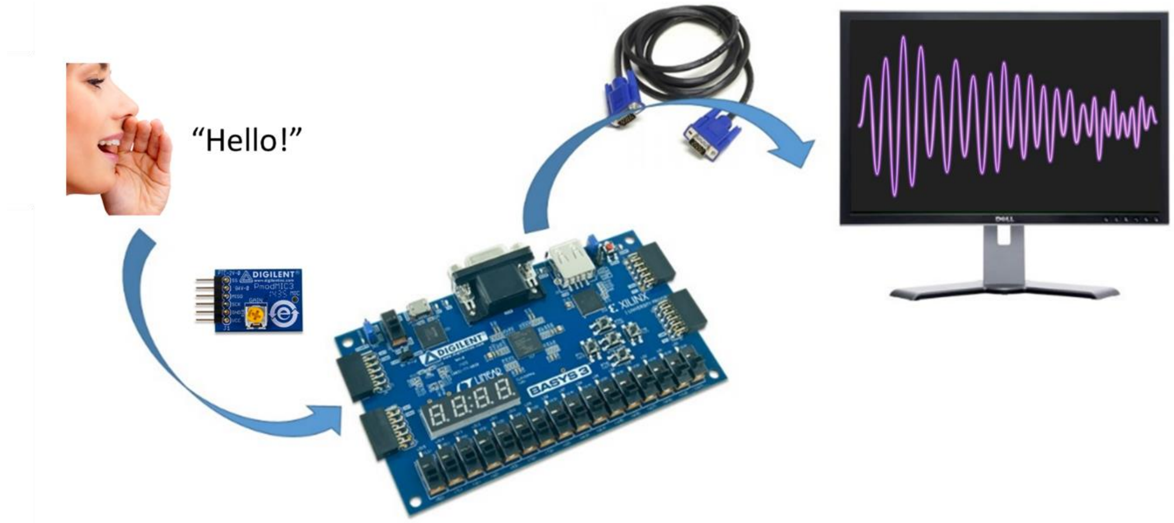
Department of Electrical & Computer Engineering

EE2026 Digital Fundamentals

FPGA Design Project: Real-Time Voice Scope

Abstract

As your EE2026 FPGA Design Project, you will create a real time voice scope (audio visualizer)! You will be provided with a MEMs microphone to capture human voice and display your voice signal on a VGA monitor. This manual introduces you to the various concepts involved, and guides (not walks!) you through getting a voice scope up and running.



Semester 1, AY 2018/2019	
EE2026/ TEE2020 Lab Instructors	DR. CHUA DINGJUAN CHRISTOPHER GAO JIEYI

1. Introduction – System Overview

This Voice Scope project is a project that includes **individual** and **teamwork** (pairwork) components.

You are provided with a MEMs microphone module that captures and digitizes a voice or sound signal. This information is then displayed on a VGA monitor as a waveform. The block diagram of the overall system is as shown in Figure 1.

To make the voice scope more functional and user-friendly, you will implement additional features. Some features are pre-defined in Section 4 of this lab manual, while you will also work on features of your own choosing to make your team's project more functional and interesting.

2. Schedule & Weightage

The schedule and weightages for the various components in the different project weeks are listed below. The requirements of the tasks will be explained in detail in Section 4.
All of the features will be assessed during the project assessment week.

Week	EE2026 Tasks	Percentage
8	Teamwork: Basic voice scope system (Microphone – VGA) (Assessed in week 9)	5%
9	Student A: Real-time audio volume indicator system (Microphone – LEDs) Student B: User-friendly scope grid and background color setting (VGA)	15%+5% individual
10	Teamwork: System Integration – User-friendly voice scope system with an LED volume indicator. Individual/Teamwork: improvement feature(s) (open-ended).	10% teamwork
11	*No Lab. Project Achrive & Report Submission.*	
12	Project Assessment	30%

Table 1. Project schedule and mark percentage breakdown

PLAGIARISM WARNING

This is a teamwork project. Discussions are encouraged, however, 'discussion' is not a valid excuse if your codes turn out to be uncomfortably similar.

- Warning from the NUS Code of Student Conduct:
  "Any student found to have committed or aided and abetted the offence of plagiarism may be subject to disciplinary action"
  http://www.nus.edu.sg/registrar/adminpolicy/acceptance.html
- Both the source and recipient of the project solutions (codes) or reports will be **unconditionally penalised** in such cases. Marks will also be adjusted and withheld from release if the codes are similar beyond a certain empirical threshold.
- Students will also be reported to BoD (Board of Discipline).
- Please always note that NUS and the EE2026 teaching team take plagiarism very seriously.

