Table of Contents

[Introduction 4](#_Toc452490091)

[Task 1 5](#_Toc452490092)

[1.1. Data collection tools & methods 5](#_Toc452490093)

[1.1. Some suitable Data collecting methods for Smart Arcade 5](#_Toc452490094)

[1.1.1. Interview users 6](#_Toc452490095)

[1.1.2. Send Questionnaires 7](#_Toc452490096)

[1.1.3. Examine suggestions and problem reports 8](#_Toc452490097)

[1.1.4. Observation 9](#_Toc452490098)

[1.2. Different system of SDLC model 10](#_Toc452490099)

[1.2.1. The different types of model in SDLC 10](#_Toc452490100)

[1.2.1.1. Water fall model 11](#_Toc452490101)

[1.2.1.1.1. Advantages of waterfall model 11](#_Toc452490102)

[1.2.1.1.2. Disadvantages of waterfall model 12](#_Toc452490103)

[1.2.1.2. Incremental model 12](#_Toc452490104)

[1.2.1.2.1. Advantages of Incremental model 13](#_Toc452490105)

[1.2.1.2.2. Disadvantages of Incremental model 13](#_Toc452490106)

[1.2.1.3. Prototype model 13](#_Toc452490107)

[1.2.1.3.1. Advantages of Prototype model 14](#_Toc452490108)

[1.2.1.3.2. Disadvantages of Prototype model 14](#_Toc452490109)

[1.2.1.4. Rapid application development model 14](#_Toc452490110)

[1.2.1.4.1. Advantages of the RAD model 15](#_Toc452490111)

[1.2.1.4.2. Disadvantages of RAD model 15](#_Toc452490112)

[1.2.1.5. Spiral model 16](#_Toc452490113)

[1.2.1.5.1. Advantages of Spiral model 16](#_Toc452490114)

[1.2.1.5.2. Disadvantages of Spiral model 16](#_Toc452490115)

[1.2.1.6. Agile Methodology 17](#_Toc452490116)

[1.2.1.6.1. Advantages 17](#_Toc452490117)

[1.2.1.6.2. Disadvantages 17](#_Toc452490118)

[1.2.3. The top 08 key factors for that I’m choosing Agile’s XP model for proposed system 18](#_Toc452490119)

[1.2.4. Why other models are not suitable for this project? 20](#_Toc452490120)

[**1.3. Importance of following a procedural / staged lifecycle in a systems investigation. 21**](#_Toc452490121)

[1.3. The phases of SDLC 21](#_Toc452490122)

[1.3.1. Requirement gathering 21](#_Toc452490123)

[1.3.2. Analysis and Specification 22](#_Toc452490124)

[1.3.3. Design 22](#_Toc452490125)

[1.3.4. Coding 22](#_Toc452490126)

[1.3.5. Testing 23](#_Toc452490127)

[1.3.5.1. Some important Testing techniques 23](#_Toc452490128)

[1.3.5.1.1. Black-Box Testing 23](#_Toc452490129)

[1.3.5.1.2. White-Box Testing 23](#_Toc452490130)

[1.3.5.1.3. System Testing 24](#_Toc452490131)

[1.3.5.1.4. Unit Testing 24](#_Toc452490132)

[1.3.5.1.5. Integration Testing 24](#_Toc452490133)

[1.3.6. Release and Maintenance 24](#_Toc452490134)

[TASK 02 25](#_Toc452490135)

[2. Developing the feasibility report 25](#_Toc452490136)

[2.1. Feasibility Study 25](#_Toc452490137)

[2.1.1. The 05 main Feasibilities 25](#_Toc452490138)

[2.1.1.1. Technology and System Feasibility 25](#_Toc452490139)

[2.1.1.2. Economic feasibility 26](#_Toc452490140)

[2.1.1.3. Legal feasibility 26](#_Toc452490141)

[2.1.1.4. Operational feasibility 26](#_Toc452490142)

[2.1.1.5. Schedule feasibility 26](#_Toc452490143)

[**2.2. Feasibility Report 27**](#_Toc452490144)

[2.2.1. Feasibility report of the Smart Arcade spare parts company 27](#_Toc452490145)

[TASK 03 34](#_Toc452490146)

[3.1. Developing a Systems Requirements Specification (SRS) 34](#_Toc452490147)

[3.1.1. The purpose of the SRS is to: 34](#_Toc452490148)

[3.2. SRS of the Smart Arcade Company 35](#_Toc452490149)

[TASK 04 39](#_Toc452490150)

[4. Use of Methodology 39](#_Toc452490151)

[4.1. Entity Relationship Diagram (ERD) 39](#_Toc452490152)

[4.1.1. Entity Relationship Diagram’s Symbols 39](#_Toc452490153)

[4.1.2. Entity Relationship Diagram of the system 40](#_Toc452490154)

[4.2. Data Flow Diagram 41](#_Toc452490155)

[4.2.1. Types of DFD 41](#_Toc452490156)

[4.2.1.1. Data Flow Diagrams are either Logical or Physical. 41](#_Toc452490157)

[4.2.2. Derivation of DFD 41](#_Toc452490158)

[4.2.3. Data Flow Diagram’s Notations - Gane & Sarson method 42](#_Toc452490159)

[4.2.4. Context Diagram of the system (DFD Level 0) 42](#_Toc452490160)

[4.2.5. DFD Level 1 Diagram of the system 43](#_Toc452490161)

[4.2.6. DFD Level 2 Diagram of the system 44](#_Toc452490162)

[TASK 05 45](#_Toc452490163)

[5.1. Quality Management 45](#_Toc452490164)

[5.1.1. Top 5 Reasons to Implement a Quality Management 45](#_Toc452490165)

[5.1.2. There are 03 types of process 46](#_Toc452490166)

[5.1.2.1. Quality Planning 46](#_Toc452490167)

[5.1.2.2. Quality Control 47](#_Toc452490168)

[5.1.2.3. Quality Assurance 46](#_Toc452490169)

[5.2. The Quality Control and Assurance of the Smart Arcade 47](#_Toc452490170)

[5.2.1. Software Quality Control and Assurance 47](#_Toc452490171)

[5.2.2. Hardware Quality Control and Assurance 48](#_Toc452490172)

[**Conclusion 50**](#_Toc452490173)

[**Harvard Reference 53**](#_Toc452490174)

[Gantt chart 58](#_Toc452490175)

## 

## 

## Introduction

This assignment is about to solve some obstacles of the Smart Arcade car spare parts Company which is manufacturing the car spare parts based on the customer orders. The company obstacles are delays on processes, misplacement of the customer orders and some customer service issues during the paper based work system. Therefore the company is looking to solve the obstacles by implementing a computer based system.

There are total 05 tasks in this assignment and I have answered them respectively. Task 1, I have explained about the Requirement gathering methods, SDLC models, and software development phases with suitable pictures. Task 2, I have described about the feasibility study categories and made report on a different criteria of the feasibility analysis. Next task, I have evaluated the system requirement specification and made a SRS analysis of the proposed system with hardware, software, functional and non-functional requirements Task 4 which is all about the designing part and I have drawn a 04 diagrams for the proposed system which are ER diagram, DFD level 0 (Context diagram), DFD level 1 and DFD level 2. Final task is a quality management analysis which I have explained about the QP, QC and QA respectively.

I have completed the assignment with the help of internet because most of the parts are analysis and reports. In the mean time, I take this opportunity to express my thanks to Mr. Nirupan who has taught me the subject (S.A.D) in a tremendous way.

# Task 1

# 

# 1.1. Data collection tools & methods

Data collection is the process of gathering and measuring information on variables of interest, in an established systematic fashion that enables one to answer stated research questions, test hypotheses, and evaluate outcomes.

## http://cdn1.hubspot.com/hub/167634/file-296310301.jpg

Figure 1

**Importance of the data collection**

* Systematic approach of gathering information from various sources to answer research questions and to test hypotheses
* Accurate data collection essential to maintain integrity of research
* Correct use of collected data important to reduce errors
* Most time and labor intensive step in an empirical study

## 1.1. Some suitable Data collecting methods for Smart Arcade

* Interview users
* Send questionnaires
* Examine suggestions and problem reports
* Observation

## 1.1.1. Interview users

Face-to-face contact with users through individual interviewing is the primary source of requirements and an important way you gather and validate their requirements. Remember that it is not the only possible technique, and that you can conduct interviews many different ways. Develop a repertoire of styles to fit different situations. Unless you use the system yourself, you will need to make an effort to understand and experience the user's problem to describe it clearly and correctly.

