Pembangunan *Class Library* Pada Domain *Recipes Apps* Berbasis *Windows Phone Apps*

Renata Kurniawan[1]

[1] Teknik Informatika – Universitas Komputer Indonesia
Jl. Dipatiukur 112-114 Bandung
Email : renatakurniawan@gmail.com[1]

ABSTRAK

Resep masakan telah menjadi kebutuhan sejak lama dan terus berkembang dalam berbagai media sampai media perangkat lunak. Menurut *mobile marketing statistic* 2015 menunjukan bahwa pengguna *mobile* mencapai 1900 juta pengguna. Berdasarkan observasi, aplikasi resep masakan pada perangkat *mobile* banyak digemari oleh pengembang aplikasi dan memiliki jumlah pengguna yang tinggi. Berdasarkan wawancara kepada beberapa pengembang aplikasi didapat fakta bahwa pengembang aplikasi lebih fokus pada pembangunan aplikasi dibandingkan analisis terhadap domain aplikasi tersebut, akibatnya pembangunan aplikasi kembali dimulai dari awal karena sulitnya penggunaan ulang. Solusi yang dapat diterapkan yaitu membangun sebuah *class library* pada domain aplikasi resep masakan.

Class library yaitu perpustakaan kelas yang dapat diguna ulang dalam pembangunan aplikasi pada suatu domain. *Class library* menyediakan fungsionalitas dasar pada suatu domain aplikasi. Penentuan fungsionalitas dasar pada *class library* didapatkan dengan cara analisa terhadap aplikasi resep masakan yang telah ada. Tahapan analisis yaitu analisis domain, analisis *frozen spot*, analisis *hotspot*, perancangan kelas, perancangan *package*, dan implementasi kepada kode program.

Setelah dilakukan pengujian menggunakan *unit testing*, *integration testing*, dan pengujian *acceptance* dapat disimpulkan bahwa *class library* yang dibangun telah menyediakan fungsionalitas dasar pada domain aplikasi resep masakan serta *class library* yang dibangun dapat mempermudah pengembang perangkat lunak dalam pembangunan aplikasi resep masakan.

Kata Kunci: *Class Library, resep masakan, aplikasi resep masakan..*

1. PENDAHULUAN

Resep masakan telah menjadi kebutuhan sejak lama dan terus berkembang dalam berbagai media sampai media perangkat lunak. Menurut *mobile marketing statistic* 2015 menunjukan bahwa pengguna *mobile* mencapai 1900 juta pengguna. Berdasarkan observasi pada market aplikasi *mobile* didapat fakta bahwa aplikasi resep masakan digemari oleh pengembang aplikasi dan memiliki jumlah pengguna yang besar. Berdasarkan wawancara yang dilakukan kepada beberapa pengembang aplikasi, dalam pembangunan aplikasi pada suatu domain pengembang perangkat lunak cenderung memulai pekerjaan dari awal dikarenakan sulitnya penggunaan ulang, hal ini disebabkan karena pengembang cenderung fokus terhadap pembangunan perangkat lunak dibandingkan analisis terhadap suatu domain aplikasi. Solusi yang dapat diterapkan yaitu membangun *class library*.

1.1. Class Library

Class library yaitu perpustakaan kelas yang dapat diguna ulang oleh pengembang aplikasi dalam pembangunan aplikasi pada suatu domain [1]. *Class library* menyediakan fungsionalitas dasar dan dirancang untuk menjadi pondasi dalam aplikasi, komponen, dan kontrol yang dibangun. Tahapan dalam pembangunan *class library* yaitu analisis domain, analisis *frozen spot*, analisis *hotspot*, perancangan kelas, perancangan *package*, implementasi *class library*, pengujian *class library*.



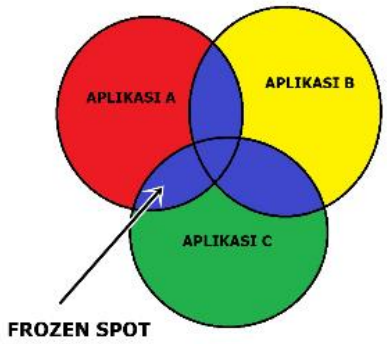
Gambar 1 Tahapan Pembangunan Class Library

1.2. Analisis Domain

Analisis domain merupakan tahap pengumpulan informasi dimana informasi tersebut didefinisikan dengan tujuan agar dapat digunakan kembali [2]. Analisis domain dilakukan dengan cara melakukan observasi terhadap tiga buah aplikasi resep masakan.

