

IDENTIFIKASI MASALAH DAN PENGUJIAN KUALITAS PERANGKAT LUNAK APLIKASI 'INTEGRATED HUMAN RESOURCES INFORMATION SYSTEM (IHRIS)

Reza Kamaluddin Isman¹, Anita Hidayati²

^{1,2}Teknik Informatika dan Komputer, Politeknik Negeri Jakarta

¹email:rezakisman@gmail.com

²email:anita.hidayati@tik.pnj.ac.id

ABSTRAK

Aplikasi Sumber Daya Manusia terintegrasi merupakan perangkat lunak yang menangani masalah pada divisi SDM dan Umum seperti absensi, penggajian, kas kecil, pergudangan, data karyawan, kontrak, dan lainnya yang menggunakan sumber data yang terintegrasi. Pengintegrasian ini bertujuan untuk meminimalisir kesalahan pada create, read, update, delete (CRUD) pada saat pengoperasian aplikasi, serta integrasi data ini bertujuan menghindari perbedaan informasi yang dihasilkan pada hasil akhir laporan software. Pada jurnal ini penulis melakukan sebuah analisa pada aplikasi Integrated Human Resources Information System (IHRIS), IHRIS merupakan aplikasi perangkat lunak yang akan mengolah data-data dari divisi SDM menjadi informasi laporan yang menggunakan database terintegrasi. Teknologi yang digunakan dalam IHRIS adalah .NET Framework 4.5 dan Microsoft OLEDB (12.0 ACE dan 4.0 JET), serta database yang digunakan adalah Microsoft SQL Server 2012, serta menggunakan bahasa pemrograman C# dengan object-oriented programming. Tujuan dari artikel ini adalah untuk mengidentifikasi berapa besar kualitas perangkat lunak IHRIS dengan dilakukan tes. Tes yang dilakukan adalah unit testing tool dan manual testing. Kemudian dilakukan analisa masalah-masalah apa saja yang dapat diatasi oleh perangkat lunak tersebut. Hasil identifikasi menunjukkan bahwa masalah yang dapat diatasi oleh perangkat lunak ini adalah CRUD, integrasi, serta pengimporan data absensi dari mesin absensi yang kemudian diproses sebelum disimpan kedalam database IHRIS, masalah ini dapat diselesaikan dengan menggunakan teknologi Microsoft OLEDB (12.0 ACE dan 4.0 JET). Hasil kualitas software berhasil diuji dengan test metrics yang menunjukkan hasil 100% lulus uji.

Kata kunci : Integrated Human Resources Information System (IHRIS), integrasi, kualitas perangkat lunak, testing, object-oriented programming.

I. PENDAHULUAN

Sebuah perusahaan yang kompeten pasti memiliki manajemen sumber daya manusia yang baik. Pengembangan sumber daya manusia merupakan serangkaian kegiatan yang terorganisir, yang dilakukan dalam waktu khusus dan dirancang untuk menghasilkan perubahan perilaku [1]. Divisi Sumber Daya Manusia (SDM) merupakan salah satu divisi yang sangat krusial di sebuah perusahaan. Divisi SDM menentukan perekrutan karyawan, pelatihan dan pengembangan karyawan, mengawasi kinerja dari setiap karyawan, memberikan rekomendasi pengangkatan jabatan dan kompensasi, distribusi penggajian karyawan, melakukan survei karyawan dan grading, mengawasi perlengkapan, melakukan pembentukan tim untuk proyek-proyek perusahaan, dan lainnya.

Divisi SDM berperan penting dalam sebuah perusahaan, karena tugasnya menghasilkan informasi yang dibutuhkan oleh divisi lain

dalam sebuah perusahaan. Oleh sebab itu data yang akan diolah menjadi informasi wajib valid dan harus selalu benar. Integrasi database merupakan salah satu cara untuk meminimalisir kesalahan tentang data, atau duplikasi data bahkan *mismatch* data. Integrasi data membantu pemusatan informasi sehingga seluruh bagian SDM yang terkait mempunyai data yang sama. Oleh karena itu, dibutuhkan aplikasi terpusat yang dapat melingkupi seluruh bidang atau bagian SDM agar data dapat terintegrasi dengan baik, salah satunya adalah *IHRIS*. *IHRIS* merupakan aplikasi SDM terintegrasi yang dirancang sedemikian mungkin untuk mengatasi masalah di divisi SDM.

