How to survive a group project

In COMP1531 and beyond 🚀

COMP1531 – Software Engineering Fundamentals

About Oliver

- Recently graduated with a Bachelor’s in Mech Eng
- Took 13 group-based courses as part of my degree
- Taught 7 group-based courses
- Department Lead at Sunswift Racing for 2 years
- I’ve seen and made all the mistakes
- Here to give you general teamwork principles

About Rani

- Experienced COMP1531 tutor (3 years 👵🏼)
- Interned at range of companies: Deloitte, Westpac, Google
- Involved in many student-led group-based activities
  - CSESoc Careers Director
  - Started my own society w/ friends
  - BITSA, WIT, UMCG, …
- Experienced slacker and high-performing leader
- 💖 Unironically love group work 💖

And yes, my friends have called me insane for this opinion.

Why teamwork?

Why teamwork?

- One of THE most important skills
- Make success better than “random chance”

Our Goals

- Give advice to our younger self
- Learn from our mistakes
- Give you strategies that you will ACTUALLY want to use
- Apply these strategies in COMP1531

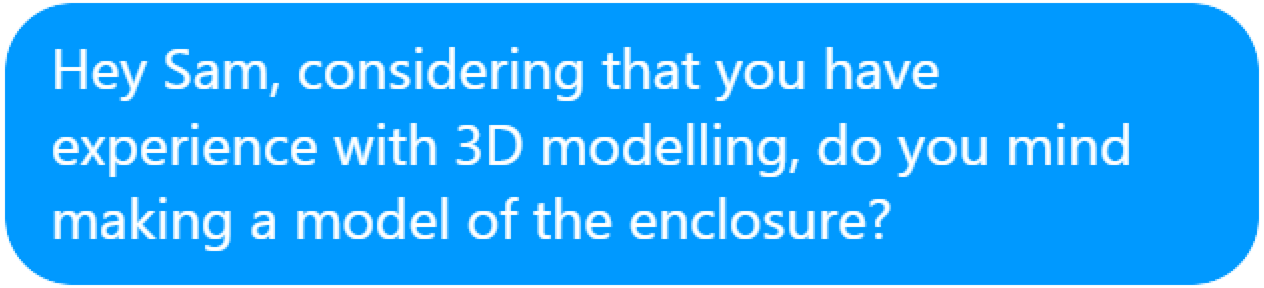
Disclaimer: these strategies are NOT compulsory.

We will cover:

- How to make sure people get their work done on time
- How to make sure tasks are done to a high standard
- How to be a better contributor
- How to distribute tasks fairly
- How to prevent arguments
- How to make decisions as a group
- How to run a meeting that isn’t a complete waste of time

How to make sure people get their work done on time

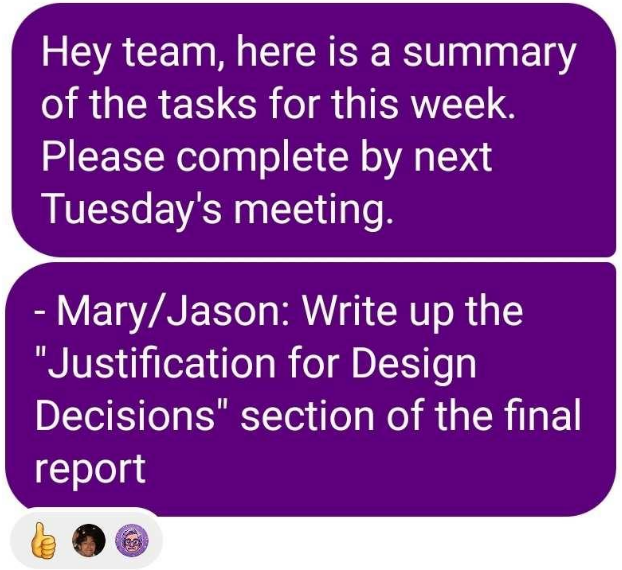
1. **Always set deadlines**



Hey Sam, considering that you have experience with 3D modelling, do you mind making a model of the enclosure?

How to make sure people get their work done on time

1. Always set deadlines
2. **Avoid the social diffusion effect**

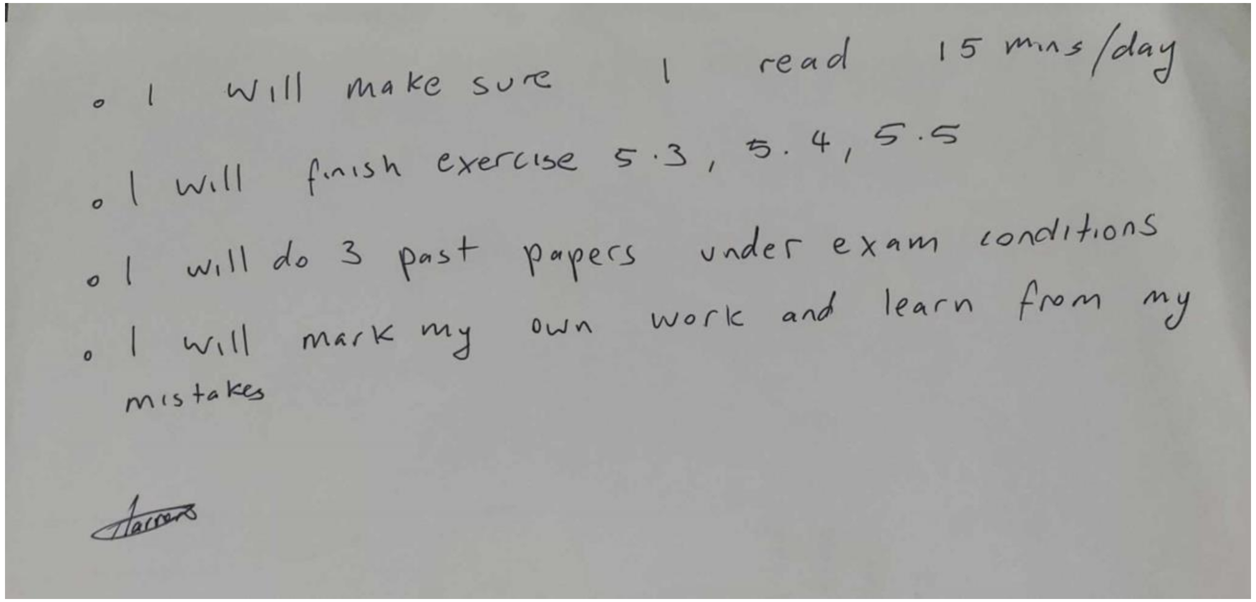


Hey team, here is a summary of the tasks for this week. Please complete by next Tuesday's meeting.

- Mary/Jason: Write up the "Justification for Design Decisions" section of the final report

How to make sure people get their work done on time

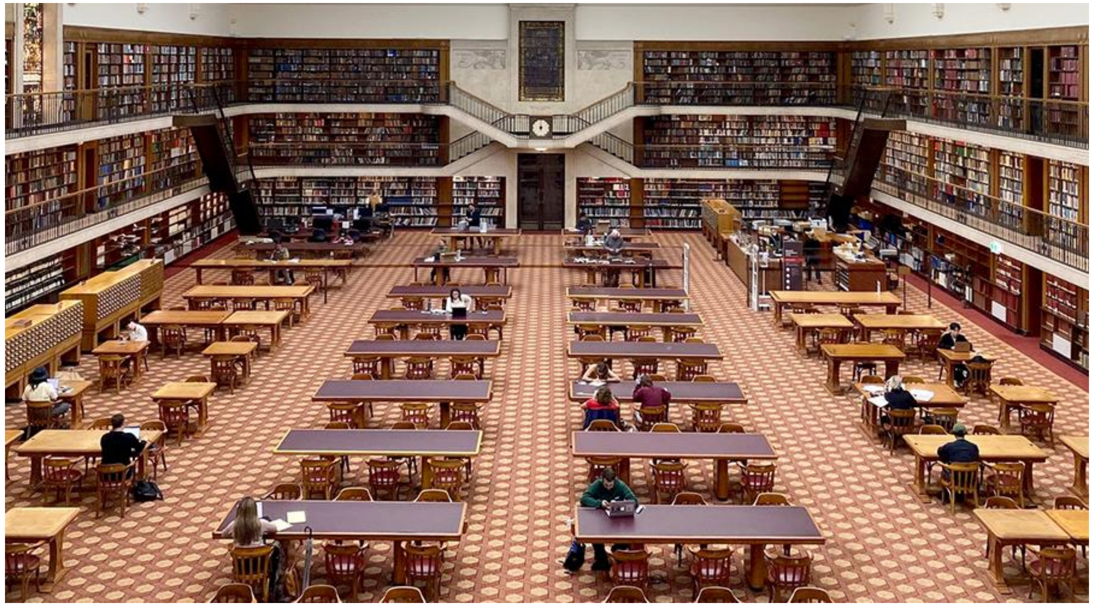
1. Always set deadlines
2. Avoid the social diffusion effect
3. **Exploit commitment bias**



- I will make sure I read 15 mins/day
- I will finish exercise 5.3, 5.4, 5.5
- I will do 3 past papers under exam conditions
- I will mark my own work and learn from my mistakes

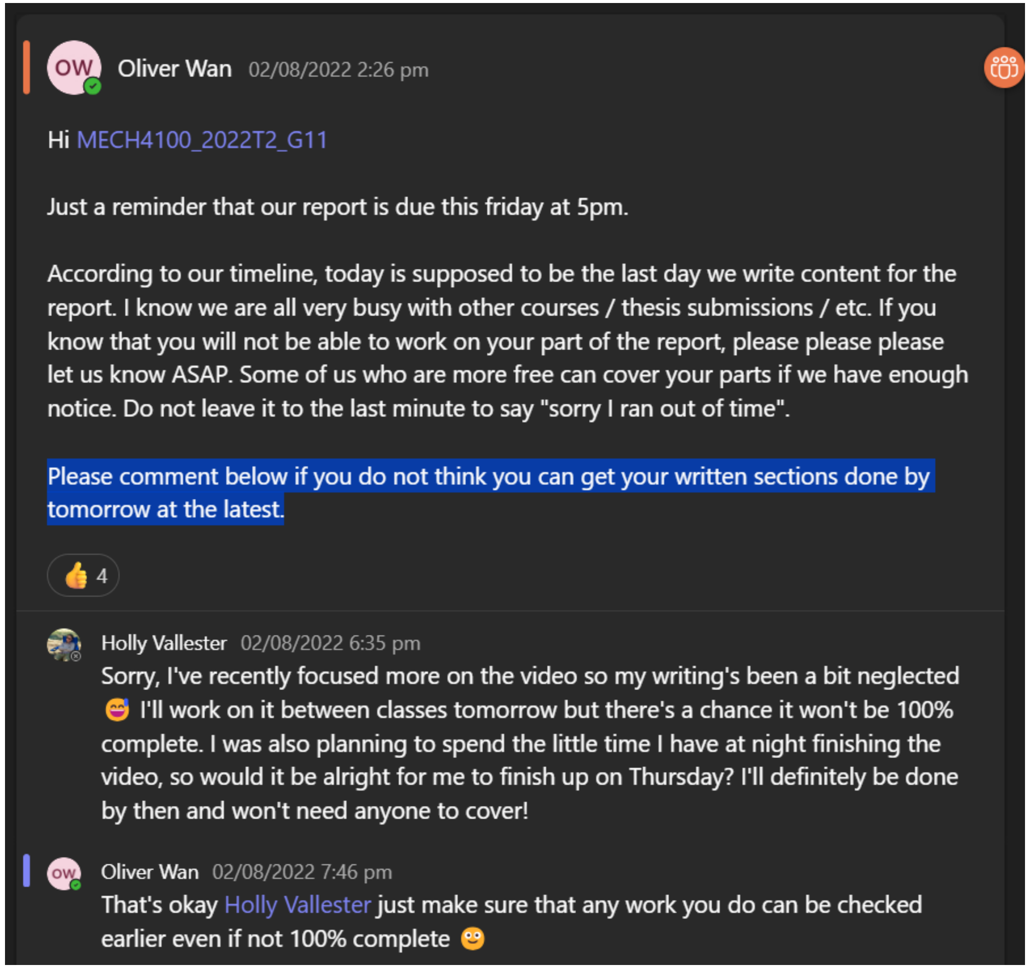
How to make sure people get their work done on time

1. Always set deadlines
2. Avoid the social diffusion effect
3. Exploit commitment bias
4. **Exploit the social facilitation effect**



How to make sure people get their work done on time

1. Always set deadlines
2. Avoid the social diffusion effect
3. Exploit commitment bias
4. Exploit the social facilitation effect
5. **Check in regularly and send reminders**



Oliver Wan 02/08/2022 2:26 pm

Hi MECH4100_2022T2_G11

Just a reminder that our report is due this friday at 5pm.

According to our timeline, today is supposed to be the last day we write content for the report. I know we are all very busy with other courses / thesis submissions / etc. If you know that you will not be able to work on your part of the report, please please please let us know ASAP. Some of us who are more free can cover your parts if we have enough notice. Do not leave it to the last minute to say "sorry I ran out of time".

Please comment below if you do not think you can get your written sections done by tomorrow at the latest.

👍 4

Holly Vallester 02/08/2022 6:35 pm

Sorry, I've recently focused more on the video so my writing's been a bit neglected 😅 I'll work on it between classes tomorrow but there's a chance it won't be 100% complete. I was also planning to spend the little time I have at night finishing the video, so would it be alright for me to finish up on Thursday? I'll definitely be done by then and won't need anyone to cover!

