3/23/2019

Kyer Potts

30003389

ACME FINANCIAL SERVICES

Software Report

Table of Contents

[Organisational Report 1](#_Toc4603019)

[Admin 1](#_Toc4603020)

[Accounts 1](#_Toc4603021)

[Sales 1](#_Toc4603022)

[International 1](#_Toc4603023)

[Technical Document Report 2](#_Toc4603024)

[Problem Statement: 2](#_Toc4603025)

[Technical Documentation Reference Report: 3](#_Toc4603026)

[Requirement Report 4](#_Toc4603027)

[Introduction 4](#_Toc4603028)

[User Requirements Definition 4](#_Toc4603029)

[Functional Requirements 5](#_Toc4603030)

[Non-Functional Requirements 6](#_Toc4603031)

[Recommendations 6](#_Toc4603032)

[Summary 6](#_Toc4603033)

[Feasibility Report 7](#_Toc4603034)

[Product 7](#_Toc4603035)

[Technical Feasibility 7](#_Toc4603036)

[Performance 7](#_Toc4603037)

[Operational Characteristics 7](#_Toc4603038)

[Scalability 7](#_Toc4603039)

[Technical Feasibility Summary 7](#_Toc4603040)

[Operational Feasibility 8](#_Toc4603041)

[Organisational Requirements 8](#_Toc4603042)

[Operational Effects 8](#_Toc4603043)

[Employee Response 8](#_Toc4603044)

[Operational Feasibility Summary 8](#_Toc4603045)

[Economic Feasibility 9](#_Toc4603046)

[Costs 9](#_Toc4603047)

[Benefits 9](#_Toc4603048)

[Economic Feasibility Summary 9](#_Toc4603049)

[Alternative Solutions 10](#_Toc4603050)

[Microsoft Azure 10](#_Toc4603051)

[Microsoft SharePoint 10](#_Toc4603052)

[Alternative Solution Summary 10](#_Toc4603053)

[Software Solution Report 11](#_Toc4603054)

[Introduction 11](#_Toc4603055)

[Software Development Life Cycle 11](#_Toc4603056)

[Planning and Requirement Analysis 11](#_Toc4603057)

[Defining Requirements 11](#_Toc4603058)

[Designing the Product Architecture 11](#_Toc4603059)

[Build and Development 11](#_Toc4603060)

[Testing 11](#_Toc4603061)

[Deployment and Maintenance 12](#_Toc4603062)

[Option One: Waterfall Model 13](#_Toc4603063)

[Waterfall Model Pros 13](#_Toc4603064)

[Waterfall Model Cons 13](#_Toc4603065)

[Option two: Agile Model 14](#_Toc4603066)

[Agile Model Pros 14](#_Toc4603067)

[Agile Model Cons 14](#_Toc4603068)

[Summary 15](#_Toc4603069)

[Recommendation 15](#_Toc4603070)

[Risk Assessment Report 16](#_Toc4603071)

[Development of Incorrect User Interface 16](#_Toc4603072)

[Introduction 16](#_Toc4603073)

[Strategies 16](#_Toc4603074)

[Surveying and End User Analytics 16](#_Toc4603075)

[Development and Testing 16](#_Toc4603076)

[Review and Implementation 16](#_Toc4603077)

[Recommendations 17](#_Toc4603078)

[Gold Plating 18](#_Toc4603079)

[Introduction 18](#_Toc4603080)

[Strategies 18](#_Toc4603081)

[Recommendations 18](#_Toc4603082)

[Changes to Requirements 19](#_Toc4603083)

[Introduction 19](#_Toc4603084)

[Strategies 19](#_Toc4603085)

[Recommendations 19](#_Toc4603086)

[Signatures 20](#_Toc4603087)

[Client Sign-Off 20](#_Toc4603088)

[ATEX Project Manager Sign-Off 20](#_Toc4603089)

[ATEX Technical Lead Sign-Off 20](#_Toc4603090)

[References 21](#_Toc4603091)

# Organisational Report

## Admin

The administration department comprises four staff. The staff perform day-to-day organisational duties within the company. The office assistants also assist the rest of the business with office related tasks such as copying, printing etc.

