**1. Ask why we are here.**

**• This is a quick reminder about why we are here, who our**

**customers are, and why we decided to do this project in the**

**first place.**

Teacher, Instructors

To create a team based on students’ preferences to create teams that are productive and happy.

**2. Create an elevator pitch.**

**• If we had thirty seconds and two sentences to describe our**

**project, what would we say?**

For teachers or instructors, Who needs to place their students in groups/teams. The Happy Teams Software is a program that organizes students into teams based on preferences. Unlike Canvas our product takes into account the student’s preferences and the overall team happiness.

**3. Design a product box.**

**• If we were flipping through a magazine and we saw an advertisement**

**for our product or service, what would it say, and,**

**more importantly, would we buy it?**



**4. Create a NOT list.**

**• It’s pretty clear what we want to do on this project. Let’s be**

**even clearer and show what we are not doing.**

* Abi’s idea of arrays
  + Basically parallel arrays that were way too complicated ( I tend to complicate programming assignments)
* Rely on one person
* Procrastinate (even though it is in our name)

**5. Meet your neighbors.**

**• Our project community is always bigger than we think. Why**

**don’t we invite them over for coffee and introduce ourselves?**

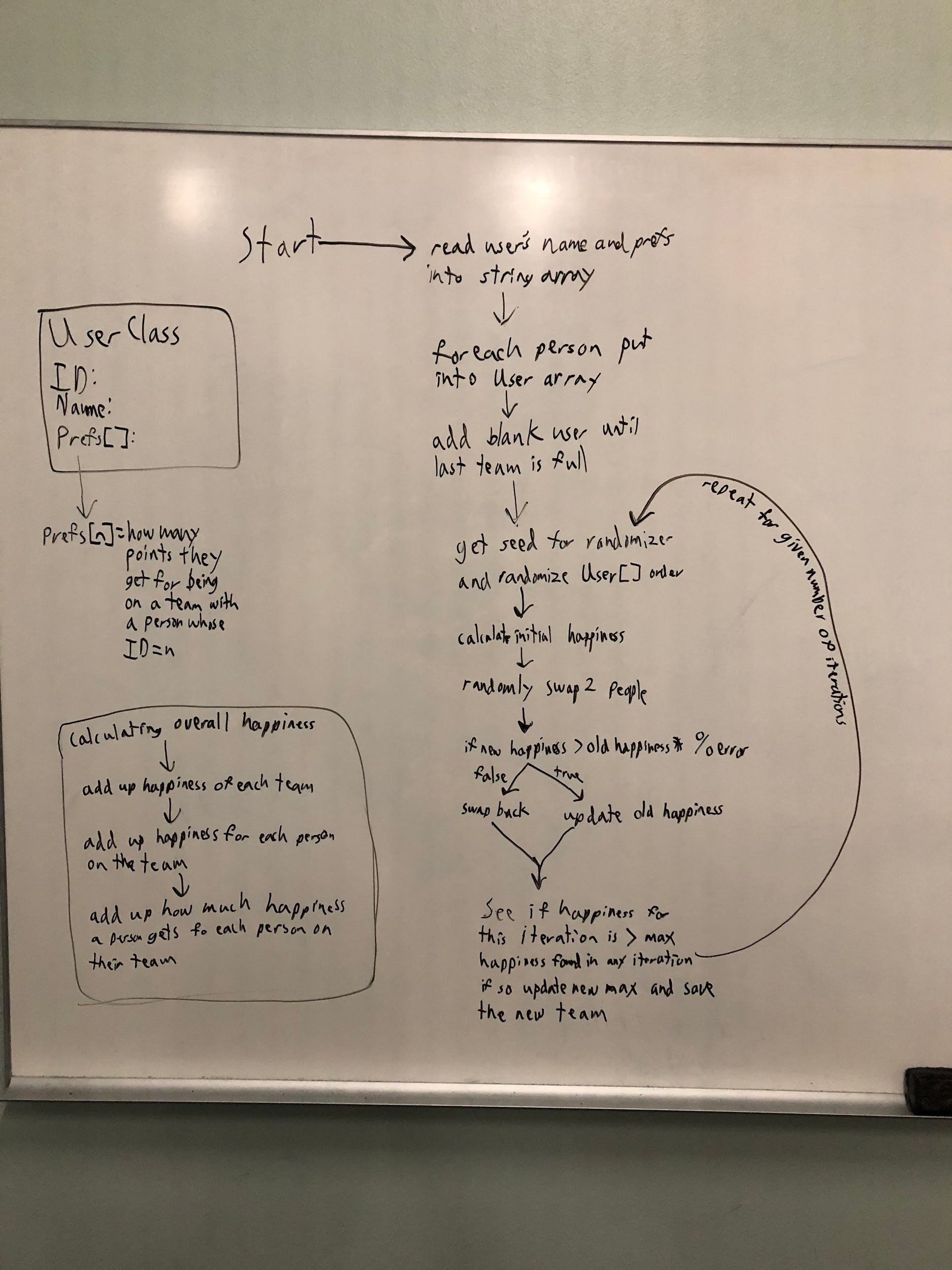
Ben, Sam, Reeves, other teams with the same project.

We introduced ourselves and also we (the class) have created a community on Group Me for when we need help!!

**6. Show the solution.**

**• Let’s draw the high-level blueprints of the technical architecture**

**to make sure we are all thinking of the same thing.**



**7. Ask what keeps us up at night.**

**• Some of the things that happen on projects are downright**

**scary. But talking about them, and what we can do to avoid**

**them, can make them less scary.**

WORK ON IT, pulling all-nighters

Having trouble understanding the assignment

Solution:

Preparation

We just need to talk through with reeves, ask questions in class to reeves.

**8. Size it up.**

**• Is this thing a three-, six-, or nine-month project?**

A couple of weeks max

Week one:

Random swap

Make sure it works

Week two:

Happiness preferences

Make sure it works

**9. Be clear on what’s going to give.**

**• Projects have levers like time, scope, budget, and quality.**

**What’s most and least important for this project at this time?**

**Most:**

Fulfilling the A proposal

* t - team size, minimum 2 maximum (class / 2)
* v - verbosity level 0 - 4 information / debugging output
* n - number-of-swaps to attempt
* l - number of times to perform N swaps
* r - percent of sub-optimal swaps allowed (2 means 2 %)

**Least:**

* The way it looks
* Working the most efficiently
* Anything not in the proposal

**10. Show what it’s going to take.**

**• How long is it going to take?**

**How much will it cost?**

**Time and Energy and whatever Software Engineering Cost**

And what kind of team are we going to need to pull this off?

Understanding

Communicating team

Responsible

Reliable