CAT404 – Software Engineering Major Project

Project Proposal

Postgraduate Practicum Management System

Subsystem 2: Practicum Presentation Scheduling

SE22230052

Teoh Yu Fei, Assoc. Prof. Dr. Wong Li Pei

yufeiteoh@student.usm.my, lpwong@usm.my

School of Computer Sciences, Universiti Sains Malaysia

11800 USM, Penang, Malaysia

Abstract

Presentation often held in tertiary education. For example, postgraduate students, who enrolled in the master degree programme offered by School of Computer Sciences, Universiti Sains Malaysia (USM), are required to present their practicum work on the Computer Science Masters Innovation and Demo (CSMInD). For each presentation, it involves a postgraduate student, a supervisor, an examiner and a chair. The company mentor of postgraduate student is optional to join. Each presentation will need to be scheduled in such a way that it meets a set of constraints. Scheduling the presentations manually is a tedious task as it needs to avoid the conflicts between the time slots. It will be time-consuming and might increase the risk of human error since there are a lot of people involved in a presentation. Therefore, the practicum presentation scheduling system is proposed to generate a satisfactory presentation schedule using the genetic algorithm method. This algorithm uses biological principles concept to provide a solution for an optimization problem. The proposed system is expected to provide a platform for people who involved in the presentations to input their constraints, allow the programme manager to schedule the presentations in an effective and efficient way, and generate multiple views for presentation schedule.

Keywords: Scheduling Problem, Genetic Algorithm, Satisfactory Schedule, Constraints

1. Project Background

By the end of CDS590 Consultancy Project and Practicum as well as CDT594 Digital Transformation Project and Practicum courses, postgraduate students are required to present their works in CSMInD, therefore, to make sure there is no conflict in the presentation schedule, scheduling comes into it. Presentation scheduling is the process of allocating student presentations to a specific time slot and venue respectively. However, it can be hard to schedule due to a presentation will involve a few attendees and it is hard to find a suitable time slot which everyone involved are available on that time. Besides, there are also other constraints which need to be considered such as the availability of the venue, the location of the venue, and etc. The constraints can be classified into two types which are hard constraints (constraints that must be applied) and soft constraints (constraints that can be violated if necessary). Hence, to make sure the presentation schedule can prevent the conflicts among the constraints and it is also able to satisfy the preferences of attendees as much as possible, a presentation scheduler is needed to facilitate the programme manager in scheduling the presentations.

There are many research articles where different methods have been proposed to solve the scheduling problem. Scheduling is a NP hard problem, therefore, in [1], it stated that to save the time in the creation of lecture schedules, they used the genetic algorithm to generate a schedule that is conflict-free. In [2], ant colony optimization approach was applied to solve the university examination timetabling problem. In [3], particle swarm optimization is used to schedule the thesis defense timetable in order to produce a timetable that is not violating the hard constraints. Based on [4], it stated that genetic algorithm has proven to be effective in scheduling and hence, genetic algorithm will be used in this project. It is a population based heuristic method which extensively used in scheduling for constraint optimization problem. It mimics the biological principles of genetic variation and natural selection. It will begin with creating a random population of timetables followed by their evaluation according to the defined criteria to select the timetables for the next generation. It is expected to produce better timetables by way of crossovers and mutations. This process will be repeated until a satisfactory timetable is reached.

As mentioned above, the proposed system will be using genetic algorithm to solve the presentation scheduling problem. This system will also provide a platform for postgraduate students and lecturers to enter their time availability (hard constraint) and also their preferences (soft constraint) for the presentations. The gathered information will be serve as the input for presentation scheduler. The scheduler will then use the genetic algorithm to produce a good quality of presentation schedule and the generated schedule will be visualized in the system. In the following sections, it will be discussing the problem statements of this project, the motivation of doing this project, the details of the proposed solution, the significance and uniqueness of this project as well as the expected outcome of this project.

2. Problem Statements

A feasible presentation schedule should be able to prevent the conflicts among all the hard constraints and it should be also satisfying the preferences of attendees (soft constraints) as much as possible. To generate the presentation schedule, several inputs are required for the constraint management in presentation scheduling. The inputs will include the hard constraints such as the availability of the lecturers, postgraduate students, and venues, as well as some soft constraints like whether the lecturers will prefer to attend the consecutive presentations, or whether they will prefer to change their presentation venue. Without a platform, it is difficult to compile or standardize all these information. Hence, these inputs should be well-captured to ease the process of scheduling a presentation schedule.

The collected information will be used as the input for presentation scheduling. To generate a satisfactory schedule, the programme manager would need to take the hard and soft constraints of everyone involved into consideration. However, it is hard to manually adjusting the schedule as the process will be time consuming and might be overlook for some constraints. Therefore, to solve this problem, genetic algorithm – a metaheuristic approach is needed for the scheduler to find a near-optimal schedule.

Without the personalized schedule, lecturers would need to make their own schedule manually by referring to the master schedule. They might overlook some presentation time slot as they need to scan through the master schedule by themselves. Thus, a personalized schedule should be provided to everyone respectively to ensure

that they will not miss any of the presentation slots and affect the progress on the demo day.

3. Motivation

The first motivation of this project is to facilitate the process of handling constraints. With a standardized input format for data entry, it is easier to handle and manage the constraints well. The programme manager would not need to manually collect the constraints one by one and transfer those data into a standardized version that can be recognized by the scheduler. With this system, it provides a platform for attendees to fill up their constraints and it will automatically change it to a version that is compatible to the scheduler.

The second motivation of proposing this project is to ease the programme manager in scheduling the presentation schedule. It is not easy to do the scheduling as many constraints would need to be considered. With this system, the programme manager will have a better performance in handling the presentation scheduling as using the genetic algorithm method will significantly reduce the time in adjusting the schedule manually and also generate a satisfactory schedule for everyone.

Another motivation for proposing this project is to make the personalized schedule visualizable by each of the attendees instantly after the master schedule is being generated. For instance, people always tend to overlook some details when all information is put together include those that are not relevant to us. Thus, we will explicitly make another individual schedule for ourselves. Hence, with this system, it will help the individual in saving time for constructing their own schedules.

