



H A R V A R D | B U S I N E S S | S C H O O L

9-815-093

REV: OCTOBER 14, 2015

JEFFREY J. BUSSGANG

LISA C. MAZZANTI

Codecademy: Monetizing a Movement?

We're a movement to make education more of a commodity. We're not just a for-profit company. Our mission would get tainted if we charged consumers for content. We need to be authentic.

— Zach Sims, Cofounder and CEO

Zach Sims and Ryan Bubinski sat in the Codecademy headquarters, an exposed-brick fourth-floor office near Madison Square Park in New York City. In 2011, while in their early twenties, the two had founded Codecademy, an open-platform, online community to teach users to code. By 2014, they had a total of 24 million unique users and a library of over 100,000 lessons. The company had raised a total of \$12.5 million in funding and was, on many fronts, an overwhelming success. However, there were still no revenues. The company's website stated, "Codecademy is free and always will be."¹

The founders, along with the board, had decided that 2014 would be a year of experimentation with different monetization strategies. By June, the cofounders had preliminarily tested two monetization models. The first charged companies for training employees offline on coding skills, a service that the training departments of these companies paid an annual fee to receive. The second monetization model focused on a labor marketplace to match Codecademy users with jobs that corporations and recruiters were seeking to fill.

But 2014 had also been busy in other arenas for the 25-employee company. In April, the company launched a redesign of its website, because, as the Codecademy blog announced, "it quickly became apparent that if we wanted to grow and mature as a brand, we required a thorough redesign of our entire product."² The next month, the company announced that they were opening an office in London to work with the British education system and also had forged partnerships with foundations and government bodies in Estonia, Argentina, and France.

As Sims and Bubinski huddled in their glass-walled conference room, they tried to focus on the task at hand—to narrow down their ideas and eventually decide on a viable business model. The two reviewed early results from both experiments to prepare for the upcoming board meeting where they planned to present their findings and propose next steps. The employee-training experiments had yielded promising initial results but would require hiring a sales force, offline instructors, and some content customization to scale. The labor marketplace model promised less friction in scaling but represented a more crowded market opportunity.

Senior Lecturer Jeffrey J. Bussgang and Case Researcher Lisa C. Mazzanti (Case Research & Writing Group) prepared this case. It was reviewed and approved before publication by a company designate. Funding for the development of this case was provided by Harvard Business School and not by the company. Jeffrey Bussgang is a general partner at Flybridge Capital, an investor in Codecademy. HBS cases are developed solely as the basis for class discussion. Cases are not intended to serve as endorsements, sources of primary data, or illustrations of effective or ineffective management.

Copyright © 2015 President and Fellows of Harvard College. To order copies or request permission to reproduce materials, call 1-800-545-7685, write Harvard Business School Publishing, Boston, MA 02163, or go to www.hbsp.harvard.edu. This publication may not be digitized, photocopied, or otherwise reproduced, posted, or transmitted, without the permission of Harvard Business School.

815-093

Sims and Bubinski were also aware that many other companies, in the United States and abroad, were gaining ground in the computer-coding education industry, including free massive open online courses (MOOCs), websites that charged for content, and offline coding boot camps. "We don't want to get beaten because of someone else's defensive moats," said Bubinski. One Codecademy investor noted, "Codecademy has played a leading role in the development of an online industry to teach people to code, and a lot of companies have cashed in on that industry. At some point, Codecademy has to capture what it created."

The cofounders wanted to avoid being pressured into decisions that went against their open platform philosophy, but they also knew that to keep the company alive and well, they had to eventually position themselves to generate revenue. But Sims had begun to question whether now was the right time to even focus on monetization: "We can monetize by helping people make use of the skills they learn on our platform. But, until we have enough content and skilled users, there is no reason for users to keep coming back or anyone to want to hire them." The company had plenty of cash relative to its modest burn rate, so fundraising was not an immediate concern. Perhaps both monetization experiments needed to be shelved in favor of a greater focus on scaling content and engagement in the community.

The Big Idea: Teach the World to Code

Sims and Bubinski became friends while studying at Columbia University when Sims was a freshman and Bubinski was a sophomore. While in school, Bubinski founded a student organization in 2009 that helped students learn to code outside of class called the Application Development Initiative (ADI).³ Sims also showed an early interest in technology, and during school he interned at GroupMe (later acquired by Skype), drop.io (acquired by Facebook), and AOL Ventures. By 2011, the two had begun to identify a set of issues that led to starting a company. "We saw that, as our friends began looking for jobs, there was a disconnect between college education and marketable skills," said Sims. Initially, they envisioned a company that would help recent college graduates discover and match their skills with job opportunities. In the summer of 2011, after Bubinski graduated with a degree in computer science (CS) and Sims had completed his junior year, they applied and got accepted to Y Combinator, a three-month entrepreneurship and seed-funding program in Mountain View, California. When Sims learned of their acceptance, he told his parents he was going to drop out of college in order to focus on the startup.