Oliver Wan 02/08/2022 7:46 pm

That's okay Holly Vallester just make sure that any work you do can be checked earlier even if not 100% complete 🙂

How to make sure people get their work done on time

1. Always set deadlines
2. Avoid the social diffusion effect
3. Exploit commitment bias
4. Exploit the social facilitation effect
5. Check in regularly and send reminders
6. **A team will not lead itself**

Roles of team leader:

- Setting goals
- Delegate tasks
- Plan timeline
- Facilitate communication
- Motivate and support
- Monitor progress
- Mediate conflict

How to make sure people get their work done on time

1. Always set deadlines
2. Avoid the social diffusion effect
3. Exploit commitment bias
4. Exploit the social facilitation effect
5. Check in regularly and send reminders
6. **A team will not lead itself**

A team leader does not have to:

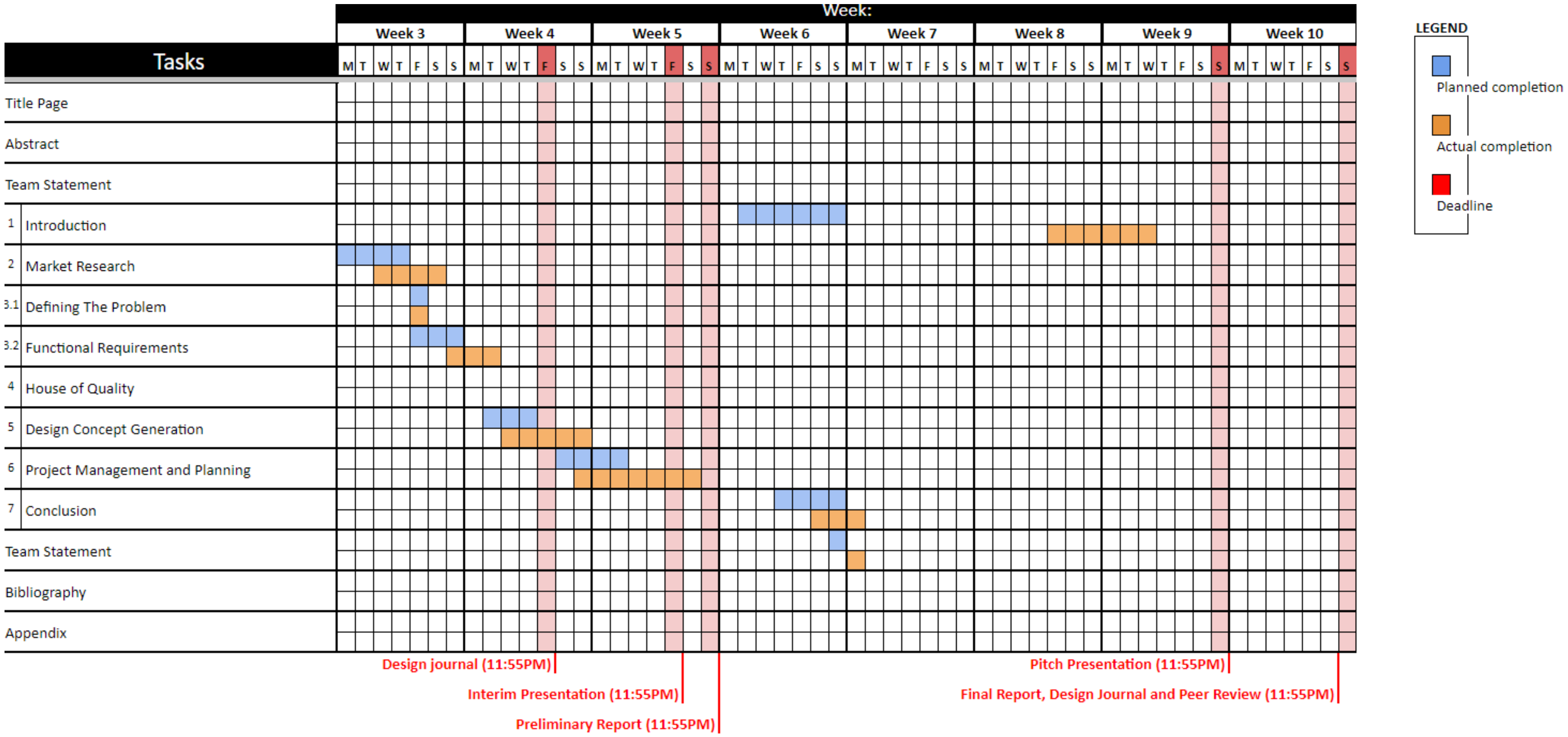
- Have the best technical skills
- Be a dictator
- Have a higher workload than everyone else
- Start fights with people
- Be the most experienced / authoritative member

GANTT Charts

- Visualise task scheduling
- Communicate dependencies between tasks
- Allow for progress tracking
- Not necessary in this course

GANTT Charts

- Visualise task scheduling
- Communicate dependencies between tasks
- Allow for progress tracking
- Not necessary in this course



How to make sure work is done to a high standard

- READ THE MARKING CRITERIA
  - If the marking criteria is too vague, add specificity yourself
- Have a rigorous review process
- Don’t be afraid to give each other feedback

How you can become a better contributor

- **80/20 rule**

  80% of your marks can be achieved from 20% of your effort. Put your time where it will make the greatest impact.

- **Embrace imposter syndrome**

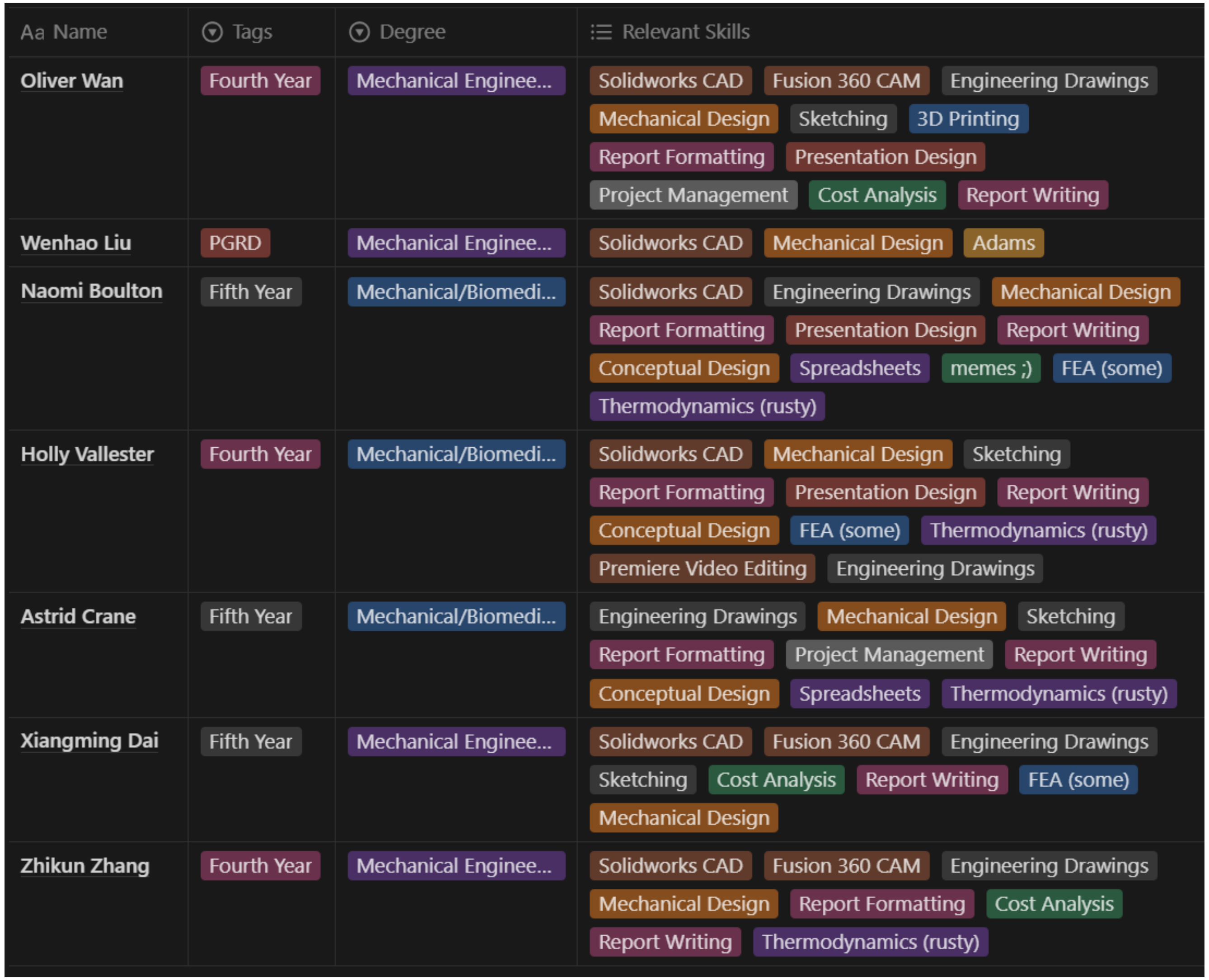
  Don’t shy away from a task just because you aren’t 100% sure you can do it. You’ll learn along the way. Rani will explain how to improve your technical skills

- **Be transparent**

  If things aren’t going well, tell the group. If you aren’t going to get a task done on time, tell the group.

How to distribute tasks fairly

- **Identify strengths and weaknesses**
  - Note: Most tasks in 1531 are similar to each other, so strengths and weaknesses have less range



Aa Name	Tags	Degree	Relevant Skills
Oliver Wan	Fourth Year	Mechanical Enginee...	Solidworks CAD Fusion 360 CAM Engineering Drawings Mechanical Design Sketching 3D Printing Report Formatting Presentation Design Project Management Cost Analysis Report Writing
Wenhao Liu	PGRD	Mechanical Enginee...	Solidworks CAD Mechanical Design Adams
Naomi Boulton	Fifth Year	Mechanical/Biomedi...	Solidworks CAD Engineering Drawings Mechanical Design Report Formatting Presentation Design Report Writing Conceptual Design Spreadsheets memes ;) FEA (some) Thermodynamics (rusty)
Holly Vallester	Fourth Year	Mechanical/Biomedi...	Solidworks CAD Mechanical Design Sketching Report Formatting Presentation Design Report Writing Conceptual Design FEA (some) Thermodynamics (rusty) Premiere Video Editing Engineering Drawings
Astrid Crane	Fifth Year	Mechanical/Biomedi...	Engineering Drawings Mechanical Design Sketching Report Formatting Project Management Report Writing Conceptual Design Spreadsheets Thermodynamics (rusty)
Xiangming Dai	Fifth Year	Mechanical Enginee...	Solidworks CAD Fusion 360 CAM Engineering Drawings Sketching Cost Analysis Report Writing FEA (some) Mechanical Design
Zhikun Zhang	Fourth Year	Mechanical Enginee...	Solidworks CAD Fusion 360 CAM Engineering Drawings Mechanical Design Report Formatting Cost Analysis Report Writing Thermodynamics (rusty)

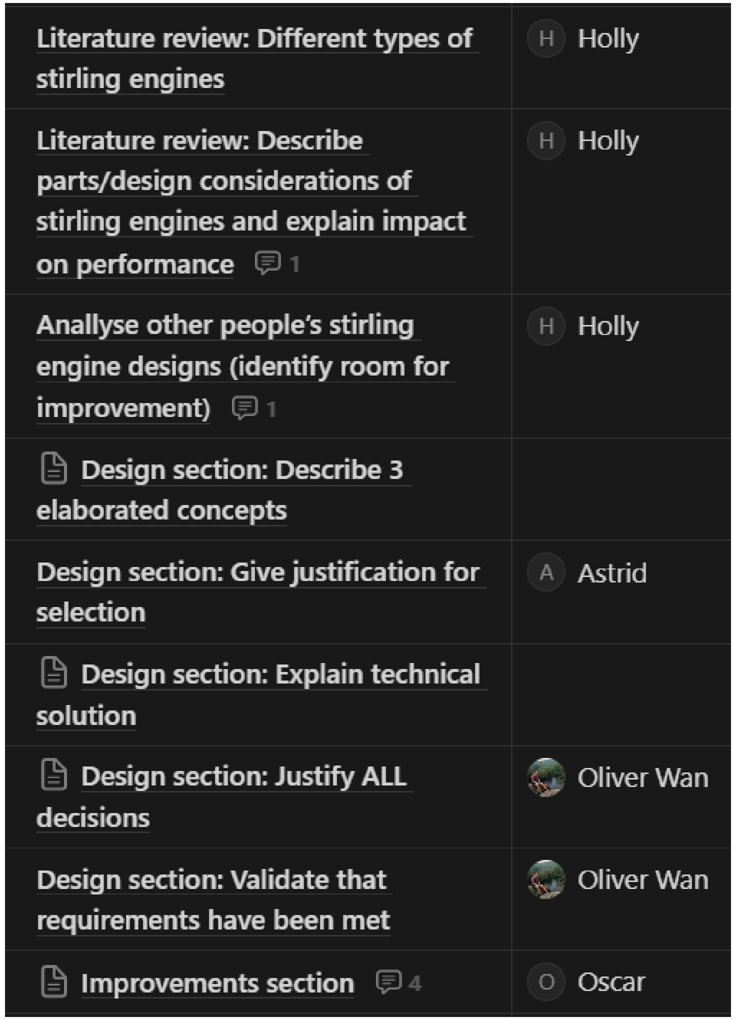
How to distribute tasks fairly

- Identify strengths and weaknesses
- Assign specific roles
- **Not all tasks require high technical competence**
  - However, in COMP1531, documentation contribution is not a substitute for code contribution

- Project management
- Sending out reminders of upcoming deadlines
- User Interface Design
- Brainstorming Test Cases
- Usability testing
- Writing Documentation
- Report writing – executive summary, intro, conclusion, etc
- Reviewing the Specification

How to distribute tasks fairly

- Identify strengths and weaknesses
- Assign specific roles
- Not all tasks require high technical competence
- **Let people pick their tasks, BUT…**



Literature review: Different types of stirling engines	H Holly
Literature review: Describe parts/design considerations of stirling engines and explain impact on performance 💬 1	H Holly
Anallyse other people's stirling engine designs (identify room for improvement) 💬 1	H Holly
📄 Design section: Describe 3 elaborated concepts	
Design section: Give justification for selection	A Astrid
📄 Design section: Explain technical solution	
📄 Design section: Justify ALL decisions	Oliver Wan
Design section: Validate that requirements have been met	Oliver Wan
📄 Improvements section 💬 4	O Oscar

How to distribute tasks fairly

- Identify strengths and weaknesses
- Assign specific roles
- Not all tasks require high technical competence
- Let people pick their tasks, BUT…
- **Verbal agreements of equity**

Is everyone happy with their tasks?

