Abstract -

Ambrosia is a project that started as a crazy idea. We wanted to make a robot which could solve a problem that we have all faced during our day-to-day life. We wanted to make a robot that could cook food for you when you're running late on time. But that's not all- we realized the untapped potential for automation in this market, which is why we adapted our project to fit the needs of everyone from unattended children to pentathletes with super rigorous diets.

We wanted to make a project that allows you to eat healthy, and we wanted to make it incredibly convenient. In today's busy world, working professionals barely have any time to cook. This has unfortunately led to a rise in eating out, which, as a lot of us know, is not great for our body in the slightest. Obesity among children and adults is on the rise[1] because of such conditions, which will increase the percentage of people who die because of complications related to obesity[2].

As health freaks ourselves, we've made our project capable of automating a diet plan tailored around the user, making it possible for users to eat healthy. Ambrosia uses artificial intelligence to recommend you food items based on your diet goals, your allergen and medicine history and most importantly, your personal food preferences.

Ambrosia hopes to disrupt the industry by offering quick, hygienic, healthy and tasty machine-cooked meals at incredibly low prices. This will not only lead to a much healthier way of eating, but it will be much more rewarding as well, since we also hope to introduce our users to new dishes from around the world, exposing them to a much more diverse food palate.

Introduction –

Ambrosia isn't just a robot. It's a platform. It not only cooks a meal for you, but also allows you to track your calories, micronutrient (and macronutrient) indexes to an incredibly precise amount. It allows you to check your food eating habits, finds trends in your food preferences, and gives you a valuable insight into what you're eating in excess, and what you're eating in scarce amounts.

The project can provide invaluable data to doctors, who can prescribe certain diets to you based on the nutrient index data. It also automatically filters out any food which contains anything that could trigger allergies and allows you to identify problematic foods (if you have undiagnosed allergies, if you have given it some symptom data).

We hope to make the project capable of monitoring this data at an excessive scale in the very near future, using the USDA food database as a benchmark for most of our nutritional data.

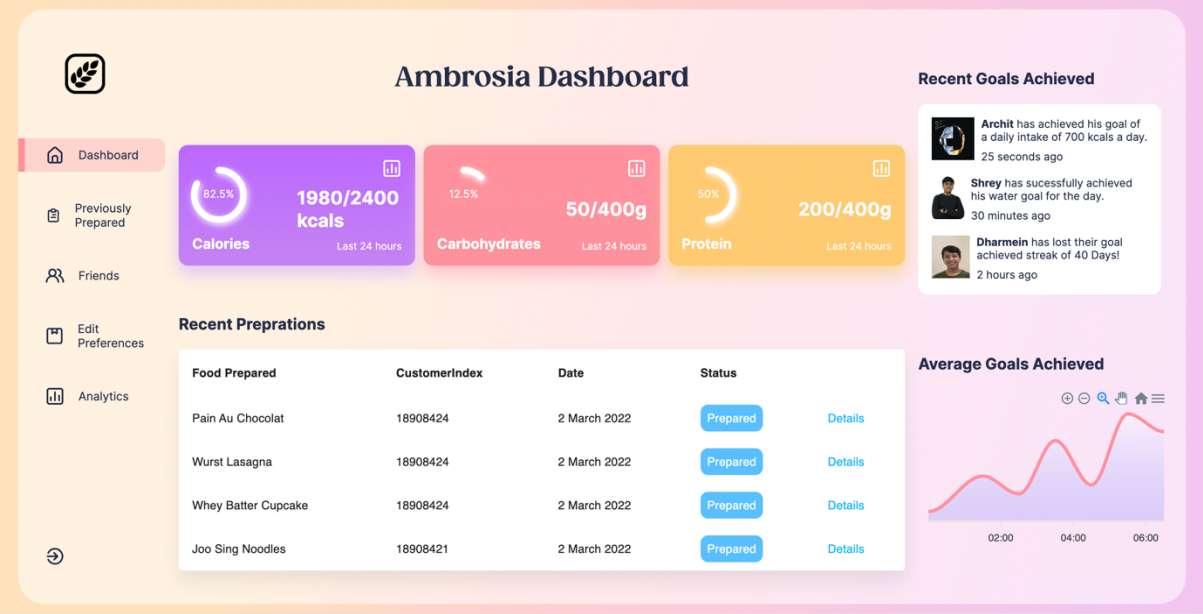
The project can be broken down into multiple parts.