But while at Y Combinator, Sims struggled to learn how to code to build the job website. He found that the educational books and videos available were insufficient, much like Bubinski had felt with programming classes (which had caused him to start ADI). The two decided they could build a service unaffiliated with a university, yet valuable to users, to address the gap they saw in coding-skills education.

The cofounders pivoted their idea to one of offering a service to teach the skills people needed to find jobs. They started designing a free product that would help coders, initially beginners, learn coding through interactive exercises. "We wanted to build something that I would like learning from, and that Ryan would like teaching through," said Sims.⁴ The lessons taught concepts by action—none included videos or books. The exercises were set up in an in-browser terminal, and the user was given simple directions and "chipper words of encouragement"⁵ in order to execute a command.⁶ (See Exhibit 1 for an example of the interactive lessons.) As users finished lessons, they accumulated points and badges. The website tracked student progress, which users could share with friends on Twitter and Facebook. Initially, website visitors could take the first lesson without signing up. After the user completed several lessons, a prompt appeared to create an account.⁷

Sims and Bubinski launched Codecademy on a Friday afternoon in August 2011 with eight lessons on JavaScript, a coding language used to create interactive applications ("apps"). Within more than 1,000 concurrent users.⁸ Within the first four days, there were 250,000 visitors who spent approximately one hour on the site and had done over 2.75 million exercises. Sims stated at the time, "This is the first time programming has been truly democratized. It's available to anyone with a computer and it's easy to follow. We think creating a new generation of programmers will help to raise employment and the standard of living for those individuals. It's preparing the world for the future."⁹ Sims and Bubinski now knew their venture could have a large impact, as Sims described: "We're closing the gap between education and labor to make sure anyone, anywhere has access to a free education that can help them make a living." Although Sims estimated that the age range of the users was predominantly 13–30 years old, he felt that the site could have a wider target audience of 12–50 years old.¹⁰

The market's reaction to the simple website attracted investors. Shortly after completing the Y Combinator program, the company closed a \$2.5 million round of funding led by Union Square Ventures (USV) along with AlphaTech, CrunchFund, Founder Collective, SV Angel, Yuri Milner, and others.¹¹ "What really intrigued us about Codecademy was the engagement it fostered with users — this isn't just a teacher lecturing. It's learning by doing," said Andy Weissman, a partner at USV. He continued, "It's designed to have millions of people use it. It's transformative because it unbundles coding from college education. And lastly," Weissman added, "We just really like these guys." USV was not concerned with the specifics of a monetization strategy right away. "Sometimes, if you decide on and pursue a monetization strategy too soon, you can get in your own way. We believe you build a barrier to entry first with a large network and then monetize. You want to retain flexibility to make sure your strategy fits with the fabric of the product," said Weissman.

By November 2011, Codecademy had more than 800,000 users.¹² Sims said of the site's success, "We started working on Codecademy simply as a product and it quickly became a company. We threw the original site together in about three weeks."¹³

Code Year: Scaling Content and Community

The buzz that ensued from the announcement of the series A funding faded quickly. By the end of 2011, Sims recalled, "There was a plunge in usage, because we barely had any content." Acting quickly, the company launched an initiative called Code Year on January 1, 2012, to take advantage of New Year's resolutions. Code Year offered a simple e-mail service that would notify users of at least one new lesson a week. Sims sought the pressure the weekly service would provide in forcing the company to create enough content to fulfill the weekly demand. Within 48 hours of the launch, 100,000 users had registered for Code Year. "People got excited about the weekly cadence," said Sims. New York Mayor Michael Bloomberg tweeted: "My New Year's resolution is to learn to code with Codecademy in 2012!"¹⁴

The founders focused on creating a viral-friendly design of the site to fuel rapid growth—users could "tweet" and "like" the Code Year page on social media even before they joined. "We built in a super tight viral loop," said Sims. Twitter drove one-third of the traffic and Facebook over 16%.¹⁵

Two weeks later, Codecademy and the White House announced a joint initiative called Code Summer+ to educate youth as part of the White House's larger initiative, Summer Jobs+.¹⁶ Sims explained that Code Summer+ provided a "condensed version of our curriculum to get [youth] on track to become engineers."¹⁷ By the Code Summer+ launch, Codecademy had 1 million users.

815-093

Simultaneously to the Code Summer+ announcement, Codecademy announced that they had launched a question-and-answer forum for users to communicate about lessons.¹⁸ They also designed a part of their website for students to organize in-person meet-ups to discuss their programming progress and roadblocks face-to-face. Within a week, 171 Codecademy meet-up groups formed worldwide.¹⁹ A tech blogger at *Gigaom* wrote, "Codecademy is on fire right now."