4. System Objectives

This project aims to develop a web-based application which revolves around the following objectives:

1. To develop a platform for managing the hard and soft constraints.
2. To automate the presentation scheduling.
3. To provide multiple views for presentation schedule.

5. Proposed Solutions

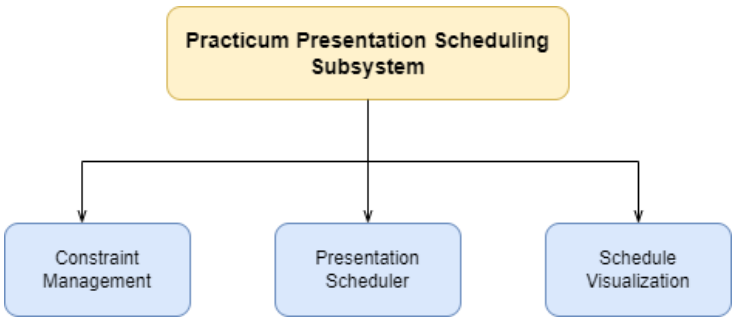


Figure 5.1: Module Breakdown Structure

The proposed presentation scheduling solution is a web-based application whereby it contains three submodules as shown in Figure 5.1. Table 5.1 shows the brief description of each submodule.

Table 5.1: Description of Each Submodule

Submodules	**Description**
Constraint Management	This submodule will help in managing the hard and soft constraints of different roles. • Lecturers can input their unavailability of time slots, unavailability of venues, maximum number of consecutive presentations and preferences in changing venue. • Students can input their unavailability of time slots. • Programme manager can input the date of demo day, the venues, and the time range.
Presentation Scheduler	This submodule will help in scheduling the presentation schedule. This scheduler will be using the genetic algorithm method to find out the near-optimized or satisfactory schedule. The generated schedule should fulfil the conditions below: ***Hard Constraints*** • All the presentations must be scheduled and each presentation must be scheduled only once.

	• No lecturer can attend two or more presentations concurrently. • The venue should be available for the scheduled presentation. • Each person must be available for the scheduled presentation. * Any violation of these constraints would make the generated schedule an infeasible schedule. ***Soft Constraints*** • The preferred number of consecutive presentations. • Preferences on changing the presentation venue if there are any consecutive presentations. * Any violation of these constraints would not make the generated schedule an infeasible schedule. However, the generated schedule should fulfil as many of these constraints as possible.
Schedule Visualization	This submodule will help in visualizing the generated schedule to the attendees. All attendees will be able to see the master schedule and each of them will be able to see their personalized schedule. Those schedules can be exported into Excel format.

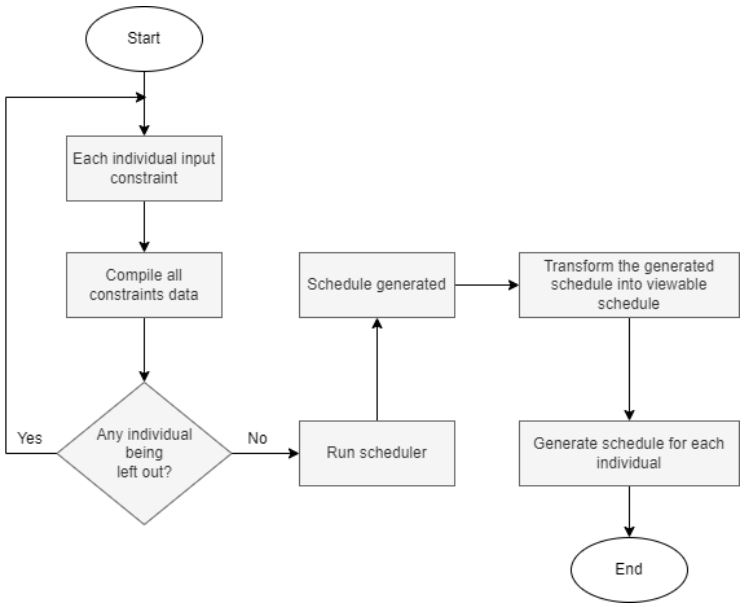


Figure 5.2: Simplified flow chart for presentation scheduling

Based on Figure 5.2, firstly, everyone including programme manager, lecturers and students are required to input their hard and soft constraints into the system. After some time, the constraints data of each individual are compiled and acts as the input of presentation scheduler. Before running the scheduler, the system should check whether all individuals have entered their constraints. If some of them did not input their constraints, then they are required to input the constraints before the scheduler can start to run. After running the scheduler, a schedule will then be generated. Based on the generated schedule, the system will transform it into a schedule which can be visualized by the people. Based on the master schedule, the system will then generate the personalized schedule respectively.

6. Benefits / Impact / Significance of Project

As mentioned in the previous section, this system will be used by programme manager, lecturers, and students. It helps to facilitate the process of collecting and managing constraints. With this system, programme manager would not need to manually call or find each of the individual to ask for their availability and other preferences. It saves time and manpower in managing the constraints.

Furthermore, with the presentation scheduler, it helps to reduce the time in scheduling the presentations. With the support of scheduler, the programme manager would not need to adjust the schedule manually based on the given constraints. The system will automatically generate a schedule that can fulfil all the hard constraints and some of the soft constraints. This will help to minimize the human error as by using manual scheduling, the programme manager might overlook some of the constraints and produce a schedule that is not satisfy by others.

This system will also provide personalized schedule for each user. The users do not need to find their time slots from the master schedule manually as each of them will have their own schedule that only include the time slots which they are involved in. Since they do not need to manually find their time slots, hence, it reduces the human error as doing manually might overlook some of the time slots and affect the progress of CSMInD.

7. Uniqueness of Proposed Solutions

This system allows the users to use the event calendar with drag and drop function to input their constraints. It facilitates the whole process of presentation scheduling because the programme manager can easily use this platform to compile all the entered constraints and make it as the input to presentation scheduler. In current practice, the programme manager will need to find each of the attendee or use another platform – Doodle to get their constraints. By using this proposed system, it will reduce the workload of programme manager and also minimize the human error.