Dengan merujuk kepada Artikel yang berjudul '*An Analytical Way to Improve Test Execution and Review of Software Metrics for The Software Quality*', tujuan dari analisis masalah ini adalah untuk mengidentifikasi seberapa besar kualitas perangkat lunak *IHRIS* dengan melakukan *unit testing tools* dan *manual testing* yang disediakan oleh *Visual Studio 2015 Community Edition*.

II. TINJAUAN PUSTAKA

Integrated Human Resource Information System (IHRIS) merupakan aplikasi perangkat lunak yang digunakan pada divisi Sumber Daya Manusia dan Umum (SDM dan Umum) PT Pelayaran Bahtera Adhiguna (PT.BAg). Aplikasi perangkat lunak ini mengatasi masalah-masalah pada divisi SDM dan Umum seperti, *create, read, update, delete* (CRUD) data karyawan, data absensi, data *payroll*, data kontrak, data pergudangan (*inventory*), dan data kas kecil (*petty cash*). Aplikasi ini di-develop oleh *JADES Code Guru* menggunakan *Visual Studio 2015 Community Edition* dan *SQL Server 2012 Express Edition* dengan bahasa pemrograman *C#* yang bersifat *object-oriented programming*.

C# (C-sharp) merupakan sebuah bahasa pemrograman yang berorientasi objek yang dikembangkan oleh Microsoft sebagai bagian dari *.NET Framework*. Bahasa ini dibuat berdasarkan bahasa *C++* yang telah dipengaruhi oleh aspek-aspek ataupun fitur bahasa yang terdapat pada *Java, Delphi, Visual Basic* dan lain-lain dengan beberapa penyederhanaan.

Visual Studio 2015 Community Edition merupakan *framework* pemrograman berbagai macam bahasa seperti *C++, Python, HTML 5, Java Script, C#, Visual Basic, F#, ASP.NET*, dan *Node.js* yang di develop oleh Microsoft. *Framework* ini merupakan *framework* pertama Microsoft yang bersifat terbuka (*open source*).

SQL Server 2012 Express Edition adalah sebuah *Relational Database Management System (RDBMS)* produk Microsoft. Bahasa *query* utamanya adalah *Transact-SQL* standar ANSI / ISO yang digunakan oleh Microsoft dan Sybase. *SQL Server 2012 Express Edition* ini gratis tetapi ada beberapa fitur yang dikunci oleh Microsoft.

.NET Framework (yang terbaru *.NET Framework 4.6*) atau dikenal *dot net* merupakan sebuah perangkat lunak kerangka kerja yang berjalan utamanya pada *Windows Operating Sytem*, saat ini *dot net* sudah mulai terintegrasi dalam distribusi standar Windows. Kerangka kerja (*framework*) ini menyediakan *library* pemrograman komputer yang besar dan mendukung beberapa bahasa pemrograman serta *interoperability* yang baik sehingga bahasa-bahasa tersebut beroperasi satu sama lain dalam pengembangan sistem. *Dot net* berjalan pada lingkungan perangkat

lunak melalui *Common Language Runtime* dan bukan perangkat keras secara langsung.

Object Linking and Embedding, Database (OLE DB) merupakan sebuah API yang dirancang oleh Microsoft, memungkinkan mengakses data dari berbagai sumber dengan cara yang seragam. API menyediakan satu set antarmuka yang diimplementasikan menggunakan *Component Object Model (COM)*. *OLE DB* awalnya ditujukan sebagai pengganti tingkat yang lebih tinggi dan penerus *ODBC*, memperluas fitur yang ditetapkan untuk mendukung jenis database non-relasional, seperti database objek dan spreadsheet yang belum tentu menerapkan *SQL*.

