Project Proposal

COFF-E NLP & API
The Coffee Language Processor

Human-Computer Interaction
Bruinsma, Chi, Feliciano, Li, Paden

I. Meet the Team

Iris Chi
- Front End Coder, Product Manager
- Major/Minor: CS, Legal Studies
- Skills: Product Management, Editing, Figma
- Fun Fact: I throw a stick for fun (javelin)

Dustin Paden
- UI/UX Designer
- Majors: Digital Media Studies, Studio Arts, Psychology; UX Certificate
- Skills: UI/UX Research and Design, Figma
- Fun Fact: I'm a triplet

Jeffrey Li
- Front End Coder
- Major/Minor: Computer Science, Business
- Skills: UI/UX, Product Presentation
- Fun Fact: I enjoy photography

Jack Feliciano
- Back End Coder
- Major/minor: Computer Science, Statistics
- Skills: Data Mining and Machine Learning
- Fun Fact: Experiences Blindness

Chris Bruinsma
- Back End Coder
- Major/Minor: Computer Science
- Skills: Machine Learning, Scrum Master, Git
- Fun Fact: I was a Starbucks Coffee Master

II. Problem Statement

There is no denying that there is an all important human aspect to the process of ordering coffee. Whether it is the person who rings up your order at the register or the one who calls your name with the drink. That being said, there are areas where human interaction is not needed and a computer would be better suited to do the job.

Starbucks Coffee represents a second wave of coffee here in America. The first wave was provided by companies like Folgers giving the United States instant coffee and the like. Then Coffee was about convenience. Starbucks was the first to bring high quality Italian coffee to the United States and to present that style of ordering. What Starbucks also brought in were unseen amounts of customization to their drinks which is part of their fame. Indeed Starbucks boasts

87,000 permutations of their drinks. Another part of the Starbucks fame has been the inability for Starbucks Employees to properly write the names of customers on the cups. Herein lies the problem. Evidenced by our experience at the University of Rochester as both students ordering drinks as baristas making the drinks we see that there is an overall need to improve the ordering process.

We have divided the improvements into a few categories. The first is to make coffee more accessible. There is a language to order Starbucks drinks all on its own including important syntax and semantics. The phrase "Grande Hot Mocha" is a well formed drink but so is "Grande Brown-Sugar-Oat milk-Shaken Espresso." In addition to the details required to order a drink which can be confusing enough, there can also be language barriers introduced that complicate the process which can make it more difficult for the baristas to understand what is being said and more difficult for the person ordering to get what they want. By using an ordering system that makes use of such a system it also accelerates the pace of ordering by removing the overhead needed to train baristas on all of the many codes that need to be understood to make the drinks. A machine would be able to type : Latte with soy milk and two pumps of vanilla faster than a person could write: L + S + 2V in sharpie and ring the drink up on the register.

Our goal is to make this easier by providing a natural language processing system that will use existing ordering methods: such as a drive-through. This stands as a way to simplify the ordering process and enable employees to be allocated to other areas such as the more important making of the drinks themselves.

III.   Needfinding Techniques

Need-finding techniques begin with *mind-mapping* or *process-mapping*; as this is a productive way to get our group's thoughts in a cohesive, structured order. Mind-mapping has allowed us to understand we must do a task-analysis on the procedure of ordering coffee, observing user interactions. From there, we plan to interview fanatics (daily consumers), servers, baristas, and others to understand the user task analysis and difficulties. Using other techniques such as contextual inquiry, observations, and designed surveys, we will be able to find more information to better understand our demographic.

By observing Starbucks at busy times to observe why people do what they do, the interactions allow us to understand emotions, frustrations, and any inefficiencies within the ordering process. Understanding where the problem is within the users emphasizes bottleneck issues and significant issues so the researchers can fix it. By knowing the boundaries and understanding structured tasks to incorporate both ends to successfully begin problem solving. We can find boundaries and limits to what we can do to help by interviewing and following through with other need-finding techniques. This is done by implementing internal communication between user tasks and research tasks and open interpretation leads to more communication to be as specific as possible. We have specifically chosen to interview Starbucks employees and managers to understand a step-by-step of the ordering process, giving us more perspective and detail step-by-step. We aim to

gather information from the baristas, managers, and other staff members to understand the intermediary between the customer and the product (ie. the drink). Due to our team member's close relationships with the on-campus Starbucks staff, we have been able to interview individual baristas to gather information about the ordering process and any additional environmental factors. With, we have gathered observational data from watching the rush hours and understanding the ordering process interaction.