**Advantages**

* Any misunderstanding and mistake can be rectified easily in an interview. Because the interviewer and interviewee physically present before the interview board.
* Interview can help to collect the fresh, new and primary information as needed.
* Interview can help to save time to select the best suitable candidate. Within a very short time communication can be accomplished with the interview
* Interview can help to save time to select the best suitable candidate. Within a very short time communication can be accomplished with the interview

**Disadvantages**

* Suitable candidate cannot be selected by interview only. The written test is more important than the interview.
* In the case of the interview some confusion may be arisen in the future as, there is no evidence actually that have been discussed at interview.
* Always there is a possibility that the interview process can be influenced by the biases of the interviewer.
* Generally interview method is expensive

## 

## 1.1.2. Send Questionnaires

If face-to-face meetings are possible, they are always preferable, because they provide a better means of uncovering the problem behind the problem. Sometimes, though, face-to-face meetings with stakeholders are not feasible (when developing products for the consumer market, for example). In those situations, consider using questionnaires.

**Advantages**

* The survey [questionnaire](https://explorable.com/questionnaire-example) can be rapidly deployed and completed by the respondents, especially if there’s an incentive that is given after their participation.
* An Internet survey facilitates low-cost and fast data collection from the target population. Sending email questionnaires and other online questionnaires are more affordable than the [face-to-face method](https://explorable.com/personal-interview-survey).
* Online survey provides the highest level of convenience for the respondents because they can answer the questionnaire according to their own pace, chosen time, and preferences.
* The responses are automatically stored in a survey database, providing hassle-free handling of data and a smaller possibility of data errors.

**Disadvantages**

* An online survey is not suitable for surveys which ask open-ended questions because there is no trained interviewer to explore the answers of the respondents.
* This method is not applicable for surveys that require respondents who do not have an access to the Internet.
* Questions that bear controversies may not be precisely answered by the participants because of the probably difficulty of recalling the information related to them.
* The survey that was used by the researcher from the very beginning, as well as the method of administering it.

## 1.1.3. Examine suggestions and problem reports

Requirements can come from change suggestions and user problems. A direct road to finding requirements is to look at suggestions and problems as first described. Most organizations have a form for reporting system problems or software defects. You can ask to look through the reports (and there will probably be many). Sort them into groups so you can identify the key areas that are troubling users. Ask users some questions about these areas to clarify the users' actual needs.

**Advantages**

1. Relatively easy to administer
2. Can be developed in less time (compared to other data-collection methods)
3. Cost-effective, but cost depends on survey mode
4. Can be administered remotely via online, mobile devices, mail, email, kiosk, or telephone.
5. Conducted remotely can reduce or prevent geographical dependence

**Disadvantages**

* Respondents may not feel encouraged to provide accurate, honest answers
* Respondents may not feel comfortable providing answers that present themselves in a unfavourable manner.
* Respondents may not be fully aware of their reasons for any given answer because of lack of memory on the subject, or even boredom.
* Surveys with closed-ended questions may have a lower validity rate than other question types.

## 1.1.4. Observation

It’s accurate watching, noting the phenomenon by which they occur in the nature with regard to the cause and effect of mutual relations. Science begins with observation and at the end also uses observation for the final validation – so it can be said that Observation acts as very elegant method in case of scientific investigation.

**Types of observation**

1. Direct observation – It is a method of collecting evaluative information in which the evaluator watches the subject in his or her usual environment without altering that environment.

2. Indirect observation - Indirect Observation is an observational technique whereby some record of past behaviour is used than observing behaviour contemporaneously.

**Advantages**

* Very direct method for collecting data or information – best for the study of human behavior.
* Data collected is very accurate in nature and also very reliable.
* Improves precision of the research results.
* Problem of depending on respondents is decreased.

**Disadvantages**

* Problems of the past cannot be studied by means of observation.
* Having no other option one has to depend on the documents available.
* Observations like the controlled observations require some especial instruments or tools for effective working, which are very much costly.
* One cannot study opinions by this means.

# 

# Conclusion: I’m going to use the questionnaire data collection method for this company. Through this method, I can get informations easily and most of the questions are based on “one word answer” which will not consume more time and will not frustrate the people who are answering. In addition, one of the sample questionnaire has been attached at the appendix.

# 1.2. Different system of SDLC model

SDLC is a process followed for a software project, within a software organization. It consists of a detailed plan describing how to develop, maintain, replace and alter or enhance specific software. The life cycle defines a methodology for improving the quality of software and the overall development process.

A systems development life cycle is composed of a number of clearly defined and distinct work phases which are used by systems engineers and systems developers to plan for, design, build, test, and deliver [information systems](https://en.wikipedia.org/wiki/Information_system). Like anything that is manufactured on an assembly line, an SDLC aims to produce high quality systems that meet or exceed customer expectations, based on customer requirements, by delivering systems which move through each clearly defined phase, within scheduled time-frames and cost estimates. Computer systems are complex and often (especially with the recent rise of [service-oriented architecture](https://en.wikipedia.org/wiki/Service-oriented_architecture)) link multiple traditional systems potentially supplied by different software vendors. To manage this level of complexity, a number of SDLC models or methodologies have been created, such as "[waterfall](https://en.wikipedia.org/wiki/Waterfall_model)"; "[spiral](https://en.wikipedia.org/wiki/Spiral_model)"; "[Agile software development](https://en.wikipedia.org/wiki/Agile_software_development)"; "[rapid prototyping](https://en.wikipedia.org/wiki/Software_prototyping#Throwaway_prototyping)"; "[incremental](https://en.wikipedia.org/wiki/Incremental_development)"; and "synchronize and stabilize".

## 1.2.1. The different types of model in SDLC

1. Water fall model
2. Incremental model
3. Prototype model
4. Rapid application development model
5. Spiral model
6. Agile Methodology

**SDLC Models can be divided into 03 categories**

1. Iterative
2. Sequential
3. Progressive

## 1.2.1.1. Water fall model

It is a Plan-driven model. Separate and distinct phases of specification and development

[](http://istqbexamcertification.com/wp-content/uploads/2012/01/Waterfall-model.jpg)

**Figure 2**

## ****1.2.1.1.1. Advantages of waterfall model****

* This model is simple and easy to understand and use.
* It is easy to manage due to the rigidity of the model – each phase has specific deliverables and a review process.
* In this model phases are processed and completed one at a time. Phases do not overlap.
* Waterfall model works well for smaller projects where requirements are very well understood.

## ****1.2.1.1.2. Disadvantages of waterfall model****

* Once an application is in the [testing](http://istqbexamcertification.com/what-is-a-software-testing/) stage, it is very difficult to go back and change something that was not well-thought out in the concept stage.
* No working software is produced until late during the life cycle.
* High amounts of risk and uncertainty.
* Not a good model for complex and object-oriented projects.

## 1.2.1.2. Incremental model

The incremental build model is a method of [software development](https://en.wikipedia.org/wiki/Software_development) where the product is [designed](https://en.wikipedia.org/wiki/Software_design), implemented and tested incrementally (a little more is added each time) until the product is finished. It involves both development and maintenance. The product is defined as finished when it satisfies all of its requirements.

[](http://istqbexamcertification.com/wp-content/uploads/2012/01/Incremental_model.jpg)

**Figure 3**

## ****1.2.1.2.1. Advantages of Incremental model****

* Generates working software quickly and early during the software life cycle.
* This model is more flexible – less costly to change scope and requirements.
* It is easier to test and debug during a smaller iteration.
* In this model customer can respond to each built.
* Lowers initial delivery cost.

## ****1.2.1.2.2. Disadvantages of Incremental model****

* Needs good planning and design.
* Needs a clear and complete definition of the whole system before it can be broken down and built incrementally.
* Total cost is higher than [waterfall](http://istqbexamcertification.com/what-is-waterfall-model-advantages-disadvantages-and-when-to-use-it/).

## 1.2.1.3. Prototype model

A prototype model is an initial version of a system used to demonstrate concepts and try out design options.

[](http://istqbexamcertification.com/wp-content/uploads/2012/01/Prototype-model.jpg)

Figure 4

## ****1.2.1.3.1. Advantages of Prototype model****

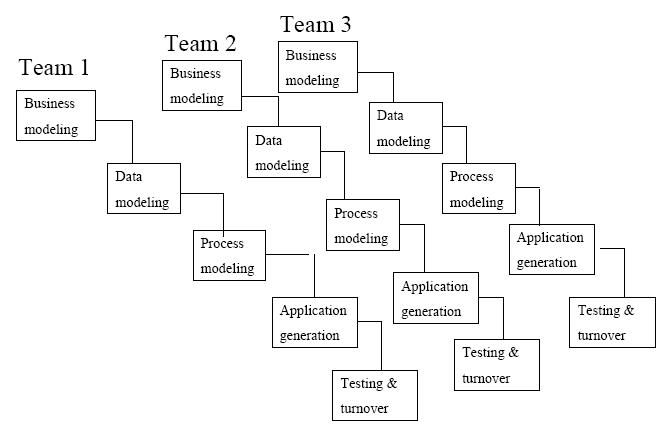
* Users are actively involved in the development
* Since in this methodology a working model of the system is provided, the users get a better understanding of the system being developed.
* Errors can be detected much earlier.
* Quicker user feedback is available leading to better solutions.
* Missing functionality can be identified easily
* Confusing or difficult functions can be identified  
  Requirements validation, Quick implementation of, incomplete, but  
  functional, application.

