Waitr
A Complete Data Solution For Restaurants And Their Guests To Make The Most Of The Dining Experience.

Authors: Daniella Erulker, Sahil Kumar, Anushka Makhija, Liz Reckart, Kevin Thomas
Advisor: Boon Thau Loo

Abstract

Over half of surveyed UPenn students eat out at least once a day, and almost a third have between four and seven food-related apps, making the restaurant industry a significant part of the local Philadelphia economy. However, the digital restaurant experience is completely decentralized; consumers need to use several applications and services to complete basic tasks, from identifying restaurants to splitting the bill. At the same time, smaller local restaurants have difficulty pinpointing meaningful data that can translate into insightful and actionable takeaways. Waitr mitigates these issues by centralizing the customer restaurant experience into one app and thereby provides restaurants with access to data that they would not have had before. Our mobile application provides consumers with a one-stop shop for restaurant identification, reservations, in-house ordering, and payment sharing, while restaurants can manage related operations - such as table management and order processing - via a web application. Restaurants can also easily view analytics based on aggregated customer data, such as total customer visit time, hourly demand, trends across popular food items, customer demographics etc. and use those analytics to inform actions like promotions, marketing, operating hours, etc.

Motivation and Needs Addressed

The average American household spends about $3,000 annually dining out. All together, these transactions have built the restaurant business into a $4 billion industry. However, across the industry there is significant fragmentation in the restaurant experience, both on the consumer and restaurant ends. On the consumer front, customers encounter multiple disconnected reservation, ordering, and payment experiences. Similarly on the restaurant side, restaurants currently utilize a variety of different platforms to manage reservations, orders, payments, and to visualize analytics on their business. For payments these applications range from point-of-sale systems like Square Up and Shopkeep to digital applications like Venmo and PayPal. While some restaurants still utilize written orders, others have integrated ordering systems like Ziosk into their establishment. In addition, with the rising popularity of delivery services like GrubHub and UberEats, more and more restaurants are offering online takeout options either through these sites or through their own. Our team saw a need to reintegrate the entire restaurant business funnel, from the point that the customer searches for a restaurant all the way to restaurant owners' evaluation of their business's sales.

First, we'll cover the pain points associated with the consumer restaurant experience and consumer needs. In the ideal solution, customers need a centralized place to make reservations, order, and pay during a restaurant experience. The current purchase journey for a consumer involves the following steps: Find a restaurant (Yelp, Maps, etc.), reserve a table (OpenTable), order (in person, Ziosk), and pay (in person, Venmo). These numerous different platforms require different accounts, do not share information with one another, and can add unnecessary time to the restaurant experience. Moreover, the payment process specifically is a significant hassle within the system, particularly with parties needing to split their bill at the end of a meal. Usually, one person person pays the bill and struggles on their own time to calculate the tax and tip for each individual, finally using Venmo or some other method to request payments. Parties need an easy way to split bills, that eliminates the middleman and avoid placing the financial burden on any one person.

On the other end, business owners need a more robust platform for integrating their daily operations and taking advantage of performance metrics to grow their brand. Currently, many restaurants utilize multiple platforms to manage reservations and ordering. This leads to a disjoint experience for staff, and also potential loss of valuable data from platform to platform. Furthermore, according to the annual CMO survey, restaurants with revenues less than $25 million spent 11% of their revenues on marketing. This budget is spread across a variety of different media, such as search engine optimization (SEO), targeted ads, social media, and a variety of non-digital platforms. With platforms such as Ziosk, the only method for determining user performance is through customer surveys taken post-purchase. Surveys like this have a low response rate, and this leaves restaurant owners with no means to determine user preferences in

order to gain meaningful and actionable insights from their business. With the right data, restaurants could develop new ways to maximize their profits such as selective pricing, responsive dynamic menus, and special, targeted promotions. These business owners need a more robust platform to track their performance and streamline their operations all in the same place.

Technical Approach

This system contains two main applications: the customer-facing application, available to everyday users, and the business-facing application, available to restaurants. The frontend applications for both of these are supported by a robust joint backend. In this section we discuss design and development decisions made for all three pieces of this system.