How did we do it? First, generally speaking, we asked respondents what component of the drink they ordered first, second, third, fourth, and etc. This was able to tell us the general sense of the University's population preference regarding the sequence. From there, we continued to ask whether these orders were successfully completed on-time and accurately. We found that 100% of our responses agreed that there was discouragement from ordering from Starbucks due to either long lines, late to class, or other frustrations within the ordering and receiving of a coffee.

As this is going on, we will have distributed surveys to better understand feedback for the ordering process targeted towards students, faculty, community. Our survey aims to gather general details for an audience, while understanding niche target problems by asking questions to simplify the ordering process entirely. We will also be asking specific questions such as the order of elements to aid in developing a user-friendly interface. These need-finding techniques will be outsourced publicly to the University of Rochester community, but mainly targeted towards those who spend time at Starbucks and who currently understand the frustrating order process. Additionally, we will be observing students and customers at Starbucks while watching staff interaction throughout rush hour and popular times. All surveys, interviews, and information gathered will be done so consensually

IV.   Prototyping

Each iteration of the prototype will be tested by 10 participants. To ensure each sample is generalizable to the larger Starbucks-consumer population, users will be selected based on diverse representations of genders, accents, ages, and order preferences.

The success of our interface and product will be evaluated for the duration of *how* smooth and accurate this process will be. The processor is expected to provide the dictated order of the\customer, in whatever form, to **a clear concise Starbucks order readable by the baristas** and based on the success rate of the processor and the efficiency of the ordering line we will know how successful our product is. Additionally, based on the way our users interact with our interface will provide us information on what we could improve upon our UX/UI designs. Realistically, because our product is dependent on speech recognition, there is not much for the user to physically do other than just speak. The users can evaluate the processor based on the UI and UX of the product, mainly focusing on smoothness and accuracy of flow of information and icons. The targeted subject users will be daily consumers, inconsistent consumers (Unaware of the ordering process, language), and staff members. This will give us the range of daily consumer to expert, such as barista and staff members while accounting for cases such as those who use medium versus grande. The range of participants will provide a diversified set of information based on the user reports on their evaluation of our product.

V.   Implementation
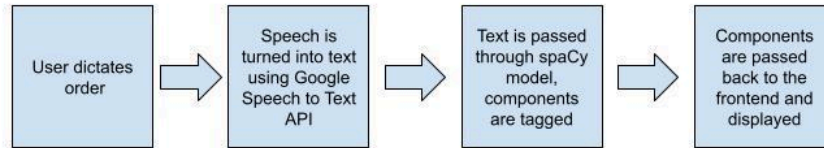
For this project, our aim is to translate real-world user-dictated Starbucks orders into a normalized form that can be passed to order placement services such as Grubhub. Below is a high level overview of our system. This section will explain our plans for implementation for each step.

As we intend for this application to be cross platform compatible, we plan on using a Flutter frontend and a HuggingFace Inference-API backend. This will ensure that our developer team does not need to code on two separate frameworks for future iOS and Android deployment. As we anticipate orders to be generally placed on mobile devices vs desktop devices, we will be making a mobile app and not a desktop application. As speech ordering could be less convenient on desktops than mobile devices, we do not anticipate this to be an issue.

The first step for a user to interact with our model is dictating their Starbucks coffee order. Flutter has a speech to text library, speech_to_text, which we plan on utilizing for this task. When speech is translated to text, we will send it to our backend where we will apply NLP techniques in a 2-stage process in order to translate a user-dictated order into a normalized form.

Stage 1 will be a Named Entity Recognition (NER) problem where principal component entities such as flavor, drink type, and size are extracted from user-dictated orders using the NLP library spaCy. NER in the context of our application can be visualized below:

I would like a large black coffee with two pumps of vanilla
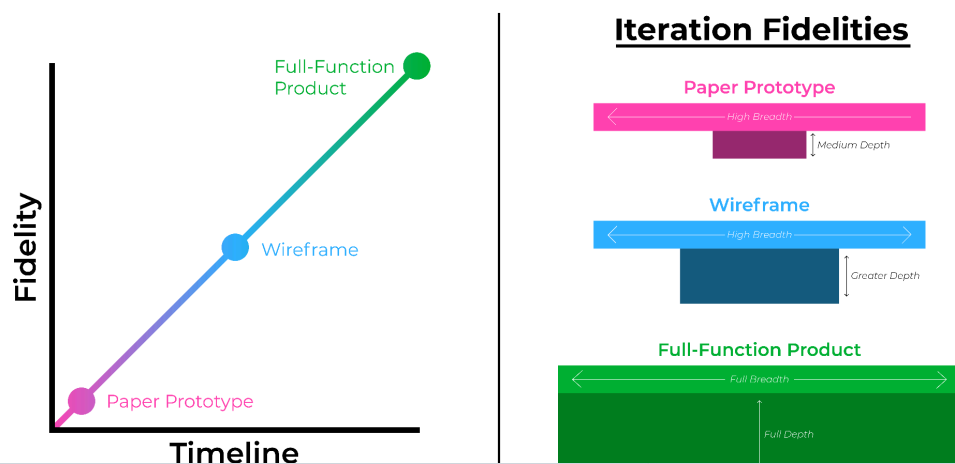
Where:
red: size
green: drink type
blue: flavor

