A System Analysis Reader

By

Nelson O. Guinmapang Jr.

For
System Analysis

TABLE OF CONTENTS

I BOOK REVIEWS

II CASE STUDIES

III. USE CASES

Appendix 1: EXERCISES

Appendix 2: FINAL PROJECT

Appendix 3: REFERENCE

http://www.wikipedia.org
http://www.google.com
http://www.yahoo.com
De La Salle – College of Saint Benilde - Learning Resource Center (EXTENSION)
Introduction to Systems Analysis and Design by: Igor T. Hawryszkiewycz

Dedication:

I dedicate this book to my family and friends, who helped me in a lot of ways to be responsible enough to do this kind of thing.

Preface:

I have been a student for 12 years, but this is the first time I have encountered this kind of requirement wherein I will compile all my paperworks and make it a book. This is good, because it will show how responsible a person is to be able to do this kind of requirement and of course for the first time, I will able to author my own book. System Analysis and Design is an introductory subject to IT majors, hope this book can be helpful to all.

CHAPTER 1 – BOOK REVIEWS

Introduction to Systems Analysis and Design

By:

Igor T. Hawryszkiewycz

T

57.6

H38

1991b

Chapter 1 – The System Design Environment

Review:

This chapter overviews what is the meaning of a system and how does it works. The importance of systems analysis is also explained in this chapter. According to the book, systems analysis is important because this is where we build a new information systems or changing existing ones. In this chapter, we saw a variety of people involved in a system development and what analysts do. System analysts must work with all the people involved in the systems development. They must have good communication skills to be successful. It is also necessary for system analysts to have assistance from different tools for their work.

Introduction to Systems Analysis and Design

By:

Igor T. Hawryszkiewycz

T
57.6
H38
1991b

Chapter 2 – Typical Information Systems

Review:

This chapter talks about what a typical information systems looks like. It shows examples of different systems. Like Human Resource System, Customer or Client System, Inventory Control System, Accounting System, and Marketing System. In these examples of systems, we can see that they are inter-related to one another. They have their own subsystems which have own tasks to do for the system. Human Resource is in charge of the payroll and the personnel, Accounting is in charge of accounts payable, general accounts and accounts receivable, Inventory control takes care of the assets and consumables, Client system checks the client history and order processing, and Marketing system is in charge of the sales, promotion and publicity as said in this chapter. Also this chapter includes 2 typical cases which exemplifies the kind of work often undertaken by analysts.

Introduction to Systems Analysis and Design
By:
Igor T. Hawryszkiewycz

T
57.6
H38
1991b

Chapter 3 - Problem - Solving steps

This chapter talks about the steps used to develop computer-based information systems. It is described as the linear cycle which is composed of five phases used to build the system also includes two additional phases that take place after the system is built. Project development proceeds in sequence through these phases.

The phases of a Liner cycle are:

Phase 1 – Problem definition
People consider this to be the most important phase. It defines the problem to be solved and thus sets the direction for the whole project.
Phase 2 – Feasibility
This phase proposes one or more conceptual solutions to the problem set for the project.
Phase 3 – Systems analysis
This phase is detailed appraisal of the existing system
Phase 4 – System Design
This phase produces a design for the new system
Phase 5 – System Construction
This phase is often broken up into two smaller phases: development and implementation. The individual system components are built during the development period.

After the product has been completed, these are the two additional phases:

Post-implementation review
This phase takes place about a year after the system is implemented.

Maintenance
Maintenance is necessary to eliminate errors in the working system during its working life and to tune the system.

Each phase in the linear cycle has defined objective and uses a number of activities to achieve this objective. Each phase produces an output that includes phase product and management report which describes the difficulties encountered in the phase presents a plan for the subsequent phase.

Introduction to Systems Analysis and Design
By:
Igor T. Hawryszkiewycz

T
57.6
H38
1991b

Chapter 4 – Gathering Information

This chapter talks about a good understanding on what the system needs and its problems. If you can identify the problem then you can come up with possible solutions in solving the problem. You have to be sure that the new system is acceptable by users and is acceptable by the organization. An understanding of the system cannot be determined by sitting down and drawing a model or setting user requirement. It can be solved by talking to people who use it and from there you examine what it needs to fix. System analysts must set objectives and use the requirements in order to build the new system.

Information gathering can be an onerous task. It must be gathered in an organized way to make sure that nothing is overlooking with the system detail.

There are different Information Sources these are:

- System Users – are usually the first information source investigated by analysts
- Forms and Documents - are useful sources of information for system data flows and transactions
- Computer Programs – can be used to determine the details of data structures or processes
- Procedure manuals - specify what people do in an organization
- Reports – kinds of outputs needed by the users

These are different Search Methods:

- Interviewing – the most common method of gathering information.
- Questionnaires – is sent to the users and replies are analyzed by analysts.

This shows how to plan a search strategy and examine the differences between search strategies where there is an existing system and those where there is no existing system.

Introduction to Systems Analysis and Design
By:
Igor T. Hawryszkiewycz

T
57.6
H38
1991b

Chapter 5 – Starting a Project

In this chapter, it describes how projects are started. The first two linear cycle phases, problem definition and feasibility analysis, are used to start a project. These are important phases because they set the pattern for the remainder of the project. He first phase defines the goal which is problem definition and the second determines a feasible way of achieving this goal.

The problem definition and feasibility are important they give direction for the remainder of the project. A system analyst must find the best possible solution without actually building the system. Once a broad solution is agreed upon, detailed analysis and design will commence.

Feasibility analysis commences once the project goal is set. The first feasibility analysis step proposes a set of solutions that can realize the project goal. These solutions are descriptions of what the new system looks like. The second step evaluates the feasibility of such solutions. Setting the project goal is perhaps the most important activity in the problem-solving cycle. It ensures that the problem that will be solved is right. One important guideline for goal-setting is to remember that goals must not be unrealizable ideals that are subsequently ignored. They must be developed within the practical bounds of the organization.

Once the proposals are generated they must be evaluated, Evaluations looks at the:

- Technical feasibility
- Operational feasibility
- Economic feasibility

To summarize this chapter, we can see the outlined methods used to define the goal and evaluate the feasibility. It also explained the necessity of

proposing a number of solutions and then evaluates it again. The final step of the feasibility is to develop an implementation plan for the project.

Introduction to Systems Analysis and Design
By:
Igor T. Hawryszkiewycz

T
57.6
H38
1991b

Chapter 6 - Data Flow Diagram

In this chapter, it explains how data flow diagrams are important. It is one of the most important tools used by system analysts. The use of data flow diagrams as modeling tools was popularized by DeMarco and Gane and Sarson through their system analysis.

The modeling components that the system analyst must have are: system processes, the data used by these processes, any external entities that interact with the system and the information flows the systems.

According to this chapter, data flow diagrams are made up of a number of symbols which represents components. There are ways where you can ensure that DFDS are self-explanatory, complete and unambiguous. These properties are:

- the absence of flow chart structures
- conservation of data
- good naming conventions

These are the difference between Flowcharts and Dataflow Diagrams:

- DFD should have no data flows that split up into a number of other data flows
- DFD must have no crossing lines
- DFD must not include flow chart loops of control elements
- And DFD must not include data flows that act as signals to activate processes

To summarize, this chapter describes the main component of structured systems analysis, the data flow diagram. It shows how dataflow diagram looks like, how you name each process and able to know its correctness.

Introduction to Systems Analysis and Design
By:
Igor T. Hawryszkiewycz

T
57.6
H38
1991b

Chapter 7 – Describing Data

In this chapter, it says that an important aspect of systems analysis is the system data. As with a data flow diagram, an abstract or implementation independent model of the data is developed first and in the end it is converted to a physical implementation.

Data analysis is a more difficult subject than data flow diagramming. A DFD almost looks like a system. It has boxes that you can actually envisage as physical operations and its flows can be imagined just as easily. A data model on the other hand is often more abstract and difficult to relate to actual system components. Often it contains data from more than one function or it must show associations that are not visible as physical things in the system.

Data modeling presents another difficulty to a beginner. It is made up of more than one level and different ideas are used at each level. In the first level, analysts develop a *conceptual model* of data. This model represents the major data objects and the relationships between them. The first level is also sometimes called *semantic analysis* because it captures the natural semantics of the system. While the next level is the one who organized the first model into a good shape. This process removes redundancies; often it is called *normalization*, which uses ideas from relational theory. The normalized model is then converted to a physical database.

Conceptual model describes the essential semantics of system data. This model consists of a number of symbols joined up according to certain conventions. Entity-relationship analysis is a method where conceptual model symbols are used. This method was introduced by Chen in 1976 and is now widely used.

Entity-relationship analysis uses three major abstractions to describe data. These are:

- entities, which are distinct things in the enterprise;
- relationships, which are meaningful interactions between the objects; and
- attributes, which are the properties of the entities and relationships.

The E-R model is refined by converting it to a relational model. It is also provides a number of criteria to construct normal form relations. Finally this chapter shows the

relationship between the E-R model and DFD diagrams and how this two can be used to ensure that correct and complete models are developed during analysis.

Introduction to Systems Analysis and Design
By:
Igor T. Hawryszkiewycz

T
57.6
H38
1991b

Chapter 8 – Advanced Modeling methods

This chapter briefly covers some alternatives methods that are now used for high-level conceptual modeling and will compare these methods to E-R modeling. The most common extensions to the E-R model are dependent entity sets and subsets.

Dependent entities are also sometimes known as weak entity sets. An entity or relationship set is made up of objects that have not only the same properties, but another property in common: their existence depends on the existence of parent entity in another set. If that parent is not of interest to the system, its dependent entities also cease to be of interest to the system.

Dependent entities are very useful for modeling historical information. Historical information must be kept whenever two entities interact more than once.

Subsets are modeling situations where we wish to treat entities from one entity set in the same way in some cases and differently in other cases. To model different methods of treatment of entities in an entity set it is necessary to show division of entity sets into subsets. It is also common for entities in different subsets to have some different attributes and some common attributes. The common attributes are shown as attributes of the parent entity set, whereas attributes particular to a subset are shown only as attributes of that subset.

Entity-relationship modeling is only one of many alternative methods that have been proposed for conceptual modeling. The alternative methods use different modeling abstractions and different modeling symbols, two such methods are entity modeling and binary-relationship modeling.

Entity-relationship models of data are static models. They describe the data structure but do not show what happens to various parts of the data structure. Entity life-cycle histories tell us what happens to entities in the system. Each entity in the system goes through a number of states. An application may b e received in which case it becomes a 'received' application. The application may then be checked, it then becomes a 'checked application' and finally the application may be approved, in which case it becomes an 'approved' application. To summarize, this chapter described some alternative methods for conceptual modeling.

Introduction to Systems Analysis and Design
By:
Igor T. Hawryszkiewycz

T
57.6
H38
1991b

Chapter 10 – Documentation

Documentation is both a communication tool and management tool. It is a communication tool because it contains a repository of all work done to date and makes it available to all persons working on a large project. Proper documentation ensures that all the information developed about the system is always available to new people joining the project.

Documentation is also a management tool because it supports management in two ways. First, it ensures greater efficiency because all project personnel have access to the latest work and thus less work is required. Second, it is only project deliverable, especially in the early project phases. A broad design and evaluation are only the outputs of the feasibility study, and system models are only the results of systems analysis and broad design.

To be useful, documentation must be mandatory and most organizations that adopt formal design methods make documentation mandatory. Mandatory documentation requires formal techniques in system analysis and design. It also ensures that the organization obtains the communication and management advantages of good documentation practices.

There are different Project Dictionary Entries:

- DFD diagrams
- Data Structure
- ER diagrams

To summarize, documentation is necessary to support design methodologies. It keeps track of work completed and makes this work available. It also helps analysts to maintain system consistency by storing cross-reference between the components. It is now up to the analyst to detect any inconsistencies in the model.

Introduction to Systems Analysis and Design
By:
Igor T. Hawryszkiewycz

T
57.6
H38
1991b

Chapter 11 – Designing the New System

Of all the linear cycle phases, design is the hardest phase to describe. It is different from analysis because in analysis it is possible to come up with a correct model of the existing system however you cannot create a correct design. It is even harder to know what the best design for a given system is. A good design is very dependent on a particular system, and we need to be aware that a good design to a system may be a bad design to a different system. Design requires considerable creativity to make the necessary system-specific changes and additions and come up with a new system that is acceptable to users and easy to implement.

System Objectives:

The objectives are the driving design factor. Any changes to the system should be made to satisfy these objectives.

Kind of Objectives:
1. Functional Objectives
2. Operational Objectives
3. Personal and job satisfaction objectives

Different design ideas are needed to satisfy each of these objectives. The first objective can be achieved by adding new processes to the system or amending existing processes. Objectives can be high-level, low-level, general or specific. These are some key factors to be analyzed:

- the information requirements to satisfy the key goals
- performance targets (such as time, volume, sample size)
- comparison to existing systems
- personnel issues such as skill levels

Broad design in this design methodology consists of two fundamental steps – producing a new logical model and following this with a new physical model. The next step is to go on with detailed system design.

Introduction to Systems Analysis and Design
By:
Igor T. Hawryszkiewycz

T
57.6
H38
1991b

Chapter 12 - Detailed System Design

In this chapter, it defines what user procedures are; these are what people must do to make the system work. There are two important factors to consider in user procedure design. One functional design, which is, making sure the system, does what it is supposed to do. Functional design defines the tasks necessary to make the system work. The other important part of detail design is job design. Job design specifies what people must do within the tasks. These two factors are related because tasks must be designed so that they can be carried out by people.

Functional Design – user procedures specify the tasks needed to capture data and pass it correctly through the system. A user procedure must specify how data is obtained, the medium used to carry the data, and the format of the data on the medium.

Describing the User Procedures – One output from design must be a description of user procedures. These descriptions are usually of a physical nature. Flowcharting is one of the earliest representation tools for physical systems and several flowcharting techniques are frequently used to describe physical procedures.

Job design – this must encourage user involvement with the system. It must cultivate considerable human satisfaction and to achieve better system acceptance. It must also allow people to improve their skills while using the system.

Job enrichment is now a common term in practice makes jobs more interesting and rewarding for system users. It gives individuals greater initiative and responsibility and encourages them to improve themselves because it involves the in a greater variety of activities.

Another important part of user procedure design is the computer interface. This defines how users in the system use computers. The interface must be designed to capture all the information without error and to provide, in an easily read form, any information needed by the users. There are 2 kinds of interface, online and offline.

Introduction to System Analysis and Design
I.T. Hawryszkiewycz [Third Edition]
T
57.6
H38
1994

Chapter 6 - Starting a Project

In this chapter, it discusses how important that a company must identify the problem that will be solve. Then to justify that solving the problem is worthwhile. Finding the right problem to solve is perhaps the most important thing that a company/management must do. It said that if a company tries to solve the wrong problems, the company will be wasting time and resources, and then the solution will not have value to anyone. If the management do not identify the right problems to solve, then the system ill not be as good as it should be and eventually this will lead to difficulties.

The chapter describes how to find problems to solve, how the management can justify solution and how projects are started. The chapter describes problem finding and justifying it in the context of the linear cycle. However, problem finding and justifying is equally important in any other kind of system development life cycle and usually precedes analysis and design activity in any project.

Setting the goal of a management can be done in two ways: Setting goals using external considerations and setting goals using internal considerations. Some of the ways in setting goals using external considerations are; using normative models, which describe an accepted or conventional way of doing something; using historical models of the ways in which organizations develop; comparing activities and analyzing changes to government policy and community attitudes.

While in setting goals internally, the management must focuses on their internal problems and tries to solve it. The management asked questions to the people that are working in the environment to further develop their knowledge about the different problems that the management encounters.

In summary, the chapter describes how a project is started in a management. The first step is to define the project goal, and the next is to evaluate the feasibility of achieving this goal.

The chapter outlined methods used to define the goal and evaluate its feasibility, together with the necessity of proposing a number of broad solutions and then evaluating the technical, operational and economic feasibility of the solution.

System Analysis Design Seventh Edition

By: Garry B. Shelly, Thomas J. Cashman Harry J Rosenblatt

QA 76.9 S84 S97 2008 (LRC Extension)

Chapter 2 – Analyzing the Business Case

This chapter caught my attention the most since this will play a big role in the development of our course. Knowing the nooks and crannies of a good business case would help progress a startup further. Through the business plan a strategic goals can be set along with the finance, marketing and human resources. But before you make a business case you ask yourself, what exactly is a business case. The business case tells why the team is doing the project, what the project is about and how this solution address key business issues does. Also how much and how long will it take and will the productivity suffer loss during the transition. Then what is the return on investment and payback period and what are the risks of doing and not doing the project. After all has been said the last questions are how will you measure success and if there are any alternatives. We can clearly see that making a business plan is not an easy task. Each and every question must be researched and answered seriously. That is why being able to produce a great business plan will really increase the growth of a company.

CHAPTER 2 – CASE STUDIES

Case Study (Foreword)

In this book's foreword, it talks about what startup company looks like at first and what kind of ideas come to people's mind when they hear the word "business". According to this book, less energy people expend on performance, the more they expend on appearance to compensate. This concept is obviously wrong, because performance in work is more important rather than how you look.

Also this book says that people think better when waking up in the morning rather than in work. Because maybe people will came up ideas when they are comfortable and relaxed. Where in work, you have to wear a certain dress code and you get stress if there are many works to de done.

This foreword inspires those people who have the abilities to do startups. Startups have usually 3 to 4 people involved. But once these 3 to 4 people to work hard and able to come up wonderful ideas for business improvement then we can consider them as successful founder of a startup and eventually be a part of a large and successful company. But of course, if you build a startup you must consider the possible things that could go wrong and able to handle them.

Case Study (PayPal)

Paypal is an e-commerce business allowing payments and money transfers through the use of internet. It was co-founded by Max Levchin on December 1998. At first, Max didn't plan on doing payments, because he was into security. When he went to Stanford University, he met Peter. Since then he and Peter became partners in developing a startup company.

Paypal encountered many problems during its first years. Many people are afraid of "fraud" that can happen if you make a transaction in Paypal. So he and his colleagues try find a way to solve these problems. They analyzed the information they got, when he was close in solving the problem, he worked 3 days straight just to finish the program and on the 3rd day he was suppose to present this to the possible buyers. When Max finally solved the problems with fraud, his work is slowly becoming popular until people from e-bay wants to buy his product for 1.5 billion US dollars.

From this story, we can see that developing a startup company takes a lot of work. There are people who want you to quit and just work for them, some will discourage you and even try to manipulate you. But once you showed how willing you are to pursue what you started then you will surely become successful in developing a startup. You must know who the people working for you, you have to trust them for every work they do.

Case Study (Hotmail)

Hotmail is one of the first free web-based email founded by Jack Smith and Sabeer Bhatia. They started to develop a startup when their employer's firewall preventing them to access their personal e-mail accounts.

They started developing a web-based email that can be accessed free by people as long as they an internet connection. At first they had problems in funding, because in order for them to develop this one of them has to quit their job so that the other can concentrate in developing Hotmail. They got their first funding from Drapier Fisher Jurvetson (DJF) who later became their competitor. Since Sabeer is just an entrepreneur, he didn't understand that he can't go to other VC. He focused in developing Hotmail, his work is becoming popular. It spread by means of word of mouth. Finally, Hotmail was sold to Microsoft for $400 million.

In this story, funding maybe the most difficult problem they encountered. They must develop and hoping that someone will like their work and give them money to develop it too a much larger scale. Also because of different companies that are rising from their time, they had many competitors. But since they were one of the first in introducing this web-based email, they are the one of who became popular. The advice that Sabeer said is in order to develop a startup, you must have a business plan because according to him it crystallize your thoughts to communicate your ideas with somebody else.

Case Study (Apple Computer)

Apple Computer is an American Multinational corporation with a focus on designing and manufacturing consumer electronics and even develops software products. It was con-founded by Steve Wozniak and Steve Jobs. Steve Wozniak met Steve Jobs while he was working at Hewlett-Packard. Steve Jobs worked part time, where he would finish up games that they designed in Grass Valley.

In 1975, the first personal computer kit, the Alistair 8800 was announced. Since Steve Wozniak couldn't afford an Alistair 8800 he decided to build his own personal computer by using cheaper chips. As circuit board alone, it could do more than Alistair. He and Steve Jobs called it Apple I, Jobs handled on marketing it while Wozniak continued to improve it. By 1977, Wozniak had built Apple II, then he and Jobs decided to form Apple Computer Inc. when it went public on 1980, its stock value was $117 million, three years later it was $985 million.

