

Software Project Management Plan

Team Name:

Keeping Up Appearances

Team Members:

Vilius Chockevicius - w18007679

Robert McNicol - w18007336

Dawid Michniuk - w18016997

Dominykas Patinskas - w18016919

Programme of Study:

Computer Science with Artificial Intelligence

1 Introduction

1.1 Project Overview

Project Summary:

“We want to try restoring poorly aged images, specifically by adding colour to, what were originally, black and white photos. We know that there is a need for this technology since there are entire communities revolving around photo restoration and colorization. After we achieve good results in this area, we could potentially try adapting the technology to colorize and restore archaic/old movies” – *Project Idea excerpt from Project Proposal*.

The project is to address the following problem statement: “If we have a black and white image of a given size, how can we develop an AI application to successfully colourise this image?”. Of course, this problem statement arises questions of its own, such as “how do we define ‘successful colouration’ of images?”. These questions, however, will also be addressed throughout the project where necessary.

1.2 Project Deliverables

This project intends to achieve

- A fully functional **Artificial Intelligent Model** which successfully converts black and white images into colour.
- A successful development of **API and GUI**, to implement a bespoke client-based service/product
- An accompanying **Mathematical Review Report**, which outlines the mathematical theory and basis for the models explored in the development process, and ultimately used in the final build.

1.3 SPMP Evolution

Due to the research style of this artificial intelligent based project, and the likely uses of an iterative software process model, this document (and other relevant documents) may be changed and revised as the project continues along its potential changing course. In light of any new information, the understanding of technical requirement changes, or change in project circumstances, the project documentation will be updated accordingly.

2 Project Organisation

2.1 Software Process Model

2.1.1 Baseline Processes

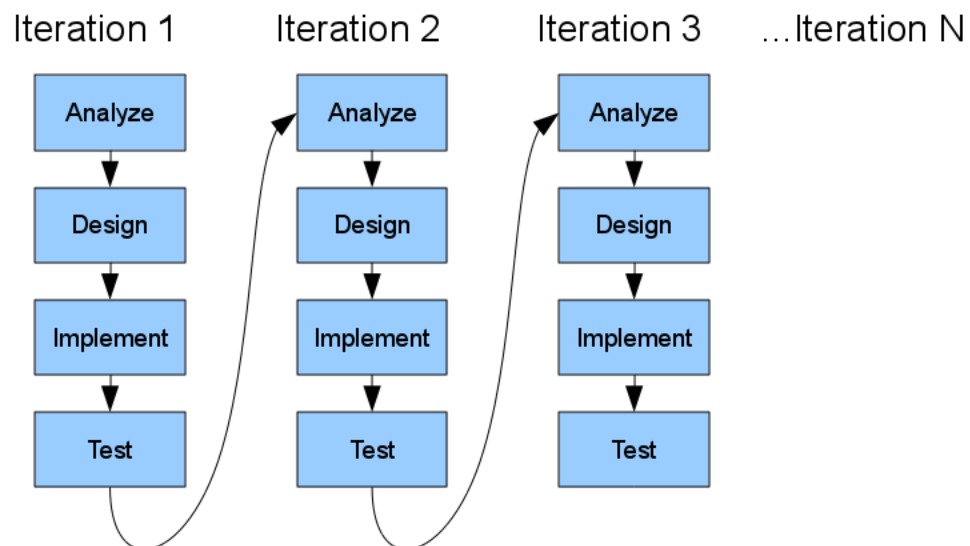
The Baseline Implementation follows a flexible experimental approach, to ensure a wide variety of ideas and models can be explored. Implementation based on successfulness and appropriateness of models follow and forms the base for the software iteration development.

