SCOTT KLEMMER STANFORD HCI GROUP

PhD Students
Jesse Cirimele (382)
Nicolas Kokkalis (3A, proposal)
Chinmay Kulkarni (377)
Ranjitha Kumar (372)

2012-13 Rotation Project Ideas

2012/13 rotations
Jonathan Bassen *(Fall, CogAid, slides)*
Justin Cheng *(Fall, Online, slides)*
BS/Masters
Kristen Leach (*CogAid*)
Maxine Lim (*Omniscent*)
Kathryn Papadopoulos (*Online*)
Cesar Torres (*Omniscent*)

2011/12 rotations
James Hegarty *(Fall, TaskGenies, slides)*
Lahiru Jayatalika *(Winter, CogAid, slides)*
Joy Kim *(Winter, Bricolage, slides)*
Haden Lee *(Spring, Bricolage, slides, examples)*
Arvind Satyanarayan *(Fall, Bricolage, slides)*

Summer 2012
Cesar Torres (*Ranjitha*)
Maxine Lim (*Ranjitha, slides*)
Arun Prasad (*Nicolas*)
Thomas Bridges-Lymon (*Nicolas*)
Ryan Globus (*Nicolas*)
Arzav Jain (*Chinmay*)
Elaine Chen (*Chinmay*)
Nisha Masharani (*Chinmay*)
Tonya Yu (*Jesse*)
Justin Lee (*Jesse*)
Kristen Leach (*Jesse*)

Information about working with Scott's group:

- The **first day** of your rotation:
  1. Set up your laptop at your desk in 3B. We can provide a Macbook, monitor, keyboard and/or mouse for the duration of your rotation. Send me email with any equipment you have so we can keep track of it.
  2. At your 3B desk, **read this page** and its linked documents. I estimate it will take you an hour or two to read and contemplate, including following the links (but not reading the mentioned books). By the end of your first day, **Email me** saying either:
     a. that you have read this Web page, and that you agree with and will follow the practices outlined here. Or,
     b. that there are some things that you don't understand or agree with, and list them.
  3. **Put your name** on your office door. (Or, if you're in an open area, above your desk.) I like Spell With Flickr for printing nametags; you can use anything you like.
  4. Take the human subjects certification.
- **Set an intention** at the beginning of your rotation, write it down, and share it with me. What will you accomplish? Make a **conservative goal** that you're reasonably certain you can accomplish, and a **stretch goal** that might happen if things go really well. Having a clear intention will enable you to focus your efforts, work effectively, and know when you're done. Ideally, we should have a rough plan before your rotation begins. Look at the presentations for recent rotation students and build on their work whenever possible. Your plans will likely bounce around a bunch for the first week-and-a-half. You should have a pretty clear goal by the middle of your second week. **Update** your conservative and stretch goals each week when you send out your weekly update email.
- **Organize all research files** (design work, code, writing, presentations, data).
  - Store **all** research files in one of four places (and **nowhere** else, because duplication causes confusion):
    - **Google Docs/Spreadsheets** (*data files, plans, outlines, and paper/proposal drafts*)
    - **Dropbox** (*talks, final papers, proposals, and other "rich" documents*)
    - **The HCI partition of the graphics file server** (*Web content*)
      From the Finder, connect to afp://ee-afp.stanford.edu. Use your graphics login. If you don't have a graphics login, email support@graphics. Database-backed information, like publications, lives in the graphics MySQL database.

    - The StanfordHCI **github / svn** (*code*)

  Use a stanford.edu (or cs... or hci...) email address for your account on these services.
  - **Clearly name and organize your files** so your collaborators can find them without having to ask you. For all of your research (study data, proposals, publications, talks, *etc*.) **name folders ProjectName-VenueYear** and files ProjectName-VenueYear-Version, e.g., dtools-UIST2006-17, or Bricolage-NSFIIS2010-23. Keep all drafts in subfolder named 'drafts', so that the only file in the main folder is the current version. **DO NOT** use generic filenames / titles like "my_chi_paper.doc", "paper.pdf", "Experiment Design", "talks", or "research." This holds for Google Docs too: **organize Google Docs and Spreadsheets** into clearly-named collections, like "cs147 Fall 2010" or "Dyad Study 2010 -- Data for Steven Dow et al. CHI 2011". I encourage you to share at the collection level.
  - There are three reasons why it's essential for you to be very diligent about keeping all of your files consistently organized in a common, shared repository.
    1. if your computer is lost, damaged, or stolen--or your hard drive crashes--**no** research should be lost.
    2. shared repositories minimize **coordination** problems when collaborating, and provide versioning support.
    3. it helps me and other members of the group **find** your work when you're unavailable (or have graduated!).

    As the group has grown, different people have gravitated toward different data management strategies. And I empathize with that. But, when different people or sub-groups have different strategies, that makes it much more difficult for me to find information and for sub-groups to collaborate with each other. I don't really care *what* repository we use. The key criteria is that whatever we use, we all use it *consistently*, and that it works for people's wide-ranging needs.
  - **Mark all your code** as BSD Open Source. Clearly name all your variables, so that others can more easily understand your code.
  - **Show Me Your Data:** Please share data early and often. Organize your spreadsheet so that the main dependent variable is a column, with its entries in the rows. **Title each column clearly** in the first row. Anyone in the group should be able to interpret them.
  - I strongly recommend the **one-computer** strategy (and suggest a Macbook). You'll spend less time keeping your machine up-to-date. And you'll never have the problem that you can't do a task because a file/application/password/configuration/setting is on your other machine. Why a Macbook? It's what everyone else in the group has, and consistency helps collaboration.
- **Find, and post, persistent information**
  - The HCI private page contains poster/slide templates, links to source code, checklists for the czars, and how-to information. Place a link to your source code (with instructions) and data files on this page so that others can find them later. If something's missing, **add it**.
- **Communicate**...

