**Reporting Dashboard for Interrogating Transactional Insurance Data within Applied Relay Broker Management System**

*Project Plan*

Paul Connolly

B00348751

Table of Contents

[Introduction 2](#_Toc497083471)

[Background 2](#_Toc497083472)

[Project Aim 3](#_Toc497083473)

[Project Objectives and Activities 3](#_Toc497083474)

[Objectives 3](#_Toc497083475)

[Activities 3](#_Toc497083476)

[Project Schedule 4](#_Toc497083477)

[Project Milestones 4](#_Toc497083478)

[Work Breakdown Structure 4](#_Toc497083479)

[Project Lifecycle 5](#_Toc497083480)

[Software Development Process Models Considered 5](#_Toc497083481)

[Waterfall Model 5](#_Toc497083482)

[Spiral Model 6](#_Toc497083483)

[Agile Model 7](#_Toc497083484)

[Project Scope 9](#_Toc497083485)

[Project Risk Assessment 10](#_Toc497083486)

[Risk Register 10](#_Toc497083487)

[Project Resources 11](#_Toc497083488)

[Required Knowledge and Skills for the Project 11](#_Toc497083489)

[Project Data Management 11](#_Toc497083490)

[Works Cited 12](#_Toc497083491)

[Appendix 13](#_Toc497083492)

# Introduction

Project aim is to create a reporting dashboard for insurance brokers using Applied Systems product Applied Relay enabling them to view their transactional data over a given period.

# Background

Applied Systems have operated for over 30 years powering the insurance industry across the USA, Canada, Ireland and the United Kingdom, providing industry leading technology to the insurance industry (Applied Systems, 2017).

In the Irish insurance market, Applied Systems provides insurance brokers with a product called Applied Relay. Applied Relay is a back-office solution enabling brokers to facilitate all needs of their customers (Applied Systems, 2017).

Key features of Applied Relay are (Applied Systems, 2017):

* Leads management
* Automatic Accounts reconciliation
* Policy Administration
* Product Builder
* Account Reconciliation

The current reporting tool in Applied Relay is limited as it is not able to provide its users with graphical representations of the reports it generates.

Reports are generated within Applied Relay and these can be executed on-demand or set-up to run automatically. The reports can be run from a predefined list or the user can generate their own by using the report building tool inside of Applied Relay.

# Project Aim

Aim of the project is to provide insurance brokers in Ireland using Applied Relay, with a secure, web-based dashboard giving them the ability to view graphical reports of their transactional data from Applied Relay.

The reporting dashboard should give insurance brokers a better and more in depth understanding of how their brokerage is functioning. It will give them more informative feedback that is easily understood and will give them the ability to interrogate the data to meet their own needs.

## Project Objectives and Activities

To give the project structure, a list of project objectives and activities have been identified and listed below:

### Objectives

1. Create website for Irish insurance brokers that will graphically represent the transactional data of their business.
2. Create SQL database for storing data that will be reported against.
3. Create service for exporting data from current Applied Relay database to the new database schema.

### Activities

1. Project requirements gathering.
2. Determine database and database tables in Applied Relay that will be used during the data export process.
3. Define user roles on the website.
4. Determine default reports, graphs and statistics to be included on the dashboard.
5. Research best programming languages to use for this project.
6. Design reporting database.
7. Mock-up user interfaces and have them signed off by Project Manager.
8. Build UI and test.
9. Create Windows service or web API for front-end communication and retrieving data from database.
10. Write SQL queries.
11. Test code base in the Windows service/web API.
12. Link front-end and service so data can be retrieved from the database and displayed on the UI.
13. Manually test all aspects of the system. This should be an ongoing process during system development but an all-around test will be done when the project is considered complete.

# Project Schedule

The project is scheduled to run for around 7 months, which breaks down in to estimated periods of 1 month of planning and design, 5 months of development and another month for testing and deployment.

## Project Milestones

The project milestones have been added to the GitHub applied-relay-reporting, available [here](https://github.com/ulster-university/applied-relay-reporting/milestones) and Appendix 1 and Appendix 1.

The milestones selected outline important stages throughout the project and determine points at which significant amounts of work have been completed.

## Work Breakdown Structure

As part of project planning, a Work Breakdown Structure, Appendix 3, has been developed to give an oversight to which project milestones fall in each stage of the Software Development Life Cycle (SDLC).