## ****1.2.1.3.2. Disadvantages of Prototype model****

* Leads to implementing and then repairing way of building systems.
* Practically, this methodology may increase the complexity of the system as scope of the system may expand beyond original plans.
* Incomplete application may cause application not to be used as the full system was designed  
  Incomplete or inadequate problem analysis.

## 1.2.1.4. Rapid application development model

RAD model is Rapid Application Development model. It is a type of incremental. In RAD model the components or functions are developed in parallel as if they were mini projects. The developments are time boxed, delivered and then assembled into a working prototype. This can quickly give the customer something to see and use and to provide feedback regarding the delivery and their requirements.



**Figure 5**

## ****1.2.1.4.1. Advantages of the RAD model****

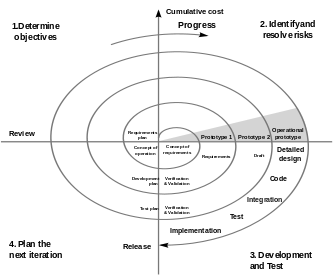
* Reduced development time.
* Increases reusability of components
* Quick initial reviews occur
* Encourages customer feedback
* Integration from very beginning solves a lot of [integration issues](http://istqbexamcertification.com/what-is-system-integration-testing/).

## ****1.2.1.4.2. Disadvantages of RAD model****

* Depends on strong team and individual performances for identifying business requirements.
* Only system that can be modularized can be built using RAD
* Requires highly skilled developers/designers.
* High dependency on modeling skills
* Inapplicable to cheaper projects as cost of modeling and automated code generation is very high.

## 1.2.1.5. Spiral model

The spiral model is similar to the [incremental model](http://istqbexamcertification.com/what-is-incremental-model-advantages-disadvantages-and-when-to-use-it/), with more emphasis placed on risk analysis. The spiral model has four phases: Planning, Risk Analysis, Engineering and Evaluation. A software project repeatedly passes through these phases in iterations (called Spirals in this model). The baseline spiral, starting in the planning phase, requirements are gathered and risk is assessed each subsequent spirals builds on the baseline spiral.



**Figure 6**

## ****1.2.1.5.1. Advantages of Spiral model****

* High amount of risk analysis hence, avoidance of Risk is enhanced.
* Good for large and mission-critical projects.
* Strong approval and documentation control.
* Additional Functionality can be added at a later date.

## ****1.2.1.5.2. Disadvantages of Spiral model****

1. Can be a costly model to use.
2. Risk analysis requires highly specific expertise.
3. Project’s success is highly dependent on the risk analysis phase.
4. Doesn’t work well for smaller projects.

## 1.2.1.6. Agile Methodology - Extreme Programming ( XP )

**Agile development model** is also a type of [Incremental model](http://istqbexamcertification.com/what-is-incremental-model-advantages-disadvantages-and-when-to-use-it/). Software is developed in incremental, rapid cycles. This results in small incremental releases with each release building on previous functionality. Each release is thoroughly [tested](http://istqbexamcertification.com/why-is-testing-necessary/) to ensure [software quality](http://istqbexamcertification.com/what-is-software-quality/) is maintained. It is used for time critical applications.  Extreme Programming (XP) is currently one of the most well known agile [development life cycle model](http://istqbexamcertification.com/what-are-the-software-development-models/).

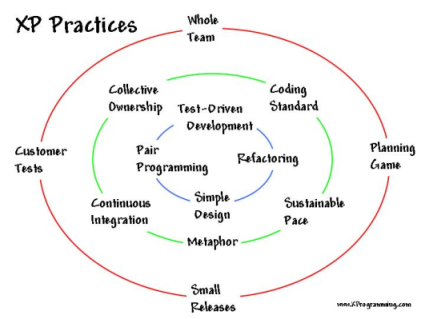


Figure 7

## 1.2.1.6.1. Advantages

* Customer satisfaction by rapid, continuous delivery of useful software.
* People and interactions are emphasized rather than process and tools. Customers, developers and testers constantly interact with each other.
* Working software is delivered frequently (weeks rather than months).
* Face-to-face conversation is the best form of communication.

## 1.2.1.6.2. Disadvantages

1. In case of some software deliverables, especially the large ones, it is difficult to assess the effort required at the beginning of the software development life cycle.
2. There is lack of emphasis on necessary designing and documentation.
3. The project can easily get taken off track if the customer representative is not clear.
4. No place for newbie programmers, unless combined with experienced resources.

## 1.2.3. The top 08 key factors for that I’m choosing Agile’s XP model for proposed system

**1. Revenue**

The iterative nature of agile development means features are delivered [incrementally](http://www.allaboutagile.com/2007/03/agile-principle-5-how-dyou-eat-elephant.html), enabling some benefits to be realized early as the product continues to develop.

**2. Speed-to-market**

Research suggests about 80% of all market leaders were first to market. As well as the higher revenue from incremental delivery, agile development philosophy also supports the notion of early and [regular releases](http://www.allaboutagile.com/2007/03/agile-principle-6-focus-on-frequent.html), and ‘perpetual beta’.

**3. Quality**

A key principle of agile development is that [testing is integrated throughout the lifecycle](http://www.allaboutagile.com/2007/04/agile-development-agile-testing-is-not.html), enabling regular inspection of the working product as it develops. This allows the product owner to make adjustments if necessary and gives the product team early sight of any quality issues.

**4. Visibility**

Agile development principles encourage [active ‘user’ involvement](http://www.allaboutagile.com/2007/02/principle-1-active-user-involvement-is.html) throughout the product’s development and a very cooperative collaborative approach. This provides excellent [visibility](http://www.allaboutagile.com/2007/05/beauty-of-not-doing-agile-development.html) for key stakeholders, both of the project’s progress and of the product itself, which in turn helps to ensure that expectations are effectively managed.

**5. Risk Management**

Small incremental releases made visible to the product owner and product team through its development help to [identify any issues early](http://www.allaboutagile.com/2007/05/beauty-of-not-doing-agile-development.html) and make it easier to respond to change. The clear visibility in agile development helps to ensure that any necessary decisions can be taken at the earliest possible opportunity, while there’s still time to make a material difference to the outcome.

**6. Flexibility / Agility**

In traditional development projects, we write a big spec up-front and then tell business owners how expensive it is to change anything, particularly as the project goes on. In fear of scope creep and a never-ending project, we resist changes and put people through a change control committee to keep them to the essential minimum. Agile development principles are different. In agile development, change is accepted. In fact, it’s expected*.* Because the one thing that’s certain in life is change. Instead the [timescale is fixed](http://www.allaboutagile.com/2007/03/agile-principle-3-time-waits-for-no-man.html) and requirements emerge and evolve as the product is developed. Of course for this to work, it’s imperative to have an [actively involved](http://www.allaboutagile.com/2007/02/principle-1-active-user-involvement-is.html) stakeholder who understands this concept and makes the necessary trade-off decisions, trading existing scope for new.

**7. Cost Control**

The above approach of [fixed timescales and evolving requirements](http://www.allaboutagile.com/2007/03/agile-principle-3-time-waits-for-no-man.html) enables a fixed budget. The scope of the product and its features are variable, rather than the cost.

**8. Business Engagement / Customer Satisfaction**

The [active involvement](http://www.allaboutagile.com/2007/02/principle-1-active-user-involvement-is.html) of a user representative and/or product owner, the [high visibility](http://www.allaboutagile.com/2007/05/beauty-of-not-doing-agile-development.html) of the product and progress, and the [flexibility to change](http://www.allaboutagile.com/2007/03/agile-principle-3-time-waits-for-no-man.html) when change is needed, create much better business engagement and customer satisfaction. This is an important benefit that can create much more positive and enduring working relationships.

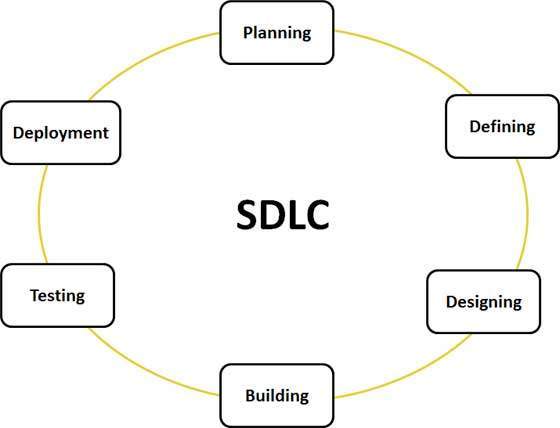
## 1.2.4. Why other models are not suitable for this project?

1. **Water fall model:** Smart arcade company’s requirements or details can be change at any time, therefore waterfall model will not be suitable for this project.
2. **Incremental model:** Same reason for waterfall model and cost is higher than waterfall model.
3. **Prototype model:** Practically, this methodology may increase the complexity of the system as scope of the system may expand beyond original plans, therefore the cost would be high.
4. **Rapid application development model:** It’s inapplicable to cheaper projects as cost of modeling and automated code generation is very high.
5. **Spiral model:** It’s not suitable for the small project and it is a costly model.