  ...with me:
  - **Email**. If you have a question for me or would like to share an update, send me an email. Give it a specific, action-oriented title. My response latency is generally inversely proportional to email length. Brevity is divine: follow the email charter.
  - **In person**. You're welcome to drop by my office if you have a question. In general, late afternoon is a good time to drop by. I try to keep mornings interruption-free for longer, focused work.
  - **Complex, thorny questions** are best handled in person, rather than email. They often require discussion to work through, and somehow I often find that in-person conversations are just better stuited to complex topics than typing email. (Also, complex emails get buried.) Shoot me an email ahead of time, so that I can noodle on it for a few days. Say "here's a heads-up that I'll have a question for you at our meeting". That'll give me more incubation time. My 'reply' will be in person.

  ...meetings:
  - When you **schedule a meeting** (*e.g.*, if I or others agree to meet with you about research) send all attendees a meeting request through Google Calendar or iCal so that is shows up in everyone's calendar.
  - The evening before/morning of the meeting, send all attendees a **reminder email** that includes a one-sentence agenda. (Feel free to send a longer agenda. The key is to send at least something.)

  ...with the group:
  - **Mailing lists**: Join the hci-research, hci-gates3, and pcd-seminar majordomo lists by going to http://lists.stanford.edu. Ask me to add you to srk-students@cs.stanford.edu.
  - Our **Google Calendar** lists the speakers for Wednesday lunches, Friday seminars, conference dates and deadlines, and other HCI-related talks and events. Log in, and in the lower-left add stanfordhci@gmail.com.

  - **Hang posters** that you present in the 3B Hallway, and email the digital versions to Maria Kazandjieva (mariakaz) for display on the first floor. It helps breathe life into the space.

...with the world:

  - Have a **Web page** for your project, and keep it up to date with papers, video, code, news, pictures, etc. See the ButterflyNet and d.tools pages for examples.
  - Use a **stanford.edu** email (or cs... or hci...) for all internal and external research communcation, shared documents, and repositories. I know who you are, but others may not. Email domain is the letterhead of the digital age.
  - The CHI work-in-progress and UIST poster/demo tracks are a great way to **share in-progress work** with the world and get feedback. These are not 'archival' venues -- they're an invitation to discussion. (They don't go on the 'papers' section of your CV and they don't preclude or inhibit future publication in archival form.) My group sees three main benefits to your sharing work through these venues: 1) You get practice with the publishing cycle: writing, revision, submitting, peer review, and presentation. That way, when it comes time to publish a 'real' paper, you will have been through the cycle already. 2) It gives you a chance to see how the world reacts to your work, offering you the opportunity for course correction. It's valuable to know sooner rather than later if someone is skeptical of your result, believes you're missing a key piece of related work, sees an opportunity you missed, or understands things better presented one way than another. 3) As Ranjitha writes, it's a "great end-of-rotation deliverable. Often students are working in the context of a larger research project and are rushed to complete research by the end of the quarter... having a written deliverable really helps them understand what their work meant in the scope of a larger research direction. It also creates a sense of ownership over that work, so that it isn't something that they do for a quarter and never think about again."
  - Demo your work regularly -- it's a great way to get feedback and users. Remember, **a demo is a story**, not a functionality test. Start by explaining why someone would want to use your system, and walk them through a scenario.

I expect you to be...

- **Present:** I strongly encourage you to work in Gates (and have lunch with your peers). Having a dedicated place for work increases your productivity. And working alongside others increases knowledge-sharing and builds esprit de corps. The importance of working in Gates is highest for junior students because it will help you learn from more senior students. The Web page Characteristics of Graduate School Superstars puts physical presence at the top of its list: "Superstars were observed to be physically present in the department, during and often after working hours." When I talk with faculty about their graduate school colleagues and who of their cohort went on to have impact, people often remark how it was the people who were in the lab most were the ones that went on to have important, impactful careers. (I believe this remains true, even in an era of laptops and WiFi.) I believe collocated work is sufficiently important that I'm willing to use bribery as a catalyst: we'll provide a large monitor for you, which demonstrably increases your productivity. If there are other things we can do to encourage people to work in Gates, let me know.

  Attend three weekly gatherings. At all three, listen actively and **ask questions**. So that you can actively participate, put away your email, and disable interruptions from your devices. Sit at the central table, not on the perimeter. If the central table seems full, ask people to make room. At the end of lunch, help clean up: put extra food in the 3B kitchen, wipe spilled food off surfaces, and open the blinds.