A.) The physical project.
B.) Client dashboard

The physical project will be capable of making the selected dish. It can be easily extensible to meet the demands of any home or store. It may have a dedicated client side inbuilt into it that

allows you to select from the cornucopia of available foods. The physical project will be discussed in depth later in the paper.

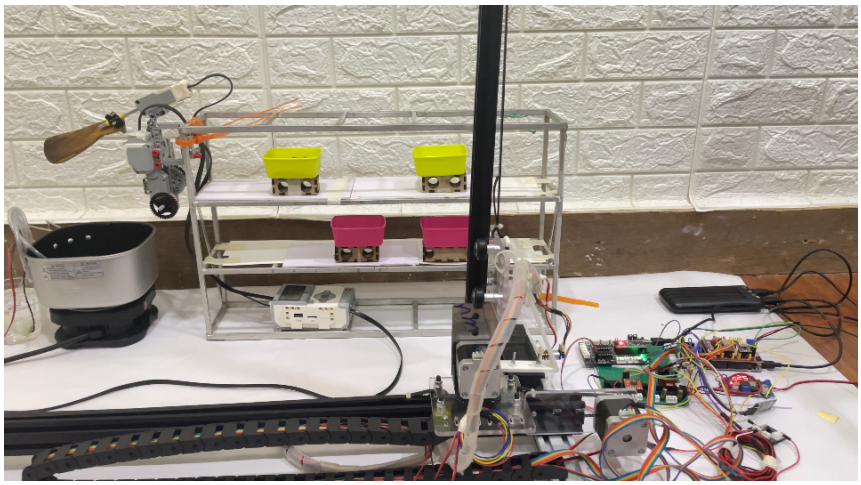
The client dashboard is arguably the most important part of ambrosia. It is the brains of the entire operation, and it houses all the algorithms that make ambrosia tick. It has a main dashboard allowing you to see all the nutritional data. It also integrates automatically with the social media of your liking to check on your friend's progress (if they choose to share it). It also allows you to enter all your medical data using the online forms integrated into the dashboard. The dashboard also shows you a comprehensive list of your previous order history, and it can also allow you to see data trends in your eating habits.



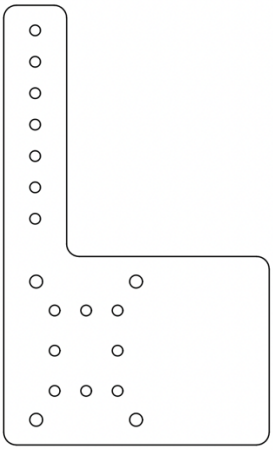
Current Ambrosia Dashboard (Built in React)

An in-depth overview of the project-

The design of the physical robot, which we have nicknamed Alfred, is heavily inspired by the design of a 3D printer. Our project fundamentally uses stepper motors, which are incredibly precise brushless DC motors that allow us incredible control over the positioning of our automata.

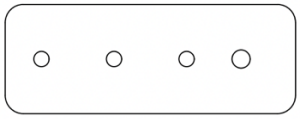

Project Ambrosia Prototype I

The main part of the project- the X-Z axis mounted automata is made up of a 20x40 V-profile aluminum bar (on the x axis) and a 20x20 aluminum profile on the Z axis. The Z axis bar is mounted on top of a movable “car” that is comprised of a custom polycarbonate cutout (made entirely in Fusion 360, explained later in the paper) which itself sits atop four ball bearings, which slide or move on the horizontal section of the X aluminum profile using the smooth conveyer belt, which is attached to the car using a zip tie on both sides.