## Accounts

The accounts department comprises four staff. The department oversees account information, payroll and payroll and staff expenditures. The department also handles the accounting and purchasing functions of the business.

## Sales

The sales department comprises four staff. This department maintains local client relationships, builds the local client portfolio and manages sales to local clients.

## International

The international department comprises two staff. The department handles international clients, builds the international client portfolio, maintains client relationships, and manages sales to international clients.

# Technical Document Report



## Problem Statement:

The problem with the current file management system, and the lack of direct customer interface is affecting business revenue, causing a decline in profits and reducing the client portfolio. This is impacting long term business and is allowing competing business to gain an edge within the market.

To solve this, a new file management system would be implemented, which also streamlines customer interface and accessibility, and improves information flow within the business.

## Technical Documentation Reference Report:

ISO/IEC 26514 is a technical documentation standard developed by the International Electrotechnical Commission in collaboration with the International Organisation for Standardisation. This standard was created to provide specifications and guidelines for the design of software user documentation implemented from the viewpoint of the software developer. The standard outlines and specifies the structure, content and format for user documentation, and provides guidance for user documentation style.

ISO/IEC 26514 is available for purchase in English for $323.31 Hardcopy, or $290.98 PDF.

It is available for purchase through:

https://infostore.saiglobal.com/en-au/Standards/ISO-IEC-26514-2008-589528\_SAIG\_ISO\_ISO\_1350435/

# Requirement Report

## Introduction

This report will outline the requirements for the software that will address the problems framed within the problem statement submitted prior to project approval. The requirements suggested within this document were compiled via a focus group implemented by the CEO and managers of ACME Financial Services to determine the functions of the software and the formats in which the functions will operate.

## User Requirements Definition

|  |  |  |
| --- | --- | --- |
| Use Case | Actor | Description |
| Log On | Client | Client logs securely onto the system |
| Edit Client Information | Client | The client can edit any personal information for the purposes of updating records |
| Display Transactions | Client, Staff, Managers | The Actor can view financial history, Managers and staff are able to view financial reports and further information regarding the client’s financial history |
| Print Transactions | Client | The client can print historical transactions in PDF format |
| Recover Login Information | Client | The client can access lost login information via secure methods. |
| Join E-mail | Client | Allows the client to opt into email notifications of new products |
| Suspend Client | Staff | Allows staff to suspend clients and lock their respective accounts |
| Access Database | Staff | Sales team members in the field will be able to remotely access client database |
| Manage Privileges | Managers | Managerial staff will be able to change the privilege levels of staff. |
| Create Client Profile | Sales Staff | Sales staff will be able to create new Client user profiles depending on sales outcomes |



## Functional Requirements

* Clients must log on to access data
* Business Rules
* Transaction corrections, adjustments and cancellations
* Administrative Functions
* Authentication
* Authorisation levels
* Audit Tracking
* External Interfaces
* Certification Requirements
* Reporting Requirements
* Historical Data
* Legal or Regulatory Requirements

## Non-Functional Requirements

* Web portal will be the first point of entry for the public/clients
* Clients will be able to log into their accounts via the web portal
* Clients will be able to edit personal information
* Clients will only be able to view their own information
* Historical transactions can be accessed by clients, staff and managers
* Historical transactions cannot be edited, but can be printed in PDF format
* Web portal will display live information about stock and currency markets
* Web portal will display various services provided by sales team
* Web portal will offer a help desk option to recover lost login information
* Database will be accessible remotely via mobile devices
* Database will offer rapid response times and current data
* Database will have hierarchical security for all users
* Managers will have administrator rights to partition privileges to users
* Web interface will be accessible for smart phones (android and IOS), 7” and 10” tablets, and pc’s
* Database interface will provide options to lock and suspend client accounts, and display financial history reports
* Database interface will provide options to update transaction and client details.