In order to train a model that can accurately do this task, we will need to annotate an abundance of real world data, namely user-dictated orders. We plan on sourcing the orders from surveys (process described in Needfinding section). This could pose a problem, as the overall accuracy of our model is contingent on the quality and magnitude of our dataset, which can vary based on how many people fill

out our survey. Regardless of how much training data we receive, however, we hope to build a pipeline for a model that can be constantly improved and retrained as more training data comes in. Upon sourcing training data, we will manually annotate a certain amount of orders (depending on magnitude of training set) using the online tool labelstud.io. This tool allows for quick annotation of data for Named Entity Recognition model training by providing a GUI that allows users to use their computer pointer to highlight and tag individual entities. The software outputs the training set in the JSON format required by NER models for custom training. Assuming a large enough sample set and some typical lexical structure that user-dictated orders follow, this process will allow us to train an accurate NER model that recognizes entities in a user-dictated phrase corresponding to the components of a coffee cup that can be sent to the front end for user confirmation.

VI. Evaluation

Start Date	End Date	Deliverables	Assignee
10/23	10/29	Low fidelity wireframe and paper prototype	Dustin,Iris
		Finalize and release need-finding survey	Jack,Jeffrey,Chris
		Starbucks drinker user and barista research, conclusion on principal components of drinks	Chris, Jack, Iris
10/30	11/5	Finish front-end-display(no backend communication required)	Jeffery, Dustin
		Summary Report on Need finding	Iris
		Collect Training Data (real world order)	Chris, Jack
11/13	11/19	Train Model	Jack, Chris
		Create Website	Dustin, Iris, Jeffrey
11/20	11/26	Connect Model to front end	Jack, Chris
		Finish Poster	Dustin, Iris, Jeffrey
11/27	11/29	Finish Documentation and practice fro final presentation	Chris, Dustin, Iris, Jack, Jeffrey

Iteration	Type	Testing Environment	Evaluation
1	Paper Prototype - Low Fidelity - high breadth - Medium Depth	- Requires 3 team members (computer, facilitator, and observer) to meet with each tester. - Detailed notes measuring time and feedback will be documented	- Rate how experience compares to traditional ordering - Explain what users liked and disliked - What additional features would users like to see.
	Interactive Wireframe	- Two team members (audio computer/facilitator, observer) meet with	- Design aesthetics - Interaction reaction time while performing tasks

VII. Alternative Solutions

Instead of addressing the problem between the baristas and the customers we could try to make it easier to make online orders. There is a huge issue with Grubhub orders getting backed up and it leaves the baristas to have to shut down online orders, if there was a better solution to get Grubhub orders from the phone to the table it would also solve the problem of communication errors, ie. Instead of printing an entire order on a small sheet of paper, the order should be split into each section so it is easier to complete the order instead of having one person having to split up the order manually.

Coffee is often lost in translation, having a way to unify the meaning of coffee would completely simplify the process for baristas. There are many ways to interpret what a regular coffee is or any generalization of terms, if there was a way to translate it into an universal meaning it would simplify the process of making coffee for people. ie. If someone orders a 'regular coffee,' does that mean milk and sugar or just black coffee, if there was a way to identify where someone was from and then translate it to what the customer deems as a 'regular coffee' there would be no reason for the barista to need to interpret what the customer means and both parties would be happy.

To cut out the middle man, there should be a way to instantly get your order sent to the espresso machine and have it make the coffee itself. This relieves a lot of stress on the baristas to keep up with the demand, each coffee making machine that chains use have source code that can be altered and changed. If it could connect to Grubhub through a third party and then make the coffee itself as long as the barista puts the cup where it's supposed to be, this saves a little bit of time per order. Instead of having the barista push the buttons, it saves a couple seconds each time but a couple seconds each time adds up to a lot of time.