CS8803 – Special Topics | Mobile Apps & Services

Ask – Project Problem Identification

by CAPA

- Alexis Durocher (adurocher3@gatech.edu) -
- Pulak Azad (pulakazad@gatech.edu) -
- Alex Faché (alexander.fache@gmail.com) -
- Carolyn Shi (cs947@cornell.edu) -
CS & MSCS student at Georgia Tech

In this assignment, we will describe our semester project – Ask. We will first start by describing the problem we are trying to solve and will talk about the feedback and research we have done. Then we will describe our use cases, project architecture and task breakdown and project plan. We will end with the resources we have at our disposal and plan to use along with any updates we have to our BMC.

1. Problem Overview | Ask

The problem that we are trying to tackle is based on two simple ideas:

- Why should I buy something that I would use only once (few times)? : as a requester.
- Why can't I set value to some of my inactive items? : as a provider.

On the first hand, having a quick access to any items, requesters don't have other choice than buying a brand-new product or second-hand one. In more than 50% of the case, the items that had to be bought for a peculiar case, won't be used anymore. This is even more real for mid-term travelers (exchange students for instance): On a new place, one will have to buy brand-new items and throw it away after one's stay.

On the other hand, providers are the ex-consumers that did the mistake: They own items they don't use anymore or they had bought for a peculiar use-case. Hence, they already solved a problem that new requester will face and solve by buying a new product. We can detect here a buying-throwing cycle which length depends of the requester type (travelers, locals, social class). The problem here is that providers are not aware of this constant cycle and thus cannot set value on their own product.

2. Feedback and Research | Ask

Requesters

- Wants quality goods for a reasonable price.
- Meet people in a community, ask local provider for tips (ie, renting a surfboard where are the best places to go)
- Get help while traveling, more information
- Have a quick turnaround time between requesting an item and receiving a response. (ensure big enough community)
- Package items. For instance I know nothing about camping, Ask would pull up camping kits and wish lists that you would need to have a good camping experience.
- Offline state so that I can still use the app (Would not have full functionality, cache most recent app state locally)
- Beautiful apps (pictures limited by Ask)

Providers

- Human nature to not put a lot of effort scrolling through items. This can be simplified with product filtering specific to what the provider has available (list of belongings for each provider)
- Find requests that can be fulfilled quickly (localization matcher)
- Insurance on my products
- Associated damage report
- Rating and reviews
- Potentially a mandatory refundable deposit in case something is damaged (Premium items)
- Not disturbed by inappropriate content, customer support
- Easy transaction
- Give help to requesters for a more enjoyable experience

3. Use Cases |Ask

A. Requesters

Travelers:

Angela and Max are traveling to San Diego two weekends from now and they want to go surfing while they're in the area. They don't have any surfing equipment and don't want to buy surfboards just to use for a day or two. They know there's a rental place an hour away but want to know if there's a local surfer they can talk to about the local waves with and maybe a restaurant recommendation or two while also receiving a cheaper price on some surfboards.

Angela and Max can use Ask to find a local surfer that's willing to lend them a board for the weekend and be able to find a good place to eat in the area.

Exchange Students:

A Georgia Tech student is going to study abroad at GTL and knows they will have a kitchenette in their dorm. They are staying there for 4 months but they do not want to bring pots, pans, kitchen cutlery with them or buy it only to use for a couple months.

They can use Ask to see if people have extra pots and pans in the area that they can borrow for a couple months or weeks where they plan to cook special local dishes and need special cooking items.

Hosting a Party:

Michael is hosting a birthday party but does not have enough tables and chairs for all of his guests. Instead of going to a party rental service which will take a hit on his budget he wants to find a simpler and cheaper alternative.

Ask will allow Michael to find someone near him that has some tables and chairs that he can rent for the day of party. This will save him money and make his party a success! ! !

