

SkillTrader - Project One

Andrew Nelson, John S Hamilton, Jithu Maliakal, Michael Schueth

Introduction

People tend to have problems, and usually they're easily fixable. For these problems, there often exist other people who already have some skills in solving these problems; they may even have problems of their own. Unfortunately, the availability of these people who solve problems are obfuscated by a lack of networking, or these people are professionals who tend to take on bigger problems than ones many people have such as fixing a car, learning to ski, or getting some help with jQuery.

SkillTrader aims to connect these people, who may either want to share their skills or seek help from others. This could be accomplished by trading skill for skill, skill for money, or just being someone pursuing a hobby who expects nothing in return. An important element in SkillTrader is that the users are facilitated rather than guided by our service.

For example, a professional painter may be having car trouble - nothing major, but the painter is definitely not experienced with what goes on under the hood and can't afford a \$100 estimate by a mechanic. Meanwhile, a car buff who has lovingly tended to his '95 Impala for years is looking for a creative field to break into. These two strangers have a goal but neither goal is quite pressing enough to consider spending a great deal of money anytime soon. Instead, they could join SkillTrader to network with others with skills such as these and find their answer.

We are taking on a market populated by people who have problems and seek solutions without worrying about dollar cost. Allowing the users to take the major role in the service as both the users and providers of the skill commodity is placing a great deal of trust on them; however, services like Craigslist have proven it sustainable and fairly reliable. The people we help are seeking skills that other people have already developed, allowing them to learn something new, disseminate or sell their knowledge, or just look for some personal connection over an interesting activity. It's a fairly human thing to do, and we're making it easier.

Requirements Gathering/Results

Requirements from Brainstorm

- log in / log out pages
- account creation page
- user / job profiles
 - name
 - contact info
 - picture
 - skills
 - short bio / description
 - city / location info
 - way to hide particular fields from other users if desired
 - amount of time available / required
 - date job needs to be done by
 - rating
- navigation menu
- Skills listing
- Job listing, with required skills
- skill keywords
- way to add skill keywords
- way to have skill keywords approved and regulated to prevent redundant key words
- way to group related key words (skill tree? Coder has the sub categories Java, C.. etc)
- way to add jobs
- way to search for jobs within **x** distance, with **y** minimum pay, and with **z** skills
- way to search for skilled people willing to travel **x** distance, work for **y** pay, and with **z** skills
- way to contact other users (personal messages / email system / phone numbers)
- way to rate and review users based on performance for particular skills
- way to enforce transfer of payments on completion
- way to flag users for various reasons (e.g., safety, fraud)

Requirements from Interviews

We are designing the interface for a Web service that connects people who have skills (java programming, bike repair, intimate knowledge of north Georgia wildlife, cooking) and people who need work done with those skills.

"worker" users post what skills they can offer and post their availability. "employer" users can post requested skills. The interface helps pair them up.

If you were to use such an interface, what would you want it to have?

Paul Schueth

- accuracy with a match, speed, options, ability to fine tune the search, contact information, click now to contact, etc

Ekriirke (online profile name of someone I interviewed)

- Outside of simple keyword searching, I'd make an extensive tree of subjects that people can drill down their expertise(s) on.
 - Computers > PC > Windows > Servers
 - Computers > PC > Windows > C Dev
 - Environment > Wildlife > Avian > Flamingo
 - Food > Dinner > Italian

David Perez

- Interface for the smart phone that will notify users of job requests
- credentials in the workers' profile

Blake Palacio

- categories for the skills

Kate Hoyt

- I'd want it to be easy to withhold my identity during the pairing process but easily share contact information (name, phone, location, email?) after an initial pairing has been made.
- A system of messaging within the website so that personal email doesn't have to be used would be nice, but with something that forwards messages to my email so I don't have to constantly check the website for results.
- During the "worker" information input, putting in my regular location plus an "I'm willing to drive ___ miles" if it fits the type of work.
- During the "employer" service requesting, a location if appropriate, and when searching for workers only provide ones that have said their locations and their willing to drive mesh correctly.
- Set up multiple options for pay, such as both having paypal accounts already or an easy agreement for paying upon completion in person.

Current UI Critique

There currently exist two analogous methods through which individuals can acquire the services of another: a digital bulletin board and a non-digital bulletin board. Both have usability problems caused by different aspects of each method.

Digital Bulletin Boards

The best example of a digital bulletin board, and probably the most widely-used system today, is Craigslist. While Craigslist is simple to use for both skill listers and skill seekers, there are a number of problems for each party when it comes to using the site.

One such problem is that service posters have to compete with a large number of other service posters that may advertise rates that are hard to compete with, even if these ads are not completely legitimate or if the quality of work is low. Ads are listed in order of post date and time, so postings that are not renewed frequently can quickly be driven to the bottom of the list.

For those seeking services, there exist many problems when trying to find the ideal skill lister. Because there is no rating or review system, finding a legitimate, competent lister becomes much more difficult. Service seekers have to manually sift through many ads without a way to sort them. Searching through ads isn't always feasible because the price posted by a service lister does not always correlate to an hourly price or the price that the lister will charge for a specific job.

There are also issues that affect both skill seekers and skill listers. Because Craigslist is completely free and there are minimal barriers to creating or replying to an ad, safety is a big concern for all users. The lack of a publicly available profile also reduces accountability for both parties.

Non-Digital Bulletin Boards

Examples of non-digital bulletin boards include actual, physical bulletin boards and publications like the Yellow Pages. Both are similar in that they are simple ways for skill listers to spread their contact information. The main differences between the two is that a bulletin board is a free system for a small community of people (e.g., workplace, library, local gym), while the Yellow Pages is a central repository for businesses who have chosen to pay to advertise their services.

One feature of the bulletin board is that it is typically free to list and view advertisements, and there are no barriers to doing either. Another inherent feature is that skill listers are typically located in the same geographical area as the bulletin board is (i.e., skill listers are local). For example, if a babysitter posts an ad on a bulletin board, it is reasonable to assume that the babysitter lives in the same neighborhood or town. While the decentralized and localized aspects of the bulletin board are beneficial in some situations, it can make it difficult for listers to advertise their skills and seekers to find skills. Seekers have to travel to multiple locations

in order to list or update their ads, and listers have to travel to multiple locations to get a more complete view of skill listers. Another feature that a bulletin board lacks is a system for rating and reviewing skill listers.

In contrast to physical bulletin boards, the Yellow Pages serves as a centralized, commercial repository for the listings of skill listers. Neither listers nor seekers have to travel anywhere to interact with listings. Also, because all listers have to pay a fee to be listed in the Yellow Pages, there is a financial barrier to entry that can potentially filter out less legitimate advertisements. A negative effect of this feature is that people who may be willing to provide their skills for free will not use this service to advertise. Additionally, the Yellow Pages book is a large, bulky book. This causes the user problems because it needs to be stored and because it is slow to navigate.

User Characteristics

Nicole - primarily provides skills

Every day, Nicole comes closer to finishing another prototype for mobile strategy games. Inspired by the Foursquare-based Risk-like battle in New York City last year, she wants to build more games that use players as pieces in a massively multiplayer strategy game. Her experience in game design is highly prized among her project groups at Zynga, where she's worked since graduation, earning her design lead positions for most projects assigned to her. In her free time though, she builds her own games in java for Android phones, hoping to open her own game business with these mobile strategy game ideas. When she hits a roadblock in her design process, she joins local game development jams to find inspiration and often meets friends from TAing for the introductory java class at Georgia Tech, her alma mater. However, in order to continue her work she must eat, and there's only so much boxed, bagged, and premade foods her stomach and life span can handle in a week. Thus she has taken up the formidable task of learning to cook.

David - primarily seeks skills

Growing up in a family so picky that a single drop more of sesame oil in the main dish could sway them to only eat rice for their meal, David is frustrated with eating the same five Cantonese dishes his family makes every week. In the last few years, he has started cooking late at night for his lunch to take to school the next day, preparing dishes from related Korean and Japanese cuisines. Now a freshman Computational Media major at Georgia Tech and the sole cook in an apartment of four, he has used his self-built culinary knowledge to bravely fuse different cuisines into a new dish for his roommates to test, usually a success. Of course he could have gone to culinary school, but that's too much effort to sink into what's more of a hobby. Instead, he came to Georgia Tech, where his intro to java programming class is giving him serious trouble. His inconvenient class times preclude him from attending most office hours, tutoring sessions, and jobs that he could use to pay for a tutor.

Constraints

- Constraints of the System
 - Alerting people of when they are wanted/needed
 - Getting the system “off the ground”
 - Generating the skill tree data required for the system to actually work
 - Getting enough users and jobs in an area for people to be matched up locally
 - Implementing an effective, easy-to-use search within the scope of the user-skill database
- Constraints on Our Implementation
 - Deadline (Only 1 semester to work on the project)
 - Available Spare Time (Balancing other classes)
 - Scheduling Conflicts (We have little opportunity to meet with each other)
 - Incomplete technical knowledge regarding the implementation of our system (we don't know everything we will be required to know to make this work)
 - Lack of a monetary investment for the server required to run everything
 - Short amount of time to gather requirements
 - Unclear instruction as to how thorough requirements need to be

Implications

Based on the requirements we have ascertained, and the constraints we have come up with, there are a number of things we must account for in our design. We need to make sure that we design the pages people are familiar with in a familiar way, for example login pages or account information pages. We also need to design the system in such a way that it attracts more and more people. When the system first begins, there will likely be many tasks people want done, and not many people to match up to those tasks. As the system spreads, we will have a larger variety of people, and as such, a larger variety of skills can be provided on request. The system also needs to be adaptable so that as new skills are introduced, they can become keywords, or suggested tasks, so that people will think to list more than simply their day job as the only role they can fulfill. This project is about requesting and providing services that are NOT part of the usual daily employment. We also need to consider adding in potential texting or other alternative forms of communication services, so that people can be reached without needing to check their email every 5 minutes. Not all people have smartphones with email delivery, and not everyone with a computer is in front of it constantly with their email open. We do hope to have a working prototype by the end of the semester, but that will strongly depend on how much time we can devote without pulling away too much time from our remaining classes

Requirements Summary

After gathering some of the system requirements, it has become apparent that SkillTrader is going to require an efficient and dynamic data structure to help drive the interface.

