World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:9, No:5, 2015

A Cost Effective Approach to Develop Mid-size Enterprise Software Adopted the Waterfall Model

M. N. Hasnine, M. K. H. Chayon, M. M. Rahman

***Abstract*—**Organizational tendencies towards computer-based information processing have been observed noticeably in the third-world countries. Many enterprises are taking major initiatives towards computerized working environment because of massive benefits of computer-based information processing. However, designing and developing information resource management software for small and mid-size enterprises under budget costs and strict deadline is always challenging for software engineers. Therefore, we introduced an approach to design mid-size enterprise software by using the Waterfall model, which is one of the SDLC (Software Development Life Cycles), in a cost effective way. To fulfill research objectives, in this study, we developed mid-sized enterprise software named "BSK Management System" that assists enterprise software clients with information resource management and perform complex organizational tasks. Waterfall model phases have been applied to ensure that all functions, user requirements, strategic goals, and objectives are met. In addition, Rich Picture, Structured English, and Data Dictionary have been implemented and investigated properly in engineering manner. Furthermore, an assessment survey with 20 participants has been conducted to investigate the usability and performance of the proposed software. The survey results indicated that our system featured simple interfaces, easy operation and maintenance, quick processing, and reliable and accurate transactions.

***Keywords*—**End-user Application Development, Enterprise Software Design, Information Resource Management, Usability.

I. Introduction

THE processes to develop computer-based information systems have been changed significantly in recent years [1]. As mentioned earlier, organizational tendencies towards ICT-based information processing have been observed noticeably in many third-world countries. In last decade, many small-sized and mid-sized companies have taken initiatives to switch to computerized working environment. Computer-based information management not only merits by saving time, but also helps the employee to perform complex tasks. Other benefits of computer-based information processing over manual-processing are record, organization, retrieval, display, dissemination, and acquisition of data. Due to these facts, end-user application development has become a dominant force in many organizations [2]. Therefore, numerous kinds of tools, techniques, and approaches have been tested in the development of enterprise software. However, limited budgets, a strict deadline, and inadequate IT-knowledge of the end-users remain the big challenge for IT analysts and software engineers.

System analysis and design are an exciting and active field where analysts continually learn new techniques and approaches to developing systems more effectively and efficiently [3]. Several approaches, methodologies, and techniques have been reported and tested in the process of software analysis. However, two widely used methodologies are: 1) Software Development Life Cycle (SDLC) and 2) Prototyping [4]. In IT project management, SDLC process applies to information system development to ensure that all functions, user requirements, strategic goals, and objectives are met. The SDLC provides a structured and standardized process for all phases of any system development effort. In this particular study, the Waterfall approach, which is one of SDLCs, has been considered as the software development methodology.

The rest of this article is structured as follows. A brief introduction to SDLCs and particularly Waterfall approach had been articulated in Section II. Section III covers the articles that have been reviewed. An introduction to the approach used to develop BSK Management System has been articulated in Section IV. Next Section V discusses on the technological specifications involved in this study and the outline of the system. The following Section VI concisely discusses on the usability test conducted to support this study including the procedures and results. Finally, Section VII summarizes the outcome of this study, and finally recommends some future works in Section VIII.

II. SDLCs & Waterfall Model

The SDLC models particularly in software development allow analysts to draw and design systems beyond traditional technological boundaries. In the development of enterprise software, the SDLC has drawn heavy attention for cost-efficient, effective, and high-quality product development. It can also be mentioned that a software lifecycle covers all the stages from its inception with requirement gathering through to maintenance. IT analysts heavily rely on SDLC methodologies to ensure that all functions, user requirements, strategic goals, and objectives are met. As stated earlier, the SDLC provides a structured and standardized process for all phases of any system

M. N. Hasnine is a full-time PhD student with the Graduate School of Engineering, Tokyo University of Agriculture and Technology, Nakacho 2-24-16, Koganei, Tokyo 184-8588, Japan (phone: +81-80-4919-4030; fax: +81-042-385-9747, e-mail: nehalhasnine@gmail.com, 50013646142@st.tuat.ac.jp).

M. K. H. Chayon is currently working with Millennium Certis Security Bangladesh Ltd., House#123, Block-E Road No 19/A. (e-mail: k.hasan@millennium-certis.com).

M. M. Rahman is working with the Kiwibank, 139 Kelburn Parade, Kelburn, Wellington 6012 (e-mail: mobasswer.rahman@kiwibank.co.nz).

M. N. Hasnine, M. K. H. Chayon & M. M. Rahman was with the Department of Computer Science and Engineering, Stamford University Bangladesh, 51 Shiddheswari Rd, Dhaka-1217, Bangladesh.

development effort.

SDLC models can be categorized as falling under three broad categories: 1) linear, 2) iterative, and 3) a combination of linear and iterative models [5]. Ruparelia also included that based on the existence and gained popularities, the Waterfall, B-Model, Incremental Model, V-Model, Spiral Model, Wheel-and-Spoke Model, and Unified Process Models are used widely.

The Waterfall model was firstly documented by Benington [6] and modified lately by Winston Royce [7] in 1970. This model has underpinned all other models since it created a firm foundation for requirements to be defined and analyzed prior to any design or development [5]. Phases in Waterfall model historically included the following with the key focus on parentheses: feasibility (readiness), analysis (what), design (how), detailed to design (how), coding and unit testing (technology), testing (correctness), and implementation (transition to operation) [8].

To our best understanding, the Waterfall model may be the best to use when developing an enterprise relational database. In addition, the Waterfall model provides benefits over other models by providing back-end functionalities, user friendly data representation, easy to implement, easy to manage due to rigidity of the model, less possibilities of phase overlapping, works well for small size projects, and cost effective. In addition, the comparative study by [9] indicates that the pros of this model dominate over other SDLC models. Therefore, we have considered the Waterfall model to design software. However, [10] scrutinized some of the common confusions that analysts often experience while working with the Waterfall model.

III. Literature Review

Research related to techniques, methodologies, and approaches has been well investigated and documented in the literature. A large number of comparative and empirical studies can be found in the literature. However, not many studies have been published for students to understand the processes of the Waterfall model. Therefore, this article targets software engineering students and IT analysts to understand the phases involved in the Waterfall model effectively.

