

## Motivation

Because of its scale and its internet-mediated nature, the OMSCS program presents a number of unique challenges in the field of educational technology.

Thus far, it has successfully addressed many of these through two patterns of adaptation. One is **a shift from centralized operations to peer support structures**. For example, the program now provides many of its own TAs in classes like Knowledge-Based Artificial Intelligence (KBAI), Computational Complexity and Algorithms (CCA), and even Educational Technology (EdTech) itself – this is possible because a class of two to four hundred students is virtually guaranteed to have at least 10-20 very good ones who are very interested in the subject and can be brought on to TA the next semester. The other is **a shift from following traditions to making decisions based on data gathered from students**. For example, Dr. David Joyner's controlled experiments on grader efficacy (performed in conjunction with the summer offering of KBAI) attempt to isolate factors governing the quality and speed of feedback, so as to develop grading "best practices" which could be applied more generally in the program.

That said, there are some pieces of the program that have not finished this transformation (and thus lag far behind in scalability). One of these is the practice of academic advising as it applies to the program. Dr. Charles Isbell, a Senior Associate Dean and one of the professors most heavily involved in the program thus far, has cited this as a key risk. It's simply not feasible to support each and every student with a professional advisor (the way a more expensive "executive Master's" program might) or a faculty advisor (the way a smaller research-oriented or PhD feeder program might) – nor will it ever be.

Advising in OMSCS appears to have begun the transition towards peer-reliant, high-data solutions. Many OMSCS students currently read the unofficial course review spreadsheet and ask their peers or the Google+ group for advice on which courses they ought to take next. To a casual observer these solutions might seem to leverage the program's size fairly well, and indeed they are significantly more feasible than a traditional advising structure in the context of OMSCS. However, because a student is still limited to manually sifting through posts or reviews, the process becomes less efficient, not more, as the program scales up; this is costly to both advisors and advisees. Furthermore, the quality of advice a student receives varies quite a bit depending on when that student asks, who happens to read and respond, which details of academic background are included in the inquiry, and so forth.

With the recent departure of Mimi Haley, the program is back down to one academic advisor, with an advisor-advisee ratio of between 1:2000 and 1:3000 – the highest in the program's history. It's now more critical than ever to offer OMSCS students the planning information they need in an approachable, scalable and consistent format.

## Vision

This project draws on two distinct traditions.

From the field of *academic advising* we learn that to be effective we must gather data on each individual student (including their goals and background / transcript info), keep abreast of overall student feedback on the available courses, offer suggestions for courses that match their preferences and allow them to complete their degree, and give them access to all the information they need to evaluate those suggestions or adapt the plan for themselves.

From the field of *recommendation systems* we learn that we must supplement the factors we know to be important (a student's specialization, for instance) with other factors we can infer as we go. We also inherit a massive repertoire of techniques that we can apply to make sense of the data we obtain.

In this first phase we will not have time to make great use of the latter (more on that under "Scope"). However, it's important to bear it in mind – the design of the system in this phase (particularly its data collection) is intended to keep all those options open for later building.

### *Advising*

The initial goal of the project is to provide advising services (specifically academic advising – career or other advising is outside the scope of the project), and to provide them in such a way that their quality improves rather than degrading as the program grows larger. This breaks down into four major components:<sup>1</sup>

- User profile & history: allows the student to input (at a minimum) their previously-completed courses, desired specialization, timetable for graduation, and study hours per week. Eventually, may serve as the basis for social features (sharing one's class schedule with others to plan study groups).
- Planner: considers the user's profile and suggests appropriate courses and enrollment dates; allows the user to save their plan and tweak it later, or start over from scratch if they want to make a major change (e.g., taking a term off or switching specializations).
- Course browser: allows the user to access additional course information that may help in evaluating how appropriate the planner's suggestions are for their unique interests and circumstances.
- Review system: allows the user to give feedback on courses they've completed, to benefit both the system (which will use some of this data, e.g. effort estimates, in recommendations) as well as other students (who will

---

<sup>1</sup> Note: I took this breakdown from my own miniproposal. That in turn largely recycled the breakdown from my project readme at <https://github.com/Arkaaito/omscs-advisor>.

look at their predecessors' opinions when deciding whether to take a course).

#### *Platform for research*

The OMSCS program is one of the largest course-based CS MS programs in the country, and the average class offers more than 200 seats per term. Moreover, the current group of OMSCS students has shown a high degree of "civic-mindedness", a willingness to go to some lengths to enhance the program experience for others. These factors combine to present a unique opportunity for data collection.