SkillTrader will need ways to sort and group skills, a way to search based on multiple constraints with user-entered data, a way to determine the distance someone must travel to get to where the job's location is, a way to filter for jobs that someone would be interested in, and a way to filter for people that would be able to accomplish jobs.

We will need to provide users with a way to advertise themselves and/or the jobs they want completed. We will also need to make sure people can trust each other since a job is likely to be the first time users will be meeting each other. We will need to find a balance that users find acceptable between protecting people's personal information and connecting people for jobs as well as providing enough information for users to trust each other. We will need a way for users to include any qualifications they may have for a particular skill, and a way for the users to rate each other on how skilled they are as well as how well the experience as a whole went. We will also need a way to flag users for misusing the SkillTrader system, and a way to warn other users what they did wrong.

We will need user profiles that are structured in a way that makes it easy to find the information you are looking for, yet open enough that users can provide additional information and draw attention to themselves without cluttering the profile page.

We will need a user to user message system to help people communicate with one another while withholding contact information. Users should have the option to turn email forwarding of messages on or off. We may also need a way to notify users remotely (on a smart phone) when someone has requested to do a job, or requested them to preform a job.

We will also need a way to transfer payments on completion. This can be with paypal, check, credit card, or anything that will generate a paper trail. The employer will then need to submit proof of paying the employee so we can have a record if the need arises.

Task Analysis

Our users will be seeking help in a skill they search for, or they will be posting a skill that they would like to offer. Generally, the users would be using any device with an internet connection, including desktop, mobile, and tablet computers. Since immediate feedback is not an issue, and the user will likely not be using it on-the-go, mobile optimization is not quite necessary; therefore, environment issues are less of a factor in the design. The only environmental variable that may be important is accessibility options and affordances for handicapped users; for example, a visually impaired student may offer tips on accessibility design in exchange for tutoring in physics.

Task Decomposition:

1. Main/landing page - user determines whether to find or post a skill
 - a. Links to about/how-to page
 - b. Includes a brief overview of the system
2. If user decides to find help with a skill:
 - a. Main page - uses the search function to narrow down list of users or scrolls through and reads list frame in current state
 - i. If search:
 1. Inputs search terms to search box and submits to regenerate list frame below (stays on main page url, the only change is the list)
 - ii. If scroll:
 1. Seeks a user's skill listing on list frame and clicks it to take further action on user page, such as reviewing details or contacting
 - b. User page - shows the selected user's full listing of skills offered and personal information. After the user moves to a particular user page from search results, it should load the page and automatically scroll to the searched-for skill via anchor
 - c. User chooses whether to contact.
 - i. If not, the user should return to the main page with the list frame
 - ii. If yes, user contacts skill-offering user with info provided on user page. This is an end condition to the task flow. We can assume the user has found what they are looking for and can close the page or make their own decision on what to do next without suggestion.
3. If user decides to post a skill they would like to offer to other users:
 - a. User navigates to their personal user page (login) or registers
 - b. Once on their personal user page, they may enter the following:
 - i. What skills to offer
 - ii. Personal contact details
 - iii. Privacy scope of skill offering (who can see what, how it's framed)
 - c. This is an end condition to the task flow. We can assume the user has found what they are looking for and can close the page or make their own decision on what to do next without suggestion.

Usage Scenarios

Scenario 1 - Nicole learns to cook, trades for help with Java

Nicole leaped over couches and leaned over tables to push the windows open for fear of another smoke alarm going off in her apartment this year. She glanced forlornly at the casualty: tonight's (almost) dinner, char siu. Too much marinade left on the pork as it sizzled under the broiler caused the liquid to pop around the oven and burn as it met the glowing grills, eschewing a puff of smoke every second and building inside the closed oven. It probably would have been helpful if she hadn't been paying so much attention to a new game she was developing and more on the edible project with the messier failure condition smoldering in the oven.

Fortunately for Nicole, she can find more help with cooking from those who actually have levels in the skill. SkillTrader users can find people with desirable skills and, in return, offer their own skills as compensation. Nicole registers with her Google account, joins the Georgia Tech and Atlanta area networks, and builds her profile with several skills that she's confident in teaching or at least providing some pointers for: game design, a few programming languages, tutoring for CS 1331 at Georgia Tech, and training for the game League of Legends. Also, being a League of Legends player for several years, she's not going to train some level 5 kid! Her services are premium, so she marks it as available by invitation only; like any scrub is going to get her supreme gaming insights. Some of the skills match common tags in the system, so that should help her be discovered more; however, it seems there aren't any tags to match her League of Legends training. Instead, she changes the skill offer to the generic "Gaming" tag and adds details to the "More Info" box for that particular skill.

With her profile configured, Nicole returns to SkillTrader's main page to search the skill marketplace for a worthy cooking mentor. The trends pane shows there's been a spike in trades for personal fitness and dieting; it is right after the winter holidays, so it makes sense. Maybe her poor cooking record and resulting starvation has helped stave off the extra pounds this time around. Submitting "cooking" to the search box directs her to a long list of users with cooking as a skill they've listed, with two figures by their name: the number of reputation score for their cooking skill and whether they're available for that skill now. The list, sorted by reputation, includes people from way outside her acceptable walking distance, so she adds a sort option for only Georgia Tech network users. "Aha!" she thinks, seeing a few users who dominate the area with their reputation score; however, he's more interested in Cantonese cuisine, and these guys probably just teach the basics to all the cooking newbies. She adds "(Cantonese || Chinese)" to the search to yield a couple promising users with the potential to teach her how to make some of the most delicious food she knows. Clicking on the user with the best reputation, the user's page opens and automatically scrolls to their cooking listing. Indeed, this guy, David Chi, offers several East Asian cooking styles and he's even from Hong Kong! He probably knows his stuff.

Nicole clicks the "Hire Me!" button next to his cooking listing, adding a few quick notes about what she's interested in and when she's available before submitting the request. David receives an email containing Nicole's message and what skill she's referencing. He doesn't see it yet, though, as he's agonizing over tonight's CS 1331 assignment. If only he had some help!

Scenario 2 - Joe needs to make some money to help get through college

Joe is a computer programmer still in college. He has good programming skills, but since he has not yet graduated, he does not have a degree that proves it. Being in college, most of Joe's

friends are also in college and are in a similar situation making it hard for him to reach out and find some work. Joe has a demanding schedule with time consuming projects and can not commit to a part time job with scheduled work hours, but he has enough time here and there and does not mind staying up late at night to finish doing his work.

Seeing a SkillTrader advertisement on one of the websites Joe frequently visits, he decided to check out the website. He doesn't know if he will be able to make much money, but since making a profile doesn't take too much effort he goes ahead and fills out the information. Joe adds some skills he has and runs a few searches for jobs he may be able to perform. Much to his surprise some one is looking for some web development. The job does not require him to actually show up, and just asks that the site be completed within a couple weeks. It is not quite what Joe wants to do for his career, but he is knowledgeable enough to be able to do the work and it doesn't require him to work specific hours. Seeing no reason not to make a little extra money on his own time, Joe takes the job. He produces a great website for the small company and receives a little extra pay and a great rating on his profile. Surprised by how easy it was for him to make some money on his own time he decides to keep looking for more jobs he can do. With his rating improving with each job he completes, Joe finds it easier to get better paying jobs that he can complete on his own time, and is able to build up some references for when he eventually graduates and seeks a full time job.

Scenario 3 - Chris provides handyman services for money

Chris works full-time as a software engineer for Google in Midtown Atlanta. While he enjoys his job there, he is spending increasing amounts of time on a personal software project. Eventually, he'd like to leave his position and spin this project off into a startup company. While he's saving as much money as he can from his job at Google, Chris is trying to bootstrap his startup and is looking for ways to earn additional income.

He's a very handy person, and in high school and college, he used to work for his dad, a licensed electrician and plumber. Chris has always helped his friends and coworkers with their home maintenance issues for free and has tried to market these skills on Craigslist, but the only people who seem interested in his services are scam artists.

After Chris discovers SkillTrader, he registers for an account. In his profile, he lists that he can offer electrical services, plumbing services, and handyman services, and he sets an hourly rate for these services. He lists his availability and his location and he chooses how far he'd be willing to travel. In addition, he types his contact information, uploads a picture of himself, and sets his availability. He finally saves his profile.

After a few days, Chris receives a request from someone with a leak underneath their sink. Communicating via SkillTrader, Chris agrees to a time to fix the leak. After this, both parties exchange their contact information via the website. The date arrives and Chris has a good experience; he performs the repairs quickly and gets paid for the work he's done. Apparently, the other person also had a good experience as well, as he gives Chris his first positive review.

Scenario 4 - Lauren needs someone to fix her dishwasher

Lauren has a successful career in finance and works for Wells Fargo in Atlantic Station, where she also lives. She has a very active lifestyle and spends a minimal amount of time in her one-bedroom apartment where she lives. An avid triathlete, she swims, bikes, and runs when she's not working in preparation for her upcoming Ironman Triathlon race in Panama City Beach, Florida. She also frequently attends concerts and goes to bars with her tight-knit group of friends.

Recently, she noticed that the hardware holding her dishwasher in place has come undone, causing her dishwasher to tip forward every time she opens the door. As she only has some basic home maintenance knowledge, she doesn't really know how to fix this problem. Because of her hours at work and her strict training schedule, she's finding it impossible to hire a handyman who will come in the evening or on the weekend.

Hearing about SkillTrader from a friend, she visits the website on her laptop and registers for an account. When filling out her profile, she skips the available skills section because she doesn't mind paying for the skill she needs and doesn't think she has anything to offer. She navigates to the search functionality and types in that she needs someone who has handyman skills or skills with appliance installation. She selects that she is willing to pay and types in her zip code. The search results page displays a few skill listers within Atlantic Station and Midtown. She selects the lister with the highest rating, Chris, and fills in the contact form. Within a few hours, Chris responds via SkillTrader. Finally, the two schedule a time to have the work done. After the mounting is repaired, Chris sends Lauren an invoice via SkillTrader. Lauren completes the transaction and proceeds to rate and review Chris.

Usability Goals

Our product has two major users groups: those who **seek** a skill and those who **list** skills. Users will likely find themselves in either group depending on their own goals, however our usability goals differ per group.