Business-Facing Application

We first discuss the business-facing application. We determined that a web application would be the best choice for three reasons. First, most of the data analytics would be done on a computer as opposed to a mobile device, and a web system allows restaurants to avoid the complications of purchasing, downloading, and installing software. Second, its easier for us to give our customers the most up-to-date version of the application and monitor our customers usage with a web app than it would have been for standalone desktop software. Third, a web-based application, as compared to an application built for tablets, allows restaurants to quickly integrate our platform without a large upfront hardware cost; most restaurants already have desktop computers.

The business-facing application consists of three key parts:

1. *The Host View*: this page allows hosts in any given restaurant to manage reservations, checking in parties who have arrived, assigning them to a table, and seeing if the party has paid for their meal all through this portal.
2. *The Waiter View*: this page allows waiters to view orders placed by their customers, with each order linked to a specific party & table. Waiters remove the orders from the queue as soon as they are served.
3. *The Business Analytics*: this page is the visual interface where restaurant owners and managers can monitor and interact with customer data. They can view information such as data and trends such as most popular dishes, demand by date, demand by time, etc.

The first two parts listed above support the restaurants operations and are interwoven with the customers experience. This is highlighted in the following section.

Customer-Facing Application

The customer-facing application was created as a PWA/web application intended for visits on mobile devices. Choosing a web app over an iOS-native or Android-native mobile application was a deliberate design decision with four key reasons backing it up. First, to reduce the barrier to entry. Visiting a URL is much easier than downloading an application. Second, a repeat user can install the web application to their phone to act as a mobile-native application thanks to the PWA compatibility installed. Third, our users can easily invite other consumers dining with them to use the application. This is explained in greater detail below. Lastly, choosing a web-based front end for both the business and customer applications allowed us to make use of Bootstrap libraries to create an elegant user interface with easier, faster, and more consistent front-end development.

The customer-facing application consisted of six key pieces that flow together to create the customer experience:

1. *Home Page*: this page is dominated by a map centered at the users current location, populated by restaurants in the area. It also features a slide-out options menu, a search bar to identify restaurants, and a reservations button which takes a user to a their reservations.
2. *Restaurant Page*: this page can be reached when a restaurant is selected through search. It features detailed information about the restaurant and allows the customer to find and make reservations at the restaurant.
3. *Reservations Page*: a page that lists out all of a customers upcoming reservations. Once the customer is in the vicinity of the restaurant and is within 15 minutes of their reservation, they can check in at the restaurant. This informs the host of the restaurant that a party has arrived in the host view of the restaurants web application (as shown in the business-facing section above).
4. *Checked In*: while a user is waiting to be seated at their table, they can copy the URL from their checked in page and share it with the other members of their party via text or other communication method. When another user clicks on the link, they are automatically added to this party.
5. *Menu Page*: once a party is checked in and seated at their table, all users with the link mentioned above are taken to the ordering page, where they are shown a menu of the items available and can place an order (which is aggregated together and displayed in the waiter view on the business end).
6. *Payment Page*: once a party has been served all of their food, all users in the party are taken to the payment page, containing a list of items ordered by the party. Users can select items they want to pay for. If more than one user selects an item, the item is split across the two users in real time, allowing users to split costs for shared dishes.

Technology & Development

Express, jQuery, Bootstrap, and Node.js form the core of both the business-facing web application and the customer-facing web application. The choice of Node.js for the backend was because our system was a low computation, high concurrency and high I/O system for which Node.js excels at. On the business front-end, Chart.js is used for analytics visuals. On the customer front-end, we use the Google Maps API & Geocoding to create the home page functionality.

Both these backends were supported by a MySQL database, manually managed via PHPMyAdmin software installed on the server. Below is a simple chart outlining our database schemas. For detailed information, please see Appendix B.

TABLE NAME	DESCRIPTION
Counters	Used as a counter lookup to generate unique values.
Users	Used to store login info.
Users_Pat	Used to store access keys.
Users_Pending	Temporarily stores user registration and SMS key.
Restaurants	Stores general restaurant information.
Menu_Items	Items listed on each restaurant's menu.
Party	Tracks a party through their restaurant visit.
Tables	Tracks availability of tables for each restaurant.
Orders	Stores all orders' information for each of the restaurants.