I have already identified some features that are likely predictors for how much a student will like a course and/or how much value they will derive from it. In the long run, though, these give us merely a starting point; application of machine learning techniques could yield a far richer set of factors to consider for each student. For example, no official prerequisites are currently established for any course in the OMSCS program; does this make sense, or are there "unofficial" prerequisites one needs in order to do well in certain courses? Can we infer that if you do well in one course requiring Java, you'll do well in another course requiring Java?

Conversely, can our data offer value to members of the OMSCS community beyond the student body? For instance, can we hope to advise professors on what factors make a course well-loved, or how many students are likely to enroll in their course for the upcoming term (and thus how many TAs they need)?

#### *Social features*

In recent years, advisors have begun to experiment with online communities of advisees. These communities are often built on course management platforms like Moodle and facilitate one-to-one or one-to-many communication. However, they rarely encourage interactions between peers. In addition, those rare communities that *do* encourage peer interactions are still largely formed around artifacts provided by the advisor (e.g., an advising syllabus) as opposed to material created by the students.

Because of the youth and size of the OMSCS program, its social landscape is still very much in flux. We've already outlined the reasons for pursuing a peer-driven overhaul of "traditional" advising functions; might the *advising community* benefit from a similar overhaul? Even very rudimentary social features, such as the ability to add "friends" and later see which friends are planning to take which classes when, might help students form smaller, more manageable study groups within larger classes.

Not only could these features benefit students directly; they would also enhance the advising and research functionality mentioned above. If a large number of students opted into the social elements, we could begin to approach such research questions as "How much does a student's connectedness change their likelihood of dropping

out?" and "Do students prefer taking classes with 'high-profile' peers?" We could also consider the presence of known study buddies as a factor influencing a student's success in a course, and invite them to adjust their schedule accordingly.

### *Future collaboration*

In the long term, the vision for omscs-advisor must evolve to more closely fit the needs of the OMSCS community. The clearest way to get there is by eventually open-sourcing the project or at least inviting code contributions from other students.

In order to attract and sustain a significant development community, I've made the decision to build the entire app on the Django web MVC framework. Django is a popular framework; it was one of the winners of my informal (and admittedly quite flawed) poll in the OMSCS Google+ group. In addition, because Django applications are written in Python, OMSCS students who participate in a Django project will be building useful skills that can be applied in many of the program's courses.

This presents some challenges of its own, as I have never previously written a web application with Django! However, I believe my background in using other web frameworks (and recent work in Python) has prepared me to pick it up relatively quickly and with a minimum of fuss.

## **Implementation**

### *Scope*

All of the foregoing is nice, but because of the constrained duration of this course, it's necessary to pick and choose somewhat. The absolute core functionality of the advisor is the ability to recommend courses, so that's where we'll start. (Where possible, I have tailored the UI design to allow for future extension in the other two areas.)

Furthermore, building an intelligent recommender system is itself an open-ended project. In order to avoid going too far down the rabbit hole, I've limited myself to a planner that considers the user's objective, explicitly stated preferences. More advanced course ranking techniques, such as collaborative filtering, would be of limited utility with a small initial set of users. With the current user interface and data collection, these techniques can still be added once the user base (and thus the set of available data) has grown.

The trimmed version of the project includes only 12 webpages. For a slightly more detailed picture of this planned interface, see the requirements draft at

<https://github.com/Arkaaito/omscs-advisor/blob/master/requirements.md>

(excluding the items listed under 2.3, "Future Items"). The requirements document should also illuminate the proposed course metadata format, statistics, and other technical details. Some current wireframes are included in *Appendix A*; however, they are not final (nor are they very legible).

### Schedule

The following is a proposed schedule for this phase of the omscs-advisor project.

Date	Theme	Deliverable(s)
10/3	Setup, ramp-up	Register omscs-advisor.com domain Set up EC2 instance running LAMPython + Django Complete Django tutorial Create landing page, login page, and account creation page Stuff code into omscs-advisor repository Update license terms Write ToC / privacy policy
10/10	Metadata	Encode metadata for specializations Encode metadata for courses
10/17	Planner	Implement core planner UI Allow user to set planner restrictions/preferences Draft & send e-mail to key supporters inviting them to a pre-alpha test
10/24	Profile	Implement ability to save preferences Implement ability to save plan
10/31	Profile, cont'd	Add ability to configure privacy settings Add ability to view others' profiles (if not private) Write alpha user recruitment post & publish it to Piazza
11/7	Course browsing	Add course index (browse & search) page Add course detail page Film project video trailer
11/14	Course reviews	Add course review / "add to course history"

		interface Account for already-completed courses in user plan Add per-course statistics on course thumbnails and course details Write beta user recruitment post & publish to Google+ groups
11/21	Course statistics	Finish course review importer (from UCRSS), populate legacy reviews Adjust course recommendation algorithm to prioritize courses with higher rating
11/28	Review & retrospective	Final paper/presentation
12/3*	Preparation for future work	Solicit beta user feedback Plan future research directions