Effectiveness

Seeker: Can find a skill listing with minimal effort, helped by thorough search methods and common tagging

Lister: Should not feel uncomfortable about sharing personal information, and can expect to be compensated sufficiently for their time potentially helping someone with their skill

Efficiency

Seeker: Can traverse the site in three steps, listed as letters in the task decomposition section

Lister: Should feel comfortable with creating an account, potentially via OpenID or Google, and can understand what and how information will be displayed

Satisfaction

Seeker: Feels more like they are interacting with people and their profiles rather than a program or website. This can be accomplished by leaving more control to the user and having minimal site interaction necessary to properly use it.

Lister: Has the potential to earn points, such as a reputation or badges system. This could allow the users who list skills to feel more involved with other users. The potential to earn a "Guru" status for those who have the best reputation for a particular common-tag skill could make the user feel more appreciated.

Reflections

Because all members of our group had conflicting schedules, we had trouble finding a time during the week when we could meet to work on this project. Instead, we had to communicate electronically via email and chat, and we used Google Docs to collaborate on this document. None of us have had experienced fleshing out a project so thoroughly in a first iteration, so approaching this document was daunting and overwhelming initially; however, the standardized design process helped us get the project started. Despite this, we are still unsure how much we needed to accomplish at this stage of the project to make later stages easier, and would have liked some examples included in the project assignment. We have since come to realize that we should have tried to meet up on the weekend to avoid schedule conflicts. We could have also coordinated a little better between group members regarding when parts of the project needed to be done so others could review individual work.

SkillTrader - Project Two

Andrew Nelson, Spencer Hamilton, Jithu Maliakal, Michael Schueth

Introduction

People tend to have problems, and usually they're easily fixable. For these problems, there often exist other people who already have some skills in solving these problems; they may even have problems of their own. Unfortunately, the availability of these people who solve problems are obfuscated by a lack of networking, or these people are professionals who tend to take on bigger problems than ones many people have such as fixing a car, learning to ski, or getting some help with jQuery.

SkillTrader aims to connect these people, who may either want to share their skills or seek help from others. This could be accomplished by trading skill for skill, skill for money, or just being someone pursuing a hobby who expects nothing in return. An important element in SkillTrader is that the users are facilitated rather than guided by our service.

For example, a professional painter may be having car trouble - nothing major, but the painter is definitely not experienced with what goes on under the hood and can't afford a \$100 estimate by a mechanic. Meanwhile, a car buff who has lovingly tended to his '95 Impala for years is looking for a creative field to break into. These two strangers have a goal but neither goal is quite pressing enough to consider spending a great deal of money anytime soon. Instead, they could join SkillTrader to network with others with skills such as these and find their answer.

We are taking on a market populated by people who have problems and seek solutions without worrying about dollar cost. Allowing the users to take the major role in the service as both the users and providers of the skill commodity is placing a great deal of trust on them; however, services like Craigslist have proven it sustainable and fairly reliable. The people we help are seeking skills that other people have already developed, allowing them to learn something new, disseminate or sell their knowledge, or just look for some personal connection over an interesting activity. It's a fairly human thing to do, and we're making it easier.

Requirements Summary

After gathering some of the system requirements, it has become apparent that SkillTrader is going to require an efficient and dynamic data structure to help drive the interface.

SkillTrader will need ways to sort and group skills, a way to search based on multiple constraints with user-entered data, a way to determine the distance someone must travel to get to where the job's location is, a way to filter for jobs that someone would be interested in, and a way to filter for people that would be able to accomplish jobs.

We will need to provide users with a way to advertise themselves and/or the jobs they want completed. We will also need to make sure people can trust each other since a job is likely to be the first time users will be meeting each other. We will need to find a balance that users find acceptable between protecting people's personal information and connecting people for jobs as well as providing enough information for users to trust each other. We will need a way for users to include any qualifications they may have for a particular skill, and a way for the users to rate each other on how skilled they are as well as how well the experience as a whole went. We will also need a way to flag users for misusing the SkillTrader system, and a way to warn other users what they did wrong.

We will need user profiles that are structured in a way that makes it easy to find the information you are looking for, yet open enough that users can provide additional information and draw attention to themselves without cluttering the profile page.

We will need a user to user message system to help people communicate with one another while withholding contact information. Users should have the option to turn email forwarding of messages on or off. We may also need a way to notify users remotely (on a smart phone) when someone has requested to do a job, or requested them to preform a job.

We will also need a way to transfer payments on completion. This can be with paypal, check, credit card, or anything that will generate a paper trail. The employer will then need to submit proof of paying the employee so we can have a record if the need arises.

Design Space

SkillTrader is at its core a networking system catering to those who seek or offer skills. Considering this, we're building a website to be displayed on any device capable of text input - the sole requirement of interaction. Considering this, nearly all desktop and mobile computers are open and are our intended design space. While simply navigating the site can be done with any selecting tool like a mouse or tabbing, text input is required for the full experience: searching for offered skills. Considering our use of tagging for skills and badges for reputation, these will be clickable and allow a limited search function.

To ensure cross-platform and cross-device consistency, the search function can use javascript to generate results as the user types, which would decrease navigation actions and potential errors. To ensure the user receives adequate feedback here, the search results can automatically display a message when actual results are not available due to either error or no results. Javascript would support the searching task more easily, however we should provide a fallback option. Since some users may not enable javascript, or the load time for javascript is too heavy for users with a slow connection, we could possibly provide an entirely server-side system like PHP. This would require more full page refreshes which may cause errors not immediately recognizable to less computer-savvy users.

Aside from the content and functionality, the device used to access the site should also be considered when configuring the layout. For example, when accessed on a desktop with a strong connection, we can display all of the features without considerable fear of errors. However, mobile devices like phones and PDAs operating on a likely weak wifi connection would require the layout to change to the much smaller viewport afforded by the device and use fewer heavy loads such as images and unnecessary scripting. One unique capability of mobile devices to use with SkillTrader is geolocation. When a user who offers skills is available, they may either use a permanent location to meet up for trading skills or they may use their phone's location system to automatically update their current location. While this is also possible on more stationary devices like desktops and some laptops, phones often already have this functionality and is a potential feature to be explored.

Mobile only interface

SkillTrader is a system that connects users together. It can be assumed that most of the users will end up meeting together at some point in time to perform most jobs. We have a good reason to support mobile devices for our system since the users will be moving around to meet with one another. It would be useful to send and check messages while they are trying to meet up in case someone is running late/early or to ask the other person if they would mind meeting someplace else close by to the original meeting place. Our system has a fairly simple interface and considering people use Facebook on their phones fairly regularly, we can assume that our SkillTrader interface will be able to run well enough on smart phones. We don't necessarily need the larger website for the computer, so it would be feasible to leave it out all together and

to implement a mobile device only interface. Some users are likely to prefer having a larger screen with more information displayed so this may not be the best solution, but it is certainly an option.

Prompting for reviews

SkillTrader relies heavily on helping people find the best person they would want for doing a job. While that may be the best qualified, closest, most friendly, cheapest, best priced person or even some other quality we have not yet accounted for, the core of our system needs a way to rank the users. Prompts to write reviews can be annoying and not everyone wants to spend the time filling out a review for each job they have someone do, so we should implement a system that will be acceptably annoying and optional yet encouraged.

Weighted Review/Rating/Rep

Since SkillTrader does rely heavily on user ratings, we need to implement the rating system in a way that is most helpful to the users. People just starting to find jobs to do may find it difficult to be selected when they do not have a rating on their profile. Others that are just bad at doing jobs may end up with really low ratings. Both groups of people may try to boost their own rating artificially, which would be harmful to the SkillTrader website's purpose. The easiest way to boost your own rating would be to create fake accounts and give jobs and reviews to your self. It may be difficult to catch people doing this, so one solution would be to have a weighted rating system. This would make the reviews given by the fake accounts worth less than reviews given by actual accounts and would hopefully help to give a more honest rating. One possible drawback to the weighted review system is that it would encourage users to work with others who have a higher reputation which may make it even harder for new users to get their rating up.

Design One

Rationale

The major ideas here are functionality and simplicity. The user should have all the tools and instructions within a single viewport to explore the capabilities and results of SkillTrader. The layout and content is also fairly bare, with only links and a brief intro text to hint at what to do (or follow up on the about page) taking up the top 50% of the viewport and the search frame starting at the bottom 50% and extending as long as the results of the search provide up to a potentially user-defined limit. The three sections - navigation, intro, and search function - are in three distinguishably colored areas to reinforce the intended workflow.

Views

The screenshot shows the SkillTrader website interface. At the top is a dark navigation bar with the site name 'SkillTrader' and links for 'Home', 'About', and 'Contact'. A 'Login' button is located in the top right corner. Below the navigation bar is a light gray section with the heading 'Get help!'. The text explains that SkillTrader helps find people with skills and includes a 'Learn more' button. To the right of the text are two curved arrows: one pointing up and right with the text 'level up', and another pointing down and left with the text 'gain experience'. Below this is a search section titled 'Start typing to search' with a search input field. Below the search field is a table of search results.

Skill Match	User	Reputation	Available?
C++ · C#	C++Hack	★★★★☆	Yes
C++ · C	Codedog	★★★★☆	Yes
C	RandomBro	★★★★☆	No
Programming	yoruska	★★★★☆	Yes
Programming	shyGenius	★★★★☆	No