Figure 1: *Database Tables Descriptions students.*

To create the real-time payment system, all individual parties were linked using Socket.io, chosen because of its simple API and easy integration into a scalable Node.js system.

Our restaurant data was acquired through data scraped off of sites like Zomato using a home-built Selenium web-scraper. This scraper helped us populate the database with seed data on restaurants.

Our infrastructure was built using Git (hosted on GitHub) as the versioning tool to organize our work with a deployment webhook to automatically deploy to an Ubuntu 16.04 server.

Evaluation

Since our application centers on the customer and restaurant experience, the best form of evaluation is feedback from our target users. For this reason, we set out with the goal of collecting feedback from both customers and restaurants on the usability, relevance, and improvements for Waitr.

On the customer side, we demoed the application for 25 potential target users. These users each had different eating habits, some ate out more frequently than others. We attempted to put together a diverse pool of users of different ages and demographics. After presenting the demo, we asked the users to answer a brief survey of questions, rating their responses on a scale from 1 to 5. The questions used to evaluate the customer application, along with the average response to each question can be seen documented in Table 1 of the appendix.

Aside from quantitative results we also recorded qualitative feedback from users. Some of the most common feedback we got about the app was that the UI needed to be improved to compete with existing applications. We also found that users did not value the restaurant search or ordering functionalities as much and this can be seen reflected in the responses. Users relayed that they would rather use platforms like Yelp where there are reviews and ratings to choose a restaurant, which indicated to us that a review system would be a valuable future addition to the Waitr application. On the other hand, ordering is usually handled by the restaurant, and incorporating it into the app did not immediately provide value for users. We did receive positive feedback about the bill splitting feature of the application. This can be seen in the response table as well, the majority of users thought that real-time bill splitting was the most valuable feature of the app. This opinion seemed to be more common among younger users who split bills more often when they go to restaurants with friends. We also received positive feedback about our decision to make the customer app a web app as opposed to a mobile app. When deciding how to build the customer application, we considered the adoption plan for our platform, and realized that most people do not want to download another app onto their phone. All of our surveyed users said that they would be much more likely to use the app since it was a web application.

Transitioning to the business side, we did have some difficulty getting in touch with restaurants to review the web application. We initially received positive correspondences from restaurants about setting up meetings to discuss the app. However, upon trying to schedule a meeting the correspondences fell through. In the end we send out a demo presentation to a number of local restaurants along with an evaluation survey to get their feedback. However, none of the restaurants responded to our demo in time. The questions asked in the restaurant evaluation can be found in Table 2 of the Appendix and the restaurant demo presentation can also be found in the Appendix. Although we did not receive concrete feedback from restaurants, we proceed to evaluate some of the aspects of our web app that we foresee being points of interest for potential restaurant clients.

We developed the app without having concrete information on restaurant needs that we could address. After not being able to set up meetings with local businesses, instead we tried to anticipate these needs. In regards to the waiter view, there are many aspects of the ordering process that are specific to the operation of the kitchen and the waiters in every restaurant, and we did not have the perspective on these considerations during development. As such, in improving the app we would need to work more closely with a restaurant to understand the daily intricacies of their operation. When considering the host view, the UI is the biggest issue, since there are other apps out there with more advanced and interactive UIs. However, the general functionality of the host view is consistent with other platforms and we do think it would be useful to integrate those features into our appli-

cation. The analytics view has the potential to be the most useful section of the application. However, we would need to work more closely with restaurants to capture and display the most relevant metrics. The fact that the data is being generated from one consistent pipeline is a very important feature we anticipate being the most valuable for restaurants.

Discussion

Key Findings

Waitr was started with the purpose of creating information value for restaurant owners while streamlining the restaurant experience for everyday consumers. The development was approached with the perspective that the greatest value the application created was for businesses through the analytics platform. Despite hypothesizing that wed provide consumers with the convenience afforded by a centralized service and a greater speed of service that only technology can provide, we still anticipated that the benefit created by this platform would primarily be for businesses.