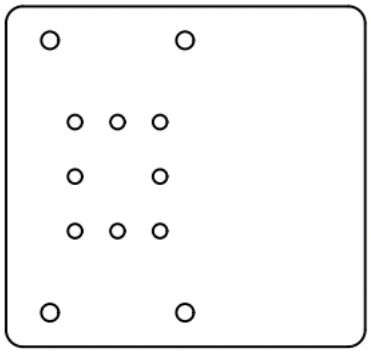
Above - Vertical Gantry Design in Fusion 360

The conveyer belt, and the cars by extension are moved by the X stepper motor (NEMA17 Stepper motor, to be precise), which is connected to a 36-tooth gear using another custom polycarbonate part, which produces a pushing-pulling motion system using the ball bearings and motor. The X motor spins the toothed gear, producing motion in the conveyer belt, causing the ball bearings on the car to move, and thus allowing motion in the Z mounted system.



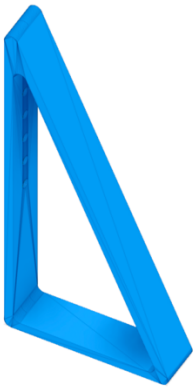
Above – End Bracket CAD in Fusion 360

The vertical system too moves in a similar way. The Z system, which produces the movement in the vertical direction is mounted on the horizontal (X) car, as described earlier. The X car has a vertically mounted 20x20 aluminum profile bar, with the Z stepper motor sitting snugly on bottom head of the profile bar. Like the motion in X dimension, the motion in the Z dimension is based on the motion produced by the Z stepper.



Above – Horizontal Gantry CAD in Fusion 360

The purpose behind the Z axis system is to allow the Z car system to the freely along the Z axis. The purpose behind the Z axis system is to allow the Z car system to the freely along the Z axis. Attached to the car is a custom designed Polycarbonate chassis which itself is connected to the custom 3D-printed clamp, which houses two very, very important components that make up the heart of our project.



Above – 3D printed clamp rendered in fusion 360

These two components are the Metal-Gear MG90 servo motor and the 75:1 Micro-metal gear motor. These two components, along with the car that they're housed on make up the core of our vertical movement in the Z direction as they make up the actual "pickup-and-drop" part of the robot, allowing us to collect the custom-built crates from our container with incredible accuracy.

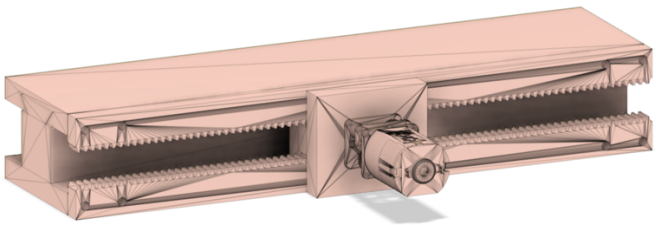
MG90 Servo Motor

The first motor, the MG90 servo motor allows us to pick up the crates and rotate it once the vertical bar is right in position, causing the crate to drop all the food components directly into

the induction cooker right underneath. The servo motor of choice was originally one with plastic gears (SG90), but we switched to the MG90 with metal hears because the torque was way too little for the weight, we wanted the robot to transport, which is why we decided to go for a much, much powerful motor. (The SG90 failed to transport heavier food components while the MG90 did them with ease.) The servo motor is attached to two Lego axels, which allows it to “grip” the holes at the bottom of the food containing crate, itself stored on the shelf of the container that it is kept in. Once the robot grabs it, the hand moves back, out of the way, allowing the X stepper to transport the Z bar to the other end of the X bar, positioning the servo (and the suspended food container), over the induction cooker. Once this occurs, the servo tilts from its original inclination of 20 degrees to 110 degrees, causing the food contained within the food container to drop directly below into the induction cooker.

Earlier, we described the clamp moving out of the way once the servo motor picked up the container, but this motion was not elaborated upon. The motion is produced using a linear actuator which is built into the clamp, positioned right behind the servo motor. The motion itself is a rack and pinion mechanism, driven using a N20 polulu motor. But that’s not all the polulu motor does. The motor itself functions as the hall effect sensor for the entire project, allowing us to send feedback and micro-adjust the steps (or micro-steps) of the X and Z stepper motors.

