Oliver Collins-Cope

2102775@rutc.ac.uk

Learning Aim A

Investigate the principles and methodologies of IT project management as used in the industry.

unit 09 it project management

Assignment 1

Contents

[Introduction 3](#_Toc116653126)

[What is a project? 3](#_Toc116653127)

[Difference between a project and a process 4](#_Toc116653128)

[What do IT projects include? 4](#_Toc116653129)

[What is project management? 5](#_Toc116653130)

[Planning 5](#_Toc116653131)

[Organisation 6](#_Toc116653132)

[Controlling 6](#_Toc116653133)

[Project life cycle 6](#_Toc116653134)

[Initiating 6](#_Toc116653135)

[Planning 6](#_Toc116653136)

[Executing 6](#_Toc116653137)

[Monitoring and controlling 7](#_Toc116653138)

[Closing 7](#_Toc116653139)

[Different project management methodologies 7](#_Toc116653140)

[Projects IN Controlled Environments (PRINCE2) 7](#_Toc116653141)

[RAD 9](#_Toc116653142)

[Waterfall 10](#_Toc116653143)

[Agile 10](#_Toc116653144)

[Project Management structures 12](#_Toc116653145)

[User requirements 12](#_Toc116653146)

[Project job roles and responsibilities 12](#_Toc116653147)

[Quality Assurance 13](#_Toc116653148)

[Operational test environment 13](#_Toc116653149)

[Live deployment 13](#_Toc116653150)

[Website Development Case Study – Waterfall methodology 14](#_Toc116653151)

[Stages of Waterfall in Website Development 14](#_Toc116653152)

[Requirements 14](#_Toc116653153)

[Analysis 14](#_Toc116653154)

[Design 14](#_Toc116653155)

[Implementation 19](#_Toc116653156)

[Testing 22](#_Toc116653157)

[Deployment 25](#_Toc116653158)

[Conclusion 25](#_Toc116653159)

[Comparison of different methodologies and structures 25](#_Toc116653160)

[Evaluation of different methodologies and structures 25](#_Toc116653161)

[Bibliography 26](#_Toc116653162)

# Introduction

IT project management is involved in almost every part of IT, as when working through different projects, whether it be developing software, making a game, or even producing an operating system, it is crucial to plan and manage the project constantly in order to ensure constant communication between departments, with everyone knowing what they have to specifically work on. Furthermore, IT project management ensures that the processes of the project are efficient and effective.

# What is a project?

A project can be defined as a sequence of tasks that have to be completed to finish a larger task, usually within a define beginning and end period (deadline). Projects often include a budget to meet these goals and are undertaken by individuals or groups with the tasks broken down to complete sequentially in order to achieve the next goal.