Another uniqueness of this proposed system is it used the genetic algorithm to generate a presentation schedule that is satisfied by the users as it is based on the defined constraints which compiled from every user. The automation of presentation scheduling brings convenience to the programme manager as he does not need to manually adjust the schedule, and this saves the time as well as the manpower in doing it.

The other uniqueness of the proposed solution is the personalized schedule. This schedule will only include the time slots which related to the users only. It helps the users to see their time slots at a glance of their eyes. This will prevent the users from entering the wrong presentation slots.

8. Expected Outcomes

The expected output of this project is a web application that has the following features:

Table 8.1: Expected Outcomes of Each Submodule

Submodules	**Expected Outcomes**
Constraint Management	• Provide an event calendar which allows users to input their unavailability of time slots. The event calendar will have the editor pop-ups, drag and drop, and resizing actions features. • Allows users to choose their preferences of soft constraints.
Presentation Scheduler	• Allows programme manager to compile all the entered constraints.

	• Allows programme manager to generate a presentation schedule using genetic algorithm.
Schedule Visualization	• The master schedule is viewable by everyone. • The personalized schedule is generated respectively based on different users. • The schedule is allowed to be exported into Excel format.

9. Status of the Project

This project is not a continuation of the final year project which done by the previous students. However, this project had referred to the student presentation scheduling system. The development of this sub-system will start from scratch and this project is still in the planning stage.

10. References

[1] R. T. Subagio, Kusnadi, T. E. Putri, P. Sokibi, S. Z. Harahap, and Darmansah, "Application of Genetic Algorithm to Optimize Lecture Scheduling Based on Lecturers' Teaching Day Willingness," in *Journal of Physics: Conference Series*, Mar. 2021, vol. 1842, no. 1. doi: 10.1088/1742-6596/1842/1/012007.

[2] A. Firdaus Khair, M. Makhtar, M. Mazlan, M. A. Mohamed, M. Nordin, and A. Rahman, "Solving examination timetabling problem in UniSZA using ant colony optimization," *International Journal of Engineering and Technology(UAE)*, vol. 7, no. 2, pp. 132-135, 2018, doi: 10.14419/ijet.v7i2.15.11369

[3] G. Christopher and A. Wicaksana, "Particle swarm optimization for solving thesis defense timetabling problem," *Telkomnika (Telecommunication Computing Electronics and Control)*, vol. 19, no. 3, pp. 762–769, Jun. 2021, doi: 10.12928/TELKOMNIKA.v19i3.18792.

[4] I. A. Abduljabbar and S. M. Abdullah, "An evolutionary algorithm for solving academic courses timetable scheduling problem," *Baghdad Science Journal*, vol. 19, no. 2, pp. 399–408, 2022, doi: 10.21123/BSJ.2022.19.2.0399.

11. Appendix

[Praticum Presentation Scheduling] Project Schedule

Project Start Date: 17 Oct 2022 | Display Week: 1
Project Lead: Teoh Yu Fei

WBS	TASK	START	END	DAYS	% DONE	WORK DAYS
1	**Pre-Project Activities (Planning)**		-			-
1.1	Project bidding	17 Oct 2022	21 Oct 2022	5	100%	5
1.2	Study project background	22 Oct 2022	24 Oct 2022	3	100%	1
1.3	Identify problems and objectives	25 Oct 2022	25 Oct 2022	1	100%	1
1.4	Determine project scope	26 Oct 2022	29 Oct 2022	4	100%	3
1.5	Write initial proposal	30 Oct 2022	5 Nov 2022	7	100%	5
1.6	Project proposal vetting	6 Nov 2022	10 Nov 2022	5	0%	4
1.7	Amendment on project proposal	11 Nov 2022	20 Nov 2022	10	0%	6
2	**Analysis**		-			-
2.1	Interview on stakeholders	21 Nov 2022	26 Nov 2022	6	0%	5
2.2	Determine system capabilities, functional, and non-functional features	27 Nov 2022	3 Dec 2022	7	0%	5
2.3	Verify and validate on the requirements	4 Dec 2022	7 Dec 2022	4	0%	3
2.4	Write System Requirement and Design (SRD) Report	8 Dec 2022	25 Dec 2022	18	0%	12
3	**Design**		-			-
3.1	Identify the detailed workflow of each use case	26 Dec 2022	1 Jan 2023	7	0%	5
3.2	Design user interface layout for applications	2 Jan 2023	8 Jan 2023	7	0%	5
3.3	Design the database structure	9 Jan 2023	15 Jan 2023	7	0%	5
3.4	Design the overall structure of the system	16 Jan 2023	22 Jan 2023	7	0%	5
4	**Implementation**		-			-
4.1	Write code for constraint management module	23 Jan 2023	21 Feb 2023	30	0%	22
4.2	Write code for presentation scheduler module	22 Feb 2023	23 Mar 2023	30	0%	22
4.3	Write code for schedule visualization module	24 Mar 2023	22 Apr 2023	30	0%	21
5	**Testing & Integration**		-			-
5.1	Perform unit testing	23 Apr 2023	2 May 2023	10	0%	7
5.2	Perform integration testing	3 May 2023	12 May 2023	10	0%	8
5.3	Perform system testing	13 May 2023	22 May 2023	10	0%	6
6	**Deployment**		-			-
6.1	Deploy the whole system	23 May 2023	1 Jun 2023	10	0%	8