Object-oriented programming merupakan paradigma pemrograman yang berorientasikan kepada objek. Semua data dan fungsi di dalam paradigma ini dibungkus dalam kelas-kelas atau objek-objek [3]. *Test metrics* merupakan standar pengukuran yang penting pada fase *Software Testing Life Cycle (STLC)* [4].

Tes perangkat lunak adalah investigasi yang dilakukan untuk memberikan *stakeholder* informasi tentang kualitas produk perangkat lunak atau *service* yang diuji [2]. Tes perangkat lunak ini menggunakan *testing tools*. *Testing tools* adalah alat yang digunakan untuk menguji coba kualitas perangkat lunak [6].

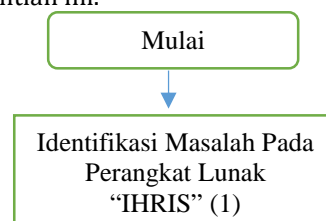
Ada beberapa contoh *testing tools* seperti, *unit testing tools* dan *manual testing tools*.

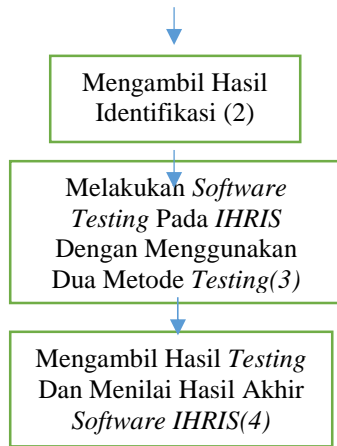
Unit testing mengacu pada tes yang memverifikasi fungsi bagian tertentu dari kode, biasanya pada tingkat fungsi. Pada program yang bersifat *object-oriented*, tes ini dilakukan pada tingkatan kelas, dan mencakup *constructor* dan *destructor* [3].

Manual testing merupakan uji coba yang dilakukan berdasarkan kriteria masalah yang dikumpulkan dari beberapa standar yang dilakukan secara berurutan serta dilakukan tanpa menggunakan alat atau *software*, biasanya uji coba ini dilakukan dengan membandingkan antara ekspektasi *tester* dengan fungsi perangkat lunak yang menghasilkan sebuah *output* [6].

III. METODOLOGI PENELITIAN

Gambar 1 merupakan gambar metodologi penelitian ini.





Gambar 1. Tahapan Metodologi Penelitian

- (1) Identifikasi selesai dilakukan dengan menentukan masalah apa saja yang dapat diatasi oleh perangkat lunak IHRIS, dengan cara menjalankan dan melihat baris kode dari perangkat lunak.
- (2) Mengumpulkan masalah-masalah yang dapat diatasi oleh perangkat lunak IHRIS.
- (3) Menentukan *test case* yang akan dilakukan kepada perangkat lunak IHRIS. Melakukan tes terhadap perangkat lunak IHRIS sesuai dengan *test case* dan *testing tools* yang telah ditentukan.
- (4) Mengumpulkan hasil tes perangkat lunak IHRIS dan menghitung prosentase lulus uji dari perangkat lunak IHRIS.

IV. IDENTIFIKASI MASALAH DAN PENGUJIAN KUALITAS PERANGKAT LUNAK

Masalah yang dapat ditangani oleh perangkat lunak IHRIS didapatkan dengan melihat fungsi kode dari IHRIS, setelah mendapatkan semua fungsi kode dalam perangkat lunak IHRIS, kemudian fungsi ini dikelompokkan sesuai modul sehingga menjadi 6 kelompok modul, kelompok modul tersebut adalah *CRUD* data karyawan, *CRUD* data absensi, *CRUD* data penggajian, *CRUD* data kontrak, *CRUD* data pergudangan, *CRUD* data kas kecil. Tabel 1 merupakan hasil penelitian

yang mengidentifikasi dari fungsi kode dan masalah dikelompokkan sesuai modul.