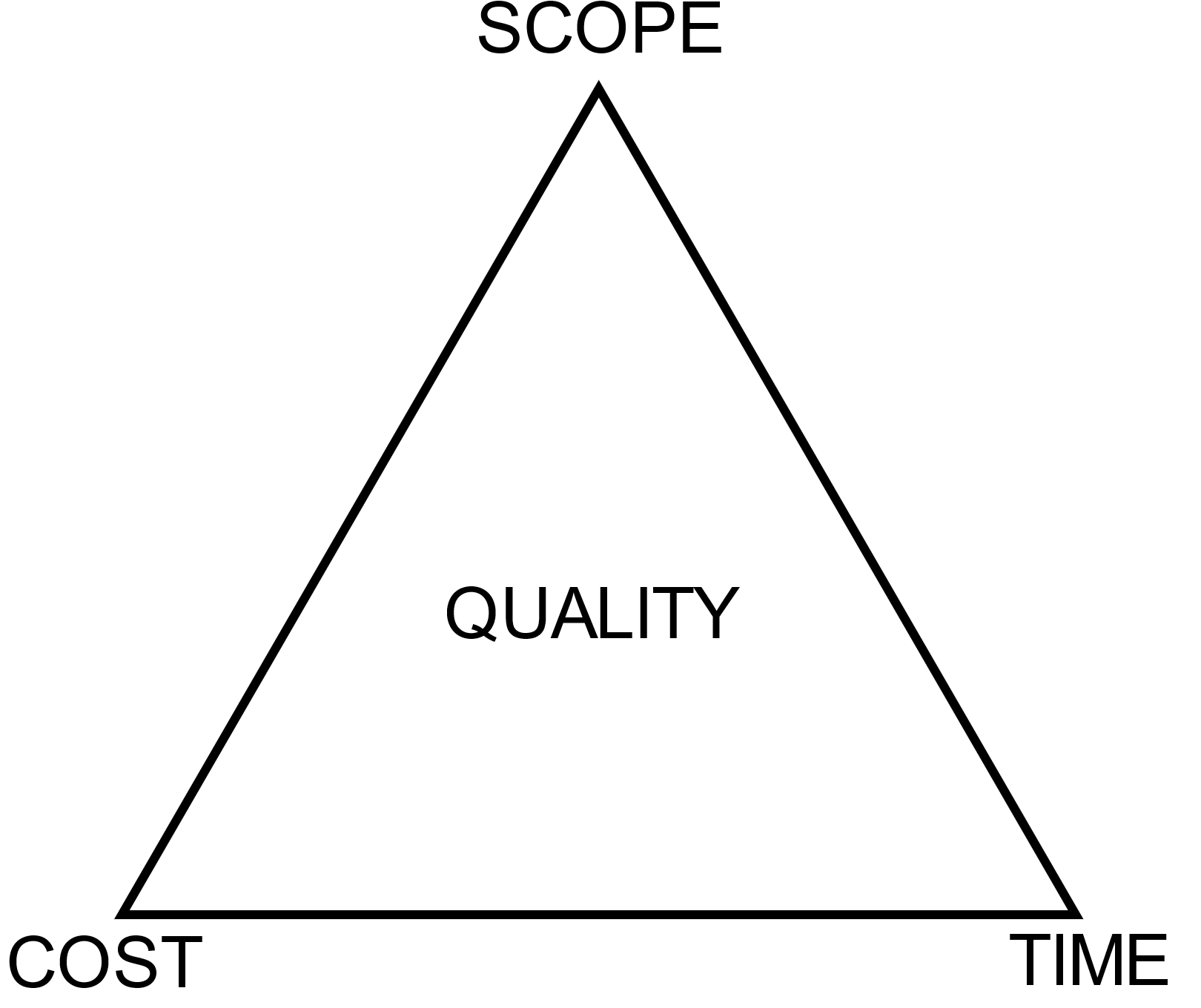
Within projects it is crucial to break down the goal into manageable and doable tasks which can be completed by the members, otherwise the chances of becoming overwhelmed with the work and designing and developing in the incorrect order, which can cause catastrophic consequences later down the line due to your code and design being dependent on each other.

A project can also be displayed through three primary points of focus:

* Time
* Budget
* Scope

Time refers to the time spent on the project, including beginning dates and deadlines to be completed by.  
Budget refers to how many resources have been allocated to the project, whether it be manpower or funding.  
Scope of the project refers to documenting the project boundaries and what the main objectives are.

This diagram can help to understand the relationship between the points of focus.



## Difference between a project and a process

It can often be confusing to understand the difference between projects and processes and their relationship with each other, including how they communicate, so it is important to clarify how these vary and what the key differences are.

While a project is usually long term, including a start and end date, processes are established procedures that help with ongoing work. These can be changed constantly and are not strict in what they contain. To help further understand the difference between a project and processes it can be helpful to view them through their relationship.

C:\Users\2102775\Downloads\pjct x prcss.drawio.png

Through this we can understand that a project has many processes, however a process belongs to the instance of that project. This means that a project is a parent process (usually) and the process is born out of the project. While there are some scenarios where a project can also be a process or be born out of a project (making it a process), this is rare and only seen in much larger scale projects made up of multiple smaller ones.

