Kyle Somers – API Implementation, Database Implementation

Shreyas Tendulkar - API Implementation, Search Implementation

Tirtha Chavan – Application Developer

Ting-Ju Yang – Application Developer

Yu Zhao – Application Developer
INTRODUCTION

Traditionally, universities have implemented large career services platforms with portals for employers to set up profiles and post details about their offerings for students. This has traditionally served as a de facto resource for finding out which companies the university has connections with. While that might work for firms that are familiar with your school, what about startups - they don’t have the resources to scope every university in America and connect with them? They lack the funds, the staff, and the time to hit all the regions in which universities reside. And let’s not forget about the schools that aren’t Harvard or Yale or MIT? Startups in Boston have no business reaching out to UCLA when they can select young fresh talent right in their own backyard. This leaves many corporate connections untapped and the schools at a big disadvantage since the world of startups are unable, or unwilling to devote resources to connect with them.

At StartupEDU, we’re creating a network where school staff, faculty, and career services officials can find corporate connections by tapping into the world of startups, a much underappreciated market in the career services community. When users have connected with the right startup and made connections, they can share their engagement history with the campus to create a single repository for others to find startups meeting their needs. This makes the task of planning career fairs, internship acquisition, campus business connections, and class experiential learning projects as simple as it can be all while tapping into a newer market - startups.

**Ok, so why Startups?** In the United States alone, the startup scene is bustling, with Angel List, considered the “holy grail” for information on startups, reporting over 40,000 new ventures on their site\(^1\). Startups are young, they are passionate and most important of all, and they are the easiest to gain access to. Our company was created after we realized how difficult it was for faculty advisors in our Masters Consulting course to find viable clients for future experiential learning projects. The startup community offers far easier access to internships, projects, and recruiting than the slow tedious process of connecting with a Fortune 500 firm can provide. While StartupEDU does not suggest that campuses alienate larger firms or disassociate with them, we believe that smaller and newer companies can provide all the same benefits, are currently not being addressed.

**Our Objective**

- Create a Platform for university officials (career coordinators, faculty, and staff) to find startups
- Find new corporate partnership opportunities in any market (ie. Healthcare, Banking, E-Commerce)
- Share your existing connections with the campus. Leverage existing campus connections for your gain
- Make it simpler for universities to establish experiential learning, internships, investments and projects

The goal of StartupEDU is to develop a web hosted platform for any university in America to join and share their connections amongst each other. This creates a social platform of sorts for users to continuously grow their network with the help of referrals that make the most sense...from fellow university colleagues. By sourcing the company data directly from third parties without requiring companies to create profiles on our site, **we eliminate the geographical burden that negatively affects the career services platforms used on campus.**

---

\(^1\) Angel List is a social networking site for the startup community. The data on Angel List is entirely user generated via the creation of profiles. StartupEDU sources the majority of its company data from Angel List. To learn more about Angel List, visit [www.angel.co](http://www.angel.co)
THE STARTUPEDU PLATFORM

StartupEDU is a web platform providing end user access to a vast repository of startup company information and user generated connection content. Our application serves data from a handful of sources and provides users with a single location to search for startups, find prior connections from colleagues, and reach out to ventures, all within a single environment. The following section outlines our features and functionalities as they currently exist in the platform.

Features & Functionalities

“One and Done” Search Functionality

The largest component to our application is our search functionality, providing a single unified bar to do searching on company information by company name, location(s), their tagline description, and the market(s) they operate in. While our system stores far more detail than this, these are the most important facets by which a user would want to navigate and find startup companies.

This search functionality allows the user to type in any word or phrase that they want and it will return results meeting their criteria in order of results relevance, which will explained in further detail in the analytics section. Since our database system treats each company and their respective information as a single aggregated document, any words or phrases being stored in the StartupEDU company database will be used to help in the returning of the results. This makes the search experience very seamless for our users when they don’t want to spend time selecting checkbox fields and various filters to narrow results. Our search functionality is quite accurate despite only using one single search bar. Simply search what you want, and our application will handle the returning and ranking of results. In the example shown above, Kyle is automatically shown results in Tucson by default because he works at the University of Arizona. Depending on the user logged in, the default results displayed will be different. If Kyle is interested in finding projects for students in the field of big data with companies in San Francisco, he can search those terms accordingly, as shown to the right.
Company Profiles

Another major component to our system functionality is the company profiles that we have created for all the startups collected by StartupEDU. When a user finds a suitable venture to connect with, they can select the “Learn More” option on the search page and view the StartupEDU profile on that company. This information is being collected from several sources, both real-time and from our stored database records.

