Searching for a Scalable Solution to Diabetes Management

Kevin Tseng

Electrical Engineering and Computer Sciences
University of California at Berkeley

Technical Report No. UCB/EECS-2012-128
http://www.eecs.berkeley.edu/Pubs/TechRpts/2012/EECS-2012-128.html

May 30, 2012

Searching for a Scalable Solution to Diabetes Management

By

Kevin Tseng

A Master's project written report submitted in

partial satisfaction of the requirements for the degree of

Master of Engineering

in

Engineering – Electrical Engineering and Computer Science

in the

Gradudate Division

of the

University of California, Berkeley

Committee in charge:
Professor Ikhlaq Sidhu, Capstone Advisor
Professor Pieter Abbeel

Spring 2011

Abstract

Searching for a Scalable Solution to Diabetes Management

By

Kevin Tseng

Master of Engineering in Electrical Engineering and Computer Science

University of California, Berkeley

Professor Ikhlaq Sidhu, Capstone Faculty Advisor

Over 25 million people are diagnosed with diabetes in the United States [1]. If current trends continue, more than half of American adults will have diabetes or pre-diabetes by the year 2020 [2]. This diabetes epidemic is growing as the popularity of Western diets and fast food restaurants continues to expand [3]. In order to address this alarming trend, my capstone team has spent the past year exploring solutions that can 1) help diabetics' live healthier lifestyles and 2) form the basis of a scalable business.

Each of our solutions has followed the mantra "mobile first." With worldwide mobile shipments outpacing those of PC's [4], mobile devices have become a ubiquitous technology and will give us a wide-reaching platform to reach diabetics around the world. In order to efficiently search the space of scalable diabetic solutions, my team adopted the Lean Startup approach [5] of garnering information from customers and pivoting from one business idea to another. This paper details the technical challenges my group has faced and the solutions we've proposed to address them.

Introduction:

The number of new Type 2 diabetics diagnosed every year is rapidly increasing. If current rates continue, 1 in 3 United States adults will be diagnosed with diabetes by 2050 [1]. With the globalization of Western diets and the convenience of fast food restaurants, this alarming trend looks to hold for the foreseeable future [3].