## What do IT projects include?

IT projects in particular include more specific definitions for what is typically involved in the overall project. In no particular order, this commonly includes:

* Software application development
* Installation of IT systems and networks
* Information collection
* Analysis
* Maintenance

### Software application development

Software application development consists of physically making the software that the company or individual has been contracted to make (in the example of business). This can also be extended to include processes like design, testing, and refinement. Generally, software application development will happen once the requirements and design of the application has been reviewed and approved by the client and project manager, ensuring that everyone is happy with how the software will be developed.

### Installation of IT systems and networks

Installation of IT systems and networks can refer to both physically and digitally installing IT systems for a company or individual.

Physically installing IT systems and networks is comprised of being on site in the location and manually installing the IT system, whether it be a desktop computer or a server rack for the company, both vital components for businesses to function. This can also include mundane jobs like establishing cable links between the servers and desktops so they are able to access each other.

Digitally installing IT systems consists of installing software for the users to use so they are able work with the required applications for their business or company projects.

### Information collection

Information collection within IT projects refers to activities where members of the project are able to share data with each other to help advance the project.

An example of a common information collection activity would a meeting, where people establish what they are doing and what support or blockers might be affecting them in their work. Furthermore, meetings will also happen between the client and company in order to remain updated about their project and allow for new input from the client incase situations arise.

### Analysis

Analysis within an IT project refers to analysing the requirements of the job and what will need to be accomplished in order to achieve success. This can vary from analysing what the program that will be developed is supposed to do and seeing what steps have to be taken to achieve this goal.

### Maintenance

Maintenance in an IT project means to ensure that the delivered program will continue to function smoothly, providing software updates to the technology and fixing any bugs that occur to destabilise the program.

## What is project management?

Project management can be defined as the process of planning, organizing, and controlling a project and its activities in order to ensure that the project is completed successfully within the previously referenced three primary points of focus within the project. Alongside this, it is also important for project management to mitigate any appropriate risks and resolve any problems or issues that may arise during the project’s lifetime. These three processes mentioned can be further broken down to gain a better understanding of them.

### Planning

Planning within a project can vary vastly in what must be accounted for and achieved within this time period of the project. For example, this can range from planning the tasks that have to be achieved in the project in order for successful completion of the project, to planning how to split up personnel and departments and assigning work. Planning ties in closely to organisation and as such the lines between what is planning and what is organisation can be regularly blurred, so it is necessary to understand that these outlines, the outlines being planning, organisation, and controlling, are intentionally vague and only serve as to help establish the project, and thus can be easily manipulated and merged depending on the situation.

### Organisation

As previously mentioned, organisation and planning can often merge together depending on the situation and the individual’s interpretation on what organisation and planning include. Typically, in organisation within project management is responsible for identifying the roles and responsibilities of each position that facilitates coordination and implementation of different project activities. This means that it is necessary to identify and define all the roles of the different team members in order to effectively organise and execute the project.

### Controlling

Controlling a project refers to how the project is managed while it is ongoing. This includes responsibilities like tracking and managing the core project elements with regards to the three primary points of focus, those being time, cost, and scope, in order to effectively run the project and achieve the product decided on before the deadline and within the budget. Further duties of controlling the project can be adjusting the course of activities and swiftly making changes in order to resolve blockers which might inhibit the progression of the project due to unaccounted, or accounted for, variables which impact it.

## Project life cycle

A project generally follows a life cycle of five main stages, and although that most projects will follow this structure, the order in which these stages and steps are implemented depends on the project methodology that is being applied to the project.

### Initiating

Initiating a project is the first stage in the project life cycle, and this step includes outlining the justification for the project alongside the requirements to achieve a successful outcome.

### Planning

Planning a project is the second stage in the project life cycle and includes the information regarding how the project will work, with regards to how it is going to be carried out, and then monitored and controlled to achieve the end of the project successfully.

### Executing

Executing a project is the third stage in the project life cycle and handles the human resources, i.e. dividing the teams up to develop separate parts of the program, and the other available resources such as programs to help aid development. This is all to help undertake all of the tasks of the project within the constraints time, cost and quality.

### Monitoring and controlling

Monitoring and controlling is the fourth and penultimate stage in the project life cycle and it includes, as the name says, monitoring and controlling the project in line with the initiation and planning documentation in order to achieve the end of the project.