At the beginning of our study, we have investigated the failures and risks involved in software development. Empirical study conducted by [11] investigated some major reasons behind information systems failure in developing countries. The investigation concluded that Design-Acuity gaps are the main reasons behind many information system failures in developing countries. Besides, various attributes (such as project control, ease of use, and communicability) associated with methodologies might cause full or partial system failure. In addition, the empirical study conducted by [12] revealed that a large gap observed between the definition of project success defined by software engineers and the popular definition in the society.

Chen [13] implemented a preliminary framework for classifying entity-relationship models and discussed how these models might be translated from one to the other.

However, to our best understanding, not significant amount of researches have been published yet which emphasized on the software design in cost effective ways. Therefore, in this study we introduced an approach to developing small-sized and/or mid-sized enterprise software at a budget cost. Our proposed system will also assist software engineering students, young software engineers, and IT analysts to understand the phases involved in the Waterfall model effectively. Moreover, our proposed method will help software engineers to understand how to develop software under a strict deadline.

IV. The Approach

This section briefly describes the approach followed to design the software.

A. In-a-Nutshell

A non-profitable organization located in Bangladesh had been selected to implement our proposed approach. The organization is involved in activities like to promote reading habits, enlightenment, and progressive ideas among students and the general public in Bangladesh. For the sake of confidentiality, this article does not reveal the company name and certain operations. In this study, the term 'BSK' is used as a pseudonym to represent the organization. It can be mentioned that the system design and implementation phases have been done based on the given information by the organization.

In this project, we have used the Waterfall model because this model allows analysts to draw and design systems beyond traditional technological boundaries. We have followed a generic Waterfall model where all steps are iterative. Fig. 1 shows the iterative feedback approach used to design our system.

Preliminary investigation and interviewing employees regarding opportunities and problems in the organization have been taken into account as *Requirement Gathering Phase*. In the *System Analysis Phase*, 4-main analyses which are 1) Problem 2) Requirement 3) Decision and 4) Feasibility had been analyzed. While into *System Implementation (Design) Phase*, technical architecture, system standard, and technical specifications have been put into action. *Coding*, *Testing*, *Conversion,* and *Maintenance* phases have been executed depending on the requirements of the user.

The proposed system purposed to assist current employees of the organization with their information resource management in a cost effective way.

At first, Data Flow Diagram (DFD) of three levels of the organization including Context Level DFD, System Level DFD, and Level-1 DFD have been designed and documented. Then we analyzed the Level-1 DFD into different subsystems. After that, we have drawn the Rich Picture depending on the organization's different activities. The Entity Relationship Diagram (ERD), the Data Dictionary based on various databases, Entity Life Cycle Diagram, and Structured English have been designed and documented respectively.

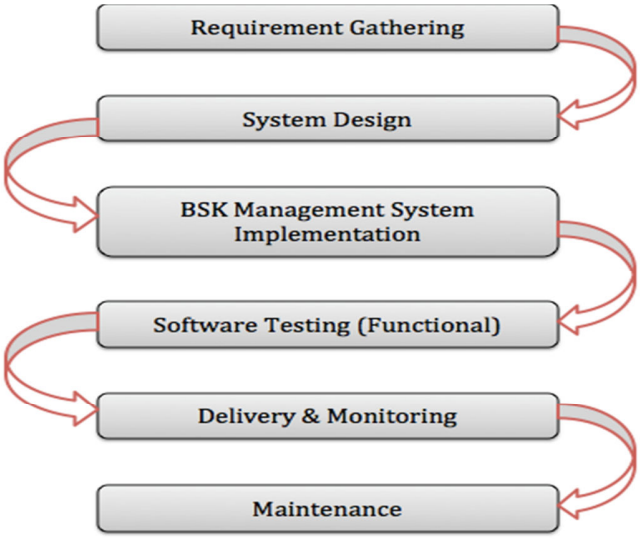

Fig. 1 Waterfall model with iterative feedback

B. Requirement Gathering

Requirement gathering (synonymously data collection) was the initial phase. The non-profitable organization we have developed the software for, is working actively with the motivation to enlighten people. The organization receives donation as a source of fund, to maintain their organizational expenses. The organization's operation can be mentioned as-student enrollment, the program progress, examination handling, donor handling, book purchasing, publications, and book supplying. To determine the functional requirements, employees in various positions have been interviewed. Received data have been well analyzed and our investigation concludes that the organization requires 9-different modules to run their basic operation.

As parts of non-functional requirement, 1) security requirements and 2) software quality attributes have been taken into account. The major considerations under Software Quality Attributes (SQA) were portability, reliability, availability, efficiency, safety, flexibility, correctness, testability, accuracy, error tolerance, maintainability, expandability, access control, and installation. Besides, some other non-functional attributes such as the platform, performance, backup, fallback, restart, and usability were taken into account while specifying the organizational requirements.

In this study, we have designed 4-modules to perform company's four major operations which are Library Management, Account Management, Examination Control, and Human Resource Management.

We have ensured the technical, economical, and operational feasibilities of the organization. To ensure technical feasibility, operational computer configurations and IT skills of the employee have been inspected. Our investigation found that as a non-profitable social awareness organization, the software maintenance cost might be a challenge for the organization to bear. Therefore, economically feasible software was taken into consideration. Finally, based on the existing systems and organizational framework, an operational feasibility has been ensured and reported to the project supervisor.

C. System Analysis & Design

DFDs have been drawn prior to system implementation. DFD is considered as a graphical representation to illustrate details of data flows and functions. Chun [14] articulated some characteristics of DFDs as 1) it supports analysis and requirement stages of system design, 2) diagramming technique with annotation, 3) describes a network of activities/processes of the target system, and 4) allows for behaviors of parallel and asynchronous; and 5) stepwise refinement through hierarchical decomposition of process.

Context Level DFD, System Level DFD, and Level-1 DFD have been drawn as a part of system analysis in BSK Management System. Figs. 2-4 display the Context Level DFD, the System Level DFD, and the Level-1 DFD of BSK Management System respectively.