**Conclusion:**

According to the above key factors which clearly indicate why Agile Methodology is ideally suitable for this project rather than the other development models. Therefore, I’m going to choose the Agile’s XP model for our software development.

## 1.3. Importance of following a procedural / staged lifecycle in a systems investigation.



**Figure 8**

## 1.3. The phases of SDLC

1. Requirement gathering

2. Analysis and Specification

3. Design

4. Coding

5. Testing

6. Release and maintenance

## 1.3.1. Requirement gathering

The process of determining user expectations for a new or modified product and these features, called requirements, must be quantifiable, relevant and detailed. In [software](http://searchsoa.techtarget.com/definition/software) engineering, such requirements are often called [functional specification](http://searchsoftwarequality.techtarget.com/definition/functional-specification)s.

## 

## 1.3.2. Analysis and Specification

Once the requirement analysis is done the next step is to clearly define and document the product requirements and get them approved from the customer or the market analysts. This is done through .SRS. . Software Requirement Specification document which consists of all the product requirements to be designed and developed during the project life cycle.

## 1.3.3. Design

In [systems design](https://en.wikipedia.org/wiki/Systems_design), the design functions and operations are described in detail, including screen layouts, business rules, process diagrams and other documentation. The output of this stage will describe the new system as a collection of modules or subsystems.

The design stage takes as its initial input the requirements identified in the approved requirements document. For each requirement, a set of one or more design elements will be produced as a result of interviews, workshops, and/or prototype efforts.

Design elements describe the desired system features in detail, and generally include functional hierarchy diagrams, screen layout diagrams, tables of business rules, business process diagrams, pseudo-code, and a complete entity-relationship diagram with a full data dictionary. These design elements are intended to describe the system in sufficient detail, such that skilled developers and engineers may develop and deliver the system with minimal additional input design.

## 1.3.4. Coding

In this stage of SDLC the actual development starts and the product is built. The programming code is generated as per DDS during this stage. If the design is performed in a detailed and organized manner, code generation can be accomplished without much hassle.

Developers have to follow the coding guidelines defined by their organization and programming tools like compilers, interpreters, debuggers etc are used to generate the code. Different high level programming languages such as C, C++, Pascal, Java, and PHP are used for coding. The programming language is chosen with respect to the type of software being developed.

## 

## 1.3.5. Testing

This stage is usually a subset of all the stages as in the modern SDLC models, the testing activities are mostly involved in all the stages of SDLC. However this stage refers to the testing only stage of the product where products defects are reported, tracked, fixed and retested, until the product reaches the quality standards defined in the SRS.

## 1.3.5.1. Some important Testing techniques

1. Black-Box Testing

2. White-Box Testing

3. System testing

4. Unit testing

5. Integration testing

## 

## 1.3.5.1.1. Black-Box Testing

The technique of testing without having any knowledge of the interior workings of the application is called black-box testing. The tester is oblivious to the system architecture and does not have access to the source code. Typically, while performing a black-box test, a tester will interact with the system's user interface by providing inputs and examining outputs without knowing how and where the inputs are worked upon.

## 1.3.5.1.2. White-Box Testing

White-box testing is the detailed investigation of internal logic and structure of the code. White-box testing is also called **glass testing** or **open-box testing**. In order to perform **white-box** testing on an application, a tester needs to know the internal workings of the code. The tester needs to have a look inside the source code and find out which unit/chunk of the code is behaving inappropriately.

## 

## 1.3.5.1.3. System Testing

System testing of software or hardware is testing conducted on a complete, integrated system to evaluate the system's compliance with its specified [requirements](https://en.wikipedia.org/wiki/Requirements). System testing falls within the scope of [black-box testing](https://en.wikipedia.org/wiki/Black-box_testing), and as such, should require no knowledge of the inner design of the code or logic.

As a rule, system testing takes, as its input, all of the "integrated" software components that have passed [integration testing](https://en.wikipedia.org/wiki/Integration_testing) and also the software system itself integrated with any applicable hardware system(s). The purpose of integration testing is to detect any inconsistencies between the software units that are integrated together (called *assemblages*) or between any of the *assemblages* and the hardware. System testing is a more limited type of testing; it seeks to detect defects both within the "inter-assemblages" and also within the system as a whole.

## 1.3.5.1.4. Unit Testing

**Unit testing** is a [level of software testing](http://softwaretestingfundamentals.com/software-testing-levels/) where individual units/ components of a software are tested. The purpose is to validate that each unit of the software performs as designed.

## 1.3.5.1.5. Integration Testing

The phase in [software testing](https://en.wikipedia.org/wiki/Software_testing) in which individual software modules are combined and tested as a group and It occurs after [unit testing](https://en.wikipedia.org/wiki/Unit_testing) and before [validation testing](https://en.wikipedia.org/wiki/Verification_and_validation_%28software%29). Integration testing takes as its input [modules](https://en.wikipedia.org/wiki/Module_%28programming%29) that have been [unit tested](https://en.wikipedia.org/wiki/Unit_testing), groups them in larger aggregates, applies tests defined in an integration [test plan](https://en.wikipedia.org/wiki/Test_plan) to those aggregates, and delivers as its output the integrated system ready for [system testing](https://en.wikipedia.org/wiki/System_testing).

## 1.3.6. Release and Maintenance

The [deployment](https://en.wikipedia.org/wiki/Software_deployment) of the system includes changes and enhancements before the decommissioning or sunset of the system. [Maintaining](https://en.wikipedia.org/wiki/Software_maintenance) the system is an important aspect of SDLC. As key personnel change positions in the organization, new changes will be implemented. There are two approaches to system development; there is the traditional approach (structured) and [object oriented](https://en.wikipedia.org/wiki/Object_oriented).

# TASK 02

# 2. Developing the feasibility report

## 2.1. Feasibility Study

A feasibility study is an analysis of the ability to complete a project successfully, taking into account legal, economic, technical, operational, scheduling, scope and other factors. Rather than just diving into a project and hoping for the best, a feasibility study allows project managers to investigate the possible negative and positive outcomes of a project before [investing](http://www.investopedia.com/terms/i/investing.asp) too much time and money.

## 2.1.1. The 05 main Feasibilities

1. Technology and System Feasibility
2. Economic Feasibility
3. Legal Feasibility
4. Operational Feasibility
5. Schedule Feasibility

## 2.1.1.1. Technology and System Feasibility

This assessment is based on an outline design of system requirements, to determine whether the company has the technical expertise to handle completion of the project. When writing a feasibility report, the following should be taken to consideration:

* A brief description of the business to assess more possible factors which could affect the study
* The part of the business being examined
* The human and economic factor
* The possible solutions to the problem

## 2.1.1.2. Economic feasibility

The purpose of the economic feasibility assessment is to determine the positive economic benefits to the organization that the proposed system will provide. It includes quantification and identification of all the benefits expected. This assessment typically involves a cost/ benefits analysis.

## 2.1.1.3. Legal feasibility

Legal feasibility is to determine whether the proposed system conflicts with the legal requirements.

**Example:** A data processing system must comply with the local data protection regulations and if the proposed venture is acceptable in accordance to the laws of the land

## 2.1.1.4. Operational feasibility

Operational feasibility is a measure of how well a proposed system solves the problems, and takes advantage of the opportunities identified during scope definition and how it satisfies the requirements identified in the requirements analysis phase of system development.

The operational feasibility assessment focuses on the degree to which the proposed development projects fits in with the existing business environment and objectives with regard to development schedule, delivery date, [corporate culture](https://en.wikipedia.org/wiki/Corporate_culture), and existing business processes.

## 2.1.1.5. Schedule feasibility

A project will fail if it takes too long to be completed before it is useful. Typically this means estimating how long the system will take to develop, and if it can be completed in a given time period using some methods like payback period. Schedule feasibility is a measure of how reasonable the project timetable is. Given our technical expertise, are the project deadlines reasonable? Some projects are initiated with specific deadlines. It is necessary to determine whether the deadlines are mandatory or desirable.

## 2.2. Feasibility Report

A feasibility report is a document that assesses potential solutions to the business problem or opportunity, and determines which of these are viable for further analysis.

Feasibility reports for complex technical ideas requiring large investments are more sophisticated than straightforward, low risk ones. In general, a feasibility report should address the marketing, technical, financial and organizational aspects of the idea and its positive and negative outcomes for the business.