Tabel 1. Hasil Identifikasi Fungsi Kode yang Kemudian Dikelompokkan Sesuai Modul

No.	Fungsi Kode	Kelompok
1.	(AddContract)	CRUD data kontrak
2.	(AddDivision)	CRUD data penggajian
3.	(AddEmployee)	CRUD data pegawai
4.	(AddInventory)	CRUD data pergudangan
5.	(AddPayroll)	CRUD data penggajian
6.	(AddPettyCash)	CRUD data kas kecil
7.	(AddPosition)	CRUD data penggajian
8.	(Authentication)	CRUD data karyawan
9.	(BindAddPayrollDetails)	CRUD data penggajian
10.	(BindAllContractDetails)	CRUD data kontrak
11.	(BindAllItemDetails)	CRUD data pergudangan
12.	(BindCBDivision)	CRUD data penggajian
13.	(BindCBDivisionForEdit AndSearchEmployeeForm)	CRUD data karyawan
14.	(BindCBPosition)	CRUD data penggajian
15.	(BindCBPositionForEdit AndSearchEmployeeForm)	CRUD data karyawan
16.	(BindDGVPayrollData)	CRUD data penggajian
17.	(BindEmpDetails)	CRUD data karyawan
18.	(BindOngoingContract)	CRUD data kontrak
19.	(BindPayrollData)	CRUD data penggajian
20.	(BindPayrollDataPerEmployee)	CRUD data penggajian
21.	(BindPayrollDetails)	CRUD data penggajian
22.	(BindPettyCashDetails)	CRUD data kas kecil
23.	(CheckContractExpiration)	CRUD data kontrak
24.	(ContractDGV)	CRUD data kontrak
25.	(DivisionDGV)	CRUD data penggajian
26.	(EditContractDetails)	CRUD data kontrak

27.	(EditEmployee)	CRUD data karyawan
28.	(EditPayrollData)	CRUD data penggajian
29.	(EmployeeDGV)	CRUD data karyawan
30.	(GetAttendanceTime)	CRUD data absensi
31.	(GetCompletedContract DGV)	CRUD data kontrak
32.	(GetEditContractDGV)	CRUD data kontrak
33.	(GetElement)	CRUD data absensi
34.	(GetExtendedContractD GV)	CRUD data kontrak
35.	(GetOngoingContractDG V)	CRUD data kontrak
36.	(GetSearchContractDGV)	CRUD data kontrak
37.	(InventoryAutoGenerate ID)	CRUD data pergudangan
38.	(InventoryDGV)	CRUD data pergudangan
39.	(PayrollEmpDGV)	CRUD data penggajian
40.	(PettyCashAutoGenerate ID)	CRUD data kas kecil
41.	(PettyCashDGVlend)	CRUD data kas kecil
42.	(PettyCashDGVReturn)	CRUD data kas kecil
43.	(PositionDGV)	CRUD data penggajian
44.	(GetSearchPayrollData)	CRUD data penggajian
45.	(GetSearchPhyFormula)	CRUD data penggajian
46.	(UpdateContractStatus)	CRUD data kontrak
47.	(UpdatePassword)	CRUD data karyawan
48.	(UpdateStatusAttendanc e)	CRUD data absensi
49.	(UpdateStock)	CRUD data pergudangan
50.	(UserUpdatePhoto)	CRUD data karyawan

Data absensi diatasi dengan meng-import data dari file berformat .xls terlebih dahulu, dengan menggunakan OLE DB JET dan ACE ADAPTER. Gambar 2 merupakan baris kode untuk mengimpor file.xls

```

try
{
    string PathConn =
    @"Provider=Microsoft.ACE.OLEDB.12.0;Dat
a Source=" + tbPath.Text +
    "; Extended Properties=\\"Excel 12.0
XML;HDR=YES;IMEX=1\\";";
    OleDbConnection con = new
    OleDbConnection(PathConn);
    OleDbDataAdapter oledbda = new
    OleDbDataAdapter("SELECT * FROM [" +
    tbName.Text + "$]", con);
    DataTable dt = new DataTable();
    oledbda.Fill(dt);
    dgExcel.DataSource = dt;
}
catch (Exception E)
{
    MessageBox.Show(E.ToString());
}
}
    
```