New Functionality

Although Codecademy had achieved early market acceptance, it was soon criticized for oversimplifying coding complexities (see Exhibit 2 for a discussion thread's critique). Sims explained, "We wanted to focus on the essence of programming, the terminal.^a We wanted to bring the magic of programming back to people and to take away everything else, like the process of choosing a text editor, a language to learn, or a place to store your code."²⁰ Sims felt that the company had removed many of the barriers to coding. "We made this so simple for users. But coding is hard. I think some people thought, 'Oh, I'll just do this for a few weeks and then be able to get a job at Google,'" noted Sims.

The shortage of lesson content continued to be a point of friction for the company. By the end of 2011, the cofounders were managing a dozen lessons. Users complained about only having tutorials in JavaScript and a lack of lessons in general. In January 2012, *TechCrunch* reported, "Codecademy has had one big problem: there just aren't that many lessons available. And the ones that are on there sometimes seem to be moving too quickly, without many practice exercises to explore and reinforce what you've just learned."²¹

In an attempt to radically increase the number of courses available, Sims and Bubinski decided to launch a beta test for peer-to-peer course creation to help fill the instruction gaps. Coders had already been e-mailing the company offering their know-how, so the cofounders wanted to formalize the community's involvement. The response to the beta test was positive—2,000 users signed up to design lessons across a range of subject areas and convinced the founders to go ahead with a new peer-to-peer course-creation platform.²² On January 30, Codecademy launched Course Creator, allowing anyone to create and post lessons, and provided a "Creation Guide," which outlined the process, tools, and tips for lesson development.²³ The company incorporated a submission test function into Course Creator and posted steps for testing submission correctness for quality-control purposes.

Course Creator could be used to create a course of general interest or for a user's own specific purposes—whether it was a teacher using the Codecademy platform to teach their own class or a professional demonstrating a company's proprietary tool. Publishing a lesson required no approval. Once someone created a lesson, they could send out the link to the lesson to anyone. However, Codecademy curated the lessons for users by featuring the high-quality ones on its homepage.²⁴ Sims estimated that creating a course took 10–20 hours, and, though contributors were unpaid, he believed prominently featuring lessons incentivized programmers to post them.

A Marketplace Model

Course Creator moved the company toward a blended model for content creation. Approximately one-third of Codecademy lessons were created by third parties, and usage of third-party lessons tended to be lower than lessons created in-house. "We make sure we have content against curricula

^a A terminal was the on-screen display where code could be entered.

or topics that we think interest our students by working with content-creation partners and authors from the general public. Through that curation, we try to create a cohesive learning experience," said Bubinski. He explained that a pure marketplace experience around content is not effective for beginners: "We realized you have to strike a balance. In an ideal world, every student gets the perfect content for the skill that's most important for them to be learning at that time. What's tricky is that we don't always know what the learner should be learning, and the learner doesn't know either. There's a huge discoverability problem."

Though a marketplace suggested that a customer could sift through myriad options to find the best fit based on value and quality, Bubinski felt that, "for education, people pick by brand or some notion of quality of the institution." But new students were not the best judges of education quality. He explained, "If you want to start studying electrical engineering, you're not really qualified to say what a great curriculum is for that, since you haven't studied it yet. You have to be able to trust the brand. The marketplace dynamics for education aren't quite as cut-and-dry as products on eBay yet."

Expanding and Refining

As the company expanded its course library, it was able to expand its coding language offering. In April 2012, the company unveiled tutorials for CSS and HTML,²⁵ two coding languages generally used in tandem to build websites. The same month, Codecademy launched an interactive coding console, Labs, an in-browser feature that allowed users to practice what they had already learned without having to download an integrated development environment (IDE) or code editor.²⁶ (See Exhibit 3 for a screenshot of Labs.) The new setup placed the text editor on the left side of the browser and a console to the right to show what a user's executed code would look like. Sims noted, "Labs makes it really easy to continue along the path of learning stuff without any of the complications that go along with programming."²⁷

In June 2012, Codecademy raised \$10 million, led by Index Ventures, bringing total funding to \$12.5 million.²⁸ The founders planned to use the new funds to grow their base of nine employees and scale globally. Already the company was working to launch sites in Chinese, German, Japanese, Russian, and Spanish. Once again, the company reached out to its network: the translations into other languages were being done by Codecademy users for free. Sims described the users' desire to lend a hand: "The day we launched, someone downloaded the site and translated it into Japanese and then sent it back to us. Unfortunately we had no way of hosting a Japanese site then."²⁹

User Community Dynamics

Codecademy's user community was diverse, with both skilled and unskilled learners. Bubinski explained, "We don't see Codecademy as a product that has fixed teachers and fixed students. Every student can be a teacher and vice versa. It's more of a dorm-room study-group mentality than having professors who stand on a stage." Instead of featuring industry gurus as teachers on its website, the company depended on its community network of experienced coders to create courses. "Advanced students who create courses for us either become evangelists or use Codecademy as a tool for distribution," said Bubinski. He explained that advanced programmers wanted to create courses to spread their knowledge. "Giveback is huge," Bubinski remarked.