# Project Lifecycle

The Software Development Life Cycle is a process that aims to produce software with the highest quality and with the lowest cost possible in the shortest amount of time (Stackify, 2017).

According to (Stackify, 2017), the advantages of following the SDLC are:

* It allows a high level of management control
* Gives developers a good understanding of what they are trying to build
* An agreement is made upfront on what the project outcome should be
* It sets out an agreed plan on how to reach the proposed goal

The Software Development Life Cycle defines six stages that can be implemented in various ways by different SDLC models.

These 6 stages are:

* Planning
* Defining
* Designing
* Building
* Testing
* Deployment

The models, also called Software Development Process Models, will follow steps unique to each model but will still all either strictly or loosely follow the Software Development Life Cycle stages (Tutorials Point, 2017).

## Software Development Process Models Considered

The following models have been considered as a development model to use on this project.

### Waterfall Model

The first process model to be introduced to the software development industry, it was designed to be used in a wat that processes do not overlap; one process must finish before the other starts (Tutorials Point, 2017).



Source: https://xbsoftware.com/blog/software-development-life-cycle-waterfall-model/

#### Advantages of Waterfall Model

* Simple to use and understand
* Each process has specific goals and outcomes
* Stages of the project are well defined

#### Disadvantages of Waterfall Model

* A working product is not delivered until near the end of the life cycle
* Not suitable for projects with requirements that are likely to change
* Stages must wait on their predecessors to finish before they can start

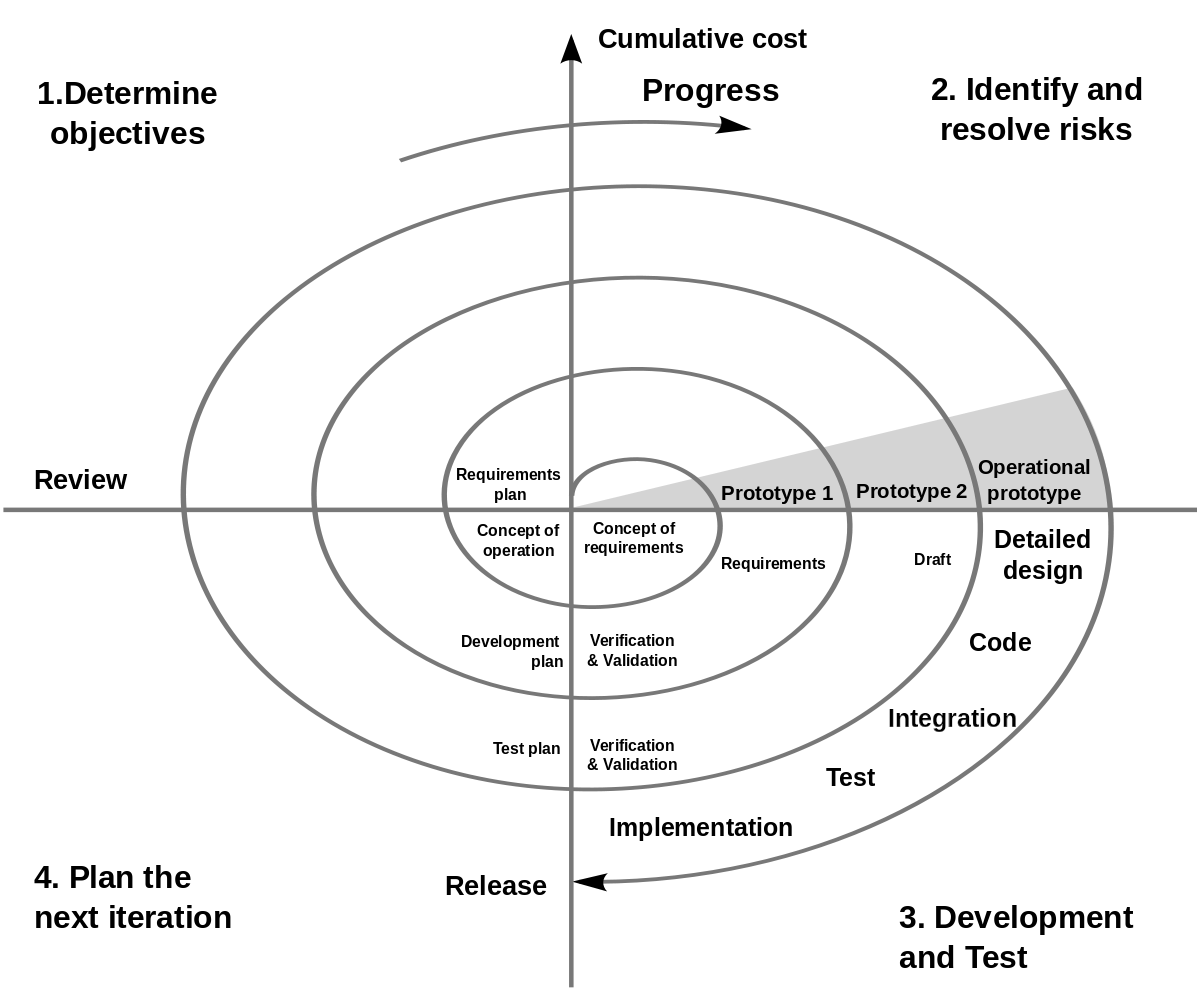
#### Why the Waterfall model was not chosen