## Recommendations

* Database redundancies for system failure
* Personalised contact information provided to clients for specific sales representatives

## Summary

By implementing a new streamlined client interface and information storage database, Acme will be able to maintain a competitive edge within the market. The new web portal will also act to lead in prospective new clients.

# Feasibility Report

## Product

Acme Financial Solutions have engaged the services of ATEX IT to provide a cohesive file management database with front and backend user functionality. This solution will be implemented with functionality across a range of devices, with offsite access to database services. This product will be developed in response to declining revenue streams due to a lack of client retention and high staffing costs caused by archaic file management procedures.

## Technical Feasibility

### Performance

The proposed software solution will need to have relevant characteristics to ensure that performance targets are achieved throughout the system, this includes bandwidth operations for user to server communication, speed and reliability optimisation and operational uptime. Considering the constraints and the relatively small scope of function of the system, the technology requirements to implement the system fall well within currently available technology.

### Operational Characteristics

The product will need to have a dedicated information storage system in place that will interface with the web portal and hold user accounts and personal information. This will also need to be accessible remotely for travelling staff. Cloud based database servers are becoming increasingly popular to manage information and offer greater functionality and portability over traditional static server-based solutions. Several integratable database management software packages are available to interface with cloud-based data management systems and implementing personalised changes to the software to ensure that it is business relevant would not pose a great deal of difficulty for the team. Many cloud-based solutions require very little upkeep and allow redundancy measures to be implemented on site. This will result in little if any downtime for server maintenance related duties.

### Scalability

The scope of works has defined that the proposed solution will need to operate on several devices effectively, particularly mobile devices that operating on specialised mobile device systems. Portability of software regarding scalability will be key in implementing the tech effectively across these devices. Fortunately, specialised programming languages designed to allow software portability are widely available and will allow the technology to be natively run within relevant runtime environments on relevant devices.

A cloud-based data management system will also off increased viability over a conventional server as it will allow scalable data capture and storage methods. As data sizes increase over time, cloud-based servers will allow relevant increases in capacity of storage to allow adequate data storage and speed of use.

### Technical Feasibility Summary

Considering the implementation of the proposed software solution, the availability to end users, operational requirements and scalability requirements, and the availability of easily implement database management solutions, the project is can be regarded as technically feasible within the scope of works.

## Operational Feasibility

### Organisational Requirements

Acme Financial Services are currently operating with an archaic file management system that consumes a large portion of resources to operate. Digitising these files and allowing access through a database solution not only reduces resource cost but also improves accessibility for staff and end users through the web portal.

### Operational Effects

The proposed system will introduce several changes that will affect the current operational behaviour of staff and the interaction between clients and business operatives. As a result, current members of the organisation will need to be trained in the effective operation of the system, with emphasis placed upon information handling within the proposed software. A problem area regarding this issue will be with any departments that regularly need access to data to perform day to day operations, such as the administration department and sales. Some training will also be necessary for accounts departments regarding the implementation of the proposed system interfacing with existing accounts software, however, most of this interface will be occurring backend. All the software implemented will be as user friendly as possible to facilitate necessary operation.

The decrease in effective resource allocation will allow these resources to be repartitioned, as substantial man hours will no longer be dedicated to data and file management, and client interfacing. This may result in relocation and upskilling of some staff so that they may be more effectively utilised in other areas. It may also result in the redundancy of some positions within the workplace.

### Employee Response

The technology readiness index questionnaire administered to the employees of ACME Financial Services highlighted several keys areas that indicate staff sentiment regarding the project. The proposed software solution generally seems to be well received in terms of optimistic response from all teams, however there is an overall high rate of discomfort and insecurity regarding the implementation of the solution and the effect that will have on employment and workplace function. This can be easily alleviated with targeted information sessions responding to concerns and evaluating feedback from employees.

### Operational Feasibility Summary

After examination of the effect that the proposed software solution will have on operating procedures and staff, this project can be considered operationally feasible.