The community also gave back in other ways. Community moderators (generally users who had over 500 Q&A forum posts and were invited to be moderators by Codecademy) were in close contact with the company. Their help was multi-faceted: moderators gave important feedback on frustrations or successes of product launches, made edits to courses, and were also given guidelines for what posts to delete from the site and in what direction they should encourage conversations to go. Sims

815-093

said, "To us and our moderators, a community is more than just responding to social media; you have to have a certain ethos of making the world a better place."

Moderators were unpaid, and as one put it, "There are no benefits to becoming a moderator aside from the satisfaction gained from the effort made while helping others. This may be hard for some people to comprehend; however, this is a trait all moderators have in common."³⁰ Approximately 100 moderators spent three hours a day on the site to service the community.

To keep a pulse on community reactions and news, all Codecademy staff was copied on the moderators' e-mails. Originally, Codecademy had a community manager to engage with the moderators and the larger user base. "But when our community manager left, we wanted to see if we could spread out the responsibility and also build community into the product," Sims recalled. "Instead of a squadron of community managers, could we build amazing features that would be mostly intuitive in regard to how users should act and what the norms should be?" However, without a community manager, the moderators had felt that Codecademy was being unresponsive. "Now we're back to hiring a community manager," explained Sims. "We're trying to expand the definition of community. We started with Q&A forums, and now we also have groups, which allow users to be grouped into subsets based on languages, interests, and skills. We don't just want islands of learners off to the side. We're trying to figure out how to build a more inclusive and intuitive learning community."

Demand for Tech Talent

By 2012, demand for computer engineers to build and maintain website interfaces, mobile apps, and other web services continued to grow. The Bureau of Labor Statistics estimated that in 2020, there would be 1.4 million new CS jobs, but only 400,000 CS students.³¹ Software developer employment was projected to grow by 22% from 2012 to 2022.³² Typical full-time software developers in technology hubs such as Boston, New York City, and Silicon Valley were paid \$100,000–\$150,000 per year. Because of fierce competition for talent, some major employers were offering college juniors six-figure starting salaries if they committed to joining the employer after graduation. Technology recruiters were often employed to assist with hiring and were paid as much as 20%–30% of the salary of successful hires.

Companies looking to address their coding needs with freelancers, on the other hand, looked to websites like Elance-oDesk,^b guru.com, and others. Employers posted descriptions of coding tasks they needed, and freelancers from around the world would sift through posts and bid on work that corresponded to their skill sets. Employers who hired freelance coders typically paid roughly \$80 per hour, but rates could be much lower depending on factors such as the particular skills required, the cost of living for the freelancer (many lived outside the U.S.), and the brand name of the company searching for freelance talent. Both oDesk and Elance had tens of thousands of posts at a time for web and software development, while guru.com had roughly 1 million at a time.³³

To address the growing need for coders, governments around the world were evaluating their CS curricula and making changes to public education programs. The *New York Times* reported, "From Singapore to Tallinn, governments, educators and advocates from the tech industry argue that it has become crucial to hold at least a basic understanding of how the devices that play such a large role in modern life actually work."³⁴ In 2012, the United Kingdom's education secretary announced a bold

^b Elance and oDesk merged in 2013 but maintained separate identities.

initiative focused on coding by "scrapping the existing [information and communications technology] curriculum. In its place, [the U.K.] will introduce new courses of study in Computer Science. . . . [to] equip pupils with the skills employers want."³⁵ The same year, Estonia announced that its public schools would teach students to write code from first grade on.³⁶

In the U.S., the dissemination of coding education was being spearheaded by private companies and individuals. Tech investor Hadi Partovi started the nonprofit organization Code.org to help bring CS education to K-12 schools in the U.S. and advocated for CS as a core academic subject.³⁷ Large tech companies such as Amazon, Apple, Google, and Microsoft, as well as President Obama, threw their support behind Code.org's campaign "Hour of Code," where "33,000 classrooms in 167 countries, 2012, Codecademy launched its first mobile app for Hour of Code, a "minimum viable product" to experiment with teaching coding on a mobile phone in a one-hour module. Sims explained, "The app was designed fresh . . . to take advantage of the iPhone while still preserving all the design tenets from our web app. . . . It gives anyone all they need to get started anywhere at any time with the basics of programming."³⁹ (See Exhibit 4 for a screenshot of the mobile app.)