### Closing

Closing the project is the fifth and final stage of the project life cycle and is the formal process of closing and ending the project. During this time, it can be common to review the original goals again and compare it to the final product, learning from the success and mistakes of the project.

# Different project management methodologies

There are a variety of methodologies that a project can utilize in order to effectively complete and execute a project. It is important to understand the different methods of project management available as it allows the project manager to choose the best method available for and effectively execute the project.

## Projects IN Controlled Environments (PRINCE2)

The PRINCE2 methodology is a methodology that primarily focuses on a process-based approach with heavily emphasizes organisation and control from start to finish over the whole project. As the name suggests, PRINCE2 is the sequel to PRINCE and was initially developed over four decades ago. It is primarily used by the British government and the UN currently. It ensures six goals that should be defined, as well as seven principles that every PRINCE2 project should ensure. The six goals include:

* Scope
* Timescale
* Risk
* Quality
* Benefits
* Cost

And the principles of PRINCE2 include:

* Continued business justification where each project should have a clear need, a defined customer, realistic benefits, and a cost assessment.
* Learn from experience where each team member should gain experience in order to help improve future performance.
* Roles and responsibilities should be clearly defined between each member so everyone knows who is responsible for what.
* Manage by stages where the work is broken up into individual phases to be worked through with periodic reviews.
* Manage by exception where in the instance of issues arriving that impact established requirements, rather than the project manager resolving it the issue is escalated and a board will resolve it.
* Focus on quality where the deliverable projects are checked for quality to ensure the product is of high quality.
* Tailored approach where each project has its own needs and adjustments that must be made in order to achieve a successful project.

Finally, there are the seven stages of the PRINCE2 project:

* Start up where the project board is made and the initial project brief is proposed and initial staff members are appointed like a project manager and team members.
* Directing a project where authorization of project, authorization of initiation and other key aspects of the project are covered as well.
* Project initiation where an initiation document can be made by the project manager and discussion about risk management, planning, refinement, and other important parts of starting a project.
* Controlling a stage where the project manager breaks down the work into smaller work packages for team members to complete and controlling risks/escalating issue takes place during this stage.
* Managing product delivery where the project manager checks the progress against the project brief and makes sure the quality is ensured in the project.
* Managing stage boundaries where the project manager updates the business case and the project board is decided whether to continue to the next stage or be abandoned completely.
* Closing a project where the project is decommissioned and is reviewed through the project documentation.

Some of the benefits of PRINCE2 include:

* It includes scalability
* PRINCE2 is flexible due to its nature as general use
* Built to deal with the unknown
* Widely recognised and well known
* Has established fundamentals and teaches good practice
* It has good and clear consistency

Some of the limitations include:

* Cost of training and qualification
* Requires senior management to approve of project
* Requires experience
* Documentation heavy
* Too rigid and can be unable to adapt

## RAD

The RAD methodology is a precursor to the Agile development methodology and is generally suited for short-term, large-scale projects, similar to Agile, and is known for having a short turnaround time. This short turnaround time is what makes RAD such an attractive project management methodology for specific fields like IT. It also ensures to remain open-ended meaning that the client is kept involved in the project and it remains easy to make changes in the project.

The different stages of RAD include:

* Requirements planning where the project members, including the client, communicate to determine the goals and expectations of the project along with any current and potential issues that might need to be addressed.
* User design is the stage where the developers work hand in hand with the clients in order to ensure that their needs are met at each stage of the design process and the prototypes that are developed function smoothly and meet the clients needs. In this phase, the bugs and kinks are also resolved through an iterative process much like Agile. This phase also includes various stages internally, such as verification and validation of code, configuration of the program, metrics and measurements, and general project management such as scheduling.
* Rapid construction involves taking the prototypes and beta design systems and from there converts them into a working model. Due to the previous iterative phase most of the bugs and kinks have been resolved and therefore development can be focused on the final working product. This phase of development can be broken down to smaller steps, those being preparation for rapid construction, program and application development, coding, unit integration, and system testing. This stage also includes communication with the client which allows for the client to add input and possible help solve any problems, ensuring they are satisfied with the final project.
* Cutover is the stage where the finished project goes to launch and includes systems like data conversion and final testing. Final changes are made now while bugs can be discovered and resolved.

