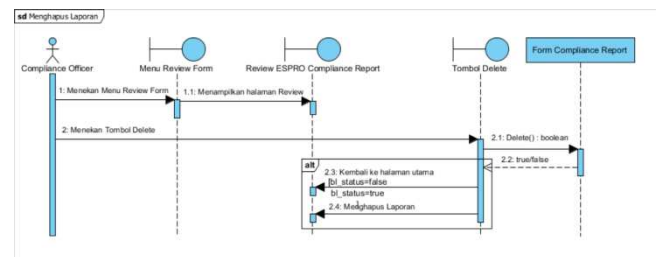
Gambar 13. Diagram Urutan Ringkasan Laporan

Proses berikutnya adalah memperbarui laporan yang digambarkan pada Gambar 14.



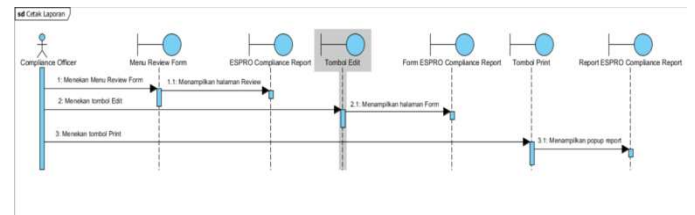
Gambar 14. Diagram Urutan Memperbarui Laporan

Diagram Urutan selanjutnya yaitu menggambarkan proses Menghapus Laporan.



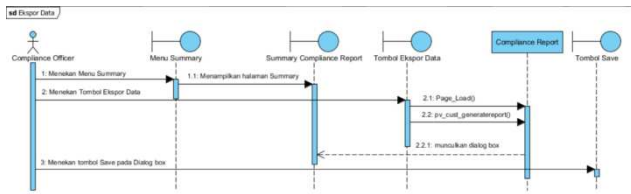
Gambar 15. Diagram Urutan Menghapus Laporan

Proses berikutnya adalah Diagram Urutan untuk Cetak Laporan.



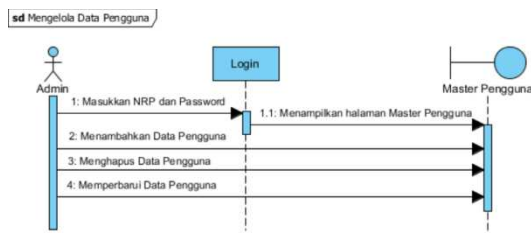
Gambar 16. Diagram Urutan Cetak Laporan

Diagram Urutan untuk proses Ekspor Data digambarkan pada Gambar 17.



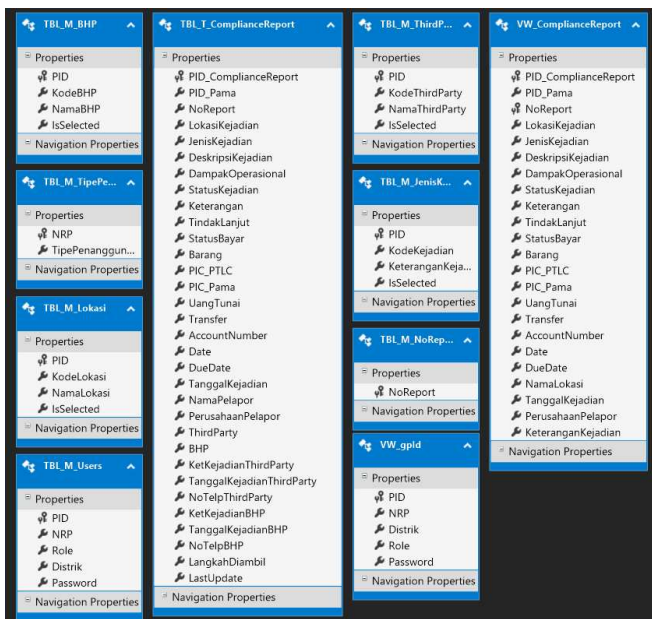
Gambar 17. Diagram Urutan Ekspor Data

Proses selanjutnya pada Diagram Urutan dilakukan oleh aktor *Admin* digambarkan pada Gambar 18.