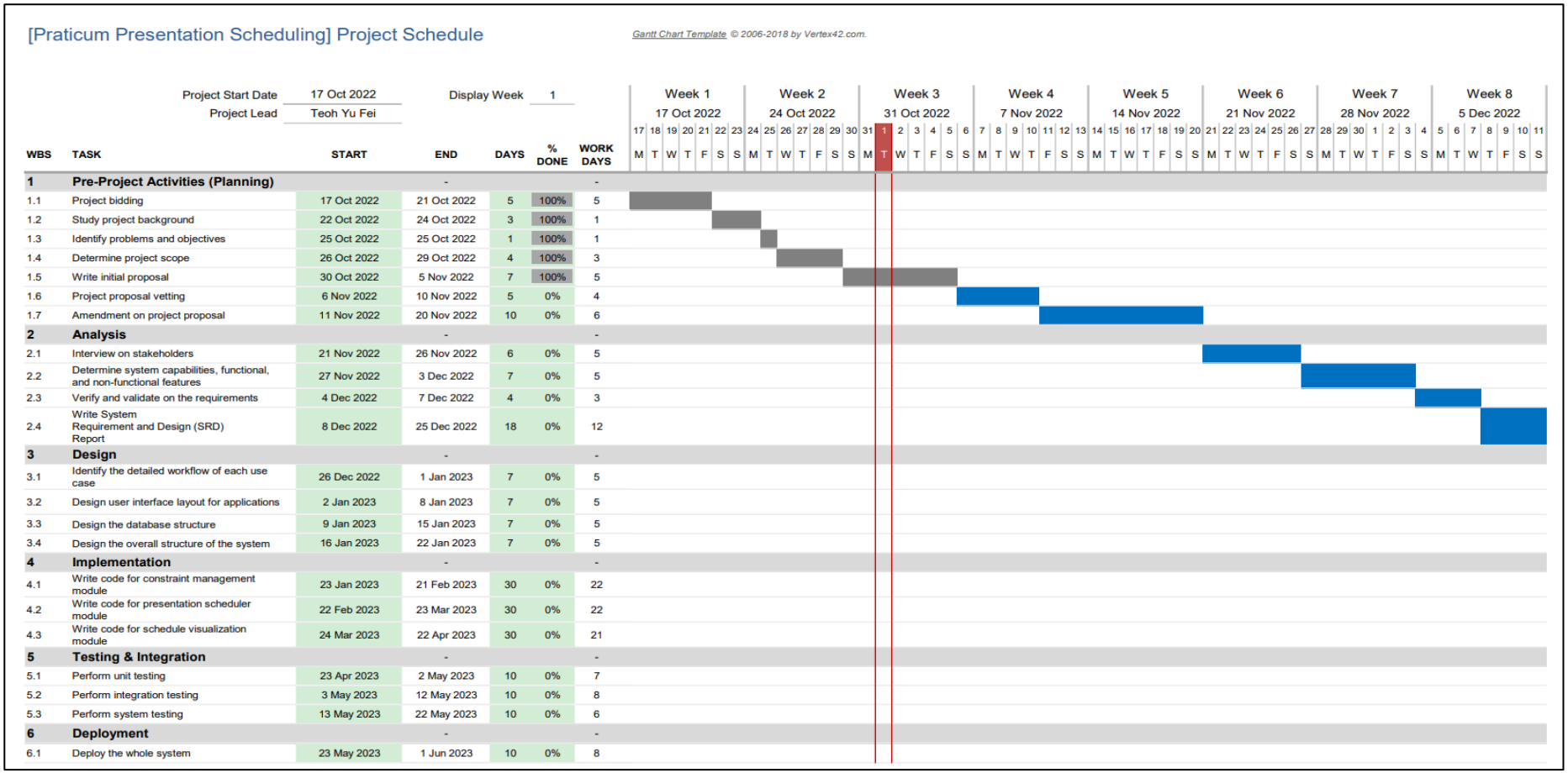
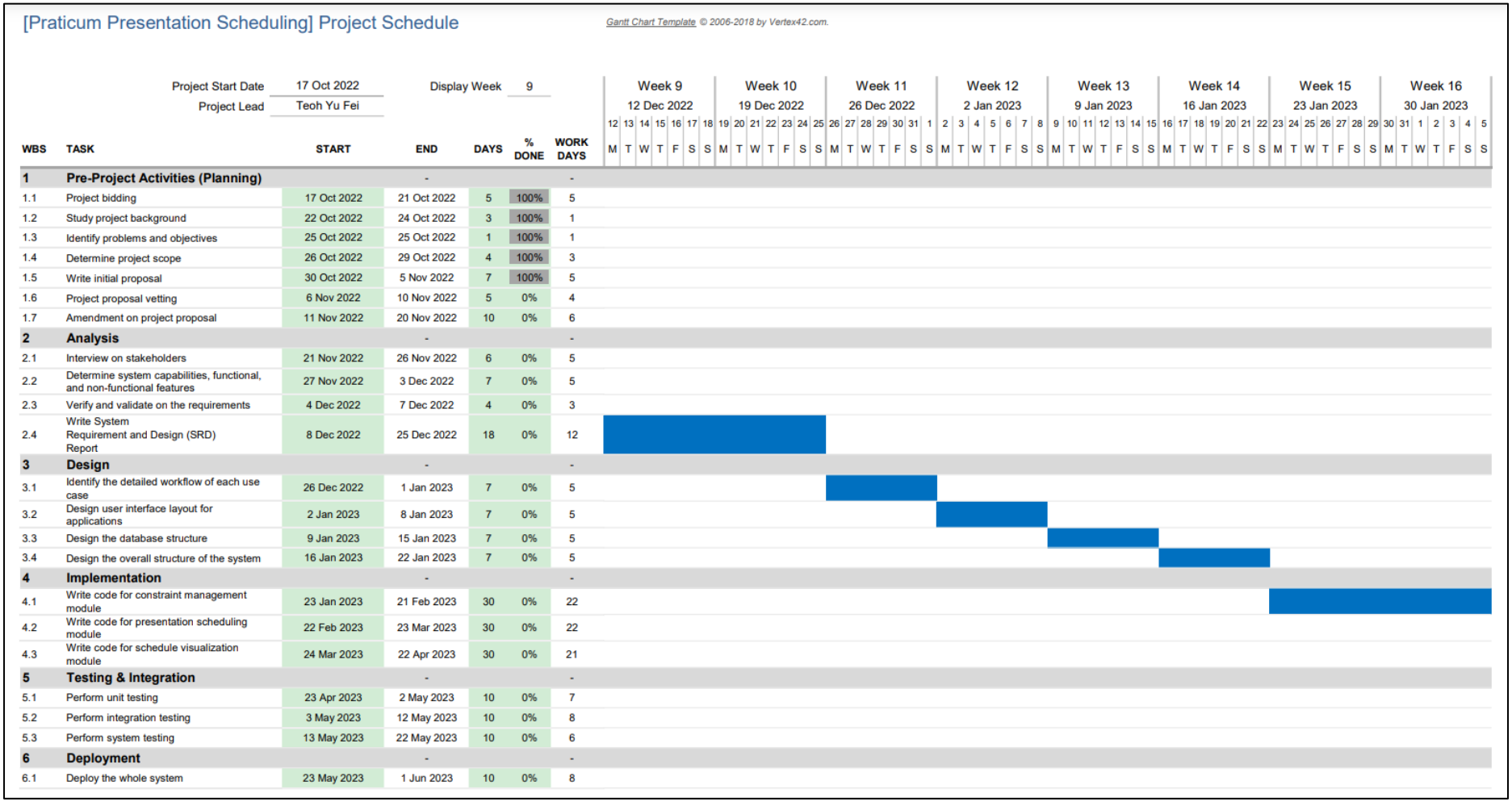


