# Project Management Report

**Group Name: MCS08**

**Word Counts: 2697**


**Team members:**

1. **Yew Yee Perng (32205481)**
2. **Yeoh Ming Wei (32205449)**
3. **Toh Xi Heng (33200548)**

# Table of Contents

# 1. Introduction

Our team, MCS08 was given the project topic, Singing Video Generation with Music Separation which is supervised by Dr Arghya Pal. Our goal in this project is to create a cutting-edge system capable of generating lifelike singing face videos synchronised with music. This can be separated into two parts of work, which are audio separation and lip synchronisation to generate a video. In our team, we are focused on the lip sync part and we will use SadTalker as our source code.

The final result of the project will be a web application that can easily generate the lip synchronisation video with the input audio and image.The development of this web application involves a collaborative effort, combining frontend development for user interface and user experience, backend development for server-side logic and architecture. To ensure the quality, the model testing will also be done.

As a team, we are all developers. However, we all have our own contributions distribution, such as Xi Heng and Ming Wei for front-end development, Ming Wei and Yee Perng for back-end development, and Xi Heng and Yee Perng for model testing to ensure the quality of the application. These roles will likely change as we progress with the project.

The team members of this project includes:

- Yeoh Ming Wei  (Team Leader)
- Yee Perng Yew (Software Architect)
- Toh Xi Heng (Quality Inspector)

In our research, we are focused on developing generative AI capabilities with a primary focus on AI-driven facial generation. By integrating cutting-edge techniques in facial animation, we strive to create innovative methods that produce highly realistic and expressive facial animations. Through this work, we aim to expand the boundaries of AI-generated visual content, making a significant contribution to the field of multimedia synthesis and unlocking new possibilities for creative AI applications.

# 2. Project Management

## 2.1 Brief Introduction to Project Management

Project management is a systematic approach to ensuring the successful completion of a project. It involves planning, organisation, resources and tasks control, and schedules to achieve specific goals within specified constraints. To achieve this, project managers must define clear and achievable goals, identify specific tasks and deliverables required, develop a timeline and deadlines, allocate the resources for example team members and budget. Besides that, he needs to identify and mitigate potential risks and make effective communication between team members and stakeholders, and also continuously monitor progress to ensure projects are on track. By following these principles, organisations and individuals can more easily navigate complex projects, increase efficiency, and increase the likelihood of success. In the next section, we will outline the reasons for choosing a project management approach and how we implement it in our projects.

## 2.2 Our Project Management Methodology

In order to have a better process for our project that leads us to success, project management is necessary to ensure a better flow of process and understanding on what should be done step by step. We did some research on some methodology such as Scrum and Kanban.

In short, to explain the respective methodologies, Kanban is a framework that manages work through visual tasks. Scrum framework assists on team structure and manages work through a set of values, principles and practices. Since both have differences, we thought that both are actually useful and beneficial for our project. Hence, we decided to follow or modify some of the principles to adapt and develop our own methodology inspired by the other methodology.

In terms of roles and responsibilities, we did not follow any of the methodologies and instead had everyone to be responsible for it. For example, everyone is required to attend the meeting with the supervisor and work has to contribute equally in terms of coding or reports. As we based off the scrum roles, we think that everyone in the team is considered as product owner and development team.

We had a meeting with our supervisor once a week. In the meeting, we tried to present our findings for our topic to our supervisor, discussing what we did for the past week and also what we are going to do for the current week. This method is similar to daily stand-up proposed by Scrum methodology which were required to perform a stand-up every day. Then, it is also considered a sprint review where our team presents to our supervisor and gathers feedback from our supervisor to improve it until the next meeting. Lastly, we did a recap on our mistake and brainstormed on what improvements should be made which is similar to sprint retrospective.

Besides that, it also aligns Kanban's continuous improvement philosophy, where our team dedicates a specific amount of time to complete our research and enhance work quality. In addition, we had more group meetings as well so that we could help each other during that time. We also tried to use Kanban board as well to provide a better visualisation on what should be done for our project. We think that it is very useful especially to keep track of our progress and appearance might help us reduce mistakes too.