1.3. Analisis *Frozen spots*

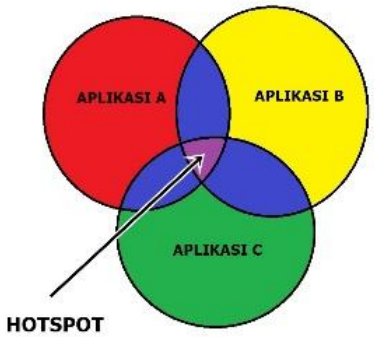
Frozen spot merupakan suatu fungsional yang biasanya ada pada suatu domain aplikasi [3]. *Frozen spot* dapat diurai menjadi *hotspot*, namun tidak semua *frozen spot* dapat diurai menjadi hotspot sehingga suatu fungsionalitas yang tidak bisa diurai menjadi *hotspot* akan tetap menjadi *frozen spot*.



Gambar 2 Analisis *Frozen Spot*

1.4. Analisis *Hotspots*

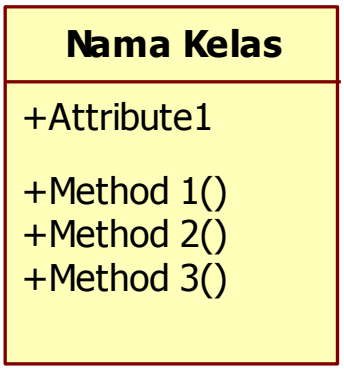
Hotspot merupakan inti dari pembangunan *class library*, *hotspot* berisi kumpulan fungsionalitas yang akan diimplementasikan menjadi *class library*. *Hotspot* dapat disesuaikan tingkat fleksibilitasnya sesuai dengan kebutuhan pengguna [3].



Gambar 3 Analisis *Hotspot*

1.5. Perancangan Kelas

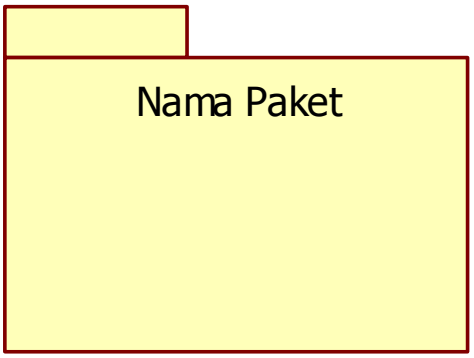
Perancangan kelas dilakukan berdasarkan *hotspot* yang terbentuk dimana *hotspot* dikelompokan berdasarkan penggunaan data yang sama selanjutnya dirancang struktur kelas baik atribut, metode, dan hubungan antar kelas. Perancangan kelas disesuaikan dengan pedoman pembangunan *class library*.



Gambar 4 Perancangan Kelas

1.6. Perancangan *Package*

Perancangan *package* merupakan pengelompokan kelas dalam suatu paket. Perancangan *package* bertujuan agar mempermudah pengguna dalam memahami *class library* yang dibangun.



Gambar 5 Perancangan Paket

1.7. Pengujian *Unit Testing*

Unit testing adalah sebuah kode yang ditulis oleh pengembang yang digunakan untuk menguji bagian kecil atau area spesifik dari suatu fungsionalitas kode yang akan diuji [4]. Pengujian

unit testing pada *class library* dilakukan pada setiap *method* yang ada pada setiap *class*.

1.8. Pengujian *Integration Testing*

Pengujian *integration testing* merupakan perluasan dari pengujian *unit testing*. Dua buah unit yang telah diuji diintegrasikan dan dibentuk menjadi sebuah komponen, selanjutnya dilakukan pengujian terhadap komponen tersebut [5]. Pengujian *integration testing* dilakukan dengan cara membangun perangkat lunak resep masakan berbasis *windows phone*.

1.9. Pengujian *Acceptance*

Pengujian *acceptance* adalah pengujian yang berhubungan dengan kebutuhan pengguna, persyaratan, dan proses bisnis yang dilakukan untuk menentukan apakah suatu sistem memenuhi kriteria penerimaan atau tidak, dan pengguna atau badan lain yang berwenang sangat mungkin untuk menentukan untuk menerima sistem atau tidak [6].