Figure 11.1: Gantt Chart of Week 1 to Week 8



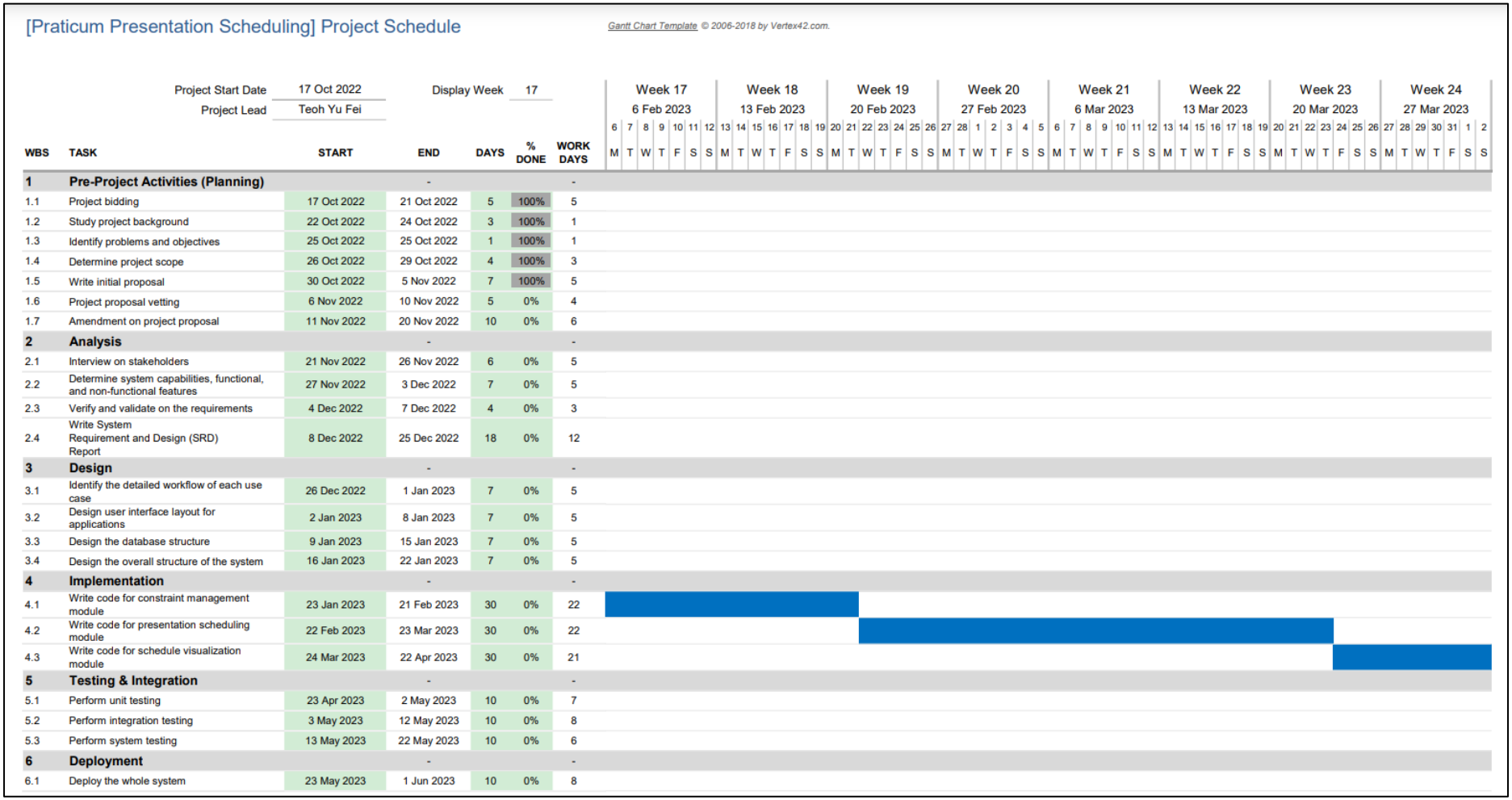
[Praticum Presentation Scheduling] Project Schedule

Gantt Chart Template © 2006-2018 by Vertex42.com.

