Case Study: the Use of Agile on Mortgage Application: Evidence from Thailand

Kreecha Puphaiboon
Faculty of Computer and Information Technology
Kasem Bundit University
Bangkok, Thailand

***Abstract*—This paper presents a case study of a mortgage loan origination project using SCRUM Agile model and Business Process Management and Business Rule Management System (BPMS and BRMS). From the Waterfall model (Stage 1), a web-based self-developed had been developed using opensource frameworks: Spring and Sarasvati. But, several problems were detected and the project failed due to insufficient project management, rapid requirement changes and developer coding skills. The project was continued (Stage 2) selecting a BPMS and BRMS tool. Later, Stage 3 SCRUM was executed with proper project management and the new tool, which suited better for rapid business needs, and minimum coding. An efficient team communication and the frequent delivery of code releases increasingly contributed to the sponsor and user's satisfaction. However, due to political influenced timeline, inexperienced project management and requirement changes, the budget exceeds and SCRUM is not favored. Nonetheless, Open-end questionnaire and interview results with core team members both business users and developers as well as software usability measurement inventory (SUMI) conducted with 14 users, it shows that SCRUM and the new tool rescue the project. Empirically, this paper demonstrates a method to evaluate the use of Agile augmented with usability measurement to Agile development community.**

Keywords*—*SCRUM; BPM; BRE; Mortgage Loan; LOS; Usability

I. Introduction

Mortgage loan origination process is complex, although common process flows are: Application, Processing, Underwriting, Closing and Post Closing [1]. However, a system that can handle a large number of Loan numbers is hard to find. In addition for IT, the pressure to revise flows, policy changes and credit risk calculations in a few days and automatically make correct lending decisions has been a great challenge in retail banking [2]. There's no room for customer and financial information and loan process errors as banks need to have certain confidence in every lending decision.

Agile's principles encourage the formation of collaborative and self-organization teams [3]. The Agile Manifesto is as follows: 1) Individuals and interactions over processes and tools. 2) Working software over comprehensive documentation. 3) Customer collaboration over contract negotiation. 4) Responding to change over following a plan. However, a continual debate surrounds the effectiveness of agile software development practices. Some organizations adopt agile practices to become more competitive, improve software development processes, and reduce costs. Other organizations are skeptical about whether agile development is beneficial. Additionally, large organizations face an additional challenge in integrating agile practices with existing standards and business processes [4].

Whilst it is generally accepted that SCRUM development improves the cost reduction and it helps to accelerate the software product to the market. Importantly, it improves customer satisfaction [5] [6]. However, no field studies research has been reported when Agile and SCRUM is first being used in a large organization in Thailand. Research [5] mentioned factors to run Scrum in aligned with PMP BOK principles (see Table I). Hence, the researcher wants to study and report the usage of Agile/SCRUM to satisfy business needs and assess the impacts over the IT development an example for software engineering community.

TABLE I. step 1. 4, often the projects have been assigned with the timeline [6]. Political forces at work within a project or company can often drive estimation inaccuracy [8]. This is usually in the form of managerial pressure to stay within or meet the estimate timeline [8]. The estimation process can be impacted negatively by these pressures resulting in project timeline or cost constraints [8] [9].

The rate of change in business and bank is accelerating [1] [2] . A number of techniques for addressing that change have emerged independently to provide for automated solutions in this environment. Business Process Management (BPM) and Business Rule Engine (BRE) that are large as well as distributed are becoming more prevalent [10] [11]. Both technologies tend to offer the promise of easy to change. As change is common in large projects; the case where the entirety of a project's complexity is understood in the early stages is quite rare. Large, distributed projects that involve user requirements present a unique challenge that neither agile methods nor waterfall approaches alone can effectively address. Hence, combining an effective software development tool with agile process may be very beneficial.

Koch [12] has proposed three criteria for evaluating the effectiveness of the agile method adopted: 1) project performance with schedule performance and budget performance; 2) management acceptance; 3) customer relationship and 4) team satisfaction. However, usability was not included. Thus, it is important to evaluate all these five criteria for agile adoption for which they were deserved.

I am thankful to Kasem Bundit who has sponsored this research

TABLE I. THE KEY PROCESSES OF RUNNING SCRUM

The key processes of running scrum		
1. Determinate phase	***2. Planning phase***	***3. Start-up phase***
1.1. Develop the real requirements of customers;	2.1. Define all the work of the project;	3.1. Recruit project manager;
1.2. Write a one page project description;	2.2. Establish the schedule of initial project;	3.2. Build the scope change management process;
1.3. Recode the requirement of customers;	2.3. Assess the time required to complete the project;	3.3. Recruit the project team members;
1.4. Gain the senior managers' permission to run the project;	2.4. Analyze and adjust the project schedule;	3.4. Manage the team communication;
1.5. Discuss how to meet the requirements with the customers.	2.5. Assess the resource required to complete the project;	3.5. Write the descriptive document of project;
	2.6. Write the risk management plan;	3.6. Determine the schedule;
	2.7. Assess the whole cost of the project;	3.7. Build the team operating rules;
	2.8. Record the project plan;	3.8. Write the work package.
	2.9. Sort the work in chronological order;	
	2.10. Get the senior management's permission to start the project.	
4. Supervision and control phase	***5. Decided to start the iteration phase***	***6. Closeout phase***
4.1. Build the running and reporting system;	5.1. Decision-making process for customer management;	6.1. Get the confirmation of the customer;
4.2. Report the schedule;	5.2. Customers must be fully involved in this process;	6.2. Prepare for the deliverables and installations.
4.3. Supervise the running;	5.3. The atmosphere must be complete open and honest;	6.3. Write the closeout report;
4.4. Deal with the request of scope change;	5.4. Determination must base on the expected commercial value;	6.4. Start the audit of the running.
4.5. Supervise the risks;	5.5. Solution must be formed according to the project's goal.	
4.6. Identify and solve the problems.		

Agile and usability aim to build quality software. As noted in research [13], agile and usability the two methods have much to offer when they share iterations because the iterations used in agile facilitate usability testing and allow developers to incorporate results of these tests in subsequent iterations. However, research [13] commented that improving the usability of a product does not come without costs. In order to integrate agile and usability and at the same time minimize these costs and risks, we need the use of usability artifacts and practices in a condensed form. SUMI for Software Usability Measurement Inventory is commonly used for usability evaluations [14]. SUMI consists of over 50 questions; it is method of measuring software quality from the end user's point of view. There is a need of usability measurement integrated with agile methodology to determine whether the software supported mortgage loan needs or any domains.

This article presents the case of mortgage loan origination project called LOS. The purpose of LOS was to replace an existing mortgage application as there were problems such as: legacy application (over 15 years old), lack of Power Builder developers to support, difficulty in changing business flows, incapable of scalability and performance issue. There were over 8,000 users using the mortgage system, roughly around 500 concurrent users across the nation. There are needs for new application with features: easy-to-change, web-based and scalable, user to make routine changes. Also, the emphasis is on IT to have a fast delivery of the software application to compete with market trends and attract customers, the faster the delivery of software are the chances that the bank will gain profits. Note that the writer is the technical manager of this project. The bank is ranked in the top five banks in Thailand. The revenue of mortgage loan was over 1 billion us dollars in year 2012. So, it is a highly critical system for the bank.