Are you comfortable with this workload?

Does that seem fair?

What to do if someone is slacking off

- Understand the root cause:
  - Procrastination
    - Lack of intrinsic motivation
      - Have a private, candid conversation with them
      - Give them tasks that they genuinely enjoy
    - Misunderstanding of deadline importance
      - Emphasise the consequences of missing the deadline
    - Overwhelming task size
      - Break big tasks up into smaller tasks, which are checked up on each week
  - Personal issues
    - Give their tasks to other team members
    - Offer them support
  - Heavy workload from other courses
    - Very rarely valid

How to make Decisions as a Group!

Which kanban board should we use?

Jira!

Gitlab Issues!

Gitlab Issues!

Gitlab Issues!

Gitlab Issues!

Gitlab Issues!

How to make Decisions as a Group!

Which kanban board should we use?

I know everyone wants to use Gitlab Issues, but do you mind if I share why I think Jira is better?

How to make Decisions as a Group!

Let’s use an array of objects!

No, let’s use a single object!

No, let’s use multiple arrays!

How to make Decisions as a Group!

Let’s call the variable `data`

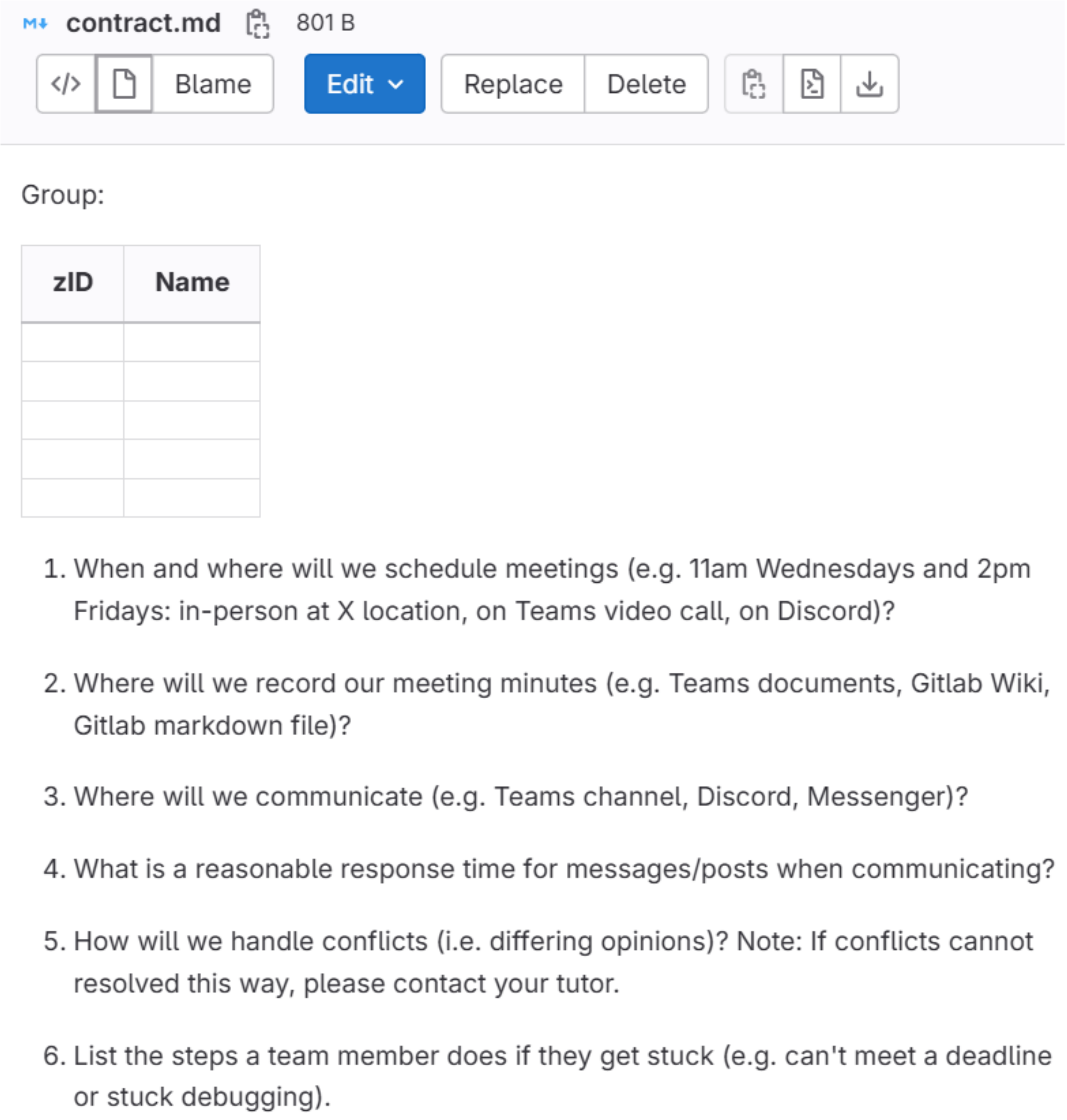
No, let’s call it `dataStore`

Most common causes of arguments

- Strong, conflicting opinions
  - Design decisions
- People not willing to listen to others
- People not getting a chance to share their ideas
- Having different rules for acceptable behaviour
- Workload not being shared equally
- Not having clear rules about when and where to communicate
- People not attending meetings / doing work on time

Group contract

- Assessed in iteration 0
- Use the template in the repo



Group:

zID	Name

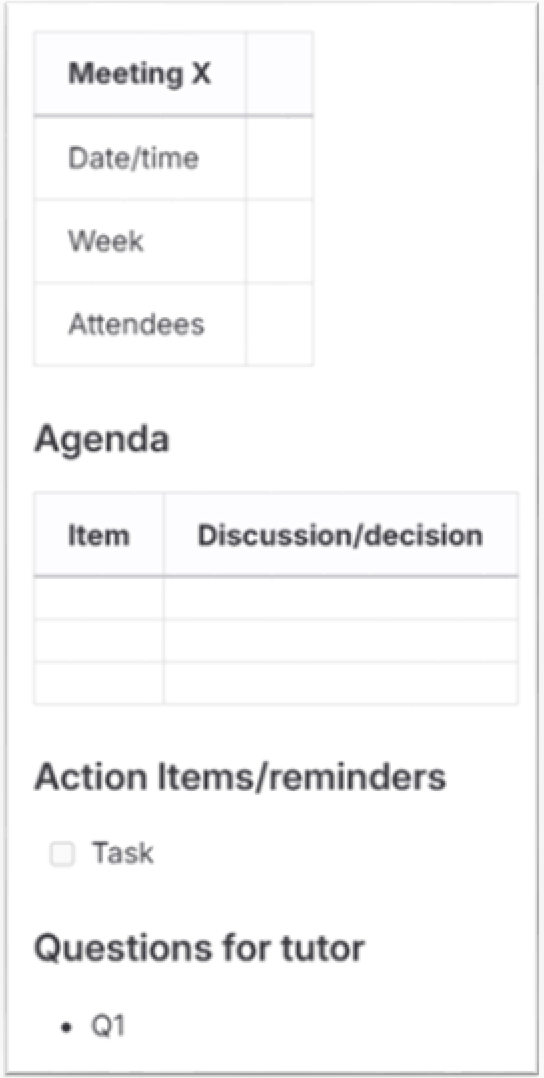
1. When and where will we schedule meetings (e.g. 11am Wednesdays and 2pm Fridays: in-person at X location, on Teams video call, on Discord)?
2. Where will we record our meeting minutes (e.g. Teams documents, Gitlab Wiki, Gitlab markdown file)?
3. Where will we communicate (e.g. Teams channel, Discord, Messenger)?
4. What is a reasonable response time for messages/posts when communicating?
5. How will we handle conflicts (i.e. differing opinions)? Note: If conflicts cannot resolved this way, please contact your tutor.
6. List the steps a team member does if they get stuck (e.g. can't meet a deadline or stuck debugging).

How to run a meeting that isn’t a huge waste of time

Types of meetings

- Kick-off meeting
- Weekly status update meetings
- Daily stand-ups
- Decision making meetings
- Brainstorming meetings
- Design reviews
- Debrief Meetings
- Team building meetings
- Co-working sessions

Meeting Minutes



Meeting X	
Date/time	
Week	
Attendees	

Agenda

Item	Discussion/decision

Action Items/reminders

□ Task

Questions for tutor

- Q1

- This is the level of complexity that your marker expects
- A markdown meeting minutes template has been provided in the COMP1531 repo
  - You don't have to use markdown
  - Use any tool you wish, to create your minutes

Specific Strategies for COMP1531

We will cover:

- How to make sure people get their work **done on time**
- How to make sure tasks are done to a **high standard**
- How to be a **better contributor**
- How to distribute tasks fairly
- How to prevent arguments
- How to make decisions as a group
- How to run a meeting that isn’t a complete waste of time

Getting Work Done on Time

- Plan for dependencies
- Definition of Completion
- Frequent check-ins
- Understand Progress

Getting Work Done on Time

- Plan for dependencies

Getting Work Done on Time

- Plan for dependencies

Each iteration will have a list to implement:

- clear
- adminAuthRegister
- adminUserDetails
- adminQuizCreate
- adminQuizList
- adminNameQuizUpdate
- , etc

Getting Work Done on Time

- Plan for dependencies

Each iteration will have a list to implement:

- clear
- adminAuthRegister
- adminUserDetails
- adminQuizCreate
- adminQuizList
- adminQuizNameUpdate
- , etc

Is there an order?

Getting Work Done on Time

- Plan for dependencies

Each iteration will have a list to implement:

- clear
- adminAuthRegister
- adminUserDetails
- adminQuizCreate
- adminQuizList
- adminQuizNameUpdate
- , etc

Is there an order?

No, you can work on branches in parallel.
Yes, some functions rely on others to test.

Getting Work Done on Time

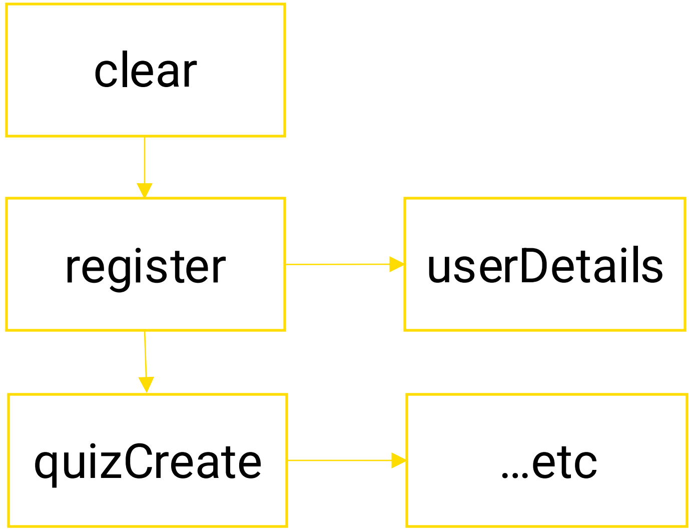
- Plan for dependencies → *dependencies graph*

Each iteration will have a list to implement:

- clear
- adminAuthRegister
- adminUserDetails
- adminQuizCreate
- adminQuizList
- adminQuizNameUpdate
- , etc

Is there an order?

No, you can work on branches in parallel.
Yes, some functions rely on others to test.



Getting Work Done on Time

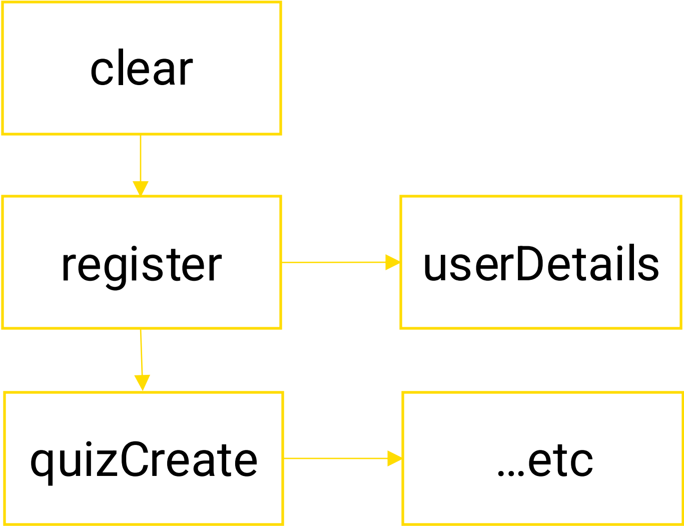
- Plan for dependencies → *dependencies graph*

Each iteration will have a list to implement:

- clear
- adminAuthRegister
- adminUserDetails
- adminQuizCreate
- adminQuizList
- adminQuizNameUpdate
- , etc

Is there an order?

No, you can work on branches in parallel.
Yes, some functions rely on others to test.