  1. **Group research lunch**: Wednesdays @noon, Gates 104. These are informal, discussion-driven talks where you should regularly ask questions. **Sign up** to give a wednesday talk at the end of your rotation/internship. PhD students should expect to present 2-3 times a year. Sometimes it will be a practice talk for a formal presentation, like a conference talk. Other times, it will be a chance to share early thoughts on a new project. If you're the speaker, ask a friend to take notes for you.
  2. Lunch with the **HCI Seminar** speaker: Fridays @noon, Gates 392 *(Academic year)*. We get a world-class set of speakers and this is your chance to ask them questions. This is also your chance to offer a 'calling card' about your research. At the beginning of lunch, we'll go around and do introductions. Say something provocative, not administrative. DON'T SAY, "I'm working with Scott on the donut project." a) that's boring. b) they don't know what the donut project is. DO SAY, "I'm investigating whether we can achieve a new level of donut tastiness by cooking them in sous vide." "I'm investigating" is one good lead. "I'd like to find out" is another. Don't dine and dash: eat the food, attend the seminar. (In fact, officially register for

cs547 -- it's one unit and helps get us resources.)

3. **Weekly project meeting**, Gates 384. *(Fridays during the academic year; Wednesdays in summer.)*
   - Each week, before our meeting, send a **weekly update** email to srk-students@cs comprising: what you accomplished that week, what you plan to accomplish in the coming week, and what your updated safe and stretch goals for the quarter are. Use that email to structure our meeting.
   - At our scheduled meeting time, please **come in** and set up. Do not wait outside (even if someone else is in my office, or I haven't arrived yet). Plug your laptop into the shared display. Open up your to-do list for the week, any artifacts you would like to share/discuss, and a place to take notes. Prior to our meeting, close all extraneous browser tabs and **disable all interruptions**, like IM. Email/chat notifications should not pop up during our meeting. (For fast web and AirPlay display access, connect your WiFi to EZE.)
   - Begin the meeting by presenting the **agenda**.
   - **Take notes** about what we discuss and how you will follow up. Right **after meeting, email me** the summary of decisions and your revised plan for the coming week.
     Taking notes is important for two reasons. First, they will serve as an important reminder for you. Second, they convey to everyone the meeting that what we are discussing is important. (I use Evernote. Many use Google Docs. Others like paper. You can use anything you like, as long as you're consistent. Do not have a set of random files scattered on your desktop. I recommend against using TextEdit because somehow it seems to encourage people to have lots of scattered files named 'untitled'. Digital tools have the advantage that they're searchable. Paper has the advantage that it makes clear to all what you're doing... and it won't buzz or tempt you to check your email.)
   - **Keep a rank-ordered list** of programming tasks with estimated time. You can use anything you like; I like Google Spreadsheets because they're easy for multiple people to edit simultaneously. **Show this list** in our weekly meeting so we can discuss it.
   - Having a **consistent meeting time** helps keep me sane. I love research and working with students. I hate spending time on coordination emails. Work hard to make our weekly meeting, so we don't have to reschedule. That said, there will be times when our scheduled meeting won't work, e.g., one of us is out of town. In those cases, please **let me know** well in advance (e.g., when you book your travel). My calendar can keep track of things far into the future, and so knowing a month in advance that we won't meet on a certain day provides me with valuable flexibility in case something else comes up.

- **Proactive**:
  - **Tell me** what's on your mind, and what you believe the next steps to be. Do those next steps of your own initiative. Find related work, read it, and share what you think the connections/insights are. Form ideas for new projects. Etc.
  - **If a problem arises, let me know**. Examples: a conflict or misunderstanding with an intern, or a disgruntled study participant. I understand that problems happen. If you tell me, we can work together to fix it as best we can -- and create guidelines and safeguards to prevent similar problems in the future. Often, my wisdom and perspective can help. If you don't tell me, I can't help -- or prevent similar problems in the future -- yet I can be ethically and legally responsible.
  - When you see something 'broken', **fix it** -- at least raise it as an issue. If you have an idea for how to make something better, say so. It's part of having esprit de corps in our group. It doesn't matter what your 'role' is, if you have ideas, share them. Follow this principle for everything from high-level research ideas to low level brass tacks. (For example, if there are errors or omissions in this web page, suggest changes/additions.)
  - Speak up when you're **confused** or don't understand something. This is true in talks (if you're confused, someone else is too). And it's especially true in our meetings. Nodding along when you're confused isn't polite -- it wastes both our time.
  - Actively **build theories**, think about how to test them, and share your theory-building thoughts with our group (and eventually the world). What are the implications of your work? What does it mean? What principles does it illustrate that were previously unknown? What are the limitations, and what next steps would best address them? How can you frame your work to have the broadest possible impact? If you're looking for strategies for coming up with ideas, I highly recommend William McGuire's Creative Hypo] Generation.
  - Over the years, I have participated in many faculty conversations about 'what is the most important trait in a PhD student.' The most common answer -- regardless of discipline -- is **initiative**. While of course many skills are important, initiative reliably tops the list. (This principle is pervasive: entrepeneurs, artists, ... also list initiative as the most important trait.) If you are a wallflower or sitting around waiting for someone to tell you what to do, you are failing to meet your responsibility.

