

HOW TO WORK TOGETHER

without killing each other

STAND UP

A STAND UP A DAY KEEPS THE MERGE CONFLICTS AWAY

- **Goal: keep the group updated on each other's progress, organize each other efficiently, and help remove each other's blockers**
- **Should be short (~15 minutes or less)**
- **Each person in round robin style:**
 - What did I do yesterday?
 - What am I doing today?
 - Do I have any *blockers*?
- **Blocker: something *outside of your control* that is preventing you from making progress on the thing you're doing today**

- **Alice:** “Yesterday, I refactored most of the product and orders routes, but there’s still a bit left to do. Today, I’m going to finish those routes. I don’t have any blockers.
- **Bob:** “Yesterday, I finished scaffolding the Orders page with dummy data. Today, I want to make the UI interact with the live data, but my blocker is that I need Alice’s backend changes to be merged before I do that.
- **Alice:** “Good to know - the orders changes are ready, it’s just the product routes that need fixing. I can make a pull request right after this, if you can review it.

GIT'ING STARTED

- **Create a Github organization**
- **Add members and give them appropriate access**
 - Everyone should be an owner
- **Create a repo**
- **Push an initial commit**
- **Protect the master branch (Settings > Branches)**
- **Create a project**
 - Add at least three columns: *To Do*, *In Progress*, and *Done*

WRITING ISSUES

USER STORIES

- *As a <persona>, I want to <do something> so that <reason>*
- Software should be designed based on what the users of that software want
- Stories !== implementation details

IMPLEMENTATION DETAILS

- **How you will actually fulfill a user story**
- **Should always serve a user story**
- **Typically, you will write a ticket for each implementation detail (not for each user story)**



Write a route to serve up all the products



As a visitor, I want to see all of the products,
so that I can choose one to get more details or
add one to my cart right away



Story: As a visitor, I want to see all of the products, so that I can choose one to get more details or add one to my cart right away

Implementation: an Express route on the backend (GET `/api/products`) should serve up the name, price, quantity, and imageUrl of all the products from our Postgres database. No special access control is needed, since any visitor should be able to receive this information.

DOING WORK

FEATURE BRANCHES

- **Never work directly on master!**
- **Always work from another branch (“feature” branch)**
 - ``git checkout -b my-feature``