However, throughout our development, evaluation, and presentation phases, feedback received suggested otherwise. It seemed that some of the side effects of our service seemed to be more valuable than some of the core offerings. Three key findings are important to highlight here:

First is the effect Waitr can have on server tips. By allowing diners to pay and tip through an application rather than paper receipt, Waitr can set a default amount or percentage for how much a waiter is tipped. According to a study by the Economic Policy Institute, tipped workers are overwhelmingly low-wage workers where there is significant wage inequality. The reason providing a default tip percentage works to mitigate this problem is because of the default heuristic as highlighted by Johnson & Goldstein in 2003; defaults act as implicit organizational recommendations and people are less likely to take the effort, physical or emotional, to deviate from the default. Our application still gives consumers the ability to change and customize this percentage, but has the side effect of potentially increasing the amount servers are given for their services.

Second, those evaluating our application seemed to find the bill-splitting functionality as important value created. The ease of splitting the bill as part of your payment process seemed to engender situations where diners are less likely to follow the convenient but less inclusive heuristic of splitting the bill evenly across all despite what each individual may have ordered.

Third, across all evaluations, we discovered that young users, specifically college students, found an application like this more valuable than older users. They are the ones who typically work as servers in underpaid areas and are also the ones who benefit the most from Waitrs bill-splitting features as they are more likely to be living on a budget than older individuals. The impact of these three findings is discussed in the Relevance of Findings and Future Work sections.

Related Work

Before addressing the implications of our work and findings, its important to highlight what similar work is already being done in this space. Companies mentioned throughout our paper, such as Ziosk and Yelp, have their own analytics platforms for restaurants and businesses. Startups like Upserve, Venga, PosIQ, and Fishbowl are working to tie together customer data from a variety of sources, including Point-of-Sale, email campaigns, credit card and purchase information, social media etc. to create customer profiles and give restaurants recommendations on how to maximize their profitability. Many of these companies address the business need without touching the customer-side experience.

The benefit behind such systems is that they avoid the challenge of customer adoption entirely. Restaurant adoption is much easier, because the application is directly linked to profit and monetary outcomes. The benefit for customers is less tangible, and these systems avoid that entirely. Their analytics platforms may well be more robust than what the Waitr platform could offer, but they beg the ethical question of where to draw the line regarding customer consent and privacy. Data found from credit card purchase information and social media is almost never explicitly consensual. Furthermore, they dont integrate information from the search and reservation steps of the customer restaurant experience a key step in the funnel used by customers to arrive at a restaurant and order. Additionally, these companies do not integrate with large consumer brand names that already contain a large amount of customer preference data, such as Yelp and OpenTable.

Relevance of Findings & Future Work

In this section, we discuss the relevance of our findings and the implications of those in future work in both corporate and research spaces.

In the Key Findings section, we discussed how consumers found some of Waitrs side effects as more valuable than some of its core offerings. In the Similar Work section, we introduced several companies working on creating similar analytics platforms for restaurants. Considering these two points, our findings suggest that it may be more valuable to integrate some of these side features, which proved more valuable to consumers, into existing platforms rather than create an entire new flow for customers and restaurants alike. This would clearly require a more in-depth study, but it is a takeaway for companies that already provide in this space.

Specifically, companies like Ziosk and Square can begin introducing default tipping in their purchase services. This can even be extended to receipts, where rather than having an empty blank for tipping amount, restaurants can offer three options (15%, 18%, and 20%) followed by a blank line. In this case, all consumers have to do is circle one of the three default options which is significantly less effort on their part than calculating tip themselves. Companies like Venmo should create a receipt processing system where they can scan receipts and make it easy for users to charge their friends based off of the prices on the receipt. Ziosk can add a feature where users can select which items they want to pay for on the tablet. Overall, all companies in the restaurant services and analytics space should recognize that young consumers found these features valuable and recognize that there is a need in this consumer segment left unmet.

From a research standpoint, our findings beg the question of whether needs such as these are a generational effect or an age-based effect. In other words, will future generations, when they reach the college and young adult ages, find an application like this beneficial? Why? If not, is it Generation Z specifically that finds value in an application such as Waitr? Answers to questions like these would allow for the maturing of an application like Waitr, better targeting, and better feature development in the future.

Limitations

Throughout Waitrs planning, development, and evaluation processes, we discovered limitations to the platforms development and viability. We highlight four limitations here, and end with a discussion regarding the last of the four.