N20 Polulu Motor



Prototype of early Rack and Pinion Mechanism rendered in Fusion 360

Essentially, Stepper motors cannot tell the relative direction that they’re heading in, and simply assume that wherever they’re going is the right direction. This is easily fixed by using the polulu motor as a hall-effect-sensor, which without going into too much depth, allows us to calculate the precise relative position of the Z bar, allowing us to code in “micro-adjustments” in the stepping of the X motor, making the project semi-adaptable to changes.

Once the X stepper reaches the end of its path, it lightly interacts with a sensitive Lego touch sensor. This touch sensor receives the input and sends a message to the EV3 brick, which interacts with the medium and large motors situated on top of the container, which causes the

mixing mechanism to lower. The large motor is connected to a large Lego axel motor which is connected to a wooden spatula using two axel pins which are inserted into drilled holes in the spatula. We're using a water pump to pump up water from the glass into the pan. The water pump triggers when the induction cooker is switched on, allowing it to fill the water before the ready-mix is dumped into the cooker.

Ambrosia is a complex project, requiring many intricate components that all need to coordinate together effectively. To run these components effectively, we need to use a very efficient power system. We draw our main power directly from the 240V source, which we step down to 12 volts using the SMPS. We further step it down to about 5V (with more current) using a buck converter (with a handy display, allowing us to see exactly how much we've stepped it down).

To control our project, we're using an Arduino nano running off a custom PCB to control the servo. We're using an Arduino Mega connected to a CNC shield with DRV 8825 Stepper motor drivers to control the stepper motors. We're tentatively using an Ardutooth application to send inputs to the stepper drivers via the Arduino Mega. We've used some libraries including a stepper-regulation library called Accel-Stepper in our code.

Frequently asked questions

1.) Who's in your team ? Who manages what tasks ?

    Ambrosia is made up of three members namely, Archit Shanbhag, Dharmein Shah, Shrey Shah. Archit is the head of programming and presentation, Dharmein is the head of non-Lego hardware development. Shrey's the head of Lego-based hardware development and the CAD associated with the project. Each member is vital to this project, and everyone is equally as important.

Team ambrosia- Left to right; Shrey, Archit, Dharmein

2.) Where are we from ? How did we come up with this idea?

We're from Mumbai- the city of foods. You could say that we were a little inspired by our surroundings :p . Our parents are working parents, which is why we're alone at home, causing us to learn to cook for ourselves. This led us to eat outside a lot, which didn't really work diet-wise as it triggered allergens or led us to gain a lot of weight when we shouldn't have. This caused us to think about ways to solve this problem- and thus, an idea was born.

3.) What problem are you solving ?

Meal preparation is incredibly important, and it takes time and energy to cook food for yourself, and even if you do cook food, it's not going to be very healthy, unless you follow along a recipe or a cookbook. This process is even harder when we have no time- which is exactly the case in our busy, busy world. This has led to the immense growth of food delivery businesses such as Uber eats. While tasty and convenient, it's very hard to stick to your diets when you're selecting foods. Therefore, we created ambrosia- to make cheap, tasty and healthy foods, fast- and we wanted to make it easy. Easy enough that unattended children could become a little more independent.

4.) What other ideas did you investigate?
Originally, we tried making designs for rescue robots. We tried making small insect-like robots that could provide reconnaissance to rescuers, but that proved to be too difficult to implement, leading us to shelf it. We then decided to make an automated doctoral checkup system, which also fizzled out because of the sheer number of vectors we needed to track in the project- it would prove to be very, very difficult. Therefore, we thought, and thought, ideated to the end. Once we got comfortable with a few ideas, we selected a niche, and we stuck to it.

5.) Did you face any challenges during the development process?

No project is complete without failures- and we should know, we had a lot of them. The first version of our prototype was incredibly lacking in features compared to our robot now. We've had to calibrate our stepper motors using code with nothing but sheer trial and error, we've faced problems with our servos (causing us to simply switch to one with greater torque), we've even had instances where our Z stepper bar collided with our mixing mechanism because of timing based bugs in our software. It was not an easy journey till here, but it was one that was worth all the pain in the end.

6.) Who will the project target ? Who'll benefit from it ?

As stated before, our project wants to target everyone from unattended children to athletes with super-rigorous diets. Our design is appealing and easy enough for children to use it, yet it's just as information oriented as it needs be, allowing athletes to track their diet data in incredible depth, providing insights into calorie and macronutrient trends.

