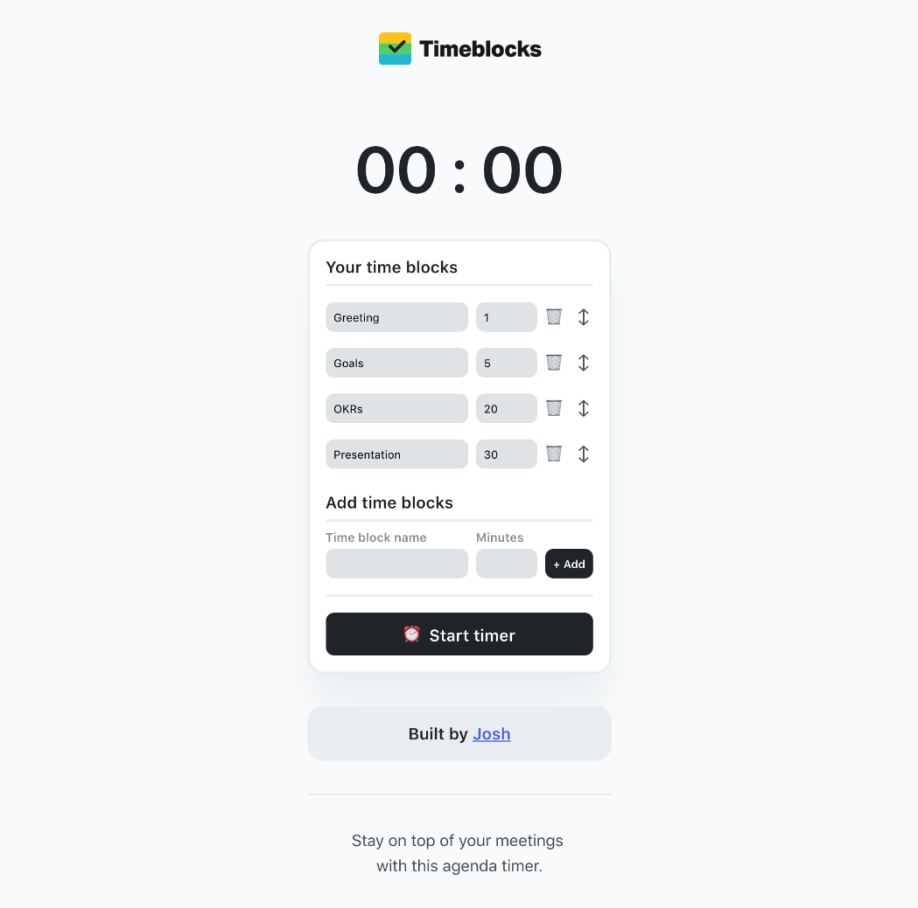
Assignment 1 – Project

v1.0

Objective:

Effective time management are essential for business and for study. It can be used as a tool to support software development, such as managing team meetings. In this assignment, you're responsible for the development of a time management tool. You'll practice engineering workflow and tool chains for software development.

In this assignment, you'll develop a countdown timer designed for team meetings. An example of the countdown timer can be found at:



https://www.timeblocks.co/

The basic feature of your countdown timer should include:

1. Allowing users to specify sub-tasks and time.
2. Allowing users to export and import sub-task and time configuration as a CSV file.
3. Graphical or browser-based user interface design.
4. Every member is required to develop one feature to enhance countdown timer, designed for a team meeting scenario.

The development of **feature 1** should remain in the **master branch**. Other features should be **firstly developed in a separate branch**, and then merge to the master once the feature development is complete.

Upon completing each of the **feature 1, 2, and 3** listed above, you should create **tags v1.0, v2.0, and v3.0**, respectively.

In this assignment, you can discuss with your group can choose to program in either java, javascript, or python. As this course focuses on tools for software development, please select the programming language that you're already familiar with. For simple countdown timer in these programming languages, you can take a look at the following links to get start with (note: these samples are not the solution, just to give you some ideas to start with):

Python	https://www.youtube.com/watch?v=KZkQFab3NnA
Java	https://examples.javacodegeeks.com/desktop-java/awt/event/a-simple-timer-example/
Javascript	https://www.w3schools.com/howto/howto_js_countdown.asp

For user-interface, you can choose any GUI framework or browser-based interface. As a reminder, this course focuses on tools for software development, and we'll not teach coding. You should choose a programming language that you're familiar with, or you are confidence that you can be self-taught or learn with your group to complete the assignment.