    - **Form your own opinions**, don't blindly follow what I and others in the field believe. This doesn't mean that you can believe whatever you want regardless of your knowledge or the evidence. Quite the opposite: question assumptions, dig deep, and investigate.
    - Don't let the best be the enemy of the good. **Do something now.**I can't tell you how many times I've had students who have the same to-do item on their list every week. Because it's a big ball of thorniness. Don't suffer from this analysis paralysis. Take the first step: make an outline, fix the stuff that's obvious, whatever it takes to make some progress. Don't have something linger forever because you've wanted to find the time to think about it but just haven't. The do-something-now strategy increases your desire and ability to execute, and lowers your stress level.
    - **Iteration** is your friend. Iteration is not only valuable in system-building. It's also really essential for writing. And for designing experiments. For both your writing and your experiments, plan to go through several versions before you get to something excellent.
    - **Manage Expectations** of the people you work with by keeping them updated on your status. Ranjitha writes, "first-years are sometimes too 'eager to please' and bite off more than they can chew. Also, they don't seek help immediately when they are stuck; they want to show they can fix it themselves, and then weeks later they speak up about it." In a ten-week quarter, we simply don't have the time to recover from your being stuck on something for 3 weeks without telling us. I fully understand the impulse to try and hide the fact that one day's productivity is low with the plan that you'll make it up the next day. Because it's tough to share struggles, you may be surprised by our empathy -- we've all been there. Don't be like the gambler who hopes to hide a losing evening by doubling down and getting lucky; it rarely goes well.
    - Minimize risk and maximize flexibility through **rapid prototyping**. And remember, the Wizard of Oz is your friend.

- **Focused:** Dedicate time each day for serious research, and use that time to make progress toward the research goals you outline in your weekly plan. During this time, limit distractions: **turn off** your email/chat/... so you won't be interrupted, and close all applications/documents/tabs that are unrelated to that day's research. If you have 18 applications open, 27 random tabs open, and six things bouncing and blinking on your screen, that's not an environment conducive to the focused, reflective work that is essential to performing deep research. Of course, not all your time needs to be deep and focused, but some of your time should be.
    - In general, the best predictor of performance in any field is perserverence and time on task. You may have heard Anders Ericsson's oft-publicized estimate that it takes 10,000 hours to become an expert in a domain. That's obviously an oversimplification. But the high-level takeaway is spot on: expertise requires many hours of focused practice. And – even at the highest levels of achievement – time on task is the best predictor for differentiating the really good from the pretty good. Malcolm Gladwell summarizes Anders Ericsson's research on this wonderfully in Outliers. Corollary: high-quality time is essential. If you're chronically sleep-deprived, you can't have high-quality time.
- **In selecting your research project**:
    - remember that the state of the practice may not reflect the **state of the art**. Doug Engelbart invented the mouse in 1964; it was first widely available as a product in 1984 -- 20 years between invention and commercialization. Not everything takes that long. Conversely, some great ideas never see the light of day. I bring this up because you don't want to work on a project for a year only to realize someone else invented it a decade earlier. What's the best way to avoid 'wasting' a year because you don't know prior work?
    - To be a great researcher, you need to have **encyclopedic** knowledge of the literature. Read voraciously. Until you have encyclopedic knowledge, I recommend working on a project related to a current research thrust in the group.
    - So that you can leverage each other's talents -- and build a community -- it's important for our research group to have a **coherent** intellectual agenda. A goal of our research group, to borrow a phrase from Cornel West, is to "be a thermostat, not a thermometer." That is, our goal is to set the conversation for the field through focused leadership. This focus is defined by an iterative, dialectical process that everyone participates in. The senior PhD students play an especially important role. We should be doing work that other people wouldn't think to do. And then we should inspire others to build on it. Science is like masonry, building up structures brick by brick.
    - A senior PhD student will be your rotation mentor. The best way I know how to learn to be a researcher is the **apprenticeship model**.
- **Apply for fellowships.**
    - If you are an incoming or first-year student (and are eligible), apply for things like the NSF Graduate Fellowship.