### Risk mitigation

The project does not depend extensively on external integrations. While use of omscs-advisor is limited to students at Georgia Tech, integrating with IAP appeared too involved for the duration of this project – therefore, identity verification will be performed purely through e-mail address verification (i.e., students will be required to sign up with a Georgia Tech e-mail address). This avoids complex data use restrictions and scary FERPA requirements; it does require that the students manually enter any course history data they might wish the advisor to consider, but even this is not such a great burden since they should have fewer than 10 courses in total (far fewer than the ~50 for an undergraduate).

However, the project does suffer from another source of risk:

*No project plan survives first contact with the user.*

As the project is somewhat over-planned, this could hit it particularly hard. The plan calls for mitigating this risk through the use of staged rollouts:

- By 10/17, a hand-picked set of users will be solicited to give feedback on just the planner UI. (I identified these students during the Spring 2015 registration period, the Fall 2015 registration period, and the first few weeks of this class. They represent a small group very interested in the subject of course selection.)
- By 10/31, any users enrolled in Educational Technology will be solicited to try the initial version of the webapp.

- By 11/14, all users in the program will be invited to create accounts and give feedback.

In the event that beta users express dissatisfaction with the app, the work currently planned for the week of 11/21 and 11/28 can be swapped out for more in-demand features. No such accommodation is currently in the schedule to allow for major changes suggested by pre-alpha and alpha users. This may prove to be a major weakness of the plan.

### **Endnote**

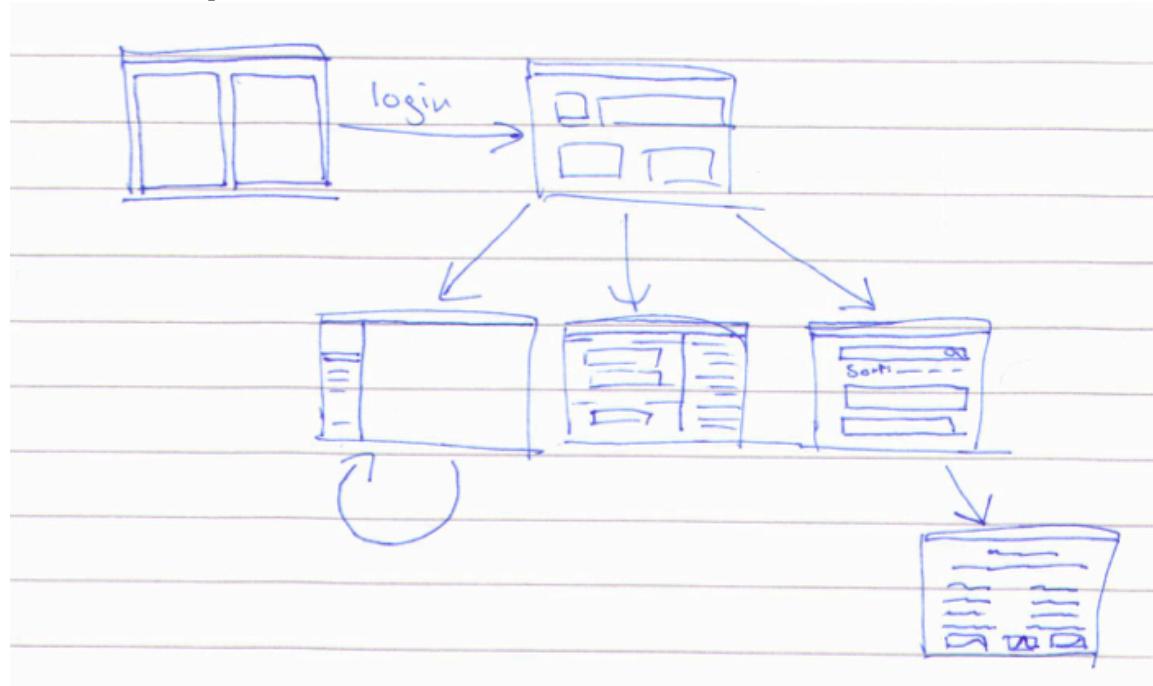
Many large swaths of this proposal were taken from my previous assignments. Pieces of my motivation section, and my “advising” description, were recycled wholesale from my miniproposal – I was very attached to the explanation I gave there.