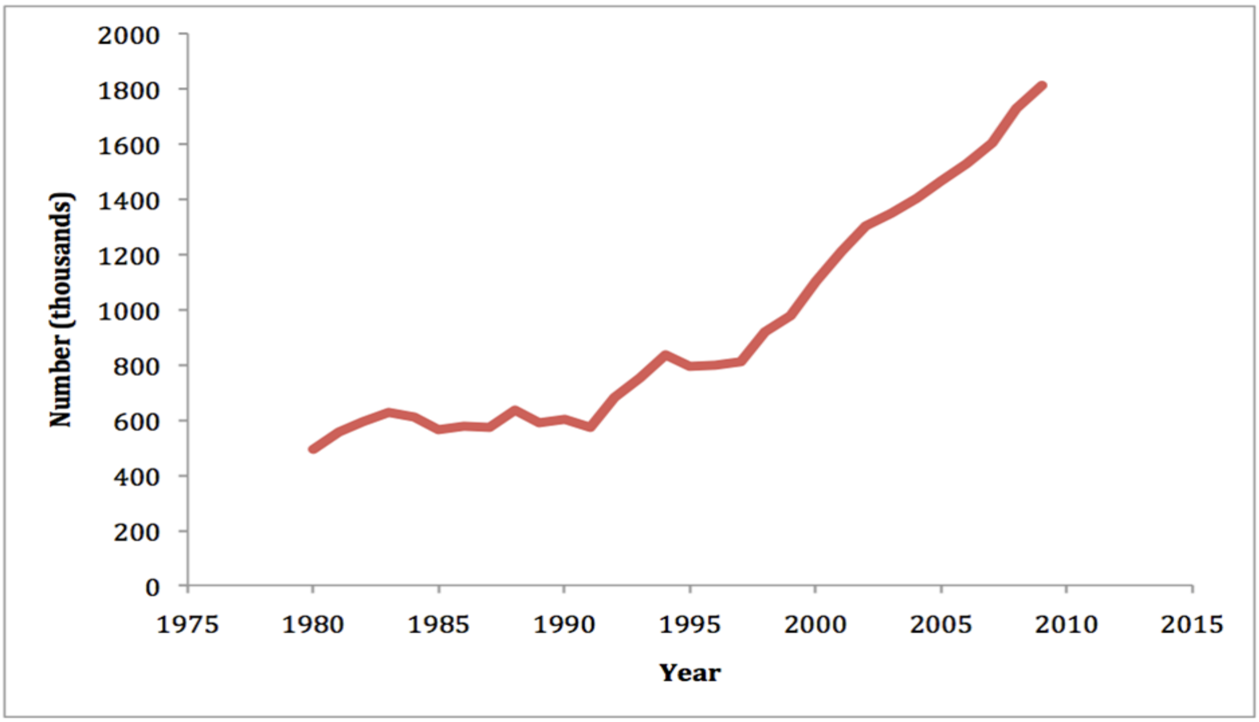

Figure 1: Increase in number of diabetics from 1975-2010 [1]

This massive increase in the number of diabetics will have a profound impact on the health care system. Currently, $218 billion of our health care funds go into the treatment of diabetes [6]. The health care industry is already over-burdened and as the need for diabetes medications and treatments increases, exorbitant health care costs for the public will grow. Diabetes care direly needs new innovations within the technology space to curtail these troubling trends, and many of these new solutions will undoubtedly utilize the booming smart phone market.

With the advent of the smart phone, consumers now have a handheld device capable of accessing the Internet and running complicated applications. This new category of device has ushered in a new era of mobile computing that has pushed the boundaries of technology and has driven innovation over the past four years. The smart phone movement is sure to gain momentum in the coming years. According to one study [7], 300 million mobile smart phones were sold in 2010. This number is expected to balloon to about 1 billion smart phones by 2015, representing more than a threefold increase in four years. The future of technology lies in taking advantage of this mobile ubiquity.

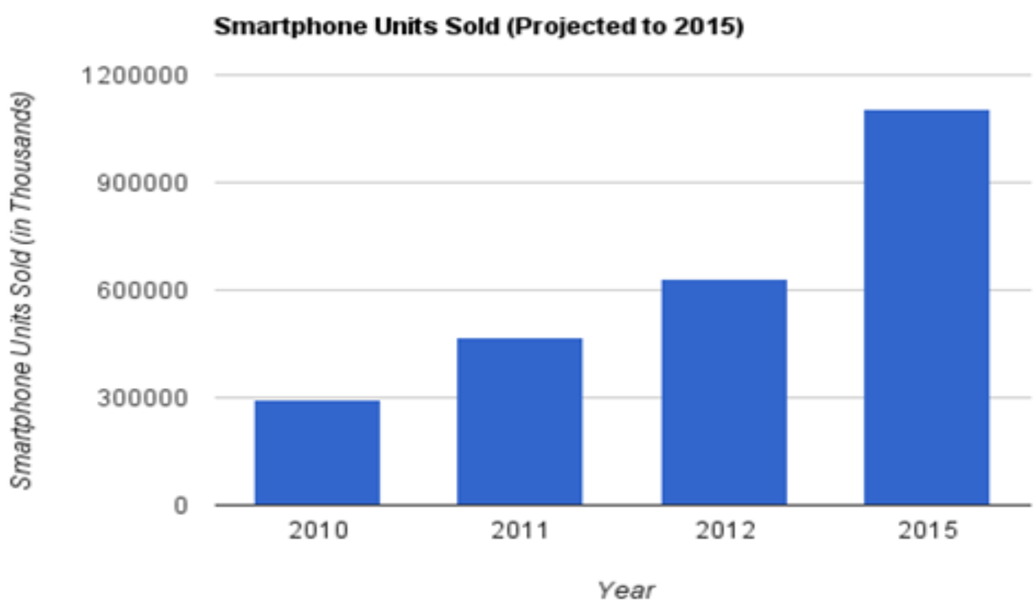

Figure 2: Projected increased smart phone usage through 2015 [7]

Although the marketplace for mobile applications has been expanding along with the increasing number of mobile devices, effective solutions for diabetic management have been scarce. The challenges of creating a useful and scalable solution for diabetics are two-fold. First, it must address a significant pain point shared by diabetics in an innovative, yet intuitive manner. This aspect is necessary in order to drive user adoption. Secondly, the application must be easily distributable. Developing for the smart phone platform will allow us to obtain this widespread channel.