Gambar 18. Diagram Urutan Mengelola Data Pengguna

Pada perancangan perangkat lunak, desain basisdata merupakan suatu tahap yang sangat penting, dimana dalam tahap ini akan dilakukan pembuatan desain dari skema penyimpanan data dari perangkat lunak.



Gambar 19. Entity Relationship Diagram (ERD) Aplikasi

Dari desain ERD yang telah dibuat Tabel 1. sampai dengan Tabel 5. menunjukkan kamus data yang akan diterapkan pada basisdata SQL Server seperti berikut.

Tabel 1. Kamus Data TBL_M_NoReport

Kolom	Tipe	Atribut
NoReport	Int	primary key, not null, unique

Tabel 2. Kamus Data TBL_M_Users

Kolom	Tipe	Atribut
PID	varchar(50)	primary key, not null, unique
NRP	varchar(50)	Not null
Roles	varchar(50)	Not null
Distrik	varchar(50)	Not null

Tabel 3. Kamus Data TBL_M_Lokasi

Kolom	Tipe	Atribut
PID	varchar(50)	primary key, not null, unique
KodeLokasi	varchar(10)	not null
NamaLokasi	varchar(50)	not null

Tabel 4. Kamus Data TBL_M_JenisKejadian

Kolom	Tipe	Atribut
PID	varchar(50)	primary key, not null, unique
KodeKejadian	varchar(10)	not null
NamaKejadian	varchar(200)	not null

Tabel 5. Kamus Data TBL_M_TipePenanggungJawab

Kolom	Tipe	Atribut
NRP	varchar(50)	primary key, not null, unique
TipePenanggungJawab	varchar(50)	not null

Tabel 5. Kamus Data TBL_T_ComplianceReport

Kolom	Tipe	Atribut
PID_ComplianceReport	varchar(50)	primary key, not null, auto increment
PID_Pama	varchar(10)	not null
NoReport	varchar(15)	not null
LokasiKejadian	varchar(3)	not null
JenisKejadian	varchar(3)	not null
DeskripsiKejadian	varchar(300)	not null
LangkahDiambil	varchar(10)	not null
DampakOperasional	varchar(300)	not null
StatusKejadian	varchar(5)	not null
Keterangan	varchar(300)	not null
TindakLanjut	varchar(300)	not null
StatusBayar	varchar(10)	not null
Barang	varchar(20)	not null
PIC_PTLC	varchar(20)	not null
PIC_Pama	varchar(20)	not null
UangTunai	money	not null
Transfer	numeric(18,0)	not null

DueDate	Datetime	not null
TanggalKejadian	Datetime	not null
NamaPelapor	varchar(20)	not null
PerusahaanPelapor	varchar(10)	not null
ThirdParty	varchar(3)	not null
BHP	varchar(3)	not null
KetKejadianThirdParty	varchar(300)	not null
TanggalKejadianThirdParty	Datetime	not null
NoTelpThirdParty	Int	not null
KetKejadianBHP	varchar(300)	not null
TanggalKejadianBHP	Datetime	not null
NoTelpBHP	Int	not null

IV. IMPLEMENTASI DAN PENGUJIAN

A. Tahap Pembuatan

Pembangunan aplikasi ESPRO Compliance Report dibagi dalam dua tahap yaitu; tahap pembuatan basis data dan tahap pembuatan aplikasi atau *front end*.

1. Tahap Pembuatan Basisdata

Pembuatan basisdata dilakukan menggunakan aplikasi SQL Server Management Studio. SQL Server Management Studio menyediakan pembuatan basisdata secara GUI, hanya dengan mengetikkan beberapa hal dan memilih beberapa pilihan saja basisdata dapat dibuat. SQL Server Management Studio juga menyediakan *Menu SQL* yang dapat digunakan untuk melakukan eksekusi *query* langsung ke basisdata.

```
#TABEL NO REPORT
CREATE TABLE [dbo].[TBL_M_NoReport] (
    [NoReport] [int] NOT NULL
)
```

Pada kode di atas menunjukkan bagaimana tabel No Report dibuat, tabel ini merupakan tabel master yang kemudian diberi nama TBL_M_NoReport.

```
#TABEL LOKASI KEJADIAN
CREATE TABLE [dbo].[TBL_M_Lokasi] (
    [KodeLokasi] [varchar] (3) NOT NULL,
    [NamaLokasi] [varchar] (50) NOT NULL
)
```