Project Start Date: 17 Oct 2022 | Display Week: 9
Project Lead: Teoh Yu Fei

WBS	TASK	START	END	DAYS	% DONE	WORK DAYS
1	**Pre-Project Activities (Planning)**		-			-
1.1	Project bidding	17 Oct 2022	21 Oct 2022	5	100%	5
1.2	Study project background	22 Oct 2022	24 Oct 2022	3	100%	1
1.3	Identify problems and objectives	25 Oct 2022	25 Oct 2022	1	100%	1
1.4	Determine project scope	26 Oct 2022	29 Oct 2022	4	100%	3
1.5	Write initial proposal	30 Oct 2022	5 Nov 2022	7	100%	5
1.6	Project proposal vetting	6 Nov 2022	10 Nov 2022	5	0%	4
1.7	Amendment on project proposal	11 Nov 2022	20 Nov 2022	10	0%	6
2	**Analysis**		-			-
2.1	Interview on stakeholders	21 Nov 2022	26 Nov 2022	6	0%	5
2.2	Determine system capabilities, functional, and non-functional features	27 Nov 2022	3 Dec 2022	7	0%	5
2.3	Verify and validate on the requirements	4 Dec 2022	7 Dec 2022	4	0%	3
2.4	Write System Requirement and Design (SRD) Report	8 Dec 2022	25 Dec 2022	18	0%	12
3	**Design**		-			-
3.1	Identify the detailed workflow of each use case	26 Dec 2022	1 Jan 2023	7	0%	5
3.2	Design user interface layout for applications	2 Jan 2023	8 Jan 2023	7	0%	5
3.3	Design the database structure	9 Jan 2023	15 Jan 2023	7	0%	5
3.4	Design the overall structure of the system	16 Jan 2023	22 Jan 2023	7	0%	5
4	**Implementation**		-			-
4.1	Write code for constraint management module	23 Jan 2023	21 Feb 2023	30	0%	22
4.2	Write code for presentation scheduling module	22 Feb 2023	23 Mar 2023	30	0%	22
4.3	Write code for schedule visualization module	24 Mar 2023	22 Apr 2023	30	0%	21
5	**Testing & Integration**		-			-
5.1	Perform unit testing	23 Apr 2023	2 May 2023	10	0%	7
5.2	Perform integration testing	3 May 2023	12 May 2023	10	0%	8
5.3	Perform system testing	13 May 2023	22 May 2023	10	0%	6
6	**Deployment**		-			-
6.1	Deploy the whole system	23 May 2023	1 Jun 2023	10	0%	8

Figure 11.2: Gantt Chart of Week 9 to Week 16



[Praticum Presentation Scheduling] Project Schedule

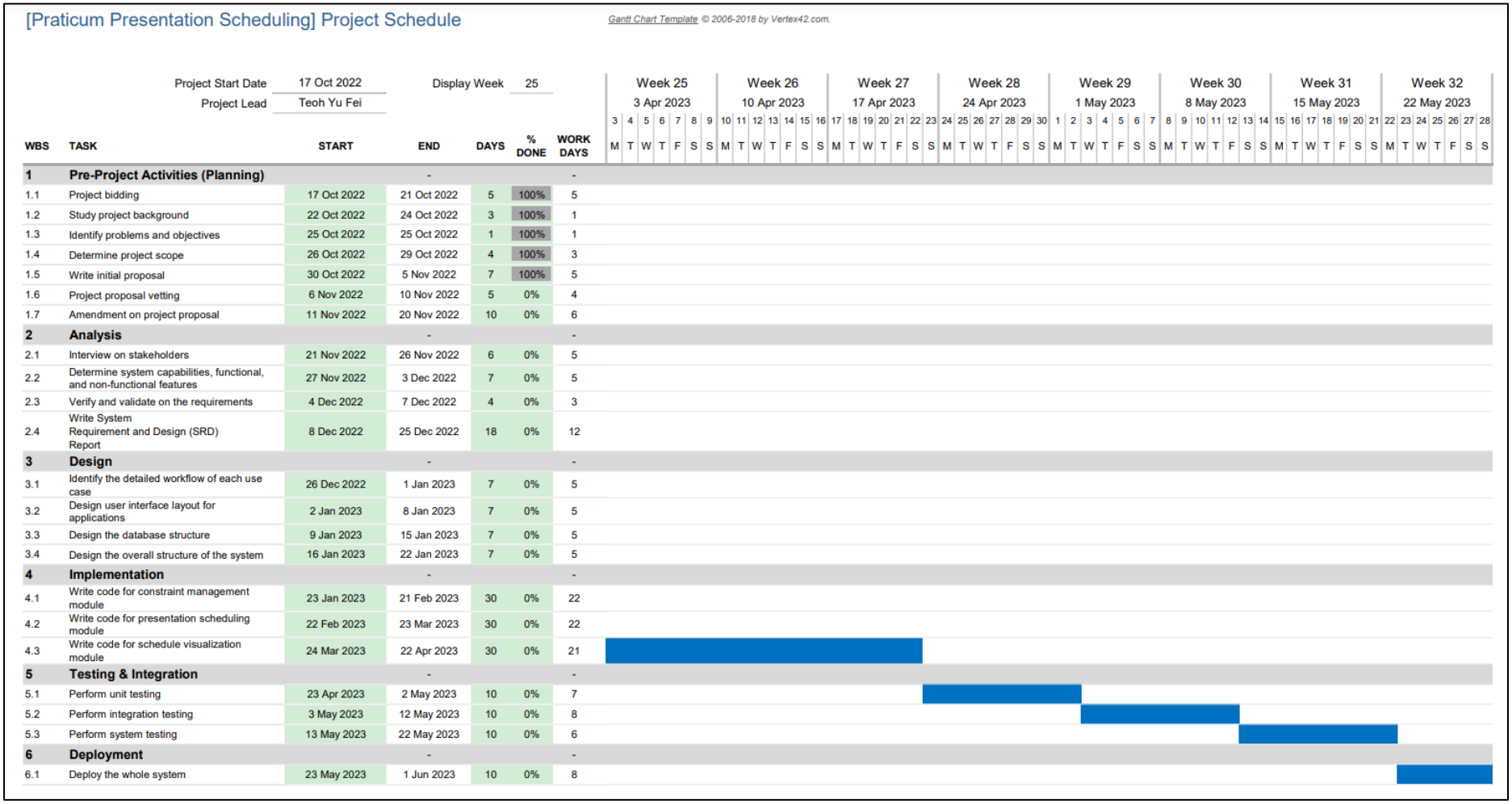
Gantt Chart Template © 2006-2018 by Vertex42.com.