Online Education: A Movement

When first founded, Codecademy was a pioneer in offering free coding-specific online education. However, the market opportunity to help close the skills gap was not lost on other companies. Thus, by 2014, competitors had emerged from many segments of the online education market.

University-Affiliated Online Education

By 2013, MOOCs were attracting much attention by expanding access to education and prompting educators to revisit how people learned.⁴⁰ While the MOOCs offered a broad catalog of courses outside of CS, their CS subject matter overlapped with Codecademy's. Three major MOOCs (EdX, Coursera, and Udacity) offered free classes to anyone with an Internet connection.

EdX A nonprofit partnership between Harvard University and the Massachusetts Institute of Technology (MIT), EdX was created in 2012 and expanded to a consortium of schools that contributed courses, such as the University of Texas and the University of California, Berkeley. EdX featured over 175 free courses that ranged from biology to literature; approximately 40 of the courses were CS. Over 400 faculty contributed. Students could earn certificates for completing courses, which required roughly 50–100 hours of work. One EdX principle was financial sustainability, although revenue generation was still being worked out. Harvard and MIT had each made an initial investment of \$30 million.⁴¹ Potential monetization models included third parties paying to access users or recruiting schemes with employers who would pay for the identification of successful students.⁴²

Coursera Coursera was a for-profit company started in 2012 by two Stanford University CS professors. By 2013, it offered over 200 courses from 33 institutions from around the world and served more than 2 million users.⁴³ Each course called for about 40–60 hours of work and ranged from medicine to the arts; roughly one-third were in CS.⁴⁴ In 2013, Coursera launched career services, which students could opt into for free. Companies paid a flat fee for each introduction to a student. The same year, Coursera introduced the "Signature Track," which provided verification of a student's work and allowed students to receive certifications. Course admission through the Signature Track was limited to certain courses, and students were charged \$30–\$100 per course.⁴⁵

815-093

Udacity Although not officially Stanford-affiliated, Udacity was founded by former Stanford professor and Google employee Sebastian Thrun. A for-profit venture started in 2012, it focused on math and science, including CS. The courses were self-paced and included data science, web development, software engineering, and classes for an online master's program with Georgia Tech. By the end of 2012, Udacity had raised over \$21 million.⁴⁶ By 2013, it was testing various business models: "matching students with employers, licensing content to schools and charging for proctored exams."⁴⁷ In 2013, *Fast Company* reported that Thrun was considering pivoting the company to charge a fee for classes and become more vocationally focused.⁴⁸

Unaffiliated Online Education

Online education companies unaffiliated with higher education institutions (Lynda.com, Khan Academy, and Udemy) were gaining ground, benefiting from the network effects of larger MOOCs.

Lynda.com Lynda.com started in 1995 and was profitable by 1997. It offered what *Wired* called "high-end how-to content . . . including tutorials on the Ruby on Rails programming language, search engine optimization, basic business software like Excel, and digital photography."⁴⁹ The company adopted the contributor model early on: experts were screened and flown to the company's studios to shoot instructional videos.⁵⁰ By 2013, it had a staff of 400, 550 courses, 1,400 hours of content, and 3 million clients (including individual subscribers and enterprise clients from government, corporate, and academic institutions).⁵¹ The Lynda.com business model was freemium-based. After gaining access to a few free lectures, subscribers paid \$25 monthly or \$250 yearly, and Lynda.com in turn paid content creators based on traffic.⁵² In 2013, it raised \$103 million in series A funding to grow internationally and build into more content areas. Cofounder Lynda Weinman said the company would never offer accreditation, because the company simply promoted the joy of learning.⁵³

Khan Academy A nonprofit started in 2004 that initially posted its 10-minute videos on YouTube before having its own website, Kahn Academy was known for tutorials on middle school and high school math concepts. However, by 2014, it also offered tutorials on humanities, computing, and standardized test preparation. The lessons were self-paced, combined videos and exercises, and tracked users' progress. Khan Academy had roughly 6 million users per month and 4,000 lessons.⁵⁴ It had successfully piloted programs with school districts, and in 2013 Mexican multimillionaire Carlos Slim pledged to fund translation of the lessons into Spanish.⁵⁵ Its homepage boasted, "Completely free, forever," and founder Sal Khan had claimed, "I don't have a business model."⁵⁶

Udemy Udemy was founded in 2010 with a marketplace model that "skewed towards the skills workers need to remain employable, rather than the academic course content offered by many MOOCs."⁵⁷ Instructors could create and upload courses to the platform and decide how much to charge for their course, if at all.⁵⁸ By 2014, Udemy had raised \$48 million and had 16,000 courses, and 8,000 instructors had published on its platform.⁵⁹ Udemy's courses were available in 10 languages, and 60% of its students were from abroad.⁶⁰

Online Education for Coding

Online education companies focused on teaching coding had emerged by 2013 with freemium and fee models.