2.1.2.a Iteration Development

The Iteration Development stage of this project adopts an Agile Model approach. The Agile Model can be described as an iterative approach, where, “during each iteration a single feature or small set of features are chosen and implemented completely”¹

As this is a student project for a single module, the semester dictates the maximum length of the project, and as such, means that everything involved in a software engineering project is effectively squashed down into a much shorter timeframe. Consequently, the project makes some manipulations to the standard agile method, and rather following a specific industry standard, is more inspired by the concept and uses the iterative idea. Hence the project implements a ‘light touch agile scrum’.

The iterative basis for agile can be represented as such by the diagram below²



^{1, 2} <http://users.csc.calpoly.edu/~jdalbey/308/Lectures/SoftwareProcessModels.html>

2.1.2.b 'Light Touch Agile Scrum'

Within the iterative development stages, the project adopts separate scrum meetings to the main meetings, to help identify and have understanding about the following things:

- What are the priorities?
 - o (Must have's, Should have's, Could have's, for example)
- How can these priorities be achieved?
- What is the realistic timeframe for these?

Further, once each scrum meeting has taken place, there will be a review upon the achievements made from the latter meeting's priority list. This will help further identify the following:

- What was achieved since the last meeting?
- Did we do everything we had planned?
 - o If yes: have the priorities implemented had the desired effect on the progress if the project?
 - o If no: Why did these not get achieved? Do they even need these at all?

Using this 'light touch agile scrum', the project will have a clear layout of the critical path being taken and what features are on that path.

2.2 Roles and Responsibilities

The allocated roles and responsibilities for this project, in accordance with the assignment specifications, are as follows:

- **Dawid Michniuk** will be taking on responsibility for ***Baseline Implementation*** and ***Application Programming Interface***.
- **Villus Chockevicius** will be taking on responsibility for ***Project Ideation*** and ***Solution Design***.
- **Dominikas Patinskas** will be taking on responsibility for ***Solution Testing*** and ***Graphical User Interface***.
- **Robert McNicol** will be taking responsibility for ***Project Management*** and ***Mathematical Review***.

Since we are on the **Artificial Intelligence** stream of the software engineering practice module, the group task we will all be responsible for is ***Iterative Development***.

The following table describes the roles and responsibilities of the above project phases in more detail:

| Project Phase | Description | Phase Lead |
|-------------------------|--|---|
| Project Management | Preparation of management documentation, adapt and update documentation where needed, responsible for settling disputes, and keeps project critical path heading forwards. | Robert McNicol |
| Project Ideation | Creation of the project problem statement, discussing the history of the problem, points out key importance of the project, and outlines the project plan for problem tackling. | Villus Chockevicius leads this phase and its documentation, while the rest of the team contribute to this phase with project idea discussion. |
| Solution Design | Specification of initial solution creation based upon project working title. Design choices and requirements decided upon for implementation. | Villus Chockevicius |
| Baseline Implementation | A foundation layer for the rest of the project, implementing the very first solution to the problem. Used also to test infrastructure and document a benchmark for performance comparison as the project progresses. | Dawid Michniuk |
| Iterative Development | Using the baseline solution as the foundation, our solution will be iteratively improved and changed, and data pre-processing will be experimented with in parallel. Parameter tuning and experimenting with various machine learning models will occur. | Team Responsibility: This phase will be led in different sections of requirements. Where appropriate, each member may head up particular requirement sections as iterative development proceeds. |
| Solution Testing | Weakness identification within the system by actively trying to confuse machine learning model implemented. Provides a plan for testing and | Dominikas Patinskas |

| | | |
|-----------------------------------|--|----------------------------|
| | completes and reports findings on the testing process. | |
| Mathematical Review | Formalise the problem mathematically. Provide possible mathematical approaches to the solutions for the problem. Implement findings within iterative development stages to better the system performances. | Robert McNicol |
| Application Programming Interface | Interface development to provide a vessel for the model to be used and be accessible in the real world | Dawid Michniuk |
| Graphical User Interface | Develop a user interface working with the API to enable the interaction between the user and the model. | Dominikas Patinskas |