Kode *query* di atas digunakan untuk membuat tabel master lokasi kejadian yang diberi nama TBL_M_Lokasi dan bertujuan untuk menampung data lokasi kejadian.

```
#TABEL JENIS KEJADIAN
CREATE TABLE [dbo].[TBL_M_JenisKejadian] (
    [KodeKejadian] [varchar] (3) NOT
    NULL,
    [KeteranganKejadian] [varchar] (200)
    NOT NULL
)
```

Tabel master yang ketiga adalah tabel jenis kejadian yang diberi nama TBL_M_JenisKejadian.

```
#TABEL TIPE PENANGGUNG JAWAB
CREATE TABLE
[dbo].[TBL_M_TipePenanggungJawab] (
    [NRP] [varchar] (50) NOT NULL,
    [TipePenanggungJawab] [varchar] (50)
    NOT NULL
)
```

Tabel master selanjutnya adalah tabel TBL_M_TipePenanggungJawab yang terdiri dari dua kolom.

```
#TABEL BHP
CREATE TABLE [dbo].[TBL_M_BHP] (
    [KodeBHP] [varchar] (3) NOT NULL,
    [NamaBHP] [varchar] (50) NOT NULL
)
```

Tabel BHP dibuat menggunakan kode diatas dengan nama TBL_M_BHP.

```
#TABEL ThirdParty
CREATE TABLE [dbo].[TBL_M_ThirdParty] (
    [KodeThirdParty] [varchar] (3) NOT
    NULL,
    [NamaThirdParty] [varchar] (50) NOT
    NULL
)
```

Query diatas menunjukkan bagaimana tabel *third party* dibuat. Tabel ini bertujuan untuk menampung data-data *third party*.

```
#TABEL COMPLIANCE REPORT
CREATE TABLE
[dbo].[TBL_T_ComplianceReport] (
    [PID_ComplianceReport]
    [varchar] (50) NOT NULL,
    [PID_Pama] [varchar] (10) NULL,
```

```

[NoReport] [varchar](15) NOT NULL,
[LokasiKejadian] [varchar](3) NULL,
[JenisKejadian] [varchar](3) NULL,
[DeskripsiKejadian] [varchar](300)
NULL,
[DampakOperasional] [varchar](300)
NULL,
[StatusKejadian] [varchar](5) NULL,
[Keterangan] [varchar](300) NULL,
[TindakLanjut] [varchar](300) NULL,
[StatusBayar] [varchar](10) NULL,
[Barang] [varchar](20) NULL,
[PIC_PTLC] [varchar](20) NULL,
[PIC_Pama] [varchar](20) NULL,
[UangTunai] [money] NULL,
[Transfer] [numeric](18, 0) NULL,
[AccountNumber] [bigint] NULL,
[Date] [datetime] NULL,
[DueDate] [datetime] NULL,
[TanggalKejadian] [datetime] NULL,
>NamaPelapor] [varchar](20) NULL,
[PerusahaanPelapor] [varchar](10)
NULL,
[ThirdParty] [varchar](3) NULL,
[BHP] [varchar](3) NULL,
[KetKejadianThirdParty]
[varchar](300) NULL,
[TanggalKejadianThirdParty]
[datetime] NULL,
[NoTelpThirdParty] [int] NULL,
[KetKejadianBHP] [varchar](300)
NULL,
[TanggalKejadianBHP] [datetime]
NULL,
[NoTelpBHP] [int] NULL,
[LangkahDiambil] [varchar](10)
NULL,
[LastUpdate] [datetime] NULL
)

```

Pada tabel ini data sistem pengawasan kepatuhan akan disimpan. Kode *query* diatas digunakan untuk membuat tabel TBL_T_ComplianceReport.

```

#VIEW VW_ComplianceReport
SELECT
C.PID_ComplianceReport,
C.PID_Pama, C.NoReport, C.LokasiKejadian,
C.JenisKejadian, C.DeskripsiKejadian,
C.LangkahDiambil, C.DampakOperasional,
C.StatusKejadian, C.Keterangan,
C.TindakLanjut, C.StatusBayar, C.Barang,
C.PIC_PTLC, C.PIC_Pama, C.UangTunai,
C.Transfer, C.AccountNumber, C.Date,
C.DueDate, L>NamaLokasi,
J.KeteranganKejadian, C.LastUpdate,
C.TanggalKejadian, C.PerusahaanPelapor
FROM
dbo.TBL_T_ComplianceReport AS
C.INNER JOIN