This story of how Apple started is one of my favorite. We watched a film titled "Pirates of the Silicon Valley" last year that's why Apple story is a little familiar for me. We can see that Steve Wozniak is determined that he will develop a computer even if he was still a child. I can say that if we really put out hard work and commitment to do something then we can be successful. According to Wozniak, if we try to start our own company, we must have the highest ethics and be open and truthful about things, not hide. We must not lead people. Know in your heart that you're a good person with good goals because it will carry over to our own self-confidence: make our own product better that the average person would.

Case Study (Excite)

Excite is an Internet portal, it offers a variety of services, including web search, instant messaging, web-based e-mail, and customable user homepage. It was founded by Stanford University students, Graham Spencer, Joe Kraus, Mark Van Haren, Ryan McIntyre, Ben Lutche and Martin Reinfried. All of them are Computer Science major except for Joe Kraus who was a political science major. Even if Joe Kraus is a political science major, that didn't stop him in helping his co-founders in developing Excite. In a matter of fact, he did more coding than some of them. Except Graham Spencer, Spencer almost did all the work and he was the one who come up with the idea of how Excite will work.

Like other start-ups, Excite encountered some problem, since they were 6 of them who founded this Internet portal. They have to find a way to redistribute the equity in a way that isn't even. They approved that Graham will get more since he has more technical ability than other guys. From this experience they had we can see that in business, you must choose wisely your partners. Because maybe in some future time there will be some regrets or arguments that will be encountered. Also you must consider what other things can be helpful for your company.

Case Study (Software Arts)

Software arts was a software company who develops VisiCalc, the first electronic spreadsheets. It was co-founded by Dan Bricklin and Bob Frankston.

VisiCalc became a "killer application" for computers at first, but because it was not produced by a company organized like a modern startup. That maybe caused the downfall of this company although it was distributed by Daniel Fylstra's Personal Software (later renamed VisiCorp). As a business, Software Art's fall was as fast as its rise, but it had influence than many longer-lived companies.

In this story, I can say that startups don't last forever. Eventually something big will happen, whether it can be good or bad for your company, you must be ready for the consequences that you will face. You must know the people you're working and able to trust them to make your company a successful one. But even though their company doesn't exist today, it helped a lot of people in doing a startup nowadays and also in the business of company.

Case Study (Lotus Development)

Lotus Software is a software company who is commonly famous for developing 1-2-3 spreadsheet application, by far the first feature-rich, user friendly and highly reliable product to be used in early days and replaced Visicalc as the "killer application". Lotus also released a groupware and email system called Lotus notes. It was co-founded by Mitchell Kapor and Jonathan Sachs. Lotus went public in 1983 and on 1995 IBM purchased Lotus for $3.5 billion.

Mitchell Kapor first worked for VisiCorp (makers of Visicalc) and he even sell his rights for the Visiplot and VisiTrend before he left. By the time he left, he and Jonathan Sachs produced an integrated spreadsheet and graphics program. By 1983, Lotus developed 1-2-3, way more superior than VisiCalc. They named it 1-2-3 because it refers to the three ways the product could be used, as a spreadsheet, graphics package and database manager.

Lotus Software was involved in several lawsuits according to Mitch Kapor. But the most significant is the "look and feel" cases. Lotus sued Paperback Software and Mosaic for copyright infringement, false and misleading advertising and unfair competition. Lotus software won the case and Paperback and Mosaic went out of business. Mitchell Kapor was surprised when this company became successful, he admitted that he and Jonathan are technical people and they know that their product will become useful for a lot of people, but he didn't expect to be in the place where they are right now. He even became popular at their town, he felt like he was a celebrity.

For Mitchell Kapor, if we are trying to do our own startup. We must have our own compatible set of values, must be inspired by a vision and we must be passionate about disruptive technology. We must create something that really has a value for people. He also said that in order to be successful entrepreneurs, we must know the people we are working with, this will help you to work easier and better.

Today, Lotus has many products, namely Lotus Connections, Lotus Domino, Lotus Expeditor, Lotus Notes, Lotus Forms, etc. Base from this example, if we are willing to give our best ideas and hard work, then someday we also can achieve what Mitch Kapor and Jonathan Sachs. We must consider some risks, because startups don't easily become popular, we must be patient because we

know that our time will come. And you should not underestimate people and you should not judge them from their appearances. In a company, you should have trust in your employees and as well as your co-workers

Case Study (Iris Associates, Groove Networks)

Iris Associates was a software development company founded in Massachusetts by Ray Ozzie to build the software known as Lotus Notes. Tim Halvorsen and Len Kawell, joined Iris shortly in January 1985, met Ray Ozzie years before when all of them were working on the University of Illinois PLATO system. They are widely regarded as the core team behind the creation of Lotus Notes.

The first thing they did was to code a primitive version of their synchronization algorithm. In 1982, it was the concept that they will be using computers on the desktop s and therefore they might want to use them as communication tools. This was a time when PC's were just emerging as spreadsheet tools and word processing replacements, still available only on a subset of desks, and definitely no networks. The company's primary funding came from a contractual relationship with Lotus Development Corp., best known at the time for its Lotus 1-2-3 spreadsheet software.

Ray Ozzie also founded Groove networks, this company specialized in productivity software that allows multiple users to work collaboratively on computer files simultaneously.

Before he started his company, Ray Ozzie wrote a couple of founding documents, one is a scenario-based document, describing the high-level challenge that the end user scenarios they are trying to solve. This explains the reason why the company is developing that kind of program, so that people who will join will have enough ideas. Another document that he wrote is a bottom-up document which describes the different technologies that will have to be assembled to accomplish a vision.

The author/publisher contractual relationship between Iris Associates and Lotus Development continued until 1994, when Iris was purchased by Lotus for approximately US$84 million. In 2004, nearing the 20th anniversary of the founding of Iris Associates, IBM reported that Notes has over 110 million users. This is a good story, because the way he started his company is very hard. He tool some risks and able to overcome them. And finally his hard work pays off when many invested in his company.

Case Study (Pyra Labs)

Pyra Labs was a web application which would combine a project manager, contact manager and to-do list. It was founded by Evan Williams; he lived in Nebraska doing internet stuff and even had a company there. He went to college but dropped-out, because for him, he doesn't need a degree because he was not trying to work with anyone. He moved to California because clearly, Nebraska was not the place for him. He worked with O'Reilly.

He worked as a developer with HP and Intel. He was not happy with his work, because he knows that he is capable of starting his own company. He already had visions on how to raise money and building something new. The idea of "Pyra" was already in Evan's mind, it is a web-based project management system or collaboration and it is assigned to build projects for clients around intranets and help them organize their work and personal information. It is a web application where you would put your stuff, things you are thinking about, things you had to do, things you wanted to share with other people.

Evan was on the verge on starting his own company so he decided to consult with his friend Meg Hourian, who was working as a management consultant. The first year of Pyra Labs was funded by HP, their company is just doing work for HP. They started working full time by November 1998 and even hired their first employee, Paul Bausch. Paul turned his site, onfocus.com, into a blog which Evan did. Blog started when Evan took a script and wrote it to publish the site, and then made an internal site where they could do the same thing. They made an internal blog called "stuff". They build real collaboration tool, with all kinds of structure and bid ideas trying to be implemented to it. Paul also wrote a little addition to Stuff so that certain things they posted to their internal blog could put on the external company blog. Luckily they had investors like O' Reilly. They raised one and a half million dollars from Advance.net. A half million dollars was a ton of money at that time. After earning money they decided to focus on Blogger and develop it. When launched publicly, Blogger grew rapidly and Pyra Labs decided to focus on it full-time. Blogger also reached its decline. But Williams remained as the only employee and manage to bring back the company. By 2003, Blogger had 1 million users.

When I read this article, I felt envy Williams, because he risked studying and having a degree because he felt that he will develop something that people will be using. He thought that his idea was simple, bit the people proved him wrong. Now, many people are doing Blogs because of his idea.

Case Study (Non-founding employee, Yahoo)

As well know, Yahoo was started by two Stanford University students, Jerry Yang and David Filo. But behind their success was a friend named Tim Brady. Tim Brady was Jerry Yang's college roommate; he was supposed to graduate by the end of the semester, but because of Yahoo's rapid growth he needed to do a business plan as soon as possible. So he made the business plan for Yahoo as his final requirement on one of his course to pass. He was Yahoo's actual first employee.

Tim Brady's title during his 8 years stay at Yahoo was VP of production. His responsibility is more on of course, "products". He was the editor of the Yahoo site. Yahoo won many portal wars because it was a better site, and it was the site it was largely because of the works of Tim Brady.

According to Tim Brady, Yahoo started as Jerry and David's PhD theses. They would add some categories especially in the things they are interested in. It grew bigger when people would e-mail to add something. That's how Yahoo began. The time came when VC's were interested in their product, they would tell them to come to their office and show them what they got. But they decided to just start their own company. That's the time when Tim Brady will come up with a business plan. Tim Brady used his simple business plan format, incorporated it with Jerry and David's ideas and added a few of his own to come up with a business plan.

Their mindset is to never let their company sink, they must keep it going. Some part of it is making money, so they would do a little advertising in their site. For him, it is tough to convince people to join Yahoo, because mostly the people they hired were friend, or friends of friends. There was a high level of trust between them because they all know each other. For him, the proud moments they had were the Gates memo, going public and being added to the NASDAQ.

I like this story, because I once researched about Yahoo and I never heard about Tim Brady. All I knew was Jerry Yang and David Filo. Now I realized that if it weren't for Tim Brady, Jerry and David wouldn't have a business plan for Yahoo and Brady is the one who is in-charge of the products. His advice for those who wants to do a start-up is that you must "know yourself". You must know what's ahead for you, you must be ready for the challenges it will give you as a founder or part of a start-up.

Case Study (Research in Motion)

Mike Lazaridis founded Research in Motion or RIM with his friend Doug Fregin in 1984 while they were still students at University of Waterloo. One of his first projects was a local area network that ran industrial displays. Their first contract was with General Motors where they got $600,000 to build a similar network they first built. Also RIM was one of the first companies to appreciate the importance of wireless networks. RIM went public on 1997 and it became Canada's most admired technology companies.

Research in Motion is best known for developing Blackberry, handheld communication device. Mike Lazardis met Doug Fregin way back while they were just in grade school, and they started working in high school. They studied at one of the state-of-the-art electronics shop and shop program and that was donated by local industrialist.

He got his first contract when he was in third year, then in his fourth year he started with RIM. Also while working at the University, he developed a programming language which now we know as Java.

When he started working with General Motors, which the one who gave them funding of $600,000, he got a copy of RFP and able to recognize some things in there that you couldn't do without the knowledge from studying at a state-of-the-art techniques he learned from the University. Once he was able to analyze what he needs which is a LAN network. So now they need to build it from scratch, they had to make sure that it will be rugged, because the place where they are going to us it is a hostile environment. He was able to develop a LAN thanks to the first IBM PCS. They ordered hard drives as the bigger hard drives were coming out.

For him, the turning point of his company was when he developed a wireless data. He is one of the people who came up with the idea of Wireless network. Also one of his dreams is able to work for some kind of space-technology. He taught that those were just dreams but he was able to work for SPAR aerospace, a Canadian company. The company needs the wireless network for the upcoming Canadarm2 on the International Space Station.

I like this story, because there is no difference even if you have different nationality. As long as you have the proper education to develop something that is very useful, then you can be successful. And one important reason why he became successful is because the school he went to. If you came from a

prestigious school and you are very hard work then someday, you will become like Mike Lazardis who is very successful.

Case Study (Marimba)

Arthur Van Hoff founded Marimba accompanied by a Java team, Sami Shaio, Jonathan Payne, and Kim Polese, Java's product manager. They all left Sun Microsystems and founded Marimba. At first, they didn't know what to develop so they just funded for themselves and find an office space where it can serve as their headquarters. Kim and Arthur wanted to do a user interface builder and Java. But then they decided to do software distribution instead of user interface builder. They came up with the idea of subscription-based software instead of buying software. Once you subscribe, you can get automatic updates.

They had problem with the CEO position, because they hired Kim Polese as their CEO, she is an inexperienced CEO although she's very good with media stuff, when it comes to decision-making, she had a really tough time. So they replaced Kim for John Olsen, it was a little bit easy for John because he had experienced being a CEO from big companies before.
]
Marimba's progress is fast, from 0 employees to 40 employees in the first year and grew to 300 during the IPO. During its first year, they had their first release, which was very important for their company. They also got a lawsuit filed on their first year. Marimba was acquired by BMC software and later on Arthur and Jonathan started another start-up which was called the Strangeberry.

The important factors that a startup needs are of course a business plan. This is what Tim Brady did for Yahoo. For Arthur it is really important because it shows where your company is headed. Also VCs are important; because they gave the funding you need and even endorse you. Business plan can affect your VCs. If your business plan accepts your VC then go for with it if not, change it. Try to brainstorm again until you come with ideas that can help your company improve. You must able to share your ideas with your group mates and respect each others opinion regarding the subject.

Case Study (Gmail)

We all know that Google was founded by Sergey Brin and Larry Page. But not all the services of Google came from Sergey and Larry's idea. Gmail, is a free web-based email which is provided by Google. The lead developer of Gmail is Paul Buchheit. He was the 23rd employee of Google. He also developed the first prototype of AdSense, Google's program for running ads on other websites. As I said, although he is not a founder of Google, he contributed more to Google than other founders of startups. Gmail is a startup within Google; it started by a small group and was brought to fruition against a good resistance.

For several years, Gmail was only available internally, as an exclusive e-mail for Google employees. The challenging parts that Paul encountered in developing Gmail were people thought that it was not real; they thought it was a joke because they released Gmail on April 1, which is the same day for "April Fools day". Also people thought it wasn't possible because a lot of people will use it and that means a lot of data to store.

For Paul, the best resource he got were the people in Google, because many smart people work in Google, you can brainstorm with them different ideas and finally they will come up with solutions. According to Paul, some people said that Larry and Sergey just got lucky but actually they are smart people with great ideas. For him, it takes more than luck to build something successful.

Also, Paul Buchheit is the one who came up with the "Don't be evil" principle. For him this principle is a little bit funny from other principle like "strive for excellence". In order for you to be successful in a start-up, you must be ready for the risk or in other words you must be risk-averse.

I like this story, because it shows one of the most popular web-based email nowadays. At first I thought that Gmail was one of Larry and Sergey's ideas. But now that I've read the story behind Gmail, I found out that it was one of their employees, Paul Buchheit who was the developer of Gmail. He had many ideas for Google but his two most popular works were Google AdSense and of course Gmail.

Case Study (WebTV)

WebTV was founded by Steve Perlman. His friend Bruce Leak knew the first time he saw Steve Perlman's work that they can make something big from his work. But for Perlman, it was just another day of work, because he when he was working with Apple, he was one of the leading experts in display technology; he even gave color to the Mac, which was Apple's brand of computer. Later on his startup called the Catapult Entertainment, he build one of the first systems for network games. A little over a year after the first prototype, Sony and Philips sold the first WebTV set-up boxes to the public. And finally Microsoft acquired WebTV (now called MSNTV) for $500 million.

Base from the story, I can say that Steve Perlman is really a hardworking person. He can do his work in just one night as long as he is in the correct mood for doing work and he got enough sleep. His wife sometimes get surprised if someone is already sleeping by the pizza box, Steve and his team have the right chemistry to do the works they needed to do.

Because of his hard work and technical abilities, although he is not engineering major, actually he has liberal arts background. That didn't stop him from developing WebTV. He had many great plans for his startup. He even made a business from the time he started WebTV. He admitted that he is a hobbyist; he seamlessly goes from software, hardware, networking and material science.

The biggest issue that they are concern with is that people don't want to interact with their TV. Even though they showed prototypes, for the people it wasn't enough. For Steve Perlman, the main goal for WebTV is connect average people together doing their engineering things, things that interest them and better communication, sharing of ideas, and of course for entertainment.

And finally, Microsoft brought WebTV, during that time, WebTV have only 56,000 subscribers which for them was a fairly modest number.

I like this story because at first I didn't know that the original name of MSNTV was WebTV. I like the way how Steve Perlman works, because you can totally see how dedicated he his to do something. He knows how to handle different things even though he only has liberal arts background. That means that if you really want to pursue or accomplish something then no thing can stop you if you are really dedicated.

Case Study (TiVo)

TiVo was co-founded by Mike Ramsay and Jim Barton. Their original plan was to create a network server for selling homes. They had difficulties when they realized that they need to explain why the consumers needed one of those, so instead they narrowed their original plan and came up with the DVR or the digital video recorder.

TiVo can manipulate the existing information on televisions. With using TiVo, you can skip commercials, pasue live TV programs and schedule the recording of every TV series you like. With this technology, it brought some controversy on Hollywood. Networks are worried about losing control over show how people watched TV. TiVo brought a revolution in the way that people watch TV.

Originally, Mike Ramsay studied in Scotland but went to the United States to work for HP. During his time, there were no PCs. The microprocessor idea has just started. The 4 bit processor was a state-of-the-art during that time.

Somebody told him that he was amazed by Mike Ramsay's work, because he was fascinated by this space and wanted to do something, but he hadn't seen other companies with bright ideas. If you are able to come up with ideas that are useful and have sense then you can do something like what Ramsay did. All you have to do is be resourceful and become hard working with all the work that needs to be done.

As I said earlier, the original plan of Mike Ramsay was not TiVo, just a network server for selling homes. Because of VC's they were able to think about different ideas that made them change their minds in changing a network server to DVR.

I also liked this story, again because of the hard work that we can see. I realized that in a start-up, you must plan ahead so that you can think of the possibility problems that you will encounter once you are doing the start-up.

Case Study (Viaweb)

We all know that Viaweb is a web-based application that allows its users to build and host their own stores. It was co-founded by Paul Graham and his friend Robert Morris. At first they were building Viaweb as software for building online stores. During their first prototype, they had a crazy idea that the users run their software on their own server and let the users control it by using a browser. In this idea, they started building a web-based online store to show the demo to their investors.

Before they started on Viaweb, they already a startup, which was Artix. Artix was art galleries which was online. According to Graham the problem with art galleries they are not for online. From this concept they were able to create online stores which were not art galleries. Rather this was the beginning of Viaweb.

The first problem that they encountered was funding, during that time the only fund that they get were coming from him and Robert Morris in an apartment. That time they were doing a prototype, ay first it wasn't running on the web, but as is said earlier, they come up with the idea to use this store to be available in the web. But since they were able to make it work, they became excited and they got $10,000 funding from their friend Julian who worked for them when they were developing Artix.

When they needed more programmers for the new startup, Graham asked Robert Morris who is the smartest grad student in the computer science program and he suggested Trevor Blackwell. Graham knew Trevor but he didn't know that he was that brilliant, he thought that he was a goofball.

There was a time when Paul Graham almost quit his job which they were told by their investors to refinance the company. Paul Graham already had an appointment to his lawyer where he can quit his job without getting sued. But good thing Fred Egan grabbed him and told him that he will think of something that will fix this problem. Fred Egan was their COO and was willing to do all the business stuff and let Paul and others with their technical work. For Paul Graham, Fred Egan saved them from a summer of horror, where Robert Morris was away and their investors were pressuring them with work.

Finally by the time when their investors were pressuring them was the same time when they were brought by Yahoo. Which was a great deal for them, because originally, if someone was trying to buy you, what company would you choose right? For Paul Graham, Yahoo was the ideal acquirer for their company.

But Paul got a little sentimental when the time came when he must change the front page of Viaweb to Yahoo. He realized that Viaweb is gone, it doesn't exist anymore. Someone told him to stop whining and move on, since the whole point of doing their startup is to sell it to a buyer which was Yahoo.

I like this story of Paul Graham, for me his story is what happening in a start-up. At first you will have financial problem in developing it. You have friend to help you build it. But when you come up with a business plan for the company then you know that your company's future will go somewhere. In the end, I like how Paul felt when he was changing the front page of Viaweb to Yahoo. That particular reaction of a person for me is normal. Because you worked hard for something and in a glimpse it will not exist anymore. But you know that you contributed big for the company and Paul should be proud of himself.

Case Study (del.icio.us)

When we surf the internet sometimes it make us uncomfortable when we need to open a lot of pages in order for to find what we are looking for. It is a waste of time especially when some people have a lot of work to do. In order for the users to have the convenience in searching the site they are looking for, someone by the name of Joshua Schachter founded del.icio.us.

del.icio.us is a social bookmarking web service for storing, and sharing bookmarks. Before developing del.icio,us, Schachter was at Morgan Stanley doing data mining and proprietary trading algorithms. He also created a website called Memepool, an editor with a reader submission. It was a blog that was chronologically sorted and updated. He also built another application, Muxway. It was a bookmarklet wherein you can save things, describe and tag them. This was actually a single player but the actual website was open for viewing. Users got interested with this project. There were actually 10,000 people who are viewing his stuff. So, he found it interesting and got idea of doing some startup.

Before he built this Muxway, he had an idea of tagging his 20,000 bookmarks in order to help him to find the links. He was also doing several stuffs like the GeoURL. So during his spare time he was focused on his del.icio.us. del.icio.us is a multiplayer version and is better than Memepool and have some similarities with Muxway and Memepool.

Joshua Schachter met a guy named Albert Wenger; he had some good experience in operational management. So, Wenger was a big help for him. He just focuses on technical stuff and Wenger will be focusing on operations. Union Square Ventures and Amazon were some of his VCs. For Joshua Schachter, there were a lot of challenges and problems he faced including technical problems, payroll, the network, the product itself and of course the people. Some of the technical problems he encountered were, getting the hardware racked, building machines, ordering stuff, dealing with Dell and the most difficult was the tagging because there was a time that site went down. All his tables and databases in MySQL crashed. So, it was a big deal for him because he had a limited process management in charged to help to cope with these complications. He also worried for the people. He thought that this project was a very technical in order to the users to be familiar in using it. Yahoo brought del.icio.us for a rumored amount of $30 million.

I also like this story of Joshua Schachter, he was already developing different things before he started del.icio.us. So in that sense, he has already ideas coming from his mind, and in that concept, he was able to come up with del.icio.us. At first he created Memepool and Muxway, which gave way to

del.icio.us. He encountered some challenges and problems, but since he was a technical person, he was able to resolve it.

Case Study (Bloglines)

Bloglines, a web-based news aggregator for reading syndicated feeds using the RSS formats. Bloglines was founded by Mark Fletcher; he was a senior engineer for Sun Microsystems when he started ONElist, a free email list service. As we can see, he is working at Sun Microsystems while developing ONElist was just his side project. But a year later, he received the funding he needed and just focused on developing ONElist until it was brought by Yahoo and was renamed to eGroups.

His start-up started when he wanted to start a mailing list on the net for his parents. Then, he thought that this could be interesting because other people might use this. It started as his personal problem in developing mailing lists, then he want to manage his own bookmark list. Actually, he just uses this only to solve his own problems. But because this project will also help other people, he decided to do these as a startup. Before he came up with these ideas, he had worked for Trustic, an anti-spam company.

Early on, he recruited some core groups in order to help him with ONElist and Bloglines. Both were self-funded. But later on, he was able to deal with VCs. He got million users a month. ONEllist became a 150-person company. He has some problems and challenges to face in doing these projects. Mostly of his problems was on Bloglines. In ONElist, his problems was funding because they were growing fast, they got million of users a month. They also had some scaling problems because he had no any idea of how to set up a monitoring system. Also with the ONElist, he had poor communication skills so he asked by their VC to replace as a CEO. In, Bloglines one of these was the people. He was worried because people at that time had no any background in doing blogs. So, he had worries on explaining it to the people. He also worried of adding features because a lot of people emailed to him asking for new features. Competitors came out, but it wasn't big deal for him. Ask Jeeves acquired Bloglines in February 2005.

This story is interesting for me, because I was surprised how Mark Fletcher managed his time by working at the same time he was developing ONElist. For me, that takes hard work and time management, because they maybe sometime when there is a conflict. I was curious on how he able to manage it. Hope someday I can the same the way he did it.

Case Study (Craigslist)

Craigslist is a central network of online communities, featuring free classified advertisements and forums on certain topics. It is founded by Craig Newmark, at first it started as an email list to publicize events at San Francisco. As Craig's list grew in popularity, he switched from mailing list to website and added some categories. Without realizing it, he was about to embark a big bite out of the classified business.

And finally in 1999, Newmark decided that it was time for Craigslist to morph the craigslist.org from a hobby to a real business. Jim Buckmaster joined on as the lead programmer and CTO in early 2000, and was promoted to CEO later that year. Newmark is dedicated to his mission to build a community on the internet; he has this held fast plan to keep craigslist free as possible. All listings are free, except help wanted ads in his plan to select cities and broker apartment listings in New York City, there are no banner ads.

Despite that Craigslist had the opportunities to increase it revenues, it never compromises the experience of its users. And because it is able to operate cheaply and let the users do much work, craigslist has only about 20 employees and several orders of magnitude less than other top-ten sites.

Though eBay purchased a 25 percent stake from a former craigslist employee in 2004, craigslist remains a privately held company. It continues to expand as we speak and now has sites for over 300 cities worldwide.

I like this story of Craigslist, because I thought at first that their ideas were not going anywhere. Buy as it progressed, we can see what the developments were done in order for craigslist to be successful. The interview with Craig Newmark is quite short compare to other start-up developers, I think he is being concise about how he started this company and just elaborate important factors to consider in building a start-up. For him, the importance of trust is important, this can show how good of a leader can you be and for him having trust is a golden compass.

Case Study (Flickr)

Flickr started with Caterina Fake. Caterina Fake is married to Stewart Butterfield. When they met, Caterina was living in San Francisco while Stewart lived in Canada. Both of them were doing web development at that time and his idea was that they do something some type of transitional web development company which is kind of a scheme. Caterina and Stewart had a long-distance relationship. Caterina eventually moved up and they got married.

After their honeymoon, two days later they started Ludicorp. The real name is ludus, a latin word which means "play". They were building a multiplayer online game called Game Neverending. It was a lightweight web-based game, and a typical for massively multiplayer games. Most of those have sword and sorcery or fiction themes and are usually CD-ROM based. Neverending was very much based around social interactions that you could form groups, instant message each other and a social network associated with it. When they came up with the idea for the game, Stewart had been working at the CBC on the children's site. In doing research, he started playing all these online games.

Neopets was one of the inspirations for Game neverending. They have these pets which are Tamagotchi-like and you can buy those presents and give them toys. Both of them have backgrounds in web design and web development. Caterina has a focus on social software. Before Ludicorp, she worked on or participated in online communities including the WELL, Electric minds, the Netscape online communities and various sites she started on her own. At interval research, she worked on a collaborative animation game, which was a cousin to the Game Neverending idea.

At the beginning, it was Caterina, Stewart and Jason Classon. Jason and Stewart had a company together in 1999 that was acquired by a venture backed startup out of Boston after about 6 to 9 months. Jason went and worked in Boston for a year and came back and then the three of them started working on the game together. Caterina did the game design, Stewart did the interaction design and Jason did the PHP for the prototype. In 2004, they added a new feature, a chat environment with photo sharing; which quickly surpassed Game Neverending itself in popularity. With its emphasis on user-generated content and its devoted online community, Flickr is one of the most commonly cited examples of Web 2.0 companies.

I like the story of how Flickr stared, because at first they were developing games which is interesting because I like playing computer games, when they were able to come up with the idea of Flickr, I was amazed because I didn't expect them to comp up with that kind of idea, I thought that they will just focus

on developing games. But now, Flickr is one of the examples of Web 2.0 company, that's how successful Flickr has become.

(Case Study) WAIS, Internet Archive and Alexa Internet

The thinking machines teams were founded by Danny Hills and Sheryl Handler. Brewster Kahle was on the project team at MIT. They had been working for couple of years before there was a company. They did a full couple rounds of the connection machine at MIT before they started the company. For them, it was very helpful to get your lessons learned basically on somebody else nickel, in a research phase. Another lesson Kahle learned was if you are trying to get your company to think differently, to do something interesting; pick your setting carefully. Thinking machines was set in an 1800's Victorian mansion on 100 acres of forest just outside of Boston. It was a park, working in an environment where, if you got stuck, you'd go for a long walk is very different than trying to do a startup and think differently if you're in Suite 201 in some major office complex.

Thinking machines had the great fortune of starting with $8 million in the bank because some rich individuals really believed in it. It was not venture funded and it was not founded with the idea that it was going to take years to actually get something real done. It allowed thinking machines to attract a very interesting set of people. People who worked there were Richard Feyman, Stephen Wolfram and Marvin Minsky. They hired a person from Digital Equipment Corporation and he was VP of reality. It stirred up the culture to try to get it so that they could actually produce working machines.

The idea of WAIS was to make network services, stuff that you take completely for granted now, but the idea was that you could use remote machines to answer questions. Brewster started WAIS in the late 80's while an employee of thinking machines. He left in 1993 to found WAIS. It is one of the earliest forms of internet software developed before the web. Kahle sold WAIS to AOL in 1995.

I like the story of WAIS because their idea is unique. They were able to create a site where you can compare each sites to other sites. I think this one is one of the most interesting start-up that I've read.

Case Study (Adobe Systems)

Adobe systems was founded by Charles Geschke and John Warnock, who established this company after leaving Xerox PARC in order to develop and sell the Postscript page description language. During their stay at Xerox PARC they were able to develop a language called Interpress that would allow any computer to talk to any printer. When the Xerox PARC is slow to commercialize this technology they decided to establish Adobe which produced the successor of Interpress called Postscript.

Apple computer licensed Postscript for use in its LaserWriter printers, which helped spark the desktop publishing revolution. Adobe went public on 1986 and is the recognized industry leader in graphics and desktop publishing software through its typefaces and its popular Photoshop, Illustrator, and Acrobat applications.

Charles Geschke parents thought that he lost his mind because he had a great job at Xerox; he even had a big nice office overlooking the whole Bay Area. He just explained to them that he will try something new and branch out on their own.

He also said that Adobe is getting harder as it grows bigger. So they came up to a plan to reinforce the current CEO, Bruce Chizen, the importance of innovation and the importance of taking some investment of the company and not immediately pouring it back into the current business. His advice for those people who is thinking about a startup is one must be passionate in his/her work. One must work smart and not long, because for him one must preserve his/her life, not just on work life.

I like the story of how Adobe was started, because they showed courage that they can do it by themselves. We can easily identify that they are passionate people; once they want to accomplish something they really work hard in order to achieve that goal. Now, a lot of people are using Adobe especially Photoshop… thanks to them, editing files and pictures can easily be done nowadays.

Case Study (Open Systems and Hummer Winblad)

Ann Winblad was taking double majors, major in mathematics and major in business administration in College of St. Catherine. Then, she had also interested in computer science and acting as well. During her masteral degree, she got this job at Federal Reserve Bank. So, every weekdays she had her masteral degree and every Saturday she had her job at Federal Reserve Bank. Later on, she decided to start a company with the three guys from the Federal Reserve Bank. At first they had no money to invest but thanks to the Y Combinator and her brother borrowed them $6000 and $500. Later on, they were chosen under a Request for Proposal bid to build a student accounting system for a vocational school in the state of Minnesota, which can test their accounting knowledge. With the help of these three guys who were working for the Federal Reserve Bank for 3 and ½ years they got this idea. Although Ann had a limited experience in her job, she still determined to do some new things. They decided to program and build accounting systems for smaller computers. When they were setting some discussions with the CADO computer guys, Ann had the opportunity to talk with these guys. With her credibility and accounting skills she got about $10,000 from the checks of these 15 guys which was a big help for them.

They had some problems and challenges to cope with. One of these problems was the fire started on their office. Fortunately, all their computers including software and hardware were safe but other stuffs of Ann was burned by the fire. They also had problems with pricing strategy and how they collect money from people. Prior to co-founding Hummer Winblad Venture Partners, the first
venture firm to focus exclusively on software in 1989, she served as a consultant for clients such as IBM, Microsoft, and Price Waterhouse. Ann Winblad and her colleuges sold their company for more than $15 million.

Most of the startup companies, their main goal is to change the world and help people for their convinience. As Winblad said as a business investor, "Your job is not to tell, but to teach." Also, she stated that if you have your own company, the best way to succeed is to think for the best strategy because in running a company, there is a possibility of failing and running out of money. So the best way to cope with these complications you should do the best gameplan or strategy.

Since Technologies are growing fast and changing over a period of time. Through the help of technology, people can have convenience on what they are doing, on what they are wanted to do and other stuffs. Nowadays, we have this

ecommerce to help business people and customers to have transaction online. A good example of this technology is what Ann Winblad created, the Open System, an accounting software company.

Case Study (37 Signals)

The web is now on changing and growing generations. Different web applications are now implemented to do a certain thing like in business firms, education and things how to socialize with other people. 37 signals is a good example. 37 signals is basically a privately held web application company. David Heinemeier Hansson was one of the founders lead the company launched. 37 signals was actually founded by Jason Fried.

Before the 37 signals was released, Hansson developed first the Basecamp. Hansson was part of the 37 signals 2.0 management team. With this Basecamp, they picked simple things: a project weblog, milestones tracking, file and to-do list sharing. At first they had this idea of making blog and applied it to project management. In developing the Basecamp, they try to do it to a simpler application because they would probably have some complications with the codes. Hansson was actually the only programmer and designer who had primary concerns of the applications. So, they decided to do it to simpler programs. While doing the Basecamp, Hansson was able to have an opportunity to develop Ruby on Rails.

They had some technical and bank problems encountered. In the bank issue, before they had the billing system of yearly. They didn't figure out that the bank wouldn't let them bill that way until about 3 days before they were ready to launch. The bank wouldn't let them sell a service that they were going to promise for an entire year, because they'd be on the hook for the money if they went out of business a few months into a $500 agreement. They wouldn't allow that because they didn't have a long history with them. In the technical issue, they had problems in creating services firms. They had a year and a half to fix this problem. Time was also their primary problems because some users in different countries can't see the accurate time of the files that they were created. They had a long period of time on fixing this particular problem.

Most of the startup, their business plan is to do 100 percent of the problem in order to help people. In Hansson's situation, they did the 80 percent of the problems and they did some simpler solution to the problem.

Case Study (ArsDigita)

ArsDigita is a name that I have never heard before. I had no idea what kind of start up it was. I don't have any idea also on which start up can we relate ArsDigita except from Fog Creek Software. ArsDigita was founded by Philip Greenspun. Philip Greenspun was the inspiration of Fog Creek Software that's why it became successful. Philip Greenspun started building internet applications in the early 1980's. He liked developing multi-user applications. He thought of connecting people over the network if they were separated by space and time. Unfortunately that time, it was hard to write popular applications because whatever you do, only one kind of computer system will function. If you are using Apple Macintosh, and you want to connect to a network and edit a document together or play games with Windows or any other Operating System, it would not work, because at that time, there was no standard operating system and no real programming environment.

Then there came the Web in 1993. That was the time when Philip started to build computer applications. He doesn't need to write custom code to the operating system anymore. He will create that is already specified on the server side and user experience will be rendered by the browser. One of the turning points in ArsDigita was when they got Levi Strauss as a customer. They had acquired a small company that made custom-cut khaki pants and they wanted a web front end for this new factory that they were building that could take your measurements and sew you a pair of khakis to your specs. For them, it was one good thing about when working for non-technical companies. Levi made a mark for the programmers.

ArsDigita grew out of the software that Philip Greenspun wrote for managing photo.net, a popular photography site. Philip released the software under an open source license and was deluged by requests from big companies for custom features. What I like about this startup is that it was very successful. I really idolize programmers because of their hard work. ArsDigita was successful because by 2000, the company grew fast and earns about $20,000,000 in annual revenue for its monthly contracts. On the same year, it took $ 38,000,000 from venture capitalists. In the year 2002, ArsDigita was dissolved, but not before establishing an important new model for the consulting business.

Case Study (Fog Creek Software)

When I first heard the name Fog Creek Software, I had no idea what that was, because for me, it is a new and unique start-up. Fog Creek Software were founded by Joel Spolsky and Michael Pryor. They were friends in Juno Online Services which was co founded in 2000. Joel on Software was a one of the most widely read programming blogs. He made blogs to share his thoughts, about building software, about management, business and about the internet. Joel on Software was one of the first examples of a common strategy for software startups. The key inspiration for this start up was Philip Greenspun of ArsDigita. He had a particular business plan that seemed to work at that time. They had this point of view that they were doing a lot of great stuffs but without the use of Microsoft technology. For them, it was already an achievement.

ArsDigita Community System is the product that they were developing. The theory about this system is that the product that they created would support the consulting and the consulting will support the product. The product needs to be open source. But consulting is a business where your revenue is just a multiple of the number of people you can hire. The first application they made was Fogbugz. An internal bug-tracking application which made people interested and started to buy the product. The second application made was CityDesk, which was a market failure. The third product was Tintin, that was never wrote and let alone shipped. Because of what happened to their products, they created the idea of combining the three applications that would work in various ways. Fogbugz would provide workflow, Tintin will provide content for management server and CityDesk was going to be this content management client. That was their long-term vision because they had Fogbugz.

They started making $5000 to $10,000 a month selling the product. It was enough for them to pay their expenses and live off once they lay off two consultants they hired. Luckily, one of them is back to a job who works full-time. What was good about this start up is that they never had an outside investment. The money came from the founders itself. At one time, it should be programmers who are the stars. Coding maybe is the hardest part of making or developing a system. It is the soul of the program. Marketing, designing and analyzing the system are secondary parts in making the product. When you can't code, then everything will be a failure. I hope that in the future, they will be more startups that programmers will be the stars.

Case Study (TripAdvisor)

Today, internet maybe is the best way to gather different kind of information. One of the useful website is TripAdvisor. TripAdvisor is a free travel guide and research website that offers reviews and information to help plan a vacation. This company was founded by Steve Kaufer, Langley Steinert, Nick Shanny, and Thomas Palka. Kaufer and his wife were looking for a place for a vacation. They got some travel agents and they were not satisfied with their offerings and got frustrated on some unbiased information. So, later on they came up with the idea of you could build a better search engine to find what you're looking for in travel and not the published opinion, but the unpublished, unbiased opinion about a place, a location, something to do.

They made the idea of they can help other people with relevant and detailed information of the place they want to go. The idea was that when you search a place all possible and relevant information about the place will appear. With this startup, people can have the convenience and get some detailed information of the place they want. This startup can also help people to access or do other stuff regarding with the place they searched. They got some deal with some company like the Lycos and Expedia. Unfortunately with Lycos deal, their quarterly revenue check wouldn't buy the weekly free lunch that they offered to their employees. On the other hand, they had some link with the page of booking of Expedia. Fortunately, it worked and was good with the link. They actually got a lot of people clicking. 10 percent of the time that people saw that page, they were clicking on one of those links. Click-through rates at the time were a quarter of a percent or half a percent. There, they were sitting at 10 percent because the links were so relevant to the topic at hand. Expedia was their first client. This was also led them for a good path.

In doing this particular startup, of course they had encountered some challenges to cope with. One of the common problems in startups and their basic problems was marketing and business development. Although some founders and their cofounders were good in programming and some sort of engineering, they still had no any idea of business stuff including selling, improving the management, motivating their workers, etc. This startup was different at that time, because some sites in the web which was primary concern was kind of travel stuffs. I agree with Kraufer's advice, look for the better cofounder because they are one of the factors of being a successful company. In, Kraufer's situation, he got Langley Steinert to help him to launch TravelAdvisor. TripAdvisor became the largest online travel community in the world, and was acquired in 2004 by Barry Diller's InterActiveCorp.

Case Study (HOT or NOT)

A good example of a successful startup was the HOT or NOT. This was found by James Hong, his brother and Jim Young. Jim was working on his PhD. James was just graduated form business school was actually unemployed. They thought the idea just for fun but they had no any business plan did with their startup. The idea was actually started when the three of them were hanging out drinking and there they met a girl that was so hot and their rate for this girl was a perfect 10.