2. ISI PENELITIAN

2.1. Analisis Domain

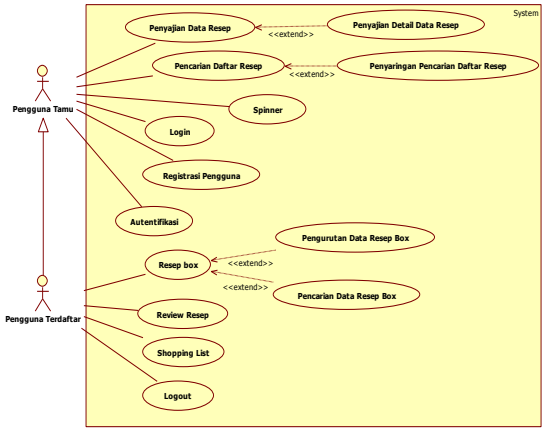
Analisis domain aplikasi dilakukan dengan observasi terhadap tiga aplikasi resep masakan. Aplikasi resep masakan yang diambil yaitu Allrecipes Dinner Spinner, Big Oven dan Allthecooks.

Tabel 1 Analisis Domain

No	Nama Aplikasi	Rating	Jumlah Unduhan
1	Allrecipes Dinner Spinner	4.3	1 Juta
2	Big Oven	4.3	1 Juta
3	Allthecooks	4.3	10 Juta

1. Allrecipes

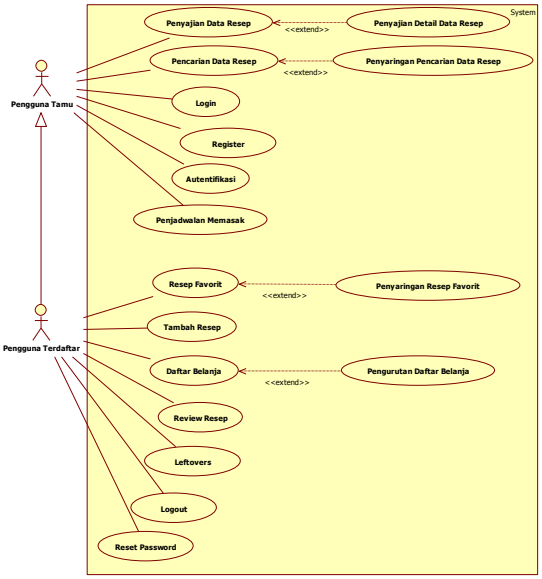
Berikut analisis use case untuk aplikasi allrecipes:



Gambar 6 Analisis *Use Case* Aplikiasi Allrecipes

2. BigOven

Berikut analisis *use case* untuk aplikasi BigOven:



Gambar 7 Analisis Use Case Aplikasi BigOven

3. Allthecooks

Berikut analisis *use case* untuk aplikasi Allthecooks:



Gambar 8 Analisis Use Case Aplikasi Allthecooks

2.2. Analisis *Frozen Spot*

Setelah analisis domain dilakukan, maka ditentukan fungsionalitas yang merupakan *frozen spot* dari ketiga aplikasi tersebut. Berikut hasil *frozen spot* yang didapat.

Tabel 2 Analisis *Frozenspot*

Fungsionalitas	Keterangan
Penyajian daftar resep masakan	Penyajian daftar masakan pada tampilan utama setiap aplikasi.
Penyajian detail resep masakan.	Penyajian detail resep masakan terdapat pada setiap aplikasi.
Pencarian resep	Pencarian terdapat pada setiap aplikasi.
Penambahan resep masakan	Menu penambahan resep masakan hanya terdapat pada aplikasi Big Oven dan Allthecooks
Tambah Daftar Belanja	Tambah daftar belanja terdapat pada setiap aplikasi
Penyajian Daftar belanja	Penyajian daftar belanja terdapat pada setiap aplikasi.
Hapus Daftar Belanja	Hapus Daftar belanja terdapat pada setiap aplikasi
Tambah *Review* resep	Review tambah resep masakan terdapat pada setiap aplikasi.
Penyajian *Review* Resep	Penyajian review resep terdapat pada setiap aplikasi.
Penjadwalan masak	Penjadwalan masak hanya terdapat pada aplikasi Big Oven dan Allthecooks, sedangkan pada aplikasi allrecipes tidak terdapat
Penambahan favorit	Penambahan favorit terdapat pada aplikasi.
Penyajian Resep favorit	Resep favorit pengguna terdapat pada setiap aplikasi.
Login	*Login* pengguna terdapat pada setiap aplikasi.