Project Start Date: 17 Oct 2022 | Display Week: 17
Project Lead: Teoh Yu Fei

WBS	TASK	START	END	DAYS	% DONE	WORK DAYS
1	**Pre-Project Activities (Planning)**		-			-
1.1	Project bidding	17 Oct 2022	21 Oct 2022	5	100%	5
1.2	Study project background	22 Oct 2022	24 Oct 2022	3	100%	1
1.3	Identify problems and objectives	25 Oct 2022	25 Oct 2022	1	100%	1
1.4	Determine project scope	26 Oct 2022	29 Oct 2022	4	100%	3
1.5	Write initial proposal	30 Oct 2022	5 Nov 2022	7	100%	5
1.6	Project proposal vetting	6 Nov 2022	10 Nov 2022	5	0%	4
1.7	Amendment on project proposal	11 Nov 2022	20 Nov 2022	10	0%	6
2	**Analysis**		-			-
2.1	Interview on stakeholders	21 Nov 2022	26 Nov 2022	6	0%	5
2.2	Determine system capabilities, functional, and non-functional features	27 Nov 2022	3 Dec 2022	7	0%	5
2.3	Verify and validate on the requirements	4 Dec 2022	7 Dec 2022	4	0%	3
2.4	Write System Requirement and Design (SRD) Report	8 Dec 2022	25 Dec 2022	18	0%	12
3	**Design**		-			-
3.1	Identify the detailed workflow of each use case	26 Dec 2022	1 Jan 2023	7	0%	5
3.2	Design user interface layout for applications	2 Jan 2023	8 Jan 2023	7	0%	5
3.3	Design the database structure	9 Jan 2023	15 Jan 2023	7	0%	5
3.4	Design the overall structure of the system	16 Jan 2023	22 Jan 2023	7	0%	5
4	**Implementation**		-			-
4.1	Write code for constraint management module	23 Jan 2023	21 Feb 2023	30	0%	22
4.2	Write code for presentation scheduling module	22 Feb 2023	23 Mar 2023	30	0%	22
4.3	Write code for schedule visualization module	24 Mar 2023	22 Apr 2023	30	0%	21
5	**Testing & Integration**		-			-
5.1	Perform unit testing	23 Apr 2023	2 May 2023	10	0%	7
5.2	Perform integration testing	3 May 2023	12 May 2023	10	0%	8
5.3	Perform system testing	13 May 2023	22 May 2023	10	0%	6
6	**Deployment**		-			-
6.1	Deploy the whole system	23 May 2023	1 Jun 2023	10	0%	8

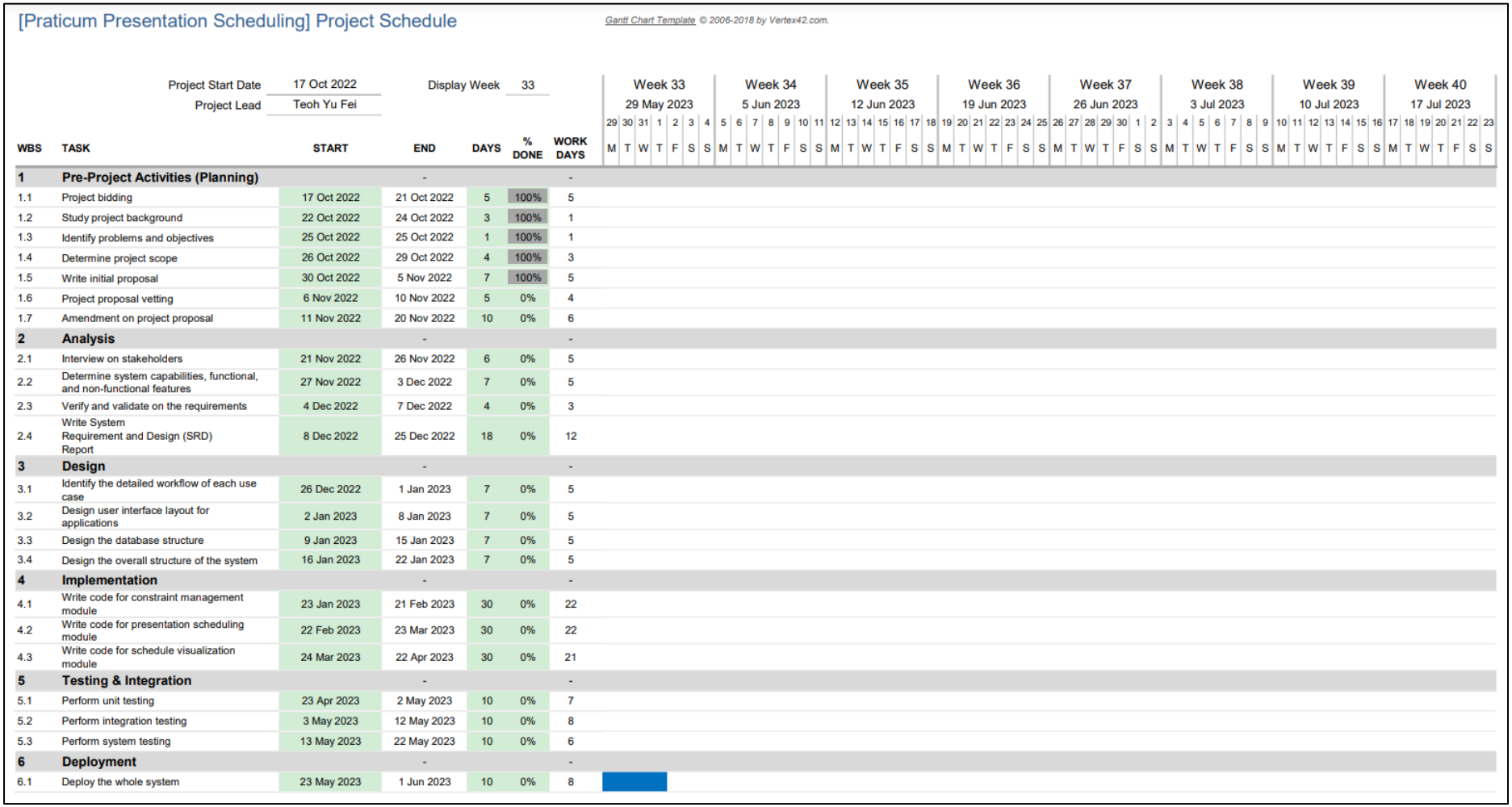
Week 17 – 6 Feb 2023 | Week 18 – 13 Feb 2023 | Week 19 – 20 Feb 2023 | Week 20 – 27 Feb 2023 | Week 21 – 6 Mar 2023 | Week 22 – 13 Mar 2023 | Week 23 – 20 Mar 2023 | Week 24 – 27 Mar 2023