Due to its strict phase completion rules, the Waterfall model is not suited for this project that is open to changing requirements throughout the duration of the project.

The plan is also to have regular feedback from stakeholders and users which again does not fit with the Waterfall model process.

### Spiral Model

Consisting of four phases; planning, Risk, Engineering and Evaluation, a software project using this model will pass through each phase iteratively until the project is delivered (International Software Testing Qualifications Board, 2017).



Source: https://en.wikipedia.org/wiki/Spiral\_model

#### Advantages of Spiral Model

* Emphasis on risk analysis means risk are identified early and can be managed or avoided
* Software is produced early and frequently
* Software functionality can change or be added late in to the project

#### Disadvantages of Spiral Model

* Can be an expensive model
* Not suited to small projects
* Risk analysis requires experienced analysts

#### Why the Spiral model was not chosen

The Spiral model was not selected for use on this project due to the short life span of the project. Spiral is more suited to larger projects with a long-term commitment.

The project is also considered low risk and the requirements of the project are clear and concise.

### Agile Model

The Agile SDLC model put focus on delivering products to the customer on a regular basis. It combines iterative and incremental models and focuses on customer satisfaction and welcomes change (Tutorials Point, 2017).

A project using the Agile model will break a project down in to smaller pieces of work and deliver these in an iterative manner.

Each iteration is able to have multiple processes active at any one time. The processes include:

* Planning
* Requirements Analysis
* Design
* Coding
* Unit testing
* Acceptance testing

The iterations that produce working software are usually time boxed to an amount of time decided by the team.

Iterations can happen in a similar way to the image below.



Source:

http://istqbexamcertification.com/what-is-agile-model-advantages-disadvantages-and-when-to-use-it/

The Agile model come with an Agile Manifesto (Agile Manifesto, 2017) that states the following principles:

* Individuals and interaction over processes and tools
* Working software over comprehensive documentation
* Customer collaboration over contract negotiations
* Responding to change over following a plan

#### Advantages of Agile

* A realistic take on how software is best developed
* Functionality is developed quickly
* Processes work with pre-defined or changing requirements
* Little or no planning required

#### Disadvantages of Agile

* Depends heavily on stakeholder and customer communication and feedback
* Less focus on documentation can lead to problems when onboarding new team members or handing a project on to another team
* Changing requirements and functionality can have an adverse effect on the project delivery deadline and can lead to scope creep on the project

#### Why Agile was chosen for this project

Agile has been chosen as the software development model for the project as the Agile model is one widely practised within Applied Systems.

The opportunity to develop software quickly and get regular feedback on it means the end product is more likely to meet the Project Sponsor’s expectations.

Not having to focus on detailed documentation also means the emphasis can be placed on building a working system. Due to the tight schedule of this project, this is a benefit that cannot be overlooked.

# Project Scope

The delivered system should consist of a secure, web-based website that communicates with a server side service and SQL database.

A registered user of the system will be able to login to the site and view data associated to the role they have been assigned. User roles will restrict the reports a user can execute and view.

Users will also be restricted to only viewing data of the company/insurance brokerage they are employed by.

The website will be an interactive dashboard with graphs and statistics showing an insurance brokerages performance over a specified period. As a default this data will be from the previous day’s business but the time period can be changed to show reports for a custom date period.

Users will be able to configure the settings for their dashboard so they can choose which reports and statistics they see on their dashboard by default.

For this project, the data is currently stored in an old database that over time has become disjointed and difficult to work with. To get the data in a more suitable structure for reporting a database schema will be designed as one of the initial tasks on this project. This will be a relational SQL database designed with emphasis on being in a performant structure suitable for reporting.