Treehouse Treehouse, which TechCrunch called “essentially the lovechild of Lynda.com and Codecademy”⁶¹ because of its mix of video instruction and project-based learning, launched in 2011^c and was backed by LinkedIn cofounder Reid Hoffman. In its first year, the site generated \$3.6 million in revenues from 19,000 users who had completed 380,000 quizzes and watched 330,000 hours of videos.⁶² The website charged \$25 per month for a basic subscription, which included access to more than 100 lessons, 1,000 videos, a coding console and text-based interface called Code Challenge Engine, and access to forums. The “Pro” subscription (\$49 per month) included videos of talks by industry professionals and workshops. The website advertised that for every new Pro account, the company donated one to a public school student. In 2012, the company committed \$3 million worth of scholarships to give college students two years of its Pro subscription.⁶³

Code School Code School offered 30 courses and nearly 2,000 “coding challenges” to “learn by doing” on a freemium model that charged \$29 per month for access to the full range of courses. Its five paths covered HTML/CSS, iOS, JavaScript, Ruby, and others. It partnered with Google in 2013 to launch an online course, Discover DevTools, to teach the developer tools available in Google’s web browser, Chrome.

Other online resources for those interested in learning to code were HTML5Rocks, a free, open project by Google to give developers a resource for learning about HTML5 coding capabilities with presentations and tutorials, and HTML5 Playground, an in-browser platform to practice coding. Another free online resource was the Mozilla Developer Network, which provided information about open web technologies such as APIs, CSS, and HTML. TheCodePlayer was a free website to watch videos of programmers’ screens while writing code to create certain projects, such as logos.

Coding Boot Camps

Coding boot camps or “hack schools” were an offline option for those eager to learn the skill. These 8- to 16-week programs were in-person and cost thousands of dollars. Hack schools publicized stories of high-earning job offers extended to those who had finished their programs. Dev Bootcamp was a nine-week program with locations in San Francisco, New York, and Chicago that charged \$12,200 and maintained a ratio of 1 instructor per 10 students.⁶⁴ Hackbright Academy, also of San Francisco, was an all-female boot camp that charged \$15,000 for its 10-week program.⁶⁵ Coding Dojo offered a 12-week program that charged just under \$11,000 and operated in Mountain View and Seattle.⁶⁶ The Flatiron School in New York City charged \$12,000 for 16 weeks,⁶⁷ while Hack Reactor of San Francisco charged \$17,780 for a 12-week program and advertised that its graduates landed jobs that paid an average salary of \$105,000.⁶⁸ Hackbright and the Flatiron School refunded several thousand dollars to graduates who landed jobs through their career-matching programs, and many of the hack schools provided some form of fellowship or scholarship. One 2014 survey polled “39 unaccredited U.S. programming schools offering 40 or more hours of instruction per week” and found that the schools would graduate nearly 6,000 students in 2014, a 175% increase over 2013.⁶⁹

Codecademy in 2014

If 2013 was the year to focus on content and rolling out a marketplace model, Sims and Bubinski were determined to have 2014 be a year of continued growth and experimentation with viable business models. In April 2014, Codecademy unveiled a “reimagined” website that “redefined every component under our brand, from a single button on our dashboard to our email template, business

^c Treehouse’s predecessor, Think Vitamin, started in 2006.

815-093

cards, slides and even apparel.⁷⁰ TechCrunch reported that the redesign was "the first step toward Codecademy's broader vision that will bring in universities, business clients and employers and employment networks, to help Codecademy students move toward actual careers through the site."⁷¹

By the time of the relaunch, Codecademy had more users than the next-largest education website by three-fold (see Figure 1) and roughly 70% of Codecademy's 24 million users were international, up from half in 2012.⁷² Registered users spanned 190 countries.⁷³

Figure 1 Education Website Users (Total, Cumulative)

	Codecademy	Coursera	Udacity	Treehouse
Users	24,000,000	7,400,000	1,600,000	100,000

Source: Ingrid Lunden, "Codecademy Takes Its Online Learning Portal To The UK, France, Brazil, Estonia And Argentina," TechCrunch, May 22, 2014, <http://techcrunch.com/2014/05/22/codecademy-takes-its-online-learning-portal-global-partners-in-uk-france-brazil-estonia-argentina/>, accessed June 2014.

As another phase of the reimagined website, Codecademy's dashboard was redesigned to help increase the ease of discoverability of content for users. The company simplified the user's experience of browsing content "by putting all the skills you can learn in a single page, and secondly, to make it easier to gauge your overall progress on the site."⁷⁴ (See Exhibit 5 for an image of the dashboard.)