2.3. Analisis Hotspot

Setelah analisis *frozen spot*, maka *hotspot* dapat didefinisikan berdasarkan penguraian *frozen spot*. Berikut daftar *hotspot* yang didapatkan.

Tabel 3 Analisis *Hotspot*

No	Nama *Hotspot*
1	Penyajian daftar resep masakan
2	Pencarian resep masakan
3	Penyajian detail resep masakan.
4	Daftar Belanja
5	Penyajian Review Resep
6	Tambah Review
7	Resep Favorit
8	Login
9	Register
10	Tambah Resep

Setelah *hotspot* didefinisikan maka dibuat kartu *hotspot* untuk masing-masing fungsionalitas. Kartu *hotspot* akan menjelaskan mengenai detail setiap fungsionalitas *hotspot*.

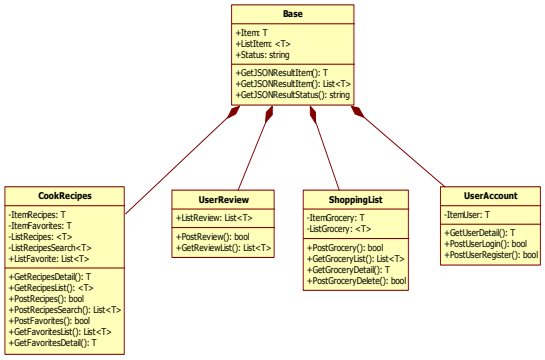
2.4. Perancangan Kelas

Perancangan kelas didefinisikan berdasarkan pengelompokan *hotspot* yang memiliki data yang sama, perancangan kelas disesuaikan dengan pedoman pembangunan *class library*.

Tabel 4 Percangan Kelas

Hotspot	Kelas
Penyajian daftar resep masakan	CookRecipes
Pencarian resep masakan	CookRecipes
Penyajian detail resep masakan.	CookRecipes
Daftar Belanja	ShoppingList

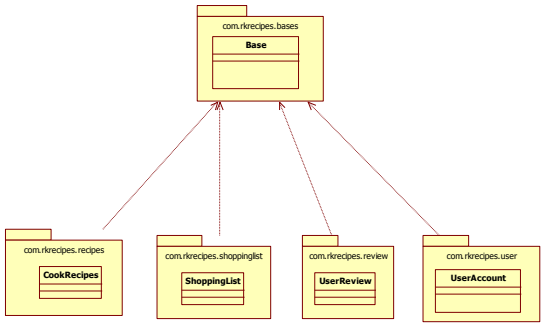
Penyajian Review Resep	UserReview
Tambah Review	UserReview
Resep Favorit	CookRecipes
Login	UserAccount
Register	UserAccount
Tambah Resep	CookRecipes



Gambar 9 Perancangan Kelas *Class Library*

2.5. Perancangan *Package*

Setelah perancangan kelas terbentuk, dilakukan perancangan *package* guna mempermudah pengguna *class library* dalam mengetahui struktur kelas.



Gambar 10 Perancangan *Package*

2.6. Implementasi *Class Library*

Setelah perancangan kelas maka dilakukan implementasi class library kedalam kode program dan di *built* menjadi sebuah *Dynamic Link Library* (.dll)



Gambar 11 *Dynamic Link Library*

2.7. Rencana Pengujian

Sebelum pengujian dilakukan, diperlukan rencana pengujian untuk menentukan tahap pengujian yang akan dilakukan. Berikut rencana pengujian pada penelitian ini:

1. Rencana Pengujian
2. Skenario Pengujian
3. Hasil Pengujian
4. Evaluasi Pengujian

2.8. Skenario Pengujian

Pengujian *class library* dilakukan dengan cara *unit testing*, *integration testing*, dan *acceptance testing*. Berikut skenario untuk masing-masing pengujian tersebut

Tabel 5 Skenario Pengujian *Unit Testing*

Kelas Uji	Jenis Pengujian
Kelas CookRecipes	*Unit Testing*
Kelas UserReview	*Unit Testing*
Kelas ShoppingList	*Unit Testing*
Kelas UserAccount	*Unit Testing*

Tabel 6 Skenario Pengujian Integration Testing

Komponen Pengujian	Jenis Pengujian
Tampil Detail Resep	*Integration Testing*
Tampil Daftar Resep	*Integration Testing*
Tambah Resep	*Integration Testing*