© SkillTrader 2012

Scenario

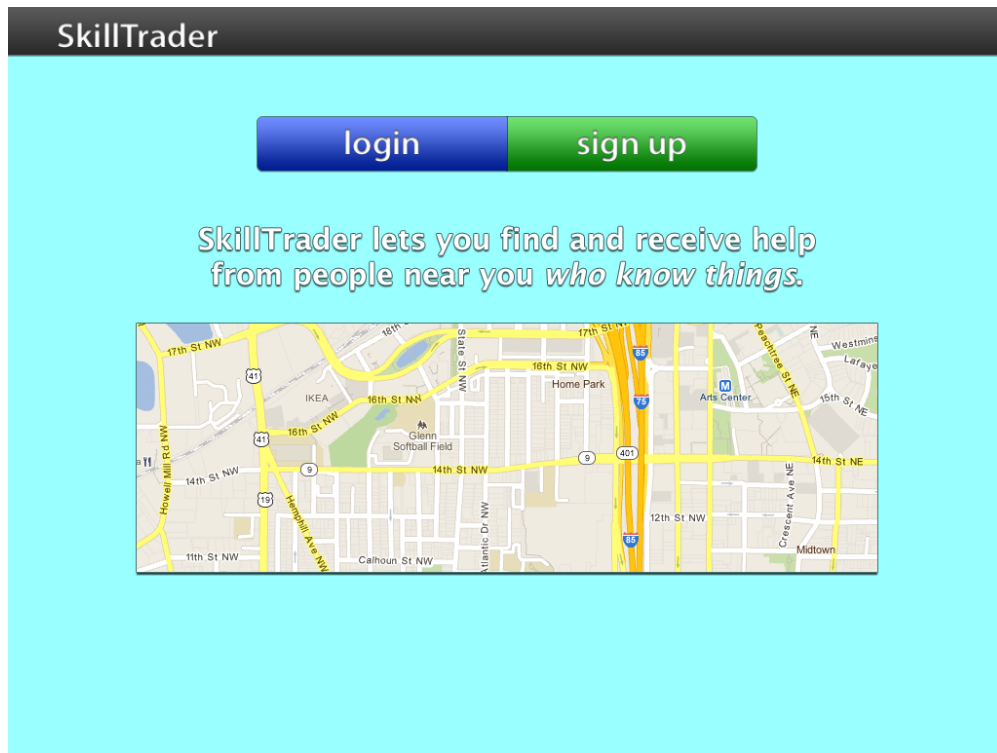
Uma is new to SkillTrader and was just linked to it from a friend, who found some help with cooking there. Naturally skeptical of another potential trading site like QuiBids, she's waiting for the inevitable catch - an esoteric currency, Nigerian princes, feature creep. Checking out the site from her laptop, she sees the interface is fairly bare, with only a few navigational buttons to the usual home, about, contact, and login pages, and the main focus of the page seems to be the search box visible under the intro banner. Seems innocent enough, but just in case she clicks the "learn more" link to the about page from the intro banner to completely understand what the site is all about. Now prepared for what to do, although it was probably easy enough to figure that out on her own, Uma returns to the main page and enters "C programming" to the search box. The fast search loads in the page as she types to prevent any unnecessary navigation, returning five users who offer skills related to her request. It seems to be sorted by relevance to her search, but they're all related enough, so she decides to sort by reputation, which the about page indicated was a weighted community rating system. Two users with four out of five stars stand above the rest, especially that guy with only one star and the vague skill offer of "Programming." Both the four star users are available according to their status, but it looks like user Codedog knows both C and C++. To find out more, Uma clicks the user's name and moves to his profile to have a chat and arrange a deal.

Design Two

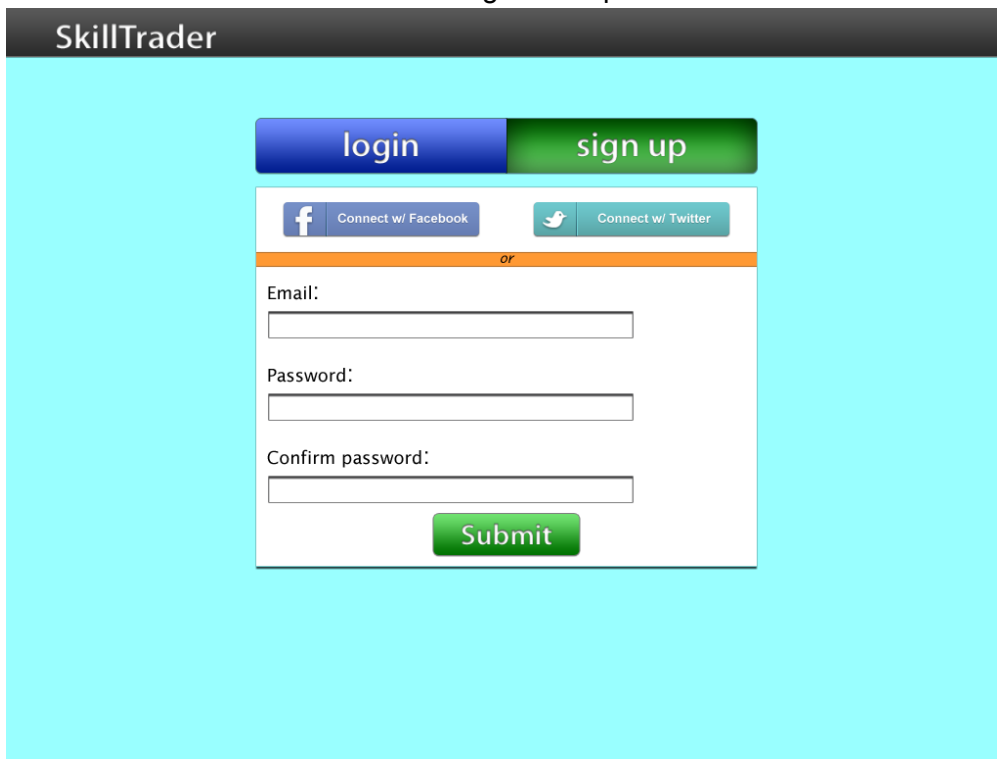
Rational

Considering the small number of truly independent actions that can be performed on SkillTrader, the design reflects this by omitting a standard navigation bar. The design is based around the user's home screen when logged into the system. The main actions, creating a new request from help and viewing other requests for help, can be accessed from this home screen, simplifying and condensing the user experience. The views for these actions can be thought of as being layered on top of the home screen; when the user has completed or would like to back out of the current action, he or she can return to the home screen. This design also incorporates a "badge" system that would allow users to display what skills they had or needed with a series of large, visual badges. Users can interact with these badges by dragging them into certain areas when creating new requests or filling their profile with skills. This physical interaction with the badges emphasizes to the user the importance of the skills to the system.

Views



Landing mock-up



Sign up mock-up



Jithu Maliakal



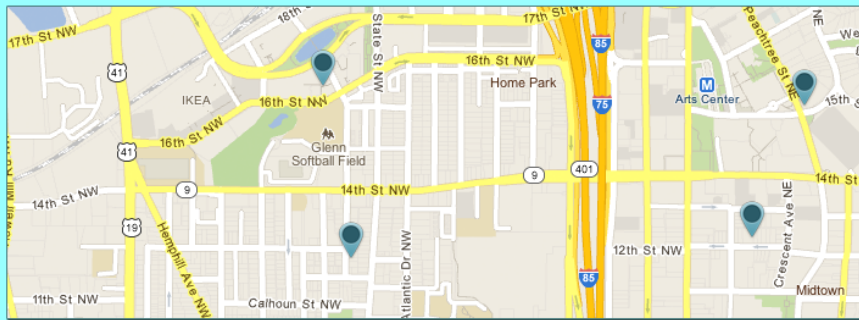
Atlantic Station, Atlanta, GA
Member for 2 months

My Badges:

Programming C++ Cooking Asian Fusion Repair Bicycle Programming MATLAB + Add Skill

Helpable Requests:

[View Requests](#)



My Requests:

[Add Request](#)

Need help with an SQL database	2/10/12
Want to change the oil in my car	2/20/12
Dishwasher mounting repair	2/23/12

Profile/home screen mock-up

SkillTrader

login	new user
-------	----------

SkillTrader information, statistics
e.g, we've solved 'x' # of problems

about / contact / privacy / etc...

landing screen

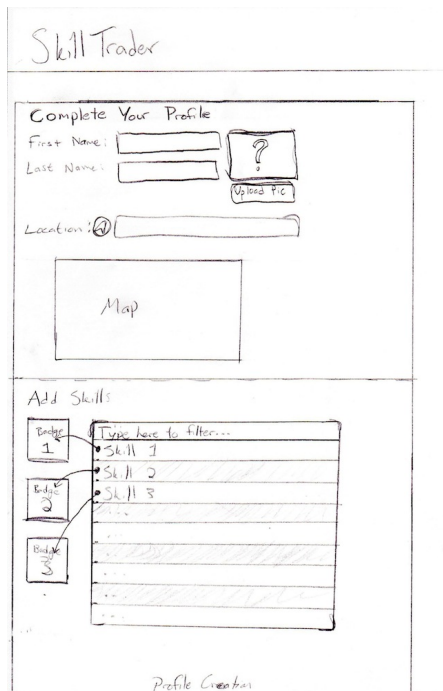
Landing wireframe

SkillTrader

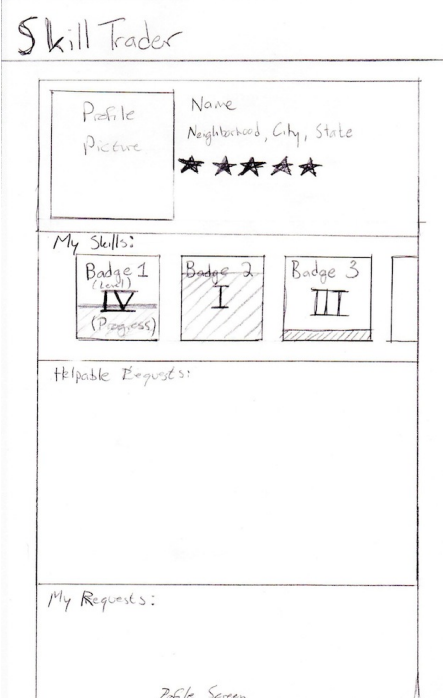
login	new user
<input type="checkbox"/> Connect to facebook	
<input type="checkbox"/> Connect to twitter	
Enter E-mail:	
<input type="text"/>	
Enter password:	
<input type="text"/>	
Confirm password	
<input type="text"/>	

new user creation

Sign up wireframe



Profile filling wireframe



Profile home wireframe

Skill Trader

Skill Request

Request title:

Description:

Skill(s) required:

Enter skills here..	
<input type="checkbox"/>	Skill 1
<input type="checkbox"/>	Skill 2
<input type="checkbox"/>	Skill 3
<input type="checkbox"/>	Skill 4

Location:

Come to me (results in Address bar)

I'll come to you (results in "w/in X mi")

New Request Screen

New request wireframe

Scenario

Fresh out of college, Kevin has recently moved into an apartment. While he loves the convenience of take out food, it's hurting both his physique and his wallet. He wants to learn how to cook, but Epicurious recipes and instruction videos on Youtube just aren't cutting it. His friend told him about SkillTrader, a service that connects skilled people with each other. Interested, he decides to check out the site.