Going Global

In the U.S., thousands of after-school programs used Codecademy. However, incorporation of computer programming into public education standards was more complex due to the state- and district-run education system. Sims said the complications of convincing U.S. school systems to incorporate computer programming into state standards were "like a street war."⁷⁵

In May 2014, Codecademy announced it was opening an office in London, where the education system was mandated by the central government. The U.K. was officially adding computing to the national curriculum in September (the first G8 country to do so) and looked to Codecademy for help. Codecademy was being used not only to teach students but also to teach the teachers (in conjunction with more formal training) free of charge.⁷⁶ Over 1,000 British schools were using the website to prepare for the new curriculum.⁷⁷ Sims explained, "We have a curriculum that works along with the standards that were set by the government. . . . It made it easy to shift a lot of the burden of teaching the material onto a platform that's already been vetted."⁷⁸ Although Codecademy was not generating any revenue through the agreement, Sims felt that it was an important initiative: "We look at this as a really good opportunity to look at learning at scale It promotes the work of people who have already created content on the site."⁷⁹

Meanwhile, Codecademy was establishing noncommercial partnerships with other non-English-speaking countries and their education systems. Although Codecademy had announced that they would concentrate on internationalization back when the company got its second round of funding, Sims and the team soon found out that launching foreign-language sites was complicated from the perspective of back-end technology. He explained, "At first we thought, 'Oh, we'll just take our website and translate it to different languages.' But it's not that easy."

Codecademy forged partnerships in Argentina, Estonia, France, and Brazil. Argentina's capital city of Buenos Aires had committed to incorporating programming into all city schools with

Codecademy's help, the first South American city to do so. In Estonia, the company would work with Tiger Leap Foundation, a public-private partnership created in the mid-1990s to promote technology in Estonian schools. In France, Libraries without Borders (*Bibliothèques Sans Frontières*), a nongovernmental organization (NGO) that spread knowledge throughout French-speaking nations, would do translations into French. In Brazil, Codecademy partnered with Lemann Foundation, the largest education foundation in the country, to do Portuguese translations and pilot Portuguese programs on the website.

Still No Revenues

Sims and Bubinski agreed that they had been successful at creating a high-quality learning experience and brand, achieving scale, and building a large community of users. Even though the company still had much opportunity for user growth, the founders felt they needed to begin focusing on monetization to ensure sustainability. Although they had a range of monetization ideas to test, they were in agreement that the site could not charge users for content. "We fundamentally believe educational content is a commodity. There's a strong movement towards open and free education, and if you charge the consumer directly for content, then you're going against that and you can't lead the movement," Sims asserted. "Offering a free product is sometimes a great business strategy," said Weissman, "Codecademy's cost of delivery is getting lower, no one can undercut on price, and being free allows you to reach the greatest audience." He added, "User engagement is so high right now that it seems like the right time to experiment with monetization. But the strategy has to be one that doesn't mess up the user experience for millions and millions of people."

Before testing monetization strategies, the founders thought hard about the company's strengths. Bubinski recalled, "We asked ourselves, 'What are our qualifying criteria for monetization?' We had volume and we believed we could teach valuable skills." Bubinski separated coding skills into three tiers. At the top were what Bubinski called "highly paid Google software engineers." In the middle were "technical non-techs," white collar workers that dealt with data and/or digital products but needed technical skill complements. He believed that "technical non-tech" employees worked at companies that would value a Codecademy enterprise training program. The last tier of skills were those held by freelancers who could find projects posted to digital temp sites, a space that some experts predicted would be \$2.7 billion in 2014. These included sites like Elance or oDesk.⁸⁰

Sims and Bubinski decided to test monetization strategies for helping students enter the bottom or second tiers. "We were pretty comfortable that we could already qualify people for the bottom tier — to do freelance work. But we were unsure of enterprise training at the middle tier." The cofounders decided to run two tests simultaneously: one on a labor marketplace and one on an enterprise service.

Labor marketplace model To understand Codecademy's best approach to entering the labor marketplace, the cofounders felt that the company had to determine its inventory of skills and how best to match those skills to jobs. The company set up a test to run on users, putting dozens of groups into 10 cohorts of 40 people, each cohort testing a different skill. To quickly access a supply of job opportunities, Bubinski and his team sifted through job postings on Elance and mimicked their specifications in an e-mail to test groups made up of users who had completed the corresponding curricula on Codecademy. The test group was told that the tasks were to test their skills and gain more practice using the knowledge they had acquired. The tasks were unpaid. "We wanted to see completion rates first without pay," Sims explained.