2.3 Tools and Techniques

The following are the tools, techniques, and general standards used in the different phases of the project:

| Project Phase | Tools/Techniques/Standards | Phase Lead |
|--------------------|--|----------------------------|
| Project Management | All project management documentation should look professional and maintain some consistency in style, although not essential. This software project management plan generally follows an industry standard by loosely resembling IEEE/EIA documentation templates designed for student projects in software engineering. | Robert McNicol |
| Project Ideation | Documentation standards to look professional and include industry standard referencing where applicable. Documentation style up to phase lead. | Villus Chockevicius |

| | | |
|-----------------------------------|--|-----------------------------|
| Solution Design | <p>Documentation standards to look professional and include industry standard referencing where applicable.</p> <p>Documentation style up to phase lead. Any other industry standards used in the design stages to be documented within the mission documentation.</p> | Villus Chockevicius |
| Baseline Implementation | <p>Documentation standards to look professional and include industry standard referencing where applicable.</p> <p>Documentation style up to phase lead.</p> <p>Software uses/ required packages: Python, jupyter notebook, Keras, Skimage,numpy.</p> | Dawid Michniuk |
| Iterative Development | N/A due to COVID19 | Team Responsibility: |
| Solution Testing | <p>Documentation standards to look professional and include industry standard referencing where applicable.</p> <p>Documentation style up to phase lead. Any other industry standards used in the testing stages to be documented within the mission documentation.</p> | Dominikas Patinskas |
| Mathematical Review | <p>Documentation standards to look professional and include academic standard referencing where applicable.</p> <p>Documentation style up to phase lead but must be designed with academic papers in mind. Any other academia standards used in the review stages to be documented within the mission documentation.</p> | Robert McNicol |
| Application Programming Interface | <p>Documentation standards to look professional and include industry standard referencing where applicable.</p> <p>Documentation style up to phase lead.</p> | Dawid Michniuk |

| | | |
|--------------------------|--|----------------------------|
| | <p>Software uses/ required packages: Flask</p> <p>Any other API standards used in this stage is to be documented within the mission documentation</p> | |
| Graphical User Interface | <p>Documentation standards to look professional and include industry standard referencing where applicable. Documentation style up to phase lead.</p> <p>Software uses/ required packages: Flask</p> <p>Any other GUI standards used in this stage is to be documented within the mission documentation.</p> | Dominikas Patinskas |

3 Project Management and Control

3.1 Team Meetings

The team will generally meet twice weekly, once on Mondays immediately after our lectures, and once in our Thursday timetabled seminar sessions. The Monday meeting will focus on planning and strategy as well as documenting and discussing the project elements, while the Thursday meeting will provide an opportunity for the team to work collaboratively on segments of the project or complete any outstanding work for their own missions. There will usually be an agenda for the Monday meetings to keep the team on track, but there is space for deviation and more or less discussion where needed. A member of the team will take meeting minutes, most likely on a turn-based system. Any key decisions made in these meetings (or any other meeting) will be recorded in dated meeting document and filed. Further, the scrum meetings during iterative development will most likely take place during the Monday meeting itself as a segment on the agenda.

3.2 Project Log

For ease of access to files and meeting notes and any other mission specific documentation, the team has opted for the use of Google Drive to keep hold of all the project files. The dated entries and abilities to comment and edit on the fly with Google Docs makes this a sensible and efficient solution leaving more time for working and less time making up extra logging documentation.

3.3 Project Risk Management

The objective of this section is to identify and mitigate risk, or potential sources of expense or delay. Some will be very common, and some are associated with specific phases of the project.