Once the site loads, Kevin is greeted by the introductory sentence on the landing page: "SkillTrader lets you find and receive help from people near you *who know things*." Reading this, Kevin has a basic idea of what SkillTrader can do. Deciding he wants to try it out, he is able to quickly figure out what to do next, and he clicks the large, green sign up button. Kevin is sent to a sign up sheet that only asks for his email address and password, allowing him to quickly move on to creating his profile. This screen asks for some more information, including his name, a photo, his location, and what skills he has. He fills out his name and clicks the compass icon next to the location bar, allowing SkillTrader to detect his location through his browser. Kevin types in "programming" to add his first skill. The search bar also returns C, C++, and Java, among others. He clicks on the C++ line and notices that the empty boxes on the left side start glowing, prompting him to drag the line into the box; this creates Kevin's first badge. Kevin then repeats this with a few other skills. Finally, Kevin submits the form and is brought to his profile page where he can see the status of his badges, people in his area who he can help, and an empty list of his requests for help.

As he originally came to the site to find help with cooking, he clicks on the green Add Request button. The resulting view prompts him for a title for the request, a description of the request, the skills that may be required, and the location of the problem. The small amount of information required by the system allows Kevin to quickly create his new request.

Design Three

Rationale

This design used a fairly standard website layout with navigation links across the top of the page and information organized in boxes. This allows users to quickly be familiar with the basic navigation of the UI. It also has skills organized into collapsible boxes of similar skills. This allows users to more quickly find skills on the profile page, and reduces the amount of clutter for the profile. The layout has a 5 star rating system for both skill and for rating. The skill rating is how qualified the user believes himself to be, or how qualified the user is determined to be based on degrees or certificates and tests. The rating rating is given by how well the person has been rated by people who have hired him/her to complete a job.

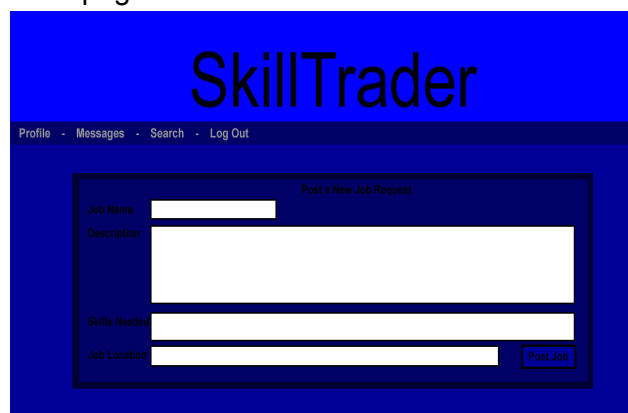
Views

Log in page



The screenshot shows the login page of the SkillTrader website. The page has a dark blue background. At the top, the word "SkillTrader" is written in a large, white, sans-serif font. Below the logo, there is a navigation bar with the following links: "Profile", "Messages", "Search", and "Log Out". In the center of the page, there is a white box containing a login form. The form has the following fields: "LOG IN" (title), "User Name" (input field), "Password" (input field), and a "Log In" button.

Job creation page



The screenshot shows the job creation page of the SkillTrader website. The page has a dark blue background. At the top, the word "SkillTrader" is written in a large, white, sans-serif font. Below the logo, there is a navigation bar with the following links: "Profile", "Messages", "Search", and "Log Out". In the center of the page, there is a white box containing a job creation form. The form has the following fields: "Job Name" (input field), "Description" (text area), "Skills Needed" (input field), "Job Location" (input field), and a "Post Job" button.

Search page

SkillTrader

Profile - Messages - Search - Log Out

SEARCH

Key Word:

Distance: Only Show Jobs I Qualify For

Programming a Java Applet

Skills Needed: Java, Object Oriented Design, User Interface Design

Description: Design a small application for recording prices of resources and predicting the market trends. It should be able to sort data on location, resource, retail company, and shipping company.

[\[More Information\]](#)

Profile page

SkillTrader

Profile - Messages - Search - Log Out

John Johnson

[Message Me](#)



Programming	Skill	Rating	Reviews	
Java	★★★★★	★★★★★	10 available	I have been programming in Java for 5 years
C++	★★★★★	★★★★★	4 available	I have learned C++ at school at GA Tech
HTML	★★★★★	★★★★★	2 available	I have messed around enough to do some basic layouts
PHP	★★★★★	★★★★★	10 available	I have messed around enough to work with PHP

Cooking	Skill	Rating	Reviews	
Baking	★★★★★	★★★★★	8 available	I have been cooking for about 2 years on and off
Grilling	★★★★★	★★★★★	2 available	I have been cooking for about 2 years on and off
Deserts	★★★★★	★★★★★	1 available	I have been cooking for about 2 years on and off

Mechanic	★★★★★	★★★★★	0 available	I have worked with random stuff... yea
Balloon Shaper	★★★★★	★★★★★	2 available	I can pop stuff with pins
Hunter	★★★★★	★★★★★	5 available	I can shoot guns at stuff for you
Lego Builder	★★★★★	★★★★★	0 available	I have played with legos for like FOREVER

Tutoring	Skill	Rating	Reviews	
CS @ GT	★★★★★	★★★★★	2 available	I am a 4th year computer science major at GT
Java @ GT	★★★★★	★★★★★	3 available	I have been programming in Java for 5 years
Math @ GT	★★★★★	★★★★★	1 available	I know that 1 + 1 = tree

Edit profile page

SkillTrader

Profile - Messages - Search - Log Out

John Johnson

[Add New Skill]

Prog	Skill Name		
Java	Proficiency	2	<input type="button" value="U"/>
C++			
HTA	Description		
PHP			<input type="button" value="Add Skill"/>



I have been programming in Java for 5 years	edit
I have learned C++ at school at GA Tech	edit
I have messed around enough to do basic layouts	edit
I have messed around enough to work with PHP	edit

Cooking	Skill	Rating	Reviews	
Baking	★★★★★	★★★★★	8 available	I have been cooking for about 2 years on and off edit
Grilling	★★★★★	★★★★★	2 available	I have been cooking for about 2 years on and off edit
Deserts	★★★★★	★★★★★	1 available	I have been cooking for about 2 years on and off edit

Mechanic	★★★★★	★★★★★	0 available	I have worked with random stuff... yea edit
----------	-------	-------	-------------	---

Balloon Shaper	delete 2	★★★★★	2 available	I can pop stuff with pins save
----------------	---------------------	-------	-------------	--

Hunter	★★★★★	★★★★★	5 available	I can shoot guns at stuff for you edit
--------	-------	-------	-------------	--

Lego Builder	★★★★★	★★★★★	0 available	I have played with legos for like FOREVER edit
--------------	-------	-------	-------------	--

Tutoring	Skill	Rating	Reviews	
CS @ GT	★★★★★	★★★★★	2 available	I am a 4th year computer science major at GT edit
Java @ GT	★★★★★	★★★★★	3 available	I have been programming in Java for 5 years edit
Math @ GT	★★★★★	★★★★★	1 available	I know that 1 + 1 = tree edit

Send message page

SkillTrader

Profile - Messages - Search - Log Out

Title Recipient

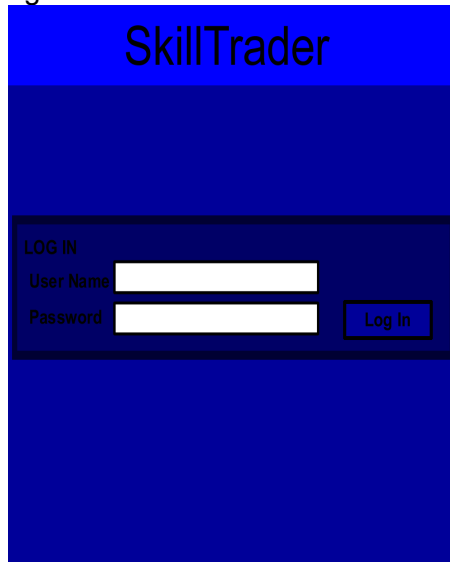
Message

Receive message page



Phone Layout

Log in page



Profile page

SkillTrader

Profile - Messages - Search - Log Out

John Johnson
[Message Me](#)

Programming	Skill	Rating	Reviews
Java	★★★★★	★★★★★	10 available I have been programming in Java for 5 years
C++	★★★★★	★★★★★	4 available I have learned C++ at school at GA Tech
HTML	★★★★★	★★★★★	2 available I have messed around enough to do some basic layouts
PHP	★★★★★	★★★★★	10 available I have messed around enough to work with PHP

Cooking	Skill	Rating	Reviews
Baking	★★★★★	★★★★★	8 available I have been cooking for about 2 years on and off
Grilling	★★★★★	★★★★★	2 available I have been cooking for about 2 years on and off
Deserts	★★★★★	★★★★★	1 available I have been cooking for about 2 years on and off

Mechanic	★★★★★	★★★★★	0 available I have worked with random stuff... yea
Balloon Shaper	★★★★★	★★★★★	2 available I can pop stuff with pins
Hunter	★★★★★	★★★★★	5 available I can shoot guns at stuff for you
Lego Builder	★★★★★	★★★★★	0 available I have played with legos for like FOREVER

Tutoring	Skill	Rating	Reviews
CS @ GT	★★★★★	★★★★★	2 available I am a 4th year computer science major at GT
Java @ GT	★★★★★	★★★★★	3 available I have been programming in Java for 5 years
Math @ GT	★★★★★	★★★★★	1 available I know that 1 + 1 = 2

Search page

SkillTrader

Key Words

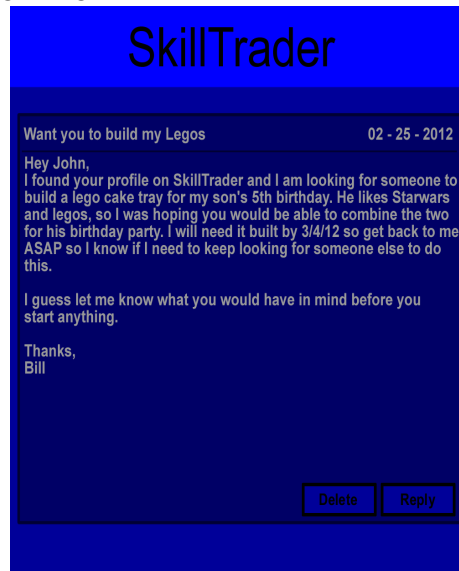
Distance Only Show Jobs I Qualify For

Programming a Java Applet
 Skills Needed: Java, Object Oriented Design, User Interface Design
[\[More Information\]](#)

Programming in Java
 Skills Needed: Java, User Interface Design
[\[More Information\]](#)

Developing a PHP Website
 Skills Needed: HTML, PHP, User Interface Design, Graphic Design
 CSS, MySQL
[\[More Information\]](#)

Message page



Scenario

John is a computer science student still at college, and has been using the SkillTrader website for about a month. He is familiar with the interface, has filled out his profile and has even done a couple jobs already in the past. He wouldn't consider himself a long time user, but he is experienced with the system and usually logs in a couple times a week to check on things.