* break up any circular dependencies

Getting Work Done on Time

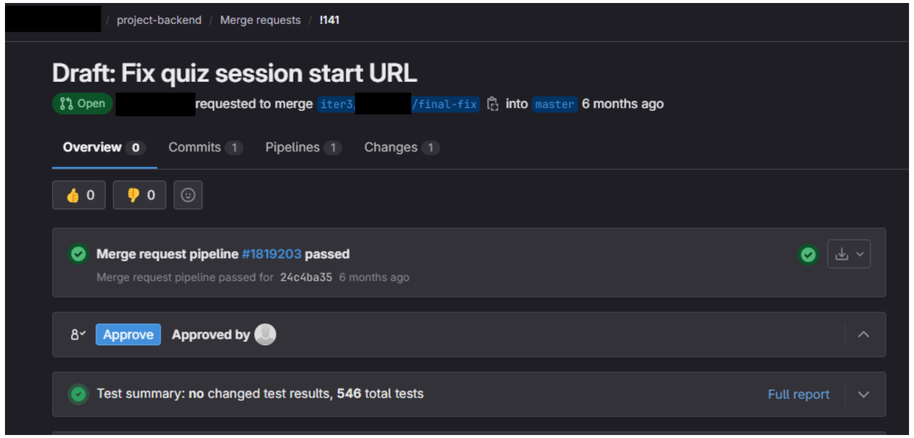
- Definition of completion → what does “*finished*” mean?

Getting Work Done on Time

- Definition of completion → what does "*finished*" mean?
- What if your team has different definitions of "*finished*"?

Getting Work Done on Time

- Definition of completion → what does “*finished*” mean?
- What if your team has different definitions of “*finished*”?



[redacted] / project-backend / Merge requests / !141

Draft: Fix quiz session start URL

Open [redacted] requested to merge iter3[redacted]/final-fix into master 6 months ago

Overview 0 Commits 1 Pipelines 1 Changes 1

👍 0 👎 0

Merge request pipeline #1819203 passed

Merge request pipeline passed for 24c4ba35 6 months ago

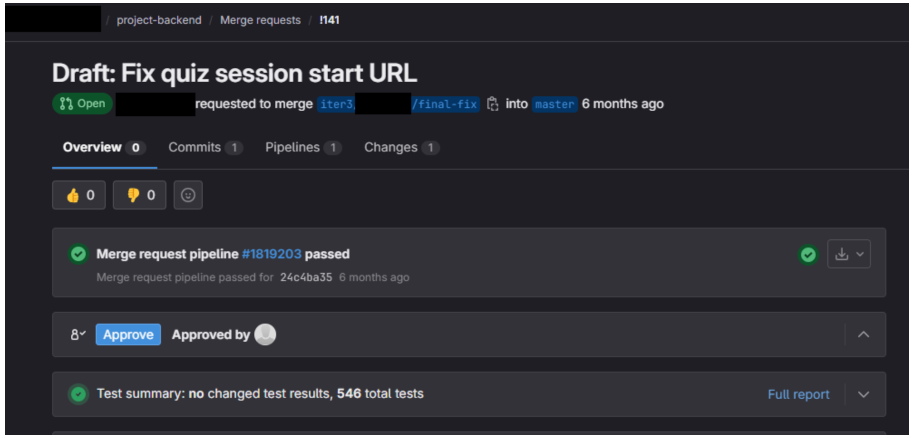
Approved by

Test summary: no changed test results, 546 total tests

Merge Request
1-click away from merging

Getting Work Done on Time

- Definition of completion → what does “*finished*” mean?
- What if your team has different definitions of “*finished*”?



[redacted] / project-backend / Merge requests / !141

Draft: Fix quiz session start URL

Open [redacted] requested to merge `iter3`[redacted]`/final-fix` into `master` 6 months ago

Overview 0 Commits 1 Pipelines 1 Changes 1

👍 0 👎 0

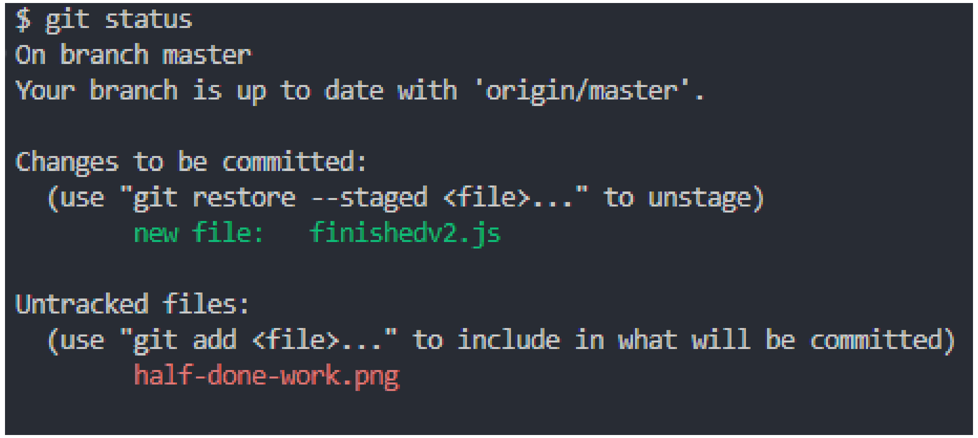
Merge request pipeline #1819203 passed

Merge request pipeline passed for 24c4ba35 6 months ago

Approve Approved by

Test summary: no changed test results, 546 total tests Full report

Merge Request
1-click away from merging



```
$ git status
On branch master
Your branch is up to date with 'origin/master'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   finishedv2.js

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        half-done-work.png
```

Local Code
untested, unreviewed, ready to rumble

Getting Work Done on Time

- Frequent check-ins

- An iteration is ~two weeks…
- Meetings are too long
- What else can we do?

Getting Work Done on Time

- Frequent check-ins → Standups

- An iteration is ~two weeks…
- Meetings are too long
- What else can we do?

Getting Work Done on Time

- Frequent check-ins →

- An iteration is ~two weeks…
- Meetings are too long
- What else can we do?

Standups

- Frequent (often daily) short progress updates

Getting Work Done on Time

- Frequent check-ins → Standups

- An iteration is ~two weeks…
- Meetings are too long
- What else can we do?

- Frequent (often daily) short progress updates
- Answer 3 key questions
  - What did I do?
  - What problems did I face? (Blockers)
  - What am I going to do?

Getting Work Done on Time

- Frequent check-ins →

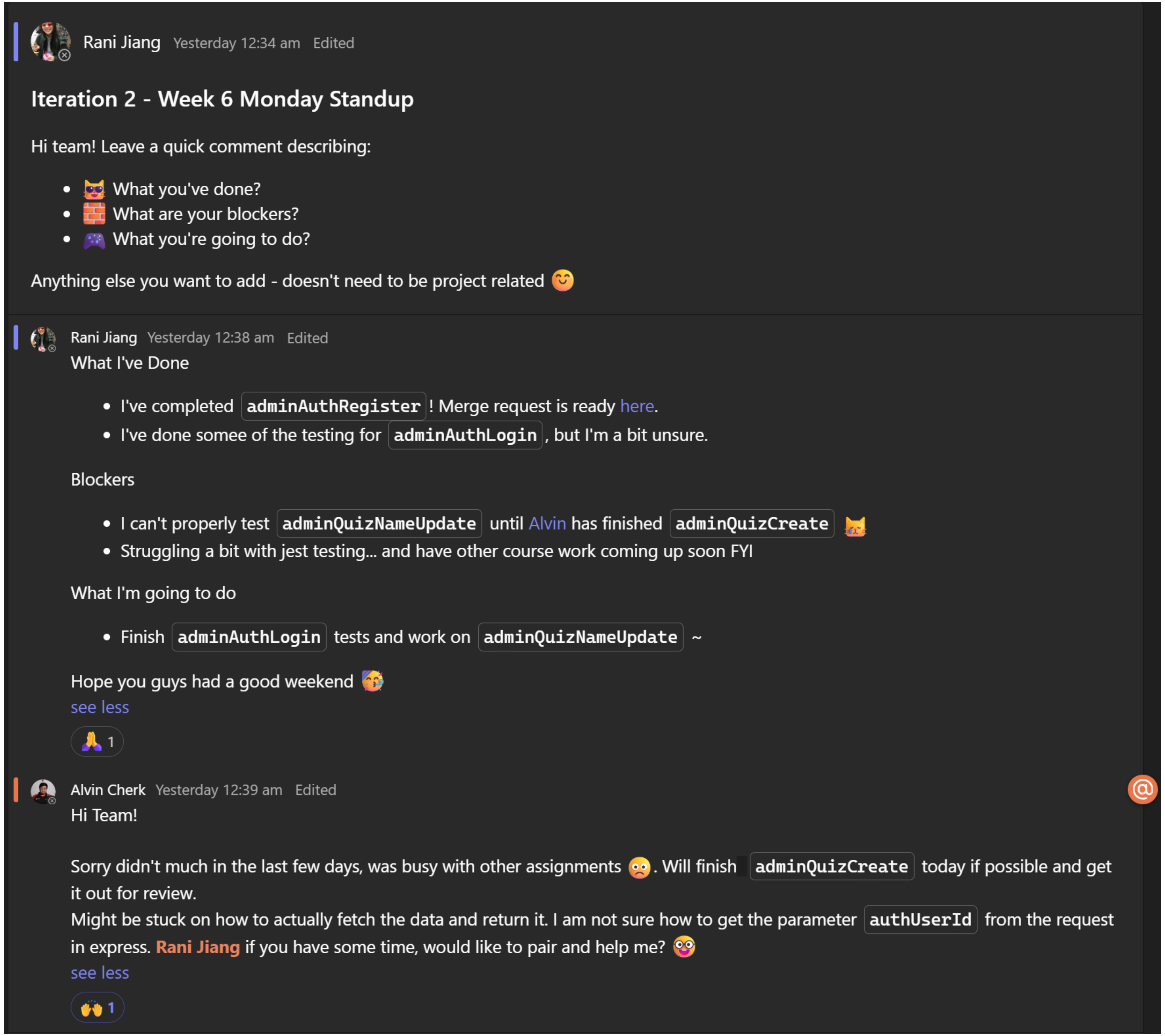
- An iteration is ~two weeks…
- Meetings are too long
- What else can we do?

Standups

- Frequent (often daily) short progress updates
- Answer 3 key questions
  - What did I do?
  - What problems did I face? (Blockers)
  - What am I going to do?

- Can be synchronous or asynchronous
  - sync: in real-time e.g. stand up in-person
  - async: not in real-time e.g. via email

Example *Async* Standup



Rani Jiang Yesterday 12:34 am Edited

Iteration 2 - Week 6 Monday Standup

Hi team! Leave a quick comment describing:

- 😼 What you've done?
- 🧱 What are your blockers?
- 🎮 What you're going to do?

Anything else you want to add - doesn't need to be project related 😊

Rani Jiang Yesterday 12:38 am Edited

What I've Done

- I've completed `adminAuthRegister`! Merge request is ready here.
- I've done somee of the testing for `adminAuthLogin`, but I'm a bit unsure.

Blockers

- I can't properly test `adminQuizNameUpdate` until Alvin has finished `adminQuizCreate` 😿
- Struggling a bit with jest testing... and have other course work coming up soon FYI

What I'm going to do

- Finish `adminAuthLogin` tests and work on `adminQuizNameUpdate` ~

Hope you guys had a good weekend 🥳

see less

🙏 1

Alvin Cherk Yesterday 12:39 am Edited

Hi Team!

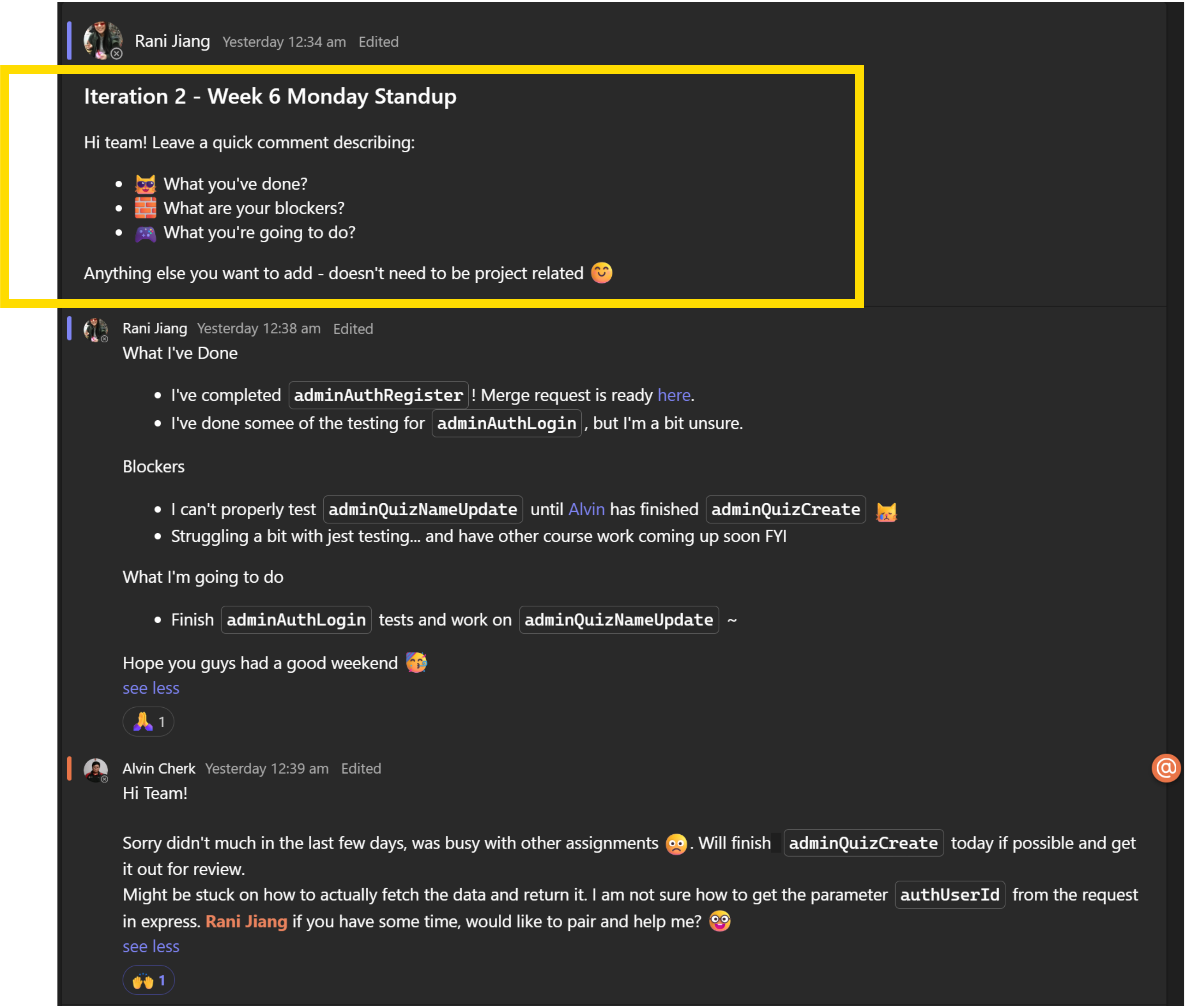
Sorry didn't much in the last few days, was busy with other assignments 🥺. Will finish `adminQuizCreate` today if possible and get it out for review.

Might be stuck on how to actually fetch the data and return it. I am not sure how to get the parameter `authUserId` from the request in express. Rani Jiang if you have some time, would like to pair and help me? 🤓

see less

🙌 1

Example *Async* Standup



Rani Jiang Yesterday 12:34 am Edited

Iteration 2 - Week 6 Monday Standup

Hi team! Leave a quick comment describing:

- 😼 What you've done?
- 🧱 What are your blockers?
- 🎮 What you're going to do?

Anything else you want to add - doesn't need to be project related 😊

Rani Jiang Yesterday 12:38 am Edited

What I've Done

- I've completed `adminAuthRegister`! Merge request is ready here.
- I've done somee of the testing for `adminAuthLogin`, but I'm a bit unsure.

Blockers

- I can't properly test `adminQuizNameUpdate` until Alvin has finished `adminQuizCreate` 😿
- Struggling a bit with jest testing... and have other course work coming up soon FYI

What I'm going to do

- Finish `adminAuthLogin` tests and work on `adminQuizNameUpdate` ~

Hope you guys had a good weekend 🥳

see less

🙏 1

Alvin Cherk Yesterday 12:39 am Edited

Hi Team!

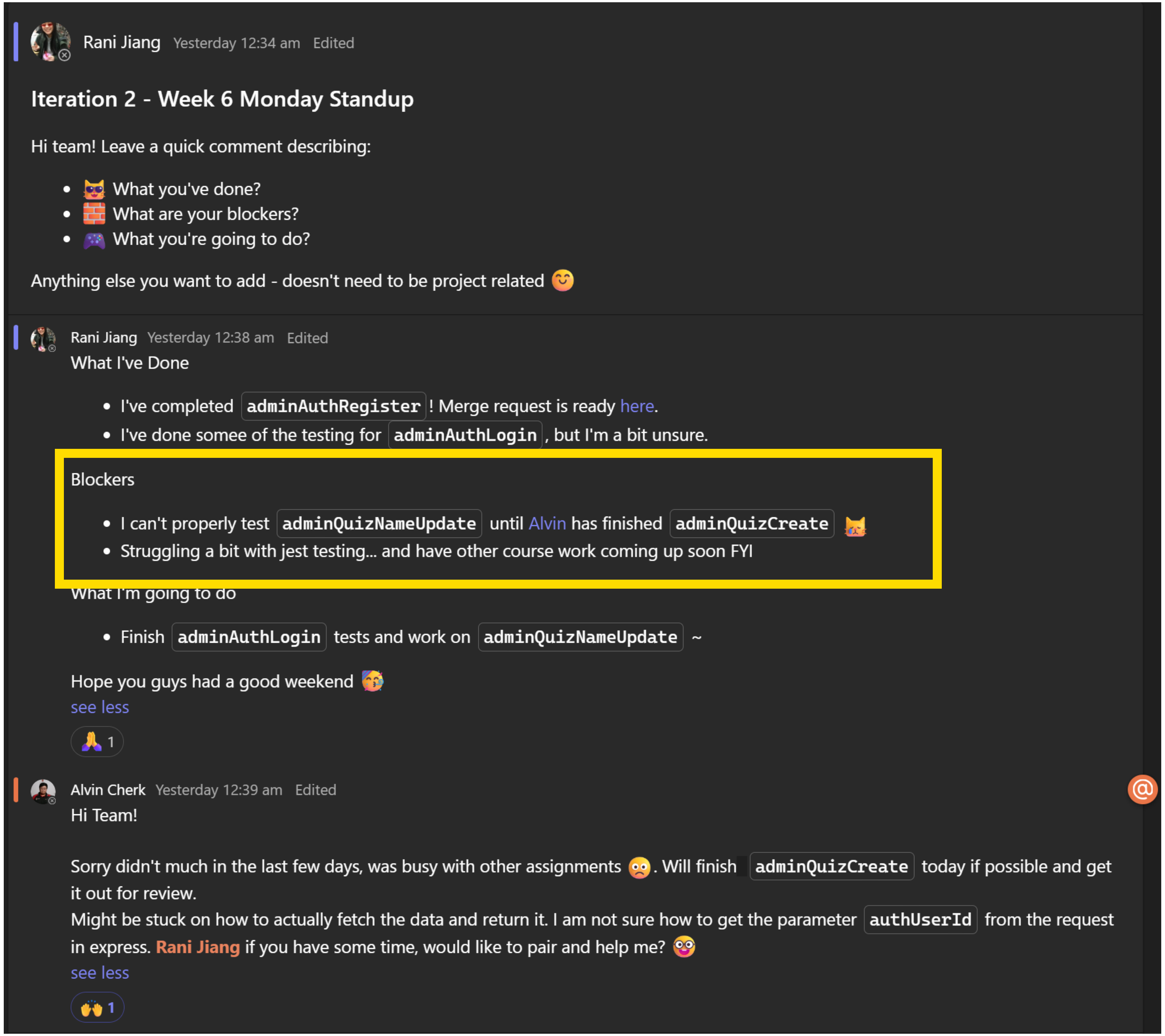
Sorry didn't much in the last few days, was busy with other assignments 😦. Will finish `adminQuizCreate` today if possible and get it out for review.

Might be stuck on how to actually fetch the data and return it. I am not sure how to get the parameter `authUserId` from the request in express. Rani Jiang if you have some time, would like to pair and help me? 😳

see less

🙌 1

Example *Async* Standup



Rani Jiang Yesterday 12:34 am Edited

Iteration 2 - Week 6 Monday Standup

Hi team! Leave a quick comment describing:

- 😼 What you've done?
- 🧱 What are your blockers?
- 🎮 What you're going to do?

Anything else you want to add - doesn't need to be project related 😊

Rani Jiang Yesterday 12:38 am Edited

What I've Done

- I've completed `adminAuthRegister`! Merge request is ready here.
- I've done somee of the testing for `adminAuthLogin`, but I'm a bit unsure.

Blockers

- I can't properly test `adminQuizNameUpdate` until Alvin has finished `adminQuizCreate` 😿
- Struggling a bit with jest testing... and have other course work coming up soon FYI

What I'm going to do

- Finish `adminAuthLogin` tests and work on `adminQuizNameUpdate` ~

Hope you guys had a good weekend 🥳

see less

🙏 1

Alvin Cherk Yesterday 12:39 am Edited

Hi Team!

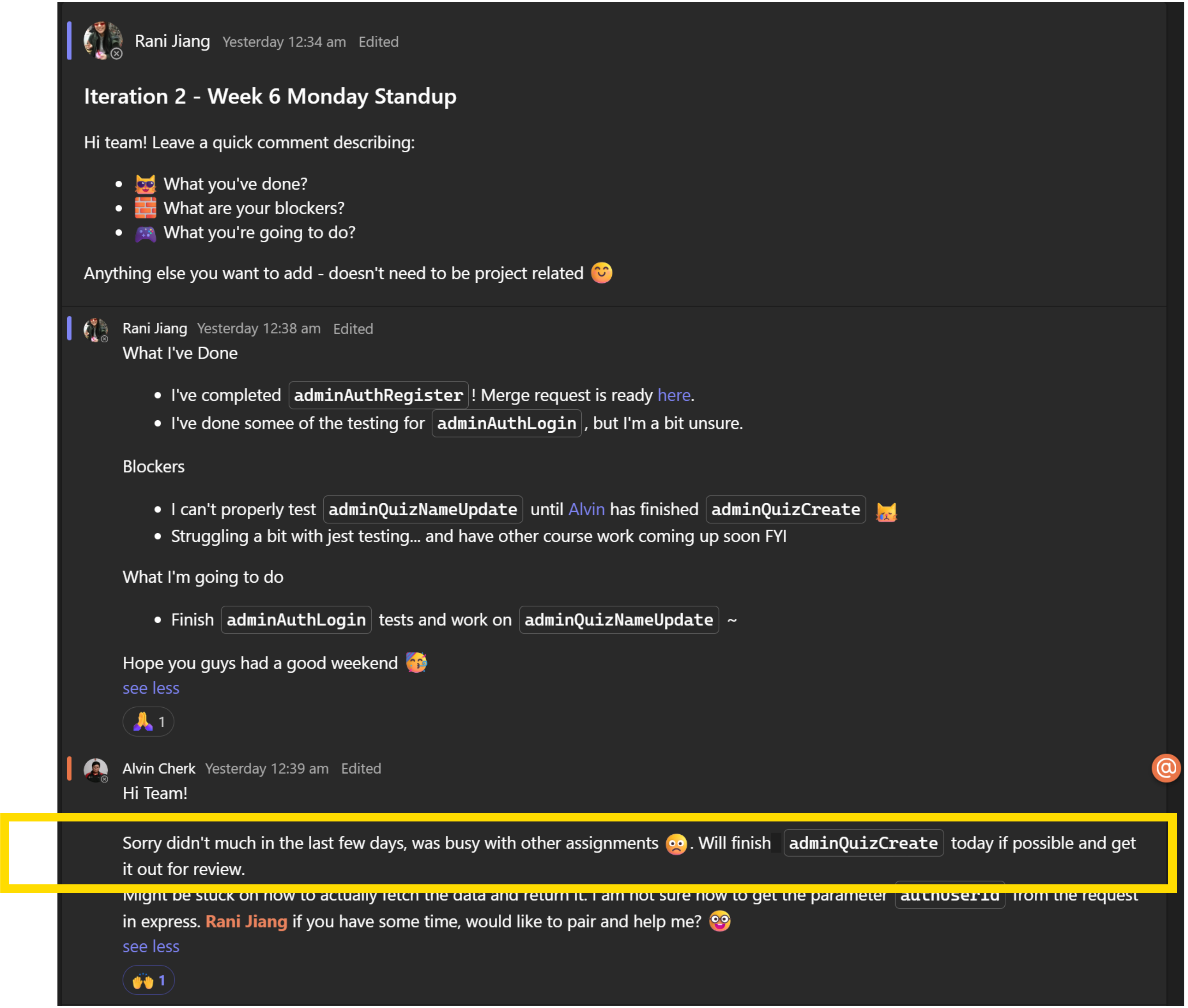
Sorry didn't much in the last few days, was busy with other assignments 🙁. Will finish `adminQuizCreate` today if possible and get it out for review.

Might be stuck on how to actually fetch the data and return it. I am not sure how to get the parameter `authUserId` from the request in express. **Rani Jiang** if you have some time, would like to pair and help me? 😳

see less

🙌 1

Example *Async* Standup



Rani Jiang Yesterday 12:34 am Edited

Iteration 2 - Week 6 Monday Standup

Hi team! Leave a quick comment describing:

- 😼 What you've done?
- 🧱 What are your blockers?
- 🎮 What you're going to do?

Anything else you want to add - doesn't need to be project related 😊

Rani Jiang Yesterday 12:38 am Edited

What I've Done

- I've completed `adminAuthRegister`! Merge request is ready here.
- I've done somee of the testing for `adminAuthLogin`, but I'm a bit unsure.

Blockers

- I can't properly test `adminQuizNameUpdate` until Alvin has finished `adminQuizCreate` 😿
- Struggling a bit with jest testing... and have other course work coming up soon FYI

What I'm going to do

- Finish `adminAuthLogin` tests and work on `adminQuizNameUpdate` ~

Hope you guys had a good weekend 🥳

see less

🙏 1

Alvin Cherk Yesterday 12:39 am Edited

Hi Team!

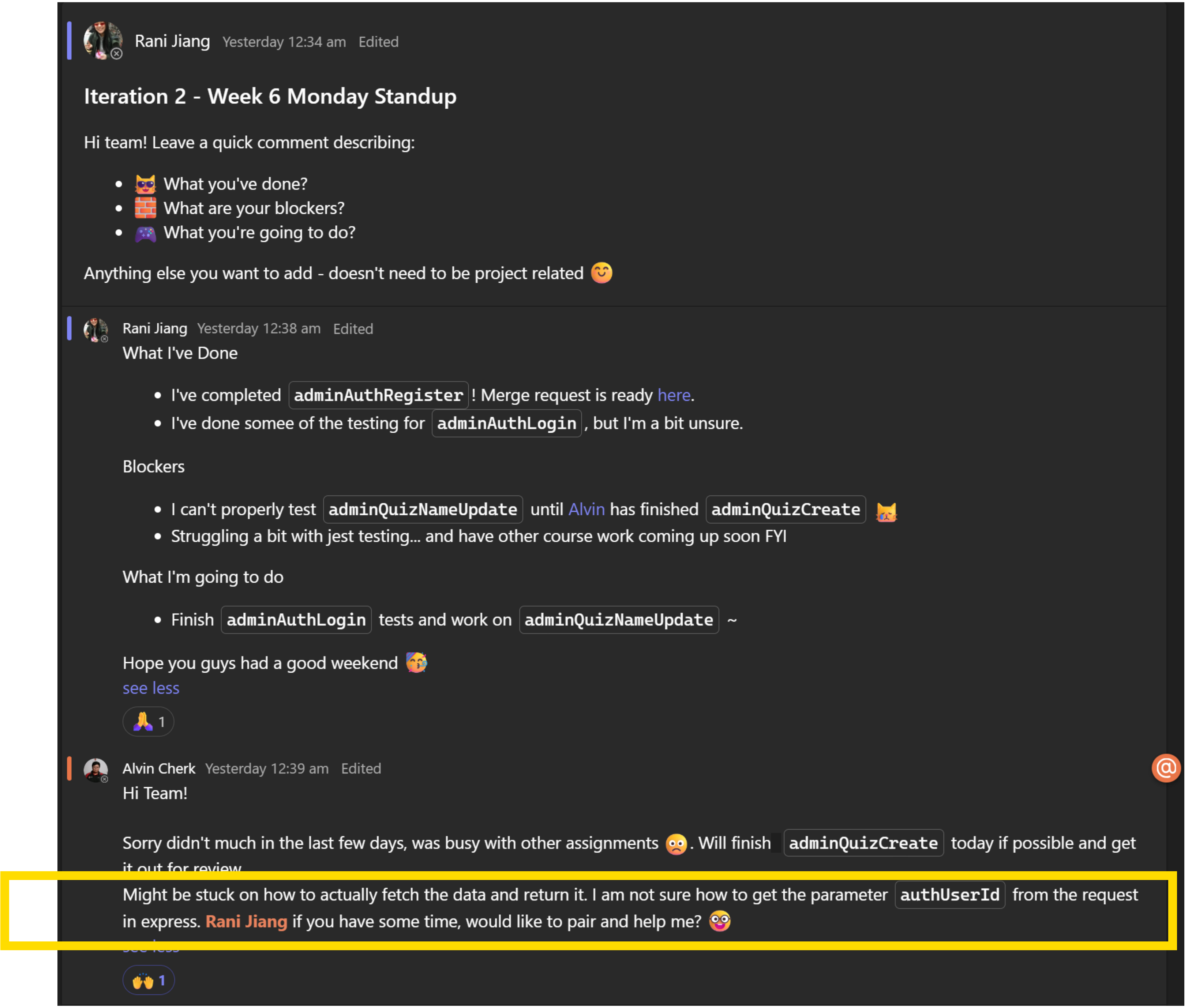
Sorry didn't much in the last few days, was busy with other assignments 😦. Will finish `adminQuizCreate` today if possible and get it out for review.

Might be stuck on how to actually fetch the data and return it. I am not sure how to get the parameter `authUserId` from the request in express. Rani Jiang if you have some time, would like to pair and help me? 😳

see less

🙌 1

Example *Async* Standup



Rani Jiang Yesterday 12:34 am Edited

Iteration 2 - Week 6 Monday Standup

Hi team! Leave a quick comment describing:

- 😼 What you've done?
- 🧱 What are your blockers?
- 🎮 What you're going to do?

Anything else you want to add - doesn't need to be project related 😊

Rani Jiang Yesterday 12:38 am Edited

What I've Done

- I've completed `adminAuthRegister`! Merge request is ready here.
- I've done somee of the testing for `adminAuthLogin`, but I'm a bit unsure.

Blockers

- I can't properly test `adminQuizNameUpdate` until Alvin has finished `adminQuizCreate` 😿
- Struggling a bit with jest testing... and have other course work coming up soon FYI

What I'm going to do

- Finish `adminAuthLogin` tests and work on `adminQuizNameUpdate` ~

Hope you guys had a good weekend 🥳

see less

🙏 1

Alvin Cherk Yesterday 12:39 am Edited

Hi Team!

Sorry didn't much in the last few days, was busy with other assignments 🥺. Will finish `adminQuizCreate` today if possible and get it out for review.

Might be stuck on how to actually fetch the data and return it. I am not sure how to get the parameter `authUserId` from the request in express. **Rani Jiang** if you have some time, would like to pair and help me? 😳

see less

🙌 1

Getting Work Done on Time

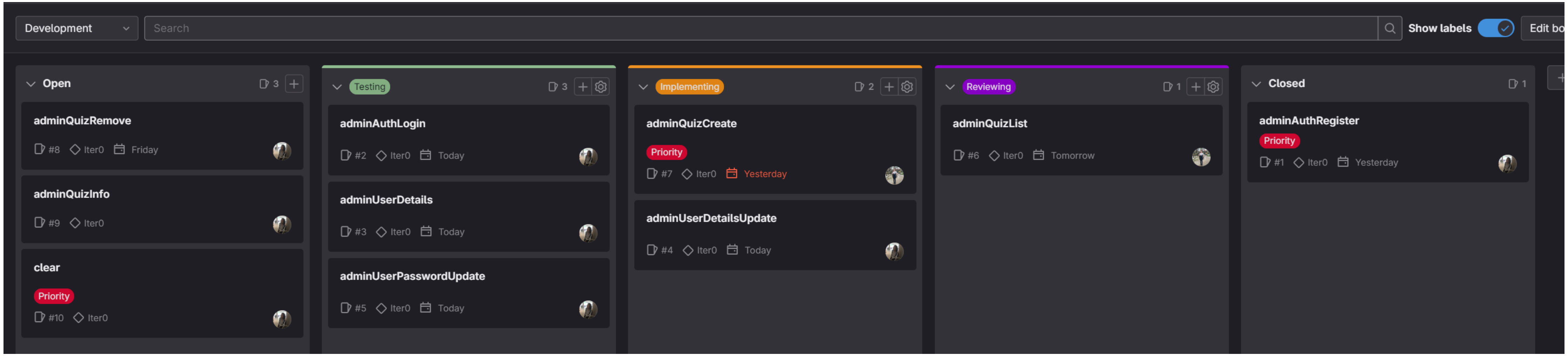
- Understand Progress→ Task Board

Getting Work Done on Time

- Understand Progress→ Task Board
- Project MGMT tool to track tasks: description, status, assignee

Getting Work Done on Time

- Understand Progress→ Task Board
- Project MGMT tool to track tasks: description, status, assignee
- Progress at a glance



Open 3

adminQuizRemove
#8 Iter0 Friday

adminQuizInfo
#9 Iter0

clear
Priority
#10 Iter0

Testing 3

adminAuthLogin
#2 Iter0 Today

adminUserDetails
#3 Iter0 Today

adminUserPasswordUpdate
#5 Iter0 Today

Implementing 2

adminQuizCreate
Priority
#7 Iter0 Yesterday

adminUserDetailsUpdate
#4 Iter0 Today

Reviewing 1

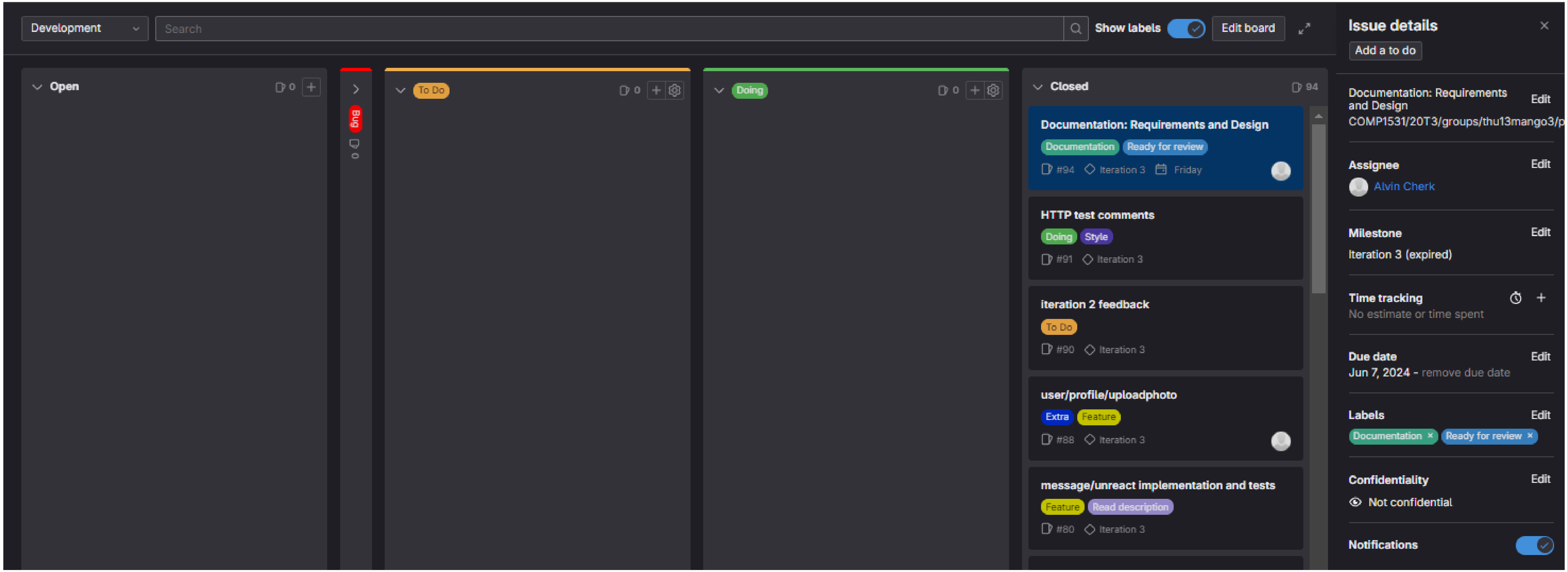
adminQuizList
#6 Iter0 Tomorrow

Closed 1

adminAuthRegister
Priority
#1 Iter0 Yesterday

Simpler Layout

To Do, Doing, Done



Documentation: Requirements and Design
Documentation | Ready for review
#94 Iteration 3 Friday

HTTP test comments
Doing | Style
#91 Iteration 3

iteration 2 feedback
To Do
#90 Iteration 3

user/profile/uploadphoto
Extra | Feature
#88 Iteration 3

message/unreact implementation and tests
Feature | Read description
#80 Iteration 3

Issue details

Add a to do

Documentation: Requirements and Design
COMP1531/20T3/groups/thu13mango3/p

Assignee: Alvin Cherk

Milestone: Iteration 3 (expired)

Time tracking: No estimate or time spent

Due date: Jun 7, 2024 - remove due date

Labels: Documentation × | Ready for review ×

Confidentiality: Not confidential

Notifications

Getting Work Done on Time

- Plan for dependencies → dependencies graph
- Definition of completion → what does “*finished*” mean?
- Frequent check-ins → Standups
- Understand Progress→ Issues Board

Applying this advice to COMP1531….

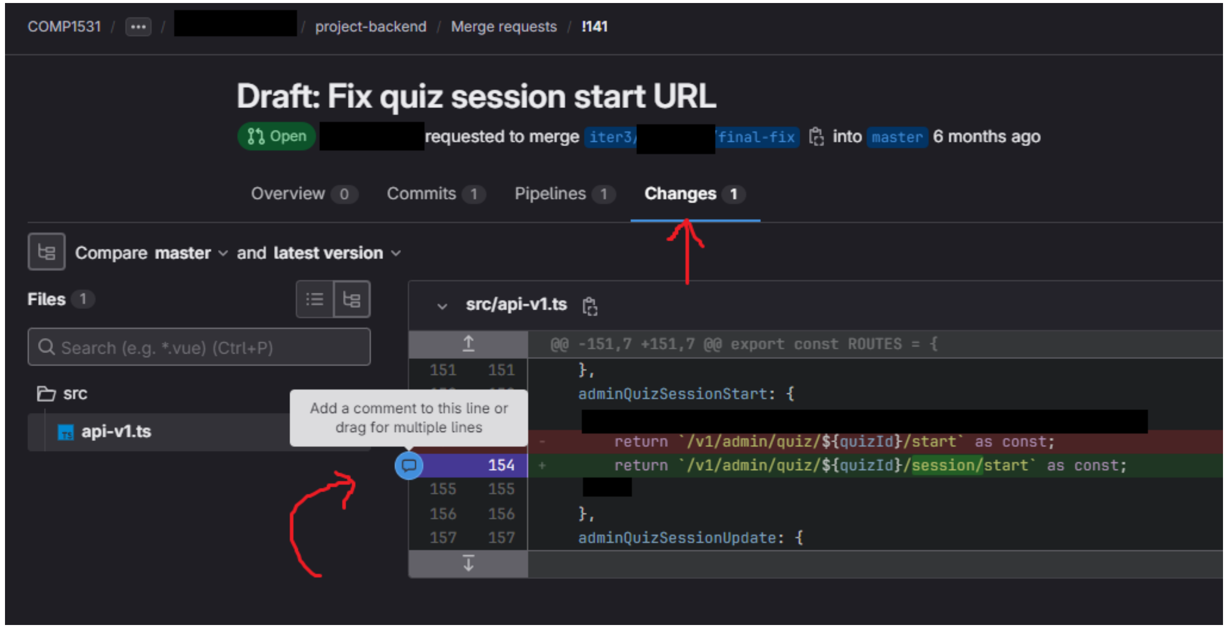
- How to make sure people get their work done on time
- How to make sure tasks are done to a **high standard**
- How to be a better contributor
- How to distribute tasks fairly
- How to prevent arguments
- How to make decisions as a group
- How to run a meeting that isn’t a complete waste of time

Tasks done to a high standard

- Code Review
- Pair Programming
- Setting git specific standards

Tasks done to a high standard

- **Code Review**
- Quality assurance process
- Different types; sync, async
- Benefits:
  - Share knowledge
  - Discover bugs earlier
  - Maintain standards
  - Improve code quality
- No single point of failure



COMP1531 / ··· / [redacted] / project-backend / Merge requests / !141

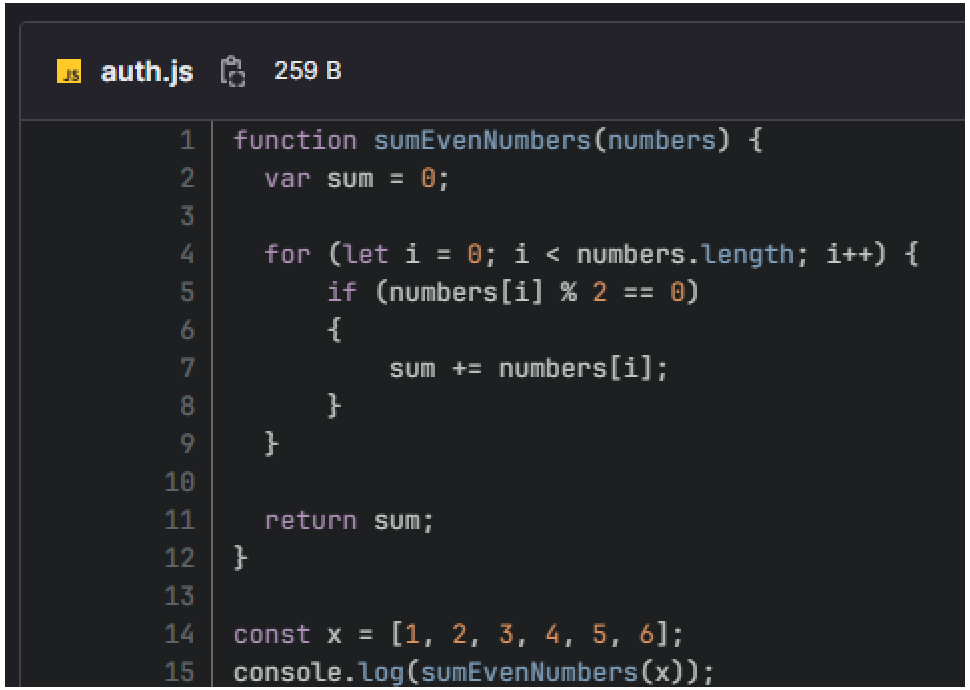
Draft: Fix quiz session start URL

Open [redacted] requested to merge iter3/[redacted]final-fix into master 6 months ago

Add a comment to this line or drag for multiple lines

```
@@ -151,7 +151,7 @@ export const ROUTES = {
  },
  adminQuizSessionStart: {
    [redacted]
-     return `/v1/admin/quiz/${quizId}/start` as const;
+     return `/v1/admin/quiz/${quizId}/session/start` as const;
    [redacted]
  },
  adminQuizSessionUpdate: {
```

Example:
Code Review



auth.js 259 B

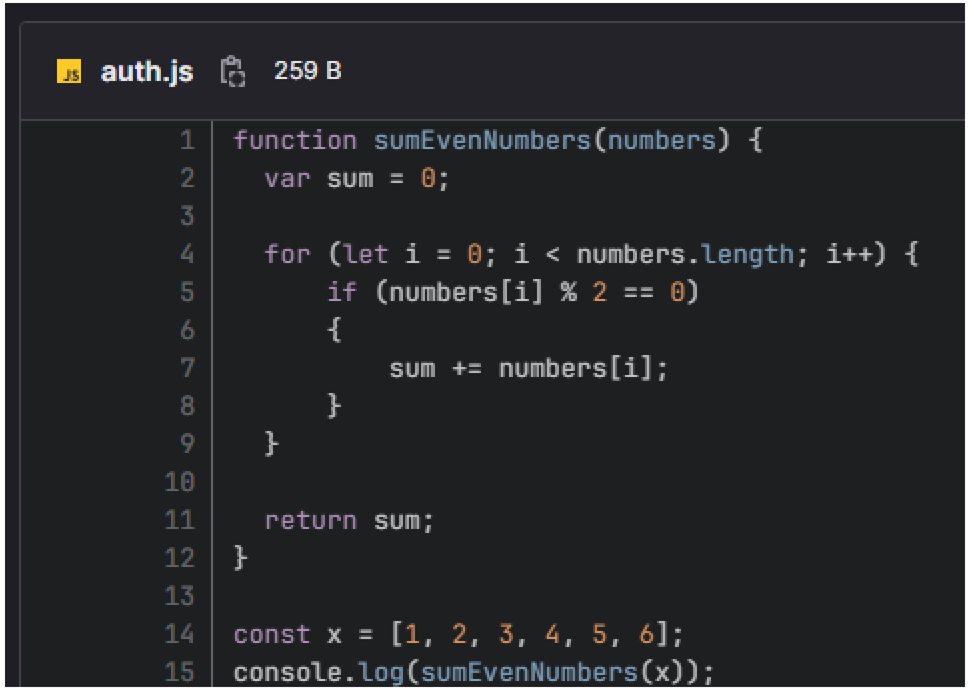
```
function sumEvenNumbers(numbers) {
  var sum = 0;

  for (let i = 0; i < numbers.length; i++) {
      if (numbers[i] % 2 == 0)
      {
          sum += numbers[i];
      }
  }

  return sum;
}

const x = [1, 2, 3, 4, 5, 6];
console.log(sumEvenNumbers(x));
```

Example: Code Review



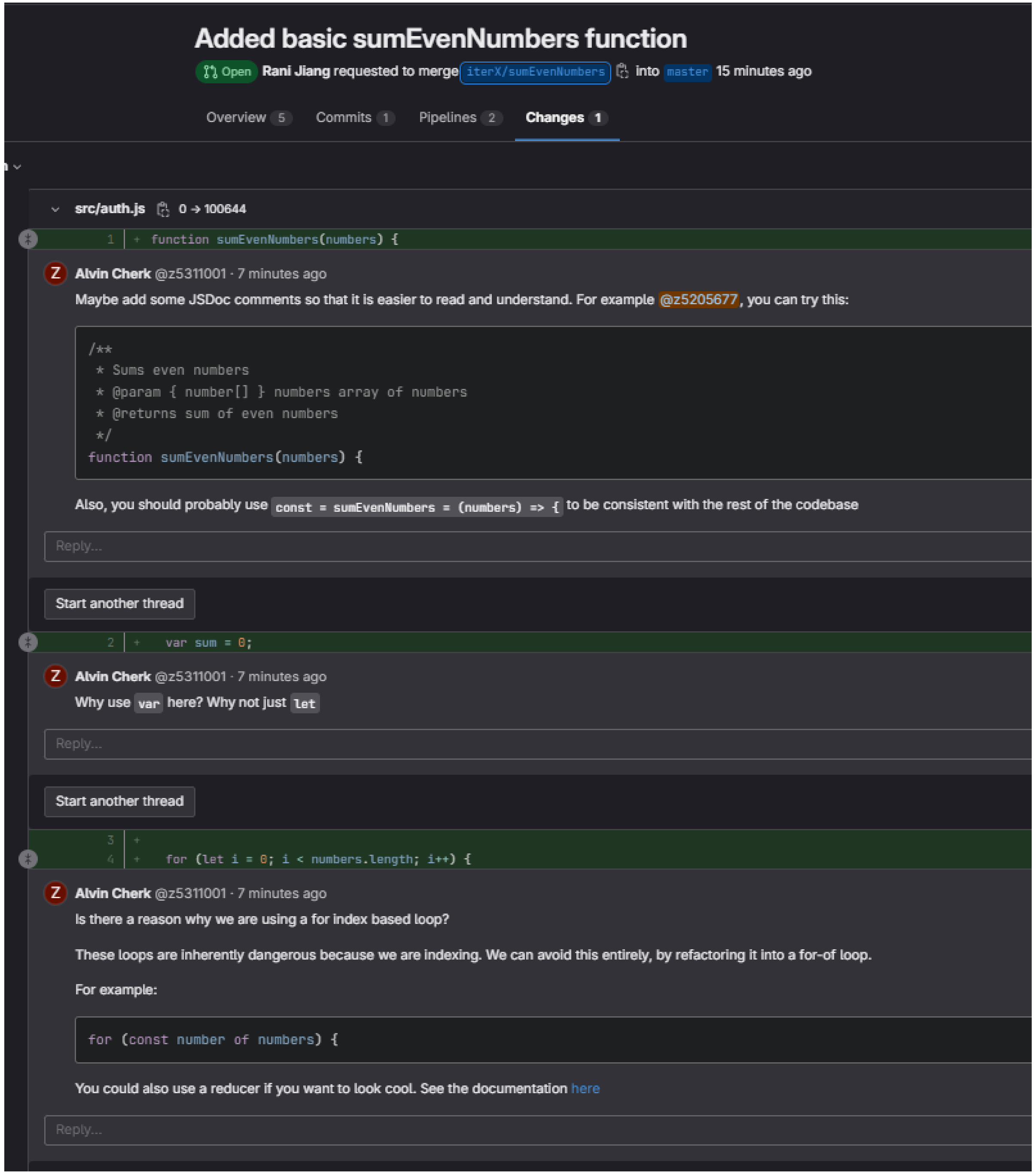
auth.js 259 B

```
function sumEvenNumbers(numbers) {
  var sum = 0;

  for (let i = 0; i < numbers.length; i++) {
      if (numbers[i] % 2 == 0)
      {
          sum += numbers[i];
      }
  }

  return sum;
}

const x = [1, 2, 3, 4, 5, 6];
console.log(sumEvenNumbers(x));
```



Added basic sumEvenNumbers function

Open Rani Jiang requested to merge `iterX/sumEvenNumbers` into `master` 15 minutes ago

src/auth.js 0 → 100644

```
+ function sumEvenNumbers(numbers) {
```

Alvin Cherk @z5311001 · 7 minutes ago

Maybe add some JSDoc comments so that it is easier to read and understand. For example @z5205677, you can try this:

```
/**
 * Sums even numbers
 * @param { number[] } numbers array of numbers
 * @returns sum of even numbers
 */
function sumEvenNumbers(numbers) {
```

Also, you should probably use `const = sumEvenNumbers = (numbers) => {` to be consistent with the rest of the codebase

```
+   var sum = 0;
```

Alvin Cherk @z5311001 · 7 minutes ago

Why use `var` here? Why not just `let`

```
+
+   for (let i = 0; i < numbers.length; i++) {
```

Alvin Cherk @z5311001 · 7 minutes ago

Is there a reason why we are using a for index based loop?

These loops are inherently dangerous because we are indexing. We can avoid this entirely, by refactoring it into a for-of loop.

For example:

```
for (const number of numbers) {
```

You could also use a reducer if you want to look cool. See the documentation here

Tasks done to a high standard

- **Pair Programming**
- Two developers, one monitor and keyboard – working together
- Fosters knowledge sharing, real time code review, increase collaboration

Tasks done to a high standard

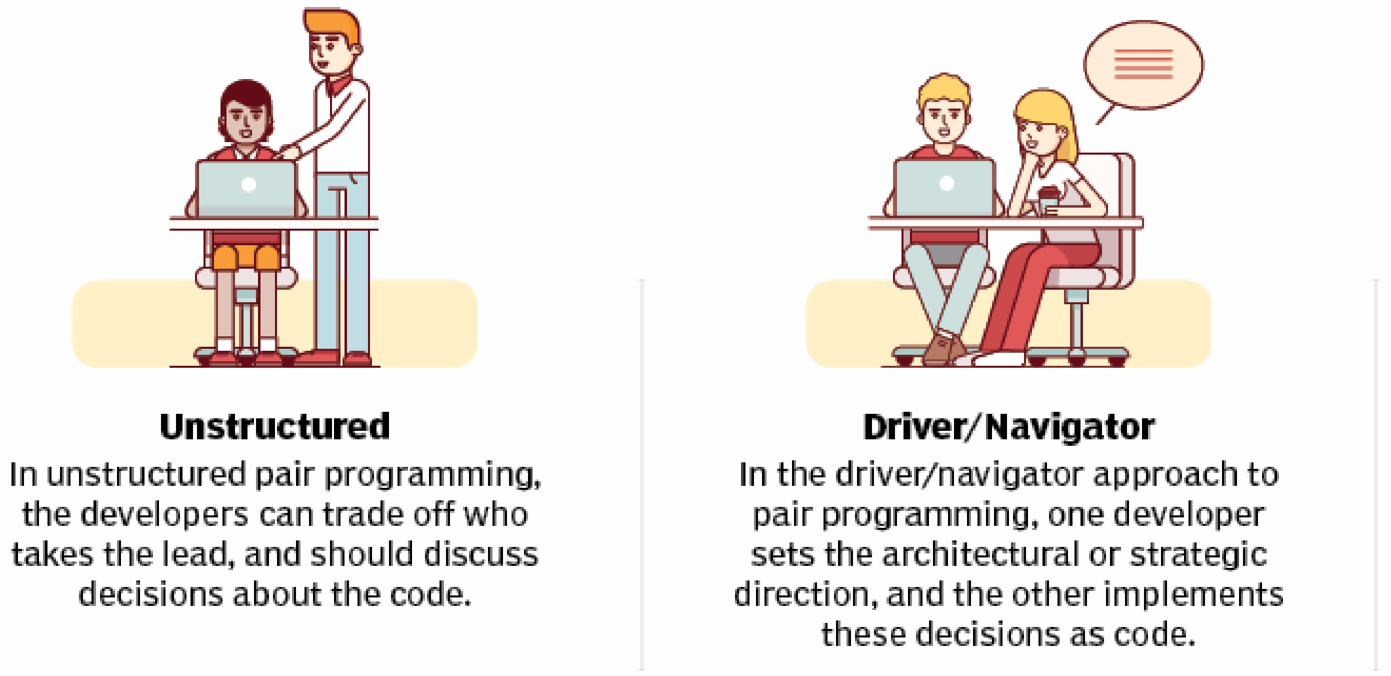
- **Pair Programming**
- Two developers, one monitor and keyboard – working together
- Fosters knowledge sharing, real time code review, increase collaboration

Unstructured

In unstructured pair programming, the developers can trade off who takes the lead, and should discuss decisions about the code.