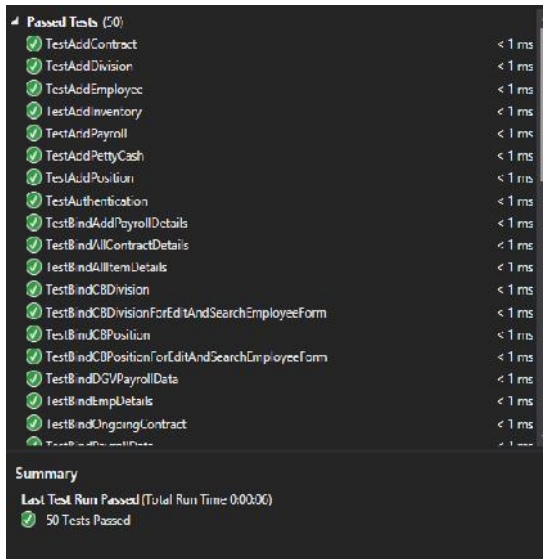
Gambar 2. Kode untuk meng-import file .xls kedalam datagridview yang bernama 'dgExcel'.

Dari Gambar 2, diketahui penggunaan OLEDB ADAPTER versi 12.0 yaitu versi ACE. Sebelum melakukan penulisan kode tersebut harus dipastikan Microsoft Access Database Engine 2010 sudah terpasang didalam komputer.

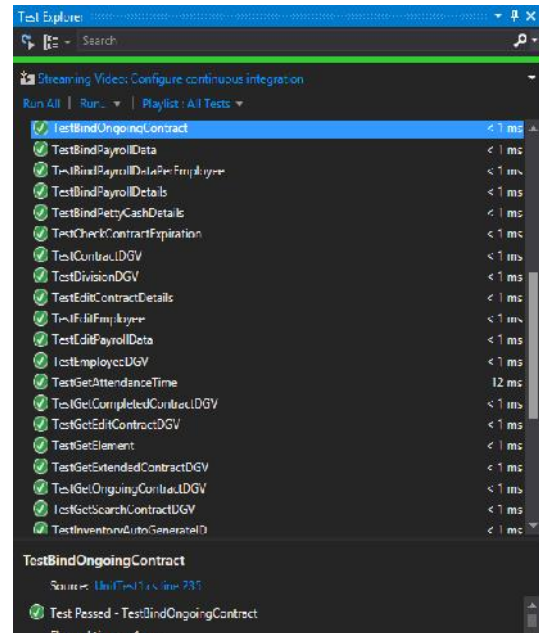
Untuk penentuan versi OLE DB ADAPTER dapat diidentifikasi dengan melihat aplikasi perangkat lunak Microsoft Office apa terpasang didalam komputer, apabila aplikasi perangkat lunak Microsoft Office 2007 yang terpasang maka digunakan OLE DB ADAPTER 4.0 JET sedangkan untuk perangkat lunak aplikasi Microsoft Office 2010 or higher yang terpasang gunakan OLE DB ADAPTER 12.0 ACE [5].

Untuk cara pemasangan OLE DB ADAPTER, pertama dilakukan pemasangan driver software (driver software OLE DB ADAPTER 4.0 JET bisa didapatkan dengan mengunduh di laman Microsoft Download Centre : <https://www.microsoft.com/en-sg/download/details.aspx?id=23734>; Sedangkan untuk versi 12.0 ACE, dapat diunduh di laman Microsoft Download Centre: <https://www.microsoft.com/en-sg/download/details.aspx?id=13255>), kemudian dilakukan instalasi, setelah instalasi selesai, restart komputer [5].