Other than this I didn’t directly copy anything, but I couldn’t avoid repeating myself quite a bit. My hope is that the feedback on this iteration will give me a good feeling for how some of that could be avoided.

(Note: obviously, this doesn’t apply to the wireframes in the appendix. Unfortunately these are rather unprofessional-looking at the moment. For the final proposal draft they will be recreated in Balsamiq or Bootstrap.)

## Appendix A: Wireframes

Overall sitemap:



Dashboard / logged in homepage:

A hand-drawn wireframe of a logged-in homepage. At the top, there is a navigation bar with 'Home', 'Profile', 'Courses', and a circled 'Logout' button. Below the navigation, there is a profile section with a placeholder icon and the text 'TITLE'. To the right of the profile, there are sections for 'Join date', 'Location', 'Specialization: Main (unofficial, minor)', and 'Planned graduation date'. In the middle left, a box contains the text: 'You are enrolled in \_\_\_\_\_ and \_\_\_\_\_ changes? Next term (Spring 2016) we project \_\_\_\_\_ courses.' In the middle right, another box contains: 'Please review CS7630!, Sept. 28, 2015' and 'Blogpost: v1.1 released!'. At the bottom left, there is a note: 'Your plan says \_\_\_\_\_. Replan?'.

## Profile (others'):

	<p>Time zone: Location: 351 miles away Specialization: 2 in common</p> <p>Bio History Places</p> <p>Started: Planned graduation: Info: Hi, here's some stuff I wrote about myself... Contact: <a href="#">✉</a> <a href="#">✉</a> <a href="#">G+</a> <a href="#">linked</a> ...</p>
---	---

<p>Bio History Plan</p>	<p>Spring 2015 —</p> <p>CS7637 "Here's my review of..."</p> <p>CS7641 "..."</p> <p>Summer 2015 —</p> <p>CS6300 Software... (not yet reviewed)</p> <p>In Progress</p> <p>CS6460 (not yet reviewed) <small>(you'll start this class)</small></p>
---------------------------------	--

<p>Bio History Plan</p>	<p>Spring 2016 —</p> <p>CS6505</p> <p>You'll start this class!</p>
---------------------------------	--

## Profile (own/edit):

<b>Bio History Plan</b>  <b>Settings Account</b>	<b>General settings</b> E-mail: <input type="text"/> Change password: <input type="password"/>	<p style="color: blue;">may need to redo specialization edit  for multiple primary</p> <p style="color: blue;">add friends (2 people) icon later</p> <p style="color: blue;">privacy: private (mask icon), verified (checkmark), world (globe icon)</p> <p style="color: blue;">contact channels including e-mail, facebook, Google+, LinkedIn, web, Jabber, ... post poll on EdTech Plaza?</p> <p style="color: blue;">states dropdown should show 'Student', 'Prospective Student', 'Alumnus', 'On Sabbatical', 'Former Student' - 'Student' gets overridden w/ New Admit * Student Emeritus or from d.o. other things.</p> <p style="color: blue;">ability to mark as withdrawn</p> <p style="color: blue;">takes to planner</p>
	<b>Privacy settings</b> Share bio with <input checked="" type="checkbox"/> Verified Share plan with <input checked="" type="checkbox"/> Everybody Share timezone with <input checked="" type="checkbox"/> All Share reviews with <input checked="" type="checkbox"/> All You can still mark individual reviews as 'anonymous'	
	<b>Location:</b> Kirkland, WA <input type="text"/> <b>Timezone:</b> UTC-7 <input checked="" type="checkbox"/> <input type="checkbox"/> <b>Specializations:</b> <input type="checkbox"/> <b>Primary:</b> <input checked="" type="checkbox"/> Interactive ... <b>Other official:</b> <input checked="" type="checkbox"/> Teaching <input type="checkbox"/> <input checked="" type="checkbox"/> Computer <input type="checkbox"/> <a href="#">+Add another</a>	
	<b>Started:</b> Spring 2015 <input type="checkbox"/> <b>Graduating:</b> June 2016 <input type="checkbox"/> <b>Info:</b> <input type="text"/>	
<b>Spring 2015</b>  <input type="checkbox"/> <a href="#">Edit / 00</a>   <a href="#">Edit / 00</a>   <a href="#">Add class</a> <b>Summer 2015</b>  <a href="#">Review</a>   <a href="#">Add class</a> ...		
<b>Bio History Plan</b>  <b>Settings</b> Proctored: <input checked="" type="radio"/> OK <input type="radio"/> Exclude Group projects: <input checked="" type="radio"/> OK <input type="radio"/> Exclude Hours per week (standard): <input type="checkbox"/> 20 <b>Plan</b> Spring 2016:  <input type="checkbox"/> Summer 2016:		

