Dravid,Francinaldo,Jinuk,Richard

EECS 448

Prof. John Gibbons

11 May 2016

Deployment Plan for Rock Chalk Jaymovies

**SYNOPSIS**

We see Rock Chalk Jaymovies as having plenty of applications for the average consumer. In today’s fragmented consumer viewing market, television shows are pushed to all corners of the Internet-one television show may have an exclusivity deal with Netflix, while another may have an exclusivity deal with Vudu or Amazon. This means that the user, when looking for movies or television, will have to know which network the show is streaming on. However, this is not necessarily an easy task. For example, the popular AMC television show *The Walking Dead* has an exclusivity deal with Netflix, which means that all streaming content from that show will only show up on Netflix (usually a season late). However, *The Walking Dead* spinoff *Fear The Walking Dead* is not on Netflix, but on Hulu, and you can watch all current seasons of the show. This leads to a fragmented experience on most watching programs, which are network app-centric. However, on Rock Chalk Jaymovies, the user just needs to type in their TV show (or movie) and will instantly find their content. Once they’ve found it, it’s just a click away to find out where the content can be bought or streamed (for certain services), and then another click redirects the user to the streaming service itself.

**INITIAL REVAMP**

Since we feel the Rock Chalk Jaymovies experience is useful in most contexts, we would eventually like to ramp up to provide the app not only online, but also on mobile app platforms like iOS and Android, TV set top box platforms like Roku, Apple TV, and Chromecast, and pretty much any service that connects to a television. Prior to launch on these different platforms, however, we would most likely have to retool our backend. Most streaming and video content services are reluctant to provide public API access to their library. Netflix, for example, famously shut down its public developer API platform a few years ago. In order to circumvent the issue and get the product off the ground quickly with minimal fuss, Rock Chalk Jaymovies relies on Guidebox, a Web scraper API that provides movie and TV show data to third-party developers. The Guidebox API was a convenient way to add complex features such as search, trailers, and streaming information in a way that did not expend a lot of hardware resources on our end. However, this comes with significant limitations. First of all, Guidebox does not provide real-time access to streaming services, instead scraping data over a fixed time interval. This means that the data Guidebox provides for streaming services is not necessarily up-to-date. For example, when Amazon pulls all of its streaming content from Comedy Central, Guidebox may not be able to recognize that the pull has happened, leading to situations where users are frustrated because the app tells them a movie exists on a service when it doesn’t any more. Second, Guidebox provided us with a production key for their API with very specific guidelines, guidelines that going to market would likely violate. To avoid getting our access revoked, we would most likely have to pay a hefty licensing fee for Guidebox, in order to use tools and features that should be fairly easy to replicate in a standalone fashion. Third, Guidebox is locked out from certain services completely-all Netflix streaming data is inaccessible from our app, which is highly problematic as a large amount of streaming traffic comes from there. As a result, we would most likely have to build our own scraper backend in order to provide users with the same level of service they enjoy now.

We may have to retool the services that we provide access to in order to provide a consistent user experience, since we will have to implement scraping functionality (Netflix, for example, does not provide public developer APIs) that scrapes regularly and watches Netflix addition and removals regularly. As a result, we may not be able to provide the same breadth of services as Guidebox is able to produce from the outset. Ultimately, by the end of the year we would hope to have a working database of our own streaming service data that we could use before we relaunch Rock Chalk Jaymovies as a web app into the market.

**SERVER COSTS**

Currently, buying 96GB of cloud storage on a service like Linode (running a server farm in house would likely be very cost-prohibitive at this point) would run us about $1000 at this point. Seeing as the only data that is stored on our servers is text and low-quality images, that seems like a perfectly reasonable amount of storage for our service to start out with. Combined with the average developer salary in Kansas being around $70000, we’re probably looking at around $150000 at this point for the year in server and talent costs alone (2 developers plus server time). However, we could probably get labor costs down by providing equity, so at the lowest we could probably be looking at around $95000 for the first year of retooling.

We would most likely be making most of our money by running ads on the web app. Hopefully that would be enough to subsidize the mobile applications, but if not we could easily place ads there as well. We could find a way to package data in a way that we could sell it, thus competing with Guidebox or creating a new niche for movie and TV streaming information.

**MULTIPLATFORM GOALS**

Our primary concern after finishing the backend would be on platform universality. Our first focus would be on getting Rock Chalk Jaymovies on the App Store. Since most of the content on Rock Chalk Jaymovies has an iTunes base (although the service supports Google Play as well), it seems like a more natural fit. We would need to add another mobile developer at a cost of $70000 specifically for development of an iOS app, since Apple has sandboxing requirements for their mobile apps (for example, we would not be able to link to certain streaming services directly like we can in the web app), as well as other style guidelines that we would need to adhere to. We would also need to pay the $99 Apple developer fee in order to get into the App Store, although after that there would be no cost as far as hosting the app since it would likely be free to download (paid apps don’t tend to do well, and an app like Rock Chalk Jaymovies needs market share in order to grow).

The next natural step would depend on the profitability of the mobile application, or how talented our mobile developer would be. If we would have to eat the cost of another developer anyway, it may be worthwhile to branch out into the nascent world of TV apps. Since we would already be on iOS, moving to tvOS, the Apple TV platform, would be a good direction to go into, and would hopefully be a trivial addition to the product. Aside from development costs, the marginal cost of this project should be very near zero.

**CONCLUSION**

We like Rock Chalk Jaymovies as a product-it provides a valuable service in a way that isn’t really out there in a convenient way. However, it is apparent why a service like Rock Chalk Jaymovies doesn’t exist in the real world-it’s very expensive. Our back of the envelope calculation suggests that the retool alone would be at its cheapest slightly south of six figures. We’re also concerned about the legal and ramifications if we were to expand the product further-while scraping Netflix in the context that Guidebox is doing it seems fine, if the product ever gets to the point where we accidentally scrape sensitive user data (just as a hypothetical), the product could be in deep legal trouble. These are all factors to consider before we deploy the product to market.