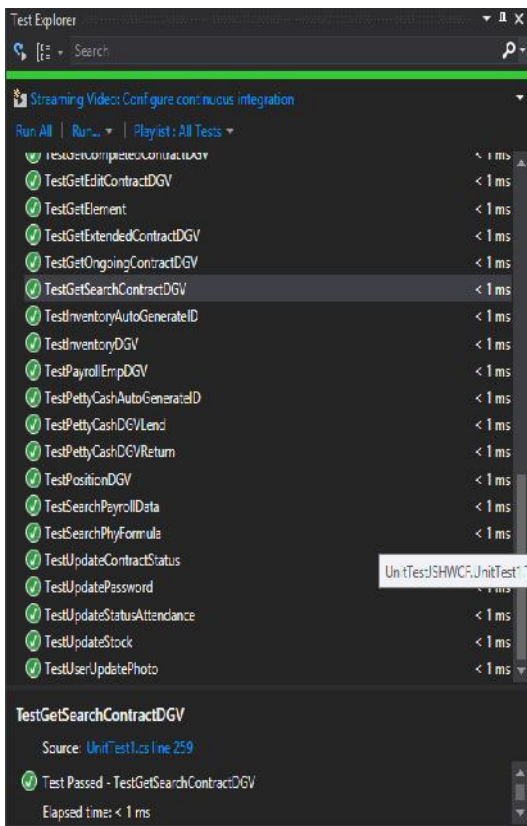
Setelah penulis melakukan identifikasi terhadap masalah dan penyelesaian masalah yang terdapat pada perangkat lunak IHRIS, penulis juga melakukan dua testing kepada IHRIS untuk mengetahui berapa besar kualitas perangkat lunak IHRIS. Gambar 3, 4 dan 5 merupakan hasil pengujian perangkat lunak dengan menggunakan testing tools.



Gambar 3. Hasil pengujian kode perangkat lunak IHRIS dengan menggunakan unit testing tools Visual Studio 2015 Community Edition.



Gambar 5. (lanj.) Hasil pengujian kode perangkat lunak IHRIS dengan menggunakan unit testing tools Visual Studio 2015 Community Edition



Gambar 4. (lanj.) Hasil pengujian kode perangkat lunak IHRIS dengan menggunakan unit testing tools Visual Studio 2015 Community Edition

Tabel 2. Hasil Pengujian Kode Perangkat Lunak Ihris dengan Menggunakan Unit Testing Tools Visual Studio 2015 Community Edition

Unit Testing Tools			
No	Kondisi	Hasil	Lulus Uji (Yes/No)
1.	User memasukan data kontrak dengan parameter yang benar dan tidak ada kesalahan	(TestAddContract) The result will be true and data will be saved	Yes
2.	User memasukan data divisi dengan parameter yang benar dan tidak ada kesalahan	(TestAddDivision) The result will be true and data will be saved	Yes
3.	User memasukan data karyawan dengan parameter yang benar dan tidak ada kesalahan	(TestAddEmployee) The result will be true and data will be saved	Yes
4.	User memasukan data pergudangan	(TestAddInventory) The result will be true and data	Yes

	dengan parameter yang benar dan tidak ada kesalahan	will be saved	
5.	User memasukan data penggajian dengan parameter yang benar dan tidak ada kesalahan	(TestAddPayroll) The result will be true and data will be saved	Yes
6.	User memasukan data kas kecil dengan parameter yang benar dan tidak ada kesalahan	(TestAddPettyCash) The result will be true and data will be saved	Yes
7.	User memasukan data posisi dengan parameter yang benar dan tidak ada kesalahan	(TestAddPosition) The result will be true and data will be saved	Yes
8.	User login dengan username dan password yang benar	(TestAuthentication) The result will be true	Yes
9.	User membuka halaman penggajian	(TestBindAddPayrollDetails) The data will be appear in data grid	Pass
10.	User membuka halaman kontrak	(TestBindAllContractDetails) The data will be appear in data grid	Pass
11.	User membuka halaman pergudangan	(TestBindAllItemDetails) The data will be appear in data grid	Pass
12.	User membuka halaman posisi dalam penggajian	(TestBindCBDivision) The data will be appear in combo box	Pass
13.	User membuka halaman divisi dalam karyawan	(TestBindCBDivisionForEditAndSearchEmployeeForm) The data will be appear in	Pass