Deliverables

(unless otherwise specified, the deadline is by the end of the week (Sunday 11:59pm)

Group formation (week 2) and README.md checkpoint (Week 4)

- Students are given an opportunity to self-nominate groups with **3 members** before **the end of week 2**. Note, this is **optional**, and you can leave for random allocation in week 3. If you opt for forming your self-nominated group, the self-nomination process is through a wiki page on the course website. You should closely follow the instruction on the wiki page which will be open in week 2. Fail to follow the instruction will result in losing the 5% checkpoint mark.
- Students without self-nomination after the deadline will be randomly put in groups or allocated to existing groups. If your self-nominated group has 2 members, you might get an additional member assigned.
- Group members are required to be in the **same practical session** (note: all external students are considered under the external-practical session.)
- If your group member drops out from the course, for example do not respond to your emails, please keep your email log and proceed without the missing member.
- Note: the group formation is part of the assessment. If you fail any of the above requirement, you will lose the group formation mark.

by the end of week 4. You're required to:

- Create a **private** team repository on **gitlab.com**;

- Each team members should create **one** account to access the project repository (you will lose mark if you have more than one account). Your account name should be the **same as your UniSA network username**.
- In the **REAEDME.md** file, put all member's **name, user ID, and student ID**. Commit the change into the repository.
- You should make a first commit towards your project.

Project submission (mid study period break, 25 Apr 2021)

- Your code should be completed by mid-SP break for submission.
- You should prepare a user manual written in markdown, with filename **manual.md**, saved under the **doc/** sub-directory in your code repository.
- You should commit your changes and push to your work to your group repository on gitlab on a regular basis. At mid-SP break, you should clone the entire repository and save it as a ZIP file and submit to the course website.
- Every member is required to submit the ZIP file (and they should be the same for the group). **No mark if you do not submit the ZIP file.**

Final report (mid study period break, 25 Apr 2021)

No mark will be given for a joint report. A report up-to 20 pages in length, including the following sections:

1. Project overview

   Give a brief summary of your project.
2. Individual contribution

   Provide details of your contribution in Features 1, 2, and 3 in this project.

   Provide details of the new feature you're responsible for.

   Describe how you collaborate with other group members.
3. Software workflow process and software development environment

   Describe the workflow process you practice in this assignment.

   Provide details of the tools you used in this project: for instance, addons that has been configured if you use an IDE, or scripts used if you use command line interface. Explain the benefit of your development environment.

Other guidelines

- Penalties apply for reports exceeding the maximum page limit.
- There is **no minimum length**. Lengthy report may not necessarily better.
- You can optionally include appendix which does not count towards the page limits.
- Please get in touch with your group members early and start your project early.

The report will be submitted through the course website.

Individual presentation (Week 10)

- **Everyone** is required to present their individual task. Up to **3 minutes per person**.
- Internal students are required to attend the **practical class** for individual presentation.
- External Students are required to submit **voice-record slide presentation** in an **mp4 video file**. You can use Microsoft PowerPoint to record voice and export -> create a video. You can also use software other than Microsoft PowerPoint.

- This is an **Individual Presentation.** You should include the common introduction of your group project in your 3 minutes, to demonstrate your understanding of the overall project, instead of only one group member deliver a project overview. (i.e. you can repeat the project introduction slides among the group members.)
- The presentation should be individual (i.e. multiple 3-minute presentations). **No mark will be given for a joint presentation.**

Extension Policy

To ensure fairness, extension will only be considered with a

All students seeking extensions are required to:

- Provide supporting evidence for the extension (e.g. medical certificate, access plan, etc.)
- Submit your most up-to-date report as a supporting document
- Late submission penalty: 20% per day (including the weekend)

Marking Scheme

Assessment Components	Weight
Checkpoint	5%
Software design and implementation	20%
Version control	20%
Bug tracking	10%
User manual	10%
Individual presentation	10%
Individual report	25%