James Hong and his brother were actually working at XMethods. Then, they thought that they will do some stuff just for fun. They were basically had an idea of people could post their pictures into the system and then other people could rate them form 1 to 10. It coded immediately and they got 40,000 hits after they launched it. People got interested with their site at that time because they were looking for hot chicks. No one had just seen it before and thought about it. This was a big impact for the people because they got interested with the pictures. And this was led to certain problem for them. Some people used and uploaded some pornographic pictures on the site. They had a difficult time to fix this particular problem.

So, pornography was one of the problems they encountered. They had this idea of doing the "meet system", allowing the people to meet each other without letting all these porn people in. this system required a little more work than most dating sites because it requires both people to be active. Another challenging moment in doing the project was the site's bandwidth driven by the huge pictures. People got interested with the site, everyday they got more views from these people so the bandwidth got slower and slower. James and Jim got to sleep 8 hours in 8 days to fix this technical problem.

For me, I like how they were able tom come up with the concept of Hot or Not, because it was unique. At first, I didn't know anyone can think about this concept. They encountered problems because they didn't make a business plan from the moment they started doing this start-up. That's why business plan is important. Finally, doing entrepreneurship is always difficult.

Case Study (Tickle)

Tickle was one of the companies that had the idea of not just to make profits and money but to help people in other stuffs through the online community, led by James Currier, a former venture capitalist with a passion for digital media and social sciences. They had the idea of helping people to learn more about themselves which led to an online testing company.

James Currier worked at Star TV in Hong Kong and did more digital media and got back to venture capital in Boston where the company was investing in the early stage Internet companies like Infoseek. Before making any product, James consulted many people for assistance. Trying to find out what product will he be doing? The spark point or climax of Tickle was when they launched the dog test. When they started the company, they have this motto of changing the world. They had all these tests on the site to help people about their lives. They had the anxiety test, the parenting, and relationship and communication test. And no one came. Because of that, they made a fun test. They made it so that people will or can remember the advertisement which includes babies and puppies. They made a test of the kind of dog breed. They also came up with a 15-question test but not scientific.

In doing this particular project, Currier also encountered come challenging problems. One of these problems was other people won't understand what he really do with the startup. When he was asking other people's idea with his startup, people might confused and misunderstood with his ideas. Another problem, he encountered was several steps that people asking him and he also confused on what things to do first to help improve the idea and the business plan too. So, he had difficult time to fix these problems.

All those problems vanished after a while, the company's success followed due to James' hard work. Based from my reading, Tickle is an interesting and a unique startup. Many people will benefit from this kind of start-up. Hopefully, someday one of us or some of us will able to do this and do not conflict our time with other people.

Case Study (Firefox)

Based from my experience about the internet, I have a lot of ideas about Firefox. Because it is a known browser worldwide aside from Microsoft Internet Explorer. The co-founders of Firefox were Blake Ross and Dave Hyatt. As we know, Firefox grew out of Mozilla. Blake Ross started working on the Mozilla project in 2000. It was an open source software that anyone could work on it. Blake Ross started with the help of the Netscape team. They were basing their product on Mozilla and Blake was helping them fix all those bugs. And was invited out for an internship one summer. Blake was only 14 yrs old at that time. After working in California, he worked from home then came back next summer.

Netscape kept sliding further in the market. They had something like 5% market share. It got worse when AOL tanked and started to demand more revenue from the browser. They wanted a return on investment. They bought Netscape for about $4,000,000. Firefox was a solution to their experience at Netscape than the Microsoft Internet Explorer. IE had been dissolved at that time in 2001. Microsoft disbandled the IE team. By that kind of scenario, they started Firefox as a way to work on a browser that we knew we could make if we they were not controlled by sales, marketing and others. Firefox started with only four members on the team. They were David, Hyatt Joe Hewitt. Blake was focusing on the development side, with Brian Ryner and Asa Dotzlec providing build and Quality Assurance support. Before it was named Firefox, it was first called Phoenix. After that, they encountered problems. They had trademark issues. Phoenix technologies complained because they had some kind of web browser also. They renamed it Firebird. It has the same imagery but there was an open source database already called Firebird. They renamed it Firefox, a Chinese name for red panda.

All of the firefox developers came from different places. Some were from France, others were from New Zealand, Pierre Chanial from France and Jan Varga from Slovakia. Firefox was a different startup from traditional or usual startups. Because companies usually worry competition for financial reasons, but when they did Firefox, money was just a sort. There were donations, seed money from AOL, etc. It was actually not for money, but for a hobby. Firefox was something new, it was an open source project both concern for the end-user and in the attention paid to marketing. This is one of my favorite startups. Because this is known worldwide and being used by many people. I really idolized the founders of this startup. I also like Firefox because not like Internet Explorer, you can open a tab while other program is running. That's what I like

about Firefox. I want to thank the people involved in making this product because they made it easier for us users when we are browsing the web.

Case Study (Six Apart)

Six Apart was co-founded by two people. They were Mena Trott and Beb Trott. They were couples. They named their startup Six Apart because of the number of days between their birthdays.

Mena started a blog called Dollarshort in 2001. She did the blog because she felt that she needed a creative outlet. It was her job to write a blog. After creating her blog, she noticed that the blog that she made was getting more and more popular.After their company closed, they started working on a blogging tool. When Ben and Mena were in college, they created Movable Type, it became very popular very fast and it became a full-time job. After that, in 2002, there was a big demand for what the couple was creating. Movable Type really became popular. After Movable Type in 2002, they formed ALC in July of 2002, right before they decided to start doing TypePad. But they had a problem, they don't have funding. They spend their time in their apartment. They used spare bedroom and their desks are literally back to back. They totally spend their time in that apartment for 18 months.

Of course, in making startups, there are always problems before you can achieve your goal. In this situation, the reason why Mena started her blog is because she thought and felt like she did not have any friends. When Mena and her husband were in High School, they did not force themselves to make friends with other people. Because they have each other to cope with. Mena wanted her blog to have a connection with people online, all these people that she wanted to be friends with. In their blog, they had two factors why the people should donate, first was they like and appreciate the product, second is that they give recently update keys. In December of 2002, the couple met Joi at the Supernova conference. Today, she was Six Apart's CEO.

What I liked about this startup was the story between the couple. That they did not have friends before, but after creating their product, they had a lot of friends. Not only that they had did had many friends, they also had business and income due to their hard work.

Case Study (Lycos)

The co-founder of Lycos was Bob Davis. Lycos started when the technology was invented back in 1994 by a brilliant computer scientist at Carnegie Mellon University (CMU) named Michael Mauldin. His nickname was Fuzzy. The thing they were doing was a research project. It was Fuzzy all alone in a closeted office at the research lab at CMU. He worked with CMU's technical transfer office to try to sell the technology. While he was there, he met Dan Nova of CMGI which at that time was small, $35,000,000 venture capital fund and grew into one of the most successful internet investment firms of its era.

CMGI's venture was founded by Dave Wetherell. He acquired 80% of the company and 20% of it continued to be owned by a combination of Fuzzy and Carnegie Mellon, 10% a piece. When building Lycos, their first job on their first month was about building a team, getting some core people in place trying to understand what they were doing for a living and how they were to go about doing it. When they were building and licensing their technology, they coined the phrase " Technomedia". It is their branded site, selling advertising. They were fortunate because Carnegie Mellon gave the group a good draw of students, postgraduates and alumni in that area. They hired their first few technical engineers out of CM. One was Fuzzy's assistant and another that was working in their data tabs. They peaked 300 employees in Pittsburgh and certainly substantial pieces of their engineering operation were there.

Chapter 3 – Use Cases

Tin Number

Shows birth certificate

Checks birth certificate

Fill up application form

Gives application form to be filled up

Submits application form

Gives TIN number to the applicant

Pay amount

Applicant
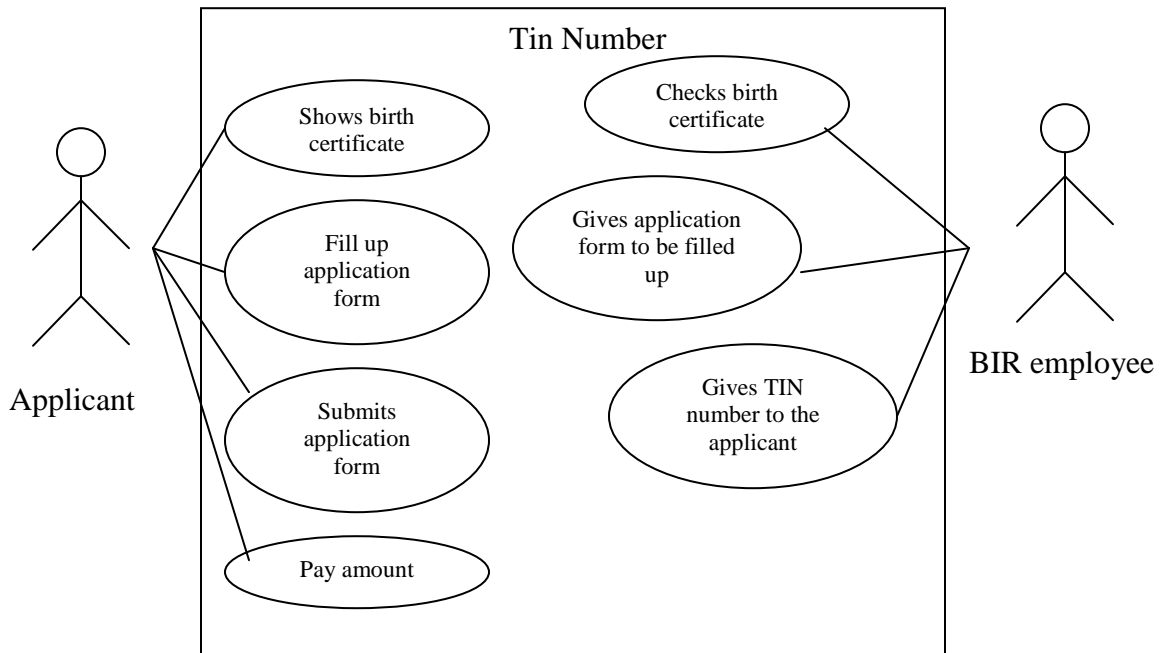
BIR employee

Identification Summary:

Title: Getting a TIN number
Summary: This Use-Case Diagram shows how you can get a TIN number from a BIR office
Actors: Applicant, BIR employee
Creation Due: July 9, 2008   *Date of Update:* September 12, 2008
Version: 1.0   *Person in Charge:* Nelson O. Guinmapang Jr.

Flow of Events:

     Preconditions:
         1. The BIR office must be open

     Main Success Scenario:
         1. The applicant will show his birth certificate
         2. The applicant will fill-up the application form
         3. The applicant will get his valid TIN number

     Alternative Sequences:

A1: The applicant has not enough money to pay the amount for the TIN number
        The BIR employee will not give the applicant his/her TIN number unless he/she pay the amount.

A2: The applicant put the wrong necessary information in the fill up form
        The BIR employee will nullify the form and let the applicant fill up another form.

Error Sequences:

E1: The applicant has no birth certificate
        The BIR employee will not allow the applicant to get a TIN number unless he/she shows his/her valid birth certificate.

Post Conditions:

1. The applicant will get a valid TIN number
2. The BIR record of TIN number will be updated.

UI (User Interface) Requirements:

- application form

Non-Functional Requirements:

- The BIR office is open Mondays to Fridays, from 8:00 am – 4:00 pm
- The BIR places its records in a secure location

Barangay Clearance
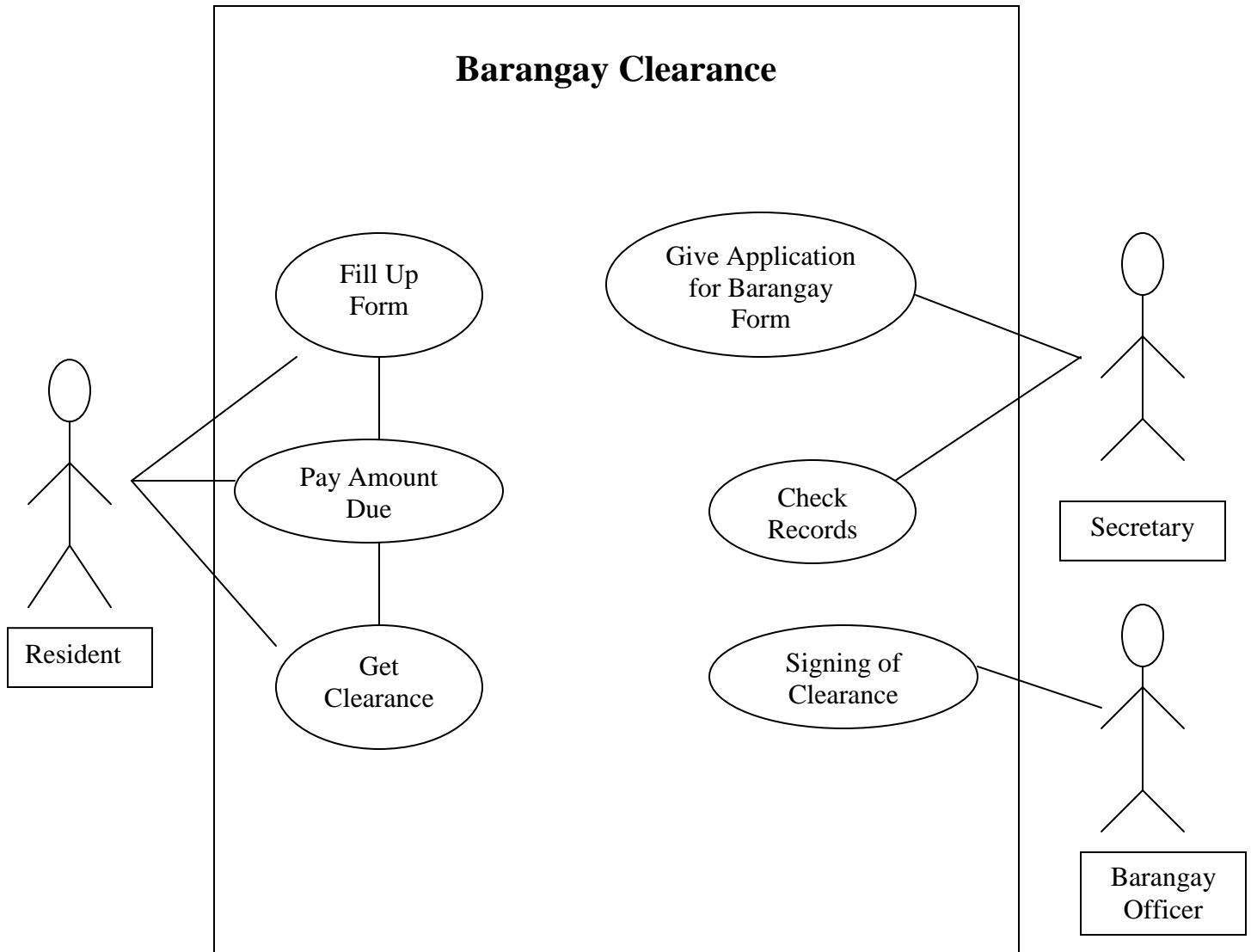
Identification Summary:

Title: Barangay Clearance
Summary: This Use-Case Diagram shows how you can get a Barangay Clearance at
Brgy. San Antonio, Sucat, Parañaque City
Actors: Residents, Secretary, Barangay Officer
Creation Due: June 11, 2008   *Date of Update:* September 12, 2008
Version: 1.0   *Person in Charge:* Nelson O. Guinmapang Jr.

Flow of Events:

Preconditions:
2. The Barangay office must be open
3. The residents must bring money to pay a certain amount

Main Success Scenario:
4. The resident will give filled-out application form to the secretary
5. The resident will get his Barangay clearance

Alternative Sequences:
A1: Resident has no Barangay record
   The secretary will not allow the resident to get a Barangay clearance

Error Sequences:
E1: The resident lives in their Barangay illegally
   Barangays don't allow illegal aliens to get a Barangay clearance

Post Conditions:
3. The resident will have an approved Barangay clearance
4. The residents record will be updated

UI (User Interface) Requirements:

- application form

Non-Functional Requirements:

- The Barangay hall is open Mondays to Fridays, from 9:00 am – 5:00 pm
- The Barangay keeps the records in a secure place
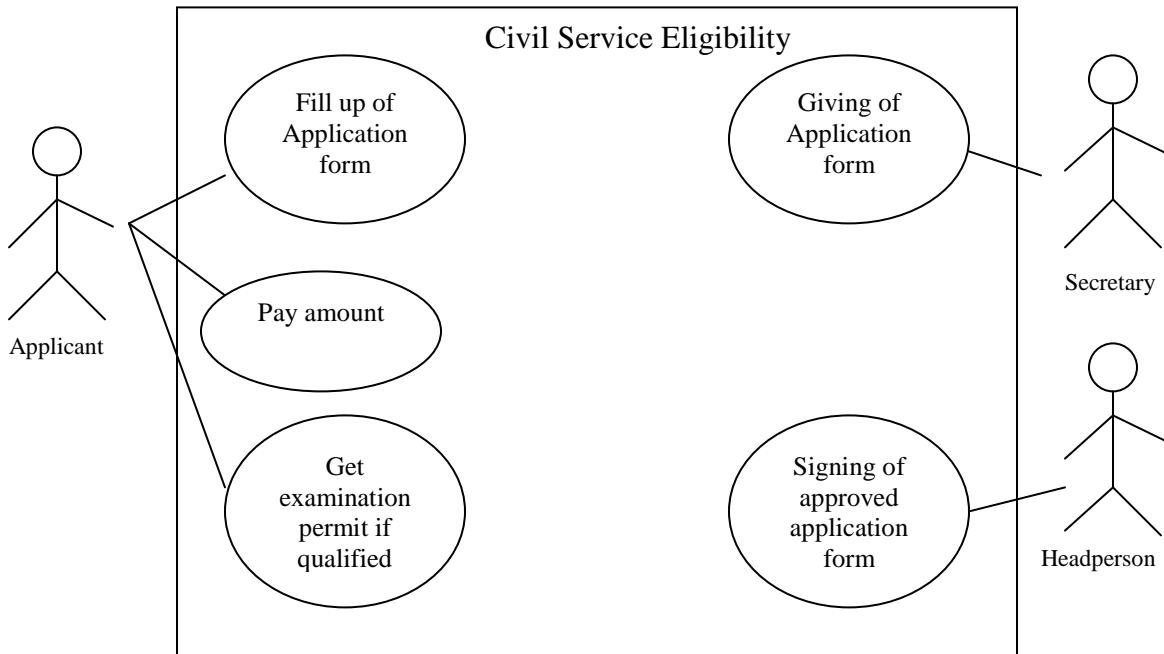
Identification Summary

Title: Civil Service Eligibilty
Summary: This use case shows how to apply for getting a Civil Service Eligibilty
Actors: Applicant, Secretary, Headperson
Creation date: June 25, 2008   **Date of Update**: September 19, 2008
Version: 1.0   **Person in charge**: Nelson Guinmapang Jr

Flow of Events

Preconditions:
1. The government must be open
2. The applicant must present a birth certificate for checking
3. The applicant must have money for payment

Main Success Scenario:
1. The resident had filled up the application form
2. The headperson must approve the application
3. The applicant will have a test permit given to him

Alternative Sequences:
   A1: The applicant is not qualified to take the civil service eligibility test
   1. The applicant has something wrong in his/her file.