## 2.2.1. Feasibility report of the Smart Arcade spare parts company

**Table of content**

1. Executive summary ……………….…………………………….. 28
2. Background of the company …….…………………………….… 28
3. Outline of the system ……………….…………………………… 29
4. Different Feasibility criteria ….…………………………………. 30
   1. Technological Feasibility ………..……....………………….. 30
      1. Hardware requirements ……………………………... 30
      2. Software requirements ……………………………… 30
   2. Economic Feasibility …………………………......…………. 31
      1. Cost or Benefit analyze ………….………………….. 31
   3. Legal Feasibility …………...………..………..……………... 31
   4. Operational Feasibility ……………………………………... 32
   5. Schedule Feasibility ……………………...………………… 33
   6. Estimation of SDLC Schedule …………………………...… 33
5. Conclusion ……………...……………...………………………... 33

**1. Executive summary**

This report is an analysis of viability of an idea through disciplined and documented process of thinking through the idea from its logical brining to its logical end. Feasibility report is carried out in order to assess the viability of a new project and it is primary and most important thing in development of a project. This report will consider, can we carry out the project with actual customer requirements, if we accept the project how about the cost of the development and time limit to complete the project. This report will help us to make decision whether take or drop rather than simply accept and struck in a half way. It helps to analyze the cost and duration for developing the each phase. This report helps us to identify, what are tools and resources have to be used in the project. Through this analysis we can decide whether continue or just drop.

**2. Background of the company**

Smart arcade is a car spare parts company which is manufacturing the spare parts and selling all over the world. Recently, the company has realized the obstacles of their business which are

1. The processes are getting delay
2. Misplacement of the customer orders
3. Customer service issues

I’m going to create the system for solve the above mentioned obstacles and this system is going to have some current trend features. Those are,

1. Consolidate your multiple sales channels, along with your Customer, Order, Inventory, and Marketing information all within this system.
2. Allow Customer Service Reps to have access to data at their fingertips for keeping in touch with their customers.
3. Search, display, export, update, and reference all customer data within one user interface for CSRs, sales, and marketing teams.
4. View your transactions and order statuses live, online, 24/7, while maintaining your customer’s order history for future up-selling and reporting.
5. Place orders, process drop shipments, and amend existing orders from a single domain or from multiple domains and sales channels via your system.

**3. Outline of the system**

Smart Arcade is car spare part manufacturing company based in UK. Smart Arcade is a global supplier of car spare parts to various car manufacturers in the world.

Customers of Smart Arcade place orders for the spare parts they require, and delivery is estimated to take three weeks. When a customer places an order, the model of spare part required, quantity and the destination to which it has to be shipped is recorded by the receptionist. Customers can order for more than one model of a spare part in a single order.

The customer orders are then being forwarded to the manufacturing department. At the end of the day, a forecast is made based on the customer orders and the spare part quantity in hand to determine the count of spare parts that have to be produced in the next line of manufacture.

Once the required quantity of spare parts is manufactured, they are quantity in hand is updated and the spare parts are subject to verification by the chief mechanic and upon success, they are being sent to the procurement department. The procurement officer is responsible for checking whether the model & quantity of spare parts manufactured supplements the customer orders and then does arrangements for delivery of spare parts to customers. The status of delivery is also recorded.

The operations of Smart Arcade are hindered by paper based operations which results in delays, misplacement of customer orders, customer service issues which ultimately leads to loss of potential customers. you have been hired by Smart Arcade as a systems analyst to propose a systems design that will help overcome limitations with the existing system.

**4. Different feasibility criteria on a system investigation**

1. Technology and System Feasibility
2. Economic Feasibility
3. Legal Feasibility
4. Operational Feasibility
5. Schedule Feasibility

**4.1. Technological Feasibility**

This system will be written in Visual C++ and also will have a Graphic user interface (GUI). It is capable to run with the average computer specifications such as below mentioned as well as future enhancements will be easy to comply into the system.

**4.1.1 Hardware requirements**

1. Motherboard – Asus ATX factor or current trend one
2. CPU – i3 or higher
3. Ram – 4 GB or higher
4. Hard drive – 500 GB or higher
5. VGA – 512 MB
6. Monitor – 19’’monitor or user wish
7. Printer – All-in-one HP LaserJet Pro MFP **( optional )**
8. Keyboard & mouse – Asus U3500 or user wish
9. Wireless Router – Asus RT-AC87U Router or User wish

**4.1.2 Software requirements**

1. Operating system – Windows 7 or higher edition
2. Antivirus – Vipre or user wish
3. Utility software – CC cleaner or user wish **( optional )**

**4.2. Economic Feasibility**

**4.2.1. Cost or Benefits analyze**

Through this new system, the company will get more profit and business growth by these things,

**To Smart Arcade,**

* 1. Smart Arcade can reduce or sustain the man power.
  2. No chance for reports misplaced or lost.
  3. Rapid customer service access will satisfy the customer needs and improve the reputation of the company.
  4. This system will be completed its intended tasks in a seconds therefore; the Smart Arcade orders will not be getting any delay or struck in a middle.
  5. It’s capable to generate the reports and those reports are more accurate and doesn’t consist the calculation errors which human do.

**To Developer,**

The total cost of this system is Rs. 249,950.00 in Srilankan money or £ 1700.00. The initial payment 25% of the money has to be paid once the agreement is signed and rest of the amount to be paid on the every software development phase completion. In addition, the late requirement changes may cost extra.

**4.3. Legal Feasibility**

The process of check and verify that the system’s functions are suitable for the Government Laws and regulations. Once the system is completed that should have to be registered under UK e-Commerce business act because the company is based in UK and also it will be registered under some criteria. Those are Copy rights, trade mark, patent rights and confidential information. Only the authorized persons or company will be able to access the system. Legal action will be taken against the misuse or pirates activity.

**4.4. Operational Feasibility**

Operational feasibility is dependent on how the new system is going to solve the current obstacles.

**Obstacles,**

1. Delay on process
2. Misplacing the customer orders
3. Customer service issues

**Solutions,**

1. **Delays:** Computerized system will be completed the intended task in a seconds. Because it will not get tired or frustrated. Even we can boost up its speed with upgrading higher grade components.
2. **Misplacement:** This system is fully automated with the structured way therefore, no more misplacement customer orders in future.
3. **Customer service issues:** Customers will be satisfied with new system’s rapid access and immediate responds.

**To developer,**

For the operational feasibility, I’m going to arrange the 05 software engineers and 01 experienced leader for the system development. On the other hand, I expected to collaborate company staffs (who are going to use the system) with our developing team because I want to know whether the staffs are clearly understand the system or not. This gathering will be arranged once in a month or in a convenient time of the related parties.

**4.5. Schedule Feasibility**

Assessing schedule feasibility is to assess the duration of the project whether it is too long to be complete before it is useful. System analysts have to estimate how long the system will take to develop, and whether all potential timeframes and the completion date schedules can be met, as well as whether meeting these date will sufficient for dealing with the needs of the organization.

I expected to finish up the system by 06 months of time and each phase will take approximately 02 weeks to 02 month to complete its intended task. The user (staffs from the company) meetings will be arranged based on the user convenient times which is mandatory for develop the system properly and avoid the unwanted changes in future.

**4.6. Estimation of SDLC Schedule**

1. Requirement Gathering - 2 weeks
2. Analysis and Specification - 2 weeks
3. Designing - 2 month
4. Coding - 2 month
5. Testing - 1 month
6. Release and Maintenance - 2 weeks

**Note:** Hopefully, we are expecting to finish up the whole thing in 05 months of time.

**Conclusion:**

According to this feasibility report, Developers and clients will have the clear knowledge on feasibility criteria. Those are technological, economical, legal, operational and schedule. I hope this feasibility report will help to make a decision whether proceed or not.

# TASK 03

# 3.1. Developing a Systems Requirements Specification (SRS)

A software requirements specification (SRS) is a description of a software system to be developed. It lays out functional and non-functional requirements, and may include a set of use cases that describe user interactions that the software must provide.

## 3.1.1. The purpose of the SRS is to:

1. **Establish the basis for agreement between the customers and the suppliers on what the software product is to do.** The complete description of the functions to be performed by the software specified in the SRS will assist the potential user to determine if the software specified meets their needs or how the software must be modified to meet their needs
2. **Provide a basis for developing the software design.** The SRS is the most important document of reference in developing a design
3. **Reduce the development effort.** The preparation of the SRS forces the various concerned groups in the customer's organization to thoroughly consider all of the requirements before design work begins. A complete and correct SRS reduces effort wasted on redesign, recoding and retesting. Careful review of the requirements in the SRS can reveal omissions, misunderstandings and inconsistencies early in the development cycle when these problems are easier to correct
4. **Provide a basis for estimating costs and schedules.** The description of the product to be developed as given in the SRS is a realistic basis for estimating project costs and can be used to obtain approval for bids or price estimates
5. **Provide a baseline for**[**validation and verification**](http://www.chambers.com.au/glossary/verification_validation.php)**.** Organizations can develop their test documentation much more productively from a good SRS. As a part of the development contract, the SRS provides a baseline against which compliance can be measured
6. **Facilitate transfer.** The SRS makes it easier to transfer the software product to new users or new machines. Customers thus find it easier to transfer the software to other parts of their organization and suppliers find it easier to transfer it to new customers
7. **Serve as a basis for enhancement.** Because the SRS discusses the product but not the project that developed it, the SRS serves as a basis for later enhancement of the finished product. The SRS may need to be altered, but it does provide a foundation for continued product evaluation.