7.) How will the project generate revenue ?

Ambrosia, after all is a product- it's meant to be purchased and kept at home. So yes, we're going to sell our physical product, but that's only a very, very small part of our

revenue streams. We would also like to partner with gas stations, clubs, schools, etc. to install our machines in their premises, modifying the machine to work like a multipurpose vending machine, allowing the users to simply scan an QR code and be served one of their favorite meals hot on the spot. Once we're big enough, we'd also like to partner with the ready-mix companies to make our own partnered ready-mix components, allowing us to essential acquire our customer base using the low-cost model, and slowly leverage the acquired customer base to generate more revenue. This revenue would allow us to partner with prominent chefs, the best in the world, to curate gourmet foods, thus expanding our already sizable food catalogue. This, in term will allow us to generate more customers, which leads to essentially explosive customer growth.

8.) What did you use to build the dashboard ?

While the dashboard is far from being finished, with most of the backend remaining, the current version is built mostly with react, utilizing MongoDB, NodeJS and a package manager called Yarn, which allows our dashboard to not only function smoothly, but also look incredibly modern and clean.

9.) What if I want to have a cheat day ? Does Ambrosia account for that ?

We get it. Eating healthy doesn't always mean cutting down on guilty pleasures. Ambrosia does indeed recommend monthly cheat days where you could get something beyond the regular diet scope. (In fact, we do plan to introduce a separate algorithm for tracking popular cheat-day foods !).

Ambrosia's process flow -

Defining the problem - In today's busy world, barely anyone has the time to cook anything nutritious and healthy, leading to the massive rise of modern delivering apps such as Swiggy, Zomato and Uber eats. This is not healthy in the slightest and has led to mass scale obesity and a myriad of health issues.

Research and Brainstorming – Originally, our project was just supposed to be a cooking robot which could cook healthy meals fast. Soon enough, we realized the untapped potential in this market after conducting our own market research, we were amazed at the fact that no one had managed to integrate diet-oriented and preference-based algorithms into a cooking robot before. We decided to go even further, by adding allergen detection, allergen avoidance and health-based exceptions. We've even planned to add facial detection, voice assistant support, integrations with your favorite tracking apps (NRC, health, etc.)

Prototype building - Ambrosia began life as a simple cooking robot we codenamed Alfred. We've iterated every single part on that robot for the better. The robot you see in front of you now is a result of improvement and tweaks made on the skeletons of multitudes of failures. Every detail has been refined- everything from the most optimal hollow shape to be printed onto the container (our prototypes succeeded the most times with round holes) to every "step" our stepper motor takes- everything has been tested, everything has been refined.

Building Ambrosia – A huge part of Ambrosia was trying to keep the project *as cheap as possible without compromising on performance*. Lower prices lead to a larger reach and a larger customer base. Our project is also scalable, meaning that depending on the size and features, it can cost practically anything upwards of 30,000 INR and beyond. It takes a multitude of parts and processes to make ambrosia work, the most important one being the stepper motor.

Deployment – Ambrosia has been tested time and time again. Ambrosia is safe, fast and hygienic. Our project has an astounding accuracy rate (*for the prototype, the final form will be significantly more refined.*) of ~80%.

Detailed Business Model

Food Trucks – Most online food today is not produced in actual kitchens. In fact, it is prepared in special food trucks known as *ghost kitchens*. These ghost kitchens are usually very, very unhygienic, as a result of which, food quality and brand identity is usually compromised. Ambrosia wants to replace such ghost kitchens by providing its own mobile kitchen units, allowing us to push as many meal orders as possible while providing nutritional integrity, and making sure that they're eating healthy.

Vending Machines – Ambrosia wants to provide passive incomes to gas stations in a similar way as ATMs, allowing us to install a stationary unit at a store, university cafeteria, etc.

These public institutions would get a royalty for every sale that they make at the location, allowing them to generate passive income in this incredibly lucrative space.