Pencarian Resep	*Integration Testing*
Tambah Resep Favorite	*Integration Testing*
Tampil Daftar Favorite	*Integration Testing*
Tampil Detail Favorite	*Integration Testing*
Uji method PostReview	*Integration Testing*
Uji method GetReviewList	*Integration Testing*
Uji method GetGroceryList	*Integration Testing*
Uji method GetGroceryDetail	*Integration Testing*
Uji method PostGrocery	*Integration Testing*
Uji method PostGroceryDelete	*Integration Testing*
Uji method GetUserDetail	*Integration Testing*
Uji method PostUserLogin	*Integration Testing*
Uji method PostUserRegister	*Integration Testing*

2.9. Hasil Pengujian

Hasil pengujian dibagi tiga yaitu hasil pengujian *unit testing*, *integration testing*, dan pengujian *acceptance*.

1. Hasil Pengujian *Unit Testing*

Berikut hasil pengujian untuk setiap kelas

a. Kelas CookRecipes Dengan Data Benar

Tabel 7 Kelas CookRecipes Dengan Data Benar

Kasus Uji	Kesimpulan
Uji *method GetRecipesDetail*	Diterima
Uji *method GetRecipesList*	Diterima
Uji *method PostRecipes*	Diterima
Uji *method PostRecipesSearch*	Diterima
Uji *method PostFavorites*	Diterima
Uji *method GetFavoriteList*	Diterima
Uji *method GetFavoriteDetail*	Diterima

b. Kelas CookRecipes Dengan Data Salah

Tabel 8 Kelas CookRecipes Dengan Data Salah

Kasus Uji	Kesimpulan
Uji *method GetRecipesDetail*	Ditolak
Uji *method GetRecipesList*	Ditolak
Uji *method PostRecipes*	Ditolak
Uji *method PostRecipesSearch*	Ditolak
Uji *method PostFavorites*	Ditolak
Uji *method GetFavoriteList*	Ditolak
Uji *method GetFavoriteDetail*	Ditolak

c. Kelas UserReview Dengan Data Benar

Tabel 9 Kelas CookRecipes Dengan Data Salah

Kasus Uji	Kesimpulan
Uji *method PostReview*	Diterima
Uji *method GetReviewList*	Diterima

d. Kelas UserReview Dengan Data Salah

Tabel 10 Kelas UserReview Dengan Data Salah

Kasus Uji	Kesimpulan
Uji *method PostReview*	Ditolak
Uji *method GetReviewList*	Ditolak

e. Kelas ShoppingList Dengan Data Benar

Tabel 11 Kelas ShoppingList Dengan Data Benar

Kasus Uji	Kesimpulan
Uji *method GetGroceryList*	Diterima
Uji *method GetGroceryDetail*	Diterima
Uji *method PostGrocery*	Diterima
Uji *method PostGroceryDelete*	Diterima

f. Kelas ShoppingList Dengan Data Salah

Tabel 12 Kelas ShoppingList Dengan Data Salah

Kasus Uji	Kesimpulan
Uji *method GetGroceryList*	Ditolak
Uji *method GetGroceryDetail*	Ditolak

Kasus Uji	Kesimpulan
Uji *method PostGrocery*	Ditolak
Uji *method PostGroceryDelete*	Ditolak

g. Kelas UserAccount Dengan Data Benar

Tabel 13 Kelas UserAccount Dengan Data Benar

Kasus Uji	Kesimpulan
Uji *method GetUserDetail*	Diterima
Uji *method PostUserLogin*	Diterima
Uji *method PostUserRegister*	Diterima

h. Kelas UserAccount Dengan Data Salah

Tabel 14 Kelas UserAccount Dengan Data Salah

Kasus Uji	Kesimpulan
Uji *method GetUserDetail*	Ditolak
Uji *method PostUserLogin*	Ditolak
Uji *method PostUserRegister*	Ditolak

2. Hasil Pengujian *Integration Testing*

Pengujian *integration testing* merupakan perluasan dari pengujian *unit testing*. Dua buah unit yang telah diuji diintegrasikan dan dibentuk menjadi sebuah komponen, selanjutnya dilakukan pengujian terhadap komponen tersebut. Pengujian *integration testing* dilakukan dengan cara membangun perangkat lunak resep masakan berbasis *windows phone*. Pembangunan perangkat lunak dilakukan dengan menggunakan *class library* yang telah dibangun.

Tabel 15 Hasil Pengujian Integration Testing

No	Implementasi Kelas	Metode yang digunakan
1	*GetRecipesDetail*	*GetRecipesDetail :T*
2	*GetRecipesList*	*GetRecipesList* : List<T>
3	*PostRecipes*	*PostRecipes*:T
4	*PostRecipesSearch*	*PostRecipesSearch*: List<T>
5	*PostFavorites*	*PostFavorites*: bool
6	*GetFavorites*	*GetFavoritesList* : List<T>
7	*PostReview*	*PostReview*: bool
8	*GetReviewList*	*GetReviewList* : List<T>
9	*GetGroceryList*	*GetGroceryList* : List<T>
10	*PostGrocery*	*PostGrocery*:bool
11	*PostUserLogin*	*PostUser:bool*
12	*PostUserRegsiter*	*PostUser:bool*

3. Hasil Pengujian *Acceptance*

Tabel 16 Hasil Pengujian Acceptance

No	Daftar Pertanyaan	Kesimpulan Wawancara
1	Apakah *class library recipes apps* dapat membantu anda dalam membangun aplikasi resep masakan?	Iya
2	Apakah *class library recipes apps* dapat membantu anda dalam membuat fungsionalitas pada pembangunan aplikasi resep masakan?	Iya
3	Apakah dengan adanya dokumentasi *class library recipes apps*, penggunaan *class library* jadi lebih mudah?	Iya
4	Apakah fungsional-fungsional yang ada pada *class library recipes apps* sudah cukup mewakili fungsionalitas dasar pada aplikasi resep masakan?	Iya
5	Pada *class library recipes apps* tidak dibangun *exception* atau penanganan *error*, apakah menurut anda hal tersebut diperlukan?	Sangat Perlu

2.10. Evaluasi Pengujian

Berdasarkan hasil pengujian yang telah dilakukan maka dilakukan evaluasi terhadap ketiga hasil pengujian tersebut. Berikut penejelasan untuk masing-masing hasil pengujian

1. Evaluasi Pengujian *Unit Testing*

Berdasarkan pengujian *unit testing* yang telah dilakukan, dapat disimpulkan bahwa setiap unit pada *class library* yaitu *method* telah berjalan dengan baik, namun tidak ada penanganan *exception* atau *error* untuk data masukan yang salah.

2. Evaluasi Pengujian *Integration Testing*

Berdasarkan pengujian *integration testing* yang telah dilakukan, dapat disimpulkan bahwa setiap unit pada *class library* dapat berintegrasi dengan unit lainnya menjadi sebuah komponen yang berjalan

dengan baik pada pembangunan perangkat lunak aplikasi resep masakan.

3. Evaluasi Pengujian *Acceptance*

Berdasarkan hasil wawancara terhadap tiga orang pengembang aplikasi dapat disimpulkan bahwa:

a. Fungsionalitas pada *class library* yang dibangun telah mewakili fungsionalitas dasar untuk domain aplikasi resep masakan.
b. C*lass library* yang dibangun telah mempermudah pengembang perangkat lunak dalam membangun aplikasi resep masakan.
c. *Class library* yang dibangun perlu ditambahkan *exception* atau penanganan *error*.

3. KESIMPULAN

Berdasarkan penelitian yang telah dilakukan maka dapat disimpulkan bahwa *class library* untuk domain aplikasi resep masakan yang dibangun telah menyediakan fungsionalitas dasar aplikasi resep masakan dan telah mempermudah pengembang aplikasi dalam membangun aplikasi resep masakan.

4. DAFTAR PUSTAKA

[1] Microsoft Corporation, "Class Library," [Online]. Available: https://msdn.microsoft.com/en-us/library/d11h6832%28v=vs.71%29.aspx.

[2] A. Valerio, G. Succi and M. Fenaroli, "Domain Analysis and Framework-based Software Development," [Online]. Available: http://www.sigapp.org/acr/Issues/V5.2/valerio.html. [Accessed 17 April 2015].

[3] M. E. Fayad, D. C. Schmidt and R. E. Johnson, Building Application Frameworks Object-Oriented Foundations of Framework Design, Wiley Computer Publishing, 1999.

[4] L. Hunt, C# Coding Standards for .NET, Lance Hunt, 2007.

[5] Microsoft Corporation, "Integration Testing," [Online]. Available: https://msdn.microsoft.com/en-us/library/aa292128(v=vs.71).aspx. [Accessed 27 07 2015].

[6] Wikipedia, "Acceptance Testing," [Online]. Available: https://en.wikipedia.org/wiki/Acceptance_testing. [Accessed 25 07 2015].