MERGING TOGETHER

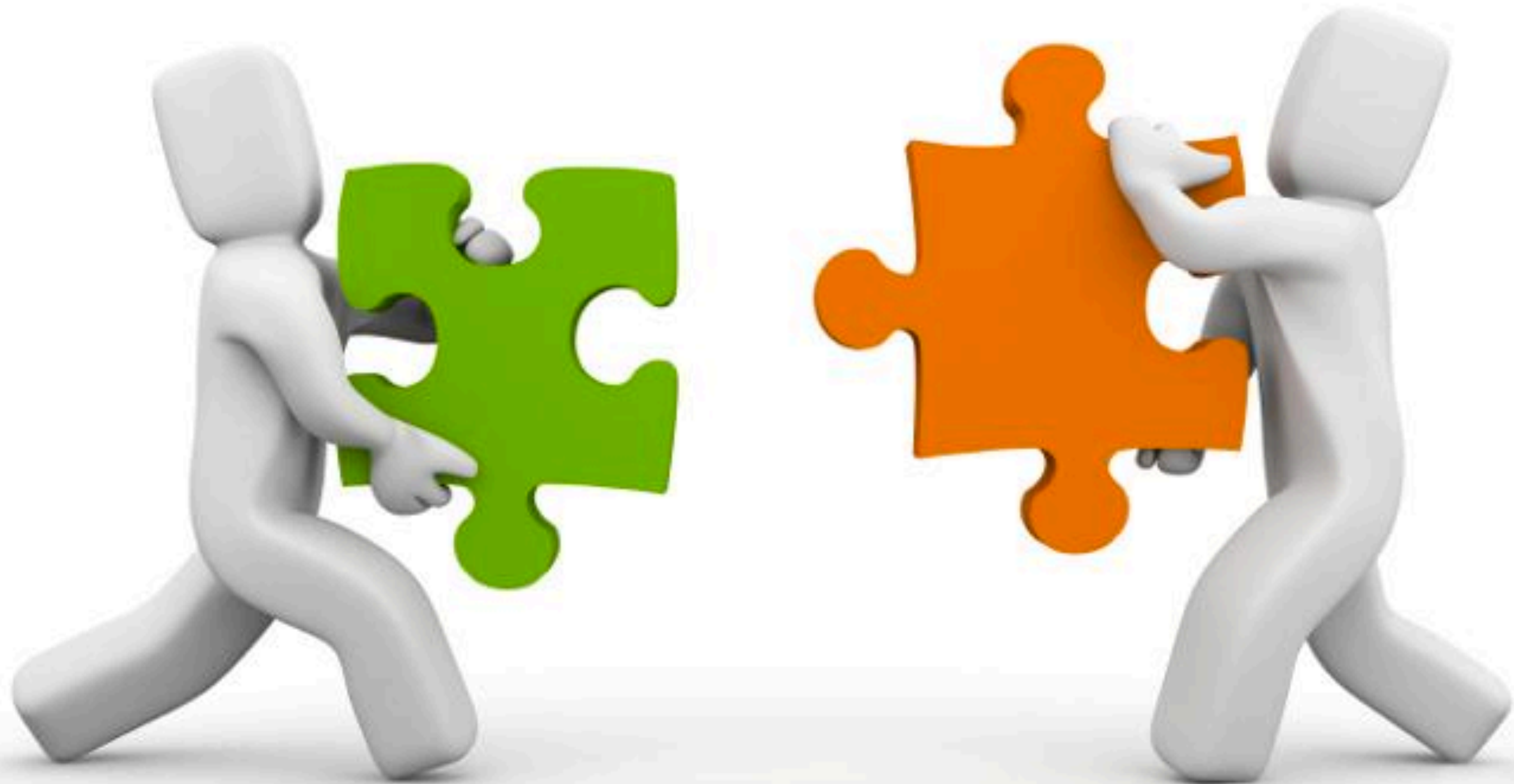
PULL REQUESTS

- **Push your branch up to the remote**
 - ``git push origin my-feature``
- **Go to your repo on Github - you will be prompted to make a pull request**
- **Another member of your team should review your pull request**
- **If changes are requested, make them locally and then push again**

MERGING

- **Once the pull request is ready, merge it on Github**
- **All team members can now update their local master branch**
 - ``git checkout master``
 - ``git pull origin master``
- **If you are currently working on a feature, you should update it with the latest changes from master**
 - ``git checkout my-other-feature``
 - ``git merge master``

WORKING TOGETHER





WHY IS IT HARD TO WORK IN A GROUP?

NORMALIZE

- Establish a set of rules that the group agrees to abide by
- These rules act as a *source of truth* for how the group behaves, manages work, and deals with conflict
- When in doubt: “What are our norms?”

STARTING NORMS

1. **Share the mic**
2. **Don't interrupt - raise your hand**
3. **Share frustrations in a constructive matter**
4. **Write a GitHub issue**
5. **Ask for help**
6. **Commit frequently**
7. **Deploy early**
8. **Choose a code style and stick to it**
9. **Keep each other updated**
10. **Disagreements many, arguments few**

SET YOUR NORMS

- **Each group: take our list of starting norms and add to them based on the way your group wants to work**
- **Answer the questions on the accompanying document**
- **Put them into a NORMS.md file, which will go into your project tracking**

Up next

- First: Lecture on how starting point works
- After that: get set up!
 - Find your assigned team, grab an area to sit together
 - ** Read the workshop **
 - Create a Github Organization and repo
 - Establish your master branch with *boilermaker*
 - ** Read the README.md **
 - Add your NORMS.md to your project
 - Organize the work you have to do with a Github project