## Economic Feasibility

### Costs

There will be several costs involved with the design and implementation of the proposed software and database solution. These consist of design and hardware costs, cost of labour, hardware and software upgrades, and training costs. These costs are itemised below

* Design and Labour - $22,650
* Cloud Database Solution, User Interface, Web Portal and relevant Software - $4,329
* Hardware upgrades, data redundancy technology and equipment - $4,389
* Expected Operational costs - $2,350
* Unexpected Operational budget (Conditional Return) - $3,000
* Training costs - $3,275

These costs total to $39,993 and fall within the bounds of the budget provided by ACME Financial Solutions (including conditional return of up to $3,000).

### Benefits

The benefits of this solution post implementation include; the reduction of operation costs, reduced personnel and staffing costs, increased revenue due to the acquisition and retention of clientele and improved operational efficiency. The benefit values are itemised below

* Improved record keeping efficiency - $216,000 per year (based on expected 80% reduction in operational resources spent on record keeping)
* Client Acquisition and Retention - $550,000 per year (projected yearly sales growth post implementation)

The total financial benefits total at an estimated $766,000 annually (not adjusted for inflation or growth). There are a number of qualitative benefits associated with the implementation of the software that are unable to be quantified with a financial value, however remain relevant and valuable to the operations of the business. These can be associated with updated ease of use for sales staff accessing client information, resulting in improved turnaround and more accurate information accessed and given, maintaining a competitive presence within the market and increasing the standing reputation and public perception of the company as it adjusts to a changing business climate.

### Economic Feasibility Summary

After reviewing the cost benefit analysis and determining that the proposed works fall within the project budget given by ACME Financial Solutions, it can be considered that the project is economically feasible and viable.

## Alternative Solutions

### Microsoft Azure

Microsoft cloud database solution (Microsoft Azure) can be interfaced with the pre-existing SharePoint package to provide a total solution for ACME Financial Services. The benefit of this would be to reduce the production of training material and documentation, reduce software design costs and ensure that data redundancies are implemented securely. The cost of this solution is an increase in monthly operating cost (calculated data storage on Azure cloud platform returns $1,990 monthly) and an increased cost to implementing portability across mobile device range (Interfacing web portal software costs estimated at $8,000)

This alternative also results in an increase to unnecessary user end functionality, and an increased cost for unused functionality.

### Microsoft SharePoint

Microsoft SharePoint provides an out of the box end user interface that can be directly linked with Azure cloud databases to provide a complete web and database solution. This however results in total costs that fall outside of the projected budget and result in a loss of functionality and scalability.

### Alternative Solution Summary

Designing a company specific based solution will better optimise data flow and management by personalising the software to specific necessary operations within the company, and as such, will result in greater value and usability than Microsoft Azure.

# Software Solution Report

## Introduction

The SDLC is a necessary component of the software design process. It is a broad high-level plan that outlines the stages of product development based upon client and stakeholder requirements, and functionality requested by the scope. There are several SDLC methodologies that have been defined and each have benefits and risks associated with each methodology. Therefor it is important to decide on which model to use based upon the nature of the project.

## Software Development Life Cycle

The software development life cycle are predefined processes that lay the framework for the progress and development of a software project. These processes are implemented to ensure that the development of the project runs efficiently and succinctly. SDLC processes provide direction for the team when developing, changing or upgrading software. SDLC methodology emerged in the 1960’s in response to the development of large-scale business systems. These systems often took a great deal of development time and resources to construct and implement and required processes to coordinate team efforts. The typical development life cycle consists of six stages:

### Planning and Requirement Analysis

Planning and requirement analysis is integral to the development life cycle. Requirement information is taken from a variety of sources, which is then analysed and utilised to create a comprehensive plan that considers quality assurance, and risks associated with the project.

### Defining Requirements

Requirements for the project are then defined by laying out expectations of the product in terms of functionality and performance. These requirements will usually require approval from the customer. This is compiled into an SRS – Software Requirement Specification.

### Designing the Product Architecture

After the requirements have been analysed and defined, the architecture for the project is designed by product architects. The architecture is based of the requirements for the project and will provide a modular blueprint for the development of the software. This will include communication and data flow representations and charts. This information is documented within a DDS – Design Document Specification. This document will include definitions of all of the modules used within the specification.