Error Sequences:

E1: Rejected application form
      1. The applicant will not receive a test permit from the center

Post conditions:
1. The applicant will know whether you were qualified or not to take the test
2. The person taking the test has been added

Non-Functional Requirements:

- Response time: The applicant can only go one transaction at a time.
- Availability: The office is only open during Monday to Friday.
- Integrity: The secretary make sure that the form given by the applicant is secured

CSB Enrollment

Log-In

Pre-Enroll

Get EAF from the Registrar

Approving of enrolled subjects

Enroll

Pay at the Accounting

Counting of amount paid

Get approved EAF at the Accounting

Student

Academic Adviser
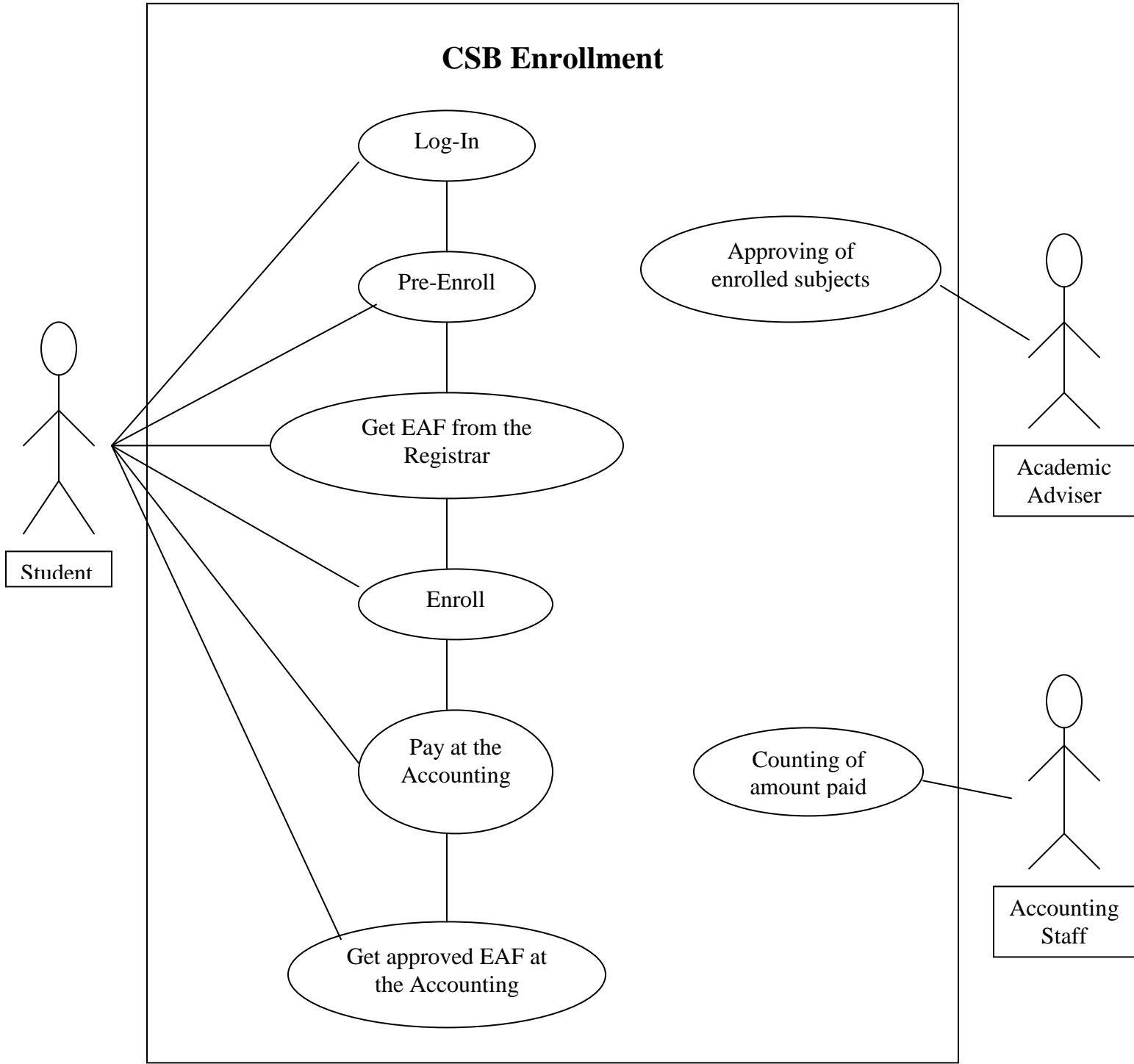
Accounting Staff

Identification Summary:

Title: CSB Enrollment Procedure
Summary: This Use-Case Diagram allows students to enroll at DLS-CSB
Actors: Students, Accounting Staff, Academic Adviser
Creation Due: June 4, 2008   *Date of Update:* January 2009
Version: 1.0   *Person in Charge:* School Administrator

Flow of Events:

Preconditions:
4. The school must be open to enrollees
5. The computers must be on
6. The registrar must be open
7. The accounting office must be open

Main Success Scenario:
6. The student enlists the subjects he intend to enroll
7. Student pays in cash/check the amount for the enrollment
8. Student gets his approved EAF

Alternative Sequences:
A1: Overload Units

   The academic adviser will not allow the student to enroll if they exceed the maximum units.

A2: Failed Pre-Requisite Subjects

   The academic adviser will not allow the student to enroll if he had not taken their failed pre-requisite subject.

Error Sequences:
E1: Forget EAF

   The enrollee forget his/her EAF, the accounting cannot do transaction. Use case fails.

E2: Check is not accepted (Payment)

   The accounting does not accept "bouncing check".

Post Conditions:
1. The available slots for a block becomes less
2. The student is enrolled for the term
3. The accounting will receive a lot of money

UI (User Interface) Requirements:

- Student Information System (SIS)
- Username and Password

Non-Functional Requirements:

- The students are given a time to enroll depending on their course
- The accounting office and registrar is open 8:00 AM until 4:00 PM
- The registrar sends a letter to the students that contains all his grade for the term
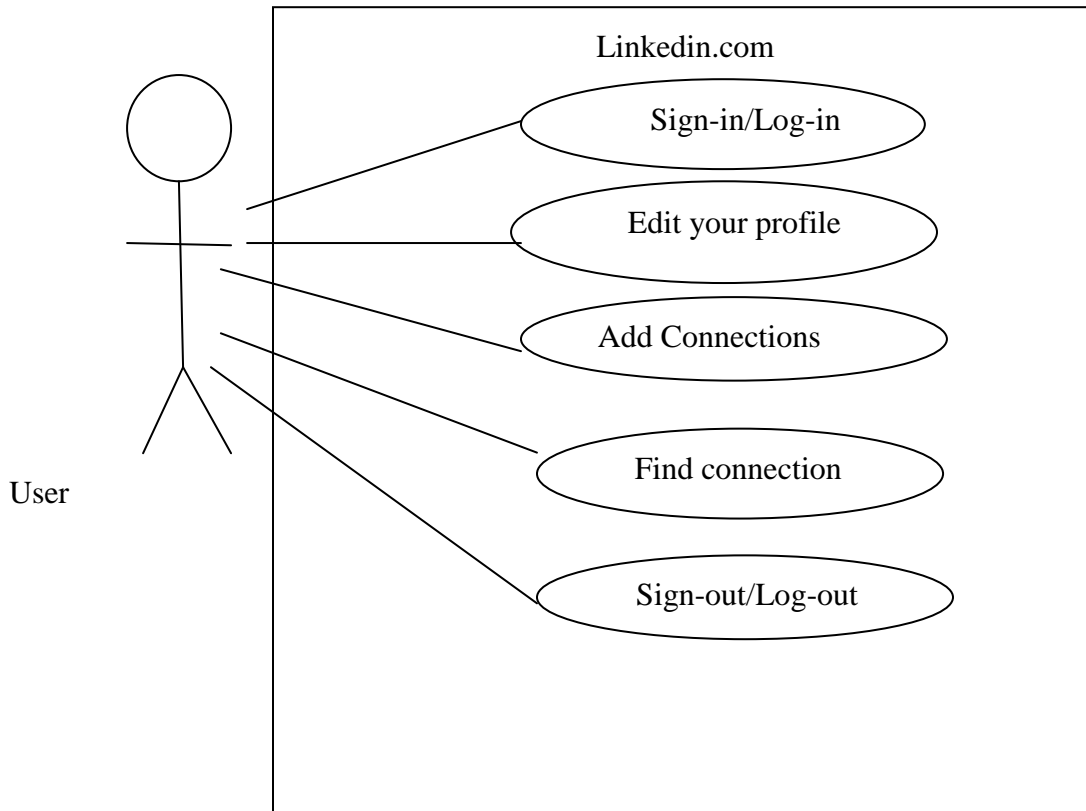
Use Case Narrative
Title: Sign in to Linkedin.com
Summary: This use case allows a user to log in his/her account at linkedin.com.
Actors: User

Creation Date: July 23, 2008   Update Date: December 2008
Version 1.1   Person-in-Charge: Nelson Guinmapang

Flow of Events
    Preconditions:
- The user has an account at linkedin.com.
- The computer is connected to the internet.

    Main Success Scenario:
    1. The user goes to the home page of linkedin.com.
    2. The user clicks the sign-in button.
    3. The user inputs his/her email address and password.
    4. The system checks if the email address and password inputted by the user match an existing account.
    5. The user waits for the confirmation.
    6. The user is now in his/her account.

"Alternative" Scenario

A1: Email address and password did not match

- The site notifies the user that the email address and password didn't match. The user will retype the email address and password.

A2: Remember me

starts at point 4 of main success scenario

- The user chooses if he/she wants the site to remember his/her email address and password.

Error Sequences

E1: System Maintenance

-user will not be able to access his/her account due to system failure.

Post Condition

- User can access his account
- User can edit his account
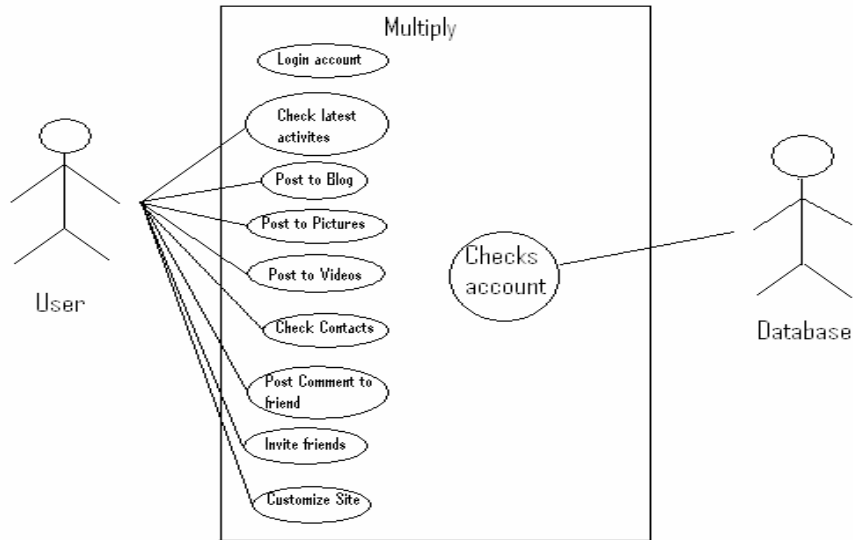
Use Case Narrative (Post to Blog)

Title: Post to Blog using Multiply.com
Summary: This Use-Case Diagram shows how to post a blog using Multiply.com
Actors: User, Database
Creation Due: July 15, 2008   *Date of Update:* December 20, 2008
Version: 1.2   *Person in Charge:* Nelson O. Guinmapang Jr.

Flow of Events:

   Preconditions:
8. The user must have internet connection
9. The system must be online

   Main Success Scenario:
9. The user inputs his valid multiply account
10. The database checks if the account is valid/active
11. The user clicks the post blog/journal icon
12. The user types his/her journal
13. The user now has a blog/journal on his homepage

Alternative Sequences:

    A1: System Maintenance

        The system is currently offline due to fixing some bugs in the site

    A2: Incorrect Password

        The site will tell the user that he/she has inputted the wrong password for his/her account

Error Sequences:

    E1: No internet connection

        The user cannot go to the site if he/she doesn't have internet connection

    E2: No Multiply account

        The user cannot post a blog/journal is he/she does not have a multiply account

Post Conditions:

5. The user has now a blog/journal on his/her page
6. His/her contacts can now view his blog/journal

UI (User Interface) Requirements:

- Multipy.com

Non-Functional Requirements:

- The website is available 24 hours a day, 7 days a week
- All the account is secured in a database
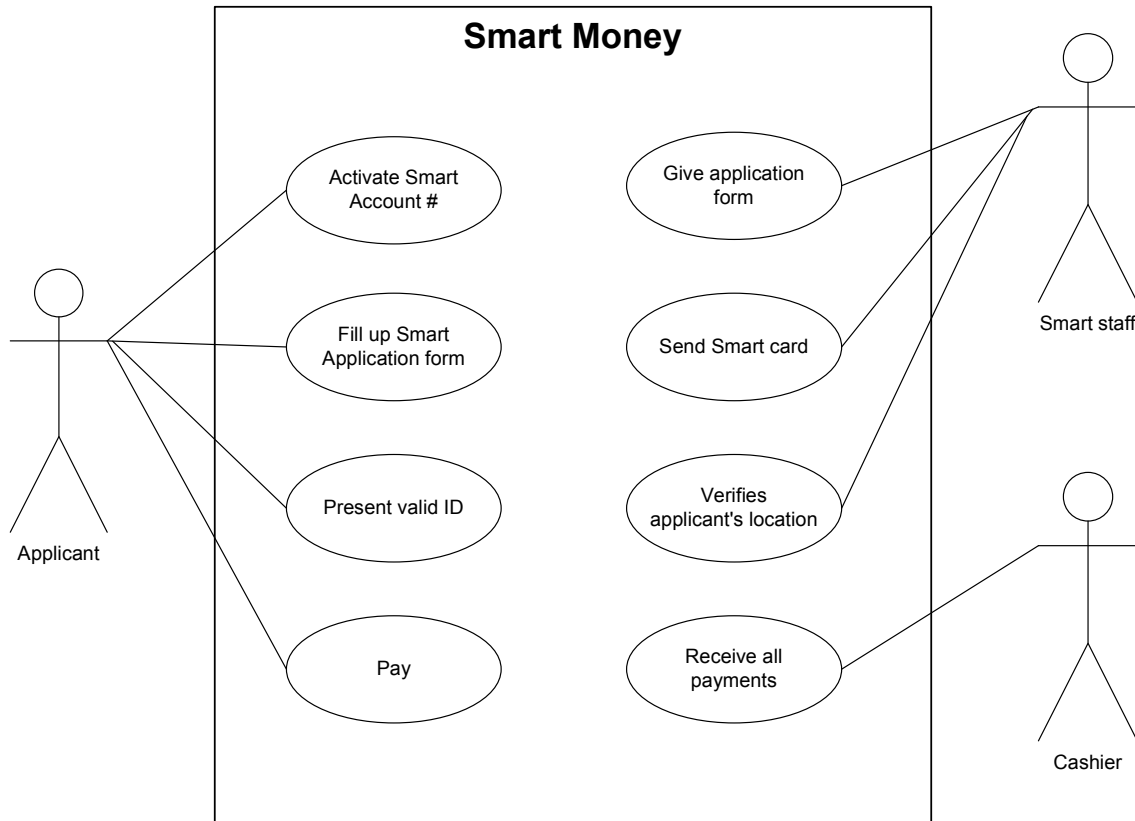
Identification Summary

Title: Smart Money

Summary: This use case allows the applicant to apply for Smart Money Card

Actors: Applicant, SMART Staff, Cashier

Creation date: July 16, 2008   **Date of Update**: September 2, 2008

Version: 1.1   **Person in charge**: Nelson Guinmapang Jr.

Flow of Events

Preconditions:

4. Applicant must be a Smart Subscriber
5. Applicant must be 18 years and above
6. Applicant must have a valid ID

Main Success Scenario:

4. Applicant activates his/her Smart Money Account via text message
5. Employee gives application form to be filled up
6. Applicant fills up the form and present valid ID
7. Employee verifies and checks the form and ID
8. Employee issues amount of 30 Php to the applicant
9. Applicant pays the amount in the cashier
10. Cashier issues receipt
11. Smart staff sends the Smart Money Card to the applicant's address

Alternative Sequences:

A1: Invalid ID
   2. Employee informs the applicant that his/her ID is invalid or has incomplete information

A2: Incomplete information
   a. Employee sends back the application form to the applicant for verification.

Error Sequences:
   E1: Applicant is not a Smart subscriber
   1. Applicant must be a Smart user to be able to access Smart Money
   E2: Applicant's age is 17 years old and lower
      1. Employee informs the applicant that he/she is not able to apply for Smart Money Card
      2. Minor applicants can access once Smart Money only through their sim card.

Post conditions:
   3. Applicant received his/her card via mail
   4. Applicant's information card was added to the databases in Smart administration

Non-Functional Requirements:
   - Availability: The SMART Wireless Center has branches in different parts of the Philippines
   - Integrity: The staff of the SMART Wireless Center makes sure of the security of inventories regarding with the applicants' information.
   - Confidentiality: The transaction between the applicant and staff is done privately.
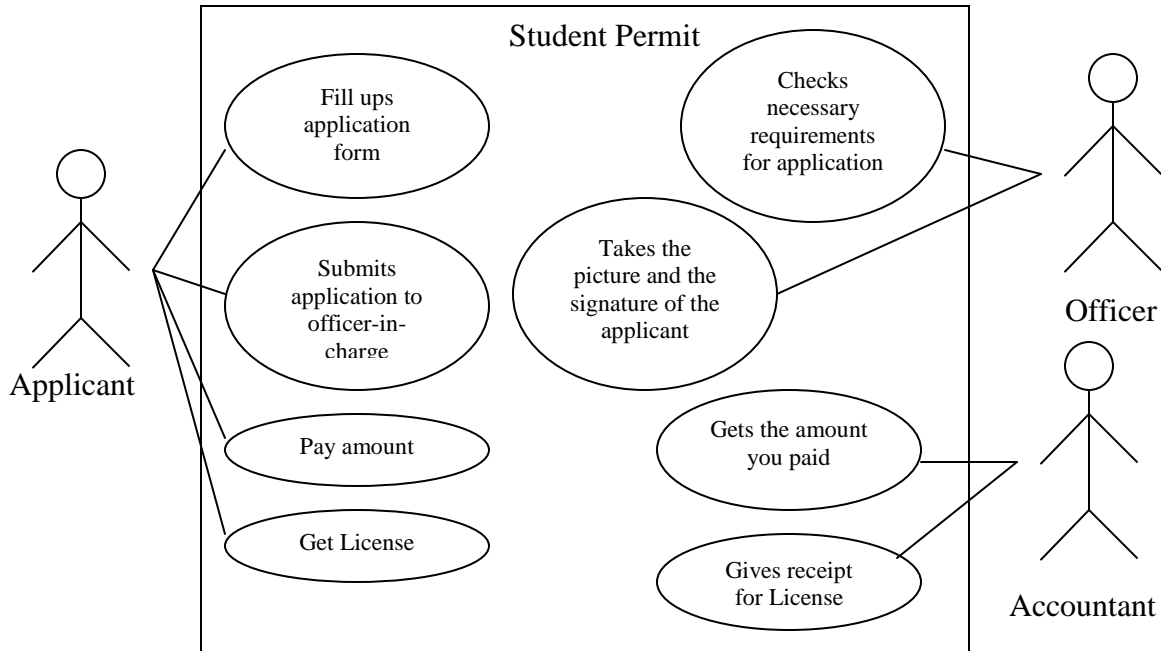
Identification Summary:

Title: Getting a Student License
Summary: This Use-Case Diagram shows how you can get a Students License at LTO office at Parañaque City
Actors: Applicant, Officer-in-charge, Accountant
Creation Due: June 2, 2008   *Date of Update:* September 12, 2008
Version: 1.0   *Person in Charge:* Nelson O. Guinmapang Jr.

Flow of Events:

    Preconditions:
          10. The LTO must be open

    Main Success Scenario:
          14. The applicant will fill-up the application form
          15. The applicant will get his students license

    Alternative Sequences:
          A1: The applicant has no TIN number
                    The officer-in-charge will not accept the application form if the applicant has no TIN number. The applicant can get a TIN number from a BIR agent near the office
          A2: The parent consent of the applicant is not notarized.

If a minor is applying and he/she must have a legal consent form from his/her parent and it must be notarized.

Error Sequences:

E1: The applicant is a minor and doesn't have parent consent.