The paper is structured as follows: Section 2 presents the development stages of delivering a prototype, including detected problems. Section 3 explains ways in which BPM/BRE tools were selected. Section 4 describes the full development SCRUM model and some properties of the methodology are summarized of the system and its iterations. Section 5 shows team comments on Agile/SCRUM base on questionnaire and feedbacks via Sticky Notes and user experience assessment to the software via SUMI usability questionnaire. Lesson learned and conclusions are presented in Section 6.

II. STAGE ONE OF MORTGAGE APPLICATION – PROTOTYPE SELF-DEVELOPMENT

Using Waterfall model, the development of web-based mortgage application the project was initiated (2011–2012) timeline and main tasks were planned by senior management with four months for each step of requirements, coding and UAT testing. Two main representatives' of business users were given. Requirement gathering in June – September 2011, there were 18 main flows from start to finish entire loan process e.g. data entry, manager, credit approval officer, credit approval manager, legal contract and legal managers. The direction was to used open-source framework and in-house development the J2EE, Spring Framework (2009) and Sarasvati proposed by IT developers. Spring is used to handle simultaneous runs and as an interface to database DB2 (see in Appendix for Fig. 4). The development effort was plan roughly for 14,400 man hours for the entire project for one project manager, two business users, 10 Java developers (average experienced 3.5 years) and 3 testers. The budget was 180,000 USD.

Application development was carried out in October 2011 - January 2012, but different problems threatened the project continuity. Considering the SIT was unable to finish within the first two month (1 month delayed) due to numerous bugs. The team had difficulties to follow defined plans and the senior executives were unsatisfied with the progress of the software development after 6 months. The following main deficiencies were detected:

- Deficient project management and communication; project manager, domain experts and end users were present only six-hours per week. They also sit in different buildings with IT developers. Telephone and e-mail were used. These means did not result efficient when problem arisen and need immediate action from BAs.
- Requirements were broad without enough details and cannot integrate entire flow. For example, details of different collateral types between two units are not in sync (processing and underwriting). Therefore, most of the coding tasks required impact analysis, modifications and re-implementations, causing continuous delays to the development process.
- The implications are poor change control, developers' software design skill and software framework to adopt dynamic changes.
- The developer stated that complicated flow, business rule calculations and changes were the root cause of failures.

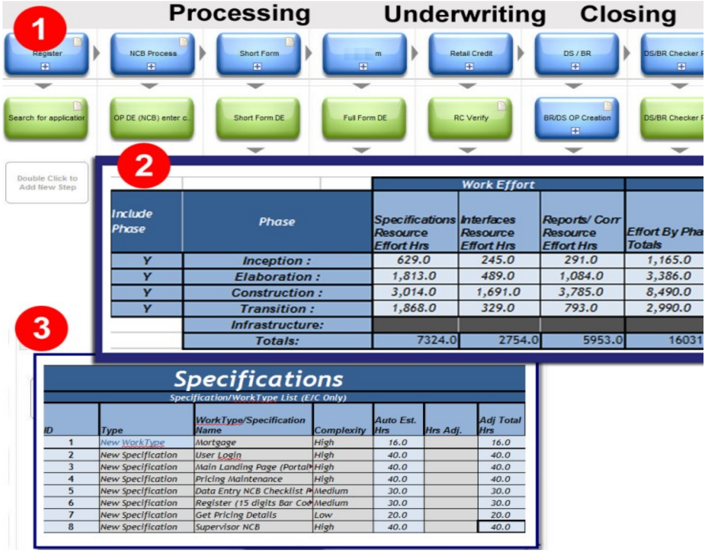
The objectives of self-develop web application project were clearly not met. Significantly the budget was exceeded by 60,606 USD. After a meeting between the team and executives (CFO, COO and CIO), all decided to discontinue the self-development. CIO advised the project to acquire BPM and BRE tools which support developer to code as less as possible and users can manipulate the flow and rules within the system. This is to downgrade possible failure risks. Importantly, the team requested access to the expert user on a daily and full-time basis.

III. Stage Two of Mortgage Application – Selecting and Assessing the Tool

On February 2012 the BPM and BRE Vendor Selection was executed. The top product listed in Gartner and Forrester ranking reports were invited. Importantly and practically, two weeks of POC project to build almost half of an entire mortgage process arisen from Stage 1 including SIT and UAT were conducted. Moreover, stress test with 500 concurrent users was conducted and passed 3 seconds respond time criteria. Developers and users from the bank also involved in the development as well as evaluation of the software. This is to prove that the selected BPM/BRE framework provided software flexibility for fast development without much coding and ease of change when users want to change various calculation schemes. For end users, they appreciated the fact that they can input decision rules and the software provides friendly and intuitive screens for users. Another additional benefit for developers is that the software supports Agile development: 1) users can draw flow and design screen with developers in real time without coding this helps to improve business gaps with users; 2) flows and rules can be drawn/changed without coding; and 3) requirements and documents are saved in the system so developers can identify changes. Another benefit for the bank is that financially, 5 Year Total Cost of Ownership (TCO) is less than other products. The product also offers Cloud Amazon EC2 service, thus it leverages maintenance agility and investment cost. 5 year TCO of the project is around 10 million US dollars. But for mortgage application is funded with 1.5 million USD where outsource developers budget is around 330,000 USD. Time-and-Material contracts and Labor Hour (LH) contracts are used. Three weeks of training was also provided to local staff including business analysts.

IV. Stage Three of Mortgage Application – Application of SCRUM

On June 2012, the Stage 2 of the software development finally started. As it was learnt that requirement documentation was not clear, hence the re-documentation/requirement were captured and put directly into the system. Two weeks period with multiple sessions captured the details of the use cases and flow which give the project traceability and determine application requirements. During the requirement gathering, a close seating and direct communication environment within a big room with a projector, design sketches and white-board, notes and mocked up screen were conducted. These meetings focus on efficiency, getting 2 subject matter experts (SME) from 2 departments into the room to focus on the implementation details of mocking up UI, validation of inputs and outputs and aware of each other impacts. They agreed up front on how the application processes will work, avoiding costly rework later. The outcomes determined the number of iterations, sprints and effort required for the project. Due to business confidentiality, Fig. 1 shows some of business process flow of Application Registration, Processing and Underwriting (without Closing and Post Closing). More details will be discussed in Section 4.1 – 4.5. The sizing effort by developers was produced with 11,480 man hours for coding and unit testing (see Fig. 1 circled number 2) with total of 213 use cases (see some specification below in Fig. 1 circled number 3).