John is sitting in class with a teacher he can hardly understand and is having difficulty following the lecture, so he decides to check on his SkillTrader account. He clicks his bookmark and heads to the log in screen. John types his user info, and logs into the site. He then is directed to his profile, and so he decides to double check he has it filled out the way he wants. He notices he has a new message, so he clicks to open it and reads an offer to build some Legos for a little kid's birthday party. John has a lot he should be doing this week and would prefer to do something more related to computers to get more experience, so he doesn't really want to the Legos. He quickly hits the reply button and types out a message explaining he is really to busy at the moment and can't do the job. John then clicks the link to go to the search page, and does a quick search for computer related jobs near by. He finds a job offer for programming a Java application that will do a little data analysis for finding market trends to help a smaller company buy and ship resources. He clicks for more information and decides to send a message saying that he is interested in the job. After classes are over, John takes the bus back too his dorm and decides to check his SkillTrader account on his smart phone. He logs in and notices he has a new message. He opens it and sees the business is interested in having him make the Java application, so John shoots back another message asking when would be a good time to meet up and figure out the details.

Design Assessment

Criteria	Design 1	Design 2	Design 3
Supported platform	Desktop	Desktop/Mobile	Desktop/Mobile
Navigation	Navigation bar, search	Modal navigation	Links are displayed at the top of the page
Accessibility	Uses few, contrasting colors to help the user see what is important.	Large badges allow users to easily see skills; however, too many colors may be distracting.	Low contrast and dark colors may make site usage difficult.
Load time	May take longer to load search functionality	May take longer to load badge images and interface	Very lightweight, should load the fastest
Ease of signing up	Can be completed via login button and verified via email. Further account requirements are done via user page	Allows connection to other social services.	Same as any other website. Click new user and fill in the blanks.
Ease of searching	Search is a primary feature, auto-complete	Can find requests with the map, full search is linked	Link to the search, out of plain view
Affordances	Sections are clearly marked between navbar, info area, and search frame	Location features affords usage of the system from many locations.	grouped skills based on categories
Precision	Large, colored buttons and search bar	Buttons are large allowing the user to easily perform actions	Things are a little compressed in the edit profile view, which may make it a little hard on the user
Assumed User Knowledge	General browsing, such as how to recognize links	Does not require the user to be familiar with standard website navigation (no navigation bar).	Assume user is familiar with website navigation
Constraints	Requires a broadband Internet connection for the auto-complete search interface.	Dragging would require a desktop browser or a native mobile application (i.e., not supported on mobile web); location would work better on mobile.	Low contrast requires the user to have good vision

Feedback	Auto-complete search	Draggable badges	
----------	----------------------	------------------	--

Reflection

During this phase of the project, we had a few difficulties when making our initial designs and our poster. The first was during the creation of wireframes and mock-ups. We realized that it is very difficult to come up with significantly different designs for the same functionality and feature set, especially after already deciding to build a website interface. This was made even more difficult after we met for the first time during this phase and discussed our initial ideas. This caused certain design elements to be transferred among what were supposed to be three entirely different designs. After finishing this step, we realized that it may have been better to create each design in its entirety in a vacuum (i.e., without contacting each other and discussing the designs). During the assessment of the designs, we also found that it was difficult to separate discussion of the user interface design itself from discussion of the features of the system. Finally, after we created our designs and explained our rationales for these designs, we created the poster. Unfortunately, the plotters in the Library and school of Biomedical Engineering were unavailable by the time the poster had been finalized.

SkillTrader - Part Three

Andrew Nelson, Spencer Hamilton, Jithu Maliakal, Michael Schueth

Introduction

People tend to have problems, and usually they're easily fixable. For these problems, there often exist other people who already have some skills in solving these problems; they may even have problems of their own. Unfortunately, the availability of these people who solve problems are obfuscated by a lack of networking, or these people are professionals who tend to take on bigger problems than ones many people have such as fixing a car, learning to ski, or getting some help with jQuery.

SkillTrader aims to connect these people, who may either want to share their skills or seek help from others. This could be accomplished by trading skill for skill, skill for money, or just being someone pursuing a hobby who expects nothing in return. An important element in SkillTrader is that the users are facilitated rather than guided by our service.

For example, a professional painter may be having car trouble - nothing major, but the painter is definitely not experienced with what goes on under the hood and can't afford a \$100 estimate by a mechanic. Meanwhile, a car buff who has lovingly tended to his '95 Impala for years is looking for a creative field to break into. These two strangers have a goal but neither goal is quite pressing enough to consider spending a great deal of money anytime soon. Instead, they could join SkillTrader to network with others with skills such as these and find their answer.

We are taking on a market populated by people who have problems and seek solutions without worrying about dollar cost. Allowing the users to take the major role in the service as both the users and providers of the skill commodity is placing a great deal of trust on them; however, services like Craigslist have proven it sustainable and fairly reliable. The people we help are seeking skills that other people have already developed, allowing them to learn something new, disseminate or sell their knowledge, or just look for some personal connection over an interesting activity. It's a fairly human thing to do, and we're making it easier.

Requirements Summary

After gathering some of the system requirements, it has become apparent that SkillTrader is going to require an efficient and dynamic data structure to help drive the interface.

SkillTrader will need ways to sort and group skills, a way to search based on multiple constraints with user-entered data, a way to determine the distance someone must travel to get to where the job's location is, a way to filter for jobs that someone would be interested in, and a way to filter for people that would be able to accomplish jobs.

We will need to provide users with a way to advertise themselves and/or the jobs they want completed. We will also need to make sure people can trust each other since a job is likely to be the first time users will be meeting each other. We will need to find a balance that users find acceptable between protecting people's personal information and connecting people for jobs as well as providing enough information for users to trust each other. We will need a way for users to include any qualifications they may have for a particular skill, and a way for the users to rate each other on how skilled they are as well as how well the experience as a whole went. We will also need a way to flag users for misusing the SkillTrader system, and a way to warn other users what they did wrong.

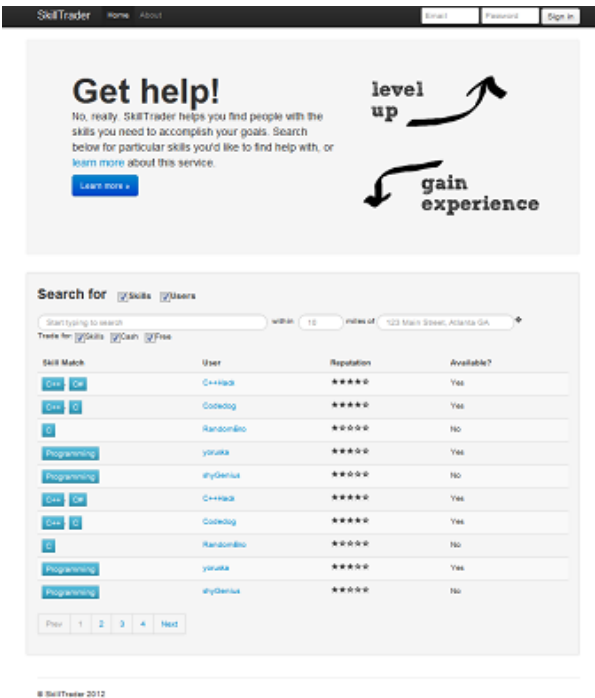
We will need user profiles that are structured in a way that makes it easy to find the information you are looking for, yet open enough that users can provide additional information and draw attention to themselves without cluttering the profile page.

We will need a user to user message system to help people communicate with one another while withholding contact information. Users should have the option to turn email forwarding of messages on or off. We may also need a way to notify users remotely (on a smartphone) when someone has requested to do a job, or requested them to perform a job.

We will also need a way to transfer payments on completion. This can be with paypal, check, credit card, or anything that will generate a paper trail. The employer will then need to submit proof of paying the employee so we can have a record if the need arises.

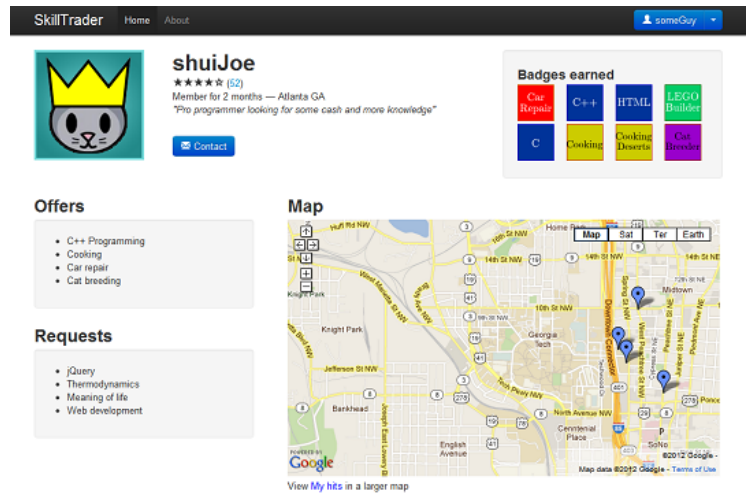
Final Design Summary

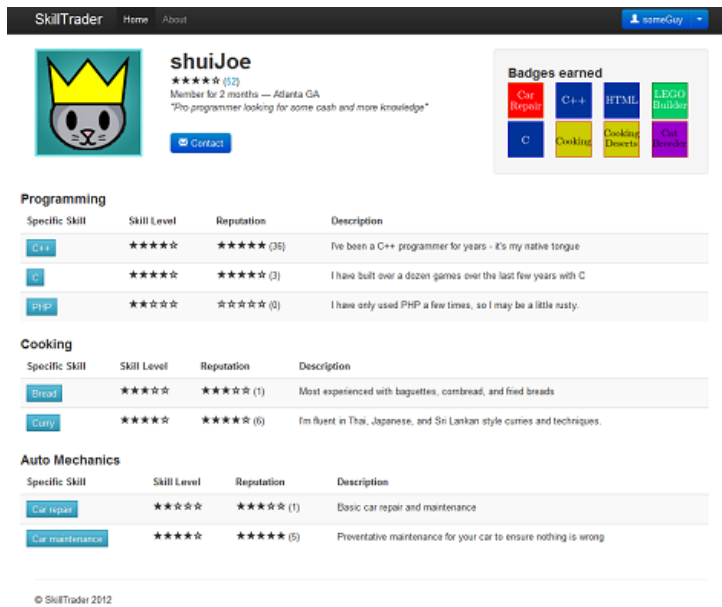
For our final design, we chose to merge two designs (the first and third) from our last milestone and integrate more detailed elements from the second design.