The project will require a Windows service to be created that will have open endpoints the websites can make requests to for retrieving the data that will be drawn up in graphs and output as statistics on the user interface.

# Project Risk Assessment

The purpose of a good risk assessment before a project begins is to help expose potential risks in a project at an early stage. It is important to identify the likelihood of the risk occurring and to hopefully find an early solution to remove or limit the risk. It is a key tool in project planning.

## Risk Register

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Risk Id** | **Description** | **Probability of Occurrence** | **Preventive Measures** | **Severity** |
| 1 | Unrealistic time schedule | Medium | Incremental development, modify milestones, requirement prioritisation | High |
| 2 | Data loss | Low | Backup database regularly, secure repository for source code | High |
| 3 | User interface does not meet requirements | Low | Regular customer interaction, careful design, attention to requirements | Medium |
| 4 | Poor product performance | Medium | Ensure data is structured correctly, performant UI components and database | Medium |
| 5 | Scope creep | Medium | Set functionality boundaries, closely manage changing requirements | Medium |
| 6 | Lack of skills | Medium | Build software with skills I have, iterative builds to get more complex if time allows | Medium |

# Project Resources

To ensure delivery of this project, the following required resources have been identified:

* Development computer (PC and/or laptop)
* Microsoft Visual Studio
* Microsoft Visual Studio Code
* SQL server
* Hosting server

## Required Knowledge and Skills for the Project

Programming skills for this project will include development languages:

* C#
  + NCrunch and SpecFlow will be used when testing this code
* SQL

Web development languages include:

* HTML
* CSS
* JavaScript
  + React – a JavaScript library
  + Wallaby – a JavaScript testing library

Development framework

* ASP.NET

## Project Data Management

All data stored in the database is held on a secure server inside of Applied Systems Ireland.

Source code for the project will be held in a private GitHub repository so that it is controlled and versioned. The GitHub repository will be called ‘applied-relay-reporting’.

The finished system will use ASP.NET to authenticate users. Users will be required to have a unique username and unique password in order for them to be able to login to the website.

# Works Cited

Agile Manifesto. (2017, 10 29). *Manifesto for Agile Software Development*. Retrieved from Agile Manifesto website: www.agilemanifesto.org

Applied Systems. (2017, 10 28). *About Applied*. Retrieved from Applied Systems website: https://www1.appliedsystems.com/en-ie/about-us/

Applied Systems. (2017, 10 28). *Applied Relay Software for Brokers*. Retrieved from Applied Systems website: https://www1.appliedsystems.com/en-ie/solutions/for-brokers/managing-your-brokerage/applied-relay/

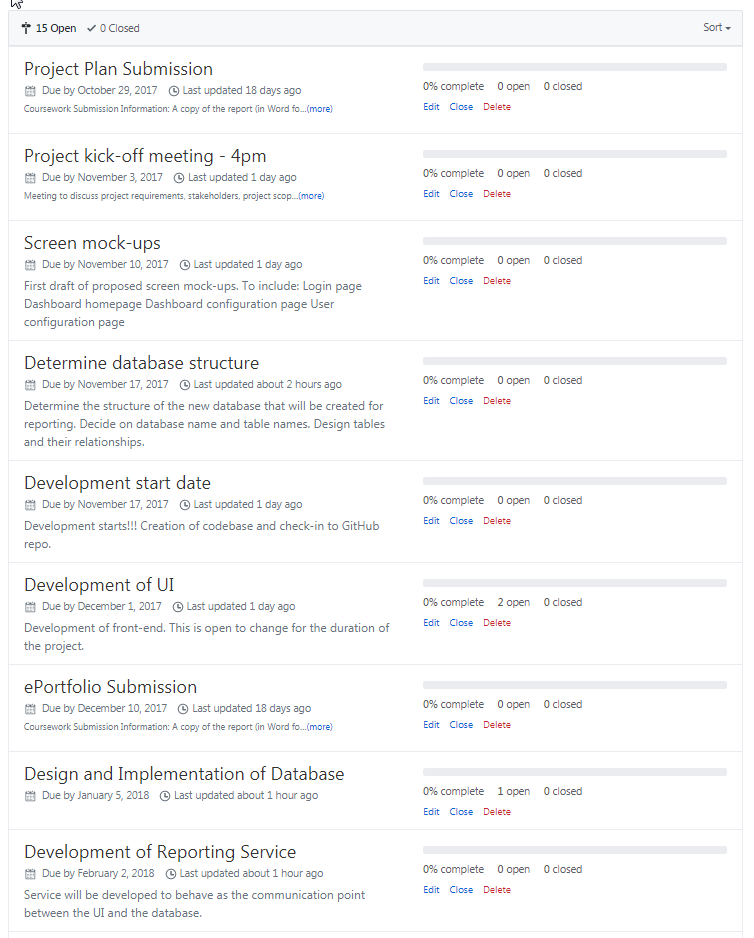
International Software Testing Qualifications Board. (2017, 10 29). *What is Spiral model- advantages, disadvantages and when to use it?* Retrieved from ISTQB website: http://istqbexamcertification.com/what-is-spiral-model-advantages-disadvantages-and-when-to-use-it/