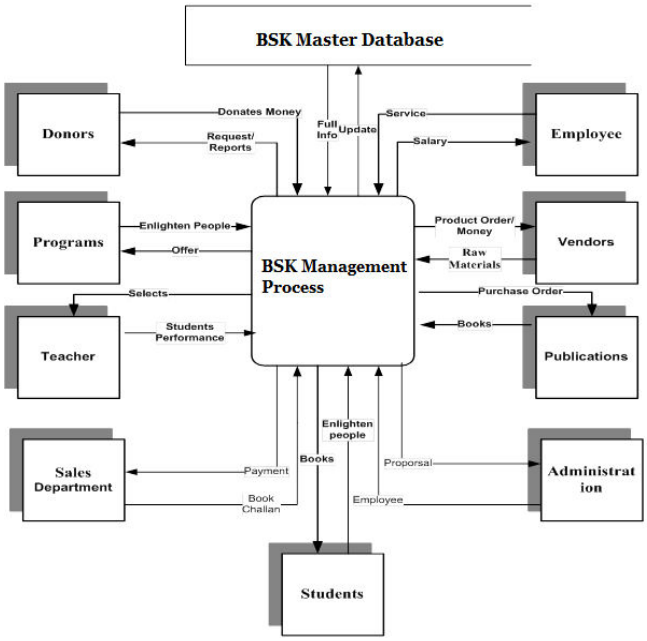

Fig. 2 Context Level DFD

In Figs. 2-4, shaded rectangles were used to represent external entities. External entities in BSK Management Systems can be defined as the departments will be allowed to give access. Processes have been represented by universal process icons. Processes of System Level and Level-1 DFDs are decomposed into a finer level for smooth processing. Databases have been represented with inventory icons. Data flows with an arrow head have been used to show the flow of the connectors.

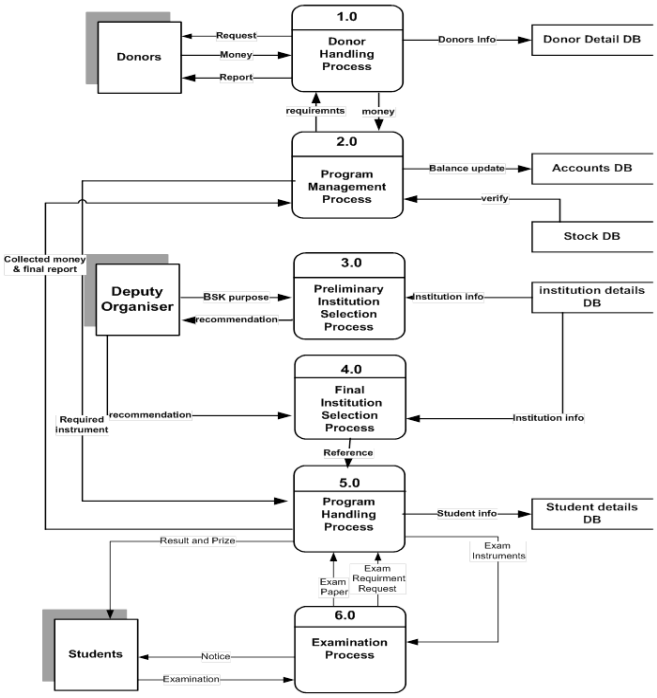

Fig. 3 System Level DFD

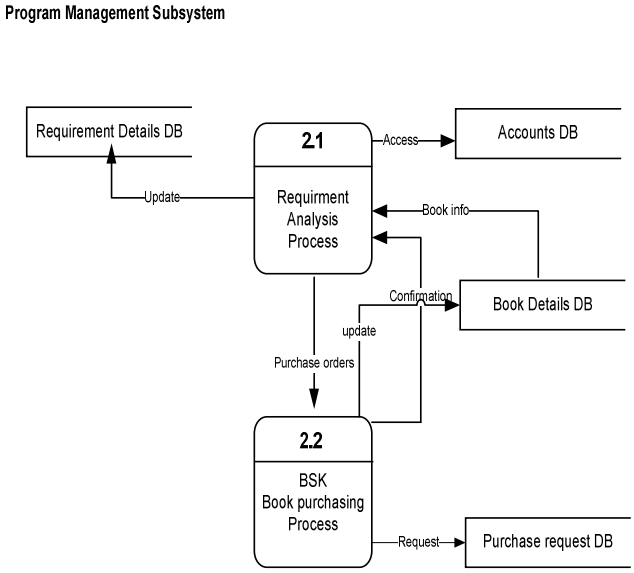

Fig. 4 (a) Level-1 DFD for Program Management Subsystem

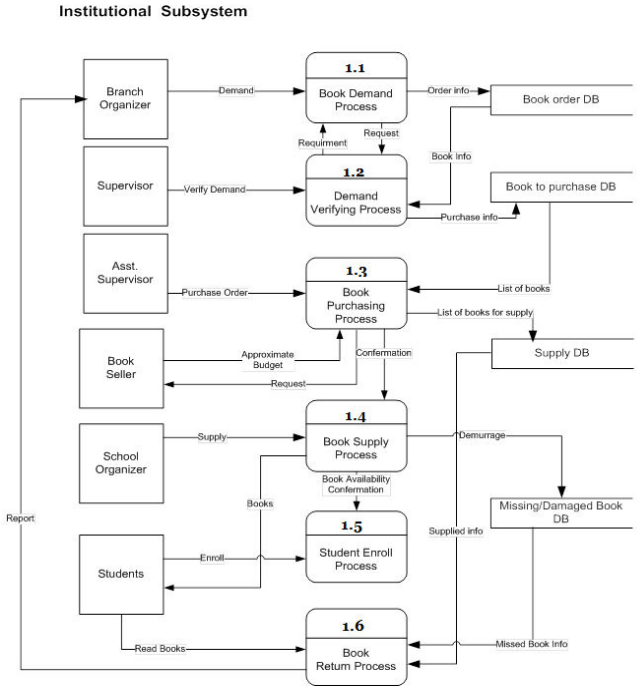

Fig. 4 (b) Level-1 DFD for Institutional Subsystem

D. Entity Lifecycle Diagram

Entity Life-Cycle Diagram is a graphical notation for Object-Oriented (OO) software development, which purposed to depict entity states and the possible state changes for entities of a single type [15]. In addition, Brain [16] argued that there is no object-oriented software model of life-cycle that has gained universal acceptance. Therefore, by considering the designing flexibility and project objectives, we have drawn Entity Life-Cycle Diagram to represent our software graphically. Fig. 5 shows the Entity Life-Cycle Diagram for BSK Management System.