3. Design Files and Resources

The following design files accompany this project manual, and are available on IVLE. Open the archived Vivado project to get started.

Hardware Sources

Each team member can check out 1 piece of **PmodMIC3** and 1 **VGA cable** from the Digital Electronics Lab. You would already have your assigned 1 x **Basys 3 Development Board**.

Design Sources

Voice_Scope_TOP.v: the top-module of the design. This will be your main module, or typically called the Top Level module, where you instantiate the sub-modules and make the necessary links between these modules.

VOICE_CAPTURER.v: an interface module between the microphone and your design. This module works with the PmodMIC3 module / peripheral board to convert the analog sound signal into a digital 12-bit parallel *MIC_in* sample data. This conversion needs a sampling clock and a serial clock. *You are NOT supposed to make changes in this module.*

Draw_Waveform.v: A waveform drawing module defines the waveform that is to be shown on the VGA monitor. The waveform signal is displayed by configuring the colours (red, green and blue, 4 bits respectively) of each pixel.

Draw_Background.v: A background drawing module (Draw_Background.v) controls the background of the screen. This allows you to draw grids and ticks as the monitor background to assist users read the amplitudes of the waveforms. In the provided code, only one horizontal and one vertical line is predefined. You may make modification to create your desired grid lines. The monitor background colour can also be defined in this module.

VGA_DISPLAY.v: A VGA display module displays pre-defined pixels on the VGA monitor. The display reflects the waveform and background defined in **Draw_Waveform.v** and **Draw_Background.v**. *You are NOT supposed to make changes in this module.*

Constraint Sources

Basys3_Master.xdc: a master constraints file that defines the I/O constraints for BASYS 3.

Project Wiki

tiny.cc/ee2026wiki : The project wiki includes a troubleshooting guide, vhdl vs verilog code comparison and useful resources when you are working on additional features.