Our company profile pages provide the following information and functionality on the startup, if it exists:

1. Company Email and URL
2. Description of the company
3. Vimeo or YouTube video for the company (our site can handle either type!)
4. Contact information for the employees, founders, and board members at the company to connect
   ✓ This includes LinkedIn, Facebook, and Twitter accounts
5. Distance on map between the user’s university and the startup location
6. Ability to “Shout Out” to StartupEDU and your university, any prior engagements with the company
7. Ability to send emails to anyone, including the company, directly from the profile page
8. See who else is connected to the startup and what university they are affiliated with.
Shout Out Your Connections to the Campus

If Kyle has a prior engagement with “Flicksea”, he can select the “Shout out” button and let his university know about the connection. An additional feature is the ability to list any collaborators or “Affiliates” that were also a part of this connection. These can be users that exist within the StartupEDU community or people that do not have accounts in the system. This allows the referral network to grow as a single connection to a Startup could have several folks associated!

Email the Company…and others…in the App!

StartupEDU can send emails on your behalf, allowing you to connect directly with the startup and with colleagues conveniently in the application. Don’t worry, when you sign up we grab your email address so any recipient will know that the email came from you, even if it sends from the StartupEDU SMTP server!
Personalized User Dashboard

When a user first logs into the StartupEDU site, they will be directed to a personalized dashboard displaying their connection history and a breakdown of all StartupEDU connections at their university grouped by the department on campus. Currently, our dashboard displays all connections across all the schools since there are very few users in our system. Once more users join the platform, we can narrow this to only show the breakdown at the university of the logged-in user. This dashboard allows the user to see all of their past connections while learning about the distribution of the connections across their campus.

Kyle works at University of Arizona so his dashboard acknowledges this!

Tirtha works at Stanford so her dashboard acknowledges this!
System Architecture/ Components

StartupEDU was developed on a database architecture consisting of both MySQL and Amazon’s NoSQL database service, DynamoDB. Our system is spread across two Windows EC2 servers on Amazon, one serving as webserver and user database, and the other running as our crawling server. Our crawling server runs the Python collection scripts that gather our thousands of records on companies from Angel List and also contains the scripts that save our data to DynamoDB via the DynamoDB Python API.

DynamoDB NoSQL Service

Our DynamoDB database is a cloud service offered by Amazon that acts as a non-relational environment for the storage of our crawled data. We implemented a NoSQL environment for several reasons, the first being the fast scalability it offered as we continued to find new data points that we wanted to store. By ditching the traditional relational environment, we are able to continuously add new
columns of any type on the fly without the constraints of a predefined schema. This gives us significant flexibility as we grow and want to add more data sources to our environment such as from Socrata. This environment stores our records via a hash key (primary) and a range key (our facet). We have built many indexes on the original base tables that are used to rapidly serve requests that hit the environment. These indexes were specifically designed to execute the queries which we planned for our site to handle. DynamoDB is a very flexible option for us because we pay for what we use, nothing more. We are charged based on the storage of our indexes and the read/write throughput our application uses. Since our users do not write to our NoSQL environment, only write to it, we have very minimal write costs which are more expensive than the read costs. Our small team of 5 can focus on the business, spending very little time managing our environment. Likewise, the service offers us the ability to adjust the read/write throughput at any time we please, ensuring our service remains flawless when more universities join our platform.