Tasks done to a high standard

- **Pair Programming**
- Two developers, one monitor and keyboard – working together
- Fosters knowledge sharing, real time code review, increase collaboration



Unstructured

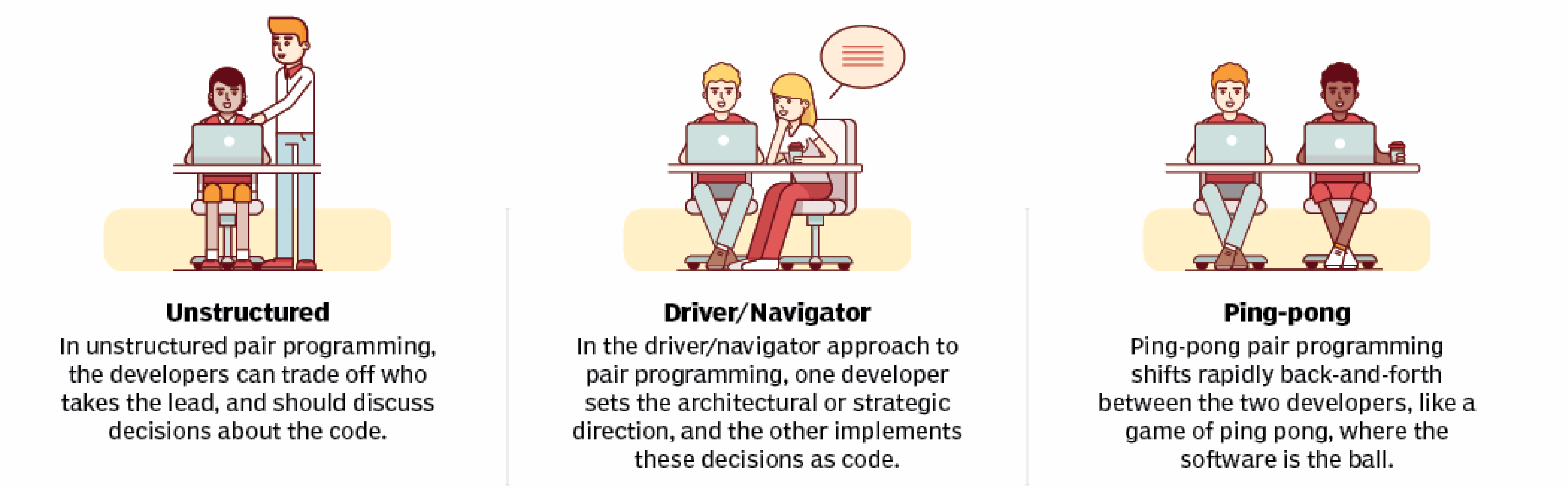
In unstructured pair programming, the developers can trade off who takes the lead, and should discuss decisions about the code.

Driver/Navigator

In the driver/navigator approach to pair programming, one developer sets the architectural or strategic direction, and the other implements these decisions as code.

Tasks done to a high standard

- **Pair Programming**
- Two developers, one monitor and keyboard – working together
- Fosters knowledge sharing, real time code review, increase collaboration



Unstructured

In unstructured pair programming, the developers can trade off who takes the lead, and should discuss decisions about the code.

Driver/Navigator

In the driver/navigator approach to pair programming, one developer sets the architectural or strategic direction, and the other implements these decisions as code.

Ping-pong

Ping-pong pair programming shifts rapidly back-and-forth between the two developers, like a game of ping pong, where the software is the ball.

Tasks done to a high standard

- READ THE MARKING CRITERIA
  - If the marking criteria is too vague, add specificity yourself

Tasks done to a high standard

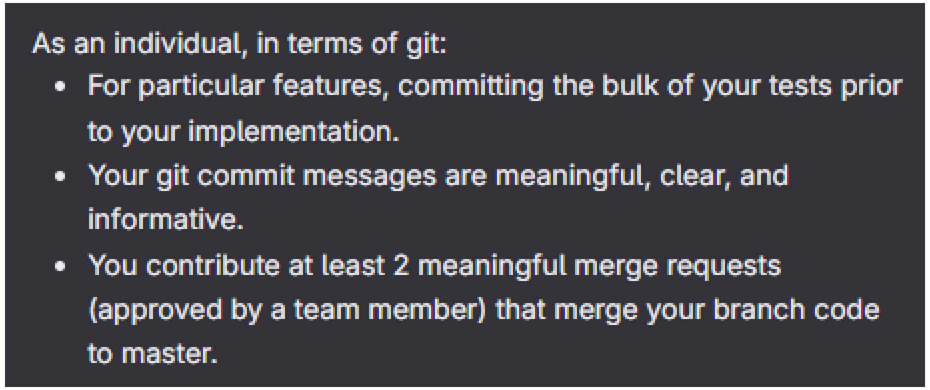
- READ THE MARKING CRITERIA
  - If the marking criteria is too vague, add specificity yourself

- All commits follow some format e.g.

  “test: added login tests”

- All branches start with iteration e.g.

  “iter1/adminAuthLogin"



As an individual, in terms of git:

- For particular features, committing the bulk of your tests prior to your implementation.
- Your git commit messages are meaningful, clear, and informative.
- You contribute at least 2 meaningful merge requests (approved by a team member) that merge your branch code to master.

git Marking Criteria

Applying this advice to COMP1531….

- How to make sure people get their work done on time
- How to make sure tasks are done to a high standard
- How to be a **better contributor**
- How to distribute tasks fairly
- How to prevent arguments
- How to make decisions as a group
- How to run a meeting that isn’t a complete waste of time

How you can improve your technical skills

- **Understand what you find hard**
  - "If you are unable to understand the cause of a problem, it is impossible to solve it."
- **Break problems down into smaller pieces**
  - “Any problem can be solved when broken… and attacked one small piece at a time”
- **Understand how concepts fit together**
  - Track the flow of your program from beginning to end, each line it executes
- **Try it yourself and then ask for help**
  - Don’t stay blocked/stuck for too long
- **Read others code**
  - Be proactive in learning from others (pair programming)
- **Practice makes better, perfect is an illusion**
  - Done is better than perfect. Something is better than nothing.

When things go wrong?

When things go wrong?

- Internal group issue? E.g. unequal contribution, merging bad code, rude

When things go wrong?

- Internal group issue? E.g. unequal contribution, merging bad code, rude
  - **Communicate** the issue clearly and politely with your group members
  - **Work together** to solve it
  - If responses are not constructive, contact your tutor privately

When things go wrong?

- Internal group issue? E.g. unequal contribution, merging bad code, rude
  - **Communicate** the issue clearly and politely with your group members
  - **Work together** to solve it
  - If responses are not constructive, contact your tutor privately
- Disappearing group members?

When things go wrong?

- Internal group issue? E.g. unequal contribution, merging bad code, rude
  - **Communicate** the issue clearly and politely with your group members
  - **Work together** to solve it
  - If responses are not constructive, contact your tutor privately
- Disappearing group members?
  - Message them on MS Teams asking them for their whereabouts
  - If they don’t reply in 48-72 hours, continue as if they won’t reappear
  - If they reappear and have lost the opportunity to work – that’s on them
  - Avoid putting yourself in life-or-death situations waiting for a reply within 48 hours

When things go wrong?

- Internal group issue? E.g. unequal contribution, merging bad code, rude
  - **Communicate** the issue clearly and politely with your group members
  - **Work together** to solve it
  - If responses are not constructive, contact your tutor privately
- Disappearing group members?
  - Message them on MS Teams asking them for their whereabouts
  - If they don’t reply in 48-72 hours, continue as if they won’t reappear
  - If they reappear and have lost the opportunity to work – that’s on them
  - Avoid putting yourself in life-or-death situations waiting for a reply within 48 hours
- Communicate clearly with your team and your tutor.

When things go wrong?

- Internal group issue? E.g. unequal contribution, merging bad code, rude
  - **Communicate** the issue clearly and politely with your group members
  - **Work together** to solve it
  - If responses are not constructive, contact your tutor privately
- Disappearing group members?
  - Message them on MS Teams asking them for their whereabouts
  - If they don't reply in 48-72 hours, continue as if they won't reappear
  - If they reappear and have lost the opportunity to work – that's on them
  - Avoid putting yourself in life-or-death situations waiting for a reply within 48 hours

- Communicate clearly with your team and your tutor.
- NOTE: Tutors WILL check for contribution via peer evaluation, scripts, git

Contribution & Peer Evaluation

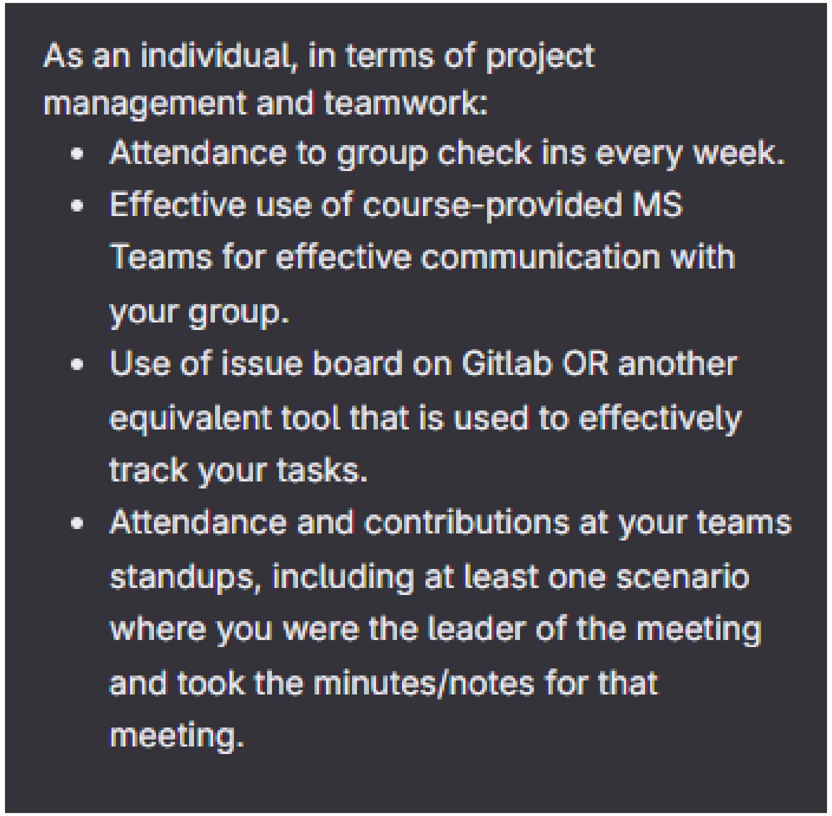
- Peer Evaluation
  - Participation
  - Dependability
  - Team Wellbeing
  - Work Contribution
- Reviewed by your tutor

Do you need to do everything we've mentioned?

No!

No! Marking Criteria is Flexible

- The marking criteria expects a minimum standard of
  - meeting notes/minutes
  - issues board
  - standups



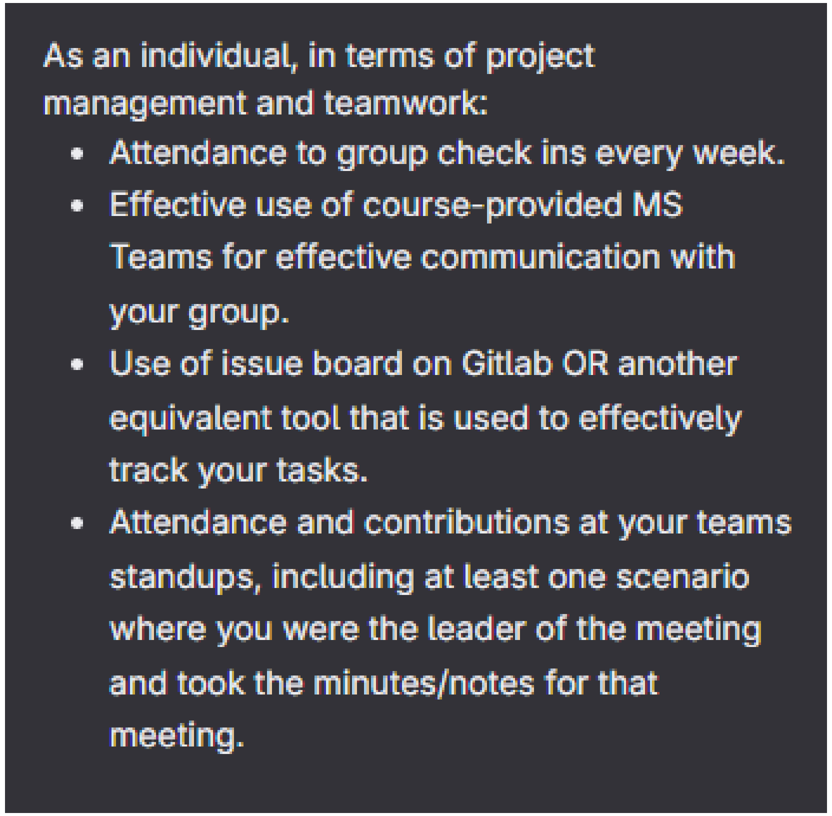
As an individual, in terms of project management and teamwork:

- Attendance to group check ins every week.
- Effective use of course-provided MS Teams for effective communication with your group.
- Use of issue board on Gitlab OR another equivalent tool that is used to effectively track your tasks.
- Attendance and contributions at your teams standups, including at least one scenario where you were the leader of the meeting and took the minutes/notes for that meeting.

Project Marking Criteria

No! Marking Criteria is Flexible

- The marking criteria expects a minimum standard of
  - meeting notes/minutes
  - issues board
  - standups
- Your group might impose additional standards
- Pick and choose relevant strategies effective for your group



As an individual, in terms of project management and teamwork:

- Attendance to group check ins every week.
- Effective use of course-provided MS Teams for effective communication with your group.
- Use of issue board on Gitlab OR another equivalent tool that is used to effectively track your tasks.
- Attendance and contributions at your teams standups, including at least one scenario where you were the leader of the meeting and took the minutes/notes for that meeting.

Project Marking Criteria

Questions?

Questions from the form

- Communicating in a 2nd language?
- Communicating with an age gap? Or knowledge gap?
- Reconciling different individual goals? e.g. HD vs Pass

Thanks for Listening!

End.