```

```

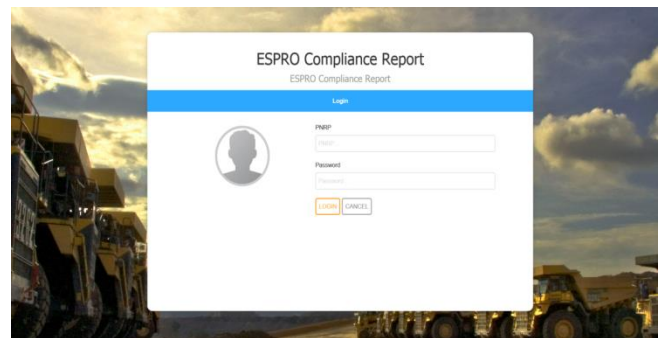
dbo.TBL_M_Lokasi AS L ON C.LokasiKejadian
= L.KodeLokasi INNER JOIN
dbo.TBL_M_JenisKejadian AS J ON
C.JenisKejadian = J.KodeKejadian

```

Penggabungan tabel-tabel baik tabel sebenarnya maupun *virtual* dilakukan dengan penggunaan fungsi *JOIN*, pada MySQL fungsi ini dibagi lagi menjadi tiga fungsi *LEFT JOIN*, *RIGHT JOIN*, serta *INNER JOIN*. *LEFT JOIN* digunakan untuk menggabungkan tabel dengan mengacu pada kunci tabel kiri, *RIGHT JOIN* digunakan untuk menggabungkan tabel dengan mengacu kunci pada tabel kanan, sedangkan *INNER JOIN* digunakan untuk menggabungkan tabel dengan mengacu pada kunci kedua tabel yang sama. *ON* digunakan untuk mewakili karakteristik kunci yang diberikan, jika *ON TRUE* maka kedua tabel akan digabungkan untuk semua nilai yang ada.

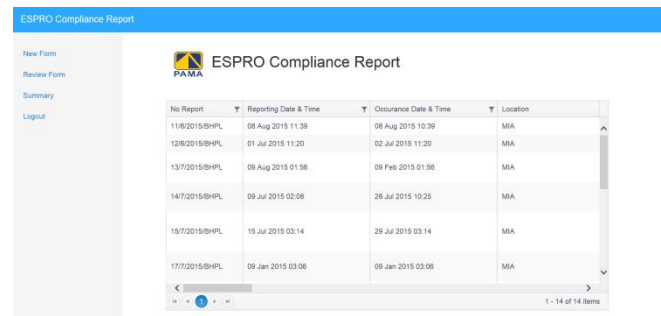
2. Tahap Pembuatan Aplikasi

Pembuatan aplikasi yang dilakukan pertama kali adalah membuat halaman *login*, halaman ini merupakan halaman yang akan diakses pertama kali oleh pengguna ketika membuka aplikasi, pada halaman ini terdapat dua *field* yaitu *field* NRP dan Kata Sandi yang berfungsi untuk menerima masukkan dari pengguna. Pada Gambar 20. merupakan tampilan dari Halaman *Login*.



Gambar 20. Tampilan halaman *Login*

Tahapan berikutnya adalah membuat tampilan halaman utama. Halaman utama terdiri dari *navigation bar* disebelah kiri dan tabel **ESPRO Compliance Report** di sebelah kanan. *Navigation bar* terdiri dari **New Form**, **Review Form**, **Summary**, dan **Logout**.



Gambar 21. Tampilan halaman utama

Selanjutnya membuat halaman **New Form** untuk melakukan pencatatan laporan. Pada halaman ini pengguna akan melakukan pencatatan laporan dan menyimpannya ke basisdata.