My team began our search of a solution fulfilling these two criteria by developing a mobile glucose monitoring system. A team attending Berkeley's Engineering Leadership Professional Program investigated the promise of such a device, and we used their research as a starting point for our quest. From this initial idea, we surveyed close to 100 diabetics around the Bay Area to ask for customer feedback and adapted our model accordingly.

Our final product eventually evolved through three distinct phases. Our first idea aimed at predicting glucose levels based off a diabetic's previous glucose readings. We sought to correlate food choices and portion sizes with a diabetic's previous blood glucose spikes and dips to make predictions on what a diabetics' glucose levels might be. Our second iteration centered on building a glucose-monitoring platform that would analyze past glucose readings to provide the diabetic an informed assessment of his or her diabetes management. Part of this solution included employing gaming mechanics to help motivate diabetics to be more proactive with their diabetes management. Our third and final product pivot was the development of a mobile web application that would advise diabetics on what restaurants around the area are most diabetic friendly. Our aim is to build a database of diabetic-curated restaurant ratings that shows healthy food menu items so that other diabetics can select restaurant food items more intelligently.

Literature Review

Each of our three potential products had unique technical challenges that we faced. Luckily, my team had an extensive literature base to look through to aid us in our endeavors. The following subsections detail some of the obstacles we faced and how prior research aided in our progress.

Glucose Level Prediction

The biggest challenge with this idea was discovering what mechanisms are involved in predicting blood glucose levels (BGL). The first step in this process was to find a way to accurately model a diabetic's past glucose readings.

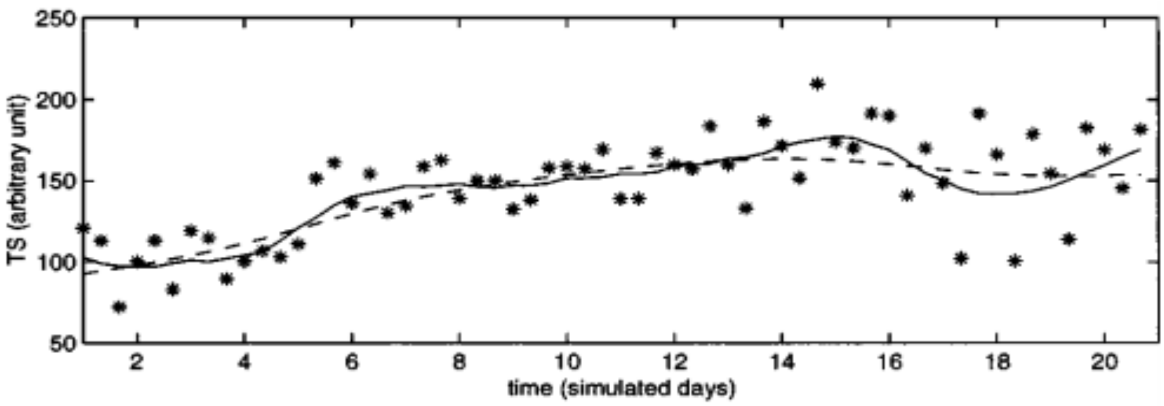

Figure 3: Time series modeling of blood glucose levels [8]

Historically, BGL modeling has been analyzed as a time series [9]. In this approach, BGL are broken down into two components: 1) a cyclic component describing the variations in BGL that occur on a daily basis and 2) a trend component describing the long-term variations in BGL over a time period. Unfortunately, these sorts of time series analyses fail to take into consideration the myriad of parameters that can account for BGL variations. For example, patho-physiological changes to the diabetic's body (e.g. fighting the common cold) could affect the time series structural analysis and return an inaccurate modeling.

To compensate for this, [8] recommends treating the cyclic and trend components as randomly varying with time, thereby mitigating the effect these sorts of fluctuations may have on our modeling. In order to handle the extra complexity that go along with these assumptions, [8] employs the use of Markov Chain Monte Carlo (MCMC) algorithms, which rely on an iterative estimation process.