The front page should be able to convey all necessary information about the service and provide an easy point of entry for actually using the service. Since the main feature is in fact this point of entry, the majority of the service's functionality is contained in this main page.

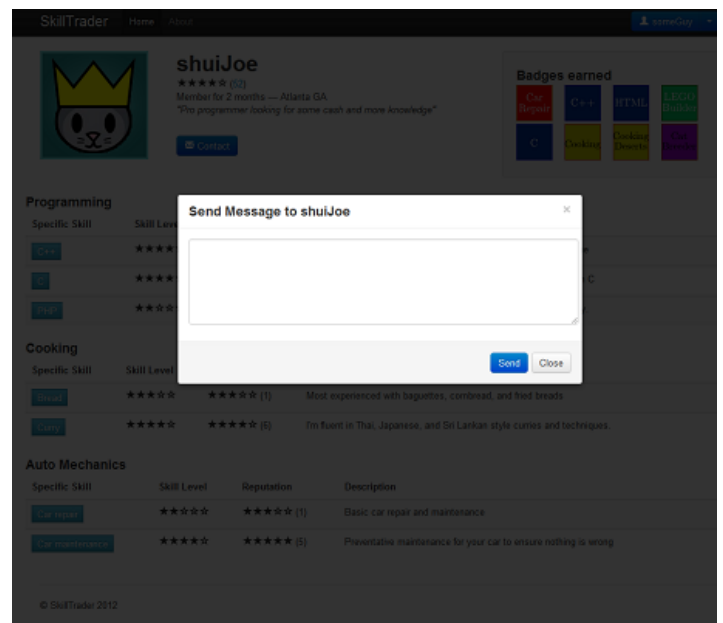
Next we incorporated elements of all three designs in the user page, which can double as both a personal and public profile page. Mechanics of the inter-user interactions were fleshed out starting here, as this is the second step, following finding a result on the main page, in most of our scenarios and is the launching point for other operations like checking a user's reviews, finding specific skills, and communicating with the user.





Following finding the user, our scenarios described determining what the user offers in particular. Such as a backyard auto mechanic offering some preventative care tips. On this skill-view page, other users may more accurately understand a user's skills and how other users have rated them.

Finally, users may send a message to other users to request, offer, or trade their skills through an internal messaging system to ensure privacy. This as well as some other single-function actions like updating account information use modals like this.



Evaluation Plan

The first few areas of evaluation are

1. What do users want to do with their skills?
2. What's the easiest way to find others' skills on our site?
3. How comfortable would they be sharing their skills?

For these questions, a combination of interviewing and thinking out loud as users navigate similar websites (Craigslist, Fiverr) would be helpful in gathering the basics of what users are seeking. Once our findings are applied to SkillTrader, we can repeat the process with users navigating SkillTrader. First we may familiarize the test-users with the website, largely described on its About page, and then let them navigate the site and complete a questionnaire (see below: "Tasks for the users"). Following this, we may end with an unstructured interview to address their most pressing concerns and anything they were confused about. The feedback from the questionnaire and interview should be the foundation of testing our user interface, with successive rounds adding or changing the questionnaire.

Tasks for the users:

Please perform the following tasks to help us evaluate our design. We ask that you think out loud while you do the tasks.

1. Please login to the service.
Completion time goal: 10 seconds

Observations:

How easy was it to carry out this task? (using a 10-point scale, 10= very easy [could do this in your sleep], 1= very difficult [would need hours of training])

Very Difficult

1

2

3

4

5

6

7

8

Very Easy

9

10

Why did you give that rating (esp. if low score)?

How would you change the process to do this task?

2. Please update your profile.
Completion time goal: 30 seconds

Observations:

How easy was it to carry out this task? (using a 10-point scale, 10= very easy [could do this in your sleep], 1= very difficult [would need hours of training])

Very Difficult										Very Easy
	1	2	3	4	5	6	7	8	9	10

Why did you give that rating (esp. if low score)?

How would you change the process to do this task?

3. Please search for a job you need completed.
Completion time goal: 1 minute

Observations:

How easy was it to carry out this task? (using a 10-point scale, 10= very easy [could do this in your sleep], 1= very difficult [would need hours of training])

Very Difficult
1 2 3 4 5 6 7 8 9 10
Very Easy

Why did you give that rating (esp. if low score)?

How would you change the process to do this task?

5. Please send a message to another user.
Completion time goal: 2 minutes

Observations:

How easy was it to carry out this task? (using a 10-point scale, 10= very easy [could do this in your sleep], 1= very difficult [would need hours of training])

Very Difficult											Very Easy
1	2	3	4	5	6	7	8	9	10		

Why did you give that rating (esp. if low score)?

How would you change the process to do this task?

Interview Questions

- 1) What parts of the design did you like?
- 2) What parts of the design did you not like?
- 3) What functionality would you want to see on such a website?
- 4) Are there any parts to the site that you feel are not needed?
- 5) How much detail should we put in the badges? Small icon? Large icon? Text? images?

6) **How easy was it to use this website overall?**

(1= very difficult [would need hours of training], 10= very easy [could do this in your sleep])

Very Difficult											Very Easy
1	2	3	4	5	6	7	8	9	10		

Why did you rate it as such?

- 7) Would you actually use a website like this? Why or why not?

Reflection

As we built our prototype during the course of this project phase, we found ourselves thinking about some of the core concepts of our project. One such example was whether the design should emphasize users offering their own skills or seeking the skills of others. After discussing this, we determined that offering skills should be emphasized as it would help the service grow by offering more chances to learn or earn something and promote a meta goal of fostering a knowledge ecosystem.

We really have not hashed out how much detail we will need for the badges or really how the badges will actually work. Originally, badges were to be a publicly visible mark on a user's profile that they are a sort of guru or at least has attained some higher level of competence in a particular skill. We still need to figure out how a user will earn a badge, though. Do they just add the skill/badge to the profile? Do other users rate them and at a certain point they qualify for a badge? Do they have to pass some sort of test to prove they are worthy of the badge? We also need to figure out the best way to display the badges. Do we want small icons that are easily displayed or do we want more detailed larger badges that give more information to the users? We could also conduct more research into how a badge system has succeeded in this context compared to instances of badge systems in other popular websites.

Thanks to Twitter Bootstrap, we're able to prototype easily with a much higher degree of function and design than other tools could afford us. Considering only one or two of us were particularly confident in using Bootstrap (or HTML/CSS in general), the workload was a little off-balance as we jumped between generating designs and implementing them. However, now that we have enough material live and mostly functional, we can split our tasks by design, implementation, and feedback. Since we all have some experience in each, we can better distribute work and get results faster to allow our iterative process to actually have more than one or two iterations.

SkillTrader - Final Project

Andrew Nelson, Michael Schueth, Jithu Maliakal, Spencer Hamilton

Evaluation Plan:

The first few areas of evaluation are

1. What do users want to do with their skills?
2. What's the easiest way to find others' skills on our site?
3. How comfortable would they be sharing their skills?

For these questions, a combination of interviewing and thinking out loud as users navigate similar websites (Craigslist, Fiverr) would be helpful in gathering the basics of what users are seeking. Once our findings are applied to SkillTrader, we can repeat the process with users navigating SkillTrader. First we may familiarize the test-users with the website, largely described on its About page, and then let them navigate the site and complete a questionnaire (see below: "Tasks for the users"). Following this, we may end with an unstructured interview to address their most pressing concerns and anything they were confused about. The feedback from the questionnaire and interview should be the foundation of testing our user interface, with successive rounds adding or changing the questionnaire.

Tasks:

1. Login to the service.
2. Update your profile.
3. Search for a job you need completed.
4. Assess a user to complete the job from (3) above.
5. Send a message to another user.

Questionnaire:

1. What parts of the design did you like?
2. What parts of the design did you not like?
3. What functionality would you want to see on such a website?
4. Are there any parts to the site that you feel are not needed?
5. How much detail should we put in the badges? Small icon? Large icon? Text? images?
6. How easy was it to use this website overall?
7. Would you actually use a website like this? Why or why not?

Users/Testers:

1. Sho Kitamura
2. Blake Palacio
3. Craig Burgess
4. Luis Simauchi
5. Laurissa Rybacki

Evaluation Plan Rational:

We chose our 5 tasks because they are fundamental to using the system. It is impossible to use the system without logging in, so we put that one first. Hiring someone over the internet can make a user somewhat skeptical, so we encourage users to keep profiles up to date and stress the importance of completely filling them out. So, we decided that updating a profile is an essential task. We believe that users should be searching for jobs rather than jobs searching for people, so our next task was to have the user try to find a job to complete.

It is unclear whether the system should put stress on finding jobs or finding people to do the jobs. Another essential task to do is to choose a user that you would want to complete a job for you, so we added that to the list of tasks to have users test. Communication is essential for working out the details between the worker and the employee, but people said protecting privacy was also important... so we added an internal messaging system to the system. Since it is an essential function, we also had users test it out.

Just having a user complete a couple tasks and recording how well they do does not get their opinion of the overall system, so we also included the questionnaire to ask the testers after they have completed the test tasks.

Results of the Study:

User: Sho Kitamura

Initial reaction:

The first thing he did was read the intro text on the front page, which he remarked was sort of obnoxious looking - possibly the size or font choice. After that, he immediately wanted to use the search box.