**Amazon CloudSearch**

For our search functionality, we connected our NoSQL environment to Amazon’s CloudSearch, which can natively understand our DynamoDB data with minimal customizations. This service takes each of our records and indexes them into document sets, allowing the entire tuple (document) to be searched on with keyword and phrase searches. This environment is in a live sync with our DynamoDB environment, meaning our end users can search our new data as soon as it is indexed into DynamoDB from our crawling server. Even more beneficial, since our CloudSearch instance and our DynamoDB instance both reside in the same US Region (datacenter), we are not charged for data transfer rates between the services, keeping the investment low in these services. With CloudSearch, we are charged for the per-hour consumption of our indexed data. This is scalable for the company as the majority of usage of our platform will come between the working hours of university officials, or roughly 8 hours of the day. We also pay when we do new document (data) uploads which fortunately do not occur real time. Since company information does not tend to change by the second, we have our crawling server performing crawls and Database writes on a nightly basis. For this project we performed this manually from the server. However, in the future we could set these on scheduled cron jobs and automate our entire process.

**Our System Architecture**
Our Novelty – The StartupEDU API

StartupEDU offers an API for our “Summa Cum Laude” universities to access our data programmatically. When they visit the API Access tab on our site, they will be directed to the URL endpoint displayed below. This endpoint illustrates a JSON readable format of our indexed DynamoDB data. Conveniently, this is the same endpoint that our search functionality hits when we pass through queries for the front end search requests. By creating a similar second endpoint, we now have the ability to offer our users access to our data for their development and application needs!

The Endpoint for our StartupEDU API...notice it is a CloudSearch URL

APIs Used to Create StartupEDU

StartupEDU is a platform specifically for establishing connections and sharing these connections among the different parties at a university. We utilized several different APIs to obtain the functionality for our site.

<table>
<thead>
<tr>
<th>Web</th>
<th>Database &amp; Search</th>
<th>Data Handling</th>
</tr>
</thead>
<tbody>
<tr>
<td>Google Map API</td>
<td>DynamoDB API</td>
<td>Angellist Startup roles API</td>
</tr>
<tr>
<td>Youtube API</td>
<td></td>
<td>Angellist Tags API</td>
</tr>
<tr>
<td>Twitter Fellow API</td>
<td>Amazon CloudSearch API</td>
<td>Angellist Search API</td>
</tr>
<tr>
<td>Vimeo API</td>
<td></td>
<td>Crunchbase API</td>
</tr>
</tbody>
</table>

Analytics

For our analytics component of the platform, we incorporate fuzzy search and word stemming for our search functionality. To perform this, we utilized the components available via the Amazon CloudSearch service to harness the power of Amazon’s textual analysis. After our DynamoDB data had been successfully synced with the CloudSearch service, the setup of this textual analysis was very seamless. To implement this, all we had to do was turn on the CloudSearch stemming feature and insert into our query endpoint the logic to perform fuzzy search to 2 letters. CloudSearch provides the ability to do fuzzy search on over 3 letters, but we limited this to two letters by declaring this in our search endpoint URL as shown below –
Fuzzy Search

~2 tells CloudSearch to perform fuzzy searching up to two letters

We chose two letters because anything more would have been counterproductive and started serving results that did not make sense for our users.

With two letter fuzzy searching, when a user searches for “tuccon”, intending to search for “Tucson”, they are still served the results for Tucson!

In this result set of eight companies, the final one returned is “Tycoon Real Estate”. This is served because the CloudSearch fuzzy functionality deemed “tycoon” to be close to what the user was searching for, “tuccon”.

Stemming

We also implemented Stemming, a functionality that was easily activated via the Amazon CloudSearch interface, providing stemming functionality for our application. This allows searches such as “network” to return results for occurrences of words like “networking” in our database. We activated the “light” setting for stemming simply to provide the basic functionality of the feature. We did not introduce a higher level of stemming because this would have negatively impacted searches on company markets by serving unwanted results.
Implicit vs. Explicit Searching, Plus Document Scoring

When a user types in entries into our search bar, the search is simply checking the occurrences of keywords and displaying the results based on the frequency of those parameters in the data that it checked against. This could end up in unwanted results if a user makes a search such as the one to the right. Since the word “advertising”, a market, tends to appear more in a document than a city name, the results could display the market you wanted but not necessarily the city. This would be an implicit search in our system.