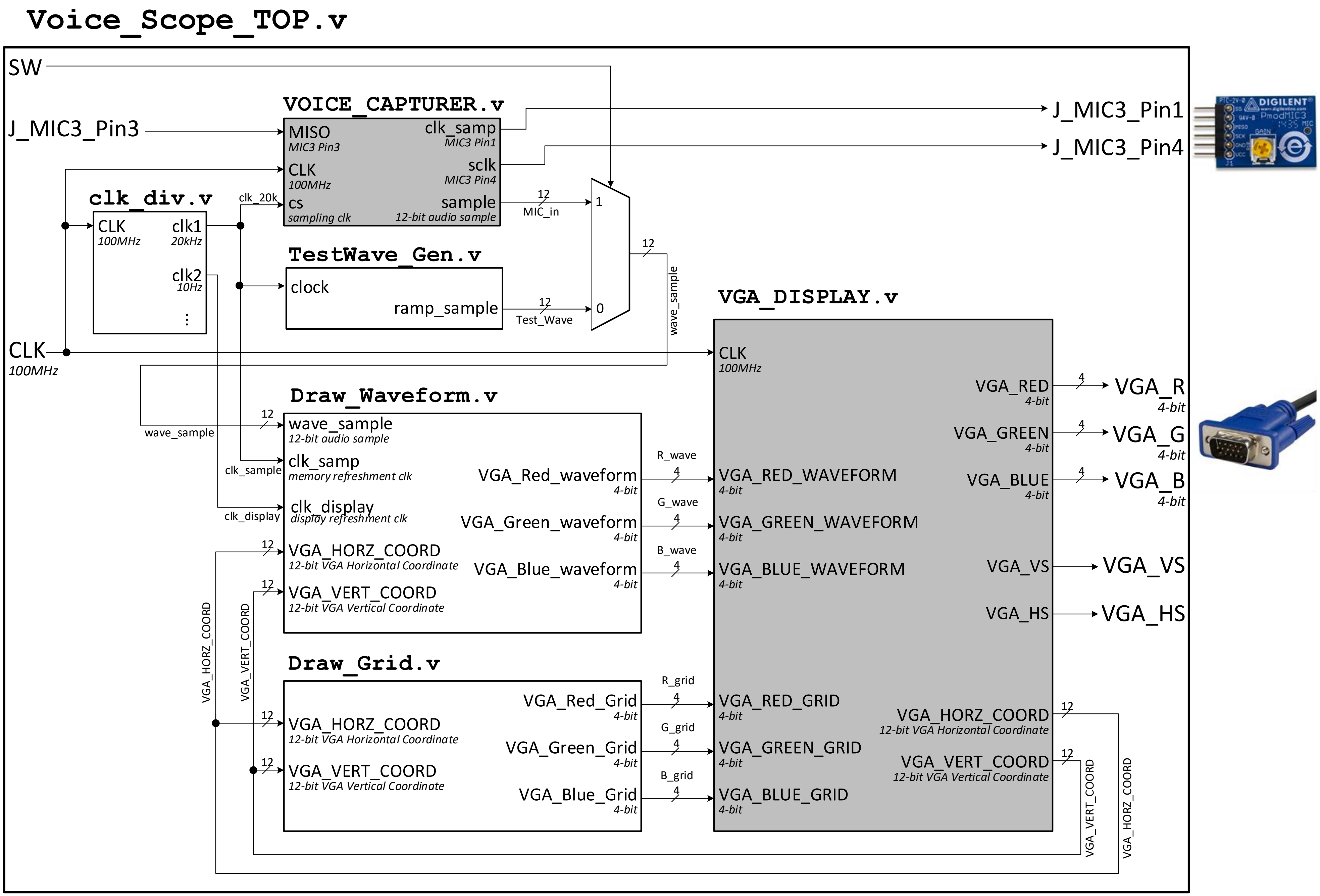


Figure 1. System block diagram of a basic voice scope

4. Tasks & Requirements

4.1 Basic Voice Scope System

PROJECT TASK 1.
Basic Voice Scope System

In this session, the objective is to setup the basic voice scope system using the PmodMIC3 and a VGA monitor with Basys3. Your system will capture and digitize the audio signal input from the microphone on the PmodMIC3. The VGA monitor will display the audio signal as points which represent the audio waveform.

Student A : : Set up Microphone and Generate VGA Test Waveform(s)

1) **Create a `clk_div.v` module which generates a 20kHz clock. Name it `clk_20k`.**
   The (exactly) 20kHz clock is needed as a sample clock for capturing the audio signal. For example, every 1s audio signal will be recorded by 20,000 discrete samples.

2) **Create a `TestWave_Gen.v` module which generates a 10-bit ramp wave.**
   This ramp wave acts as a test signal for the VGA display. You would be able to observe it on the VGA display after combining with your partner's tasks.

3) **Instantiate below modules in `Voice_Scope_TOP.v` according to Figure 1.**
   `clk_div.v`
   `VOICE_CAPTURER.v`

4) **Display `MIC_in` on LEDs.**
   This allows you to visualize the microphone input data. This is for testing microphone purposes only.

5) **Attach the PmodMIC3 with FPGA. Edit the constraints file.**
   IMPT *To avoid damaging the boards*, make sure the GND and VCC pins on the Pmod and Basys3 are connected correspondingly.

6) **Generate bitstream and verify that the LEDs light up.**

Student B : : Set up VGA and Display the Test Waveform(s) on Monitor

1) **Instantiate below modules in `Voice_Scope_TOP.v` according to Figure 1.**
   `TestWave_Gen.v`
   `Draw_Waveform.v`
   `Draw_Background.v`
   `VGA_DISPLAY.v`
   You do not have to modify the given codes in these modules at this stage.
   Your partner will create **`TestWave_Gen.v`** for you. You can take the ramp signal as your `wave_sample`.

2) **Attach the VGA monitor to the FPGA using a VGA cable. Edit the constraints file.**

3) **Generate bitstream and verify that the VGA monitor displays a ramp signal one red horizontal and one red vertical line.**

Teamwork : : System Integration

1) **Combine the codes from both members.**

2) **Code the logic of wave_sample to let it choose between the microphone input and a pre-defined ramp wave.**

   Hint:

   The 12-bit sample values of MIC_in may be too large to fit in the monitor, since the vertical size is 1024 (10-bit).

3) **Generate your bitstream and upload it to FPGA.**

   When displaying from a microphone mode, you can output a sinusoidal tone from your handphone/earpiece and put it near to the PmodMIC3.

   a) Observe that either a sound waveform or ramp signal is displayed on the VGA display.

   b) Observe one RED horizontal and one RED vertical line showing on the VGA monitor. The screen background color should be BLACK.

4) **Demonstrate the above to the instructors in Week 9.**

4.2.A. Volume Indicator

PROJECT TASK 2.A.
Volume Indicator

An audio volume indicator displays a representation of the intensity in audio equipment. It helps people to observe the changes in the audio signal visually. An example is as shown in Figure 2.

In this task, we use the LED array on Basys 3 as an audio volume indicator (from a vertical view), displaying a digital value which reflects the intensity of the input audio sample.

Use LED array on FPGA to display the peak intensity of the audio signal (`MIC_in`).
The LED array displays intensity linearly and refreshes at an observable rate.

Hint:

1) The peak volume mapped to the LEDs must remain for a reasonable period of time to be observable by human eyes. You may need to record the peak value of a set of sampled data within this period of time.

2) Linear display means the LEDs should light up as a bar, instead of individual bits. That is, once the volume reaches a certain level, all LEDs below the level should light up.
Hint : The volume gaps between two adjacent two LEDs do not have to be equal. Think about practical considerations to make the display user-friendly.

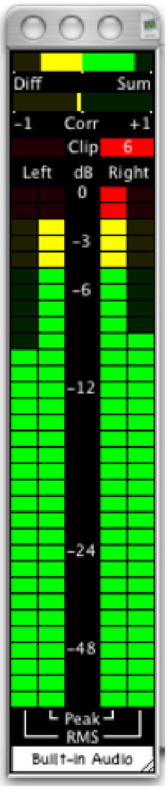


Figure 2. An example of an audio volume indicator

4.2.B. Drawing Grid Lines & Setting Pixel Color on VGA

PROJECT TASK 2.B.
Basic Scope Displays on VGA

To improve the user-friendliness of the voice scope in terms of reading the waveform amplitudes, axis, ticks, grid lines, and the screen background color all play a role. You may refer to an example of an oscilloscope as shown in Figure 3.

In this task, you need to draw 16x16 grids, x- and y-axis, ticks on the two axis, and configure the colors of the lines and the background to make it user-friendly.

Hint:

1) Work on modifying `Draw_Background.v` to achieve the above.

2) You may employ Verilog's modulus operator % to draw grid lines.
   - Every 80-th pixel along the horizontal axis
   - Every 64-th pixel along the vertical axis.

3) Ticks are short lines on the x and y axes to show the horizontal and vertical unit divisions. You will need to insert the logic in Line #38.
   Modification of Lines #43 – 45 may be needed to fit in the ticks.

4) The colors of lines and background must be different from the default red color, and must be user-friendly.

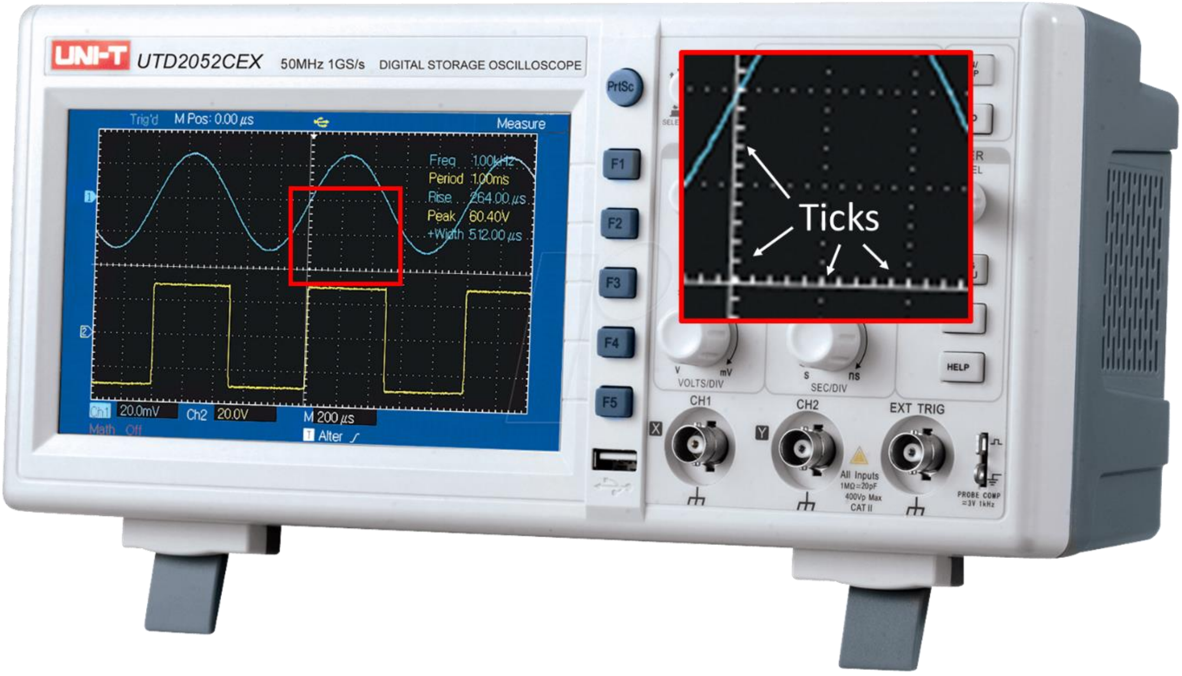


Figure 3. An example of an oscilloscope

4.3. System Integration

Working together with your partner, integrate the TASK 1, TASK 2.A. and TASK 2.B. to display the audio signal on the VGA. Ensure that your voice scope is easy to use and user-friendly.