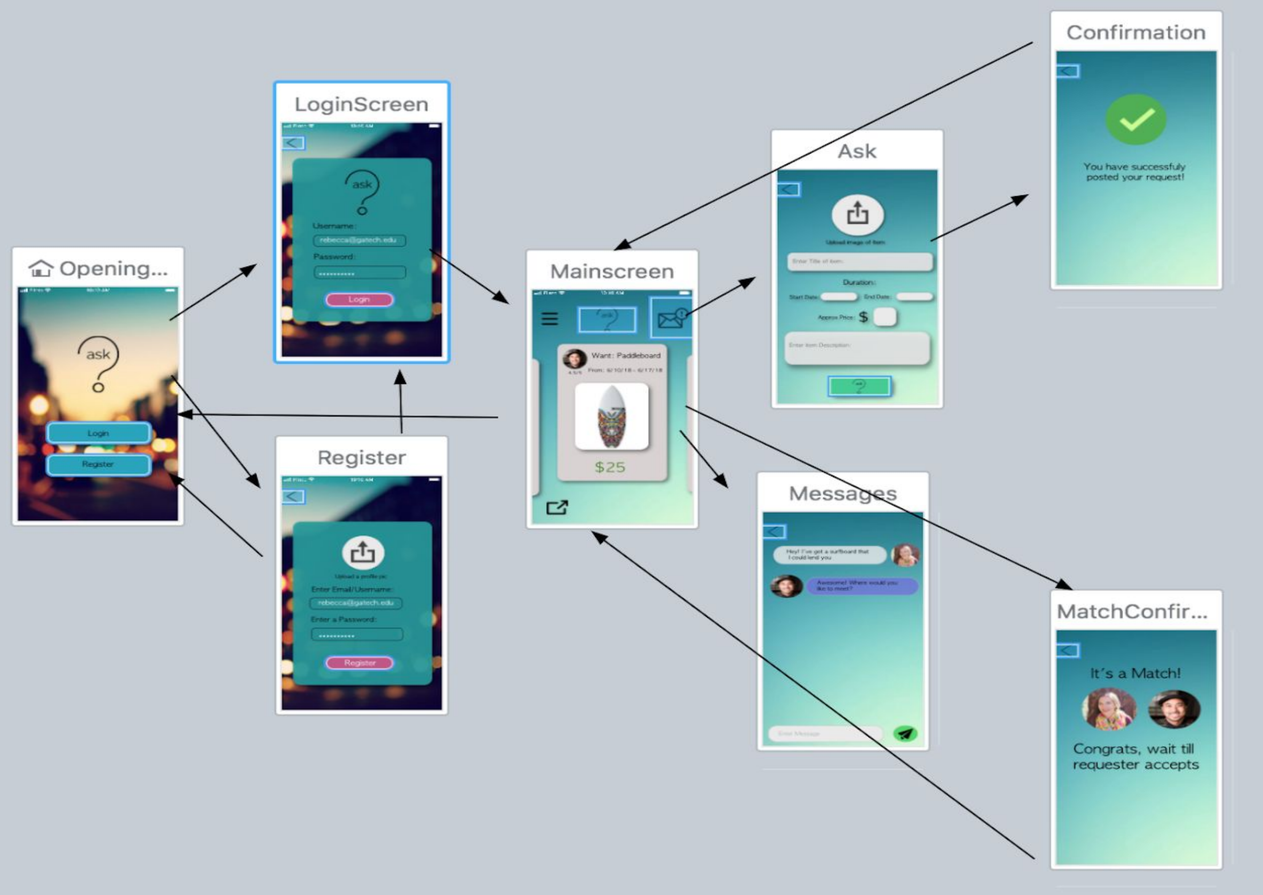
B. Provider Use Cases

Renting Out Unused Items:

Rebecca used to go surfing all the time before he hurt his knee but now he has two surfboards and swimming suit equipments laying around in his house that's just collecting dust.

Rebecca will use Ask to rent out his surf equipment to earn a little bit of money on the side to help with his physical therapy sessions or to simply enjoy.

4. StoryBoard | Ask



Link To interactive StoryBoard: https://drive.google.com/open?id=11GjKjoQNbZf5VQbkU2vVWJWFEzpSNgIw *(Must have flinto app on an iOS device to run it).*

Here we have a use case that we have implemented using Flinto. We have Max as the requester looking for a paddleboard on his trip to California. He is going to be at the beach for a week and and wants to find a cheap alternative to a rental service. By going on Ask, Max is able to submit a paddleboard request which is then received by a provider, in our use case Rebecca. Rebecca a local to California beaches has a paddle board that she is not currently using because she must take some time off for an injury. However, she wants to find a use for the paddleboard other than just collecting dust in her garage. Going on to Ask and confirming the request she found made by Max, they are matched! Once the transaction is confirmed by both parties, payment will be transferred and messaging will be sent up for communication.