First, before beginning development, we were unable to extract detailed information regarding existing operational and point-of-sale systems used by restaurants. This was primarily because of the sensitive nature of data on these systems as well as the fact that many restaurants were hesitant to take this time out for a student group. This limited our ability to develop a robust operations support branch of the application, and therefore shifted our focus to the customer application and analytics platform. Second, after building our application, we found it difficult to evaluate the restaurant-side platform for a few reasons. The operational efficiency and impact on speed-of-service was impossible to measure and evaluate quantitatively without a restaurant actually adopting our platform and integrating it into their systems. Third, our evaluation was primarily with Penn undergraduates. We received a handful of valuable feedback from adults over the age of 25, a lot of which informed the analysis conducted throughout much of this report. However, because of the ease-of-access to undergraduates, we were left mostly blind to how the rest of the world would perceive an application like this.

The three limitations discussed so far pertain to challenges we faced throughout the development process. The final limitation pertains to a challenge for Waitrs business model; in order for a service like Waitr to succeed, it relies on customer adoption. Getting customers to begin using a new service is notoriously difficult, with the average American adult downloading zero applications each month. Our team recognized this early on, and made design decisions to facilitate customer adoption easily. There are two solutions we considered regarding this problem. First, the customer-side application is a mobile web application accessible by a URL, not an iOS or Android application to download from an application store. This reduces barrier to entry. Second, the method to split payments only requires the user to copy the URL from their phone and share it via text or other mobile communication method to the individuals they want to split bills with. This means that if one individual in a group of friends is a Waitr user, they can quickly and easily spread our application to their friend group.

Ethical Considerations & Societal Impact

By streamlining the consumer restaurant experience into a singular app, we must address the aspect of human interaction. Our app does not completely eliminate human interaction and therefore does not eliminate jobs. Hosts are still needed to assign and sit people to a table. Kitchen staff are still needed to cook the food. Waiters are still needed to serve the food. Instead, our app simplifies the entire experience so as to maximize the consumers social interactions and experiences.

Due to the decreased dependency on waiters, however, there is a concern over tips. If the consumer interacts less with the waiter, will they tip less? Studies have shown that implementing an opting-out system rather than opting-in system increases participation. An opting-out system is where the choice has already been selected for the user and they must actively change or deselect the choice. An opting-in system is where the choice has not already been selected for the user and they must actively select a choice. A study done by Stanford on organ donations showed that countries with opt-out policies for organ donations, had 90% of the population became organ donors. Countries with opt-in policies for organ donations, had only 15% of the population became organ donors. We can apply this strategy to tipping. Future plans for our app would include a section for suggested tips at 18%, 20%, 25%, and other. The 18% would be automatically selected for the consumer, but they have the freedom to change it to whatever they want. With this strategy, we predict that tips would not decrease, and could even have the potential to increase. Furthermore, Waitr has the ability to track this so restaurants could monitor the situation.

With this product we wanted to address economic diversity within the restaurant industry. We wanted to make sure our product could be used by restaurants that do not have the budgets for a mass overhaul into a new system using iPads and Ziosk. By structuring our system so that assigning tables and serving food are the only actions in the restaurant experience workflow a restaurant needs to complete using the app, as well as allowing the product be accessible through any electronic device, we have eliminated the need for iPads and an expensive system such as Ziosk.

We also wanted to address economic diversity within the consumer pool. Our target consumers are college students, whose social lives rely heavily on going out to eat. Students with smaller budgets for food can become cut out of these important social interactions due to the cost of eating out. By making the split the bill system very flexible, students have more control over their money and have the ability to easily only pay for what they ate rather than defaulting to a generic split the bill method leaving people paying more or less for what they ate.

Business Analysis

The main inspiration for this product - and consequently its source of value - is in improving the restaurant experience for young people in social settings. However, in a product setting this app would be sold to restaurant owners as a SAS platform granting them access to full customer profiles collected on the mobile app. Thus, the business viability of this product will be analyzed in two ways: As a standalone mobile app for the everyday consumer and a software platform

to help restaurants optimize their business.