Some benefits of the RAD methodology include:

* Task oriented structure
* Client is involved
* Easy to make changed
* Regular communication between team and client
* Risk control

Some of the limitations of RAD include:

* Less control on the overall project
* Can be rushed design and app
* Lack of scalability
* Unable to plan for the future

## Waterfall

The Waterfall methodology is based on planning and accounting for every variable in each of stage of the project. This means that it can take months before any development takes place in a project as there will be a large focus on the planning and requirements aspect before it can advance onto the next stages. Waterfall is often suitable for plans that are very clearly outlined from the beginning with a concrete timeline and well-defined deliverables.

The different stages of Waterfall are:

* Requirements where the projects product requirements and end goals are documented for the future development.
* Analysis where the project management develops the system for the project that will be followed.
* Design where the different elements of the project that might need design planning, like program architecture and UI elements.
* Implementation of the three previous steps and ongoing development of the project and software, etc.
* Testing where the project focuses on searching for bugs within the code and fixes any uncovered issues.
* Operation/Deployment of the final project and the delivery of the software/program.
* Maintenance where the final product is updated and any issues that arise are dealt with.

Some of the benefits of Waterfall methodology include:

* Easily copied in the future for successful projects
* Simple to understand
* Timeline of events, all planned out
* Outputs of project visible at each stage

Some of the downsides and limitations of the Waterfall methodology includes:

* Difficult to make any changes to plan
* Expensive to make changes in plan
* Inflexible
* Not suited for constantly changing programs/projects
* Difficult to know everything required during the requirements and planning phases

## Agile

The Agile methodology is based on working through ‘sprints’, which usually last 1-2 weeks and consist of daily meetings where team members mention what stage in their work that they are in and what is blocking them from continuing their work. In this, work is prioritized based on what is important and critical to what can be dealt with later, which leads to a backlog of work and therefore a constant workflow. It is more suited for longer term projects where changes can be made regularly. Feedback is critical within Agile as it allows for the product to move forward in a positive direction that will ensure that the customer is satisfied with the final result.

The different stages of Agile include:

* Concept or iteration phase 1 where the scope of the project is decided and different projects and processes are prioritised.
* Inception or iteration phase 0 where the projects team is slowly built and the initial planning and modelling of the project takes place. The prototypes and MVP can be decided during this phase as well.
* Iteration or construction where the project team work to turn all of the design and feedback from the customer through ‘sprints’ of work. This is one of the most important phases of Agile development as it allows developers to create software swiftly and make improvements that will be able to satisfy the customer. Each sprint is reviewed by a scrum master who manages the sprints and is then adapted for the average workload possible during the sprint.
* Transition and release phase where the product is released on a small scale alongside software testing to smooth out any kinks in the software and ensure a wild scale release is as bug free as possible.
* Maintenance or production phase where the product is released to the customer or wider market while maintenance and bug fixes are available and support from the developers is available to understand how the program works.
* Retirement phase where the program is either removed from development due to being replaced or has become obsolete, either way the project and product is no longer being worked on.

Some of the benefits of Agile methodology include:

* Flexibility
* Clear outputs from each iteration
* Appropriate use of resources
* Satisfied customers do to constant feedback
* Adaptability
* Predictability due to working in short bursts
* Less risks as it is cheaper to catch mistakes and errors earlier on
* Better communication i.e daily meetings

Some of the limitations of the Agile methodology include:

* Requires fast communication and effective teamwork, work often depends on others progressing at a similar rate
* Limited documentation due to fast work pace
* No finite end
* Difficulty to tell when project will be finished

# Project Management structures

There are several structures within project management that should all be followed in order to achieve a successful outcome and a high level of quality with the final release. These structures are:

* User requirements
* Project job roles and responsibilities
* Quality assurance
* Operation test environment
* Live deployment

## User requirements

Within user requirements there are two section to focus, those being both functional and non-functional requirements.