Codecademy offered support services for the projects so they could see where users were running into problems and gauge whether Codecademy was failing to teach certain necessary skills. The

815-093

company collected the responses internally and graded the users' output to make assumptions about whether Codecademy's base of users could qualify for Elance-type tasks. Depending on the cohort, a handful of the 40 users in a cohort completed the jobs proficiently. "The experiment wasn't meant to maximize completion. We were simply testing if our users who had completed corresponding curricula on Codecademy qualified to do the work. We wanted to see if we could take a project on Elance without any post processing and give it to users to complete," said Bubinski. "The interesting thing is that people were doing unpaid projects like this just to improve their skills," Sims noted.

The cofounders agreed that the next stage of experimentation would pay the users for their work to test the impact of completion incentives. They also wanted to test what kind of additional support services might be needed to increase the ability and motivation for the users to complete freelance tasks and to test what Bubinski called the "flow"—setting up systems to get the finished projects from the user to the party that had requested the task. "We can solve the flow process to improve our conversion rates," said Bubinski, "That's just engineering."

The other challenge with expanding the labor marketplace model would be content development. Codecademy would need to create a library of learning modules ("Learning Prototypes") that would allow the user to build skills that directly matched the skills employers were seeking in job candidates. This would require a substantial investment in content development and an exhaustive research effort with employers to match skills to the open jobs.

Enterprise model To test an enterprise model where Codecademy would offer a custom curriculum to company clients for their employees, Codecademy ran several pilot programs with a handful of Fortune 500 companies. Each company commissioned the programs and worked with Sims and Bubinski to tailor the content for their learning objectives. The companies all paid for programs. "We wanted money to be involved, because we wanted to get good information. If you give a company something for free, the information you collect from them isn't necessarily information you can depend on. We wanted their expectations of quality to match the dollars they were spending," explained Bubinski.

The company developed one enterprise curriculum targeted not at programmers, but rather at technical non-techs, which was shared across all pilot programs. "The curriculum basically breaks up into helping people use digital tools to build digital products, communicate data, or process information," noted Bubinski. To implement the pilots, Codecademy sent an instructor to each company to teach classes on Codecademy to the employees for a day. For the next three months, Codecademy sent the company data on employee performance and online course completions.

After completing the enterprise pilot programs, each of the four companies requested company-wide training programs. Demand for those training programs emerged across all the departments of the companies who wanted to make sure their staff was technically proficient. Sims commented, "When a company comes to us and says that they want to roll out our platform to their entire company, we look at it as we're getting paid to acquire more customers and grow our network."

After the pilot programs, Codecademy gathered data on the perceived value of the program. They asked enterprise employees if they learned skills they could apply to their job and how well they felt they were taught. "We also surveyed the companies to see where they found the value," said Bubinski. The client surveys showed favorable results and that companies were willing to pay for Codecademy services and its platform, as much as six to seven figures in annual contracts. It was not completely clear where the budgets would come from. On the one hand, training budgets were available if there was a direct link to a return on the investment in terms of hard skills development. But the original courses were more focused on softer skills, such as technical literacy for nontechnical

employees, and Codecademy would need to adjust the content and the sales efforts to better fit the training department buyer.

In 2013, total spending for employee learning in the U.S. was \$164 billion, or \$1,195 per employee.⁸¹ Many platforms were competing for training employees, such as Lynda.com, OpenSesame, or publishing company Wiley's Digital Classroom. Coursera, Udacity, and Udemy were also piloting and launching new programs for an enterprise audience.⁸² Program pricing typically ranged from \$20-\$40 per employee per month.

Conclusion

As the board meeting neared, Sims and Bubinski were still debating which experiment they should focus on: a labor marketplace or the enterprise model. Either choice would put Codecademy up against some formidable competitors.

Sims pointed out that a labor marketplace model would not require a large human capital investment and could scale relatively easily, while an enterprise model would require hiring full-time instructors to teach enterprise clients' employees and a large sales staff to call on training departments to sell the programs. Weissman had also reminded them that delivering a product to the enterprise market would mean Codecademy would have to deliver it at an enterprise level of service, which had not been a primary focus of the startup.

On the other hand, an enterprise model had a straightforward go-to-market strategy with clear demand. The labor marketplace model's go-to-market strategy was less certain, which would lead to a less concrete valuation outcome. Having an uncertain go-to-market path did not prevent them from building the largest learning community of coders, but the Codecademy team was concerned that the recruiting market was already trained to tap other communities, such as LinkedIn, and might not initially pay to access the Codecademy community.

The cofounders knew that either monetization route would pose its own set of challenges but that they had to make a decision. "As a 25-person company trying to create a movement, we have to be very focused," Sims acknowledged. "We have a long-term vision to become the vocational school for the 21st century, but what's the right next step to take us there?"