    - If you are a senior PhD student, apply for dissertation fellowships and things like the Google, Microsoft, and Qualcomm fellowships.
    - Proposal writing is one of the world's most difficult tasks because it requires you to envision and project a clear, coherent, exciting vision of the future. To manage this difficulty, I suggest a four-step process for writing a fellowship application
        1. At least a month before the deadline, select your letter-writers and contact them. If I'm one of your letter-writers, follow these instructions. In your request email, offer to write a draft for the letter-writer. If they accept...
        2. send your letter-writer a draft letter for them to work from. This can be daunting because it can feel like bragging. Ask me for an example letter that you can use as a template.
        3. At least two weeks before the deadline, bring a detailed outline to our weekly meeting for discussion. Remind me to expect this, and put it in your weekly plan, so I can help you stay on task.
        4. Turn the outline into a proposal :)
- **A team player:** We have an amazing research group. That doesn't happen automatically. It's constructed every day through your active participation. Provide your peers with feedback and help each other out. (A corollary: don't be a jerk.)
    - To keep the group running smoothly, there are several support roles that graduate students play, serving as Webmaster, PhD admissions, faculty hiring, lunch czar, seminar, and CURIS leader. Each year, you should expect to **play one of these roles**.
    - My group takes a somewhat socialist approach to supporting PhD students. If you are a PhD student in the group working hard towards one of the group's major research goals, everyone in the group will work hard to fund your work rain or shine. This means that in general:
        - when the group collectively brings in sufficient funding, that collective achievement will support your work. (provided, of course, that it contributes to a major thrust of the group and you are working really hard and making serious progress on research. We don't have a mechanism to support 'random', unrelated research ideas. And, given the value of cumulative research, that generally wouldn't warrant support even if we could. It's also not a good use of the group's grants to support students that aren't working really hard: if you aren't serious there are a lot of people that are. And we can't sustain funding unless we can **demonstrate amazing results**.)
        - Your contribution to this process is a) to **help write grants** and apply for student fellowships, and b) provide your peers with **feedback** during Wednesdays lunches and to drafts they send to srk-students.
        - Over the course of your PhD, you should **expect to be a lead (co-)author** on 1-2 'substantial' proposals (e.g., like an NSF grant that will support 1-2 students for 3-4 years) and a couple smaller proposals (e.g., a private-sector opportunity at the scale of a couple pages of writing for ~1 student year of support). Over the course of a 5-year PhD, expect grant-writing and maintence (annual reports, talks, etc.) to add up to 2 months of work on your part. (Big grants will take you 2-3 weeks. Short grants will take you 1-3 days.) In addition to providing funding, grantwriting is a great venue to force you to clarify and flesh out your ideas; most students find this an intrinsically valuable activity. And your grant/fellowship proposal can also be your thesis proposal -- kill two birds with one stone.
        - Funding is best thought of as an expected-value activity. Let's say our hit rate is ~50%. That means we'll need to apply for double the grants we need to fund the group. It also means that any given person on any given year may not hit paydirt. And on another year, one may hit paydirt multiple times. The intrinsic randomness in this process is one major reason we operate on a quasi-socialist model. So, for example, as a first-year student, you may be covered on a grant written by a senior student and I. Then, you may write a proposal that doesn't get funded. Later on, the department may successfully nominate you for a fellowship. Regardless of if/when/how many of these things work out, each of us contributes on the grantwriting side. You get the idea. We all work to bring in funding. And because of the randomness, we all share the rewards.
        - The group will work to support you on a consistent, long-term trajectory so that your dissertation research can build cumulatively. That said, it may need to bend a bit to accomodate current funding constraints. For example, if the group has funding on mobile research, it will be important for your research to have a connection to mobile.
        - The best way to write a grant is to **write down the research that you would like to do**. Think of proposals first and foremost as a research plan, as if you were writing to a colleague. Use it as an opportunity to brainstorm and flesh out future projects, and read and understand related work. The 'spin' of grantwriting should absolutely play second fiddle and happen only after you've figured out your research goals.

        - With grant-writing, it's especially important to begin with a clear and detailed outline.
- **Write clearly**! It's very important to me that we write clear papers and collaborate well. I also care deeply about what a paper *means*. I hope you do too. Therefore, you should expect many revisions, especially in the introduction.
    - **Write an abstract and outline** of your research paper very early on in your research project. This will help you focus your efforts, and make sure that your activities align with your goals. It's a real bummer to realize after the fact that you spent three months building something irrelevant. Or that you spent two months running an experiment that doesn't show what you wanted it to. Chinmay and Ed Chi have an implementation of the outline-early strategy that I strongly recommend: write your outline as a slide deck and give it as a talk. They point out that this forces you to articulate what's important and focus on it. Also, slide software is provides a good interface for rearranging content.
    - The best way to increase the odds that your paper is **rejected**: do most of the writing at the last minute. The best way to increase the odds of acceptance: have a complete and coherent draft several weeks in advance; solicit and incorporate feedback from other PhD students in the group. Your peers can serve as 'pilot' readers so you get the kinks and errors out of your paper before reviewers see it. If you don't solicit and incorporate feedback, the reviewers will get confused, find errors and reject your paper. There's a saying, "an ounce of prevention is worth a pound of cure" and it's true here in a big way. A small amount of up-front discipline and preparation can hugely increase odds of acceptance. By contrast, getting rejected likely delays publication by a year -- and a year is a long time to unnecessarily have a big incomplete goal occupying your time and attention -- you also risk getting scooped, made less relevant, or needing to rework your framing based on publications that appear in the meantime. Given all the stress and work of those downsides, don't you think it's a good idea to draft your paper early and focus on writing well?
    - Providing meaningful feedback on writing takes a significant contiguous block of time. While I try really hard to get comments to you as quickly as I can, you should expect that it will take me **1-2 weeks** to get you feedback on a paper draft. Sometimes I may be faster. And when I have a major upcoming deadline it may be slower; I may not always be able to get you feedback as quickly as you or I would like. Also, let me know what feedback you would like. In general, my editing will include careful reworking of sentences, detailed changes to graphs, etc. If your paper isn't ready for that yet, and you'd only like structural feedback (i.e., you'll send me a version for detailed editing later), please let me know. That way, neither of us spends time 'getting the design right' before we've decided on the 'right design.'
    - The edits I make are **not optional** suggestions that you can ignore if you wish. For every single edit that I mark, either make the edit or add a comment explaining why you didn't make it or did something differently. Dialogue is essential: if you disagree with an edit (e.g., if my revision would be incorrect or read poorly), don't just leave the text as it is. Think about what motivated me to try and revise it. And propose an alternative version that is more accurate, clearer, ...
    - It's imperative that the archival version of our papers is clearly written and error-free. To insure this, please provide me with a version that you believe to be absolutely **100 percent** done. I will make a final pass with a fine-toothed comb. Given that I will need to find a block of large block of free time for this careful pass, expect that it will take at least a week -- longer if I have travel or a major deadline. To the best of my memory, every "final" paper I've been given had at least one typo. And every edit has a chance of introducing a new bug. That's why it's essential for me to read a version that you are 100 percent done editing -- every figure, reference, data is fully complete.
    - **Write durable, impactful papers.** I suggest a MiniMax strategy: in order to cover maximal terrain, embed minimal assumptions in the writing. You want as many people as possible to get something out of your paper. So don't make it too domain specific. Imagine people reading it a decade from now. Write it so that it will still feel relevant, and not dated. Talks and videos, on the other hand, are more ephemeral, and consequently its more appropriate to tie them more strongly to the time and the domain.
      Related to the MiniMax strategy is Stu Card's weak methods/strong methods strategy. This is useful for both both system building and writing. Have a general ('weak') strategy that makes few assumptions -- this will have the broadest applicability. Then, have a domain-specific ('strong') strategy that provides more power in a particular setting by leveraging features of the domain.
      Also related is Phil Agre's idea of 'turf', which I think is deeply insightful. When you frame you research and when you share your research, you should try to create and capture as much 'turf' as possible (...but not any more than that because reviewers will penalize you for claiming more than you can support). As Agre points out, creating turf sets the stage for follow-on research and set up a paradigm for further normal science.
    - **Read and follow** Herb Clark's: Everyone Can Write Better (and you are no exception)