In conclusion, the combination of Kanban and Scrum, a hybrid approach, has allowed us to have better project management. One of them is continuous improvement through regular feedback from our supervisor every week. This will show our product better results every week. Besides that, more communication through meetings provides more efficiency and teamwork when working.

## 2.3 Project Resources

### I. Resource Allocation Approach

When we distribute the task for team members, we assign the task ourselves to the task that we are more interested with so that everyone can stay motivated and engaged with the project. This approach also allows us to ensure that the tasks are distributed fairly and efficiently, taking into account the strengths and weaknesses of each team member.

As mentioned above, in order to avoid time clashing between the activities of each team member and supervisor, we have designated a time to do a weekly meeting as a sprint meeting. In each sprint meeting, we review the accomplishments since the previous session, discuss any obstacles faced, and strategize for the upcoming tasks. Besides that, we also have our own group weekly meeting without the supervisor on weekends. We share and discuss our updated process before this meeting and prepare the question to ask for help from the supervisor. This helps us be able to keep track of the progress done by other team members and can do a full review to the code implementation.

For technical purposes, we use the Jupyter notebook inbuilt in Google Colab to run our pre-trained model and execute the web application. The version control for these codes will be saved in Google Drive. On the contrary, we will implement the SadTalker source code from the Visual Studio Code where it is easy to clone with GitHub to do version control. In the same place, we will develop the web interface using JavaScript, CSS and HTML. Jira will also be used to monitor our progress.

### II. Software Tools

To keep track of our team progression, we have used few software tools to do so. The first is we use Whatsapp to do our communication. We could instantly message each other for quick updates, discussions, and coordination of smaller tasks. Besides that, we also use Google Chat to communicate with our supervisor. Any question we face, we will directly contact him or get help from his PhD student from Google Chat. Furthermore, we also use Discord to do our meetings online. Discord's audio and video quality is generally good, making it possible to have high-quality meetings with clear audio and video.

We use Google Colab to do our implementation with the notebook inbuilt as it has a faster GPU to avoid the powerful hardware needed and has PyTorch inbuilt which we would use to run the pre-trained model. We can also easily use SadTalker source code because the SadTalker developers have shared the code publicly on Colab for us to run. Google Drive and GitHub will be used for us to do version control as they

are easily connected with Colab. GitHub also provides a commit history that helps other team members to do reviews for what has been updated.

In addition, we use Jira to do our project management as we use its provided features like sprint planning, backlog refinement to keep track of our progress. It is a nice software to do planning following Agile.

## 2.4 Risk Management

Risks are unexpected events that could affect the project. To avoid experiencing any bad effects on the project, monitoring for risk in a project is vital and measurements should be taken to ensure the project completes smoothly in time. Our team has created our risk register before the start of the project, which takes into account some various situations and possible risks that we could encounter and how we should mitigate them. Besides that, we do look into our risk register from time to time to see if updates are required to ensure all the risks to our current progress can be identified and mitigated.

| # | Impact score/100 | Probability (%) | Description | Risk Response | Impact | Monitoring strategy |
|---|---|---|---|---|---|---|
| 1 | 50 | 10 | Device malfunction of team member | Ensure that the work is pushed to GitHub daily to prevent any code loss and backup devices if required. | Our team progress might be slowed since one person might not be able to work on it. | Ensure that device works perfectly fine as it should be. |
| 2 | 80 | 60 | Conflict between team members | Ensure that major decisions are discussed and voted before performing on it | Might affect the collaboration between team members which slows down progress. | Do regular meetings and get to know each other better. |
| 3 | 70 | 30 | Technical challenges that appears in the project | Ensure that team members are up to date on the knowledge required for the project, having basic understanding on the model and platform the team is using. | Might cause a halt in progress if the issue is not settled. | Do meetings regularly and help one another when another is in need. |
| 4 | 80 | 20 | Project delays due to insufficient time | Ensure that the progress of the team is similar to the goals set by the team | Might not be able to present the desired completed project the team wanted. | Regular meetings to update each member’s progress. |