The officer-in-charge will not accept the application form if the applicant is a minor and doesn't have a legal consent form

Post Conditions:

7. The applicant will have a Student driver license.
8. The LTO will have a record of the applicant's profile.

UI (User Interface) Requirements:

- application form
- TIN verification

Non-Functional Requirements:

- The LTO office is open Mondays to Fridays, from 8:00 am – 4:00 pm
- The LTO places its records in a safe location
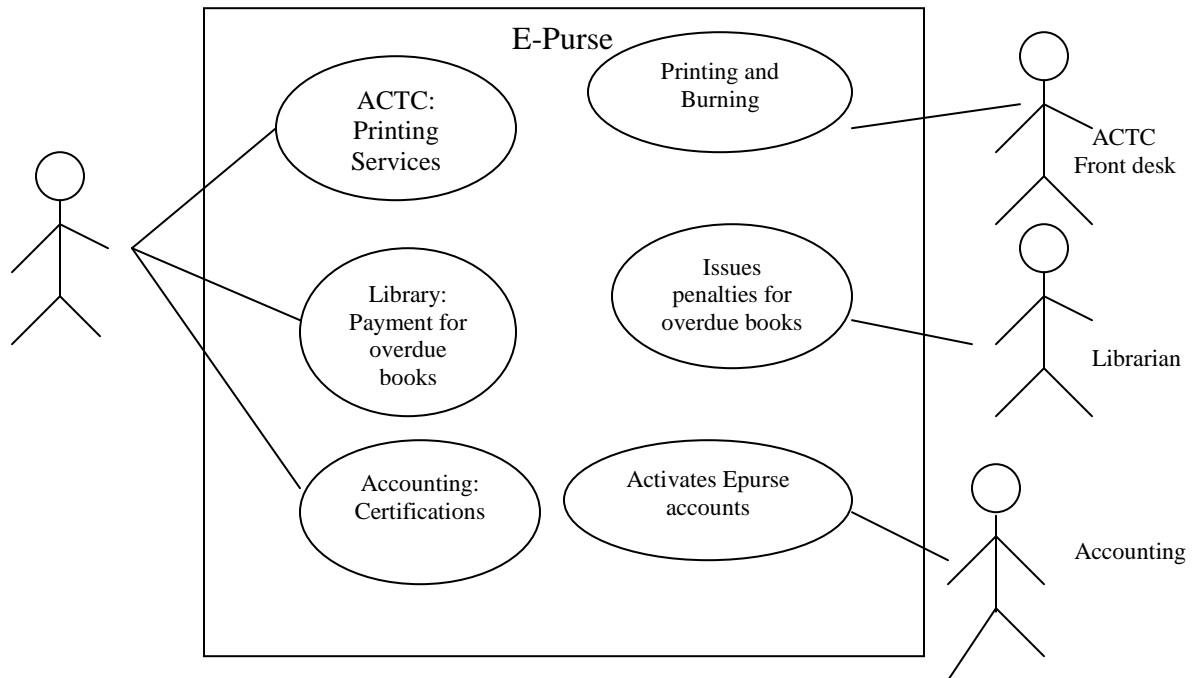
Identification summary

      Title: Using Epurse

      Summary: This use case diagram shows how students use Epurse to pay different transactions.

      Actors: Student, Accounting, ACTC, Librarian

      Creation Date: June 17, 2008   *Date of update:* September 2008

      Version: 1.0   *Person in Charge*: Nelson O. Guinmapang Jr.

Flow of Events:

      Preconditions:

1. Your id must be validated
2. The load of the Epurse must be sufficient for the transaction.
3. The school must be open
4. The student is enrolled at the school.

      Main Success Sequences:

1. You paid your transaction using your Epurse.
2. You don't have to pay at the accounting office whenever you have a transaction.

      Alternative Sequences:

A1: The load of the Epurse is insufficient to the amount of the transaction.

        A1. The transaction will be voided and the student must pay at the accounting office for the transaction or reload his/her Epurse.

A2. Id is not activated for Epurse

        A2. The student will be sent to the accounting office to get their Epurse accounts activated.

Error Sequences:

        E1: The student is not enrolled from the school.

            E2. The personnel will not accept the Epurse.

Post Conditions:

1. The student can use his Epurse to pay for the printing services, CD burning services, payment for the over due book.
2. The student will have sufficient time to pay easier.

Non-Functional Requirements

- The personnel can process your transaction faster using Epurse.
- The Accounting office is open from 8 AM to 5 PM

- The transaction between the student and the personnel is done privately.

APPENDIX 1 - EXERCISES

APPENDIX 2: FINAL PROJECT

A Systems Analysis Study on the Provident Fund of AMOSUP

Presented to the
Information Systems Program
School of Management and Information Technology
De La Salle – College of Saint Benilde

In Partial fulfillment of the
Requirements of the subject
Systems Analysis

Submitted By:
Magat, Michael Angelo
Lansang, Jairus
Guinmapang, Nelson
O0C/2nd year
August 2008

Submitted To:
Mr. Paul Pajo

Company Profile

Company Name:	Associated Marine Officers' and Seamen's Union of the Philippines
Address:	Cabildo corner Sta. Potenciana St. Intramuros, Manila
Telephone No.:	(02) 527-8491 to 98 loc. 103
Fax:	527-3534, 527-3538
Email Address:	s_center@amosup.org, capt_oca@amosup.org
Capability:	training and education, board and lodging, medical and dental plan, financial and legal services, housing

Date of Incorporation: November 11, 1960

Executive Officer: Captain Gregorio S. Oca

Manpower (as of June 2007):

- 1 – Admin
- 1 – Assistant
- 4 – Processors
- 6 – Encoders
- 1 – IT
- 1 – Accounting Clerk
- 1 – Releasing Clerk
- 1 – Filling Clerk
- 1 – Messenger
- 1 – Janitor

Associated Marine Officers' and Seamen's Union of the Philippines

On November 11, 1960, Capt. Gregorio S. Oca, concerned with the plight of the licensed crew of United President Lines, Magsaysay Lines, Inc. and the Eastern Shipping Lines, constituted the Associated Marine Officers' Union of the Philippines (AMOUP), with the PTGWO as the mother organization. At the same time, Bro. Donato Alarcon organized the unlicensed crew and formed the Associated Seamen's Union of the Philippines (ASUP). The members then working on board foreign vessels, receiving very low salaries and wages, with poor working conditions, and often not covered by necessary benefits and unprotected from accidents, sickness and death.

Guided by the same ideals and principles, the two (2) unions decided to merge into one cohesive organization in 1972 and named it the Associated Marine Officers' and Seamen's Union of the Philippines (AMOSUP-PTGWO). As early as its inception, AMOSUP affiliated itself with the International Transport Workers' Federation (ITF).

The union also realized that in order to be competitive with other seafarer supplying countries, the Filipino Seamen should be well trained, disciplined, and hard working. In return, the seafarer should be justly compensated and given all necessary benefits to afford a decent living. With these principal goals, focused on molding and upgrading the skills and improving the social status and well-being of the members, the leadership of AMOSUP began reshaping the future of the organization. The rest is history.

The unparalleled and successful programs of AMOSUP serve as lasting tributes and shrines to the people who helped organize the union and the members who believed and continuously supported its leadership.

VISION

AMOSUP, our vision is to give seafaring Filipinos valuable support in their quest for a better life. It begins as we negotiate for Collective Bargaining Agreements (CBA) that demand full commitment from all the parties involved – the Union, the employed, and the employee. With all three working towards the same goal, the seafarer receives the fruit of his labor in fair remuneration, benefits, programs and facilities available to him as an AMOSUP member

MISSION

With aims and objectives to unite all Filipino seafarers, Capt. Oca labored to fight for the social, legal and moral rights of the members in the domestic and foreign fronts. He wanted free, if not, affordable medical and dental services, sufficient education and adequate training, and a united organization with clear, definite and willful objectives.

ORGANIZATIONAL CHART
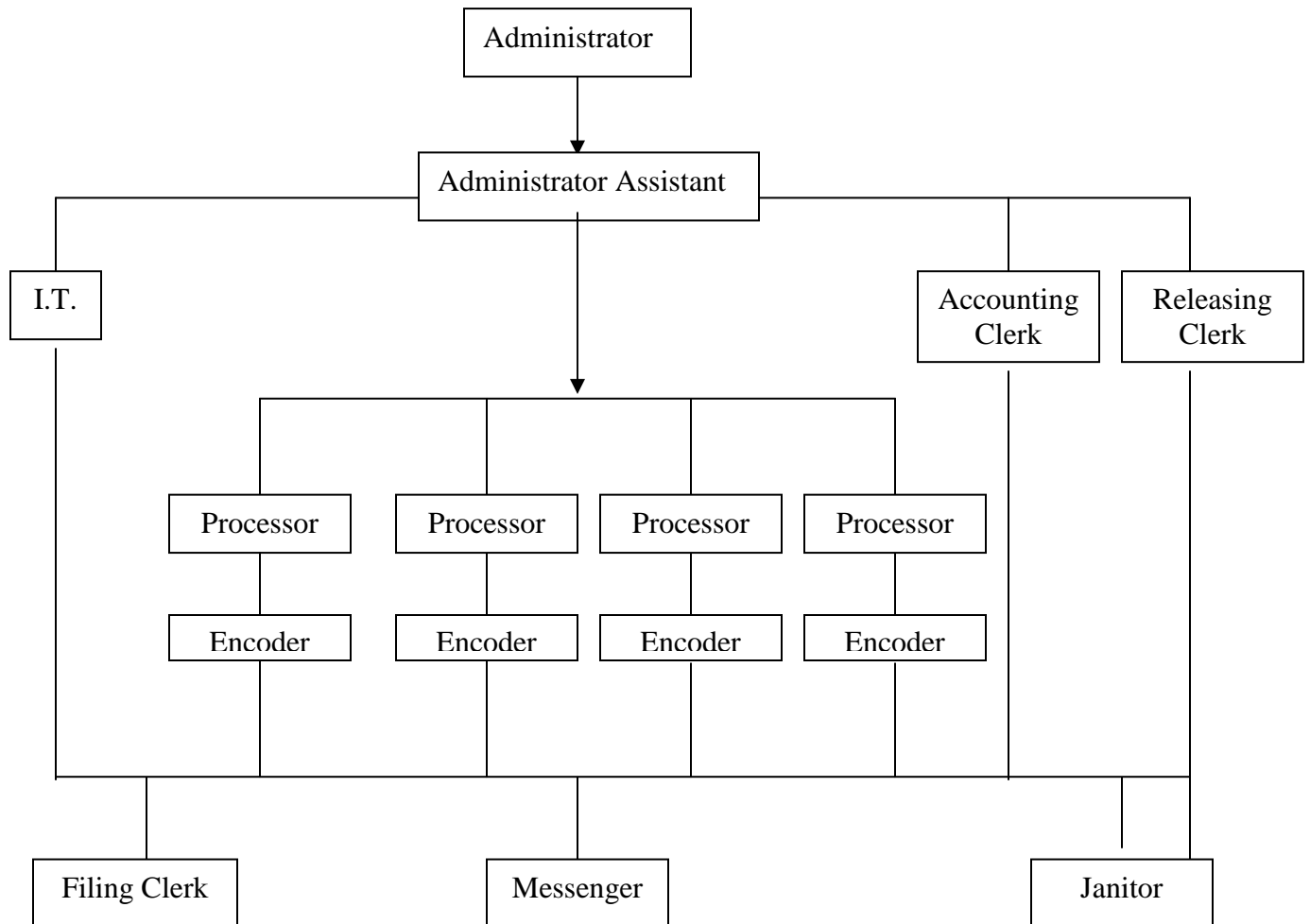
```
                        Administrator
                             │
                             ▼
                   Administrator Assistant
        ┌────────────────────┼───────────────┬──────────┐
        │                    │                │          │
      I.T.                   ▼          Accounting   Releasing
                                          Clerk        Clerk

        ┌──────────┬──────────┬──────────┐
   Processor   Processor   Processor   Processor

    Encoder     Encoder     Encoder     Encoder


   Filing Clerk          Messenger              Janitor
```

Statement of the Problem

We chose the system on how a seafarer who is 50 years or above now has the privilege to collect his or her provident fund. According to our interviews the seafarers rely on the passbooks that they receive on the start of their working contract to keep track of the money being invested in their provident fund. For each vessel that you ride you receive a passbook for it. The passbooks serve as a record of contributions from the vessels they have participated in. Each seafarer has to update his or her passbook regularly to validate the contributions in the Provident Fund Office. The problem is that a there a lot of vessels one seafarer can work in. Thus each seafarer could manage from around 3 or more passbooks. We can see that it would be difficult to keep track and update all your passbooks especially for the ones who live in provinces.

Significance of the Study

This study's significance is to show the user how our proposed system could increase efficiency in their system. We find it important to study how each process works because through it we are able to find out what needs improvements and what causes problems. With our study we will be able to make a much more efficient system and an easier way for the seafarers to check on their Provident funds. Because we know that if we want our proposed system to replace the existing one we have to carefully study how it runs. We cannot make a better system based on just what we know about the system. Studying it further would lead us to the real problems the existing system has.

Scope and Limitation

The process that we examined is how the company updates Provident fund of its union members and make it available for their applicants. We did not include their other the other services offered at AMOSUP as we plan to focus on how to increase the efficiency in the Provident fund department.
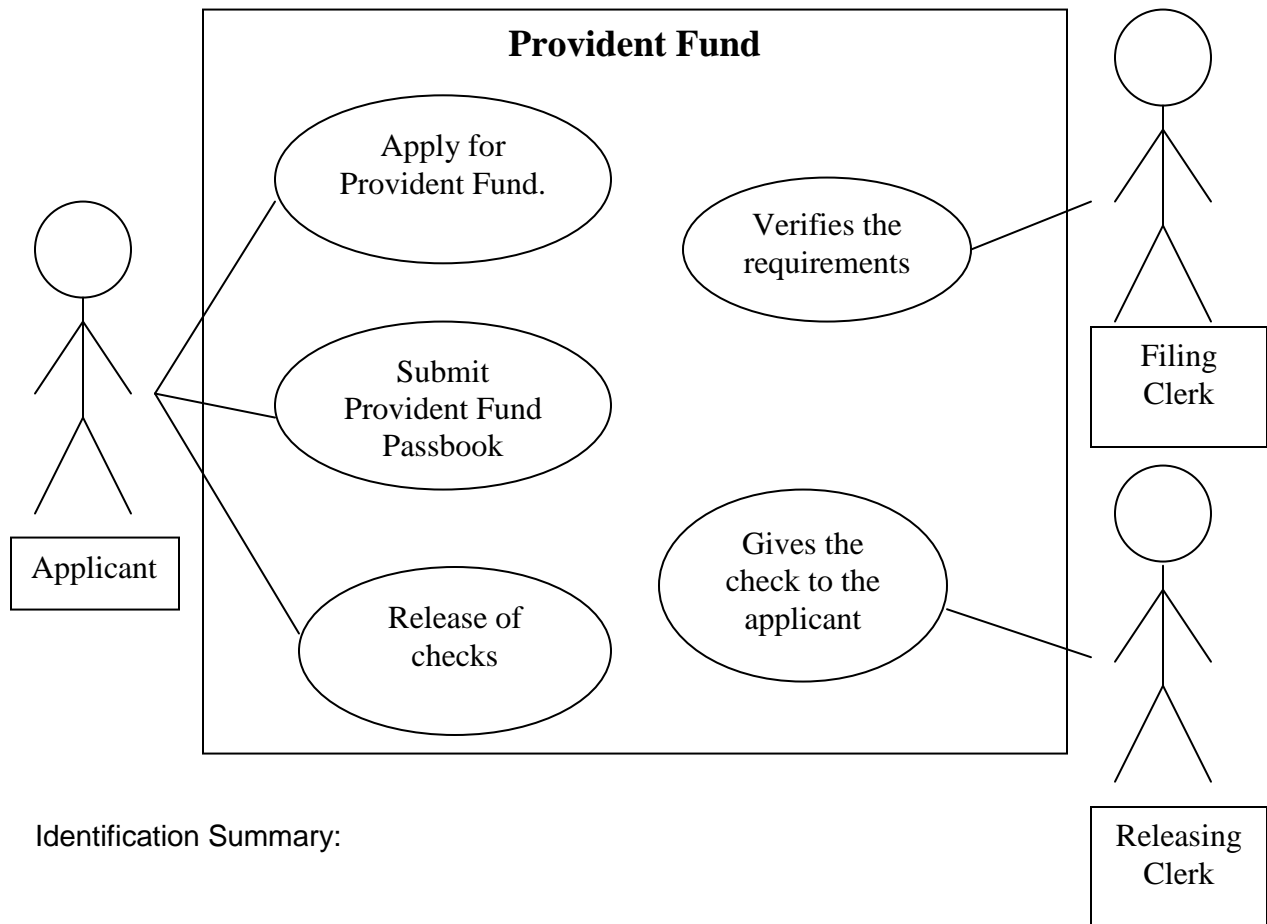
EXISTING SYSTEM

Provident Fund

Apply for Provident Fund.

Verifies the requirements

Submit Provident Fund Passbook

Gives the check to the applicant

Release of checks

Applicant

Filing Clerk

Releasing Clerk

Identification Summary:

Title: Submitting of ID/Passport or Seaman's book

Summary: This use case shows how seafarers submits their ID/Passport or Seaman's book

Actors: Applicant, Filing Clerk, Releasing Clerk

Creation Date: September 12, 1990   Date of update: Aug 2008

Version: 1.1

Person-in-charge: Jairus Lansang, Nelson Guinmapang, Michael Magat

Flow of Events:

Preconditions:

1. The office is open.

2. The staffs are present.

Main Success Scenario:

1. The applicant gets stub on counter 1.
2. The staff gives stub.
3. The applicant waits at the lounge for his stub to call.
4. The staff calls the applicant number.
5. The applicant goes to the respective counter.

(Counter 2 for JSU or Japan Seamen's Union.

Counter 3 for TCC ships.

Counter 4 for Dutch, Belgian, French, LSA, SMOU, NCSU and Swedish.

Counter 5 for British, FKSU, PRV, Italian and NCL)

6. The applicant presents his requirements to the staff.
7. The staff checks if the requirements of the applicant is complete.
8. The staff inputs the applicant's information to the database.
9. The applicant has now a provident fund.

Alternative Sequences:

A1. Incomplete requirements

A1. Scenario starts at sequence 7 of the Main Case Scenario.

8. The staff informs that the requirements are incomplete.
9. The applicant completes the requirements.

Scenario goes back at point 2.

A2. The applicant goes to the wrong counter.

A2. Scenario starts at sequence 5 of the Main Case Scenario.

6. The staff will inform the applicant that he goes to a wrong counter.

7. The applicant goes to the counter where he is under.

Scenario goes back at point 6.

Error Sequence:

E1. The database is offline.

E1. The sequence starts at point 8 of the main case scenario.

10. The staff will not be able to input the file of the applicant. Use case fails.

E2. The office experiences power failure.

E2. The sequence starts at any point of the Main Case Scenario.

The office will not able to process the applicants because they don't have electricity.

Use case fails.

Post Conditions:

1. The applicant now has a Provident Fund.
2. The stubs in counter 1 are lesser.
3. The office has all the profile of the applicants.

Non-functional Requirements:

1. Availability

   The office is open 8:00 AM to 3:00 AM during weekdays.

2. Confidentially

   The transaction between the applicant and staff is done privately.