- **Use standard sections.** If you're writing an experimental paper, use APA style: e.g., have sections marked METHOD, RESULTS, etc. Consistency helps readers.
  - **Titles**: At least at the beginning, title your paper with its contribution. For example, Trust Breaks Down in Electronic Contexts but Can Be Repaired by Initial Face-to-Face contact. I've never failed to remember the contribution of that paper. A counterweight to this advice: all things equal, shorter titles are better. If the paper introduces a system, the title in the camera-ready (but not for review) should begin with the system name, followed by a colon. Why not for review? Omitting the system name during review reminds both you and the readers that you're making a research contribution, not 'merely' building a system.
  - **Graphs**: Maximize the data-to-ink ratio. When you have bar charts showing an average, augment the bars with whiskers showing the standard error.
  - **Tables**: Get rid of the chartjunk: *i.e.*, have no rules if possible. If rules are absolutely necessary, make them 1/4 point. Whenever possible, embed sparklines or other iconic data graphics into tables. Visual presentation facilitates rapid comprehension and comparison.
  - **Figures**: If possible, put a summary figure on the top-right of page 1. Except on page 1, figures should go on the bottom of the page. Bottom-aligned figures are preferable to top-aligned figures because there is less interference/confusion between the caption text and the body text (because the figure itself serves as a separator).
  - **References**: Render references with real reference management software, like EndNote or Mendeley. Don't use Word's built-in system. Why? In addition to automatically consistent formatting and ordering, EndNote/Mendeley will check whether a reference is actually used, so there are no reference orphans. And at some point, you'll need to change how the refs are formatted (what information is included, what's in italics, whether and how to abbreviate with et al., ...). It's faster and more robust for a computer to do this.
  - **Appendices**: If possible, include your experimental materials (e.g., instructions to participants) in an appendix. If there isn't space, post them on your project Web page.
- **Editor**: For the most part, text composition and document layout are separate tasks. Separating these phases will save you time, increase your flexibility, and minimize wasted effort of fidgeting with layout or carefully placing something that may change later.
  - **Draft and share** your papers with Google Docs so that my edits address the latest version without blocking your progress. That way you can continue to work, and when I read your paper, I will see the latest version. Google Docs are ideal for this because if I get halfway through, you'll see those comments immediately. Neil Patel really pioneered this in our group.
  - Once the paper is mostly done, move to an editor that supports layout. Hopefully, a web-based tool like Google Docs will eventually be able to do this. Right now, our two main choices for a **layout editor** are Microsoft Word and Latex. Both have major idiosyncrasies, and both have major drawbacks. (My sense is that other options like Apple's Pages aren't quite ready for prime time -- for example, Pages only works on Mac. InDesign supports fancy layout, but is overkill for our purposes and limits collaboration since few have it.) Word's biggest benefits: WYSIWYG editing AND commenting; track changes; interactive spelling and grammar feedback. Word's biggest idiosyncrasy: figure placement (it's both flexible and deterministic once you know how). Word's biggest drawback: good reference management requires separate software (like EndNote or Mendeley). Latex's biggest benefits: text-only representation integrates well with source control; great for math; bibtex. Latex's biggest drawbacks: time wasted compiling and rendering documents; reading is slowed by monospace editors littered with formatting syntax; commenting/revisions are clunkier; layout is less flexible. (don't believe me? One quick example: send me a latex paper with a figure that spans 1.5 columns.) I have written papers with many dozen people, and this experience has taught me that Word's drawbacks waste less time than Latex's. (Sometimes, it does feel like a race to the bottom.) So, unless you have a math-heavy paper, please use Word so that we can collaborate most efficiently. I want to emphasize that this is a data-driven choice rather than an ideological one, so if new data or options appear, I'm happy to revisit.

- A **poster** has two primary goals:
  1. Looks exciting from a distance (say 10 feet) so that people come over and talk to you.
  2. When someone comes over to talk with you, you can augment your explanation by pointing to visual aids (graphs, screenshots, pictures)

  A poster is not a novel. As a rule of thumb, it should have no more than 500 words of text on it (including title, authors, everything).