| | | | | | | |
|---|---|---|---|---|---|---|
| 5 | 90 | 5 | Client is unsatisfied/ decides to change the required in the middle of development phase | Host a meeting and act upon it immediately by changing as much as possible based on the time and resources available to implement the changes requested. | Project might be replanned and might take up a lot of time | Have weekly meetings with the client to clarify and present the weekly findings to ensure both sides are happy on the current work |
| 6 | 10 | 90 | Team member's availability | Proceed with the meeting with the available members and update the ones that are absent on our medium of communication | This could cause miscommunication or delays in project if they fall behind on the project | Ensure that meetings/ deadlines are preplanned a few days ago to provide ample time on scheduling time |
| 7 | 95 | 5 | Loss of important data | Ensure that all members have access to important data on the cloud server so it does not cause any delays or loss of data. | Could be time consuming if required to redo/ regain the loss data | Check and backup data regularly to ensure no file is left out/lost |
| 8 | 65 | 30 | Lost contact with client | Ensure that client has kept in contact with the team for updates/ communication | Could cause a major delay in progress if client does not respond in a long time | Keep regular contact with the client. |

## 2.5 Limitations

Scope Creep

Although our aim is to enhance the video quality, we may try to add some features. In this situation, scope creep may happen when additional tasks or features are added to the project without proper control, leading to potential delays.

This is quite challenging as we will keep doing changes to the project scope without proper evaluation and approval, which can lead to a snowball effect of scope creep. This can result in a project that is no longer aligned with the original objectives and goals, and may even compromise the quality and integrity of the final product.

Time Constraints

As we had other units' assignments to work on and due to time constraints, we do not have enough time to insight into it and test it thoroughly and get the project to perfection. Therefore, it is crucial to prioritise tasks, focus on the most critical aspects of the project and adjust the schedule as necessary to avoid a rushed completion of the project that results in quality suffers.

Limited Knowledge

Since we are new with Colab and PyTorch, we need time to learn the way of implementing with them. Due to self-study, there must be a lack of knowledge in some aspects, which makes the project results not necessarily meeting our expectations.

Reliance on Pre-trained Models Limitation

We have no ability to collect the data and train the model on our own which makes us forced to rely on the pre-trained model that is published publicly. They are trained on general datasets and may lack the precision or specificity required for specialised tasks. For example, the dataset commonly published are trained with English but we expected more language. This could make the result cannot reach our expectations.

## 2.6 Reflections

Although we had our own methodology and we managed to deliver our projects successfully, there is more room for improvement that will help us to reduce our risk and limitations. The success of being able to develop our own methodology and adapt it. Then, we were able to deliver our project during our meeting with the supervisor. Besides that, we tried to take notes on the feedback given by our supervisor and make changes to improve it so that we are able to prepare a better one for our next meeting. However, we think that our work is only the bare minimum due to not being able to put more effort into our work because of time constraints. As a solution, we should discuss on which time we are available and allocate more time to do our project. It is important to have a better quality of work as well to impress our supervisor. In addition, better time allocation will give us more time to learn how to use Colab and Pytorch and get familiar with it.

Besides that, I think that we should have stricter project management such as having a more precise task breakdown and having weekly standups to keep track of our team's progress. This benefits our team by having a better process, knowing what should be done first. Weekly standup is similar to daily standup from Scrum methodology that allows effective communication between team members so that team members are able to see the progress and help each other when it is needed.

# 3. Conclusion

Project management is a very useful skill that guides our team to a better direction. Although we do not follow entirely how both Scrum and Kanban methodology works, we tried to combine both as a hybrid approach and picked the one that we think that it helps our team. In the end, it was successful but still required some modification to make it more robust for project management. Through reflection, we found some flaws but it did not really affect the progress of our project. However, we strive to have a better improvement and hence the suggestions given above. So, hopefully our team will have better management and produce a better result for our project in the future.