(intro to class) We are going to do something that is not usually done in entrepreneurship teaching, and that is merge CS and business for the customer exploration part of the project. The goal of today is not to find solutions to your area / field directly, but to get at your solution from the perspective of the customer / consumer and UI / UX.

Clarification on the use of the word "customer" - what we mean is customer / stakeholder / user. The person who is being targeted with your solution. Customer is a business term, but we don't only mean customer in terms of business.

(15 minutes for activity) Finalize your problem / customer segments. The goal is to figure out in general who your potential customers are - if there are multiple potential audiences, great. Just keep in mind that today we will be working very strongly on customers.

Goal for the end of today: really truly understand your potential customer segments.

(3 minutes per group) Share your areas / fields / arenas; major problems; potential customer segments (does not have to be very detailed).

## UI vs. UX

INTERFACE and EXPERIENCE

- interface and the experience of a product / solution make or break the product

- sometimes the experience IS the product / solution

- Filled with examples, and I want to give you things to think about

- GetTaxi experience is different than the hailing a cab on the street experience

- This product is not about the interface, it is about the experience

## Principles of UI / UX

1. **Learnability** - is it easy to use?

2. **Efficiency** - once you learn it, is it fast to use?

3. **Safety** - are errors few and recoverable?

4. **Aesthetics** - satisfaction / happiness of user?

5. **Ergonomics** - is it comfortable / exhausting to use?

- "textbook" UI / UX principles. There are more, but they can be broken into these 5 categories

- Which ones you care about matter on the application. For example: spaceshuttle launch, subway turnstile, experts, novices

## Techniques

- These are by no means comprehensive techniques. This is more to serve examples, and there are a million ways to do something, and you can always invent your own. These are just to give you things to think about

- Metaphors (but don't force it)

- What people know is easy for them to understand

- Makes it easy to learn

- Consistency

- Stay close to what people know

- related to metaphors

- If you are doing something radical, your UI has to at least be something close to what people know

- The same applies for bootstrapping your ideas to something else. For example, if your idea is to make a game to play with others, maybe a Facebook game is easier for people to use than remembering to go to another new social networking website

- "Affordance"

- Give your users clues about what is happening (this is not related to the age or demographic of your users at all. this is just good for everyone)

- User feedback. Something should change in 100ms, or the user will notice that nothing happened and get frustrated

- Angry users are not good users

- Direct mapping to something physical or known

- related to metaphors, but more direct

- People who are used to using something in the physical world will want to use this more

- Gamification

- Make it fun

- especially important for younger users, or users who have a lot of things thrown at them

- Can help you stand out

- Can help with users coming back

- Competition is good too

- This can be a whole separate lecture, but basically make it fun

- Look for your own examples of good UI / UX

- Don't mimic, take what is good and build on it

## Design process

- What kinds of users do you have?
  - computer experience, age, motivation, goals, education, language, age?
  - YOU are not the user (in most cases)
- Prototype early, prototype often
- Paper prototypes
- A / B testing

## How do you know what the user really wants?

- Telephone handsets were heavy and users were asked "what do you want the handset to weight" but when the user testing was done, it turned out that they actually wanted different from what they said

## Understanding the user

- Common mistake: describing what you WANT your users to be, rather than what they are

- **The user is always right in behavior**. If they fail in doing something with your interface, the interface was not easy to use. They are not good at giving suggestions, but they are never wrong in their reactions.

**WHO** is your user

**WHEN** is it used

**WHERE** is it used

**WHAT** features are a must / what is optional

(break)

(30 minutes) Identify customer COMPREHENSIVELY and create a CUS-TOMER PROFILE. For each possible customer segment, make sure to answer the 4 questions above in as much detail as possible. But for the WHO especially. Old/young/internet literacy/language/income/education/car owner/pet owner/where do they go regularly/special characteristics. As you are doing this, think vaguely about what kinds of solutions are better for them: mobile, web, integration to a currently-existing product (e.g. FB game, car bumper sticker)

(rest of time, depending on how much there is left) Now that you have a comprehensive customer profile, pretend that you are building a feature / app / website / whatever for them. Prototype this on paper, and role-play small features. Each group should have 2 paper prototype copies, and pair off half-groups with each other to test each other's product. Each tester group should be given a comprehensive customer profile, and understand the customer segment / problem / area. Give feedback pretending you are that customer segment. Testing team takes comprehensive notes on how the person uses their product / whatever. Later, will take this feedback and integrate it into their next-version prototype.