E. Data Modeling

In modern days computing, Entity Relationship (ER) model is considered greatly for designing databases. The conceptual ER model consists of two major concepts: 1) Entities and 2) Relationships, which are the associations or interactions between several entities. Entities can be defined as real-life objects either animate or inanimate that can be easily identifiable and distinguishable. The ER diagram shown in Fig. 6 was designed and implemented for BSK Management System. Databases have been developed based on conceptual entity relationship diagrams and functional relationship. In Fig. 6, *rectangles*, *circles*, and *diamonds* represent entity types, attributes, and relationship types respectively. In addition, lines link attributes to entity types and entity types to relationship types. To describe binary relationships based on cardinality ratios, One-to-One (1->1), One-to-Many (1->m) and Many-to-Many (m->m) notations have been used. Logical relationships have been defined as well during database implementation.

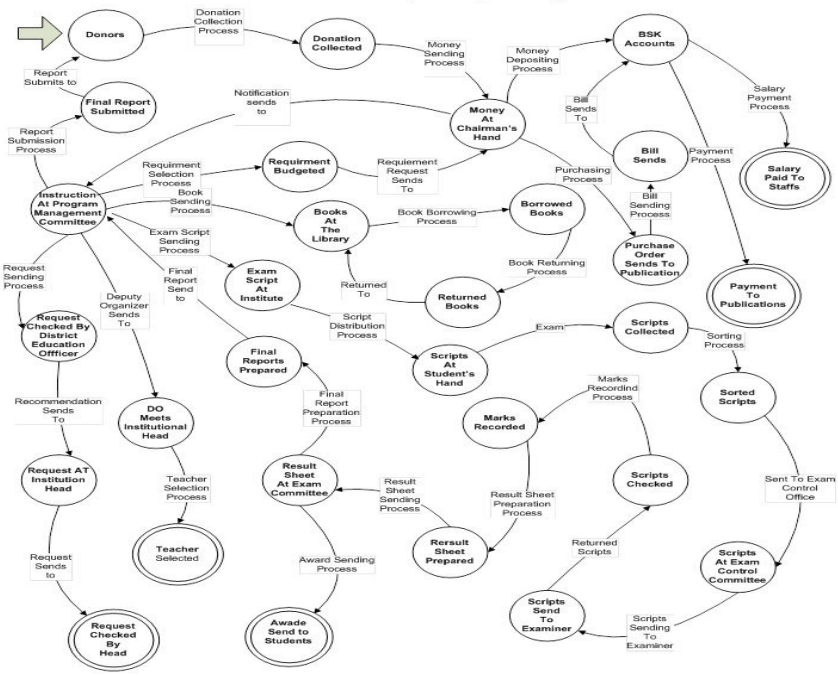

Fig. 5 Entity Lifecycle Diagram

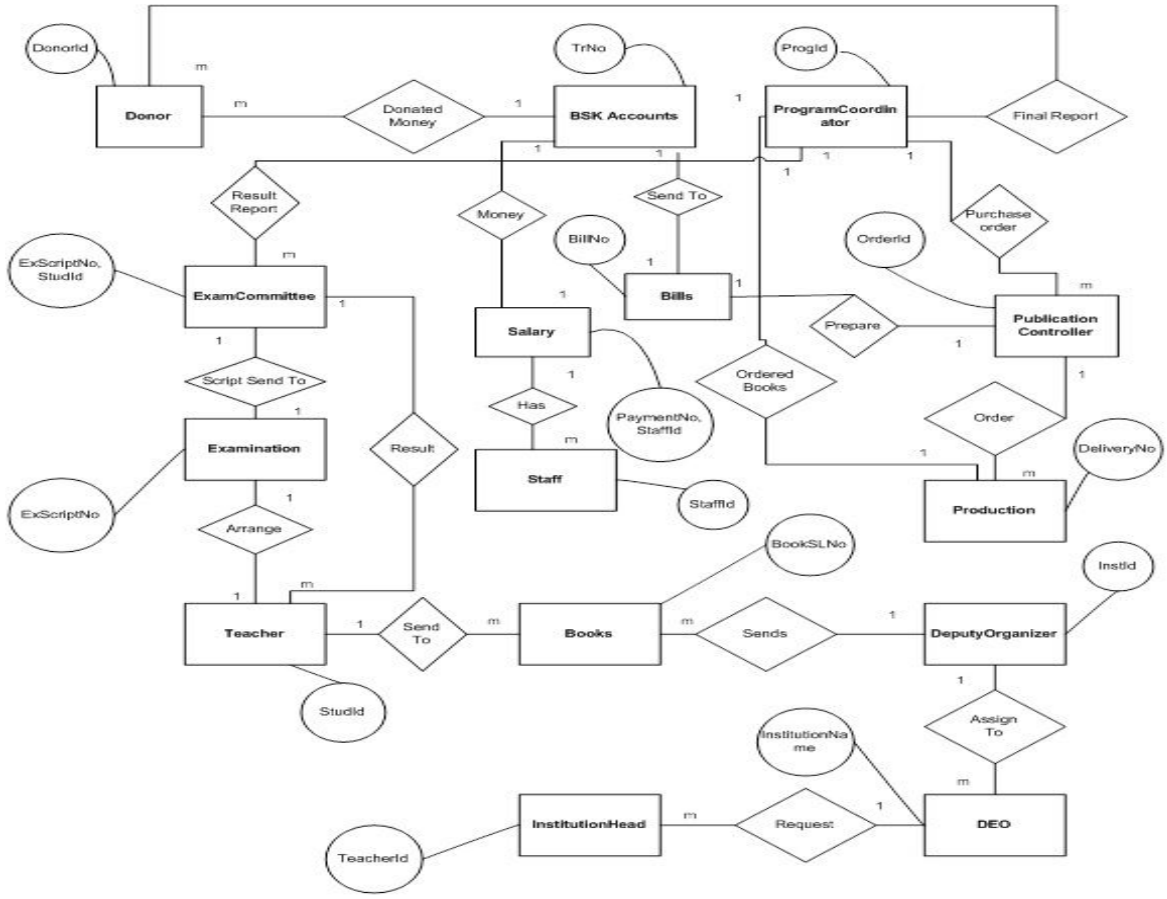

Fig. 6 Conceptual Entity Relationship Diagram

V. Software Implementation

This section briefly discusses on the implementation and the

operations of the BSK Management System.