Functional requirements are things like defined inputs, outputs, and processes. This could include things like data entry, controller inputs, and even a keyboard input. An output could be anything from a value returned in or from a table to a character moving in a game. Processes are the things that decide how these inputs turn into outputs, such as converting controller input into movement by adding force to a character or providing calculations to output data according to what was inputted.

Non-functional requirements are requirements to satisfy the system such as maintained quality and high performance. These are much more abstract compared to clearly defined functional requirements however they are just as important when developing a high level application.

## Project job roles and responsibilities

Within projects it is important to also assign job roles and the responsibilities that comes alongside those job roles in order to ensure that everyone knows who is responsible for their own part of the project, so that the overall project runs smoothly and there is no confusion between different team members.

Some of the different job roles include:

* Project Manager, who is responsible for running the project, ensuring effective communication between the teams and other important tasks like risk management, scheduling and delivery.
* Systems architect, who is responsible for the design and implementation of hardware and the software infrastructure, and explaining how the hardware meets the requirements of the project.
* Electronics engineer, who is responsible for designing the electronic components, along with different circuits and system to ensure that software implementation is possible.
* Product owner, who is generally the key stakeholder and is responsible for identifying requirements of the product, and has a vision for the products future.
* Lead developer, who has a responsibility for the underlying architecture and communication between management and the standard developers. Typically mentor the junior new developers as well.
* Developers, who are responsible for writing software code and testing.
* Quality assurance test lead who are responsible for designing the test plans and test suites, and assigning work to the QA testers.
* QA tester, who are responsible for testing the functional and non-functional requirements against the product and general individual testing.
* IT support technician, who are responsible for resolving user issues/faults and ensuring that the IT services are operating to allow the organisation to work and meet the users’ needs to work.
* Users, who use the product during deployment and produce requirements based on usability and testing while sending feedback to upper management and stakeholders.

## Quality Assurance

As previously mentioned, quality assurance is a key process and part of the project structure. The role of quality assurance is primarily to ensure that the product meets the quality standards that has been previously defined by the product owner and customer.

If quality assurance is not there, it can lead to customers being unhappy with the final product and would overall be bad for business, and is therefore critical for the company charged with the project.

## Operational test environment

An operational test environment is a pre-release environment with the architecture being close to live deployment. This part of the project allows for the project team members to measure the field performance of the software alongside measuring the user system interaction effects. It also facilitates the consideration of different environmental factors that can influence system behavior, such as different hardware of user’s performance, and evaluate the effectiveness of the system.

## Live deployment

Finally, there is live deployment. This is where the product is deployed and an operational deployment environment for the final product of the project is created and the product is released to it.

# Website Development Case Study – Waterfall methodology

As mentioned above, the Waterfall project management methodology is suited for website development due to the nature of websites and the fact that it is easy to create a roadmap of features that can be developed in line and it is easy to define the requirements for this project.

## Stages of Waterfall in Website Development

In the Waterfall methodology it follows 6/7 set steps that must be completed in full in order to progress to the next stage of development, similarly to how a waterfall fills an empty vessel before flowing onto the stage.

The website in question is a website designed to raise awareness for a charity that rescues animals, primarily, in the Netherlands however they also rescue animals in Spain and Romania.

### Requirements

It is important to lay out the requirements of the website, which are already defined in the assignment criteria, and from there it is possible to expand on it to include other possible features that might improve the website. The requirements mentioned include:

* Be at least five pages
* Feature an image carousel/slide show and social media integration
* Feature suitable images, text and at least 1 video
* Use scripts in at least 1 way i.e., a booking/quote/email form

Listing out the requirements like this allows for the project members to understand what their goals are and what they have to achieve by the end of the project, allowing them to focus on the important tasks before focusing on anything else.

From this, the analysis can begin along with development for the systems, for the project.

### Analysis

The analysis from here consists of the theoretical design of the system for the project without the software and hardware technologies in the way. For a relatively simple website like this, the analysis is limited to discussion on how the image carousel and script for the email form might work.