### Build and Development

This stage marks the beginning of the construction of the product. The DDS is used during this stage to develop the code that follows the correct infrastructure and processes outlined within the product architecture documentation. Code guidelines should be followed during this stage to ensure uniformity of code, and readability for other members of the team.

### Testing

The testing stage is used to identify and problems within the product, in the form of bugs or functionality errors. These errors are identified and reported, and returned to the development team to implement fixes, which are retested to ensure that the product works as intended.

### Deployment and Maintenance

Once the development of the product has been completed and tested, it is released for use. Post release use might identify further errors or additional required functionality. These problems are addressed, or, if they fall outside of the scope, maybe incur additional cost to the customer.

## Option One: Waterfall Model

Waterfall methodology consists of sequential steps or phases of development that follow a linear progression. These steps will usually consist of logical sequences for planning, design, construction, testing and implementation, deployment and maintenance if required.

### Waterfall Model Pros

* Ease of use
* Clearly defined stages
* Clearly defined goals
* Organisationally simple
* Processes and results are well documented throughout stages
* Structure of the model ensures simple management of deliverables

### Waterfall Model Cons

* Testable or usable software is not available until late during the life cycle
* High risk and uncertainty factor due the rigid nature of the model
* Simplicity of the model does not mesh well with complex object orientated projects
* Structure of the model makes changing requirements difficult to implement
* Measuring progress of the product development is difficult
* Integration is done at the end of the project and doesn’t allow identification of technology or business bottleneck or other issues earlier within the development life cycle.

## Option two: Agile Model

Agile methods segment the project into incremental builds which are provided in iterations. These iterations usually last several weeks, allowing deliverable versions of the software to be produced to the client for continual sign off until all software requirements are met. These iterations are produced by cross functional teams working simultaneously on various areas of the life cycle of the project.

### Agile Model Pros

* Promotes teamwork and cross training
* Rapid development of functionality and features
* Low resource requirements
* Adaptable to changing or added requirements
* Delivers basic working products incrementally
* Documentation is easily employed
* Little to no planning required before development begins
* Allows the developers to be flexible with the project
* Minimises scope creep
* Development progress is easily measured

### Agile Model Cons

* Difficult to handle complex dependencies
* Risk of sustainability and maintainability of the project
* A high-level plan and appropriate team leader is required to implement agile methods
* Deadlines for deliverables require strict adherence
* Depends on customer involvement with the project. Deliverables require continuous sign off during the development of the project
* Individual responsibility of each team member is high as there is little documentation for them to follow.
* Transfer of technology to additional team members can be difficult as little documentation is produced
* Requires strict adherence to code practices to ensure that code is uniform and readable across all areas of the project

## Summary

SDLC methodology is integral to the development of marketable software. The processes outlined allow technical leads to define the requirements and deliverables of the project, as well as provide the architecture necessary to develop software that effectively meets those requirements. This ensures that the product deployed to the market is of a high quality and falls within the project constraints.

### Recommendation

Agile methodologies are the most appropriate for a project of this size and scope. It is also appropriate for the implementation of changes to the scope, such as additional desired functionality, while minimising gold plating during development due to the short development cycles. It will also allow the product to be deployed in increments, minimising the amount of training needed while also familiarising them with the systems.

# Risk Assessment Report

## Development of Incorrect User Interface

## Introduction

Incorrect development of the product UI has been identified as a critical risk of high probability within the risk assessment matrix. Incompatible or incorrect UI elements will eventually lead to user dissatisfaction and ease of use issues that will arise with the implementation of superfluous functionality. Further issues will also arise when implementing training programs for staff to use the software. Additional unnecessary functionality, while often useful for power users, will not need to be implemented within the software for end users, as it exacerbates confusion, and therefor training time, with little benefit to day to operative requirements when using the software.