A. Interface Design

In Human-Computer Interaction (HCI), user interfaces have traditionally been about designing effective and efficient systems. By considering users ability to perform computer operations, we have designed simple and easy-to-operate interfaces. Proposed software ensures the security and accessibility by password protection.

B. Technical Specification

We used Microsoft Office Visio 2007 for any sort of designing including DFD, Rich Picture, and Entity Relationship Diagram. To prepare Structured English we used Microsoft Office XP and for presentation, we used Microsoft PowerPoint.

To develop the final software for the organization, we used Visual Studio 6.0. We also used JAVA for few activities. We used Microsoft Access, My SQL, SQL, and Oracle for database maintenance.

C. Modules & Operations

As stated earlier, organizational behavior and tasks needed to perform (set by BSK employee) were the key considerations during software implementation. Due to that fact, the BSK Management System V 1.0.0 consists of 7-key operational buttons under each module. The button 1) 'Find' performs the search operations; 2) 'Add' performs basis addition operation to database; 3) 'Save' performs saving data operation that users might need for future references; 4) 'Delete' button erases existing data from the database; 5) 'Cancel' performs cancellation; 6) 'Print' button assists users in printing documents if the computer is connected to local or network printer; and 7) 'Close' is used to quit the current module and switch to other (if access allowed). In addition, other operational buttons were created based on modules' operations.

BSK Management System was developed with password protection policy. Only authorized users are allowed to access the system. To access to the modules, users need to insert accurate user id and password. Fig. 7 displays the login window that users will experience to access the functionalities.

Fig. 7 Login Window to Access BSK Management System

In this study, we report four modules that have been developed and delivered to the client. The four modules were designed to perform enterprise's operation under four departments, which are: 1) Library management, 2) Account management, 3) Examination control, and 4) Human resource management.

Library Management Module supported with functionalities to perform basic book search operation. Librarians and registered students were targeted users of this module. This module was designed such a way that students can order (purchase) a book online. However, the software does not function online payment. Fig. 8 displays a sub-module of library management module.

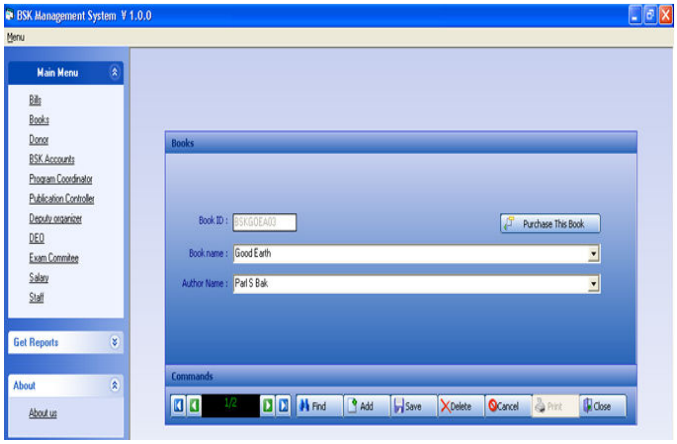

Fig. 8 Library Management Module

Account Management Module processes daily financial transactions taken place into the enterprise. Employers under the finance department were given access to the module. This module assists users to all finance-related works including automatic ledge update, create a new transaction, delete existing etc. Fig. 9 shows a snapshot of the sub-module of the account management module.

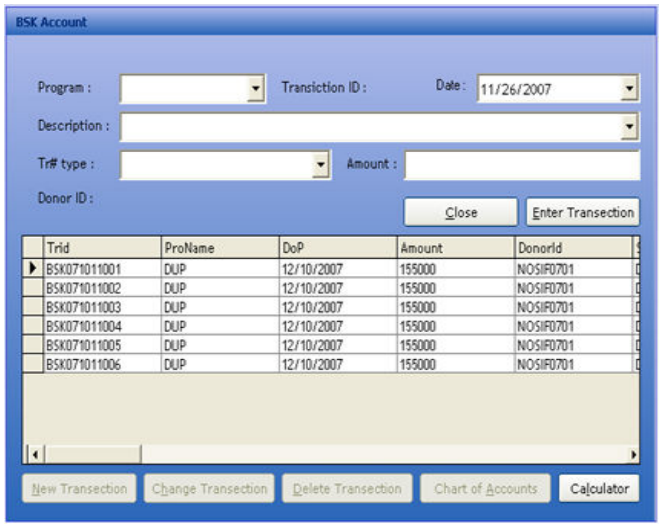

Fig. 9 Account Management Module

Examination Control Module performs tasks related to students' education. Teachers can create student profiles and analyze their performances by using this module. Teachers and people involved in teaching were main users of this module.

Fig. 10 shows working environment with this examination control module.

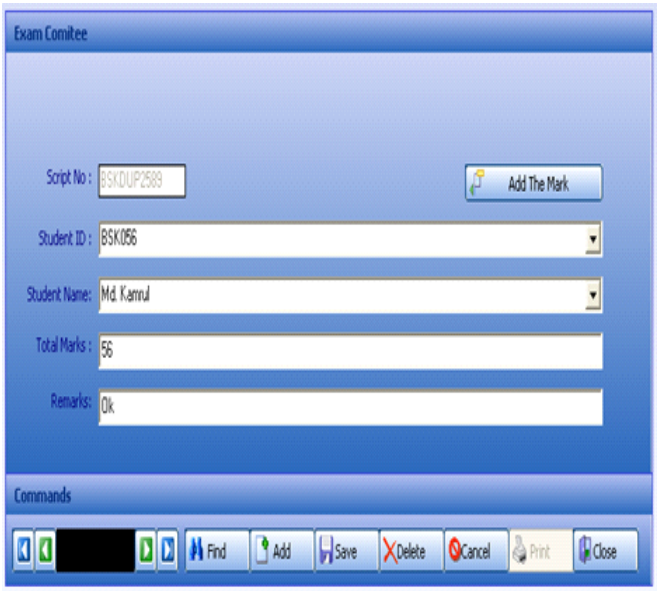

Fig. 10 Examination Control Module

Human Resource Module was developed to help employers under HR department to perform their tasks related to human affairs. Access to this module were set to HR staffs and people working in the top management of the organization. A snapshot of human resource module has been shown in Fig. 11.