Tasks:

1. **Logging in / registering:** Easily found the actionable items and knew what to do
2. **Updating profile:** First, he clicked his profile button in the top right corner, but he was confused when it took him to his user page instead of a preference page. He looked around the profile page body for clues, but it required my hint to realize there was an arrow by his name in the profile button. He realized this and then understood it to mean there were further options if the arrow were clicked. After clicking, he found the preferences button and continued with ease and a degree of familiarity. He said he would prefer a button on his profile page body to do this.
3. **Searching for a job he needs someone to complete:** He clicked the SkillTrader button in the top left corner to return to the main page and then clicked in the search box. Before typing, he noticed the skill/user check buttons and unchecked users to only search for skills, and then typed in the skill he was seeking. At this point I told him to pretend results were given in the list below, which he expected. I asked why he didn't enter location and that his search would turn up results from all SkillTrader users, even if they were from another country. He responded that he expected the search to turn up only local results, and didn't bother entering location details despite acknowledging the field while he was searching. He suggested using Craigslist-style sub-domains for cities, (e.g. atlanta.skilltrader.com), as he has used Craigslist. Moving on, looking through the results, he first identified which users had the highest reputation and then checked whether they were available, narrowing his choice to three. He assumed that the users were ranked by geographic proximity and thus chose the user at the top of the list, thinking that user was the closest. I explained that they were actually listed by relevance to the search term and I had to explain the skill hierarchy. He clicked the user's name to continue.
4. **Assess the chosen user:** After a glance over the user's profile page, he noticed the number beside his overall rating and understood it to mean the number of reviews. He did not, however, realize it was a link until I told him. After clicking it through to the skill breakdown page, he looked for the skill he was looking for. He understood both the review and confidence ratings but did not find the badges useful (he assumed they were simply to imply the trustworthiness, but he did not put much value to them). He also wanted to know what the user wanted in exchange for his skill, which he remembered from the main search page but did not see here.
5. **Send the user a message:** From the skills breakdown page, he clicked the contact button and sent the user a message. He wanted to wait on the site for a few minutes to see if the person would immediately respond, so he restarted the process and further investigating the site. He appreciated the in-mail system for not giving out his actual email but wished that the site emailed him when the user replied.

General questions

1. **What parts of the interface did you like?** He appreciates the "Trade for"

options when searching.

2. **What parts of the interface did you dislike?** He did not notice the about page or “Learn more” button on the main page until after sending the message, although he understood enough from his internet experience to function as expected anyway. The image on the front page was unhelpful and should be easy to understand; he would not want an imageless front page, though. Also, the downward arrow was inferred as a thumbs-down or negative idea instead of the intended direction idea.
3. **What functionality would you want to see on such a web service?** He would personally use it mostly for school help and thought it would be cool if organizations could post jobs that required a particular skill (such as a one-time gig for photography, which Sho is skilled in). Text with feedback on the skill breakdown page would have been more helpful than just stars and number of reviews; he also would like to see a thread style to the feedback (like eBay and Newegg reviews) so that the reviewed user can respond to negative feedback he considered incorrect or misleading.
4. **Are there any parts to the site that you feel are not needed?** He did not find badges necessary (he assumed they were simply to imply the trustworthiness, but he did not put much value to them).
5. **How easy was it to use this website overall?** Very easy
6. **Would you actually use a website like this? In what context?** Initially, he would use it more to get help until he felt comfortable with system before offering help. He explained that he’d feel worse about being bad at offering a skill to teach than getting bad help with a skill from someone else. Although, if he were burned by receiving bad help once, he would not be turned off by the service and would give it more than one chance.

User: Blake Palacio

Tasks:

1. **Logging in / registering:** It was easy; I would make enter login
2. **Updating profile:** Was called Preferences, a little confusing; took a while to find button; rename the button
3. **Searching for a job he needs someone to complete:** The user could not find the functionality, and based on the wording of the task he did not believe that he completed the task after the search bar was pointed out to him
4. **Assess the chosen user:** Same as 3... sigh...
5. **Send the user a message:** Very easy

General questions

1. **What parts of the interface did you like?** Like the top bar and the layout of sidebar
2. **What parts of the interface did you dislike?** The labeling of buttons in drop downs. Also the search was hard to see at first on a smaller screen
3. **What functionality would you want to see on such a web service?** Ability to contact individuals through means other than the website (phone number, email, etc.).
4. **Are there any parts to the site that you feel are not needed?** No, everything that's there is functional and relevant.
5. **How easy was it to use this website overall?** Was not drawn to some important functionality at first (search bar at bottom of site).
6. **Would you actually use a website like this? In what context?** Yes, could use some help on things

User: Luis Simauchi

Tasks:

1. **Logging in / registering:** He found the task easy and straightforward
2. **Updating profile:** He found the task was a bit difficult and indicated that there was no clear path to edit the information. He says the link should be more clearly labeled.
3. **Searching for a job he needs someone to complete:** It was simple to find and to use the search bar
4. **Assess the chosen user:** He spent too much time trying to find the link and believes that it is too small.
5. **Send the user a message:** it was easy to do and worked well.

General questions

1. **What parts of the interface did you like?** Search bar, recommended searches, Messaging system, rating system
2. **What parts of the interface did you dislike?** Would like an auto complete search, editing the profile is difficult to figure out, lots of white space, no clear way to rate other users.
3. **What functionality would you want to see on such a web service?** He liked the map functionality.
4. **Are there any parts to the site that you feel are not needed?** No
5. **How much detail should we put in the badges? Small icon? Large icon? Text? images?** Little stale, round them, make them icons
6. **How easy was it to use this website overall?** It was moderately difficult to use the site. Not entirely intuitive, make all tasks more apparent, give more buttons
7. **Would you actually use a website like this? In what context?** Maybe

User: Craig Burgess

Tasks:

1. **Logging in / registering:** Registration was unnecessary.
2. **Updating profile:** Changing the profile should be a button on the profile page, not in preferences.
3. **Searching for a job he needs someone to complete:** Does not appear to make sense, what job am I searching for? I can search for skills only.
4. **Assess the chosen user:** Only shuiJoe seems to work. The star layout for skill level and reputation is intuitive.
5. **Send the user a message:** Contact button is clearly visible, easy to find

General questions

1. **What parts of the interface did you like?** The star layout for rating and skill level, Badges good, google map
2. **What parts of the interface did you dislike?** How to search for a job is confusing
3. **What functionality would you want to see on such a web service?** Compare users side by side, Search by skill level, more search options in general
4. **Are there any parts to the site that you feel are not needed?** No
5. **Badge changes?** Image, with hover over explanation, Click them, link to a list of users with that badge
6. **How easy was it to use this website overall?** Make the steps searching for a job more obvious
7. **Would you actually use a website like this? In what context?** Yes, if I needed a job done and it was easy to use

User: Laurissa Rybacki

Tasks:

1. **Logging in / registering:** It was very easy
2. **Updating profile:** It took a little while, but the user was able to figure it out. The user recommends adding an edit button on the profile page
3. **Searching for a job he needs someone to complete:** The user was a bit confused by this task,
4. **Assess the chosen user:** N/A
5. **Send the user a message:** The user easily figured this out and was able to send off a message.

General questions

1. **What parts of the interface did you like?** The user said that it is easy to register, log in, send a message, and edit a profile. The map of the skills was a nice touch, and they liked the badges. She thought the overall site was clean and pleasing to use.
2. **What parts of the interface did you dislike?** She hates the search bar. skills are listed twice, and the search is unclear.
3. **What functionality would you want to see on such a web service?** Vouching system where highly rated users could vouch for new users or friends.
4. **Are there any parts to the site that you feel are not needed?** no
5. **Badge changes?** Large icon with images on the profile
6. **How easy was it to use this website overall?** Everything was easy except for searching.
7. **Would you actually use a website like this? In what context?** No, the user believes she has no skills.

Analysis of the Results:

After doing some user testing, we came to the realization that the site was not quite as user friendly as we had hoped for. Some of the wording could have been better, and some of the functionality could be moved around. We were told that important functions of the website should be near the top of the page and need to be intuitive and easy to use. It wasn't really anything we didn't already know, but we were surprised that some users did not think our design was entirely intuitive or easy to use. We learned that the method for editing the user profile could be done better, and that some users didn't quite like how the search bar was placed.

We made a couple changes:

- Added extra edit buttons to profile page.
- Viewing the skill breakdown page requires finding a single small button, so we enlarged button and provided a text description.
- The user assumed search results were listed by proximity, so we changed search listing from search term relevance to proximity to entered location.
- Changed search action to onkeyup to create an automatic Google Instant-like searching experience.

Possible changes if we had more time:

Problem: Users did not see a way to provide feedback for a user

Lesson: Provide more functionality for the user to experiment with. If we had more dedicated time, we could have added more functionality and tasks

Problem: Front page header was not helpful enough, lots of white space

Lesson: Research landing pages and web service layouts to use space more efficiently

Problem: Search could be better scoped or more comprehensive

Lesson: Investigate potential users in earlier planning phases to plan for what they would search

Reflections:

Spencer: Always knew but always needs repetition, Test Early, Test Often. Also glad that in this group, members almost always showed up to class, so that even if there was no meeting scheduled, if something came up it could be addressed. That regular meeting time is a huge benefit to group work, even if we didn't work directly on the project during this time.

Jithu: Rather than immediately working directly with a markup framework (in our case, Bootstrap), I feel like we would have benefitted by using a prototyping tool like Balsamiq. While Bootstrap is a great way to rapidly prototype a functional design, it is far too easy to just use the prepackaged design elements like buttons and menus. While this packaged design is certainly clean and aesthetic, it was far easier to just keep those elements rather than experiment with potentially better elements. Additionally, I learned that a design needs to actually work to some degree before giving it to a user to test. While it is easy to tell a user to pretend that search results appear or that the login actually worked, I think it affects their feedback and the outcome of the test. Finally, it certainly helped that this is a service I could see myself using in the future. Many of my design decisions were based on things I would like to see as a user of the site.

Andrew: It would be better to start designing with lower fidelity prototypes to allow a more malleable startup idea; we had launched straight into building with Twitter Bootstrap's ready-for-deployment quality elements. Also, testing is not a phase meant for final iterations; rather, each iteration (for this class, each deliverable) should have been shorter with design, test, and evaluate subphases. That way we could identify problems and strong points more rapidly. Finally, as this was a four-person project, scheduling conflicts were rife, with only weekends and weekdays late at night being common free time to work. This caused our work schedules to either be seriously disjointed (working on separate things with little feedback) or single 5-hour sessions grinding out our design and testing.

Michael: I think we would have benefitted more from merging the designs a bit more than we did. Some of the prototypes had more detail than others, which made judging them a bit difficult. The main reason I wanted the design we went with was because it was already coded and somewhat functional, so I could have evaluated the designs more fairly.