- **Speak Effectively** by being prepared.
    - Content first; then form. Presentation software makes it easy and alluring to spend hours making fancy slides. And that's fine, **after** you've honed the content. Your initial goal is to nail what you'll say. For conference talks: Sign up to give a practice talk at our group lunch about two weeks in advance.
    - Always arrive at the room early to **test the projector** (**and audio**, if needed). Arrive early enough that you'll have a moment to relax after you've set everything up. You don't want to be the person at the front of the room five minutes after the talk was supposed to start asking if anyone has a power supply, dongle, or knows how to operate the projector. This will significantly increase your stress level, which will decrease the quality of your talk (and your enjoyment of it). It will also sour the audience. On a related note, make sure to end on time. You are not obligated to 'get through all your slides'.
    - For conference talks, **give a practice talk** in the room where you'll be speaking the day before.
    - create **high-contrast slides** so that people can read the text: white (or a very light color) on black (or a very dark color). Why light-on-dark? When giving a talk, you'll inevitably cross in front of your slides. With a white background, the slide edge creates harsh lines on your face – and shadows on the screen. Makes for a cool art project, but maybe not for an expository talk. Dark backgrounds avoid this problem. (It's also It's long been a rule of thumb among graphic designers that for large projected content, light-on-dark is easier to read than dark-on-light. Conversely, for paper, dark-on-light is recommended.)
    - Explain the **value proposition** of the talk early: what does your research seek to find out, and what will be the take-away for the audience? Organize your talk using standard sections. If it's an experimental talk, you should have slides marked 'hypothesis', 'methods', 'measures', and 'results'.
    - Our group currently uses Apple's Keynote for presentations; you can buy it from the App Store. (Once online tools like Google Presentations get better, we may switch. My sense is they aren't quite ready for prime time.)
    - Use a remote so you aren't tethered to your keyboard. I currently use the Logitech Cube because it's compact. I also recommend the Targus remote. The Apple one is pretty but in my experience unreliable (probably because it's using infrared, so requires line-of-sight to the receiver).
- To **design effective experiments** and analyze their results...
    - I highly recommend taking Psych 252 (fall) or Psych 254 (winter)
    - for guidance in book form, I recommend Doing Psychology Experiments and Statistics as Principled Argument. I have copies of both in my office, which you are welcome to borrow. (As long as you *immediately* send me an e-mail that you have borrowed them – email is my memory for who has my books. When something is missing, I search my email.)
    - for practical guidance on selecting statistical tests, see the book Learning to use statistical tests in psychology. For example, it walks you through how, with three conditions, to use an ANOVA to test for an overall difference, and follow-up corrected t-tests to test for pairwise differences.
    - Two papers from the HCI Group that are excellent templates are Blueprint and Parallel Prototyping. Note that in both papers, care is taken to have a **minimal-pairs** experimental design, where the two conditions vary in only one dimension. Conditions that vary in multiple ways prevent ascertaining cause. (There are times when one might elect a non-minimal-pairs design, but one should do so eyes wide open as to the challenges.) Note also that both papers present multiple related experiments to support their conclusions. One good experiment can accomplish a lot. Multiple related experiments can often accomplish a lot more -- especially when one is trying both to build theory and to inform practice.
    - **Write the paper before doing the research**. (I got this from Bjoern who tells me he got it from Marc Levoy.) It really helps crystallize whether what you're doing has the potential to uncover what you hope to learn/contribute. I think this strategy is especially valuable for experiments. At the very least, write the intro (including hypotheses) and methods (including measures) sections pre-experiment. And it's useful to make up a results section. That'll let you do things like estimate effect size so you can get a sense of how many participants you need. And if you show it to others, that'll help debug the experiment. For those that have mastered the write-first approach, the next level is to use the paper as a spec for what you need to implement. This prevents you from wasting time on things that don't matter from a research perspective.
    - oDesk is a great place to get participants. (Among other things, you can constrain based on geography and 'test' performance on various skills.) For crowdsourcing n00bs, Steven put together an introductory slide deck.
    - Use **specific measures**, concrete questions, and comparison to deliver meaningful results. ...and avoid "people liked it" studies that come from Likert questions like "this is a useful tool: agree/disagree".
    - **Reuse** existing measures and paradigms. Two reasons: 1) when multiple experiments use the same measures, it's easier to compare the results. This helps build knowledge cumulatively. 2) It's hard to come up