Stackify. (2017, 10 04). *What is SDLC? Understand the Software Development Life Cycle*. Retrieved from Stackify website: https://stackify.com/what-is-sdlc/

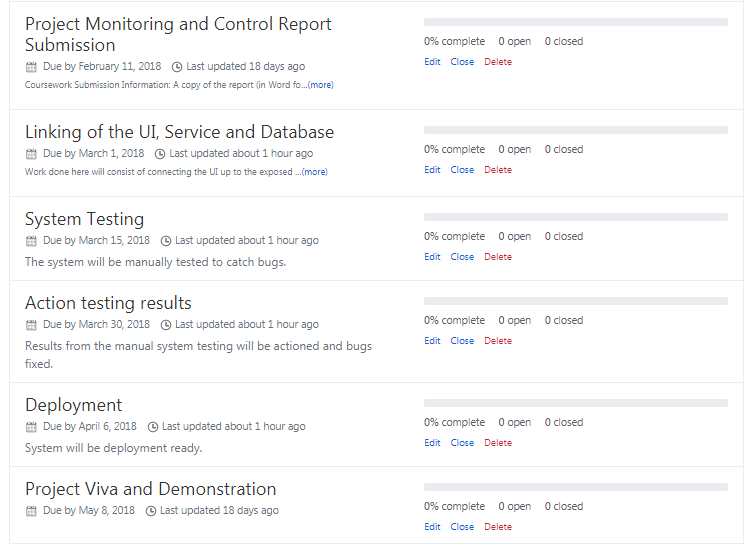
Tutorials Point. (2017, 10 29). *SDLC Agile Model*. Retrieved from Tutorials Point website: https://www.tutorialspoint.com/sdlc/sdlc\_agile\_model.htm

Tutorials Point. (2017, 10 29). *SDLC Overview*. Retrieved from TutorialsPoint: https://www.tutorialspoint.com/sdlc/sdlc\_overview.htm

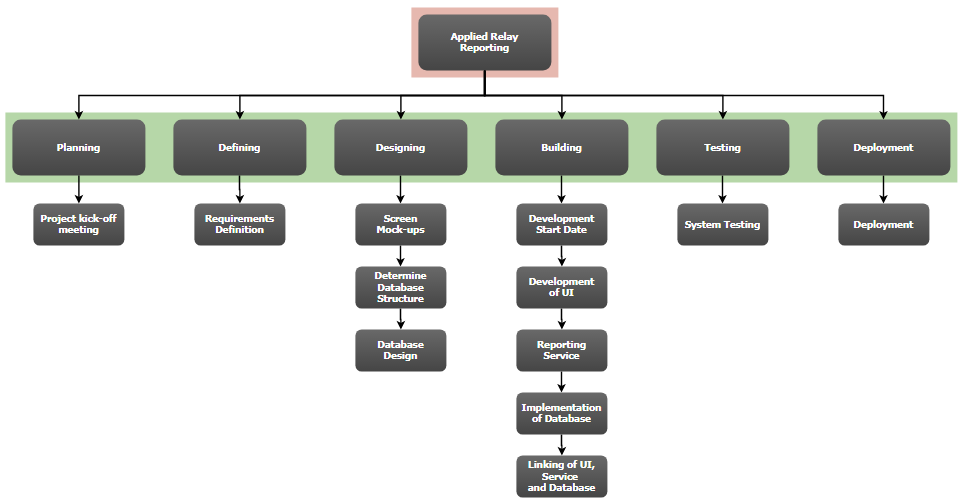
# Appendix



Appendix



Appendix



Appendix