## Strategies

There are several strategies when dealing with the risk of developing wrong or unnecessary UI’s. These strategies can be divided into three main groups: Surveying and End User Analytics, Development and Testing and Review and Implementation.

### Surveying and End User Analytics

* Survey end users to define the functionality and level of ease of use that will be required by end users. This should be undertaken as an initial step, before development of the software begins.
* Analyse end users of the software, with emphasis on examining mental models in relationship to the user interface.
* Define end user work scenarios in relation to the software and analyse them in terms of interface goals.
* Define user interface goals and requirements based upon the information that surveys and end user analysis have provided. Ensure that the goals and requirements are properly documented and updated when necessary.

### Development and Testing

* Develop a prototype of the user interface in accordance with interface goals and requirements identified at the survey and analytical stage.
* Test the prototype within a sample user group.
* Collect information and feedback from the sample user group.

### Review and Implementation

* Review the data and feedback collected from the development and testing stage and cross reference it with the goals and requirements defined within the surveying and analytics stages.
* Ensure that any new goals or requirements that have arisen from testing are updated and documented.
* Implement changes as requested or necessary to the prototype.
* Return to the testing phase for any changes or new functionality implemented as appropriate.

## Recommendations

* Survey and test the prototype on staff that have been identified as key operatives of the software, and on staff that have reported high levels of discomfort and insecurity on the technology readiness index questionnaire.
* Create and implement a plan for the UI development process using the strategies highlighted above.
* Ensure that the UI designer, and architecture developer, and back end developers are working collaboratively to ensure the UI has appropriate functionality for all end users.
* Develop secondary UI for high level users with further understanding and a broader scope of required functionality. This UI should be aimed towards administrative users.

## Gold Plating

## Introduction

Gold plating has been rated as a relatively critical risk of decent probability within the risk assessment matrix. Gold plating within projects further compounds issues relating to development, often increasing development times and costs, interfering with proper operation and requirements of the software, and in the improper use of resources during development of the product. These issues make Gold Plating a key area of concern, especially in relation to lower budget/lower turnaround projects where gold plating can cut revenue from the product sales by implementing functionality for free.

## Strategies

* Create a development plan that addresses only the functionality defined within the scope of works.
* For larger projects that require multiple teams, delegate tasks appropriately to each team to ensure that resources are used effectively.
* Employ project management techniques and processes that ensure that software development staff are kept on task.
* Allocate additional resources, or resources that free up onto critical areas of functionality defined within the original scope of work that may be behind schedule.

## Recommendations

* Ensure that the scope of works is clearly communicated to staff and emphasise that work is to be aimed at delivering on the scope, not on extras.
* Implement agile software development practises. Short sprints of work ensure that there isn’t excess time spent in development that is then wasted on gold plating. Additionally, the customer then reviews and accepts the results quickly, ensuring that the product is up to the expected standard of the customer, not the standards of any project team members that may have perfectionist qualities.

## Changes to Requirements

## Introduction

The risk assessment matrix identifies additions or changes to requirements of the project as the most likely and most critical issue to arise during the development process. Changes to requirements that have not been defined within the scope causes several problems during development. It increases the amount of work needed to be completed before any set deadlines. It adds additional features that have not been analysed during costing reviews and analysis during the planning stages of the project, and often it will also interfere with existing functionality, and require additional resources to implement, all at the cost of the developer, unless otherwise agreed upon.

## Strategies

* Define and baseline the requirements for the project prior to commencement.
* Ensure that any additional features or requirements defined by the customer incur additional cost after the scope has been initially defined.
* Ensure that the development team have a solid uniform understanding of what the customers requires with the system.
* Identify existing products on the market with similar functionality that can be used to baseline or establish firm requirements.
* Administer surveys to stakeholders to identity desired user functionality within the system.
* Identify any known change requests that are pending.
* Identify whether the customer has a history of requesting changes beyond the agreed scope.
* Analyse whether the change will impact the performance or the existing desired functionality of the system.
* Identify the resources required to implement the change.