  with robust, bug-free measures. Stand on the shoulders of giants -- no use reinventing the wheel. Reading widely will help you discover valuable methods and measures. Douglas Adams wrote that flying is the art of throwing yourself at the ground and failing. I feel its often valuable to take a similar approach to innovation. Try to solve new problems with existing methods. I promise that sometimes you'll fail, and be forced to invent.
  - **Pilot** the exact experiment you intend to run before running it with a full slate of participants. I have never seen a bug-free experiment, ever. Sometimes, these are software bugs. Other times, it's errors with wording, flow, measures, ...
  - Remember to begin your experiment with a consent form. In general, ask demographic information at the end (to omit the possibility of a stereotype-threat-like influence on the task).
  - Before launching your experiment, please send me the final version of all your materials: recruitment, tasks, everything. This is important for two reasons. First, I may catch bugs that will save you from a wasted experiment. Second, it's important that we take our human subjects obligations very seriously.
- **Book travel early:** If you attend a conference (such as CHI or UIST), please book your travel early. Most conferences have an early and/or member rate that is cheaper. Also, flights and hotels are often cheaper when booked well in advance. And Visas can take a long time. In general, you should take care of registration, hotel, flight, and visa two months in advance. If you procrastinate or forget, and your tardiness means higher registration/flight/..., you are responsible for covering the difference. To keep hotel costs reasonable, please find a roommate -- A good strategy is to email hci-gates3@lists. Attending conferences is a great way to interact with the people and ideas that compose our field. Consequently, I work hard to make this possible for as many members of the group as I can -- much more than most groups. To pull this off, I need your help in keeping costs as low as possible. Whenever you save money, that's money that we can repurpose to bring an additional member of the group. At conferences, your primary responsibility is to talk with other attendees. If there's anyone you'd like me to introduce you to, let me know.
- **Enlist undergraduates**: In January, submit requests for CURIS and Symsys interns. Having interns is a big win for all involved. You get help with your project and students get valuable training and first-person insight into the research process. If you look at the HCI group publications, you'll note that nearly all of them have an intern as an author.
  During the school year, students can perform research with you and earn academic credit by signing up for independent-study units (cs199 for undergraduates, cs399 for masters students). As a rule of thumb, students should plan on 4 hours of work per week per unit. Often, students may want to frontload this -- doing more work at the beginning of the quarter before classes become time-consuming. Front-loading is fine; even encouraged. Both during the summer and the school year, the first step is for you and your intern to create a plan together that we all agree on. I expect to see a written plan (which of course can change) by the end of the first day of the quarter. This can of course be updated.
- Here is **my advice** for graduate students. (And here is a page on How to be a Terrible Graduate Student). My #1 piece of advice: academia is a place where success compounds. When something compounds, early investment pays enormous long-term dividends. What this means for you as a graduate student is that your effort and ambition during this period of your life will have a disproportionately large impact on your lifelong opportunities.
- Strategic advice for your first year (or two) beyond your rotation with me
  - To help graduate students learn about other fields, Stanford recently introduced **short summer courses** in areas like design thinking and entrepeneurship. I strongly encourage you to take advantage of these. The deadline is usually mid-May, but sometimes courses have open spots much later than that.
  - When planning your three rotations, think about ways that you can find **synergy** and connections across multiple quarters.
  - The department provides a small amount of money for each incoming graduate student. I suggest spending it on attending a conference, like CHI or UIST.
  - Make a **CA** plan to CA the courses that will help you most. For many HCI-related students, this will include cs147. In July, apply with the department and check in with me. If we agree you'll CA cs147, don't pad your application with lots of other classes because that just lowers the chances that you'll get what you want.
  - Research comes first. Classes come second. They're important, just not dominant. (e.g., don't think of your first year as "I'll get all my classes out of the way so I can do research later." Focus on research, even in year 1.)
  - Be helpful, but not too helpful. Over your ~5-year PhD, volunteer for some departmental service role. Good choices include being the student liaison for faculty hiring or graduate admissions. Or helping to run PhD admit day. Or organizing TGIF. It's fun, important, and builds esprit de corps. But don't do too much of this (see previous bullet). If you seek a faculty job, in your final year, participate in a professional conference

committee and/or attend a small, high-powered workshop like HCIC. It's a great way to meet faculty elsewhere. When the time comes, ask me for suggesions. These things usually get figured out at the previous year's conference. (e.g., if you'd like to SV chair UIST year N, that's usually decided during UIST N-1, so bring it up with me right before then.)

- What qualifies someone to **be a co-author** on a paper in our group? (Primarily written for undergraduate and masters RAs.)
    - We expect that all RAs in the group will diligently complete their work, and our publications will credit your work in the **acknowledgements** section of the paper. You can of course include this work on your resume, and we'll obviously include it in a recommendation letter for you.
    - To be a **co-author**, your contribution should go beyond the tasks handed to you. A co-author should make an important contribution to the paper and remain invested through to completion. 'Important' means that your creative insights significantly improved the research and/or you contributed a major portion of the work in implementing a system. 'Remain invested' means that you will help see your portion of the project to it's conclusion. This usually includes a lot of tedious, non-innovative work that needs to get done to push out a good paper (running studies, figure making, etc.). You should contribute to writing and be available to help make revisions. This often means sticking around through multiple review cycles. (For CURIS students, it definitely means that your contribution continues past the official end of CURIS with sustained effort through the CHI deadline.)
    - The difference in mentality between 'acknowledgements' and co-author is the difference between crew and captain, between hourly employee and cofounder. One does what they're told and heads home at 5. The other does whatever it takes to make the project a success. This means that the choice of whether to be a co-author is really up to you and whether you go the extra mile.
    - The author list for a paper should be in place well in advance of any deadline. Given the importance and delicacy of the topic, this isn't something that can be done at the last minute.
    - I've written these guidelines down to minimize the 'gray area' for students working with the group. The potential for misunderstanding also arises when working with others. I encourage you to solicit the advice of other students and faculty. If a faculty member gives you good, meaningful advice, include their name in the acknowledgements -- their time is busy and you should be grateful. If your discussions expand beyond a conversation or two, it's best to explicitly discuss authorship with them and me so that no one has mistaken impressions.
    - The acknowledgements section of papers should always include funding source(s).
    - **Reviewing:**Your first 2-3 years, volunteer to review things like the CHI work-in-progress track and UIST/CSCW posters/demos. In general, until you have published a full paper, you should not review full papers. As a senior graduate student, you should review full papers at the top venues in your field (e.g., CHI, UIST, CSCW, TOCHI). At first, these will come as a trickle, so load management won't be a problem. If/when the load gets heavy, stop reviewing 'in-progress' stuff or papers from venues that you wouldn't publish at yourself.
- Here's my advice for writing your dissertation and organizing your oral defense.