The MCMC approach proved useful in understanding the complexity of modeling BGL but did not yield a suitable technique to make predictions. Ultimately, the computational burden of the MCMC method proved to be too significant for mobile devices and the proposed approach was never intended to make predictions [8]. Furthermore, FDA restrictions on medical devices as well as the legal consequences of making incorrect predictions deterred us from the prediction route.

Analysis Platform

After deciding to move away from predicting glucose levels, we turned instead to an analysis platform for diabetics. Instead of predicting future blood glucose levels, we focused on displaying relevant information such as average BGL, variance, and unhealthy spikes or dips. To increase user participation and encourage a healthier lifestyle, we incorporated simple gaming mechanics in the form of a leaderboard, displaying which diabetics managed their diabetes the best.

Hybrid Applications

In order to reach the largest market of diabetic users, my team wanted a solution that could reach both Android and iOS users. Normally, this would require developing two separate natively running applications for both sets of hardware. However, breakthroughs in mobile technology have introduced what are known as hybrid applications that allow developers to program smart phone applications using HTML5, CSS3, and JavaScript [10]. Because applications are developed using these standard web technologies, programmers can develop a mobile application using a single framework and deploy it to multiple smart phone platforms.

While shifting to hybrid applications has its benefits, there are also significant disadvantages with the move. First off, native applications generally perform better than their hybrid counterparts. Secondly, hybrid applications tend to have less access to the smart phone's hardware components, such as the phone's camera, accelerometer, or GPS. Lastly, user interface development is much more difficult for hybrid applications because native platforms give developers abstractions for common user-interface controls and experiences [10].

Luckily, many of these detriments are being rapidly addressed within the developer community. While native applications currently outperform hybrid applications, these performance differences are only noticeable for CPU intensive processes such as 3D games. Furthermore, with improvements in JavaScript interpreters, the performance gap between native and hybrid code is closing. Furthermore, application platforms such as PhoneGap [11] and Titanium [12] are providing hybrid applications access to hardware peripherals and extensive JavaScript/CSS libraries such as Sencha Touch [13] and JQuery Mobile [14] are providing developers a way to easily program hybrid applications with a touch-friendly user interface.

Security Concerns

One distinct difference between the traditional client-server model of web applications and the hybrid application model for mobile phones is that in the former case, web pages are served to users by a server, while in the latter case, pages are stored on the phone. This leads to an interesting security problem when trying to retrieve data from a server.

Normally, if a web page wishes to retrieve information from a server, it performs an AJAX request to do so. However, the same-origin policy in browsers restricts data transfers of this nature between server and browser to occur only for web pages that have been serviced by that specific server. All modern browsers have implemented this policy in order to protect users from malicious JavaScript behaviors. For hybrid applications, however, this becomes a problem because web pages are essentially stored within the phone (and not on the server). Thus, retrieving leaderboard updates from a hybrid application would violate the same-origin policy.

In order to get around this problem, some changes needed to be made on both the server-side and client-side code to implement the JSON-P protocol [16]. JSON-P stands for "JSON with padding." Normally, servers and web pages communicate with each other by sending JSON objects back and forth. However, if unaccounted for, malicious web pages may make requests to servers asking for JSON objects filled with personal information. This is one of the reasons why browsers institute the same-origin policy: to protect viewers from this sort of scenario.

JSON-P counteracts this by forcing another level of handshaking between the server and the client. When requests are made to the server, the client must send an extra parameter designating a callback function. This callback function will be executed immediately after the client receives the data from the server. When the server receives a JSON-P request from the client, it wraps the JSON object it returns with this extra callback function so that the client can make an exec call on the object it receives. This extra measure needs to be taken on both the server and client side and allows JSON-P requests to be more secure than typical JSON requests.

Diabeats

After conducting multiple customer interviews, we found that many diabetics had a strong need for finding healthy restaurants to eat at. Oftentimes, diabetics are uncertain of the nutrition contents of food they don't prepare themselves. As a result, diabetics don't know what food portions they must eat to maintain healthy blood sugar levels or what insulin dosages to take before their meal.