As was mentioned in the Motivation section above, the consumer issue being addressed is that when deciding to eat out, consumers must rely on a host of different apps to guide their experience. A simple search on the Apple App Store returns more than 200 apps simply for restaurant reservations. One issue with this is overload, where the number of different apps to consider is so large that face-to-face interaction becomes the simplest option. Another is that preferences are not transferred between apps, leading to redundancy and a less smooth customer experience. For the business end, the problem is that small businesses lack an affordable method to gain robust analytic insights about their customer base. Without partnerships with large companies like Ziosk, these small businesses rely on primitive data sources like customer surveys to understand their audience, while larger competitors have access to expansive customer tracking profiles through third-party sources.

The main value proposition is the payment section of the mobile app. Through personal experience and surveys of other college students, we found that one of the biggest pain points of the restaurant experience is splitting the bill. For college students who tend to eat out in large groups (of whom there are many, as we found through our polls), deciding on who should pay a large bill and how to properly pay each other the required amounts can be very time-consuming. The situation is compounded further by the sharing of dishes, varying prices per dish, and tips having to be parsed to be split among all diners. To address this use case, our app provides an intuitive payment system where each user can view the full order and select which items they would like to pay for. Cost splitting is done automatically if multiple users select the same item, and the tip amount can be added manually. We also allow users to share payment information so one individual is not required to ”pick up the tab” for the entire group. This streamlined payment system is made possible because of our online ordering platform; if college students are transferring money electronically, why should they have to parse paper receipts?

From the webapp perspective, the value proposition is the end-to-end customer data that small businesses could access to provide more insights into their customers. Instead of having disjointed customer experience information from multiple apps, restaurants would be able to easily track their consumers from initial purchase decision to final transaction. We would also be able to provide them with user segmentation information: What cuisine is most popular among your customers? What is their demographic breakdown? Which dishes are most popular for certain types of consumers? Assuming we gain a large enough consumer base (some strategies for which are discussed below), we would be able to provide small businesses with consumer behavior data previously reserved for large chains with proprietary ordering/payment platforms. It is important to note that the value offered by the restaurant webapp is entirely dependent on the richness of data in the mobile consumer app.

In terms of market research, the average American eats out approximately 4.9 times each week for an average cost of $36.40 per meal. In total, Americans spend on average $2,300 per year on eating out. This figure becomes even more pronounced for college students - after conducting a survey of Penn students, we found that more than 50% of respondents ate restaurant or takeout on a daily basis. Assuming approximately 24,000 undergraduates, these eating-out-daily individuals alone would account for approximately 1.5 million restaurant experiences per year at Penn alone. Of these restaurant experiences, we found that the biggest opportunity is in reservations. 63% of respondents said they used OpenTable for reservations, and 57.6% said they call to make reservations. The goal of a product like Waitr should be to convert those call-to-reserve users through the convenience of the app, then to focus on stealing market share from competitors by displaying the benefits of a centralized app.

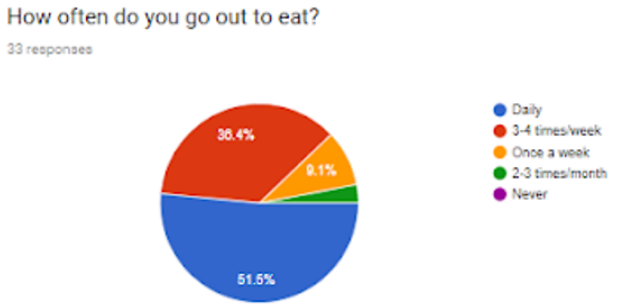

Figure 2: *Eating habits of surveyed Penn students.*

We can also break down the above market sizing by generation to better understand our target demographic. According to The Bureau of Labor Statistics, Gen Xers spend the most amount of money per year eating out at $4,200. However, at age 35-54 we would expect them to go out more with their families and may not be as interested in optimizing their dining experience. On the other hand, while younger millennials (age 25-34) spend less eating out at $3,400, this spending makes up a greater proportion of their overall spending than Gen Xers (6.8% vs. 5.8%). This means that millennials would care more about their restaurant experience, and since they are more likely to eat out with friends, they would be more in need of our payment splitting feature. Therefore, we hypothesize that *our target customers are college millennials who eat out approximately daily with their friends*.