## 

## 3.2. SRS of the Smart Arcade Company

This SRS has integrated with user and system requirements in a single description.

**Table of content**

* 1. Introduction …………………………………………….. 36
  2. Purpose …………………………………………………. 36
  3. Scope ………………………………………………….... 36
  4. Hardware Requirement ………………………………… 36
  5. Software Requirement ……………………………….… 37
  6. Functional Requirement ……………………………….. 37
  7. Non functional Requirement …………………………... 37
  8. Disadvantages of current system ……………………… 38
  9. Conclusion …………………………………………..… 38

**1. Introduction**

Smart Arcade is the car spare part manufacturing company and it has a customer all over the world. The company is handling their works in a paper based system. Due to the paper based system that the company is having some major obstacles. The company is expecting to solve their obstacles by implementing a computer based system.

**2. Purpose**

The purpose of this document is to give a detailed description of the requirements for the Smart Arcade spare parts system/software. It will illustrate the purpose and complete declaration for the development of system. It will also explain system hardware, software, functional and non functional requirements. This document is primarily intended to be proposed to a client for its approval and a reference for developing the first version of the system for the development team.

**3. Scope**

The Scope of the system is to solve the current obstacles and enhance the system for future challenges. Through this new system, we expecting to reduce the working load of the staffs and giving fast and flexible services to the customers.

**4. Hardware Requirements**

1. Motherboard – Asus ATX factor or current trend one
2. CPU – i3 or higher
3. Ram – 4 GB or higher
4. Hard drive – 500 GB or higher
5. VGA – 512 MB
6. Monitor – 19’’monitor or user wish
7. Printer – All-in-one HP LaserJet Pro MFP **( optional )**
8. Keyboard & mouse – Asus U3500 or user wish
9. Wireless Router – Asus RT-AC87U Router or User wish

**5. Software Requirements**

1. Operating system – Windows 7 or Greater versions
2. Antivirus – Vipre or User wish
3. Utility software – CC cleaner or User wish **( optional )**

**6. Functional Requirement**

1. Record the business details
2. User-friendly interfaces
3. Ledgers handling
4. Search particular customer or customer order by id / name / date
5. Modify the customer details
6. Generates reports automatically
7. Back up and store the business details safely

**7. Non Functional Requirement**

1. **Performance/Time**

The processing time should be taken less than1 seconds since the request given.

1. **Usability**

Usability is a crucial point in the system. As most of the users/staffs who don’t like to adapt a new type of systems and they are expecting to use the system in a way like a traditional word processor. This system should simulate the traditional operating system and word processor.

# 

1. **Security**

This system will be secured from SQL injections and internal antivirus will handle the virus activity. The credential right has been categorized as a Superior level and Operational level.

**Superior Level:** Full access of the system.

**Operational Level:** Restriction level can be assigned by Superior level.

1. **Reliability**

This system is available for 24/7 and during any interruption, it will not loss any datas because it is designed with auto saving and auto recovery modes which will records each and every single events of the software.

1. **Reusability**

It will be designed for adapt a future enhancement.

**8. Disadvantages of the current system**

**­**

Smart arcade is car spare parts manufacturing company which is dealing the business all over the world. Currently they are being used the manual or paper based system which causes lot of obstacles of the company’s routine job. Let’s see, what are the defects that the company has been facing due to the manual system?

* Inconsistency in data entry, room for errors, miskeying information.
* Large on going staff training cost.
* System is dependent on good individuals.
* Reduction in sharing information and customer services.
* Time consuming and costly to produce reports.

**Conclusion**

I hope, through this report that the reader can understand what problems are currently having and how new system is going to full fill their expectations. At the mean time, company can figure out the financial benefits of the new system.

# TASK 04

# 4. Use of Methodology

## 4.1. Entity Relationship Diagram (ERD)

An entity-relationship diagram (ERD) is a data modeling technique that graphically illustrates an information system's entities and the relationships between those entities. An ERD is a conceptual and representational model of data used to represent the entity framework infrastructure.

## 4.1.1. Entity Relationship Diagram’s Symbols

# C:\Users\Shan89\Desktop\Chen-ERD-symbols.png

## 4.1.2. Entity Relationship Diagram of the system

## C:\Users\Nazi\Desktop\ER.PNG

## 4.2. Data Flow Diagram

Data flow diagram is graphical representation of flow of data in an information system. It is capable of depicting incoming data flow, outgoing data flow and stored data. The DFD does not mention anything about how data flows through the system.

There is a prominent difference between DFD and Flowchart. The flowchart depicts flow of control in program modules. DFDs depict flow of data in the system at various levels. DFD does not contain any control or branch elements.

## 4.2.1. Types of DFD

1. Logical DFD
2. Physical DFD

## 

## 4.2.1.1. Data Flow Diagrams are either Logical or Physical.

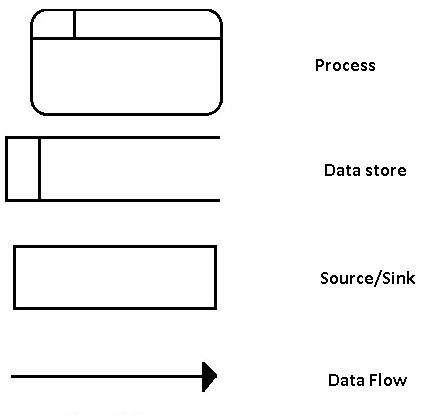
* **Logical DFD** - This type of DFD concentrates on the system process, and flow of data in the system. For example in a Banking software system, how data is moved between different entities.
* **Physical DFD** - This type of DFD shows how the data flow is actually implemented in the system. It is more specific and close to the implementation.

## 4.2.2. Derivation of DFD

1. Level 0 DFD - Context Diagram
2. Level 1 DFD
3. Level 2 DFD

***Note: This diagram can go further level according to the requirements*.**

## 4.2.3. Data Flow Diagram’s Notations - Gane & Sarson method



## 4.2.4. Context Diagram of the system (DFD Level 0)

## C:\Users\Nazi\Desktop\dfd level 0.png

## 

## 4.2.5. DFD Level 1 Diagram of the system

## C:\Users\Nazi\Desktop\ddd.tif

## 4.2.6. DFD Level 2 Diagram of the system

**Order Process**

## C:\Users\Shan89\Desktop\Snart arcade Diagrams\DFD 2\dfd2.PNG

# 

# TASK 05

# 5.1. Quality Management

Software quality management is a management process the goal of which is to develop and manage the quality of software to make sure the product satisfies the user. The goals of SQM (software quality management ) is to make sure a product is in good levels of quality and meets the regulations and standards the customer needs from the product. Software quality managers have to test the product before it's released to the market, they do this by a series of steps called the software cycle in order to get the errors in software and try to fix it before being released.