4.4. Improvement Feature(s)

As part of the project requirements, you are required to implement improvement feature(s) to add value to your voice scope (audio visualizer) and distinguish your product from the rest. You are allowed to make use of the other peripherals available on the Basys 3 to make the project – your own product – more interesting/functional/user-friendly.

This additional feature is open-ended and your team will be evaluated on the metrics of quality, complexity, functionality, creativity, functionality and uniqueness.

5. Project Submission

Each team is required to submit two items (ITEM A and ITEM B) as indicated below:

5.1 ITEM A: PROJECT ARCHIVE SUBMISSION

- Only ONE Vivado project archive (.zip) for your team. The team marks are not awarded if all the features from the team are not combined into one bitstream
- Ensure that your bitstream has been successfully generated and tested on your Basys 3 development board BEFORE archiving your Vivado workspace
- Name your project archive in the format indicated below to avoid losing marks:
  Official lab day_Name of any one team member as indicated on the matriculation card_Matriculation number 1_Matriculation number 2_Archive
  Example: Monday_Claude_Shannon_A0300416Z_A0131086Y_**Archive**.xpr.zip

5.2 ITEM B: REPORT SUBMISSION

- Include the following in your report:
  - ➢ Name, matriculation number, official lab session (Monday | Tuesday | Wednesday | Thursday | Friday)
  - ➢ **User guide** which consists of:
    - ▪ Features that you have designed and implemented
    - ▪ Instructions on how these features can be operated by the user, either through **flowcharts** or by using a **table** format as indicated below:

No.	**Feature**	**Owner**	**FPGA Input used**	**Feature Description**	**Display (Figures)**
1	Basic Voice Scope	Team	SW0	When SW0 = 0, the VGA displays a 50Hz ramp wave; When SW0 = 1, the VGA displays the voice waveform.	Figure 1

2.A.	Volume Indicator	Jieyi	N.A.	{LED11, …., LED0} ……	
2.B.	VGA	Christopher	N.A.	… grids … … ticks …	
3	Improvement 1	Jieyi	SW2 …	…	Figure 2
	Improvement 2	Christopher	SW3 …	…	Figure 3
	Improvement 3	Team	SW4 …	…	Figure 4

Table 2. Example of Project Summary

    - Describe the key features that have been designed and implemented, while appending code segments to aid in your description if necessary
    - Clearly indicate the **lead designer(s) of each feature**. You are not allowed to indicate both students of the team
  - Include all images in the Appendix in colour.
  - Describe the key features that have been designed and implemented, while appending code segments to aid in your description if necessary
  - Feedbacks: Please include your experience on what made the project more doable and enjoyable, and which parts caused unnecessary pressure during your learning journey. From the perspective of a fresh student in this research-focused university, let us know if the practical parts of the module have an appropriate balance between spoon-feeding / independence when it comes to learning the fundamentals given a short-time frame of 10 weeks
    - Note that feedbacks, whether positive or negative, DO NOT have any effects on your grades ☺
  - **References**: Include references to sources that you have obtained template codes / ideas from

- Name your report submission in the format indicated below to avoid losing marks (The one team member name MUST be the same as that indicated for the Project Archive):

  Official lab day_Name of any one team member as indicated on the matriculation card_Matriculation number 1_Matriculation number 2_Report

  Example: Monday_Claude_Shannon_A0300416Z_A0131086Y_**Report**.pdf

5.3 SUBMISSION DEADLINES:

- ITEM A and ITEM B should be submitted to IVLE in the correct folders by the deadlines as stated in the table below:

Official Session	ITEM A: Project Archive Submission	ITEM B: Report Submission
Monday	Week 11 Saturday 11:59 P.M.	Week 11 Sunday 11:59 P.M.
Wednesday		

Table 3. Project archive and report submission deadlines

- Excuses about technical issues during upload will not be entertained, hence you are required to avoid uploading at the last minute
- Penalties apply for late submissions