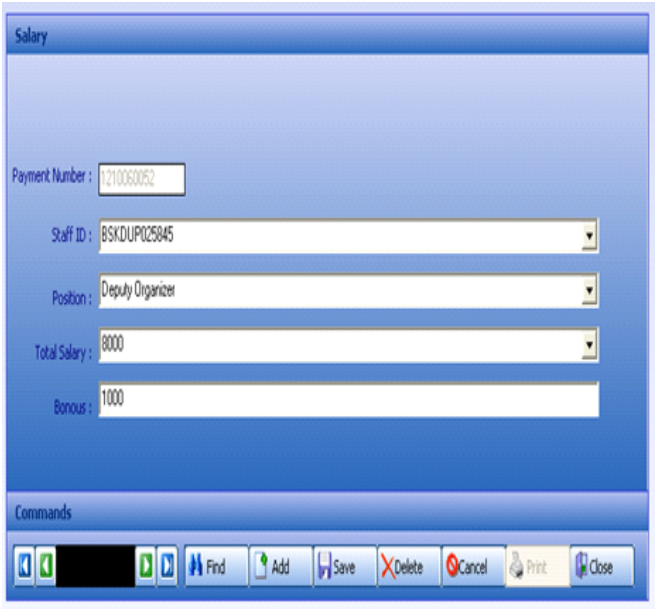

Fig. 11 Human Resource Module

An access to these modules was set based on client's demand. Another access was granted to the person who will be responsible to change privacy policies.

D. Structured English

Structured English QUEry Language (SEQUEL) is considered as a relational data sublanguage intended for ad hoc interactive problem solving by non-computer specialists [17]. Keeping end-users into consideration, a version of SEQUEL has been implemented and documented. Fig. 12 shows the SEQUEL for BSK Management System.

```
IF District Education Officer or Upozilla Education Officer Doesn't Accepts BSK Proposal
        THEN the BSK Programs of that particular District is been cancelled.
ELSE
IF the Institution Committee Permits the BSK Proposal
    IF the Institutional Head Selects a Responsible Teacher
        THEN BSK Program Management Committee sends Book and Required
        Instruments.
            IF the Student is an enrolled Student under BSK Program
                THEN Borrow one Book for One Week to Read
                    IF the Student is 16 Weeks Old Student
                        THEN the Student Sit for Performance Test
                            IF the Student is Performed Very Well
                                THEN Reward Student
                            ELSE No Prize will be given.
                        ELSE
                        DO UNTILL 16 Weeks Study Complete.
                ELSE don't Issue Book but Encourage the Student about the
                Program
            ELSE BSK does not send Books.
    ELSE try to convince the Committee and Waits for the Committee Decision
END IF
END
```

Fig. 12 SEQUEL of BSK Management System

E. Rich Picture

The Rich Pictures are generally constructed by interviewing people [18]. Rich pictures originated in the Soft System Methodology (SSM) [19]: Soft systems methodology (SSM) is an approach for tackling problematical, messy situations of all kinds. It is an action-oriented process of inquiry into problematic situations in which users learn their way from finding out about the situation, to taking action to improve it [20].

The importance of drawing a rich picture scenario was because of the assumption that, the newly developed system will affect the way people work. Fig. 13 shows the Rich Picture of BSK Management System.

As a non-profitable organization, donors are the key source of financial support in BSK. Therefore, we have designed rich picture from a donor. It can be mentioned that drawing rich pictures are the matter of debate. Depending on the type organization and type of activities, the rich picture might vary. In this study, we have tried to reflect the operation of the enterprise by drawing the rich picture (shown in Fig. 13).