Homes – Ambrosia is, at its core *home automation*, and the core business model is meant to be marketed towards homes, despite most of our projected revenues coming from other sources. Ambrosia will be marketed and promoted as a home item, because it's perfect and affordable for busy middle- and upper-class families. Once refined, we also hope to monopolize home-based revenue by partnering with ready-mix companies to sell our own container refills, giving us a theoretical monopoly in this space (once we've acquired enough customers). We also want to introduce subscriptions (totally optional- no subscriptions will be forced on the user, ever) that give you access to a much wider range of food, curated specifically by professional chefs and giving you access to many, many more intricate forms of data-tracking.

Motels, Hotels – Hotel chefs can be messy and unorganized. As the hospitality industry grows, so does the need for quick, organized and tasty food. Ambrosia hopes to expand into this space as well, once it has scaled up enough. We hope to provide quicker ways of providing meals to them on a mass scale. We hope to become an early part of the quickly automating hospitality industry by providing delicious meal services to our clients or partners.

Subscriptions – Subscriptions form the backbone of every internet business these days. Ambrosia is no different – we hope to bring subscriptions to our product by partnering with ready-mix companies to provide custom made ready mixes to customers who've bought our monthly subscriptions. They can also buy a "premium" membership, giving them access to curated recipes prepared especially for their tastes by our in-house chefs. That's not all- we also hope to use AI to rapidly generate new, never before seen food items tailored to our user's tastes. These subscriptions will also include new, powerful features beyond the scope of our regular feature set such as micronutrient dosage tracking, health supplement tracking, workout integrations with other partnered apps, etc. *Ambrosia will never force subscriptions on the user – the subscriptions are meant to add to the feature-set, not replace it*.

Future Implementations :-

Ambrosia hopes to introduce the following features in the future –

1.) Facial Recognition for automatic account sign in.
2.) Voice Recognition for automatic account sign in.
3.) Voice Assistant Integration with Alexa, Google Home, and Apple HomePod.
4.) Workout Tracking Integration with Apple health, Nike Run Club, etc.
5.) Fully functioning dashboard with highly responsive breakdowns of food(s).
6.) Fully functioning android (and IOS) apps with profiles and food selections.
7.) Blazing fast API allowing for rapid integration with supported apps.
8.) Utilizing AI to predict food preferences, to create new meals and to customize diet plans using other data.

Bibliography :-

Citations :-

[1] - Venkatrao M, Nagarathna R, Majumdar V, Patil SS, Rathi S, Nagendra H. Prevalence of Obesity in India and Its Neurological Implications: A Multifactor Analysis of a Nationwide Cross-Sectional Study. Ann Neurosci. 2020 Jul;27(3-4):153-161. doi: 10.1177/0972753120987465. Epub 2021 May 29. PMID: 34556954; PMCID: PMC8455012.

[1] - Malik, V.S., Willet, W.C. & Hu, F.B. Nearly a decade on — trends, risk factors and policy implications in global obesity. *Nat Rev Endocrinol* **16**, 615–616 (2020). https://doi.org/10.1038/s41574-020-00411-y

[2] - Abdelaal M, le Roux CW, Docherty NG. Morbidity and mortality associated with obesity. Ann Transl Med. 2017 Apr;5(7):161. doi: 10.21037/atm.2017.03.107. PMID: 28480197; PMCID: PMC5401682.

Important links that we referred -

Food Database source - https://fdc.nal.usda.gov/fdc-app.html

MG90 Specification Sheet - https://components101.com/motors/mg90s-metal-gear-servo-motor

Polulu Motor Specification Sheet - https://www.pololu.com/product/997

Stepper motor Specification Sheet - https://www.omc-stepperonline.com/e-series-nema-17-bipolar-0-9deg-30ncm-42-48oz-in-1-5a-42x42x38mm-4-wires-17me15-1504s

Image sources –

MG90 - https://components101.com/motors/mg90s-metal-gear-servo-motor

N20 Polulu Motor - https://www.electronicscomp.com/n20-3v-15-rpm-micro-metal-gear-motor?gclid=CjwKCAjw-L-ZBhB4EiwA76YzOVVu5awmL5JEoQurA-B7S3iB2kEWNoIAhlAEYZ9Bn28UTX-XHA84rxoCcC0QAvD_BwE