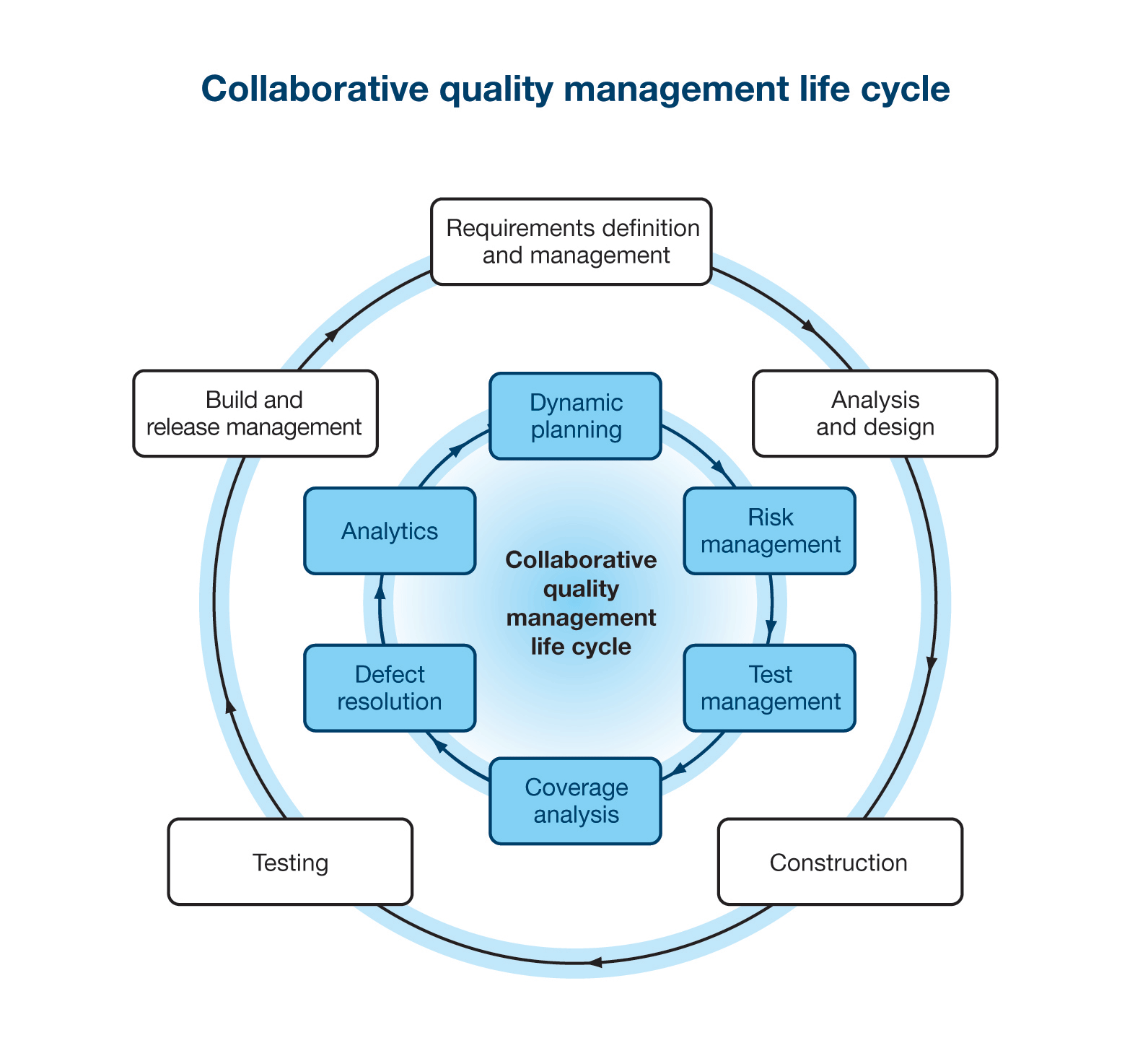


Figure 9

## 5.1.1. Top 5 Reasons to Implement a Quality Management

1. Improve the Organizational Effectiveness
2. Improve the Customer Satisfaction
3. Improve the Compliance
4. Improve the Organizational Culture
5. Improve the Documentation

## 5.1.2. There are 03 types of process

1. Quality Planning – QP

2. Quality Assurance – QA

3. Quality Control – QC

## 5.1.2.1. Quality Planning

A quality plan is a document, or several documents, that together specify quality standards, practices, resources, specifications, and the sequence of activities relevant to a particular product, service, project, or contract. Quality plans should define:

* Objectives to be attained (for example, characteristics or specifications, uniformity, effectiveness, aesthetics, cycle time, cost, yield, dependability, and so on)
* Steps in the processes that constitute the operating practice or procedures of the Smart Arcade.
* Allocation of responsibilities, authority, and resources during the different phases of the process or project
* Specific documented standards, practices, procedures, and instructions to be applied.
* Suitable testing, inspection, examination, and audit programs at appropriate stages

## 5.1.2.3. Quality Assurance

Quality assurance (QA) is a process that ensures that developed software meets and complies with defined or standardized quality specifications of the Smart Arcade. QA is an ongoing process within the software development life cycle (SDLC) that routinely checks the developed software to ensure it meets desired quality measures.

## 5.1.2.2. Quality Control

It is the set of procedures used by organizations to ensure that a software product will meet its quality goals at the best value to the Smart Arcade, and to continually improve the organization’s ability to produce software products in the future.

Software quality control refers to specified functional requirements as well as non-functional requirements such as supportability, performance and usability. It also refers to the ability for software to perform well in unforeseeable scenarios and to keep a relatively low defect rate.

Software development requires quality control.

# 5.2. The Quality Assurance and Control of the Smart Arcade

## 5.2.1. Software Quality Assurance and Control

**1. Operating system - Windows**

This software is capable to run with Windows 7 or 8 or 10 versions therefore, the user doesn’t need to change their current operating system and also this software will not contain a higher capacity.

**2. Antivirus – Vipre or User wish**

In order to secure our Datas from the Virus activity, we should have to use Anti-virus because the client (Smart Arcade) may use the internet or removable disks therefore, the computer has a frequent chance to have a virus or malware attacks.

**3. Utility software – CC cleaner or User wish (optional)**

The Utility softwares are taking care of the computer maintenance such as analyze, configure, optimize and boost up the computer speed and gain the internet speed by clear the cookies and temporary files.

## 5.2.2. Hardware Quality Control and Assurance

**1. Motherboard – Asus ATX factor or current trend one**

Mother board is a component which integrates the functions of the other components. In order to get a full usage of the other components that the user has to use the higher grade mother board such as above mentioned or similar one.

**2. CPU – i3 or higher**

The processor can be considered to be the “engine” of the computer because it executes information processing. i3 or higher range of processors will make a room for run our software rapidly and smoothly.

**3. RAM – 4 GB or higher**

RAM is temporary storage which loads the softwares in it and let the processors to run it. When we have a higher capacity of RAM which will make our software run smoothly therefore, I recommend the Ram 4 GB or higher than that.

**4. Hard drive – 500 GB or higher**

While the softwares are running, they need a higher transferring speed to generate their output instantly and this company’s datas capacity will be increased day by day basis therefore, I recommend minimum 500 GB hard disk for the company.

**5. VGA – 512 MB**

Higher capacity VGA card helps to produce the clear and quality display outputs Therefore, I recommend the VGA card.

**6. Monitor – 19’’monitor or User wish**

The 19’’ monitor ideally suitable for the solution software and some special features are higher resolution of 1280 x 1024 and light emission technology which helps to see the display crystal clearly Therefore, I recommend this monitor.

**7. Printer – All-in-one HP LaserJet Pro MFP (Optional)**

This printer will be used for printing out the reports and other contents and also it has some special features which are scanner, fax, can be printed via blue tooth or wifi and so on.

**8. Keyboard & mouse – Asus U3500 or user wish**

I recommend the Asus U3500 Key board & Mouse set for Smart Arcade because it is the best and long lasting brand.

**9. Wireless Router – Asus RT-AC87U Router or User wish**

It will enable to access internet and share the informations to others over the Network.

Conclusion

This solution software has developed based on the future enhancement basis. In other words that the owner easily can upgrade the software or change the platform into web based. For the future needs, I prefer the owner to choose the web based system because current solution is using the local database to store their data and can not be accessed remotely.

**The benefits of web based system over the stand alone system**

1. No needs for any further installation, can be accessed on browsers.
2. Cross platform compatibility.
3. Possible to connect anywhere at any time.
4. Data back up facilities such as : cloud storage, store on the remote servers.

Here, I have given some important benefits of the web based system and the future trend relies on it.

Eventually, I reckon that I have a clear knowledge on basic software development. Those are how to gathering a software requirement, SDLC models, feasibility study, feasibility reports, database diagrams and quality management of the software. Therefore, I feel very confident and I would like to take this opportunity to express my thanks to **Esoft** and **Pearson.**

Appendix

Sample Questionnaire of the proposed system

Note : One of the sample hard copy of the filled questionnaire has converted into electronic format.

Name : Bernard Shaw Job description : General Manager

1. What are the key objectives of the Proposed system?

1. The processes are getting delay.

2. Misplacement of the customer orders.

3. Customer service issues.

1. Do you have any related existing computer based system? If yes, Please specify the details

No, existing system.

1. What kind of system that you require?
   1. Stand alone [ x ]
   2. Web based [ ]
2. What type of interface that you prefer?
3. Command line interface / CLI [ ]
4. Graphical user interface / GUI [ x ]
5. Do you have specific color or logo that need to be used? If yes, Please specify the details

Yes, Smart Arcade Logo.

1. What is the user Operating system?

Windows

1. Where do you want to store your data?
   1. Local database [ x ]
   2. Remote servers [ ]
2. Do you want to maintain a inventory system? If yes, Please specify the details.

Yes and inventory details will be provided as soon as possible.

1. Do you need to process credit cards or other financial data?

For the moment, we are not going to do that.

1. Who will use the system?
   1. Receptionist
   2. Manager
2. How many users are going to use the system?

Less than 3 users.

1. What kind of security features that you want to be implemented?
   1. Username & password [ x ]
   2. Thumb impression [ ]
   3. Voice recognition [ ]
2. Do you have any ideas or example something like in your mind? If yes Please specify the details.

No idea.

1. What is the time line of the project?

06 months.

1. What is the budget of the system?

£ 1700.00

## Harvard Reference - Web pages

1.1.1The Business Communication. 2016. *Advantage and disadvantage of interview*. [ONLINE] Available at:<http://thebusinesscommunication.com/advantage-and-disadvantage-of-interview/>. [Accessed 02 Mar 2016].

1.1.2 Online Surveys - Pros and cons of web-based questionnaires. 2016. *Online Surveys - Pros and cons of web-based questionnaires*. [ONLINE] Available at: <https://explorable.com/online-surveys>. [Accessed 02 Mar 2016].

1.1.3 Snap Surveys Blog. 2016. *Advantages and Disadvantages of Surveys*. [ONLINE] Available at:<http://www.snapsurveys.com/blog/advantages-disadvantages-surveys/>. [Accessed 2 Mar 2016].