		combo box	
14.	User membuka halaman divisi dalam penggajian	(TestBindCBPosition) The data will be appear in combo box	Pass
15.	User membuka halaman posisi dalam karyawan	(TestBindCBPositionForEditAndSearchEmployeeForm) The data will be appear in combo box	Pass
16.	User membuka halaman penggajian	(TestBindDGVPayrollData) The data will be appear in data grid	Pass
17.	User membuka halaman karyawan	(TestBindEmpDetails) The data will be shown on the text box	Pass
18.	User membuka halaman kontrak	(TestBindOngoingContract) The data will be appear in data grid	Pass
19.	User membuka halaman penggajian	(TestBindPayrollData) The data will be appear in data grid	Pass
20.	User membuka halaman penggajian	(TestBindPayrollDataPerEmployee) The data will be appear in text box	Pass
21.	User membuka halaman penggajian	(TestBindPayrollDetails) The data will be appear in datagrid	Pass
22.	User membuka halaman kas kecil	(TestBindPettyCashDetails) The data will be appear in datagrid	Pass
23.	User membuka halaman utama	(TestCheckContractExpiration) The result will be true and notification will alert if there is	Pass

		expired contract	
24.	User membuka halaman kontrak	(TestContractDGV) The data will be appear in datagrid	Pass
25.	User membuka halaman divisi	(TestDivisionDGV) The data will be appear in datagrid	Pass
26.	User menyunting halaman kontrak dengan data yang benar tanpa ada kesalahan	(TestEditContractDetails) The result will be true	Pass
27.	User menyunting halaman karyawan dengan data yang benar tanpa ada kesalahan	(TestEditEmployee) The result will be true	Pass
28.	User menyunting halaman penggajian dengan data yang benar tanpa ada kesalahan	(TestEditPayrollData) The result will be true	Pass
29.	User membuka halaman karyawan	(TestEmployeeDGV) The data will be appear in datagrid	Pass
30.	User mengimpor data absensi dari file .xls dengan menggunakan OLEDB Data ADAPTER	(TestGetAttendanceTime) The result will be returned is list of time	Pass
31.	User membuka halaman kontrak	(TestGetCompletedContractDGV) The data will be appear in datagrid	Pass
32.	User membuka halaman kontrak	(TestGetEditContractDGV) The data will be appear in datagrid	Pass
33.	User	(TestGetElement)	Pass

	membuka main menu	nt) The result will be true	
34.	User membuka halaman kontrak	(TestGetExtendedContractDGV) The data will be appear in datagrid	Pass
35.	User membuka halaman kontrak	(TestGetOngoingContractDGV) The data will be appear in datagrid	Pass
36.	User mencari dan mengklik tombol SEARCH pada halaman kontrak	(TestGetSearchContractDGV) The data will be appear in text box	Pass
37.	User membuka halaman pergudangan	(TestInventoryAutoGenerateID) The result will be returned is list of number	Pass
38.	User membuka halaman pergudangan	(TestInventoryDGV) The data will be appear in datagrid	Pass
39.	User membuka halaman karyawan	(TestPayrollEmployeeDGV) The data will be appear in datagrid	Pass
40.	User membuka halaman kas kecil	(TestPettyCashAutoGenerateID) The result will be returned is list of number	Pass
41.	User membuka halaman kas kecil	(TestPettyCashDGV) The data will be appear in datagrid	Pass
42.	User membuka halaman kas kecil	(TestPettyCashDGVReturn) The data will be appear in datagrid	Pass
43.	User membuka halaman posisi	(TestPositionDGV) The data will be appear in datagrid	Pass
44.	User mencari dan mengklik tombol	(TestGetSearchPayrollData) The data will	Pass