5. Learning Prototype | Ask

A. Gathering user information

- It is crucial that we are able to gather precise user information in order to ensure that user experience is positive and persistent.

- We will need to gather the items being requested by requesters. This will include the item itself, date range, price range, and a description to aid in an efficient and timely match making process.
- The provider needs to let the application now what items they will make available to fulfill requests. The provider will be given more specialized items requests associated to the items that they are capable of providing.

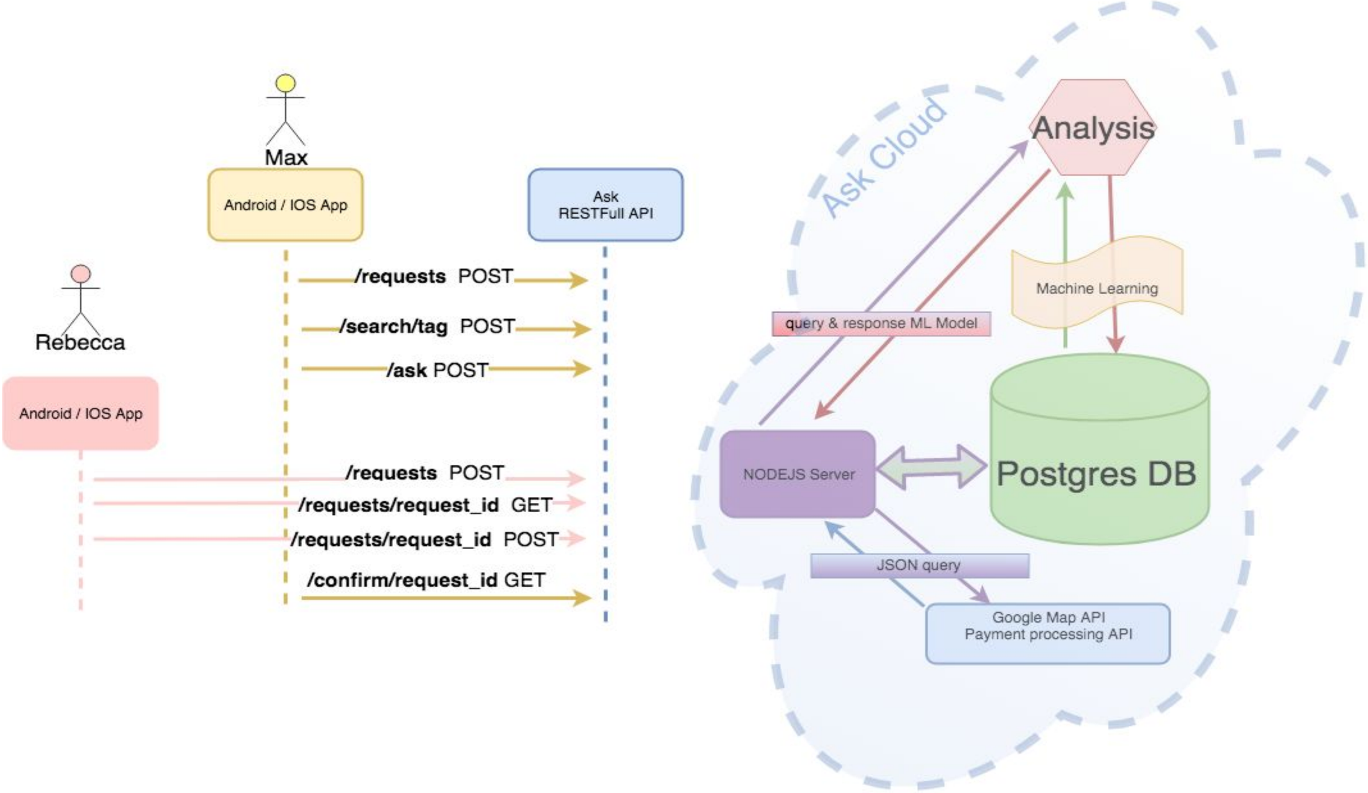
B. Determining what items are in most commonly in demand

- Our application will need to gather information regarding the categories of items and specific items that are requested. This information gathered through requests will let us know how often and when these item requests are made. With this information we will able to "suggest" to providers to add more rentable items so that the supply can keep up with the demand.
- How long with both requesters and providers be using Ask in one sitting.
- Our idea for Ask is to have a quick, simple, and cool way of renting out items and satisfying requests. It is necessary that the time between logging into the app and sending a request or logging into the app and fulfilling a request is minimal.