Activity Diagram: With Swimlane
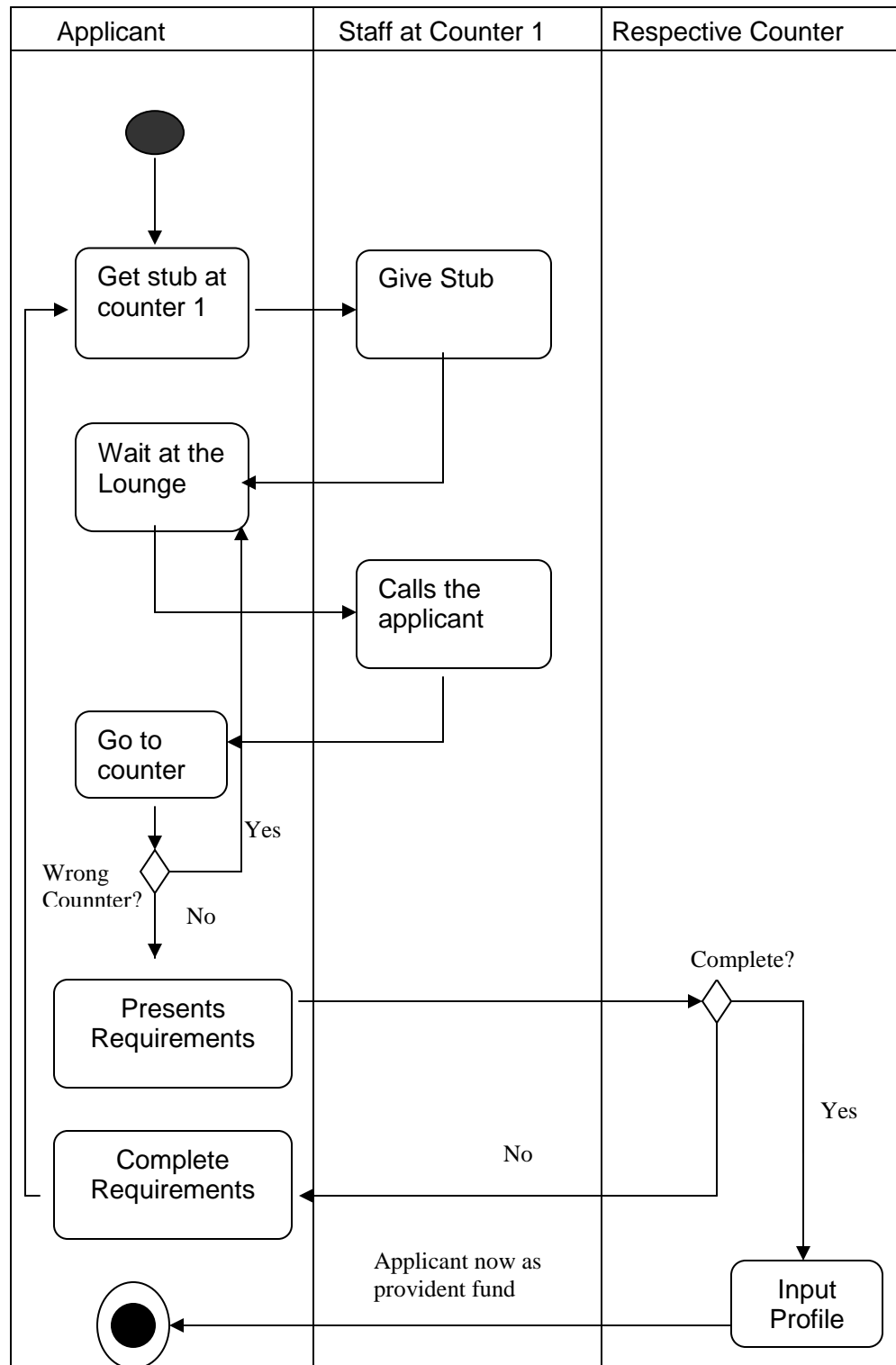
Applicant	Staff at Counter 1	Respective Counter

Get stub at counter 1

Give Stub

Wait at the Lounge

Calls the applicant

Go to counter

Wrong Counnter?

Yes

No

Presents Requirements

Complete?

Complete Requirements

No

Yes

Applicant now as provident fund

Input Profile

Identification Summary:

Title: Submitting of Provident Fund Passbook

Summary: This use case shows how seafarers submits their Provident Fund passbook

Actors: Applicant, Filing Clerk, Releasing Clerk

Creation Date: September 12, 1990   Date of update:

Version: 1.1

Person-in-charge: Jairus Lansang, Nelson Guinmapang, Michael Magat

Flow of Events:

Preconditions:

1. The applicant has a Provident Fund.

2. The office is open.

3. The staff is present.

Main Success Scenario:

1   The applicant gets stub on counter 1.

2.   The staff gives stub.
3.   The applicant waits at the lounge for his stub to call.
4.   .The staff calls the applicant number.
5.   The applicant goes to the respective counter.
(Counter 2 for JSU or Japan Seamen's Union.

Counter 3 for TCC ships.

Counter 4 for Dutch, Belgian, French, LSA, SMOU, NCSU and Swedish.

Counter 5 for British, FKSU, PRV, Italian and NCL)

6.   The applicant presents submits his Provident fund passbook.
7.   The staff checks if the content of the passbook from their database match.
8.   The staff approves the passbook.
9.   The staff processes the passbook for the releasing of the check.

10. The staff will give a claim stub of the schedule on when does the applicant can get his check.

11. The applicant waits for the releasing of the check.

Alternative Sequences:

A1. The applicant goes to the wrong counter.

A2. Scenario starts at sequence 5 of the Main Case Scenario.

6. The staff will inform the applicant that he goes to a wrong counter.

7. The applicant goes to the counter where he is under.
Scenario goes back at point 6

A2. The content in the passbook did not match to the database of the office.

The scenario starts at point 7 of the Main Case Scenario.

8. The staff will match the content of the passbook and the data in their database.

Scenario goes back at point 8.

Error Sequence:

E1. The database is offline.

E1. The sequence starts at point 7 of the main case scenario.

11. The staff will not be able to check the passbook of the applicant in their database.
Use case fails.

E2. The office experiences power failure.

E2. The sequence can starts at any point of the Main Case Scenario.

The office will not able to process the applicants because they don't have electricity.

Use case fails.

Post Conditions:

The applicant gives his passbook to the staff.

The applicant waits for the schedule of the releasing of the check.

The staff has lesser paper for the schedule of the releasing of the check.

Non-functional Requirements:

1. Availability

   The office is open 8:00 AM to 3:00 AM during weekdays.
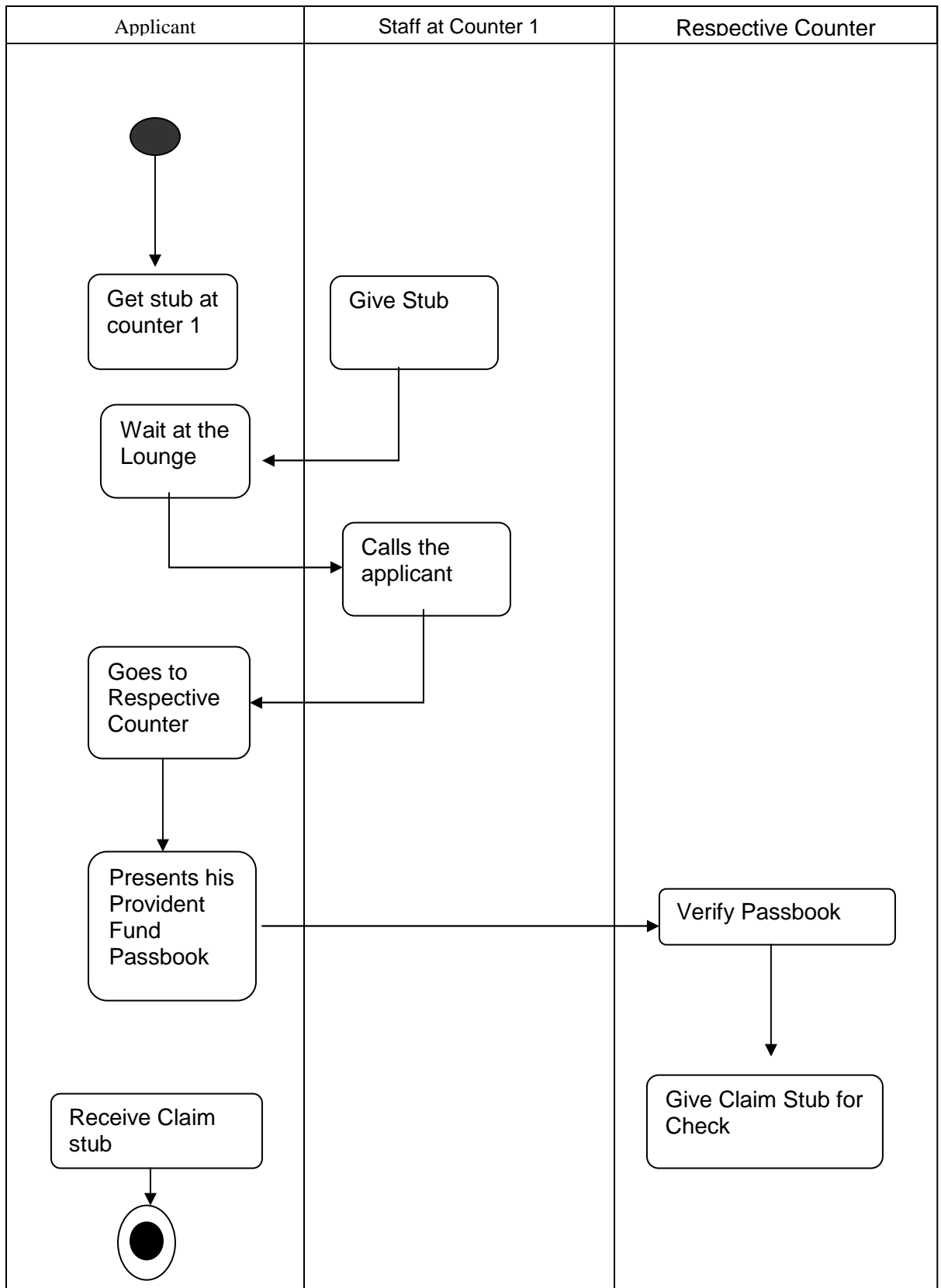
2. Confidentially

   The transaction between the applicant and staff is done privately.

Applicant	Staff at Counter 1	Respective Counter

Get stub at counter 1

Give Stub

Wait at the Lounge

Calls the applicant

Goes to Respective Counter

Presents his Provident Fund Passbook

Verify Passbook

Receive Claim stub

Give Claim Stub for Check

Identification Summary:

Title: Releasing of Checks

Summary: This use case shows how seafarers receive their checks from the releasing clerk

Actors: Applicant, Releasing Clerk

Creation Date: September 12, 1990   Date of update:

Version: 1.1

Person-in-charge: Jairus Lansang, Nelson Guinmapang, Michael Magat

Flow of Events:

Preconditions:

1. The office is open.

2. The staffs are present.

Main Success Scenario:

1   The applicant goes to counter 1.
2   The applicant presents his claim stub.
3   The staff checks if the date of the claim stub and the current date match.
4   The staff checks if the check of the applicant is ready.
5   The staff asks for a identification card from the applicant.
6   The applicant presents identification to the staff.
7   The applicant will sign a paper stating that he has received the check for his provident fund.
8   The staff gives the check to the applicant.
9   The applicant claims the check.

Alternative Sequences:

A1. Lost claim stub.

A1. The scenario starts at point 2 of the Main Case Scenario.

3. The staff will ask the applicant to get an affidavit of loss stating that he lost his claim stub.

4. The applicant gets an affidavit of loss.

5. The applicant notarizes his affidavit of loss

6. The applicant presents his affidavit of loss.

Scenario goes back at point 3.

Error Sequences

E1. Unreleased check

E1. The scenario starts at point 4 of the Main Case Scenario.

5. The staff will tell to the applicant that the check is not yet available.

6. The applicant will not get the check.

Use case fails

E2. The office experiences power failure.

E2.   The sequence starts at any point of the Main Case Scenario.

The office will not able to process the applicants because they don't have electricity.

Use case fails.

Post Conditions:

- The applicant now has his check.

- There are fewer papers in counter 1.

Non-functional Requirements:

1. Availability

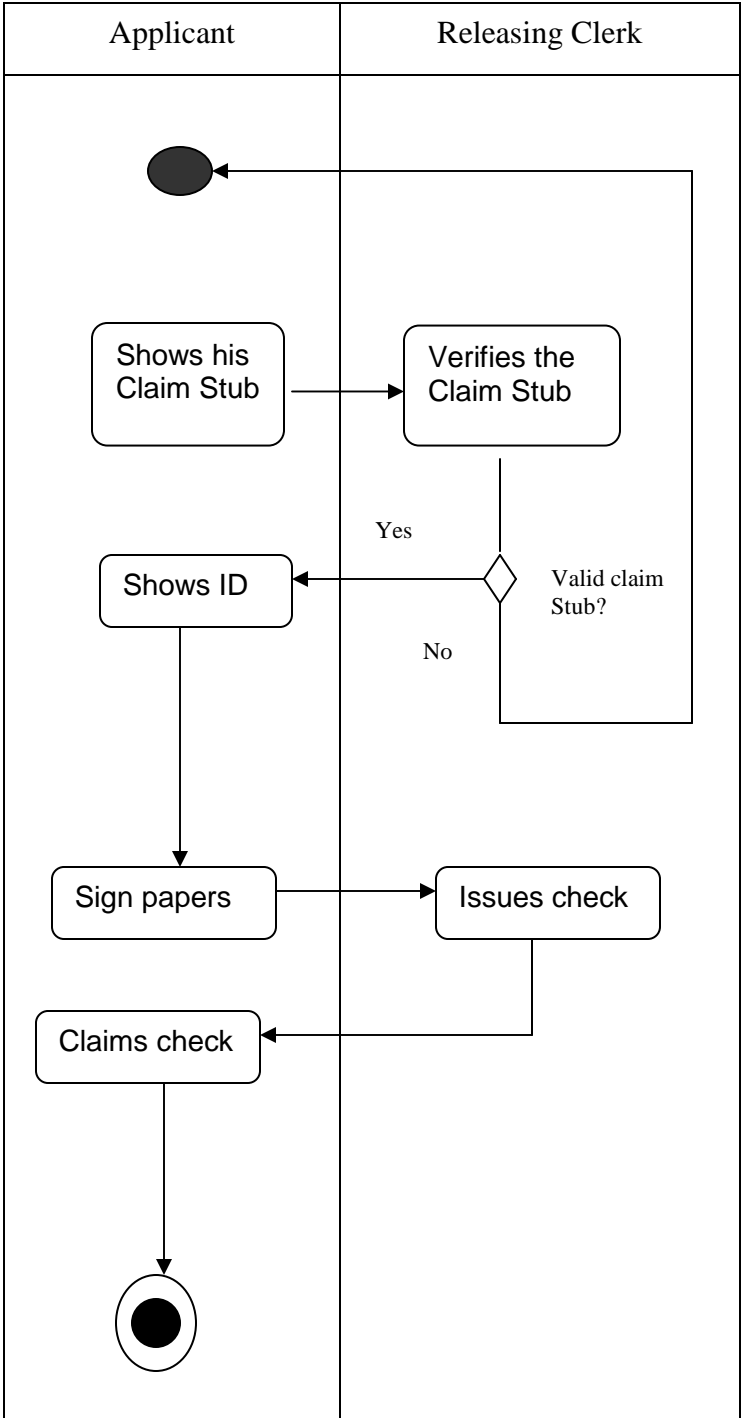   The office is open 8:00 AM to 3:00 AM during weekdays.

2. Confidentially

   The transaction between the applicant and staff is done privately.

Applicant	Releasing Clerk

Shows his Claim Stub → Verifies the Claim Stub

Valid claim Stub?

Yes → Shows ID

No

Shows ID → Sign papers

Sign papers → Issues check

Issues check → Claims check

Claims check →

Identification Summary:

Title: Verifies Requirement

Summary: This use case shows how filing clerk verifies the requirement of the applicant.

Actors: Applicant, Filing Clerk

Creation Date: September 12, 1990   Date of update:

Version: 1.1

Person-in-charge: Jairus Lansang, Nelson Guinmapang, Michael Magat

Flow of event

Precondition:

1. The office is open.
2. There is an applicant.

Main Success scenario

1. The applicant goes the staff to pass the requirements.
2. The staff checks if the applicant completes all the needed requirements.
3. The staff verifies the requirement and transfers the data to the database

Alternative Scenario:

A1. Incomplete Requirements

The scenario starts at step 2 of the Main Case Scenario.

3. The applicant's requirements are not complete.
4. The staff asks the applicant to complete his requirements.

The scenario goes back to point 2.

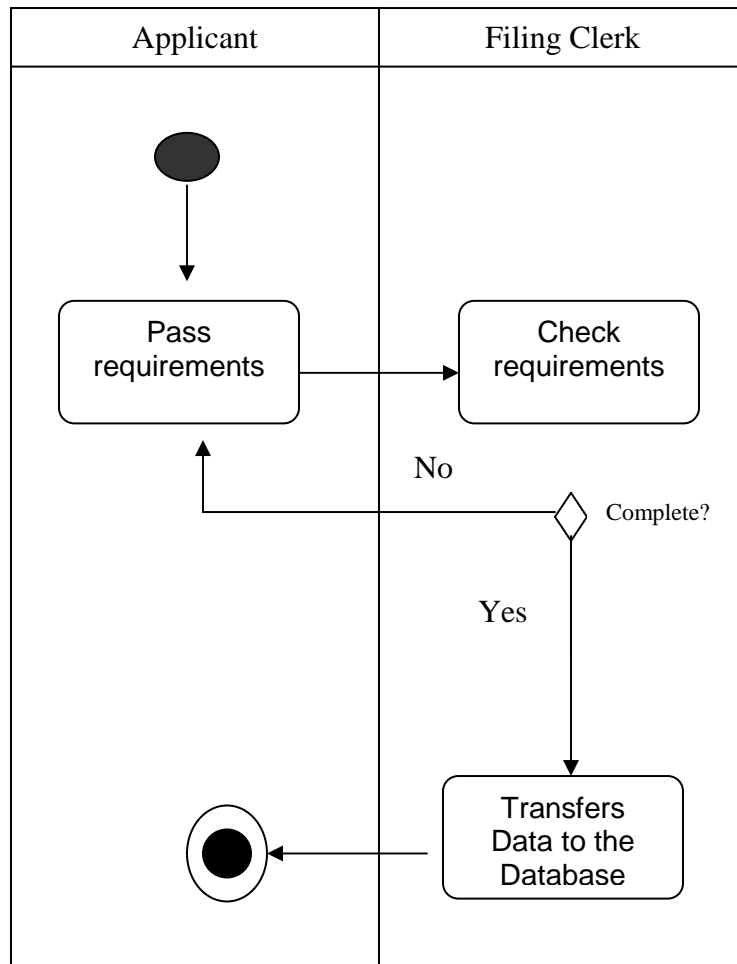Error Scenario:

E1. The office experiences power failure.

E1. The sequence starts at any point of the Main Case Scenario.

The office will not able to process the applicants because they don't have electricity.
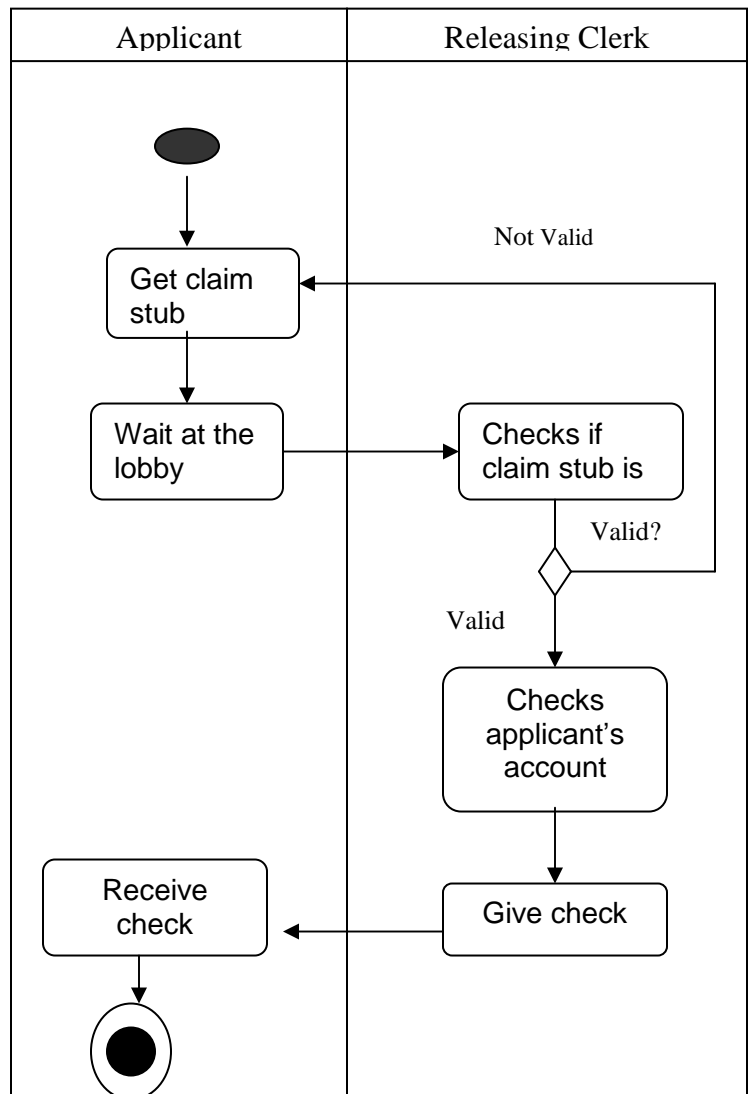
Use case fails

Post Conditions:

1.   Availability
The office is open 8:00 AM to 3:00 AM during weekdays.

2.   Confidentially
The transaction between the applicant and staff is done privately.

Applicant	Filing Clerk

```
        Applicant                    Filing Clerk

          ●
          │
          ▼
    ┌───────────┐              ┌───────────┐
    │   Pass    │─────────────▶│   Check   │
    │requirements│              │requirements│
    └───────────┘              └───────────┘

                          No
    ┌─────────────────────────◇  Complete?

                              │ Yes
                              │
                              ▼
    ◉ ◀──────────────┌───────────┐
                     │ Transfers │
                     │Data to the│
                     │ Database  │
                     └───────────┘
```

Identification Summary:

Title: Gives the check to the applicant

Summary: This use case shows how releasing clerk releases the check to the applicant

Actors: Applicant, Releasing Clerk

Creation Date: September 12, 1990   Date of update:

Version: 1.1

Person-in-charge: Jairus Lansang, Nelson Guinmapang, Michael Magat

Flow of event

Precondition:

1. The office is open.
2. There is an applicant.

Main Success scenario

1. The applicant gets claim stub for the check
2. The applicant will wait at the lobby to be called using the claim stub.
3. The releasing clerk verifies the claim stub.
4. The releasing clerk checks the applicant's account
5. Once approved, the releasing clerk will give to the applicant the check that contains his Provident Fund money.