## Recommendations

* Ensure that changes to the scope are identified within the scope of works, and will require additional cost on behalf of the customer to implement changes beyond the originally agreed upon scope of works
* Communicate any requested changes to key stakeholders to ensure that the changes to functionality are necessary.
* Research and analyse existing products already on the market when defining the scope and or changes to the scope.

# Signatures

### Client Sign-Off

………………………………………………………………….

### ATEX Project Manager Sign-Off

………………………………………………………………….

### ATEX Technical Lead Sign-Off

………………………………………………………………….

# References

Ambler, S. W. (2014). *Justifying a Software Development Project*. Retrieved from Ambysoft: http://www.ambysoft.com/essays/projectJustification.html

Australian Computer Society. (2016). Retrieved from https://www.acs.org.au/home.html

Azi. (2012, May 23). *A brief comparison of SDLC Models*. Retrieved from Asm's Blogosphere: http://asma-aziz.blogspot.com/2012/05/brief-comparison-of-sdlc-models.html

Business Analysis Center of Excellence. (2013). *Documenting Non-Functional Requirements*. Retrieved from Business Analysis: http://www.businessanalysis.in/2013/08/documenting-non-functional-requirements.html

Construx. (2019). Retrieved from https://www.construx.com/

Gordon, J. (2019). *How to Write a Feasibility Report*. Retrieved from bizfluent: https://bizfluent.com/how-5554221-write-feasibility-report.html

IT Pro Portal. (2010, July 4). *Comparison of Various Software Development Life Cycle*. Retrieved from ITProPortal: https://www.itproportal.com/2010/07/04/comparison-various-software-development-life-cycle/

Kirinde, R. (2016, April 13). *How to control the scope creep*. Retrieved from The Project Management Corner: https://www.projectmanagement.com/blog-post/19863/How-to-control-the-scope-creep

Lake, M. (2010, September 9). *Epic failures: 11 infamous software bugs*. Retrieved from Computerworld: https://www.computerworld.com/article/2515483/epic-failures--11-infamous-software-bugs.html

Open Group. (2001). *Business Scenarios*. Retrieved from OpenGroup: http://www.opengroup.org/public/arch/p4/bus\_scen/bus\_scen.htm

ReQtest. (2019). *Functional vs Non Functional Requirements*. Retrieved from ReQtest: https://reqtest.com/requirements-blog/functional-vs-non-functional-requirements/

Sai Global. (2019). *ISO/IEC 26514 : 2008*. Retrieved from Sai Global Store: https://infostore.saiglobal.com/en-au/Standards/ISO-IEC-26514-2008-589528\_SAIG\_ISO\_ISO\_1350435/

Salonek, T. (2013, July 15). *Top 15 Worst Computer Software Blunders*. Retrieved from Intertech: https://www.intertech.com/Blog/15-worst-computer-software-blunders/

Sauter, V. L. (2013, September 25). *Data Flow Diagrams Examples*. Retrieved from University of Missouri - St. Louis: http://www.umsl.edu/~sauterv/analysis/dfd/dfd\_intro.html

Standards Australia. (2018). *ISO/IIEC 26514:2008*. Retrieved from Standards Australia: https://www.standards.org.au/standards-catalogue/international/iso-slash-iec--jtc--1-slash-sc--7/iso-slash-iec--26514-colon-2008

Steffens, R. (2018, December 10). *18 of the Very Best Mission Statement Examples*. Retrieved from Blueleadz: https://www.bluleadz.com/blog/15-of-the-very-best-mission-statement-examples

Tutorials Point. (2019). *SDLC - Overview*. Retrieved from tutorialspoint: https://www.tutorialspoint.com/sdlc/sdlc\_overview.htm

Wikibooks. (2018, July 18). *Professional and Technical Writing/Feasibility*. Retrieved from Wikibooks: https://en.wikibooks.org/wiki/Professional\_and\_Technical\_Writing/Feasibility

Wikipedia. (2017, April 12). *Risk analysis (engineering)*. Retrieved from Wikipedia: https://en.wikipedia.org/wiki/Risk\_analysis\_(engineering)