	SEARCH pada halaman penggajian	be appear in text box	
45.	User mencari dan mengklik tombol SEARCH pada halaman penggajian	(TestGetSearchPhyFormula) The data will be appear in text box	Pass
46.	User mengklik tombol UPDATE pada halaman kontrak	(TestUpdateContractStatus) The result will be true and data will be updated	Pass
47.	User mengklik tombol UPDATE pada halaman karyawan	(TestUpdatePassword) The result will be true and data will be updated	Pass
48.	User mengimpor data absensi dari file .xls dengan menggunakan OLEDB Data ADAPTER	(TestUpdateStatusAttendance) The result will be true	Pass
49.	User mengklik tombol UPDATE pada halaman pergudangan	(TestUpdateStock) The result will be true and data will be updated	Pass
50.	User mengklik tombol UPDATE pada halaman karyawan	(TestUserUpdatePhoto) The result will be true and data will be updated	Pass

Hasil tes yang telah dilakukan dan didokumentasikan adalah sebagai berikut:

- Dari tabel 2 diketahui jumlah tes sebanyak 50 rangkaian dilakukan dengan menggunakan *unit testing tools* dan sebanyak 50 tes berhasil dijalankan.
- Sebanyak 50 rangkaian tes dilakukan dengan menggunakan *manual testing tools* dan mencapai keberhasilan 100%.
- Apabila hasil tes dihitung dengan *test metrics*, jumlah tes dianalogikan dengan simbol “nt” dan jumlah komponen yang di tes dianalogikan dengan simbol “N”, dan hasil tes dianalogikan dengan “q” [4].

Maka kualitas tes dihitung dengan menggunakan rumus:

$$q = \frac{nt}{N} \times 100\%$$

$$q = \frac{50}{50} \times 100\%$$

$$q = 100\%$$

V. KESIMPULAN

Hasil identifikasi masalah yang sudah dilakukan penulis mengidentifikasi sebanyak 6 masalah yang dapat diatasi oleh perangkat lunak *IHRIS*, yaitu *CRUD* data karyawan, data absensi, data penggajian, data kontrak, data pergudangan, data kas kecil. Data absensi berasal dari *file .xls* sehingga perlu dikonversi terlebih dahulu sebelum melakukan *CRUD*, masalah konversi ini dapat diatasi dengan meng-*import* data dari *file* berformat *.xls* terlebih dahulu, dengan menggunakan *OLE DB JET* dan *ACE ADAPTER*.

Setelah mengidentifikasi masalah yang dapat diatasi aplikasi perangkat lunak *IHRIS*, kemudian dilakukan tes untuk mengetahui berapa besar kualitas perangkat lunak *IHRIS*.

Hasil tes menghasilkan q = 100 %, ini menunjukkan bahwa perangkat lunak *IHRIS* berhasil uji secara keseluruhan bila diuji dengan menggunakan *unit testing tools* dan *manual testing tools*.

VI. DAFTAR PUSTAKA

- [1] James L. Noel (1988). 'Managing human resource development by Leonard Nadler and Garland Wiggs', Human Resource Management Vol. 27, Issue 1, 1988. pp.103-107
- [2] Cem Kaner (2006). 'Exploratory Testing'. Florida Institute of Technology, Quality Assurance Institute Worldwide Annual Software Testing Conference, Orlando, FL.
- [3] Binder, Robert V. (1999). Testing Object-Oriented Systems: Objects, Patterns, and Tools. Addison-Wesley Professional. p. 45. ISBN 0-201-80938-9.
- [4] Chandu P.M.S.S (2015). 'An Analytical Way to Improve Test Execution and Review of Software Metrics for The Software Quality'. Journal of Theoretical and Applied Information Technology. March . Vol 73.
- [5] Microsoft Download Centre (2015). <https://www.microsoft.com/en-sg/download/details.aspx?id=23734>. Dikunjungi tanggal 13 Desember 2015.
- [6] NIIT Team (2010) 'Software Testing and Quality Assurance'. Book number: 133170. India.