Gambar 22. Tampilan halaman *New Form*

Review Compliance Report berisi tabel data seperti pada halaman utama namun tiap baris disertai dengan **button Update**. Ketika pengguna menekan **button** ini maka aplikasi langsung mengarahkan pengguna ke halaman **Update Form** untuk melakukan perubahan data. Data otomatis diambil dari basisdata ketika pengguna menekan **button Update**. Pada halaman ini pengguna juga dapat mencetak laporan yang dapat di *generate* ke dalam format PDF, Word, dan Excel. Pada Gambar 23 menampilkan halaman **Review Form** dan Gambar 24 menampilkan halaman **Update Form**.

Action	Action	No Report	Reporting Date & Time	Occurrence Date & Time
Edit	Delete	11/6/2015/BH-PL	08 Aug 2015 11:38	08 Aug 2015 10:38
Edit	Delete	12/6/2015/BH-PL	01 Jul 2015 11:20	02 Jul 2015 11:20
Edit	Delete	13/7/2015/BH-PL	09 Aug 2015 01:56	09 Feb 2015 01:56
Edit	Delete	14/7/2015/BH-PL	09 Jul 2015 02:08	28 Jul 2015 10:25
Edit	Delete	15/7/2015/BH-PL	15 Jul 2015 03:14	29 Jul 2015 03:14
Edit	Delete	17/7/2015/BH-PL	09 Jan 2015 03:06	09 Jan 2015 03:06

Gambar 23. Tampilan halaman *Review Form*

Gambar 24. Tampilan halaman *Update Form*

KESIMPULAN

Kesimpulan dari penelitian tugas akhir ini adalah sebagai berikut:

1. Aplikasi sistem pengawasan kepatuhan berbasis *website* yang memiliki fitur inti dapat melakukan pencatatan laporan untuk mengawasi tingkat kepatuhan perusahaan kontraktor pertambangan terhadap peraturan yang berlaku.
2. Pengujian yang dilakukan pada aplikasi ini adalah menggunakan metode *whitebox* dan memperoleh hasil *cycloomatic complexity* yang diperoleh sudah sesuai dengan tabel jalur proses yang dibuat, sehingga dapat disimpulkan bahwa logika yang diberikan untuk sistem sudah sesuai dengan kebutuhan sistem.

SARAN

Saran yang diberikan dalam upaya pengembangan sistem yang lebih baik dikemudian hari diantaranya:

1. Aplikasi sistem pengawasan kepatuhan memungkinkan untuk pengembangan aplikasi selanjutnya dalam penambahan fitur-fitur tambahan sesuai dengan kebutuhan pengguna.
2. Perlu dilakukan penelitian lanjutan untuk mengembangkan aplikasi sistem pengawasan kepatuhan menggunakan metode pengembangan perangkat lunak Scrum, yang kemudian dibandingkan dengan metode *Rapid Application Development* (RAD) untuk dapat diperoleh metode yang terbaik.

DAFTAR PUSTAKA

- [1.] Freeman, A. (2013). Pro ASP.NET MVC 5. New York: Apress.
- [2.] M. Zainul, "Thinking Original Entity Framework," 1 July 2011. [Online]. Available: <https://zainulmasadi.wordpress.com/2011/07/01/entity-framework>. [Accessed 12 march 2015].
- [3.] M. Ross and M. Stacia, *Introducing Microsoft SQL Server 2008 R2*, USA: Microsoft Press, 2010
- [4.] Moore, A. (2010). *Visual Studio 2010 All-in-One Form Dummies. Building* (p.916). New Jersey: Wiley Publishing, Inc.
- [5.] R. Falzle, *Jump Start Bootstrap*, (p. 4)
- [6.] Aiskahendra, "Perbedaan Razor dan aspx," 17 December 2010. [Online]. Available: <https://aiskahendra.wordpress.com/2010/12/17/asp-net-mvc-3-mengenal-syntax-razor>. [Accessed 12 March 2015].
- [7.] *Panduan Praktis Menguasai Pemrograman Web dengan Javascript 2009*.
- [8.] P. Jeffrey and dkk, *ASP.NET MVC 4 IN ACTION*, Shelter Island.
- [9.] Noertjahyana, A., 2002. *Studi Analisis Rapid Application Development Sebagai Perangkat Lunak*, 3(2), (p.74).