While we were not able to survey restaurant owners for their preferences, our hypothesis is that small businesses are the target market for this app. There are a few reasons for this: (1) Large chains typically implement their own POS systems, and significant complications would arise from connecting our payment service to their existing platform. This would lead to greater cost to ourselves and the client. (2) Well-established restaurants do not feel the need for analytics as greatly as smaller companies. Data collection services use GPS tracking, cookies, and browsing history to track customers across multiple apps, and this information is sold to large businesses to give them greater insight into

their customers. This is exactly what we hope to offer in our SAS platform, but because our collection method is much simpler our lower costs would make it affordable for small businesses.

In terms of intellectual property, the only feature which is truly differentiated from other products is the payment splitting platform. Based on our research, the use of socket.io for bill management has not been implemented before. A possible extension could be using this platform for all payment services involving multiple people, such as paying rent or purchasing movie tickets. A likely extension we would make to this project is a proprietary machine learning model that segments users based on their purchasing habits on the app. By using data such as their culinary preference, demographic information, and group eating habits, we could implement a k-clustering algorithm, giving businesses trends among their customers. This model would be considered intellectual property.

With our target market established, we will now discuss the cost and revenue models associated with Waitr. The main costs associated with the mobile app would simply be upkeep, such as database storage fees and maintenance to ensure the app is working properly. However, with a large enough user base we would also expect significant costs to come from API calls such as Google's Geolocation API. Also, both the webapp and mobile app would have large costs associated with data storage, as logging information for more than a thousand users would require large Amazon AWS options. Our main revenue source is advertising. Like Yelp, we would allow businesses to pay for advertising on mobile in the form of paid promotions or "premium" spots in search results. And as we mentioned earlier, revenue would also come from the webapp through partnerships with restaurants. For onboarding, we would provide a "free" version which allowed restaurants to see incoming reservations and manage their tables. However, we would also allow restaurants to pay for a license that gives them full access to our restaurant management service (view tables, mark orders as completed, check in users) as well as a suite of analytics packages which can be used to gain insight into their business. Due to ethics considerations, all data would be anonymized, but restaurants would essentially be paying to learn information such as who their target customer is, what their peak hours are, best dishes, and a variety of other insights. The main challenge for this section is making the platform "worth it" to consumers - small restaurants do not have much cash to spend, so we would need to add insights on the SAS platform that they truly believe can improve their business.

With the above cost breakdowns, we can estimate prices for our products. Obviously, the mobile app must be free in order to compete with existing products such as Yelp and OpenTable. However, we can implement a restaurant promotion system similar to the Google SEO algorithm. Restaurants could request to be placed in a "Featured Restaurants" tab at the top of the page, and pay based on the exposure received through clicks on that page. For example, we could charge $0.10 per click or more per reservation resulting from that click. While variable costs vary greatly depending on

	Fixed Costs	Variable Costs	Revenue Sources
B2C Mobile App	• Server maintenance • Website hosting fees	• Calls to developer APIs, ex. Google Geolocation • User storage fees (Amazon AWS storage for user data)	Restaurant advertising fees
B2B Webapp	• Server maintenance	• Queries for restaurant data (hosted on restaurant database) • Analytics dashboard usage (Tensorflow.js)	Business licenses for data access

Figure 3: *Cost and revenue breakdown for both apps.*

the size of our user base, we estimate that this cost would offset the cost of API calls for the app.

It should be noted that one of the major difficulties one would expect to encounter from an app such as this is volume. Waitr depends heavily on having a large variety of users and restaurants on the app to provide the best overall experience. Thus, we would expect some difficulty climbing the adoption curve early on, but after a certain critical mass social contagion could carry the product further. Our strategy for marketing would be to attract consumers using promotions and coupons for using the app to drive consumer adoption, as well as giving users access to the bill splitting feature. With these consumers, restaurants would be incentivized to advertise their menu on the platform, even if they do not opt into paid promotion. This increase in restaurant volume would allow users access to the other features of the app - reservations and ordering. Thus, consumer adoption of the mobile app drives restaurant adoption of the webapp, and vice-versa. By providing the initial "spark" of consumer adoption via bill splitting, we could grow our user base large enough to transition into the other features of our app.