In this project, the image carousel will be written in JavaScript, as most work in scripting is done for HTML, along with the email form and validation for it. The video will be in MP4 format and set to auto-play with the audio muted as this is required for videos to auto-play in Chrome.

The pages will be planned before hand and will remain with a similar colour scheme to the original site. This will allow for the site to remain a neutral colour and allow developers to focus on working on the production of the website and different features.

### Design

The design work of this project can now begin with storyboarding the website and mapping out all of the requirements, from the website font to the colour scheme and images/videos used. Due to the simplicity of this website, it will be easy to make minor changes to suite the website, however large changes will not happen.

Here is the design plan:

Diagram

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Graphical user interface

Description automatically generated

Graphical user interface, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Graphical user interface

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Table

Description automatically generated with medium confidence

### Implementation

With the design mapped out of the website, the waterfall can now flow into implementation and writing the code for the website. The outcome of this implementation and development can be seen in the website as its development progresses.

Below is the JavaScript that was developed for the website.  
Text

Description automatically generated

Text

Description automatically generatedAnd finally, there is the website, once the HTML has been made.

Graphical user interface, website

Description automatically generated

Graphical user interface

Description automatically generated

Graphical user interface, text, application

Description automatically generated

A cat sitting on a chair

Description automatically generated with low confidence

Graphical user interface, text

Description automatically generated

Graphical user interface

Description automatically generated

### Testing

With the website designed and developed, it is important to now begin testing and ensure that all of the parts of the website are functional.

Below is the documentation of testing if the JavaScript works.

Graphical user interface

Description automatically generated with low confidence

Graphical user interface, application

Description automatically generated

Here we can see that the email validation validates that the email has been submitted.

Graphical user interface, application

Description automatically generated

And here is an example if one of the required fields is missing.

And finally for testing there is the image carousel.

A picture containing text, cat, screenshot

Description automatically generated

A cat sitting on a cat tree

Description automatically generated with low confidence

Graphical user interface, text

Description automatically generated

Here we can see that the image changes and the image carousel works, meaning that the script works and therefore the requirements have been met, with all the different parts of the website functional.

The testing used in this is iterative testing, where as the application is developed the code is tested alongside it in order to fix any bugs or issues that might develop and prevent it from working. This is commonly used in software development and is a core feature of development for all methodologies as testing only once development has been completed is impractical and expensive.

### Deployment

Now that the testing stage has completed and all the parts of the website are functioning, it is now possible to deploy the project and close it.

## Conclusion

In conclusion, although the Waterfall methodology is effective for website development, its restrictive nature around changes in unfriendly when it comes to the everchanging nature of software development and therefore it I believe that there are more effective software project methodologies out there that could be more adequately suited for website development. All in all, the project was a success and Waterfall is a viable project management methodology.

# Mobile Application Development Case Study – PRINCE2 Methodology

Here, the PRINCE2 methodology will be looked at due to its nature of focusing on six goals and the numerous stages that allow a successful program to be developed and its ability to work for a variety of different project types, even those outside of the IT industry.

## Stages of PRINCE2 in Mobile Application Development

PRINCE2 is noted for having 6/7 stages of development and remaining thorough in its development, ensuring that the work is not wasted and that a purpose for what is done remains in order to ensure integrity within the project.

### Start up

The startup phase in PRINCE2 is important as it allows for the project to begin gaining traction within an organisation, from creating the project board to appointing project managers and the team. For this project, although there is no larger organisation to present the project to, it is still important to follow the different stages as it allows the project to be developed properly. In this instance as there is no board, the equivalent will be external feedback from peers and the assignment authorisor.

There is no previous experience to learn from in this situation so it is important that there is a lot of documentation for this project. While the project management will be documented here, the project itself can be viewed further upon request in another document to ensure there is no confusion between the documentation.

### Directing the project

This stage of PRINCE2 is important as it is where authorization of the project comes from as well as further documentation of the different stages of development. Here, although there is no board to gain authorization from, the feedback from my peers and assignment authorisor allows me to progress forward from there previous stages and advance to the initialization of my project.