1.1.4 Features, Advantages and Disadvantages of Observation | Total MBA Guide. 2016. *Features, Advantages and Disadvantages of Observation | Total MBA Guide*. [ONLINE] Available at:<http://www.mbaofficial.com/mba-courses/research-methodology/features-advantages-and-disadvantages-of-observation/>. [Accessed 02 Mar 2016].

1.2 SDLC - Overview. 2016. SDLC - Overview. [ONLINE] Available at: <http://www.tutorialspoint.com/sdlc/sdlc_overview.htm>. [Accessed 02 Mar 2016].

1.2.2 What is Waterfall model- advantages, disadvantages and when to use it?. 2016. What is Waterfall model- advantages, disadvantages and when to use it?. [ONLINE] Available at: <http://istqbexamcertification.com/what-is-waterfall-model-advantages-disadvantages-and-when-to-use-it/>. [Accessed 02 Mar 2016].

1.2.3 Testing Excellence. 2016. Incremental Model - Advantages and Disadvantages. [ONLINE] Available at: <http://www.testingexcellence.com/incremental-model/> [Accessed 03 Mar 2016].

1.2.4 Software Evolution: Past, Present and Future | American journal of Engineering Research (AJER) - Academia.edu. 2016. Software Evolution: Past, Present and Future | American journal of Engineering Research (AJER) - Academia.edu. [ONLINE] Available at: <http://www.academia.edu/7296972/Software_Evolution_Past_Present_and_Future>

[Accessed 03 Mar 2016].

1.2.5 What is RAD model- advantages, disadvantages and when to use it?. 2016. What is RAD model- advantages, disadvantages and when to use it?. [ONLINE] Available at: <http://istqbexamcertification.com/what-is-rad-model-advantages-disadvantages-and-when-to-use-it/>. [Accessed 03 Mar 2016].

1.2.6 What is Spiral model- advantages, disadvantages and when to use it?. 2016. What is Spiral model- advantages, disadvantages and when to use it?. [ONLINE] Available at: <http://istqbexamcertification.com/what-is-spiral-model-advantages-disadvantages-and-when-to-use-it/>. [Accessed 03 Mar 2016].

1.2.7 What is Agile model – advantages, disadvantages and when to use it?. 2016. What is Agile model – advantages, disadvantages and when to use it?. [ONLINE] Available at: <http://istqbexamcertification.com/what-is-agile-model-advantages-disadvantages-and-when-to-use-it/>. [Accessed 04 Mar 2016].

1.2.8 10 Good Reasons To Do Agile Development | All About Agile. 2016. 10 Good Reasons To Do Agile Development | All About Agile. [ONLINE] Available at: <http://www.allaboutagile.com/10-good-reasons-to-do-agile-development/>. [Accessed 04 Mar 2016].

1.2 Search Software Quality. 2016. What is systems development life cycle (SDLC)? - Definition from WhatIs.com. [ONLINE] Available at: <http://searchsoftwarequality.techtarget.com/definition/systems-development-life-cycle> [Accessed 04 Mar 2016].

1.2.2. SDLC - Overview. 2016. SDLC - Overview. [ONLINE] Available at: <http://www.tutorialspoint.com/sdlc/sdlc_overview.htm>. [Accessed 04 Mar 2016].

1.2.4. SDLC - Overview. 2016. SDLC - Overview. [ONLINE] Available at: <http://www.tutorialspoint.com/sdlc/sdlc_overview.htm>. [Accessed 06 Mar 2016].

2.1 Feasibility Study Definition | Investopedia. 2016. Feasibility Study Definition | Investopedia. [ONLINE] Available at: <http://www.investopedia.com/terms/f/feasibility-study.asp>. [Accessed 07 Mar 2016].

2.2 What is a feasibility report?. 2016. What is a feasibility report?. [ONLINE] Available at: <http://www.ask.com/business-finance/feasibility-report-e50affa556c0fc6a>. [Accessed 09 Mar 2016].

2.2 osarome ogbebor: 1. TECHNICAL FEASIBILITY 2. OPERATIONAL FEASIBILITY 3. ECONOMIC FEASIBILITY. 2016. osarome ogbebor: 1. TECHNICAL FEASIBILITY 2. OPERATIONAL FEASIBILITY 3. ECONOMIC FEASIBILITY. [ONLINE] Available at: <http://osarome.blogspot.com/2011/10/1-technical-feasibility-2-operational.html>. [Accessed 10 Mar 2016].

3.1 Search Software Quality. 2016. What is software requirements specification (SRS)? - Definition from WhatIs.com. [ONLINE] Available at: <http://searchsoftwarequality.techtarget.com/definition/software-requirements-specification>

[Accessed 11 Mar 2016].

3.1 ProjectManagement.com - System Requirements Specification. 2016. ProjectManagement.com - System Requirements Specification. [ONLINE] Available at: <http://www.projectmanagement.com/deliverables/238236/System-Requirements-Specification>. [Accessed 12 Mar 2016].

3.2 SRS - Software Requirements Specification | Software Specification | Application Development | Requirements and Specifications | Software Engineering. 2016. *SRS - Software Requirements Specification | Software Specification | Application Development | Requirements and Specifications | Software Engineering*. [ONLINE] Available at:<http://www.chambers.com.au/glossary/software_requirements_specification.php>. [Accessed 13 Mar 2016].

5.1 Revolvy, LLC. 2016. Software quality management. [ONLINE] Available at:

<http://www.revolvy.com/main/index.php?s=Software%20quality%20management> [Accessed 20 Mar 2016].

5.1 Best Quality Management Software | 2016 Reviews of the Most Popular Systems. 2016. Best Quality Management Software | 2016 Reviews of the Most Popular Systems. [ONLINE] Available at: <http://www.capterra.com/quality-management-software/>. [Accessed 21 Mar 2016].

5.1.1 Top 5 Reasons to Implement a Quality Management System — Quality Management System. 2016. *Top 5 Reasons to Implement a Quality Management System — Quality Management System*. [ONLINE] Available at: <http://qualitymanagementsystem.com/top-5-reasons-to-implement-a-quality-management-system/>. [Accessed 21 Mar 2016].

5.1.2.1 Quality Plans. 2016. *Quality Plans*. [ONLINE] Available at: <http://asq.org/learn-about-quality/quality-plans/index.html>. [Accessed 22 Mar 2016].

5.1.2.2 Benefits of ISO 9000. 2016. Benefits of ISO 9000. [ONLINE] Available at: <http://www.pjr.com/standards/iso-90012008/benefits-of-iso-9000> [Accessed 22 Mar 2016].

5.1.2.3 Techopedia.com. 2016. *What is Software Quality Assurance (SQA)? - Definition from Techopedia*. [ONLINE] Available at: <https://www.techopedia.com/definition/4363/software-quality-assurance-sqa>. [Accessed 24 Mar 2016].

## Harvard Reference - Pictures

Figure 1 2016. . [ONLINE] Available at: <http://cdn1.hubspot.com/hub/167634/file-296310301.jpg>. [Accessed 02 Mar 2016].

Figure 2 . 2016. . [ONLINE] Available at: <http://istqbexamcertification.com/wp-content/uploads/2012/01/Waterfall-model.jpg>. [Accessed 02 Mar 2016].

Figure 3 . 2016. . [ONLINE] Available at: <http://istqbexamcertification.com/wp-content/uploads/2012/01/Incremental_model.jpg>. [Accessed 04 Mar 2016].

Figure 4 . 2016. . [ONLINE] Available at: <http://istqbexamcertification.com/wp-content/uploads/2012/01/Prototype-model.jpg>. [Accessed 04 Mar 2016].

Figure 5 . 2016. . [ONLINE] Available at: <http://3.bp.blogspot.com/-dvQKTQ2Sn7Q/UyxVq0IBdKI/AAAAAAAAAC8/LqEvhu3i5Wg/s1600/13.PNG>. [Accessed 04 Mar 2016].

Figure 6 . 2016. . [ONLINE] Available at: <https://upload.wikimedia.org/wikipedia/commons/thumb/e/ec/Spiral_model_%28Boehm,_1988%29.svg/333px-Spiral_model_%28Boehm,_1988%29.svg.png>. [Accessed 05 Mar 2016].

Figure 7 . 2016. . [ONLINE] Available at: istqb. 2015. istqbexamcertification. [ONLINE] Availableat: http://istqbexamcertification.com/wp-content/uploads/2012/01/Agile-model.jpg. [Accessed 05 Mar 2016].

Figure 8 . 2016. . [ONLINE] Available at: <http://www.tutorialspoint.com/sdlc/images/sdlc_stages.jpg>. [Accessed 06 Mar 2016].

Figure 9 2016. [ONLINE] Available at:<http://twimgs.com/ddj/images/article/2010/1110/MosheFig2.gif>. [Accessed 22 Mar 2016].

# 

# Gantt chart