C. How long will items be rented

- Knowing the time range for items in requests will let us target providers according to their item availability. It will also allow us to set heads up notifications for potential wait times on common item requests.

6. Project Architecture & Workflow | Ask



Ask's workflow

A. Front-end

Requesters and Consumers use Android | IOS mobile application to request the Ask API. Tinder-like design. Chat embedded. Restrain to around 100 of items categories to ask or provide (start small and grow).

B. Back-end | Cloud

RESTful API. NodeJS server deployed on Heroku online server. The server is connected with our Ask Postgres DataBase.

Third party's API are accessible only via our Ask API. Only the Ask server should interact with third-party's API and build a JSON response on-top of it. The front-end should not have a direct access. (Easier to assure payment processing and restriction on the use of third-parties' services).

Perspective of evolution : Machine Learning model to classify users and predict their needs (filtering).

7. Resources & references | Ask

Development environment: Android Studio, XCode. RESTful API.

a. User visualization of activity

Determine where users are making requests from and for what locations. Where are providers located. Collect and Visualize Information.

b. Third parties' API

Location/Maps : Use to search for requests within a target area.

Image : Control image views.

Payment: Transaction between requester and provider.

Messaging: Communicated with requester/provider regarding details of product.

c. Machine Learning, Image Recognition

References :

How to build NodeJs & Postgres server + RESTful API (http://mherman.org/blog/2016/03/13/designing-a-restful-api-with-node-and-postgres/#.Wo2EURPOW9Z).

8. High-Level Team Task Breakdown and Project Plan

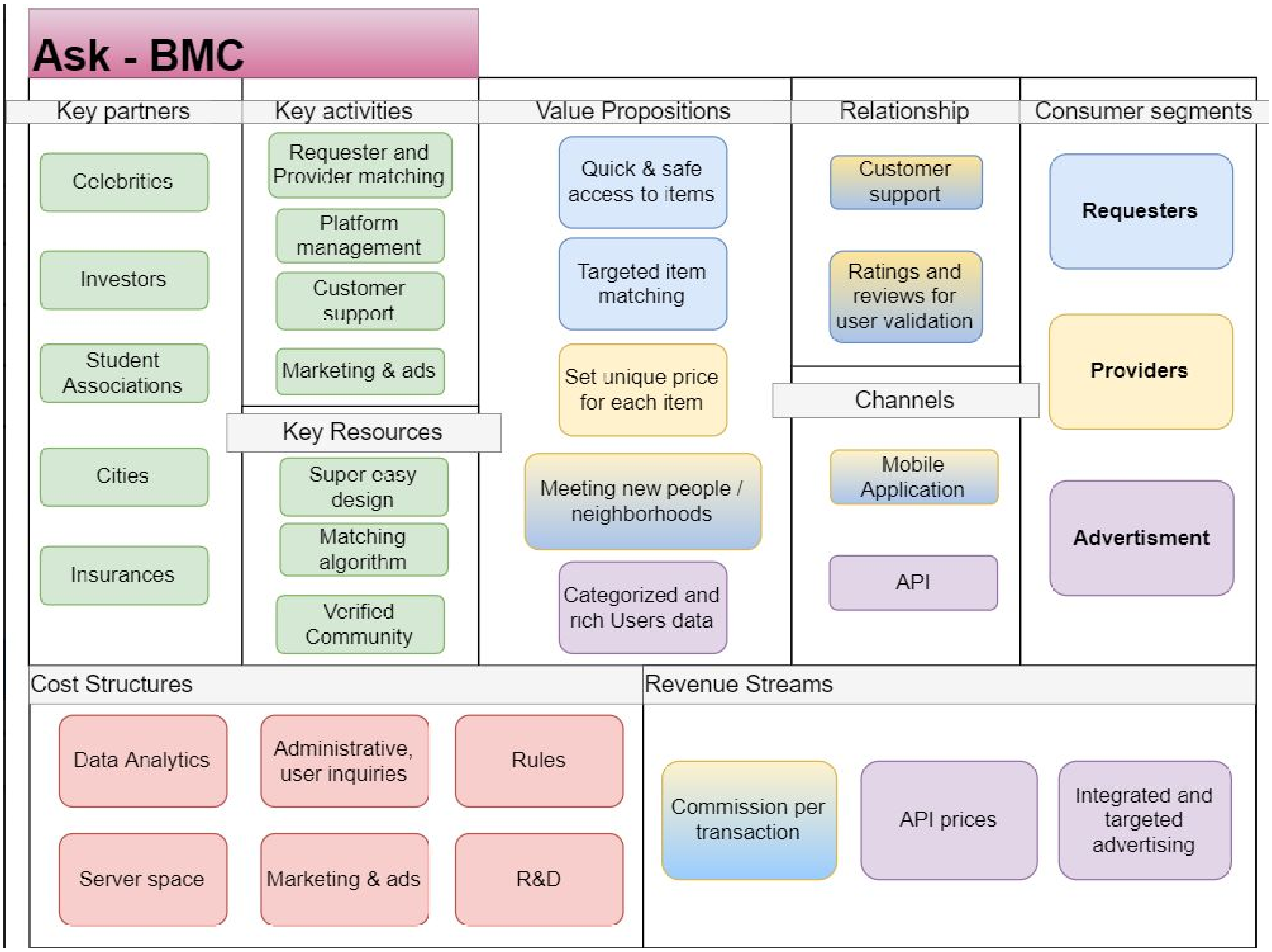
Final Deliverables:

- Website
- Video
- 3 minute presentation
- App

Feb 18	Feb 19	Feb 20	Feb 21	Feb 22	Feb 23	Feb 24
			Presentation 2 / PIVOT			
Feb 25	Feb 26	Feb 27	Feb 28	Mar 1	Mar 2	Mar 3
	More user interviews			Finalize back end		
Mar 4	Mar 5	Mar 6	Mar 7	Mar 8	Mar 9	Mar 10
Mar 11	Mar 12	Mar 13	Mar 14	Mar 15	Mar 16	Mar 17
	CODE	CODE	Presentation 3 / PIVOT	CODE	CODE	
Mar 18	Mar 19	Mar 20	Mar 21	Mar 22	Mar 23	Mar 24
	Start video filming CODE	CODE	CODE	CODE	End filming, begin editing	
Mar 25	Mar 26	Mar 27	Mar 28	Mar 29	Mar 30	Mar 31
	CODE	CODE	Presentation 4 / PIVOT	Start final presentation	Video rough draft	
Apr 1	Apr 2	Apr 3	Apr 4	Apr 5	Apr 6	Apr 7
	CODE	CODE	CODE	CODE	Finish final presentation	
Apr 8	Apr 9	Apr 10	Apr 11	Apr 12	Apr 13	Apr 14
	Brush up	Brush up	Brush up	CIC		

Apr 15	Apr 16	Apr 17	Apr 18	Apr 19	Apr 20	Apr 21
		Presentation 5				
Apr 22	Apr 23	Apr 24	Apr 25	Apr 26	Apr 27	Apr 28

9. Updated Business Model Canvas



Ask - BMC

Key partners

Celebrities

Investors

Student Associations

Cities

Insurances

Key activities

Requester and Provider matching

Platform management

Customer support

Marketing & ads

Key Resources

Super easy design

Matching algorithm

Verified Community

Value Propositions

Quick & safe access to items

Targeted item matching

Set unique price for each item

Meeting new people / neighborhoods

Categorized and rich Users data

Relationship

Customer support

Ratings and reviews for user validation

Channels

Mobile Application

API

Consumer segments

Requesters

Providers

Advertisment

Cost Structures

Data Analytics

Administrative, user inquiries

Rules

Server space

Marketing & ads

R&D

Revenue Streams

Commission per transaction

API prices

Integrated and targeted advertising

Modification

a. We have modified are user terminology as follows:

Users making item requests, "Requesters"

Users fulfilling item requests, "Providers"

b. Integrated and targeted advertising is a potential revenue stream