

DESN2000 (Computer Engineering) 2024 T2: Assessment Guide for Project

Last edited: 20/07/2024

Project Documentation (10% of the total grade)

There will be two submissions, namely, the project user guide and the project developer guide. Per each submission, only one member of the group should do the submission. There will also be a form that every one of you must fill about the contribution made by each member of the group towards the documentation. Demonstrators will mark the submissions and the individual marks will be scaled based on the information in contribution forms. A plagiarism check will be conducted through Turnitin and may result in a mark of 0 in case plagiarism is detected.

Deadline for the submissions and filling out the contribution form: 11:59 PM, Friday (Week 10)

$$\textit{individual mark} = \textit{contribution_factor} * (\textit{user_guide_mark} + \textit{developer_guide_mark}),$$

where,

$$\textit{contribution_factor} = (\textit{average percentage assigned to the individual}) * (\textit{no. of group members in the group}) / 100$$

See below for the marking table of *user_guide_mark* and *developer_guide_mark*.

Project User Guide

The user guide will be marked from the perspective of a user with limited technical knowledge of embedded systems. Imagine the user is a lab technician. What we are looking for in a good quality user guide are the following:

- **Completeness.** All subsystem relevant to the user is clearly explained.
- **Figures.** Great sketches/figures that help illustrate the working of your product
- **Accessibility.** It is deemed that a layman can pick up the user manual, and read and understand everything without the need for technical knowledge.

Marks will be awarded as per the table below:

0	None submitted or nearly empty submission
1	The user guide is only partially complete and the user is likely to get lost in most of the scenarios
2	The user guide is only half done and the user is likely to get lost half of the time
3	Quality is good, however, requires considerable improvements to reach the quality of a near-perfect user manual
4	Quality is good and only minor improvements are sufficient to reach the quality of a near-perfect user manual.

5	Quality is close to that of a user manual of a commercial product. The user manual is outstanding, well-structured, complete and clear. Figures have been used for better illustrations where applicable.
---	---

Project Developer Guide:

The user guide will be marked from the perspective of a future developer who has taken over the project after you completed the prototype development.

What are we looking for in a good quality developer guide is as below:

- **Completeness.** Every functionality in the system is covered, and what design approach is agreed upon when developing your system. It covers all subsystems and outlines how they integrate.
- **Figures.** Use of appropriate figures to demonstrate the working of the systems and subsystems
- **Conciseness.** It dives deep enough in the technical aspect/coding side just enough to cover the important part, but not too deep to the detriment of the reader (developers are human as well after all)

Marks will be awarded as per the table below:

0	None submitted or nearly empty submission
1	Some effort has been made, however, figuring out the code is not made much easier with this developer guide
2	Half done and half of the details are missing. Would require a substantial rewrite of the manual to be deemed more useful.
3	Good quality, however, there are a few ambiguous places that may require setting up a meeting with the developers to figure out.
4	Very high quality. Only some minor unclear places and a short email conversation with the developers will be adequate to clarify any ambiguities.
5	Outstanding quality. Understood everything about the implementation and no need to contact the developers for any clarification.

Project Implementation (30% of the total grade)

There will be one submission for this component and only one member of the group should do the submission. There will also be a form that every one of you must fill about the contribution made by each member of the group. This is a separate form to the contribution for project documentation. The submitted source code will be checked for plagiarism and will attract a mark of 0 in the case where plagiarism is detected. The implementation will be marked in a viva where each group will demonstrate their work to a tutor and will answer the questions asked by the tutors that checks for your understanding. Note that if you fail to answer a question, you will not get a full mark even

if that component worked. If tutors determine that the work is not done by you (lack of understanding) you may even get a 0.

Deadline for the submissions and filling contribution form: 11:59 PM, Friday (Week 10)

Viva: 0.5 hours per group which you can select from 2.00 pm-4.00 pm on Wednesday (week 11) and 1:00 pm-3:00 pm on Thursday (Week 11). We will release a booking form and the booking times will be first come first served basis.

The marks for the group will be assigned as follows:

Laboratory timer (~30%)

1. Configuration

- a. During the configuration, the user should be able to give a number that indicates how many timers they need (1 to 4, check for nonvalid inputs). **3 marks**
- b. The user should then be able to set the timeout for each of those tasks while giving a short label for each task (maximum time for a task can be assumed to be 60 minutes and the minimum 30 seconds). **3 marks**
- c. The user should be able to change a saved value later if they wish. **3 marks**
- d. The saved configuration must be stored in memory as long as the power to the device is provided (assume the power supply will be replaced using a battery at the production stage – you don't have to worry about this). **3 marks**

2. Running

- a. Each task should have a designated button to start the relevant timer. **3 marks**
- b. All the timers should be displayed on the LCD screen appropriately in a countdown style. **3 marks**
- c. When each timer reaches its end, notify using a sound. **3 marks**
- d. Ensure each timer has a unique notification (e.g., different sounds) to distinguish it from others. **3 marks**
- e. User should be able to start tasks in parallel (ignore any little delays by the user to press buttons) or start them sequentially in any order. **3 marks**
- f. Once a certain task has finished, the user should be able to start the same task again later. **3 marks**

Standard clock (~25%)

1. Time of the day clock

- a. Today's date and time are displayed on the LCD in a format like 19/06/2024 Wed 12:01:00 am. **3 marks**

- b. A method to set this date and the time. **3 marks**
- 2. One alarm
 - a. User should be able to set an alarm for a given time. **2 marks**
 - b. Should be able to distinguish between a.m. and p.m. No need to worry about the dates for the alarms, let them ring every day. **2 marks**
 - c. There should be a way to enable/disable the alarm. **2 marks**
 - d. There should be a way to snooze and stop the alarm. **2 marks**
- 3. One countdown timer
 - a. User should be able to set the countdown time in a format like hh:mm:ss. **2 marks**
 - b. The countdown should be displayed on the LCD. **2 marks**
 - c. Once the countdown is reached, a notification should be played (both auditory and visual). **2 marks**
- 4. One stopwatch
 - a. User should be able to reset, start, pause and stop the stopwatch. **3 marks**
 - b. The stopwatch should run in a format like hh:mm:ss. **2 marks**

Switching between the two modes (~15%)

- 1. Make sure there is a way to switch between the two modes. **5 marks**
- 2. Switching the modes should not clear any running timers/alarms/countdowns/stopwatches and instead should run in the background. **5 marks**
- 3. The user should be able to clearly distinguish in which mode the clock is currently in. **5 marks**

Distinguishable Features (~30%)

At least three distinguishable features are required to make your implementation user-friendly and attractive. **Each distinguishable feature will get a mark to a maximum of 10. The sum of marks for the distinguishable features will be capped at 30.**

The sum of the above marks will be then scaled based on the contribution to calculate the individual mark as follows.

$$\text{Individual mark} = \text{contrib}_x * \text{contrib}_y * (\text{sum of the marks above}) * 30 / 100$$

contrib_x:

Each individual must bring their own board and laptop with the cubelDE project available. You should demonstrate to the tutor how you would flash the board in a maximum of 5 minutes. The tutor will assess your familiarity based on how you perform this and will assign a value between 0.8 to 1.0. Don't worry, technical issues beyond your control will not be penalised.

contrib_y:

This is based on the individual contribution form.

*contribution_factor = (average percentage assigned to the individual)
* (no. of group members in the group) / 100*