My team saw an opportunity to solve this pervasive problem by creating a mobile web application that collects feedback from diabetics on how healthy it is to eat at various restaurants. Thus, when other diabetics are curious to see what food

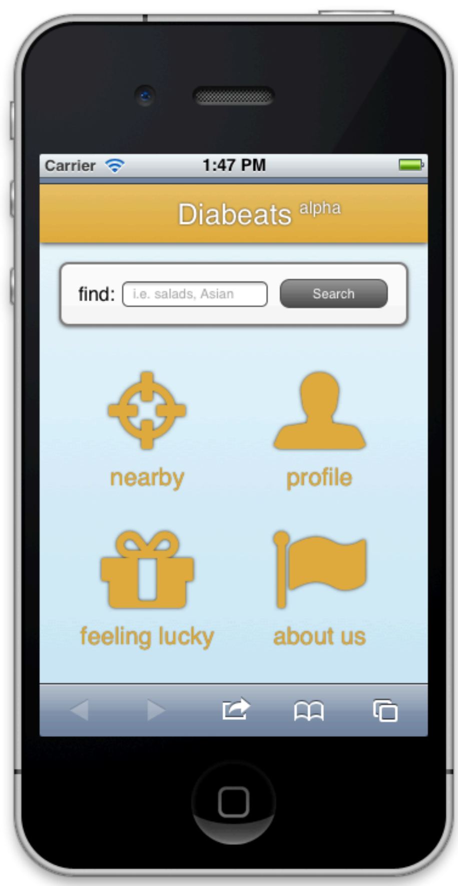

Figure 4: Screenshot of Diabeats Application

choices would be the best for them at a particular restaurant, they can use Diabeats to help plan out their meal.

OAuth

When my team first set out to build Diabeats, we wanted to develop an application that would appeal most directly to mobile phone users. In order to accomplish this, we optimized the user interface for mobile screens and built around geolocation to find restaurants around the user. Instead of forcing users to type in restaurants not currently in our database, our service queries Factual's [17] extensive database of locations to populate restaurant information.

While Factual's database proved to be a very useful resource, one of the limitations was that Factual severely limited the number of queries for unsigned requests. Factual's servers support larger rate limits, but only if the user makes requests using two-legged OAuth authentication [18].

Most major web services, such as Twitter and Facebook, only allow access to their APIs through OAuth authentication [19] [20]. Much like the JSON-P format mentioned earlier, OAuth enables the secure exchange of data between client and server. This is done primarily through the use of public and private key encryption and signatures [21] using well-established encryption methods such as HMAC-SHA1. By incorporating OAuth into our web service, Diabeats was allowed to query Factual's database with much higher frequency and volume. This was crucial in order to support many users and maintain scalability.

Methodology and Approach

Our methodology and approach to our capstone project continually evolved throughout the year. My team steadily adapted our plans as we learned more about diabetes, its treatment, and the concerns diabetics face on a daily basis. In addition to our own investigating, we performed interviews with endocrinologists, healthcare corporations, non-profit diabetic associations, and diabetics that greatly influenced our trajectory as a team.

At the beginning of the school year, my team had no experience or prior knowledge about diabetes. Thus, for the first few months, my team aggressively researched diabetes as a disease and an industry. We learned what diabetes' causes and treatments were, what percentage of Americans were diabetics, and what profile a typical diabetic has. We also studied who the major players were within the diabetes ecosystem. This analysis encompassed a wide range of topics, from finding what corporations were the predominant manufacturers of diabetic supplies to what mobile applications have been released that claim to help diabetics.

In order for us to relate to diabetics on a more personal level, two of my teammates purchased glucose meters and pricked themselves for a month while measuring their blood glucose levels, essentially living the life of a diabetic. This process helped us empathize with our would-be customer base by allowing us to uncover pain points diabetics have that we wouldn't have discovered otherwise.

Once our group had accrued enough data about diabetes, we began to research possible solutions we might pursue. For example, we began reading papers on what mechanisms are involved in predicting future glucose levels. Another potential avenue we looked in to was developing a cheap glucose meter that could sync directly with a mobile application. Additionally, my team looked into possible patent violations and FDA regulations our application might need to follow. By the end of the first semester, my team had a firm grasp of what players were in the diabetic market as well as what significant barriers to entry we needed to overcome.