Alternative Scenario:

A1. Lost claim stub

1. The scenario starts at step 3 of the Main Case Scenario.

2. The applicant lost his stub number to get his Provident Fund

Error Scenario:

E1. The office experience power failure.

1. The sequence starts at any point of the Main Case
Scenario.

2. The office will not able to process the applicants because they
don't have electricity.

3. Use case fails

Applicant	Releasing Clerk

Not Valid

Get claim
stub

Wait at the
lobby

Checks if
claim stub is

Valid?

Valid

Checks
applicant's
account

Receive
check

Give check

Table Recommendation

Problem	Recommended Change	Affected Activities
Seafarers find it difficult to manage and update their Provident Fund passbooks.	A newer system that automatically updated the seafarers Provident funds without the help their passbooks. This would also allow the seafarer to keep track of the money stored in his or her Provident Fund.	The submission of requirements can be submitted online through the seafarer's personal Provident Fund account.

.
Proposed System:
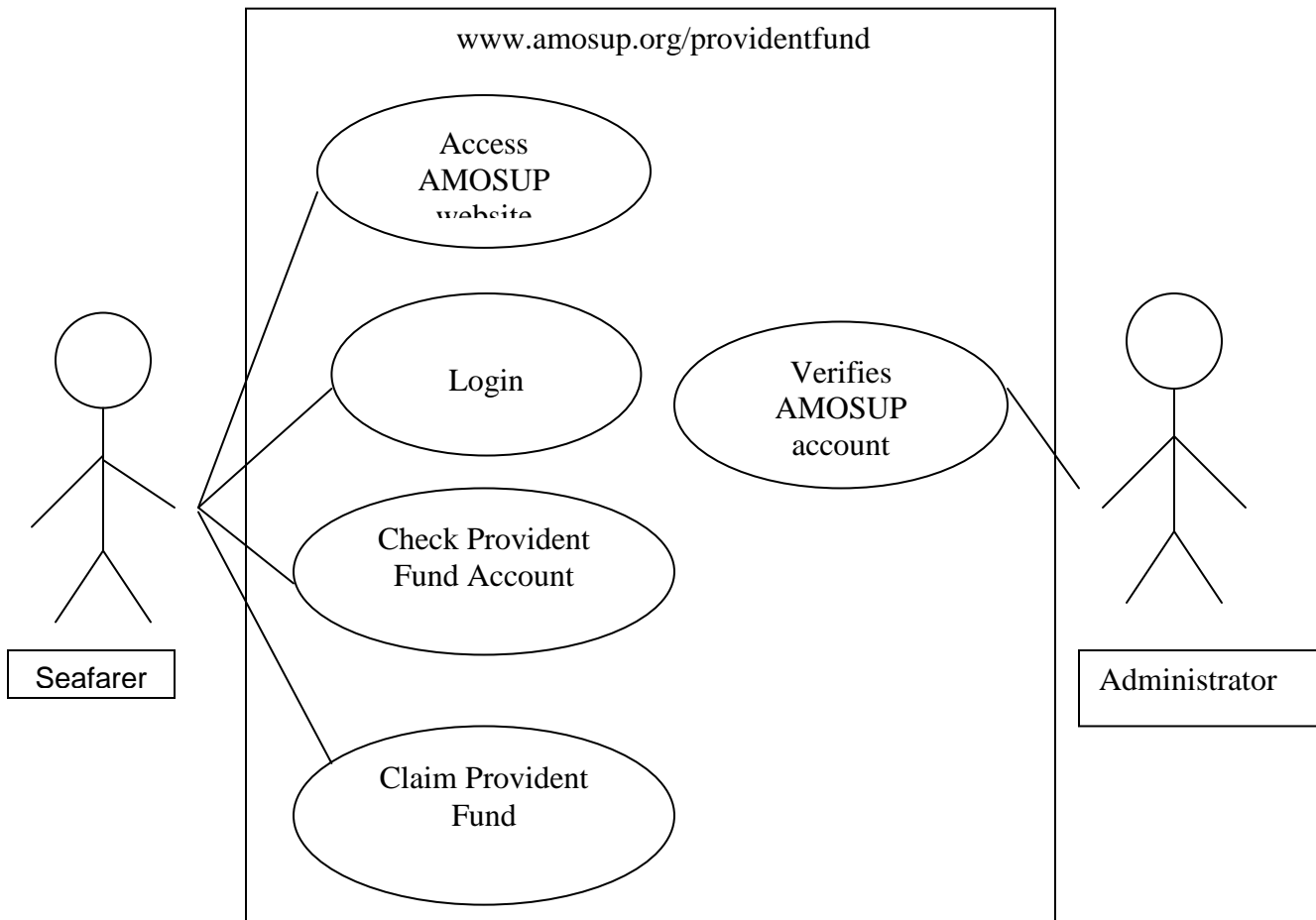
Identification Summary:

Title: Access AMOSUP website

Summary: This use case shows how the applicant can access the AMOSUP website

Actors: Applicant, Administrator

Creation Date: October 12, 1990   Date of update:

Version: 1.2

Person-in-charge: Jairus Lansang, Nelson Guinmapang, Michael Magat

Flow of Events:

Preconditions:

1. The applicant has Internet access
2. The applicant has a working browser

Main Success Scenario:

1. The applicant access the internet
2. The applicant inputs AMOSUP's URL.
3. The applicant access' the AMOSUP website.

Alternative Sequences:

A1. The server is offline

A1. Scenario starts at sequence 2 of the Main Case Scenario.

3. The applicant cannot access the website if the server is offline
4. The applicant must then wait until the server is back online

The scenario goes back to point 2.

A2. The Internet connection was disconnected

    A2. Scenario starts at sequence 1 of the Main Case Scenario.

    2. The applicant must reconnect to the Internet to access the AMOSUP website

    The scenario goes back to point 1.

A3. The URL accessed is incorrect

    A3. Scenario starts at sequence 2 of the Main Case Scenario.
    3. The correct URL must be placed in order to access the AMOSUP website.
    The scenario goes back to point 2.

Error Sequence:

E1. There is no Internet access available.

    E1. Scenario starts at sequence 1 of the Main Case Scenario.

    1. The applicant must first have an Internet connection in order to access the AMOSUP website.

    Use case fails

E2. The applicant does not know the AMOSUP URL.

    E2. Scenario starts at sequence 1 of the Main Case Scenario.

    1. The applicant would not be able to access the AMOSUP website unless he or she knows the AMOSUP URL.

    Use case fails

Post Conditions:

- The number of people visited the website increases

Applicant

```
        ●
        │
        ▼
   ┌─────────┐
   │ Access the │
   │ internet   │
   └─────────┘
        │
        ▼
   ┌─────────┐
   │ Enter    │
   │ AMOSUP   │
   │ URL      │
   └─────────┘
        │
        ▼
   ┌─────────┐
   │ Enter    │
   │ AMOSUP   │
   │ website  │
   └─────────┘
        │
        ▼
       ◉
```
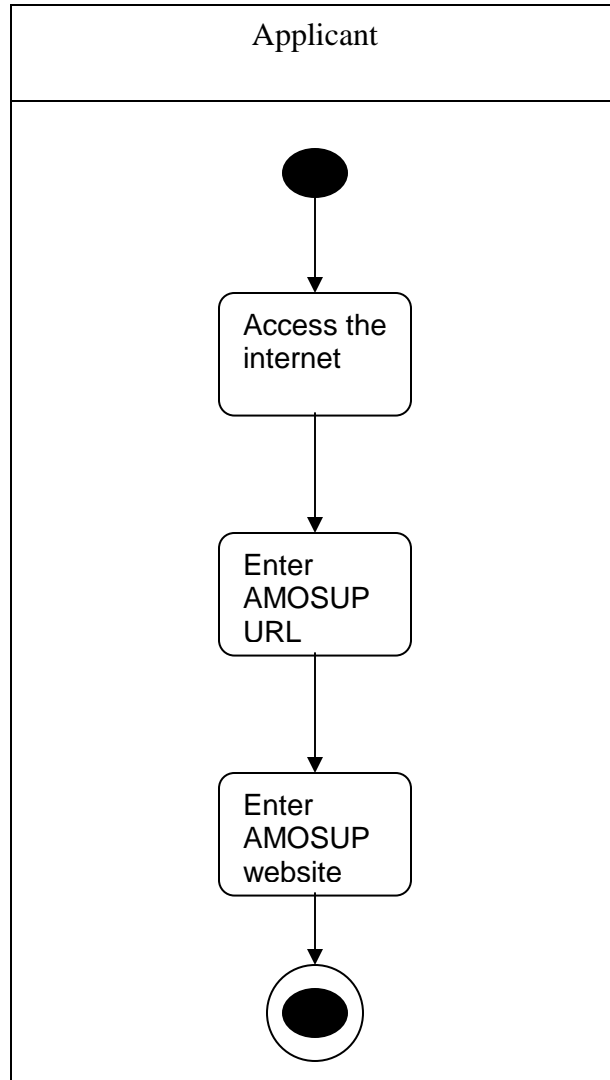
Identification Summary:

Title: Login

Summary: This use case shows how the applicant can login at AMOSUP website

Actors: Applicant, Administrator

Creation Date: October 12, 1990   Date of update:

Version: 1.2

Person-in-charge: Jairus Lansang, Nelson Guinmapang, Michael Magat

Flow of Events:

Preconditions:

1. The applicant must have a working internet connection
2. The applicant must be at the AMOSUP website
3. The applicant must have an account

Main Success Scenario:

1. The applicant inputs his username and password

2. The system administrator verifies the username and password

3. The applicant is successfully login

Alternative Sequences:

A1. Incorrect username or password

A1. Scenario starts at sequence 2 of the Main Case Scenario

3. The applicant must input the correct username and password

The scenario goes back to point 1

Error Sequence:

E1. Does not have an account

E1. Scenario starts at sequence 1 of the Main Case Scenario.

2. The applicant cannot login unless he or she has created his or her own personal AMOSUP account

Use Case fails

E2. Accessed the wrong website

E2. Scenario starts at sequence 1 of the Main Case Scenario.

2. The applicant can only access his or her AMOSUP account through the AMOSUP website
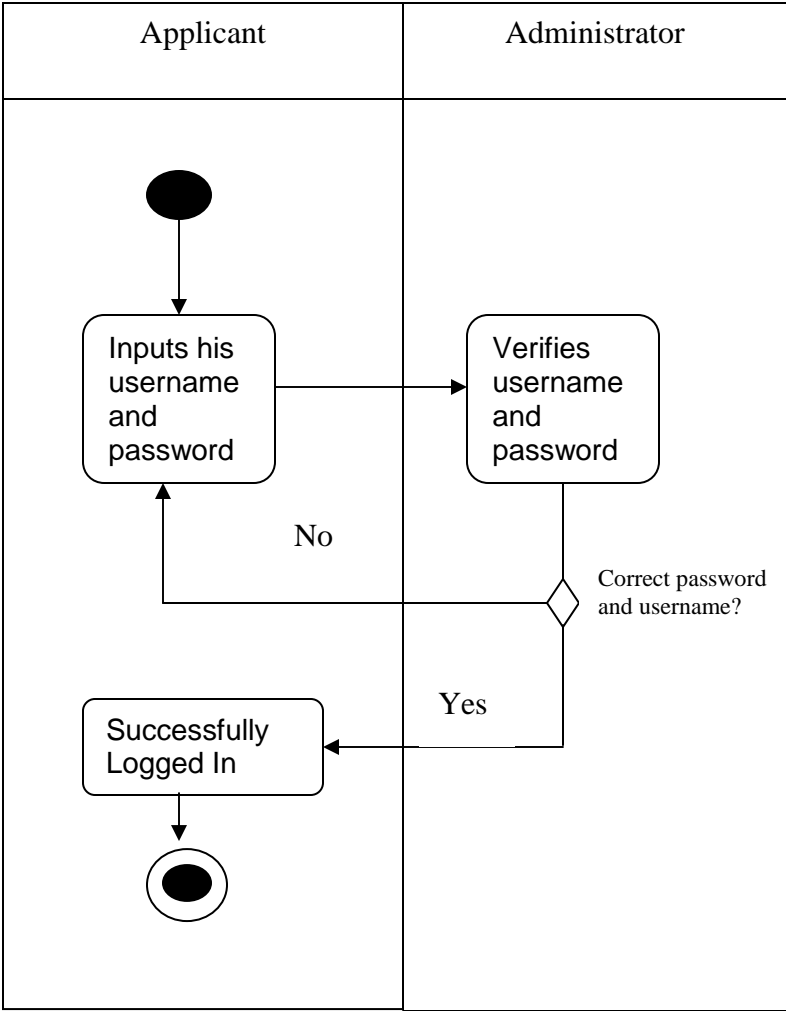
Use case fails

Post Conditions:
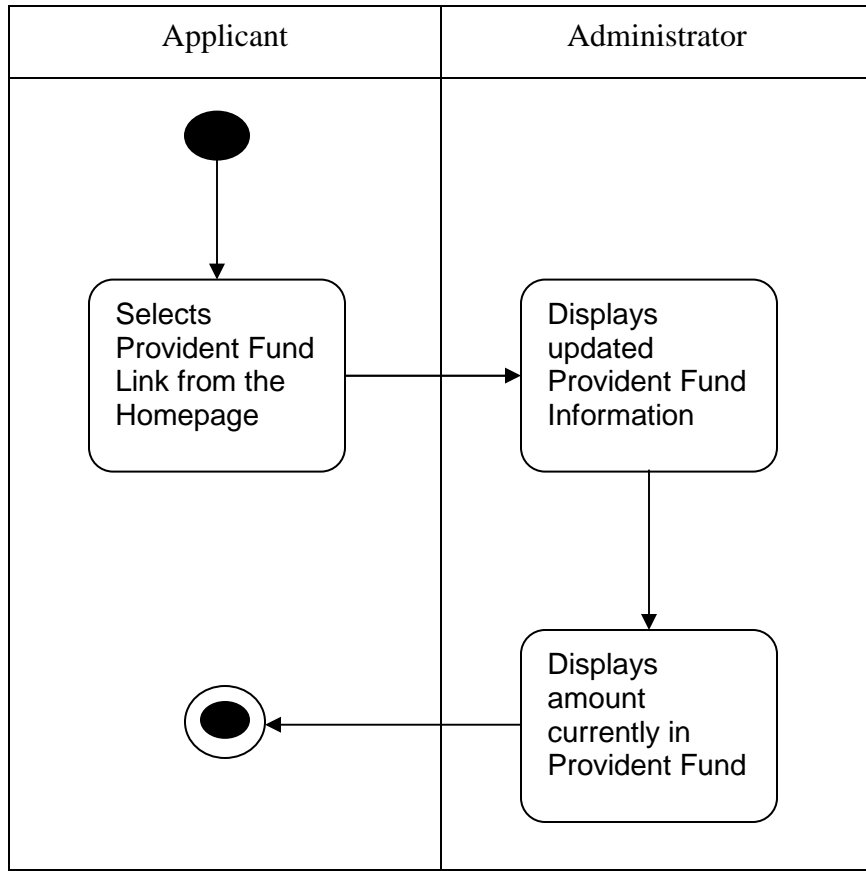
- The log of users who have logged in the site has increased

Applicant	Administrator

Inputs his username and password

Verifies username and password

No

Correct password and username?

Yes

Successfully Logged In

Identification Summary:

Title: Check Provident Fund account

Summary: This use case shows how applicant can check their Provident Fund account

Actors: Applicant, Administrator

Creation Date: October 12, 1990   Date of update:

Version: 1.2

Person-in-charge: Jairus Lansang, Nelson Guinmapang, Michael Magat

Flow of Events:

Preconditions:

1. The applicant must have a working internet connection
2. The applicant must be logged in his own AMOSUP account
3. The applicant must have an Provident Fund account

Main Success Scenario:

1. The applicant selects the Provident Fund link in the AMOSUP website

2. The applicant access the Provident Fund page

3. The applicant's Provident Fund information is displayed in the page

4. The applicant can view the amount of his or her Provident Fund account has stored through the web page.

Alternative Sequences:

A1. Incorrect link accessed

A1. Scenario starts at sequence 1 of the Main Case Scenario.

2. The applicant must access the correct link in order to view his or her own Provident Fund page.

The scenario goes back to point 1

A2. The web page timed out

A2.Scenario starts at any point of the Main Case Scenario.

1. The applicant did not enabled the "remember me" option and has to log in his account again to access his account

Error Sequence:

E1. Website is under maintenance

E1. Scenario starts at any sequence of the Main Case Scenario.

1. The applicant cannot view the website until the maintenance is over

Use case fails

E3. Accessed the wrong website

E3. Scenario starts at any point of the Main Case Scenario

1. The applicant can only access his or her AMOSUP account through the AMOSUP website

Use case fails

Post Conditions:

- The view of Provident Fund in the site has increased
- The seafarer now knows how much is currently in his account

Applicant	Administrator

Selects Provident Fund Link from the Homepage

Displays updated Provident Fund Information

Displays amount currently in Provident Fund

Identification Summary:

Title: Claim Provident Fund

Summary: This use case shows how applicant can claim their Provident Fund

Actors: Applicant, Administrator

Creation Date: October 12, 1990   Date of update:

Version: 1.2

Person-in-charge: Jairus Lansang, Nelson Guinmapang, Michael Magat

Flow of Events:

Preconditions:

1. The applicant must be logged in his account
2. The applicant must have a Provident Fund account.

Main Success Scenario:

1. The company will receive a request claim from the applicant.
2. The company will issue letter to the applicant on where he can claim his fund. (The company will cite the bank that is affiliated with them, if the seafarer is living in the province.)
3. The applicant will go the go the bank where his or her money is delivered.

Alternative Sequences:

A1. The company is not affiliated with any of the banks where the applicant lives

A1. Scenario starts at sequence 2 of the Main Case Scenario.

3. The bank then chooses to send the check through the applicant's bank account.

The scenario goes back to point 1

A2. The money is not delivered to the bank

A2. Scenario starts at sequence 3 of the Main Case Scenario

2. The bank informs the company that the money has not been received.

3. The company verifies and resends the check

The scenario goes back to point 2

Error Sequence

E2. A letter stating that the money can be claimed was not received by the applicant

1. The applicant does not know if he or she can claim the money

Use case fails

Post Conditions:

- There is less money in the whole Provident Fund
- The applicant now has his Provident funds

Applicant	Administrator	Bank

● (start)

Send a request letter for claim → **Receive letter from the applicant**

Issue check and send to applicant what bank to claim the check

Receive information where can get the check ←

Go to Bank → **Give Check** → ◉

APPENDIX 3 - REFERENCE

http://www.wikipedia.org
http://www.google.com
http://www.yahoo.com
De La Salle – College of Saint Benilde - Learning Resource Center (EXTENSION)
Introduction to Systems Analysis and Design by: Igor T. Hawryszkiewycz