Common Risks:

- **Team member unavailability.** It is most likely certain that at some point during the course of the project some members will unfortunately not be able to make meetings or various other group activities for the project due to illnesses, emergencies, or general exceptions.
 - **Probability:** High
 - **Impact:** Low
 - **Prevention:** Team members should always alert the team lead or meeting leader at the soonest opportunity for any absences that they know they will have to take. For any responsibilities that need to be covered they will coordinate with other members of the team for the duration of their absence.
 - **Correction measures:** In the unlikely case of excessive unavailability or unannounced availability a team member will have to trigger a meeting for discussion. This discussion will cover whether continued

unavailability will mean that the team member has to be contacted by tutors or module staff.

- **Miscommunications.** Due to the length of the project and the volume of communication occurring there is likely to be miscommunications between team members.
 - **Probability:** High
 - **Impact:** Medium
 - **Prevention:** To avoid miscommunication of any important verbal communications about project deliverables, these will be documented for verification by ensuring that these discussions are held in meetings. Further, due to the nature of using communication via digital means, there will be a natural log of communication between the team and its members. In any confusion, members will be able to refer back to conversations had in these forums.
 - **Correction:** In the case that miscommunication is common throughout the project, there will have to be a review on the current methods of recording important information regarding the project and an agreed upon approach between all team members of how to fix the issue.
- **Changes to project scope.** As this is the first major group project that the team is undertaking at the University, and the nature of an artificial intelligent project, it is likely that our desired achievements and project progress may not line up. It may also be the case that the project scope isn't wide enough, and we have achieved all we want to achieve and therefore would like to make more progress within the time frame.
 - **Probability:** High
 - **Impact:** High
 - **Prevention:** It is certainly the case that we do not want this to be happening regularly and therefore we should come up with a definite project scope that although allows for some changes will mainly stay the same. However, changes to the project scope may be necessary so long as they don't derail project timelines.
 - **Correction:** All deadlines will be honoured unless the circumstances allow for deviation and therefore any project scope changes but keep these deadlines will be considered and allowed if all the team members deem fit.
- **Missed Deadlines.** As this is the first team project carried out by us as a team, it is entirely possible that as students, external responsibilities or other reasons may hinder the ability to make deadlines or complete necessary tasks on time.
 - **Probability:** Medium
 - **Impact:** High
 - **Prevention:** During weekly meetings where appropriate there will be a review on the project specific deadlines and other non-project related time commitments going on throughout the University. It is expected that no member takes on more responsibility than they can

handle, and they only accept work that they can reasonably expect to accomplish by the deadlines.

- **Correction:** The missing of some deadlines may not be too impactful and therefore can be rescheduled, however, some deadlines may need to be incredibly strict and multiple failures to meet other deadlines will trigger a team discussion. In the unlikely event that there is continued failures to meet deadlines by a specific team member then there will be a discussion with the faculty and module advisors on how to proceed.
- **Mismanagement.** Due to the nature of this project mistakes will be made in the project management as the students involved have never done this before and are still learning what project management requires.
 - **Probability:** High
 - **Impact:** High
 - **Prevention:** Project manager will make sure that they understand the feedback from team members regarding their performance. There will also be project management reviews with tutors scheduled into the project timeline where any issues with project management can be discussed with the module tutors.
 - **Correction:** the project manager will not be able to be unseated from this position due to the way missions work, however, given continuous mismanagement the project manager may be taken up to the module tutors under discussion will take place on how to proceed the management and understand why mismanagement has occurred in specific instances present.
- **Misunderstanding the Problem Domain.** With a vast array of ways of which to address the problem, it may be easy to forget the understanding of the problem itself, under there for become not precise in a project goal.
 - **Probability:** Low
 - **Impact:** High
 - **Prevention:** To ensure there isn't a misunderstanding of what the project is trying to address at its root, there will be more than sufficient research conducted throughout the different missions to ensure that all team members have a thorough understanding of the problem at hand.
 - **Correction:** in the case that there has been a misunderstanding, there will need to be a review into how precisely the problem was formalised and the various ways a solution may present itself. Assuming all team members have a rough idea of where the project is meant to be heading, there can be a meeting the discuss is precisely what the aims of the project are and where the deliverables come from in relation to the problem.