Add course [to history] / review course modal:

NOTES

Some random thoughts on the Add Course interface:

- Should encourage, but not require, a review
- Should clearly distinguish required (semester taken, withdrawn/not) fields from optional (ratings for effort/value/difficulty, grade earned)

modal box

Course: CS6300
When: <input checked="" type="text"/> Summer 2015 <input checked="" type="checkbox"/>
Withdrawn: <input type="checkbox"/>
Help others advisor learn about you.
Value: ★★★★★ Effort max: <input type="text"/> avg: <input type="text"/>
Difficulty: <input type="text"/> Grade: <input type="text"/> <input checked="" type="checkbox"/>
Help your classmates learn about this course
<input type="checkbox"/> Leave this review anonymously: <input checked="" type="checkbox"/> Done

If triggered from profile, "When" populated based on term selected. If triggered from course detail page, populated based on most recent course attempt date; if none listed yet, user must specify manually. Corresponding course history entry will be created. Note that any time a student creates a course history entry for a course, only the latest history entry for that course should count; all others should be set to 'withdrawn' if not already set to grade <C>.

Planner UI:

⊕ Add new course

Spring 2016  
10 hours/week, 20 hours/week peak, max

CS8803  
...

CS8803  
...

Summer 2016  
15 hours/weekly, 25 hours/week peak, max

CS8803  
...

Specializations:

Primary  
Interactive Int...  
Machine Learning

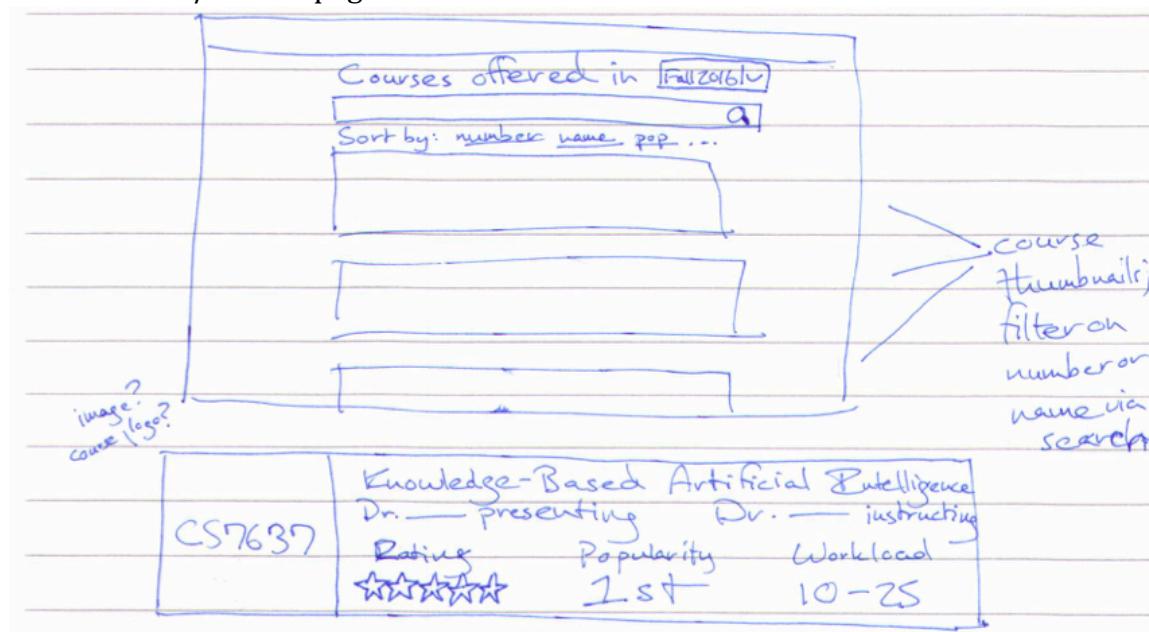
Secondary  
Computer...  
Add...

Graduation  
Goal:  Full-time  
Constraints  
Max hours/week  
  
Group projects:  
 Yes  No  
...

Cancel Save

Drag & dropable

## Course index / search page:



## Course detail page:

