Development Of The Class Library In Personal Finance Domain Applications

Argi Sugiyarsa[1]

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

ABSTRAK

Each individual requires the ability to manage financial resources either manually or by using a mobile application. Many users require a personal finance application, became the trigger for software developers in building software personal finance. From the results of interviews with some of the programmers, found that information in building software in a domain always starts from the beginning because developers focusing on the process of implementation rather than its analysis. Solutions that can be applied is to build the class library.

Class library is a collection of classes that were already written before it can be used again by the programmer in developing software. In building a class library, it required some step to do, the first step is to get a functional domain analysis is the basis of the applications analyzed, the next step is the analysis of frozen spots and hot spots to determine the functional equation of the analyzed applications. The next step is the design of a class of hot spot that has been obtained, the class was formed later grouped with the design of the package.

Class libraries that are already designed and then implemented in the code and do the testing. To test the class libraries is by using unit testing, integration testing, and acceptance testing. From the test results it can be concluded that the construction of the class library domain case on personal finance can ease the programmer in developing personal finance software.

Key Word: class library, personal finance.

1. FOREWORD

Each individual requires the ability to manage financial resources either manually or by using a mobile application [1]. *At this time the management of financial resources are already using technology such as mobile software* [2]. *Mobile Marketing Research Statistics 2015 the number of global mobile applications the user was 1900 million. Every day new applications are born no exception application personal finance* [3]. *From observations in the windows store found more than 1000 software personal finance, analysis of IDC (International Data Corporation) says windows phone market share will increase by 10% in 2017* [4].

Build or develop software on a domain is complex and requires a long process [5] . *From the results of interviews with some of the programmers, found that information in building software in a domain always starts from the beginning because developers focusing on the process of implementation rather than its analysis. Solutions that can be applied is to build the class library.*

1.1 Class Library

The class library is a library of classes that were already written before and it can be used again by the programmer in software development [6].

1.2 Method Of Construction Class Library

Method of construction class library is composed by several steps such as domain analysis, frozen spot analysis, hot spot analysis, design of class, design of package, implementation class library, and testing [7].

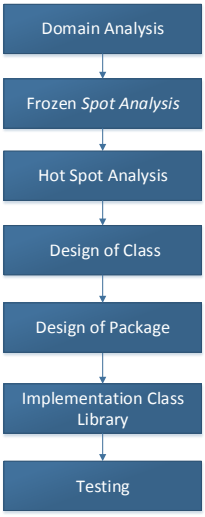


Figure 1 Method Of Construction Class Library

1.3 Personal Finance Domain

Personal finance management is the knowledge of how individuals or families to manage their financial resources [8]. *individual or family do*

recording the flow of money, whether it's outflow or inflow of financial resources, the process can be done many ways like taking notes on a book of finance or by remembering the flow of money that occurs in a certain period of time, but is currently more relevant to the use of technology in personal finance management [2].

1.4 Domain Analysis

Domain analysis can present a systematic approach to identify the scope, features, and points variations on a domain [9]. *Domain analysis is a grouping of software process seen from the similarity of their functionality.*

1.5 Frozen Spot

Frozen spot identical functionality is present on the domain of the software [7].

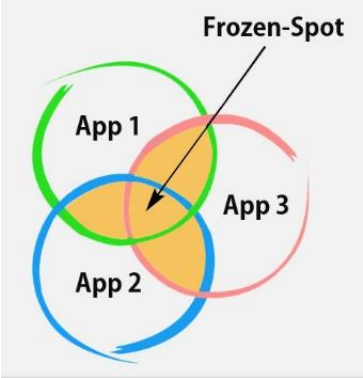


Figure 2 Frozen Spot

1.6 Hot Spot

Hot spot is functional on each software in a domain.

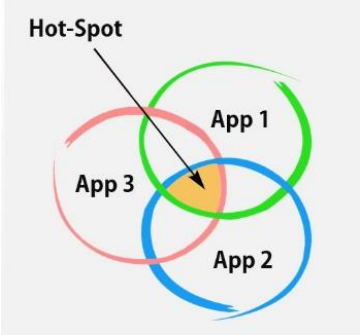


Figure 3 Hot Spot

1.7 Hot Spot Card

Hot spot card used to know the behavior of a functionality that will be implemented. The contents of a host spot card is the name of hot spot, flexibility, description, and functionality.

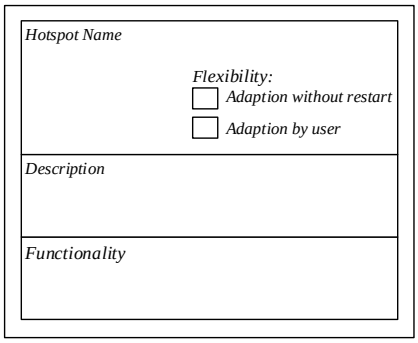


Figure 4 Hot Spot Card

1.8 Testing

The test class library can be implemented some way of testing such as unit testing, integration testing, and acceptance testing.

1. *Unit Testing*

   Unit Tesing is a code written by developers who are used to test a small section of a specific area or the functionality of the code will be tested [3].

2. *Integration testing*

   Integration testing is a logical extension of unit testing[8]. *Step of integration testing is to input modules that have been tested on the unit testing phase, processed into sub integration testing (interaction testing, UI testing, etc.).*

3. *Acceptance testing*

   Acceptance testing is a formal testing with respect to user needs, requirements, and business processes conducted to determine whether or not a system satisfies the acceptance criteria and to enable the user, customers or other authorized entity to determine whether or not to accept the system [11].

2. CONTENT

2.1 Domain software analysis

An analysis of the three domains of personal finance software. Personal finance software selected by the rating of the site windows store. The results of domain analysis can be seen in ***Error! Reference source not found.***.

Table 1 Domain Analysis Software

Application Name	*Rating*
IMoney V 1.2.0	***4 Star***
myFinance V 1.5.1.0	***4 Star***

Toshl Finance V 1.8.9.3	***4 Star***

2.2 Frozen Spot Analysis

After domain analysis has been done, do the analysis of frozen spot with the Group and defines the functionality that exists on every software. The results of frozen spot analysis can be seen in Table 2.

Table 2 Frozen Spot Analysis

Fitur	Personal Finance			Ket
	App 1	App 2	App 3	
Create Account	✔	✔	✔	*Each application has this feature with different parameters.*
List of Currency	✔	✔	✔	*Each application has a feature list of currencies*
Add Transaction	✔	✔	✔	*Each application has a feature add transaction*
List of Category	✔	✔	✔	*Each application has a feature list of category*
Add Transaction Category				*Each application has a feature add transaction category*
Detail of Transaction	✔	✔	✔	*Each application has a feature detail of transaction*
Add Currency		✔	✔	*This feature only found on myFinance and toshl finance*
Add Schedule	✔			*This feature only found on I-Money*
Report	✔	✔	✔	*Each application has a feature report*
Login			✔	*This feature only found toshl finance*
Sign Up			✔	*This feature only found toshl finance*

2.3 Hot Spot Analysis

After analysis of the frozen spot was completed, then a hot spot analysis can be performed, candidates hot spots obtained from the definition of frozen spot. The results of hot spot analysis can be seen in Table 3.

Table 3 Hot Spot Analysis

No	*Hot Spot*
1	*Add transaction*
2	*List of transaction*
3	*Detail of transaction*
4	*Report of transaction*
5	*Add transaction category*
6	*Add account*
7	*Add type of currency*
8	*List of currency type*
9	*Report of transaction*
10	*Login*
11	*Sign Up*

2.4 Design of Class

This section will do design classes from any hot spot that is already defined. Every hot spot will be grouped based on data managed. . The results of domain analysis can be seen in Table 4 and Figure 5.

Table 4 Design of class

Hot Spot	*Name of Class*
Add transaction	*Transaction*
List of transaction	

Detail of transaction	
Report of transaction	
Add transaction category	*CategoryTransaction*
Add account	
Add type of currency	*Wallet*
List of currency type	*CurrencyTypes*
Report of transaction	
Login	*Report*
Sign Up	*User*
Add transaction	
-	*Bases*

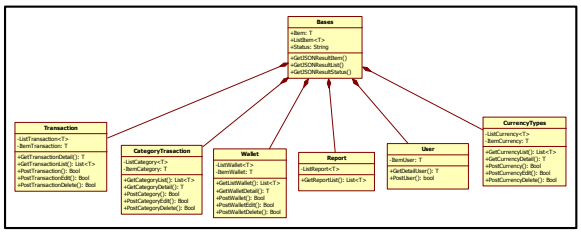


Figure 5 Class Diagram personal finance class library

2.5 Design of Package

*The design of package is the process of grouping related classes or similar, with a design package understanding class structure can be easily. The design of package on a personal finance class library can be seen in Figure 6****Error! Reference source not found.****.*

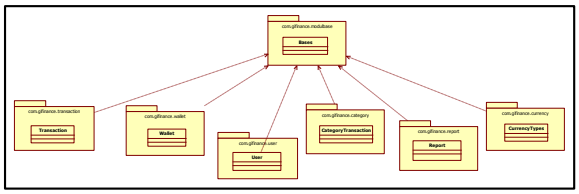


Figure 6 Design of package

2.6 Implementation of Class Library

Class library implementations is a class that is already designed in advance, then the class will be compiled into Dynamic Link Library files (dll).