Software/Model Design Risks

- **Incorrect Design:** Software on model design is the foundation for the implementation of the project. Any errors relating to the design will cause dramatic effects for the rest of the project.

- **Probability:** Low to Medium
- **Impact:** High
- **Prevention:** it is absolutely essential that there isn't an implementation of an incorrect design during the software implementation and iterative development stages. Therefore, any model designs that the team is unsure of will be run past module tutors just to be sure that they won't derail the entire project.
- **Correction:** In the case that there is a design flaw that happens to be implemented there will be a review into how this can be manipulated to the best possible design as to not waste the time that has been put in whilst ensuring that the project deliverables are met.

Software/model implementation Risks

- **Lack of experience with the given technologies:** Most, if not all, of the software used in this project is unfamiliar to the whole team. It is not unlikely because of this that the lack of experience will affect the standard and quality of code being written.
 - **Probability:** High
 - **Impact:** High
 - **Prevention:** It will be necessary for the learning of the technologies being used by either all of the team members or by specific members in their individual missions to be learnt to a thorough level.
 - **Correction:** If the necessary groundwork to the understanding of the given technologies for the project is not met or achieved, and this is noticeable in a team member's or the entire team's work, Then there will have to be a period of time where implementation is put on hold to go back and fully understand how to implement the necessary features for the project.
- **Inconsistency with Design.** With there being time pressures on the implementation stages due to the short project length of a single semester there may be the temptation to make shortcuts which will create a deviation from the desired product discussed in the design phase.
 - **Probability:** Medium
 - **Impact:** Medium
 - **Prevention:** To ensure that unplanned deviations to the design do not take place, there will be an attempt to keep the design resources accessible at all times throughout
 - **Correction:** If the necessary groundwork to the understanding of the given technologies for the project is not met or achieved, and this is noticeable in a team member's or the entire team's work, Then there will have to be a period of time where implementation is put on hold to go back and fully understand how to implement the necessary features for the project.

3 Iterative Development Plan

Side-lined due to COVID19:

Unfortunately, due to the disruption caused by the COVID-19 pandemic, the iterative development mission was dropped and therefore this plan is incomplete. However, see below how the descriptions, deliverables, resources and dependencies as well as risks would have been documented for every task performed in the project from the following skeleton: Note the project schedule exists due to that being created before the pandemic.

3.1 Tasks

3.1.n Task-n

3.1.n.1 Description

3.1.n.2 Deliverables and Milestones

3.1.n.3 Resources Needed

3.1.n.4 Dependencies and Constraints

3.1.n.5 Risks and Contingencies

3.2 Assignments

3.3 Timetable – Project and Task Schedules

3.3.1 Project Schedule

| Missions | Mission Responsibility | Start | Approx No. Weeks | Pre-holiday termtime | | | | |
|-----------------------------------|------------------------|------------|------------------|----------------------|------------|------------|------------|------------|
| | | | | | | | | |
| Main Missions | | | | 24/02/2020 | 02/03/2020 | 09/03/2020 | 16/03/2020 | 23/03/2020 |
| Project Ideation | Vilius | 28/05/2020 | 3 | | | | | |
| Solution Design | Vilius | 02/06/2020 | 2 | | | | | |
| Baseline Implementation | Dawid | 25/05/2020 | 2 | | | | | |
| Iterative Development | Team | 17/06/2020 | 3 | | | | | |
| Solution Testing | Dominykas | 03/06/2020 | 4 | | | | | |
| Side Missions | | | | | | | | |
| Mathematal Review | Robert | 03/06/2020 | 3 | | | | | |
| Application Programming Interface | Dawid | 04/06/2020 | 4 | | | | | |
| Graphical User Interface | Dominykas | 12/06/2020 | 4 | | | | | |
| | | | | | | | | |

SPMP Signed by:

David Michaud

Jim

Greg

John