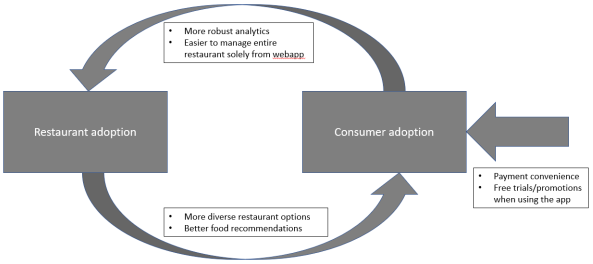

Figure 4: *Positive adoption loop cross-platform.*

Conclusion

At the crux of it's existence Waitr was created with the motivation of making the restaurant experience more seamless from start to finish. We saw a need to solve some typical pain points for college students, such as splitting the bill, and from there we went about identifying other inefficiencies in the dining experience. Throughout the course of developing Waitr we used an entrepreneurial process to motivate our decisions and fuel subsequent iterations and improvements to features. As we went about this process, we faced numerous obstacles in contacting restaurants, and this experience helped us see the importance of developing according to the needs of your target users.

From a business standpoint, Waitr has the potential to solve major pain points for millennial college students who eat out. Chief among these issues is bill splitting, which becomes easier when the entire restaurant experience is managed on one app. However, the biggest challenge for this app is climbing the adoption curve. We believe that by offering the bill-splitting feature and paid promotions first, we can gain the consumer adoption necessary to entice restaurants to take advantage of the data insights of the webapp. Using this strategy (and carefully managing costs), Waitr has the potential to dominate the online restaurant experience space.

References

Gould, E., Cooper, D. (2018, May 31). Seven facts about tipped workers and the tipped minimum wage. Retrieved April 29, 2019, from https://www.epi.org/blog/seven-facts-about-tipped-workers-and-the-tipped-minimum-wage/

Johnson, E. J., Goldstein, D. G. (2003). Do Defaults Save Lives? Science, 302, 1338-1339. doi:10.1017/cbo9780511618031.038

Miles, S. (2016, May 03). 6 Customer Analytics Solutions for Restaurants. Retrieved April 29, 2019, from https://streetfightmag.com/2016/05/03/6-customer-analytics-solutions-for-restaurants/

2018 Dining Trends Survey: Highest Tippers, Social Media Habits, and More. (2018, January 7). Retrieved April 29, 2019, from https://www.zagat.com/b/2018-dining-trends-survey-highest-tippers-social-media-habits-and-more

Kiersz, A. (2018, September 18). How much the average American millennial, Gen Xer, and baby boomer spends each year eating out. Retrieved April 29, 2019, from https://www.businessinsider.com/average-restaurant-takeout-spending-generation-united-states-2018-9

Appendix A: Evaluation Tables

Table 1: Customer App Feedback Results

Question	Avg. Response
How likely would you be to use an app like Waitr?	3.12
How likely would you be to use Waitr to find a restaurant?	2.88
How likely would you be to use Waitr to order at a restaurant?	2.60
How likely would you be to use Waitr to split the bill?	4.28
How easy to use was Waitr overall?	3.68

Table 2: Resaurant Web App Feedback Questions

Questions
How likely would you be to use an app like Waitr?
How likely would you be to use Waitr manage reservations?
How likely would you be to use Waitr to assign tables?
How likely would you be to use Waitr to manage orders?
How useful were the analytics shown in Waitr?
How easy to use was Waitr overall?

Appendix B: In-Depth Database Schema

Used as a counter lookup to generate unique values.

Counters
main
pat
pending
pkey
user

Used to store login info.

Users
user_id
phone
password
first
last
misc
date

Used to store access keys.

Users_Pat
pat_id
token
user_id
active
os
date

Temporarily stores user registration and SMS key.

Users_Pending
verify_id
private_key
sms_key
first
last
phone
password
date
attempts

General restaurant information.

Restaurants
restaurant_id
name
location
lat
lng
phone
category
payment_info
price
open_time
close_time
img_url
password

Items listed on each restaurant's menu.

Menu_Items
restaurant_id
menu_item_id
title
price
description
options
header_name

Tracks a party through their restaurant visit.

Party
party_id
restaurant_id
name
size
created
arrived
reservation
seated
table_id
paid
have_left

Tracks availability of tables for each restaurant.

Tables
table_id
restaurant_id
capacity
available

Stores all orders' information for each of the restaurants.

Orders
order_id
restaurant_id
party_id
menu_item_id
details
date