To more accurately return results, our application also allows explicit searching, by placing quotes “” around any word that the user declares must exist in the search results. This will guarantee that the city “Denver” exists in the results returned to Tirtha, as displayed below.

By implementing the CloudSearch functionality into our application, we were able to provide enterprise quality search and benefit from the many additional functionalities that the service had to offer without coding this ourselves.
Our Revenue Model

StartupEDU is a paid hosted web service that is offered to universities on a subscription basis. The revenue for our company comes directly from the service fees that we charge universities based on the package that they select. We charge based on the features which are made accessible to the paying school as well as the number of users that the university will support.

Our packages are illustrated below:

1. **Cum Laude**
   - Analytics Dashboard
   - Full Search Access
   - Limited # of Users

2. **Magna Cum Laude**
   - Analytics Dashboard
   - Full Search Access
   - Increased # of Users

3. **Summa Cum Laude**
   - Analytics Dashboard
   - Full Search Access
   - Maximum # of Users
   - API Access
   - Single Sign On Integration

The first two packages we offer to our customers are entirely based on the number of users that will use the system at the university. We plan to assign the # of users into predetermined buckets with the university paying for the package representing the bucket in which their # of users falls into. These two packages, the “Cum Laude” and “Magna Cum Laude” offerings, provide the user the full functionality of StartupEDU. This includes the company searching, the customized dashboard, and access to the company profiles with associated contact information. For our third package, the “Summa Cum Laude” package, the university will have access to the maximum number of users. At this time we have not decided if this will be capped at a certain number or if we will allow unlimited access. Our team hopes to determine this after a small GO-LIVE to select schools and see the system usage by these schools. This third package will also give the school unlimited access to our StartupEDU API, which can be integrated with any preexisting systems or applications to give machine readable format to our customers. Finally, this package also comes with a customized integration with the university’s single sign on identity access platform to provide a simpler logon process for the school. Instead of logging in with a StartupEDU user account, the user will be able to select their university and login via their school’s identity management portal.

Since a product offering similar to ours is not currently available in the market, we are unable to benchmark prices and packages against any set standards. Therefore, at this time, we have not been able to successfully pinpoint accurate prices to charge our customers or find specific cutoffs for the number of users that will be associated with each specific package. Our team is hoping that with investment from an experienced Angel, we can be provided the necessary industry knowledge on how to competitively price our product while still supporting a realistic model. While the foundation has been laid for our revenue model, we did not want to arbitrarily pick price offerings and risk over assessing the value of our product.
Competitors...Kind of

StartupEDU does not currently have a true competitor as no current website targets our market – university officials. However, many sites offer a lot of unstructured information on startups, focusing on financial background, job postings, events, and news related to new venture funding. Despite not having a true competitor, we believe that career services software companies will consider our business an infringement on their space, therefore provided healthy competition for StartupEDU.

Although our company does not currently have direct competitors, we rely on several other businesses as the source of our crawled company data. The two major sources are Crunchbase, a subsidiary business of TechCrunch, and the social networking website Angel List. Fortunately, these platforms are free and offer their data via machine readable APIs. While these platforms serve much of the same startup data that we provide our users, they do not target the university community and therefore are not considered our competition. They are unable to provide a personalized connection network specific to the university community. We believe that our product serves a much different purpose than these sites. While they provide lots of information focusing on funding and employee history, our competitive advantage is built on the idea that a university faculty member wants to spend their time connecting and less time reading about the financials of a company. Additionally, this same faculty member is not likely interested in using his daughter’s best friend’s father’s brother as a referral to connect with a hot startup (as would happen on Angel List). He would much rather create that connection via a colleague at his university whom already has a preexisting university relationship with the venture. This makes the process that much easier on the startup as well - if they already have experience working with the university community!