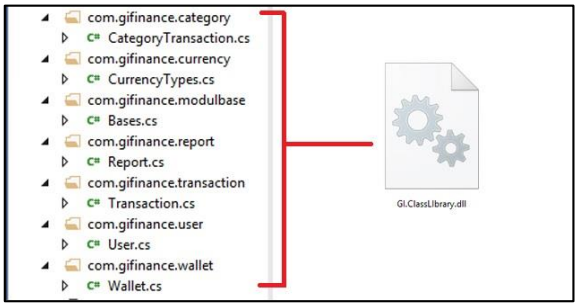


Figure 7 Implementation of class library

2.7 Testing of Class Library

This section will explain the testing on a personal finance class libraries that have been built, testing on a personal finance class library is unit testing, integration testing, and acceptance testing. The test class library scenario can be seen in Table 5 and Table 6.

Table 5 Scenario of Unit Testing

Class Testing	*Type of Testing*
Transaction	*Unit Testing*
Wallet	*Unit Testing*
CategoryTransaction	*Unit Testing*
CurrencyTypes	*Unit Testing*
User	*Unit Testing*
Report	*Unit Testing*

Table 6 Scenario Integration Testing

Item	*Type of Testing*
Detail Transaction	*Integration Testing*
Get Transaction	*Integration Testing*
Add Transaction	*Integration Testing*
Edit Transaction	*Integration Testing*
Delete Transaction	*Integration Testing*
Detail Wallet	*Integration Testing*
Get Wallet	*Integration Testing*
Add wallet	*Integration Testing*
Edit wallet	*Integration Testing*
Delete wallet	*Integration Testing*
Detail User	*Integration Testing*
Add User	*Integration Testing*
Detail Currency	*Integration Testing*
Get Currency	*Integration Testing*
Add Currency	*Integration Testing*
Edit Currency	*Integration Testing*
Delete Currency	*Integration Testing*
Detail Category	*Integration Testing*
Get Category	*Integration Testing*
Add Category	*Integration Testing*
Edit Category	*Integration Testing*
Delete Category	*Integration Testing*
Report	*Integration Testing*

3. CONCLUSION

From the results of research, the developer of a class library personal finance can make easier programmers to build software personal finance. Class library personal finance the main functionality has been providing needed in personal finance software development.

The suggestion from the results of this study are able to add error handling and exception or currency exchange rates information which are not performed in this study.

4. REFERENCES

[1] D. Nababan and I. Sadila, "Analisis Personal Financial Literacy dan Financial Behavior Mahasiswa Strata 1 Fakultas Ekonomi Iniversitas Sumatra Utara," *Jurnal Universitas Sumatra Utara*, 2013.

[2] G. Satriaji, Merencanakan Keuangan Pribadi dengan GNUCASH, Tulungagung: Galihsatria Publishing, 2012.

[3] A. Hunt, D. Thomas and M. Hargett, Pragmatic Unit Testing In C# With NUnit Second Edition, United States of America: The Pragmatic Programmers, 2007.

[4] R. Edmonds, "IDC: Bright future for Windows Phone with 10 percent of the global smartphone market by 2017," windows central, 5 September 2013. [Online]. Available: http://www.windowscentral.com/idc-bright-future-windows-phone-10-per-globally-2017. [Accessed 11 June 2015].

[5] M. F. Rhomadhoni, "Rancang Bangun Class Library dan Integrasi Sistem pada Game Sosial Food Merchant Saga Pada Perangkat Android," *Teknik Informatika FTIF-ITS*, 2014.

[6] Microsoft Corporation, "Class Library," Microsoft Developer Network, [Online]. Available: https://msdn.microsoft.com/en-us/library/d11h6832%28v=vs.71%29.aspx. [Accessed 27 May 2015].

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

[8] L. J. Gitman, Principle of Finance 11th ed, New Jersey: Prentice Hall, 2004.

[9] A. Bakar, B. Setiawan, G. Syukriawati, L. M. Rahma and W. Wijaya, "Ananlisis Domain pada Website Toko Online," *Program Teknologi Informasi dan Ilmu Komputer Universitas Brawijaya*, 2012.

[10] Microsoft Corporation, "Integration Testing," Microsoft Developer Network, [Online]. Available: https://msdn.microsoft.com/en-us/library/aa292128(v=vs.71).aspx. [Accessed 28 July 2015].

[11] Wikipedia, "Acceptance testing," Wikipedia, [Online]. Available: https://en.wikipedia.org/wiki/Acceptance_testing. [Accessed 29 July 2015].

[12] S. Lynch, "Top 50 Personal Finance and Budget Planning Apps," Direct Capital, 13 May 2014. [Online]. Available: http://blog.directcapital.com/business-insights/finance-lending/top-50-personal-finance-and-budget-planning-apps/. [Accessed 11 June 2015].