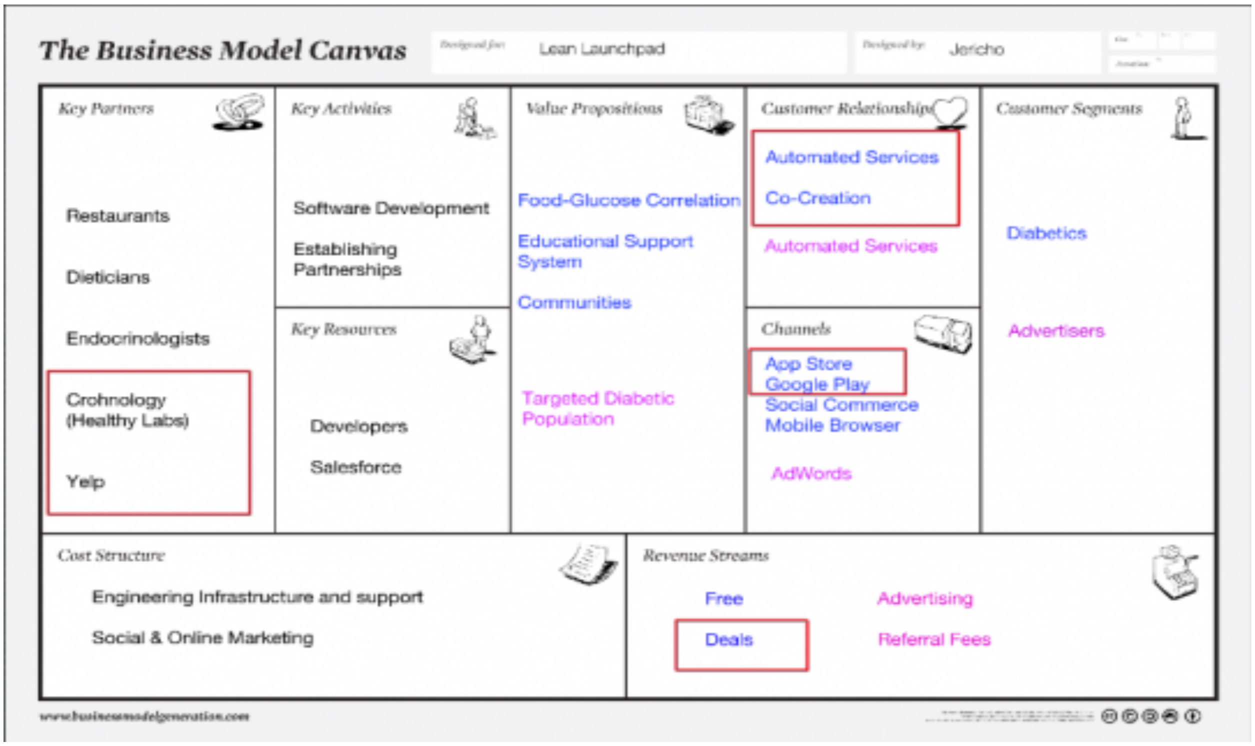

Figure 5: Business Model Canvas

At the start of the second semester, our approach became much more customer-oriented. We adopted the Lean Startup mentality and made decisions based upon customer feedback. Every week, we set out to speak to ten different people who could give us useful insight. This meant reaching out to diabetics, doctors, endocrinologists, and health bloggers for their opinions. We continually set out to prove or disprove hypotheses that we proposed while making the appropriate changes to our business model canvas. Eventually, we amassed a catalog of diabetics whom we could contact via email for quick surveys.

While we collected customer feedback using detailed wireframe models as a sort of minimum viable product, our group started work on developing a mobile application. I was tasked with most of the software development, including writing the client-side HTML and JavaScript and the server-side logic to would handle all the incoming requests. For our hybrid application, we used the PhoneGap library because it provided us an easy way to deploy to both Android and iOS. When we pivoted to the mobile web application, we decided to host our application on Google App Engine and continued developing using these same web technologies.