Figure 11.3: Gantt Chart of Week 17 to Week 24



WBS	TASK	START	END	DAYS	% DONE	WORK DAYS
1	**Pre-Project Activities (Planning)**		-			-
1.1	Project bidding	17 Oct 2022	21 Oct 2022	5	100%	5
1.2	Study project background	22 Oct 2022	24 Oct 2022	3	100%	1
1.3	Identify problems and objectives	25 Oct 2022	25 Oct 2022	1	100%	1
1.4	Determine project scope	26 Oct 2022	29 Oct 2022	4	100%	3
1.5	Write initial proposal	30 Oct 2022	5 Nov 2022	7	100%	5
1.6	Project proposal vetting	6 Nov 2022	10 Nov 2022	5	0%	4
1.7	Amendment on project proposal	11 Nov 2022	20 Nov 2022	10	0%	6
2	**Analysis**		-			-
2.1	Interview on stakeholders	21 Nov 2022	26 Nov 2022	6	0%	5
2.2	Determine system capabilities, functional, and non-functional features	27 Nov 2022	3 Dec 2022	7	0%	5
2.3	Verify and validate on the requirements	4 Dec 2022	7 Dec 2022	4	0%	3
2.4	Write System Requirement and Design (SRD) Report	8 Dec 2022	25 Dec 2022	18	0%	12
3	**Design**		-			-
3.1	Identify the detailed workflow of each use case	26 Dec 2022	1 Jan 2023	7	0%	5
3.2	Design user interface layout for applications	2 Jan 2023	8 Jan 2023	7	0%	5
3.3	Design the database structure	9 Jan 2023	15 Jan 2023	7	0%	5
3.4	Design the overall structure of the system	16 Jan 2023	22 Jan 2023	7	0%	5
4	**Implementation**		-			-
4.1	Write code for constraint management module	23 Jan 2023	21 Feb 2023	30	0%	22
4.2	Write code for presentation scheduling module	22 Feb 2023	23 Mar 2023	30	0%	22
4.3	Write code for schedule visualization module	24 Mar 2023	22 Apr 2023	30	0%	21
5	**Testing & Integration**		-			-
5.1	Perform unit testing	23 Apr 2023	2 May 2023	10	0%	7
5.2	Perform integration testing	3 May 2023	12 May 2023	10	0%	8
5.3	Perform system testing	13 May 2023	22 May 2023	10	0%	6
6	**Deployment**		-			-
6.1	Deploy the whole system	23 May 2023	1 Jun 2023	10	0%	8

Figure 11.4: Gantt Chart of Week 25 to Week 32



[Praticum Presentation Scheduling] Project Schedule

Gantt Chart Template © 2006-2018 by Vertex42.com.

Project Start Date: 17 Oct 2022
Project Lead: Teoh Yu Fei
Display Week: 33

WBS	TASK	START	END	DAYS	% DONE	WORK DAYS
1	**Pre-Project Activities (Planning)**		-			-
1.1	Project bidding	17 Oct 2022	21 Oct 2022	5	100%	5
1.2	Study project background	22 Oct 2022	24 Oct 2022	3	100%	1
1.3	Identify problems and objectives	25 Oct 2022	25 Oct 2022	1	100%	1
1.4	Determine project scope	26 Oct 2022	29 Oct 2022	4	100%	3
1.5	Write initial proposal	30 Oct 2022	5 Nov 2022	7	100%	5
1.6	Project proposal vetting	6 Nov 2022	10 Nov 2022	5	0%	4
1.7	Amendment on project proposal	11 Nov 2022	20 Nov 2022	10	0%	6
2	**Analysis**		-			-
2.1	Interview on stakeholders	21 Nov 2022	26 Nov 2022	6	0%	5
2.2	Determine system capabilities, functional, and non-functional features	27 Nov 2022	3 Dec 2022	7	0%	5
2.3	Verify and validate on the requirements	4 Dec 2022	7 Dec 2022	4	0%	3
2.4	Write System Requirement and Design (SRD) Report	8 Dec 2022	25 Dec 2022	18	0%	12
3	**Design**		-			-
3.1	Identify the detailed workflow of each use case	26 Dec 2022	1 Jan 2023	7	0%	5
3.2	Design user interface layout for applications	2 Jan 2023	8 Jan 2023	7	0%	5
3.3	Design the database structure	9 Jan 2023	15 Jan 2023	7	0%	5
3.4	Design the overall structure of the system	16 Jan 2023	22 Jan 2023	7	0%	5
4	**Implementation**		-			-
4.1	Write code for constraint management module	23 Jan 2023	21 Feb 2023	30	0%	22
4.2	Write code for presentation scheduling module	22 Feb 2023	23 Mar 2023	30	0%	22
4.3	Write code for schedule visualization module	24 Mar 2023	22 Apr 2023	30	0%	21
5	**Testing & Integration**		-			-
5.1	Perform unit testing	23 Apr 2023	2 May 2023	10	0%	7
5.2	Perform integration testing	3 May 2023	12 May 2023	10	0%	8
5.3	Perform system testing	13 May 2023	22 May 2023	10	0%	6
6	**Deployment**		-			-
6.1	Deploy the whole system	23 May 2023	1 Jun 2023	10	0%	8

Week 33 29 May 2023 | Week 34 5 Jun 2023 | Week 35 12 Jun 2023 | Week 36 19 Jun 2023 | Week 37 26 Jun 2023 | Week 38 3 Jul 2023 | Week 39 10 Jul 2023 | Week 40 17 Jul 2023

29 30 31 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23

M T W T F S S M T W T F S S M T W T F S S M T W T F S S M T W T F S S M T W T F S S M T W T F S S M T W T F S S M T W T F S S

Figure 11.5: Gantt Chart of Week 33 to Week 40