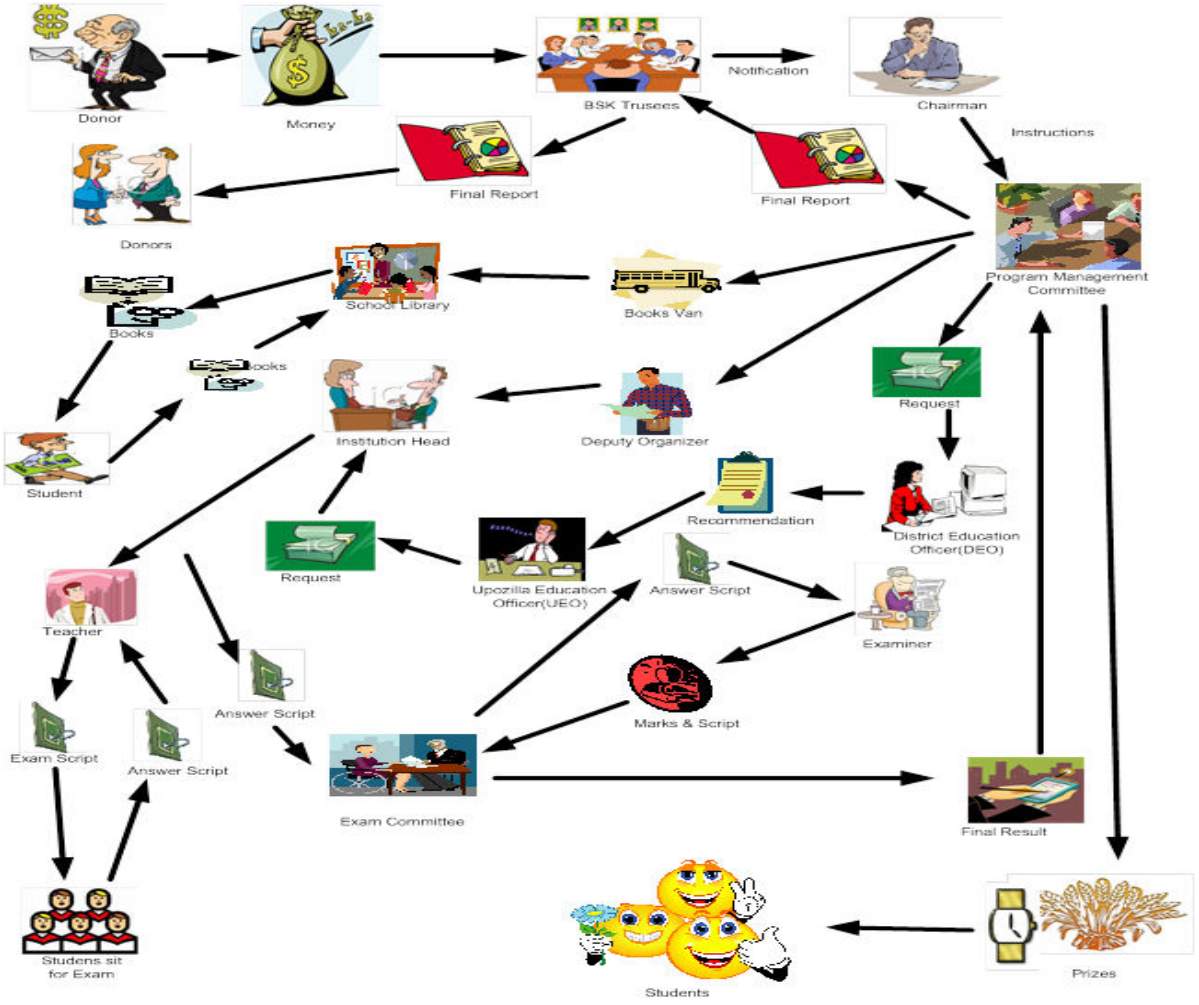

Fig. 13 Rich Picture of the Enterprise

F. Data Dictionary

Databases have been managed with active data dictionary. In Data Management System (DMS), Data Dictionary is an approach to represent a set of information describing the contents, format(s) and structure(s) of databases and the relationship between its elements. Werts [21] described step-by-step instructions and demonstrated how to customize planning so the installed dictionary meets an organization's specific needs. The article also reviewed basic data concepts and data-related problems, discussed diverse approaches to these problems, and described ideas and features of data dictionaries.

G. Testing

Our proposed software BSK Management System has been tested in engineering manner. During software testing, bugs have been removed and compatibilities have been ensured.

H. Documentation Aid

After found out all required information for our software we have recorded all the requirements accurately and made a reliable and complete documentation for the further improvements.

I. Delivery and Maintenance

The software was delivered to the client within the given timeframe. Software installation and proper guidance to access the system have been provided in engineering manner.

VI. Usability Test

We have conducted a survey to investigate the usability of our proposed system. We have also interviewed the company employees along with an assessment survey to ensure the accuracies and performances of the installed new software.

A. Participant Details:

10 employees and 10 students participated in the survey. The participated students were involved in voluntary social works and also involved in BSK's daily operations.

B. Materials & Method:

Participants were, at first, asked to perform some basis transactions on the newly installed software. Then, participants were asked to evaluate our software under 8 perspectives (shown in Fig. 14) based on their experiences. Each participant was asked to use our system for 30 minutes. The system's

operation and functionalities have been explained in advance.

After the 30-minutes session, we asked them to feedback their impression on the usability of the software. An assessment questionnaire with 8 different perspectives of software was provided to them. We also interviewed the participants about the software afterwards.

	Very Difficult to Operate	Difficult to Operate	Satisfactory	Very Good	Excellent
Efficiency					
Readability					
Navigability					
Accessibility					
Speed					
User Experience					
Contents					
Reliability					

Fig. 14 Questionnaire Used in Usability Test

C. Result

The survey result revealed that, all of the participants evaluated our system positively. The result indicated that, approximately 60% evaluation (8 perspectives & 20 participants) was found 'Excellent'. Approximately 30% feedback was noted as 'Very Good' and approximately 10% noted as 'Satisfactory'. None of the participants assessed our system as 'Difficult to Operate' and 'Very Difficult to Operate'. Therefore, we can conclude that our system features simple interfaces, easy operation and maintenance, quick processing, reliable and accurate operations. After interviewing the users of this software, we knew that they took the solution very frankly and they were satisfied with the software. Therefore, we can conclude that the employee will get everything they require by using this software.

VII. Discussion

ICT is a potentially powerful tool for extending business. Not only big-size enterprises are investing in ICT-based information management, but also many mid-size and small-size enterprises moving forward to it. The massive flow of ICT in organizations has been noticed significantly in third-world countries in recent decades. Enterprises are ready to adopt ICTs because it features transcend time and space. However, limited financial budgets lack of IT skills, and lack of enough infrastructures enterprises are stepping back. To design information resource management software for small and mid-size enterprises under budget cost and strict deadline was always problematic for engineers. At the same time, inadequate knowledge of different approaches to software engineering is a major challenge while working under the limited budget and a strict deadline.

Therefore, in this study, we have reported an approach to design enterprise software under a strict deadline. Our proposed approach will assist software engineering students and IT analysts to understand the phases involved in the Waterfall model effectively.

At first, we introduced mid-sized enterprise software named "BSK Management System", which assists enterprise software clients with information resource management and perform complex organizational tasks. The proposed system assists enterprise software clients to assist with information resource management under limited financial budget. The system was designed and developed on a non-profitable organization located in Bangladesh to perform its daily operations. The organization is involved in activities like to promote reading habits, enlightenment, and progressive ideas among students and the general public in Bangladesh. The system analysis and design phases have been done based on the given information on the organization.

In the system analysis phase, generic 7-phases (Preliminary Investigation, System Analysis, System Design, Coding, Testing, Conversion, and Maintenance) of the Waterfall model had been investigated properly in engineering manner. The reasons for using the Waterfall model were 1) it allowed designing flexibility, 2) rigidity, 3) phases do not overlap with each other, 4) software testing in every stage, 5) easy debugging, 6) less ambiguity in requirement gathering phase, 7) limited budget, 8) strict timeline, and 9) quick implementation.