Once we had enough customer data to validate that a significant number of diabetics wanted help with eating out at restaurants, my team concentrated on the programming aspect. This included setting up the necessary software resources (code repository, testing environment, etc.) careful planning of the software architecture, assigning of tasks, good communication between team members, and extensive testing.

Currently, our initial product has been released to the public (http://www.diabeats.co), and we are actively searching for diabetics to evangelize and spread our product. We have emailed our diabetic contacts, contacted several diabetic blogs, and setup targeted Facebook ads. Customer acquisition continues to be an ongoing process.

Discussion

As an entrepreneurial capstone team, we were given the flexibility to make decisions on what form our final product would take based on market analysis and technical concerns. While many decisions were made without perfect information, all of our choices had a strong rationale associated with it.

Our initial product idea of predicting blood glucose levels was decided upon after speaking with an expert in the healthcare field. Given the heavy computer science background of the team, we felt we could leverage our knowledge of machine learning to make predictions based off of blood glucose history.

However, once we dug deeper into the literature, we quickly shifted product ideas. The research papers found suggested that making blood glucose level predictions would require heavy computation [8]. Additionally, it was highly unlikely we could generate accurate predictions without considering biological factors other than diet [8]. Furthermore, FDA regulations as well as possible legal repercussions for making incorrect predictions swayed us from continuing this route.

We eventually traded prediction for analysis to be the cornerstone of our application. Instead of making predictions, we would display relevant statistics regarding diabetic health. Initially, we would simply calculate average blood glucose levels and variance and use beta tests to uncover what other statistics would be valuable for diabetics to have. While we felt this could be of great use for diabetics, my team also noted that this idea was not novel and likely to be insufficient to change diabetic behavior. Thus, in order to promote healthier changes in lifestyle, we added a social gaming component where diabetics using our smart phone application would compete against each other to see who could obtain the highest “health score” for best diabetes management.

This new pivot introduced several technical challenges. The first was deciding what mobile platform we should develop on. We ultimately decided that going for a large target audience would be the best route, and the quickest way to do this would be to develop a hybrid application because of its program once, deploy everywhere scheme. The second challenge was how to create a simple social network without exposing private information to malicious parties. In our case, we needed to make sure that everyone could view other users' leaderboard scores without allowing hackers the opportunity to steal potentially sensitive information. To solve this, I devised a moving average formula that would save all blood glucose values on the diabetics' device and send only enough information to calculate each user's health score, without exposing individual blood glucose measurements.

After demonstrating our wireframe prototypes to diabetics, we found alarmingly little excitement about our idea, despite our many attempts at tweaking the product to be more attractive. However, one recurring sentiment expressed from diabetics was how difficult it was to select healthy items on restaurant menus. My team quickly established that this could be a new market for us to explore, and we pivoted our product idea accordingly.

One major issue with our idea was the platform with which we wanted to distribute our application. While our previous development always targeted mobile applications, we felt that it would be more convenient if we, instead, switched to a mobile web application. This meant developing a user interface that would work optimally with mobile devices but could still be used by conventional computers. Developing "mobile first" was a point we stressed because my team felt that most interactions that diabetics would take with our service would be when they were on the go, searching for healthy restaurants around them.

Once we completed our initial beta release, our contact list of endocrinologists and diabetes bloggers greeted our new idea with warm acceptance and excitement. One contact at UCSF distributed our application to their mailing lists of hundreds of diabetics, and we were even featured as an article on http://www.diabeticconnect.com [22]. While we are still trying to accrue a critical mass of users, we are optimistic about our initial results.

Conclusion

While my team's goal for this project has been to create a scalable business, our interests have always been altruistic: to build a product that can help diabetics lead healthier lifestyles. Operating under these guidelines has proved to be a challenging, but extremely fulfilling journey this school year, and I have learned immensely about topics varying from diabetes, the healthcare industry, creating a startup, and developing a mobile application.

From our initial research stage, my team learned about diabetes as a disease and its effect on those who are diagnosed with it. We realized that diabetes is an epidemic, and that healthcare costs are quickly rising due to the growing population of diabetics. While this meant that there was a definite market for our product, it also meant that our work could impact millions of lives as well.