It is again important to mention that in PRINCE2 there is consistent constant documentation of the process in order to look back on it later and understand what happens at all the various stages of development, from the start up all the way to the conclusion and closing of the project. Therefore, it may be required to assign someone whose sole responsibility is to documentation of the project.

### Initialisation

The initialization of the project is not so different from the direction phase, however a key difference here is that this phase is more focused on progressing the project rather than authorization and ensuring that the project is actually able to run.

It is necessary here to plan the project and refine the business cases, along with any risk planning and project documentation.

For the plan, this project will produce a mobile application which runs a set of artificial traffic lights according to a set timer that is predefined. The plan can be viewed below. The risk planning for this is minimal as it is a one-man, short term small-scale project that will bear minimal consequences, and certainly none that are financial or business wise. The project documentation will continue in the other paper that can be viewed.

Shape, square

Description automatically generated

The pseudocode for my application will be this, as I believe this is the best way to do this.

IF BUTTON START PRESSED Then

Start coroutine sequence1

Or I will establish a timer and when that timer reaches past a certain value, it will run a specific function which sets the light to the right colours.

IF BUTTON START PRESSED Then

timerLights += Time.deltatime

IF timerLights > int step1Time Then

Sequence1();

IF timerLights > int step2Time Then

Sequence2();

### Controlling the stage

Controlling the stage for this project involves breaking down the work into more manageable chunks and slowly working through it. Any issues that arise during this phase of development that are unrelated to programming would be escalated to the assignment authorisor and as such there is no need to worry about that part of development.

To split up the work properly, I followed the unit specification and worked my way through it day by day, doing as much work as I could make myself do in that time and then focusing on developing my application when that part of the project came up. This would be the best way to do is at it allowed for me to finish my project properly and work on a good schedule.

### Managing product delivery

Managing product delivery is relatively simple as it only just making sure that the work mentioned above is handled properly and work is done on time in order to ensure that the deadline will be met in a timely manner.

### Managing stage boundaries

This part of PRINCE2 development consists of updating the board, and in my case my assignment authorisor on the development of my project and receiving feedback from both peers and project authorisors alike in order to move forward and finally focus on closing development and being able to conclude my report about development, allowing me to close the project entirely.

### Closing a project

Finally, the project is closed and development on the application alongside the application development documentation can be finalized and closed out. The project will be decommissioned by the assignment authorisor. The project end also means that it will be reviewed, and in my case this means that the project will be graded and I will receive a result for my project work.

This review also allows me to focus on the process of my whole project and look at the benefits and limitations of developing my application in the way that I did.

## Conclusion

To conclude, PRINCE2 turns out to be a very effective and thorough project management methodology. Its constant documentation allows for the ability to look back at a certain period in development and understand what was going on at the time. Furthermore, its thorough stages ensured that the project was covered on all ends, from authorization to project closing and review, or in my case grading, and allowed for me to achieve the highest review possible for this project.

Moreover, its constant documentation also allows me to repeat this project in the same way if I so wished, meaning that it can be used for repeated application development.

# Social Media Charity Case Study – Agile methodology

## Stages of Agile with a Social Media Campaign

### Concept

### Inception

### Iteration

### Transition and release

### Production

### Retirement

## Conclusion

# Comparison of different methodologies and structures

# Evaluation of different methodologies and structures

# Bibliography

Chart, L. (2022, October 6). *Lucid Chart*. Retrieved from Lucid Chart: https://www.lucidchart.com/blog/rapid-application-development-methodology

Chart, L. (2022, October 6). *Waterfall project management methodology*. Retrieved from Lucid Chart: https://www.lucidchart.com/blog/waterfall-project-management-methodology

train, K. (2021, July 21). *PRINCE2 Benefits, advantages and disadvantages*. Retrieved from Knowledge Train: https://www.knowledgetrain.co.uk/project-management/prince2/prince2-course/prince2-benefits

Wrike. (2022, October 6). *What is Agile*. Retrieved from Wrike: https://www.wrike.com/project-management-guide/faq/what-is-agile-methodology-in-project-management/