**Let’s talk incentives!** At StartupEDU, we create a mutually beneficial environment for the startups and the universities. In fact, we even help drive user traffic to our biggest source of information, Angel List. Since our web platform directly sources information from this social network, which is entirely generated by user-created profiles, we need them to thrive for our platform to be successful. As our network grows and more schools join the service, the startup community has a major incentive to join the Angel List social network and fill up their profile. As they learn that a simple profile creation on Angel List could lead universities all over the country to come knocking on their doors with internship and experiential learning opportunities, they have a big reason to head over to Angel List. This therefore gives us a big reason to collaborate with Angel List and CrunchBase.

**Partnerships with Crunchbase and AngelList**

- Provide due credit to our data sources on our site
- Similar to this

Portions of this content provided by:
AngelList - Where startups meet investors

(From CrunchBase.com)
Target Users

StartupEDU’s target users are people that work at universities! More specifically, we target faculty, career services coordinators, professors...anyone with a stake in helping classes acquire real projects, students acquire real jobs, and the university acquire real partnerships.

FUTURE PLANS

Improved Search

StartupEDU has many future plans we’re looking to implement. First of all, we want to grow our searching ability by taking our MySQL users and connection data to introduce it into the NoSQL environment that houses our company data. By doing so, we be able to provide users with the ability to search companies by connections and people, rather than just details about the company itself. We’d also like to introduce the Amazon CloudSearch faceted search functionality as an add-on to our current “single search bar” functionality. By doing so, we can provide users with an additional method of searching our records. With this increased search and faceting ability, we’d be able to capitalize on the CloudSearch weighting mechanisms that come built in with the service, giving our users the ability to manipulate the results rankings based on their predefined criteria and interests. Believe it or not most of this is feasible with the built in functionality of CloudSearch and simply involves developing new search API endpoints for our queries to hit.

Enhanced Connections

Our next plans include modifying the way we categorize connections so that they can be used to display better results in the dashboard. Currently, our dashboard is displaying information in the aggregate, but our future plans include drill down functionality and viewing of actual users connected at your school rather than the distributions by department. We’re also looking to make better use of the interactive chart we have already implemented to allow the user more analysis within the dashboard.

Prep our API for Production

We were very successful in getting an API built for machine access to our company information. While the API is fully functioning, it does not currently have any user access controls or throttling as would be expected from a production API. Our future plans include laying an API control service, perhaps from Windows Azure or Amazon AWS, on top of the StartupEDU API to monitor and control its usage. Additionally, once we introduce connections information into our DynamoDB NoSQL environment, we’ll need to split this API into different endpoints to distribute the load of our data hungry university users!

Introduce Socrata City Open Data

Finally, our last goal is to begin introducing new company datasets such as Socrata’s Open Data access. This startup, based in Seattle, has already successfully made thousands of endpoints accessible for the public and we’re hoping that as the Open Data movement expands, so will the number of reliable and structured datasets on startups throughout the country.
**Kyle Somers – Database Development & Company Data APIs**
Kyle contributed to the team through database development and the implementation of the Angel List data collection APIs (Angel List) with Python. Kyle is the mastermind behind our DynamoDB NoSQL environment that supports StartupEDU’s company data storage. He also built the MySQL environment that hosts our user authentication and connections data.

**Shreyas Tendulkar – Search & Company Data APIs**
Shreyas is our search guru. He is responsible for deploying our search functionality via Amazon CloudSearch. He implemented our dynamic searching and worked with Kyle to lead the PHP on the front end. He is also responsible for our implementation of the CrunchBase API for getting additional company information.

**Ting-Ju Yang – Front End Developer & Social APIs**
Ting-Ju (AKA Anny) led our front end development with Yu. She is a web design expert responsible for much of the design that our users boast about! She led the development of our company search page. She is also responsible for getting our social interaction APIs functioning.

**Yu Zhao – Front End Developer & Social APIs**
Yu led the front end development with Anny. She is responsible for our company profile page along with much of the design our site. She became a bootstrap expert in the process of building this site, helping providing our users with the great site they see today.

**Tirtha Chavan – Front End App Developer / Video APIs**
Tirtha worked as a front end developer leading the implementation of our video APIs and the initial whiteboarding of our site. She has a knack for “what the user will want” and was responsible for much of the testing that went into the confirmation of our site functionality before our Go-Live.