At the beginning of this study, we have gathered all the requirements to proceed to system analysis phase. We have interviewed the employees of the enterprise and analyzed the infrastructure of the enterprise as the part of requirement collection. Secondly, based on the gathered information, we have drawn the DFDs in context level, system level, and level 1 of the enterprise. Thirdly, we have drawn the entity lifecycle diagram for the flexibility of the project. Fourthly, conceptual entity relationship diagram has been designed. Later on, databases have been implemented base on conceptual and logical relationships between entities. Fifthly, we have developed and reported 4-major modules for the enterprise. Four modules for library, accounts, examination and HR have been developed and delivered to the client. We have designed simple interfaces because of users' inadequate knowledge on technologies. Developed software has been tested and debugged with proper engineering ways.

Then, we have supported this study with SEQUEL, rich picture of the enterprise, and built a data dictionary. A documentation of the entire system has been prepared for further improvements.

Finally, the software has been delivered to the client within the given time frame. The software had been implemented and users of the system have been trained to use the system.

Moreover, we have conducted a questionnaire survey with 20 participants to investigate the usability of the system. The result revealed that users found our system easy to operate, accurate transactions, fast processing and easy to navigate. An interview followed by the questionnaire survey has been done. The interview indicated that users of the system are quite happy to use the system. Most of the participants agreed that the

proposed system would help them to fulfill their daily transaction.

Time to complete the entire project set by the enterprise was 20-weeks. Table I displays the allocation of time planned and required to complete each phase.

TABLE I
PROJECT TIMELINE

Phases	Planned Time (Days)	Required Time (Days)
Preliminary Investigation	12	11
Requirement Analysis	22	22
Data Flow Diagram	28	24
Entity Relationship Diagram	14	12
Backend	7	7
User Interface	14	10
Testing	7	3
Handover & Implementation	7	5
Usability Test	7	5
Others	7	6

A complete understanding of the approach followed to develop enterprise software in this article will give engineers ideas to plan, schedule, and deliver a project successfully under strict deadline and a budget cost.

VIII. FUTURE WORKS

Software development by using SDLC's other models- in a similar environment would be a recommended future work for new software engineers and analysts. As design-acuity gap remains a challenge for software failures, further investigation needs to be taken to the improve design-acuity gap. A large-scale project by using this approach needs to be conducted.

Currently, BSK Management System v1.0.0 supports four major modules. Developing five more modules are under consideration and will certainly be taken into action as future works.

ACKNOWLEDGMENT

We, hereby acknowledge Mr. Shamimul Hasan Azim (Researcher, VirginiaTech USA; ex-Faculty, Stamford University Bangladesh) and Assistant Professor Kamruddin Md. Nur (PhD Candidate, Pompeu Fabra University SPAIN; Faculty, Stamford University Bangladesh) for their constant advice and guidance throughout the project. In addition, we show our deepest appreciation to the honorable Chairman, BSK to grant us permission to conduct research activities and publish the outcome.

We would also like to express our special thanks to the participants of the experiments. Without their help and support, this study would not have completed successfully.

Authors would like to dedicate this project to the victims of Cyclone Sidr in 2007, Bangladesh.

REFERENCES

[1] C. R. Necco, C. L. Gordon, and N. W. Tsai, "Systems analysis and design: current practices", *MIS Quarterly*, 461-476, 1987

[2] K. S. Nantz. "Supporting End-User Application Development with the Information Transformation-Analysis-Management Model", *In Managing Information Resources in the 1990s: Proceedings of 1990 Information Resources Management Association International Conference (p. 66). IGI Global*, 1990.

[3] J. L. Whitten, V. M. Barlow, and L. Bentley, "Systems analysis and design methods", *McGraw-Hill Professional*, 1997.

[4] P. Palvia, J. T. Nosek, P. Palvia, and J. T. Nosek, "An empirical evaluation of system development methodologies", *In Managing information resources in the 1990s: proceedings of 1990 Information Resources Management Association international conference* (p. 72), 1990.

[5] N. B. Ruparelia, "Software development lifecycle models", *ACM SIGSOFT Software Engineering Notes*, 35(3), 8-13, 2010.

[6] H. D. Benington, "Production of large computer programs", In *ICSE* (Vol. 87, pp. 299-310), 1987.

[7] W. W. Royce, "Managing the development of large software systems", *In proceedings of IEEE WESCON* (Vol. 26, No. 8), 1970.

[8] S. Conger, "Software development life cycles and methodologies: Fixing the old and adopting the new", 2010.

[9] S. Balaji, and M. S. Murugaiyan. "Waterfall vs v-model vs Agile: A Comparative Study on SDLC" *JITBM & ARF 2.1*: 26-30, 2012

[10] C. Sue. "Software development life cycles and methodologies: Fixing the old and adopting the new." (2010)

[11] R. Heeks "Information systems and developing countries: Failure, success, and local improvisations." *The information society* 18.2: 101-112, 2002.

[12] K. R. Linberg, "Software developer perceptions about software project failure: a case study", *Journal of Systems and Software* 49, no. 2: 177-192, 1999.

[13] P. P. Chen, "A Preliminary Framework for Entity-Relationship Models", In *ER* (pp. 19-28), 1981.

[14] Y. L Chen, "Data Flow Diagram." *Modeling and Analysis of Enterprise and Information Systems. Springer Berlin Heidelberg,* 85-97, 2009.

[15] https://support.ca.com/cadocs/0/CA%20Gen%208%205-JPN/Bookshelf_Files/HTML/AnalysisGuide/index.htm?toc.htm?1100359.html

[16] H. S. Brian, and J. M. Edwards. "The object-oriented systems life cycle." *Communications of the ACM* 33.9: 142-159, 1990.

[17] M. M. Astrahan, and D. D. Chamberlin. "Implementation of a structured English query language." *Communications of the ACM* 18.10: 580-588, 1975.

[18] A. Monk, and S. Howard. "Methods & tools: the rich picture: a tool for reasoning about work context." *interactions*5.2: 21-30, 1998.

[19] P. B. Checkland, "Soft systems methodology." *Human systems management* 8.4: 273-289, 1989

[20] P. B. Checkland, "Soft systems methodology." *Encyclopedia of Operations Research and Management Science. Springer US*, 1430-1436, 2013.

[21] C. J. Wertz, "Data Dictionary: Concepts and Uses", *John Wiley & Sons, Inc.*, 1986.