Once we started our market analysis of the diabetes industry, we began to realize how only a few players dominate the diabetic supply market and what this meant for us in terms of business opportunities. For example, creating glucose meters or test strips as a solution would most likely lead to a dead end due to our lack of clout and reputation in this space. Furthermore, FDA regulations and other legal restrictions deterred us from pursuing these routes.

When it became time to focus on building a product, my team was aided tremendously by adapting the Lean Startup methodology. I experienced how important it was to seek out customer verification and that pivots and business model changes are a frequent habit in the startup world. I learned the value in of a minimum viable product and why creating and testing hypotheses are truly the keys to being agile and efficient.

Finally, I learned much more about my technical craft. This included the benefits of JSON-P, how to develop hybrid applications, and how OAuth authentication allows for secure communication between server and client.

This past school year has marked a significant period of growth and learning in my life. While I've earned a technical Masters degree by taking deep coursework in my field of study, I have also learned several concepts in business and leadership. The Masters of Engineering program has opened my eyes to the opportunities for engineers beyond technical work, and the capstone project has been the catalyst for this change. After I graduate, I hope to take with me the concepts of engineering leadership and apply them and make a difference in the world.

Sources

[1] Center for Disease Control, "Diabetes: At a Glance" (2011). www.cdc.gov/chronicdisease/resources/publications/aag/pdf/2011/Diabetes-AAG-2011-508.pdf

[2] UnitedHealth Group, "The United States of Diabetes, *Working Paper 5" (2010).* http://www.unitedhealthgroup.com/hrm/UNH_WorkingPaper5.pdf

[3] Mintel - The US Casual Dining Market - August 2011

[4] "Android Rises, Symbian 3 and Windows Phone 7 Launch as Worldwide Smartphone Shipments Increase 87.2% Year Over Year, According to IDC", IDC (2011). http://www.idc.com/about/viewpressrelease.jsp?containerId=prUS22689111&sectionId=null&elementId=null&pageType=SYNOPSIS

[5] Lean Startup: http://theleanstartup.com/

[6]. Dall, T.M. et al., "The Economic Burden of Diabetes", Health Affairs (2010).

[7]. “Gartner Says Android to Command Nearly Half of Worldwide Smartphone Operating System Market by Year-End 2012”, Gartner (2011).
http://www.gartner.com/it/page.jsp?id=1622614
[8] Riccardo Bellazzi, Paolo Magni, and Giuseppe De Nicolao. “Bayesian Analysis of Blood Glucose Time Series from Diabetes Home Monitoring”, IEEE TRANSACTIONS ON BIOMEDICAL ENGINEERING, VOL. 47, NO. 7, JULY 2000
[9] T. Deutsch, E. D. Lehmann, E. R. Carson, A. V. Roudsari, K. D. Hopkins, and P. Sönksen, “Time series analysis and control of blood glucose levels in diabetic patients,” Comput Meth Programs Biomed, vol. 41, pp. 167–182, 1994.
[10] Andre Charland, Brian LeRoux, “Mobile Application Development: Web vs. Native,” Communications of the ACM, Volume 54 Issue 5, May 2011
[11] PhoneGap: http://phonegap.com/
[12] Titanium: http://www.appcelerator.com/
[13] Sencha Touch: http://www.sencha.com/products/touch/
[14] JQuery Mobile: http://jquerymobile.com/
[15] Same origin policy: http://www.w3.org/Security/wiki/Same_Origin_Policy
[16] JSON-P: http://www.json-p.org/
[17] Factual: http://www.factual.com/
[18] Adrian Hannah, “A Primer to the OAuth Protocol,” Linux Journal Issue #206, June 2011
[19] Twitter OAuth: https://dev.twitter.com/docs/auth/oauth/faq
[20] Facebook OAuth: https://developers.facebook.com/docs/authentication/
[21] Randall C. Reid, Richard G. Platt, June Wei, “A Teaching Module to Introduce Encryption for Web Users,” InfoSecCD '05 Proceedings of the 2nd annual conference on Information security curriculum development
[22] Diabetes Connect: http://www.diabeticconnect.com/discussions/15305-diabeats-co-empowering-diabetics-to-eat-out-smartly