		Work Effort			
Include Phase	Phase	Specifications Resource Effort Hrs	Interfaces Resource Effort Hrs	Reports/ Corr Resource Effort Hrs	Effort By Pha Totals
Y	Inception :	629.0	245.0	291.0	1,165.0
Y	Elaboration :	1,813.0	489.0	1,084.0	3,386.0
Y	Construction :	3,014.0	1,691.0	3,785.0	8,490.0
Y	Transition :	1,868.0	329.0	793.0	2,990.0
	Infrastructure:				
	Totals:	7324.0	2754.0	5953.0	16031

Specifications

Specification/WorkType List (E/C Only)

ID	Type	WorkType/Specification Name	Complexity	Auto Est. Hrs	Hrs Adj.	Adj Total Hrs
1	New WorkType	Mortgage	High	16.0		16.0
2	New Specification	User Login	High	40.0		40.0
3	New Specification	Main Landing Page (Portal	High	40.0		40.0
4	New Specification	Pricing Maintenance	High	40.0		40.0
5	New Specification	Data Entry NCB Checklist P	Medium	30.0		30.0
6	New Specification	Register (15 digits Bar Coc	Medium	30.0		30.0
7	New Specification	Get Pricing Details	Low	20.0		20.0
8	New Specification	Supervisor NCB	High	40.0		40.0

Fig. 1. Flows of Mortgage Application

However, TABLE II shows the output sizing sheet and effort for the project which was influenced by the senior executives to 7,524 man-hours (reduced 34%).

TABLE II. ITERATIONS PLAN AND SIZING EFFORTS

Steps	Political Forced Effort and Initial Effort in Man-Hour		
	Name	*Political*	*Plan*
1	Application Registration	1,332	440
2	Application Processing	720	380
3	Underwriting and Closing	480	3,296
4	User Management and Change of Condition	2,256	502
5	Risk Analysis and Interface	2,736	6,862
Total		**7,524**	**11,480**

In this project, the system development team was integrated by four groups with the following roles:

- Developers: agile defines specific categories team lead designer and programmers. In this case, they were cross-functional and allocated dynamically depending on particular needs of the running iterations. Four developer teams were agreed between three different vendors and the bank. 23 developers were engaged and 16 were outsources from India, Singapore and Hong Kong averaged 4.0 years experienced with the product. Higher number of outsource the reason being that because bank staff had little experience with the product.

- Product owner is IT business analyst lead manager: she works with SME and users.

- Scrum master is technical manager who does the code review, release management, network, and security. He serves as a resource to help the teams make appropriate system and component level design decisions during implementation. He defines and split use cases and features for the program backlog, and allocating respective items to the individual team for implementation.

- Team involves 3 SIT testers, an architecture leader from the vendor who establishes software architecture design support of upcoming user and business needs and helped developers when required. Subject matter experts: three domain experts each for processing, underwriting and closing who can evacuate doubts or give a rapid opinion as required by developers. Different users carried out this role during the development depending on the issue under review. A dedicated team for BRE was also provided to deal with all underwritings and A-Score models.

A daily "scrum" or standup meeting was held with all the stakeholders. Every day, developers answered three questions: 1) what have you done since yesterday; 2) what are you planning to do by tomorrow; and 3) Do you have any problems preventing you from accomplishing your goal. To satisfy quality assurance of development, unit test with capture screens was applied by developers and reviewed by business lead. Frequent delivery and test with every two weeks, a release was delivered to SIT environment with bug fixes and new features in product backlog. For fast testing, QTP was used as an automated testing tool. Once a month, the product was released to UAT and a meeting with steering committees was conducted to inform the status of bug fixes and project status. This is to ensure that the team and all stakeholders have reviewed the product and meet the expectation.

A. Iteration 1 (Application Registration)

Following the plan, a short first iteration (15 days) was designed (02/07/12 - 23/07/12) but it was taken 34 days to complete - 19 days later than planned. This iteration involves the first three columns in **Error! Reference source not found.**1 where users register a mortgage application with the barcode and confirm application creation. The data entry person will enter customer information from a hard copy document and use the citizen identification to interface with National Credit Bureau and obtain the credit score result which will be used later in underwriting calculation. User can check and search if the borrower or co-borrowers information exist in the core bank systems, if so retrieve all the information. Once completed, they can send to the supervisor whose role is to review and revise wrong/missing data given by their staff. Later, if required they can submit the work to another unit whose roles are to comment and record missing data or document for further analysis of sale sufficiency. They also need to follow up with sales or customers about the missing information.

The first integration of the GUI presentation layer with the bank systems was achieved as a working software delivery. However, there were two factors which affected this iteration timeline. Firstly, Bank of Thailand regulation states that the bank cannot keep customer's sensitive information e.g. name, surname and id outside of Thailand. As a consequence, a special HTML/Ajax control was developed which maps all sensitive information kept in Cloud and the bank. The control is developed and can be reusable on all screens. The bank utilized the 'Mapper' server using Java and Spring Framework developed in Stage 1 to keep the real customer information (see Fig. 2) and interface with the bank internal systems via MQ. Secondly, during the reflection workshop carried out at the end of the present iteration, different code conventions were specified and refactored to facilitate maintenance and readability of the code. As a result the iteration was finished 17/08/2012, 19 working days delay.

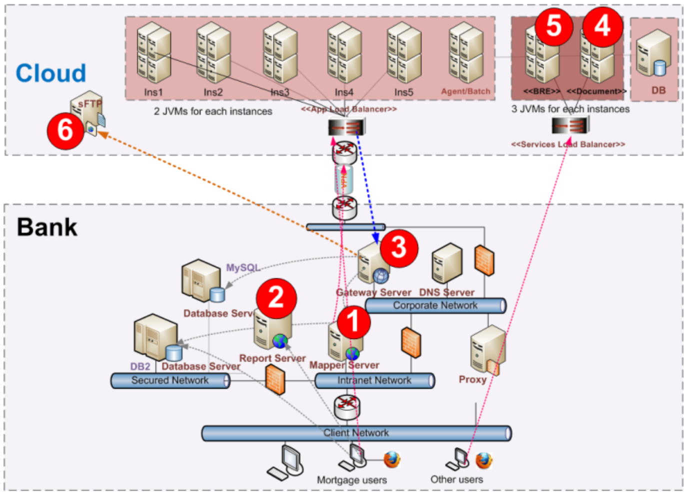


Fig. 2. Architecture of the System

B. Iterations 2 (Application Processing)

The second iteration started on 25/07/2012 to 03/09/2012 with an incremental on top of the previous iteration, developers enhanced features: 1) employee loan and employee search capability (edit, delete, and add) which can be reused in all screens. 2) Borrower search with edit, delete, add and online information retrieval of related loan and customer details resided in old mortgage application as data migration is not implemented and 3) validation for previous screen input fields were done.

To enhance business value, the iteration developed routing capability which is to deliver task to designed users and unit. In addition, all managers can track and store for key performance indicators (KPIs). So the manager can analyze process performance as well as create service level agreements (SLAs). However, a challenge of this iteration is that processing a mortgage loan, there were over 180 input fields in one page such as pricing plan detail, fee detail, insurance service, payback plan, guarantor, secure collateral types i.e. land and building, deed, condominium, debenture, bond and etc. These field values were highly inter-related and significant for the loan outcome. Therefore, developers required business knowledge, times for develop and unit test. On this iteration, prior to the delay 5 additional users came into help after office hour to test and identify missing requirements on the screen and validation rules and prevent further delay. Nevertheless, due to business complexity of all fields, the iteration faced 18 days delay and completed in 03/09/2012.

C. Iterations 3 (Underwriting and Closing)

This iteration was carried out from 23/08/12 to 13/11/12 (58 days). Two main processes were developed where underwriter officer approved the loan will be routed to loan closing unit. In this project underwriter officers' main tasks include: 1) calling the borrower about his/her income, wealth, credit history; 2) verifying borrower information with third parties such as social security department, other banks and employers via phone and online government website; and 3) approving or overriding the underwriting result (risk analysis, see Section 4.4). Loan closing process which is triggered automatically when underwriter approved. The loan application will be delivered to appraisal unit, once having an appraisal approved. The system initiated the process of finalizing documentations and printing between the bank and customer i.e. insurance, contract and cheque.

There were challenges in this iteration resulting in 48 days delay. With underwriting development (20 days delay), as data model was from the previous processing unit. However, the underwriters can add comments and adjust all fields resulting in 225 input fields appeared on screen (see Fig. 5). In addition, auto population of historical data (e.g. statements, debt, and insurance) via 7 interface bank systems was requested to reduce their time to key-in in one single page. Nonetheless, midway development underwriter users were informed on performance issues with HTML streaming and interface time (time > 3 seconds), so data grouping was proposed. UI was redesigned collaboratively with users where all input fields were groups into 3 tabs: customer (name and address, guarantor), finance (income, debt) and loan (mortgage term, rate, installment amount) information. Once, the user wants to view or edit then they can click each tab individually to improve loading time of less than 3 seconds. For interfacing issue, a manual click was proposed and accepted with seven interface buttons provided for users to enquire when needed.

With the closing state, there were resource and technical problems with the development of legal contracts. Not knowing before, the BPM tool the bank bought is not sufficient in generating documents. There were over 140 legal contracts in mortgage loan. So, legal contract users were trained to use iReport and designed the contracts which fulfill legal compliances such as font size, paragraph, margin and etc. Additional developers also were employed. Technically, loan data was kept in Cloud and real time data was required. So a dedicated web-service server was used to transfer loan data from Cloud to the on-premise report server (see **Error! Reference source not found.** circled numbers 2 and 4). It was also found out that legal contracts consumed JVM memory, as each loan requires 30 legal contracts at least. As a result, a dedicated server was provided with 2 JVMs inside. The iteration was once finished but during the iteration review, business executives requested a flow change (delete of appraisal manager), they were not cognizant of the impact these changes would have on the project budget or timeline, leading to significant tension across the project teams.

D. Iterations 4 (User Management and Change of Condition)

The fourth iteration started on 05/09/2012 and finished on 19/12/2012 19/12/12 taken 76 days. By then, the team understood many requirements of housing loan and in a continuous improvement, favoring a high team motivation. Another module was developed to handle user profile management, which defines allowed access rights, authority levels, skills, approval limits, department, views and outcomes by profile of different user groups. This is used to control startup features for the user's screen and session, the types of authorized approval limit and screen that s/he can operate.

The late part of business process is the 'change of condition'. The purpose of change of condition is to enable any information adjustment of the latest approved loan such as details of customer, finance and loan. The business process starts by allowing users to search by borrower or loan application number. System will return the latest approved loan application for the borrower or the loan. But specific complication arose as a 'change in requirement' when validation required recursive search whether all the related parties (borrower, borrower-spouse, authorize signature person, all the parties in collateral) in the latest application having any in-progress loan application within the bank. For example, a borrower may be a co-borrower or a guarantor in multiple loans. So, a change of borrower's name should not impact other loan. After retrieving all results in the background of the latest loan, all data will be auto-populate, users can register the change of condition loan process with minimum to key-in and validation is ensured. After, the loan operation reiterates the same steps as 4.1 – 4.3. Due to complexity of this recursive validation requirement, the project tried multiple ways to achieve the objective and stress tests to prevent future system failure. Hence, the iteration was delayed 76.

E. Iterations 5 (Risk Analysis and Interface)

The fifth iteration started on 16/07/2012 and finished on 18/03/2013 taken 176 day. However, every two weeks some rules and interfaces were delivered in accordance with the previous iteration sprints. Significant usages of BRE are used to empower business users to quickly create and manage underwriting rules with minimal involvement from IT staff. To facilitate the mortgage underwriting process, reduce costs, and promote consistency for all loans, ''credit scoring'' models have been developed that numerically weigh or ''score'' some or all of the factors considered in the underwriting process and provide an indication of the relative risk posed by each application (see TABLE III). At the time, the bank's mortgage underwriting rules had over 2,600 rules. So, four dedicated developers and three business users involved in the underwriting and analytic score development. BRE and its complex calculations were also involved such as loan to value, loan term, loan history, delinquent and etc. During the development, we also found with many accounts and debt history can result in a long processing. So, a dedicated BRE server was employed (see Fig. 2 circled number 5).

In the loan process, there were needs for real time online integration with 13 external systems and databases such as credit bureau, deposit, debt, fraud and so on. The application controlled the response sent in turn to a response received from bank systems via web service. Information was mapped directly to loan application properties or parsed and transformed. The application servers served as the endpoint for an external connection—as a means to provide data to other systems (see Fig. 2). Additionally, at night time, for non real time data mortgage system sent/received file integration via SFTP with external systems e.g. enterprise data warehouse (EDW), insurance, human resource, anti-money laundry and etc. Time was also a constraint as users finished work at midnight only 6 hours permit to complete file transfer. As there were many loan applications only new and updated loans information were extracted and sent to the external systems. A dedicated agent node and SFTP was utilized (see Fig. 2 circled number 6).

TABLE III. PARTIAL EXAMPLE OF APPLICATION SCORE MODEL

Mortgage and characteristic	**Credit score**							
	Mean	Median	Low		Medium		High	
			% of characteristic	% of score range	% of characteristic	% of score range	% of characteristic	% of score range
Loan-to-value ratio (%)								
< 81	817	842	3.8	86.8	8.2	85.5	88.1	86.8
81 to 90	801	821	3.7	12.6	8.8	13.6	87.5	12.6
> 90	770	782	3.4	0.6	12.8	1	83.9	0.6
Loan size								
2 M	814	840	4.4	44.2	8.3	37.7	87.3	37.5
2 M –5 M	812	836	3.9	39.1	8.7	39.9	87.5	37.8
> 5 M	819	840	2.6	16.8	7.6	22.4	89.8	24.7
Location characteristic								
ZIP code								
< 80	788	811	5.5	13.4	12.6	13.9	81.9	8.5
80 to 120	811	836	4.2	52.9	8.6	50.1	87.2	47.5
> 120	824	847	2.9	33.6	6.9	36	90.1	44.1
….	….	….	….	….	….	….	….	….

The SFTP file integration with other systems was executed last, as these was between system to system and the assumption that business application needed to be processed correctly first. However, during the test, it was continuously detected by related systems that our interface data caused their systems to collapse regularly. For examples with the insurance system, firstly, date of birth need to be in Christian year format "yyyy-mm-dd". So, the software needed to convert the Thai data before transferring.

Secondly, insured company document identification number needed to be 13 digits, so there was a need for screen formatter and validation to ensure users have entered data correctly. Thirdly, KYC Level the system needed to set a default level value at 100 (low risk). Fourthly, in the case of selecting a life insurance of a particular company, we needed to set a sum insured to be greater than 0. These file integrations significantly impacted the project timeline, as meetings and agreements with IT and business owners were required to agree upon the inputs and signed-off. These systems have some regulations that they have to comply and cannot change. As a result of these new findings, various parts of the software were revised e.g. data format and types, business rules and screens. SIT, UAT and regression tests were done resulting in 124 days delay in total.

V. EVALUATIONS OF SCRUM AND MORTGAGE APPLICATION

The research aimed to answer five areas: 1) project performance with schedule and cost; 2) management acceptance; 3) customer relationship; 4) team satisfaction and 5) usability acceptance.

For project performance, TABLE IV shows actual effort and duration, the last column highlighted differences from the plan. The actual efforts were over two thousands man hours (33%) more than estimated and 200 days delayed resulting in 300,000 USD over budget. Therefore, top senior management decided that in the next phase a turn-key project managed by a vendor will be utilized in order to manage the cost. Agile will no longer be favored.

TABLE IV. ACTUAL EFFORT AND DURATION

Process	Plan (hr)	Duration (day)	Act. Work (hr)	Act. Duration (day)	Diff (hr)
Application Registration	1,332	16	420	35	912
Application Processing	720	12	774	29	-54
Underwriting and Closing	480	11	4,164	59	-3,684
User Management and Change of Condition	2,256	32	790	76	1,477
Risk Analysis and Interface	2,736	52	3,885	176	-1,149
Total	7,524	123	10,033	375	-2509

A week after the production date (3/06/2013), assessment to find customer relationship and team satisfaction with Agile, an interview with project sponsor, was conducted. She was happy with Agile as its approach promoted teamwork, facilitated the deliveries of periodic working software. However, she was disappointed with the control over costs. From working team (PM, Dev, BA, SME, testers) points of views towards the project (with writing comments on sticky note colored in green and red to represent 'good' and 'no good'), the result shows that developers liked the Agile approach, new tool and office environment. They felt the team was congenial (outsources and in-house developers) and were comfortable on working and understanding with each other despite language challenges. Developers felt SME and business leads are open to discussions, able to explain and clarify queries on existing business processes. However, the team shared similar negative opinions: 1) high resource turnover, resulting in substantial time and effort spent on orientation and initiation of new resources; 2) There are many situations where changes to requirements were made without analysis and approvals from stakeholders "*Better utilization of time and effort could have been achieved if there was a more comprehensive process in assessing suitable resources*". "*Change management process needs further refining and governance*"; 3) Project status updates in weekly meetings with senior management of outsource companies were often postponed.

In terms of interface usability evaluation, fourteen participants, from all units, 7 males and 7 females, voluntarily participated. Their ages ranged from 22 to 31 years with a mean age of 25 (std. = 0.48). LOS was given an overall usability of 58% which is considered above average. The other factors met the standard requirement of usability scales (see **Error! Reference source not found.**). For each scale, the median value is shown circled in the middle of the line; the 95% confidence levels are shown by the opening and closed points. These limits mean that we can be 95% certain that true scale median for the software can be found. LOS made the circles over 50% line, except for the Efficiency scale showing that users felt the software and navigation were complicated.

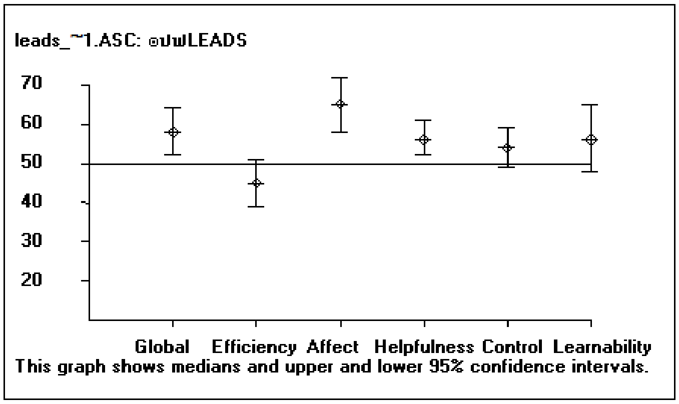


Fig. 3. SUMI Result

The main conclusion for each of the Sub-scales is summarized in TABLE V.

TABLE V. SUMMARY OF SUMI RESULTS

Subscale	Main Results
Efficiency	LOS was complicated to navigate. It required too many interactions (text inputs, buttons and conditions) to achieve an intended task. The software is robust and sufficient to work in a network environment. Even though, the software did take the issues of sensitive data into account, many of the save buttons were too many. The users need to save of sensitive data for any modification because of Cloud.
Affect	The users were satisfied with working with this software and did not feel tense while using it. Still, the presentation needs to be improved.
Learnability	The interface is informative; most functions of menu and buttons represented what it did quite clearly except for the sensitive data.
Control	The software was fast and robust. The user could move from one part to another fairly easily. However, there were too many clicks and keystroke and the user felt they were not in control.
Helpfulness	The help file was informative but some texts were difficult for the staff. The error and software messages were adequate. Each screen had its own help presentation.

From the SUMI questionnaire, LOS was usable (> 50). Overall end users had no problem in operating the software. This indicated that interface's factors such as clearly seen buttons and layout of UI elements received positive feedback from the end users.

VI. Discussion and Conclusion

This paper aims to contribute with an understanding of agile development failure in a large scale project, by identifying learning lessons which may contribute to other financial systems and other complex domains. Importantly, this study shows that Agile and the project was not failed, but due to political pressure on reducing the effort by senior management (Stage 3) which aligned with previous studies [7] [8]. Currently, Agile is not adopted by the bank as a result of effort, timeline and cost overruns. Waterfall model is currently employed. However, TABLE VI shows that if using the initial estimated (last column); the total project effort was in a safe zone (over estimated by 1447 man-hours or 12%). Indeed, agile should have been adopted, if efforts were not determined by senior management.

TABLE VI. Comparison of Actual and Initial Planning

Process	Political	Actual	Team plan
1. Application Registration	1,332	420	440
2 Application Processing	720	774	380
3 Underwriting and Closing	480	4,164	3,296
4 User Management and Change of Condition	2,256	790	502
5 Risk Analysis and Interface	2,736	3,885	6,862
Total	7,524	10,033	11,480

Even though, one of the main principles of agile methods is to "welcome changing requirements" [4] [7] [12] however this research showed that changing requirements especially with technically complicated challenge (Iteration 4) can contribute to extended timeline and cost [9]. The project appeared to have timeline and budget is strictly determined by senior management, then it implies that working over-time is mandatory. In this project, developers worked over-time on a regular basis to meet the political forced timeline. Their efforts were mostly un-clocked and un-billed to help the project and team. Therefore, the actual hours of over-time were not recorded. Consequently, six outsources and two local staff resigned adding problems to the project.

Rigid timeline increased pressure, when timeline is restricted, poor planning and analysis of related interface systems (Iteration 5) were not focused causing re-works significantly and miss of timeline for the project. Future empirical research is needed to investigate under which related interface systems and their messages should be reconciled in terms of their required fields to avoid reworks and delay of project. For example, if a middle named is a required field in a customer information system, then the field is a required field in mortgage system.

As reported in the PMKBOK [4] lack of fulltime SME staff was an important side effect of the detected problems (Stage 1), leading to project delays and failure. Besides, the previously mentioned problems potentially linked to political force and project management, there were additional difficulties during software development. At the beginning of Stage 3 of our project, the local team was novice. Employing new technology in any project implies certain inherent risks. Although, a training period to use BPM/BRE tool was carried out in Stage 2, it resulted insufficient; since there were areas of the tool to serve complicated mortgage requirements. Special care was employed by *side-by-side programming* practice between local and outsource developers. This turns beneficial to outsource as well for handling complex nature of mortgage domain where multiple disciplines interact [1] and specific Thai mortgage rules. It is therefore recommended that local and foreign staff sitting together.

Efficient communication is one of the key issues of Agile [3] and frequent delivery of tested working software in an iterative way brought high visibility of project progress [10]. The email update of bug fixes and project status update for every two weeks too all stakeholders give direct feedback to development amelioration. This context helped to share the 'big picture' of the project state and to build a strong camaraderie and team spirit, which definitely were the key drivers to sustain focus and commitment during all stages.

Although, the SUMI score was adequate, a range of interface problems was uncovered. Firstly, the major problem arising from the SUMI analysis related to the 'Efficiency' scale (46%). It was found that the users were required to key-in substantial data in one single page, check data synchronization between tabs and sections in order to complete the procedure. For example, firstly, in financial data section, users were not allowed to step next if they had not completed typing in eight tabs of the account. Secondly, the application instructed the user to save sensitive information in many places (see Fig. 6). In the next version of LOS in terms of Efficiency scale, therefore the software will minimize the number of save button. Also, collapsible section will be used (see Fig. 7) without displaying all fields[1]. This may reduce the user perception that they have to key-in all.

One of the limitations of this research study was the constitution of the sample i.e. mortgage application specific. Nonetheless, mortgage was only part of the activities of this research in agile development where an institute attempted to adopt Agile. BPM/BRE tool used is proprietary where line of code cannot be counted to assess software size. Additionally, from the authors' knowledge, the BPM/BRE cloud-based loan origination system is the first experience in the world. So in terms of Koch Agile evaluation [10] there is no benchmarking available to compare in terms of speed of delivery, software size, performance, robustness or adaptation of Agile in banking industry. Therefore, the results might not generalize to other agile development, particularly those in different culture or free of political influences.

While the need for agile approach has been widely recognized, making an agile approach work in a long established waterfall bank is challenging. A number of recommendations can be drawn; firstly even a political pressure on the effort estimated by the developers, still in the end the actual effort reflects the early estimate. Secondly, inexperience developers may not be able to design a core engine of BPM/BRE for the bank. Tool may be needed.

[1] At the time of this writing, it has been implemented as a result of SUMI and interviews where users requested for screen redesigned changes

Thirdly, constantly changing requirements increases difficulties and workload during development where timeline cannot be changed producing a significant dissatisfaction from developers and the sponsor. The research revealed several paradox-like phenomena that need further research and investigation. The agile method was found feasible in a large project; within the team Scrum was highly effective in rescuing this mortgage web-based project. The use of Scrum resulted highly positive at working team since it improved the communication between team members and as a consequence increases the team flexibility and productivity and maintaining focus on those tasks more relevant to the project.

ACKNOWLEDGMENT

The author is very grateful to Anders Markvardsen for continued reviews of this paper.

REFERENCES

[1] K.Temkin, D. Levy and D. Levine, "A Case Study of the Mortgage Application Process”," in *Mortgage Lending Discrimination: A Review of the Evidence*, Washington DC,, Urban Institute Press, 1999, pp. 137-160.

[2] R. Avery, R. Bostic and P. Calem, "Credit Risk, Credit Scoring, and the Performance of Home Mortgages," Federal Reserve Bulletin July 1996, Washington, 1996.

[3] L. Williams, "What Agile Teams Think of Agile Principles," *Communications of the ACM,* vol. 55, no. 4, pp. 71-76, 2012.

[4] B. Jordan, J. Giboney, M. Keith, J. Mark, D. Wilson, R. Schuetzler, P. Lowry and A. Vance, "Overview and Guidance on Agile Development in Large Organizations," *Communications of the Assosiation for Information Systems,* pp. 25-44, July 2011

[5] H. Mohammad, T. Alwada and J. Ababneh, "Agile Software Methodologies: Strength and Weakness," *International Journal of Engineering Science and Technology,* vol. 5, no. 3, pp. 455-459, 2013

[6] D. Batra, W. Xia, D. VanderMeer and K. Dutta, "Balancing Agile and Structured Development Approaches to Successfully Manage Large Distributed Software Projects: A Case Study from the Cruise Line Industry," *Communications of the Association for Information Systems,* vol. 27, no. 21, pp. 378-394, 2010.

[7] J. Wan, Y.Zhu and M. Zeng, "Case Study on Critical Success Factors of Running Scrum," *Journal of Software Engineering and Applications,,* vol. 6, pp. 59-64, 2013.

[8] M. Santos, P. Bermejo, M. Oliveira, A. Tonelli and E. Seidel, "Improving The Managment of Cost and Scope in Software Projects using Agile Practices," *International Journal of Computer Science & Information Technology,* vol. 5, no. 1, pp. 47-64, 2013

[9] S. Keaveney and K. Conboy, "Cost estimation in Agile Development Projects," in *ECIS*, 2006.

[10] S. Kruba, S. Baynes and R. Hyer, "BPM, Agile, and Virtualization Combine to Create Effective Solutions," *International Journal of Advanced Computer Science and Applications,* 2012.

[11] K. G. Royce, "Integration of a Business Rules Engine to Manage Frequently Changing Workflow: A Case Study of Insurance Underwriting Workflow," in *Americas Conference on Information Systems*, Colorado, 2007.

[12] A. S. Koch, Agile Software Development Evaluating the Methods for Your Organization, Boston: Artech House, 2005.

[13] T. Silva, M. Silveira, F. Maurer and T. Hellmann, "User Experience Design and Agile Development: From Theory to Practice," *Journal of Software Engineering and Applications,* vol. 5, pp. 743-751, 2012.

[14] A. Narasimhadevara, T. Radhakrishnan, B. Leung and R. Jayakumar, "On Designing a Usable Interactive System to Support Transplant Nursing," *Journal of Biomedical Informatics,* vol. 42, p. 137–151, 2008.

[15] G. G. Angel, PMP Certification A Beginner’s Guide, New York: McGraw-Hill, 2010.

APPENDIX

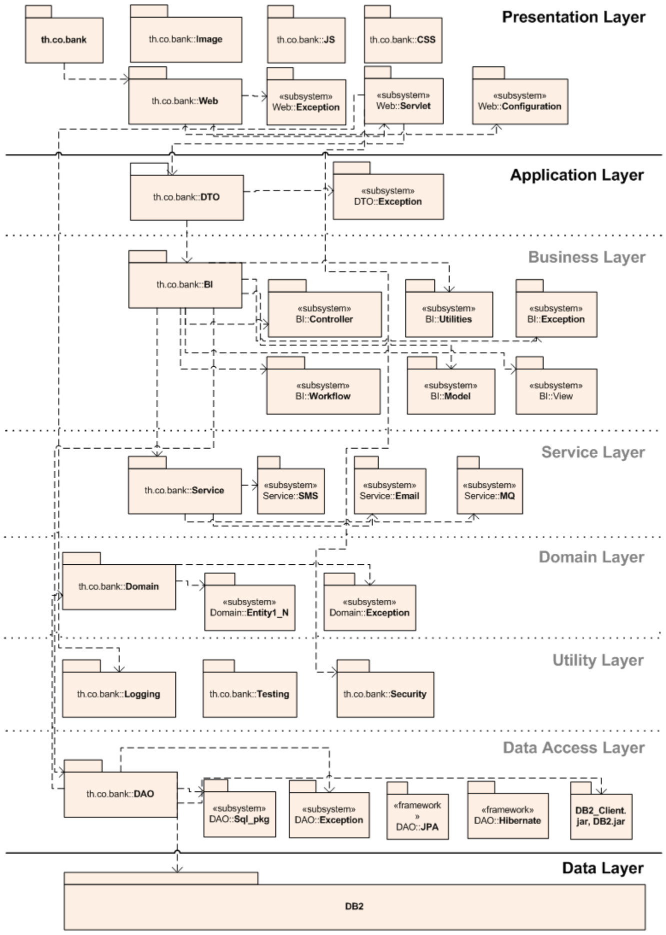
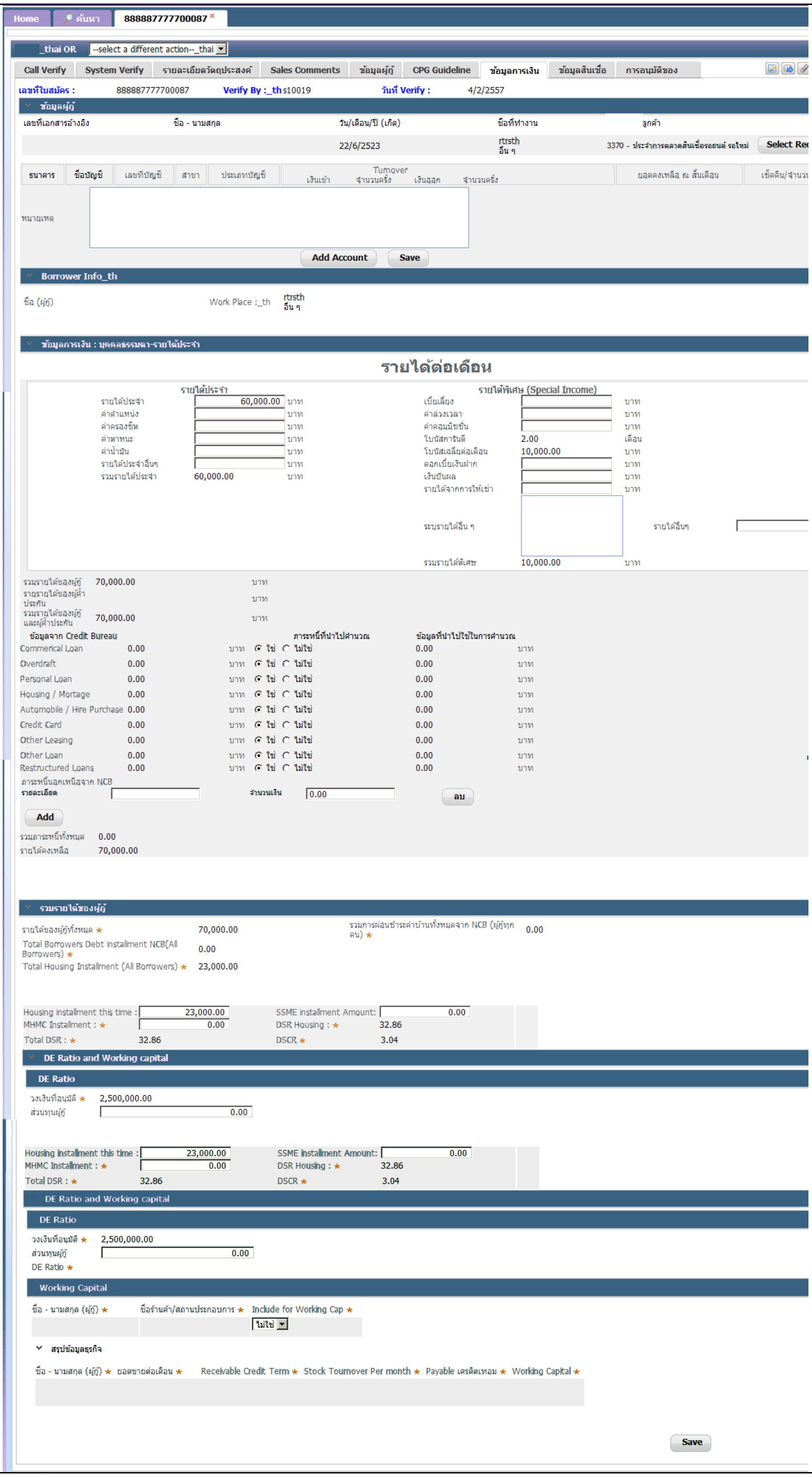


Fig. 4. Three Tier Architecture of Mortgage Application



เลขที่ใบสมัคร : 888887777700087 Verify By :_th s10019 วันที่ Verify : 4/2/2557

ข้อมูลผู้กู้

เลขที่เอกสารอ้างอิง	ชื่อ - นามสกุล	วัน/เดือน/ปี (เกิด)	ชื่อที่ทำงาน	ลูกค้า	
		22/6/2523	rtrsth อื่น ๆ	3370 - ประจำการตลาดสินเชื่อรถยนต์ รถใหม่	Select Re

ธนาคาร	ชื่อบัญชี	เลขที่บัญชี	สาขา	ประเภทบัญชี	Turnover เงินเข้า	 จำนวนครั้ง	 เงินออก	 จำนวนครั้ง	ยอดคงเหลือ ณ สิ้นเดือน	เช็คคืน/จำนว

หมายเหตุ

Add Account Save

Borrower Info_th

ชื่อ (ผู้กู้) Work Place :_th rtrsth อื่น ๆ

ข้อมูลการเงิน : บุคคลธรรมดา-รายได้ประจำ

รายได้ต่อเดือน

รายได้ประจำ			รายได้พิเศษ (Special Income)		
รายได้ประจำ	60,000.00	บาท	เบี้ยเลี้ยง		บาท
ค่าตำแหน่ง		บาท	ค่าล่วงเวลา		บาท
ค่าครองชีพ		บาท	ค่าคอมมิชชั่น		บาท
ค่าพาหนะ		บาท	โบนัสการันตี	2.00	เดือน
ค่าน้ำมัน		บาท	โบนัสเฉลี่ยต่อเดือน	10,000.00	บาท
รายได้ประจำอื่นๆ		บาท	ดอกเบี้ยเงินฝาก		บาท
รวมรายได้ประจำ	60,000.00	บาท	เงินปันผล		บาท
			รายได้จากการให้เช่า		บาท
			ระบุรายได้อื่น ๆ		รายได้อื่นๆ
			รวมรายได้พิเศษ	10,000.00	บาท

รวมรายได้ของผู้กู้	70,000.00	บาท
รายรายได้ของผู้ค้ำประกัน		บาท
รวมรายได้ของผู้กู้และผู้ค้ำประกัน	70,000.00	บาท

ข้อมูลจาก Credit Bureau			ภาระหนี้ที่นำไปคำนวณ	ข้อมูลที่นำไปใช้ในการคำนวณ	
Commerical Loan	0.00	บาท	● ใช่ ○ ไม่ใช่	0.00	บาท
Overdraft	0.00	บาท	● ใช่ ○ ไม่ใช่	0.00	บาท
Personal Loan	0.00	บาท	● ใช่ ○ ไม่ใช่	0.00	บาท
Housing / Mortage	0.00	บาท	● ใช่ ○ ไม่ใช่	0.00	บาท
Automobile / Hire Purchase	0.00	บาท	● ใช่ ○ ไม่ใช่	0.00	บาท
Credit Card	0.00	บาท	● ใช่ ○ ไม่ใช่	0.00	บาท
Other Leasing	0.00	บาท	● ใช่ ○ ไม่ใช่	0.00	บาท
Other Loan	0.00	บาท	● ใช่ ○ ไม่ใช่	0.00	บาท
Restructured Loans	0.00	บาท	● ใช่ ○ ไม่ใช่	0.00	บาท

ภาระหนี้นอกเหนือจาก NCB

รายละเอียด จำนวนเงิน 0.00 ลบ

Add

รวมภาระหนี้ทั้งหมด 0.00

รายได้คงเหลือ 70,000.00

รวมรายได้ของผู้กู้

รายได้ของผู้กู้ทั้งหมด *	70,000.00	รวมการผ่อนชำระค่าบ้านทั้งหมดจาก NCB (ผู้กู้ทุกคน) *	0.00
Total Borrowers Debt installment NCB(All Borrowers) *	0.00		
Total Housing Installment (All Borrowers) *	23,000.00		

Housing installment this time :	23,000.00	SSME installment Amount:	0.00
MHMC Installment : *	0.00	DSR Housing : *	32.86
Total DSR : *	32.86	DSCR *	3.04

DE Ratio and Working capital

DE Ratio

วงเงินที่อนุมัติ * 2,500,000.00

ส่วนทุนผู้กู้ 0.00

Housing installment this time :	23,000.00	SSME installment Amount:	0.00
MHMC Installment : *	0.00	DSR Housing : *	32.86
Total DSR : *	32.86	DSCR *	3.04

DE Ratio and Working capital

DE Ratio

วงเงินที่อนุมัติ * 2,500,000.00

ส่วนทุนผู้กู้ 0.00

DE Ratio *

Working Capital

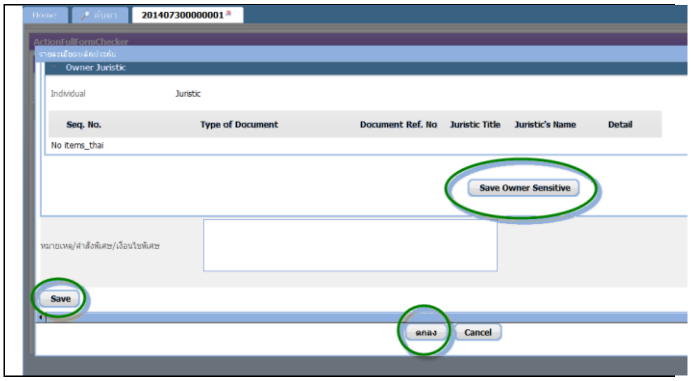
ชื่อ - นามสกุล (ผู้กู้) *	ชื่อร้านค้า/สถานประกอบการ *	Include for Working Cap *
		ไม่ใช่

สรุปข้อมูลธุรกิจ

ชื่อ - นามสกุล (ผู้กู้) *	ยอดขายต่อเดือน *	Receivable Credit Term *	Stock Tournover Per month *	Payable เครดิตเทอม *	Working Capital *

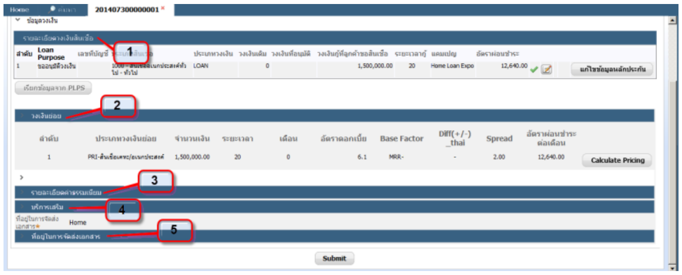
Save

Fig. 5. Screen Example



Seq. No.	Type of Document	Document Ref. No	Juristic Title	Juristic's Name	Detail
No items_thai					

Fig. 6. Save Buttons



ลำดับ	Loan Purpose	เลขที่บัญชี	ประเภทสินเชื่อ	ประเภทวงเงิน	วงเงินเดิม	วงเงินที่อนุมัติ	วงเงินกู้ที่ลูกค้าขอสินเชื่อ	ระยะเวลากู้	แคมเปญ	อัตราผ่อนชำระ
1	ขออนุมัติวงเงิน		1000 - สินเชื่ออเนกประสงค์ทั่วไป - ทั่วไป	LOAN	0		1,500,000.00	20	Home Loan Expo	12,640.00 ✓

ลำดับ	ประเภทวงเงินย่อย	จำนวนเงิน	ระยะเวลา	เดือน	อัตราดอกเบี้ย	Base Factor	Diff(+/-)_thai	Spread	อัตราผ่อนชำระต่อเดือน
1	PRI-สินเชื่อเคหะ/อเนกประสงค์	1,500,000.00	20	0	6.1	MRR-	-	2.00	12,640.00

Fig. 7. Collapsible Sections